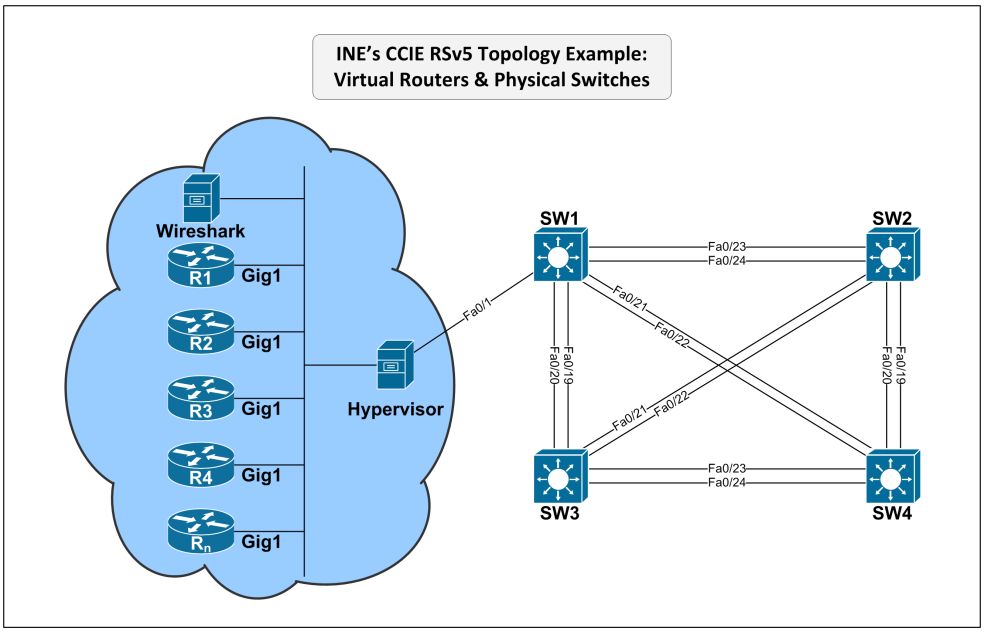
CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Workbook Overview

CCIE R&S v5 Topology Diagrams & Initial Configurations

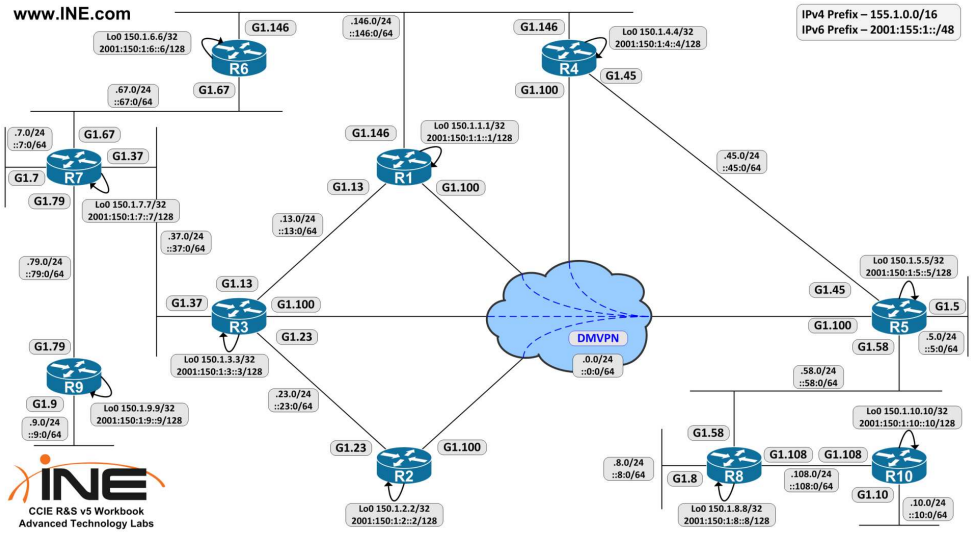
Click the **Resources** button on the right to download the initial configurations and PDF diagrams for the Advanced Technology Labs. PDF diagrams are optimized for Legal print size (8.5in x 14in / 215.9mm × 355.6mm). Diagrams below are optimized for full-screen viewing at 1920 x 1080 (1080p).

Topology Wiring: Virtual Routers & Physical Switches Diagram

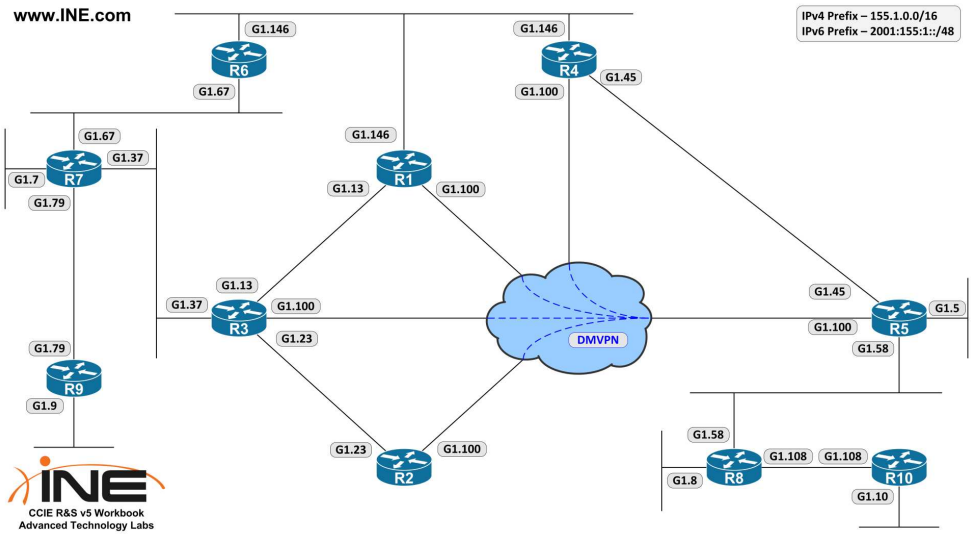


Advanced Technology Labs With Addressing Diagram

https://t.me/learningnets

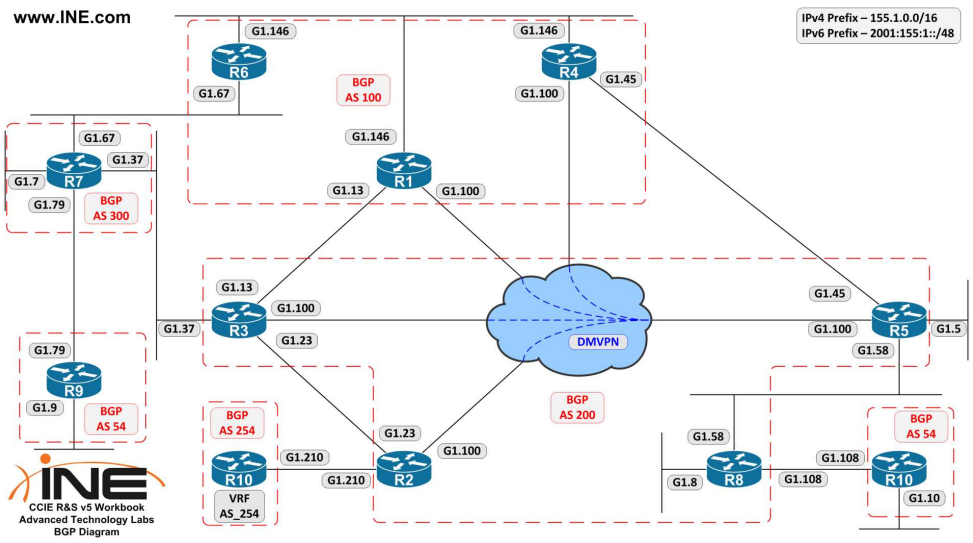


Advanced Technology Labs Without Addressing Diagram

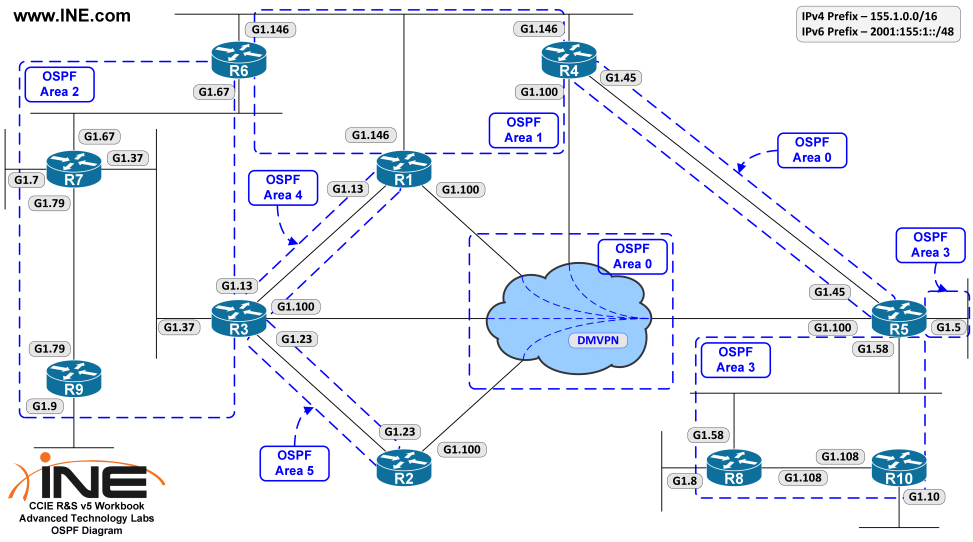


Advanced Technology Labs BGP Diagram

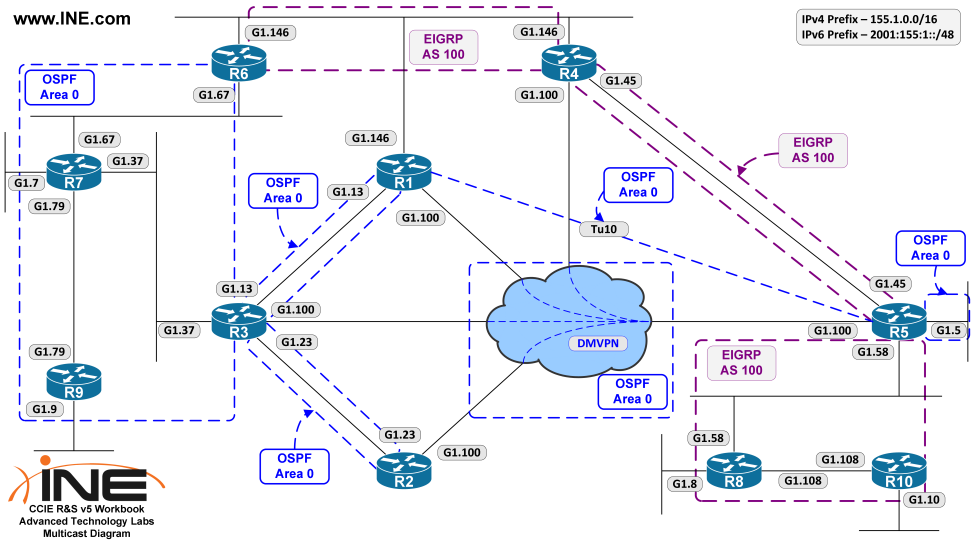
https://t.me/learningnets



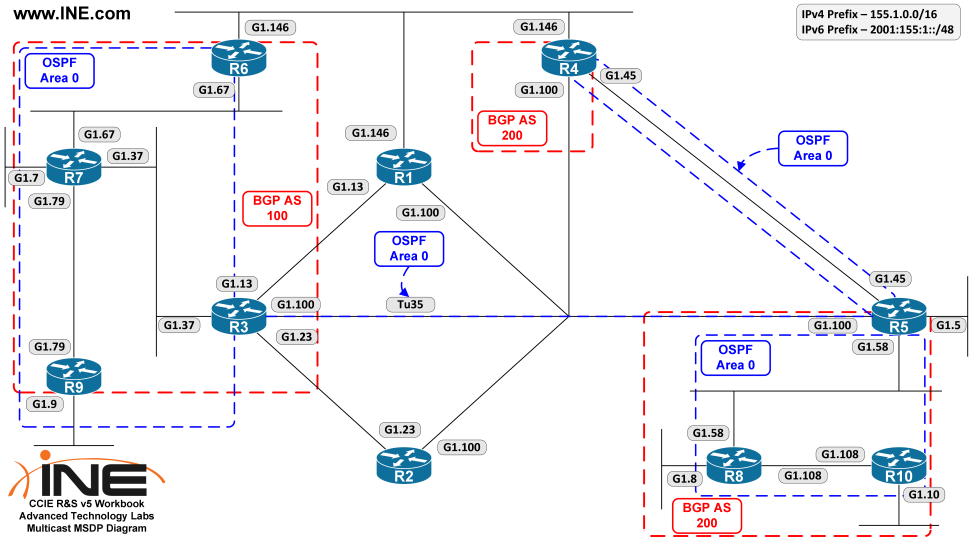
Advanced Technology Labs OSPF Diagram



Advanced Technology Labs Multicast Diagram



Advanced Technology Labs Multicast MSDP Diagram



https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Workbook Overview

CCIE R&S v4 Topology Diagrams & Initial Configurations

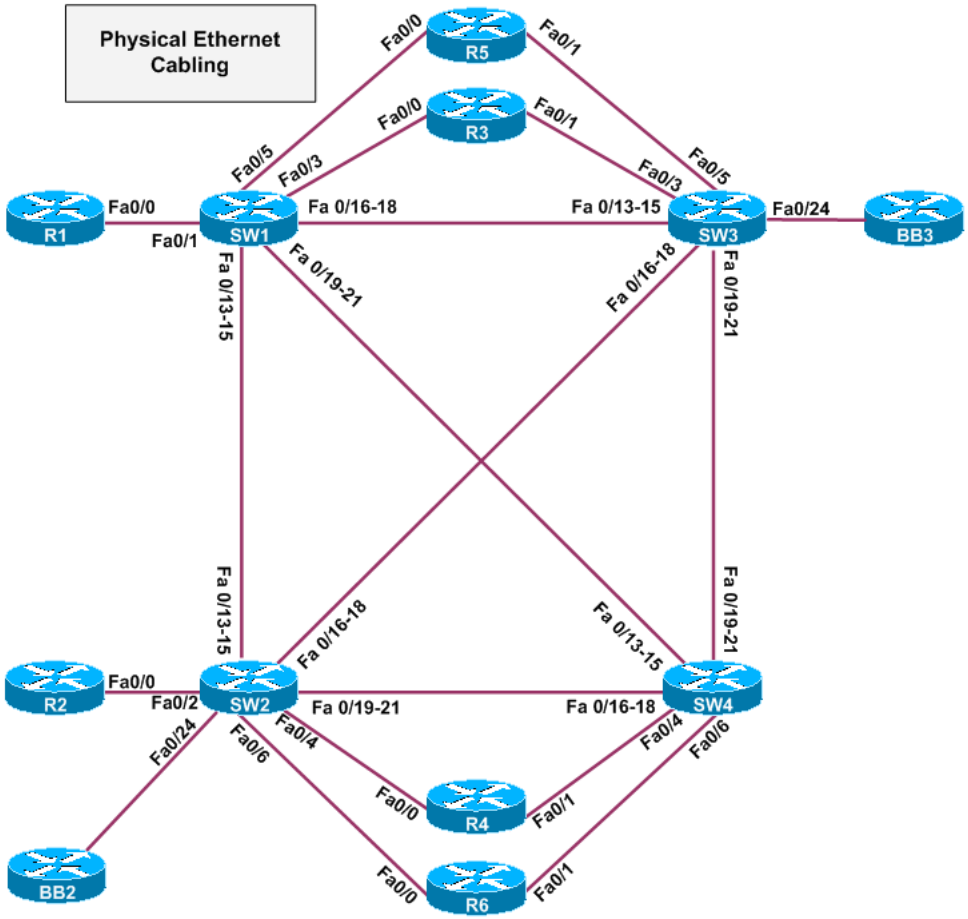
> Use these diagrams and initial configurations for tasks that are listed as **(pending update)** in the table of contents.

There are three main diagrams supplied with this workbook: two physical cabling diagrams and the Logical Layer 3 addressing diagram. These should be used together to give you a complete understanding of the network topology. In general, there are no separate diagrams per section. For sections that have specific pre-configurations, such as parts of BGP and Multicast, additional diagrams are provided.

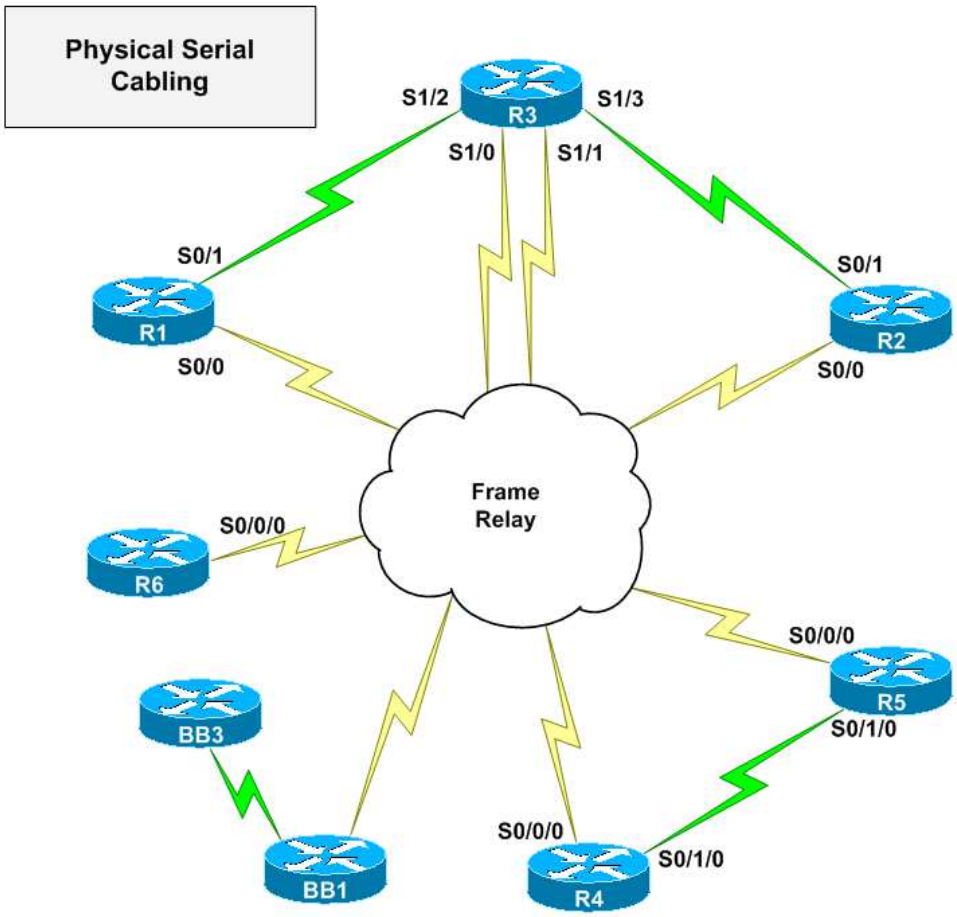
Assume that these three main diagrams are the foundation for every section in this workbook. We highly recommend that you re-draw the Logical Layer 3 diagram and extend it as appropriate for every section—for example, adding routing protocol domains and additional addressing if used. Remember that some sections, such as those centered around Layer 2 technologies, may not make use of the Layer 3 diagram at all, because they concentrate mainly on bridging and switching topics.

Click the **Resources** button on the right to download the initial configurations and diagrams for these labs.

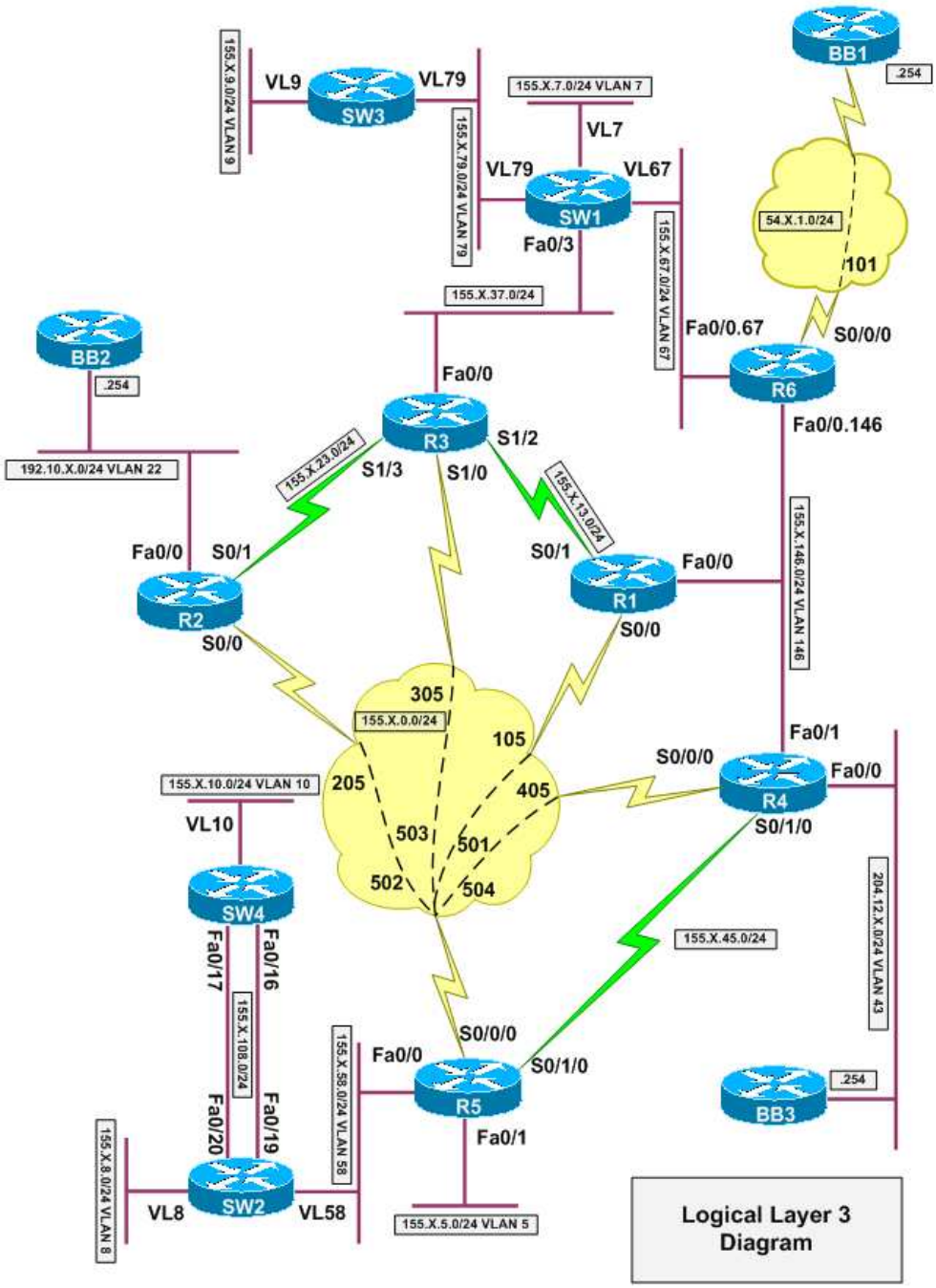
https://t.me/learningnets



https://t.me/learningnets



https://t.me/learningnets



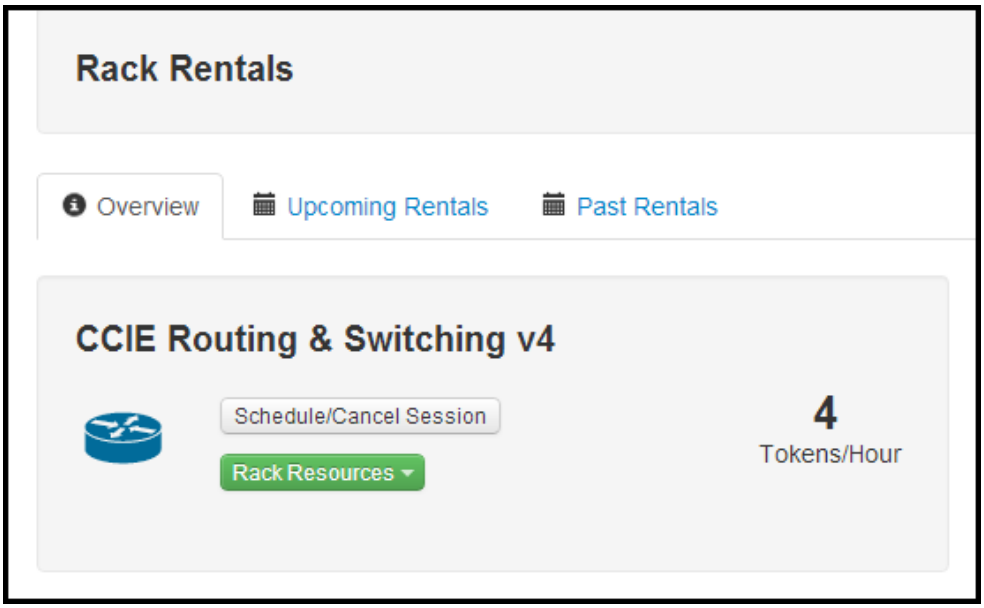
https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Workbook Overview

README: CCIE R&S v5 Topology Changes

> Rack rentals for the v5 topology will be available in beta starting the first week of May. A discussion thread about the CCIE RSv5 Equipment Build can be found here.

Currently the CCIE R&S v5 Workbook is in a state of change between our CCIE R&S v4 Topology and CCIE R&S v5 Topology. Tasks that are still formatted for the v4 topology are listed as **(pending update)** in the table of contents. When working on these tasks please reference the CCIE R&S v4 Topology Diagrams and Initial Configurations. If you are renting rack time from INE to configure these tasks you should use the following scheduler on the Rack Rentals Dashboard:



For all other tasks, please reference the CCIE R&S v5 Topology Diagrams and Initial Configurations.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Workbook Overview

CCIE R&S v5 Workbook Release Notes

> Please check back here periodically for release notes on workbook updates.

Changes by Date

Oct 4, 2014

- Added Troubleshooting Lab 1
- Added Full-Scale Lab 1

Jul 7, 2014

- Changes to Initial Configs, for Multicast and IPv6 tasks.
- Multicast and IPv6 tasks have been finished.

Jun 5, 2014

- Changes to Initial Configs, mainly for the LAN Switching Tasks.

May 16, 2014

- Minor change to Initial Configs .zip file to fix directory naming structure.

May 15, 2014

- Updated Initial Configs .zip file

May 13, 2014

- Added *INE's CCIE R&S v5 Hardware Topology* document

https://t.me/learningnets

May 8, 2014

- Added DMVPN Initial Configurations
- Added the following new sections
  - DMVPN without IPsec
  - DMVPN with IPsec
  - DMVPN Phase 1 with EIGRP
  - DMVPN Phase 1 with OSPF
- Added *Advanced Technology Labs BGP Diagram*

May 2, 2014

- Initial workbook release.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Workbook Overview

CCIE R&S v5 Workbook Overview

INE's CCIE R&S v5 Workbook is currently in intial beta release and will be continually updated in the coming days and weeks. Be sure to track the CCIE R&S v5 Workbook Release Notes, where workbook additions and changes will be listed. Also be sure read about the CCIE R&S v5 Workbook Topology Changes. Finally, join us on this IEOC discussion thread about the CCIE RSv5 Equipment Build.

About INE’s CCIE Routing & Switching v5 Workbook

INE’s CCIE Routing & Switching v5 Workbook is the definitive resource to master the technologies covered on the CCIE lab exam. The workbook follows a structured design that covers not only the necessary topic domains, but also lab strategy and other key test-taking skills. The workbook is broken into four main sections, as described below.

View the IEOC discussion boards for this workbook here.

Advanced Technology Labs

The Advanced Technology Labs are one of the first steps toward CCIE lab preparation. This section consists of nearly 500 hands-on labs that walk you through each and every technology, and provide in-depth explanations of how their configurations work. Topics are presented in an easy-to-follow, goal-oriented, step-by-step approach. These scenarios feature detailed breakdowns and thorough verifications to help you completely understand each technology at an expert level.

Join the IEOC discussion for this section here.

https://t.me/learningnets

Advanced Foundation Labs

The Advanced Foundation Labs are where the overall pieces of the puzzle start to fit together. These labs are designed to refine your configuration skills on the core technologies used in the CCIE lab exam. Each lab guides you through the critical steps necessary for building and verifying a working networking topology. The labs are designed to increase your speed and refine your task-management skills, capacities that are crucial when working in a timed full-scale lab environment.

Join the IEOC discussion for this section here.

Advanced Troubleshooting Labs

The Advanced Troubleshooting Labs present you with pre-built network topologies, in which you are tasked with resolving various problems that have been introduced. This section will help you develop a structured troubleshooting approach and improve your time-management skills, with a final result of troubleshooting becoming second nature. Improving your troubleshooting skills will not only help you pass the CCIE lab exam, but also help you with real-world job scenarios, which often require timely and accurate troubleshooting.

Join the IEOC discussion for this section here.

Full-Scale Practice Labs

The Full-Scale Practice Labs are the culmination of all your preparation, as you ready yourself for the actual CCIE lab exam. The full-scale labs are designed to simulate the CCIE Routing & Switching Lab Exam, while still illustrating the principles behind the technologies. Building upon your expert level understanding of the fundamentals, this section teaches you to be able to predict advanced and sometimes subtle interactions that occur when multiple technologies are combined together. When you have fully mastered the full-scale labs, you'll be ready to take and pass the CCIE lab exam!

Join the IEOC discussion for this section here.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Workbook Overview

INE's CCIE R&S v5 Hardware Topology

How To Build a CCIE Rack for CCIE R&S v5

This document details INE’s reference topology used in our CCIE Routing & Switching v5 products, such as our CCIE Routing & Switching v5 Workbook and CCIE Routing & Switching v5 Advanced Technologies Class. Specifically this document outlines what you would need in order to build the topology on your own.

Topology Overview

The topology can be built in a completely physical manner, a completely virtual manner, and a combination of both. Which option you choose depends on a number of factors, such as your budget, and space, power, & cooling limitations.

A full build of this topology consists of the following:

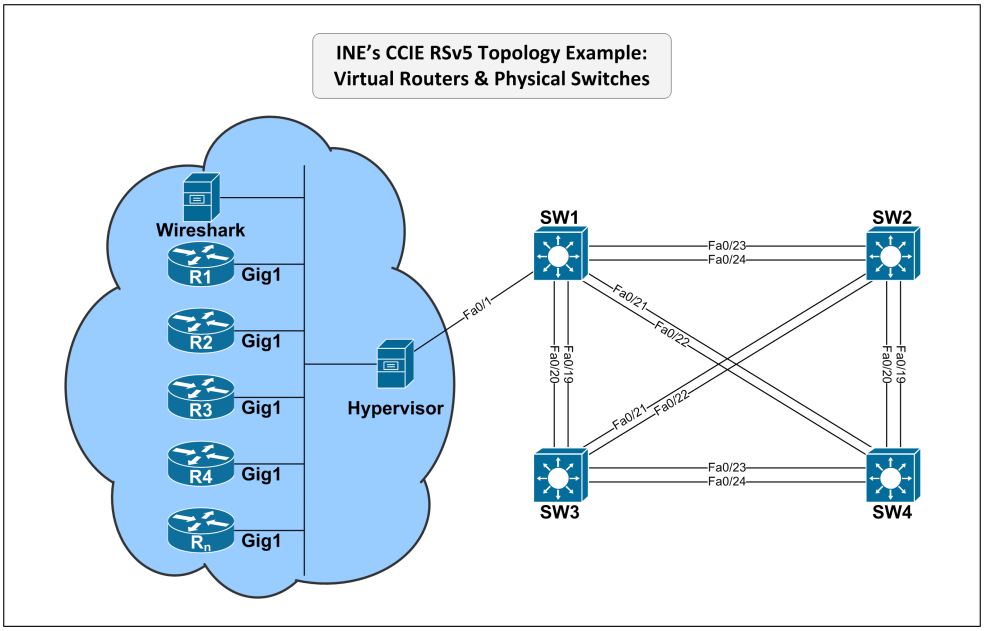
- QTY 20 IOS Routers running version 15.4S or 15.3T (virtual or physical)
- QTY 4 Catalyst IOS Switches running version 15.0SE (virtual or physical)
- Terminal Server / Access Server (optional)
- Remote Power Controllers (optional)

Physical & Virtual Wiring

Example topology wiring can be seen below when using a combination of virtual routers and physical switches, and when using a fully physical topology. For a fully physical topology a breakout switch is only required if you do not want to have to modify the initial configurations of SW1 in the INE workbook lab material.

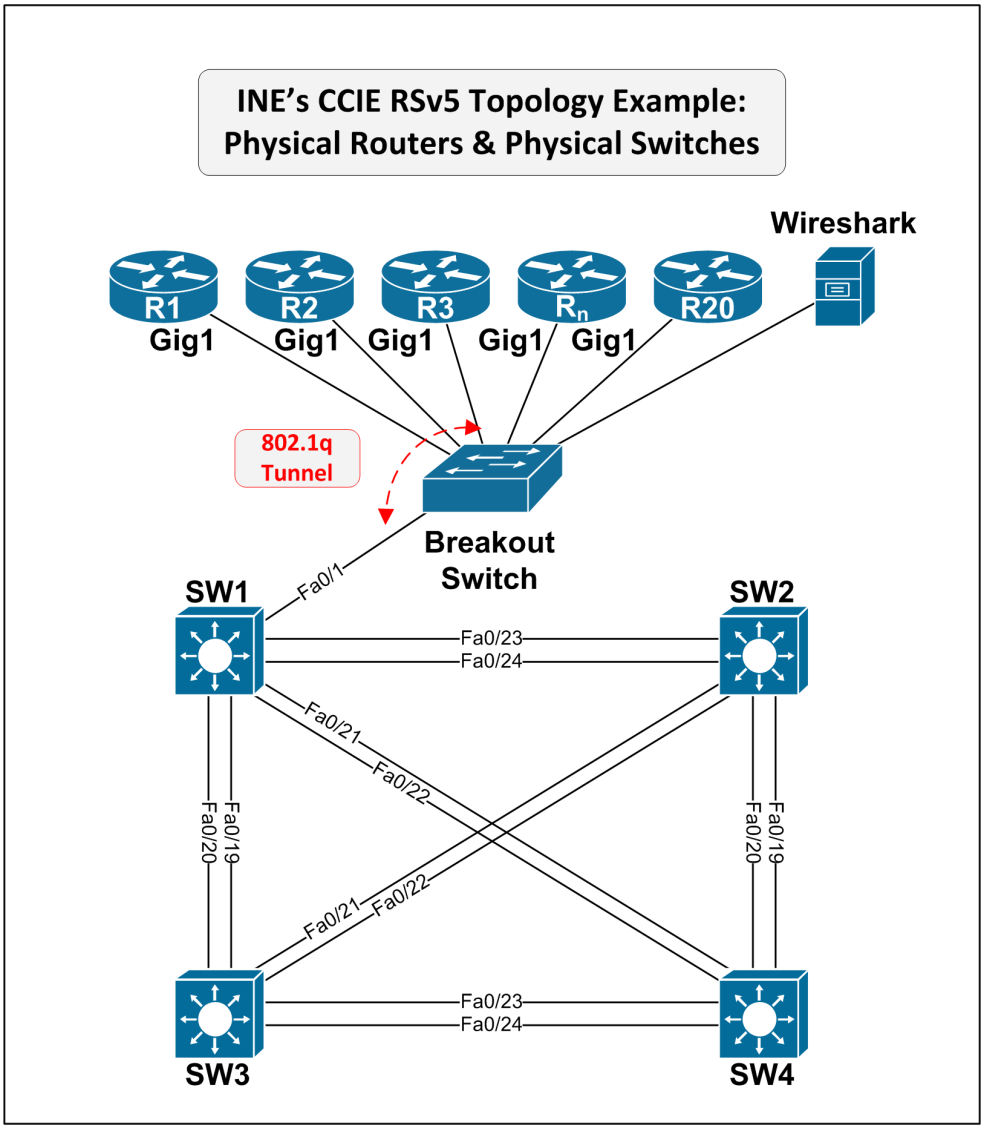
Topology Example: Virtual Routers & Physical Switches

https://t.me/learningnets



Topology Example: Physical Routers & Physical Switches

https://t.me/learningnets



Physical Router Platforms

Below are some examples of potential platforms that can be used when building the topology with physical routers. Note that the IOS version and feature set is more important than the actual platform itself, and that either newer or older platforms could also be used.

Ideal platform - ISR G2 (1900/2900/3900)

https://t.me/learningnets

The advantage of using ISR G2s is that 100% of all needed IOS features are supported when running IOS 15.3T Universal with feature sets IP Base, Data, & Security enabled. The disadvantage of this platform is generally the cost of the physical box plus full licensing is high, in addition to space, power, and cooling requirements.

Alternate platform - ISR G1 (1800/2800/3800)

The advantage of using ISR G1s is that the cost is generally lower than ISR G2. The disadvantage is that ISR G1 only officially supports up to IOS 15.1T with feature set Advanced Enterprise Services. Not all features tested on in CCIE RSv5 will be supported, but the vast majority will be. Space, power, and cooling requirements are still a large consideration with ISR G1, just as ISR G2.

Virtual Router Platforms

Below are some examples of potential platforms that can be used when building the topology with virtual routers.

Ideal platform – Cloud Services Router (CSR) 1000v

The advantage of using the CSR1000v is that 99% of all needed IOS features are supported when running IOS XE 3.11S (15.4S) with premium feature set. The disadvantage is that CSR1000v has large CPU & RAM requirements, and that Serial links are not supported. If using CSR1000v it is highly recommended to run it on a dedicated baremetal Hypervisor (i.e. a native install of ESXi, KVM, or XenServer) as opposed to inside desktop virtualization software (e.g. VirtualBox or VMWare Workstation).

Alternate platform - GNS3 with 7200 series routers

The advantage of using GNS3 is that the CPU & RAM requirements are lower than CSR1000v, and that most features are supported when emulating 7200 series routers running IOS 15.2S with feature set Advanced Enterprise Services. The disadvantage is that GNS3 is not as stable as CSR1000v or physical platforms, and some features may be unsupported or have unpredictable results. IOU or IOL could also be used, but are outside the scope of this document.

https://t.me/learningnets

Physical Switch Platforms

Below are some examples of physical switches that could be used to build the topology. Again note that the IOS version and feature set is more important than the actual platform itself, and that either newer or older platforms could also be used.

Ideal platform - Catalyst E or X (3560E/3560X/3750E/3750X)

The advantage of using Catalyst E or X is that 100% of all needed features are supported when running Catalyst IOS 15.0SE Universal with feature set IP Services. The disadvantage is generally the cost of the physical box plus full licensing is high.

Alternate platform - Non E/X Catalyst (3560/3560G/3750/3750G)

The advantage of using regular Catalyst switches is that their cost is generally much lower than E or X equivalents, while still supporting the vast majority of features needed. The disadvantage is that only platforms with 32MB Flash can run 15.0SE, and that platforms with 16MB Flash support only up to 12.2SE.

Virtual Switch Platforms – GNS3 with L2IOU

Switches can be emulated using L2IOU and GNS3, which is outside the scope of this document.

Terminal Server Platforms

A Terminal Server, sometimes called an Access Server or Console Server, can be used as a central point of management for the console sessions to any of the physical routers and switches in your lab build. A number of platforms could be used for this, such as:

- NM-16A or NM-32A modules in any modular router (2600/2800/3600/3800, etc.) with CAB-OCTAL-ASYNC cables.
- HWIC-16A or SM-32A in ISR G1 or ISR G2 with CAB-HD8-ASYNC cables.
- Non-Cisco solutions such as Opengear or Digi

https://t.me/learningnets

Remote Power Controllers

A Remote Power Controller (RPC) can be used to remotely power-on, power-off, or reboot your equipment. These can be especially useful not only to save energy, but allow you to do remote password recovery if you get locked out of any of your devices. Make sure that the device matches your power specifications and your outlet types, as lots of variations exist. A number of vendors make RPC devices, such as:

- APC
- Synaccess
- BayTech

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Workbook Overview

CCIE Routing & Switching v5 Rack Rental Guide

Click here for the CCIE Routing & Switching v5 Rack Rental Guide.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Layer 2 Access Switchports

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure SW1's port FastEthernet0/19 as a Layer 3 interface with the IP address 169.254.1.1/24.
- Configure SW2's port FastEthernet0/19 as a Layer 3 interface with the IP address 169.254.1.2/24.
- Configure ports FastEthernet0/19 on SW3 and SW4 to be access ports in VLAN 169.
- Configure FastEthernet0/23 and FastEthernet0/24 between SW3 and SW4 as dot1q trunk ports.
- For verification, test that SW1 and SW2 have IPv4 reachability to each other over VLAN 169.

Configuration

```
SW1:
interface FastEthernet0/19
 no switchport
 ip address 169.254.1.1 255.255.255.0
SW2:
interface FastEthernet0/19
 no switchport
 ip address 169.254.1.2 255.255.255.0
SW3:
```

https://t.me/learningnets

```
vlan 169
!
interface FastEthernet0/19
 switchport mode access
 switchport access vlan 169
!
interface range FastEthernet0/23 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
SW4:

vlan 169
!
interface FastEthernet0/19
 switchport mode access
 switchport access vlan 169
!
interface range FastEthernet0/23 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
```

Verification

SW1 and SW2 in this example are acting as end hosts. When end hosts are connected to different physical switches but are in the same VLAN, IP connectivity will be obtained only when Spanning-Tree Protocol is forwarding the VLAN end to end between switches connecting to the hosts. On the Catalyst platforms, an STP instance is automatically created for a VLAN when the VLAN is created. This implies that the first step in getting connectivity between the hosts is to create the VLAN.

Although the VLAN could also be learned through VTP, in this design the VLAN is simply manually defined on both switches, removing the need for VTP to be configured. Additionally, trunking must be configured on transit switches, SW3 and SW4, so that VLAN tagged frames can be sent over the links between them; optionally, as in this case we have a single VLAN required to be carried between SW3 and SW4, the links can be configured as access in VLAN 169.

Final verification in this example would be to ensure that the VLANs are assigned correctly according to the **`show interface status`** or **`show vlan`** output, and that end-to-end connectivity exists:

```
SW1#ping 169.254.1.2

Type escape sequence to abort.
```

https://t.me/learningnets

```
Sending 5, 100-byte ICMP Echos to 169.254.1.2, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
!
!SW2#ping 169.254.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.1.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
!
!SW3#show interface status


Port      Name               Status       Vlan       Duplex  Speed Type
Fa0/1                        notconnect   1            auto   auto 10/100BaseTX
Fa0/2                        notconnect   1            auto   auto 10/100BaseTX
Fa0/3                        notconnect   1            auto   auto 10/100BaseTX
Fa0/4                        notconnect   1            auto   auto 10/100BaseTX
Fa0/5                        notconnect   1            auto   auto 10/100BaseTX
Fa0/6                        notconnect   1            auto   auto 10/100BaseTX
Fa0/7                        notconnect   1            auto   auto 10/100BaseTX
Fa0/8                        notconnect   1            auto   auto 10/100BaseTX
Fa0/9                        notconnect   1            auto   auto 10/100BaseTX
Fa0/10                       notconnect   1            auto   auto 10/100BaseTX
Fa0/11                       no
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Layer 2 Dynamic Switchports

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure all inter-switch links on SW2, SW3, and SW4 to be in dynamic auto state.
- Configure all inter-switch links on SW1 to be in dynamic desirable state.
- For verification, ensure that:
  - SW1 Ethernet links to SW2, SW3, and SW4 are negotiated as trunks.
  - Ethernet links between SW2, SW3, and SW4 do not negotiate trunking and fallback to access mode.

Configuration

https://t.me/learningnets

```
SW1:

interface range FastEthernet0/19 - 24
 switchport mode dynamic desirable
SW2:

interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
SW3:

interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
SW4:

interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
```

Verification

With SW1's inter-switch links configured in dynamic desirable state, and all other inter-switch links configured in dynamic auto state, trunks will only be negotiated between SW1 to SW2, SW1 to SW3, and SW1 to SW4. This is because SW1 initiates trunking negotiation through DTP (desirable), and SW2, SW3, and SW4 only respond to DTP negotiation requests (auto). This can be verified as shown below, note that the output may differ for the "Vlans in spanning tree forwarding state and not pruned" based on which of the switches is the STP root bridge for VLAN 1.

```
SW1#show interface trunk



Port        Mode             Encapsulation  Status        Native vlan Fa0/19 desirable n-isl trunking
      1 Fa0/20 desirable n-isl trunking
      1 Fa0/21 desirable n-isl trunking
      1 Fa0/22 desirable n-isl trunking
      1 Fa0/23 desirable n-isl trunking
      1 Fa0/24 desirable n-isl trunking
      1


Port        Vlans allowed on trunk
Fa0/19      1-4094
Fa0/20      1-4094
Fa0/21      1-4094
```

https://t.me/learningnets

```
Fa0/22      1-4094
Fa0/23      1-4094
Fa0/24      1-4094

Port        Vlans allowed and active in management domain
Fa0/19      1
Fa0/20      1
Fa0/21      1
Fa0/22      1
Fa0/23      1
Fa0/24      1

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/19      1
Fa0/20      1
Fa0/21      1
Fa0/22      1
Fa0/23      1
Fa0/24      1
```

The output on SW3 is the same as on SW2 and SW4. None of these switches are trunking directly with each other, only with SW1.

```
SW3#show interfaces trunk


Port        Mode             Encapsulation  Status        Native vlan Fa0/19 auto n-isl trunking
      1 Fa0/20 auto n-isl trunking
      1

Port        Vlans allowed on trunk
Fa0/19      1-4094
Fa0/20      1-4094

Port        Vlans allowed and active in management domain
Fa0/19      1
Fa0/20      1

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/19      1
Fa0/20      none
```

As seen from above outputs, by default switches will also negotiate ISL instead of 802.1q as the trunking protocol. Verify the DTP port state of "dynamic desirable"

https://t.me/learningnets

and "dynamic auto"; also note the difference between "Administrative Mode," which defines how the port was configured to operate, and "Operational Mode," which defines how the port actually operates after DTP negotiation.

```
SW3#show interfaces fastEthernet0/19 switchport
Name: Fa0/19
Switchport: EnabledAdministrative Mode: dynamic auto
Operational Mode: trunk
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: islNegotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
!
!SW3#show interfaces fastEthernet0/21 switchport
Name: Fa0/21
Switchport: EnabledAdministrative Mode: dynamic auto
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: nativeNegotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
```

https://t.me/learningnets

```
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
!
!SW1#show interfaces fastEthernet0/19 switchport
Name: Fa0/19
Switchport: EnabledAdministrative Mode: dynamic desirable
Operational Mode: trunk
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: islNegotiation of Trunking: On

Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
```

https://t.me/learningnets

```
Appliance trust: none
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

802.1q Dynamic Trunking

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure all inter-switch links on SW2, SW3, and SW4 to be in dynamic auto state.
- Configure all inter-switch links on SW1 to be in dynamic desirable state.
  - Configure the trunking encapsulation on SW1's inter-switch links as static 802.1q.
- For verification, ensure that:
  - SW2, SW3, and SW4 are negotiating 802.1q as the trunking encapsulation to SW1.
  - SW1 is not negotiating 802.1q as the trunking encapsulation to SW2, SW3, and SW4.

Configuration

```
SW1:
interface range FastEthernet0/19 - 24
 switchport mode dynamic desirable
 switchport trunk encapsulation dot1q
SW2:
interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
SW3:
interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
```

https://t.me/learningnets

```
SW4:

interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
```

Verification

Similar to the previous case, SW1 is running in DTP desirable mode, so it is negotiating trunking but now has its trunking encapsulation statically set to 802.1q.

```
SW1#show interface trunk

Port        Mode             Encapsulation  Status        Native vlan
Fa0/19      desirable        802.1q
         trunking      1 Fa0/20      desirable        802.1q
         trunking      1 Fa0/21      desirable        802.1q
         trunking      1 Fa0/22      desirable        802.1q
         trunking      1 Fa0/23      desirable        802.1q
         trunking      1 Fa0/24      desirable        802.1q
         trunking      1
<output omitted>
```

SW2, SW3, and SW4 must now agree to use dot1q trunking through DTP negotiation, as seen in the **n-802.1q** output, which stands for **negotiated-802.1q**.

```
SW2#show interface trunk

Port        Mode         Encapsulation  Status        Native vlan Fa0/23      auto          n-802.1q
       trunking      1 Fa0/24      auto          n-802.1q
       trunking      1
<output omitted>
!
!SW3#show interface trunk

Port        Mode         Encapsulation  Status        Native vlan Fa0/19      auto          n-802.1q
       trunking      1 Fa0/20      auto          n-802.1q
       trunking      1
<output omitted>
!
!SW4#show interface trunk

Port        Mode         Encapsulation  Status        Native vlan Fa0/21      auto          n-802.1q
       trunking      1 Fa0/22      auto          n-802.1q
       trunking      1
```

https://t.me/learningnets

```
<output omitted>
```

The fact that SW1 has its trunking protocol manually configured while all other switches negotiate it can be seen in following output.

```
SW1#show interfaces fastEthernet0/19 switchport
Name: Fa0/19
Switchport: Enabled
Administrative Mode: dynamic desirable
Operational Mode: trunkAdministrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
<output omitted>
!
!SW2#show interfaces fastEthernet0/19 switchport
Name: Fa0/19
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: static accessAdministrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native

Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
<output omitted>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

802.1q Native VLAN

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure all inter-switch links on SW1 to be in dynamic desirable state.
- Configure all inter-switch links of SW2, SW3, and SW4 toward SW1 to be in dynamic auto state.
- Configure the trunking encapsulation on SW1's inter-switch links as static 802.1q.
- Configure the switches so that traffic between devices in VLAN 146 is not tagged when sent over the trunk links.

Configuration

```
SW1:
vlan 146
!
interface range FastEthernet0/19 - 24
 switchport mode dynamic desirable
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 146
SW2:
vlan 146
!
interface range FastEthernet0/23 - 24
 switchport mode dynamic auto
 switchport trunk native vlan 146
SW3:
```

https://t.me/learningnets

```
vlan 146
!
interface range FastEthernet0/19 - 20
 switchport mode dynamic auto
 switchport trunk native vlan 146
SW4:

vlan 146
!
interface range FastEthernet0/21 - 22
 switchport mode dynamic auto
 switchport trunk native vlan 146
```

Verification

The IEEE 802.1q trunking encapsulation standard uses the term *native VLAN* to describe traffic sent and received on an interface running 802.1q encapsulation that does not have an 802.1q tag actually inserted. Native VLAN was preserved for backward compatibility so that frames can still transit switches not yet capable for 802.1q.

When a switch needs to forward a frame outbound on a trunk link and the frame was received from a VLAN that is the same as the native VLAN of the trunk link, the frame is sent untagged as if 802.1q were not configured. When the switch receives a untagged frame on an interface running 802.1q, it associates the frame with the native VLAN of its trunk port on which the frame was received. The native VLAN is not configured switch-wide, it is port specific. For example, a switch may be configured to have VLAN 20 as native VLAN on its FastEthernet0/19 port and VLAN 40 as native VLAN on its FastEthernet0/20 port. The switches on both ends of an 802.1q trunk link must agree on what the native VLAN is; otherwise, traffic can unexpectedly leak between broadcast domain boundaries. The native VLAN is not negotiated between switches; it is your responsibility to configure it the same on both ends of the trunk link.

If, however, you've configured a different native VLAN on the two ends of a trunk link, this will be detected through CDP which will log a warning messages, and STP which will logically disable the port to avoid forwarding loops. The native VLAN defaults to 1 on all links unless modified. In this case, the native VLAN is modified to 146 on both ends of the link.

```
SW1#show interface trunk
 Port        Mode         Encapsulation  Status Native vlan
Fa0/19      desirable        802.1q         trunking 146
```

https://t.me/learningnets

```
Fa0/20      desirable        802.1q         trunking 146
Fa0/21      desirable        802.1q         trunking 146
Fa0/22      desirable        802.1q         trunking 146
Fa0/23      desirable        802.1q         trunking 146
Fa0/24      desirable        802.1q         trunking 146
<output omitted>
!
!SW2#show interface trunk
Port        Mode         Encapsulation  Status Native vlan
Fa0/23      auto         n-802.1q       trunking 146
Fa0/24      auto         n-802.1q       trunking 146
<output omitted>
!
!SW3#show interface trunk
Port        Mode         Encapsulation  Status Native vlan
Fa0/19      auto         n-802.1q       trunking 146
Fa0/20      auto         n-802.1q       trunking 146
<output omitted>
!
!SW4#show interface trunk
Port        Mode         Encapsulation  Status Native vlan
Fa0/21      auto         n-802.1q       trunking 146
Fa0/22      auto         n-802.1q       trunking 146

<output omitted>
```

Verify that the default native VLAN of 1 has been changed to VLAN 146.

```
SW1#show interfaces fastEthernet0/23 switchport
Name: Fa0/23
Switchport: Enabled
Administrative Mode: dynamic desirable
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 146 (VLAN0146)
Administrative Native VLAN tagging: enabled
!
!SW2#show interfaces fastEthernet0/23 switchport
Name: Fa0/23
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: trunk
Administrative Trunking Encapsulation: negotiate
```

https://t.me/learningnets

```
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 146 (VLAN0146)

Administrative Native VLAN tagging: enabled
```

Let's break the configuration by using a different native VLAN on the ends of the trunk link.

```
SW1#configure terminal
SW1(config)#interface range fastEthernet0/23 - 24
SW1(config-if-range)#shutdown
SW1(config-if-range)#switchport trunk native vlan 1
SW1(config-if-range)#no shutdown
```

The following log messages will be triggered by CDP, as the native VLAN value is sent through CDP advertisements.

```
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch
 discovered on FastEthernet0/23 (1), with SW2 FastEthernet0/23 (146).%CDP-4-NATIVE_VLAN_MISMATCH:
Native VLAN mismatch
 discovered on FastEthernet0/24 (1), with SW2 FastEthernet0/24 (146).
```

The following log messages will be triggered by STP, logically blocking the port.

```
%SPANTREE-2-RECV_PVID_ERR: Received BPDU with inconsistent peer vlan id 146 on FastEthernet0/24 VLAN1.
%SPANTREE-2-BLOCK_PVID_PEER: Blocking FastEthernet0/24 on VLAN0146. Inconsistent peer vlan.
%SPANTREE-2-BLOCK_PVID_LOCAL: Blocking FastEthernet0/24 on VLAN0001. Inconsistent local vlan.
%SPANTREE-2-RECV_PVID_ERR: Received BPDU with inconsistent peer vlan id 146 on FastEthernet0/23 VLAN1.
%SPANTREE-2-BLOCK_PVID_PEER: Blocking FastEthernet0/23 on VLAN0146. Inconsistent peer vlan.
%SPANTREE-2-BLOCK_PVID_LOCAL: Blocking FastEthernet0/23 on VLAN0001. Inconsistent local vlan.
```

Verify that from STP perspective, ports are blocked, which means no data-plane traffic can be forwarded out on the trunks and all inbound data-plane frames are dropped; however, ports are in the **UP** state.

```
SW1#show ip interface brief | i 0/23|0/24
FastEthernet0/23       unassigned      YES unset  up                    up
FastEthernet0/24       unassigned      YES unset  up                    up
!
!SW1#show spanning-tree vlan 1 interface fastEthernet0/23
```

https://t.me/learningnets

```
Vlan                 Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
VLAN0001             Desg BKN*19        128.25 P2p *PVID_Inc
!
!SW1#show spanning-tree inconsistentports

Name                 Interface                Inconsistency
-------------------- ------------------------ ------------------VLAN0001
FastEthernet0/23          Port VLAN ID Mismatch
VLAN0001             FastEthernet0/24         Port VLAN ID MismatchVLAN0146
FastEthernet0/23          Port VLAN ID Mismatch

VLAN0146             FastEthernet0/24         Port VLAN ID Mismatch

Number of inconsistent ports (segments) in the system : 4
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

DTP Negotiation

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure static 802.1q trunk links between SW1 and all other switches.
  - Disable Dynamic Trunking Protocol on all of these ports.
- Configure all other inter-switch links between SW2, SW3, and SW4 to be in dynamic auto state.
- For verification, ensure that trunk links between SW1 and all other switches do not use DTP.

Configuration

```
SW1:
interface range FastEthernet0/19 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate
SW2:
interface range FastEthernet0/19 - 22
 switchport mode dynamic auto
!
interface range FastEthernet0/23 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate
SW3:
```

https://t.me/learningnets

```
interface range FastEthernet0/19 - 20
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate
!
interface range FastEthernet0/21 - 24
 switchport mode dynamic auto
```

SW4:

```
interface range FastEthernet0/19 - 20
 switchport mode dynamic auto
!
interface range FastEthernet0/21 - 22
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate
!
interface range FastEthernet0/23 - 24
 switchport mode dynamic auto
```

Verification

DTP negotiation can be disabled with either the **`switchport mode access`** command or the **`switchport nonegotiate`** command. If trunking is needed but DTP is disabled, the port must be statically configured with the **`switchport mode trunk`** command. This design is most commonly used when a switch is trunking to a device that does not support DTP, such as an IOS router's routed Ethernet interface (not an EtherSwitch interface) or a server's NIC card.

```
SW1#show interface fastethernet0/19 switchport | include Negotiation
Negotiation of Trunking: Off
!
!SW1#show interface trunk
Port Mode
         Encapsulation  Status        Native vlan Fa0/19 on
               802.1q         trunking      1 Fa0/20 on
               802.1q         trunking      1 Fa0/21 on
               802.1q         trunking      1 Fa0/22 on
               802.1q         trunking      1 Fa0/23 on
               802.1q         trunking      1 Fa0/24 on
               802.1q         trunking      1
<output omitted>
!
```

https://t.me/learningnets

```
!SW2#show interface trunk

Port Mode
        Encapsulation  Status        Native vlan Fa0/23 on
          802.1q         trunking      1 Fa0/24 on
          802.1q         trunking      1
<output omitted>
!
!SW3#show interface trunk
Port Mode
        Encapsulation  Status        Native vlan Fa0/19 on
          802.1q         trunking      1 Fa0/20 on
          802.1q         trunking      1
<output omitted>
!
!SW4#show interface trunk
Port Mode
        Encapsulation  Status        Native vlan Fa0/21 on
          802.1q         trunking      1 Fa0/22 on
          802.1q         trunking      1
<output omitted>
```

Verify DTP statistics on both DTP-enabled and DTP-disabled interfaces, and note that interface access/trunk state is displayed.

```
SW2#show dtp interface fastEthernet0/19
DTP information for FastEthernet0/19:
  TOS/TAS/TNS:                              ACCESS/AUTO/ACCESS
  TOT/TAT/TNT:                              NATIVE/NEGOTIATE/NATIVE
  Neighbor address 1:                       001AA1742515
  Neighbor address 2:                       000000000000
Hello timer expiration (sec/state):       15/RUNNING
  Access timer expiration (sec/state):      never/STOPPED
  Negotiation timer expiration (sec/state): never/STOPPED
  Multidrop timer expiration (sec/state):   never/STOPPED
FSM state:                                S2:ACCESS
  # times multi & trunk                     0
  Enabled:                                  yes
  In STP:                                   no

  Statistics
  ----------
  372 packets received (372 good)
  0 packets dropped
      0 nonegotiate, 0 bad version, 0 domain mismatches,
```

https://t.me/learningnets

```
      0 bad TLVs, 0 bad TAS, 0 bad TAT, 0 bad TOT, 0 other
  748 packets output (748 good)
      374 native, 374 software encap isl, 0 isl hardware native
  0 output errors
  0 trunk timeouts
  1 link ups, last link up on Wed Mar 24 1993, 12:07:57
  0 link downs
!
!
SW2#show dtp interface fastEthernet0/23
DTP information for FastEthernet0/23:
  TOS/TAS/TNS:                              TRUNK/NONEGOTIATE/TRUNK
  TOT/TAT/TNT:                              802.1Q/802.1Q/802.1Q
  Neighbor address 1:                       0013605FF019
  Neighbor address 2:                       000000000000
  Hello timer expiration (sec/state):       never/STOPPED
  Access timer expiration (sec/state):      never/STOPPED
  Negotiation timer expiration (sec/state): never/STOPPED
  Multidrop timer expiration (sec/state):   never/STOPPED
  FSM state:                                S6:TRUNK

  # times multi & trunk                     0
  Enabled:                                  yes
  In STP:                                   no

  Statistics
  ----------
  243 packets received (243 good)
  0 packets dropped
      0 nonegotiate, 0 bad version, 0 domain mismatches,
      0 bad TLVs, 0 bad TAS, 0 bad TAT, 0 bad TOT, 0 other
  247 packets output (247 good)
      244 native, 3 software encap isl, 0 isl hardware native
  0 output errors
  0 trunk timeouts
  3 link ups, last link up on Wed Mar 24 1993, 13:06:13
  2 link downs, last link down on Wed Mar 24 1993, 13:05:34
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

VTP Domain

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure all inter-switch links on SW2, SW3, and SW4 to be in dynamic auto state.
- Configure all inter-switch links on SW1 to be in dynamic desirable state.
- Configure SW2 as a VTP server in the domain named **CCIE**.
  - Configure SW1, SW3, and SW4 as VTP clients in the domain **CCIE**.
  - Authenticate VTP messages using the string of **VTPPASS**.
- Configure VLANs 5, 7, 8, 9, 10, 22, 43, 58, 67, 79, and 146 on SW2.
- For verification, ensure that SW1, SW3, and SW4 learn about these new VLANs through VTP.

Configuration

```
SW1:
vtp domain CCIE
vtp mode client
vtp password VTPPASS
!
interface range FastEthernet0/19 - 24
 switchport mode dynamic desirable
SW2:
vtp domain CCIE
vtp password VTPPASS
vlan 5,7,8,9,10,22,43,58,67,79,146
```

https://t.me/learningnets

```
!
interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
SW3:
vtp domain CCIE
vtp mode client
vtp password VTPPASS
!
interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
SW4:

vtp domain CCIE
vtp mode client
vtp password VTPPASS
!
interface range FastEthernet0/19 - 24
 switchport mode dynamic auto
```

Verification

VLAN Trunking Protocol (VTP) can be used in the Ethernet domain to simplify the creation and management of VLANs, but it does not dictate the traffic flow of VLANs or the actual port assignments to VLANs. The first step in running VTP is to ensure that the switches are trunking with each other (it can be ISL or 802.1q; VTP runs over both). Next, the VTP domain name is configured, and all other switches without domain names configured will dynamically learn the domain name. VTP password is optional but it cannot be learned through VTP because it is not sent in VTP messages; an MD5 hash is sent instead, so it must be manually configured on all devices. Finally, the VLAN definitions are created on the VTP server.

To verify this configuration, compare the output of the **`show vtp status`** command on all devices in the domain. If the domain name, the number of existing VLANs, and the Configuration Revision Number match, the domain is converged. If authentication is configured, the MD5 digest field should be compared as well.

```
SW1#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1 VTP Domain Name                 : CCIE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 000a.b832.3580
Configuration last modified by 0.0.0.0 at 3-1-93 02:36:18
```

https://t.me/learningnets

```
Feature VLAN:
--------------VTP Operating Mode                :Client
Maximum VLANs supported locally   : 1005Number of existing VLANs          : 16
Configuration Revision            : 1
MD5 digest                        : 0xD2 0x47 0xDC 0xAD 0x66 0xEE 0x31 0x42
                                    0xEF 0x6E 0x13 0x4B 0xD4 0x1C 0x37 0x65
!
!SW2#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1VTP Domain Name                 : CCIE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 001c.576d.4a00
Configuration last modified by 0.0.0.0 at 3-1-93 02:36:18
Local updater ID is 0.0.0.0 (no valid interface found)

Feature VLAN:
--------------VTP Operating Mode                :Server
Maximum VLANs supported locally   : 1005Number of existing VLANs          : 16
Configuration Revision            : 1
MD5 digest                        : 0xD2 0x47 0xDC 0xAD 0x66 0xEE 0x31 0x42
                                 0xEF 0x6E 0x13 0x4B 0xD4 0x1C 0x37 0x65
!
!SW3#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1VTP Domain Name                 : CCIE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 001d.45cc.0580
Configuration last modified by 0.0.0.0 at 3-1-93 02:36:18

Feature VLAN:
--------------VTP Operating Mode                :Client
Maximum VLANs supported locally   : 1005Number of existing VLANs          : 16
Configuration Revision            : 1
MD5 digest                        : 0xD2 0x47 0xDC 0xAD 0x66 0xEE 0x31 0x42
                                    0xEF 0x6E 0x13 0x4B 0xD4 0x1C 0x37 0x65
!
!SW4#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1VTP Domain Name                 : CCIE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 001c.576d.3d00
Configuration last modified by 0.0.0.0 at 3-1-93 02:36:18
```

https://t.me/learningnets

```
Feature VLAN:
--------------VTP Operating Mode                :Client
Maximum VLANs supported locally   : 1005Number of existing VLANs          : 16
Configuration Revision            : 1

MD5 digest                        : 0xD2 0x47 0xDC 0xAD 0x66 0xEE 0x31 0x42
                                    0xEF 0x6E 0x13 0x4B 0xD4 0x1C 0x37 0x65
```

The output of **`show vtp status`** confirms that the VTP password has been correctly configured on all switches, because the same MD5 digest has been computed on all devices, but the password can be verified separately.

```
SW1#show vtp password
VTP Password:VTPPASS
!
!SW2#show vtp password
VTP Password:VTPPASS
!
!SW3#show vtp password
VTP Password:VTPPASS
!
!SW4#show vtp password
VTP Password:VTPPASS
```

The commands **`show vlan`** and **`show vlan brief`** can also be compared to ensure that the VLAN numbers and names properly propagated throughout the VTP domain.

```
SW1#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Gi0/1, Gi0/25    VLAN0005
                        active    7    VLAN0007
                        active    8    VLAN0008
                        active    9    VLAN0009
                        active    10   VLAN0010
                        active    22   VLAN0022
                        active    43   VLAN0043
                        active    58   VLAN0058
                        active    67   VLAN0067
```

https://t.me/learningnets

```
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
!
!SW2#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Gi0/1, Gi0/25
                         VLAN0005
                         active    7    VLAN0007
                         active    8    VLAN0008
                         active    9    VLAN0009
                         active    10   VLAN0010
                         active    22   VLAN0022
                         active    43   VLAN0043
                         active    58   VLAN0058
                         active    67   VLAN0067
                         active    79   VLAN0079
                         active
146  VLAN0146                         active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
!
!SW3#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24, Gi0/1, Gi0/25
                         VLAN0005
                         active    7    VLAN0007
                         active    8    VLAN0008
                         active    9    VLAN0009
                         active    10   VLAN0010
                         active    22   VLAN0022
                         active    43   VLAN0043
                         active    58   VLAN0058
```

https://t.me/learningnets

```
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
!
!SW4#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/23, Fa0/24, Gi0/1, Gi0/25
5    VLAN0005
                         active
7    VLAN0007
                         active
8    VLAN0008
                         active
9    VLAN0009
                         active
10   VLAN0010
                         active
22   VLAN0022
                         active
43   VLAN0043
                         active
58   VLAN0058
                         active
67   VLAN0067
                         active
79   VLAN0079
                         active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

VTP Transparent

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Ethernet links between SW1 and all other switches as static 802.1q trunks.
- Configure VTP version 2 in domain **CCIE** as follows:
  - SW1 in transparent mode
  - SW2 in server mode
  - SW3 and SW4 in client mode
- Configure VLANs 5, 7, 8, 9, and 10 on VTP server.
- Ensure that traffic between hosts within same VLAN is functional regardless of the switch being connected to.

Configuration

```
SW1:
vtp domain CCIE
vtp version 2
vtp mode transparent
vlan 5,7,8,9,10
!
interface range FastEthernet0/19 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
SW2:
```

https://t.me/learningnets

```
vtp domain CCIE
vtp version 2
vlan 5,7,8,9,10
!
interface range FastEthernet0/23 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
SW3:
vtp domain CCIE
vtp version 2
vtp mode client
!
interface range FastEthernet0/19 - 20
 switchport trunk encapsulation dot1q
 switchport mode trunk
SW4:

vtp domain CCIE
vtp version 2
vtp mode client
!
interface range FastEthernet0/21 - 22
 switchport trunk encapsulation dot1q
 switchport mode trunk
```

Verification

VTP version, just like VTP domain name, can be dynamically learned from VTP advertisements (if the VTP mode is client or server), but it is configured on all switches for consistency; VTP version cannot be changed on devices running in client mode. VTP devices running in transparent mode do not install VTP updates received, but will continue to forward them unmodified if the domain name of received VTP advertisements matches its locally configured domain. The configuration revision number of zero confirms that received VTP updates do not affect the local VLAN database.

```
SW1#show vtp status
VTP Version capable             : 1 to 3 VTP version running             : 2
VTP Domain Name                 : CCIE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 0013.605f.f000
Configuration last modified by 0.0.0.0 at 3-24-93 21:11:43
```

https://t.me/learningnets

```
Feature VLAN:
--------------VTP Operating Mode                : Transparent
Maximum VLANs supported locally   : 1005
Number of existing VLANs          : 10Configuration Revision            : 0

MD5 digest                        : 0x6B 0x36 0x65 0xF9 0xD9 0x10 0x51 0xED
                                    0xA8 0x25 0xC5 0x35 0xC9 0x38 0x9F 0x0F
```

Because VTP is control-plane only and does not directly relate to STP forwarding, VTP traffic from the server/client or from an entirely different VTP domain can be in the same broadcast domain as VTP transparent switches. In this particular case, SW1 must be locally configured with all VLANs from the VTP domain, because it is in the physical Layer 2 transit path for data-plane traffic within those VLANs. If a switch receives tagged frames for which the VLAN does not exist in the database, frames are silently dropped; this can be seen from the fact that the switch does not have any of its ports in STP forwarding state for non-existing VLANs. Before VLANs are manually configured on SW1:

```
SW2#show spanning-tree interface fastEthernet0/23

Vlan                Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
VLAN0001            Desg FWD 19        128.25   P2p
VLAN0005            Desg FWD 19        128.25   P2p
VLAN0007            Desg FWD 19        128.25   P2p
VLAN0008            Desg FWD 19        128.25   P2p
VLAN0009            Desg FWD 19        128.25   P2p
VLAN0010            Desg FWD 19        128.25   P2p
!SW1#show spanning-tree interface fastEthernet0/23


Vlan                Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
VLAN0001            Root FWD 19        128.25   P2p
```

Verify that the VLAN database has been learned by VTP clients, so the VTP device in transparent mode, SW1, has relayed the VTP messages between its trunk ports (from VTP server to VTP clients).

```
SW2#show vlan  brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
```

https://t.me/learningnets

```
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Gi0/1, Gi0/25    VLAN0005
                         active     7    VLAN0007
                         active     8    VLAN0008
                         active     9    VLAN0009
                         active     10   VLAN0010
                         active
1002 fddi-default                       act/unsup
1003 token-ring-default                 act/unsup
1004 fddinet-default                    act/unsup
1005 trnet-default                      act/unsup
!
!SW3#show vlan brief

VLAN Name                               Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                            active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                  Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                  Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                  Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                  Fa0/17, Fa0/18, Fa0/21, Fa0/22
                                                  Fa0/23, Fa0/24, Gi0/1, Gi0/25    VLAN0005
                         active     7    VLAN0007
                         active     8    VLAN0008
                         active     9    VLAN0009
                         active     10   VLAN0010
                         active
1002 fddi-default                       act/unsup
1003 token-ring-default                 act/unsup
1004 fddinet-default                    act/unsup
1005 trnet-default                      act/unsup
```

Changes in the rest of the VTP domain, such as VLAN adds or removes, does not affect the transparent switches, which just relay VTP messages.

```
SW1#debug sw-vlan vtp events
vtp events debugging is on
!
!SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW2(config)#vlan 123
!
!SW3#show vlan | include ^123
123  VLAN0123                           active
```

https://t.me/learningnets

```
123  enet  100123     1500  -      -      -        -    -        0      0
!
!SW4#show vlan | include ^123
123  VLAN0123                         active

123  enet  100123     1500  -      -      -        -    -        0      0
!
!SW1#show vlan | include ^123
SW1#
```

The following log messages will appear on the SW1 console, confirming that VTP messages are received from the VTP server and relayed on all other switches.

```
VTP LOG RUNTIME:Relaying packet received on trunk Fa0/23 - in TRANSPARENT MODE
 (nc = false)
VTP LOG RUNTIME: Relaying packet received on trunk Fa0/23 - in TRANSPARENT MODE (nc = false)
VTP LOG RUNTIME:Relaying packet received on trunk Fa0/19 - in TRANSPARENT MODE
 (nc = false)VTP LOG RUNTIME:Relaying packet received on trunk Fa0/21 - in TRANSPARENT MODE
 (nc = false)
VTP LOG RUNTIME: Relaying packet received on trunk Fa0/19 - in TRANSPARENT MODE (nc = false)
VTP LOG RUNTIME: Relaying packet received on trunk Fa0/21 - in TRANSPARENT MODE (nc = false)
VTP LOG RUNTIME:Relaying packet received on trunk Fa0/20 - in TRANSPARENT MODE
 (nc = false)VTP LOG RUNTIME:Relaying packet received on trunk Fa0/22 - in TRANSPARENT MODE
 (nc = false)
VTP LOG RUNTIME: Relaying packet received on trunk Fa0/20 - in TRANSPARENT MODE (nc = false)
VTP LOG RUNTIME: Relaying packet received on trunk Fa0/22 - in TRANSPARENT MODE (nc = false)
```

Note that when a switch is in VTP transparent mode, the VLAN configuration statements appear in the running-configuration. If the switch is in VTP client/server mode, the configured VLANs do not appear in the running-configuration; these are kept in the VLAN database file.

```
SW1# show running-config | i vlan

vlan internal allocation policy ascendingvlan 5,7-10
!
!SW2#show running-config | i vlan

vlan internal allocation policy ascending
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

VTP Pruning

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial VTP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- All switches are pre-configure in VTP domain **CCIE**.
  - Ensure that SW1 is in VTP client mode.
- Enable VTP pruning in the Layer 2 network so that inter-switch broadcast replication is minimized.
- Verify that this configuration is functional through the `show interface trunk` output.

Configuration

```
SW1:
vtp mode client
SW2:

vtp pruning
```

Verification

VTP pruning eliminates the need to statically remove VLANs from the allowed trunking list of a port by having the switches automatically communicate to each other which VLANs they have locally assigned or are in the transit path for.

The `show interface pruning` command indicates what traffic the local switch told its

https://t.me/learningnets

neighbor that it needs, via the **VLAN traffic requested of neighbor** field. These VLANs are either locally assigned to certain ports, or those for which the local switch is in the Layer 2 transit path and traffic was requested by neighbor switches. The **Vlans pruned for lack of request by neighbor** field indicates the VLANs that the upstream neighbor did not request. VTP pruning can be enabled only on the device running in server mode, and the settings will be inherited by all devices in the same VTP domain.

In the below output, this means that SW1 is not forwarding VLAN 7 to SW3, because SW3 did not request it. This output can be confusing because what SW1 sees as pruned for lack of request is the opposite of what SW3 sees as requested.

```
SW1#show interface fastethernet0/19 pruning

Port                Vlans pruned for lack of request by neighbor Fa0/19 5,7-10,22,43,58,67,79,123,146

Port                Vlan traffic requested of neighbor Fa0/19 1,5,7-10,22,43,58,67,79,123,146
!
!SW3#show interface fastethernet0/19 pruning

Port                Vlans pruned for lack of request by neighbor Fa0/19 none

Port                Vlan traffic requested of neighbor Fa0/19 none
```

If the network is converged, all devices in the VTP domain should agree that pruning is enabled, as shown in the below `show vtp status` output. Note that transparent switches cannot participate in pruning because they do not read the payload of the VTP updates they are receiving from their adjacent neighbors, they just relay it.

```
SW1#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1
VTP Domain Name                 : CCIE VTP Pruning Mode                  : Enabled
VTP Traps Generation            : Disabled
Device ID                       : 000a.b832.3580
Configuration last modified by 0.0.0.0 at 3-1-93 05:42:56

Feature VLAN:
--------------
VTP Operating Mode                : Client
Maximum VLANs supported locally   : 1005
Number of existing VLANs          : 17
Configuration Revision            : 3
MD5 digest                        : 0xC0 0x28 0xD7 0xD0 0x3D 0xA3 0x1D 0xB7
```

https://t.me/learningnets

```
                                    0x13 0xC9 0xD1 0xE6 0x57 0xD0 0x09 0x58
!
!SW2#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1
VTP Domain Name                 : CCIE VTP Pruning Mode                 : Enabled
VTP Traps Generation            : Disabled
Device ID                       : 001c.576d.4a00
Configuration last modified by 0.0.0.0 at 3-1-93 05:42:56
Local updater ID is 0.0.0.0 (no valid interface found)

Feature VLAN:
--------------
VTP Operating Mode                : Server
Maximum VLANs supported locally   : 1005
Number of existing VLANs          : 17
Configuration Revision            : 3
MD5 digest                        : 0xC0 0x28 0xD7 0xD0 0x3D 0xA3 0x1D 0xB7
                                    0x13 0xC9 0xD1 0xE6 0x57 0xD0 0x09 0x58
!
!SW3#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1
VTP Domain Name                 : CCIE VTP Pruning Mode                 : Enabled
VTP Traps Generation            : Disabled
Device ID                       : 001d.45cc.0580
Configuration last modified by 0.0.0.0 at 3-1-93 05:42:56

Feature VLAN:
--------------
VTP Operating Mode                : Client
Maximum VLANs supported locally   : 1005
Number of existing VLANs          : 17
Configuration Revision            : 3
MD5 digest                        : 0xC0 0x28 0xD7 0xD0 0x3D 0xA3 0x1D 0xB7
                                    0x13 0xC9 0xD1 0xE6 0x57 0xD0 0x09 0x58
!
!SW4#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1
VTP Domain Name                 : CCIE VTP Pruning Mode                 : Enabled

VTP Traps Generation            : Disabled
Device ID                       : 001c.576d.3d00
Configuration last modified by 0.0.0.0 at 3-1-93 05:42:56
```

https://t.me/learningnets

```
Feature VLAN:
--------------
VTP Operating Mode                : Client
Maximum VLANs supported locally   : 1005
Number of existing VLANs          : 17
Configuration Revision            : 3
MD5 digest                        : 0xC0 0x28 0xD7 0xD0 0x3D 0xA3 0x1D 0xB7
                                    0x13 0xC9 0xD1 0xE6 0x57 0xD0 0x09 0x58
```

To quickly view the traffic that is not being pruned, and therefore actually forwarded, issue the `show interface trunk` command. The final field of **Vlans in spanning tree forwarding state and not pruned** means that the VLAN is created, is allowed on the link, is running STP, and is not pruned.

```
SW1#show interface trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not pruned
Fa0/19      1
Fa0/20      1
Fa0/21      1
Fa0/22      1 Fa0/23       1,5,7-10,22,43,58,67,79,123,146
Fa0/24      1,5,7-10,22,43,58,67,79,123,146
!
!SW2#show interface trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not pruned
Fa0/23      1,5,7-10,22,43,58,67,79,123,146
Fa0/24      none
!
!SW3#show interface trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not pruned
Fa0/19      1,5,7-10,22,43,58,67,79,123,146
Fa0/20      none
!
!SW4#show interface trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not pruned
Fa0/21      1,5,7-10,22,43,58,67,79,123,146

Fa0/22      none
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

VTP Prune-Eligible List

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial VTP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- All switches are pre-configure in VTP domain **CCIE**.
- Enable VTP Pruning in the VTP domain.
- Edit the prune-eligible list to ensure that traffic for VLAN 7 is carried on all active trunk links in the Layer 2 network.
- Verify that this configuration is functional through the `show interface trunk` output.

Configuration

https://t.me/learningnets

```
SW1:

interface range FastEthernet0/19 - 24
 switchport trunk pruning vlan 2-6,8-1001
SW2:

vtp pruning
!
interface range FastEthernet0/23 - 24
 switchport trunk pruning vlan 2-6,8-1001
SW3:

interface range FastEthernet0/19 - 20
 switchport trunk pruning vlan 2-6,8-1001
SW4:

interface range FastEthernet0/21 - 22
 switchport trunk pruning vlan 2-6,8-1001
```

Verification

The implementation of the prune eligible list, which is controlled by the **`switchport trunk pruning vlan`** command, is commonly confusing because it is essentially the opposite of editing the allowed list of the trunk. By default, all VLANs 2–1001 (not the default or extended VLANs) can be pruned off on a trunk link.

This means that if the switch does not have VLAN 7 assigned to any ports and is not in the STP transit path for VLAN 7, it can tell its adjacent switches not to send VLAN 7 traffic. However, if VLAN 7 is removed from the prune eligible list, the switch must report that it does need VLAN 7, and the traffic cannot be pruned.

This can be seen in the change of the output below, where SW1 sends VLAN 7 traffic over all links that are forwarding for STP, even though the devices on the other end of the link do not actually need VLAN 7. Note that output may differ based on the STP root bridge, and therefore which ports are in STP FW state and which are blocking.

```
SW1#show interfaces trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not prunedFa0/19      1,7
Fa0/20      1,7
Fa0/21      1,7
Fa0/22      1,7
```

https://t.me/learningnets

```
Fa0/23      1,5,7
-10,22,43,58,67,79,146 Fa0/24 none
!
!SW2#show interface trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not pruned Fa0/23      1,5,7
-10,22,43,58,67,79,146 Fa0/24      1,5,7
-10,22,43,58,67,79,146
!
!SW3#show interface trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not pruned Fa0/19      5,7
-10,22,43,58,67,79,146 Fa0/20 none
!
!SW4#show interface trunk | begin pruned
Port        Vlans in spanning tree forwarding state and not pruned Fa0/21      5,7
-10,22,43,58,67,79,146 Fa0/22 none
```

SW1's FastEtherhet0/24 displays "none", as SW3's FastEthernet0/20 and SW4's FastEthernet0/22, because it is in the blocking state for all VLANs.

```
SW1#show spanning-tree interface fastEthernet0/24

Vlan                Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
VLAN0001            Altn BLK
 19        128.26   P2p  VLAN0005            Altn BLK
 19        128.26   P2p  VLAN0007            Altn BLK
 19        128.26   P2p  VLAN0008            Altn BLK
 19        128.26   P2p  VLAN0009            Altn BLK
 19        128.26   P2p  VLAN0010            Altn BLK
 19        128.26   P2p  VLAN0022            Altn BLK
 19        128.26   P2p  VLAN0043            Altn BLK
 19        128.26   P2p  VLAN0058            Altn BLK
 19        128.26   P2p  VLAN0067            Altn BLK
 19        128.26   P2p  VLAN0079            Altn BLK
 19        128.26   P2p  VLAN0146            Altn BLK
 19        128.26   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Layer 2 EtherChannel

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Layer 2 EtherChannels between SW1 and all other switches as follows:
  - Do not use any negotiation protocols.
  - SW1 should initiate 802.1q trunking negotiation.
  - Use port-channel numbers in the format of **1Y**, where *Y* is the switch number for SW2, SW3, and SW4.

Configuration

```
SW1:
interface range FastEthernet0/19 - 20
 channel-group 13 mode on
!
interface range FastEthernet0/21 - 22
 channel-group 14 mode on
!
interface range FastEthernet0/23 - 24
 channel-group 12 mode on
!
interface Port-channel12
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
!
interface Port-channel13
```

https://t.me/learningnets

```
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
!
interface Port-channel14
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
SW2:
interface range FastEthernet0/23 - 24
 channel-group 12 mode on
SW3:
interface range FastEthernet0/19 - 20
 channel-group 13 mode on
SW4:

interface range FastEthernet0/21 - 22
 channel-group 14 mode on
```

Verification

For an EtherChannel to form, all member interfaces must have the same configuration, and both ends of the channel must agree on the same negotiation protocol; in this case there is no negotiation used between the switches forming the EtherChannel. In the below `show etherchannel summary` output, the **Protocol** field is null, which means that no negotiation was used. This comes from the on mode of the `channel-group` command. This output also shows that the port-channel is in the **SU** state, which means that it is Layer 2 and up; member ports are in the **P** state, which means that interfaces have successfully joined the EtherChannel.

```
SW1#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port



Number of channel-groups in use: 3
Number of aggregators:           3
```

https://t.me/learningnets

```
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
12     Po12(SU)          -        Fa0/23(P)   Fa0/24(P)
   13     Po13(SU)          -        Fa0/19(P)   Fa0/20(P)
   14     Po14(SU)          -        Fa0/21(P)   Fa0/22(P)

!
!SW1#show etherchannel protocol
                Channel-group listing:
                ----------------------

Group: 12
----------Protocol:   -   (Mode ON)

Group: 13
----------Protocol:   -   (Mode ON)

Group: 14
----------Protocol:   -   (Mode ON)
```

As interfaces have been bundled into EtherChannels, the switch will show the logical interface as being trunk, not the physical interface; also, from STP perspective, STP will run over the logical interface as well.

```
SW1#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Po12        desirable        802.1q         trunking      1
Po13        desirable        802.1q         trunking      1
Po14        desirable        802.1q         trunking      1

Port        Vlans allowed on trunk
Po12        1-4094
Po13        1-4094
Po14        1-4094
<output omitted>
!
!SW1#show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0013.605f.f000
             This bridge is the root
```

https://t.me/learningnets

```
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/1               Desg FWD 19        128.3    P2p Po12
                Desg FWD 12        128.152  P2p Po13
                Desg FWD 12        128.160  P2p Po14
                Desg FWD 12        128.168  P2p
```

Verify that physical interfaces have inherited the configuration from the logical interface, which is the port-channel.

```
SW1#show interfaces fastEthernet0/19 switchport
Name: Fa0/19
Switchport: EnabledAdministrative Mode: dynamic desirable
Operational Mode: trunk (member of bundle Po13)
Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
```

https://t.me/learningnets

```
Unknown multicast blocked: disabled
Appliance trust: none
```

Another way to verify that a Layer 2 channel is working correctly is to view the spanning-tree topology. If STP runs over the logical port-channel interface instead of the physical interfaces, channeling has occurred properly. This is because without channeling, some member interfaces would be in the STP forwarding state and some blocking, but with channeling all interfaces have the same STP state and role as displayed by the logical port-channel interface. Note that STP port states and roles may differ from the output below based on which switch is the STP root bridge.

```
SW2#show spanning-tree vlan 1



VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     000a.b832.3a80
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p
Fa0/21              Desg FWD 19        128.23   P2p
Fa0/22              Desg FWD 19        128.24   P2p Po12                Desg FWD 12        128.152  P2p
!
!SW3#show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     000a.b832.3a80
             Cost        19
             Port        23 (FastEthernet0/21)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0022.5627.1f80
```

https://t.me/learningnets

```
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/1               Desg FWD 19        128.3    P2p
Fa0/21              Root FWD 19        128.23   P2p
Fa0/22              Altn BLK 19        128.24   P2p
Fa0/23              Altn BLK 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p Po13                Altn BLK 12        128.160  P2p

!
!SW4#show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     000a.b832.3a80
             Cost        19
             Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/1               Desg FWD 19        128.3    P2p
Fa0/19              Root FWD 19        128.21   P2p
Fa0/20              Altn BLK 19        128.22   P2p
Fa0/23              Desg FWD 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p Po14                Altn BLK 12        128.168  P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Layer 2 EtherChannel with PAgP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Layer 2 EtherChannels between SW1 and all other switches as follows:
  - Use Cisco's proprietary protocol for negotiation.
  - Only SW1 should actively initiate the EtherChannel negotiation.
  - SW1 should initiate 802.1q trunking negotiation.
  - Use port-channel numbers in the format of **1Y**, where *Y* is the switch number for SW2, SW3, and SW4.

Configuration

SW1:

```
interface range FastEthernet0/19 - 20
 channel-group 13 mode desirable
!
interface range FastEthernet0/21 - 22
 channel-group 14 mode desirable
!
interface range FastEthernet0/23 - 24
 channel-group 12 mode desirable
!
interface Port-channel12
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
!
```

https://t.me/learningnets

```
interface Port-channel13
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
!
interface Port-channel14
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
SW2:
interface range FastEthernet0/23 - 24
 channel-group 12 mode auto
SW3:
interface range FastEthernet0/19 - 20
 channel-group 13 mode auto
SW4:

interface range FastEthernet0/21 - 22
 channel-group 14 mode auto
```

Verification

Port Aggregation Protocol (PAgP) is a Cisco proprietary negotiation protocol for EtherChannel links. The desirable mode of PAgP, like DTP, is used to initiate negotiation, whereas the auto mode is used to listen for negotiation. This implies that one side running desirable with the other side running desirable or auto will result in a channel being formed, but both sides running auto will not. Verify that EtherChannel has been successfully negotiated and the protocol used is PAgP.

```
SW1#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port



Number of channel-groups in use: 3
Number of aggregators:           3

Group  Port-channel  Protocol    Ports
```

https://t.me/learningnets

```
------+-------------+-----------+-----------------------------------------------
12     Po12(SU)         PAgP       Fa0/23(P)   Fa0/24(P)
13     Po13(SU)         PAgP       Fa0/19(P)   Fa0/20(P)
14     Po14(SU)         PAgP       Fa0/21(P)   Fa0/22(P)

!
!SW1#show etherchannel protocol
                Channel-group listing:
                ----------------------

Group: 12
----------Protocol:  PAgP

Group: 13
----------Protocol:  PAgP

Group: 14
----------Protocol:  PAgP
```

Verify that SW1 is configured for desirable mode, and all other switches for auto mode.

```
SW1#show etherchannel 12 port-channel
                Port-channels in the group:
                ---------------------------

Port-channel: Po12
------------

Age of the Port-channel   = 0d:03h:01m:55s
Logical slot/port   = 2/12          Number of ports = 2
GC                  = 0x000C0001      HotStandBy port = null
Port state          = Port-channel Ag-Inuse
Protocol            =   PAgP
Port security       = Disabled

Ports in the Port-channel:

Index   Load   Port     EC state        No of bits
------+------+------+------------------+-----------  0     00 Fa0/23   Desirable-Sl
       0  0     00 Fa0/24   Desirable-Sl
       0

Time since last port bundled:    0d:03h:01m:46s    Fa0/24
```

https://t.me/learningnets

```
!
!SW2#show etherchannel 12 port-channel
                Port-channels in the group:
                ---------------------------

Port-channel: Po12
------------

Age of the Port-channel   = 0d:03h:02m:02s
Logical slot/port   = 2/12          Number of ports = 2
GC                  = 0x000C0001      HotStandBy port = null
Port state          = Port-channel Ag-Inuse
Protocol            =   PAgP
Port security       = Disabled

Ports in the Port-channel:

Index   Load   Port     EC state        No of bits
------+------+------+------------------+-----------  0     00Fa0/23   Automatic-Sl
       0  0     00Fa0/24   Automatic-Sl
       0

Time since last port bundled:    0d:03h:02m:00s    Fa0/24
```

Verify that trunking has been negotiated and logical port-channel interfaces are used by STP.

```
SW1#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Po12        desirable        802.1q         trunking      1
Po13        desirable        802.1q         trunking      1
Po14        desirable        802.1q         trunking      1

Port        Vlans allowed on trunk
Po12        1-4094
Po13        1-4094
Po14        1-4094
<output omitted>
!
!SW1#show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
```

https://t.me/learningnets

```
             Address     000a.b832.3a80
             Cost        12
             Port        152 (Port-channel12)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/1               Desg FWD 19        128.3    P2p Po12
                Root FWD 12        128.152  P2p Po13
                Desg FWD 12        128.160  P2p Po14
                Desg FWD 12        128.168  P2p
```

Verify the EtherChannel state on all switches. However, because switches use a negotiation protocol, if EtherChannel shows as functional on one side, it has to be functional on the other side as well. With no negotiation protocol being used, EtherChannel may not actually be functional from the perspective of both switches, and you must verify that.

```
SW2#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
12     Po12(SU)         PAgP       Fa0/23(P)   Fa0/24(P)
!
!SW3#show etherchannel summary
```

https://t.me/learningnets

```
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
13     Po13(SU)         PAgP      Fa0/19(P)   Fa0/20(P)
!
!SW4#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
14     Po14(SU)         PAgP      Fa0/21(P)   Fa0/22(P)
```

Verify that physical interfaces have inherited the configuration from the logical interface—for example, FastEthernet0/19 of SW1.

```
SW1#show interfaces fastEthernet0/19 switchport
```

https://t.me/learningnets

```
Name: Fa0/19
Switchport: EnabledAdministrative Mode: dynamic desirable
Operational Mode: trunk (member of bundle Po13)
Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
```

If configuration is done on one side of the EtherChannel only, for example, only on SW1, because PAgP negotiation will fail, the following messages will be logged:

```
%EC-5-L3DONTBNDL1: Fa0/19 suspended: PAgP not enabled on the remote port.
%EC-5-L3DONTBNDL1: Fa0/20 suspended: PAgP not enabled on the remote port.
```

To avoid any problems in the network, physical interfaces will be logically disabled at Layer 2 and will appear in the suspended state for the EtherChannel.

```
SW1#show etherchannel summary
Flags:  D - down        P - bundled in port-channel        I - stand-alones - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
```

https://t.me/learningnets

```
        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
13     Po13(SU)        PAgP      Fa0/19(s)   Fa0/20(s)
!
!SW1#show ip interface brief | i 19|20
FastEthernet0/19       unassigned      YES manual up down
FastEthernet0/20       unassigned      YES manual up down
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Layer 2 EtherChannel with LACP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Layer 2 EtherChannels between SW1 and all other switches as follows:
  - Use industry standard protocol for negotiation.
  - Only SW1 should actively initiate the EtherChannel negotiation.
  - SW1 should initiate 802.1q trunking negotiation.
  - Use port-channel numbers in the format of **1Y**, where *Y* is the switch number for SW2, SW3, and SW4.

Configuration

SW1:

```
interface range FastEthernet0/19 - 20
 channel-group 13 mode active
!
interface range FastEthernet0/21 - 22
 channel-group 14 mode active
!
interface range FastEthernet0/23 - 24
 channel-group 12 mode active
!
interface Port-channel12
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
!
```

https://t.me/learningnets

```
interface Port-channel13
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
!
interface Port-channel14
 switchport trunk encapsulation dot1q
 switchport mode dynamic desirable
SW2:
interface range FastEthernet0/23 - 24
 channel-group 12 mode passive
SW3:
interface range FastEthernet0/19 - 20
 channel-group 13 mode passive
SW4:

interface range FastEthernet0/21 - 22
 channel-group 14 mode passive
```

Verification

Similar to the previous variation of EtherChannel, Link Aggregation Control Protocol (LACP) is used to negotiate the formation of the channels from SW1 to SW2, SW3, and SW4. LACP is an open standard defined in IEEE 802.3ad. The active mode of LACP, like the desirable mode of PAgP, is used to initiate LACP negotiation, whereas the passive mode is used only to respond to negotiation. Like PAgP, this implies that a channel will form via LACP if one side is active and the other side is active or passive, but a channel will not form if both sides are passive. Verify that EtherChannel has been successfully negotiated and the protocol used is LACP.

```
SW1#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port



Number of channel-groups in use: 3
Number of aggregators:           3
```

https://t.me/learningnets

```
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
12     Po12(SU)         LACP      Fa0/23(P)   Fa0/24(P)
13     Po13(SU)         LACP      Fa0/19(P)   Fa0/20(P)
14     Po14(SU)         LACP      Fa0/21(P)   Fa0/22(P)
!
!SW1#show etherchannel protocol
                Channel-group listing:
                ----------------------

Group: 12
----------Protocol:  LACP

Group: 13
----------Protocol:  LACP

Group: 14
----------Protocol:  LACP
```

Verify that SW1 is configured for active mode, and all other switches for passive mode.

```
SW1#show etherchannel 12 port-channel
                Port-channels in the group:
                ---------------------------

Port-channel: Po12    (Primary Aggregator)

------------

Age of the Port-channel   = 0d:00h:03m:50s
Logical slot/port   = 2/12          Number of ports = 2
HotStandBy port = null
Port state          = Port-channel Ag-Inuse
Protocol            =   LACP
Port security       = Disabled

Ports in the Port-channel:

Index   Load   Port     EC state        No of bits
------+------+------+------------------+-----------  0     00Fa0/23   Active
            0  0     00Fa0/24   Active
            0
```

https://t.me/learningnets

```
Time since last port bundled:    0d:00h:03m:35s    Fa0/24
!
!SW2#show etherchannel 12 port-channel
                Port-channels in the group:
                ---------------------------

Port-channel: Po12    (Primary Aggregator)

------------

Age of the Port-channel   = 0d:00h:03m:46s
Logical slot/port   = 2/12          Number of ports = 2
HotStandBy port = null
Port state          = Port-channel Ag-Inuse
Protocol            =   LACP
Port security       = Disabled

Ports in the Port-channel:

Index   Load   Port     EC state        No of bits
------+------+------+------------------+-----------  0     00Fa0/23   Passive
            0  0     00Fa0/24   Passive
            0

Time since last port bundled:    0d:00h:03m:38s    Fa0/24
```

Verify that trunking has been negotiated and logical port-channel interfaces are used by STP.

```
SW1#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Po12        desirable        802.1q         trunking      1
Po13        desirable        802.1q         trunking      1
Po14        desirable        802.1q         trunking      1

Port        Vlans allowed on trunk
Po12        1-4094
Po13        1-4094
Po14        1-4094
<output omitted>
!
!SW1#show spanning-tree vlan 1
```

https://t.me/learningnets

```
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     000a.b832.3a80
             Cost        12
             Port        152 (Port-channel12)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/1               Desg FWD 19        128.3    P2p Po12
                Root FWD 12        128.152  P2p Po13
                Desg FWD 12        128.160  P2p Po14
                Desg FWD 12        128.168  P2p
```

Verify the EtherChannel state on all switches. However, because switches use a negotiation protocol, if EtherChannel shows as functional on one side, it has to be functional on the other side as well. With no negotiation protocol being used, EtherChannel may not actually be functional from the perspectives of both switches, and you must verify that.

```
SW2#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
12     Po12(SU)         LACP      Fa0/23(P)   Fa0/24(P)
```

https://t.me/learningnets

```
!
!SW3#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol
------+-------------+-----------+-----------------------------------------------
13     Po13(SU)        LACP      Fa0/19(P)   Fa0/20(P)
!
!SW4#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
14     Po14(SU)        LACP      Fa0/21(P)   Fa0/22(P)
```

https://t.me/learningnets

Verify that physical interfaces have inherited the configuration from the logical interface—for example, FastEthernet0/19 of SW1.

```
SW1#show interfaces fastEthernet0/19 switchport
Name: Fa0/19
Switchport: EnabledAdministrative Mode: dynamic desirable
Operational Mode: trunk (member of bundle Po13)
Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
```

If configuration is done on one side of the EtherChannel only, for example only on SW1, because LACP negotiation will fail, the following messages will be logged.

```
%EC-5-L3DONTBNDL2: Fa0/19 suspended: LACP currently not enabled on the remote port.
%EC-5-L3DONTBNDL2: Fa0/20 suspended: LACP currently not enabled on the remote port.
```

To avoid any problems in the network, physical interfaces will be logically disabled at Layer 2 and will appear in the suspended state for the EtherChannel.

https://t.me/learningnets

```
SW1#show etherchannel summary

Flags:  D - down        P - bundled in port-channel        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
13     Po13(SU)        LACP      Fa0/19(s)   Fa0/20(s)
!
!SW1#show ip interface brief | i 19|20
FastEthernet0/19       unassigned      YES manual up down
FastEthernet0/20       unassigned      YES manual up down
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Layer 3 EtherChannel

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Layer 3 EtherChannels between SW2 and SW4 as follows:
  - Use both directly connected Ethernet links.
  - Use port-channel number 24 and the subnet 155.1.108.Y/24, where *Y* is the switch number.
  - Use an industry standard protocol for negotiation.
  - Both switches should actively initiate the EtherChannel negotiation.
- Ensure that IPv4 reachability is obtained between SW2 and SW4.

Configuration

```
SW2:
interface Port-channel24
 no switchport
 ip address 155.1.108.2 255.255.255.0
!
interface range FastEthernet0/19 - 20
 no switchport
 channel-group 24 mode active

SW4:

interface Port-channel24
```

https://t.me/learningnets

```
 no switchport
 ip address 155.1.108.4 255.255.255.0
!
interface range FastEthernet0/19 - 20
 no switchport
 channel-group 24 mode active
```

Verification

Pitfall

One common problem with forming Layer 3 EtherChannel links is the order of operations. The important point to remember is that when the **`channel-group`** command is issued, the attributes of the member interfaces are immediately inherited by the port-channel interface. This means that if the **`channel-group`** command is issued before the **`no switchport`** command on the physical interfaces, the logical port-channel interface will be created as the default of Layer 2, and this cannot be changed afterward. A subsequent attempt to issue the **`channel-group`** command will generate an error message saying that the channel interface and the members are not compatible. To resolve this problem, simply issue the **`no switchport`** command before the **`channel-group`** command. If configured properly, the state of the port-channel from the **`show etherchannel summary`** command should show **RU** for routed and in use.

```
SW2#show etherchannel 24 summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 2
Number of aggregators:           2

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
24     Po24(RU)        LACP      Fa0/19(P)   Fa0/20(P)
```

https://t.me/learningnets

```
!
!SW4#show etherchannel 24 summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port



Number of channel-groups in use: 2
Number of aggregators:           2

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
24     Po24(RU)        LACP      Fa0/19(P)   Fa0/20(P)
```

Verify IPv4 connectivity between SW2 and SW4.

```
SW4#ping 155.1.108.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.108.2, timeout is 2 seconds:.!!!!

Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
```

The port-channel interface should appear as a normal Layer 3 routed interface in the IPv4 routing table.

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW2(config)#ip routing
!SW2#show ip route connected
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override
```

https://t.me/learningnets

```
Gateway of last resort is not set

      155.1.0.0/16 is variably subnetted, 2 subnets, 2 masks
C        155.1.108.0/24 is directly connected, Port-channel24

L        155.1.108.2/32 is directly connected, Port-channel24
```

Note that a Layer 3 interface in a switch is just a "one-legged" VLAN. SW4 is using VLAN 1006 internally for this Layer 3 port-channel.

```
SW2#show vlan internal usage



VLAN Usage
---- --------------------1006 Port-channel24
```

This is not just for port-channels but also for any Layer 2 port on the switch that is converted to a Layer 3 port by using the **`no switchport`** command. The internal VLAN cannot be used for anything else. The policy that controls how the VLAN numbers are assigned is managed by the following command: **`vlan internal allocation policy ascending`** .

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW2(config)#vlan 1006
SW2(config-vlan)#exit
!
% Failed to create VLANs 1006
VLAN(s) not available in Port Manager.
%Failed to commit extended VLAN(s) changes.
%PM-4-EXT_VLAN_INUSE: VLAN 1006 currently in use by Port-channel24
%SW_VLAN-4-VLAN_CREATE_FAIL: Failed to create VLANs 1006: VLAN(s) not available in Port Manager
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP Root Bridge Election

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure the inter-switch links between SW1 and all other switches as static 802.1q trunk links.
  - Disable all other inter-switch links.
- Configure VLAN 2 on all switches and:
  - Configure SW4 as the STP Root Bridge for VLAN 2 using the lowest possible priority.
  - If SW4 goes down, SW1 should take over the STP Root Bridge role for VLAN 2 using the second-lowest possible priority.

Configuration

```
SW1:
vlan 2
spanning-tree vlan 2 priority 4096
!
interface range FastEthernet0/19 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
SW2:
vlan 2
interface range FastEthernet0/19 - 22
 shutdown
```

https://t.me/learningnets

```
!
interface range FastEthernet0/23 - 24
 switchport trunk encapsulation dot1q
 switchport mode trunk
SW3:
vlan 2
!
interface range FastEthernet0/19 - 20
 switchport trunk encapsulation dot1q
 switchport mode trunk
!
interface range FastEthernet0/21 - 24
 shutdown
SW4:

vlan 2
spanning-tree vlan 2 priority 0
!
interface range FastEthernet0/19 - 20
 shutdown
!
interface range FastEthernet0/21 - 22
 switchport trunk encapsulation dot1q
 switchport mode trunk
!
interface range FastEthernet0/23 - 24
 shutdown
```

Verification

STP root bridge election is based on the priority and MAC address fields of the Bridge ID. The device with the lowest priority value is elected the root. If there is a tie in priority, the device with the lowest MAC address is elected root. SW4 with the local priority of two, the configured priority of zero, plus the system id extension (VLAN number), shows that **This bridge is the root**. The root bridge should show the same priority and MAC address for both the Root ID and the Bridge ID, and list all interfaces as Designated (downstream facing). In this case, SW4's BID is **001a.a174.2500**.

```
SW4#show spanning-tree vlan 2


VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    2
```

https://t.me/learningnets

```
             Address     001a.a174.2500 This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    2      (priority 0 sys-id-ext 2)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- -------------------------------Fa0/21 Desg FWD
 19        128.23   P2p  Fa0/22 Desg FWD
 19        128.24   P2p
```

Verify that all other switches agree that SW4 is the root bridge for VLAN 2 and have selected their Root Port, which is the physical path with the lowest STP cost to reach the root; because both SW1's FastEthernet0/21 and FastEthernet0/22 have the same cost of 19, the tie breaker is the lowest port-priority of the upstream neighbor, which is SW4.

```
SW1#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee Root ID    Priority    2
 Address     001a.a174.2500
 Cost        19
 Port        23 (FastEthernet0/21)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4098   (priority 4096 sys-id-ext 2)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- -------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/21              Root FWD 19        128.23   P2p
Fa0/22              Altn BLK 19        128.24   P2p
Fa0/23              Desg FWD 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p
!
!SW1#show spanning-tree vlan 2 root detail
VLAN0002
  Root ID    Priority    2
             Address     001a.a174.2500
```

https://t.me/learningnets

```
              Cost        19 Port         23 (FastEthernet0/21)
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
!
!SW4#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    2
             Address     001a.a174.2500
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    2      (priority 0 sys-id-ext 2)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/21              Desg FWD 19 128.23
   P2p  Fa0/22              Desg FWD 19 128.24
   P2p
```

SW2 agrees that the device with the BID **001a.a174.2500** is the root bridge and uses its Fa0/23 port with a total cost of 38 to reach it. SW2's local BID is a priority of 32770, the default of 32768 plus the system id extension 2, and the MAC address **000a.b832.3a80**.

```
SW2#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    2
             Address     001a.a174.2500
             Cost        38
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/23              Root FWD 19        128.25   P2p
```

https://t.me/learningnets

```
Fa0/24              Altn BLK 19        128.26   P2p
```

SW3 agrees that the device with the BID **001a.a174.2500** is the root bridge and uses its Fa0/19 port with a total cost of 38 to reach it. SW3's local BID is a priority of 32770, the default of 32768 plus the system id extension 2, and the MAC address **0022.5627.1f80**.

```
SW3#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    2
             Address     001a.a174.2500
             Cost        38
             Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     0022.5627.1f80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD 19        128.21   P2p

Fa0/20              Altn BLK 19        128.22   P2p
```

If SW4's trunk links to SW1 are disabled, SW1 will become the root bridge because it has the next-best priority value of 4096.

```
SW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#interface range fastEthernet0/21 - 22
SW1(config-if-range)#shutdown
!
!
SW1#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieeeRoot ID    Priority    4098
Address     0013.605f.f000
This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

https://t.me/learningnets

```
  Bridge ID  Priority    4098   (priority 4096 sys-id-ext 2)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------Fa0/19Desg FWD
 19        128.21   P2p  Fa0/20Desg FWD
 19        128.22   P2p  Fa0/23Desg FWD
 19        128.25   P2p  Fa0/24Desg FWD
 19        128.26   P2p
```

Verify that both SW2 and SW3 agree that SW1 is the new current root bridge for VLAN 2:

```
SW2#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieeeRoot ID    Priority    4098
 Address     0013.605f.f000
             Cost        19
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/23              Root FWD 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
!
!SW3#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieeeRoot ID    Priority    4098
 Address     0013.605f.f000
             Cost        19
             Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     0022.5627.1f80
```

https://t.me/learningnets

```
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD 19        128.21   P2p

Fa0/20              Altn BLK 19        128.22   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP Path Selection with Port Cost

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Using Spanning-Tree cost, ensure that traffic for all active VLANs from SW2 to SW1 uses Fa0/20 link between SW2 and SW4.
  - If this link goes down, traffic should use Fa0/19 link between SW2 and SW4.

Configuration

```
SW2:

interface range FastEthernet0/23 - 24
 spanning-tree cost 1000
!
interface FastEthernet0/19
 spanning-tree cost 2
!
interface FastEthernet0/20
 spanning-tree cost 1
```

Verification

https://t.me/learningnets

Check the STP path for one VLAN on SW2 before configuration changes, and note that Root Port is Fa0/23, which directly connects SW2 to SW1.

```
SW2#show spanning-tree vlan 10


VLAN0010
  Spanning tree enabled protocol ieee Root ID    Priority    4106
             Address     0013.605f.f000 Cost        19
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    16394  (priority 16384 sys-id-ext 10)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/23              Root FWD 19        128.25   P2p


Fa0/24              Altn BLK 19        128.26   P2p
```

Note that the path from SW2 to SW1 through SW4 and SW3 is blocked at the SW4 level.

https://t.me/learningnets

```
SW4#show spanning-tree vlan 10


VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        38
             Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8202   (priority 8192 sys-id-ext 10)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD 19        128.21   P2p
Fa0/20              Altn BLK 19        128.22   P2p Fa0/23              Altn BLK 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

The STP cost to the root bridge from SW2 before configuration changes is 19. Changing the links to SW1 to a cost of 1000 makes them the least-preferred path. By changing the last link to SW4 to a cost of 1, the end-to-end path cost on that link becomes 39, which is the most preferred (1 to SW4, 19 from SW4 to SW3, 19 from SW3 to SW1). With the second-to-last link having a cost of 2, the end-to-end path cost will be 40 and will therefore be the second-most preferred link:

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#interface range FastEthernet0/23 - 24
SW2(config-if-range)#spanning-tree cost 1000
!SW2(config-if-range)#interface FastEthernet0/19
SW2(config-if)#spanning-tree cost 2
!SW2(config-if)#interface FastEthernet0/20
SW2(config-if)#spanning-tree cost 1
!
!SW2#show spanning-tree vlan 10


VLAN0010
  Spanning tree enabled protocol ieee Root ID    Priority    4106
```

https://t.me/learningnets

```
             Address     0013.605f.f000 Cost         39
             Port        22 (FastEthernet0/20)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    16394  (priority 16384 sys-id-ext 10)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Altn BLK 2         128.21   P2p Fa0/20              Root FWD 1         128.22   P2p

Fa0/23              Altn BLK 1000      128.25   P2p
Fa0/24              Altn BLK 1000      128.26   P2p
```

As shown above, the direct path from SW2 to SW1 is in blocking state, and the path via SW4-->SW3-->SW1 is now the preferred path by STP.

```
SW4#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        38
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8202   (priority 8192 sys-id-ext 10)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/23              Root FWD 19        128.25   P2p

Fa0/24              Altn BLK 19        128.26   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP Path Selection with Port Priority

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Using Spanning-Tree priority, ensure that traffic for all active VLANs from SW4 to SW1 uses Fa0/20 link between SW2 and SW4.
  - If this link goes down, traffic should use Fa0/19 link between SW2 and SW4.

Configuration

```
SW2:

interface FastEthernet0/19
 spanning-tree port-priority 16
!
interface FastEthernet0/20
 spanning-tree port-priority 0
```

Verification

Check the STP path for one VLAN on SW4 before configuration changes, and note that Root Port is Fa0/19 between SW4 and SW2.

```
SW4#show spanning-tree vlan 10

VLAN0010
```

https://t.me/learningnets

```
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        38
             Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8202   (priority 8192 sys-id-ext 10)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD 19        128.21   P2p

Fa0/20              Altn BLK 19        128.22   P2p
Fa0/23              Altn BLK 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

Before configuration changes:

```
SW4#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    10 Address     000a.b832.3580
             Cost        38 Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     001c.576d.3d00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD 19        128.21   P2p

Fa0/20              Altn BLK 19        128.22   P2p
Fa0/23              Altn BLK 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
!
!SW4# show spanning-tree vlan 10 detail
```

https://t.me/learningnets

```
 VLAN0010 is executing the ieee compatible Spanning Tree protocol
  Bridge Identifier has priority 8192, sysid 10, address 001a.a174.2500
  Configured hello time 2, max age 20, forward delay 15
Current root has priority 4106, address 0013.605f.f000
 Root port is 21 (FastEthernet0/19), cost of root path is 38
  Topology change flag not set, detected flag not set
  Number of topology changes 7 last change occurred 00:07:38 ago
          from FastEthernet0/23
  Times:  hold 1, topology change 35, notification 2
          hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0, aging 300
 Port 21 (FastEthernet0/19) of VLAN0010 is root forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Designated root has priority 4106, address 0013.605f.f000
Designated bridge has priority 16394, address 000a.b832.3a80
 Designated port id is 128.21, designated path cost 19
   Timers: message age 2, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   BPDU: sent 1276, received 1585
 Port 22 (FastEthernet0/20) of VLAN0010 is alternate blocking
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 4106, address 0013.605f.f000
Designated bridge has priority 16394, address 000a.b832.3a80
 Designated port id is 128.22, designated path cost 19
   Timers: message age 2, forward delay 0, hold 0
   Number of transitions to forwarding state: 2
   Link type is point-to-point by default
   BPDU: sent 1274, received 1582
 Port 25 (FastEthernet0/23) of VLAN0010 is alternate blocking
   Port path cost 19, Port priority 128, Port Identifier 128.25.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 32778, address 0022.5627.1f80
   Designated port id is 128.25, designated path cost 19
   Timers: message age 3, forward delay 0, hold 0
   Number of transitions to forwarding state: 2
   Link type is point-to-point by default
   BPDU: sent 1173, received 1681
 Port 26 (FastEthernet0/24) of VLAN0010 is alternate blocking

   Port path cost 19, Port priority 128, Port Identifier 128.26.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 32778, address 0022.5627.1f80
   Designated port id is 128.26, designated path cost 19
   Timers: message age 2, forward delay 0, hold 0
```

https://t.me/learningnets

```
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  BPDU: sent 1167, received 1678
```

Because SW4 is connected to SW2 out both links and the STP cost is the same, there is also a tie in the designated bridge-id; the designated (upstream) port priority is compared. Because the upstream port priority of Fa0/19 is 128.21, versus 128.22 for Fa0/20, Fa0/19 is the Root Port on SW4. By changing the upstream priority on SW2 for ports Fa0/19 and Fa0/20, SW4 will prefer the path via Fa0/20. If interface Fa0/20 on SW4 goes down, it will fall back to Fa0/19.

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW2(config)#interface FastEthernet0/20
SW2(config-if)#spanning-tree port-priority 0
!SW2(config-if)#interface FastEthernet0/19
            SW2(config-if)#spanning-tree port-priority 16
!
!SW4#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106 Address     0013.605f.f000
             Cost        38 Port         22 (FastEthernet0/20)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8202   (priority 8192 sys-id-ext 10)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Altn BLK 19        128.21   P2p Fa0/20              Root FWD 19        128.22   P2p
Fa0/23              Altn BLK 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
!
!SW4#show spanning-tree vlan 10 detail

 VLAN0010 is executing the ieee compatible Spanning Tree protocol
  Bridge Identifier has priority 8192, sysid 10, address 001a.a174.2500
  Configured hello time 2, max age 20, forward delay 15
 Current root has priority 4106, address 0013.605f.f000
 Root port is 22 (FastEthernet0/20), cost of root path is 38
  Topology change flag not set, detected flag not set
  Number of topology changes 8 last change occurred 00:00:47 ago
```

https://t.me/learningnets

```
          from FastEthernet0/19
  Times:  hold 1, topology change 35, notification 2
          hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0, aging 300
 Port 21 (FastEthernet0/19) of VLAN0010 is alternate blocking
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 16394, address 000a.b832.3a80
 Designated port id is 16.21, designated path cost 19
   Timers: message age 3, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   BPDU: sent 1276, received 1877
 Port 22 (FastEthernet0/20) of VLAN0010 is root forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 16394, address 000a.b832.3a80
 Designated port id is 0.22, designated path cost 19
   Timers: message age 3, forward delay 0, hold 0
   Number of transitions to forwarding state: 3
   Link type is point-to-point by default
   BPDU: sent 1275, received 1874
 Port 25 (FastEthernet0/23) of VLAN0010 is alternate blocking
   Port path cost 19, Port priority 128, Port Identifier 128.25.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 32778, address 0022.5627.1f80
   Designated port id is 128.25, designated path cost 19
   Timers: message age 2, forward delay 0, hold 0
   Number of transitions to forwarding state: 2
   Link type is point-to-point by default
   BPDU: sent 1173, received 1973
 Port 26 (FastEthernet0/24) of VLAN0010 is alternate blocking

   Port path cost 19, Port priority 128, Port Identifier 128.26.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 32778, address 0022.5627.1f80
   Designated port id is 128.26, designated path cost 19
   Timers: message age 3, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   BPDU: sent 1167, received 1969
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Tuning STP Convergence Timers

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure the root bridge so that switches generate Spanning-Tree hello packets every 3 seconds.
- When a new port becomes active, it should wait 20 seconds before transitioning to the forwarding state.
- If the switches do not hear a configuration message within 10 seconds, they should attempt reconfiguration.
- This configuration should affect all currently active VLANs and any additional VLANs created in the future.

Configuration

```
SW1:

spanning-tree vlan 1-4094 hello-time 3
spanning-tree vlan 1-4094 forward-time 10
spanning-tree vlan 1-4094 max-age 10
```

Verification

https://t.me/learningnets

Verify the default STP timers before being changed.

```
SW1#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee Root ID    Priority    4106
             Address     0013.605f.f000
             This bridge is the root Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
 Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     0013.605f.f000 Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p
Fa0/23              Desg FWD 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p
!
!SW2#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        19
             Port        25 (FastEthernet0/23) Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
 Bridge ID  Priority    16394  (priority 16384 sys-id-ext 10)
             Address     000a.b832.3a80 Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p
Fa0/23              Root FWD 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

Modify STP timers, and verify that these have been learned from the STP root bridge.

```
SW1#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
```

https://t.me/learningnets

```
Root ID    Priority    4106

             Address     0013.605f.f000
             This bridge is the root Hello Time   3 sec  Max Age 10 sec  Forward Delay 10 sec
Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     0013.605f.f000 Hello Time   3 sec  Max Age 10 sec  Forward Delay 10 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p
Fa0/23              Desg FWD 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p
!
!SW2#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        19
             Port        25 (FastEthernet0/23) Hello Time   3 sec  Max Age 10 sec  Forward Delay 10 sec
Bridge ID  Priority    16394  (priority 16384 sys-id-ext 10)
             Address     000a.b832.3a80 Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p
Fa0/23              Root FWD 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

Downstream devices from the root bridge inherit the timers configured on the root. With a forward delay of 10 seconds configured on SW1, the downstream switches should take 10 seconds in each of the listening and learning phases during convergence. The below timestamps indicate that a new root port was elected at 23:36:59 on SW3, which transitions from blocking to listening. 10 seconds later, at 23:37:09, the port transitions from listening to learning. Finally, 10 seconds after that, at 23:37:19, the port transitions into forwarding.

```
SW3#debug spanning-tree events

Spanning Tree event debugging is onSW3#debug condition vlan 10
```

https://t.me/learningnets

```
Condition 1 set
!SW3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW3(config)#service timestamps debug
SW3(config)#logging console 7
SW3(config)#interface FastEthernet0/19
SW3(config-if)#shutdown
!
23:36:59: STP: VLAN0010 new root port Fa0/20, cost 19 23:36:59: STP: VLAN0010 Fa0/20 -> listening
23:36:59: STP[10]: Generating TC trap for port FastEthernet0/19
!
23:37:02: STP: VLAN0010 sent Topology Change Notice on Fa0/20 23:37:09: STP: VLAN0010 Fa0/20 -> learning
!
23:37:19: STP: VLAN0010 sent Topology Change Notice on Fa0/20
23:37:19: STP: VLAN0010 Fa0/20 -> forwarding
!
!SW3#undebug all
All possible debugging has been turned off
!SW3#undebug condition all

Removing all conditions may cause a flood of debugging
messages to result, unless specific debugging flags
are first removed.

Proceed with the removal of all conditions? [yes/no]: yes
1 conditions have been removed
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP PortFast

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Spanning-Tree PortFast on SW1 so that port FastEthernet0/1 does not have to wait for the Spanning-Tree listening and learning phases to begin forwarding.
  - Configure FastEthernet0/1 on SW1 as a trunk port using 802.1q encapsulation.
  - Do not use any global Spanning-Tree commands to accomplish this.

Configuration

```
SW1:

interface FastEthernet0/1
 switchport trunk encapsulation dot1q
 switchport mode trunk
 spanning-tree portfast trunk
```

Verification

Portfast is used to override the listening and learning phases of Spanning-Tree, also called the forwarding delay, and transition immediately to forwarding. Note that in this example, we are enabling PortFast on a trunk port, which requires the additional `trunk` keyword attached to the `spanning-tree portfast` command. This is useful

https://t.me/learningnets

when you have a trunk port connected to a server, such as a hypervisor, that is running multiple virtual machines and using a virtual switch with multiple VLANs. Although trunking can be used between switches, trunking can also be used to connect to single devices that can tag traffic with 802.1q. Another example would be a VOIP phone with a PC connected to it that tags voice traffic with one VLAN but data traffic from the PC with another VLAN; this was actually the voice implementation before the voice VLAN functionality, also known as the auxiliary VLAN, was added to the switch.

```
SW1#show spanning-tree interface fastethernet0/1 portfast
VLAN0001            enabled
VLAN0005            enabled
VLAN0007            enabled
VLAN0008            enabled
VLAN0009            enabled
VLAN0010            enabled
VLAN0022            enabled
VLAN0043            enabled
VLAN0058            enabled
VLAN0067            enabled
VLAN0079            enabled
VLAN0146            enabled
!
!SW1#show spanning-tree interface fastEthernet0/1 detail

 Port 3 (FastEthernet0/1) of VLAN0001 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 4097, address 0013.605f.f000
   Designated bridge has priority 4097, address 0013.605f.f000
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
The port is in the portfast mode by portfast trunk configuration
   Link type is point-to-point by default
   BPDU: sent 686, received 0

 Port 3 (FastEthernet0/1) of VLAN0005 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 4101, address 0013.605f.f000
   Designated bridge has priority 4101, address 0013.605f.f000
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
The port is in the portfast mode by portfast trunk configuration
```

https://t.me/learningnets

```
   Link type is point-to-point by default
   BPDU: sent 686, received 0
<output omitted>
!SW1#debug spanning-tree event
Spanning Tree event debugging is on
!SW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#service timestamps debug datetime msec
SW1(config)#logging console 7
SW1(config)#interface FastEthernet0/1
SW1(config-if)#shutdown

!
*Mar  3 05:15:45.999: STP: VLAN0001 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0005 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0007 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0008 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0009 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0010 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0022 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0043 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0058 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0067 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0079 we are the spanning tree root
*Mar  3 05:15:45.999: STP: VLAN0146 we are the spanning tree root
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to administratively down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down
```

When interface FastEthernet0/1 is shut down and subsequently brought back up, it immediately transitions to the forwarding state for all VLANs allowed on the trunk.

```
SW1(config-if)#no shutdown
!
*Mar  3 22:52:28.732: set portid: VLAN0005 Fa0/1: new port id 8003*Mar  3 22:52:28.732:
STP: VLAN0005 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.732: set portid: VLAN0007 Fa0/1: new port id 8003*Mar  3 22:52:28.732:
STP: VLAN0007 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0008 Fa0/1: new port id 8003*Mar  3 22:52:28.740:
STP: VLAN0008 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0009 Fa0/1: new port id 8003*Mar  3 22:52:28.740:
STP: VLAN0009 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0010 Fa0/1: new port id 8003*Mar  3 22:52:28.740:
STP: VLAN0010 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0022 Fa0/1: new port id 8003*Mar  3 22:52:28.740:
STP: VLAN0022 Fa0/1 ->jump to forwarding from blocking
```

https://t.me/learningnets

```
*Mar  3 22:52:28.740: set portid: VLAN0043 Fa0/1: new port id 8003 *Mar  3 22:52:28.740:
STP: VLAN0043 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0058 Fa0/1: new port id 8003 *Mar  3 22:52:28.740:
STP: VLAN0058 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0067 Fa0/1: new port id 8003 *Mar  3 22:52:28.740:
STP: VLAN0067 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0079 Fa0/1: new port id 8003 *Mar  3 22:52:28.740:
STP: VLAN0079 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.740: set portid: VLAN0146 Fa0/1: new port id 8003 *Mar  3 22:52:28.740:
STP: VLAN0146 Fa0/1 ->jump to forwarding from blocking
*Mar  3 22:52:28.749: set portid: VLAN0001 Fa0/1: new port id 8003 *Mar  3 22:52:28.749:
STP: VLAN0001 Fa0/1 ->jump to forwarding from blocking

!
%LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP PortFast Default

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Spanning-Tree PortFast on SW1 - SW4 so ports in access mode do not have to wait for the Spanning-Tree listening and learning phases to begin forwarding.
  - Configure FastEthernet0/1 on SW1 as an access port in vlan 146.
  - Do not use any interface-level Spanning-Tree commands to accomplish this.

Configuration

```
SW1:
spanning-tree portfast default
!
interface FastEthernet0/1
 switchport mode access
 switchport access vlan 146
SW2, SW3, SW4:

spanning-tree portfast default
```

Verification

Portfast default has the same effect as the interface-level portfast command, but it is automatically enabled at the same time only on interfaces statically configured in

https://t.me/learningnets

access mode. This command is equivalent to issuing the `spanning-tree portfast` command under an interface range that encompasses all interfaces. The port must be an access port for this to work, however. If the port is configured as a trunk, the global portfast command will not convert the port to an edge port; you must manually configure the port with the `spanning-tree portfast trunk` command. Regardless of how portfast is enabled, if BPDUs are received on the port, the port will lose its portfast and edge status, and STP port state will be negotiated, thus the port may actually transition to blocking.

Verify that PortFast is enabled at the global level, appearing as **by default**.

```
SW1#show spanning-tree summary
Switch is in pvst mode
Root bridge for: VLAN0001, VLAN0005, VLAN0007-VLAN0010, VLAN0022, VLAN0043
  VLAN0058, VLAN0067, VLAN0079, VLAN0146
Extended system ID           is enabled Portfast Default             is enabled

PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short

Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
VLAN0001                      0         0        0          3          3
VLAN0005                      0         0        0          3          3
VLAN0007                      0         0        0          3          3
VLAN0008                      0         0        0          3          3
VLAN0009                      0         0        0          3          3
VLAN0010                      0         0        0          3          3
VLAN0022                      0         0        0          3          3
VLAN0043                      0         0        0          3          3
VLAN0058                      0         0        0          3          3
VLAN0067                      0         0        0          3          3
VLAN0079                      0         0        0          3          3
VLAN0146                      0         0        0          4          4
---------------------- -------- --------- -------- ---------- ----------
12 vlans                      0         0        0         37         37
```

Verify that PortFast is actually also enabled at the interface level, thus port is an STP Edge port, because it is in access mode and no BPDUs have been received

https://t.me/learningnets

inbound.

```
SW1#show interfaces fastEthernet0/1 switchport
Name: Fa0/1
Switchport: EnabledAdministrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 146 (VLAN0146)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
<output omitted>
!
!SW1#show spanning-tree interface fastEthernet0/1 portfast
VLAN0146            enabled
!
!SW1#show spanning-tree interface fastEthernet0/1 detail
 Port 3 (FastEthernet0/1) of VLAN0146 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 4242, address 0013.605f.f000
   Designated bridge has priority 4242, address 0013.605f.f000
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1 The port is in the portfast mode by default
   Link type is point-to-point by default   BPDU: sent 72, received 0
```

Enable STP debugging to verify that ports transitions into forwarding state on link up.

```
SW1#debug spanning-tree event
Spanning Tree event debugging is onSW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#service timestamp debug datetime msec
SW1(config)#logging console 7
SW1(config)#interface FastEthernet0/1
SW1(config-if)#shutdown

!
*Mar  3 06:03:34.735: STP: VLAN0146 we are the spanning tree root
2d05h: %LINK-5-CHANGED: Interface FastEthernet0/1, changed state to administratively down
2d05h: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down
```

https://t.me/learningnets

When interface Fa0/1 is shut down and subsequently brought back up, it immediately transitions to the forwarding state.

```
SW1(config-if)#no shutdown
!
*Mar  3 06:04:03.634: set portid: VLAN0146 Fa0/1: new port id 8003*Mar  3 06:04:03.634:
STP: VLAN0146 Fa0/1 ->jump to forwarding from blocking

!
2d05h: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
2d05h: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP UplinkFast

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure SW2 - SW4 to use the default STP priority for all VLANs.
- Configure SW2 - SW4 with Spanning-Tree UplinkFast so that if their Root Port is lost, they immediately select a new Root Port.
- Verify this by shutting down the Root Port of SW2.

Configuration

```
SW2 - SW4:

default spanning-tree vlan 1-4094 priority
spanning-tree uplinkfast
```

Verification

https://t.me/learningnets

The Cisco-proprietary UplinkFast feature is used to speed up convergence time when the direct failure of the local Root Port occurs. This feature can be used only if the switch runs legacy STP, 802.1D, because functionality is built in to RSTP, 802.1w. To ensure that the switch with UplinkFast configured does not become a transit switch, its bridge priority will be automatically increased to 49152 and its ports cost increased with 3000. UplinkFast cannot be enabled on a switch that has its default STP priority modified.

Verify SW2's STP bridge priority, port costs, and port states after configuring the default STP priority but before configuring uplinkfast; for example, for VLAN 2:

```
SW2#show spanning-tree summary
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled UplinkFast                    is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short

Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
VLAN0001                      1         0        0          3          4
VLAN0002                      1         0        0          3          4
VLAN0005                      1         0        0          3          4
VLAN0007                      1         0        0          3          4
VLAN0008                      1         0        0          3          4
VLAN0009                      1         0        0          3          4
VLAN0010                      1         0        0          3          4
VLAN0022                      1         0        0          3          4
VLAN0043                      1         0        0          3          4
VLAN0058                      1         0        0          3          4
VLAN0067                      1         0        0          3          4
VLAN0079                      1         0        0          3          4
VLAN0146                      1         0        0          3          4
---------------------- -------- --------- -------- ---------- ----------
13 vlans                     13         0        0         39         52
!
!SW2#show spanning-tree vlan 2


VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    4098
```

https://t.me/learningnets

```
             Address     0013.605f.f000
             Cost        19
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
 Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p
Fa0/23              Root FWD 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

Enable STP event debugging only for VLAN 2, shut down the Root Port that is FastEthernet0/23, and notice that FastEhernet0/24 transitions to listening and learning before forwarding, so it is not automatically selected as the new Root Port.

```
SW2#debug spanning-tree events
Spanning Tree event debugging is on
!SW2#debug condition vlan 2
Condition 1 set
!SW2#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#service timestamps debug datetime msec
SW2(config)#interface fastEthernet0/23
SW2(config-if)#shutdown
!
!
*Mar  1 00:19:07.980: STP: VLAN0002 new root port Fa0/24, cost 19
*Mar  1 00:19:07.980: STP: VLAN0002 Fa0/24 -> listening
*Mar  1 00:19:07.980: STP[2]: Generating TC trap for port FastEthernet0/23
*Mar  1 00:19:09.985: STP: VLAN0002 sent Topology Change Notice on Fa0/24
*Mar  1 00:19:22.988: STP: VLAN0002 Fa0/24 -> learning
*Mar  1 00:19:37.995: STP[2]: Generating TC trap for port FastEthernet0/24
*Mar  1 00:19:37.995: STP: VLAN0002 sent Topology Change Notice on Fa0/24
*Mar  1 00:19:37.995: STP: VLAN0002 Fa0/24 -> forwarding
!
!SW2#show spanning-tree vlan 2

VLAN0002
```

https://t.me/learningnets

```
  Spanning tree enabled protocol ieee
  Root ID    Priority    4098
             Address     0013.605f.f000
             Cost        19
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
 Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/24              Root FWD 19        128.26   P2p
```

Configure uplinkfast and notice how SW2's STP bridge priority and port costs have been increased; for example, for VLAN 2:

```
SW2#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    4098
             Address     0013.605f.f000
             Cost        3019
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
 Bridge ID  Priority    49154  (priority 49152 sys-id-ext 2)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec
  Uplinkfast enabled

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 3019      128.21   P2p
Fa0/20              Desg FWD 3019      128.22   P2p
Fa0/23              Root FWD 3019      128.25   P2p
Fa0/24              Altn BLK 3019      128.26   P2p
```

Enable STP event debugging only for VLAN 2, shut down the Root Port that is

https://t.me/learningnets

FastEthernet0/23, and notice that FastEhernet0/24 transitions directly to forwarding, so it is automatically selected the new Root Port.

```
SW2#debug spanning-tree events
Spanning Tree event debugging is on
!SW2#debug condition vlan 2
Condition 1 set
!SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#service timestamps debug datetime msec
SW2(config)#interface fastEthernet0/23
SW2(config-if)#shutdown
!
!%SPANTREE_FAST-7-PORT_FWD_UPLINK: VLAN0001 FastEthernet0/24 moved to Forwarding (UplinkFast).
*Mar  1 01:09:03.619: STP: VLAN0002 new root port Fa0/24, cost 3019
*Mar  1 01:09:03.619: STP[2]: Generating TC trap for port FastEthernet0/24
*Mar  1 01:09:03.619: STP[2]: Generating TC trap for port FastEthernet0/23
*Mar  1 01:09:05.624: STP: VLAN0002 sent Topology Change Notice on Fa0/24
!
!SW2#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    4098
             Address     0013.605f.f000
             Cost        3019
             Port        26 (FastEthernet0/24)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
 Bridge ID  Priority    49154  (priority 49152 sys-id-ext 2)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec
  Uplinkfast enabled

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 3019      128.21   P2p
Fa0/20              Desg FWD 3019      128.22   P2p Fa0/24              Root FWD 3019      128.26   P2p
```

Verify that UplinkFast is enabled, and based on statistics it has been used when the Root Port has changed. The CAM table was flooded out the new Root Port to expedite the learning phase of upstream neighbors.

https://t.me/learningnets

```
SW2#show spanning-tree summary
Switch is in pvst mode

Root bridge for: none
Extended system ID           is enabled
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled UplinkFast                    is enabled
BackboneFast                 is disabled
Configured Pathcost method used is short

Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
VLAN0001                      0         0        0          3          3
VLAN0002                      0         0        0          3          3
VLAN0005                      0         0        0          3          3
VLAN0007                      0         0        0          3          3
VLAN0008                      0         0        0          3          3
VLAN0009                      0         0        0          3          3
VLAN0010                      0         0        0          3          3
VLAN0022                      0         0        0          3          3
VLAN0043                      0         0        0          3          3
VLAN0058                      0         0        0          3          3
VLAN0067                      0         0        0          3          3
VLAN0079                      0         0        0          3          3
VLAN0146                      0         0        0          3          3
---------------------- -------- --------- -------- ---------- ----------
13 vlans                      0         0        0         39         39

Station update rate set to 150 packets/sec.

UplinkFast statistics
-----------------------Number of transitions via uplinkFast (all VLANs)            : 26
Number of proxy multicast addresses transmitted (all VLANs) : 84
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP BackboneFast

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Spanning-Tree BackboneFast so that if the links between SW1 and SW2 go down, SW4 immediately expires its maxage timer and begins Spanning-Tree reconvergence.

Configuration

```
SW1 - SW4:

spanning-tree backbonefast
```

Verification

The Cisco proprietary BackboneFast feature is used to speed up convergence when an indirect failure occurs upstream in the network by immediately expiring the max_age timer. The feature is useful when running legacy 802.1d STP, because RSTP and MSTP standards were designed to have a built-in functionality to resolve the max_age timer expiry issue. It must be enabled on all switches in the Layer 2 topology to be functional.

Based on configured STP priorities from initial configurations, in this design, SW4's Root Port is toward SW2 on Fa0/19. If Fa0/19 goes down, Fa0/20 toward SW2 will

https://t.me/learningnets

be selected as the new Root Port.

```
SW4#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee Root ID    Priority    4098
             Address     0013.605f.f000
             Cost        38 Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8194   (priority 8192 sys-id-ext 2)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD 19        128.21   P2p

Fa0/20              Altn BLK 19        128.22   P2p
Fa0/23              Altn BLK 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

Before configuring BackboneFast, enable conditional STP events debugging for VLAN 2 on SW4 and shut down both Ethernet ports between SW2 and SW1. Check the log messages on SW4 and note that it ignores the inferior BPDU messages received from SW2 until the max_age timer of 20 seconds expires, allowing SW4 to age out the old root bridge stored BPDU on the respective port. SW2 sends inferior BPDU messages claiming itself as root bridge because it is isolated as it lost its Root Port.

```
SW4#debug spanning-tree events
Spanning Tree event debugging is on
!SW4#debug condition vlan 2
Condition 1 set
!SW4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW4(config)#service timestamps debug datetime msec
!
!SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#interface range fastEthernet0/23 - 24
SW2(config-if-range)#shutdown
!
!*Mar  3 02:45:30.829: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
```

https://t.me/learningnets

```
*Mar  3 02:45:30.829: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:31.895: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:31.895: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:33.891: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:33.891: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:35.904: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:35.904: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:37.909: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:37.909: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:39.914: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:39.914: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:41.911: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:41.911: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:43.924: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:43.924: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:45.929: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:45.929: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
SW4#
*Mar  3 02:45:47.934: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 02:45:47.934: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
```

After max_age timer expires, SW4 starts reconverging its STP port states and finally selects the new Root Port to be Fas0/23, its connection to SW3.

```
*Mar  3 02:45:48.814: STP: VLAN0002 new root port Fa0/20, cost 38
*Mar  3 02:45:48.814: STP: VLAN0002 Fa0/20 -> listening
*Mar  3 02:45:48.814: STP: VLAN0002 new root port Fa0/23, cost 38
*Mar  3 02:45:48.814: STP: VLAN0002 Fa0/23 -> listening
SW4#
*Mar  3 02:45:49.855: STP: VLAN0002 Topology Change rcvd on Fa0/19
*Mar  3 02:45:49.855: STP: VLAN0002 sent Topology Change Notice on Fa0/23
SW4#
*Mar  3 02:46:03.822: STP: VLAN0002 Fa0/20 -> learning
*Mar  3 02:46:03.822: STP: VLAN0002 Fa0/23 -> learning
SW4#
*Mar  3 02:46:18.829: STP[2]: Generating TC trap for port FastEthernet0/20
```

https://t.me/learningnets

```
*Mar  3 02:46:18.829: STP: VLAN0002 sent Topology Change Notice on Fa0/23
*Mar  3 02:46:18.829: STP: VLAN0002 Fa0/20 -> forwarding
*Mar  3 02:46:18.829: STP[2]: Generating TC trap for port FastEthernet0/23
*Mar  3 02:46:18.829: STP: VLAN0002 Fa0/23 -> forwarding
!
!SW4#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    4098
             Address     0013.605f.f000
             Cost        38
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8194   (priority 8192 sys-id-ext 2)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/23              Root FWD 19        128.25   P2p

Fa0/24              Altn BLK 19        128.26   P2p
```

Re-enable SW2's Ethernet links to SW1, Fa0/23, and Fa0/24, configure BackboneFast on all switches, enable STP BackboneFast debugging, and test the same scenario again. Note that as soon as SW4 receives the inferior BPDU from SW2, it no longer waits for max_age timer to expire; instead, SW4 sends RLQ (Root Link Query) messages out on all its non-designated ports.

```
SW4#debug spanning-tree backbonefast
Spanning Tree backbonefast general debugging is on
!SW4#debug spanning-tree backbonefast detail
Spanning Tree backbonefast detail debugging is on
!SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#interface range fastEthernet0/23 - 24
SW2(config-if-range)#shutdown
!
!
*Mar  3 06:40:27.876: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/19
*Mar  3 06:40:27.876: STP FAST: received inferior BPDU on VLAN0002 FastEthernet0/19.
*Mar  3 06:40:27.876: STP FAST: sending RLQ request PDU on VLAN0002(2) Fa0/20 Vlan2
*Mar  3 06:40:27.876: STP FAST: sending RLQ request PDU on VLAN0002(2) Fa0/23 Vlan2
```

https://t.me/learningnets

```
*Mar  3 06:40:27.876: STP: VLAN0002 heard root 16386-000a.b832.3a80 on Fa0/20
*Mar  3 06:40:27.885: STP FAST: received inferior BPDU on VLAN0002 FastEthernet0/20.
*Mar  3 06:40:27.885: STP FAST: sending RLQ request PDU on VLAN0002(2) Fa0/23 Vlan2
*Mar  3 06:40:27.885: STP FAST: sending RLQ request PDU on VLAN0002(2) Fa0/24 Vlan2
```

SW4 receives a negative RLQ response from SW2 (the root bridge is not accessible via SW2) and positive RLQ response from SW3 (the root bridge is still accessible via SW3).

```
*Mar  3 06:40:27.901: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/20.
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/23.
*Mar  3 06:40:27.910: STP FAST: received RLQ response PDU was expected on VLAN0002 FastEthernet0/23 - resp root id 4
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/24.
*Mar  3 06:40:27.910: STP FAST: received RLQ response PDU was expected on VLAN0002 FastEthernet0/24 - resp root id 4

*Mar  3 06:40:27.910: STP FAST: received_rlq_bpdu on VLAN0002 FastEthernet0/19 - making FastEthernet0/19 a designate
*Mar  3 06:40:27.910: STP FAST: received_rlq_bpdu on VLAN0002 FastEthernet0/20 - making FastEthernet0/20 a designate
```

Based on received RLQ responses, SW4 will mark its ports to SW2 as designated and will select its new Root Port via SW3. All ports, however, still transition through regular STP port states, listening, learning, and forwarding or blocking.

```
*Mar  3 06:40:27.910: STP: VLAN0002 new root port Fa0/23, cost 38
*Mar  3 06:40:27.910: STP: VLAN0002 Fa0/23 -> listening
*Mar  3 06:40:27.910: STP: VLAN0002 Fa0/20 -> listening
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/23.
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/24.
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/23.
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/24.
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/23.
*Mar  3 06:40:27.910: STP FAST: Received RLQ response PDU on VLAN0002 FastEthernet0/24.
*Mar  3 06:40:28.405: STP: VLAN0002 Topology Change rcvd on Fa0/19
*Mar  3 06:40:28.405: STP: VLAN0002 sent Topology Change Notice on Fa0/23
*Mar  3 06:40:42.917: STP: VLAN0002 Fa0/23 -> learning
*Mar  3 06:40:42.917: STP: VLAN0002 Fa0/20 -> learning
*Mar  3 06:40:57.924: STP[2]: Generating TC trap for port FastEthernet0/23
*Mar  3 06:40:57.924: STP: VLAN0002 sent Topology Change Notice on Fa0/23
*Mar  3 06:40:57.924: STP: VLAN0002 Fa0/23 -> forwarding
*Mar  3 06:40:57.924: STP[2]: Generating TC trap for port FastEthernet0/20
*Mar  3 06:40:57.924: STP: VLAN0002 Fa0/20 -> forwarding
!
!SW4#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
```

https://t.me/learningnets

```
  Root ID    Priority    4098
             Address     0013.605f.f000
             Cost        38
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8194   (priority 8192 sys-id-ext 2)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/23              Root FWD 19        128.25   P2p

Fa0/24              Altn BLK 19        128.26   P2p
```

Verify that BackboneFast is enabled, and also note the exchange of RLQ messages; for example, on SW4:

```
SW4#show spanning-tree summary
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled BackboneFast                  is enabled
Configured Pathcost method used is short

Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
VLAN0001                     1         0        0          4          5
VLAN0002                     1         0        0          3          4
VLAN0005                     1         0        0          3          4
VLAN0007                     1         0        0          3          4
VLAN0008                     1         0        0          3          4
VLAN0009                     1         0        0          3          4
VLAN0010                     1         0        0          3          4
VLAN0022                     1         0        0          3          4
VLAN0043                     1         0        0          3          4
VLAN0058                     1         0        0          3          4
```

https://t.me/learningnets

```
VLAN0067                     1         0        0          3          4
VLAN0079                     1         0        0          3          4
VLAN0146                     1         0        0          3          4
---------------------- -------- --------- -------- ---------- ----------
13 vlans                    13         0        0         40         53

BackboneFast statistics
-----------------------
Number of transition via backboneFast (all VLANs)           : 13
Number of inferior BPDUs received (all VLANs)               : 26
Number of RLQ request PDUs received (all VLANs)             : 0
Number of RLQ response PDUs received (all VLANs)            : 117
Number of RLQ request PDUs sent (all VLANs)                 : 65

Number of RLQ response PDUs sent (all VLANs)                : 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP BPDU Guard

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure a port-channel between SW3 and SW4 as follows:
  - SW3's side should be a Layer 3 port-channel with IP address 169.254.34.3/24.
  - SW4's side should be a Layer 2 port-channel in VLAN 10.
  - Both switches should actively initiate negotiation using a standard protocol.
- Configure Spanning-Tree BPDU Guard on SW4 so that the etherchannel is disabled if a BPDU is detected.
  - SW4 should attempt to re-enable the etherchannel after two minutes.
  - Do not use the global `portfast` command to accomplish this.

Configuration

```
SW3:
default interface range FastEthernet0/23 - 24
!
interface Port-channel34
 no switchport
 ip address 169.254.34.3 255.255.255.0
!
interface range FastEthernet0/23 - 24
 no switchport
 channel-group 34 mode active
```

https://t.me/learningnets

```
SW4:

default interface range FastEthernet0/23 - 24
!
interface range FastEthernet0/23 - 24
 channel-group 34 mode active
!
interface Port-channel34
 switchport mode access
 switchport access vlan 10
 spanning-tree bpduguard enable
!
errdisable recovery cause bpduguard
errdisable recovery interval 120
```

Verification

The STP BPDU Guard feature is used to enforce access layer security on the termination of the STP domain. When an interface running BPDU Guard receives a BPDU (STP packet), the interface is transitioned into err-disable state. This ensures that unauthorized switches cannot be plugged in to the network, for example, to perform a Layer 2 man-in-the-middle (MiM) attack. If configured, the **`errdisable recovery`** feature can then be used to bring the interface out of err-disable state automatically after a configured interval.

Note that no BPDUs have been received on the interface and BPDu Guard is enabled at the interface level:

```
SW4#show spanning-tree interface port-channel34 detail
 Port 328 (Port-channel34) of VLAN0010 is designated forwarding
   Port path cost 12, Port priority 128, Port Identifier 128.328.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 8202, address 001a.a174.2500
   Designated port id is 128.328, designated path cost 38
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default Bpdu guard is enabled
   BPDU: sent 19, received 0
!
!SW4#show spanning-tree interface fastEthernet0/23 detail
 Port 328 (Port-channel34) of VLAN0010 is designated forwarding
   Port path cost 12, Port priority 128, Port Identifier 128.328.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 8202, address 001a.a174.2500
```

https://t.me/learningnets

```
   Designated port id is 128.328, designated path cost 38
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default Bpdu guard is enabled
   BPDU: sent 34, received 0
!
!SW4#show spanning-tree interface fastEthernet0/24 detail
 Port 328 (Port-channel34) of VLAN0010 is designated forwarding
   Port path cost 12, Port priority 128, Port Identifier 128.328.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 8202, address 001a.a174.2500
   Designated port id is 128.328, designated path cost 38
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default Bpdu guard is enabled
   BPDU: sent 45, received 0
```

By re-configuring SW3's port-channel as Layer 2, STP BPDUs are generated and SW4 will err-disable the port-channel:

https://t.me/learningnets

```
SW3#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
!SW3(config)#no interface port-channel34
SW3(config)#default interface range fastEthernet0/23 - 24
SW3(config)#interface range fastEthernet0/23 - 24
SW3(config-if-range)#channel-group 34 mode active

!
!SW4#show interfaces port-channel34 status err-disabled


Port      Name               Status       Reason               Err-disabled Vlans
Po34                         err-disabled bpduguard

!
!SW4#show interfaces fastEthernet0/23 status err-disabled


Port      Name               Status       Reason               Err-disabled Vlans
Fa0/23                       err-disabled bpduguard

!
!SW4#show interfaces fastEthernet0/24 status err-disabled


Port      Name               Status       Reason               Err-disabled Vlans
Fa0/24                       err-disabled bpduguard
```

As soon as Etherchannel is negotiated via LACP the following log messaged will be displayed by SW4, identifying the problem:

```
%PM-4-ERR_DISABLE: bpduguard error detected on Fa0/23, putting Fa0/23 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Fa0/24, putting Fa0/24 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Fa0/23 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Fa0/24 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Po34 in err-disable state
```

Based on the configured err-disable recovery functionality, after two minutes SW4 will try to recover it and thus re-activate it, however the scenario will repeat as

https://t.me/learningnets

BPDUs are received on the ports:

```
%PM-4-ERR_RECOVER: Attempting to recover from bpduguard err-disable state on Fa0/23
%PM-4-ERR_RECOVER: Attempting to recover from bpduguard err-disable state on Po34
%PM-4-ERR_RECOVER: Attempting to recover from bpduguard err-disable state on Fa0/24
!%SPANTREE-2-BLOCK_BPDUGUARD: Received BPDU on port Po34 with BPDU Guard enabled. Disabling port.

%PM-4-ERR_DISABLE: bpduguard error detected on Fa0/23, putting Fa0/23 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Fa0/24, putting Fa0/24 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Fa0/23 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Fa0/24 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Po34 in err-disable state
```

Verify that err-disable recovery is activated for BPDU Guard and check timers:

```
SW4#show errdisable recovery
ErrDisable Reason            Timer Status
-----------------            --------------
arp-inspection               Disabled bpduguard                    Enabled
channel-misconfig (STP)      Disabled
dhcp-rate-limit              Disabled
dtp-flap                     Disabled
gbic-invalid                 Disabled
inline-power                 Disabled
l2ptguard                    Disabled
link-flap                    Disabled
mac-limit                    Disabled
loopback                     Disabled
pagp-flap                    Disabled
port-mode-failure            Disabled
pppoe-ia-rate-limit          Disabled
psecure-violation            Disabled
security-violation           Disabled
sfp-config-mismatch          Disabled
small-frame                  Disabled
storm-control                Disabled
udld                         Disabled
vmps                         Disabled
psp                          Disabled
Timer interval: 120 seconds


Interfaces that will be enabled at the next timeout:

Interface       Errdisable reason       Time left(sec)
```

https://t.me/learningnets

```
---------       -----------------       --------------
Fa0/23                  bpduguard           66
Fa0/24                  bpduguard           66
Po34                    bpduguard           66
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP BPDU Guard Default

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure a port-channel between SW3 and SW4 as follows:
  - SW3's side should be a Layer 3 port-channel with IP address 169.254.34.3/24.
  - SW4's side should be a Layer 2 port-channel in VLAN 10.
  - Both switches should actively initiate negotiation using a standard protocol.
- Configure Spanning-Tree BPDU Guard on SW4 so that the etherchannel is disabled if a BPDU is detected.
  - SW4 should attempt to re-enable the etherchannel after two minutes.
  - Do not use interface-level commands to accomplish this.

Configuration

```
SW3:
default interface range FastEthernet0/23 - 24
!
interface Port-channel34
 no switchport
 ip address 169.254.34.3 255.255.255.0
!
interface range FastEthernet0/23 - 24
 no switchport
 channel-group 34 mode active
```

https://t.me/learningnets

```
SW4:

default interface range FastEthernet0/23 - 24
!
spanning-tree portfast bpduguard default
spanning-tree portfast default
!
interface range FastEthernet0/23 - 24
 channel-group 34 mode active
!
interface Port-channel34
 switchport mode access
 switchport access vlan 10
!
errdisable recovery cause bpduguard
errdisable recovery interval 120
```

Verification

The BPDU Guard Default feature works in conjunction with Portfast to automatically enable BPDU Guard on any interfaces in the Portfast state. Portfast can be enable at the global level, or interface level, task requirements restrics interface-level commands. Verify that BPDU Guard is enabled by default, at the global level:

```
SW4#show spanning-tree interface port-channel34 detail
 Port 328 (Port-channel34) of VLAN0010 is designated forwarding
   Port path cost 12, Port priority 128, Port Identifier 128.328.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 8202, address 001a.a174.2500
   Designated port id is 128.328, designated path cost 38
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1 The port is in the portfast mode by default
   Link type is point-to-point by default Bpdu guard is enabled by default
   BPDU: sent 5, received 0
!
!SW4#show spanning-tree interface fastEthernet0/23 detail
 Port 328 (Port-channel34) of VLAN0010 is designated forwarding
   Port path cost 12, Port priority 128, Port Identifier 128.328.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 8202, address 001a.a174.2500
   Designated port id is 128.328, designated path cost 38
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1 The port is in the portfast mode by default
```

https://t.me/learningnets

```
   Link type is point-to-point by default Bpdu guard is enabled by default
   BPDU: sent 20, received 0
!
!SW4#show spanning-tree interface fastEthernet0/24 detail
Port 328 (Port-channel34) of VLAN0010 is designated forwarding
   Port path cost 12, Port priority 128, Port Identifier 128.328.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 8202, address 001a.a174.2500
   Designated port id is 128.328, designated path cost 38
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1 The port is in the portfast mode by default
   Link type is point-to-point by default Bpdu guard is enabled by default
   BPDU: sent 26, received 0
!
!SW4#show spanning-tree summary
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled Portfast Default              is enabled
PortFast BPDU Guard Default  is enabled

Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short
<output omitted>
```

By re-configuring SW3's port-channel as Layer 2, STP BPDUs are generated and SW4 will err-disable the port-channel:

```
SW3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
!SW3(config)#no interface port-channel34
SW3(config)#default interface range fastEthernet0/23 - 24
SW3(config)#interface range fastEthernet0/23 - 24
SW3(config-if-range)#channel-group 34 mode active
!
!SW4#show interfaces port-channel34 status err-disabled

Port      Name               Status       Reason               Err-disabled Vlans
Po34                         err-disabled bpduguard
!
!SW4#show interfaces fastEthernet0/23 status err-disabled
```

https://t.me/learningnets

```
Port      Name               Status       Reason               Err-disabled Vlans
Fa0/23                       err-disabled bpduguard
!
!SW4#show interfaces fastEthernet0/24 status err-disabled

Port      Name               Status       Reason               Err-disabled Vlans
Fa0/24                       err-disabled bpduguard
```

As soon as Etherchannel is negotiated via LACP the following log messaged will be displayed by SW4, identifying the problem:

```
%PM-4-ERR_DISABLE: bpduguard error detected on Fa0/23, putting Fa0/23 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Fa0/24, putting Fa0/24 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Fa0/23 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Fa0/24 in err-disable state
%PM-4-ERR_DISABLE: bpduguard error detected on Po34, putting Po34 in err-disable state
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP BPDU Filter

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Shut down FastEthernet0/19 on both SW2 and SW4 and:
  - Configure SW2's FastEthernet0/20 as Layer 3 with IP address 169.254.24.2/24.
  - Configure SW4's FastEthernet0/20 as an access port in VLAN 10.
- Configure SW4 so that it filters all inbound/outbound BPDU packets on its FastEthernet0/20 port.

Configuration

```
SW2:
interface Fastethernet0/19
 shutdown
!
default interface Fastethernet0/20
!
interface Fastethernet0/20
 no switchport
 ip address 169.254.24.2 255.255.255.0
SW4:

interface Fastethernet0/19
 shutdown
!
```

https://t.me/learningnets

```
default interface Fastethernet0/20
!
interface Fastethernet0/20
 shutdown
 switchport mode access
 switchport access vlan 10
 spanning-tree bpdufilter enable
 no shutdown
```

Verification

The BPDU Filter feature, like the BPDU Guard feature, is used to terminate the STP domain, but has a different functionality. It can also be configured globally or at the interface level, however behaviour is different based on this, while this was not the case for BPDU Guard which had the same functionality regardless of how it was enabled.

When configured at the interface level, BPDU Filter silently drops all received inbound BPDUs and does not send any outbound BPDUs on the port. There is no violation option for BPDU Filter, thus the port never goes into err-disabled state. BPDU Filter needs to be carefully enabled at the port level, because it will cause permanent loops if on the other end of the link a switch is connected and the network is physically looped; in this case STP will not be able to detect the loop and the network will become unusable within seconds.

Verify that BPDU Filter is enabled at the port level, confirmed by the fact that no BPDUs are sent, BPDUs are not received on the port anyways as the remote link is Layer 3, thus SW2 does not run STP on this link:

https://t.me/learningnets

```
SW4#show spanning-tree interface fastEthernet0/20 detail

 Port 22 (FastEthernet0/20) of VLAN0010 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 8202, address 001a.a174.2500
   Designated bridge has priority 8202, address 001a.a174.2500
   Designated port id is 128.22, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   The port is in the portfast mode by default
   Link type is point-to-point by default
   Bpdu guard is enabled by default Bpdu filter is enabled
 BPDU: sent 0, received 0
```

By configuring SW2's FastEthernet0/20 as Layer 2, it will start sending BPDUs to negotiate the port STP state, however SW4 will filter BPDUs inbound:

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
!SW2(config)#default interface fastEthernet0/20
SW2(config)#interface FastEthernet0/20
SW2(config-if)# switchport access vlan 10
SW2(config-if)# switchport mode access
!
!SW4#show cdp neighbors fastEthernet0/20
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW2              Fas 0/20          155              S I   WS-C3560- Fas 0/20
!
!SW4#show spanning-tree interface fastEthernet0/20 detail
 Port 22 (FastEthernet0/20) of VLAN0010 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 8202, address 001a.a174.2500
   Designated bridge has priority 8202, address 001a.a174.2500
   Designated port id is 128.22, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   The port is in the portfast mode by default
   Link type is point-to-point by default
   Bpdu guard is enabled by default
```

https://t.me/learningnets

```
   Bpdu filter is enabled BPDU: sent 0, received 0
!
!SW2#show spanning-tree interface fastEthernet0/20 detail
 Port 22 (FastEthernet0/20) of VLAN0010 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 16394, address 000a.b832.3a80
   Designated port id is 128.22, designated path cost 19
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default BPDU: sent 26, received 0
```

At this point we have a logical layer 2 loop in the network for VLAN 10, however because there is no live traffic within VLAN 10, network is still stable:

```
SW1#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD
 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/23              Desg FWD
 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p
!
!SW2#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        19
```

https://t.me/learningnets

```
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    16394  (priority 16384 sys-id-ext 10)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Desg FWD
 19        128.22   P2p Fa0/23              Root FWD
 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
!
!SW3#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        19
             Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32778  (priority 32768 sys-id-ext 10)
             Address     0022.5627.1f80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD
 19        128.21   P2p
Fa0/20              Altn BLK 19        128.22   P2p Fa0/23              Desg FWD
 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p
!
!SW4#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        38
             Port        25 (FastEthernet0/23)
```

https://t.me/learningnets

```
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8202   (priority 8192 sys-id-ext 10)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Desg FWD
 19        128.22   P2p Edge Fa0/23              Root FWD
 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP BPDU Filter Default

> You must load the initial configuration files for the section, **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Shut down FastEthernet0/19 on both SW2 and SW4 and:
  - Configure SW2's FastEthernet0/20 as Layer 3 with IP address 169.254.24.2/24.
  - Configure SW4's FastEthernet0/20 as an access port in VLAN 10.
- Configure BPDU Filter on SW4 so that it applies only to its PortFast-enabled ports.
  - Ensure that FastEthernet0/20 is subject to this policy.

Configuration

```
SW2:
interface Fastethernet0/19
 shutdown
!
default interface Fastethernet0/20
!
interface Fastethernet0/20
 no switchport
 ip address 169.254.24.2 255.255.255.0
SW4:

interface Fastethernet0/19
 shutdown
!
```

https://t.me/learningnets

```
spanning-tree portfast bpdufilter default
!
default interface Fastethernet0/20
!
interface Fastethernet0/20
 shutdown
 switchport mode access
 switchport access vlan 10
 spanning-tree portfast
 no shutdown
```

Verification

When BPDU Filter is configured globally, it only affects PortFast-enabled ports; PortFast can be configured globally or at the port level. With BPDU Filter globally enabled, the switch still sends out exactly 11 BPDUs on all Filter-enabled ports, and received BPDUs are not filtered. When the switch receives a BPDU inbound on a BPDU Filter-enabled port, the port also loses its PortFast status and its STP state is negotiated. You can say that BPDU Filter globally enabled is a safer mechanism to terminate the STP domain, because loops cannot be formed. Additionally, if the port is truly Edge, and thus connected to a host, after the first 11 BPDUs the switch no longer sends BPDUs on the port until a BPDU is received inbound.

Verify that BPDU Filter is enabled at the global level and PortFast is enabled at the port level.

```
SW4#show spanning-tree summary
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled Portfast BPDU Filter Default is enabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short
<output omitted>
!
!SW4#show spanning-tree interface fastEthernet0/20 detail
 Port 22 (FastEthernet0/20) of VLAN0010 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 8202, address 001a.a174.2500
```

https://t.me/learningnets

```
   Designated port id is 128.22, designated path cost 38
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1 The port is in the portfast mode
   Link type is point-to-point by default Bpdu filter is enabled by default

   BPDU: sent 11, received 0
```

When you configure SW2's FastEthernet0/20 as Layer 2, it will start sending BPDUs to negotiate the port STP state. This time SW4 will accept the BPDUs and start STP negotiation, and PortFast and BPDU Filter states are lost for the port.

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
!SW2(config)#default interface fastEthernet0/20
SW2(config)#interface FastEthernet0/20
SW2(config-if)# switchport access vlan 10
SW2(config-if)# switchport mode access
!
!SW4#show cdp neighbors fastEthernet0/20
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW2              Fas 0/20          155              S I   WS-C3560- Fas 0/20
!
!SW2#show spanning-tree interface fastEthernet0/20 detail
 Port 22 (FastEthernet0/20) of VLAN0010 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 16394, address 000a.b832.3a80
   Designated port id is 128.22, designated path cost 19
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default BPDU: sent 31, received 1
!
!SW4#show spanning-tree interface fastEthernet0/20 portfast
VLAN0010            disabled
!
!SW4#show spanning-tree interface fastEthernet0/20 detail
 Port 22 (FastEthernet0/20) of VLAN0010 is root forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.22.
   Designated root has priority 4106, address 0013.605f.f000
   Designated bridge has priority 16394, address 000a.b832.3a80
   Designated port id is 128.22, designated path cost 19
```

https://t.me/learningnets

```
    Timers: message age 3, forward delay 0, hold 0
    Number of transitions to forwarding state: 1
    Link type is point-to-point by default BPDU: sent 1, received 31
```

This time, because SW4 automatically disabled the BPDU Filter on the port and allows STP to negotiate, a loop is no longer formed in the network for VLAN 10. The loop is terminated by SW4, which has its Fa0/23 port toward SW3 in BLK.

```
SW1#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD
 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/23              Desg FWD
 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p
!
!SW2#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        19
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    16394  (priority 16384 sys-id-ext 10)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec
```

https://t.me/learningnets

```
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Desg FWD
 19        128.22   P2p Fa0/23              Root FWD
 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
!
!SW3#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        19
             Port        21 (FastEthernet0/19)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32778  (priority 32768 sys-id-ext 10)
             Address     0022.5627.1f80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Root FWD
 19        128.21   P2p
Fa0/20              Altn BLK 19        128.22   P2p Fa0/23              Desg FWD
 19        128.25   P2p
Fa0/24              Desg FWD 19        128.26   P2p
!
!SW4#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
             Cost        38
             Port        22 (FastEthernet0/20)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8202   (priority 8192 sys-id-ext 10)
             Address     001a.a174.2500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec
```

https://t.me/learningnets

```
Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Root FWD
 19        128.22   P2p Fa0/23              Altn BLK
 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP Root Guard

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure SW1 so that STP logically blocks Ethernet links connected to SW2 and SW3 if any of SW2 - SW4 tries to become Root Bridge for any VLAN.

Configuration

```
SW1:

interface range FastEthernet0/19 - 20
 spanning-tree guard root
!
interface range FastEthernet0/23 - 24
 spanning-tree guard root
```

Verification

Root Guard is similar to the BPDU Guard feature in the manner in which it is used to detect STP packets and disable the interface they were received on. The difference between them is that with Root Guard, the interface is only logically disabled (via **Root Inconsistent** state) if a superior BPDU is received on the port with Root Guard enabled. **Root Inconsistent** state is similar to blocking state, in that BPDUs are not sent outbound, but accepted inbound, and of course all received frames are

https://t.me/learningnets

dropped. The switch automatically recovers the port from **Root Inconsistent** and starts negotiating the new port state and role, as soon as superior BPDUs are no longer received inbound.

A superior BPDU indicates a better cost to the root bridge than what is currently installed. Therefore, design-wise this feature is used to prevent a rogue device from announcing itself as the new root bridge and possibly implementing a layer 2 man-in-the-middle attack. Root Guard can be enabled only at the port level and basically prevents a Designated port from becoming Non-Designated. You will want to configure this functionality on the Root Bridge itself.

Verify that Root Guard is enabled for all VLANs, for example on FastEthernet0/19 port:

```
SW1#show spanning-tree interface fastEthernet0/19 detail | i Port|Root
 Port 21 (FastEthernet0/19) of VLAN0001
 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21. Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0002
 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21. Root guard is enabled on the port

 Port 21 (FastEthernet0/19) of VLAN0005 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0007 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0008 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0009 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0010 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0022 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0043 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0058 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
```

https://t.me/learningnets

```
 Port 21 (FastEthernet0/19) of VLAN0067 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0079 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0146 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Root guard is enabled on the port
```

Although Root Guard is enabled at the port level, it works on a per-VLAN basis. For example let's configure SW2 with a better bridge priority for VLAN 2, which means that SW1 will logically disable its port to SW2 only for VLAN 2:

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW2(config)#spanning-tree vlan 2 priority 0
!
!SW1#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    4098
             Address     0013.605f.f000
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4098   (priority 4096 sys-id-ext 2)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg BKN*19
        128.21   P2p *ROOT_Inc Fa0/20              Desg BKN*19
        128.22   P2p *ROOT_Inc Fa0/23              Desg BKN*19
        128.25   P2p *ROOT_Inc Fa0/24              Desg BKN*19
        128.26   P2p *ROOT_Inc
!
!SW1#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    4106
             Address     0013.605f.f000
```

https://t.me/learningnets

```
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD
 19        128.21   P2p Fa0/20              Desg FWD
 19        128.22   P2p Fa0/23              Desg FWD
 19        128.25   P2p Fa0/24              Desg FWD
 19        128.26   P2p
```

SW1 will also log messages similar to the following, notifying of the problem:

```
%SPANTREE-2-ROOTGUARD_BLOCK: Root guard blocking port FastEthernet0/23 on VLAN0002.
%SPANTREE-2-ROOTGUARD_BLOCK: Root guard blocking port FastEthernet0/24 on VLAN0002.
```

Because SW1 no longer sends BPDUs outbound on its **Root Inconsistent** port, note that SW2 and SW3 have their ports towards SW1 in **FWD** state for VLAN 2:

```
SW2#show spanning-tree vlan 2

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    2
             Address     000a.b832.3a80
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    2      (priority 0 sys-id-ext 2)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/23              Desg FWD
 19        128.25   P2p Fa0/24              Desg FWD
 19        128.26   P2p
!
```

https://t.me/learningnets

```
!SW3#show spanning-tree vlan 2



VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    2
             Address     000a.b832.3a80
             Cost        38
             Port        25 (FastEthernet0/23)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     0022.5627.1f80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD
 19        128.21   P2p Fa0/20              Desg FWD
 19        128.22   P2p
Fa0/23              Root FWD 19        128.25   P2p
Fa0/24              Altn BLK 19        128.26   P2p
```

Once superior BPDUs are no longer received, SW1 will start to send BPDUs outbound on the ports to negotiate the STP state and role, it will also log messages similar with the following:

```
%SPANTREE-2-ROOTGUARD_UNBLOCK: Root guard unblocking port FastEthernet0/23 on VLAN0002.
%SPANTREE-2-ROOTGUARD_UNBLOCK: Root guard unblocking port FastEthernet0/24 on VLAN0002.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

STP Loop Guard

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Spanning-Tree Loop Guard to prevent unidirectional links from forming on any of the inter-switch links in the Layer 2 network.
  - Do not use any interface level commands on SW1 and SW2.

Configuration

```
SW1 - SW2:
spanning-tree loopguard default
SW3 - SW4:

interface range FastEthernet0/19 - 20
 spanning-tree guard loop
!
interface range FastEthernet0/23 - 24
 spanning-tree guard loop
```

Verification

STP Loop Guard is used to prevent STP loops from occurring because of unidirectional links. This feature is similar to Unidirectional Link Detection (UDLD), but it uses STP BPDUs to determine whether there is a unidirectional link. Loop

https://t.me/learningnets

Guard can be enabled globally at the switch level, or specific at the port level. Loop Guard prevents a Non-Designated port from becoming Designated, thus it is the opposite of Root Guard; for this reason Root Guard and Loop Guard cannot be actively enabled at the same time on the same ports, and it doesn't even make sense to do it. When globally configured, although from the output it seems as being enabled on all ports in the **UP** state, actually Loop Guard only monitors Non-Designated ports.

In normal STP operation in a redundant topology, some links will be in designated forwarding while the other end will be in alternate blocking or root forwarding. If one of these blocking links transitions to forwarding state erroneously, a loop can occur. Specifically, this can happen if there is a unidirectional link and the blocking port stops receiving the BPDUs that the designated port it sending (on any given segment BPDUs are sent only by the designated port, unless bridge assurance is configured in which case all ports generate BPDUs regardless of the state and role). Loop guard prevents this by transitioning blocking ports into **Loop Inconsistent** state instead of forwarding if BPDUs stop being received from the designated port.

Just like Root Guard, although is enabled for a port, Loop Guard takes actions on a per-VLAN level; for example if a trunk port is in blocking state and stops receiving BPDUs for VLAN 2 from the designated port on the segment, it transitions the port into **Loop Inconsistent** only for VLAN 2. Switch will automatically recover the port from **Loop Inconsistent** state when it starts receiving BPDUs and the STP port state is re-negotiated. **Loop Inconsistent** is also similar to blocking state, as no BPDUs are sent outbound, BPDUs are accepted inbound and all received inbound data frames are dropped.

Verify that Loop Guard is enabled at the global level on SW1 and SW2, but not on SW3 and SW4:

```
SW2#show spanning-tree summary
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled Loopguard Default            is enabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short
<output omitted>
!
!SW3#show spanning-tree summary
```

https://t.me/learningnets

```
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled Loopguard Default            is disabled

EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short
```

Verify that Loop Guard is enabled on all ports of SW1, because it was globally configured:

```
SW1#show spanning-tree interface fastEthernet0/1 detail
 Port 3 (FastEthernet0/1) of VLAN0001 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 4097, address 0013.605f.f000
   Designated bridge has priority 4097, address 0013.605f.f000
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default Loop guard is enabled by default on the port
   BPDU: sent 1142, received 0
!
!SW1#show spanning-tree interface fastEthernet0/19 detail | i Port|Loop
 Port 21 (FastEthernet0/19) of VLAN0001 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
 Loop guard is enabled by default on the port
 Port 21 (FastEthernet0/19) of VLAN0002 is designated forwarding

   Port path cost 19, Port priority 128, Port Identifier 128.21.
 Loop guard is enabled by default on the port

 Port 21 (FastEthernet0/19) of VLAN0005 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Loop guard is enabled by default on the port
 Port 21 (FastEthernet0/19) of VLAN0007 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Loop guard is enabled by default on the port
 Port 21 (FastEthernet0/19) of VLAN0008 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.21.
   Loop guard is enabled by default on the port
 Port 21 (FastEthernet0/19) of VLAN0009 is designated forwarding
```

https://t.me/learningnets

```
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
  Port 21 (FastEthernet0/19) of VLAN0010 is designated forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
  Port 21 (FastEthernet0/19) of VLAN0022 is designated forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
  Port 21 (FastEthernet0/19) of VLAN0043 is designated forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
  Port 21 (FastEthernet0/19) of VLAN0058 is designated forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
  Port 21 (FastEthernet0/19) of VLAN0067 is designated forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
  Port 21 (FastEthernet0/19) of VLAN0079 is designated forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
  Port 21 (FastEthernet0/19) of VLAN0146 is designated forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled by default on the port
```

Verify that on SW3 and SW4, Loop Guard is enabled at the port level:

```
SW3#show spanning-tree interface fastEthernet0/19 detail | i Port|Loop
 Port 21 (FastEthernet0/19) of VLAN0001 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21. Loop guard is enabled on the port
 Port 21 (FastEthernet0/19) of VLAN0002 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21. Loop guard is enabled on the port

  Port 21 (FastEthernet0/19) of VLAN0005 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0007 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0008 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0009 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0010 is root forwarding
```

https://t.me/learningnets

```
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0022 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0043 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0058 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0067 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0079 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
  Port 21 (FastEthernet0/19) of VLAN0146 is root forwarding
    Port path cost 19, Port priority 128, Port Identifier 128.21.
    Loop guard is enabled on the port
```

Configure BPDU Filter on SW3's FastEthernet0/19 port, at the port level, which means SW3 will no longer send outbound BPDUs and filter all inbound BPDUs. Although, as seen from above output, SW1 seems to have Loop Guard enabled on its FastEthernet0/19 port, this will not cause SW1 to transition the port into Loop Inconsistent, as Loop Guard only monitors Non-Designated ports and the failure on receiving BPDUs, and SW1 being the root bridge has all ports as Designated forwarding. This will however cause SW3 itself to transition FastEthernet0/19 into Loop Inconsistent. Initially, SW3's Root Port was FastEthernet0/19, thus the port was Non-Designated; upon enabling BPDU Filter on the port, as inbound BPDUs are filtered, port should transition to Desginated after the max_age expires, however Loop Guard will prevent this from happening, FastEthernet0/20 will be elected as the new Root Port:

```
SW3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW3(config)#interface fastEthernet0/19
SW3(config-if)#spanning-tree bpdufilter enable
!
!SW3#show spanning-tree interface fastEthernet0/19

Vlan                 Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- -------------------------------VLAN0001Desg BKN*
19        128.21P2p *LOOP_Inc
VLAN0002Desg BKN*19        128.21P2p *LOOP_Inc
```

https://t.me/learningnets

```
VLAN0005            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0007            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0008            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0009            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0010            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0022            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0043            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0058            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0067            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0079            Desg BKN*19        128.21   P2p *LOOP_Inc
VLAN0146            Desg BKN*19        128.21   P2p *LOOP_Inc
!
!SW3#show spanning-tree interface fastEthernet0/20

Vlan                Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------VLAN0001Root FWD
 19        128.22   P2p  VLAN0002Root FWD
 19        128.22   P2p
VLAN0005            Root FWD 19        128.22   P2p
VLAN0007            Root FWD 19        128.22   P2p
VLAN0008            Root FWD 19        128.22   P2p
VLAN0009            Root FWD 19        128.22   P2p
VLAN0010            Root FWD 19        128.22   P2p
VLAN0022            Root FWD 19        128.22   P2p
VLAN0043            Root FWD 19        128.22   P2p
VLAN0058            Root FWD 19        128.22   P2p
VLAN0067            Root FWD 19        128.22   P2p
VLAN0079            Root FWD 19        128.22   P2p
VLAN0146            Root FWD 19        128.22   P2p
```

SW3 will also log messages similar with the following, notifying of the problem:

```
%SPANTREE-2-LOOPGUARD_BLOCK: Loop guard blocking port FastEthernet0/19 on VLAN0001.
%SPANTREE-2-LOOPGUARD_BLOCK: Loop guard blocking port FastEthernet0/19 on VLAN0002.
```

Upon removing BPDU Filter on Fa0/19 of SW3, as inbound BPDUs are accepted, SW3 will remove the port from Loop Inconsistent state and negotiate the STP port state and role. Fa0/19 is re-elected as the Root Port:

```
SW3#show spanning-tree interface fastEthernet0/19

Vlan                Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------VLAN0001Root FWD
 19        128.21   P2p  VLAN0002Root FWD
 19        128.21   P2p
```

https://t.me/learningnets

```
VLAN0005            Root FWD 19        128.21   P2p
VLAN0007            Root FWD 19        128.21   P2p
VLAN0008            Root FWD 19        128.21   P2p
VLAN0009            Root FWD 19        128.21   P2p
VLAN0010            Root FWD 19        128.21   P2p
VLAN0022            Root FWD 19        128.21   P2p
VLAN0043            Root FWD 19        128.21   P2p
VLAN0058            Root FWD 19        128.21   P2p
VLAN0067            Root FWD 19        128.21   P2p
VLAN0079            Root FWD 19        128.21   P2p
VLAN0146            Root FWD 19        128.21   P2p
```

SW3 will also log messages similar with the following, notifying that port was recovered:

```
%SPANTREE-2-LOOPGUARD_UNBLOCK: Loop guard unblocking port FastEthernet0/19 on VLAN0001.
%SPANTREE-2-LOOPGUARD_UNBLOCK: Loop guard unblocking port FastEthernet0/19 on VLAN0002.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Unidirectional Link Detection

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure UDLD to prevent unidirectional links from forming on any of the inter-switch links in the Layer 2 network.

Configuration

```
SW1 - SW4:

interface range FastEthernet0/19 - 20
 udld port aggressive
!
interface range FastEthernet0/23 - 24
 udld port aggressive
```

Verification

UDLD, like Loop Guard, is used to prevent loops caused by unidirectional links. The difference between the features is that Loop Guard uses STP BPDUs to detect these failures, whereas UDLD uses its own keepalive mechanism. UDLD is a Cisco proprietary feature in which peers discover each other by exchanging frames sent to the well-known MAC address **01:00:0C:CC:CC:CC**. Each switch sends its own device ID along with the originator port ID and timeout value to its peer. Additionally,

https://t.me/learningnets

a switch echoes back the ID of its neighbor. If no echo frame with the switch's own ID has been seen from the peer for a certain amount of time, the port is suspected to be unidirectional. What happens next depends on UDLD mode of operation.

In **Normal** mode, if the physical state of port (as reported by Layer 1) is still up, UDLD marks this port as **Undetermined** but does NOT shutdown or disable the port, and it continues to operate under its current STP status. This mode of operation is informational and potentially less disruptive (although it does not prevent physical loops). If UDLD is set to **Aggressive** mode, when the switch loses its neighbor it actively tries to re-establish the relationship by sending a UDLD frames 8 times every 1 second. If the neighbor does not respond after that, the port is considered to be unidirectional and sent to err-disable state. The port is not automatically recovered unless UDLD err-disable recovery is configured.

In certain designs there are unidirectional links that Loop Guard can prevent and UDLD cannot, and likewise ones that UDLD can prevent but Loop Guard cannot. For example, if a loop occurs because of a physical wiring problem (for example, someone mistakenly mixes up the send and receive pairs of a fiber link), UDLD can detect this, but Loop Guard cannot. Likewise, if there is a unidirectional link caused by a failure in the STP software itself, although much more rare, Loop Guard can detect this but UDLD cannot. Based on this, the features can be configured at the same time to protect against all possible unidirectional link scenarios.

Although in this design UDLD is configured on copper UTP interfaces, this case is usually not needed in a real network design because of the Fast Link Pulse (FLP) signals that already track the interface status on wired interfaces. Instead, UDLD is more commonly run on fiber optic interfaces. UDLD can be enabled globally or at the port level in both modes, however if configured globally it only applies to fiber link ports. For this reason, we're enabling it at the port level on all switches.

Verify that UDLD is enabled, and neighbors have been discovered, for example betweem SW1 and SW3:

```
SW1#show udld fastethernet0/19


Interface Fa0/19
---
Port enable administrative configuration setting: Enabled / in aggressive mode
Port enable operational state: Enabled / in aggressive mode
Current bidirectional state: Bidirectional
Current operational state: Advertisement - Single neighbor detected
Message interval: 15000
Time out interval: 5000


    Entry 1
    ---
```

https://t.me/learningnets

```
    Expiration time: 42300
    Device ID: 1 Current neighbor state: Bidirectional
    Device name: FDO1227X2KE   Port ID: Fa0/19
    Neighbor echo 1 device: CAT0906R10C
    Neighbor echo 1 port: Fa0/19

    Message interval: 15
    Time out interval: 5
    CDP Device name: SW3
!
!SW3#show udld fastethernet0/19

Interface Fa0/19
---
Port enable administrative configuration setting: Enabled / in aggressive mode
Port enable operational state: Enabled / in aggressive mode
Current bidirectional state: Bidirectional
Current operational state: Advertisement - Single neighbor detected
Message interval: 15000
Time out interval: 5000

    Entry 1
    ---
    Expiration time: 34600
    Device ID: 1 Current neighbor state: Bidirectional
    Device name: CAT0906R10C   Port ID: Fa0/19
    Neighbor echo 1 device: FDO1227X2KE
    Neighbor echo 1 port: Fa0/19

    Message interval: 15
    Time out interval: 5
    CDP Device name: SW1
!
!SW1#show udld neighbors
Port     Device Name   Device ID     Port ID    Neighbor State
----     -----------   ---------     -------    --------------
Fa0/19   FDO1227X2KE     1            Fa0/19 Bidirectional
Fa0/20   FDO1227X2KE     1            Fa0/20 Bidirectional
Fa0/23   CAT1025NMR8     1            Fa0/23 Bidirectional
Fa0/24   CAT1025NMR8     1            Fa0/24 Bidirectional
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

MST Root Bridge Election

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Multiple Spanning-Tree on SW1 - SW4 as follows:
  - Use region named **MST** and a revision of **1**.
  - Instance 1 should service VLANs 1 - 100.
  - Instance 2 should service VLANs 101 - 200.
  - Instance 3 should service all other VLANs.
- Ensure the following Root Bridge selection:
  - Configure SW1 as the STP Root Bridge for instance 1.
  - Configure SW4 as the STP Root Bridge for instance 2.
  - If SW1 goes down, SW2 should become the STP Root Bridge for instance 1.
  - If SW4 goes down, SW3 should become the STP Root Bridge for instance 2.
  - Use the lowest priority values to achieve task requirements.

Configuration

```
SW1:
spanning-tree mst configuration
 name MST
 revision 1
 instance 1 vlan 1-100
 instance 2 vlan 101-200
 instance 3 vlan 201-4094
!
```

https://t.me/learningnets

```
spanning-tree mode mst
spanning-tree mst 1 priority 0
SW2:
spanning-tree mst configuration
 name MST
 revision 1
 instance 1 vlan 1-100
 instance 2 vlan 101-200
 instance 3 vlan 201-4094
!
spanning-tree mode mst
spanning-tree mst 1 priority 4096
SW3:
spanning-tree mst configuration
 name MST
 revision 1
 instance 1 vlan 1-100
 instance 2 vlan 101-200
 instance 3 vlan 201-4094
!
spanning-tree mode mst
spanning-tree mst 2 priority 4096
SW4:

spanning-tree mst configuration
 name MST
 revision 1
 instance 1 vlan 1-100
 instance 2 vlan 101-200
 instance 3 vlan 201-4094
!
spanning-tree mode mst
spanning-tree mst 2 priority 0
```

Verification

Multiple Spanning-Tree (MST) is an IEEE standard defined in 802.1s which allows user-defined STP instances to be mapped to multiple VLANs. Unlike the Cisco proprietary Per-VLAN Spanning-Tree (PVST), MST can be used to eliminate the overhead of redundant STP instances in topologies where multiple VLANs, but not all VLANs, follow the same layer 2 forwarding path, while at the same time allowing for flexible failure domain separation and traffic engineering. MST essentially takes the best features of IEEE 802.1D Spanning-Tree, AKA Common Spanning-Tree,

https://t.me/learningnets

and the Cisco extensions to STP, PVST, PVST+, Rapid PVST+, and combines them.

For example, in this design STP instances are created for VLANs 1-4094. In Common Spanning-Tree, all 4094 VLANs would map to one instance. This has very little overhead but does not allow for detailed traffic engineering. With PVST, there would be 4094 separate instances of STP, which allows for detailed traffic engineering but creates immense overhead in the control-plane. With MST, three user-defined instances are created that map different portions of the VLAN space into separate instances with a similar forwarding path.

Like CST and PVST, MST uses the lowest Bridge-ID (BID) in the network to elect the Root Bridge. The BID is made up of the priority value and the MAC address. The lower priority wins the election, and if there is a tie in priority the lowest MAC address is the tie breaker. In PVST, there is one root bridge election per VLAN, because there is one STP instance per VLAN, but in MST there is one election per user-defined instance.

From the `show spanning-tree mst` output, we can see which VLANs are mapped to the particular MST instance, who the root bridge is, and how the root port election has occurred. In this case, SW1 is the root for instance 1, whereas SW4 is the root for instance 2. SW1 is the root for instance 1 because it has a priority value of 1, which is made up of the configured priority of 0 plus the sytem-id extension of 1. In MST the sysid field is the instance number, whereas in PVST the sysid is the VLAN number.

Verify the MST configuration and that switches run in MST mode, for example on SW1:

```
SW1#show spanning-tree mst configuration
Name      [MST]
Revision  1     Instances configured 4

Instance  Vlans mapped
--------  -------------------------------------------------------------------
0         none
1         1-100
2         101-200
3         201-4094
---------------------------------------------------------------------------
!
!SW1#show spanning-tree summary
Switch is in mst mode (IEEE Standard)

Root bridge for: MST1
Extended system ID           is enabled
```

https://t.me/learningnets

```
Portfast Default             is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short (Operational value is long)

Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
MST0                          1         0        0          4          5
MST1                          0         0        0          5          5
MST2                          3         0        0          1          4
---------------------- -------- --------- -------- ---------- ----------
3 msts                        4         0        0         10         14
```

Verify that SW1 is Root Bridge for instance 1 and SW4 is Root Bridge for instance 2, thus both have all ports as Designated Forwarding for the respective instance:

https://t.me/learningnets

```
SW1#show spanning-tree mst 1

##### MST1    vlans mapped:   1-100
Bridge        address 0013.605f.f000  priority      1     (0 sysid 1)
Root          this switch for MST1


Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/1            Desg FWD 200000    128.3    P2p
Fa0/19           Desg FWD 200000    128.21   P2p
Fa0/20           Desg FWD 200000    128.22   P2p
Fa0/23           Desg FWD 200000    128.25   P2p
Fa0/24           Desg FWD 200000    128.26   P2p
!
!SW4#show spanning-tree mst 2

##### MST2    vlans mapped:   101-200
Bridge        address 001a.a174.2500  priority      2     (0 sysid 2)
Root          this switch for MST2


Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Desg FWD 200000    128.21   P2p
Fa0/20           Desg FWD 200000    128.22   P2p
Fa0/23           Desg FWD 200000    128.25   P2p
Fa0/24           Desg FWD 200000    128.26   P2p
```

MSTP does not generate per-instance BPDUs as RSTP did with per-VLAN BPDUs. Instead, it uses a single BPDU which has additional records to advertise STP data for all configured instances. Even though with our configuration no VLANs are assigned to the default MST instance of 0 (to which by default all VLANs are assigned), STP still runs for this instance, as STP actually runs over instance zero which forms the IST, or CIST in the case of multiple MST regions. As we did not modify the default MST priority for instance zero, in this case SW2 was selected as the CIST Root Bridge and also the Regional Root as MST is configured for a single region, but this may depend on your rack assignment:

```
SW2#show spanning-tree mst 0
##### MST0    vlans mapped:   none
Bridge        address 000a.b832.3a80  priority      32768 (32768 sysid 0)
Root          this switch for the CIST)
```

https://t.me/learningnets

```
Operational   hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured    hello time 2 , forward delay 15, max age 20, max hops    20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Desg FWD 200000    128.21   P2p
Fa0/20           Desg FWD 200000    128.22   P2p
Fa0/23           Desg FWD 200000    128.25   P2p
Fa0/24           Desg FWD 200000    128.26   P2p
!
!SW1#show spanning-tree mst
##### MST0    vlans mapped:   none
Bridge        address 0013.605f.f000  priority      32768 (32768 sysid 0)
Root          address 000a.b832.3a80  priority      32768 (32768 sysid 0)
port    Fa0/23          path cost     0
        Regional Root address 000a.b832.3a80  priority      32768 (32768 sysid 0)
                                      internal cost 200000    rem hops 19
Operational   hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured    hello time 2 , forward delay 15, max age 20, max hops    20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/1            Desg FWD 200000    128.3    P2p
Fa0/19           Desg FWD 200000    128.21   P2p
Fa0/20           Desg FWD 200000    128.22   P2p
Fa0/23           Root FWD 200000    128.25   P2p
Fa0/24           Altn BLK 200000    128.26   P2p
##### MST1    vlans mapped:   1-100
Bridge        address 0013.605f.f000  priority      1     (0 sysid 1)Root          this switch for MST1

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/1            Desg FWD 200000    128.3    P2p
Fa0/19           Desg FWD 200000    128.21   P2p
Fa0/20           Desg FWD 200000    128.22   P2p
Fa0/23           Desg FWD 200000    128.25   P2p
Fa0/24           Desg FWD 200000    128.26   P2p
##### MST2    vlans mapped:   101-200
Bridge        address 0013.605f.f000  priority      32770 (32768 sysid 2)
Root          address 001a.a174.2500  priority      2     (0 sysid 2)
port    Fa0/19          cost          400000    rem hops 18


Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Root FWD 200000    128.21   P2p
```

https://t.me/learningnets

```
Fa0/20           Altn BLK 200000    128.22   P2p
Fa0/23           Altn BLK 200000    128.25   P2p
Fa0/24           Altn BLK 200000    128.26   P2p
```

Test that if SW1 is offline, SW2 becomes the Root Bridge for instance 1:

```
SW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#interface range fastEthernet0/19 - 20, fastEthernet0/23 - 24
SW1(config-if-range)#shutdown
!
!SW2#show spanning-tree mst 1

##### MST1    vlans mapped:   1-100
Bridge         address 000a.b832.3a80  priority      4097  (4096 sysid 1)
Root           this switch for MST1


Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Desg FWD 200000    128.21   P2p
Fa0/20           Desg FWD 200000    128.22   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

MST Path Selection with Port Cost

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Multiple Spanning-Tree on SW1 - SW4 as follows:
  - Use region named **MST** and a revision of **1**.
  - Instance 1 should service VLANs 1 - 100.
  - Instance 2 should service VLANs 101 - 200.
  - Configure SW1 as the STP Root Bridge for instance 1, using the lowest priority value.
- Using Spanning-Tree cost, modify the Layer 2 transit network so that traffic for MST instance 1 from SW2 to SW1 uses the last link between SW2 and SW4.
  - If this link goes down, traffic should fall over to the remaining link between SW2 and SW4.

Configuration

```
SW1 - SW4:
 spanning-tree mst configuration
  name MST
  revision 1
  instance 1 vlan 1-100
  instance 2 vlan 101-200
 !
 spanning-tree mode mst
 SW1:
```

https://t.me/learningnets

```
spanning-tree mst 1 priority 0
SW2:

interface range FastEthernet0/23 - 24
 spanning-tree mst 1 cost 500000
!
interface FastEthernet0/19
 spanning-tree mst 1 cost 2
!
interface FastEthernet0/20
 spanning-tree mst 1 cost 1
```

Verification

Similar to CST and PVST, MST uses a cost value derived from the inverse bandwidth of the interface (higher bandwidth means lower cost). The root port is chosen based on the lowest end-to-end cost to the root bridge. The **`show spanning-tree mst`** command shows the local cost values of the outgoing ports on the local switch. Verify that initially, SW2's Root Port for instance 1 is the direct Fa0/23 link to SW1:

```
SW2#show spanning-tree mst 1

##### MST1    vlans mapped:   1-100
Bridge        address 000a.b832.3a80  priority      4097  (4096 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
              port    Fa0/23          cost          200000    rem hops 19

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Desg FWD 200000    128.21   P2p
Fa0/20           Desg FWD 200000    128.22   P2p Fa0/23           Root FWD 200000    128.25   P2p

Fa0/24           Altn BLK 200000    128.26   P2p
```

Configure the STP cost on SW2 and verify now that SW2's Root Port changed to its Fa0/20 link towards SW4:

```
SW2#show spanning-tree mst 1

##### MST1    vlans mapped:   1-100
Bridge        address 000a.b832.3a80  priority      4097  (4096 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
```

https://t.me/learningnets

```
              port    Fa0/20          cost          400001    rem hops 17

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Altn BLK 2
         128.21   P2p Fa0/20           Root FWD 1         128.22   P2p

Fa0/23           Altn BLK 500000    128.25   P2p
Fa0/24           Altn BLK 500000    128.26   P2p
```

To see the entire end-to-end cost of a path, the **`show spanning-tree mst 1 detail`** command should be used. The end-to-end cost is made up of the upstream (designated) cost plus the local port cost. In this output, the alternate ports Fa0/23 – Fa0/24 have a total cost of 500.000 because of the manual cost change. Fa0/19 has a total cost of 400.002, which is 2 to SW4, 200.000 from SW4 to SW3, and 200.000 from SW3 to SW1. Fa0/20 wins the Root Port election because it has a total cost of 400.001.

```
SW2#show spanning-tree mst 1 detail

##### MST1    vlans mapped:   1-100
Bridge        address 000a.b832.3a80  priority      4097  (4096 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
              port    Fa0/20          cost          400001    rem hops 17
 FastEthernet0/19 of MST1 is alternate blocking
Port info             port id         128.21  priority    128  cost           2
Designated root       address 0013.605f.f000  priority      1  cost      400000
Designated bridge     address 001a.a174.2500  priority  32769  port id   128.21
Timers: message expires in 5 sec, forward delay 0, forward transitions 2
Bpdus (MRecords) sent 147, received 142
 FastEthernet0/20 of MST1 is root forwarding
Port info             port id         128.22  priority    128  cost           1
Designated root       address 0013.605f.f000  priority      1  cost      400000

Designated bridge     address 001a.a174.2500  priority  32769  port id   128.22
Timers: message expires in 5 sec, forward delay 0, forward transitions 2
Bpdus (MRecords) sent 146, received 142

FastEthernet0/23 of MST1 is alternate blocking
Port info             port id         128.25  priority    128  cost      500000
Designated root       address 0013.605f.f000  priority      1  cost           0
Designated bridge     address 0013.605f.f000  priority      1  port id   128.25
Timers: message expires in 5 sec, forward delay 0, forward transitions 2
Bpdus (MRecords) sent 146, received 145
```

https://t.me/learningnets

```
FastEthernet0/24 of MST1 is alternate blocking
Port info             port id         128.26  priority    128  cost      500000
Designated root       address 0013.605f.f000  priority      1  cost           0
Designated bridge     address 0013.605f.f000  priority      1  port id   128.26
Timers: message expires in 5 sec, forward delay 0, forward transitions 0
Bpdus (MRecords) sent 147, received 144
```

When SW2's port Fa0/20 is down, the next lowest cost path is 400.002 through Fa0/19:

```
SW2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW2(config)#interface fastEthernet0/20
SW2(config-if)#shutdown
!
!SW2#show spanning-tree mst 1

##### MST1    vlans mapped:   1-100
Bridge        address 000a.b832.3a80  priority      4097  (4096 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
              port    Fa0/19          cost          400002    rem hops 17

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------Fa0/19            Root FWD
 2         128.21   P2p
Fa0/23           Altn BLK 500000    128.25   P2p
Fa0/24           Altn BLK 500000    128.26   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

MST Path Selection with Port Priority

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Multiple Spanning-Tree on SW1 - SW4 as follows:
  - Use region named MST and a revision of 1.
  - Instance 1 should service VLANs 1 - 100.
  - Instance 2 should service VLANs 101 - 200.
  - Configure SW1 as the STP Root Bridge for instance 1, using the lowest priority value.
  - Configure SW2 with the next available priority value for instance 1, so that it becomes root bridge in case of SW1 failure.
- Using Spanning-Tree priority, modify the Layer 2 transit network so that traffic for MST instance 1 from SW4 to SW1 uses the last link between SW2 and SW4.
  - If this link goes down, traffic should fall over to the remaining link between SW2 and SW4.

Configuration

```
SW1 - SW4:
spanning-tree mst configuration
 name MST
 revision 1
 instance 1 vlan 1-100
 instance 2 vlan 101-200
```

https://t.me/learningnets

```
!
spanning-tree mode mst
SW1:
spanning-tree mst 1 priority 0
SW2:

spanning-tree mst 1 priority 4096
!
interface FastEthernet0/20
 spanning-tree mst 1 port-priority 0
```

Verification

Like CST and PVST, MST uses the designated (upstream) port-priority as a tie breaker if the end-to-end cost is the same on multiple ports to the same upstream switch. Before changing the port-priority on SW2, note that SW4 uses its Fa0/19 as the Root Port for instance 1:

```
SW4#show spanning-tree mst 1

##### MST1    vlans mapped:   1-100
Bridge        address 001a.a174.2500  priority      32769 (32768 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
              port    Fa0/19          cost          400000    rem hops 18

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Root FWD 200000    128.21   P2p

Fa0/20           Altn BLK 200000    128.22   P2p
Fa0/23           Altn BLK 200000    128.25   P2p
Fa0/24           Altn BLK 200000    128.26   P2p
```

The `show spanning-tree mst 1` only shows the local port-priority, so the output below doesn't tell us why Fa0/19 is chosen as the Root Port. The `show spanning-tree mst 1 detail` shows that the lowest end-to-end cost of 400.000 is equal on all its ports. Because all of these ports share the same cost, the next differentiator is the lowest upstream bridge priority, which in this case is SW2 with its configured priority of 4096. As SW4 has two links towards SW2 the designated port-id is checked. The port-id is made of the port-priority and the internally assigned port number. Fa0/19 has the lowest designated port-id of 128.21, versus Fa0/20 with port-id of 128.22:

https://t.me/learningnets

```
SW4#show spanning-tree mst 1 detail


##### MST1    vlans mapped:   1-100
Bridge        address 001a.a174.2500  priority      32769 (32768 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
              port    Fa0/19          cost          400000    rem hops 18
FastEthernet0/19 of MST1 is root forwarding
Port info             port id         128.21  priority    128 cost      200000
Designated root       address 0013.605f.f000  priority      1 cost      200000
Designated bridge     address 000a.b832.3a80  priority   4097  port id   128.21
Timers: message expires in 5 sec, forward delay 0, forward transitions 2
Bpdus (MRecords) sent 660, received 663
FastEthernet0/20 of MST1 is alternate blocking
Port info             port id         128.22  priority    128 cost      200000
Designated root       address 0013.605f.f000  priority      1 cost      200000
Designated bridge     address 000a.b832.3a80  priority   4097  port id   128.22

Timers: message expires in 5 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 366, received 367

FastEthernet0/23 of MST1 is alternate blocking
Port info             port id         128.25  priority    128  cost      200000
Designated root       address 0013.605f.f000  priority      1  cost      200000
Designated bridge     address 0022.5627.1f80  priority  32769  port id   128.25
Timers: message expires in 5 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 671, received 650

FastEthernet0/24 of MST1 is alternate blocking
Port info             port id         128.26  priority    128  cost      200000
Designated root       address 0013.605f.f000  priority      1  cost      200000
Designated bridge     address 0022.5627.1f80  priority  32769  port id   128.26
Timers: message expires in 5 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 668, received 650
```

Configure the port priority on SW2 as in the solution, based on this SW4 should select its Fa0/20 as the new Root Port, verify that priority changed:

```
SW4#show spanning-tree mst 1


##### MST1    vlans mapped:   1-100
Bridge        address 001a.a174.2500  priority      32769 (32768 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
              port    Fa0/20          cost          400000    rem hops 18
```

https://t.me/learningnets

```
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/19           Altn BLK 200000    128.21   P2p Fa0/20           Root FWD 200000    128.22   P2p
Fa0/23           Altn BLK 200000    128.25   P2p
Fa0/24           Altn BLK 200000    128.26   P2p
!
!SW4#show spanning-tree mst 1 detail

##### MST1    vlans mapped:   1-100
Bridge        address 001a.a174.2500  priority      32769 (32768 sysid 1)
Root          address 0013.605f.f000  priority      1     (0 sysid 1)
              port    Fa0/20          cost          400000    rem hops 18
 FastEthernet0/19 of MST1 is alternate blocking
Port info             port id         128.21  priority    128  cost      200000
Designated root       address 0013.605f.f000  priority      1  cost      200000
Designated bridge     address 000a.b832.3a80  priority   4097  port id   128.21
Timers: message expires in 4 sec, forward delay 0, forward transitions 2
Bpdus (MRecords) sent 913, received 915
 FastEthernet0/20 of MST1 is root forwarding
Port info             port id         128.22  priority    128  cost      200000
Designated root       address 0013.605f.f000  priority      1  cost      200000
Designated bridge     address 000a.b832.3a80  priority   4097  port id     0.22

Timers: message expires in 4 sec, forward delay 0, forward transitions 2
Bpdus (MRecords) sent 619, received 618

FastEthernet0/23 of MST1 is alternate blocking
Port info             port id         128.25  priority    128  cost      200000
Designated root       address 0013.605f.f000  priority      1  cost      200000
Designated bridge     address 0022.5627.1f80  priority  32769  port id   128.25
Timers: message expires in 4 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 923, received 902

FastEthernet0/24 of MST1 is alternate blocking
Port info             port id         128.26  priority    128  cost      200000
Designated root       address 0013.605f.f000  priority      1  cost      200000
Designated bridge     address 0022.5627.1f80  priority  32769  port id   128.26
Timers: message expires in 4 sec, forward delay 0, forward transitions 1
Bpdus (MRecords) sent 920, received 902
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

MST and Rapid Spanning Tree

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure Multiple Spanning-Tree on SW1 - SW4 as follows:
  - Use region named MST and a revision of 1.
  - Instance 1 should service VLANs 1 - 100.
  - Instance 2 should service VLANs 101 - 200.
- Configure PortFast on SW1's FastEthernet0/1 so that it immediately begins forwarding when enabled.

Configuration

https://t.me/learningnets

```
SW1 - SW4:

spanning-tree mst configuration
 name MST1
 revision 1
 instance 1 vlan 1-100
 instance 2 vlan 101-200
!
spanning-tree mode mst
SW1:

interface FastEthernet0/1
 spanning-tree portfast
```

Verification

When MST is enabled, Rapid Spanning-Tree Protocol (RSTP) is automatically enabled. RSTP is an IEEE standard defined in 802.1w that speeds up convergence through a reliable handshaking process. RSTP defines new port roles to automatically allow for the functionality built into Cisco proprietary features such as PortFast and UplinkFast.

RSTP Edge ports behave the same as PVST PortFast-enabled ports. However, to maintain backward-compatible configurations, Cisco's implementation of RSTP does not automatically elect edge ports as the standard suggests. Instead, a port must be configured as an edge port with the `spanning-tree portfast` command:

```
SW1#show spanning-tree interface fastEthernet0/1

Mst Instance        Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
MST0                Desg FWD 200000    128.3 P2p Edge
MST1                Desg FWD 200000    128.3 P2p Edge
!
!SW1#show spanning-tree mst interface fastEthernet0/1

FastEthernet0/1 of MST0 is designated forwarding Edge port: edge            (enable)
         port guard : none        (default)
Link type: point-to-point (auto)           bpdu filter: disable     (default)
Boundary : internal                        bpdu guard : disable     (default)
Bpdus sent 2954, received 0

Instance Role Sts Cost      Prio.Nbr Vlans mapped
```

https://t.me/learningnets

```
-------- ---- --- --------- -------- ------------------------------
0        Desg FWD 200000    128.3    none
1        Desg FWD 200000    128.3    1-100
!
!SW1#show spanning-tree interface fastEthernet0/1 portfast
MST0                 enabled
MST1                 enabled
```

If there were a device connected to FastEthernet0/1 such as a hypervisor that could tag frames with dot1q tags, we could configure this port as a trunk port and also configure it as an Edge port. This will ensure that it immediately begins forwarding when enabled for all VLANs being trunked:

```
SW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW1(config)#interface FastEthernet0/1
SW1(config-if)#switchport trunk encapsulation dot1q
SW1(config-if)#switchport mode trunk
 SW1(config-if)#spanning-tree portfast trunk

%Warning: portfast should only be enabled on ports connected to a single
 host. Connecting hubs, concentrators, switches, bridges, etc... to this
 interface  when portfast is enabled, can cause temporary bridging loops.
 Use with CAUTION
!
!SW1#show spanning-tree interface fastEthernet0/1

Mst Instance        Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
MST0                Desg FWD 200000    128.3 P2p Edge
MST1                Desg FWD 200000    128.3 P2p Edge
MST2                Desg FWD 200000    128.3 P2p Edge
!
!SW1#show spanning-tree mst interface fastEthernet0/1

FastEthernet0/1 of MST0 is designated forwarding Edge port: edge            (trunk)
          port guard : none        (default)
Link type: point-to-point (auto)           bpdu filter: disable     (default)
Boundary : internal                        bpdu guard : disable     (default)
Bpdus sent 9, received 0

Instance Role Sts Cost      Prio.Nbr Vlans mapped
-------- ---- --- --------- -------- ------------------------------
0        Desg FWD 200000    128.3    none
1        Desg FWD 200000    128.3    1-100
```

https://t.me/learningnets

```
2        Desg FWD 200000    128.3    101-200
!
!SW1#show spanning-tree interface fastEthernet0/1 portfast
MST0                 enabled
MST1                 enabled
MST2                 enabled
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Protected Ports

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Shutdown all Ethernet links from SW4 towards SW2 and SW3.
- Shutdown ports Fa0/19 and Fa0/23 on SW1.
- Create an SVI for VLAN 10 on SW1, assign it the IP address of 169.254.23.1/24.
- Configure Fa0/20 and Fa0/24 on SW1 as access ports in VLAN 10.
- Configure port Fa0/20 on SW3 and Fa0/24 on SW2 as Layer 3 ports on with IP addresses of **169.254.23.Y/24**, where **Y** is the switch number.
- Configure port protection on SW1 so that SW2 and SW3 cannot directly communicate with each other, but can communicate with SW1's VLAN 10 interface.

Configuration

```
SW4:
interface range FastEthernet0/19 - 20
 shutdown
!
interface range FastEthernet0/23 - 24
 shutdown
SW1:
interface range FastEthernet0/19 , FastEthernet0/23
 shutdown
!
default interface range FastEthernet0/20 , FastEthernet0/24
```

https://t.me/learningnets

```
!
interface range FastEthernet0/20 , FastEthernet0/24
 switchport mode access
 switchport access vlan 10
 switchport protected
!
interface Vlan10
 ip address 169.254.23.1 255.255.255.0
 no shutdown
SW2:
interface FastEthernet0/24
 no switchport
 ip address 169.254.23.2 255.255.255.0
SW3:

interface FastEthernet0/20
 no switchport
 ip address 169.254.23.3 255.255.255.0
```

Verification

Protected ports are used to prevent traffic from being directly exchanged at Layer 2 between two or more hosts that are within the same VLAN. Traffic received in a protected port cannot be sent out another protected port, but traffic received in a protected port can be sent out a non-protected port. This feature is a much smaller subset of the Private VLAN feature, and it cannot span between multiple physical switches; you cannot configure a protected port on SW1 and a protected port on SW2 and expect traffic between these to be disallowed.

In this particular design, the result of port protection is that SW1 and SW2 can communicate, SW1 and SW3 can communicate, but SW2 and SW3 cannot communicate, although are attached to the same VLAN:

```
SW3#ping 169.254.23.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/8 ms
!
!SW3#ping 169.254.23.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.2, timeout is 2 seconds:.....
```

https://t.me/learningnets

```
Success rate is 0 percent (0/5)
!
!SW2#ping 169.254.23.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
!
!SW2#ping 169.254.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.3, timeout is 2 seconds:.....

Success rate is 0 percent (0/5)
```

Notice that ARP traffic is also not allowed between protected ports, basically all traffic is dropped:

```
SW1#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  169.254.23.1            -   0013.605f.f041  ARPA   Vlan10
Internet  169.254.23.2            0   000a.b832.3ac1  ARPA   Vlan10
Internet  169.254.23.3            9   0022.5627.1fc1  ARPA   Vlan10
!
!SW2#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  169.254.23.1           18   0013.605f.f01a  ARPA   FastEthernet0/24
Internet  169.254.23.2            -   000a.b832.3ac1  ARPA   FastEthernet0/24
Internet  169.254.23.3            0   Incomplete      ARPA
!
!SW3#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  169.254.23.1           18   0013.605f.f016  ARPA   FastEthernet0/20
Internet  169.254.23.2            0   Incomplete      ARPA

Internet  169.254.23.3            -   0022.5627.1fc1  ARPA   FastEthernet0/20
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Storm Control

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure SW1 to limit unicast traffic received from SW2 to 100 pps.
- Configure SW2 to limit broadcast traffic received from SW4 to 10Mbps.
- Configure SW4 to limit broadcast traffic received from SW1 to 1Mbps using a relative percentage of the interface bandwidth.

Configuration

```
SW1:
interface range FastEthernet0/23 - 24
 storm-control unicast level pps 100
SW2:
interface range FastEthernet0/19 - 20
 storm-control broadcast level bps 10m
SW4:

interface range FastEthernet0/19 - 20
 storm-control broadcast level 1
```

Verification

Storm control is used to limit the amount of unicast, multicast, or broadcast traffic

https://t.me/learningnets

received inbound on a port. The most common application of this feature is to prevent broadcast storms, but it can also be used to police individual ports not to exceed a desired rate. Depending on the version of IOS, the **`storm-control`** command may take units in percentage, packets per second, bits per second, or others. Make sure to use the question mark when implementing this command so that the units entered achieve the desired result.

```
SW2#show storm-control
Interface  Filter State   Upper        Lower        Current
---------  -------------  -----------  -----------  ----------
Fa0/19     Forwarding         10m bps      10m bps        0 bps
Fa0/20     Forwarding         10m bps      10m bps        0 bps
!
!SW4#show storm-control

Interface  Filter State   Upper        Lower        Current
---------  -------------  -----------  -----------  ----------
Fa0/19     Forwarding        1.00%        1.00%        0.00%
  Fa0/20     Forwarding        1.00%        1.00%        0.00%
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

MAC-Address Table Static Entries and Aging

> You must load the initial configuration files for the section, **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Shut down all Ethernet links from SW4 toward SW2 and SW3.
- Shut down ports Fa0/19 and Fa0/23 on SW1.
- Create an SVI for VLAN 10 on SW1 and assign it the IP address 169.254.23.1/24.
- Configure Fa0/20 and Fa0/24 on SW1 as access ports in VLAN 10.
- Configure port Fa0/20 on SW3 and Fa0/24 on SW2 as Layer 3 ports with IP addresses of **169.254.23.*Y*/24**, where **Y** is the switch number.
- Configure static CAM entries on SW1 as follows:
  - Frames destined to the MAC address of SW2's Layer 3 interface are dropped.
  - SW3's MAC address is not allowed to commute between ports or switches.

Configuration

```
SW4:
interface range FastEthernet0/19 - 20 , FastEthernet0/23 - 24
 shutdown
SW1:
interface range FastEthernet0/19 , FastEthernet0/23
 shutdown
!
default interface range FastEthernet0/20 , FastEthernet0/24
!
```

https://t.me/learningnets

```
interface range FastEthernet0/20 , FastEthernet0/24
 switchport mode access
 switchport access vlan 10
!
interface Vlan10
 ip address 169.254.23.1 255.255.255.0
 no shutdown
!
mac address-table static 0022.5627.1fc1 vlan 10 interface FastEthernet0/20
mac address-table static 000a.b832.3ac1 vlan 10 drop
SW2:
interface FastEthernet0/24
 no switchport
 ip address 169.254.23.2 255.255.255.0
SW3:

interface FastEthernet0/20
 no switchport
 ip address 169.254.23.3 255.255.255.0
```

Verification

Normally, switches populate the CAM table, or MAC address table, by flooding unknown frames everywhere in the VLAN in which they were received and by looking at the source MAC address of frames received in its ports. In certain circumstances this can be undesirable, such as when someone attempts to do a Layer 2 MAC address spoofing attack. A simple way to prevent these types of attacks is to statically hard-code which MAC addresses are reachable via which ports.

Another static feature of the CAM table is the ability to Null route MAC addresses. Because static entries always override dynamically learned entries, if the drop keyword or an unused interface is used in the **`mac address-table static`** command, traffic destined to that MAC address will be silently dropped.

In this particular design, SW1, SW2, and SW3 exchange traffic on VLAN 10. Before configuring static MAC entries, SW1 has connectivity with both SW2 and SW3.

```
SW1#ping 169.254.23.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
!
!SW1#ping 169.254.23.3
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.3, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/9 ms
```

SW1 dynamically learns the MAC addresses of both SW2 and SW3.

```
SW1#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  169.254.23.1            -   0013.605f.f041  ARPA   Vlan10 Internet
169.254.23.2           46   000a.b832.3ac1
  ARPA   Vlan10 Internet 169.254.23.3           46   0022.5627.1fc1
  ARPA   Vlan10
!
!SW1#show mac address-table dynamic address 000a.b832.3ac1
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    ----- 10    000a.b832.3ac1    DYNAMIC     Fa0/24
Total Mac Addresses for this criterion: 1
!
!SW1#show mac address-table dynamic address 0022.5627.1fc1
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    ----- 10    0022.5627.1fc1    DYNAMIC     Fa0/20

Total Mac Addresses for this criterion: 1
```

After SW1 is configured with static entries for both SW2 and SW3, these will override the dynamically learned ones. The result is that any traffic destined to SW2 is dropped in the Layer 2 transit path by SW1.

```
SW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#mac address-table static 000a.b832.3ac1 vlan 10 drop
!
!SW1#ping 169.254.23.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.2, timeout is 2 seconds:.....
Success rate is 0 percent (0/5)
!
!SW1#show mac address-table address 000a.b832.3ac1
```

https://t.me/learningnets

```
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    ----- 10    000a.b832.3ac1    STATIC       Drop

Total Mac Addresses for this criterion: 1
```

Likewise, as soon as we add the static entry for SW3's Layer 3 interface, traffic going to SW3 uses the static entry instead of the dynamically learned entry.

```
SW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#mac address-table static 0022.5627.1fc1 vlan 10 interface FastEthernet0/20
!
!SW1#ping 169.254.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 169.254.23.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
!
!SW1#show mac address-table address 0022.5627.1fc1
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----  10    0022.5627.1fc1 STATIC
      Fa0/20
Total Mac Addresses for this criterion: 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

SPAN

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure SW1 so that all traffic transiting VLAN 146 is redirected to a host located on port Fa0/24.
- Configure SW4 so that all interface Fa0/8 traffic is redirected to a host located on port Fa0/24.
  - Untagged inbound traffic from the host should be placed into VLAN 146.

Configuration

```
SW1:
monitor session 1 source vlan 146
monitor session 1 destination interface FastEthernet0/24
SW4:

monitor session 1 source interface FastEthernet0/8
monitor session 1 destination interface FastEthernet0/24 ingress vlan 146
```

Verification

The Switchport Analyzer (SPAN) feature is used to redirect traffic from a port or VLAN onto another port for analysis by devices such as a packet sniffer or Intrusion Prevention Sensor (IPS). There are three variations of SPAN: Local SPAN (or just

https://t.me/learningnets

SPAN), Remote SPAN (or RSPAN), and Encapsulated Remote SPAN (or ERSPAN). ERSPAN is only supported on high end platforms, such as the Cisco 6500/7600 or Nexus 7000. Instead of having the destination of the SPAN be a local port (SPAN) or a VLAN (SPAN), ERSPAN can send the traffic to be analyzed over a Layer 3 network using GRE encapsulation.

With Local SPAN, as shown in this design, traffic coming from or going to a particular port is redirected to another local port. The source of traffic can also be a VLAN, as shown on SW1. Normally when the SPAN feature is configured, the switch drops all traffic coming back in from the SPAN destination port. The **`ingress`** keyword tells the switch to accept inbound traffic from a SPAN destination port and assign the traffic to a particular VLAN.

```
SW1#show monitor session 1
Session 1
---------
Type                    : Local Session
Source VLANs            :     Both                : 146
Destination Ports       : Fa0/24
    Encapsulation       : Native Ingress       : Disabled
!
!SW4#show monitor session 1
Session 1
---------
Type                    : Local Session
Source Ports            :      Both                : Fa0/8
Destination Ports       : Fa0/24
    Encapsulation       : Native Ingress       : Enabled, default VLAN = 146

    Ingress encap : Untagged
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

RSPAN

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Disable the Ethernet links between SW1 and SW2.
- Create VLAN 500 as an RSPAN VLAN on all switches in the topology.
- Configure Fa0/5 on SW2 to be an access port on VLAN 43 and redirect all traffic on this port to the RSPAN VLAN.
- Configure SW1 to capture traffic on RSPAN VLAN and redirect it to a host connected to port Fa0/24.
  - Accept inbound tagged traffic for VLAN 146.

Configuration

```
SW1:
interface range FastEthernet0/23 - 24
 shutdown
!
vlan 500
 remote-span
!
monitor session 2 destination interface Fa0/24 ingress dot1q vlan 146
monitor session 2 source remote vlan 500
SW2:

interface range FastEthernet0/23 - 24
```

https://t.me/learningnets

```
 shutdown
!
interface FastEthernet0/5
 switchport mode access
 switchport access vlan 43
!
monitor session 2 source interface FastEthernet0/5
monitor session 2 destination remote vlan 500
```

Verification

The Remote SPAN, or RSPAN, feature is used when the source port or VLAN that is being monitored is on a different physical switch than the destination sniffer or sensor. The SPAN session can be spanned across multiple switches (a Layer 2 network). With ERSPAN, the SPAN session can be sent across a routed Layer 3 network.

The first step in configuring RSPAN is to ensure that the switches in the Layer 2 transit path from the source port/VLAN to the destination port are trunking at Layer 2, and know about the RSPAN VLAN that is used to encapsulate and transport the monitored traffic. In this case VTP is used, so only the VTP server SW1 needs to create the VLAN. Note the remote-span keyword under the VLAN: this is a special attribute that affects how traffic is processed when it is received in this VLAN.

Next, the switch attached to the source port or VLAN creates a SPAN session. The source of this span session, in the case of SW2, is all traffic on port Fa0/5. The destination of the session is the RSPAN VLAN 500 itself. This means that all traffic on port Fa0/5 will receive a new 802.1q header with a VLAN 500 tag and be sent out in the Layer 2 network.

Finally, the switch attached to the sniffer/sensor creates a SPAN session with the source as the RSPAN VLAN, and the destination as the local port. This means that the switch wants to listen for all traffic received in the RSPAN VLAN and redirect it out a local port. In this case, SW1 says that the source of the session is the remote VLAN 500. On SW1, therefore, all traffic coming in a trunk link with a tag of 500 will be redirected out port Fa0/24. Because the **`ingress dot1q vlan 146`** keyword is also used, SW1 accepts only inbound tagged traffic with tag 146.

```
SW2#show monitor session 2
Session 2
---------
Type                    : Remote Source Session
Source Ports            :      Both                : Fa0/5
Dest RSPAN VLAN         : 500
```

https://t.me/learningnets

```
!
!SW1#show monitor session 2
Session 2
---------
Type                   : Remote Destination Session Source RSPAN VLAN       : 500
Destination Ports      : Fa0/24
    Encapsulation      : Native Ingress       : Enabled, default VLAN = 146
Ingress encap : DOT1Q
```

Verify that VLAN 500 was propagated through VTP as RSPAN.

```
SW2#show vlan id 500

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
500  VLAN0500                         active    Fa0/19, Fa0/20

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
500  enet  100500     1500  -      -      -        -    -        0      0
Remote SPAN VLAN
----------------Enabled


Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Voice VLAN

> You must load the initial configuration files for the section, **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure port Fa0/1 of SW1 to assign data packets in VLAN 146 and voice packets in VLAN 600.
- Configure port Fa0/4 of SW1 as an 802.1q trunk link.
  - Configure VLAN 146 as the native VLAN for this port.
  - Configure VLAN 600 to be advertised as voice VLAN via CDP.
  - Allow only traffic for VLANs 146 and 600.
- Configure port Fa0/6 of SW1 to assign data packets in VLAN 146.
  - Ensure that voice VLAN frames use dot1p tagging.

Configuration

SW1:

```
interface FastEthernet0/1
 switchport mode access
 switchport access vlan 146
 switchport voice vlan 600
!
interface FastEthernet0/4
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport trunk native vlan 146
```

https://t.me/learningnets

```
 switchport trunk allowed vlan 146,600
 switchport voice vlan 600
!
interface FastEthernet0/6
 switchport mode access
 switchport access vlan 146
 switchport voice vlan dot1p
```

Verification

Many models of Cisco IP Phones have a built-in three-port switch with one port to connect to the upstream switch, one port for the IP Phone itself, and the last port to connect to a desktop PC. The built-in switch is capable of separating the IP Phone and the desktop PC traffic using different VLANs. Additionally, the internal switch can also use different 802.1p markings in the Class of Service (CoS) field to distinguish the IP Phone and the desktop PC frames. Based on this, there are three different options for connecting the IP Phone and the desktop PC to the Catalyst switches.

- Option 1 is to separate the Data VLAN for the PC and the Voice VLAN for the IP Phone. The internal IP Phone switch will tag VoIP traffic with the respective VLAN number and apply a CoS value of 5. The data frames are sent untagged and received by the upstream switch on the configured access VLAN. The connection between the IP Phone and the upstream switch is an 802.1q trunk with the native VLAN equal to the Data VLAN.
- Option 2 is to use a single VLAN for Data and Voice. The IP Phone's internal switch does not tag the frames and acts as a simple bridge. The connection between the IP Phone and the upstream switch is an access port.
- Option 3 is to use a single VLAN for Data and Voice, but to add an 802.1p CoS tag. Data frames received from the PC on the phone, along with VoIP frames sent from the phone, get a special 802.1q header that carries a VLAN ID equal to zero and has the CoS field set to 5 for VoIP and the value instructed from the switch for data frames. The Catalyst switch accepts the frames with VLAN zero as if they are in the access VLAN, but it also honors the CoS bits to calculate the switch's internal QoS tag.

For all three options, the IP Phone's built-in switch should be instructed which mode to use. The command `switchport voice vlan` configured on an access port will

https://t.me/learningnets

communicate with the IP Phone via CDP and tell its internal switch which VLAN to use for voice traffic. The IP Phone's internal switch will then apply the instructed VLAN tag to the voice traffic and send the PC's data untagged. Note that there is no need to configure the port as an 802.1q trunk via the `switchport mode trunk` command. The switchport ASIC will automatically convert the port into a rudimentary trunk.

If no `switchport voice vlan` command is configured, Option 2 applies automatically. Both voice and data packets are received on the same VLAN (the access VLAN). If the command `switchport voice vlan dot1p` is configured on a switchport, the connected IP Phone's switch is instructed to apply VLAN 0 to voice traffic along with the corresponding CoS bits. Both voice and data frames will share the same VLAN configured on the access port.

> Note that as soon as the **`switchport voice vlan`** command is applied to the port, PortFast feature is automatically enabled, although not visible in running-config.

Verify the voice and data VLAN configuration on port Fa0/1, and note that PortFast was automatically enabled.

```
SW1#show interfaces fastEthernet0/1 switchport
Name: Fa0/1
Switchport: EnabledAdministrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: OffAccess Mode VLAN: 146 (VLAN0146)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabledVoice VLAN: 600 (VLAN0600)
<output omitted>
!
!SW1#show spanning-tree interface fastEthernet0/1 detail
Port 3 (FastEthernet0/1) of VLAN0146 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 4242, address 0013.605f.f000
   Designated bridge has priority 4242, address 0013.605f.f000
   Designated port id is 128.3, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1The port is in the portfast mode
   Link type is point-to-point by default
   BPDU: sent 18, received 0
Port 3 (FastEthernet0/1) of VLAN0600 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   Designated root has priority 4696, address 0013.605f.f000
```

https://t.me/learningnets

```
    Designated bridge has priority 4696, address 0013.605f.f000
    Designated port id is 128.3, designated path cost 0
    Timers: message age 0, forward delay 0, hold 0
    Number of transitions to forwarding state: 1 The port is in the portfast mode

    Link type is point-to-point by default
    BPDU: sent 18, received 0
```

Verify the voice and data VLAN configuration on ports Fa0/4 and Fa0/6.

```
SW1#show interfaces fastEthernet0/4 switchport
Name: Fa0/4
Switchport: Enabled Administrative Mode: trunk
Operational Mode: down Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default) Trunking Native Mode VLAN: 146 (VLAN0146)
Administrative Native VLAN tagging: enabled Voice VLAN: 600 (VLAN0600)
<output omitted>
!
!SW1#show interfaces fastEthernet0/6 switchport
Name: Fa0/6
Switchport: Enabled Administrative Mode: static access
Operational Mode: down
Administrative Trunking Encapsulation: negotiate
Negotiation of Trunking: Off Access Mode VLAN: 146 (VLAN0146)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled Voice VLAN: dot1p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Smartport Macros

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **LAN Switching Initial Spanning Tree**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure a macro on SW1 named **VLAN_146** so that when applied to an interface:
  - It enables the interface as access mode in VLAN 146.
  - It enables BPDU Filter on the port.
- Apply this macro to ports Fa0/7 and Fa0/8 on the switch.

Configuration

This is the old command syntax:

```
SW1:

macro name VLAN_146
 switchport mode access
 switchport access vlan 146
 spanning-tree bpdufilter enable
 @
!
interface range FastEthernet0/7 - 8
 macro apply VLAN_146
```

https://t.me/learningnets

This is the new command syntax:

```
SW1:

define interface-range VLAN_146 fastEthernet0/7 - 8
!
interface range macro VLAN_146
 switchport mode access
 switchport access vlan 146
 spanning-tree bpdufilter enable
```

Verification

Smartport Macros are used to define a well-known template of configuration to apply onto multiple interfaces. This feature is useful in large switching environments where general categories of ports can be defined, such as access, server, and uplink, and have them share common configuration templates. Note that with newer codes on the switches, the command to create the macro is hidden.

In this particular design, the macro is used to apply three attributes to the interface: the switchport mode, the access VLAN, and the BPDU Filter feature. The result shown in the show run output is identical to what would be achieved by manually entering these commands on both interfaces, with the addition of the **`macro description`**, telling us which macro was applied:

```
SW1#show running-config | section 0/7|0/8
interface FastEthernet0/7
 switchport access vlan 146
 switchport mode access macro description VLAN_146
 spanning-tree bpdufilter enable interface FastEthernet0/8
 switchport access vlan 146
 switchport mode access macro description VLAN_146
 spanning-tree bpdufilter enable
!
!SW1#show interfaces fastEthernet0/7 switchport
Name: Fa0/7
Switchport: Enabled Administrative Mode: static access
Operational Mode: down
Administrative Trunking Encapsulation: negotiate
Negotiation of Trunking: Off Access Mode VLAN: 146 (VLAN0146)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
<output omitted>
```

https://t.me/learningnets

```
!
!SW1#show interfaces fastEthernet0/8 switchport
Name: Fa0/8
Switchport: EnabledAdministrative Mode: static access
Operational Mode: down
Administrative Trunking Encapsulation: negotiate
Negotiation of Trunking: OffAccess Mode VLAN: 146 (VLAN0146)

Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
<output omitted>
```

Several default Smartport Macros exist in the switch, which can be seen by issuing the **`show parser macro`** command.

https://t.me/learningnets

```
SW1#show parser macro brief

    default global   : cisco-global
    default interface: cisco-desktop
    default interface: cisco-phone
    default interface: cisco-switch
    default interface: cisco-router
    default interface: cisco-wireless customizable      : VLAN_146

!
!SW1#show parser macro name cisco-router

Macro name : cisco-router
Macro type : default interface
# macro keywords $native_vlan
# Access Uplink to Distribution switchport trunk encapsulation dot1q


# Define unique Native VLAN on trunk ports
# Recommended value for native vlan should not be 1 switchport trunk native vlan $native_vlan


# Update the allowed VLAN range such that it
# includes data, voice and native VLANs switchport trunk allowed vlan ALL


# Hardcode trunk switchport mode trunk


# Configure qos to trust this interface auto qos voip trust
mls qos trust dscp


# Ensure fast access to the network when enabling the interface.
# Ensure that switch devices cannot become active on the interface. spanning-tree portfast trunk
spanning-tree bpduguard enable
```

A default macro can be applied as follows.

```
SW1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.SW1(config)#interface FastEthernet0/10
SW1(config-if)#macro apply cisco-desktop $access_vlan 10
```

https://t.me/learningnets

```
%Warning: portfast should only be enabled on ports connected to a single
 host. Connecting hubs, concentrators, switches, bridges, etc... to this
 interface  when portfast is enabled, can cause temporary bridging loops.
 Use with CAUTION

%Portfast has been configured on FastEthernet0/10 but will only
 have effect when the interface is in a non-trunking mode.
!
!SW1#show running-config interface fastethernet0/10
Building configuration...

Current configuration : 332 bytes
!interface FastEthernet0/10
 switchport access vlan 10
 switchport mode access
 switchport port-security
 switchport port-security aging time 2
 switchport port-security violation restrict
 switchport port-security aging type inactivity
 macro description cisco-desktop
 spanning-tree portfast
 spanning-tree bpduguard enable
end
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

Private VLANs

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic Layer 2 Switching**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- Configure private-vlan on SW1 as follows:
  - Use VLAN 100 as the primary VLAN.
  - Use VLAN 1000 as community VLAN.
  - Use VLAN 2000 as isolated VLAN.
- Configure IP addressing as follows:
  - Assign 169.254.100.1/24 to SW1's SVI on VLAN 100.
  - Assign 169.254.100.2/24 to SW2's Fa0/24 port.
  - Assign 169.254.100.3/24 to SW3's Fa0/20 port.
  - Assign 169.254.100.4/24 to SW4's Fa0/22 port.
- Assign SW1's interfaces to required VLANs so that:
  - SW1, SW2, and SW3 can communicate directly at Layer 2.
  - SW4 can only communicate with SW1 directly at Layer 2.

Configuration

```
SW1:
vtp mode transparent
!
vlan 1000
  private-vlan community
!
```

https://t.me/learningnets

```
vlan 2000
  private-vlan isolated
!
vlan 100
  private-vlan primary
  private-vlan association 1000,2000
!
interface range FastEthernet0/20 , FastEthernet0/24
 switchport private-vlan host-association 100 1000
 switchport mode private-vlan host
!
interface FastEthernet0/22
 switchport private-vlan host-association 100 2000
 switchport mode private-vlan host
!
interface Vlan100
 ip address 169.254.100.1 255.255.255.0
 private-vlan mapping 1000,2000
SW2:
interface FastEthernet0/24
 no switchport
 ip address 169.254.100.2 255.255.255.0
SW3:
interface FastEthernet0/20
 no switchport
 ip address 169.254.100.3 255.255.255.0
SW4:

interface FastEthernet0/22
 no switchport
 ip address 169.254.100.4 255.255.255.0
```

Verification

The Private VLAN (PVLANs) feature is similar in theory to the Protected Ports feature, in which two or more ports can be in the same VLAN but cannot directly communicate at Layer 2. Private VLANs expand this concept much further, however, and allow very complex security policies that can span between multiple physical switches. Private VLANs split a single broadcast domain, that is normally defined by a single VLAN, into multiple isolated broadcast subdomains, that are defined by primary VLAN and its secondary VLANs. In essence, the feature allows us to configure VLANs inside a VLAN.

https://t.me/learningnets

From a design perspective, this feature is commonly used in environments like shared ISP co-location, in which customers are on the same VLAN and same IP subnet, but should not communicate directly with each other, or in Multiple Dwelling Units (MDUs) such as hotels or office buildings, where two hotel rooms or offices may be in the same subnet and VLAN but should not communicate directly.

Pitfall

The Private VLAN feature requires VTP to run in transparent or off mode.

Although the theory of PVLANs is relatively straightforward, the implementation can be confusing because of the different terms that Cisco uses to describe VLANs and ports and the syntax in which they are bound together. First we must define the port roles used in PVLANs. These are promiscuous ports, community ports, and isolated ports:

- Promiscuous ports are allowed to talk to all other ports within the VLAN.
- Isolated ports are only allowed to talk to promiscuous ports.
- Community ports are allowed to talk to other ports in their own community, but not ports in different communities, and can talk to any promiscuous ports.

The port roles are defined by the interface's association to a primary VLAN and one or more secondary VLANs. First the secondary VLANs are created, and defined as either community or isolated. Then the primary VLAN is defined, and the secondary VLANs are associated with the primary VLAN.

Next the command `switchport mode private-vlan promiscuous` or `switchport mode private-vlan host` is configured at the physical interface level. As you might guess, the promiscuous option indicates that the port role is promiscuous, and the host option indicates that the port role is either community or isolated. Last, the port is assigned to both the primary and secondary VLANs, which defines what other ports it can talk to. The links to SW2 and SW3 have the command `switchport private-vlan host-association 100 1000` configured, which means that it is a member of the primary VLAN 100 and the secondary VLAN 1000. VLAN 1000 was defined as a community VLAN, which implies that SW2 and SW3 can talk to all other ports in VLAN 1000 and any promiscuous ports belonging to VLAN 100. The SVI interface in VLAN 100 can only be a promiscuous port, and it needs the secondary VLANs mapped using command `private-vlan mapping 1000,2000` ; this is so that you can actually restrict which secondary VLANs can communicate with the promiscuous port.

Verify the private VLAN configuration and port assignment.

```
SW1#show vlan private-vlan
```

https://t.me/learningnets

```
Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
100     1000      community         Fa0/20, Fa0/24
100     2000      isolated          Fa0/22
!
!SW1#show interfaces vlan100 private-vlan mapping
Interface Secondary VLANs
--------- ------------------------------------------------------------------vlan100   1000, 2000
!
!SW1#show interfaces fastEthernet0/20 switchport
Name: Fa0/20
Switchport: EnabledAdministrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: noneAdministrative private-vlan host-association: 100 (VLAN0100) 1000 (VLAN1000)
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: noneOperational private-vlan:
 100 (VLAN0100) 1000 (VLAN1000)
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL
<output omitted>
!
!SW1#show interfaces fastEthernet0/24 switchport
Name: Fa0/24
Switchport: EnabledAdministrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: noneAdministrative private-vlan host-association: 100 (VLAN0100) 1000 (VLAN1000)
```

https://t.me/learningnets

```
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: noneOperational private-vlan:
 100 (VLAN0100) 1000 (VLAN1000)

Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL
<output omitted>
```

Final verification for this configuration can be obtained by sending traffic to the broadcast address of 255.255.255.255 from all devices. As defined in the requirements, SW1 can communicate with all switches because it is a promiscuous port.

```
SW1#ping 255.255.255.255 repeat 1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:
 Reply to request 0 from 169.254.100.4, 8 ms
Reply to request 0 from 169.254.100.2, 8 ms
Reply to request 0 from 169.254.100.3, 8 ms
```

SW2 and SW3 can communicate with each other, as members of the community VLAN, and with SW1, which is the promiscuous host.

https://t.me/learningnets

```
SW2#ping 255.255.255.255 repeat 1

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:
Reply to request 0 from 169.254.100.3, 9 ms
Reply to request 0 from 169.254.100.1, 9 ms

!
!SW3#ping 255.255.255.255 repeat 1

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:
Reply to request 0 from 169.254.100.2, 8 ms
Reply to request 0 from 169.254.100.1, 8 ms
```

SW4, as member of the isolated VLAN, can only communicate with SW1, which is the promiscuous host.

```
SW4#ping 255.255.255.255 repeat 1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:
Reply to request 0 from 169.254.100.1, 8 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - LAN Switching

VTP Version 3

> You must load the initial configuration files for the section, **LAN Switching Initial VTP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Virtual Routers & Physical Switches Diagram** to complete this task.

Task

- All switches are pre-configured in VTP domain **CCIE**.
- Configure VTP Version 3 on all switches.
- Set the VTP password to **CISCO** and ensure that it cannot be retrieved through **show commands** or by looking at the **vlan.dat** file.
- Ensure that SW2 can modify the VLAN database.

Configuration

https://t.me/learningnets

```
SW1:

vtp version 3
vtp password CISCO hidden
SW2:

vtp version 3
vtp password CISCO hidden
end
!
vtp primary vlan
SW3:

vtp version 3
vtp password CISCO hidden
SW4:

vtp version 3
vtp password CISCO hidden
```

Verification

VTP Version 3 comes with multiple VLAN database security improvements, the most significant being that only the VTP switch designated as **primary** can update the VLAN database within one VTP domain, regardless of the configuration revision number value. The switch designated as **primary** must run in server mode. By default, all switches running in server mode are designated as **secondary** servers. Note that the command `vtp primary` is configured from `#` mode. VTP Version 3 is modular, in that it supports advertisement propagation for several databases or instances:

- **VLAN** database configuration
- **MST** configuration
- **Unknown**, reserved for future use

For each of the above modules, a switch can run in the following modes:

- Server
- Client
- Transparent

https://t.me/learningnets

- Off

Version 3 also brings support for advertising Private-VLAN configuration and extended-range VLANs. It also has the ability to hide the password so that it cannot be retrieved by means of show commands. If a hidden password was configured, the administrator would need to provide the password before promoting a secondary server to primary. A VTP client cannot be promoted to primary server, but it can still participate by listening and processing VTP updates from the primary server. The output of SW1 will be similar to SW3 and SW4.

```
SW1#show vtp status
VTP Version capable             : 1 to 3VTP version running             : 3
VTP Domain Name                 : CCIE
VTP Pruning Mode                : Enabled
VTP Traps Generation            : Disabled
Device ID                       : 000a.b832.3580

Feature VLAN:
--------------VTP Operating Mode                : Client
Number of existing VLANs          : 18
Number of existing extended VLANs : 0
Maximum VLANs supported locally   : 1005
Configuration Revision            : 0
Primary ID                        : 0000.0000.0000
Primary Description               :
MD5 digest                        : 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
                                    0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00



Feature MST:
--------------
VTP Operating Mode                : Transparent



Feature UNKNOWN:
--------------
VTP Operating Mode                : Transparent
!
!SW2#show vtp status
VTP Version capable             : 1 to 3VTP version running             : 3
VTP Domain Name                 : CCIE
VTP Pruning Mode                : Enabled
VTP Traps Generation            : Disabled
Device ID                       : 001c.576d.4a00
```

https://t.me/learningnets

```
Feature VLAN:
--------------VTP Operating Mode                : Server

Number of existing VLANs          : 18
Number of existing extended VLANs : 0
Maximum VLANs supported locally   : 1005
Configuration Revision            : 0
Primary ID                        : 0000.0000.0000
Primary Description               :
MD5 digest                        : 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
                                    0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00


Feature MST:
--------------
VTP Operating Mode                : Transparent


Feature UNKNOWN:
--------------
VTP Operating Mode                : Transparent
```

With the other versions of VTP, the password could be easily looked at in clear text by using the **`show vtp password`** command. Look at the output with VTP Version 3 and the new hidden password feature.

```
SW1#show vtp password
 VTP Password: 14F81D29C1B9FBF90576F97120429250
```

Now we will promote SW2 to the primary server role and add VLAN 2055. Note that this VLAN would have not been able to get propagated with Versions 1 or 2 because it is higher than 1001.

https://t.me/learningnets

```
SW2#vtp primary vlan force
 This system is becoming primary server for feature vlan

Enter VTP Password: CISCO
%SW_VLAN-4-VTP_PRIMARY_SERVER_CHG: 001c.576d.4a00 has become the primary server for the VLAN VTP feature
!SW2#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.SW2(config)#vlan 2055
SW2(config-vlan)#name TEST_VLAN
```

Note in the output of `show vtp status` that the **Primary ID** value has changed from **0000.0000.0000** to a switch-derived MAC address of SW2 and **Primary Description ** value is the hostname of the primary VTP server.

```
SW1#show vtp status

VTP Version capable             : 1 to 3
VTP version running             : 3
VTP Domain Name                 : CCIE
VTP Pruning Mode                : Enabled
VTP Traps Generation            : Disabled
Device ID                       : 0013.605f.f000

Feature VLAN:
--------------
VTP Operating Mode                : Client
Number of existing VLANs          : 16
Number of existing extended VLANs : 1
Maximum VLANs supported locally   : 1005
Configuration Revision            : 2Primary ID                         : 000a.b832.3a80
Primary Description               : SW2
MD5 digest                        : 0x41 0x9B 0x84 0xFA 0x2E 0x10 0x9B 0x37
                                    0x72 0x1D 0x28 0x58 0xA4 0x2F 0xE6 0xC0
<output omitted>
!
!                                                                    SW2#show vtp status

VTP Version capable             : 1 to 3
VTP version running             : 3
VTP Domain Name                 : CCIE
VTP Pruning Mode                : Enabled
VTP Traps Generation            : Disabled
Device ID                       : 000a.b832.3a80
```

https://t.me/learningnets

```
Feature VLAN:
--------------
VTP Operating Mode                : Primary Server
Number of existing VLANs          : 16
Number of existing extended VLANs : 1
Maximum VLANs supported locally   : 1005
Configuration Revision            : 1 Primary ID                          : 000a.b832.3a80
Primary Description               : SW2

MD5 digest                        : 0x57 0xA9 0x31 0x33 0x2A 0xC6 0x64 0x1C
                                    0x9B 0x83 0x55 0x15 0x86 0xA7 0x0C 0x0A
<output omitted>
```

Note that as soon as there is a promotion to primary server, all members of the VTP domain output a syslog message.

```
%SW_VLAN-4-VTP_PRIMARY_SERVER_CHG: 000a.b832.3a80 has become the primary server for the VLAN VTP feature
```

Now check that the extended range VLAN has been propagated within the VTP domain.

```
SW1#show vlan id 2055

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
2055 TEST_VLAN                        active
    Fa0/19, Fa0/20, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
2055 enet  102055     1500  -      -      -        -    -        0      0

Remote SPAN VLAN
----------------
Disabled

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

When all switches have selected their primary VTP server from which to accept updates, this is reported in VTP messages so that you get a complete map of all

https://t.me/learningnets

devices in the VTP domain.

```
SW1#show vtp devices
Retrieving information from the VTP domain. Waiting for 5 seconds.

VTP Feature  Conf Revision Primary Server Device ID      Device Description
------------ ---- -------- -------------- -------------- ----------------------
VLAN         No   2 000a.b832.3a80=000a.b832.3a80 SW2
                   VLAN         No   2 000a.b832.3a80 001a.a174.2500 SW4
                   VLAN         No   2 000a.b832.3a80 0022.5627.1f80 SW3
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Routing to Multipoint Broadcast Interfaces

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 with a IPv4 static route for R4's Loopback0 prefix, using only the next-hop with a value of R4.
- Configure R1 with a IPv4 static route for R6's Loopback0 prefix, using only R1's Ethernet outgoing interface.
- Disable Proxy ARP on R6's connections to VLAN 146.
  - Ensure that R1 can ping the Loopback0 interfaces of R4 and R6.
  - Modify R1's ARP table so that it still has IPv4 reachability to the Loopback0 interface of R6.

Configuration

```
R1:
ip route 150.1.4.4 255.255.255.255 155.1.146.4
ip route 150.1.6.6 255.255.255.255 GigabitEthernet1.146
!
arp 150.1.6.6 0011.93da.bf40 arpa

R6:

interface GigabitEthernet1
 mac-address 0011.93da.bf40
!
interface GigabitEthernet1.146
```

https://t.me/learningnets

```
no ip proxy-arp
```

Note that configuring the MAC address statically on R6 is not required; it is simply listed here to clarify that this is the MAC address assigned to R6's interface.

Verification

When the router needs to route a packet which matches an entry in the routing table with a next-hop value, it performs Layer 3 to Layer 2 resolution for the next-hop address. If it matches an entry in the routing table with just the outgoing/exit local interface, without a next-hop value, it performs Layer 3 to Layer 2 resolution for the final destination of the IP packet. In this particular case, this means that R1 will ARP and use the MAC address of next-hop 155.1.146.4 to reach 150.1.4.4, but will ARP for the address 150.1.6.6 to reach 150.1.6.6. Because all routers have Proxy ARP enabled by default on all interfaces, when R1 sends an ARP Request for 150.1.6.6, R6 will reply with its own MAC address from VLAN 146:

```
R1#show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  155.1.146.1             -   000e.83f8.0d00  ARPA   GigabitEthernet1.146
Internet  155.1.146.4             5   0011.20d2.4641  ARPA   GigabitEthernet1.146
Internet  155.1.146.6             5   0011.93da.bf40  ARPA   GigabitEthernet1.146
!R1#ping 150.1.4.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
!R1#ping 150.1.6.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/4 ms
!R1#show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  155.1.146.1             -   000e.83f8.0d00  ARPA   GigabitEthernet1.146
Internet  155.1.146.4             5   0011.20d2.4641  ARPA   GigabitEthernet1.146Internet
155.1.146.6             5   0011.93da.bf40
  ARPA   GigabitEthernet1.146Internet150.1.6.6               0   0011.93da.bf40
  ARPA   GigabitEthernet1.146
```

When Proxy ARP is disabled on R6, R1 cannot resolve the destination 150.1.6.6

https://t.me/learningnets

through ARP. This is seen from the **encapsulation failed** message that R1 generates in the `debug ip packet detail` output. Encapsulation failed means that the router does not know the destination Layer 2 MAC address to use for building the Layer 2 frame. Depending on the platform and IOS code message may be different, as in this case where the relevant message is **not enough info to forward via fib**:

```
R1#clear arp
!R1#debug arp
ARP packet debugging is on
!R1#debug ip packet
IP packet debugging is on
!R1#ping 150.1.6.6 repeat 1

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:.
Success rate is 0 percent (0/1)
!
!
IP: s=155.1.146.1 (local), d=150.1.6.6, len 100, local feature
    ICMP type=8, code=0, feature skipped, Logical MN local(14), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk
FIBipv4-packet-proc: route packet from (local) src 155.1.146.1 dst 150.1.6.6
FIBfwd-proc: Default:150.1.6.6/32 process level forwarding
FIBfwd-proc: depth 0 first_idx 0 paths 1 long 0(0)
FIBfwd-proc: try path 0 (of 1) v4-ap-GigabitEthernet1.146 first short ext 0(-1)
FIBfwd-proc: v4-ap-GigabitEthernet1.146 valid
FIBfwd-proc: GigabitEthernet1.146 no nh type 3  - deag
FIBfwd-proc: ip_pak_table 0 ip_nh_table 65535 if GigabitEthernet1.146 nh none deag 1 chg_if 0 via fib 0 path type at
FIBfwd-proc: Default:150.1.6.6/32 not enough info to forward via fib (GigabitEthernet1.146 none)
FIBipv4-packet-proc: packet routing failed
IP: tableid=0, s=155.1.146.1 (local), d=150.1.6.6 (GigabitEthernet1.146), routed via RIB
IP: s=155.1.146.1 (local), d=150.1.6.6 (GigabitEthernet1.146), len 100, sending
    ICMP type=8. code=0
IP ARP: creating incomplete entry for IP address: 150.1.6.6 interface GigabitEthernet1.146

IP ARP: sent req src 155.1.146.1 0050.568d.2e27,
                 dst 150.1.6.6 0000.0000.0000 GigabitEthernet1.146
```

There are two ways to resolve this problem: either change the static routing configuration on R1 so that it does not ARP for the final destination from the IP header, or statically configure the ARP cache of R1 so that it knows which MAC address to use when it sends the packet to 150.1.6.6:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#arp 150.1.6.6 0011.93da.bf40 arpa
```

https://t.me/learningnets

```
!R1#show arp

Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  155.1.146.1             -   000e.83f8.0d00  ARPA   GigabitEthernet1.146
Internet  155.1.146.4             4   0011.20d2.4641  ARPA   GigabitEthernet1.146
Internet  155.1.146.6             4   0011.93da.bf40  ARPA   GigabitEthernet1.146
Internet  150.1.6.6               -   0011.93da.bf40  ARPA

!R1#ping 150.1.6.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms
```

From a design perspective, the ideal solution for this problem is to never configure a static route to point out a multipoint interface. Static routes should either point to the next-hop value of the neighbor on the multipoint interface or point to an interface only if it is point-to-point, such as a GRE tunnel, PPP or HDLC link.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Routing to NBMA Interfaces

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 and R2 with IPv4 default routes through the DMVPN cloud with a next-hop of R5.
  - Ensure that the route is valid as long as the Tunnel interface is in the **UP** state.

- Configure R5 with IPv4 static routes for R1's and R2's Loopback0 prefixes through the DMPVN cloud.
- Ensure that R1, R2, and R5 can all ping each other's Loopback0 interfaces.

Configuration

https://t.me/learningnets

```
R1:

ip route 0.0.0.0 0.0.0.0 Tunnel0 155.1.0.5

R2:

ip route 0.0.0.0 0.0.0.0 Tunnel0 155.1.0.5

R5:

ip route 150.1.1.1 255.255.255.255 155.1.0.1
ip route 150.1.2.2 255.255.255.255 155.1.0.2
```

Verification

When configuring a static route, the following options are available:

- specify only the next-hop value; route is valid as long as a route exists for the next-hop value.
- Specify only the local outgoing interface; route is valid as long as the interface is in the **UP/UP** state.
- Specify both next-hop value and local outgoing interface.

When the third option is selected, the local outgoing interface behaves like a condition for the next-hop value and should be read like: this static route is valid only if the configured next-hop value is reachable over the configured interface, which means as long as the interface is in the **UP/UP** state and has nothing to do with IP/ARP/NHRP functionality with the next-hop. Check connectivity between Loopback0 interfaces of the routers:

```
R1#ping 150.1.5.5

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/64/88 ms
!R1#ping 150.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
!R2#ping 150.1.5.5

Type escape sequence to abort.
```

https://t.me/learningnets

```
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 56/59/60 ms
!R2#ping 150.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
!R5#ping 150.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/58/60 ms
!R5#ping 150.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

DMVPN Phase 2 and Phase 3 use a multipoint GRE (mGRE) interface on both hubs and spokes; read more on this in the DMVPN section of the workbook. When configuring static routes over mGRE interfaces in DMVPN, the following restrictions apply:

- On spokes, all three options outlined above are available: next-hop value, local outgoing interface, or both.
- On hubs, you must specify the next-hop value at all times, so using only the local outgoing interface is not a functional solution.

When traffic is routed over the mGRE interface of the DMVPN cloud, the router must perform GRE encapsulation, so it needs to know the source IP address (which is identified from the configured `tunnel source` command) and destination IP address (NBMA IP address of the remote spoke/hub, which is determined through NHRP unless statically configured), so ARP has no role in this process. Given the dynamic built-in design of DMVPN, the only static NHRP mappings are configured on spokes for the hub, to allow spokes to successfully and dynamically register their NBMA and tunnel IP addresses with the hub. When a DMVPN member needs to resolve an NBMA address dynamically, it sends an NHRP Resolution Request to the NHS (Next-Hop-Server), which is always the hub. For this reason, because the hub is the only NHS in the cloud and it cannot query itself, static routing with only the local outgoing interface on the hub is not functional. To better understand the process, let's first configure static routing on the spokes with only the outgoing exit interface; make sure to first remove the routes provided by the solution on spokes.

https://t.me/learningnets

```
R1:

ip route 0.0.0.0 0.0.0.0 Tunnel0

R2:

ip route 0.0.0.0 0.0.0.0 Tunnel0
```

Before generating any traffic, note that on spokes, only the static NHRP mappings for the hub exist.

```
R1#show ip nhrp
155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 00:42:45, never expire Type: static, Flags: used
NBMA address: 169.254.100.5
!
!R2#show ip nhrp
155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 00:43:15, never expire Type: static, Flags: used
NBMA address: 169.254.100.5
```

Generate traffic between Loopbck0 interfaces of spokes and notice the newly created NHRP entries; the first packet is dropped until the NHRP Resolution Request process is successful.

```
R1#ping 150.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/2 ms
!R2#ping 150.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/2 ms
!R1#show ip nhrp
155.1.0.1/32 via 155.1.0.1
   Tunnel0 created 00:00:04, expire 00:04:55
   Type: dynamic, Flags: router unique local
   NBMA address: 169.254.100.1
    (no-socket) 155.1.0.2/32 via 155.1.0.2
   Tunnel0 created 00:00:04, expire 00:04:55 Type: dynamic, Flags: router implicit used nhop
NBMA address: 169.254.100.2
155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 00:29:30, never expire
   Type: static, Flags: used
```

https://t.me/learningnets

```
   NBMA address: 169.254.100.5 155.1.2.2/32 via 155.1.2.2

   Tunnel0 created 00:00:11, expire 00:02:53 Type: dynamic, Flags: used temporary
 NBMA address: 169.254.100.5
```

Now configure the static routes on the hub using only the outgoing interface, to see that packets are dropped due to NHRP failure; make sure to first remove the routes provided by the solution on hub.

```
R5:

ip route 150.1.1.1 255.255.255.255 Tunnel0
ip route 150.1.2.2 255.255.255.255 Tunnel0
```

Before generating any traffic, note that on the hub, dynamic NHRP entries exist as all spokes have registered to the hub.

```
R5#show ip nhrp
 155.1.0.1/32 via 155.1.0.1
   Tunnel0 created 00:35:29, expire 00:04:31 Type: dynamic, Flags: unique registered used nhop
 NBMA address: 169.254.100.1
155.1.0.2/32 via 155.1.0.2
   Tunnel0 created 00:05:17, expire 00:04:42 Type: dynamic, Flags: unique registered used nhop
 NBMA address: 169.254.100.2
155.1.0.3/32 via 155.1.0.3
   Tunnel0 created 01:28:56, expire 00:04:01 Type: dynamic, Flags: unique registered used nhop
 NBMA address: 169.254.100.3
155.1.0.4/32 via 155.1.0.4
   Tunnel0 created 01:28:56, expire 00:03:21 Type: dynamic, Flags: unique registered used nhop
 NBMA address: 169.254.100.4
```

Generate traffic to Loopback0 interfaces of R1 or R2, and note the debug output and that traffic is not functional as the tunnel destination cannot be resolved through NHRP (no NHS to query).

```
R5#debug nhrp
 NHRP protocol debugging is on
!R5#debug ip packet detail
 IP packet debugging is on (detailed)
!R5#ping 150.1.1.1 repeat 1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:.
```

https://t.me/learningnets

```
Success rate is 0 percent (0/1)
!
NHRP: nhrp_ifcache: Avl Root:7F498830D308
NHRP: NHRP could not map 150.1.1.1 to NBMA, cache entry not found
NHRP: MACADDR: if_in null netid-in 0 if_out Tunnel0 netid-out 1
NHRP: Checking for delayed event NULL/150.1.1.1 on list (Tunnel0).
NHRP-MPLS:  tableid: 0 vrf:
NHRP: No delayed event node found.
NHRP: nhrp_ifcache: Avl Root:7F498830D308.
!
IP: s=155.1.0.5 (local), d=150.1.1.1, len 100, local feature
    ICMP type=8, code=0, feature skipped, Logical MN local(14), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk
FIBipv4-packet-proc: route packet from (local) src 155.1.0.5 dst 150.1.1.1
FIBfwd-proc: Default:150.1.1.0/24 process level forwarding
FIBfwd-proc: depth 0 first_idx 0 paths 1 long 0(0)
FIBfwd-proc: try path 0 (of 1) v4-ap-Tunnel0 first short ext 0(-1)
FIBfwd-proc: v4-ap-Tunnel0 valid
FIBfwd-proc: Tunnel0 no nh type 3  - deag
FIBfwd-proc: ip_pak_table 0 ip_nh_table 65535 if Tunnel0 nh none deag 1 chg_if 0 via fib 0 path type attached prefix
FIBfwd-proc: Default:150.1.1.0/24 not enough info to forward via fib (Tunnel0 none)
FIBipv4-packet-proc: packet routing failed

IP: tableid=0, s=155.1.0.5 (local), d=150.1.1.1 (Tunnel0), routed via RIB
IP: s=155.1.0.5 (local), d=150.1.1.1 (Tunnel0), len 100, sending
    ICMP type=8, code=0
IP: s=155.1.0.5 (local), d=150.1.1.1 (Tunnel0), len 100, output feature
    ICMP type=8,.
code=0, feature skipped, TCP Adjust MSS(56), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
```

Although the technically correct solution is to fix the static routing by using a next-hop value, a static NHRP mapping can be configured on the hub for both R1 and R2 Loopback0 prefixes.

```
R5:

interface Tunnel0
 ip nhrp map 150.1.2.2 169.254.100.2
 ip nhrp map 150.1.1.1 169.254.100.1
```

Test IPv4 connectivity again and note the added static NHRP mappings on the hub.

```
R5#ping 150.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
!R5#ping 150.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
!R5#show ip nhrp static
150.1.1.1/32 via 150.1.1.1
   Tunnel0 created 00:01:02, never expire Type: static, Flags:
 NBMA address: 169.254.100.1
150.1.2.2/32 via 150.1.2.2
   Tunnel0 created 00:01:32, never expire Type: static, Flags:
 NBMA address: 169.254.100.2
```

For a better understanding, verify the CEF entries on the hub.

https://t.me/learningnets

```
R5#show ip cef 150.1.1.1 internal

150.1.1.1/32, epoch 2, flags attached, refcount 5, per-destination sharing
  sources: Adj
  subblocks:
   Adj source: IP midchain out of Tunnel0, addr 150.1.1.1 7F4987C7E3B8
    Dependent covered prefix type adjfib, cover 150.1.1.0/24
  ifnums:
   Tunnel0(14): 150.1.1.1
  path 7F4990A5C010, path list 7F498E385E00, share 1/1, type adjacency prefix, for IPv4
attached to Tunnel0, adjacency IP midchain out of Tunnel0, addr 150.1.1.1 7F4987C7E3B8

output chain: IP midchain out of Tunnel0, addr 150.1.1.1 7F4987C7E3B8 IP adj out of GigabitEthernet1.100, addr 169.2

!R5#show ip cef 150.1.2.2 internal

150.1.2.2/32, epoch 2, flags attached, refcount 5, per-destination sharing
  sources: Adj
  subblocks:
   Adj source: IP midchain out of Tunnel0, addr 150.1.2.2 7F4987C7E1D8
    Dependent covered prefix type adjfib, cover 150.1.2.0/24
  ifnums:
   Tunnel0(14): 150.1.2.2
  path 7F4990A5C220, path list 7F498E385FE0, share 1/1, type adjacency prefix, for IPv4
attached to Tunnel0, adjacency IP midchain out of Tunnel0, addr 150.1.2.2 7F4987C7E1D8

output chain: IP midchain out of Tunnel0, addr 150.1.2.2 7F4987C7E1D8 IP adj out of GigabitEthernet1.100, addr 169.2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Longest Match Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 and R5 with IPv4 static routes to each other's Loopback0 prefixes via the Ethernet segment between them.
- Configure R4 and R5 with IPv4 static routes for 150.1.0.0/16 prefix via the DMVPN cloud.
- Ensure that traffic between R4's and R5's Loopback0 prefixes is primarily routed over the Ethernet segment, and DMVPN cloud is used only if Ethernet link is **DOWN**.

Configuration

```
R4:
ip route 150.1.5.5 255.255.255.255 GigabitEthernet1.45
ip route 150.1.0.0 255.255.0.0 155.1.0.5

R5:

ip route 150.1.4.4 255.255.255.255 GigabitEthernet1.45
ip route 150.1.0.0 255.255.0.0 155.1.0.4
```

Verification

IPv4 routing logic uses **longest match** routing to determine which entry to use from

https://t.me/learningnets

the routing table for forwarding. This principle can be used to achieve both redundancy and traffic engineering, as is the case for traffic routed between Loopback0 prefixes of R4 and R5. When R5 performs a lookup on the final destination 150.1.4.4, the longest match is 150.1.4.4/32 out GigabitEthernet1.45. Although technically there are multiple routes to this destination, 150.1.4.4/32 and 150.1.0.0/16, the /32 route wins. For any other destination from 150.1.0.0/16 range, such as 150.1.3.3, the longest match will be 150.1.0.0/16 via 155.1.0.4. This allows traffic for a portion of the IP address space to be routed one way, and traffic for a different portion of the IP address space to be routed another way:

```
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks:   * directly connected, via GigabitEthernet1.45
      Route metric is 0, traffic share count is 1
!R5#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 3 msec *  2 msec
```

Redundancy is accomplished in this example when the Ethernet link between R4 and R5 fails. As long as this link is **UP** and associated IPv4 static routes can be installed in the routing table, the 150.1.4.4/32 route will be installed in the routing table. However, if the GigabitEthernet1.45 link is in the **UP/DOWN** or **DOWN/DOWN** states, it cannot be installed in the routing table; the same goes for any routes that recurse to GigabitEthernet1.45. The result is that when the link is down, traffic between Loopbacks of R4 and R5 is rerouted out the DMVPN cloud using the 150.1.0.0/16 prefix as the longest match.

https://t.me/learningnets

```
R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface GigabitEthernet1.45
R5(config-if)#shutdown

!R5#show ip route 150.1.4.4
Routing entry for 150.1.0.0/16

  Known via "static", distance 1, metric 0
  Routing Descriptor Blocks:   * 155.1.0.4

      Route metric is 0, traffic share count is 1
!R5#traceroute 150.1.4.4

Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.4 5 msec *  5 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Floating Static Routes

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 and R5 with indentical IPv4 static routes for each other's Loopback0 prefixes through both the directly connected Ethernet segment and the DMVPN cloud.
- Use administrative distance to ensure that traffic is primarily routed over the Ethernet segment, but is rerouted through the DMVPN cloud if the Ethernet link is down.

Configuration

```
R4:
ip route 150.1.5.5 255.255.255.255 GigabitEthernet1.45 10
ip route 150.1.5.5 255.255.255.255 155.1.0.5 20

R5:

ip route 150.1.4.4 255.255.255.255 GigabitEthernet1.45 10
ip route 150.1.4.4 255.255.255.255 155.1.0.4 20
```

Verification

When a router receives identical routes (prefix and prefix-length) through multiple routing protocols (static or dynamic), the decision to which one gets installed in the

https://t.me/learningnets

routing table is based on the lowest administrative distance; lower AD values have higher preference. With static routing, this principle can be used for simple redundancy by configuring a backup route with a higher administrative distance than the primary route. In this example, R5 installs the route to 150.1.4.4/32 via GigabitEthernet1.45 with an administrative distance of 10. When the link GigabitEthernet1.45 is down, the route with the next-lowest administrative distance, 150.1.4.4/32 via 155.1.0.4 with a distance of 20, is installed. The result is that traffic is routed out Ethernet link unless it is down, in which case traffic is rerouted out the DMVPN cloud.

Verify the active route in the routing table, as well as the RIB entries:

```
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "static", distance 10
, metric 0 (connected)
  Routing Descriptor Blocks:   * directly connected, via GigabitEthernet1.45
      Route metric is 0, traffic share count is 1
!R5#show ip static route

Codes: M - Manual static, A - AAA download, N - IP NAT, D - DHCP,
       G - GPRS, V - Crypto VPN, C - CASA, P - Channel interface processor,
       B - BootP, S - Service selection gateway
       DN - Default Network, T - Tracking object
       L - TL1, E - OER, I - iEdge
       D1 - Dot1x Vlan Network, K - MWAM Route
       PP - PPP default route, MR - MRIPv6, SS - SSLVPN
       H - IPe Host, ID - IPe Domain Broadcast
       U - User GPRS, TE - MPLS Traffic-eng, LI - LIIN
       IR - ICMP Redirect
Codes in []: A - active, N - non-active, B - BFD-tracked, D - Not Tracked, P - permanent

Static local RIB for default
 M  150.1.4.4/32 [10/0] via GigabitEthernet1.45 [A]
M               [20/0] via 155.1.0.4 [N]
```

Use traceroute to verify the active path of the traffic, before and after disabling the Ethernet link:

```
R5#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 3 msec *  3 msec
```

https://t.me/learningnets

```
!R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface GigabitEthernet1.45
R5(config-if)#shutdown
!R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/24
  Known via "static", distance 20
, metric 0
  Routing Descriptor Blocks:   * 155.1.0.4
      Route metric is 0, traffic share count is 1
!R5#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.4 4 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

GRE Backup Interface

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure two GRE tunnels between R4 and R5 as follows:
  - **Tunnel45** with IPv4 addresses 155.45.0.Y/24, where *Y* is the router number, sourced from VLAN 45 Ethernet link.
  - **Tunnel100** with IPv4 addresses 155.100.0.Y/24, where *Y* is the router number, sourced from VLAN 100 Ethernet link.
- Configure IPv4 static routes on R5 for R4's Loopback0 interface via both Tunnel100 and Tunnel45.
- Configure IPv4 static routes on R4 for R5's Loopback0 interface via both Tunnel100 and Tunnel45.
- The static routes on R4 and R5 via the Tunnel45 should have a higher administrative distance than those on Tunnel100.
- Configure the backup interface feature on R4 and R5 so that if the Tunnel100 goes down, Tunnel45 is activated.
- Ensure that the backup link is activated 3 seconds after the main link fails, and deactivated when the main link is active for 60 seconds.
- To verify this configuration, ensure that traffic between Loopback0 prefixes of R4 and R5 is routed out Tunnel100:
  - If Tunnel100 interface state goes **DOWN**, traffic is rerouted out on Tunnel45.

https://t.me/learningnets

Configuration

```
R4:
interface Tunnel45
 ip address 155.45.0.4 255.255.255.0
 tunnel mode gre ip
 tunnel source 155.1.45.4
 tunnel destination 155.1.45.5
!
interface Tunnel100
 ip address 155.100.0.4 255.255.255.0
 tunnel mode gre ip
 tunnel source 169.254.100.4
 tunnel destination 169.254.100.5
 backup interface Tunnel45
 backup delay 3 60
!
ip route 150.1.5.5 255.255.255.255 Tunnel100 10
ip route 150.1.5.5 255.255.255.255 Tunnel45 20

R5:

interface Tunnel45
 ip address 155.45.0.5 255.255.255.0
 tunnel mode gre ip
 tunnel source 155.1.45.5
 tunnel destination 155.1.45.4
!
interface Tunnel100
 ip address 155.100.0.5 255.255.255.0
 tunnel mode gre ip
 tunnel source 169.254.100.5
 tunnel destination 169.254.100.4
 backup interface Tunnel45
 backup delay 3 60
!
ip route 150.1.4.4 255.255.255.255 Tunnel100 10
ip route 150.1.4.4 255.255.255.255 Tunnel45 20
```

Verification

https://t.me/learningnets

In this example, R4 and R5 use the backup interface feature along with duplicate routing information to perform both traffic engineering and redundancy. With the backup interface configured on R4's and R5's point-to-point GRE Tunnel100 interface, R4 and R5 wait for the line protocol of Tunnel100 interface to go **DOWN** before GRE interface Tunnel45 is activated. The following rules and restrictions apply when implementing the backup interface functionality:

- The primary/active interface being backed up must be a point-to-point interface type, because its state can be better determined.
- The secondary/standby interface acting as backup can be any interface except sub-interface, because the state of the main interface determines the state of sub-interfaces in general.

Verify that backup interface is correctly configured, and Tunnel45 waits for Tunnel100 to go **DOWN** to become active.

```
R4#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   normal operation
!R4#show ip interface brief | i Tunnel
Tunnel0                155.1.0.4       YES manual up                    up
Tunnel45               155.45.0.5      YES manual standby mode          down
Tunnel100              155.100.0.5     YES manual up                    up
!R5#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   normal operation
!R5#show ip interface brief | i Tunnel
Tunnel0                155.1.0.5       YES manual up                    up
Tunnel45               155.45.0.5      YES manual standby mode          down

Tunnel100              155.100.0.5     YES manual up                    up
```

Verify that traffic between Loopback0 prefixes of R4 and R5 is primarily routed over GRE Tunnel100.

```
R4#show ip route 150.1.5.5
Routing entry for 150.1.5.5/32
  Known via "static", distance 10, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel100
      Route metric is 0, traffic share count is 1
!R5#show ip route 150.1.4.4
```

https://t.me/learningnets

```
Routing entry for 150.1.4.4/32
  Known via "static", distance 10, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel100
      Route metric is 0, traffic share count is 1
!R5#traceroute 150.1.4.4 source loopback0
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.100.0.4 3 msec *  12 msec
```

Disable VLAN 100 interface on both R4 and R5, which will trigger the backup Tunnel45 interface to go **UP** after the configured delay of 3 seconds.

```
R4#debug backup
Backup events debugging is on
!R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet 1.100
R4(config-subif)#shutdown
!R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface gigabitEthernet 1.100
R5(config-subif)#shutdown
!R4#
BACKUP(Tunnel100): event = primary interface went down
BACKUP(Tunnel100): changed state to "waiting to backup"
BACKUP(Tunnel100): event = timer expired on primary
BACKUP(Tunnel100): secondary interface (Tunnel45) made active
BACKUP(Tunnel100): changed state to "backup mode"
!
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel45, changed state to up
BACKUP(Tunnel45): event = secondary interface came up

%LINK-3-UPDOWN: Interface Tunnel45, changed state to up
```

Verify that the backup interface is now active.

https://t.me/learningnets

```
R4#show backup

Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   backup mode

!R4#sho ip interface brief | i Tunnel

Tunnel0                155.1.0.4       YES manual up                    down
Tunnel45               155.45.0.4      YES manual up                    up

Tunnel100              155.100.0.4     YES manual up                    down
```

Verify that traffic between Loopback0 is now routed over GRE Tunnel45.

```
R4#show ip route 150.1.5.5
Routing entry for 150.1.5.5/32
  Known via "static", distance 20, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel45
      Route metric is 0, traffic share count is 1
!R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "static", distance 20, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel45
      Route metric is 0, traffic share count is 1
!R5#traceroute 150.1.4.4 source loopback0
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.45.0.4 5 msec *  3 msec
```

When R4's and R5's VLAN 100 interfaces are re-enabled, Tunnel100 interface is reactivated after the configured delay of 60 seconds.

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet 1.100
R4(config-subif)#no shutdown
!R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface gigabitEthernet 1.100
R5(config-subif)#no shutdown
!R4#debug backup
Backup events debugging is on
!R4:
BACKUP(Tunnel100): event = primary interface came up
BACKUP(Tunnel100): changed state to "waiting to revert"
```

https://t.me/learningnets

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to up
```

Verify that the primary interface is active and traffic is re-routed over Tunnel100.

```
R4#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   normal operation
!R5#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   normal operation
!R5#traceroute 150.1.4.4 source loopback0
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.100.0.4 5 msec *  5 msec
```

Because end-to-end connectivity between GRE tunnel endpoints is not implemented, the design flaw with this configuration is that if Tunnel100 interface goes **DOWN** on one side only, traffic is blackholed.

Let's disable the VLAN 100 Ethernet link on one side only, for example on R4; note that R4 and R5 have different perspectives of the network state.

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet 1.100
R4(config-subif)#shutdown
!R4#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45 backup mode
!R5#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45 normal operation
```

https://t.me/learningnets

This results in traffic being blackholed as R5 routes traffic over Tunnel100, which is disabled on R4, and R4 routes traffic over Tunnel45, which is in standby mode on R5.

```
R4#show ip route 150.1.5.5
Routing entry for 150.1.5.5/32
  Known via "static", distance 20, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel45
      Route metric is 0, traffic share count is 1
!R4#show ip interface brief | i Tunnel
Tunnel0                155.1.0.4       YES manual up                    down
Tunnel45               155.45.0.4      YES manual up                    up
Tunnel100              155.100.0.4     YES manual up                    down
!R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "static", distance 10, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel100
      Route metric is 0, traffic share count is 1
!
R5#show ip interface brief | i Tunnel
Tunnel0                155.1.0.5       YES manual up                    up
Tunnel45               155.45.0.5      YES manual standby mode          down
Tunnel100              155.100.0.5     YES manual up                    up
!R5#traceroute 150.1.4.4 source loopback0 ttl 1 2
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1  *  *  *  2  *  *  *
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Reliable Static Routing with Enhanced Object Tracking

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 with IPv4 static route for R4's Loopback0 prefix through the DMVPN cloud.
- Configure R5 with IPv4 static routes for R1's and R4's Loopback0 prefixes through the DMVPN cloud.
- Configure R4 with a primary IPv4 static route for R1's Loopback0 prefix via its VLAN146 Ethernet connection.
  - use SLA and Object Tracking to ensure the route is valid as long as ICMP connectivity exists between R1 and R4's Ethernet connection.
  - configure R4 to verify connectivity each 5 seconds.
  - ensure R1 replies within 2 seconds.
- Configure R4 with a backup IPv4 static route for R1's Loopback0 prefix through the DMVPN cloud using administrative distance of 2.

Configuration

```
R1:
ip route 150.1.4.4 255.255.255.255 155.1.0.4


R4:
ip sla 1
 icmp-echo 155.1.146.1 source-interface GigabitEthernet1.146
```

https://t.me/learningnets

```
 threshold 2000
 timeout 2000
 frequency 5
ip sla schedule 1 life forever start-time now
!
track 1 ip sla 1 state
!
ip route 150.1.1.1 255.255.255.255 155.1.146.1 track 1
ip route 150.1.1.1 255.255.255.255 155.1.0.1 2
```

R5:

```
ip route 150.1.1.1 255.255.255.255 155.1.0.1
ip route 150.1.4.4 255.255.255.255 155.1.0.4
```

Verification

Although R1 and R4 are on the same Layer 2 segment in VLAN 146, their physical Ethernet interfaces are not on the same Layer 1 network; there is no back-to-back Ethernet cable between the two routers, connectivity is achieved through a switching infrastructure. This means that the Layer 1 link status of R1's connection to VLAN 146 is independent of R4's Layer 1 link status, and vice-versa. From a static routing redundancy design point of view, the possible problem with this scenario is that routers have no way of detecting the other peer link failure, which may result in traffic being blackholed and silently dropped in the transit path.

To visualize this, before implementing the tracking functionality for the static route let's shutdown R1's Ethernet interface, which will still keep the primary route in the routing table, however IPv4 connectivity will fail:

```
R1#ping 150.1.4.4 source 150.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/58/60 ms
!R1(config)#interface gigabitEthernet1.146
R1(config-subif)#shutdown

!R4#show ip route 150.1.1.1
Routing entry for 150.1.1.1/32
  Known via "static", distance 1, metric 0
  Routing Descriptor Blocks:   * 155.1.146.1
```

https://t.me/learningnets

```
      Route metric is 0, traffic share count is 1
!R1#ping 150.1.4.4 source 150.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 .....
```

To fix the problem, we need to actively monitor IPv4 connectivity between R1 and R4 on the Ethernet segment and mark the primary static route as invalid for being installed in the routing table when connectivity fails. For this scope, IP Service Level Agreement (SLA) and Enhanced Object Tracking features is used. First, R4 is configured with a SLA instance that actively monitors IPv4 connnectivity with R1 over the Ethernet link by sending ICMP Echo Request packets each 5 seconds. SLA will consider connectivity to be functional through the Return Code of **OK** as long as ICMP Echo Reply is received within the configured 2 seconds timeout window; otherwise the Return Code will be **Timeout**:

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#ip sla 1
R4(config-sla-monitor)#icmp-echo 155.1.146.1 source-interface GigabitEthernet1.146
     R4(config-sla-monitor-echo)#frequency 5
R4(config-sla-monitor-echo)#timeout 2000
R4(config-sla-monitor-echo)#exit
R4(config)#ip sla schedule 1 life forever start-time now
!R4#show ip sla configuration 1
IP SLAs Infrastructure Engine-III
Entry number: 1
Owner:
Tag: Operation timeout (milliseconds): 2000
Type of operation to perform: icmp-echo
Target address/Source interface: 155.1.146.1/GigabitEthernet1.146
Type Of Service parameter: 0x0
Request size (ARR data portion): 28
Verify data: No
Vrf Name:
Schedule: Operation frequency (seconds): 5  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
```

https://t.me/learningnets

```
Threshold (milliseconds): 2000
Distribution Statistics:
   Number of statistic hours kept: 2
   Number of statistic distribution buckets kept: 1
   Statistic distribution interval (milliseconds): 20
Enhanced History:
History Statistics:
   Number of history Lives kept: 0
   Number of history Buckets kept: 15
   History Filter Type: None
!R4#show ip sla statistics

IPSLAs Latest Operation Statistics

IPSLA operation id: 1
        Latest RTT: 1 milliseconds
Latest operation start time: 15:59:53 UTC Sat May 3 2014Latest operation return code: OK
Number of successes: 2

Number of failures: 0
Operation time to live: Forever
```

Next, a Enhanced Object Tracking is created that monitors the IP SLA instance Return Code. If SLA Return Code is **OK**, the tracking state is **UP**, while if the SLA Return Code has any other value, the tracking state is **DOWN**:

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#track 1 ip sla 1 state
!R4#show track

Track 1 IP SLA 1 state
State is Up

    2 changes, last change 00:00:26
  Latest operation return code: OK
  Latest RTT (millisecs) 2
```

Next the primary static route is configured with the tracking object attached as a condition. This will instruct the router to consider the route as valid for being entered in the routing table as long as the tracking state is **UP**. Also note that tracking now shows it is attached to static routing:

https://t.me/learningnets

```
R4(config)#ip route 150.1.1.1 255.255.255.255 155.1.146.1 track 1

!R4#show track

Track 1
  IP SLA 1 state
  State is Up
    2 changes, last change 00:02:02
  Latest operation return code: OK
  Latest RTT (millisecs) 1
  Tracked by: Static IP Routing 0
!R4#show ip route static | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 1 subnetsS        150.1.1.1 [1/0] via 155.1.146.1
```

We simulate the same network failure, however due to tracking being configured for the primary route, once R4 will detect loss of IPv4 connectivity with R1, it will mark the primary route as invalid and inject the backup route in the routing table:

```
R4#traceroute 150.1.1.1 source 150.1.4.4

Type escape sequence to abort.
Tracing the route to 150.1.1.1
  1 155.1.146.1 36 msec *  36 msec
!R4#debug track state

track state debugging enabled
!R4#debug ip routing
IP routing debugging is on
!R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface gigabitEthernet1.146
R1(config-subif)#shutdown

!R4#
Track: 1 Change #3 rtr 1, state Up->Down
%TRACK-6-STATE: 1 ip sla 1 state Up -> Downtrack-sta (1) ip sla 1 state Up -> Down
RT: del 150.1.1.1 via 155.1.146.1, static metric [1/0]RT: delete subnet route to 150.1.1.1/32
RT: updating static 150.1.1.1/32 (0x0)  :
    via 155.1.0.1   0 1048578
 RT: add 150.1.1.1/32 via 155.1.0.1, static metric [2/0]
RT: updating static 150.1.1.1/32 (0x0)  :
```

https://t.me/learningnets

```
via 155.1.0.1    0 1048578

!R4#traceroute 150.1.1.1 source 150.1.4.4

Type escape sequence to abort.
Tracing the route to 150.1.1.1
  1 155.1.0.5 28 msec 28 msec 28 msec
  2 155.1.0.1 56 msec *  56 msec
!R4#show ip route 150.1.1.1
Routing entry for 150.1.1.1/32
  Known via "static", distance 2, metric 0
  Routing Descriptor Blocks:   * 155.1.0.1

      Route metric is 0, traffic share count is 1
```

Verify the SLA and tracking states:

```
R4#show ip sla statistics 1
IPSLAs Latest Operation Statistics

IPSLA operation id: 1
        Latest RTT: NoConnection/Busy/Timeout
Latest operation start time: 16:36:54 UTC Sat May 3 2014 Latest operation return code: Timeout
Number of successes: 82
Number of failures: 70
Operation time to live: Forever
!R4#show track

Track 1
  IP SLA 1 state State is Down
    3 changes, last change 00:05:39 Latest operation return code: Timeout

  Tracked by:
    Static IP Routing 0
```

When we re-activate R1's VLAN 146 Ethernet connection, the SLA instance reports itself as back up, the tracking instance reports itself as back up, and the static route with the lower administrative distance is re-installed in the routing table:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface gigabitEthernet1.146
R1(config-if)#no shutdown
```

https://t.me/learningnets

```
!
!R4#
track-sta (1) Change #4 ip sla 1, state Down->Up
%TRACK-6-STATE: 1 ip sla 1 state Down -> Up track-sta (1) ip sla 1 state Down -> Up
RT: updating static 150.1.1.1/32 (0x0)  :
    via 155.1.0.1   0 1048578
RT: updating static 150.1.1.1/32 (0x0)  :
via 155.1.146.1   0 1048578
RT: closer admin distance for 150.1.1.1, flushing 1 routes
RT: add 150.1.1.1/32 via 155.1.146.1, static metric [1/0]
RT: updating static 150.1.1.1/32 (0x0)  :
    via 155.1.0.1   0 1048578

RT: rib update return code: 17
!R4#traceroute 150.1.1.1 source 150.1.4.4

Type escape sequence to abort.
Tracing the route to 150.1.1.1
  1 155.1.146.1 40 msec *  36 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Policy Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure IPv4 default routes on R4 and R6 pointing to R1's IPv4 address from the shared Ethernet segment.
- Configure IPv4 default route on R3 pointing to R1's IPv4 address from the shared Ethernet segment.
- Configure IPv4 default route on R5 pointing to R1's DMVPN cloud IPv4 address.
- Configure IPv4 static routes on R3 for R5's Loopback0 prefix and on R5 for R3's Loopback0 prefix through the DMVPN cloud.
- Configure IPv4 policy-routing on R1 so that traffic from R4 is routed through R3 over the Ethernet link, and traffic from R6 is routed through R5 over the DMVPN cloud.
  - Create two extended access-lists on R1, named **FROM_R4** and **FROM_R6**:
  - Access-list **FROM_R4** should match all IPv4 traffic sourced from R4's Ethernet segment.
  - Access-list **FROM_R6** should match all IPv4 traffic sourced from R6's Ethernet segment.
  - Use traceroute on R4 and R6 for R3's and R5's Loopback0 prefixes to verify your configuration.

Configuration

```
R1:
```

https://t.me/learningnets

```
ip access-list extended FROM_R4
 permit ip host 155.1.146.4 any
!
ip access-list extended FROM_R6
 permit ip host 155.1.146.6 any
!
route-map POLICY_ROUTING permit 10
 match ip address FROM_R4
 set ip next-hop 155.1.13.3
!
route-map POLICY_ROUTING permit 20
 match ip address FROM_R6
 set ip next-hop 155.1.0.5
!
interface GigabitEthernet1.146
 ip policy route-map POLICY_ROUTING
```

R3:

```
ip route 0.0.0.0 0.0.0.0 155.1.13.1
ip route 150.1.5.5 255.255.255.255 155.1.0.5
```

R4:

```
ip route 0.0.0.0 0.0.0.0 155.1.146.1
```

R5:

```
ip route 0.0.0.0 0.0.0.0 155.1.0.1
ip route 150.1.3.3 255.255.255.255 155.1.0.3
```

R6:

```
ip route 0.0.0.0 0.0.0.0 155.1.146.1
```

Verification

Policy routing allows the router to forward traffic based on user-defined criteria without even consulting the IP routing table. In this example, we can see that R1 does not have routing information for either of the Loopbacks of R3 and R5, so it cannot route locally originated traffic.

```
R1#show ip route 150.1.3.3
% Subnet not in table
!R1#show ip route 150.1.5.5
% Subnet not in table
!R1#debug ip packet
```

https://t.me/learningnets

```
IP packet debugging is on
!R1#ping 150.1.3.3 repeat 1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:.
Success rate is 0 percent (0/1)
!
IP: s=150.1.1.1 (local), d=150.1.3.3, len 100, local feature, feature skipped, Logical MN local(14), rtype 0, forus
IP: s=150.1.1.1 (local), d=150.1.3.3, len 100, unroutable.
!R1#ping 150.1.5.5 repeat 1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:.
Success rate is 0 percent (0/1)
!
IP: s=150.1.1.1 (local), d=150.1.5.5, len 100, local feature, feature skipped, Logical MN local(14), rtype 0, forus
IP: s=150.1.1.1 (local), d=150.1.5.5, len 100, unroutable.
```

If traffic is received inbound on R1's VLAN 146 Ethernet segment and is sourced from R4's or R6's IPv4 addresses attached to VLAN 146, it is policy-routed accordingly to the route-map attached to the interface:

https://t.me/learningnets

```
R4#traceroute 150.1.3.3

Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 63 msec 40 msec 17 msec 2 155.1.13.3 16 msec

!R4#traceroute 150.1.5.5

Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 12 msec 3 msec 4 msec 2 155.1.13.3 2 msec 4 msec 2 msec
 3 155.1.0.5 9 msec *  4 msec

!R6#traceroute 150.1.3.3

Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 11 msec 4 msec 2 msec 2 155.1.0.5 3 msec 1 msec 3 msec
 3 155.1.0.3 3 msec *  5 msec

!R6#traceroute 150.1.5.5

Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 7 msec 2 msec 8 msec 2 155.1.0.5 5 msec *  2 msec
```

Verify policy-routing configuration and that traffic has matched the ACL:

```
R1#show ip policy

Interface      Route map Gi1.146         POLICY_ROUTING
!R1#show ip interface gigabitEthernet 1.146 | i Policy
 Policy routing is enabled, using route map POLICY_ROUTING
  BGP Policy Mapping is disabled
  Input features: Policy Routing, MCI Check
!R1#show route-map
 route-map POLICY_ROUTING, permit, sequence 10
  Match clauses: ip address (access-lists): FROM_R4
```

https://t.me/learningnets

```
  Set clauses:
    ip next-hop 155.1.13.3
Nexthop tracking current: 155.1.13.3
155.1.13.3, fib_nh:7F01B9C1BD10,oce:7F01B9F6EDD8,status:1
 Policy routing matches: 9 packets, 414 bytes
route-map POLICY_ROUTING, permit, sequence 20
  Match clauses: ip address (access-lists): FROM_R6
  Set clauses:
    ip next-hop 155.1.0.5
Nexthop tracking current: 155.1.0.5
155.1.0.5, fib_nh:7F01B9C1BCB0,oce:7F01B9F6FAF8,status:1
 Policy routing matches: 9 packets, 414 bytes
```

R1's route-map used for policy routing does not match traffic sourced from other interfaces of R4 and R6, so this traffic is dropped when it is by R1 inbound on its VLAN 146:

```
R4#ping 150.1.5.5 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 150.1.4.4 .....
Success rate is 0 percent (0/5)
!R6#ping 150.1.5.5 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 150.1.6.6 .....

Success rate is 0 percent (0/5)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Reliable Policy Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure IPv4 default routes on R4 and R6 pointing to R1's IPv4 address from the shared Ethernet segment.
- Configure IPv4 default route on R3 pointing to R1's IPv4 address from the shared Ethernet segment.
- Configure IPv4 default route on R5 pointing to R1's DMVPN cloud IPv4 address.
- Configure IPv4 static routes on R3 for R5's Loopback0 prefix and on R5 for R3's Loopback0 prefix through the DMVPN cloud.
- Configure R1 and R5 to run CDP over the DMVPN cloud with each other.
- Configure an IP SLA instance on R1 that pings R3's connection to VLAN 13 every five seconds.
- Configure IPv4 policy-routing on R1 so that traffic from R4 is routed through R3 over the Ethernet link, and traffic from R6 is routed through R5 over the DMVPN cloud.
  - Create two extended access-lists on R1, named **FROM_R4** and **FROM_R6**:
    - Access-list **FROM_R4** should match all IPv4 traffic sourced from R4's Ethernet segment.
    - Access-list **FROM_R6** should match all IPv4 traffic sourced from R6's Ethernet segment.
  - Use traceroute on R4 and R6 for R3's and R5's Loopback0 prefixes to verify your configuration.
- Modify R1's policy routing so that if R1 loses ICMP reachability to R3, traffic from R4

https://t.me/learningnets

is rerouted to R5 over the DMVPN cloud.

- Modify R1's policy routing so that if R1 loses R5 as a CDP neighbor, traffic from R6 is rerouted to R3 over the Ethernet link.

Configuration

```
R1:
ip sla 1
 icmp-echo 155.1.13.3 source-interface GigabitEthernet1.13
 frequency 5
!
ip sla schedule 1 start-time now life forever
track 1 ip sla 1 state
!
ip access-list extended FROM_R4
 permit ip host 155.1.146.4 any
!
ip access-list extended FROM_R6
 permit ip host 155.1.146.6 any
!
route-map POLICY_ROUTING permit 10
 match ip address FROM_R4
 set ip next-hop verify-availability 155.1.13.3 1 track 1
 set ip default next-hop 155.1.0.5
!
route-map POLICY_ROUTING permit 20
 match ip address FROM_R6
 set ip next-hop 155.1.0.5
 set ip next-hop verify-availability
 set ip default next-hop 155.1.13.3
!
interface GigabitEthernet1.146
 ip policy route-map POLICY_ROUTING
!
interface Tunnel0
 cdp enable

R3:
ip route 0.0.0.0 0.0.0.0 155.1.13.1
ip route 150.1.5.5 255.255.255.255 155.1.0.5

R4:
ip route 0.0.0.0 0.0.0.0 155.1.146.1
```

https://t.me/learningnets

```
R5:

ip route 0.0.0.0 0.0.0.0 155.1.0.1
ip route 150.1.3.3 255.255.255.255 155.1.0.3
!
interface Tunnel0
 cdp enable

R6:

ip route 0.0.0.0 0.0.0.0 155.1.146.1
```

Verification

Verify the IP SLA configuration and its state, and also that R1 and R5 are CDP neighbors over the DMVPN cloud.

```
R1#show ip sla configuration
IP SLAs Infrastructure Engine-III Entry number: 1
Owner:
Tag:
Operation timeout (milliseconds): 5000 Type of operation to perform: icmp-echo
Target address/Source interface: 155.1.13.3/GigabitEthernet1.13
Type Of Service parameter: 0x0
Request size (ARR data portion): 28
Verify data: No
Vrf Name:
Schedule: Operation frequency (seconds): 5  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
   Number of statistic hours kept: 2
   Number of statistic distribution buckets kept: 1
   Statistic distribution interval (milliseconds): 20
Enhanced History:
History Statistics:
   Number of history Lives kept: 0
   Number of history Buckets kept: 15
```

https://t.me/learningnets

```
   History Filter Type: None
!R1#show ip sla statistics
IPSLAs Latest Operation Statistics

IPSLA operation id: 1
        Latest RTT: 1 milliseconds
Latest operation start time: 16:40:43 UTC Thu May 15 2014 Latest operation return code: OK
Number of successes: 86
Number of failures: 0
Operation time to live: Forever
!R1#show track
Track 1
  IP SLA 1 state State is Up
    1 change, last change 00:08:43 Latest operation return code: OK
  Latest RTT (millisecs) 1
  Tracked by:
    Route Map 0
!R1#show cdp neighbors tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R5               Tunnel0           175              R I   CSR1000V  Tunnel0



Total cdp entries displayed : 1
```

Verify that traffic is policy-routed as requested.

```
R4#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 16 msec 4 msec 2 msec 2 155.1.13.3 3 msec *  2 msec
!R4#traceroute 150.1.5.5
Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 7 msec 2 msec 6 msec 2 155.1.13.3 3 msec 5 msec 5 msec
 3 155.1.0.5 6 msec *  9 msec
!R6#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
```

https://t.me/learningnets

```
  1 155.1.146.1 12 msec 24 msec 14 msec 2 155.1.0.5 23 msec 50 msec 22 msec
3 155.1.0.3 5 msec *  2 msec
!R6#traceroute 150.1.5.5
Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 14 msec 2 msec 15 msec 2 155.1.0.5 14 msec *  3 msec
```

Verify policy-routing configuration and that traffic has matched the ACL, and note the tracking object in the **UP** state.

```
R1#show ip policy
Interface      Route map Gi1.146         POLICY_ROUTING
!R1#show ip interface gigabitEthernet 1.146 | i Policy
Policy routing is enabled, using route map POLICY_ROUTING
  BGP Policy Mapping is disabled
  Input features: Policy Routing, MCI Check
!R1#show route-map
route-map POLICY_ROUTING, permit, sequence 10
  Match clauses:
    ip address (access-lists): FROM_R4
  Set clauses: ip next-hop verify-availability 155.1.13.3 1 track 1  [up]
    ip default next-hop 155.1.0.5 Policy routing matches: 12 packets, 552 bytes
route-map POLICY_ROUTING, permit, sequence 20
  Match clauses:
    ip address (access-lists): FROM_R6
  Set clauses: ip next-hop 155.1.0.5
    ip next-hop verify-availability
    ip default next-hop 155.1.13.3 Policy routing matches: 9 packets, 414 bytes
```

Because a regular policy routing configuration is only locally significant, network failures do not automatically update the routing policy of the router. To resolve this design problem, R1 needs some way to track end-to-end reachability on these links used for the outbound forwarding through policy routing. The two ways illustrated in this example are through the IP SLA and Enhanced Object Tracking features, and through CDP. With IP SLA configured, R1 tracks the end-to-end circuit status of VLAN 13 through ICMP ping. When R3's connection to VLAN 13 goes down, R1's SLA instance reports its status down, which in turn causes the tracked object to go down. The tracked object is called from the route-map syntax `set ip next-hop verify-availability 155.1.13.3 1 track 1` . This means that if tracked object 1 goes down, do not use the next-hop 155.1.13.3. Instead, this route-map sequence fails

https://t.me/learningnets

over to the “default” next-hop of 155.1.0.5. Let's disable R3's Ethernet link on VLAN 13:

```
R1#debug track state
track state debugging enabled
!R3#configure terminal
 R3(config)#interface gigabitEthernet1.13
R3(config-subif)#shutdown
```

With debug track being enabled on R1, the following log message should be displayed; verify that tracking object state is down.

```
R1:
%TRACK-6-STATE: 1 ip sla 1 state Up -> Down
!R1#show track
Track 1
  IP SLA 1 state State is Down

    2 changes, last change 00:02:55
  Latest operation return code: Timeout
  Tracked by:
    Route Map 0
```

Verify that traffic received from R4 is now rerouted over the DMVPN cloud, based on the `set ip default next-hop 155.1.0.5` route-map entry.

```
R4#traceroute 150.1.5.5
Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 15 msec 2 msec 2 msec 2 155.1.0.5 4 msec *  3 msec
```

Re-activate R3's Ethernet link on VLAN 13.

```
R3#configure terminal
R3(config)#interface gigabitEthernet1.13
R3(config-subif)#no shutdown
```

https://t.me/learningnets

With CDP tracking for policy routing, R1 looks into the CDP table to see if there is a neighbor installed with the IP address that matches the next-hop value being set in the route-map. In this case, the syntax `set ip next-hop 155.1.0.5 , set ip next-hop verify-availability` and `set ip default next-hop 155.1.13.3` means if there is no CDP neighbor with the IP address 155.1.0.5, traffic that matches this sequence will be routed to 155.1.13.3. Let's disable R1's DMVPN interface to trigger CDP failure:

```
R1#configure terminal
 R1(config)#interface Tunnel0
R1(config-if)#shutdown
```

Normally, you would disable R5's DMVPN interface to trigger CDP failure on R1, but on CSR 1000v routers, it seems that CDP next-hop tracking does not work as expected.

Slowly, after 180 seconds (the default CDP holdtime), the CDP entry of R5 will timeout from R1's CDP table.

```
R1#show cdp neighbors Tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R5               Tunnel0 55
               R I   CSR1000V  Tunnel0

Total cdp entries displayed : 1
!R1#show cdp neighbors Tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
Total cdp entries displayed : 0
```

Verify that traffic received from R6 is now rerouted over the Ethernet link to R3, based on the `set ip default next-hop 155.1.13.3` route-map entry.

```
R6#traceroute 150.1.3.3
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.146.1 6 msec 8 msec 2 msec 2 155.1.13.3 5 msec *  6 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

Local Policy Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R3 with an IPv4 static route for R5's Loopback0 through the DMVPN cloud and for R1's Loopback0 through VLAN 13.
- Configure R5 with IPv4 static routes for R1's and R3's Loopback0 through the DMVPN cloud.
- Create two access-lists named **TO_R3** and **TO_R5** on R1.
  - Access-list **TO_R3** should match all packets sourced from R1's Loopback0 going to the Loopback0 network of R3.
  - Access-list **TO_R5** should match all packets sourced from R1's Loopback0 going to the Loopback0 network of R5.
- Configure local policy-routing on R1 as follows:
  - Locally generated traffic matched by the list **TO_R3** is routed out the DMVPN cloud to R5.
  - Locally generated traffic matched by the list **TO_R5** is routed out the Ethernet link to R3.
  - Use traceroute on R1 for R3's and R5's Loopback0 networks to verify that this configuration is functional.

Configuration

```
R1:
```

https://t.me/learningnets

```
ip access-list extended TO_R3
 permit ip host 150.1.1.1 host 150.1.3.3
!
ip access-list extended TO_R5
 permit ip host 150.1.1.1 host 150.1.5.5
!
route-map LOCAL_POLICY permit 10
 match ip address TO_R3
 set ip next-hop 155.1.0.5
!
route-map LOCAL_POLICY permit 20
 match ip address TO_R5
 set ip next-hop 155.1.13.3
!
ip local policy route-map LOCAL_POLICY

R3:
ip route 150.1.1.1 255.255.255.255 155.1.13.1
ip route 150.1.5.5 255.255.255.255 155.1.0.5

R5:

ip route 150.1.1.1 255.255.255.255 155.1.0.1
ip route 150.1.3.3 255.255.255.255 155.1.0.3
```

Verification

Local policy routing is similar in operation to normal policy routing, except that it affects locally generated traffic from the router instead of traffic transiting the router, like received inbound on an interface. In the below output, we can see that R1 does not have a route to either of the destinations 150.1.3.3 or 150.1.5.5, but traffic is successfully routed because of the locally configured policy:

```
R1#show ip route 150.1.3.3
% Subnet not in table
!R1#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.5 4 msec 1 msec 2 msec
  2 155.1.0.3 2 msec *  2 msec
!R1#show ip route 150.1.5.5
% Subnet not in table
!R1#traceroute 150.1.5.5
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.13.3 3 msec 2 msec 1 msec

  2 155.1.0.5 2 msec *  2 msec
```

Verify policy-routing configuration and that traffic has matched the ACL:

```
R1#show ip policy
Interface      Route map local           LOCAL_POLICY
!R1#show route-map
route-map LOCAL_POLICY, permit, sequence 10
  Match clauses:
    ip address (access-lists): TO_R3
  Set clauses: ip next-hop 155.1.0.5
 Policy routing matches: 44 packets, 2925 bytes
route-map LOCAL_POLICY, permit, sequence 20
  Match clauses:
    ip address (access-lists): TO_R5
  Set clauses: ip next-hop 155.1.13.3
 Policy routing matches: 32 packets, 2477 bytes
```

Pitfall

Note that when the remote devices receive traffic from R1, it is sourced from the Loopback0 interface of R1. Normally the router uses the IP address of the outgoing interface in the routing table as the source IP address in its own packets. However, because the routing table is not consulted for the lookup, you may see inconsistencies in the source address of the local traffic. This behavior could have a negative impact on protocols such as BGP, which need to agree on the source and destination IP addresses for a peering.

```
R1#debug ip icmp
ICMP packet debugging is on
!R1#ping 150.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
!ICMP: echo reply rcvd, src 150.1.3.3, dst 150.1.1.1
, topology BASE, dscp 0 topoid 0ICMP: echo reply rcvd, src 150.1.3.3, dst 150.1.1.1
, topology BASE, dscp 0 topoid 0ICMP: echo reply rcvd, src 150.1.3.3, dst 150.1.1.1
, topology BASE, dscp 0 topoid 0ICMP: echo reply rcvd, src 150.1.3.3, dst 150.1.1.1
, topology BASE, dscp 0 topoid 0
```

https://t.me/learningnets

```
ICMP: echo reply rcvd, src 150.1.3.3, dst 150.1.1.1
, topology BASE, dscp 0 topoid 0
!R1#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.0.5 4 msec 0 msec 1 msec
  2 155.1.0.3 2 msec
!
ICMP: time exceeded rcvd from 155.1.0.5
ICMP: time exceeded rcvd from 155.1.0.5
ICMP: time exceeded rcvd from 155.1.0.5 ICMP: dst (150.1.1.1)
 port unreachable rcv from 155.1.0.3  *  2 msec ICMP: dst (150.1.1.1)
 port unreachable rcv from 155.1.0.3
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

GRE Tunneling

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Create a GRE tunnel between R1 and R3 using the IPv4 addresses 155.13.0.Y/24, where **Y** is the router number.
- The tunnel should be sourced from and destined to these devices' VLAN 13 IPv4 addresses.
- Configure static routes so that traffic between R1 and R3 Loopback0 networks is routed over the tunnel.
- Use traceroute on R1 and R3 to verify that this configuration is functional.

Configuration

```
R1:
interface Tunnel13
 ip address 155.13.0.1 255.255.255.0
 tunnel mode gre ip
 tunnel source 155.1.13.1
 tunnel destination 155.1.13.3
 no shutdown
!
ip route 150.1.3.3 255.255.255.255 Tunnel13

R3:
```

https://t.me/learningnets

```
interface Tunnel13
 ip address 155.13.0.3 255.255.255.0
 tunnel mode gre ip
 tunnel source 155.1.13.3
 tunnel destination 155.1.13.1
 no shutdown
!
ip route 150.1.1.1 255.255.255.255 Tunnel13
```

Verification

Generic Routing Encapsulation (GRE) tunneling is used to take another protocol payload, such as IPv4, IPv6, IPX, etc., and tunnel it over an IPv4 or IPv6 transit network, by using IP protocol number 47. In this case, GRE is used to tunnel IPv4 packets between the Loopback0 networks of R1 and R3. Because the GRE tunnel is configured as point-to-point, static routes can be configured to point out the Tunnel directly, and a next-hop address is not required. We can verify that traffic between these networks is going out the tunnel because the traceroute output shows only one hop between the networks, which is the GRE tunnel.

```
R1#show ip route 150.1.3.3
Routing entry for 150.1.3.3/32
  Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel0
      Route metric is 0, traffic share count is 1
!R1#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.13.0.3 4 msec *  2 msec
!R3#show ip route 150.1.1.1
Routing entry for 150.1.1.1/32
  Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel13
      Route metric is 0, traffic share count is 1
!R3#traceroute 150.1.1.1
Type escape sequence to abort.
Tracing the route to 150.1.1.1
VRF info: (vrf in name/id, vrf out name/id) 1 155.13.0.1 3 msec *  2 msec
```

Verify the GRE tunnel state and encapsulation, and notice that MTU has been automatically lowered to accommodate for the new IP header of 20 bytes and GRE

https://t.me/learningnets

header of 4 bytes.

```
R3#show interfaces tunnel13
Tunnel13 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 155.13.0.3/24
  MTU 17868 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255 Encapsulation TUNNEL, loopback not set
  Keepalive not set Tunnel source 155.1.13.3, destination 155.1.13.1
 Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255, Fast tunneling enabled Tunnel transport MTU 1476 bytes
```

To verify that packets are GRE encapsulated, we can use the embedded packet capture feature.

https://t.me/learningnets

```
R1#monitor capture GRE match any interface gigabitEthernet1.13 both
R1#monitor capture GRE start

!R1#ping 150.1.3.3 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
!R1#monitor capture GRE stop

!R1#show monitor capture GRE buffer brief

 ----------------------------------------------------------
 #   size   timestamp     source             destination protocol

 ----------------------------------------------------------
   0  122    0.000000    FE80:*:2E27     ->   FF02:*:0001     IPv6-ICMP
   1  142    8.135003   155.1.13.1       ->  155.1.13.3 GRE
   2  142    8.135995   155.1.13.3       ->  155.1.13.1 GRE

   3  142    8.137002   155.1.13.1       ->  155.1.13.3       GRE
   4  142    8.137002   155.1.13.3       ->  155.1.13.1       GRE
   5  142    8.137994   155.1.13.1       ->  155.1.13.3       GRE
   6  142    8.137994   155.1.13.3       ->  155.1.13.1       GRE
   7  142    8.139001   155.1.13.1       ->  155.1.13.3       GRE
   8  142    8.139001   155.1.13.3       ->  155.1.13.1       GRE
   9  142    8.139001   155.1.13.1       ->  155.1.13.3       GRE
  10  142    8.139993   155.1.13.3       ->  155.1.13.1       GRE
!R1#no monitor capture GRE
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

GRE Tunneling and Recursive Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Create a GRE tunnel between R3 and R4 using the IPv4 addresses 155.34.0.Y/24, where **Y** is the router number.
- The tunnel should be sourced from and destined to these devices' Loopback0 IPv4 addresses.
- Configure Loopback1 on R3 and R4 with IPv4 addresses 150.1.YY.YY/32, where **Y** is the router number.
- Configure RIP version 2 on R1, R3. and R4 as follows:
  - Enable it on VLAN 13 and VLAN 146.
  - Enable it on the GRE tunnel.
  - Advertise Loopback0 and Loopback1 prefixes of R3 and R4.
- Use traceroute on R3 and R4 to verify connectivity between Loopback1 networks of R3 and R4 through the GRE tunnel.
- Configure distribute-list filtering on R3 and R4 to fix the recursive routing error for the GRE tunnel.

Configuration

```
R1:
router rip
 version 2
 no auto-summary
```

https://t.me/learningnets

```
 network 155.1.0.0
```

R3:

```
interface Tunnel34
 ip address 155.34.0.3 255.255.255.0
 tunnel mode gre ip
 tunnel source 150.1.3.3
 tunnel destination 150.1.4.4
 no shutdown
!
interface Loopback1
 ip address 150.1.33.33 255.255.255.255
!
ip prefix-list STOP_RECURSIVE_ERROR seq 5 deny 150.1.3.3/32
ip prefix-list STOP_RECURSIVE_ERROR seq 10 permit 0.0.0.0/0 le 32
!
router rip
 version 2
 no auto-summary
 network 155.1.0.0
 network 150.1.0.0
 network 155.34.0.0
 distribute-list prefix STOP_RECURSIVE_ERROR out Tunnel34
```

R4:

```
interface Tunnel34
 ip address 155.34.0.4 255.255.255.0
 tunnel mode gre ip
 tunnel source 150.1.4.4
 tunnel destination 150.1.3.3
 no shutdown
!
interface Loopback1
 ip address 150.1.44.44 255.255.255.255
!
ip prefix-list STOP_RECURSIVE_ERROR seq 5 deny 150.1.4.4/32
ip prefix-list STOP_RECURSIVE_ERROR seq 10 permit 0.0.0.0/0 le 32
!
router rip
 version 2
 no auto-summary
 network 155.1.0.0
 network 150.1.0.0
 network 155.34.0.0
```

https://t.me/learningnets

```
 distribute-list prefix STOP_RECURSIVE_ERROR out Tunnel34
```

Verification

Verify the RIP routing table on R3 and R4 before enabling RIP on Tunnel34, and note that there is IPv4 connectivity between Loopback0 and Loopback1 prefixes.

```
R4#show ip route rip | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 4 subnets
R        150.1.3.3 [120/2] via 155.1.146.1, 00:00:09, GigabitEthernet1.146
R        150.1.33.33 [120/2] via 155.1.146.1, 00:00:09, GigabitEthernet1.146
      155.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
R        155.1.13.0/24 [120/1] via 155.1.146.1, 00:00:09, GigabitEthernet1.146
R        155.1.23.0/24 [120/2] via 155.1.146.1, 00:00:09, GigabitEthernet1.146
R        155.1.37.0/24 [120/2] via 155.1.146.1, 00:00:09, GigabitEthernet1.146
!R3#show ip route rip | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 4 subnets
R        150.1.4.4 [120/2] via 155.1.13.1, 00:00:16, GigabitEthernet1.13
R        150.1.44.44 [120/2] via 155.1.13.1, 00:00:16, GigabitEthernet1.13
      155.1.0.0/16 is variably subnetted, 10 subnets, 2 masks
R        155.1.45.0/24 [120/2] via 155.1.13.1, 00:00:16, GigabitEthernet1.13
R        155.1.146.0/24 [120/1] via 155.1.13.1, 00:00:16, GigabitEthernet1.13
!R3#ping 150.1.4.4 source loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.1.3.3 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
!R3#ping 150.1.44.44 source loopback1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.44.44, timeout is 2 seconds:
Packet sent with a source address of 150.1.33.33 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/8/23 ms
```

When RIP is enabled on the GRE tunnel of R3 and R4, a recursive routing error will happen on either R3 or R4, based on the order of configuration. A recursive routing error is when a lookup for prefix **X** points at prefix **Y** for the next-hop, and a lookup for prefix **Y** points at prefix **X** for the next-hop. For tunnel interfaces, this occurs when the tunnel destination is dynamically learned through the tunnel interface

https://t.me/learningnets

itself. This problem can be easily identified by the IOS log message **%TUN-5-RECURDOWN: Tunnel34 temporarily disabled due to recursive routing**. The step-by-step process by which this error occurs is as follows. First, the GRE tunnel interface is enabled and goes into **UP** state as the router has a route for tunnel destination learned through RIP; IOS logs the message **%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel34, changed state to up** to signal this. At this moment, let's enable RIP over the tunnel interface, which will determine both routers, R3 and R4, to prefer routes learned over the GRE tunnel against routes learned over the physical Ethernet path, because of the lower metric/hop-count of 1 on the GRE tunnel versus the metric/hop-count of 2 on the physical Ethernet path:

```
R4#show ip route rip | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 4 subnetsR        150.1.3.3 [120/1] via 155.34.0.3, 00:00:02, Tunnel34

R        150.1.33.33 [120/1] via 155.34.0.3, 00:00:02, Tunnel34
      155.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
R        155.1.13.0/24 [120/1] via 155.34.0.3, 00:00:02, Tunnel34
                       [120/1] via 155.1.146.1, 00:00:22, GigabitEthernet1.146
R        155.1.23.0/24 [120/1] via 155.34.0.3, 00:00:02, Tunnel34
R        155.1.37.0/24 [120/1] via 155.34.0.3, 00:00:02, Tunnel34
```

Because at this moment R4 learns about the GRE tunnel destination through the GRE tunnel itself, a route-recursive error is triggered, and the Tunnel interface goes to **DOWN** state. This will allow the router to re-learn the RIP routes over the physical path, but this process will repeat itself when R4 will bring **UP** again the GRE tunnel interface:

```
R4:
%ADJ-5-PARENT: Midchain parent maintenance for IP midchain out of Tunnel34 - looped chain attempting to stack
%TUN-5-RECURDOWN: Tunnel34 temporarily disabled due to recursive routing
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel34, changed state to down
!R4#show ip route rip | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 4 subnets
R        150.1.3.3 [120/2] via 155.1.146.1, 00:00:05, GigabitEthernet1.146

R        150.1.33.33 [120/2] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
      155.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
R        155.1.13.0/24 [120/1] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
R        155.1.23.0/24 [120/2] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
```

https://t.me/learningnets

```
R        155.1.37.0/24 [120/2] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
```

In general, there are several solutions to the problem, depending on the IPv4 addressing scheme, routing protocol being used, and network design:

- Do not advertise in the routing protocol the same networks over both the physical/Ethernet path and the tunneling/GRE path.
- Do not advertise in the routing protocol used over the tunneling/GRE path the tunnel endpoints.
- Use route filtering techniques to filter tunnel endpoints IPv4 addresses from being learned over the GRE tunnel.
- Do not use same routing protocol over both the physical and tunneling paths; this is recommended but it requires correct routing protocol configuration.

In this case, one of multiple route filtering solutions has been selected, basically configuring R3 and R4 to not advertise Loopback0 prefix over the GRE tunnel. After applying the filtering, the network is stable and routes are correctly learned over both physical and tunneling paths. Note that route filtering was configured strictly to fix the recursive routing problem; some Ethernet link subnets will be learned over the GRE tunnel, which is functional but maybe not optimal.

https://t.me/learningnets

```
R3#show ip route rip | b Gateway

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 4 subnets
R        150.1.4.4 [120/2] via 155.1.13.1, 00:00:01, GigabitEthernet1.13
R        150.1.44.44 [120/1] via 155.34.0.4, 00:00:15, Tunnel34

      155.1.0.0/16 is variably subnetted, 10 subnets, 2 masks
R        155.1.45.0/24 [120/1] via 155.34.0.4, 00:00:15, Tunnel34
R        155.1.146.0/24 [120/1] via 155.34.0.4, 00:00:15, Tunnel34
                        [120/1] via 155.1.13.1, 00:00:01, GigabitEthernet1.13
!R4#show ip route rip | b Gateway

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 4 subnets
R        150.1.3.3 [120/2] via 155.1.146.1, 00:00:23, GigabitEthernet1.146
R        150.1.33.33 [120/1] via 155.34.0.3, 00:00:04, Tunnel34

      155.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
R        155.1.13.0/24 [120/1] via 155.34.0.3, 00:00:04, Tunnel34
                       [120/1] via 155.1.146.1, 00:00:23, GigabitEthernet1.146
R        155.1.23.0/24 [120/1] via 155.34.0.3, 00:00:04, Tunnel34
R        155.1.37.0/24 [120/1] via 155.34.0.3, 00:00:04, Tunnel34
```

Verify there is IPv4 connectivity between Loopback0 and Loopback1 prefixes of R3 and R4, and that traffic for Loopback1 is routed over the GRE tunnel.

```
R4#ping 150.1.3.3 source loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 150.1.4.4 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/10/23 ms
!R4#ping 150.1.33.33 source loopback1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.33.33, timeout is 2 seconds:
Packet sent with a source address of 150.1.44.44 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/13 ms
!R4#traceroute 150.1.33.33
Type escape sequence to abort.
Tracing the route to 150.1.33.33
VRF info: (vrf in name/id, vrf out name/id) 1 155.34.0.3 18 msec *  3 msec
```

https://t.me/learningnets

```
!R3#traceroute 150.1.44.44

Type escape sequence to abort.
Tracing the route to 150.1.44.44
VRF info: (vrf in name/id, vrf out name/id) 1 155.34.0.4 6 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

GRE Reliable Backup Interface

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure two GRE tunnels between R4 and R5 as follows:
  - **Tunnel45** with IPv4 addresses 155.45.0.Y/24, where **Y** is the router number, sourced from VLAN 45 Ethernet link.
  - **Tunnel100** with IPv4 addresses 155.100.0.Y/24, where **Y** is the router number, sourced from VLAN 100 Ethernet link.
- Configure IPv4 static routes on R5 for R4's Loopback0 interface via both the DMVPN cloud and Tunnel45.
- Configure IPv4 static routes on R4 for R5's Loopback0 interface via both the DMVPN cloud and Tunnel45.
- The static routes on R4 and R5 via the DMVPN cloud should have a higher administrative distance than those on Tunnel45.
- Configure the backup interface feature on R4 and R5 so that if the Tunnel100 goes down, Tunnel45 is activated.
- To verify this configuration ensure that traffic between Loopback0 prefixes of R4 and R5 is routed out DMVPN cloud:
  - if R4's VLAN 100 interface is disabled, traffic is rerouted out on Tunnel45.

Configuration

```
R4:
```

https://t.me/learningnets

```
interface Tunnel45
 ip address 155.45.0.4 255.255.255.0
 tunnel mode gre ip
 tunnel source 155.1.45.4
 tunnel destination 155.1.45.5
!
interface Tunnel100
 ip address 155.100.0.4 255.255.255.0
 tunnel mode gre ip
 tunnel source 169.254.100.4
 tunnel destination 169.254.100.5
 keepalive 1 3
 backup interface Tunnel45
!
ip route 150.1.5.5 255.255.255.255 Tunnel45 10
ip route 150.1.5.5 255.255.255.255 155.1.0.5 20
```

R5:

```
interface Tunnel45
 ip address 155.45.0.5 255.255.255.0
 tunnel mode gre ip
 tunnel source 155.1.45.5
 tunnel destination 155.1.45.4
!
interface Tunnel100
 ip address 155.100.0.5 255.255.255.0
 tunnel mode gre ip
 tunnel source 169.254.100.5
 tunnel destination 169.254.100.4
 keepalive 1 3
 backup interface Tunnel45
!
ip route 150.1.4.4 255.255.255.255 Tunnel45 10
ip route 150.1.4.4 255.255.255.255 155.1.0.4 20
```

Verification

By default the state of a point-to-point GRE interface is determined by routing availability for the tunnel destination, thus as long as the router has a route for the tunnel destination, tunnel interface state will be **UP**. This, however, does not account for possible transit problems or maybe devices filtering GRE which is IP protocol number 47. To fix the problem, GRE keepalives can be enabled on point-to-

https://t.me/learningnets

point GRE tunnels. GRE keepalives are implemented in such a way that it can be enabled on one side of the tunnel only, which means only that side can track end-to-end GRE connectivity between the tunnel endpoints and update the GRE interface status accordingly. GRE keepalives are enabled with interface level command `keepalive <interval> <number_of_retries>` , with interval defining the frequency in seconds for sending keepalives and retries defining the maximum number of keepalives being sent after first failed keepalive before tunnel interface state changes to **DOWN.**. So with configuration `keepalive 1 3` , the router will send a GRE keepalive each 1 second, upon first failed keepalive will send additional 3 keepalives and if all failed, interface goes into **DOWN** state.

The state of multipoint GRE tunnel interface, such as those used in DMVPN scenarios, cannot be monitored through GRE keepalives, as there is no single destination for the tunnel. The mGRE tunnel interface is always in the **UP** state. In DMVPN setups, the spoke mGRE tunnel interface can be determined by the spoke being able to successfully register to the hub or not via NHRP if the `if-state nhrp` interface level command is configured, but this is not possible for the hub, so the hub interface is always in the **UP** state.

The design problem in this case is that R4 and R5 cannot actively determine if the DMVPN path is still functional. Based on the NHRP entries and possibly IPsec state if configured with DMVPN, both hub and spokes will know if there is hub-to-spoke connectivity or not. However this does not affect the mGRE interface which is always in the **UP** state and if static routing is configured over DMVPN, this may result in traffic blackholing. If dynamic routing is used over DMVPN cloud and problems appear in the transit path, this will trigger the routing protocol to converge over alternate paths if available.

In this case the problem is fixed through the use of backup interface functionality. Point-to-point GRE Tunnel100 interface is using the same source and destination IPv4 addresses as the DMVPN network between R4 and R5. By implementing GRE keepalive, any problems in the transit path which may affect the DMVPN network will be detected by Tunnel100 and cause the interface status to go **DOWN**. When Tunnel100 interface goes **DOWN**, this will trigger the backup interface which is Tunnel45 to go **UP**, which also activates the static route configured over Tunnel45.

Verify that backup interface is correctly configured, and Tunnel45 waits for Tunnel100 to go **DOWN** in order to become active:

```
R5#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   normal operation
!R5#sho ip interface brief | i Tunnel
```

https://t.me/learningnets

```
Tunnel0                155.1.0.5       YES manual up                    up
Tunnel45               155.45.0.5      YES manual standby mode          down

Tunnel100              155.100.0.5     YES manual up                    up
!R4#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   normal operation
!R4#sho ip interface brief | i Tunnel
Tunnel0                155.1.0.4       YES manual up                    up
Tunnel45               155.45.0.4      YES manual standby mode          down

Tunnel100              155.100.0.4     YES manual up                    up
```

Verify that traffic between Loopback0 is primarily routed over DMVPN cloud:

```
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32  Known via "static", distance 20
, metric 0 (connected)
  Routing Descriptor Blocks: * 155.1.0.4
      Route metric is 0, traffic share count is 1
!R5#ping 150.1.4.4 source loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.1.5.5 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
!R5#traceroute 150.1.4.4 source loopback0
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.4 7 msec *  8 msec
```

Enable debugging on R5 and disable R4's VLAN 100 interface, note that Tunnel45 is activated:

```
R5#debug backup
Backup events debugging is on
!R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet1.100
R4(config-subif)#shutdown
!R5#
BACKUP(Tunnel100): event = primary interface went down
BACKUP(Tunnel100): changed state to "waiting to backup"
BACKUP(Tunnel100): event = timer expired on primary
BACKUP(Tunnel100): secondary interface (Tunnel45) made active
```

https://t.me/learningnets

```
BACKUP(Tunnel100): changed state to "backup mode"
!
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel45, changed state to up
BACKUP(Tunnel45): event = secondary interface came up

%LINK-3-UPDOWN: Interface Tunnel45, changed state to up
```

Verify that backup interface is now active:

```
R5#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   backup mode
!R5#sho ip interface brief | i Tunnel
Tunnel0                155.1.0.5       YES manual up                    up
Tunnel45               155.45.0.5      YES manual up                    up

Tunnel100              155.100.0.5     YES manual up                    down
```

Verify that traffic between Loopback0 is now routed over GRE Tunnel45, although both DMVPN and Tunnel45 interfaces are active, static route is preferred via Tunnel45 due to lower administrative distance:

```
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32  Known via "static", distance 10
, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Tunnel45
      Route metric is 0, traffic share count is 1
!R5#show ip static route
Codes: M - Manual static, A - AAA download, N - IP NAT, D - DHCP,
       G - GPRS, V - Crypto VPN, C - CASA, P - Channel interface processor,
       B - BootP, S - Service selection gateway
       DN - Default Network, T - Tracking object
       L - TL1, E - OER, I - iEdge
       D1 - Dot1x Vlan Network, K - MWAM Route
       PP - PPP default route, MR - MRIPv6, SS - SSLVPN
       H - IPe Host, ID - IPe Domain Broadcast
       U - User GPRS, TE - MPLS Traffic-eng, LI - LIIN
       IR - ICMP Redirect
Codes in []: A - active, N - non-active, B - BFD-tracked, D - Not Tracked, P - permanent

Static local RIB for default
M  150.1.4.4/32 [10/0] via Tunnel45 [A]
M               [20/0] via 155.1.0.4 [N]
!R5#ping 150.1.4.4 source loopback0
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.1.5.5 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
!R5#traceroute 150.1.4.4 source loopback0
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.45.0.4 5 msec *  2 msec
```

When R4's VLAN 100 interface is re-enabled, Tunnel100 interface is re-activated as GRE keepalives are functional and all traffic is re-routed over the DMVPN cloud:

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet1.100
R4(config-subif)#no shutdown
!R5#show backup
Primary Interface          Secondary Interface        Status
-------------------------  -------------------------  ------
Tunnel100                  Tunnel45                   normal operation
!R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "static", distance 20, metric 0 (connected)
  Routing Descriptor Blocks: * 155.1.0.4
      Route metric is 0, traffic share count is 1
!R5#traceroute 150.1.4.4 source loopback0
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.4 3 msec *  4 msec
```

Verify that GRE tunnel keepalives are enabled on Tunnel100:

```
R5#show interfaces tunnel100
Tunnel100 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 155.100.0.5/24
  Backup interface Tunnel45, failure delay 0 sec, secondary disable delay 0 sec,
  kickin load not set, kickout load not set
  MTU 17868 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set Keepalive set (1 sec), retries 3
  Tunnel source 169.254.100.5, destination 169.254.100.4
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
```

https://t.me/learningnets

```
    Checksumming of packets disabled
  Tunnel TTL 255, Fast tunneling enabled
  Tunnel transport MTU 1476 bytes
!R5#debug tunnel keepalive
Tunnel keepalive debugging is on
!Tunnel100: sending keepalive, 169.254.100.4->169.254.100.5 (len=24 ttl=255), counter=1
Tunnel100: keepalive received, 169.254.100.4->169.254.100.5 (len=24 ttl=253), resetting counter

Tunnel100: sending keepalive, 169.254.100.4->169.254.100.5 (len=24 ttl=255), counter=1
Tunnel100: keepalive received, 169.254.100.4->169.254.100.5 (len=24 ttl=253), resetting counter
Tunnel100: sending keepalive, 169.254.100.4->169.254.100.5 (len=24 ttl=255), counter=1
Tunnel100: keepalive received, 169.254.100.4->169.254.100.5 (len=24 ttl=253), resetting counter
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Routing

ODR - On-Demand Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic IP Addressing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Disable all physical interfaces on R1, R2, R3, R4, and R5, with the exception of their connections to the DMVPN cloud.
- Ensure that CDP is enabled on the DMVPN cloud between these devices and enable ODR on R5.
- Ensure that all DMVPN routers have IPv4 reachability to each other's Loopback0 prefixes.

Configuration

```
R1:
interface gigabitEthernet1.13
 shutdown
!
interface gigabitEthernet1.146
 shutdown
!
interface Tunnel0
 cdp enable

R2:
interface gigabitEthernet1.23
 shutdown
!
```

https://t.me/learningnets

```
interface Tunnel0
 cdp enable
```

R3:

```
interface gigabitEthernet1.13
 shutdown
!
interface gigabitEthernet1.23
 shutdown
!
interface gigabitEthernet1.37
 shutdown
!
interface Tunnel0
 cdp enable
```

R4:

```
interface gigabitEthernet1.45
 shutdown
!
interface gigabitEthernet1.146
 shutdown
!
interface Tunnel0
 cdp enable
```

R5:

```
interface gigabitEthernet1.5
 shutdown
!
interface gigabitEthernet1.45
 shutdown
!
interface gigabitEthernet1.58
 shutdown
!
interface Tunnel0
 cdp enable
!
router odr
```

https://t.me/learningnets

Verification

On-Demand Routing (ODR) uses Cisco Discovery Protocol (CDP) to advertise connected IPv4 routes of a stub router to the hub. The hub router then advertises a default IPv4 route back to the spokes via CDP. For ODR to be functional, there should be no dynamic routing protocol configured on spokes. Assuming that the hub and spokes are already running CDP, the configuration of this feature requires only one command on the hub, **`router odr`** . In this design, CDP is disabled on the mGRE Tunnel interface of all routers by default; therefore, the additional command **`cdp enable`** is required on all routers. Verify that spoke routers are CDP neighbors with the hub router, which is R5.

```
R1#show cdp neighbors tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R5               Tunnel0           138              R I   CSR1000V  Tunnel0

Total cdp entries displayed : 1
!R2#show cdp neighbors tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R5               Tunnel0           126              R I   CSR1000V  Tunnel0

Total cdp entries displayed : 1
!R3#show cdp neighbors tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R5               Tunnel0           123              R I   CSR1000V  Tunnel0

Total cdp entries displayed : 1
!R4#show cdp neighbors tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay
```

https://t.me/learningnets

```
Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R5               Tunnel0           122              R I   CSR1000V  Tunnel0



Total cdp entries displayed : 1
!R5#show cdp neighbors tunnel0
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R2               Tunnel0           140              R I   CSR1000V  Tunnel0
R3               Tunnel0           143              R I   CSR1000V  Tunnel0
R1               Tunnel0           134              R I   CSR1000V  Tunnel0
R4               Tunnel0           149              R I   CSR1000V  Tunnel0



Total cdp entries displayed : 4
```

Verify that all spokes received the default route form the hub, and the hub has learned about the Loopback0 prefixes of spokes.

```
R1#show ip route odr | b Gateway
Gateway of last resort is 155.1.0.5 to network 0.0.0.0
o*    0.0.0.0/0 [160/1] via 155.1.0.5, 00:00:27, Tunnel0
!R2#show ip route odr | b Gateway
Gateway of last resort is 155.1.0.5 to network 0.0.0.0
o*    0.0.0.0/0 [160/1] via 155.1.0.5, 00:00:34, Tunnel0
!R3#show ip route odr | b Gateway
Gateway of last resort is 155.1.0.5 to network 0.0.0.0
o*    0.0.0.0/0 [160/1] via 155.1.0.5, 00:00:36, Tunnel0
!R4#show ip route odr | b Gateway
Gateway of last resort is 155.1.0.5 to network 0.0.0.0
o*    0.0.0.0/0 [160/1] via 155.1.0.5, 00:00:38, Tunnel0
!R5#show ip route odr | i 150.1.


      150.1.0.0/32 is subnetted, 5 subnets
o        150.1.1.1 [160/1] via 155.1.0.1, 00:00:10, Tunnel0
o        150.1.2.2 [160/1] via 155.1.0.2, 00:00:12, Tunnel0
o        150.1.3.3 [160/1] via 155.1.0.3, 00:00:10, Tunnel0
o        150.1.4.4 [160/1] via 155.1.0.4, 00:00:10, Tunnel0
```

Verify IPv4 connectivity between Loopback0 prefixes.

https://t.me/learningnets

```
R1#ping 150.1.2.2 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
!R1#ping 150.1.3.3 source loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
!R1#ping 150.1.4.4 source loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/8/31 ms
!R1#ping 150.1.5.5 source loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
```

Verify that ODR is enabled on the hub, but not on the spokes.

```
R5#show ip protocols | section odr
Routing Protocol is "odr"
  Sending updates every 60 seconds, next due in 52 seconds
  Invalid after 180 seconds, hold down 0, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Maximum path: 4
  Routing Information Sources:
    Gateway         Distance      Last Update 155.1.0.2             160      00:00:37
 155.1.0.3            160       00:00:41
 155.1.0.1            160       00:00:42
 155.1.0.4            160       00:00:40
  Distance: (default is 160)
!R1#show ip protocols | section odr
R1#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Basic Configuration

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial RIP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure RIPv2 on all interfaces in the 150.1.0.0 and 155.1.0.0 networks.
- Disable auto summarization.
- Verify your configuration by testing that all routers have full IPv4 reachability.

Configuration

```
R1 - R10:

router rip
 version 2
 network 150.1.0.0
 network 155.1.0.0
 no auto-summary
```

Verification

By default, split-horizon is enabled on interfaces, thus R5 has it enabled on its DMVPN tunnel interface. This can be verified as seen below:

```
R5#show ip interface tunnel0
```

https://t.me/learningnets

```
Tunnel0 is up, line protocol is up
  Internet address is 155.1.0.5/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1400 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.9
  Outgoing Common access list is not set
  Outgoing access list is not set
  Inbound Common access list is not set
  Inbound  access list is not set
  Proxy ARP is enabled
  Local Proxy ARP is disabled
  Security level is default Split horizon is enabled

  ICMP redirects are never sent
  ICMP unreachables are always sent
  <snip>
```

Because split-horizon rule dictates that a router cannot send updates out the same interface that it received it on (given that is selected as best path and thus present in the routing table), R5 does not reflect RIP updates between spokes. For example R2 does not receive the routes from other spokes:

```
R2#show ip route rip

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets
R        150.1.4.4 [120/2] via 155.1.0.5, 00:00:06, Tunnel0
R        150.1.5.5 [120/1] via 155.1.0.5, 00:00:06, Tunnel0
R        150.1.8.8 [120/2] via 155.1.0.5, 00:00:06, Tunnel0
R        150.1.10.10 [120/3] via 155.1.0.5, 00:00:06, Tunnel0
      155.1.0.0/16 is variably subnetted, 8 subnets, 2 masks
```

https://t.me/learningnets

```
R        155.1.5.0/24 [120/1] via 155.1.0.5, 00:00:06, Tunnel0
R        155.1.8.0/24 [120/2] via 155.1.0.5, 00:00:06, Tunnel0
R        155.1.10.0/24 [120/3] via 155.1.0.5, 00:00:06, Tunnel0
R        155.1.45.0/24 [120/1] via 155.1.0.5, 00:00:06, Tunnel0
R        155.1.58.0/24 [120/1] via 155.1.0.5, 00:00:06, Tunnel0
R        155.1.108.0/24 [120/2] via 155.1.0.5, 00:00:06, Tunnel0
```

Note that several routes, such as the Loopback of R1, are missing from R2's routing table:

```
R2#show ip route 150.1.1.1
% Subnet not in table
```

To resolve this, R5 must disable split-horizon on the DMVPN tunnel interface:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface tunnel0
R5(config-if)#no ip split-horizon
!
!R5#show ip interface tunnel0 | include Split
Split horizon is disabled
```

Now R5 reflects the RIP updates and thus R2 can learn the reachability information about the rest of the network:

```
R2#show ip route rip

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets
R        150.1.1.1 [120/2] via 155.1.0.1, 00:00:01, Tunnel0
R        150.1.3.3 [120/2] via 155.1.0.3, 00:00:01, Tunnel0
```

https://t.me/learningnets

```
R        150.1.4.4 [120/2] via 155.1.0.5, 00:00:01, Tunnel0
R        150.1.5.5 [120/1] via 155.1.0.5, 00:00:01, Tunnel0
R        150.1.6.6 [120/3] via 155.1.0.1, 00:00:01, Tunnel0
R        150.1.7.7 [120/3] via 155.1.0.3, 00:00:01, Tunnel0
R        150.1.8.8 [120/2] via 155.1.0.5, 00:00:01, Tunnel0
R        150.1.9.9 [120/4] via 155.1.0.3, 00:00:01, Tunnel0
R        150.1.10.10 [120/3] via 155.1.0.5, 00:00:01, Tunnel0
      155.1.0.0/16 is variably subnetted, 16 subnets, 2 masks
R        155.1.5.0/24 [120/1] via 155.1.0.5, 00:00:01, Tunnel0
R        155.1.7.0/24 [120/3] via 155.1.0.3, 00:00:01, Tunnel0
R        155.1.8.0/24 [120/2] via 155.1.0.5, 00:00:01, Tunnel0
R        155.1.9.0/24 [120/4] via 155.1.0.3, 00:00:01, Tunnel0
R        155.1.10.0/24 [120/3] via 155.1.0.5, 00:00:01, Tunnel0
R        155.1.13.0/24 [120/2] via 155.1.0.3, 00:00:01, Tunnel0
R        155.1.23.0/24 [120/2] via 155.1.0.3, 00:00:01, Tunnel0
R        155.1.37.0/24 [120/2] via 155.1.0.3, 00:00:01, Tunnel0
R        155.1.45.0/24 [120/1] via 155.1.0.5, 00:00:01, Tunnel0
R        155.1.58.0/24 [120/1] via 155.1.0.5, 00:00:01, Tunnel0
R        155.1.67.0/24 [120/3] via 155.1.0.1, 00:00:01, Tunnel0
R        155.1.79.0/24 [120/3] via 155.1.0.3, 00:00:01, Tunnel0
R        155.1.108.0/24 [120/2] via 155.1.0.5, 00:00:01, Tunnel0
R        155.1.146.0/24 [120/2] via 155.1.0.1, 00:00:01, Tunnel0
```

A full reachability test can be performed by pinging around the network manually or with a basic TCL shell script, as seen below:

```
R2#tclsh
R2(tcl)#foreach ADDRESS {
+>(tcl)#150.1.1.1
+>(tcl)#155.1.0.1
+>(tcl)#155.1.13.1
+>(tcl)#155.1.146.1
+>(tcl)#150.1.2.2
+>(tcl)#155.1.0.2
+>(tcl)#150.1.3.3
+>(tcl)#155.1.0.3
+>(tcl)#155.1.13.3
+>(tcl)#155.1.37.3
+>(tcl)#150.1.4.4
+>(tcl)#155.1.0.4
+>(tcl)#155.1.45.4
+>(tcl)#150.1.5.5
+>(tcl)#155.1.0.5
+>(tcl)#155.1.5.5
+>(tcl)#155.1.45.5
```

https://t.me/learningnets

```
+>(tcl)#155.1.58.5
+>(tcl)#150.1.6.6
+>(tcl)#155.1.67.6
+>(tcl)#155.1.146.6
+>(tcl)#150.1.7.7
+>(tcl)#155.1.7.7
+>(tcl)#155.1.37.7
+>(tcl)#155.1.67.7
+>(tcl)#155.1.79.7
+>(tcl)#150.1.8.8
+>(tcl)#155.1.8.8
+>(tcl)#155.1.58.8
+>(tcl)#155.1.108.8
+>(tcl)#150.1.9.9
+>(tcl)#155.1.9.9
+>(tcl)#155.1.79.9
+>(tcl)#150.1.10.10
+>(tcl)#155.1.10.10
+>(tcl)#155.1.108.10+>(tcl)#} { ping $ADDRESS }
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.13.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.13.3, timeout is 2 seconds:!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.37.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.45.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.5.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.45.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.58.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.7.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.37.7, timeout is 2 seconds:!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.79.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.8.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.8.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.58.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.108.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.9.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.79.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.10.10, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.10.10, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/8 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.108.10, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Authentication

> You must load the initial configuration files for the section, **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure RIPv2 authentication on the Tunnel interfaces in the DMVPN cloud.
  - Use the MD5 key number **1** with the password **CISCO**.
  - All the spoke routers should be learning RIP routes from R5.

- Configure clear-text RIP authentication on the segment between R1 and R6 using the password **CCIE**.

Configuration

```
R1:
key chain RIP
 key 1
  key-string CISCO
!
key chain RIP_146
 key 1
  key-string CCIE
!
interface Tunnel0
 ip rip authentication mode md5
 ip rip authentication key-chain RIP
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.146
 ip rip authentication mode text
 ip rip authentication key-chain RIP_146
```

R2 - R5

```
key chain RIP
 key 1
  key-string CISCO
!
interface Tunnel0
 ip rip authentication mode md5
 ip rip authentication key-chain RIP
```

R6:

```
key chain RIP_146
 key 1
  key-string CCIE
!
interface GigabitEthernet1.146
 ip rip authentication mode text
 ip rip authentication key-chain RIP_146
```

Verification

R1 has been configured for both MD5 and clear-text authentication, but on different RIP-enabled interfaces.

```
R1#debug ip rip
RIP protocol debugging is on RIP: received packet with MD5 authentication
RIP: received v2 update from 155.1.0.5 on Tunnel0
!
!RIP: received packet with text authentication CCIE
RIP: received v2 update from 155.1.146.6 on GigabitEthernet1.146
```

When RIP authentication is configured on all devices, R5 will correctly receive all prefixes through RIP and install it in the routing table.

```
R5#show ip route rip | b Gateway

Gateway of last resort is not set
```

https://t.me/learningnets

```
  150.1.0.0/32 is subnetted, 10 subnets
R     150.1.1.1 [120/1] via 155.1.0.1, 00:00:12, Tunnel0
R     150.1.2.2 [120/1] via 155.1.0.2, 00:00:10, Tunnel0
      150.1.3.3 [120/1] via 155.1.0.3, 00:00:07, Tunnel0
R     150.1.4.4 [120/1] via 155.1.45.4, 00:00:12, GigabitEthernet1.45
                [120/1] via 155.1.0.4, 00:00:11, Tunnel0
R     150.1.6.6 [120/2] via 155.1.0.1, 00:00:12, Tunnel0
R     150.1.7.7 [120/2] via 155.1.0.3, 00:00:07, Tunnel0
R     150.1.8.8 [120/1] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
R     150.1.9.9 [120/3] via 155.1.0.3, 00:00:07, Tunnel0
R     150.1.10.10 [120/2] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
  155.1.0.0/16 is variably subnetted, 18 subnets, 2 masks
R     155.1.7.0/24 [120/2] via 155.1.0.3, 00:00:07, Tunnel0
R     155.1.8.0/24 [120/1] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
R     155.1.9.0/24 [120/3] via 155.1.0.3, 00:00:07, Tunnel0
R     155.1.10.0/24 [120/2] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
R     155.1.13.0/24 [120/1] via 155.1.0.3, 00:00:07, Tunnel0
                    [120/1] via 155.1.0.1, 00:00:12, Tunnel0
R     155.1.37.0/24 [120/1] via 155.1.0.3, 00:00:07, Tunnel0
R     155.1.67.0/24 [120/2] via 155.1.0.3, 00:00:07, Tunnel0
                   [120/2] via 155.1.0.1, 00:00:12, Tunnel0
R     155.1.79.0/24 [120/2] via 155.1.0.3, 00:00:07, Tunnel0
R     155.1.108.0/24 [120/1] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
R     155.1.146.0/24 [120/1] via 155.1.0.1, 00:00:12, Tunnel0
```

Pitfall

Whitespace counts as a valid character for key chain authentication. Use the **`show key chain`** command to ensure that whitespace is not appended at the end of the authentication string.

The most common error leading to whitespaces being appended is using the **`?`** after typing the password string, which results in authentication errors and updates being rejected.

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R2(config)#key chain RIP
R2(config-keychain)#key 1
R2(config-keychain-key)#key-string CISCO ?
LINE    <cr>
!R2(config-keychain-key)#key-string CISCO
!R2#show key chain
Key-chain RIP: key 1 -- text "CISCO "
        accept lifetime (always valid) - (always valid) [valid now]
        send lifetime (always valid) - (always valid) [valid now]
R2#debug ip rip
```

https://t.me/learningnets

```
RIP protocol debugging is on
!RIP: received packet with MD5 authentication
RIP: ignored v2 packet from 155.1.0.5 (invalid authentication)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Split Horizon

> You must load the initial configuration files for the section, **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Enable split-horizon on R5's connection to the DMVPN cloud.
- Test IPv4 reachability to all networks and note any changes within the topology.

Configuration

```
R5:

interface Tunnel0
 ip split-horizon
```

Verification

https://t.me/learningnets

The split-horizon rule applies to distance-vector protocols such as RIP and EIGRP. It can be enabled or disabled per-interface, and it is enabled by default on all Ethernet and Tunnel interfaces for which RIP or EIGRP is enabled. When enabled, it does not allow the router to send/reflect routing updates back on the interface it was received on if it is selected as best route and installed in the routing table. By default, split-horizon is enabled on the Tunnel interface, but in this case it has been disabled from the initial configurations. Note the routing table of R1, for example, before split-horizon is enabled:

```
R1#show ip route rip | i Tunnel
R        150.1.2.2 [120/2] via 155.1.0.2, 00:00:13, Tunnel0
R        150.1.4.4 [120/2] via 155.1.0.4, 00:00:13, Tunnel0

R        150.1.5.5 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
R        150.1.8.8 [120/2] via 155.1.0.5, 00:00:13, Tunnel0
R        150.1.10.10 [120/3] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.5.0/24 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.8.0/24 [120/2] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.10.0/24 [120/3] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.45.0/24 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.58.0/24 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.108.0/24 [120/2] via 155.1.0.5, 00:00:13, Tunnel0
```

Verify that R1 has connectivity with R2's and R4's Loopback0 prefixes, and that these are also present in the RIP database.

```
R1#show ip rip database 150.1.2.2 255.255.255.255
150.1.2.2/32 [2] via 155.1.0.2, from 155.1.0.5, 00:00:06, Tunnel0
!R1#ping 150.1.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds: !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
!R1#show ip rip  database 150.1.4.4 255.255.255.255
150.1.4.4/32 [2] via 155.1.0.4, from 155.1.0.5, 00:00:21, Tunnel0
!R1#ping 150.1.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds: !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Enable split-horizon on R5's DMVPN connection and note the differences in R1's table. Because R5 no longer reflects routes received on its Tunnel0 interface, those

https://t.me/learningnets

entries will slowly be removed from the routing table of other routers based on the flush timer, which has a default value of 240 seconds. You can see this behavior in the routing table; certain routes have been installed in the routing table for longer than the default RIP update timer of 30 seconds, which refreshes the routing table entries as well.

```
R1#show ip route rip | i Tunnel
R        150.1.2.2 [120/2] via 155.1.0.2, 00:01:20
, Tunnel0 R        150.1.4.4 [120/2] via 155.1.0.4, 00:01:20
, Tunnel0
R        150.1.5.5 [120/1] via 155.1.0.5, 00:00:23, Tunnel0
R        150.1.8.8 [120/2] via 155.1.0.5, 00:00:23, Tunnel0
R        150.1.10.10 [120/3] via 155.1.0.5, 00:00:23, Tunnel0
R        155.1.5.0/24 [120/1] via 155.1.0.5, 00:00:23, Tunnel0
R        155.1.8.0/24 [120/2] via 155.1.0.5, 00:00:23, Tunnel0
R        155.1.10.0/24 [120/3] via 155.1.0.5, 00:00:23, Tunnel0
R        155.1.45.0/24 [120/1] via 155.1.0.5, 00:00:23, Tunnel0
R        155.1.58.0/24 [120/1] via 155.1.0.5, 00:00:23, Tunnel0
R        155.1.108.0/24 [120/2] via 155.1.0.5, 00:00:23, Tunnel0
```

Verify that after 3 minutes R1 no longer has routes for R2's and R4's Loopback0 installed in the routing table; these are also not available in the RIP database.

```
R1#show ip route rip | i Tunnel
R        150.1.5.5 [120/1] via 155.1.0.5, 00:00:14, Tunnel0
R        150.1.8.8 [120/2] via 155.1.0.5, 00:00:14, Tunnel0
R        150.1.10.10 [120/3] via 155.1.0.5, 00:00:14, Tunnel0
R        155.1.5.0/24 [120/1] via 155.1.0.5, 00:00:14, Tunnel0
R        155.1.8.0/24 [120/2] via 155.1.0.5, 00:00:14, Tunnel0
R        155.1.10.0/24 [120/3] via 155.1.0.5, 00:00:14, Tunnel0
R        155.1.45.0/24 [120/1] via 155.1.0.5, 00:00:14, Tunnel0
R        155.1.58.0/24 [120/1] via 155.1.0.5, 00:00:14, Tunnel0
R        155.1.108.0/24 [120/2] via 155.1.0.5, 00:00:14, Tunnel0
!R1#show ip rip database 150.1.2.2 255.255.255.255
%Route not in database
!R1#show ip rip database 150.1.4.4 255.255.255.255
%Route not in database
```

Verify that split-horizon is enabled on R5's DMVPN interface and that R1 has no reachability with R2's and R4's Loopback0.

```
R5#show ip interface tunnel0 | i Split
```

https://t.me/learningnets

```
Split horizon is enabled

!R1#ping 150.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:.....
Success rate is 0 percent (0/5)
!R1#ping 150.1.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:.....

Success rate is 0 percent (0/5)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Auto-Summary

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure RIPv2 for auto summarization on R4.
- Note any changes in the network advertisements that R4 is sending.

Configuration

```
R4:

router rip
 auto-summary
```

Verification

Before auto-summary is enabled:

```
R4#debug ip rip
RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.45 (155.1.45.4)
RIP: build update entries 150.1.4.4/32 via 0.0.0.0, metric 1, tag 0
    155.1.0.0/24 via 0.0.0.0, metric 1, tag 0
RIP: sending v2 update to 224.0.0.9 via Tunnel0 (155.1.0.4)
```

https://t.me/learningnets

```
RIP: build update entries 150.1.4.4/32 via 0.0.0.0, metric 1, tag 0

    155.1.45.0/24 via 0.0.0.0, metric 1, tag 0
```

After auto-summary is enabled, R4 summarizes the specific 150.1.4.4/32 prefix of its Loopback0 into the 150.1.0.0/16 and sends it outbound on its RIP enabled interfaces. Since R4's Ethernet interface on VLAN 146 is disabled, there are no RIP updates sent or received on the interface:

```
RIP: sending v2 update to 224.0.0.9 via Tunnel0 (155.1.0.4)
RIP: build update entries 150.1.0.0/16 via 0.0.0.0, metric 1, tag 0
    155.1.45.0/24 via 0.0.0.0, metric 1, tag 0
RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.45 (155.1.45.4)
RIP: build update entries 150.1.0.0/16 via 0.0.0.0, metric 1, tag 0

    155.1.0.0/24 via 0.0.0.0, metric 1, tag 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Send and Receive Versions

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Remove the `version 2` commands under the RIP processes of R8 and R10.
- Configure R8 to send and receive only RIPv2 updates on VLAN 58.
- Configure R8 to send and receive only RIPv1 updates on VLAN 108.
- Note any changes in IPv4 reachability throughout the network topology.

> Although RIPv1 will not be tested on in the CCIE Lab Exam, understanding the problems with legacy protocol design can help you better understand the routing logic for IPv4.

Configuration

```
R8:
interface GigabitEthernet1.58
 ip rip send version 2
 ip rip receive version 2
!
interface GigabitEthernet1.108
 ip rip send version 1
 ip rip receive version 1
!
```

https://t.me/learningnets

```
router rip
 no version 2

R10:

router rip
 no version 2
```

Verification

R8 and R10's routing tables before the change to RIPv1:

```
R8#show ip route rip
     150.1.0.0/32 is subnetted, 9 subnets
R        150.1.1.1 [120/3] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.2.2 [120/2] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        150.1.3.3 [120/2] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        150.1.4.4 [120/2] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        150.1.5.5 [120/1] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        150.1.6.6 [120/4] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        150.1.7.7 [120/3] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        150.1.9.9 [120/4] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        150.1.10.10 [120/1] via 155.1.108.10, 00:00:02, GigabitEthernet1.108
     155.1.0.0/16 is variably subnetted, 17 subnets, 2 masks
R        155.1.0.0/24 [120/1] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.5.0/24 [120/1] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.7.0/24 [120/3] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.9.0/24 [120/4] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.10.0/24
          [120/1] via 155.1.108.10, 00:00:02, GigabitEthernet1.108
R        155.1.13.0/24 [120/2] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.37.0/24 [120/2] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.45.0/24 [120/1] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.67.0/24 [120/3] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.79.0/24 [120/3] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
R        155.1.146.0/24 [120/4] via 155.1.58.5, 00:00:23, GigabitEthernet1.58
!R10#show ip route rip

     150.1.0.0/32 is subnetted, 10 subnets
R        150.1.1.1 [120/4] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        150.1.2.2 [120/3] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        150.1.3.3 [120/3] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        150.1.4.4 [120/3] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        150.1.5.5 [120/2] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        150.1.6.6 [120/5] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
```

https://t.me/learningnets

```
R        150.1.7.7 [120/4] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        150.1.8.8 [120/1] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        150.1.9.9 [120/5] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
     155.1.0.0/16 is variably subnetted, 16 subnets, 2 masks
R        155.1.0.0/24 [120/2] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.5.0/24 [120/2] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.7.0/24 [120/4] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.8.0/24 [120/1] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.9.0/24 [120/5] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.13.0/24 [120/3] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.37.0/24 [120/3] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.45.0/24 [120/2] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.58.0/24 [120/1] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.67.0/24 [120/4] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.79.0/24 [120/4] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
R        155.1.146.0/24
          [120/4] via 155.1.108.8, 00:00:20, GigabitEthernet1.108
```

After RIPv1 has been enabled between R8 and R10:

```
R8#show ip route rip
    150.1.0.0/16 is variably subnetted, 10 subnets, 2 masks
R        150.1.0.0/16 [120/1] via 155.1.108.10, 00:00:06, GigabitEthernet1.108
R        150.1.1.1/32 [120/2] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.2.2/32 [120/2] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.3.3/32 [120/2] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.4.4/32 [120/2] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.5.5/32 [120/1] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.6.6/32 [120/3] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.7.7/32 [120/3] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        150.1.9.9/32 [120/4] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
   155.1.0.0/16 is variably subnetted, 17 subnets, 2 masks
R        155.1.0.0/24 [120/1] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.5.0/24 [120/1] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.7.0/24 [120/3] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.9.0/24 [120/4] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.10.0/24
           [120/1] via 155.1.108.10, 00:00:01, GigabitEthernet1.108
R        155.1.13.0/24 [120/2] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.37.0/24 [120/2] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.45.0/24 [120/1] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.67.0/24 [120/3] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.79.0/24 [120/3] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
R        155.1.146.0/24 [120/2] via 155.1.58.5, 00:00:04, GigabitEthernet1.58
!R10#show ip route rip
```

https://t.me/learningnets

```
     150.1.0.0/16 is variably subnetted, 2 subnets, 2 masks
R        150.1.0.0/16 [120/1] via 155.1.108.8, 00:01:33, GigabitEthernet1.108

     155.1.0.0/16 is variably subnetted, 16 subnets, 2 masks
R        155.1.0.0/24 [120/2] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.5.0/24 [120/2] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.7.0/24 [120/4] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.8.0/24 [120/1] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.9.0/24 [120/5] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.13.0/24 [120/3] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.37.0/24 [120/3] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.45.0/24 [120/2] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.58.0/24 [120/1] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.67.0/24 [120/4] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.79.0/24 [120/4] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
R        155.1.146.0/24 [120/3] via 155.1.108.8, 00:00:12, GigabitEthernet1.108
```

Because R8 and R10 are running RIPv1 on the directly connected Ethernet link, which does not include subnet mask information in the update, only classful networks are advertised, along with contiguous subnets that share the same major network and subnet mask as the transit link between the RIP routers. This is why R10 still learns about all the 155.1.X.0/24 networks, but cannot learn any of the 150.1.X.0/24 Loopback networks. Likewise, R8 cannot learn the 150.1.10.0/24 network from R10. When traffic is sent to 150.1.10.10, it is routed based on the RIPv1 generated update of 150.1.0.0/16 by R10, which is classful:

```
R8#show ip route 150.1.10.10
Routing entry for 150.1.0.0/16
  Known via "rip", distance 120, metric 1
  Redistributing via rip
  Last update from 155.1.108.10 on GigabitEthernet1.108, 00:00:04 ago
  Routing Descriptor Blocks: * 155.1.108.10, from 155.1.108.10, 00:00:04 ago, via GigabitEthernet1.108
      Route metric is 1, traffic share count is 1
!R8#ping 150.1.10.10 source Loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.8.8 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 2/11/33 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Manual Summarization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R5 to generate a IPv4 RIP summary route for Loopback0 prefixes of R1 to R7 routers, send the update only to R8.
- Ensure that the summary does not overlap with any IPv4 address space that R5 does not have a longer match for.

Configuration

```
R5:

interface GigabitEthernet1.58
 ip summary-address rip 150.1.0.0 255.255.248.0
```

Verification

The configured summary route on R5 should be received by R8 and installed in the routing table with a metric of 1 hop. Based on the output of `show ip route 150.1.1.1` or any other Loopback0 prefixes of routers R1 to R7, the summary prefix 150.1.0.0/21 is being referenced as the best route:

https://t.me/learningnets

```
R8#show ip route 150.1.1.1
 Routing entry for 150.1.0.0/21
   Known via "rip", distance 120, metric 1
   Redistributing via rip
   Last update from 155.1.58.5 on GigabitEthernet1.58, 00:00:23 ago
   Routing Descriptor Blocks: * 155.1.58.5, from 155.1.58.5, 00:00:23 ago, via GigabitEthernet1.58

   Route metric is 1, traffic share count is 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Convergence Timers

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure RIP timers throughout the topology to be three times lower than default values.
- Ensure R7 and R8 still use the default update interval on their Ethernet links to R9 and R10.

Configuration

https://t.me/learningnets

```
R1 - R10:

router rip
 timers basic 10 60 60 80

R7:

interface GigabitEthernet1.79
 ip rip advertise 30

R8:

interface GigabitEthernet1.108
 ip rip advertise 30
```

Verification

Verify default RIP timers before changing it:

```
R7#show ip protocols | include seconds
 Sending updates every 30 seconds
, next due in 24 seconds Invalid after 180 seconds, hold down 180, flushed after 240
```

Verify RIP timers after changing it:

```
R7#show ip protocols | include seconds
     Sending updates every 10 seconds
, next due in 19 seconds Invalid after 60 seconds, hold down 60, flushed after 80
```

All routers have been globally configured for a 10 seconds update interval, however it can be overriden at the interface level if required. R7 and R8 are configured to send routing updates outbound on VLAN 79 and VLAN 108 segments every 30 seconds.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Offset List

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure an offset-list on R1 so that all traffic destined to R3's Loopback0 is routed over VLAN 146 segment.
- If the VLAN 146 Ethernet link is down, traffic should be rerouted over the directly connected Ethernet link to R3.

Configuration

```
R1:

access-list 1 permit host 150.1.3.3
!
router rip
 offset-list 1 in 3 GigabitEthernet1.13
 offset-list 1 in 3 Tunnel0
```

Verification

Verify traffic path before metric offset is applied:

https://t.me/learningnets

```
R1#traceroute 150.1.3.3

Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.13.3 3 msec *  2 msec
```

Verify traffic path after metric offset is applied:

```
R1#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.6 3 msec 1 msec 1 msec
 2 155.1.67.7 1 msec 2 msec 1 msec
 3 155.1.37.3 1 msec *  1 msec
```

Verify route metric after metric offset is applied:

```
R1#show ip route 150.1.3.3
    Routing entry for 150.1.3.3/32    Known via "rip", distance 120, metric 3
    Redistributing via rip
    Last update from 155.1.146.6 on GigabitEthernet1.146, 00:00:09 ago
    Routing Descriptor Blocks: * 155.1.146.6, from 155.1.146.6, 00:00:09 ago, via GigabitEthernet1.146

              Route metric is 3, traffic share count is 1
```

Verify that traffic is re-routed over the directly connected Ethernet link between R1 and R3, if R1's Ethernet VLAN 146 interface is down:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface GigabitEthernet1.146
R1(config-if)#shutdown
!
!R1#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.13.3 3 msec *  2 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Filtering with Passive Interface

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure passive interface on R8 so that it learns RIP updates from R10 but does not advertise any routing information to R10.

Configuration

```
R8:

router rip
 passive-interface GigabitEthernet1.108
```

Verification

Once VLAN 108 Ethernet interface is configured as passive, R8 stops sending routing updates outbound on that particular segment. However, passive-interface in RIP does not affect the inbound updates on the segment:

```
R10#debug ip rip
RIP protocol debugging is on
RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.108 (155.1.108.10)
```

https://t.me/learningnets

```
RIP: build update entries
        150.1.10.10/32 via 0.0.0.0, metric 1, tag 0
        155.1.10.0/24 via 0.0.0.0, metric 1, tag 0
!        R8#debug ip rip
RIP protocol debugging is on
RIP: received v2 update from 155.1.108.10 on GigabitEthernet1.108


        150.1.10.10/32 via 0.0.0.0 in 1 hops
        155.1.10.0/24 via 0.0.0.0 in 1 hops
```

Verify that R8 still receives and accepts RIP updates from R10:

```
R8#sh ip route | inc GigabitEthernet1.108
R        150.1.10.10 [120/1] via 155.1.108.10, 00:01:03, GigabitEthernet1.108


             [120/1] via 155.1.108.10, 00:01:03, GigabitEthernet1.108
C        155.1.108.0/24 is directly connected, GigabitEthernet1.108
L        155.1.108.8/32 is directly connected, GigabitEthernet1.108
```

Verify that R10 does not receive any RIP updates from R8:

```
R10#show ip route


Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set


     150.1.0.0/32 is subnetted, 1 subnets
C        150.1.10.10 is directly connected, Loopback0
     155.1.0.0/16 is variably subnetted, 4 subnets, 2 masks
C        155.1.10.0/24 is directly connected, GigabitEthernet1.10
L        155.1.10.10/32 is directly connected, GigabitEthernet1.10
C        155.1.108.0/24 is directly connected, GigabitEthernet1.108
L        155.1.108.10/32 is directly connected, GigabitEthernet1.108
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Filtering with Prefix-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Using a prefix-list, configure R5 to stop advertising IPv4 Loopback0 prefixes of R6 and R7 to R8.
  - Ensure that all other prefixes are still advertised.

- Using a prefix-list, configure R5 to filter any IPv4 updates received inbound from R4 over the DMVPN cloud.
  - Ensure that inbound updates from other RIP routers on the DMVPN cloud are accepted.

Configuration

R5:

```
ip prefix-list NOT_FROM_R4 seq 5 deny 155.1.0.4/32
ip prefix-list NOT_FROM_R4 seq 10 permit 0.0.0.0/0 le 32
!
ip prefix-list PERMIT_ALL seq 5 permit 0.0.0.0/0 le 32
!
ip prefix-list RIP_FILTER_TO_R8 seq 5 deny 150.1.6.6/32
ip prefix-list RIP_FILTER_TO_R8 seq 10 deny 150.1.7.7/32
ip prefix-list RIP_FILTER_TO_R8 seq 15 permit 0.0.0.0/0 le 32
```

https://t.me/learningnets

```
!
router rip
 distribute-list prefix RIP_FILTER_TO_R8 out GigabitEthernet1.58
 distribute-list prefix PERMIT_ALL gateway NOT_FROM_R4 in
```

The prefix-list named **RIP_FILTER_TO_R8** filters R6 and R7 Loopback0 prefixes from being advertises out on VLAN 58 and permits all others. The syntax **`0.0.0.0/0 le 32`** in a prefix-list means match all routes, similar to the **`any`** keyword from access-lists. The second route filtering is based on both the routes being learned and whom they are learned from. This filter says match any route coming in any interface, per the **PERMIT_ALL** prefix-list, and allow them to come in as long as they were not learned from R4, per the **`deny 155.1.0.4/32`** syntax.

Verification

Verify that R5 learns through RIP about R6 and R7 Loopback0 prefixes.

```
R5#show ip route rip
 150.1.0.0/32 is subnetted, 10 subnets
R        150.1.1.1 [120/1] via 155.1.0.1, 00:00:11, Tunnel0
R        150.1.2.2 [120/1] via 155.1.0.2, 00:00:18, Tunnel0
R        150.1.3.3 [120/1] via 155.1.0.3, 00:00:18, Tunnel0
R        150.1.4.4 [120/1] via 155.1.45.4, 00:00:13, GigabitEthernet1.45
                 [120/1] via 155.1.0.4, 00:02:43, Tunnel0
R        150.1.6.6 [120/2] via 155.1.0.1, 00:00:11, Tunnel0
R        150.1.7.7 [120/2] via 155.1.0.3, 00:00:18, Tunnel0

R        150.1.8.8 [120/1] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
R        150.1.9.9 [120/3] via 155.1.0.3, 00:00:18, Tunnel0
R        150.1.10.10 [120/2] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
      155.1.0.0/16 is variably subnetted, 18 subnets, 2 masks
R        155.1.7.0/24 [120/2] via 155.1.0.3, 00:00:18, Tunnel0
R        155.1.8.0/24 [120/1] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
R        155.1.9.0/24 [120/3] via 155.1.0.3, 00:00:18, Tunnel0
R        155.1.10.0/24 [120/2] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
R        155.1.13.0/24 [120/1] via 155.1.0.3, 00:00:18, Tunnel0
                       [120/1] via 155.1.0.1, 00:00:11, Tunnel0
R        155.1.37.0/24 [120/1] via 155.1.0.3, 00:00:18, Tunnel0
R        155.1.67.0/24 [120/2] via 155.1.0.3, 00:00:18, Tunnel0
                       [120/2] via 155.1.0.1, 00:00:11, Tunnel0
R        155.1.79.0/24 [120/2] via 155.1.0.3, 00:00:18, Tunnel0
R        155.1.108.0/24 [120/1] via 155.1.58.8, 00:00:25, GigabitEthernet1.58
```

https://t.me/learningnets

```
R        155.1.146.0/24 [120/1] via 155.1.0.1, 00:00:11, Tunnel0
```

Verify that R8 does not learn through RIP about the same Loopback0 prefixes, but other routers do as R5 advertises it out.

```
R8#show ip route 150.1.6.6
% Subnet not in table
!R8#show ip route 150.1.7.7
% Subnet not in table
!R2#show ip route 150.1.6.6
Routing entry for 150.1.6.6/32
    Known via "rip", distance 120, metric 3
    Redistributing via rip
    Last update from 155.1.0.1 on Tunnel0, 00:00:09 ago
    Routing Descriptor Blocks: * 155.1.0.1, from 155.1.0.5, 00:00:09 ago, via Tunnel0
        Route metric is 3, traffic share count is 1
!R2#show ip route 150.1.7.7
Routing entry for 150.1.7.7/32
    Known via "rip", distance 120, metric 3
    Redistributing via rip
    Last update from 155.1.0.3 on Tunnel0, 00:00:06 ago
    Routing Descriptor Blocks: * 155.1.0.3, from 155.1.0.5, 00:00:06 ago, via Tunnel0

        Route metric is 3, traffic share count is 1
```

Verify IPv4 reachability from R2 with the respective Loopback0 prefixes. Because the Ethernet link between R2 and R3 has been disabled from the initial configuration, traffic will be routed over the DMVPN cloud.

```
R2#ping 150.1.6.6
    Type escape sequence to abort.
    Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds: !!!!!
    Success rate is 100 percent (5/5), round-trip min/avg/max = 5/12/31 ms
!   R2#ping 150.1.7.7
    Type escape sequence to abort.
    Sending 5, 100-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds: !!!!!

    Success rate is 100 percent (5/5), round-trip min/avg/max = 6/12/26 ms
```

Verify through debugging on R5 that Loopback0 prefixes of R6 and R7 are still advertised out on the DMVPN network, so filtering only affects R5's connection to R8.

https://t.me/learningnets

```
R5#debug ip rip
RIP: sending v2 update to 224.0.0.9 via Tunnel0 (155.1.0.5)

RIP: build update entries
    150.1.1.1/32 via 155.1.0.1, metric 2, tag 0
    150.1.2.2/32 via 155.1.0.2, metric 2, tag 0
    150.1.3.3/32 via 155.1.0.3, metric 2, tag 0
    150.1.4.4/32 via 0.0.0.0, metric 2, tag 0
    150.1.5.5/32 via 0.0.0.0, metric 1, tag 0 150.1.6.6/32 via 155.1.0.1, metric 3, tag 0
 150.1.7.7/32 via 155.1.0.3, metric 3, tag 0
    150.1.8.8/32 via 0.0.0.0, metric 2, tag 0
    150.1.9.9/32 via 155.1.0.3, metric 4, tag 0
    150.1.10.10/32 via 0.0.0.0, metric 3, tag 0
!RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.58 (155.1.58.5)

RIP: build update entries
    150.1.1.1/32 via 0.0.0.0, metric 2, tag 0
    150.1.2.2/32 via 0.0.0.0, metric 2, tag 0
    150.1.3.3/32 via 0.0.0.0, metric 2, tag 0
    150.1.4.4/32 via 0.0.0.0, metric 2, tag 0
    150.1.5.5/32 via 0.0.0.0, metric 1, tag 0
    150.1.9.9/32 via 0.0.0.0, metric 4, tag 0
    155.1.0.0/24 via 0.0.0.0, metric 1, tag 0
    155.1.5.0/24 via 0.0.0.0, metric 1, tag 0
    155.1.7.0/24 via 0.0.0.0, metric 3, tag 0
    155.1.9.0/24 via 0.0.0.0, metric 4, tag 0
    155.1.13.0/24 via 0.0.0.0, metric 2, tag 0
    155.1.37.0/24 via 0.0.0.0, metric 2, tag 0
    155.1.45.0/24 via 0.0.0.0, metric 1, tag 0
    155.1.67.0/24 via 0.0.0.0, metric 3, tag 0
    155.1.79.0/24 via 0.0.0.0, metric 3, tag 0
    155.1.146.0/24 via 0.0.0.0, metric 2, tag 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Filtering with Standard Access-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure a one-line standard access-list on R6 to filter out the IPv4 prefixes that have an even number in the third octet.

Configuration

```
R6:

access-list 1 permit 0.0.1.0 255.255.254.255
!
router rip
 distribute-list 1 in
```

Verification

Verify R6's routing table; there are no routes that have an even number in the third octet.

```
R6#show ip route rip
```

https://t.me/learningnets

```
    150.1.0.0/32 is subnetted, 5 subnets
R        150.1.3.3 [120/2] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        150.1.5.5 [120/3] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        150.1.7.7 [120/1] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        150.1.9.9 [120/2] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
    155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks
R        155.1.5.0/24 [120/3] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        155.1.7.0/24 [120/1] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        155.1.9.0/24 [120/2] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        155.1.13.0/24 [120/2] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        155.1.37.0/24 [120/1] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        155.1.45.0/24 [120/3] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
R        155.1.79.0/24 [120/1] via 155.1.67.7, 00:00:04, GigabitEthernet1.67
```

When debugging RIP, we can see that all routes are received from R7, but only odd-numbered routes are getting installed in the routing table.

```
R6#debug ip rip
RIP protocol debugging is on
RIP: received v2 update from 155.1.67.7 on GigabitEthernet1.67
150.1.2.2/32 via 0.0.0.0 in 4 hops
 150.1.3.3/32 via 0.0.0.0 in 2 hops 150.1.4.4/32 via 0.0.0.0 in 4 hops

 150.1.5.5/32 via 0.0.0.0 in 3 hops
 150.1.7.7/32 via 0.0.0.0 in 1 hops
 150.1.8.8/32 via 0.0.0.0 in 4 hops
 150.1.9.9/32 via 0.0.0.0 in 2 hops
 150.1.10.10/32 via 0.0.0.0 in 5 hops
 155.1.0.0/24 via 0.0.0.0 in 2 hops
 155.1.5.0/24 via 0.0.0.0 in 3 hops
 155.1.7.0/24 via 0.0.0.0 in 1 hops
 155.1.8.0/24 via 0.0.0.0 in 4 hops
 155.1.9.0/24 via 0.0.0.0 in 2 hops
 155.1.10.0/24 via 0.0.0.0 in 5 hops
 155.1.13.0/24 via 0.0.0.0 in 2 hops
 155.1.37.0/24 via 0.0.0.0 in 1 hops
 155.1.45.0/24 via 0.0.0.0 in 3 hops
 155.1.58.0/24 via 0.0.0.0 in 3 hops
 155.1.79.0/24 via 0.0.0.0 in 1 hops
 155.1.108.0/24 via 0.0.0.0 in 4 hops
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Filtering with Extended Access-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure an extended access-list filter on R5 so that routes for VLAN 7 and VLAN 9 are accepted only from R1:
  - additionally, routes to R1's Loopback0 and VLAN 146 are accepted only from R3.
  - this filter should not affect any other updates on this segment.

Configuration

```
R5:

access-list 100 deny ip host 155.1.0.3 host 155.1.7.0
access-list 100 deny ip host 155.1.0.3 host 155.1.9.0
access-list 100 deny ip host 155.1.0.1 host 155.1.146.0
access-list 100 deny ip host 155.1.0.1 host 150.1.1.1
access-list 100 permit ip any any
!
router rip
 distribute-list 100 in tunnel0
```

https://t.me/learningnets

Verification

When extended access-lists are used as distribute-list for IGP filtering, the functionality is different than when used for route redistribution or in BGP. With BGP and redistribution, the source field in the ACL represents the network address, and the destination field represents the subnet mask. In IGP distribute-list application, the source field in the ACL matches the update source of the route, and the destination field represents the network address. This implementation allows us to control which routes we accept, but more importantly who do we accept it from. Before the filter is applied, R5 routes to R3 for VLANs 7 and 9, and to R1 for VLAN 146 and R1's Loopback0:

```
R5#show ip route rip | include via 155.1.0.(1|3)
R        150.1.1.1 [120/1] via 155.1.0.1, 00:00:22, Tunnel0
R        150.1.3.3 [120/1] via 155.1.0.3, 00:00:27, Tunnel0
R        150.1.6.6 [120/2] via 155.1.0.1, 00:00:22, Tunnel0
R        150.1.7.7 [120/2] via 155.1.0.3, 00:00:27, Tunnel0
R        150.1.9.9 [120/3] via 155.1.0.3, 00:00:27, Tunnel0
R        155.1.7.0/24 [120/2] via 155.1.0.3, 00:00:27, Tunnel0
R        155.1.9.0/24 [120/3] via 155.1.0.3, 00:00:27, Tunnel0
R        155.1.13.0/24 [120/1] via 155.1.0.3, 00:00:27, Tunnel0
                       [120/1] via 155.1.0.1, 00:00:22, Tunnel0
R        155.1.37.0/24 [120/1] via 155.1.0.3, 00:00:27, Tunnel0
R        155.1.67.0/24 [120/2] via 155.1.0.3, 00:00:27, Tunnel0
                        [120/2] via 155.1.0.1, 00:00:22, Tunnel0
R        155.1.79.0/24 [120/2] via 155.1.0.3, 00:00:27, Tunnel0
R        155.1.146.0/24 [120/1] via 155.1.0.1, 00:00:22, Tunnel0
```

After the filter is applied, R5 routes to R1 for VLANs 7 and 9, and to R3 for VLAN 146 and R1's Loopback0:

```
R5#show ip route rip | include via 155.1.0.(1|3)
R        150.1.1.1 [120/2] via 155.1.0.3, 00:00:20, Tunnel0
R        150.1.3.3 [120/1] via 155.1.0.3, 00:00:20, Tunnel0
R        150.1.6.6 [120/2] via 155.1.0.1, 00:00:10, Tunnel0
R        150.1.7.7 [120/2] via 155.1.0.3, 00:00:20, Tunnel0
R        150.1.9.9 [120/3] via 155.1.0.3, 00:00:20, Tunnel0
R        155.1.7.0/24 [120/3] via 155.1.0.1, 00:00:10, Tunnel0
R        155.1.9.0/24 [120/4] via 155.1.0.1, 00:00:10, Tunnel0
R        155.1.13.0/24 [120/1] via 155.1.0.3, 00:00:20, Tunnel0
                       [120/1] via 155.1.0.1, 00:00:10, Tunnel0
R        155.1.37.0/24 [120/1] via 155.1.0.3, 00:00:20, Tunnel0
```

https://t.me/learningnets

```
R        155.1.67.0/24 [120/2] via 155.1.0.3, 00:00:20, Tunnel0
                       [120/2] via 155.1.0.1, 00:00:10, Tunnel0
R        155.1.79.0/24 [120/2] via 155.1.0.3, 00:00:20, Tunnel0
R        155.1.146.0/24 [120/2] via 155.1.0.3, 00:00:20, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Filtering with Offset Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure an offset-list on R7 so that R9 does not install a route for VLAN 5.
- This filter should not affect any other updates on this segment.

Configuration

```
R7:

access-list 1 permit 155.1.5.0
!
router rip
 offset-list 1 out 16 GigabitEthernet1.79
```

Verification

Before offset-list is applied, R9 has a route for VLAN 5:

```
R9#show ip route 155.1.5.0
Routing entry for 155.1.5.0/24
Known via "rip", distance 120, metric 3
```

https://t.me/learningnets

```
    Redistributing via rip
    Last update from 155.1.79.7 on GigabitEthernet1.79, 00:00:23 ago
    Routing Descriptor Blocks: * 155.1.79.7, from 155.1.79.7, 00:00:23 ago, via GigabitEthernet1
```

After offset-list is applied, R9 no longer has the route installed because it receives it from R7 with an infinite metric, infinite metric having the value of 16:

```
R9#show ip route 155.1.5.0
% Subnet not in table
!R9#debug ip rip
RIP protocol debugging is on
RIP: recei
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Filtering with Administrative Distance

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure administrative distance filtering on R5 so that devices within the network cannot reach R4's Loopback0 network.
- This filter should not affect any other networks in the topology.

Configuration

```
R5:

access-list 1 permit host 150.1.4.4
!
router rip
 distance 255 0.0.0.0 255.255.255.255 1
```

Verification

Before administrative distance filtering is applied, R5 has the route for R4's Loopback0 installed.

```
R5#show ip route 150.1.4.0
```

https://t.me/learningnets

```
    Routing entry for 150.1.4.4/32 Known via "rip", distance 120, metric 1
    Redistributing via rip
    Last update from 155.1.45.4 on GigabitEthernet1.45, 00:00:24 ago
    Routing Descriptor Blocks: * 155.1.45.4, from 155.1.45.4, 00:00:24 ago, via GigabitEthernet1.45
        Route metric is 1, traffic share count is 1 155.1.0.4, from 155.1.0.4, 00:00:25 ago, via Tunnel0
        Route metric is 1, traffic share count is 1
!R2#ping 150.1.4.4
    Type escape sequence to abort.
    Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds: !!!!!

    Success rate is 100 percent (5/5), round-trip min/avg/max = 2/6/15 ms
```

Even though administrative distance is locally significant to the router, the RIP process, like the EIGRP process, cannot advertise a route that is not actually installed in the routing table. By setting the distance of the route 150.1.4.4/32 to 255, it is invalidated from being installed in the routing table, and hence invalidated from being advertised to any neighbors. After configuration is applied, R2 not only does not have reachability to the destination, it does not have the route in its routing table or RIP database.

```
R2#show ip rip database 150.1.4.4 255.255.255.255
% Route not in database
!R2#show ip route 150.1.4.4
        % Subnet not in table
!R2#ping 150.1.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:.....
Success rate is 0 percent (0/5)
!R5#show ip route 150.1.4.4
% Subnet not in table
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Filtering with Per-Neighbor AD

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure administrative distance filtering on R7 so that traffic destined for R3's Loopback0 prefix is routed through R6.
- This configuration should not affect any other networks in the topology.

Configuration

```
R7:

access-list 2 permit 150.1.3.3
!
router rip
 distance 255 155.1.37.3 0.0.0.0 2
```

Verification

Before administrative distance filtering is applied, traffic is routed through R3.

```
R7#show ip route 150.1.3.3
    Routing entry for 150.1.3.3/32
    Known via "rip", distance 120, metric 1
```

https://t.me/learningnets

```
    Redistributing via rip
    Last update from 155.1.37.3 on GigabitEthernet1.37, 00:00:03 ago
    Routing Descriptor Blocks: * 155.1.37.3, from 155.1.37.3, 00:00:03 ago, via GigabitEthernet1.37


        Route metric is 1, traffic share count is 1
```

The **distance** command can be used to change the administrative distance globally for the routing process, globally for the routing process per route type (external vs. internal), on a per-prefix basis, or on a per-neighbor per-prefix basis. The two fields after the distance value are the source of the route (RIP speaker) and a wildcard mask to match the source. The value needed for the source is seen in the above output in the **from 155.1.37.3** field. After filtering is applied, traffic is routed through R6.

```
R7#show ip route 150.1.3.3
    Routing entry for 150.1.3.3/32
    Known via "rip", distance 120, metric 3
    Redistributing via rip
    Last update from 155.1.67.6 on GigabitEthernet1.67, 00:00:01 ago
    Routing Descriptor Blocks: * 155.1.67.6, from 155.1.67.6, 00:00:01 ago, via GigabitEthernet1.67
        Route metric is 3, traffic share count is 1
!             R7#traceroute 150.1.3.3
    Type escape sequence to abort.
    Tracing the route to 150.1.3.3
    VRF info: (vrf in name/id, vrf out name/id) 1 155.1.67.6 11 msec 5 msec 7 msec
    2 155.1.146.1 7 msec 6 msec 7 msec 3 155.1.13.3 14 msec *  7 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R6 to advertise a default route via RIP only outbound on its VLAN146 interface.
- Do not use any access-lists or prefix-lists on R6 to accomplish this.

Configuration

https://t.me/learningnets

```
R6:

route-map DEFAULT_TO_R1 permit 10
 set interface GigabitEthernet1.146
!
router rip
 default-information originate route-map DEFAULT_TO_R1

R7:

ip prefix-list NO_DEFAULT seq 5 deny 0.0.0.0/0
ip prefix-list NO_DEFAULT seq 10 permit 0.0.0.0/0 le 32
!
router rip
 distribute-list prefix NO_DEFAULT out GigabitEthernet1.67
```

Verification

Verify that R6 advertised the default route, as it is present in the RIP database, but not in the routing table.

```
R6#show ip rip database 0.0.0.0 0.0.0.0
0.0.0.0/0    redistributed
    [1] via 0.0.0.0,
!R6#show ip route | include ( 0.0.0.0)
R6#
```

Verify that other routers in the topology receive the default route, and therefore R6 advertises it.

https://t.me/learningnets

```
R1#show ip route | include ( 0.0.0.0)

Gateway of last resort is 155.1.146.6 to network 0.0.0.0
R*    0.0.0.0/0 [120/1] via 155.1.146.6, 00:00:07, GigabitEthernet1.146

!   R3#show ip route | include ( 0.0.0.0)

Gateway of last resort is 155.1.13.1 to network 0.0.0.0
R*    0.0.0.0/0 [120/2] via 155.1.13.1, 00:00:27, GigabitEthernet1.13

!R7#show ip route | include ( 0.0.0.0)

Gateway of last resort is 155.1.37.3 to network 0.0.0.0
R*    0.0.0.0/0 [120/3] via 155.1.37.3, 00:00:22, GigabitEthernet1.37
```

Pitfall

Note in the above output that R6 does not have a default route installed in the routing table. Unlike OSPF, RIP does not require that a default route actually be installed in the routing table before originating one. For this reason, route feedback of R6's default origination will occur in this topology.

Based on the route-map attached to the default route origination of R6, the default advertisement is only sent out VLAN 146 toward R1. When R1 receives this, it is installed in the routing table and advertised on to R3. Note that the metric of the default route can be changed using offset-list.

```
R1#debug ip rip
RIP protocol debugging is on RIP: received v2 update from 155.1.146.6 on GigabitEthernet1.146
 0.0.0.0/0 via 0.0.0.0 in 1 hops
<output omitted> RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.13 (155.1.13.1)
RIP: build update entries 0.0.0.0/0 via 0.0.0.0, metric 2, tag 0
```

R3 now receives the update in from R1 with a metric of 2, and forwards the announcement to R7 with a metric of 3.

```
R3#debug ip rip
RIP protocol debugging is on RIP: received v2 update from 155.1.13.1 on GigabitEthernet1.13
0.0.0.0/0 via 0.0.0.0 in 2 hops
<output omitted> RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.37 (155.1.37.3)
```

https://t.me/learningnets

```
RIP: build update entries 0.0.0.0/0 via 0.0.0.0, metric 3, tag 0

<output omitted>
```

R7 receives the default route from R3 with a metric of 3 and installs it. Normally, R7 would continue to advertise this route on to R6, but in the above example the default route is filtered out from this advertisement with a prefix-list applied in conjunction with distribute-list.

```
R7#debug ip rip
RIP protocol debugging is on
RIP: received v2 update from 155.1.37.3 on GigabitEthernet1.37  0.0.0.0/0 via 0.0.0.0 in 3 hops
<output omitted>RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.67 (155.1.67.7)

RIP: build update entries
    150.1.2.2/32 via 0.0.0.0, metric 4, tag 0
    150.1.3.3/32 via 0.0.0.0, metric 2, tag 0
    150.1.5.5/32 via 0.0.0.0, metric 3, tag 0
    150.1.7.7/32 via 0.0.0.0, metric 1, tag 0
    150.1.8.8/32 via 0.0.0.0, metric 4, tag 0
    150.1.9.9/32 via 0.0.0.0, metric 2, tag 0
    150.1.10.10/32 via 0.0.0.0, metric 5, tag 0
    155.1.0.0/24 via 0.0.0.0, metric 2, tag 0
    155.1.5.0/24 via 0.0.0.0, metric 3, tag 0
    155.1.7.0/24 via 0.0.0.0, metric 1, tag 0
    155.1.8.0/24 via 0.0.0.0, metric 4, tag 0
    155.1.9.0/24 via 0.0.0.0, metric 2, tag 0
    155.1.10.0/24 via 0.0.0.0, metric 5, tag 0
    155.1.13.0/24 via 0.0.0.0, metric 2, tag 0
    155.1.37.0/24 via 0.0.0.0, metric 1, tag 0
    155.1.45.0/24 via 0.0.0.0, metric 3, tag 0
    155.1.58.0/24 via 0.0.0.0, metric 3, tag 0
    155.1.79.0/24 via 0.0.0.0, metric 1, tag 0
    155.1.108.0/24 via 0.0.0.0, metric 4, tag 0
```

Because R6 does not receive the default route from R7, there is no route feedback. Now let’s look at what happens in the topology when there is no filter configured on R7.

https://t.me/learningnets

```
R7#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#router rip
R7(config-router)#no distribute-list prefix NO_DEFAULT out GigabitEthernet1.67
```

R6 originates the default route to R1 with a metric of 1.

```
R6#debug ip rip
RIP protocol debugging is onRIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.146 (155.1.146.6)
RIP: build update entries0.0.0.0/0 via 0.0.0.0, metric 1, tag 0
```

The route is sent from R1, to R3, to R7, and then advertised back to R6. Because R6 does not actually have the default route installed in the routing table, it accepts R7's advertisement as valid.

```
RIP: received v2 update from 155.1.67.7 on GigabitEthernet1.67  0.0.0.0/0 via 0.0.0.0 in 4 hops
```

R6 now sends a triggered update to R1 with the default route, because the metric has changed to a value of 5. R1 sends the route to R3 and then again R7, which creates a routing loop.

```
RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.146
 (155.1.146.6)
RIP: build update entries0.0.0.0/0 via 0.0.0.0, metric 5, tag 0
```

The result of this problem can also be viewed in the rest of the topology through the `debug ip routing` output.

```
R1#debug ip routing
RT: rip's 0.0.0.0/0 (via 155.1.146.6) metric changed from distance/metric [120/13] to [120/1]

RT: updating rip 150.1.2.2/32 (0x0)  :
via 155.1.13.3 Gi1.13  0 1048578
```

To fix this problem, we must ensure that R6 does not install a default route via R7. One way to fix this, as shown above, is to filter the default route from being sent

https://t.me/learningnets

from R7 to R6. Another solution is to filter the default route inbound on R6 as is received from R7. Yet another solution is to actually configure a static default route on R6, either to Null0 or to another interface, so the RIP route could not be installed in the routing table when received from R7. The simplest solution, although it does not meet the task requirements, is to have R6 advertise the default route out all interfaces. If R6 sends the default route directly to R7, R7 cannot send it back to R6 because of split-horizon, and the loop is avoided.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Conditional Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 to originate a default route into the RIP domain, as long as it has a route to R9's Loopback0 prefix.

Configuration

R4:

```
ip prefix-list ROUTE_TO_R9_LOOP seq 5 permit 150.1.9.9/32
!
route-map TRACK_ROUTE_TO_R9_LOOP permit 10
 match ip address prefix-list ROUTE_TO_R9_LOOP
!
router rip
 default-information originate route-map TRACK_ROUTE_TO_R9_LOOP
```

Verification

https://t.me/learningnets

As long as R4 has a route to the network 150.1.9.9/32 installed in the routing table, it will advertise a default route.

```
R4#show ip route 150.1.9.9 255.255.255.255
Routing entry for 150.1.9.9/32
    Known via "rip", distance 120, metric 4
    Redistributing via rip
    Last update from 155.1.45.5 on GigabitEthernet1.45, 00:00:19 ago
    Routing Descriptor Blocks: * 155.1.45.5, from 155.1.45.5, 00:00:19 ago, via GigabitEthernet1.45
          Route metric is 4, traffic share count is 1
!R5#sh ip route | include (0.0.0.0)
    Gateway of last resort is 155.1.45.4 to network 0.0.0.0
R*    0.0.0.0/0 [120/1] via 155.1.45.4, 00:00:14, GigabitEthernet1.45
```

If R4 no longer has a route to 155.1.9.9/32, the default route is no longer advertised.

```
R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#interface Loopback0
R9(config-if)#shutdown
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to down
%LINK-5-CHANGED: Interface Loopback0, changed state to administratively down
!R4#show ip route 150.1.9.9 255.255.255.255
% Subnet not in table
!R5#show ip route | include ( 0.0.0.0)
R5#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Reliable Conditional Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 to originate a default route.
- Configure IP SLA on R1 to track ICMP reachability to R7's IPv4 address on VLAN 7.
  - ICMP Echo-Request should be sent each 5 seconds.

- Configure IP SLA tracking on R1 so that if an ICMP Echo-Reply is not received from VLAN 7, R1 withdraws its default route advertisement.

Configuration

R1:

```
ip sla 1
 frequency 5
 icmp-echo 155.1.7.7
!
ip sla schedule 1 start-time now life forever
!
track 1 ip sla 1
!
ip route 169.254.0.1 255.255.255.255 Null0 track 1
```

https://t.me/learningnets

```
!
ip prefix-list DUMMY_ROUTE_TRACKED_VIA_SLA seq 5 permit 169.254.0.1/32
!
route-map RELIABLY_TRACK_LINK_TO_VLAN7 permit 10
 match ip address prefix-list DUMMY_ROUTE_TRACKED_VIA_SLA
!
router rip
 default-information originate route-map RELIABLY_TRACK_LINK_TO_VLAN7
```

Verification

With this example, an IP SLA instance is introduced into conditional default origination. The SLA instance checks reachability to VLAN7 via ICMP every 5 seconds, with a timeout of 5 seconds (the default value). The SLA instance is then called from an enhanced object, which is called from a static route. This link local route of 169.254.0.1/32 could be any arbitrary dummy prefix. The dummy prefix is then called from a route-map, which is tied to the default route origination. Therefore, if R1 loses ICMP reachability to VLAN 7, the default route is withdrawn. As long as R1 has ICMP connectivity with R7's IPv4 address from VLAN 7, R1 injects the default route in the RIP domain.

```
R3#show ip route | in ( 0.0.0.0)
Gateway of last resort is 155.1.13.1 to network 0.0.0.0
R*    0.0.0.0/0 [120/1] via 155.1.13.1, 00:00:11, GigabitEthernet1.13
```

Simulate an indirect link failure so that R1 no longer has reachability to R7's IPv4 address on VLAN 7, which will cause the IP SLA on R1 to timeout, and finally for default route to be withdrawn and no longer advertised.

https://t.me/learningnets

```
R1#debug track state

track state debugging enabled
!R1#debug ip routing

IP routing debugging is on
!R1#debug ip rip

RIP protocol debugging is on
!R7#configure terminal
R7(config)#interface GigabitEthernet1.37
R7(config-subif)#shutdown
R7(config-subif)#interface gigabitEthernet 1.67
 R7(config-subif)#shutdown
```

The following log message on R1 confirms that R1 lost ICMP connectivity with R7; the object tracking state is now **DOWN**.

```
R1#
%TRACK-6-STATE: 1 ip sla 1 state Up -> Down
```

This will cause R1 to generate a flash update, poisoning the default route with the infinite metric of 16.

```
R1#
RIP: sending v2 flash update to 224.0.0.9 via Loopback0 (150.1.1.1)
RIP: build flash update entries
 0.0.0.0/0 via 0.0.0.0, metric 16, tag 0
RIP: Update contains 1 routes
RIP: Update queued RIP: sending v2 flash update to 224.0.0.9 via GigabitEthernet1.13 (155.1.13.1)
RIP: build flash update entries
 0.0.0.0/0 via 0.0.0.0, metric 16, tag 0
RIP: Update contains 1 routes
RIP: Update queued RIP: sending v2 flash update to 224.0.0.9 via GigabitEthernet1.146 (155.1.146.1)
RIP: build flash update entries
 0.0.0.0/0 via 0.0.0.0, metric 16, tag 0
RIP: Update contains 1 routes
RIP: Update queued RIP: sending v2 flash update to 224.0.0.9 via Tunnel0 (155.1.0.1)
RIP: build flash update entries
 0.0.0.0/0 via 0.0.0.0, metric 16, tag 0
```

https://t.me/learningnets

```
RIP: Update contains 1 routes
RIP: Update queued
RIP: Update sent via Loopback0
RIP: Update sent via GigabitEthernet1.13
RIP: Update sent via GigabitEthernet1.146
RIP: Update sent via Tunnel0
```

This also causes R1 to withdraw the dummy route from the routing table.

```
R1#
RT: del 169.254.0.1 via 0.0.0.0, static metric [1/0]
RT: delete subnet route to 169.254.0.1/32
```

All routers will remove the default route from the routing table, so R3 no longer has the default route in the routing table.

```
R3#show ip route | in (0.0.0.0)
R3#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Unicast Updates

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R5 and R8 so that RIPv2 updates sent over VLAN 58 use unicasts instead of multicasts.

Configuration

```
R5:
router rip
 passive-interface GigabitEthernet1.58
 neighbor 155.1.58.8

R8:

router rip
 passive-interface GigabitEthernet1.58
 neighbor 155.1.58.5
```

Verification

Like EIGRP and OSPF, the `neighbor` statement in RIP is used to send updates out

https://t.me/learningnets

the interface as unicast. Unlike other protocols, however, the `neighbor` statement does not automatically suppress the sending of the broadcast or multicast update. The additional `passive-interface` command is required to accomplish this.

```
R5#debug ip rip
RIP protocol debugging is on
RIP: sending v2 update to 155.1.58.8 via GigabitEthernet1.58 (155.1.58.5)
RIP: build update entries
        0.0.0.0/0 via 0.0.0.0, metric 2, tag 0
        150.1.1.1/32 via 0.0.0.0, metric 2, tag 0
        150.1.2.2/32 via 0.0.0.0, metric 2, tag 0
        150.1.3.3/32 via 0.0.0.0, metric 2, tag 0
        150.1.4.4/32 via 0.0.0.0, metric 2, tag 0
        150.1.5.5/32 via 0.0.0.0, metric 1, tag 0
        150.1.6.6/32 via 0.0.0.0, metric 3, tag 0
        150.1.7.7/32 via 0.0.0.0, metric 3, tag 0
        150.1.9.9/32 via 0.0.0.0, metric 4, tag 0
        155.1.0.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.5.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.7.0/24 via 0.0.0.0, metric 3, tag 0
        155.1.9.0/24 via 0.0.0.0, metric 4, tag 0
        155.1.13.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.37.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.45.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.67.0/24 via 0.0.0.0, metric 3, tag 0
        155.1.79.0/24 via 0.0.0.0, metric 3, tag 0
        155.1.146.0/24 via 0.0.0.0, metric 2, tag 0
!            R8#debug ip packet detail
IP packet debugging is on (detailed)
IP: s=155.1.58.8 (local), d=155.1.58.5 (GigabitEthernet1.58), len 132, local feature
UDP src=520, dst=520, feature skipped, Logical MN local(14), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FAL

IP: s=155.1.58.8 (local), d=155.1.58.5 (GigabitEthernet1.58), len 132, sending UDP src=520, dst=520
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Broadcast Updates

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 and R6 so that RIPv2 updates sent over VLAN 146 use broadcasts instead of multicasts.

Configuration

```
R1:
interface GigabitEthernet1.146
 ip rip v2-broadcast

R6:

interface GigabitEthernet1.146
 ip rip v2-broadcast
```

Verification

https://t.me/learningnets

Normally, RIPv2 updates are sent as multicast. The interface-level command `ip rip v2-broadcast` reverts back to using the all host broadcast address of **255.255.255.255** for updates.

```
R1#debug ip rip
RIP protocol debugging is on
RIP: sending v2 update to 255.255.255.255 via GigabitEthernet1.146 (155.1.146.1)
RIP: build update entries
        0.0.0.0/0 via 0.0.0.0, metric 1, tag 0
        150.1.1.1/32 via 0.0.0.0, metric 1, tag 0
        150.1.2.2/32 via 0.0.0.0, metric 3, tag 0
        150.1.3.3/32 via 0.0.0.0, metric 2, tag 0
        150.1.4.4/32 via 0.0.0.0, metric 3, tag 0
        150.1.5.5/32 via 0.0.0.0, metric 2, tag 0
        150.1.8.8/32 via 0.0.0.0, metric 3, tag 0
        150.1.10.10/32 via 0.0.0.0, metric 4, tag 0
        155.1.0.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.5.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.8.0/24 via 0.0.0.0, metric 3, tag 0
        155.1.10.0/24 via 0.0.0.0, metric 4, tag 0
        155.1.13.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.37.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.45.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.58.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.108.0/24 via 0.0.0.0, metric 3, tag 0
!R1#debug ip packet detail
IP packet debugging is on (detailed)
IP: s=155.1.146.1 (local), d=255.255.255.255 (GigabitEthernet1.146), len 312, local feature
UDP src=520, dst=520, feature skipped, Logical MN local(14), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FAL

s=155.1.146.1 (local), d=255.255.255.255 (GigabitEthernet1.146), len 312, sending broad/multicast
UDP src=520, dst=520
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - RIP

RIPv2 Source Validation

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic RIP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's connection to VLAN 146 and the Ethernet Link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R9 to use its Loopback0 address on its VLAN 79 Ethernet link.
- Ensure that RIPv2 updates sent across the Ethernet link can be installed in the routing tables of both R7 and R9.

Configuration

```
R7:
router rip
 no validate-update-source

R9:

interface GigabitEthernet1.79
 ip unnumbered Loopback0
```

Verification

https://t.me/learningnets

Verify that before IP unnumbered is configured on R9, RIP routes are correctly installed in both the R7 and R9 routing tables, and there is IPv4 connectivity.

```
R7#show ip route 155.1.9.0
Routing entry for 155.1.9.0/24
Known via "rip", distance 120, metric 1
  Redistributing via rip
  Last update from 155.1.79.9 on GigabitEthernet1.79, 00:00:09 ago
  Routing Descriptor Blocks: * 155.1.79.9, from 155.1.79.9, 00:00:09 ago, via GigabitEthernet1.79
      Route metric is 1, traffic share count is 1
!R7#ping 155.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.9.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
!R9#show ip route 155.1.7.0
Routing entry for 155.1.7.0/24
Known via "rip", distance 120, metric 1
  Redistributing via rip
  Last update from 155.1.79.7 on GigabitEthernet1.79, 00:00:09 ago
  Routing Descriptor Blocks: * 155.1.79.7, from 155.1.79.7, 00:00:09 ago, via GigabitEthernet1.79
      Route metric is 1, traffic share count is 1
!R9#ping 155.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.7.7, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 2/4/13 ms
```

After applying the IP unnumbered configuration on R9, because R9 will be sending RIP updates out on VLAN 79 Ethernet segment with a IPv4 address not in the same subnet as R7's IPv4 address, R7 will ignore these updates, and all RIP routes learned from R9 will slowly be removed from the routing table based on the flush timer.

```
R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#interface GigabitEthernet1.79
R9(config-subif)#ip unnumbered Loopback0
!R7#debug condition interface gigabitEthernet1.79
Condition 1 set
!R7#debug ip rip

RIP protocol debugging is on
```

Because RIP debugging has been enabled on R7, the following log messages will

https://t.me/learningnets

be displayed, basically saying that R7 ignores the RIP updates received from R9.

```
RIP: ignored v2 update from bad source 150.1.9.9 on GigabitEthernet1.79

RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.79 (155.1.79.7)
RIP: build update entries
        150.1.1.1/32 via 0.0.0.0, metric 3, tag 0
        150.1.2.2/32 via 0.0.0.0, metric 4, tag 0
        150.1.3.3/32 via 0.0.0.0, metric 2, tag 0
        150.1.4.4/32 via 0.0.0.0, metric 4, tag 0
        150.1.5.5/32 via 0.0.0.0, metric 3, tag 0
        150.1.6.6/32 via 0.0.0.0, metric 2, tag 0
        150.1.7.7/32 via 0.0.0.0, metric 1, tag 0
        150.1.8.8/32 via 0.0.0.0, metric 4, tag 0
        150.1.10.10/32 via 0.0.0.0, metric 5, tag 0
        155.1.0.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.5.0/24 via 0.0.0.0, metric 3, tag 0
        155.1.7.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.8.0/24 via 0.0.0.0, metric 4, tag 0
        155.1.10.0/24 via 0.0.0.0, metric 5, tag 0
        155.1.13.0/24 via 0.0.0.0, metric 2, tag 0
        155.1.37.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.45.0/24 via 0.0.0.0, metric 3, tag 0
        155.1.58.0/24 via 0.0.0.0, metric 3, tag 0
        155.1.67.0/24 via 0.0.0.0, metric 1, tag 0
        155.1.108.0/24 via 0.0.0.0, metric 4, tag 0
        155.1.146.0/24 via 0.0.0.0, metric 2, tag 0
```

Note that timers for RIP routes received from R9 are no longer refreshed based on the update interval.

```
R7#show ip route rip | i 1.9.
                 R        150.1.9.9 [120/1] via 155.1.79.9, 00:02:33
, GigabitEthernet1.79 R        155.1.9.0/24 [120/1] via 155.1.79.9, 00:02:33
, GigabitEthernet1.79
!R7#show ip rip database 150.1.9.9 255.255.255.255
150.1.9.9/32    [1] via 155.1.79.9, 00:02:51
, GigabitEthernet1.79
!R7#show ip rip database 155.1.9.0 255.255.255.0
155.1.9.0/24    [1] via 155.1.79.9, 00:02:55
, GigabitEthernet1.79
```

Note that R9 does not have the same problem for RIP updates received from R7; it

https://t.me/learningnets

accepts the RIP routes. This is because when IP unnumbered is used for an interface, the routing protocol no longer performs the check to determine whether the update was received from an IPv4 address within the directly connected subnet.

```
R9#debug condition interface gigabitEthernet1.79
Condition 1 set
!R9#debug ip rip

RIP protocol debugging is on
!RIP: received v2 update from 155.1.79.7 on GigabitEthernet1.79
     150.1.1.1/32 via 0.0.0.0 in 3 hops
     150.1.2.2/32 via 0.0.0.0 in 4 hops
     150.1.3.3/32 via 0.0.0.0 in 2 hops
     150.1.4.4/32 via 0.0.0.0 in 4 hops
     150.1.5.5/32 via 0.0.0.0 in 3 hops
     150.1.6.6/32 via 0.0.0.0 in 2 hops
     150.1.7.7/32 via 0.0.0.0 in 1 hops
     150.1.8.8/32 via 0.0.0.0 in 4 hops
     150.1.10.10/32 via 0.0.0.0 in 5 hops
     155.1.0.0/24 via 0.0.0.0 in 2 hops
     155.1.5.0/24 via 0.0.0.0 in 3 hops
     155.1.7.0/24 via 0.0.0.0 in 1 hops
     155.1.8.0/24 via 0.0.0.0 in 4 hops
     155.1.10.0/24 via 0.0.0.0 in 5 hops
     155.1.13.0/24 via 0.0.0.0 in 2 hops
     155.1.37.0/24 via 0.0.0.0 in 1 hops
     155.1.45.0/24 via 0.0.0.0 in 3 hops
     155.1.58.0/24 via 0.0.0.0 in 3 hops
     155.1.67.0/24 via 0.0.0.0 in 1 hops
     155.1.108.0/24 via 0.0.0.0 in 4 hops
     155.1.146.0/24 via 0.0.0.0 in 2 hops
RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.79 (150.1.9.9)
RIP: build update entries
        150.1.9.9/32 via 0.0.0.0, metric 1, tag 0
        155.1.9.0/24 via 0.0.0.0, metric 1, tag 0
!R9#show ip route rip | b Gateway

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets
R        150.1.1.1 [120/3] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        150.1.2.2 [120/4] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        150.1.3.3 [120/2] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        150.1.4.4 [120/4] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        150.1.5.5 [120/3] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        150.1.6.6 [120/2] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
```

https://t.me/learningnets

```
R        150.1.7.7 [120/1] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        150.1.8.8 [120/4] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        150.1.10.10 [120/5] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
      155.1.0.0/16 is variably subnetted, 14 subnets, 2 masks
R        155.1.0.0/24 [120/2] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.5.0/24 [120/3] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.7.0/24 [120/1] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.8.0/24 [120/4] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.10.0/24 [120/5] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.13.0/24 [120/2] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.37.0/24 [120/1] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.45.0/24 [120/3] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.58.0/24 [120/3] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.67.0/24 [120/1] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.108.0/24 [120/4] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
R        155.1.146.0/24 [120/2] via 155.1.79.7, 00:00:11, GigabitEthernet1.79
```

So the problem is unidirectional, and it can be fixed on R7 by configuring it to no longer perform the check against the source IPv4 address of the update. This is globally configured under the RIP process with the command **`no validate-update-source`** .

```
R7#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#router rip
R7(config-router)#no validate-update-source
!R7#show ip route rip | i 1.9.
R        150.1.9.9 [120/1] via 150.1.9.9, 00:00:06
R        155.1.9.0/24 [120/1] via 150.1.9.9, 00:00:06
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Network Statement

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP AS 100 on all devices in the topology.
- Enable EIGRP on all interfaces in the 150.1.0.0/16 and 155.1.0.0/16 networks on all devices.
- Any new interfaces added should not automatically have EIGRP enabled on them, regardless of their IPv4 addresses.
- For verification, all devices should have full IPv4 reachability throughout the network.

Configuration

```
R1:
router eigrp 100
 network 150.1.1.1 0.0.0.0
 network 155.1.0.1 0.0.0.0
 network 155.1.146.1 0.0.0.0
 network 155.1.13.1 0.0.0.0

R2:
router eigrp 100
 network 150.1.2.2 0.0.0.0
 network 155.1.0.2 0.0.0.0
```

https://t.me/learningnets

R3:

```
router eigrp 100
 network 150.1.3.3 0.0.0.0
 network 155.1.0.3 0.0.0.0
 network 155.1.13.3 0.0.0.0
 network 155.1.37.3 0.0.0.0
```

R4:

```
router eigrp 100
 network 150.1.4.4 0.0.0.0
 network 155.1.0.4 0.0.0.0
 network 155.1.45.4 0.0.0.0
```

R5:

```
interface Tunnel0
 no ip split-horizon eigrp 100
!
router eigrp 100
 network 150.1.5.5 0.0.0.0
 network 155.1.0.5 0.0.0.0
 network 155.1.5.5 0.0.0.0
 network 155.1.45.5 0.0.0.0
 network 155.1.58.5 0.0.0.0
```

R6:

```
router eigrp 100
 network 150.1.6.6 0.0.0.0
 network 155.1.67.6 0.0.0.0
 network 155.1.146.6 0.0.0.0
```

R7:

```
router eigrp 100
 network 150.1.7.7 0.0.0.0
 network 155.1.7.7 0.0.0.0
 network 155.1.37.7 0.0.0.0
 network 155.1.67.7 0.0.0.0
 network 155.1.79.7 0.0.0.0
```

R8:

```
router eigrp 100
 network 150.1.8.8 0.0.0.0
 network 155.1.8.8 0.0.0.0
 network 155.1.58.8 0.0.0.0
 network 155.1.108.8 0.0.0.0
```

https://t.me/learningnets

```
R9:

router eigrp 100
 network 150.1.9.9 0.0.0.0
 network 155.1.9.9 0.0.0.0
 network 155.1.79.9 0.0.0.0

R10:

router eigrp 100
 network 150.1.10.10 0.0.0.0
 network 155.1.10.10 0.0.0.0
 network 155.1.108.10 0.0.0.0
```

Verification

The network statement in EIGRP, like in OSPF, does not control which networks are being advertised, but instead controls which interfaces are running the EIGRP process. By using a wildcard address of 0.0.0.0 in the EIGRP network statement, only the interface with that particular IPv4 address will have the EIGRP process enabled. By using all zeros in the wildcard mask, there is no question as to which interfaces are running the process, and new interfaces added to the device will not automatically be running the EIGRP process.

When the network statement is configured, your first verification should always be to check the neighbor adjacencies with the **`show ip eigrp neighbors`** command. A “Q Cnt” (queue count) of zero means that there are no updates waiting to be sent or ackonwledged, thus the network has converged:

```
R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                 (sec)         (ms) Cnt
 Num 2   155.1.146.6             Gi1.146                 11 00:45:40    2   100 0
  17 1   155.1.0.5               Tu0                     14 00:45:47   88  1398 0
  33 0   155.1.13.3              Gi1.13                  13 00:46:07    1   100 0
  31
!R2#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                 (sec)         (ms) Cnt
 Num 0   155.1.0.5               Tu0                     14 00:45:47  141  1398 0
  30
!R3#show ip eigrp neighbors
```

https://t.me/learningnets

```
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                (sec)         (ms) Cnt
 Num 2   155.1.37.7              Gi1.37                 11 00:45:27    1   100 0
  18 1   155.1.0.5               Tu0                    14 00:45:47   93  1398 0
  32 0   155.1.13.1              Gi1.13                 11 00:46:07  169  1014 0
  35
!R4#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                (sec)         (ms) Cnt
 Num 1   155.1.45.5              Gi1.45                 12 00:45:47    1   100 0
  27 0   155.1.0.5               Tu0                    14 00:45:47  140  1398 0
  31
!R5#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                (sec)         (ms) Cnt
 Num 5   155.1.58.8              Gi1.58                 14 00:45:17    1   100 0
  7 4   155.1.0.1               Tu0                    14 00:45:47    5  1362 0
  34 3   155.1.0.3               Tu0                    13 00:45:47    5  1362 0
  33 2   155.1.45.4              Gi1.45                 13 00:45:47    2   100 0
  18 1   155.1.0.4               Tu0                    14 00:45:47    5  1362 0
  20 0   155.1.0.2               Tu0                    11 00:45:47    1  1362 0
  8
!R6#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                (sec)         (ms) Cnt
 Num 1   155.1.67.7              Gi1.67                 10 00:45:27    1   100 0
  19 0   155.1.146.1             Gi1.146                12 00:45:40    1   100 0
  36
!R7#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                (sec)         (ms) Cnt
 Num 2   155.1.79.9              Gi1.79                 14 00:45:06    4   100 0
  5 1   155.1.67.6               Gi1.67                 11 00:45:27    1   100 0
  16 0   155.1.37.3              Gi1.37                 13 00:45:27    3   100 0
  32
!R8#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                (sec)         (ms) Cnt
 Num 1   155.1.108.10            Gi1.108                12 00:44:49    2   100 0
  3
```

https://t.me/learningnets

```
0   155.1.58.5              Gi1.58                   11 00:45:17    1   100 0
  29
!R9#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                 (sec)         (ms) Cnt
 Num 0   155.1.79.7              Gi1.79                   13 00:45:06    1   100 0
  17
!R10#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO Q
  Seq                                                 (sec)         (ms) Cnt
 Num 0   155.1.108.8             Gi1.108                  10 00:44:49    4   100 0
  8
```

Additionally, note that there is an implicit design problem that must be solved in this topology that relates to split-horizon. Because R2's only connection to the rest of the EIGRP network is through the DMVPN network, all advertisements that R5 receives in the DMVPN Tunnel interface cannot be sent back out to R2. This is similar to the RIP split-horizon problem previously introduced, but EIGRP split-horizon is enabled on all interfaces, regardless of what the interface type is. To resolve this issue, R5 must disable split-horizon for this EIGRP process by using the command **`no ip split-horizon eigrp 100`** under the Tunnel interface. The end result of this configuration is that R2 should be learning about all EIGRP destinations through the DMVPN Tunnel:

```
R2#show ip route eigrp

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets
D        150.1.1.1 [90/27264000] via 155.1.0.5, 00:43:55, Tunnel0
D        150.1.3.3 [90/27264000] via 155.1.0.5, 00:43:55, Tunnel0
D        150.1.4.4 [90/27008256] via 155.1.0.5, 00:51:43, Tunnel0
D        150.1.5.5 [90/27008000] via 155.1.0.5, 00:51:45, Tunnel0
```

https://t.me/learningnets

```
D        150.1.6.6 [90/27264256] via 155.1.0.5, 00:43:55, Tunnel0
D        150.1.7.7 [90/27264256] via 155.1.0.5, 00:43:55, Tunnel0
D        150.1.8.8 [90/27008256] via 155.1.0.5, 00:51:13, Tunnel0
D        150.1.9.9 [90/27264512] via 155.1.0.5, 00:43:55, Tunnel0
D        150.1.10.10 [90/27008512] via 155.1.0.5, 00:50:45, Tunnel0
      155.1.0.0/16 is variably subnetted, 15 subnets, 2 masks
D        155.1.5.0/24 [90/26880256] via 155.1.0.5, 00:51:45, Tunnel0
D        155.1.7.0/24 [90/27136512] via 155.1.0.5, 00:43:55, Tunnel0
D        155.1.8.0/24 [90/26880512] via 155.1.0.5, 00:51:13, Tunnel0
D        155.1.9.0/24 [90/27136768] via 155.1.0.5, 00:43:55, Tunnel0
D        155.1.10.0/24 [90/26880768] via 155.1.0.5, 00:50:45, Tunnel0
D        155.1.13.0/24 [90/27136256] via 155.1.0.5, 00:43:55, Tunnel0
D        155.1.37.0/24 [90/27136256] via 155.1.0.5, 00:43:55, Tunnel0
D        155.1.45.0/24 [90/26880256] via 155.1.0.5, 00:51:45, Tunnel0
D        155.1.58.0/24 [90/26880256] via 155.1.0.5, 00:51:45, Tunnel0
D        155.1.67.0/24 [90/27136512] via 155.1.0.5, 00:43:55, Tunnel0
D        155.1.79.0/24 [90/27136512] via 155.1.0.5, 00:43:55, Tunnel0
D        155.1.108.0/24 [90/26880512] via 155.1.0.5, 00:51:13, Tunnel0
D        155.1.146.0/24 [90/27136256] via 155.1.0.5, 00:43:55, Tunnel0
```

IPv4 reachability in the network could be verified manually by pinging the various routes in the routing table, or by using a simple TCL shell script, as seen below:

```
R2#tclsh
R2(tcl)#foreach ADDRESS {
+>(tcl)#150.1.1.1
+>(tcl)#155.1.0.1
+>(tcl)#155.1.13.1
+>(tcl)#155.1.146.1
+>(tcl)#150.1.2.2
+>(tcl)#155.1.0.2
+>(tcl)#150.1.3.3
+>(tcl)#155.1.0.3
+>(tcl)#155.1.13.3
+>(tcl)#155.1.37.3
+>(tcl)#150.1.4.4
+>(tcl)#155.1.0.4
+>(tcl)#155.1.45.4
+>(tcl)#150.1.5.5
+>(tcl)#155.1.0.5
+>(tcl)#155.1.5.5
+>(tcl)#155.1.45.5
+>(tcl)#155.1.58.5
+>(tcl)#150.1.6.6
+>(tcl)#155.1.67.6
```

https://t.me/learningnets

```
+>(tcl)#155.1.146.6
+>(tcl)#150.1.7.7
+>(tcl)#155.1.7.7
+>(tcl)#155.1.37.7
+>(tcl)#155.1.67.7
+>(tcl)#155.1.79.7
+>(tcl)#150.1.8.8
+>(tcl)#155.1.8.8
+>(tcl)#155.1.58.8
+>(tcl)#155.1.108.8
+>(tcl)#150.1.9.9
+>(tcl)#155.1.9.9
+>(tcl)#155.1.79.9
+>(tcl)#150.1.10.10
+>(tcl)#155.1.10.10
+>(tcl)#155.1.108.10
+>(tcl)#} { ping $ADDRESS }
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.13.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.13.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.37.3, timeout is 2 seconds:!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.45.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.5.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.45.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.58.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.7.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.37.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.7, timeout is 2 seconds:!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.79.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.8.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/4/9 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.8.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.58.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.108.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.9.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.79.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/3/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.10.10, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/3/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.10.10, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.108.10, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/5 msR2(tcl)#tclquit

R2#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Auto-Summary

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP AS 100 on all devices in the topology.
- Enable EIGRP on all links in the 150.1.0.0/16 and 155.1.0.0/16 networks.
- Enable Auto-Summary under the EIGRP process on all devices in the topology.
- Disable EIGRP split-horizon on R5's tunnel interface connecting to the DMVPN network.
- Test IPv4 reachability throughout the network, and note any connectivity problems.

Configuration

https://t.me/learningnets

```
R1 – R10:

router eigrp 100
 auto-summary
 network 150.1.0.0 0.0.255.255
 network 155.1.0.0 0.0.255.255

R5:

interface Tunnel0
 no ip split-horizon eigrp 100
```

Verification

As of IOS release 15.0, EIGRP auto-summary now defaults to disabled. With EIGRP auto-summary enabled, VLSM is supported, but only for subnets that are within the same major network boundary. As network advertisements pass between major network boundaries, they are automatically summarized to their classful mask. The result of this behavior is that EIGRP cannot support discontiguous major networks, as seen in the reachability problems in this task.

Note that in the routing table output of devices in this topology, two automatic summaries are generated, one for the 150.1.0.0/16 network and one for the 155.1.0.0/16 network. Although all of the routers are advertising the classful summary of their Loopback subnet 150.1.0.0/16 to all of their neighbors, each router also installs a local EIGRP summary route to Null0. The end result is that they cannot learn about each other's summaries to 150.1.0.0/16, and reachability is not obtained between these networks:

```
R2#show ip route eigrp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/16 is variably subnetted, 2 subnets, 2 masks
D     150.1.0.0/16 is a summary, 00:00:28, Null0
```

https://t.me/learningnets

```
      155.1.0.0/16 is variably subnetted, 16 subnets, 3 masks
D        155.1.0.0/16 is a summary, 00:00:28, Null0

D        155.1.5.0/24 [90/26880256] via 155.1.0.5, 00:00:28, Tunnel0
D        155.1.7.0/24 [90/27136512] via 155.1.0.5, 00:00:09, Tunnel0
D        155.1.8.0/24 [90/26880512] via 155.1.0.5, 00:00:26, Tunnel0
D        155.1.9.0/24 [90/27136768] via 155.1.0.5, 00:00:09, Tunnel0
D        155.1.10.0/24 [90/26880768] via 155.1.0.5, 00:00:26, Tunnel0
D        155.1.13.0/24 [90/27136256] via 155.1.0.5, 00:00:09, Tunnel0
D        155.1.37.0/24 [90/27136256] via 155.1.0.5, 00:00:09, Tunnel0
D        155.1.45.0/24 [90/26880256] via 155.1.0.5, 00:00:28, Tunnel0
D        155.1.58.0/24 [90/26880256] via 155.1.0.5, 00:00:28, Tunnel0
D        155.1.67.0/24 [90/27136512] via 155.1.0.5, 00:00:09, Tunnel0
D        155.1.79.0/24 [90/27136512] via 155.1.0.5, 00:00:09, Tunnel0
D        155.1.108.0/24 [90/26880512] via 155.1.0.5, 00:00:26, Tunnel0
D        155.1.146.0/24 [90/27136256] via 155.1.0.5, 00:00:09, Tunnel0
```

If one of the router's Loopback advertisements was removed, the Null0 summary route would be removed, and they would be able to learn the /16 auto-summary generated by their peers. For example, take the following output when R10's Loopback0 is disabled:

```
R10#show ip route 150.1.0.0
Routing entry for 150.1.0.0/16, 2 known subnets
  Attached (1 connections)
  Variably subnetted with 2 masks
  Redistributing via eigrp 100D        150.1.0.0/16 is a summary, 00:07:07, Null0
C        150.1.10.10/32 is directly connected, Loopback0
!R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#interface Loopback0
R10(config-if)#shutdown
%SYS-5-CONFIG_I: Configured from console by console
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to down
%LINK-5-CHANGED: Interface Loopback0, changed state to administratively down
!R10#show ip route 150.1.0.0
Routing entry for 150.1.0.0/16
 Known via "eigrp 100"
, distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.108.8 on GigabitEthernet1.108, 00:00:08 ago
  Routing Descriptor Blocks:   * 155.1.108.8
, from 155.1.108.8, 00:00:08 ago, via GigabitEthernet1.108
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
```

https://t.me/learningnets

```
      Loading 1/255, Hops 1
```

In this case, R10 can learn the summary 150.1.0.0/16 from R8, but only because it does not have a locally installed Null0 route that matches the summary as well. From a reachability standpoint, R10 should now be able to reach R8's Loopback0, but none further than that, because R8 will null route any other packets matching 150.1.0.0/16:

```
R10#ping 150.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.8.8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/5 ms
!R10#debug ip icmp
ICMP packet debugging is on
!R10#ping 150.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:U.U.U
Success rate is 0 percent (0/5)ICMP: dst (155.1.108.10) host unreachable rcv from 155.1.108.8

ICMP: dst (155.1.108.10) host unreachable rcv from 155.1.108.8
ICMP: dst (155.1.108.10) host unreachable rcv from 155.1.108.8
```

Although there is little practical application to having EIGRP auto-summary enabled, it is important to understand how its behaviour can negatively affect a routing design. For example, if a router's configuration from a 15.x IOS version were to be rolled back to an IOS release of 12.4T or earlier, the default EIGRP auto-summary behavior would be re-enabled and potentially have a negative impact on reachability throughout the network.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Multi-AF Mode

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure an EIGRP processed named **MULTI-AF** on all devices in the topology.
- Configure the IPv4 unicast address-family to use EIGRP Autonomous System 100.
- Enable EIGRP on all links in the 150.1.0.0/16 and 155.1.0.0/16 networks.
- Disable EIGRP split-horizon on R5's tunnel interface connecting to the DMVPN network.
- When complete, all devices should have full IPv4 reachability throughout the network.

Configuration

```
R1 - R4, R6 - R10:
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family

R5:
```

https://t.me/learningnets

```
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface Tunnel0
   no split-horizon
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

Verification

EIGRP Multi-AF Mode, also known as EIGRP Named Mode, is an enhancement to the EIGRP syntax format introduced in IOS release 15.0. In EIGRP Classic Mode, now also referred to as EIGRP Autonomous System Mode, syntax was fragmented between the global process and the interface level. Furthermore, syntax at the interface level did not follow a strict hierarchy in its command structure. With EIGRP Multi-AF mode, all configuration is now consolidated to the global process.

In EIGRP Named Mode, the name of the EIGRP process is any locally significant string. The address-family configuration specifies which particular AF EIGRP is configured in, such as IPv4, the sub-address-family identifier (SAFI), such as unicast, the routing table, such as global or VRF, and the autonomous system. Furthermore, any commands that were previously issued at the interface level, such as **`no ip split-horizon eigrp`** , are now configured under the **`af-interface`** mode, with the specific interface denoted:

```
R1(config)#router eigrp MULTI-AF
R1(config-router)# address-family ipv4 unicast autonomous-system 100
R1(config-router-af)# af-interface GigabitEthernet1.146
R1(config-router-af-interface)#?

Address Family Interfaces configuration commands:
  add-paths           Advertise add paths
  authentication      authentication subcommands
  bandwidth-percent   Set percentage of bandwidth percentage limit
  bfd                 Enable Bidirectional Forwarding Detection
  dampening-change    Percent interface metric must change to cause update
  dampening-interval  Time in seconds to check interface metrics
  default             Set a command to its defaults
```

https://t.me/learningnets

```
  exit-af-interface   Exit from Address Family Interface configuration mode
  hello-interval      Configures hello interval
  hold-time           Configures hold time
  next-hop-self       Configures EIGRP next-hop-self
  no                  Negate a command or set its defaults
  passive-interface   Suppress address updates on an interface
  shutdown            Disable Address-Family on interface
  split-horizon       Perform split horizon
  summary-address     Perform address summarization
```

Attributes that are global to the EIGRP process are either configured under the SAFI itself or under the **`topology base`**, assuming that Multi-Topology Routing (MTR) is not being used:

```
R1(config)#router eigrp MULTI-AF
                            R1(config-router)# address-family ipv4 unicast autonomous-system 100
R1(config-router-af)#?
Address Family configuration commands:
  af-interface         Enter Address Family interface configuration
  default              Set a command to its defaults
  eigrp                EIGRP Address Family specific commands
  exit-address-family  Exit Address Family configuration mode
  help                 Description of the interactive help system
  maximum-prefix       Maximum number of prefixes acceptable in aggregate
  metric               Modify metrics and parameters for address advertisement
  neighbor             Specify an IPv4 neighbor router
  network              Enable routing on an IP network
  no                   Negate a command or set its defaults
  nsf                  Non-stop forwarding
  remote-neighbors     Specify IPv4 service remote neighbors
  shutdown             Shutdown address family
  timers               Adjust peering based timers
  topology             Topology configuration mode
!R1(config-router-af)#topology base
R1(config-router-af-topology)#?

Address Family Topology configuration commands:
  auto-summary         Enable automatic network number summarization
  default              Set a command to its defaults
  default-information  Control distribution of default information
  default-metric       Set metric of redistributed routes
  distance             Define an administrative distance
  distribute-list      Filter entries in eigrp updates
  eigrp                EIGRP specific commands
  exit-af-topology     Exit from Address Family Topology configuration mode
```

https://t.me/learningnets

```
  fast-reroute          Configure Fast-Reroute
  maximum-paths         Forward packets over multiple paths
  metric                Modify metrics and parameters for advertisement
  no                    Negate a command or set its defaults
  offset-list           Add or subtract offset from EIGRP metrics
  redistribute          Redistribute IPv4 routes from another routing protocol
  snmp                  Modify snmp parameters
  summary-metric        Specify summary to apply metric/filtering
  timers                Adjust topology specific timers
  traffic-share         How to compute traffic share over alternate paths
  variance              Control load balancing variance
```

In addition to the syntax parser change, EIGRP Wide Metric scaling is automatically enabled when EIGRP runs in Named Mode. This can be seen from the delay value now being measured in picoseconds in the EIGRP topology, as well as the RIB scaling factor seen below:

```
R1#show eigrp address-family ipv4 100 topology 150.1.2.2/32
EIGRP-IPv4 VR(MULTI-AF) Topology Entry for AS(100)/ID(150.1.1.1) for 150.1.2.2/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 10485841920, RIB is 81920640
  Descriptor Blocks:
  155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0 Composite metric is (10485841920/7209041920)
, route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit Total delay is 60001250000 picoseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 2
        Originating router is 150.1.2.2
  155.1.13.3 (GigabitEthernet1.13), from 155.1.13.3, Send flag is 0x0
Composite metric is (10486497280/10485841920)
, route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit Total delay is 60011250000 picoseconds

        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 3
        Originating router is 150.1.2.2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP MD5 & SHA-256 Authentication

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure an EIGRP process named **MULTI-AF** on R1 - R5.
- Configure EIGRP in Classic Mode on R6 - R10.
- Use Autonomous System 100 on all devices.
- Enable EIGRP on all links in the 150.1.0.0/16 and 155.1.0.0/16 networks.
- Disable EIGRP split-horizon on R5's tunnel interface connecting to the DMVPN network.
- Configure EIGRP authentication on R6 - R10 as follows:
  - Create a key-chain named **MD5_KEYS**.
  - Use the key-id **1** and the key-string **MD5_PASS**.
  - Apply the key-chain for MD5 authentication on all links with EIGRP adjacencies.
- Configure EIGRP authentication on R1 - R5 as follows:
  - Create an identical key-chain named **MD5_KEYS** on R1, R3, and R5.
  - Apply the key-chain for MD5 authentication toward their neighbors running EIGRP Classic Mode.
  - R1 - R5 should use the SHA-256 password **SHA_KEY** on their DMVPN tunnel interfaces.
  - R4 and R5 should use the SHA-256 password **SHA_DEFAULT** on their VLAN 45 connection, as well as any new interfaces added to the EIGRP

https://t.me/learningnets

process in the future.

- When complete, all devices should have full IPv4 reachability throughout the network.

Configuration

```
R1:
key chain MD5_KEYS
 key 1
   key-string MD5_PASS
!
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface Tunnel0
   authentication mode hmac-sha-256 SHA_KEY
  exit-af-interface
  !
  af-interface GigabitEthernet1.146
   authentication mode md5
   authentication key-chain MD5_KEYS
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family

R2:
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface Tunnel0
   authentication mode hmac-sha-256 SHA_KEY
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

https://t.me/learningnets

R3:

```
key chain MD5_KEYS
 key 1
   key-string MD5_PASS
!
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface Tunnel0
   authentication mode hmac-sha-256 SHA_KEY
  exit-af-interface
  !
  af-interface GigabitEthernet1.37
   authentication mode md5
   authentication key-chain MD5_KEYS
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R4:

```
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   authentication mode hmac-sha-256 SHA_DEFAULT
  exit-af-interface
  !
  af-interface Tunnel0
   authentication mode hmac-sha-256 SHA_KEY
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R5:

https://t.me/learningnets

```
key chain MD5_KEYS
 key 1
   key-string MD5_PASS
!
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   authentication mode hmac-sha-256 SHA_DEFAULT
  exit-af-interface
  !
  af-interface Tunnel0
   authentication mode hmac-sha-256 SHA_KEY
   no split-horizon
  exit-af-interface
  !
  af-interface GigabitEthernet1.58
   authentication mode md5
   authentication key-chain MD5_KEYS
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R6:

```
key chain MD5_KEYS
 key 1
  key-string MD5_PASS
!
interface GigabitEthernet1.67
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
interface GigabitEthernet1.146
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
router eigrp 100
 network 150.1.0.0 0.0.255.255
 network 155.1.0.0 0.0.255.255
```

R7:

https://t.me/learningnets

```
key chain MD5_KEYS
 key 1
  key-string MD5_PASS
!
interface GigabitEthernet1.37
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
interface GigabitEthernet1.67
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
interface GigabitEthernet1.79
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
router eigrp 100
 network 150.1.0.0 0.0.255.255
 network 155.1.0.0 0.0.255.255
```

R8:

```
key chain MD5_KEYS
 key 1
  key-string MD5_PASS
!
interface GigabitEthernet1.58
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
interface GigabitEthernet1.108
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
router eigrp 100
 network 150.1.0.0 0.0.255.255
 network 155.1.0.0 0.0.255.255
```

R9:

```
key chain MD5_KEYS
 key 1
  key-string MD5_PASS
!
interface GigabitEthernet1.79
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
```

https://t.me/learningnets

```
router eigrp 100
 network 150.1.0.0 0.0.255.255
 network 155.1.0.0 0.0.255.255

R10:

key chain MD5_KEYS
 key 1
  key-string MD5_PASS
!
interface GigabitEthernet1.108
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 MD5_KEYS
!
router eigrp 100
 network 150.1.0.0 0.0.255.255
 network 155.1.0.0 0.0.255.255
```

Verification

EIGRP supports MD5 authentication in Classic (Autonomous System) Mode, and both MD5 and SHA-256 in Multi-AF (Named) Mode. For MD5 authentication in both Classic and Named modes, the key chain is defined globally. The key chain can contain multiple keys, but only the lowest active key number will be exchanged in EIGRP packets. Note that the key ID must match for authentication to occur, because this number is exchanged in the hello packets. In Classic Mode, the authentication is applied at the `af-interface` mode under the SAFI. In either case, the authentication can be verified as seen below:

```
R6#show ip eigrp interface detail GigabitEthernet1.146
EIGRP-IPv4 Interfaces for AS(100)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi1.146                  1        0/0        0/0          1       0/0           50           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 3/0
  Hello's sent/expedited: 535/2
  Un/reliable mcasts: 0/4  Un/reliable ucasts: 4/1
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 1
  Topology-ids on interface - 0
```

https://t.me/learningnets

```
 Authentication mode is md5,  key-chain is "MD5_KEYS"

! R1#show eigrp address-family ipv4 100 interfaces detail GigabitEthernet1.146
EIGRP-IPv4 VR(MULTI-AF) Address-Family Interfaces for AS(100)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi1.146                  1        0/0       0/0           1       0/0           50           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 4/1
  Hello's sent/expedited: 526/2
  Un/reliable mcasts: 0/4  Un/reliable ucasts: 5/2
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 1  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0 Authentication mode is md5,  key-chain is "MD5_KEYS"
!R6#debug eigrp packet
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on

EIGRP: Sending HELLO on Gi1.146 - paklen 60
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 1

EIGRP: Received HELLO on Gi1.146 - paklen 60 nbr 155.1.146.1
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
```

If authentication were failing, the debug output would indicate this:

```
R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface GigabitEthernet1.146
R6(config-subif)#no ip authentication mode eigrp 100 md5
R6(config-subif)#do debug eigrp packet
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on
EIGRP: Gi1.146:ignored packet from 155.1.146.1, opcode = 5 (authentication off)
EIGRP: Dropping peer,invalid authentication
EIGRP: Sending HELLO on Gi1.146 - paklen 20
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.146.1 (GigabitEthernet1.146) is down:Auth failure
```

Likewise, a missing or invalid key would be indicated in this debug output:

https://t.me/learningnets

```
R6(config-subif)#ip authentication mode eigrp 100 md5
    %DUAL-5-NBRCHANGE: EIGRP-IPv4 100:Neighbor 155.1.146.1 (GigabitEthernet1.146) is up
: new adjacencyR6(config-subif)#no  ip authentication key-chain eigrp 100 MD5_KEYS
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100:Neighbor 155.1.146.1 (GigabitEthernet1.146) is down
: keychain changedR6(config-subif)#do debug eigrp packet
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is onEIGRP: Gi1.146: ignored packet from 155.1.146.1, opcode = 5
(invalid authentication or key-chain missing)

EIGRP: Sending TIDLIST on GigabitEthernet1.146 - 1 items
EIGRP: Sending HELLO on Gi1.146 - paklen 30
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
```

In Named Mode, SHA-256 authentication can be configured at the `af-interface` mode. The current implementation does not support key-chains or key IDs, which means it supports neither multiple keys nor automatic key rotation. Another useful feature of the new EIGRP Named Mode is that options can be configured at the `af-interface default` , which applies to all links at the same time. Within the scope of authentication, this can be used to configure a default key for all interfaces, or a default fallback key for interfaces that do not have a specific key applied:

```
R5#debug eigrp packet
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on
R5#EIGRP:received packet with HMAC-SHA-256 authentication
EIGRP: Received HELLO on Tu0 - paklen 76 nbr 155.1.0.4
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: received packet with HMAC-SHA-256 authentication
EIGRP: Received HELLO on Gi1.45 - paklen 76 nbr 155.1.45.4
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Sending HELLO on Gi1.58 - paklen 60
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0EIGRP:
received packet with MD5 authentication, key id = 1

EIGRP: Received HELLO on Gi1.58 - paklen 60 nbr 155.1.58.8
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
```

Pitfall

Like RIP, a white space in the key-string can cause authentication failure:

```
R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#key chain MD5_KEYS
R6(config-keychain)#key 1
R6(config-keychain-key)#key-string CISCO ?
```

https://t.me/learningnets

```
LINE    <cr>
!R6#show key chain
Key-chain MD5_KEYS:    key 1 -- text "CISCO "

        accept lifetime (always valid) - (always valid) [valid now]
        send lifetime (always valid) - (always valid) [valid now]
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Key Chain Rotation

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on R1 - R4 using AS 100.
- Configure an EIGRP process named **MULTI-AF** on R5 using AS 100.
- Enable EIGRP on the DMVPN tunnel between these devices.
- Set the clock on R5 to the current time, and configure it as an NTP master.
- Configure R1 - R4 to get NTP time from R5.
- Authenticate the EIGRP adjacencies on the DMVPN network between R1, R2, R3, R4, and R5 with a key-chain named **KEY_ROTATION** as follows:
  - Create key ID **10** with the password **CISCO10**.
  - Create key ID **20** with the password **CISCO20**.
  - Key ID **10** should be used from 00:00:00 Jan 1 1993 until 00:05:00 Jan 1 2030, and should be accepted for 10 minutes past this time.
  - Key ID **20** should be sent starting at 00:00:00 Jan 1 2030, and should be accepted any time after this time.
- Modify R5's clock to 23:59:00 Dec 31 2029.
- Remove and re-apply R1 - R4's NTP server configuration toward R5.
- If your configuration is successful, R1 - R4 should update their clocks through NTP and roll over to the new authentication key without a loss of EIGRP adjacencies.

https://t.me/learningnets

Configuration

```
R1 - R4:
router eigrp 100
 network 155.1.0.0 0.0.0.255
!
ntp server 155.1.0.5
!
key chain KEY_ROTATION
 key 10
  key-string CISCO10
  accept-lifetime 00:00:00 Jan 1 1993 00:15:00 Jan 1 2030
  send-lifetime 00:00:00 Jan 1 1993 00:05:00 Jan 1 2030
 key 20
  key-string CISCO20
  accept-lifetime 00:00:00 Jan 1 2030 infinite
  send-lifetime 00:00:00 Jan 1 2030 infinite
!
interface Tunnel0
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 KEY_ROTATION
R5:

ntp master 1
!
key chain KEY_ROTATION
 key 10
  key-string CISCO10
  accept-lifetime 00:00:00 Jan 1 1993 00:15:00 Jan 1 2030
  send-lifetime 00:00:00 Jan 1 1993 00:05:00 Jan 1 2030
 key 20
  key-string CISCO20
  accept-lifetime 00:00:00 Jan 1 2030 infinite
  send-lifetime 00:00:00 Jan 1 2030 infinite
!
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface Tunnel0
   authentication mode md5
   authentication key-chain KEY_ROTATION
  exit-af-interface
```

https://t.me/learningnets

```
  !
  topology base
  exit-af-topology
  network 155.1.0.0 0.0.0.255
 exit-address-family
```

Verification

Pitfall

Whenever time-based authentication is configured, ensure that all devices agree on the same time. This can be manually configured with the **`clock set`** command or through NTP. Also, the additional overlap of sending/receiving keys ensures that a drift away from the accurate time will not cause routing adjacencies to be lost. NTP time synchronization can be verified as seen below:

```
R5#show ntp status
Clock is synchronized, stratum 1, reference is .LOCL.
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 15500 (1/100 of seconds), resolution is 4000
reference time is D70AEA86.AC083300 (03:20:38.672 UTC Wed Apr 30 2014)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 2.34 msec, peer dispersion is 1.20 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 16, last update was 12 sec ago.
!R1#show ntp status
Clock is synchronized, stratum 2, reference is 155.1.0.5

nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 12900 (1/100 of seconds), resolution is 4000
reference time is D70AEA13.45A1CB80 (03:18:43.272 UTC Wed Apr 30 2014)
clock offset is -34.5000 msec, root delay is 2.00 msec
root dispersion is 7916.29 msec, peer dispersion is 65.34 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 64, last update was 127 sec ago.
```

With EIGRP MD5 authentication, only the lowest active key number is sent for authentication. Current active keys can be verified as follows:

```
R1#show clock
03:27:06.485 UTC Wed Apr 30 2014
```

https://t.me/learningnets

```
!R1#show key chain

Key-chain KEY_ROTATION: key 10
 -- text "CISCO10"        accept lifetime (00:00:00 UTC Jan 1 1993) - (00:15:00 UTC Jan 1 2030)
[valid now]
        send lifetime (00:00:00 UTC Jan 1 1993) - (00:05:00 UTC Jan 1 2030) [valid now]
    key 20 -- text "CISCO20"
        accept lifetime (00:00:00 UTC Jan 1 2030) - (infinite)
        send lifetime (00:00:00 UTC Jan 1 2030) - (infinite)
!        R1#debug eigrp packet hello
    (HELLO)
EIGRP Packet debugging is on EIGRP: received packet with MD5 authentication, key id = 10

EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.5
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Sending HELLO on Tu0 - paklen 60
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
```

Next, R5 is updated with the new time:

```
R5#show clock
03:29:38.112 UTC Wed Apr 30 2014
!R5#clock set 23:59:00 Dec 31 2029
%SYS-6-CLOCKUPDATE: System clock has been updated
 from 03:29:53 UTC Wed Apr 30 2014 to 23:59:00 UTC Mon Dec 31 2029, configured from console
!R5#show clock
23:59:16.834 UTC Mon Dec 31 2029

!R5#show key chain
Key-chain KEY_ROTATION: key 10
 -- text "CISCO10"        accept lifetime (00:00:00 UTC Jan 1 1993) - (00:15:00 UTC Jan 1 2030)
[valid now]
        send lifetime (00:00:00 UTC Jan 1 1993) - (00:05:00 UTC Jan 1 2030) [valid now]

    key 20 -- text "CISCO20"
        accept lifetime (00:00:00 UTC Jan 1 2030) - (infinite)
        send lifetime (00:00:00 UTC Jan 1 2030) - (infinite)
```

To update the rest of the devices' time, remove and re-apply their NTP configuration:

```
R1#show clock
03:31:36.741 UTC Wed Apr 30 2014
!R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#no ntp server 155.1.0.5
R1(config)#ntp server 155.1.0.5
```

https://t.me/learningnets

```
!R1#show clock
00:01:31.407 UTC Tue Jan 1 2030
```

Prior to the configured rotation time, R5 should still be receiving key ID 10:

```
R5#debug eigrp packet hello
    (HELLO)
EIGRP Packet debugging is on
R5#
EIGRP: Sending HELLO on Tu0 - paklen 60
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 10
EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.3
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 10
EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.4
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 10
EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.1
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 10

EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.2
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
```

At the time of rotation, all devices should invalidate the sending of key 10, but they should still accept it as validdue to the configured ovelapping period:

https://t.me/learningnets

```
R1#show clock
00:05:02.114 UTC Tue Jan 1 2030

!R1#show key chain

Key-chain KEY_ROTATION: key 10
 -- text "CISCO10"         accept lifetime (00:00:00 UTC Jan 1 1993) - (00:15:00 UTC Jan 1 2030)
[valid now]

        send lifetime (00:00:00 UTC Jan 1 1993) - (00:05:00 UTC Jan 1 2030) key 20
 -- text "CISCO20"         accept lifetime (00:00:00 UTC Jan 1 2030) - (infinite) [valid now]
        send lifetime (00:00:00 UTC Jan 1 2030) - (infinite) [valid now]

!R1#show ip eigrp neighbor

EIGRP-IPv4 Neighbors for AS(100) H   Address                 Interface              Hold Uptime
   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
0   155.1.0.5               Tu0                      10 00:17:24
   23  2097  0  5
```

Note that the uptime of the EIGRP neighbors indicates that an adjacency flap has not occurred. R5 should now be receiving EIGRP packets with key ID 20:

https://t.me/learningnets

```
R5#debug eigrp packet hello

    (HELLO)
EIGRP Packet debugging is on
! EIGRP: received packet with MD5 authentication, key id = 10

EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.2
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Sending HELLO on Tu0 - paklen 60
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 20

EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.1
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 20

EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.4
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 20

EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.3
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: received packet with MD5 authentication, key id = 20

EIGRP: Received HELLO on Tu0 - paklen 60 nbr 155.1.0.2
```

A short time later, key ID 10 stops being accepted, and rotation is complete:

```
R5#show key chain
Key-chain KEY_ROTATION:
    key 10 -- text "CISCO10"
        accept lifetime (00:00:00 UTC Jan 1 1993) - (00:15:00 UTC Jan 1 2030)
        send lifetime (00:00:00 UTC Jan 1 1993) - (00:05:00 UTC Jan 1 2030) key 20
 -- text "CISCO20"        accept lifetime (00:00:00 UTC Jan 1 2030) - (infinite) [valid now]
        send lifetime (00:00:00 UTC Jan 1 2030) - (infinite) [valid now]
!R5#show eigrp address-family ipv4 100 neighbors
EIGRP-IPv4 VR(MULTI-AF) Address-Family Neighbors for AS(100)
H   Address                 Interface              Hold Uptime
   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
3   155.1.0.3               Tu0                      12 00:29:13
    9  1398  0  22   155.1.0.2               Tu0                      12 00:29:13
   10  1398  0  21   155.1.0.1               Tu0                      11 00:29:13
   12  1398  0  2
```

https://t.me/learningnets

```
0   155.1.0.4               Tu0                     13 00:29:13
 1280  5000  0  2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Unicast Updates

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on R1 - R4 using AS 100.
- Configure an EIGRP process named **MULTI-AF** on R5 using AS 100.
- Enable EIGRP on the DMVPN tunnel between these devices.
- Configure R4 and R5 so that they exchange EIGRP packets only as unicasts with each other on the DMVPN network.

Configuration

```
R1 - R3:
router eigrp 100
 network 155.1.0.0 0.0.0.255

R4:
router eigrp 100
 network 155.1.0.0 0.0.0.255
 neighbor 155.1.0.5 Tunnel0

R5:

router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
```

https://t.me/learningnets

```
  !
  topology base
  exit-af-topology
  neighbor 155.1.0.4 Tunnel0
  network 155.1.0.0 0.0.0.255
 exit-address-family
```

Verification

By default, EIGRP hello packets are sent to the multicast address 224.0.0.10, whereas topology synchronization between two neighbors is unicast. Like RIP, the **`neighbor`** statement under the EIGRP process is used to send hello packets as unicasts. However, unlike RIP, the **`passive-interface`** command is not needed to suppress the sending of the multicast hellos. This means that if the **`neighbor`** statement is configured on one end of the adjacency, it is required to be configured on the other end as well.

Below we see that R5 has established EIGRP adjacency with R1 - R4:

```
R5#show eigrp address-family ipv4 100 neighbors

EIGRP-IPv4 VR(MULTI-AF) Address-Family Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num 3 155.1.0.2
               Tu0                      12 00:00:07    7  1398  0  3 2 155.1.0.4
               Tu0                      12 00:00:07    8  1398  0  3 1 155.1.0.1
               Tu0                      11 00:00:07    2  1398  0  3 0 155.1.0.3
               Tu0                      11 00:00:07   11  1398  0  3
```

After the neighbor statement is configured on R5, all adjacencies are dropped, because all multicast hellos are dropped. When R4 configures the neighbor statement as well, their adjacency is re-established:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router eigrp MULTI-AF
R5(config-router)# address-family ipv4 unicast autonomous-system 100
R5(config-router-af)#neighbor 155.1.0.4 tunnel0
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.2 (Tunnel0) is down: Static peer configured
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.4 (Tunnel0) is down: Static peer configured
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.1 (Tunnel0) is down: Static peer configured
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.3 (Tunnel0) is down: Static peer configured
!R4#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#router eigrp 100
                                        R4(config-router)#neighbor 155.1.0.5 tunnel0
```

https://t.me/learningnets

```
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100:Neighbor 155.1.0.5 (Tunnel0) is up: new adjacency
```

Because R5 is configured for unicast exchange on the DMVPN tunnel, adjacencies to all other spokes besides R4 are rejected. Basically EIGRP cannot be enabled for both unicast and multicast over the same interface, thus in this case all received EIGRP multicas HELLO packets are ignored:

```
R5#show eigrp address-family ipv4 100 neighbors
EIGRP-IPv4 VR(MULTI-AF) Address-Family Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num0155.1.0.4
               Tu0                      12 00:00:11    8  1398  0  5
!R5#debug eigrp packet
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on
R5#EIGRP:Received HELLO on Tu0 - paklen 20 nbr 155.1.0.4
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Received HELLO on Tu0 - paklen 20 nbr 155.1.0.3
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0EIGRP:Ignore multicast Hello Tu0 155.1.0.3
EIGRP: Received HELLO on Tu0 - paklen 20 nbr 155.1.0.1
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0EIGRP:Ignore multicast Hello Tu0 155.1.0.1
EIGRP: Sending HELLO on Tu0 - paklen 20 nbr 155.1.0.4
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
EIGRP: Received HELLO on Tu0 - paklen 20 nbr 155.1.0.2
  AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0EIGRP:Ignore multicast Hello Tu0 155.1.0.2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Summarization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on R1 - R5 using AS 100.
- Configure an EIGRP process named **MULTI-AF** on R6 - R10 using AS 100.
- Enable EIGRP on all links in the 150.1.0.0/16 network.
- Enable EIGRP on all links in the 155.1.0.0/16 network.
- Disable Split-Horizon on R5's link to the DMVPN network.
- Configure the following interfaces on R4, and redistribute them into EIGRP:
  - Loopback40 - 4.0.0.4/24
  - Loopback41 - 4.0.1.4/24
  - Loopback42 - 4.0.2.4/24
  - Loopback43 - 4.0.3.4/24
- Configure the following interfaces on R6, and redistribute them into EIGRP:
  - Loopback60 - 6.0.0.6/24
  - Loopback61 - 6.0.1.6/24
  - Loopback62 - 6.0.2.6/24
  - Loopback63 - 6.0.3.6/24
- Configure R4 and R6 to advertise a single summary of these new Loopbacks into EIGRP.

https://t.me/learningnets

Configuration

```
R1-R5:
router eigrp 100
 network 150.1.0.0 0.0.255.255
 network 155.1.0.0 0.0.255.255
R5:
interface Tunnel0
 no ip split-horizon eigrp 100

R6-R10:
 router eigrp MULTI-AF
 !
  address-family ipv4 unicast autonomous-system 100
   !
   topology base
   exit-af-topology
   network 150.1.0.0
   network 155.1.0.0

R4:
interface loopback 40
 ip address 4.0.0.4 255.255.255.0
!
interface loopback 41
 ip address 4.0.1.4 255.255.255.0
!
interface loopback 42
 ip address 4.0.2.4 255.255.255.0
!
interface loopback 43
 ip address 4.0.3.4 255.255.255.0
!
ip prefix-list CONNECTED_TO_EIGRP seq 5 permit 4.0.0.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 10 permit 4.0.1.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 15 permit 4.0.2.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 20 permit 4.0.3.0/24
!
route-map CONNECTED_TO_EIGRP permit 10
 match ip address prefix-list CONNECTED_TO_EIGRP
!
router eigrp 100
 redistribute connected route-map CONNECTED_TO_EIGRP
```

https://t.me/learningnets

```
!
interface Tunnel0
 ip summary-address eigrp 100 4.0.0.0 255.255.252.0
!
interface GigabitEthernet1.45
 ip summary-address eigrp 100 4.0.0.0 255.255.252.0
```

R6:

```
interface Loopback 60
 ip address 6.0.0.6 255.255.255.0
!
interface Loopback 61
 ip address 6.0.1.6 255.255.255.0
!
interface Loopback 62
 ip address 6.0.2.6 255.255.255.0
!
interface Loopback 63
 ip address 6.0.3.6 255.255.255.0
!
ip prefix-list CONNECTED_TO_EIGRP seq 5 permit 6.0.0.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 10 permit 6.0.1.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 15 permit 6.0.2.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 20 permit 6.0.3.0/24
!
route-map CONNECTED_TO_EIGRP permit 10
 match ip address prefix-list CONNECTED_TO_EIGRP
!
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 100
 !
 af-interface GigabitEthernet1.67
  summary-address 6.0.0.0 255.255.252.0
 exit-af-interface
 !
 af-interface GigabitEthernet1.146
  summary-address 6.0.0.0 255.255.252.0
 exit-af-interface
 !
  topology base
   redistribute connected route-map CONNECTED_TO_EIGRP
```

https://t.me/learningnets

```
  exit-af-topology
```

Verification

Like RIP, EIGRP supports summarization at the interface level anywhere throughout the topology, but it does not have the limitation of being unable to summarize beyond the classful boundary. When a summary is configured in EIGRP, all subnets that make up the summary are suppressed from being advertised out the link. From a design perspective, this feature can be used to both reduce the size of the routing table and limit the scope of EIGRP query messages.

In the below output, we can see that R5 learns the about the EIGRP summary 4.0.0.0/22 through the Ethernet segment, and the EIGRP summary 6.0.0.0/22 through the DMVPN cloud. Based on longest match routing, we can say that R5 will route traffic destined for any subnet of the 4.0.0.0/22 aggregate out the Ethernet segment and out the DMVPN cloud for any subnet of the 6.0.0.0/22 aggregate.

```
R5#show ip route | include 4.0
      4.0.0.0/22 is subnetted, 1 subnets
D        4.0.0.0 [90/130816] via 155.1.45.4, 00:23:14, GigabitEthernet1.45
!R5#traceroute 4.0.0.4 numeric
  Type escape sequence to abort.
  Tracing the route to 4.0.0.4
  VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 45 msec *  3 msec
!R5#show ip route | include 6.0
     6.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
D        6.0.0.0/22 [90/25856544] via 155.1.0.3, 00:07:02, Tunnel0
!R5#traceroute 6.0.0.6 num
  Type escape sequence to abort.
  Tracing the route to 4.0.0.4
  VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 5 msec 8 msec 4 msec


      2 155.1.146.6 12 msec *  6 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Summarization with Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on R1 - R5 using AS 100.
  - Enable EIGRP only over the DMVPN cloud.
  - Disable Split-Horizon on R5's link to the DMVPN network.
- Configure an EIGRP process named **MULTI-AF** on R6, R7 and R9 using AS 200.
  - Enable EIGRP on all links in the 155.1.0.0/16 network.
- Configure the following interfaces on R4, and redistribute them into EIGRP:
  - Loopback40 - 4.0.0.4/24
  - Loopback41 - 4.0.1.4/24
  - Loopback42 - 4.0.2.4/24
  - Loopback43 - 4.0.3.4/24
- Configure the following interfaces on R6, and redistribute them into EIGRP:
  - Loopback60 - 6.0.0.6/24
  - Loopback61 - 6.0.1.6/24
  - Loopback62 - 6.0.2.6/24
  - Loopback63 - 6.0.3.6/24
- Configure summarization on R4 and R6 to advertise a default route into EIGRP instead of the redistributed Loopbacks.

https://t.me/learningnets

Configuration

```
R1-R5:
router eigrp 100
 network 155.1.0.0 0.0.0.255

R5:
interface Tunnel0
 no ip split-horizon eigrp 100

R6, R7, R9:
 router eigrp MULTI-AF
 !
  address-family ipv4 unicast autonomous-system 200
   !
   topology base
   exit-af-topology
   network 155.1.0.0

R4:
interface loopback 40
 ip address 4.0.0.4 255.255.255.0
!
interface loopback 41
 ip address 4.0.1.4 255.255.255.0
!
interface loopback 42
 ip address 4.0.2.4 255.255.255.0
!
interface loopback 43
 ip address 4.0.3.4 255.255.255.0
!
ip prefix-list CONNECTED_TO_EIGRP seq 5 permit 4.0.0.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 10 permit 4.0.1.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 15 permit 4.0.2.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 20 permit 4.0.3.0/24
!
route-map CONNECTED_TO_EIGRP permit 10
 match ip address prefix-list CONNECTED_TO_EIGRP
!
router eigrp 100
 redistribute connected route-map CONNECTED_TO_EIGRP
!
```

https://t.me/learningnets

```
interface Tunnel0
 ip summary-address eigrp 100 0.0.0.0 0.0.0.0
```

R6:

```
interface Loopback 60
 ip address 6.0.0.6 255.255.255.0
!
interface Loopback 61
 ip address 6.0.1.6 255.255.255.0
!
interface Loopback 62
 ip address 6.0.2.6 255.255.255.0
!
interface Loopback 63
 ip address 6.0.3.6 255.255.255.0
!
ip prefix-list CONNECTED_TO_EIGRP seq 5 permit 6.0.0.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 10 permit 6.0.1.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 15 permit 6.0.2.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 20 permit 6.0.3.0/24
!
route-map CONNECTED_TO_EIGRP permit 10
 match ip address prefix-list CONNECTED_TO_EIGRP
!
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 200
 !
 af-interface GigabitEthernet1.67
  summary-address 0.0.0.0 0.0.0.0
 exit-af-interface
 !
  topology base
   redistribute connected route-map CONNECTED_TO_EIGRP
  exit-af-topology
```

Verification

Summarization can also be used to originate a default route in EIGRP. The disadvantage of this configuration, however, is that all subnets previously advertised out an interface will be suppressed, because all IPv4 networks are a subnet of the

https://t.me/learningnets

aggregate 0.0.0.0/0:

```
R9#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.79.7 to network 0.0.0.0
D*    0.0.0.0/0 [90/16000] via 155.1.79.7, 00:00:50, GigabitEthernet1.79
      155.1.0.0/16 is variably subnetted, 8 subnets, 2 masks
D        155.1.7.0/24 [90/15360] via 155.1.79.7, 00:00:57, GigabitEthernet1.79
D        155.1.37.0/24 [90/15360] via 155.1.79.7, 00:00:57, GigabitEthernet1.79
D        155.1.67.0/24 [90/15360] via 155.1.79.7, 00:00:57, GigabitEthernet1.79
D        155.1.146.0/24 [90/20480] via 155.1.79.7, 00:00:57, GigabitEthernet1.79
!R9#show ip route 6.0.0.6
% Network not in table
!R9#traceroute 6.0.0.6
Type escape sequence to abort.
Tracing the route to 6.0.0.6
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.79.7 4 msec 1 msec 1 msec 2 155.1.67.6 3 msec *  2 msec
!R1#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.0.5 to network 0.0.0.0
D*    0.0.0.0/0 [90/27264000] via 155.1.0.5, 00:03:44, Tunnel0
!R1#show ip route 4.0.0.4
% Network not in table
!R1#traceroute 4.0.0.4
Type escape sequence to abort.
Tracing the route to 4.0.0.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.0.5 4 msec 1 msec 1 msec 2 155.1.0.4 3 msec *  9 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Summarization with Leak Map

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on R4 and R5 using AS 100.
  - Enable EIGRP only over both the DMVPN cloud and Ethernet segment.
- Configure an EIGRP process named **MULTI-AF** on R1, R3, R6 and R7 using AS 200.
  - Enable EIGRP on all links in the 155.1.0.0/16 network.
- Configure the following interfaces on R4, and redistribute them into EIGRP:
  - Loopback40 - 4.0.0.4/24
  - Loopback41 - 4.0.1.4/24
- Configure the following interfaces on R6, and redistribute them into EIGRP:
  - Loopback60 - 6.0.0.6/24
  - Loopback61 - 6.0.1.6/24
- Configure summarization on R4 and R6 to advertise a default route into EIGRP instead of the redistributed Loopbacks.
- Configure a leak-map on R4 so that traffic going to R4's Loopback40 prefix is routed out the DMVPN cloud.
  - If this link is down, traffic should still be rerouted out the Ethernet link between between these devices.
- Configure a leak-map on R6 so that traffic from R1 going to R6's Loopback60 prefix is routed out the Ethernet link between R1 and R3.

https://t.me/learningnets

- If this link is down, traffic should still be rerouted out the Ethernet link between R1 and R6.

Configuration

```
R4, R5:
router eigrp 100
 network 155.1.0.0 0.0.0.255
 network 155.1.45.0 0.0.0.255

R1, R3, R6, R7:
 router eigrp MULTI-AF
 !
  address-family ipv4 unicast autonomous-system 200
   !
   topology base
   exit-af-topology
   network 155.1.0.0

R4:
interface loopback 40
 ip address 4.0.0.4 255.255.255.0
!
interface loopback 41
 ip address 4.0.1.4 255.255.255.0
!
ip prefix-list CONNECTED_TO_EIGRP seq 5 permit 4.0.0.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 10 permit 4.0.1.0/24
!
route-map CONNECTED_TO_EIGRP permit 10
 match ip address prefix-list CONNECTED_TO_EIGRP
!
ip prefix-list LOOPBACK40 seq 5 permit 4.0.0.0/24
!
route-map LEAK_LOOPBACK40 permit 10
 match ip address prefix-list LOOPBACK40
!
router eigrp 100
 redistribute connected route-map CONNECTED_TO_EIGRP
!
interface Tunnel0
 ip summary-address eigrp 100 0.0.0.0 0.0.0.0 leak-map LEAK_LOOPBACK40
!
interface GigabitEthernet1.45
```

https://t.me/learningnets

```
 ip summary-address eigrp 100 0.0.0.0 0.0.0.0
```

R6:

```
interface Loopback 60
 ip address 6.0.0.6 255.255.255.0
!
interface Loopback 61
 ip address 6.0.1.6 255.255.255.0
!
ip prefix-list CONNECTED_TO_EIGRP seq 5 permit 6.0.0.0/24
ip prefix-list CONNECTED_TO_EIGRP seq 10 permit 6.0.1.0/24
!
route-map CONNECTED_TO_EIGRP permit 10
 match ip address prefix-list CONNECTED_TO_EIGRP
!
ip prefix-list LOOPBACK60 seq 5 permit 6.0.0.0/24
!
route-map LEAK_LOOPBACK60 permit 10
 match ip address prefix-list LOOPBACK60
!
router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 200
 !
 af-interface GigabitEthernet1.67
  summary-address 0.0.0.0 0.0.0.0 leak-map LEAK_LOOPBACK60
 exit-af-interface
 !
 af-interface GigabitEthernet1.146
  summary-address 0.0.0.0 0.0.0.0
 exit-af-interface
 !
  topology base
   redistribute connected route-map CONNECTED_TO_EIGRP
  exit-af-topology
```

Verification

The EIGRP **`leak-map`** feature of the **`summary-address`** allows the advertisement of specific subnets encompassed by the interface-level summary, similar to the **`unsuppress-map`** feature of BGP aggregation. Routes matched in the leak-map route-

https://t.me/learningnets

map will be advertised in addition to the summary. If the route-map matches all routes, all subnets of the aggregate will be advertised in addition to the aggregate. This is useful in cases where you want to originate a default route with the interface summary-address, but you don't want to stop the advertisement of specific subnets.

In this particular design, the leak-map is used to enforce longest match routing traffic engineering. Because R5 has a longer match for the prefix 4.0.0.0/24 via the DMVPN cloud, traffic for this prefix will never get routed over VLAN 45 unless the Ethernet link is down. Verify that traffic destined to Loopback40 is routed out the DMVPN cloud, while traffic destined for Loopback41 is routed out the Ethernet link:

```
R5#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.45.4 to network 0.0.0.0

D*    0.0.0.0/0 [90/3072] via 155.1.45.4, 00:00:12, GigabitEthernet1.45
      4.0.0.0/24 is subnetted, 1 subnets
D EX     4.0.0.0 [170/25984000] via 155.1.0.4, 00:00:12, Tunnel0
!R5#traceroute 4.0.0.4
Type escape sequence to abort.
Tracing the route to 4.0.0.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.4 2 msec *  4 msec
!R5#traceroute 4.0.1.4
Type escape sequence to abort.
Tracing the route to 4.0.1.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 4 msec *  3 msec
```

Verify that leak-map is configured only on the DMVPN cloud, thus there is a single entry in the EIGRP topology on R5:

```
R5#show ip eigrp topology 4.0.0.0 255.255.255.0
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.5.5) for 4.0.0.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 130816
  Descriptor Blocks: 155.1.0.4 (Tunnel0), from 155.1.0.4, Send flag is 0x0

      Composite metric is (25984000/128256), route is External
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 1
        Originating router is 150.1.4.4
      External data:
```

https://t.me/learningnets

```
        AS number of route is 0
        External protocol is Connected, external metric is 0
        Administrator tag is 0 (0x00000000)
```

Disable R5's DMVPN interface to R4 and verify that all traffic is now routed over the Ethernet link, based on the default route entry:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface Tunnel0
R5(config-if)#shutdown
!R5#traceroute 4.0.0.4
Type escape sequence to abort.
Tracing the route to 4.0.0.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 3 msec *  2 msec
!R5#traceroute 4.0.1.4
Type escape sequence to abort.
Tracing the route to 4.0.1.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 3 msec *  2 msec
```

Because R1 has a longer match for the prefix 6.0.0.0/24 via its Ethernet link to R3, traffic for this prefix will never get routed over VLAN 146 unless the Ethernet link to R3 is down. Verify that traffic destined to Loopback60 is routed out VLAN 13, while traffic destined for Loopback61 is routed out VLAN 146:

```
R1#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.146.6 to network 0.0.0.0

D*    0.0.0.0/0 [90/10880] via 155.1.146.6, 00:14:27, GigabitEthernet1.146
      6.0.0.0/24 is subnetted, 1 subnets
D EX     6.0.0.0 [170/21120] via 155.1.13.3, 00:14:27, GigabitEthernet1.13
      155.1.0.0/16 is variably subnetted, 10 subnets, 2 masks
D        155.1.7.0/24 [90/20480] via 155.1.13.3, 00:14:28, GigabitEthernet1.13
D        155.1.37.0/24 [90/15360] via 155.1.13.3, 00:14:28, GigabitEthernet1.13
D        155.1.67.0/24 [90/20480] via 155.1.13.3, 00:14:28, GigabitEthernet1.13
D        155.1.79.0/24 [90/20480] via 155.1.13.3, 00:14:28, GigabitEthernet1.13
!R1#traceroute 6.0.0.6
Type escape sequence to abort.
Tracing the route to 6.0.0.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.13.3 3 msec 2 msec 0 msec
  2 155.1.37.7 1 msec 1 msec 1 msec
  3 155.1.67.6 3 msec *  4 msec
!R1#traceroute 6.0.1.6
Type escape sequence to abort.
Tracing the route to 6.0.1.6
```

https://t.me/learningnets

```
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.6 4 msec *  1 msec
```

Verify that leak-map is configured only on VLAN 67, thus there is a single entry in the EIGRP topology on R1:

```
R1#show eigrp address-family ipv4 topology 6.0.0.0 255.255.255.0
EIGRP-IPv4 VR(MULTI-AF) Topology Entry for AS(200)/ID(150.1.1.1) for 6.0.0.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2703360, RIB is 21120
  Descriptor Blocks: 155.1.13.3 (GigabitEthernet1.13), from 155.1.13.3, Send flag is 0x0

      Composite metric is (2703360/2048000), route is External
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 31250000 picoseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.6.6
      External data:
        AS number of route is 0
        External protocol is Connected, external metric is 0
        Administrator tag is 0 (0x00000000)
```

Disable R1's VLAN 13 interface to R3 and verify that all traffic is now routed over VLAN 146, based on the default route entry:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface GigabitEthernet1.13
R1(config-subif)#shutdown
!R1#traceroute 6.0.0.6
Type escape sequence to abort.
Tracing the route to 6.0.0.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.6 5 msec *  2 msec
!R1#traceroute 6.0.1.6
Type escape sequence to abort.
Tracing the route to 6.0.1.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.6 11 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Floating Summarization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on R4, R5, and R8 using AS 100.
  - Enable EIGRP only over VLAN 45 and VLAN 58 Ethernet segments.
- Configure Loopback1 on R4 and R5 with IPv4 addresses of 160.1.Y.Y/24 , where *Y* is the router number.
  - Advertise these prefixes into EIGRP.
- Configure summarization on R4 to advertise a default route into EIGRP.
- Configure R5 to summarize Loopback1 prefixes of R4 and R5 out to R8; this route should not overlap any additional networks.
- Configure an equal longest match static route on R5 so that R8 has reachability to both Loopback1 prefixes of R4 and R5.

Configuration

```
R4:
interface Loopback1
 ip address 160.1.4.4 255.255.255.0
!
router eigrp 100
 network 155.1.45.0 0.0.0.255
 network 160.1.4.0 0.0.0.255
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.45
 ip summary-address eigrp 100 0.0.0.0 0.0.0.0
R5:
interface Loopback1
 ip address 160.1.5.5 255.255.255.0
!
router eigrp 100
 network 155.1.45.0 0.0.0.255
 network 155.1.58.0 0.0.0.255
 network 160.1.5.0 0.0.0.255
!
interface GigabitEthernet1.58
 ip summary-address eigrp 100 160.1.4.0 255.255.254.0
!
ip route 160.1.4.0 255.255.255.0 155.1.45.4
R8:

router eigrp 100
 network 155.1.58.0 0.0.0.255
```

Verification

When summaries are created in EIGRP, OSPF, and BGP, the router automatically installs a route to Null0 to match the summary. This is used to prevent the router from forwarding traffic for destinations inside the summary that it does not have a longer match for. However, in certain designs this can be an undesirable behavior. To resolve this, EIGRP sets its interface-level summaries to have an administrative distance of 5 by default. This means that any other route with a distance of 1–4 will take precedence over the summary.

In this particular case, before summarization and static routing is configured on R5, R8 has the subnet route 160.1.5.0/24 and a default route to reach 160.1.4.0/24. This is because R4 is generating a default route and suppressing its subnet advertisements.

```
R8#show ip route 160.1.4.4
 % Subnet not in table
!R8#show ip route 160.1.5.5
Routing entry for 160.1.5.0/24
  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:00:12 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 155.1.58.5, 00:00:12 ago, via GigabitEthernet1.58
```

https://t.me/learningnets

```
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R8#show ip cef 160.1.4.4
0.0.0.0/0
  nexthop 155.1.58.5 GigabitEthernet1.58
!R8#traceroute 160.1.4.4
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 139 msec 3 msec 5 msec 2 155.1.45.4 5 msec *  3 msec
```

Likewise, R5 only has a default route to 160.1.4.4, whereas 160.1.5.5 is directly connected.

```
R5#show ip route 160.1.4.4
% Subnet not in table
!R5#show ip route 160.1.5.5
Routing entry for 160.1.5.5/32
  Known via "connected", distance 0, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Loopback1
      Route metric is 0, traffic share count is 1
!R5#show ip cef 160.1.4.4
0.0.0.0/0
  nexthop 155.1.45.4 GigabitEthernet1.45
!R5#traceroute 160.1.4.4
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 8 msec *  2 msec
```

When R5 advertises the summary 160.1.4.0/23, R8 loses its more specific route to 160.1.5.0/24 but gains a longer match for 160.1.4.0/24.

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)# interface GigabitEthernet1.58
R5(config-if)#ip summary-address eigrp 100 160.1.4.0 255.255.254.0

!R8#show ip route 160.1.4.4
Routing entry for 160.1.4.0/23
  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:00:43 ago
```

https://t.me/learningnets

```
  Routing Descriptor Blocks: * 155.1.58.5, from 155.1.58.5, 00:00:43 ago, via GigabitEthernet1.58
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R8#show ip route 160.1.5.5
Routing entry for 160.1.4.0/23
  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:00:56 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 155.1.58.5, 00:00:56 ago, via GigabitEthernet1.58

      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

Because R5 previously only had a default route to reach 160.1.4.0/24, the longer match is now the summary to Null0, while the longer match for 160.1.5.0/24 remains the connected interface:

```
R5#show ip route 160.1.4.4
Routing entry for 160.1.4.0/23
  Known via "eigrp 100", distance 5, metric 128256, type internal
  Redistributing via eigrp 100
  Routing Descriptor Blocks: * directly connected, via Null0
      Route metric is 128256, traffic share count is 1
      Total delay is 5000 microseconds, minimum bandwidth is 8000000 Kbit
      Reliability 255/255, minimum MTU 1514 bytes
      Loading 1/255, Hops 0
!R5#show ip route 160.1.5.5
Routing entry for 160.1.5.5/32
  Known via "connected", distance 0, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Loopback1

      Route metric is 0, traffic share count is 1
```

This implies that R5 can forward traffic for 160.1.5.0/24, but traffic for 160.1.4.0/24 will be Null routed (dropped):

```
R8#traceroute 160.1.5.5
Type escape sequence to abort.
Tracing the route to 160.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.58.5 13 msec *  2 msec
```

https://t.me/learningnets

```
!R8#traceroute 160.1.4.4

Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 6 msec 10 msec 1 msec 2 155.1.58.5 !H  *  !H
```

To resolve this, a static route with a lower administrative distance than the summary is installed in the routing table of R5, which tells R5 to forward traffic that matches the summary toward R4:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#ip route 160.1.4.0 255.255.255.0 155.1.45.4
!R5#show ip route 160.1.4.4
Routing entry for 160.1.4.0/24
  Known via "static", distance 1, metric 0
  Routing Descriptor Blocks: * 155.1.45.4
      Route metric is 0, traffic share count is 1
!R8#traceroute 160.1.4.4
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 14 msec 2 msec 3 msec 2 155.1.45.4 14 msec *  6 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Poisoned Floating Summarization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on R4, R5, and R8 using AS 100.
  - Enable EIGRP only over VLAN 45 and VLAN 58 Ethernet segments.
- Configure Loopback1 on R4 and R5 with IPv4 addresses of 160.1.Y.Y/24, where *Y* is the router number.
  - Advertise these prefixes into EIGRP.
- Configure summarization on R4 to advertise a default route into EIGRP.
- Configure R5 to summarize Loopback1 prefixes of R4 and R5 out to R8; this route should not overlap any additional networks.
  - Modify the administrative distance of the summary that R5 is generating to R8 so that a route to Null0 is not installed.

Configuration

```
R4:
interface Loopback1
 ip address 160.1.4.4 255.255.255.0
!
router eigrp 100
 network 155.1.45.0 0.0.0.255
 network 160.1.4.0 0.0.0.255
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.45
 ip summary-address eigrp 100 0.0.0.0 0.0.0.0
```

R5:

```
interface Loopback1
 ip address 160.1.5.5 255.255.255.0
!
router eigrp 100
 network 155.1.45.0 0.0.0.255
 network 155.1.58.0 0.0.0.255
 network 160.1.5.0 0.0.0.255
 summary-metric 160.1.4.0/23 distance 255
!
interface GigabitEthernet1.58
 ip summary-address eigrp 100 160.1.4.0 255.255.254.0
```

R8:

```
router eigrp 100
 network 155.1.58.0 0.0.0.255
```

Verification

Routes with an administrative distance of 255 are not candidates to be installed in the routing table. By poisoning the interface-level summary on R5 with a distance of 255, the route to Null0 cannot be installed locally in the routing table, but the summary itself can be advertised out the interface. In previous IOS codes (before 15.x), the distance was configured along with the interface-level summary; in newer IOS codes it is globally configured at the process level using the command **`summary-metric <prefix> distance <value>`** . From a design perspective, this configuration is for cases in which you want the router to forward traffic for destinations inside the summary that it does not have a longer match for. In previous IOS codes (before 15.x), the router performing the summarization would still advertise the summary in its EIGRP updates, even though AD of 255 was configured which prohibited the router to install it in the routing table. In newer IOS codes, the summary is no longer advertised if poisoned with AD of 255, but it is active because all routes comprised within the summary are no longer advertised out on the interface the summary is configured on.

Note the routing table of R8, before summary is configured on R5; R8 has a specific route for R5's Loopback1 and matches the default route for reaching R4's Loopback1.

https://t.me/learningnets

```
R8#show ip route eigrp | b Gateway

Gateway of last resort is 155.1.58.5 to network 0.0.0.0
D*    0.0.0.0/0 [90/3328] via 155.1.58.5, 00:19:31, GigabitEthernet1.58
      155.1.0.0/16 is variably subnetted, 7 subnets, 2 masks
D        155.1.45.0/24 [90/3072] via 155.1.58.5, 00:19:31, GigabitEthernet1.58
      160.1.0.0/24 is subnetted, 1 subnets
D        160.1.5.0 [90/130816] via 155.1.58.5, 00:00:32, GigabitEthernet1.58
!R8#traceroute 160.1.5.5
Type escape sequence to abort.
Tracing the route to 160.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.58.5 26 msec *  2 msec
!R8#traceroute 160.1.4.4
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 20 msec 59 msec 2 msec 2 155.1.45.4 28 msec *  3 msec
```

When the summary is configured on R5, it will install the Null0 route, which prevents both R5 and R8 from reaching R4's Loopback1.

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface GigabitEthernet1.58
R5(config-subif)#ip summary-address eigrp 100 160.1.4.0 255.255.254.0
!R8#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.58.5 to network 0.0.0.0
D*    0.0.0.0/0 [90/3328] via 155.1.58.5, 00:23:16, GigabitEthernet1.58
      155.1.0.0/16 is variably subnetted, 7 subnets, 2 masks
D        155.1.45.0/24 [90/3072] via 155.1.58.5, 00:23:16, GigabitEthernet1.58
      160.1.0.0/23 is subnetted, 1 subnets
D        160.1.4.0 [90/130816] via 155.1.58.5, 00:00:22, GigabitEthernet1.58
!R5#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.45.4 to network 0.0.0.0

D*    0.0.0.0/0 [90/3072] via 155.1.45.4, 00:23:37, GigabitEthernet1.45
      160.1.0.0/16 is variably subnetted, 3 subnets, 3 masks
D        160.1.4.0/23 is a summary, 00:00:42, Null0
!R5#traceroute 160.1.4.4 ttl 1 2
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1  *  *  *  2  *  *  *
!R8#traceroute 160.1.4.4
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 13 msec 2 msec 2 msec 2 155.1.58.5 !H  *  !H
```

When the AD of 255 is configured for the summary on R5, this removes the Null0 route and suppresses advertisements for Loopback1 prefixes, and R8 will match the default route for both Loopback1 prefixes.

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router eigrp 100
R5(config-router)#summary-metric 160.1.4.0/23 distance 255
!R5#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.45.4 to network 0.0.0.0
D*    0.0.0.0/0 [90/3072] via 155.1.45.4, 00:27:26, GigabitEthernet1.45
!R8#show ip route eigrp | b Gateway
Gateway of last resort is 155.1.58.5 to network 0.0.0.0
D*    0.0.0.0/0 [90/3328] via 155.1.58.5, 00:27:39, GigabitEthernet1.58
      155.1.0.0/16 is variably subnetted, 7 subnets, 2 masks
D        155.1.45.0/24 [90/3072] via 155.1.58.5, 00:27:39, GigabitEthernet1.58
!R5#traceroute 160.1.4.4
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 3 msec *  4 msec
!R8#traceroute 160.1.4.4
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 26 msec 4 msec 2 msec 2 155.1.45.4 4 msec *  5 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Metric Weights

> You must load the initial configuration files for the section, **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode between R5 and R8 using AS 100, and advertise their Loopback0 prefixes.
- Configure an EIGRP process named **MULTI-AF** between R7 and R9 using AS 200, and advertise their Loopback0 prefixes.
- Configure all EIGRP routers so that only delay is used in the composite metric calculation.

Configuration

```
R5, R8:
router eigrp 100
 network 155.1.58.0 0.0.0.255
 network 150.1.0.0 0.0.255.255
 metric weights 0 0 0 1 0 0

R7, R9:

router eigrp MULTI-AF
 !
 address-family ipv4 unicast autonomous-system 200
  !
  topology base
  exit-af-topology
```

https://t.me/learningnets

```
  network 155.1.79.0 0.0.0.255
  network 150.1.0.0 0.0.255.255
  metric weights 0 0 0 1 0 0 0
 exit-address-family
```

Verification

By default, EIGRP uses only bandwidth and delay to calculate its composite metric, as K1=K3=1 and K2=K4=K5=K6=0. Load, reliability, and extended attributes can also be used, or the ratio at which bandwidth and delay are used can be changed, by modifying the `metric weights` . Specifically, the calculation is as follows for Classic EIGRP, which uses a 32-bit metric:

```
Metric = 256*[(K1*Scaled Bw) + (K2*Scaled Bw)/(256 - Load) + (K3*Scaled Delay)]*[K5/(Reliability + K4)]
```

The calculation is as follows for EIGRP Named mode, which uses a 64-bit metric:

```
Metric = [(K1*Minimum Throughput + (K2*Minimum Throughput/(256-Load) + (K3*Total Latency) + (K6*Extended Attributes)
```

If K5 equals zero, the second half of the equation is ignored in both cases. "Scaled Bw" equals 10^7/(Minimum Bw/Kbps) and "Scaled Delay" equals (Delay/10) in microseconds. "Minimum Throughput" equals (10^7 * 65535)/(Minimum Bw/Kbps), "Total Latency" equals (Delay * 65536)/10 in microseconds for links below 1 GigabitEthernet and (10^7 * 65536/10)/Bw in microseconds for links above 1 GigabitEthernet. The weighting of the metrics can be seen from the `show ip protocols` command:

```
R8#show ip protocols | section eigrp
Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 Protocol for AS(100) Metric weight K1=0, K2=0, K3=1, K4=0, K5=0
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 150.1.8.8
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4
```

https://t.me/learningnets

```
      Maximum hopcount 100
      Maximum metric variance 1
!R9#show ip protocols | section eigrp
Routing Protocol is "eigrp 200"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 VR(MULTI-AF) Address-Family Protocol for AS(200)
Metric weight K1=0, K2=0, K3=1, K4=0, K5=0 K6=0

    Metric rib-scale 128
    Metric version 64bit
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 150.1.9.9
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4
      Maximum hopcount 100
      Maximum metric variance 1
      Total Prefix Count: 3
      Total Redist Count: 0
```

The commands **`show ip eigrp topology`** and **`show eigrp address-family ipv4 topology`** show the individual vector metrics that are used in the composite calculation.

```
R8#show ip eigrp topology 150.1.8.8 255.255.255.255
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.8.8) for 150.1.8.8/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 128000
  Descriptor Blocks:
  0.0.0.0 (Loopback0), from Connected, Send flag is 0x0
Composite metric is (128000/0), route is Internal
      Vector metric: Minimum bandwidth is 8000000 Kbit
 Total delay is 5000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1514
        Hop count is 0
        Originating router is 150.1.8.8
!R9#show eigrp address-family ipv4 topology 150.1.9.9 255.255.255.255
```

https://t.me/learningnets

```
EIGRP-IPv4 VR(MULTI-AF) Topology Entry for AS(200)/ID(150.1.9.9) for 150.1.9.9/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 81920
  Descriptor Blocks:
  0.0.0.0 (Loopback0), from Connected, Send flag is 0x0 Composite metric is (81920/0), route is Internal
      Vector metric: Minimum bandwidth is 8000000 Kbit
 Total delay is 1250000 picoseconds

        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1514
        Hop count is 0
        Originating router is 150.1.9.9
```

For the connected Loopback0 prefix of R8, the total delay is 5000 microseconds, which means Scaled Delay is 500 microseconds and scaled by 256 equals the total composite metric of 128.000. For the connected Loopback0 of R9, Bw is 8000000Kbps, which means Total Latency is [10^7*65536/10]/8000000=81.920 and equals the composite metric. This indicates that only delay is weighted in the calculation.

Pitfall

The metric weights must match for EIGRP adjacency to form:

```
R8#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R8(config)#router eigrp 100
R8(config-router)#metric weights 0 1 1 1 1 1
!%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.58.5 (GigabitEthernet1.58) is down: K-value mismatch
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.58.5 (GigabitEthernet1.58) is down: K-value mismatch
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Traffic Engineering with Metric

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure a metric manipulation on R7 so that traffic from R9 to R6's Loopback0 prefix transits the link between R3 and R7.

Configuration

```
R7:

interface GigabitEthernet1.67
 delay 100000
```

Verification

Before any metric manipulation, R9's traffic to R6 is sent to R7, then directly to R6:

```
R9#traceroute 150.1.6.6
Type escape sequence to abort.
Tracing the route to 150.1.6.6
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.79.7 3 msec 1 msec 1 msec 2 155.1.67.6 2 msec *  5 msec
!R7#show ip route 150.1.6.6
Routing entry for 150.1.6.6/32
  Known via "eigrp 100", distance 90, metric 130816, type internal
```

https://t.me/learningnets

```
  Redistributing via eigrp 100
  Last update from 155.1.67.6 on GigabitEthernet1.67, 00:01:54 ago
  Routing Descriptor Blocks: * 155.1.67.6, from 155.1.67.6, 00:01:54 ago, via GigabitEthernet1.67

      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

Also note that R7 does not know the alternate route through R3 for R6's Loopback0:

```
R7#show ip eigrp topology 150.1.6.6 255.255.255.255
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.7.7) for 150.1.6.6/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 130816
  Descriptor Blocks: 155.1.67.6 (GigabitEthernet1.67), from 155.1.67.6, Send flag is 0x0

      Composite metric is (130816/128256), route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5010 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.6.6
```

For R3 to advertise the alternate path to R7, R3 must see a better composite metric through R1 than it does through R7. This can be accomplished by altering the advertised distance of the route from R7 to R3 by changing the delay:

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.67
R7(config-if)#delay 100000
```

It may be required to clear EIGRP neighbors for DUAL to be recalculated. R7 now chooses R3's route as the successor, because of better metric:

```
R7#show ip eigrp topology 150.1.6.6 255.255.255.255
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.7.7) for 150.1.6.6/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131328
  Descriptor Blocks: 155.1.37.3 (GigabitEthernet1.37), from 155.1.37.3, Send flag is 0x0
      Composite metric is (131328/131072)
, route is Internal
```

https://t.me/learningnets

```
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.6.6
155.1.67.6 (GigabitEthernet1.67), from 155.1.67.6, Send flag is 0x0

      Composite metric is (25730560/128256), route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 1005000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.6.6
```

R7 now routes through R3 to reach 150.1.6.6, which is reflected in both the routing table output of R7 and the traceroute output of R9:

```
R7#show ip route 150.1.6.6
Routing entry for 150.1.6.6/32
  Known via "eigrp 100", distance 90, metric 131328, type internal
  Redistributing via eigrp 100
  Last update from 155.1.37.3 on GigabitEthernet1.37, 00:02:56 ago
  Routing Descriptor Blocks: * 155.1.37.3, from 155.1.37.3, 00:02:56 ago, via GigabitEthernet1.37
      Route metric is 131328, traffic share count is 1
      Total delay is 5030 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3
!R9#traceroute 150.1.6.6
Type escape sequence to abort.
Tracing the route to 150.1.6.6
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.79.7 3 msec 1 msec 1 msec 2 155.1.37.3 1 msec 1 msec 1 msec

  3 155.1.13.1 15 msec 10 msec 7 msec
  4 155.1.146.6 47 msec *  67 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Unequal Cost Load Balancing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Note that R4's link to VLAN 146 and the link between R2 and R3 are disabled. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure EIGRP in Classic Mode on all routers using AS 100, and advertise their Loopback0 prefixes.
  - Ensure that only delay is used in the composite metric calculation.
- Configure unequal cost load balancing so that traffic from R6 going to VLAN 9 is load balanced between R1 and R7.
  - Traffic share should be configured so that the link to R7 is used five times as much as the link to R1.
  - Modify delay on R6 as required.

Configuration

```
R1 - R10:
router eigrp 100
 network 155.1.0.0 0.0.255.255
 network 150.1.0.0 0.0.255.255
 metric weights 0 0 0 1 0 0

R6:

interface GigabitEthernet1.67
 delay 25
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.146
 delay 131
!
router eigrp 100
 variance 5
```

Verification

The `metric weights` command is configured on all routers in EIGRP AS 100 so that only delay is weighted, used for metric calculation. Therefore, based on the interface delay values from R6 outbound toward VLAN 9, we can calculate how traffic will be routed. Recall that the delay value used in the composite calculation is *tens of microseconds scaled by 256*. To start, without any configuration changes on delta values, the path from R6 to R9 has the following delays:

```
R6#show interface GigabitEthernet1.67 | include DLY
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
,
!R7#show interface GigabitEthernet1.79 | include DLY
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
,
!R9#show interface GigabitEthernet1.9 | include DLY
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
,
```

The path from R6 -> R7 -> R9 therefore has a total delay of 30 microseconds. 3 *tens of microseconds* scaled by 256 gives us a composite metric of 768. This path is then compared to the one R6 -> R1 -> R3 -> R7 -> R9 with the following delays:

```
R6#show interface GigabitEthernet1.146 | include DLY
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec
,
!R1#show interface GigabitEthernet1.13 | include DLY
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec
,
!R3#show interface GigabitEthernet1.37 | include DLY
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec
,
!R7#show interface GigabitEthernet1.79 | include DLY
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
,
!R9#show interface GigabitEthernet1.9 | include DLY
```

https://t.me/learningnets

```
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
,
```

This path has a total delay of 50 microseconds. 5 *tens of microseconds* scaled by 256 gives us a composite metric of 1280. Because 768 is lower than 1280, the Successor is the route from R6 to R7. This can be verified from the EIGRP topology of R6:

```
R6#show ip eigrp topology 155.1.9.0 255.255.255.0
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.6.6) for 155.1.9.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 768
  Descriptor Blocks: 155.1.67.7 (GigabitEthernet1.67), from 155.1.67.7, Send flag is 0x0
      Composite metric is (768/512)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit Total delay is 30 microseconds

        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 2
        Originating router is 150.1.9.9
```

To consider the route from R6 to R1 for load balancing, the route first must pass the Feasibility Condition. Again the Feasibility Condition states that if the Advertised Distance of an alternate path is lower than the Feasible Distance of the Successor, the route is a loop-free path and can be considered for load balancing. In other words, if R1's metric to reach R9 is lower than R6's metric to reach R9, R6 can assume that the path through R1 is a loop-free path. The Advertised Distance that R1 would be sending to R6 is based on these interfaces in the transit path:

https://t.me/learningnets

```
R1#show interface GigabitEthernet1.13 | include DLY
   MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec
,
!R3#show interface GigabitEthernet1.37 | include DLY
   MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec
,
!R7#show interface GigabitEthernet1.79 | include DLY
   MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
,
!R9#show interface GigabitEthernet1.9 | include DLY
   MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
,
```

The total delay of this path is 40 microseconds, or 4 *tens of microseconds*. Scaled by 256, R1 would be advertising 1024. Because R3's Feasible Distance of 1024 is equal to R6's Feasible Distance, this path cannot be considered a Feasible Successor. Verify that R1 performs equal cost load-balancing for VLAN 9 prefix, as the paths through R3 and R6 have the same metric.

```
R1#show ip eigrp topology 155.1.9.0/24
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.1.1) for 155.1.9.0/24
  State is Passive, Query origin flag is 1, 2 Successor(s), FD is 1024
  Descriptor Blocks: 155.1.13.3 (GigabitEthernet1.13), from 155.1.13.3, Send flag is 0x0
      Composite metric is (1024/768)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 40 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.9.9
155.1.146.6 (GigabitEthernet1.146), from 155.1.146.6, Send flag is 0x0
      Composite metric is (1024/768)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 40 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.9.9
```

https://t.me/learningnets

```
  155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0
      Composite metric is (1281280/1280), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 50050 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 5
        Originating router is 150.1.9.9
!R1#show ip route 155.1.9.0
Routing entry for 155.1.9.0/24  Known via "eigrp 100", distance 90, metric 1024
, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.6 on GigabitEthernet1.146, 00:20:17 ago
  Routing Descriptor Blocks: 155.1.146.6, from 155.1.146.6, 00:20:17 ago, via GigabitEthernet1.146
      Route metric is 1024, traffic share count is 1
      Total delay is 40 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3 * 155.1.13.3, from 155.1.13.3, 00:20:17 ago, via GigabitEthernet1.13

      Route metric is 1024, traffic share count is 1
      Total delay is 40 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3
```

Because the minimum configurable delay value is 10 microseconds, which is already the default for all Ethernet links, and based on task requirements, we need to modify R6's delay values on its VLAN 67 and VLAN 146 interfaces, so that metric through R1 is five times bigger than metric through R7.

```
5 * [Delay(Gi1.9) + Delay(Gi1.79) + Delay(Gi1.67)] = [Delay(Gi1.9) + Delay(Gi1.79) + Delay(Gi1.37) + Delay(Gi1.13) +
5 * [10 + 10 +Delay(Gi1.67)] = [10 + 10 + 10 + 10 + Delay(Gi1.146)]].
```

If, for example, we configure delay on R6's VLAN 67 interface to be 250, in simple math we need to configure a delay value of 1310 on R6's VLAN 146 interface. This also means that configuring a variance of 5 will be enough so that both routes for VLAN 9 are installed in the routing table of R6 with the requested load-distribution.

```
R6#show ip eigrp topology 155.1.9.0/24
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.6.6) for 155.1.9.0/24
  State is Passive, Query origin flag is 1, 2 Successor(s), FD is 1280
  Descriptor Blocks: 155.1.67.7 (GigabitEthernet1.67), from 155.1.67.7, Send flag is 0x0
```

https://t.me/learningnets

```
      Composite metric is (6912/512)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 270 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 2
        Originating router is 150.1.9.9
155.1.146.1 (GigabitEthernet1.146), from 155.1.146.1, Send flag is 0x0
      Composite metric is (34560/1024)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 1350 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 4
        Originating router is 150.1.9.9
!R6#show ip route 155.1.9.0
Routing entry for 155.1.9.0/24
  Known via "eigrp 100", distance 90, metric 6912, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.1 on GigabitEthernet1.146, 00:01:40 ago
  Routing Descriptor Blocks: 155.1.146.1, from 155.1.146.1, 00:01:40 ago, via GigabitEthernet1.146
      Route metric is 34560, traffic share count is 1
      Total delay is 1350 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 4 * 155.1.67.7, from 155.1.67.7, 00:01:40 ago, via GigabitEthernet1.67
      Route metric is 6912, traffic share count is 5

      Total delay is 270 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
```

Verify the configured delay and variance values.

```
R6# show ip protocols | section eigrp
Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
```

https://t.me/learningnets

```
  EIGRP-IPv4 Protocol for AS(100)
    Metric weight K1=0, K2=0, K3=1, K4=0, K5=0
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 150.1.6.6
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4
      Maximum hopcount 100 Maximum metric variance 5
!R6#show interfaces gigabitEthernet1.67 | i DLY
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 250 usec,
!R6#show interfaces gigabitEthernet1.146 | i DLY
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 1310 usec,
```

As CEF cannot be disabled on CSR 1000V, so per-packet load balancing is not configurable. To verify that traffic is load balanced as requested, configure ACL matching on traffic toward VLAN 9 on the devices in transit path and generate packets to various hosts in VLAN 9. For example, ping all hosts from 155.1.9.1 to 155.1.9.12 with a single packet. First configure and apply ACL's on R7 and R1.

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R7(config)#ip access-list extended EIGRP
R7(config-ext-nacl)#permit icmp any 155.1.9.0 0.0.0.255
R7(config-ext-nacl)#permit ip any any
R7(config-ext-nacl)#interface GigabitEthernet1.67
R7(config-subif)#ip access-group EIGRP in


R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip access-list extended EIGRP
R1(config-ext-nacl)#permit icmp any 155.1.9.0 0.0.0.255
R1(config-ext-nacl)#permit ip any any
R1(config-ext-nacl)#interface GigabitEthernet1.146
R1(config-subif)#ip access-group EIGRP in
```

https://t.me/learningnets

Now generate ICMP packets for the first 12 hosts in VLAN 9 and verify the ACL counters.

```
R6#show ip cef 155.1.9.0/24 internal
155.1.9.0/24, epoch 2, RIB[I], refcount 6, per-destination sharing
  sources: RIB
  feature space:
   IPRM: 0x00028000
   Broker: linked, distributed at 4th priority
  ifnums: GigabitEthernet1.67(10): 155.1.67.7
 GigabitEthernet1.146(11): 155.1.146.1
  path 7F6115C869E8, path list 7F6115DA89B0, share 1/1, type attached nexthop, for IPv4
  nexthop 155.1.146.1 GigabitEthernet1.146, adjacency IP adj out of GigabitEthernet1.146, addr 155.1.146.1 7F6115EDA
  path 7F6115C87A68, path list 7F6115DA89B0, share 5/5, type attached nexthop, for IPv4
  nexthop 155.1.67.7 GigabitEthernet1.67, adjacency IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
  output chain:
    loadinfo 7F610E19CE30, per-session, 2 choices, flags 0003, 5 locks
    flags: Per-session, for-rx-IPv4 12 hash buckets
      < 0 > IP adj out of GigabitEthernet1.146, addr 155.1.146.1 7F6115EDA420
      < 1 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      < 2 > IP adj out of GigabitEthernet1.146, addr 155.1.146.1 7F6115EDA420
      < 3 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      < 4 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      < 5 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      < 6 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      < 7 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      < 8 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      < 9 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      <10 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
      <11 > IP adj out of GigabitEthernet1.67, addr 155.1.67.7 7F6115EAECF8
    Subblocks:
     None
!R7#show ip access-lists
Extended IP access list EIGRP    10 permit icmp any 155.1.9.0 0.0.0.255 (10 matches)
    20 permit ip any any (21 matches)
!R1#show ip access-lists
Extended IP access list EIGRP    10 permit icmp any 155.1.9.0 0.0.0.255 (2 matches)

    20 permit ip any any (44 matches)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Convergence Timers

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 through R6 so that EIGRP hellos are sent every one second.
  - These devices should inform their neighbors to declare them down if subsequent hellos are not received within three seconds.
- Configure R7 through R10 so that EIGRP hellos are sent every ten seconds.
  - These devices should inform their neighbors to declare them down if subsequent hellos are not received within thirty seconds.
- Configure AS 100 so that lost routes are considered Stuck In Active if a query response has not been heard within one minute.

Configuration

```
R1:
interface GigabitEthernet1.146
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface Tunnel0
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface GigabitEthernet1.13
 ip hello-interval eigrp 100 1
```

https://t.me/learningnets

```
 ip hold-time eigrp 100 3
!
router eigrp 100
 timers active-time 1
```

R2:

```
interface Tunnel0
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface GigabitEthernet1.23
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
router eigrp 100
 timers active-time 1
```

R3:

```
interface GigabitEthernet1.37
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface Tunnel0
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface GigabitEthernet1.13
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface GigabitEthernet1.23
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
router eigrp 100
 timers active-time 1
```

R4:

```
interface GigabitEthernet1.146
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface Tunnel0
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.45
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
router eigrp 100
 timers active-time 1
```

R5:

```
interface GigabitEthernet1.58
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface Tunnel0
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface GigabitEthernet1.45
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
```

R6:

```
interface GigabitEthernet1.67
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
interface GigabitEthernet1.146
 ip hello-interval eigrp 100 1
 ip hold-time eigrp 100 3
!
router eigrp 100
 timers active-time 1
```

R7:

```
interface GigabitEthernet1.37
 ip hello-interval eigrp 100 10
 ip hold-time eigrp 100 30
!
interface GigabitEthernet1.67
 ip hello-interval eigrp 100 10
 ip hold-time eigrp 100 30
!
interface GigabitEthernet1.79
 ip hello-interval eigrp 100 10
 ip hold-time eigrp 100 30
!
router eigrp 100
```

https://t.me/learningnets

```
 timers active-time 1

R8:
interface GigabitEthernet1.58
 ip hello-interval eigrp 100 10
 ip hold-time eigrp 100 30
!
interface GigabitEthernet1.108
 ip hello-interval eigrp 100 10
 ip hold-time eigrp 100 30
!
router eigrp 100
 timers active-time 1

R9:
interface GigabitEthernet1.79
 ip hello-interval eigrp 100 10
 ip hold-time eigrp 100 30
!
router eigrp 100
 timers active-time 1

R10:

interface GigabitEthernet1.108
 ip hello-interval eigrp 100 10
 ip hold-time eigrp 100 30
!
router eigrp 100
 timers active-time 1
```

Verification

Unlike OSPF, EIGRP hello and hold-time intervals do not need to match to form adjacencies. Just like OSPF for example, the locally configured Hello interval defines the local rate interval for sending EIGRP hello packets but the value is not transmited in EIGRP Hello packets. Unlike OSPF for example, the locally configured Hold-Time interval defines for how long the remote router will wait for a EIGRP packet before resetting the adjacency, thus the value is transmited in EIGRP Hello packets.

In this case, R3 has its hello and dead intervals configured as 1 and 3, whereas R7 has them configured as 10 and 30. This means that R3 will be expecting a hello to

https://t.me/learningnets

come in from R7 within 30 seconds, and R7 will be expecting a hello to come in from R3 within 3 seconds. In most designs, the hello and dead intervals will be set identically on both ends of the link, but as we can see from this example, it is not technically required:

```
R3#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
3   155.1.0.5               Tu02
 01:24:51   33  1398  0  282   155.1.37.7              Gi1.372
 01:24:54    1   100  0  781   155.1.23.2              Gi1.232
 01:25:14    1   100  0  220   155.1.13.1              Gi1.1323
 01:25:14    1   100  0  29
!R7#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
2   155.1.79.9              Gi1.7928
 01:25:11    1   100  0  201   155.1.67.6              Gi1.672
 01:25:21   12   150  0  540   155.1.37.3              Gi1.372
 01:25:21    1   100  0  62
```

The `timers active-time` command controls how long an EIGRP router will wait for a reply to a query message before considering the route Stuck In Active (SIA) and declaring the neighbor from which a reply was not received as down. The query and reply process is used to discover alternate paths to a route for which the successor is lost. In the case below, R10 loses the successor for 155.1.10.0/24 when its VLAN 10 interface is disabled. This causes it to send an EIGRP query message out to its neighbor, R8:

https://t.me/learningnets

```
R9#debug eigrp packets terse

    (UPDATE, REQUEST, QUERY, REPLY, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on
!R10#debug eigrp packets terse

    (UPDATE, REQUEST, QUERY, REPLY, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on
!R10#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#interface gigabitEthernet1.10
R10(config-subif)#shutdown

!EIGRP: Enqueueing QUERY on Gi1.108 - paklen 0 tid 0 iidbQ un/rely 0/1 serno 26-26
EIGRP: Sending QUERY on Gi1.108 - paklen 44 tid 0

  AS 100, Flags 0x0:(NULL), Seq 4/0 interfaceQ 0/0 iidbQ un/rely 0/0 serno 26-26
```

R8 acknowledges the reception of the query with an ACK to R10, and the query continues to be forwarded:

```
R10:
EIGRP: Received ACK on Gi1.108 - paklen 0 nbr 155.1.108.8

  AS 100, Flags 0x0:(NULL), Seq 0/4 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/1
EIGRP: GigabitEthernet1.108 multicast flow blocking cleared
```

Within one second, the query reaches the far end of the network at R9. R9 then acknowledges to R7 that it received the query:

```
R9:
EIGRP: Received QUERY on Gi1.79 - paklen 44 nbr 155.1.79.7
  AS 100, Flags 0x0:(NULL), Seq 80/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Enqueueing ACK on Gi1.79 - paklen 0 nbr 155.1.79.7 tid 0
  Ack seq 80 iidbQ un/rely 0/0 peerQ un/rely 1/0
EIGRP: Sending ACK on Gi1.79 - paklen 0 nbr 155.1.79.7 tid 0

  AS 100, Flags 0x0:(NULL), Seq 0/80 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 1/0
```

Because R9 does not have any other neighbors to send the query to, and it does not have an alternate route to 155.1.10.0/24, it replies to R7 telling it that it does not

https://t.me/learningnets

have another path. R7 then acknowledges the query reply to R9 and sends its own replies back to its other neighbors:

```
R9:
EIGRP: Enqueueing REPLY on Gi1.79 - paklen 0 nbr 155.1.79.7 tid 0 iidbQ un/rely 0/1 peerQ un/rely 0/0 serno 120-120
EIGRP: Requeued unicast on GigabitEthernet1.79
EIGRP: Sending REPLY on Gi1.79 - paklen 44 nbr 155.1.79.7 tid 0
  AS 100, Flags 0x0:(NULL), Seq 24/80 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/1 serno 120-120
EIGRP: Received ACK on Gi1.79 - paklen 0 nbr 155.1.79.7

  AS 100, Flags 0x0:(NULL), Seq 0/24 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/1
```

Within two seconds, the entire query process is completed on R10 with the reply coming back from R8. Amazingly, the query and replies are received even before the link up/down message is generated, indicating the immensely fast convergence capability of EIGRP. If a reply had not come back from R8, R10 would wait for the **`timers active-time`** to expire. If this timer had expired, the route would have been considered SIA, and the neighbor relationship to R8 would have been reset:

```
R10:
EIGRP: Received REPLY on Gi1.108 - paklen 44 nbr 155.1.108.8
  AS 100, Flags 0x0:(NULL), Seq 10/4 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Enqueueing ACK on Gi1.108 - paklen 0 nbr 155.1.108.8 tid 0
  Ack seq 10 iidbQ un/rely 0/0 peerQ un/rely 1/0
EIGRP: Sending ACK on Gi1.108 - paklen 0 nbr 155.1.108.8 tid 0

  AS 100, Flags 0x0:(NULL), Seq 0/10 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 1/0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Stub Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure the EIGRP stub feature so that R8 does not receive EIGRP query messages.
- Ensure that all devices in AS 100 still have IPv4 reachability to VLAN 8.

Configuration

```
R8:

router eigrp 100
 eigrp stub connected
```

Verification

The EIGRP stub feature is used to limit the scope of EIGRP query messages and to further limit which routes a neighbor advertises.

```
R5#show ip eigrp neighbors detail gigabitEthernet1.58
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
2   155.1.58.8              Gi1.58                   13 00:00:20   11   100  0  17
   Version 15.0/2.0, Retrans: 0, Retries: 0, Prefixes: 3
```

https://t.me/learningnets

```
   Topology-ids from peer - 0  Stub Peer Advertising (CONNECTED ) Routes
 Suppressing queries


Max Nbrs: 0, Current Nbrs: 0
```

In this case, R8 is configured to only advertise its connected routes to other EIGRP neighbors, so learned EIGRP routes are not advertised. This implies that R10 will not have reachability to any destinations behind R8, and destinations behind R8 will not have reachability to R10. This is one limitation of EIGRP stub, that the router cannot be a transit device, and therefore it no longer advertises learned EIGRP routes; it can advertise connected, summary, static, or redistributed routes.

```
R10#show ip route eigrp  | b Gateway
Gateway of last resort is not set


      150.1.0.0/32 is subnetted, 2 subnets
D        150.1.8.8 [90/130816] via 155.1.108.8, 00:03:22, GigabitEthernet1.108
      155.1.0.0/16 is variably subnetted, 6 subnets, 2 masks
D        155.1.8.0/24 [90/3072] via 155.1.108.8, 00:03:22, GigabitEthernet1.108
D        155.1.58.0/24 [90/3072] via 155.1.108.8, 00:03:22, GigabitEthernet1.108
!R5#show ip route 155.1.108.0
Routing entry for 155.1.108.0/24
  Known via "eigrp 100", distance 90, metric 3072, type internal
  Redistributing via eigrp 100
  Last update from 155.1.58.8 on GigabitEthernet1.58, 00:03:56 ago
  Routing Descriptor Blocks: * 155.1.58.8, from 155.1.58.8, 00:03:56 ago, via GigabitEthernet1.58
      Route metric is 3072, traffic share count is 1
      Total delay is 20 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R5#show ip route 155.1.10.0
% Subnet not in table
```

Output from the **`debug eigrp packet terse`** shows the progression of a QUERY messages in the network and its REPLY. First, R9 disables its VLAN 9 interface, withdrawing 155.1.9.0/24 and generating a QUERY.

```
R9#debug eigrp packets terse
(UPDATE, REQUEST, QUERY, REPLY, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on
!R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#interface GigabitEthernet1.9
R9(config-if)#shutdown
```

https://t.me/learningnets

```
!
EIGRP: Enqueueing QUERY on Gi1.79 - paklen 0 tid 0 iidbQ un/rely 0/1 serno 136-136
EIGRP: Sending QUERY on Gi1.79 - paklen 44 tid 0

  AS 100, Flags 0x0:(NULL), Seq 32/0 interfaceQ 0/0 iidbQ un/rely 0/0 serno 136-136
```

R5 receives the QUERY from R1, R2, R3, and R4.

```
R5#debug eigrp packets terse
    (UPDATE, REQUEST, QUERY, REPLY, UNKNOWN, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
EIGRP Packet debugging is on
!EIGRP: Received QUERY on Tu0 - paklen 44 nbr 155.1.0.3
  AS 100, Flags 0x0:(NULL), Seq 120/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Enqueueing ACK on Tu0 - paklen 0 nbr 155.1.0.3 tid 0
  Ack seq 120 iidbQ un/rely 0/0 peerQ un/rely 1/0
EIGRP: Sending ACK on Tu0 - paklen 0 nbr 155.1.0.3 tid 0
  AS 100, Flags 0x0:(NULL), Seq 0/120 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 1/0
!EIGRP: Received QUERY on Tu0 - paklen 44 nbr 155.1.0.1
  AS 100, Flags 0x0:(NULL), Seq 80/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Enqueueing ACK on Tu0 - paklen 0 nbr 155.1.0.1 tid 0
  Ack seq 80 iidbQ un/rely 0/0 peerQ un/rely 1/0
!EIGRP: Received QUERY on Tu0 - paklen 44 nbr 155.1.0.2
  AS 100, Flags 0x0:(NULL), Seq 56/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Enqueueing ACK on Tu0 - paklen 0 nbr 155.1.0.2 tid 0
  Ack seq 56 iidbQ un/rely 0/0 peerQ un/rely 1/0
!EIGRP: Received QUERY on Tu0 - paklen 44 nbr 155.1.0.4

  AS 100, Flags 0x0:(NULL), Seq 67/0 interfaceQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
EIGRP: Enqueueing ACK on Tu0 - paklen 0 nbr 155.1.0.4 tid 0
  Ack seq 67 iidbQ un/rely 0/0 peerQ un/rely 1/0
```

Normally, the QUERY is forwarded on to all neighbors of R5 except the stub neighbor, R8. Based on the order and speed R5 received QUERY messages from its neighbors, it will send some QUERY messages or just REPLY messages to its neighbors.

https://t.me/learningnets

R5:

```
EIGRP: Enqueueing QUERY on Tu0 - paklen 44 nbr 155.1.0.2 tid 0 iidbQ un/rely 0/0 peerQ un/rely 1/1 serno 88-88

EIGRP: Sending QUERY on Tu0 - paklen 44 tid 0
  AS 100, Flags 0x2:(CR), Seq 87/0 interfaceQ 0/0 iidbQ un/rely 0/0 serno 88-88
!EIGRP: Enqueueing QUERY on Gi1.45 - paklen 0 tid 0 iidbQ un/rely 0/1 serno 88-88

EIGRP: Sending QUERY on Gi1.45 - paklen 44 tid 0
  AS 100, Flags 0x0:(NULL), Seq 90/0 interfaceQ 0/0 iidbQ un/rely 0/0 serno 88-88
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Stub Routing with Leak Map

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure the EIGRP stub feature so that R5 does not receive EIGRP query messages.
- R5 should continue to advertise all EIGRP learned routes, with the exception of R8's Loopback0 prefix.

Configuration

```
R5:

ip prefix-list R8_LOOPBACK0 seq 5 permit 150.1.8.8/32
!
route-map STUB_LEAK_MAP deny 10
 match ip address prefix-list R8_LOOPBACK0
!
route-map STUB_LEAK_MAP permit 20
!
router eigrp 100
 eigrp stub connected leak-map STUB_LEAK_MAP
```

Verification

https://t.me/learningnets

The `leak-map` feature of EIGRP stub, like the leak-map for EIGRP summarization, allows the advertisement of routes that would normally be suppressed. When R5 is configured with only the `eigrp stub` command, it cannot be used as transit. This can be seen in the routing table views of all routers, for example R8 and R3:

```
R8#show ip route eigrp | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 3 subnets
D        150.1.5.5 [90/130816] via 155.1.58.5, 00:00:18, GigabitEthernet1.58
D        150.1.10.10 [90/130816] via 155.1.108.10, 00:01:44, GigabitEthernet1.108
      155.1.0.0/16 is variably subnetted, 10 subnets, 2 masks
D        155.1.0.0/24 [90/25856256] via 155.1.58.5, 00:00:18, GigabitEthernet1.58
D        155.1.5.0/24 [90/3072] via 155.1.58.5, 00:00:18, GigabitEthernet1.58
D        155.1.10.0/24 [90/3072] via 155.1.108.10, 00:01:44, GigabitEthernet1.108
D        155.1.45.0/24 [90/3072] via 155.1.58.5, 00:00:18, GigabitEthernet1.58
!R3#show ip route eigrp | b Gateway

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 8 subnets
D        150.1.1.1 [90/130816] via 155.1.13.1, 00:01:44, GigabitEthernet1.13
D        150.1.2.2 [90/130816] via 155.1.23.2, 00:01:44, GigabitEthernet1.23
D        150.1.4.4 [90/131072] via 155.1.13.1, 00:01:44, GigabitEthernet1.13
D        150.1.5.5 [90/131328] via 155.1.13.1, 00:01:39, GigabitEthernet1.13
D        150.1.6.6 [90/131072] via 155.1.37.7, 00:01:44, GigabitEthernet1.37
                   [90/131072] via 155.1.13.1, 00:01:44, GigabitEthernet1.13
D        150.1.7.7 [90/130816] via 155.1.37.7, 00:01:44, GigabitEthernet1.37
D        150.1.9.9 [90/131072] via 155.1.37.7, 00:01:44, GigabitEthernet1.37
      155.1.0.0/16 is variably subnetted, 16 subnets, 2 masks
D        155.1.5.0/24 [90/3584] via 155.1.13.1, 00:01:39, GigabitEthernet1.13
D        155.1.7.0/24 [90/3072] via 155.1.37.7, 00:01:44, GigabitEthernet1.37
D        155.1.9.0/24 [90/3328] via 155.1.37.7, 00:01:44, GigabitEthernet1.37
D        155.1.45.0/24 [90/3328] via 155.1.13.1, 00:01:39, GigabitEthernet1.13
D        155.1.58.0/24 [90/3584] via 155.1.13.1, 00:01:39, GigabitEthernet1.13
D        155.1.67.0/24 [90/3072] via 155.1.37.7, 00:01:44, GigabitEthernet1.37
D        155.1.79.0/24 [90/3072] via 155.1.37.7, 00:01:44, GigabitEthernet1.37
D        155.1.146.0/24
           [90/3072] via 155.1.13.1, 00:01:44, GigabitEthernet1.13
```

https://t.me/learningnets

In this design, R5 is configured to leak all dynamically learned EIGRP routes, with the exception of R8's Loopback0. If a failure in the network occurs, however, R5 will still not receive EIGRP QUERY messages. It will receive updates about network changes and send its own queries:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router eigrp 100
R5(config-router)#eigrp stub connected leak-map STUB_LEAK_MAP
!R8#show ip eigrp neighbors detail gigabitEthernet1.58
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
0   155.1.58.5              Gi1.58                   12 00:01:21    3   100  0  183
   Version 15.0/2.0, Retrans: 0, Retries: 0, Prefixes: 19
   Topology-ids from peer - 0  Stub Peer Advertising (CONNECTED LEAKMAP ) Routes
 Suppressing queries
Max Nbrs: 0, Current Nbrs: 0
!R3#show ip eigrp neighbors detail tunnel0
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
3   155.1.0.5               Tu0                      11 00:01:34   51  1398  0  182
   Version 15.0/2.0, Retrans: 0, Retries: 0, Prefixes: 21
   Topology-ids from peer - 0  Stub Peer Advertising (CONNECTED LEAKMAP ) Routes
 Suppressing queries
Max Nbrs: 0, Current Nbrs: 0
!R8#show ip route eigrp | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets
D        150.1.1.1 [90/131328] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        150.1.2.2 [90/131840] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        150.1.3.3 [90/131584] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        150.1.4.4 [90/131072] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        150.1.5.5 [90/130816] via 155.1.58.5, 00:02:22, GigabitEthernet1.58
D        150.1.6.6 [90/131328] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        150.1.7.7 [90/131584] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        150.1.9.9 [90/131840] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        150.1.10.10
           [90/130816] via 155.1.108.10, 00:06:03, GigabitEthernet1.108
      155.1.0.0/16 is variably subnetted, 18 subnets, 2 masks
D        155.1.0.0/24
           [90/25856256] via 155.1.58.5, 00:02:22, GigabitEthernet1.58
D        155.1.5.0/24 [90/3072] via 155.1.58.5, 00:02:22, GigabitEthernet1.58
D        155.1.7.0/24 [90/3840] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        155.1.9.0/24 [90/4096] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
```

https://t.me/learningnets

```
D        155.1.10.0/24
           [90/3072] via 155.1.108.10, 00:06:03, GigabitEthernet1.108
D        155.1.13.0/24 [90/3584] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        155.1.23.0/24 [90/3840] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        155.1.37.0/24 [90/3840] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        155.1.45.0/24 [90/3072] via 155.1.58.5, 00:02:22, GigabitEthernet1.58
D        155.1.67.0/24 [90/3584] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        155.1.79.0/24 [90/3840] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
D        155.1.146.0/24
           [90/3328] via 155.1.58.5, 00:02:18, GigabitEthernet1.58
!R3#show ip route 150.1.8.8
% Subnet not in table
!R3#show ip route 150.1.10.10
Routing entry for 150.1.10.10/32
  Known via "eigrp 100", distance 90, metric 131840, type internal
  Redistributing via eigrp 100
  Last update from 155.1.13.1 on GigabitEthernet1.13, 00:03:01 ago
  Routing Descriptor Blocks: * 155.1.13.1, from 155.1.13.1, 00:03:01 ago, via GigabitEthernet1.13

      Route metric is 131840, traffic share count is 1
      Total delay is 5050 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 5
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Passive Interface

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure the passive-interface feature on R5, R8, and R10 so that EIGRP Hello packets are not sent out the LAN segments without routers attached.
- Configure the passive-interface default feature on R7 and R9 so that EIGRP Hello packets are not sent out the LAN segments without routers attached.
- Ensure that full IPv4 reachability is maintained after this change is made.

Configuration

```
R5:
router eigrp 100
 passive-interface GigabitEthernet1.5

R7:
router eigrp 100
 passive-interface default
 no passive-interface GigabitEthernet1.67
 no passive-interface GigabitEthernet1.79
 no passive-interface GigabitEthernet1.37

R8:
router eigrp 100
 passive-interface GigabitEthernet1.8
```

https://t.me/learningnets

```
R9:

router eigrp 100
 passive-interface default
 no passive-interface GigabitEthernet1.79

R10:

router eigrp 100
 passive-interface GigabitEthernet1.10
```

Verification

The `passive-interface` command in EIGRP, like in RIPv2, stops the sending of updates out an interface. Unlike RIPv2, however, passive-interface in EIGRP will prevent forming of an adjacency on the interface because it stops sending EIGRP Hello packets as well, and hence the learning of any updates on the link. The `passive-interface default` command can be used to make all interfaces passive, and then interfaces can have the passive feature selectively disabled with the `no passive-interface` command:

```
R5# show ip protocols | begin eigrp
Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 Protocol for AS(100)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 150.1.5.5
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4
      Maximum hopcount 100
      Maximum metric variance 1

  Automatic Summarization: disabled
  Maximum path: 4
  Routing for Networks:
```

https://t.me/learningnets

```
    150.1.0.0
    155.1.0.0 Passive Interface(s):
GigabitEthernet1.5
  Routing Information Sources:
    Gateway         Distance      Last Update
    155.1.0.2             90      00:01:39
    155.1.0.3             90      00:01:39
    155.1.0.1             90      00:01:39
    155.1.0.4             90      00:01:39
    155.1.58.8            90      00:01:39
    155.1.45.4            90      00:01:39
  Distance: internal 90 external 170
!R7#show ip protocols | begin eigrp
Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 Protocol for AS(100)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 150.1.7.7
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4
      Maximum hopcount 100
      Maximum metric variance 1

  Automatic Summarization: disabled
  Maximum path: 4
  Routing for Networks:
    150.1.0.0
    155.1.0.0 Passive Interface(s):
GigabitEthernet1.7
Loopback0

  Routing Information Sources:
    Gateway         Distance      Last Update
    155.1.37.3            90      00:02:01
    155.1.79.9            90      00:02:01
    155.1.67.6            90      00:02:01
```

https://t.me/learningnets

```
  Distance: internal 90 external 170
```

Also note that the output of **`show ip eigrp interfaces`** displays only non-passive interfaces. For example, based on the configuration, Loopback0 of R5 is non-passive, while Loopback0 of R7 is passive:

```
R5#show ip eigrp interfaces
EIGRP-IPv4 Interfaces for AS(100)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Lo0                      0        0/0        0/0          0        0/0           0           0
Gi1.45                   1        0/0        0/0          5        0/0          50           0
Gi1.58                   1        0/0        0/0          2        0/0          50           0
Tu0                      4        0/0        0/0         33        6/227       156           0
!R7#show ip eigrp interfaces

EIGRP-IPv4 Interfaces for AS(100)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi1.67                   1        0/0        0/0          2        0/0          50           0
Gi1.79                   1        0/0        0/0          1        0/0          50           0
Gi1.37                   1        0/0        0/0          3        0/0          50           0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Prefix-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure a prefix-list on R4 so that it does not advertise its Loopback0 prefix out VLAN 45.
  - Use the most efficient list to accomplish this that will not deny any other networks than Loopback0.
- Configure prefix-list filtering on R1 so that it does not install any updates received from R4 on the VLAN 146 segment.
  - Allow all routes to be received from all other EIGRP neighbors.

Configuration

```
R1:
ip prefix-list NOT_FROM_R4 seq 5 deny 155.1.146.4/32
ip prefix-list NOT_FROM_R4 seq 10 permit 0.0.0.0/0 le 32
!
ip prefix-list PERMIT_ALL seq 5 permit 0.0.0.0/0 le 32
!
router eigrp 100
 distribute-list prefix PERMIT_ALL gateway NOT_FROM_R4 in


R4:


ip prefix-list LOOPBACK0_SUBNET seq 5 deny 150.1.4.4/32
```

https://t.me/learningnets

```
ip prefix-list LOOPBACK0_SUBNET seq 10 permit 0.0.0.0/0 le 32
!
router eigrp 100
 distribute-list prefix LOOPBACK0_SUBNET out GigabitEthernet1.45
```

Verification

Before filtering is implemented, verify that R4 advertises its Loopback0 prefix out on VLAN 45 and R1 accepts EIGRP updates from R4.

```
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.45.4 on GigabitEthernet1.45, 01:43:09 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 155.1.45.4, 01:43:09 ago, via GigabitEthernet1.45
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R1#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.4 on GigabitEthernet1.146, 01:43:23 ago
  Routing Descriptor Blocks: * 155.1.146.4, from 155.1.146.4, 01:43:23 ago, via GigabitEthernet1.146

      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

After filtering is implemented, verify that R4 no longer advertises its Loopback0 prefix out on VLAN 45 and R1 rejects all EIGRP updates from R4.

```
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "eigrp 100", distance 90, metric 25984000, type internal
  Redistributing via eigrp 100
  Last update from 155.1.0.4 on Tunnel0, 00:00:05 ago
  Routing Descriptor Blocks: * 155.1.0.4, from 155.1.0.4, 00:00:05 ago, via Tunnel0
      Route metric is 25984000, traffic share count is 1
      Total delay is 15000 microseconds, minimum bandwidth is 100 Kbit
      Reliability 255/255, minimum MTU 1400 bytes
```

https://t.me/learningnets

```
      Loading 1/255, Hops 1
!R5#show ip eigrp topology 150.1.4.4/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.5.5) for 150.1.4.4/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 130816
  Descriptor Blocks: 155.1.0.4 (Tunnel0), from 155.1.0.4, Send flag is 0x0
      Composite metric is (25984000/128256), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 1
        Originating router is 150.1.4.4 155.1.0.3 (Tunnel0), from 155.1.0.3, Send flag is 0x0
      Composite metric is (25984768/131328), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 4
        Originating router is 150.1.4.4 155.1.0.1 (Tunnel0), from 155.1.0.1, Send flag is 0x0
      Composite metric is (25985024/131584), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15040 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 5
        Originating router is 150.1.4.4 155.1.0.2 (Tunnel0), from 155.1.0.2, Send flag is 0x0
      Composite metric is (25985024/131584), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15040 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 5
        Originating router is 150.1.4.4
!R1#show ip eigrp topology | i via 155.1.146.
        via 155.1.146.6
 (3328/3072), GigabitEthernet1.146        via 155.1.146.6
 (131072/130816), GigabitEthernet1.146        via 155.1.146.6
 (3328/3072), GigabitEthernet1.146
```

https://t.me/learningnets

```
        via 155.1.146.6
 (3072/2816), GigabitEthernet1.146         via 155.1.146.6
 (3584/3328), GigabitEthernet1.146         via 155.1.146.6
 (130816/128256), GigabitEthernet1.146         via 155.1.146.6
 (131328/131072), GigabitEthernet1.146
!R1#show ip route eigrp | i 155.1.146.
D        150.1.6.6 [90/130816] via 155.1.146.6
, 01:47:21, GigabitEthernet1.146 D        150.1.7.7 [90/131072] via 155.1.146.6
, 00:10:03, GigabitEthernet1.146 D        150.1.9.9 [90/131328] via 155.1.146.6
, 00:02:13, GigabitEthernet1.146            [90/3328] via 155.1.146.6
, 00:10:03, GigabitEthernet1.146            [90/3584] via 155.1.146.6
, 00:02:13, GigabitEthernet1.146            [90/3072] via 155.1.146.6
, 00:10:05, GigabitEthernet1.146            [90/3328] via 155.1.146.6
, 00:10:03, GigabitEthernet1.146
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Standard Access-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure a one-line standard access-list on R9 to filter out all routes coming from R7 that have an odd number in the third octet.

Configuration

```
R9:

access-list 1 permit 0.0.0.0 255.255.254.255
!
router eigrp 100
 distribute-list 1 in GigabitEthernet1.79
```

Verification

Verify EIGRP topology and routing table on R9 before filtering is implemented.

```
R9#show ip eigrp topology 150.1.1.1/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.9.9) for 150.1.1.1/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131328
  Descriptor Blocks: 155.1.79.7 (GigabitEthernet1.79), from 155.1.79.7, Send flag is 0x0
      Composite metric is (131328/131072), route is Internal
      Vector metric:
```

https://t.me/learningnets

```
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.1.1
!R9#show ip route eigrp | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets
D        150.1.1.1 [90/131328] via 155.1.79.7, 00:00:35, GigabitEthernet1.79
D        150.1.2.2 [90/131328] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        150.1.3.3 [90/131072] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        150.1.4.4 [90/131328] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        150.1.5.5 [90/131584] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        150.1.6.6 [90/131072] via 155.1.79.7, 00:15:09, GigabitEthernet1.79
D        150.1.7.7 [90/130816] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        150.1.8.8 [90/131840] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        150.1.10.10 [90/132096] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
      155.1.0.0/16 is variably subnetted, 17 subnets, 2 masks
D        155.1.0.0/24
           [90/26880512] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        155.1.5.0/24 [90/3840] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        155.1.7.0/24 [90/3072] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        155.1.8.0/24 [90/4096] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        155.1.10.0/24 [90/4352] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        155.1.13.0/24 [90/3328] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        155.1.23.0/24 [90/3328] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        155.1.37.0/24 [90/3072] via 155.1.79.7, 00:15:11, GigabitEthernet1.79
D        155.1.45.0/24 [90/3584] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        155.1.58.0/24 [90/3840] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        155.1.67.0/24 [90/3072] via 155.1.79.7, 00:15:11, GigabitEthernet1.79

D        155.1.108.0/24 [90/4096] via 155.1.79.7, 00:00:34, GigabitEthernet1.79
D        155.1.146.0/24 [90/3328] via 155.1.79.7, 00:15:09, GigabitEthernet1.79
```

Verify EIGRP topology and routing table on R9 after filtering is implemented.

```
R9#show ip eigrp topology 150.1.1.1/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.9.9)%Entry 150.1.1.1/32 not in topology table
!R9#show ip route eigrp | b Gateway

Gateway of last resort is not set
```

https://t.me/learningnets

```
      150.1.0.0/32 is subnetted, 6 subnets
D        150.1.2.2 [90/131328] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        150.1.4.4 [90/131328] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        150.1.6.6 [90/131072] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        150.1.8.8 [90/131840] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        150.1.10.10 [90/132096] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
      155.1.0.0/16 is variably subnetted, 10 subnets, 2 masks
D        155.1.0.0/24
           [90/26880512] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        155.1.8.0/24 [90/4096] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        155.1.10.0/24 [90/4352] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        155.1.58.0/24 [90/3840] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        155.1.108.0/24 [90/4096] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
D        155.1.146.0/24 [90/3328] via 155.1.79.7, 00:00:43, GigabitEthernet1.79
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Extended Access-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Disable R5's Ethernet link to R4.
- Configure an extended access-list filter on R5 to achieve the following:
  - Traffic destined to Loopback0 prefixes of R4 and R6 is sent to R2.
  - Traffic destined to Loopback0 prefixes of R1 and R2 is sent to R3.
  - Traffic destined to Loopback0 prefixes of R7 and R9 is sent to R1.
- This filter should not affect any other updates on this segment.

Configuration

R5:

```
interface GigabitEthernet1.45
 shutdown
!
access-list 100 deny ip host 155.1.0.1 host 150.1.4.4
access-list 100 deny ip host 155.1.0.3 host 150.1.4.4
access-list 100 deny ip host 155.1.0.4 host 150.1.4.4
access-list 100 deny ip host 155.1.0.1 host 150.1.6.6
access-list 100 deny ip host 155.1.0.3 host 150.1.6.6
access-list 100 deny ip host 155.1.0.4 host 150.1.6.6
access-list 100 deny ip host 155.1.0.1 host 150.1.1.1
access-list 100 deny ip host 155.1.0.2 host 150.1.1.1
access-list 100 deny ip host 155.1.0.4 host 150.1.1.1
```

https://t.me/learningnets

```
access-list 100 deny ip host 155.1.0.1 host 150.1.2.2
access-list 100 deny ip host 155.1.0.2 host 150.1.2.2
access-list 100 deny ip host 155.1.0.4 host 150.1.2.2
access-list 100 deny ip host 155.1.0.2 host 150.1.7.7
access-list 100 deny ip host 155.1.0.3 host 150.1.7.7
access-list 100 deny ip host 155.1.0.4 host 150.1.7.7
access-list 100 deny ip host 155.1.0.2 host 150.1.9.9
access-list 100 deny ip host 155.1.0.3 host 150.1.9.9
access-list 100 deny ip host 155.1.0.4 host 150.1.9.9
access-list 100 permit ip any any
!
router eigrp 100
 distribute-list 100 in Tunnel0
```

Verification

Like RIP, extended access-lists when called as a distribute-list in IGP have a different meaning than in redistribution or in BGP. With BGP and redistribution, the “source” field in the ACL represents the network address, and the “destination” field represents the subnet mask. In IGP distribute-list application, the “source” field in the ACL matches the update source of the route, and the “destination” field represents the network address. This implementation allows us to control which networks we are receiving, but more importantly who we are receiving them from. With VLAN 45 interface disabled and before the filter is applied, R5 routes as follows:

```
R5#show ip route eigrp | i 150.1.
      150.1.0.0/32 is subnetted, 10 subnets
D        150.1.1.1 [90/25984000] via 155.1.0.1, 00:00:13, Tunnel0
D        150.1.2.2 [90/25984000] via 155.1.0.2, 00:00:13, Tunnel0
D        150.1.3.3 [90/25984000] via 155.1.0.3, 00:00:13, Tunnel0
D        150.1.4.4 [90/25984000] via 155.1.0.4, 00:00:13, Tunnel0
D        150.1.6.6 [90/25984256] via 155.1.0.4, 00:00:13, Tunnel0
D        150.1.7.7 [90/25984256] via 155.1.0.3, 00:00:13, Tunnel0
D        150.1.8.8 [90/130816] via 155.1.58.8, 00:30:05, GigabitEthernet1.58
D        150.1.9.9 [90/25984512] via 155.1.0.3, 00:00:13, Tunnel0
D        150.1.10.10 [90/131072] via 155.1.58.8, 00:30:05, GigabitEthernet1.58
!R5#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.4 4 msec *  3 msec
!R5#traceroute 150.1.6.6
Type escape sequence to abort.
Tracing the route to 150.1.6.6
```

https://t.me/learningnets

```
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 4 msec
    155.1.0.4 2 msec
    155.1.0.1 1 msec
  2 155.1.146.6 3 msec *  7 msec
!R5#traceroute 150.1.1.1
Type escape sequence to abort.
Tracing the route to 150.1.1.1
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 3 msec *  2 msec
!R5#traceroute 150.1.2.2
Type escape sequence to abort.
Tracing the route to 150.1.2.2
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.2 3 msec *  3 msec
!R5#traceroute 150.1.7.7
Type escape sequence to abort.
Tracing the route to 150.1.7.7
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.3 4 msec 1 msec 1 msec
  2 155.1.37.7 6 msec *  2 msec
!R5#traceroute 150.1.9.9
Type escape sequence to abort.
Tracing the route to 150.1.9.9
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.3 3 msec 1 msec 0 msec

  2 155.1.37.7 2 msec 1 msec 1 msec
  3 155.1.79.9 2 msec *  4 msec
```

Verify the EIGRP topology entries for one of the Loopbacks; for example, R6's:

```
R5#show ip eigrp topology 150.1.6.6/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.5.5) for 150.1.6.6/32
  State is Passive, Query origin flag is 1, 2 Successor(s), FD is 25984256
  Descriptor Blocks: 155.1.0.1 (Tunnel0), from 155.1.0.1, Send flag is 0x0
      Composite metric is (25984256/130816)
, route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15010 microseconds
        Reliability is 255/255
        Load is 2/255
        Minimum MTU is 1400
        Hop count is 2
        Originating router is 150.1.6.6 155.1.0.4 (Tunnel0), from 155.1.0.4, Send flag is 0x0
      Composite metric is (25984256/130816)
, route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
```

https://t.me/learningnets

```
        Total delay is 15010 microseconds
        Reliability is 255/255
        Load is 2/255
        Minimum MTU is 1400
        Hop count is 2
        Originating router is 150.1.6.6 155.1.0.3 (Tunnel0), from 155.1.0.3, Send flag is 0x0
      Composite metric is (25984512/131072), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15020 microseconds
        Reliability is 255/255
        Load is 2/255
        Minimum MTU is 1400
        Hop count is 3
        Originating router is 150.1.6.6 155.1.0.2 (Tunnel0), from 155.1.0.2, Send flag is 0x0

      Composite metric is (25984768/131328), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 15030 microseconds
        Reliability is 255/255
        Load is 2/255
        Minimum MTU is 1400
        Hop count is 4
        Originating router is 150.1.6.6
```

After distribute-list is implemented, R5 has only one possible way to route to these destinations, as requested by the task.

```
R5#show ip route eigrp | i 150.1.
      150.1.0.0/32 is subnetted, 10 subnets
D        150.1.1.1 [90/25984256] via 155.1.0.3, 00:00:04, Tunnel0
D        150.1.2.2 [90/25984256] via 155.1.0.3, 00:00:04, Tunnel0
D        150.1.3.3 [90/25984000] via 155.1.0.3, 00:00:04, Tunnel0
D        150.1.4.4 [90/25984768] via 155.1.0.2, 00:00:04, Tunnel0
D        150.1.6.6 [90/25984768] via 155.1.0.2, 00:00:04, Tunnel0
D        150.1.7.7 [90/25984512] via 155.1.0.1, 00:00:04, Tunnel0
D        150.1.8.8 [90/130816] via 155.1.58.8, 00:01:25, GigabitEthernet1.58
D        150.1.9.9 [90/25984768] via 155.1.0.1, 00:00:04, Tunnel0
D        150.1.10.10 [90/131072] via 155.1.58.8, 00:01:25, GigabitEthernet1.58
!R5#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.2 4 msec 1 msec 1 msec
```

https://t.me/learningnets

```
  2 155.1.23.3 1 msec 6 msec 1 msec
  3 155.1.13.1 2 msec 1 msec 1 msec
  4 155.1.146.4 3 msec *  6 msec
!R5#traceroute 150.1.6.6
Type escape sequence to abort.
Tracing the route to 150.1.6.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.2 3 msec 1 msec 0 msec
  2 155.1.23.3 1 msec 1 msec 1 msec
  3 155.1.13.1 1 msec 1 msec 1 msec
  4 155.1.146.6 2 msec *  4 msec
!R5#traceroute 150.1.1.1
Type escape sequence to abort.
Tracing the route to 150.1.1.1
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.3 3 msec 1 msec 1 msec
  2 155.1.13.1 2 msec *  3 msec
!R5#traceroute 150.1.2.2
Type escape sequence to abort.
Tracing the route to 150.1.2.2
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.3 4 msec 1 msec 2 msec
  2 155.1.23.2 2 msec *  3 msec
!R5#traceroute 150.1.7.7
Type escape sequence to abort.
Tracing the route to 150.1.7.7
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 3 msec 1 msec 1 msec
  2 155.1.13.3 1 msec 2 msec 1 msec
  3 155.1.37.7 2 msec *  3 msec
!R5#traceroute 150.1.9.9
Type escape sequence to abort.
Tracing the route to 150.1.9.9
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 3 msec 2 msec 1 msec

  2 155.1.146.6 11 msec 2 msec 3 msec
  3 155.1.67.7 2 msec 4 msec 1 msec
  4 155.1.79.9 4 msec *  4 msec
```

Verify the EIGRP topology entries for R6's Loopback0 after filtering; there is a single entry now.

```
R5#show ip eigrp topology 150.1.6.6/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.5.5) for 150.1.6.6/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 25984768
  Descriptor Blocks: 155.1.0.2 (Tunnel0), from 155.1.0.2, Send flag is 0x0
      Composite metric is (25984768/131328)
, route is Internal
      Vector metric:
```

https://t.me/learningnets

```
        Minimum bandwidth is 100 Kbit
        Total delay is 15030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 4
        Originating router is 150.1.6.6
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Offset Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure an offset-list on R7 so traffic destined for R3's Loopback0 prefix is sent to R6.
- If the Ethernet link to R6 is down, traffic should be rerouted directly to R3.

Configuration

```
R7:

access-list 1 permit host 150.1.3.3
!
router eigrp 100
 offset-list 1 in 2000 GigabitEthernet1.37
```

Verification

Like in RIP, the offset-list feature in EIGRP is used to modify the metric on a per-route basis or a per-interface basis. Before any metric modifications, we can see that R7 is routing directly to R3 to reach 150.1.3.3/32. There are no additional entries in the EIGRP topology table of R7 for this prefix because R6 also routes through R7 to reach it:

https://t.me/learningnets

```
R7#show ip route 150.1.3.3
Routing entry for 150.1.3.3/32

  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.37.3 on GigabitEthernet1.37, 01:05:42 ago
  Routing Descriptor Blocks: * 155.1.37.3, from 155.1.37.3, 01:05:42 ago, via GigabitEthernet1.37
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R7#show ip eigrp topology 150.1.3.3/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.7.7) for 150.1.3.3/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 130816
  Descriptor Blocks: 155.1.37.3 (GigabitEthernet1.37), from 155.1.37.3, Send flag is 0x0
      Composite metric is (130816/128256)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5010 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.3.3
!R7#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.37.3 2 msec *  2 msec

!R6#show ip route 150.1.3.3
Routing entry for 150.1.3.3/32
  Known via "eigrp 100", distance 90, metric 131072, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.1 on GigabitEthernet1.146, 00:51:42 ago
  Routing Descriptor Blocks: 155.1.146.1, from 155.1.146.1, 00:51:42 ago, via GigabitEthernet1.146
      Route metric is 131072, traffic share count is 1
      Total delay is 5020 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2 * 155.1.67.7, from 155.1.67.7, 00:51:42 ago, via GigabitEthernet1.67

      Route metric is 131072, traffic share count is 1
      Total delay is 5020 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
```

https://t.me/learningnets

```
      Loading 1/255, Hops 2
```

For R7 to route through R6 to reach this destination, the metric must be offset sufficiently so that R6 computes a lower composite metric through R1 than R7. Note that the offset value configured in EIGRP is added to the delay component of EIGRP metric. Also, because an access-list is used to match just 150.1.3.3/32, no other prefixes are affected by this traffic engineering:

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#router eigrp 100
R7(config-router)#offset-list 1 in 2000 GigabitEthernet1.37
!R7#show ip route 150.1.3.3
Routing entry for 150.1.3.3/32
  Known via "eigrp 100", distance 90, metric 131328, type internal
  Redistributing via eigrp 100
  Last update from 155.1.67.6 on GigabitEthernet1.67, 00:00:15 ago
  Routing Descriptor Blocks: * 155.1.67.6, from 155.1.67.6, 00:00:15 ago, via GigabitEthernet1.67
      Route metric is 131328, traffic share count is 1
      Total delay is 5030 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3
!R7#show ip eigrp topology 150.1.3.3/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.7.7) for 150.1.3.3/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131328
  Descriptor Blocks: 155.1.67.6 (GigabitEthernet1.67), from 155.1.67.6, Send flag is 0x0
      Composite metric is (131328/131072)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.3.3
155.1.37.3 (GigabitEthernet1.37), from 155.1.37.3, Send flag is 0x0
      Composite metric is (132816/130256), route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5088 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.3.3
```

https://t.me/learningnets

```
!R7#traceroute 150.1.3.3

Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.67.6 27 msec 5 msec 6 msec
  2 155.1.146.1 4 msec 3 msec 13 msec
  3 155.1.13.3 1 msec *  2 msec
!R6#show ip eigrp topology 150.1.3.3/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.6.6) for 150.1.3.3/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131072
  Descriptor Blocks: 155.1.146.1 (GigabitEthernet1.146), from 155.1.146.1, Send flag is 0x0
      Composite metric is (131072/130816)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5020 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 2
        Originating router is 150.1.3.3
```

Because the route through R3 is still installed in the topology table of R7, it will be used as a backup route if the path through R6 is lost:

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.67
R7(config-if)#shutdown
!%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.67.6 (GigabitEthernet1.67) is down: interface down
!R7#show ip eigrp topology 150.1.3.3/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.7.7) for 150.1.3.3/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131328
  Descriptor Blocks: 155.1.37.3 (GigabitEthernet1.37), from 155.1.37.3, Send flag is 0x0
      Composite metric is (132816/130256)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5088 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.3.3
!R7#show ip route 150.1.3.3
Routing entry for 150.1.3.3/32
```

https://t.me/learningnets

```
  Known via "eigrp 100", distance 90, metric 132816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.37.3 on GigabitEthernet1.37, 00:00:24 ago
  Routing Descriptor Blocks: * 155.1.37.3, from 155.1.37.3, 00:00:24 ago, via GigabitEthernet1.37
      Route metric is 132816, traffic share count is 1
      Total delay is 5088 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R7#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.37.3 3 msec *  2 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Administrative Distance

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure administrative distance filtering on R6 so that it does not install the route to R4's Loopback0 prefix.

Configuration

```
R6:

access-list 4 permit host 150.1.4.4
!
router eigrp 100
 distance 255 0.0.0.0 255.255.255.255 4
```

Verification

Like with IGP protocols, administrative distance can be set on a per-prefix basis in EIGRP. In this example, the source address of the route/update is ignored by matching an address of 0.0.0.0 with the wildcard 255.255.255.255, whereas access-list 4 matches the route for which to change the distance. Because the AD value of 255 is "infinite," the route in question cannot be installed in the routing table or the EIGRP topology. Verify that route is present before applying the configuration:

https://t.me/learningnets

```
R6#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32

  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.4 on GigabitEthernet1.146, 01:05:39 ago
  Routing Descriptor Blocks: * 155.1.146.4, from 155.1.146.4, 01:05:39 ago, via GigabitEthernet1.146
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R6#show ip eigrp topology 150.1.4.4/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.6.6) for 150.1.4.4/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 130816
  Descriptor Blocks: 155.1.146.4 (GigabitEthernet1.146), from 155.1.146.4, Send flag is 0x0
      Composite metric is (130816/128256)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5010 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.4.4
!R6#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.4 6 msec *  6 msec
```

Apply the configuration and verify that prefix is no longer installed in routing table and EIGRP topology:

https://t.me/learningnets

```
R6#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
R6(config)#access-list 4 permit host 150.1.4.4
R6(config)#router eigrp 100
R6(config-router)# distance 255 0.0.0.0 255.255.255.255 4

!R6#show ip route 150.1.4.4
% Subnet not in table

!R6#show ip eigrp topology 150.1.4.4/32

EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.6.6)%Entry 150.1.4.4/32 not in topology table
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Per Neighbor AD

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure administrative distance filtering on R3 so that traffic destined for R7's Loopback0 prefix is sent to R1.
  - ensure R3 never uses the direct path via R7.

Configuration

```
R3:

access-list 7 permit host 150.1.7.7
!
router eigrp 100
 distance 255 155.1.37.7 0.0.0.0 7
```

Pitfall

The administrative distance for EIGRP internal routes can be changed on a per-prefix basis, but external EIGRP routes cannot.

Verification

Before any distance modifications, R3 routes directly to R7 to reach 150.1.7.7/32.

https://t.me/learningnets

Based on the routing table and EIGRP topology table, we can see that the Feasible Distance is 130816, and the neighbor the route is learned from is 155.1.37.7:

```
R3#show ip route 150.1.7.7
Routing entry for 150.1.7.7/32
  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.37.7 on GigabitEthernet1.37, 00:19:51 ago
  Routing Descriptor Blocks: * 155.1.37.7, from 155.1.37.7, 00:19:51 ago, via GigabitEthernet1.37
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R3#show ip eigrp topology 150.1.7.7/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3) for 150.1.7.7/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 130816
  Descriptor Blocks: 155.1.37.7 (GigabitEthernet1.37), from 155.1.37.7, Send flag is 0x0
      Composite metric is (130816/128256)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5010 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.7.7
  155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0
      Composite metric is (27008768/131328), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 55030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 4
        Originating router is 150.1.7.7
!R3#traceroute 150.1.7.7
Type escape sequence to abort.
Tracing the route to 150.1.7.7
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.37.7 4 msec *  3 msec
```

As we saw in the previous example, administrative distance can be changed on a

https://t.me/learningnets

per-prefix basis. Based on matching who the route is learned from, the distance can also be changed on a per-prefix per-neighbor basis. Although the composite metric is higher through R1 than it was originally through R7, the route through R7 cannot be installed in the topology table because it has an infinite administrative distance. This implies that R3 must route through R1 or R5 to reach the destination:

```
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#access-list 7 permit host 150.1.7.7
R3(config)#router eigrp 100
R3(config-router)#distance 255 155.1.37.7 0.0.0.0 7

!R3#show ip route 150.1.7.7
Routing entry for 150.1.7.7/32
  Known via "eigrp 100", distance 90, metric 131328, type internal
  Redistributing via eigrp 100
  Last update from 155.1.13.1 on GigabitEthernet1.13, 00:00:09 ago
  Routing Descriptor Blocks: * 155.1.13.1, from 155.1.13.1, 00:00:09 ago, via GigabitEthernet1.13
      Route metric is 131328, traffic share count is 1
      Total delay is 5030 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3
!R3#show ip eigrp topology 150.1.7.7/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3) for 150.1.7.7/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131328
  Descriptor Blocks: 155.1.13.1 (GigabitEthernet1.13), from 155.1.13.1, Send flag is 0x0
      Composite metric is (131328/131072)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.7.7 155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0
      Composite metric is (27008768/131328), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 55030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 4
        Originating router is 150.1.7.7
!R3#traceroute 150.1.7.7
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 150.1.7.7
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.13.1 5 msec 2 msec 1 msec

  2 155.1.146.6 1 msec 2 msec 1 msec
  3 155.1.67.7 2 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Filtering with Route Maps

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 as follows:
  - Loopback1 as 160.1.4.4/32 and redistribute it into EIGRP with the tag value of 4.
  - Loopback2 as 170.1.4.4/32 and redistribute it into EIGRP.
- Configure a route-map filter on R2 that matches this tag value and denies the route from being installed in the routing table.
- Configure a route-map filter on R3 that denies EIGRP routes with a metric in the range of 120.000–140.000 from entering the routing table.
- These filters should not affect any other networks advertised by R4 or learned by R2 and R3.

Configuration

```
R2:
route-map FILTER_ON_TAGS deny 10
 match tag 4
!
route-map FILTER_ON_TAGS permit 20
!
router eigrp 100
 distribute-list route-map FILTER_ON_TAGS in
```

https://t.me/learningnets

```
R3:

route-map FILTER_ON_METRIC_RANGE deny 10
 match metric 130000 +- 10000
!
route-map FILTER_ON_METRIC_RANGE permit 20
!
router eigrp 100
 distribute-list route-map FILTER_ON_METRIC_RANGE in

R4:

interface Loopback1
 ip address 160.1.4.4 255.255.255.255
!
interface Loopback2
 ip address 170.1.4.4 255.255.255.255
!
ip prefix-list LOOPBACK1 seq 5 permit 160.1.4.4/32
ip prefix-list LOOPBACK2 seq 5 permit 170.1.4.4/32
!
route-map CONNECTED_TO_EIGRP permit 10
 match ip address prefix-list LOOPBACK1
 set tag 4
!
route-map CONNECTED_TO_EIGRP permit 20
 match ip address prefix-list LOOPBACK2
!
router eigrp 100
 redistribute connected metric 100000 100 255 1 1500 route-map CONNECTED_TO_EIGRP
```

Verification

Unlike BGP, filtering with route-maps in IGP is usually limited to redistribution filtering only. However, EIGRP supports route-map filtering as a distribute-list with matches on metric and tag values. Route tags are set at the time of redistribution and can be used like BGP community values to group prefixes together without having to match on the actual route in a prefix-list or access-list. In this example, we can see that R2 and R4 see the prefix 160.1.4.4/32 with a tag of 4 in the topology table. R2 installs this in the EIGRP topology and routing table until the distribute-list is applied, which denies prefixes with that tag value.

```
R4#show ip eigrp topology 160.1.4.4/32
```

https://t.me/learningnets

```
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.4.4) for 160.1.4.4/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 51200
  Descriptor Blocks: 0.0.0.0, from Rconnected, Send flag is 0x0
      Composite metric is (51200/0), route is External
      Vector metric:
        Minimum bandwidth is 100000 Kbit
        Total delay is 1000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 0
        Originating router is 150.1.4.4
      External data:
        AS number of route is 0
        External protocol is Connected, external metric is 0 Administrator tag is 4 (0x00000004)
!R2#show ip eigrp topology 160.1.4.4/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.2.2) for 160.1.4.4/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 51968
  Descriptor Blocks: 155.1.23.3 (GigabitEthernet1.23), from 155.1.23.3, Send flag is 0x0
      Composite metric is (51968/51712)
, route is External
      Vector metric:
        Minimum bandwidth is 100000 Kbit
        Total delay is 1030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 3
        Originating router is 150.1.4.4
      External data:
        AS number of route is 0
        External protocol is Connected, external metric is 0 Administrator tag is 4 (0x00000004)
 155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0
      Composite metric is (26905856/51456), route is External
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 51010 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 2
        Originating router is 150.1.4.4
      External data:
        AS number of route is 0
        External protocol is Connected, external metric is 0 Administrator tag is 4 (0x00000004)
```

https://t.me/learningnets

```
!R2#show ip route 160.1.4.4
Routing entry for 160.1.4.4/32

  Known via "eigrp 100", distance 170, metric 51968
  Tag 4, type external
  Redistributing via eigrp 100
  Last update from 155.1.23.3 on GigabitEthernet1.23, 00:01:22 ago
  Routing Descriptor Blocks: * 155.1.23.3, from 155.1.23.3, 00:01:22 ago, via GigabitEthernet1.23
      Route metric is 51968, traffic share count is 1
      Total delay is 1030 microseconds, minimum bandwidth is 100000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3 Route tag 4
```

Verify that the tag value of 4 is not applied for Loopback2.

```
R2#show ip route 170.1.4.4
Routing entry for 170.1.4.4/32
  Known via "eigrp 100", distance 170, metric 51968, type external
  Redistributing via eigrp 100
  Last update from 155.1.23.3 on GigabitEthernet1.23, 00:03:05 ago
  Routing Descriptor Blocks: * 155.1.23.3, from 155.1.23.3, 00:03:05 ago, via GigabitEthernet1.23

      Route metric is 51968, traffic share count is 1
      Total delay is 1030 microseconds, minimum bandwidth is 100000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3
```

After applying the filtering on R2, Loopback1 prefix will be removed from both the EIGRP topology and routing table, but Loopback2 will be allowed.

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R2(config)#route-map FILTER_ON_TAGS deny 10
R2(config-route-map)# match tag 4
R2(config-route-map)#route-map FILTER_ON_TAGS permit 20
R2(config-route-map)#router eigrp 100
R2(config-router)# distribute-list route-map FILTER_ON_TAGS in
!R2#show ip route 160.1.4.4
% Network not in table
!R2#show ip eigrp topology 160.1.4.4/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.2.2)%Entry 160.1.4.4/32 not in topology table
!R2#show ip route 170.1.4.4
Routing entry for 170.1.4.4/32
  Known via "eigrp 100", distance 170, metric 51968, type external
  Redistributing via eigrp 100
```

https://t.me/learningnets

```
Last update from 155.1.23.3 on GigabitEthernet1.23, 00:05:37 ago
Routing Descriptor Blocks: * 155.1.23.3, from 155.1.23.3, 00:05:37 ago, via GigabitEthernet1.23

    Route metric is 51968, traffic share count is 1
    Total delay is 1030 microseconds, minimum bandwidth is 100000 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 3
```

As a filter for metrics, the route-map can match on an absolute metric value, such as with the **`match metric 10`** command, or on a range of metrics. In this case, metrics in the range of 120.000 to 140.000 are filtered out based on matching the value 130.000, plus or minus 10.000. Verify the routing table of R3 before applying the filter.

https://t.me/learningnets

```
R3#show ip route eigrp | b Gateway

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets D        150.1.1.1 [90/130816]
 via 155.1.13.1, 00:40:50, GigabitEthernet1.13 D        150.1.2.2 [90/130816]
 via 155.1.23.2, 00:40:50, GigabitEthernet1.23 D        150.1.4.4 [90/131072]
 via 155.1.13.1, 00:40:50, GigabitEthernet1.13 D        150.1.5.5 [90/131328]
 via 155.1.13.1, 00:40:50, GigabitEthernet1.13 D        150.1.6.6 [90/131072]
 via 155.1.37.7, 00:40:50, GigabitEthernet1.37 [90/131072]
 via 155.1.13.1, 00:40:50, GigabitEthernet1.13 D        150.1.7.7 [90/130816]
 via 155.1.37.7, 00:40:50, GigabitEthernet1.37 D        150.1.8.8 [90/131584]
 via 155.1.13.1, 00:40:50, GigabitEthernet1.13 D        150.1.9.9 [90/131072]
 via 155.1.37.7, 00:40:50, GigabitEthernet1.37 D        150.1.10.10 [90/131840]
 via 155.1.13.1, 00:40:50, GigabitEthernet1.13
      155.1.0.0/16 is variably subnetted, 19 subnets, 2 masks
D        155.1.5.0/24 [90/3584] via 155.1.13.1, 00:40:50, GigabitEthernet1.13
D        155.1.7.0/24 [90/3072] via 155.1.37.7, 00:40:50, GigabitEthernet1.37
D        155.1.8.0/24 [90/3840] via 155.1.13.1, 00:40:50, GigabitEthernet1.13
D        155.1.9.0/24 [90/3328] via 155.1.37.7, 00:40:50, GigabitEthernet1.37
D        155.1.10.0/24 [90/4096] via 155.1.13.1, 00:40:50, GigabitEthernet1.13
D        155.1.45.0/24 [90/3328] via 155.1.13.1, 00:40:50, GigabitEthernet1.13
D        155.1.58.0/24 [90/3584] via 155.1.13.1, 00:40:50, GigabitEthernet1.13
D        155.1.67.0/24 [90/3072] via 155.1.37.7, 00:40:50, GigabitEthernet1.37
D        155.1.79.0/24 [90/3072] via 155.1.37.7, 00:40:50, GigabitEthernet1.37
D        155.1.108.0/24 [90/3840] via 155.1.13.1, 00:40:50, GigabitEthernet1.13
D        155.1.146.0/24 [90/3072] via 155.1.13.1, 00:40:50, GigabitEthernet1.13
      160.1.0.0/32 is subnetted, 1 subnets
D EX     160.1.4.4 [170/51712] via 155.1.13.1, 00:04:26, GigabitEthernet1.13
      170.1.0.0/32 is subnetted, 1 subnets
D EX     170.1.4.4 [170/51712] via 155.1.13.1, 00:09:18, GigabitEthernet1.13
```

Look in the EIGRP topology table for R1's Loopback0 prefix; for example, before filtering is applied, the path via VLAN 13 is the successor.

```
R3#show ip eigrp topology 150.1.1.1/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3) for 150.1.1.1/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 130816
  Descriptor Blocks: 155.1.13.1 (GigabitEthernet1.13), from 155.1.13.1, Send flag is 0x0
      Composite metric is (130816/128256)
, route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5010 microseconds
```

https://t.me/learningnets

```
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.1.1 155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0
      Composite metric is (27008512/131072), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 55020 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 3
        Originating router is 150.1.1.1
!R3#traceroute 150.1.1.1
Type escape sequence to abort.
Tracing the route to 150.1.1.1
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.13.1 3 msec *  2 msec
```

We can see from the output below that the highlighted prefixes are no longer installed in the routing table via the same path after the filter is applied; an alternate path with the metric not within the filtering intervals is used instead.

```
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# route-map FILTER_ON_METRIC_RANGE deny 10
R3(config-route-map)# match metric 130000 +- 10000
R3(config-route-map)#route-map FILTER_ON_METRIC_RANGE permit 20
R3(config-route-map)#router eigrp 100
R3(config-router)# distribute-list route-map FILTER_ON_METRIC_RANGE in
!R3#show ip route eigrp | b Gateway
Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets D        150.1.1.1 [90/27008512]
 via 155.1.0.5, 00:01:45, Tunnel0 D        150.1.2.2 [90/25984768]
 via 155.1.13.1, 00:01:45, GigabitEthernet1.13 D        150.1.4.4 [90/27008256]
 via 155.1.0.5, 00:01:46, Tunnel0 D        150.1.5.5 [90/27008000]
 via 155.1.0.5, 00:01:46, Tunnel0 D        150.1.6.6 [90/27008512]
 via 155.1.0.5, 00:01:45, Tunnel0 D        150.1.7.7 [90/27008768]
 via 155.1.0.5, 00:01:45, Tunnel0 D        150.1.8.8 [90/27008256]
 via 155.1.0.5, 00:01:46, Tunnel0 D        150.1.9.9 [90/27009024]
 via 155.1.0.5, 00:01:45, Tunnel0 D        150.1.10.10 [90/27008512]
 via 155.1.0.5, 00:01:46, Tunnel0
      155.1.0.0/16 is variably subnetted, 19 subnets, 2 masks
```

https://t.me/learningnets

```
D        155.1.5.0/24 [90/3584] via 155.1.13.1, 00:43:46, GigabitEthernet1.13
D        155.1.7.0/24 [90/3072] via 155.1.37.7, 00:43:46, GigabitEthernet1.37
D        155.1.8.0/24 [90/3840] via 155.1.13.1, 00:43:46, GigabitEthernet1.13
D        155.1.9.0/24 [90/3328] via 155.1.37.7, 00:43:46, GigabitEthernet1.37
D        155.1.10.0/24 [90/4096] via 155.1.13.1, 00:43:46, GigabitEthernet1.13
D        155.1.45.0/24 [90/3328] via 155.1.13.1, 00:43:46, GigabitEthernet1.13
D        155.1.58.0/24 [90/3584] via 155.1.13.1, 00:43:46, GigabitEthernet1.13
D        155.1.67.0/24 [90/3072] via 155.1.37.7, 00:43:46, GigabitEthernet1.37
D        155.1.79.0/24 [90/3072] via 155.1.37.7, 00:43:46, GigabitEthernet1.37
D        155.1.108.0/24 [90/3840] via 155.1.13.1, 00:43:46, GigabitEthernet1.13
D        155.1.146.0/24 [90/3072] via 155.1.13.1, 00:43:46, GigabitEthernet1.13
      160.1.0.0/32 is subnetted, 1 subnets
D EX     160.1.4.4 [170/51712] via 155.1.13.1, 00:07:22, GigabitEthernet1.13
      170.1.0.0/32 is subnetted, 1 subnets
D EX     170.1.4.4 [170/51712] via 155.1.13.1, 00:12:14, GigabitEthernet1.13
```

Look in the EIGRP topology table for R1's Loopback0 prefix; for example, the path via VLAN 13 is no longer present.

```
R3#show ip eigrp topology 150.1.1.1/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3) for 150.1.1.1/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 27008512
  Descriptor Blocks: 155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0
      Composite metric is (27008512/131072)
, route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 55020 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 3
        Originating router is 150.1.1.1
155.1.23.2 (GigabitEthernet1.23), from 155.1.23.2, Send flag is 0x0
      Composite metric is (27008768/27008512), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 55030 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 4
        Originating router is 150.1.1.1
!R3#traceroute 150.1.1.1
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 150.1.1.1
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.5 2 msec 1 msec 1 msec

  2 155.1.45.4 10 msec 2 msec 1 msec
  3 155.1.146.1 3 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Bandwidth Pacing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R5 so that EIGRP cannot use more than 200Kbps of bandwidth on its DMVPN connection, assuming that the link speed is 2Mbps.

Configuration

```
R5:

interface Tunnel0
 bandwidth 2000
 ip bandwidth-percent eigrp 100 10
```

Verification

By default EIGRP can use up to maximum 50% of the administrative bandwidth of the interface. The absolut value can be changed by modifying the bandwidth on the interface or by changing the percentage level with interface-level command `ip bandwidth-percent eigrp <AS_NR> <percentage>` . Verify the SRTT and Pacing timers for the tunnel interface before bandwidth usage is changed:

```
R5#show ip eigrp interfaces
EIGRP-IPv4 Interfaces for AS(100)
                          Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
```

https://t.me/learningnets

```
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Lo0                      0        0/0         0/0        0        0/0           0           0
Gi1.5                    0        0/0         0/0        0        0/0           0           0
Gi1.45                   1        0/0         0/0        2        0/0          50           0
Gi1.58                   1        0/0         0/0        6        0/0          50           0
Tu0                      4        0/0         0/0       30        6/227        60           0
```

Verify the SRTT and Pacing timers for the tunnel interface after bandwidth usage is changed:

```
R5#show ip eigrp interfaces
EIGRP-IPv4 Interfaces for AS(100)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Lo0                      0        0/0         0/0        0        0/0           0           0
Gi1.5                    0        0/0         0/0        0        0/0           0           0
Gi1.45                   1        0/0         0/0        2        0/0          50           0
Gi1.58                   1        0/0         0/0        5        0/0          50           0
Tu0                      4        0/0         0/0       24        0/56        120           0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Default Metric

> You must load the initial configuration files for the section, **Basic EIGRP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure Loopback1 interface on R2 with IPv4 address 160.1.2.2/32.
- Configure a static route on R3 for R2's Loopback1 prefix via the directly connected Ethernet link.
  - Advertise this prefix into EIGRP as external routes using a default metric of 100Mbps, 100 microseconds of delay, maximum reliability, minimum load, and an MTU of 1500 bytes.

Configuration

```
R2:
interface Loopback1
 ip address 160.1.2.2 255.255.255.255

R3:

ip route 160.1.2.2 255.255.255.255 155.1.23.2
!
router eigrp 100
 redistribute static
 default-metric 100000 10 255 1 1500
```

https://t.me/learningnets

Verification

When redistributing static and connected prefixes into EIGRP or between EIGRP processes, metrics are automatically derived from the source prefix. For all other redistribution, the metric must be manually set on the redistribute statement, under a route-map, or from the default metric. The default metric affects all redistributed prefixes for which a specific metric has not been configured. For example, just perform redistribution on R3 without specifying the default-metric; note the metric values on R3 and R1.

```
R3#show ip eigrp topology 160.1.2.2/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3) for 160.1.2.2/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2816
  Descriptor Blocks: 155.1.23.2, from Rstatic, Send flag is 0x0
      Composite metric is (2816/0)
, route is External
      Vector metric: Minimum bandwidth is 1000000 Kbit
 Total delay is 10 microseconds
 Reliability is 255/255
 Load is 1/255
 Minimum MTU is 1500
        Hop count is 0
        Originating router is 150.1.3.3
      External data:
        AS number of route is 0
        External protocol is Static, external metric is 0
        Administrator tag is 0 (0x00000000)
!R1#show ip eigrp topology 160.1.2.2/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.1.1) for 160.1.2.2/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 3072
  Descriptor Blocks: 155.1.13.3 (GigabitEthernet1.13), from 155.1.13.3, Send flag is 0x0
      Composite metric is (3072/2816), route is External
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 20 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 150.1.3.3
      External data:
        AS number of route is 0
        External protocol is Static, external metric is 0
        Administrator tag is 0 (0x00000000)
```

https://t.me/learningnets

```
155.1.0.5 (Tunnel0), from 155.1.0.5, Send flag is 0x0

      Composite metric is (26881024/3584), route is External
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 50040 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1400
        Hop count is 4
        Originating router is 150.1.3.3
      External data:
        AS number of route is 0
        External protocol is Static, external metric is 0
        Administrator tag is 0 (0x00000000)
!R1#show ip route 160.1.2.2
Routing entry for 160.1.2.2/32  Known via "eigrp 100", distance 170, metric 3072
, type external
  Redistributing via eigrp 100
  Last update from 155.1.13.3 on GigabitEthernet1.13, 00:06:06 ago
  Routing Descriptor Blocks: * 155.1.13.3, from 155.1.13.3, 00:06:06 ago, via GigabitEthernet1.13

      Route metric is 3072, traffic share count is 1
      Total delay is 20 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

Configure the default metric on EIGRP process and note the changes in composite metric on R3 and metric value on R1.

```
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R3(config)#router eigrp 100
R3(config-router)#default-metric 100000 10 255 1 1500
!R3#show ip eigrp topology 160.1.2.2/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3) for 160.1.2.2/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2816
  Descriptor Blocks:
  155.1.23.2, from Rstatic, Send flag is 0x0
      Composite metric is (28160/0), route is External
      Vector metric: Minimum bandwidth is 100000 Kbit
Total delay is 100 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 0
```

https://t.me/learningnets

```
        Originating router is 150.1.3.3
      External data:
        AS number of route is 0
        External protocol is Static, external metric is 0
        Administrator tag is 0 (0x00000000)
!R1#show ip route 160.1.2.2
Routing entry for 160.1.2.2/32  Known via "eigrp 100", distance 170, metric 28416
, type external
  Redistributing via eigrp 100
  Last update from 155.1.13.3 on GigabitEthernet1.13, 00:00:32 ago
  Routing Descriptor Blocks: * 155.1.13.3, from 155.1.13.3, 00:00:32 ago, via GigabitEthernet1.13

      Route metric is 28416, traffic share count is 1
      Total delay is 110 microseconds, minimum bandwidth is 100000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Neighbor Logging

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R9 so that it does not log EIGRP neighbor adjacency events.
- EIGRP warning logs should not be generated more often than every 20 seconds.

Configuration

```
R9:

router eigrp 100
 no eigrp log-neighbor-changes
 eigrp log-neighbor-warnings 20
```

Verification

Manually reset the EIGRP neighborship between R7 and R9, note that R9 no longer logs any messages when neighbors are DOWN or adjacency is back up, but R7 does as this is the default setting:

https://t.me/learningnets

```
R7#clear ip eigrp neighbors gigabitEthernet1.79

! %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.79.9 (GigabitEthernet1.79) is down: manually cleared
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.79.9 (GigabitEthernet1.79) is up: new adjacency
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Router-ID

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure Loopback1 on R2 with IPv4 address of 160.1.2.2/32 and redistribute it into EIGRP.
- Modify the EIGRP Router ID on R8 so that external EIGRP routes generated by R2 are ignored.

Configuration

```
R2:
interface Loopback1
 ip address 160.1.2.2 255.255.255.255
!
router eigrp 100
 redistribute connected

R8:

router eigrp 100
 eigrp router-id 150.1.2.2
```

Verification

https://t.me/learningnets

EIGRP uses the router-id field in external routes as a loop prevention mechanism. The router that originates the external route inserts its EIGRP router-id into the update. If an update is received back in with a router-id in this field matching the local router-id, the update is dropped. Verify that R2's Loopback2 is accepted by R8 before changing the router ID on R8:

```
R8#show ip eigrp topology 160.1.2.2/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.8.8) for 160.1.2.2/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131840
  Descriptor Blocks: 155.1.58.5 (GigabitEthernet1.58), from 155.1.58.5, Send flag is 0x0
      Composite metric is (131840/131584), route is External
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5050 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 5 Originating router is 150.1.2.2
      External data:
        AS number of route is 0
        External protocol is Connected, external metric is 0
        Administrator tag is 0 (0x00000000)
!R8#show ip route 160.1.2.2
Routing entry for 160.1.2.2/32
  Known via "eigrp 100", distance 170, metric 131840, type external
  Redistributing via eigrp 100
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:01:03 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 155.1.58.5, 00:01:03 ago, via GigabitEthernet1.58

      Route metric is 131840, traffic share count is 1
      Total delay is 5050 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 5
```

The EIGRP router ID can be visualized from the topology table:

```
R2#show ip eigrp topology 100.100.100.100/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.2.2)

%Entry 100.100.100.100/32 not in topology table
```

https://t.me/learningnets

Modify R8's EIGRP router ID value to match on R2's EIGRP router ID and verify that R8 no longer accepts the update about R2's Loopback1:

```
R8#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R8(config)#router eigrp 100
R8(config-router)# eigrp router-id 150.1.2.2
!R8#show ip eigrp topology 160.1.2.2/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.2.2)%Entry 160.1.2.2/32 not in topology table
!R8#show ip route 160.1.2.2
% Network not in table
```

There are several methods for identifying this problem, one would be to enable **`debug eigrp fsm`** on the router rejecting the update, or from the EIGRP event logs:

```
R8#show ip eigrp events | b Ignored
22   16:17:57.341 Ignored route, dup routerid int: 150.1.2.2
23   16:17:57.341 Poison squashed: 155.1.8.0/24 reverse
24   16:17:57.341 Ignored route, dup routerid int: 150.1.2.2

25   16:17:57.341 Poison squashed: 150.1.8.8/32 reverse
26   16:17:57.341 Ignored route, dup routerid int: 150.1.2.2
27   16:17:57.297 Change queue emptied, entries: 18
28   16:17:57.297 Ignored route, metric: 160.1.2.2/32 metric(131840)
29   16:17:57.294 Ignored route, neighbor info: 155.1.58.5 GigabitEthernet1.58
30   16:17:57.294 Ignored route, dup routerid ext: 150.1.2.2
31   16:17:57.294 Ignored route, dup routerid int: 150.1.2.2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - EIGRP

EIGRP Maximum Hops

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic EIGRP Routing** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure all devices in EIGRP AS 100 so that routes with a hop count of greater than 5 are considered invalid.

Configuration

```
R1 – R10:

router eigrp 100
 metric maximum-hops 5
```

Verification

Based on the logical network topology and considering all links are active, traffic from R9 to R10 will follow the Ethernet path and use the DMVPN path as seconday (feasible successor). For this reason, EIGRP prefixes advertised by R9 and R10 will have a hop-count of 6:

```
R9#show ip eigrp topology 150.1.10.10/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.9.9) for 150.1.10.10/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 132096
  Descriptor Blocks: 155.1.79.7 (GigabitEthernet1.79), from 155.1.79.7, Send flag is 0x0
```

https://t.me/learningnets

```
      Composite metric is (132096/131840), route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5060 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500 Hop count is 6
        Originating router is 150.1.10.10
!R9#show ip route 150.1.10.10
Routing entry for 150.1.10.10/32
  Known via "eigrp 100", distance 90, metric 132096, type internal
  Redistributing via eigrp 100
  Last update from 155.1.79.7 on GigabitEthernet1.79, 14:30:53 ago
  Routing Descriptor Blocks: * 155.1.79.7, from 155.1.79.7, 14:30:53 ago, via GigabitEthernet1.79
      Route metric is 132096, traffic share count is 1
      Total delay is 5060 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes      Loading 1/255, Hops 6
!R9#traceroute 150.1.10.10
Type escape sequence to abort.
Tracing the route to 150.1.10.10
VRF info: (vrf in name/id, vrf out name/id) 1
 155.1.79.7 3 msec 1 msec 1 msec 2
 155.1.67.6 2 msec 2 msec 2 msec 3
 155.1.146.4 4 msec 3 msec 3 msec 4
 155.1.45.5 3 msec 3 msec 3 msec 5
 155.1.58.8 6 msec 5 msec 10 msec 6
 155.1.108.10 7 msec *  8 msec
```

Although EIGRP does not use hop-count to compute its metric like RIP, it still enforces a maximum diameter for the EIGRP autonomous-system, by default allowing for a maximum hop-count of 100:

```
R9#show ip protocols | section eigrp
Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 Protocol for AS(100)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 150.1.9.9
```

https://t.me/learningnets

```
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4 Maximum hopcount 100


      Maximum metric variance 1
```

By modifying the maximim hop-count to 5, the end result will be that EIGRP updates of R9 will be rejected by R10 and vice-versa due to exceeding the configured hop-count, thus IPv4 connectivity between R9 and R10 will be lost:

```
R9#show ip route 150.1.10.10
% Subnet not in table
!R9#show ip eigrp topology 150.1.10.10/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.9.9)%Entry 150.1.10.10/32 not in topology table
```

Verify that EIGRP maximum hop-count was successfully modified and that R10's prefixes are rejected by R9 for this reason:

```
R9#show ip protocols | section eigrp
Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 Protocol for AS(100)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 150.1.9.9
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4 Maximum hopcount 5
      Maximum metric variance 1
!R9#show ip eigrp events
Event information for AS 100:
1    07:02:37.960 NDB delete: 155.1.10.0/24 1
2    07:02:37.960 Poison squashed: 155.1.10.0/24 rt net gone
3    07:02:37.960 RDB delete: 155.1.10.0/24 155.1.79.7
4    07:02:37.960 Find FS: 155.1.10.0/24 metric(Infinity)
5    07:02:37.960 Free reply status: 155.1.10.0/24
```

https://t.me/learningnets

```
6    07:02:37.960 Clr handle num/bits: 0 0x0
7    07:02:37.960 Clr handle dest/cnt: 155.1.10.0/24 0
8    07:02:37.960 Rcv reply met/succ met: metric(Infinity) metric(Infinity)
9    07:02:37.960 Rcv reply dest/nh: 155.1.10.0/24 155.1.79.7
10   07:02:37.960 Ignored route, hopcount: 155.1.10.0 5
11   07:02:37.960 NDB delete: 150.1.10.10/32 1
12   07:02:37.960 Poison squashed: 150.1.10.10/32 rt net gone
13   07:02:37.960 RDB delete: 150.1.10.10/32 155.1.79.7
14   07:02:37.960 Find FS: 150.1.10.10/32 metric(Infinity)
15   07:02:37.960 Free reply status: 150.1.10.10/32
16   07:02:37.959 Clr handle num/bits: 0 0x0
17   07:02:37.959 Clr handle dest/cnt: 150.1.10.10/32 0
18   07:02:37.959 Rcv reply met/succ met: metric(Infinity) metric(Infinity)
19   07:02:37.959 Rcv reply dest/nh: 150.1.10.10/32 155.1.79.7
20   07:02:37.959 Ignored route, hopcount: 150.1.10.10 5
21   07:02:37.951 Metric set: 155.1.10.0/24 metric(Infinity)

22   07:02:37.951 Active net/peers: 155.1.10.0/24 1
23   07:02:37.951 FC not sat Dmin/met: metric(Infinity) metric(26881280)
24   07:02:37.951 Find FS: 155.1.10.0/24 metric(26881280)
25   07:02:37.951 Rcv update met/succmet: metric(Infinity) metric(Infinity)
26   07:02:37.951 Rcv update dest/nh: 155.1.10.0/24 155.1.79.7
```

Verify that other routers in the topology are not affected, for example R7, because receivd EIGRP updates comply with the configured maximum hop-count:

```
R7#show ip route 150.1.10.10
Routing entry for 150.1.10.10/32
  Known via "eigrp 100", distance 90, metric 131840, type internal
  Redistributing via eigrp 100
  Last update from 155.1.67.6 on GigabitEthernet1.67, 00:00:53 ago
  Routing Descriptor Blocks: * 155.1.67.6, from 155.1.67.6, 00:00:53 ago, via GigabitEthernet1.67
      Route metric is 131840, traffic share count is 1
      Total delay is 5050 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes      Loading 1/255, Hops 5
!R7#show ip eigrp topology 150.1.10.10/32
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.7.7) for 150.1.10.10/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131840
  Descriptor Blocks: 155.1.67.6 (GigabitEthernet1.67), from 155.1.67.6, Send flag is 0x0
      Composite metric is (131840/131584), route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 5050 microseconds
        Reliability is 255/255
        Load is 1/255
```

https://t.me/learningnets

```
        Minimum MTU is 1500 Hop count is 5
        Originating router is 150.1.10.10
!R7#traceroute 150.1.10.10
Type escape sequence to abort.
Tracing the route to 150.1.10.10
VRF info: (vrf in name/id, vrf out name/id) 1
 155.1.67.6 1 msec 1 msec 1 msec 2
 155.1.146.4 32 msec 64 msec 6 msec 3
 155.1.45.5 5 msec 5 msec 4 msec 4
 155.1.58.8 154 msec 35 msec 82 msec 5
 155.1.108.10 8 msec *  8 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF over Broadcast Media

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF using process ID of 1 on R6, R7 and R9 as follows:
  - Use only interface level commands on R9.
  - Do not use any interface level commands on R6 and R7.
  - Use area 67 between R6 and R7, area 79 between R7 and R9.
  - Advertise Loopback0 prefixes of R6 in area 67 and Loopback0 of R9 in area 79.
- On R6, enable OSPF only for interfaces with the exact IP addresses of **150.1.6.6** and **155.1.67.6**.
- On R7, enable OSPF on all interfaces within the subnets **155.1.67.0/24** and **155.1.79.0/24**.
- Ensure R7 has IP connectivity with R6's and R9's Loopback0 prefixes.

Configuration

```
R6:
router ospf 1
 network 155.1.67.6 0.0.0.0 area 67
 network 150.1.6.6 0.0.0.0 area 67
R7:
router ospf 1
 network 155.1.67.0 0.0.0.255 area 67
 network 155.1.79.0 0.0.0.255 area 79
R9:
```

https://t.me/learningnets

```
interface GigabitEthernet1.79
 ip ospf 1 area 79
!
interface Loopback0
 ip ospf 1 area 79
```

Verification

This task illustrates two different ways to enable the OSPF process. These include the legacy `network` statement under the OSPF process and the interface-level command `ip ospf [process-id] area [area-id]` . Both accomplish the same thing with one minor exception. If an interface is IP unnumbered, and there is a `network` statement that matches the IP address of the primary interface, both the primary interface and the unnumbered interface will have OSPF enabled on them in the designated area. Additionally, when enabled at the interface level, by default OSPF will inject both primary and secondary subnets of the interface in the OSPF database and advertise it to its neighbors. If you want to suppress the secondary prefixes, use the `ip ospf [process-id] area [area-id] secondaries none` command.

The `network` statement in OSPF, just like the network statement under the EIGRP process, is not used to originate a network advertisement. Instead it simply enables the OSPF process on the interface. If multiple network statements overlap the same interface, the most specific match based on the wildcard mask wins.

R6 enables the OSPF area 67 only on the interface with the exact IP address of 155.1.67.6 with command `network 155.1.67.6 0.0.0.0 area 67` , which does not mean, however, that the network 155.1.67.6/32 itself will be advertised. Instead, OSPF will read the prefix from the interface configuration and bring the 155.1.67.0/24 subnet in the OSPF database.

Likewise on R7, the `network 155.1.67.0 0.0.0.255 area 67` command means that OSPF area 67 will be enabled on any interface within the 155.1.67.0/24 subnet. It is just a coincidence that the network command actually matches also the prefix-length/subnet-mask.

Regardless whether the `network` statement or the `ip ospf` statement are used, the result of them can be verified with the `show ip ospf interface brief` command. Note that in the output below, there is no functional difference seen between R6 and R9 which had OSPF enabled differently:

```
R6#show ip ospf interface brief
Interface     PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo0            1      67
               150.1.6.6/32       1      LOOP   0/0 Gi1.67         1      67
               155.1.67.6/24      1      BDR    1/1
```

https://t.me/learningnets

```
!
!R9#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo0           1      79
               150.1.9.9/32       1     LOOP  0/0 Gi1.79        1      79
               155.1.79.9/24      1     DR    1/1
```

There is, however, a detailed output which identifies how was OSPF enabled for that interface, with or without the `network` command:

```
R6#show ip ospf interface gigabitEthernet1.67
GigabitEthernet1.67 is up, line protocol is up   Internet Address 155.1.67.6/24, Area 67,
Attached via Network Statement
  Process ID 1, Router ID 150.1.6.6, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 150.1.7.7, Interface address 155.1.67.7
  Backup Designated router (ID) 150.1.6.6, Interface address 155.1.67.6
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.7.7  (Designated Router)
  Suppress hello for 0 neighbor(s)
!
!R9#show ip ospf  interface gigabitEthernet1.79
GigabitEthernet1.79 is up, line protocol is up   Internet Address 155.1.79.9/24, Area 79,
Attached via Interface Enable

  Process ID 1, Router ID 150.1.9.9, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 150.1.7.7, Interface address 155.1.79.7
  Backup Designated router (ID) 150.1.9.9, Interface address 155.1.79.9
```

https://t.me/learningnets

```
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.7.7  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

Once it is verified that the interfaces are configured in the correct areas, the next verification is to check the adjacency state of the OSPF neighbors with the **`show ip ospf neighbor`** command. In order to form an OSPF adjacency some attributes must match between neighbors, while others must be unique. The common attributes that must match are the area number, timers, authentication, stub flags, MTU, and compatible network types. The attributes that must be unique are the interface IP addresses and the router-ids. The router-id is a 32-bit number and is chosen first based on the process-level **`router-id`** command, second based on the highest active Loopback IP interface, and last based on the highest active non-Loopback interface IP address. Because the LSA origination is based on the router-id, each router needs a unique OSPF router-id within the OSPF domain, otherwise OSPF database colissions occur and the SPF tree cannot be properly calculated. Moreover, two routers with the same router-id cannot become neighbors, for the same exact reason; basically the router is saying i can't become neighbor with myself.

Verify that R7 is OSPF neighbor with both R6 and R9, and reached a functional neighbor state. For this case, where there are only two neighbors on the broadcast segment, a functional state means the **FULL** state:

```
R7#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.6.6          1   FULL
/BDR        00:00:39    155.1.67.6      GigabitEthernet1.67 150.1.9.9          1   FULL
/BDR        00:00:38    155.1.79.9      GigabitEthernet1.79
!
!R6#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.7.7          1   FULL
/DR         00:00:38    155.1.67.7      GigabitEthernet1.67
```

https://t.me/learningnets

```
!
!R9#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.7.7          1   FULL
/DR         00:00:34    155.1.79.7      GigabitEthernet1.79
```

Although adjacencies have been established, a fundamental underlying design issue still exists in the network. At this point, only areas 67 and 79 are configured. The backbone area 0 is not configured on any links. This implies that the devices can route within their own area (Intra-Area), but not between areas (Inter-Area). This is because the Area Border Router (ABR) that connects one area to area zero is responsible for generating the Network Summary LSA (LSA 3) that describes the inter-area routes. The result of this can be seen by viewing both the OSPF database table and routing tables of the routers. Because R7 is attached to both areas, it has router LSA's from both areas, while R7 and R9 only from the areas being attached to:

```
R6#show ip ospf database

            OSPF Router with ID (150.1.6.6) (Process ID 1)
 Router Link States (Area 67)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.6.6       150.1.6.6       597         0x80000002 0x003336 2
150.1.7.7       150.1.7.7       598         0x80000002 0x006BAD 1

                Net Link States (Area 67)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.67.7      150.1.7.7       598         0x80000001 0x00A6BD
!
!R9#show ip ospf database

            OSPF Router with ID (150.1.9.9) (Process ID 1)
 Router Link States (Area 79)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.7.7       150.1.7.7       459         0x80000003 0x00728D 1
150.1.9.9       150.1.9.9       490         0x80000001 0x0080BC 2

                Net Link States (Area 79)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.79.7      150.1.7.7       459         0x80000001 0x0073DE
!
```

https://t.me/learningnets

```
!R7#show ip ospf database


            OSPF Router with ID (150.1.7.7) (Process ID 1)
 Router Link States (Area 67)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.6.6       150.1.6.6       738         0x80000002 0x003336 2
150.1.7.7       150.1.7.7       737         0x80000002 0x006BAD 1

                Net Link States (Area 67)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.67.7      150.1.7.7       737         0x80000001 0x00A6BD
 Router Link States (Area 79)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.7.7       150.1.7.7       467         0x80000003 0x00728D 1
150.1.9.9       150.1.9.9       500         0x80000001 0x0080BC 2


                Net Link States (Area 79)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.79.7      150.1.7.7       467         0x80000001 0x0073DE
```

Becuase R7 is not configured as ABR to perform LSA1 to LSA3 translation, R6 and R9 actually learn no routes through OSPF, as R7 does not advertise any prefixes in area 67 or area 79, other than the directly connected Ethernet link with R6 and R9:

```
R6#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set
 R6#
!
!R9#show ip route ospf
```

https://t.me/learningnets

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set
R9#
!
!R7#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 3 subnets
O        150.1.6.6 [110/2] via 155.1.67.6, 00:32:09, GigabitEthernet1.67
O        150.1.9.9 [110/2] via 155.1.79.9, 00:27:38, GigabitEthernet1.79
```

Verify that R7 has IP connectivity with R6's and R9's Loopback0 prefixes:

https://t.me/learningnets

```
R7#ping 150.1.6.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/10 ms
!
!R7#ping 150.1.9.9

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF over DMVPN

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF area 0 on R1, R2, R3, R4, and R5's connection to the DMVPN network:
  - Use process ID of 1.
  - Advertise Loopback0 prefixes of R1 - R2 into area 0.
  - Advertise Loopback0 prefixes of R3 - R4 into area **Y**, where **Y** is the router number.
  - Use a single `network` command on the DMVPN hub which is R5.
  - Use only interface level commands on the DMVPN spokes.
- Change the default OSPF network type to **non-broadcast**.
- Ensure you have IP connectivity between Loopback0 prefixes of the DMVPN routers.

Configuration

```
R1 - R2:
interface Tunnel0
 ip ospf 1 area 0
 ip ospf network non-broadcast
!
interface Loopback0
 ip ospf 1 area 0
R3:
interface Tunnel0
 ip ospf 1 area 0
```

https://t.me/learningnets

```
 ip ospf network non-broadcast
!
interface Loopback0
 ip ospf 1 area 3
R4:
interface Tunnel0
 ip ospf 1 area 0
 ip ospf network non-broadcast
!
interface Loopback0
 ip ospf 1 area 4
R5:

interface Tunnel0
 ip ospf network non-broadcast
!
router ospf 1
 network 0.0.0.0 255.255.255.255 area 0
 neighbor 155.1.0.1
 neighbor 155.1.0.2
 neighbor 155.1.0.3
 neighbor 155.1.0.4
```

Verification

In this solution, OSPF is enabled on spokes with the interface-level **`ip ospf`** command. The **`network`** statement could have also been used under the OSPF process, but this is used only on the hub. In order to enable OSPF on the hub on both Loopback0 and Tunnel0, a very generic **`network`** command was used, which basically enabled OSPF on all interfaces in area 0:

```
R5#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo0           1      0
               150.1.5.5/32       1     LOOP  0/0 Gi1.100       1      0
               169.254.100.5/24   1     DR    0/0 Gi1.58        1      0
               155.1.58.5/24      1     DR    0/0 Gi1.45        1      0
               155.1.45.5/24      1     DR    0/0 Gi1.5         1      0
               155.1.5.5/24       1     DR    0/0 Tu0           1      0
               155.1.0.5/24       1000  DR    4/4
```

Verify also on spokes that OSPF is enabled in area 0:

```
R1#show ip ospf interface brief
```

https://t.me/learningnets

```
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo0           1      0
               150.1.1.1/32       1     LOOP  0/0 Tu0           1      0
               155.1.0.1/24       1000  BDR   1/1
!
!R2#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo0           1      0
               150.1.2.2/32       1     LOOP  0/0 Tu0           1      0
               155.1.0.2/24       1000  BDR   1/1
!
!R3#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo0           1      3
               150.1.3.3/32       1     LOOP  0/0 Tu0           1      0
               155.1.0.3/24       1000  BDR   1/1
!
!R4#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo0           1      4
               150.1.4.4/32       1     LOOP  0/0 Tu0           1      0
               155.1.0.4/24       1000  BDR   1/1
```

The default OSPF network types on the DMVPN interface (which is a mGRE interface) is **Point-to-Point**, as seen from the `show ip ospf interface` output below before changing it:

```
R1#show ip ospf interface tunnel0
Tunnel0 is up, line protocol is up
  Internet Address 155.1.0.1/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.1.1, Network Type POINT_TO_POINT
, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
```

https://t.me/learningnets

```
  Suppress hello for 0 neighbor(s)
```

The Non-Broadcast network type means that there will be a DR/BDR election, and that hellos are exchanged as unicast. To unicast OSPF hellos the **`neighbor`** statement needs to be configured under the OSPF process of the DR. When the DROTHERs and/or BDR hear the unicast hellos from the DR, they will automatically respond back with their own unicast hellos. This implies that the **`neighbor`** statement can be configured everywhere, but is only required on the DR. As within the DMVPN cloud OSPF packets are only between hub and spokes (not between spokes), R5 needs to be the DR. When R5 is configured with the neighbor statement, the **`show ip ospf neighbor`** output should be checked to verify adjacency. R1, R2, R3, and R4 are the spokes for this DMVPN cloud. They will either see themselves as BDR or DROTHER because they do not see each other's OSPF Hello Packets:

```
R5#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.1.1          1   FULL
/DROTHER    00:01:30    155.1.0.1       Tunnel0 150.1.2.2          1   FULL
/DROTHER    00:01:40    155.1.0.2       Tunnel0 150.1.3.3          1   FULL
/DROTHER    00:01:32    155.1.0.3       Tunnel0 150.1.4.4          1   FULL
/BDR        00:01:54    155.1.0.4       Tunnel0
!
!R5#show ip ospf interface tunnel0
Tunnel0 is up, line protocol is up
  Internet Address 155.1.0.5/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.5.5, Network Type NON_BROADCAST
, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no           no             Base
  Enabled by interface config, including secondary ip addresses  Transmit Delay is 1 sec, State DR
, Priority 1 Designated Router (ID) 150.1.5.5, Interface address 155.1.0.5
  Backup Designated router (ID) 150.1.4.4, Interface address 155.1.0.4
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 5
  Last flood scan time is 1 msec, maximum is 1 msec
  Neighbor Count is 4, Adjacent neighbor count is 4
```

https://t.me/learningnets

```
    Adjacent with neighbor 150.1.1.1
    Adjacent with neighbor 150.1.2.2
    Adjacent with neighbor 150.1.3.3
    Adjacent with neighbor 150.1.4.4  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
!
!R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.5.5          1   FULL
/DR         00:01:56    155.1.0.5       Tunnel0
!
!R1#show ip ospf interface tunnel0
Tunnel0 is up, line protocol is up
  Internet Address 155.1.0.1/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.1.1, Network Type NON_BROADCAST
, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no          no            Base
  Enabled by interface config, including secondary ip addresses  Transmit Delay is 1 sec, State BDR
, Priority 1 Designated Router (ID) 150.1.5.5, Interface address 155.1.0.5

  Backup Designated router (ID) 150.1.1.1, Interface address 155.1.0.1
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:08
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.5.5  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

When adjacency is established in area 0, Inter-Area routing advertisements can be propagated throughout the entire topology. This is because R1, R2, R3, R4, and R5 are now ABRs, and can originate the Network Summary LSA (LSA 3) describing Inter-Area routes to the other neighbors in their attached areas. From the `show ip ospf database` output on R5, the **Summary Net link States (Area 0)** shows ABRs that are advertising information from other areas into area 0. In this case, R3 and R4

https://t.me/learningnets

are advertising their Loopbacks, note that R5 does not know the area these prefixes were injected into:

```
R5#show ip ospf database

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.1.1       150.1.1.1       719         0x80000007 0x0034F0 2
150.1.2.2       150.1.2.2       174         0x80000006 0x0041DD 2
150.1.3.3       150.1.3.3       300         0x80000007 0x005A6A 1
150.1.4.4       150.1.4.4       1307        0x80000006 0x004878 1
150.1.5.5       150.1.5.5       1292        0x80000006 0x003145 6

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.0.5       150.1.5.5       1292        0x80000004 0x00755E
 Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum 150.1.3.3        150.1.3.3
       295         0x80000001 0x0028D8 150.1.4.4       150.1.4.4
       1558        0x80000001 0x0006F6
!
!R5#show ip ospf database summary

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 340
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.3.3 (summary Network Number)
 Advertising Router: 150.1.3.3
  LS Seq Number: 80000001
  Checksum: 0x28D8
  Length: 28 Network Mask: /32
        MTID: 0         Metric: 1

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1602
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
```

https://t.me/learningnets

```
Link State ID: 150.1.4.4 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0x6F6
  Length: 28 Network Mask: /32

        MTID: 0         Metric: 1
```

As R3 and R4 are connected to multiple areas, both take all prefixes from area 0 LSA Type-1 and generate LSA Type-3 into areas 3 and 4, for example R3:

https://t.me/learningnets

```
R3#show ip ospf database


            OSPF Router with ID (150.1.3.3) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.1.1       150.1.1.1       1007        0x80000007 0x0034F0 2
150.1.2.2       150.1.2.2       463         0x80000006 0x0041DD 2
150.1.3.3       150.1.3.3       586         0x80000007 0x005A6A 1
150.1.4.4       150.1.4.4       1594        0x80000006 0x004878 1
150.1.5.5       150.1.5.5       1580        0x80000006 0x003145 6

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.0.5       150.1.5.5       1580        0x80000004 0x00755E

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
150.1.3.3       150.1.3.3       582         0x80000001 0x0028D8
150.1.4.4       150.1.4.4       1845        0x80000001 0x0006F6

                Router Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.3.3       150.1.3.3       586         0x80000001 0x00F166 1
 Summary Net Link States (Area 3)


Link ID         ADV Router      Age         Seq#       Checksum 150.1.1.1        150.1.3.3
       582         0x80000002 0x008493 150.1.2.2        150.1.3.3
       582         0x80000002 0x006FA6 150.1.4.4        150.1.3.3
       587         0x80000001 0x0047CB 150.1.5.5        150.1.3.3
       582         0x80000002 0x0030DF 155.1.0.0        150.1.3.3
       582         0x80000002 0x004EC7 155.1.5.0        150.1.3.3
       582         0x80000002 0x0021EE 155.1.45.0       150.1.3.3
       582         0x80000002 0x006780 155.1.58.0       150.1.3.3
       582         0x80000002 0x00D703 169.254.100.0    150.1.3.3
       582         0x80000002 0x00693B
```

https://t.me/learningnets

The result of the Inter-Area routing advertisements can be seen in the routing table via the `show ip route ospf` output. These new routes are denoted as **O IA** for OSPF Inter-Area. The next important design issue in this example is how the DR processes routing advertisements over the DMVPN cloud. There is no direct adjacency between spoke routers, because regardless of the DMVPN Phase being configured, IGP packets are not sent between spokes (although with certain configuration this can be achieved, there is no reason for doing it). If you reached this step and R5 is not the DR for the segment, an additional problem will be seen that is covered in the next section regarding the DR/BDR election process. However, for the sake of this example, let's assume that R5 is the DR.

The DR state on the segment means that R5 is responsible for replicating LSA information between its adjacent neighbors. For example, R3 sends R5 the LSA type-3 route 150.1.3.3/24 describing its Loopback0 prefix. When R4 or any other spoke learns this over the DMVPN cloud it comes from the DR, R5, but the next-hop value of the route is 155.1.0.3, the originator of the LSA type-3 advertisement:

```
R5#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets
O        150.1.1.1 [110/1001] via 155.1.0.1, 04:12:00, Tunnel0
O        150.1.2.2 [110/1001] via 155.1.0.2, 04:12:00, Tunnel0
O IA     150.1.3.3 [110/1001] via 155.1.0.3
, 02:14:55, Tunnel0
O IA     150.1.4.4 [110/1001] via 155.1.0.4, 02:35:57, Tunnel0
!
!R4#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
```

https://t.me/learningnets

```
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets
O        150.1.1.1 [110/1001] via 155.1.0.1, 02:36:18, Tunnel0
O        150.1.2.2 [110/1001] via 155.1.0.2, 02:36:18, Tunnel0
O IA     150.1.3.3 [110/1001] via 155.1.0.3
, 02:15:10, Tunnel0
O        150.1.5.5 [110/1001] via 155.1.0.5, 02:36:18, Tunnel0
      155.1.0.0/16 is variably subnetted, 8 subnets, 2 masks
O        155.1.5.0/24 [110/1001] via 155.1.0.5, 02:36:18, Tunnel0
O        155.1.58.0/24 [110/1001] via 155.1.0.5, 02:36:18, Tunnel0
```

This is because the DR passes the routes along, but it does not modify any of the routing lookup attributes. The result of this behavior is seen when route recursion is performed to the final destination. From the IP Routing section of this workbook, recall that when a routing lookup is done, the router must also perform layer 3 to layer 2 mapping for the next-hop value on the link. Let's look at what happens when R4 tries to send traffic to 150.1.5.5. First, R4 finds the longest match to 155.1.5.5, which is 150.1.5.5/32 via the next-hop 155.1.0.5:

```
R4#show ip route 150.1.5.5
Routing entry for 150.1.5.5/32
  Known via "ospf 1", distance 110, metric 1001, type intra area
  Last update from 155.1.0.5 on Tunnel0, 02:38:47 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 02:38:47 ago, via Tunnel0

      Route metric is 1001, traffic share count is 1
```

R4 then must do another recursive lookup to find out how to forward toward 155.1.0.5. This is seen via the match 155.1.0.0/24 out Tunnel0:

```
R4#show ip route 155.1.0.5
Routing entry for 155.1.0.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Routing Descriptor Blocks: * directly connected, via Tunnel0

      Route metric is 0, traffic share count is 1
```

Because Tunnel0 is a DMVPN tunnel interface (mGRE), R4 must now determine the NBMA address of the next-hop through NHRP, which was statically defined in

https://t.me/learningnets

the initial configs. In this case, as the NBMA addresses of all routers share the same VLAN, R4 will also need the ARP entry for R5's NBMA. In this case, the layer 3 routing lookup is successful, and the layer 2 resolution is successful. The result is a successful ICMP PING to the destination:

```
R4#show ip nhrp 155.1.0.5
155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 06:45:13, never expire
   Type: static, Flags: used  NBMA address: 169.254.100.5
!
!R4#show ip arp 169.254.100.5
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  169.254.100.5         158 0050.568d.2798
  ARPA   GigabitEthernet1.100
!
!R4#ping 150.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 2/4/12 ms
```

All the above can be seen at the CEF and adjacency level, which is the data the router actually uses for forwarding:

```
R4#show ip cef 150.1.5.5 internal
150.1.5.5/32
, epoch 2, RIB[I], refcount 6, per-destination sharing
  sources: RIB
  feature space:
   IPRM: 0x00028000
   Broker: linked, distributed at 4th priority
  ifnums: Tunnel0(21): 155.1.0.5
  path 7F77C16D9A88, path list 7F77C13440D8, share 1/1, type attached nexthop, for IPv4
  nexthop 155.1.0.5 Tunnel0, adjacency IP midchain out of Tunnel0, addr 155.1.0.5 7F77C133EB10
  output chain: IP midchain out of Tunnel0, addr 155.1.0.5 7F77C133EB10
IP adj out of GigabitEthernet1.100, addr 169.254.100.5
 7F77C133E930
!
!R4#show adjacency 169.254.100.5 detail
Protocol Interface                 Address IP GigabitEthernet1.100       169.254.100.5(11)
                                   1755 packets, 343118 bytes
                                   epoch 0
                                   sourced in sev-epoch 477
                                   Encap length 18 0050568D2798
0050568D293981000064
```

https://t.me/learningnets

```
0800
L2 destination address byte offset 0
L2 destination address byte length 6
Link-type after encap: dot1Q
ARP
```

For spoke-to-spoke traffic, it depends on the DMVPN Phase and the OSPF network type enabled to really know how will traffic be routed finally (through the hub, or directly from spoke-to-spoke), as initially is always routed via the hub. These aspects are discussed in the DMVPN section. So spoke-to-spoke traffic will always be functional with DMVPN, which is different than with Frame-Relay case, where a full-mesh of spoke-to-spoke DLCIs was required, or alternatively a full-mesh of **`frame-relay map`** statements through the hub was required, in order to obtain spoke-to-spoke connectivity in the data-plane. This is due to the dynamic spoke-to-spoke provisioning behaviour of DMVPN, versus the static spoke-to-spoke DLCI provisioning of Frame-Relay.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF DR/BDR Election Manipulation

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF in area 0 over DMVPN cloud and on VLAN 146.
  - Use a **broadcast** OSPF network type.
- Advertise Loopback0 prefixes into area 0, on routers R1 - R6.
- Configure DMVPN routers to ensure that R5 is elected as the DR.
- Configure routers on VLAN 146 so that:
  - R6 is preferred to become the DR on the segment.
  - if R6 goes down, R1 takes over the DR role.
  - R4 does not participate in the DR election process.

Configuration

```
R1 - R4:
interface Tunnel0
 ip ospf 1 area 0
 ip ospf priority 0
 ip ospf network broadcast
!
interface Loopback0
 ip ospf 1 area 0
R1:
interface GigabitEthernet1.146
 ip ospf 1 area 0
 ip ospf priority 254
```

https://t.me/learningnets

```
R4:

interface GigabitEthernet1.146
 ip ospf 1 area 0
 ip ospf priority 0
R6:
interface GigabitEthernet1.146
 ip ospf 1 area 0
 ip ospf priority 255
!
interface Loopback0
 ip ospf 1 area 0
R5:

interface Tunnel0
 ip ospf 1 area 0
 ip ospf network broadcast
!
interface Loopback0
 ip ospf 1 area 0
```

Verification

The OSPF DR/BDR election is determined based on the interface-level OSPF priority value along with the router-id. The device with the highest priority is elected the DR, and the device with the second-highest priority is elected the BDR. If there is a tie in the priority value, the device with the higher router-id is elected the DR, and the device with second-highest router-id the BDR. However, in certain designs the election may become unpredictable. DR/BDR election takes place only for OSPF network types of **broadcast** and **non-broadcast**.

This unpredictability is caused by the fact that the OSPF DR/BDR election does not support preemption like IS-IS does for its DIS election. Preemption means that if a new device comes onto the segment with a higher priority or router-id, it can take the DR/BDR status away from the current device. Because OSPF does not support this, new devices must wait for a failure of the DR or BDR before the next election occurs. Additionally, in the order in which the routers load their OSPF process will influence the election; basically a router starts sending HELLO packets on the segment, and if no other OSPF routers are detected on the segment to negotiate the DR/BDR roles within the **WAIT** timer, the router declares itself as the DR. The **WAIT** timer is not directly configurable but always equals in value with the **DEAD** interval.

https://t.me/learningnets

In the below case on the LAN segment of VLAN 146, R1, R4, and R6 compete for the DR/BDR election. Without priorities being configured, R6 will become the DR and R4 the BDR, based on the IP addresses of their Loopback0 interfaces. This is based on the fact that the default interface priority is 1, and therefore if all three routers load the OSPF process at the same time, R6 wins the election. By changing R6's interface-level priority to 255, it is most likely to be elected the DR, and changing R4's priority to 0, means it can never be elected the DR or BDR. R6's `show ip ospf neighbor` and `show ip ospf interface` output indicate that R6 is the DR with a priority of 255, R1 is the BDR with a priority of 254, and R4 is a DROTHER because it has a priority of 0. Note also the **WAIT** timer value:

```
R6#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.1.1       254   FULL/BDR
        00:00:34    155.1.146.1     GigabitEthernet1.146150.1.4.4          0   FULL/DROTHER
    00:00:35    155.1.146.4     GigabitEthernet1.146
!
!R6#show ip ospf interface gigabitEthernet1.146
GigabitEthernet1.146 is up, line protocol is up
  Internet Address 155.1.146.6/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.6.6, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses  Transmit Delay is 1 sec,
State DR, Priority 255
 Designated Router (ID) 150.1.6.6, Interface address 155.1.146.6
 Backup Designated router (ID) 150.1.1.1, Interface address 155.1.146.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40
, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 150.1.1.1  (Backup Designated Router)
    Adjacent with neighbor 150.1.4.4
  Suppress hello for 0 neighbor(s)
```

https://t.me/learningnets

```
!
!R4#show ip ospf interface gigabitEthernet1.146 | i Priority
  Transmit Delay is 1 sec, State DROTHER, Priority 0
!
!R1#show ip ospf interface gigabitEthernet1.146 | i Priority
  Transmit Delay is 1 sec, State BDR, Priority 254
```

R6's link to VLAN 146 goes down. After R1's and R4's dead timer expires, a new DR/BDR election occurs. When R1 detects this event, it asks all other neighbors on the segment to perform a new election. Because R4's priority is 0, it does not send a response back to R1. The result is that R1 is promoted from the BDR status to the DR, but there is no new BDR elected:

```
R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface GigabitEthernet1.146
R6(config-subif)#shutdown
!R1#debug ip ospf adj

OSPF adjacency events debugging is on
!
!
OSPF-1 ADJ   Gi1.146: 150.1.6.6 address 155.1.146.6 is dead
OSPF-1 ADJ   Gi1.146: 150.1.6.6 address 155.1.146.6 is dead, state DOWN
%OSPF-5-ADJCHG: Process 1, Nbr 150.1.6.6 on GigabitEthernet1.146 from FULL to DOWN, Neighbor Down: Dead timer expire
OSPF-1 ADJ   Gi1.146: Neighbor change eventOSPF-1 ADJ   Gi1.146: DR/BDR election
OSPF-1 ADJ   Gi1.146: Elect BDR 150.1.1.1
OSPF-1 ADJ   Gi1.146: Elect DR 150.1.1.1
OSPF-1 ADJ   Gi1.146: Elect BDR 0.0.0.0
OSPF-1 ADJ   Gi1.146: Elect DR 150.1.1.1
OSPF-1 ADJ   Gi1.146: DR: 150.1.1.1 (Id)   BDR: none
OSPF-1 ADJ   Gi1.146: Remember old DR 150.1.6.6 (id)
OSPF-1 ADJ   Gi1.146: Neighbor change event
OSPF-1 ADJ   Gi1.146: DR/BDR electionOSPF-1 ADJ   Gi1.146: Elect BDR 0.0.0.0
OSPF-1 ADJ   Gi1.146: Elect DR 150.1.1.1
OSPF-1 ADJ   Gi1.146: DR: 150.1.1.1 (Id)   BDR: none
```

This can also be verified by the **`show ip ospf neighbor`** or **`show ip ospf interface`** output on R1 or R4. R1 sees R4 as a neighbor with a priority of 0, therefore it is a DROTHER:

```
R1#show ip ospf neighbor
```

https://t.me/learningnets

```
Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.4.4         0   FULL/DROTHER
    00:00:35    155.1.146.4     GigabitEthernet1.146
150.1.5.5         1   FULL/DR         00:01:33    155.1.0.5       Tunnel0
```

R4 indicates that there is a DR on the segment with the RID 150.1.1.1, but no BDR:

```
R4#show ip ospf interface gigabitEthernet1.146
GigabitEthernet1.146 is up, line protocol is up
  Internet Address 155.1.146.4/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.4.4, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State DROTHER, Priority 0
Designated Router (ID) 150.1.1.1, Interface address 155.1.146.1
No backup designated router on this network

  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.1.1  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

When R6’s connection to VLAN 146 comes back up, R1 learns about the neighbor and a new election occurs. However, because there is no preemption for the DR election, R6 can only be elected the BDR, even though its priority is higher than R1's:

```
R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface GigabitEthernet1.146
R6(config-subif)#no shutdown
!
!
```

https://t.me/learningnets

```
OSPF-1 ADJ   Gi1.146: 2 Way Communication to 150.1.6.6, state 2WAY
OSPF-1 ADJ   Gi1.146: Neighbor change event OSPF-1 ADJ   Gi1.146: DR/BDR election
OSPF-1 ADJ   Gi1.146: Elect BDR 150.1.6.6
OSPF-1 ADJ   Gi1.146: Elect DR 150.1.1.1
OSPF-1 ADJ   Gi1.146: DR: 150.1.1.1 (Id)   BDR: 150.1.6.6 (Id)
OSPF-1 ADJ   Gi1.146: Nbr 150.1.6.6: Prepare dbase exchange
OSPF-1 ADJ   Gi1.146: Send DBD to 150.1.6.6 seq 0xE89 opt 0x52 flag 0x7 len 32
OSPF-1 ADJ   Gi1.146: Neighbor change event
OSPF-1 ADJ   Gi1.146: DR/BDR election
OSPF-1 ADJ   Gi1.146: Elect BDR 150.1.6.6
OSPF-1 ADJ   Gi1.146: Elect DR 150.1.1.1
OSPF-1 ADJ   Gi1.146: DR: 150.1.1.1 (Id)   BDR: 150.1.6.6 (Id)
OSPF-1 ADJ   Gi1.146: Neighbor change event
OSPF-1 ADJ   Gi1.146: DR/BDR election
OSPF-1 ADJ   Gi1.146: Elect BDR 150.1.6.6
OSPF-1 ADJ   Gi1.146: Elect DR 150.1.1.1 OSPF-1 ADJ   Gi1.146: DR: 150.1.1.1 (Id)   BDR: 150.1.6.6 (Id)
!
!R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.4.4         0   FULL/DROTHER
    00:00:33    155.1.146.4     GigabitEthernet1.146 150.1.6.6        255   FULL/BDR
        00:00:33    155.1.146.6     GigabitEthernet1.146
150.1.5.5         1   FULL/DR         00:01:40    155.1.0.5       Tunnel0
!
!R1#show ip ospf interface gigabitEthernet1.146
GigabitEthernet1.146 is up, line protocol is up
  Internet Address 155.1.146.1/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.1.1, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses  Transmit Delay is 1 sec,
 State DR, Priority 254
 Designated Router (ID) 150.1.1.1, Interface address 155.1.146.1
 Backup Designated router (ID) 150.1.6.6, Interface address 155.1.146.6

  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
```

https://t.me/learningnets

```
  Last flood scan length is 1, maximum is 2
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 150.1.4.4
    Adjacent with neighbor 150.1.6.6  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
```

For LAN segments such as VLAN 146, it technically does not matter which device is the DR or the BDR, because everyone has direct Layer 2 connectivity with each other. The only design issue of the DR/BDR placement in this case is a function of the memory and CPU resources of the DR/BDR and how many neighbors are on the segment. On the DMVPN network between R1, R2, R3, R4, and R5, R5 needs to be the DR so that proper LSA exchange takes place. This is because all OSPF routers reach the OSPF **FULL** state only with the DR and BDR, and remain in the **TWO-WAY** state with all other neighbors on the segment, thus exchange LSAs only with DR/BDR. Because OSPF packets are not functional between spokes, if one spoke becomes the DR, as only the DR floods LSA's on the segment, OSPF database will be broken and routing table incomplete. The priority of all spokes has been lowered to zero so that they cannot become DR, regardless of which routers are first configured on the segment.

Let's leave the spoke with the default priority of one and disable R5's Tunnel0 interface. After the DEAD interval expires, all spokes will declare R5 as DOWN and start DR/BDR election. Basically all spokes will become DR's, as HELLO packets are not ecxhanged between spokes, thus there is no negotiation:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface Tunnel0
R5(config-if)#shutdown
!
!R1#show ip ospf interface tunnel0 | i Desig|State|backup
  Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 150.1.1.1, Interface address 155.1.0.1
  No backup designated router on this network
!
!R2#show ip ospf interface tunnel0 | i Desig|State|backup
  Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 150.1.2.2, Interface address 155.1.0.2
  No backup designated router on this network
!
!R3#show ip ospf interface tunnel0 | i Desig|State|backup
  Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 150.1.3.3, Interface address 155.1.0.3
  No backup designated router on this network
!
```

https://t.me/learningnets

```
!R4#show ip ospf interface tunnel0 | i Desig|State|backup

  Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 150.1.4.4, Interface address 155.1.0.4

  No backup designated router on this network
```

The problem in this design occurs when R5 tries to come back onto the segment. As R5 receives OSPF HELLO packets from all spokes, all claiming to be DR's with the same interface priority, it will finally elect R4 as the DR based on its highest router-ID and itself as the BDR:

https://t.me/learningnets

```
R5#debug ip ospf adj

OSPF adjacency debugging is on
!R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface Tunnel0
R5(config-if)#no shutdown

!
!
OSPF-1 ADJ   Tu0: Route adjust notification: UP/UP
OSPF-1 ADJ   Tu0: Interface going Up
OSPF-1 ADJ   Tu0: Interface state change to UP, new ospf state WAIT
OSPF-1 ADJ   Tu0: 2 Way Communication to 150.1.2.2, state 2WAY

OSPF-1 ADJ   Tu0: Backup seen event before WAIT timer
OSPF-1 ADJ   Tu0: DR/BDR election
OSPF-1 ADJ   Tu0: Elect BDR 150.1.5.5
OSPF-1 ADJ   Tu0: Elect DR 150.1.2.2
OSPF-1 ADJ   Tu0: Elect BDR 150.1.5.5
OSPF-1 ADJ   Tu0: Elect DR 150.1.2.2OSPF-1 ADJ   Tu0: DR: 150.1.2.2 (Id)   BDR: 150.1.5.5 (Id)

!OSPF-1 ADJ   Tu0: 2 Way Communication to 150.1.3.3, state 2WAY

OSPF-1 ADJ   Tu0: Neighbor change event
OSPF-1 ADJ   Tu0: DR/BDR election
OSPF-1 ADJ   Tu0: Elect BDR 150.1.5.5
OSPF-1 ADJ   Tu0: Elect DR 150.1.3.3OSPF-1 ADJ   Tu0: DR: 150.1.3.3 (Id)   BDR: 150.1.5.5 (Id)

!OSPF-1 ADJ   Tu0: 2 Way Communication to 150.1.1.1, state 2WAY

OSPF-1 ADJ   Tu0: Neighbor change event
OSPF-1 ADJ   Tu0: DR/BDR election
OSPF-1 ADJ   Tu0: Elect BDR 150.1.5.5
OSPF-1 ADJ   Tu0: Elect DR 150.1.3.3OSPF-1 ADJ   Tu0: DR: 150.1.3.3 (Id)   BDR: 150.1.5.5 (Id)

!OSPF-1 ADJ   Tu0: 2 Way Communication to 150.1.4.4, state 2WAY

OSPF-1 ADJ   Tu0: Neighbor change event
OSPF-1 ADJ   Tu0: DR/BDR election
OSPF-1 ADJ   Tu0: Elect BDR 150.1.5.5
```

https://t.me/learningnets

```
OSPF-1 ADJ   Tu0: Elect DR 150.1.4.4OSPF-1 ADJ   Tu0: DR: 150.1.4.4 (Id)   BDR: 150.1.5.5 (Id)
```

Verify the OSPF neighbor states over the DMVPN cloud, R4 is the DR and R5 is the BDR:

```
R5#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.1.1         1   FULL/DROTHER
    00:00:35    155.1.0.1       Tunnel0150.1.2.2         1   FULL/DROTHER
    00:00:30    155.1.0.2       Tunnel0150.1.3.3         1   FULL/DROTHER
    00:00:32    155.1.0.3       Tunnel0150.1.4.4         1   FULL/DR
         00:00:34    155.1.0.4       Tunnel0
!
!R1#show ip ospf neighbor tunnel0

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         1   FULL/BDR
        00:00:37    155.1.0.5       Tunnel0
!
!R2#show ip ospf neighbor tunnel0

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         1   FULL/BDR
        00:00:35    155.1.0.5       Tunnel0
!
!R3#show ip ospf neighbor tunnel0

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         1   FULL/BDR
        00:00:32    155.1.0.5       Tunnel0
!
!R4#show ip ospf neighbor tunnel0

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         1   FULL/BDR
        00:00:30    155.1.0.5       Tunnel0
```

Because of the desired function of the DR, the network design is now broken. For example, for R2 to advertise its Loopback0, the LSA Type-1 update must be sent to the DR and BDR, but only the DR can relay the LSA further to the neighbors on the segment. However, R2 does not have a OSPF relation with R4, which is the DR.

https://t.me/learningnets

The result is that LSA Type-1 replication in the DMVPN cloud is now incomplete, and different routers end up with different views of the topology, basically missing parts of it. Based on this inconsistency of information, SPF is not uniform, and some routers have reachability to some segments whereas others do not. The result of this can be seen in either the OSPF database or the routing tables of the devices in question. Verify that R2's Loopback0 is present in the OSPF database of the BDR (R5), and possibly in the OSPF database of the DR (R4) if the entry did not expire from when it was advertised by the old DR (R5), and is not present in the routing table of either routers:

```
R5#show ip ospf database router adv-router 150.1.2.2

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)
 Adv Router is not-reachable in topology Base with MTID 0
  LS age: 764
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.2.2
  Advertising Router: 150.1.2.2
  LS Seq Number: 80000009
  Checksum: 0xF826
  Length: 48
  Number of Links: 2

    Link connected to: a Stub Network (Link ID) Network/subnet number: 150.1.2.2
 (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.0.2
     (Link Data) Router Interface address: 155.1.0.2
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
!
!R5#show ip route 150.1.2.2
% Subnet not in table
!
!R4#show ip ospf database router adv-router 150.1.2.2

            OSPF Router with ID (150.1.4.4) (Process ID 1)

                Router Link States (Area 0)
```

https://t.me/learningnets

```
 Adv Router is not-reachable in topology Base with MTID 0
  LS age: 977
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.2.2 Advertising Router: 150.1.2.2
  LS Seq Number: 80000009
  Checksum: 0xF826
  Length: 48
  Number of Links: 2

    Link connected to: a Stub Network (Link ID) Network/subnet number: 150.1.2.2
 (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.0.2
     (Link Data) Router Interface address: 155.1.0.2
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
!
!R4#show ip route 150.1.2.2
% Subnet not in table
```

The incomplete database view of the routers is clear through the message **Adv Router is not-reachable in topology Base** from the above outputs. What this means is that the DR of the segment, which now is R4, did not advertise in its LSA Type-2 that R2 is a router on the segment, and this is because R2 is not OSPF neighbors with R4:

```
R4#show ip ospf database network adv-router 150.1.4.4

            OSPF Router with ID (150.1.4.4) (Process ID 1)

                Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1366
  Options: (No TOS-capability, DC)
  LS Type: Network Links Link State ID: 155.1.0.4 (address of Designated Router)
  Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0x7B3D
  Length: 32
```

https://t.me/learningnets

```
Network Mask: /24
 Attached Router: 150.1.4.4
Attached Router: 150.1.5.5
```

Because R5 is the oly router which has OSPF adjacency with the DR (R4), R4 and R5 will install a route for each others Loopback0 prefixes, as the OSPF database is complete for LSA Type-1 generated by these routers:

```
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "ospf 1", distance 110, metric 1001, type intra area
  Last update from 155.1.0.4 on Tunnel0, 00:28:44 ago
  Routing Descriptor Blocks: * 155.1.0.4, from 150.1.4.4, 00:28:44 ago, via Tunnel0
      Route metric is 1001, traffic share count is 1
!
!R4#show ip route 150.1.5.5
Routing entry for 150.1.5.5/32
  Known via "ospf 1", distance 110, metric 1001, type intra area
  Last update from 155.1.0.5 on Tunnel0, 00:00:05 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:00:05 ago, via Tunnel0

      Route metric is 1001, traffic share count is 1
```

The process continues as such depending on where the OSPF topology is viewed from. Therefore, at this point in the configuration, the network design works in some cases but not all cases. To ensure that the network works in all cases, we must guarantee that R5 is always elected the DR. This is because R5 is the only neighbor that can form a direct adjacency with all other devices in the DMVPN cloud. The logic of the solution, in this particular case, is somewhat backward because of how OSPF deals with preemption. Even if R5 is configured with a priority value of 255, it cannot preempt whichever router elected itself as the DR after it went down. The only thing R5 can do is re-elect itself as the BDR. Therefore, instead of trying to prefer R5 become the DR by raising its interface priority to maximum, we must ensure that R1, R2, R3, and R4 are not elected the DR. This is accomplished by configuring the OSPF priority value as 0 at the interface level of each of these devices. Because priority 0 means they will not participate in the election, R5 is the only candidate that can be elected the DR and, additionally, no one is eligible to be elected the BDR. If R5 goes down, all routing information is lost, but when R5 comes back it is fully restored. Verification of this can be seen in the `show ip ospf neighbor` output on R5. With the remote devices configured with priority 0, R5 is elected the DR, and all other devices revert back to DROTHERs:

https://t.me/learningnets

```
R5#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.1.1         0   FULL/DROTHER
    00:00:35    155.1.0.1       Tunnel0 150.1.2.2         0   FULL/DROTHER
    00:00:37    155.1.0.2       Tunnel0 150.1.3.3         0   FULL/DROTHER
    00:00:38    155.1.0.3       Tunnel0 150.1.4.4         0   FULL/DROTHER
    00:00:37    155.1.0.4       Tunnel0
```

When the correct configuration is complete, it can also be verified by the **`show ip ospf database`** output. Only the DR on an OSPF segment originates the Network LSA (LSA 2). The Network LSA is used to describe all the neighbors that the DR on the segment is adjacent with to the BDR and DROTHERs on that link:

```
R5#show ip ospf database network adv-router 150.1.5.5


            OSPF Router with ID (150.1.5.5) (Process ID 1)


                Net Link States (Area 0)


  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 130
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.0.5 (address of Designated Router)
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0x7B5B
  Length: 44 Network Mask: /24
 Attached Router: 150.1.5.5
 Attached Router: 150.1.1.1
 Attached Router: 150.1.2.2
 Attached Router: 150.1.3.3
 Attached Router: 150.1.4.4
```

Verify the LSA Type-1 generated by any routers on the DMVPN cloud, note that for the DMVPN segment where DR/BDR election takes place, the network masks is not advertised, the network mask being advertised by the DR in LSA Type-2:

```
R5#show ip ospf database router adv-router 150.1.5.5


            OSPF Router with ID (150.1.5.5) (Process ID 1)
```

https://t.me/learningnets

```
                Router Link States (Area 0)

  LS age: 12
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5 Advertising Router: 150.1.5.5
  LS Seq Number: 80000010
  Checksum: 0x4EB1
  Length: 48 Number of Links: 2

    Link connected to: a Stub Network (Link ID) Network/subnet number: 150.1.5.5
 (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.0.5 (Link Data) Router Interface address: 155.1.0.5

      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
```

Verify that now, when DR election is correct, OSPF enabled interfaces are advertised and learned, thus prefixes show up in the routing table:

```
R5#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets
O        150.1.1.1 [110/1001] via 155.1.0.1, 00:05:18, Tunnel0
O        150.1.2.2 [110/1001] via 155.1.0.2, 00:05:18, Tunnel0
O        150.1.3.3 [110/1001] via 155.1.0.3, 00:05:18, Tunnel0
O        150.1.4.4 [110/1001] via 155.1.0.4, 00:39:15, Tunnel0
      155.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
O        155.1.146.0/24 [110/1001] via 155.1.0.4, 00:39:15, Tunnel0
```

https://t.me/learningnets

```
                        [110/1001] via 155.1.0.1, 00:05:18, Tunnel0
!
!R2#show ip route ospf

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 6 subnets
O        150.1.1.1 [110/1001] via 155.1.0.1, 00:00:03, Tunnel0
O        150.1.3.3 [110/1001] via 155.1.0.3, 00:00:03, Tunnel0
O        150.1.4.4 [110/1001] via 155.1.0.4, 00:00:03, Tunnel0
O        150.1.5.5 [110/1001] via 155.1.0.5, 00:00:03, Tunnel0
O        150.1.6.6 [110/1002] via 155.1.0.4, 00:00:03, Tunnel0
                   [110/1002] via 155.1.0.1, 00:00:03, Tunnel0
      155.1.0.0/16 is variably subnetted, 5 subnets, 2 masks
O        155.1.146.0/24 [110/1001] via 155.1.0.4, 00:00:03, Tunnel0
                        [110/1001] via 155.1.0.1, 00:00:03, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Network Point-to-Point

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF in area 0 between R6, R7 and R9.
  - Use a OSPF network type of **point-to-point**.
- Advertise Loopback0 prefixes in area 0.

Configuration

```
R6:
interface GigabitEthernet1.67
 ip ospf 1 area 0
 ip ospf network point-to-point
!
interface Loopback0
 ip ospf 1 area 0
R7:
interface GigabitEthernet1.67
 ip ospf 1 area 0
 ip ospf network point-to-point
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
 ip ospf network point-to-point
!
interface Loopback0
 ip ospf 1 area 0
```

https://t.me/learningnets

```
R9:

interface GigabitEthernet1.79
 ip ospf 1 area 0
 ip ospf network point-to-point
!
interface Loopback0
 ip ospf 1 area 0
```

Verification

The Default OSPF network on Ethernet interfaces is **Broadcast**. OSPF network **point-to-point** is the default option for point-to-point interfaces such as HDLC, PPP, or point-to-point GRE tunnels. It uses multicast hellos, does not use the DR/BDR election, and only supports the adjacency of exactly two neighbors on a segment. No special design considerations need be accounted for with point-to-point OSPF interfaces. The default OSPF network type can be verified by using the `show ip ospf interface` command:

```
R6#show ip ospf interface gigabitEthernet1.67
GigabitEthernet1.67 is up, line protocol is up
  Internet Address 155.1.67.6/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.6.6, Network Type BROADCAST
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State WAITING, Priority 1
  No designated router on this network
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
    Wait time before Designated router selection 00:00:35
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 0
```

https://t.me/learningnets

```
  Suppress hello for 0 neighbor(s)
```

After changing the OSPF Network type to Point-to-Point:

```
R6#show ip ospf interface gigabitEthernet1.67
GigabitEthernet1.67 is up, line protocol is up
  Internet Address 155.1.67.6/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.6.6, Network Type POINT_TO_POINT
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.7.7
  Suppress hello for 0 neighbor(s)
!
!R6#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               150.1.6.6/32       1     LOOP  0/0 Gi1.67
       1     0               155.1.67.6/24      1 P2P
   1/1
```

The null output after the slash in the **State** column of the `show ip ospf neighbor` output indicates that no DR or BDR is elected for network type point-to-point:

```
R7#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.6.6         0   FULL/  -
        00:00:38    155.1.67.6      GigabitEthernet1.67 150.1.9.9         0   FULL/  -
        00:00:39    155.1.79.9      GigabitEthernet1.79
!
```

https://t.me/learningnets

```
!R7#show ip ospf neighbor detail
 Neighbor 150.1.6.6, interface address 155.1.67.6
    In the area 0 via interface GigabitEthernet1.67
    Neighbor priority is 0, State is FULL, 18 state changes DR is 0.0.0.0 BDR is 0.0.0.0
    Options is 0x12 in Hello (E-bit, L-bit)
    Options is 0x52 in DBD (E-bit, L-bit, O-bit)
    LLS Options is 0x1 (LR)
    Dead timer due in 00:00:38
    Neighbor is up for 00:02:02
    Index 1/1, retransmission queue length 0, number of retransmission 0
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 0, maximum is 0
    Last retransmission scan time is 0 msec, maximum is 0 msec
 Neighbor 150.1.9.9, interface address 155.1.79.9
    In the area 0 via interface GigabitEthernet1.79
    Neighbor priority is 0, State is FULL, 6 state changes DR is 0.0.0.0 BDR is 0.0.0.0

    Options is 0x12 in Hello (E-bit, L-bit)
    Options is 0x52 in DBD (E-bit, L-bit, O-bit)
    LLS Options is 0x1 (LR)
    Dead timer due in 00:00:39
    Neighbor is up for 00:04:35
    Index 2/2, retransmission queue length 0, number of retransmission 2
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 1, maximum is 1
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

The lack of DR/BDR can also be observer due to missing of Type2 LSA in the OSPF database:

```
R7#show ip ospf database

            OSPF Router with ID (150.1.7.7) (Process ID 1)
 Router Link States (Area 0)


Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.6.6       150.1.6.6       173         0x80000006 0x002195 3
150.1.7.7       150.1.7.7       177         0x80000005 0x00F415 5
150.1.9.9       150.1.9.9       251         0x80000003 0x000389 3
```

Note that for **point-to-point** OSPF network types, the Type1 LSA contains both the network segment but also information about the neighbor OSPF router, which is required for a complete database view. For example R6 LSA Type1 has information

https://t.me/learningnets

about its neighbor R7 on the **point-to-point** link:

```
R6#show ip ospf database router self-originate

            OSPF Router with ID (150.1.6.6) (Process ID 1)

                Router Link States (Area 0)

  LS age: 280
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.6.6
  Advertising Router: 150.1.6.6
  LS Seq Number: 80000006
  Checksum: 0x2195
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.6.6
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.7.7
 (Link Data) Router Interface address: 155.1.67.6
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Stub Network
 (Link ID) Network/subnet number: 155.1.67.0
 (Link Data) Network Mask: 255.255.255.0

      Number of MTID metrics: 0
       TOS 0 Metrics: 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Network Point-to-Multipoint

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF in area 12345 over the DMVPN cloud, between R1 - R5.
  - Use a OSPF network type of **point-to-multipoint**.
  - Advertise Loopback0 prefixes in area 12345.
- Configure OSPF in area 0 between R5 and R8.

Configuration

```
R1 - R5:
interface Tunnel0
 ip ospf 1 area 12345
 ip ospf network point-to-multipoint
!
interface Loopback0
 ip ospf 1 area 12345
R5:
interface GigabitEthernet1.58
 ip ospf 1 area 0
R8:

interface GigabitEthernet1.58
 ip ospf 1 area 0
```

https://t.me/learningnets

Verification

OSPF network type point-to-multipoint is specifically designed to solve reachability problems in partially meshed NBMA network designs. Like network type point-to-point, it sends hellos as multicasts and does not support the DR/BDR election. Unlike point-to-point, however, multiple adjacencies on a single interface are supported. When adjacency is established, the `show ip ospf neighbor` output indicates that there is no DR or BDR for the segment with the null output in the **State** field:

```
R5#show ip ospf neighbor tunnel0

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.4.4         0   FULL/  -
        00:01:41    155.1.0.4       Tunnel0 150.1.3.3         0   FULL/  -
        00:01:36    155.1.0.3       Tunnel0 150.1.2.2         0   FULL/  -
        00:01:34    155.1.0.2       Tunnel0 150.1.1.1         0   FULL/  -
        00:01:56    155.1.0.1       Tunnel0
```

Verify the OSPF network type used:

```
R5#show ip ospf interface tunnel0
Tunnel0 is up, line protocol is up
  Internet Address 155.1.0.5/24, Area 12345, Attached via Interface Enable
  Process ID 1, Router ID 150.1.5.5, Network Type POINT_TO_MULTIPOINT
, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:14
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 7
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 4, Adjacent neighbor count is 4
```

https://t.me/learningnets

```
    Adjacent with neighbor 150.1.4.4
    Adjacent with neighbor 150.1.3.3
    Adjacent with neighbor 150.1.2.2
    Adjacent with neighbor 150.1.1.1
  Suppress hello for 0 neighbor(s)
!
!R5#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Gi1.58       1     0               155.1.58.5/24      1     BDR   1/1
Lo0          1     12345           150.1.5.5/32       1     LOOP  0/0 Tu0
          1     12345           155.1.0.5/24       1000 P2MP
  4/4
```

The spokes of the network only form adjacency with R5, and not each other, because multicast OSPF packets are not sent between spokes:

```
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:33    155.1.0.5       Tunnel0
!
!R2#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:50    155.1.0.5       Tunnel0
!
!R3#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:42    155.1.0.5       Tunnel0
!
!R4#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:34    155.1.0.5       Tunnel0
```

https://t.me/learningnets

The output from the `show ip ospf database` in area 12345 also reinforces the fact that there is no DR or BDR for the segment. Recall from the DR/BDR election section that the Network LSA (LSA 2) is used to describe the DR and its attached neighbors. Because there are no DRs for any transit links in area 12345, the **Net Link States (Area 12345)** section does not appear in the below output for area 12345:

```
R5#show ip ospf database

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.5.5       150.1.5.5       53          0x80000002 0x00D959 1
150.1.8.8       150.1.8.8       54          0x80000002 0x009A8A 1

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.58.8      150.1.8.8       54          0x80000001 0x00EA7F

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
150.1.1.1       150.1.5.5       97          0x80000001 0x006CA8
150.1.2.2       150.1.5.5       97          0x80000001 0x0057BB
150.1.3.3       150.1.5.5       97          0x80000001 0x0042CE
150.1.4.4       150.1.5.5       97          0x80000001 0x002DE1
150.1.5.5       150.1.5.5       97          0x80000001 0x00E315
155.1.0.1       150.1.5.5       97          0x80000001 0x002CE5
155.1.0.2       150.1.5.5       97          0x80000001 0x0022EE
155.1.0.3       150.1.5.5       97          0x80000001 0x0018F7
155.1.0.4       150.1.5.5       97          0x80000001 0x000E01
155.1.0.5       150.1.5.5       97          0x80000001 0x00CF2A
 Router Link States (Area 12345)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.1.1       150.1.1.1       1780        0x80000001 0x00DCA1 3
150.1.2.2       150.1.2.2       1776        0x80000001 0x000A6C 3
150.1.3.3       150.1.3.3       1772        0x80000001 0x003737 3
150.1.4.4       150.1.4.4       1769        0x80000001 0x006402 3
150.1.5.5       150.1.5.5       97          0x80000008 0x004969 6
150.1.8.8       150.1.8.8       480         0x80000002 0x009A8A 1
 Summary Net Link States (Area 12345)
```

https://t.me/learningnets

```
Link ID         ADV Router      Age         Seq#       Checksum
155.1.58.0      150.1.5.5       92          0x80000001 0x008B38
```

When LSA replication is complete, the next change that should be evident is how the routing table is processed on the DMVPN segment between the neighbors. For example, take the following view of the network from R1's perspective:

```
R1#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets O        150.1.2.2 [110/2001] via 155.1.0.5,
 00:08:42, Tunnel0 O        150.1.3.3 [110/2001] via 155.1.0.5,
 00:08:42, Tunnel0 O        150.1.4.4 [110/2001] via 155.1.0.5,
 00:08:32, Tunnel0 O        150.1.5.5 [110/1001] via 155.1.0.5,
 00:08:52, Tunnel0
      155.1.0.0/16 is variably subnetted, 10 subnets, 2 masks O        155.1.0.2/32 [110/2000]
via 155.1.0.5,
 00:08:42, Tunnel0 O        155.1.0.3/32 [110/2000] via 155.1.0.5,
 00:08:42, Tunnel0 O        155.1.0.4/32 [110/2000] via 155.1.0.5,
 00:08:32, Tunnel0 O        155.1.0.5/32 [110/1000] via 155.1.0.5,
 00:08:52, Tunnel0 O IA     155.1.58.0/24 [110/1001] via 155.1.0.5,
 00:02:43, Tunnel0
```

For all routes that are learned from the hub of the DMVPN network, the next-hop value has been updated to the interface IP address of the hub. With the broadcast network type, the DR does not update the next-hop. The result of this is that when route recursion is performed for traffic that must be routed between the spokes, NHRP resolution needs to be performed for the hub, not for the spoke (hub NHRP entry is statically configured on all spokes):

```
R1#show ip route 150.1.2.2
Routing entry for 150.1.2.2/32
  Known via "ospf 1", distance 110, metric 2001, type intra area
```

https://t.me/learningnets

```
  Last update from 155.1.0.5 on Tunnel0, 00:14:11 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.2.2, 00:14:11 ago, via Tunnel0
      Route metric is 2001, traffic share count is 1
!
!R1#show ip route 155.1.0.5
Routing entry for 155.1.0.5/32
  Known via "ospf 1", distance 110, metric 1000, type intra area
  Last update from 155.1.0.5 on Tunnel0, 00:14:34 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:14:34 ago, via Tunnel0
      Route metric is 1000, traffic share count is 1
!
!R1#show ip nhrp 155.1.0.5
155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 1d01h, never expire Type: static
, Flags: used NBMA address: 169.254.100.5
!
!R1#show ip cef 150.1.2.2 internal
150.1.2.2/32, epoch 2, RIB[I], refcount 6, per-destination sharing
  sources: RIB
  feature space:
   IPRM: 0x00028000
   Broker: linked, distributed at 4th priority
  ifnums: Tunnel0(15): 155.1.0.5
  path 7F01B9FAF1F8, path list 7F01B9F2C488, share 1/1, type attached nexthop, for IPv4
  nexthop 155.1.0.5 Tunnel0, adjacency IP midchain out of Tunnel0, addr 155.1.0.5 7F01B9F6F880
  output chain: IP midchain out of Tunnel0, addr 155.1.0.5 7F01B9F6F880
IP adj out of GigabitEthernet1.100, addr 169.254.100.5
 7F01B9F6E200
!
!R1#ping 150.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
```

From views outside of the connected segment, the point-to-multipoint network is only seen as a collection of endpoints, not a transit segment itself. For example, review the following output on R8, which is in area 0:

```
R8#show ip route 155.1.0.0 255.255.255.0
% Subnet not in table
!
!R8#show ip route ospf | i 155.1.0.
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks O IA 155.1.0.1/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58
```

https://t.me/learningnets

```
O IA 155.1.0.2/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58 O IA 155.1.0.3/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58 O IA 155.1.0.4/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58 O IA 155.1.0.5/32
 [110/1] via 155.1.58.5, 00:02:58, GigabitEthernet1.58
```

R8 does not have an exact match for the prefix 155.1.0.0/24, which is the actual subnet assignment of the DMVPN Network. Instead, it knows that there are five endpoints on the network: 155.1.0.1, 155.1.0.2, 155.1.0.3, 155.1.0.4, and 155.1.0.5. This is the normal and desirable behavior for OSPF network type point-to-multipoint, according to the RFC standard. This can also be viewed at the database level, where each router in the DMVPN cloud, in its Type1 LSA, it advertises its own IP address with a /32 mask and additionally it advertises its OSPF neighbors on the segment which is required for a complete OSPF database view:

```
R1#show ip ospf database router self-originate

            OSPF Router with ID (150.1.1.1) (Process ID 1)
 Router Link States (Area 12345)

  LS age: 2019
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.1.1
  Advertising Router: 150.1.1.1
  LS Seq Number: 80000001
  Checksum: 0xDCA1
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.1.1
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.5.5
 (Link Data) Router Interface address: 155.1.0.1
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
 Link connected to: a Stub Network
 (Link ID) Network/subnet number: 155.1.0.1
 (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

https://t.me/learningnets

```
!
!R5#show ip ospf database router self-originate

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

  LS age: 355
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000002
  Checksum: 0xD959
  Length: 36
  Area Border Router
  Number of Links: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.58.8
     (Link Data) Router Interface address: 155.1.58.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1


Router Link States (Area 12345)

  LS age: 398
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000008
  Checksum: 0x4969
  Length: 96
  Area Border Router
  Number of Links: 6

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 150.1.4.4
```

https://t.me/learningnets

```
(Link Data) Router Interface address: 155.1.0.5

      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 150.1.3.3
(Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 150.1.2.2
(Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 150.1.1.1
(Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
 Link connected to: a Stub Network
(Link ID) Network/subnet number: 155.1.0.5
(Link Data) Network Mask: 255.255.255.255

      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Network Point-to-Multipoint Non-Broadcast

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF in area 12345 over the DMVPN cloud, between R1 - R5.
  - Use a OSPF network type of **point-to-multipoint non-broadcast**.
  - Advertise Loopback0 prefixes in area 12345.
- Configure OSPF in area 0 between R5 and R8.

Configuration

```
R1 - R5:
interface Tunnel0
 ip ospf 1 area 12345
 ip ospf network point-to-multipoint non-broadcast
!
interface Loopback0
 ip ospf 1 area 12345
R5:
router ospf 1
 neighbor 155.1.0.1
 neighbor 155.1.0.2
 neighbor 155.1.0.3
 neighbor 155.1.0.4
!
interface GigabitEthernet1.58
 ip ospf 1 area 0
R8:
```

https://t.me/learningnets

```
interface GigabitEthernet1.58
 ip ospf 1 area 0
```

Verification

OSPF network type point-to-multipoint non-broadcast is essentially the same as network type point-to-multipoint, with one exception. Point-to-multipoint network type uses multicast hellos, whereas point-to-multipoint non-broadcast uses unicast hellos. Additionally, in non-broadcast mode, you can configure per-neighbor OSPF cost using command `neighbor <IP_ADDRESS> cost <value>` , which is helpful in hub-and-spoke topology, allowing the hub to use different costs per spoke, although all spokes are attached to the same interface from hub perspective. Both do not support the DR/BDR election, automatically update the next-hop value of routes learned on partially meshed networks to the directly connected neighbor, and advertise the network as a set of endpoints instead of a transit network. The `show ip ospf neighbor` output is identical between the two network types. In this case, we can see that the spokes are adjacent with R5, the hub, and the hub is adjacent with the spokes. The null field under **State** field indicates that no DR/BDR election has occurred:

https://t.me/learningnets

```
R1#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:53    155.1.0.5       Tunnel0

!
!R2#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:42    155.1.0.5       Tunnel0

!
!R3#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:58    155.1.0.5       Tunnel0

!
!R4#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.5.5         0   FULL/  -        00:01:49    155.1.0.5       Tunnel0

!
!R5#show ip ospf neighbor tunnel0


Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.4.4         0   FULL/  -
        00:01:56    155.1.0.4       Tunnel0 150.1.3.3         0   FULL/  -
        00:01:56    155.1.0.3       Tunnel0 150.1.2.2         0   FULL/  -
        00:01:56    155.1.0.2       Tunnel0 150.1.1.1         0   FULL/  -
        00:01:56    155.1.0.1       Tunnel0
```

Verify the change is OSPF network-type, note that it shows up as **point-to-multipoint**, which can be both broadcast and non-broadcast, there is no clear differentiation in the outputs between the two:

```
R5#show ip ospf interface tunnel0
```

https://t.me/learningnets

```
Tunnel0 is up, line protocol is up
  Internet Address 155.1.0.5/24, Area 12345, Attached via Interface Enable
  Process ID 1, Router ID 150.1.5.5, Network Type POINT_TO_MULTIPOINT
, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:13
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 4, maximum is 7
  Last flood scan time is 1 msec, maximum is 1 msec
  Neighbor Count is 4, Adjacent neighbor count is 4
    Adjacent with neighbor 150.1.4.4
    Adjacent with neighbor 150.1.3.3
    Adjacent with neighbor 150.1.2.2
    Adjacent with neighbor 150.1.1.1
  Suppress hello for 0 neighbor(s)
!
!R5#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Gi1.58       1     0               155.1.58.5/24      1     BDR   1/1
Lo0          1     12345           150.1.5.5/32       1     LOOP  0/0 Tu0
          1     12345           155.1.0.5/24       1000 P2MP
  4/4
```

The routing table processing is the same between both network types. For example, take R1 which has a next-hop of the hub for all OSPF learned routes:

```
R1#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
```

https://t.me/learningnets

```
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets O        150.1.2.2 [110/2001] via 155.1.0.5,
 00:08:33, Tunnel0 O        150.1.3.3 [110/2001] via 155.1.0.5,
 00:08:33, Tunnel0 O        150.1.4.4 [110/2001] via 155.1.0.5,
 00:08:33, Tunnel0 O        150.1.5.5 [110/1001] via 155.1.0.5,
 00:08:43, Tunnel0
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks O        155.1.0.2/32 [110/2000]
via 155.1.0.5,
 00:08:33, Tunnel0 O        155.1.0.3/32 [110/2000] via 155.1.0.5,
 00:08:33, Tunnel0 O        155.1.0.4/32 [110/2000] via 155.1.0.5,
 00:08:33, Tunnel0 O        155.1.0.5/32 [110/1000] via 155.1.0.5,
 00:08:43, Tunnel0 O IA     155.1.58.0/24 [110/1001] via 155.1.0.5,
 00:08:43, Tunnel0
```

Entries in the OSPF database and LSA contents are the same as in point-to-multipoint case, for example R1:

```
R1#show ip ospf database router self-originate

            OSPF Router with ID (150.1.1.1) (Process ID 1)

                Router Link States (Area 12345)

  LS age: 793
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.1.1
  Advertising Router: 150.1.1.1
  LS Seq Number: 80000004
  Checksum: 0xD6A4
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.1.1
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.5.5
```

https://t.me/learningnets

```
(Link Data) Router Interface address: 155.1.0.1

      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
 Link connected to: a Stub Network
 (Link ID) Network/subnet number: 155.1.0.1
 (Link Data) Network Mask: 255.255.255.255

      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

Like point-to-multipoint, when neighbors outside area 12345 see this segment, it appears as a collection of endpoints, not the transit subnet itself:

```
R8#show ip route 155.1.0.0 255.255.255.0
% Subnet not in table
!
!R8#show ip route ospf | i 155.1.0.
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks O IA 155.1.0.1/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58 O IA 155.1.0.2/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58 O IA 155.1.0.3/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58 O IA 155.1.0.4/32
 [110/1001] via 155.1.58.5, 00:02:58, GigabitEthernet1.58 O IA 155.1.0.5/32
 [110/1] via 155.1.58.5, 00:02:58, GigabitEthernet1.58
```

The difference between the two types can be seen in the `debug ip packet` output. R5 sends multicast hellos to 224.0.0.5 out its LAN interface to R8 that uses network type broadcast. Out the DMVPN link running point-to-multipoint non-broadcast, R5 sends unicast hellos. This implies that the neighbor statement must be configured under the OSPF process, like the non-broadcast network type, to tell the router which devices to send hellos to:

```
R5#debug ip ospf hello
OSPF hello debugging is on
!
!
OSPF-1 HELLO Gi1.58: Rcv hello from 150.1.8.8 area 0 155.1.58.8 OSPF-1 HELLO
Gi1.58: Send hello to 224.0.0.5 area 0 from 155.1.58.5
! OSPF-1 HELLO Tu0: Send hello to 155.1.0.4 area 12345 from 155.1.0.5
OSPF-1 HELLO Tu0: Send hello to 155.1.0.3 area 12345 from 155.1.0.5
OSPF-1 HELLO Tu0: Send hello to 155.1.0.2 area 12345 from 155.1.0.5
OSPF-1 HELLO Tu0: Send hello to 155.1.0.1 area 12345 from 155.1.0.5

OSPF-1 HELLO Tu0: Rcv hello from 150.1.3.3 area 12345 155.1.0.3
```

https://t.me/learningnets

```
OSPF-1 HELLO Tu0: Rcv hello from 150.1.1.1 area 12345 155.1.0.1
OSPF-1 HELLO Tu0: Rcv hello from 150.1.2.2 area 12345 155.1.0.2
OSPF-1 HELLO Tu0: Rcv hello from 150.1.4.4 area 12345 155.1.0.4
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Network Loopback

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Connfigure OSPF in area 0 between R7 and R9.
- Create Loopback160 on R7 and R9 with IP addressing in the format of **160.1.Y.Y/24**, where **Y** is the router number.
  - Advertise these Loopbacks into OSPF area 0.
- Create Loopback170 on R7 and R9 with IP addressing in the format of **170.1.Y.Y/24**, where **Y** is the router number.
  - Advertise these Loopbacks into OSPF area 0.
  - Use a OSPF network type of **point-to-point**.

Configuration

```
R7:
interface GigabitEthernet1.79
 ip ospf 1 area 0
!
interface Loopback160
 ip address 160.1.7.7 255.255.255.0
 ip ospf 1 area 0
!
interface Loopback170
 ip address 170.1.7.7 255.255.255.0
 ip ospf 1 area 0
 ip ospf network point-to-point
```

https://t.me/learningnets

```
R9:

interface GigabitEthernet1.79
 ip ospf 1 area 0
!
interface Loopback160
 ip address 160.1.9.9 255.255.255.0
 ip ospf 1 area 0
!
interface Loopback170
 ip address 170.1.9.9 255.255.255.0
 ip ospf 1 area 0
 ip ospf network point-to-point
```

Verification

OSPF network type Loopback is a special case that is used for Loopback and looped-back interfaces. Similar to point-to-multipoint, an interface running network type Loopback is advertised as a stub endpoint instead of a transit subnet. The result of this network type that we see in our configuration is that when the Loopback160 interfaces of these devices are advertised into OSPF, they appear in the routing table as /32 host routes instead of the actual subnet mask of /24. Because network type for Loopback170 interfaces has been changed, these are correctly advertised with their subnet mask:

```
R7#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      160.1.0.0/16 is variably subnetted, 3 subnets, 2 masks O        160.1.9.9/32
 [110/2] via 155.1.79.9, 00:00:56, GigabitEthernet1.79
      170.1.0.0/16 is variably subnetted, 3 subnets, 2 masks O        170.1.9.0/24
 [110/2] via 155.1.79.9, 00:00:56, GigabitEthernet1.79
!
!R9#show ip route ospf
```

https://t.me/learningnets

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      160.1.0.0/16 is variably subnetted, 3 subnets, 2 masks O        160.1.7.7/32
 [110/2] via 155.1.79.7, 00:01:47, GigabitEthernet1.79
      170.1.0.0/16 is variably subnetted, 3 subnets, 2 masks O        170.1.7.0/24
 [110/2] via 155.1.79.7, 00:01:47, GigabitEthernet1.79
```

Verify how these Loopbacks appear in th OPSF database, for example on R7:

```
R7#show ip ospf database router self-originate

            OSPF Router with ID (170.1.7.7) (Process ID 1)

                Router Link States (Area 0)

  LS age: 41
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 170.1.7.7
  Advertising Router: 170.1.7.7
  LS Seq Number: 80000004
  Checksum: 0xFF50
  Length: 60
  Number of Links: 3
 Link connected to: a Stub Network
 (Link ID) Network/subnet number: 160.1.7.7
 (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Stub Network
 (Link ID) Network/subnet number: 170.1.7.0
 (Link Data) Network Mask: 255.255.255.0

      Number of MTID metrics: 0
       TOS 0 Metrics: 1
```

https://t.me/learningnets

```
    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.79.9
     (Link Data) Router Interface address: 155.1.79.7
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
```

Verify the OSPF network types for the two Loopbacks:

```
R7#show ip ospf interface loopback160
Loopback160 is up, line protocol is up
  Internet Address 160.1.7.7/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 170.1.7.7, Network Type LOOPBACK
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
Loopback interface is treated as a stub Host
!
!R7#show ip ospf interface loopback170
Loopback170 is up, line protocol is up
  Internet Address 170.1.7.7/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 170.1.7.7, Network Type POINT_TO_POINT
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
!
!R7#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Lo160
        1     0               160.1.7.7/24       1 LOOP
  0/0 Lo170        1     0               170.1.7.7/24       1 P2P
    0/0
```

https://t.me/learningnets

```
Gi1.79       1     0               155.1.79.7/24      1     BDR   1/1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Path Selection with Auto-Cost

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Modify the global OSPF cost calculation of all devices so that a TenGigabit Ethernet interface has a cost of 3, and an OC-3 link has a cost of 193.

Configuration

```
R1 – R10:

router ospf 1
 auto-cost reference-bandwidth 30000
```

Pitfall

Technically, OSPF auto-cost reference-bandwidth does not need to match for neighbors to form an adjacency. However, if different neighbors throughout the topology calculate SPF based on conflicting cost calculations, loops can occur. For this reason, it is always recommended to set the auto-cost value consistently throughout the entire OSPF domain.

Verification

OSPF automatic cost calculation is an inverse function of the bandwidth of an

https://t.me/learningnets

interface, meaning the higher the bandwidth value of an interface, the lower the cost value. Specifically, the formula for this calculation is:

```
Interface Cost = Reference Bandwidth / Interface Bandwidth
```

The issue with this calculation in current practical network designs is that the default reference bandwidth value used is 100Mbps. This means that a 100Mbps Fast Ethernet interface will have a cost of 1, and furthermore all interfaces with higher bandwidth values are also 1. The result of this is that OSPF routers in the topology cannot make an accurate path calculation when comparing an OC-48 POS interface to a Gigabit Ethernet interface, because both interfaces revert to a cost of 1. To resolve this, the reference bandwidth value is typically modified to allow these higher-bandwidth interfaces to have separate, more granular cost values. In this particular case, a Ten Gigabit Ethernet interface or an OC-192 POS interface are asked to have a cost value of 3, whereas a 155Mbps OC-3 POS/ATM interface is asked to have a cost value of 193.

Verify the OSPF cost value for a GigabitEthernet interface and Reference Bandwidth value before configuration changes are applied:

```
R1#show ip ospf interface gigabitEthernet1.13 | include Cost
  Process ID 1, Router ID 150.1.1.1, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
!
!R1#show ip ospf | section Reference
 Reference bandwidth unit is 100 mbps


    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
```

Verify the OSPF cost value for a GigabitEthernet interface and Reference Bandwidth value after configuration changes are applied:

https://t.me/learningnets

```
R1#show ip ospf interface gigabitEthernet1.13 | include Cost
  Process ID 1, Router ID 150.1.1.1, Network Type BROADCAST, Cost: 30

  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
!
!R1#show ip ospf | section Reference
                         Reference bandwidth unit is 30000 mbps

    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
```

For tasks like this, it is not necessary to memorize the formula for cost calculation; just to know how to derive it using various show outputs on the IOS CLI. Because the OSPF cost is derived from the bandwidth, changing the interface **`bandwidth`** value should change the cost:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface GigabitEthernet1.13
R1(config-subif)#bandwidth 10000000
!R1#show ip ospf interface gigabitEthernet1.13 | include Cost
  Process ID 1, Router ID 150.1.1.1, Network Type BROADCAST, Cost: 3

  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
```

A reference bandwidth of 30.000 results in a cost of 3 (30000 reference bw/ 10000 interface bw = 3). This calculation then stays true for 155Mbps OC-3:

```
R1(config)#interface gigabitEthernet 1.13
R1(config-subif)#bandwidth 155000
!R1#show ip ospf interface gigabitEthernet1.13 | include Cost
  Process ID 1, Router ID 150.1.1.1, Network Type BROADCAST, Cost: 193

  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Path Selection with Cost

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Use the interface-level `ip ospf cost` command so that:
  - Traffic from R10 destined to R6's Loopback0 uses the DMVPN cloud path via R1.

Configuration

```
R4:
interface GigabitEthernet1.146
 ip ospf cost 2
R5:

interface GigabitEthernet1.45
 ip ospf cost 1001
```

Verification

To understand how OSPF path selection can be modified for traffic engineering purposes, it is important to understand why the current path is being selected. OSPF is sometimes described as being partially link-state and partially distance vector. This is because SPF calculations are only performed for Intra-Area routing, whereas the Area Border Routers' (ABRs) advertised information is trusted for Inter-

https://t.me/learningnets

Area and External calculations.

In this particular scenario, the goal is to modify the traffic flow so that packets coming from R10 going to the network 150.1.6.6/32 uses the DMVPN cloud through R1. By visually mapping the traffic flow, we can see that R10 has only one possible path to reach this network in the first place, VLAN 108 link in area 3 to R8. R8 likewise has only one path, the VLAN 58 link in area 3 to R5. R5 has multiple possible paths, the DMVPN link to R1, the DMVPN link to R4, and the VLAN 45 link to R4. R5 basically has one possible path via R1 and two possible paths via R4.

Before doing any configuration changes, let's see how the current path is selected and what is the current cost towards R6's Loopback0, starting from R10:

```
R10#show ip route 150.1.6.6
Routing entry for 150.1.6.6/32  Known via "ospf 1", distance 110, metric 5, type inter area
  Last update from 155.1.108.8 on GigabitEthernet1.108, 00:25:35 ago
  Routing Descriptor Blocks: * 155.1.108.8, from 150.1.5.5, 00:25:35 ago, via GigabitEthernet1.108


      Route metric is 5, traffic share count is 1
```

R10 says that the longest match to 150.1.6.6 is an OSPF Inter-Area route to 150.1.6.6/32 via the next-hop 155.1.108.8 (R8). Note that after the next-hop value, this output shows that the route is **from 150.1.5.5**. This means that R5 is the ABR originating the Network Summary LSA (LSA 3) from area 0 into area 3. By viewing the detailed information about this Type-3 LSA, we can see what R5's cost to the destination is. Specifically, the syntax for this is `show ip ospf database summary 150.1.6.6`, where 150.1.6.6 is the link-state ID (the network without the mask) for the destination:

https://t.me/learningnets

```
R10#show ip ospf database summary 150.1.6.6


            OSPF Router with ID (150.1.10.10) (Process ID 1)


                Summary Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1672
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.6.6 (summary Network Number)
 Advertising Router: 150.1.5.5

  LS Seq Number: 80000004
  Checksum: 0xDC15
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 3
```

The output above indicates that R10 is using the ABR 150.1.5.5 as the exit point to reach 150.1.6.6/32. Note that the metric value of 3 in this output is not R10's metric to the destination, but the ABR's (R5's) metric to the destination. For R10 to actually route to this destination, it must now run SPF on the advertising router 150.1.5.5 (R5's RID).

This is why OSPF is considered partially distance vector and partially link-state. R10 does not know the detailed information about the path selection that is occurring behind R5. R10 trusts R5's calculation for the prefix, which is typically a distance vector behavior, and then calculates the shortest Intra-Area path to reach R5, which is a link-state behavior. The advantage of this design inherent to OSPF is that R10 does not need to run SPF on the entire OSPF topology, which immensely increases scalability. Now let's find out how R10 routes to get to R5.

OSPF's Intra-Area SPF determines this by finding out who R10 is directly adjacent with, and what the cost values to these neighbors are. When this is determined, R10 further sees who its neighbors are adjacent with and what their costs are to their adjacent neighbors. The process continues over and over until R10 finds the shortest path to the ABR 150.1.5.5. This Intra-Area metric is then added to the LSA 3 metric that R5 is advertising, and results in the end-to-end metric value that is actually installed in the routing table. We can verify this by first finding out who R10 is adjacent with by viewing its Router LSA (LSA 1). Specifically, this is accomplished with the `show ip ospf database router self-originate` syntax:

https://t.me/learningnets

```
R10#show ip ospf database router self-originate


            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Router Link States (Area 3)

  LS age: 460
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.10.10
  Advertising Router: 150.1.10.10
  LS Seq Number: 80000022
  Checksum: 0xC758
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.10.10
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
(Link ID) Designated Router address: 155.1.108.10
(Link Data) Router Interface address: 155.1.108.10
      Number of MTID metrics: 0       TOS 0 Metrics: 1


    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.10.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
```

https://t.me/learningnets

R10 sees that it is adjacent with the *Designated Router address: 155.1.108.10*, and that the cost is 1. Although technically the DR for this segment is R10 itself, OSPF (like IS-IS) treats the DR as a separate pseudonode, which it must consult before computing SPF. This is because VLAN 108 link connecting to R8 runs OSPF network type broadcast, which requires the DR/BDR election. Therefore, R10 must ask the DR who it is adjacent with and what its cost calculation is to its adjacent neighbors. This information can be viewed by checking the DR's Network LSA (LSA 2) via the `show ip ospf database network 155.1.108.10` . From the information below, we can see that R10 finds out that the DR for the segment is adjacent with 150.1.8.8 (R8's RID):

```
R10#show ip ospf database network 155.1.108.10

            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 601
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.108.10 (address of Designated Router)
  Advertising Router: 150.1.10.10
  LS Seq Number: 80000020
  Checksum: 0xCD3B
  Length: 32
  Network Mask: /24 Attached Router: 150.1.10.10
 Attached Router: 150.1.8.8
```

Based on this, R10 infers that it should now ask the router 150.1.8.8 who it is adjacent with. We can view this information on R10 with `show ip ospf database router 150.1.8.8` or on R8 with `show ip ospf database router self-originate` :

```
R10#show ip ospf database router 150.1.8.8

            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Router Link States (Area 3)

  LS age: 1043
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.8.8
```

https://t.me/learningnets

```
  Advertising Router: 150.1.8.8
  LS Seq Number: 80000023
  Checksum: 0x7CE4
  Length: 72
  Number of Links: 4

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.8.8
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.108.10
     (Link Data) Router Interface address: 155.1.108.8
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.58.8
 (Link Data) Router Interface address: 155.1.58.8
      Number of MTID metrics: 0       TOS 0 Metrics: 1


    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.8.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
```

The output above means that R8 is adjacent with the DR 155.1.58.8 (R8), and that the cost to the DR is 1. R8 must now ask the DR who it is adjacent with:

```
R10#show ip ospf database network 155.1.58.8

            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1131
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.58.8 (address of Designated Router)
  Advertising Router: 150.1.8.8
  LS Seq Number: 80000020
```

https://t.me/learningnets

```
  Checksum: 0xAC9E
  Length: 32
  Network Mask: /24 Attached Router: 150.1.8.8
Attached Router: 150.1.5.5
```

Because the DR is directly adjacent with 150.1.5.5, R10 infers the total cost to 150.1.5.5 as 1 + 1 = 2. Because R5 is advertising the metric 3 in its LSA 3 for 150.1.6.6/32, and R10's metric to R5 is 2, we now know why the total metric in the routing table of R10 is 5. However, we do not yet know how R5's calculation to the final destination occurs. Because area 3 has no visibility on complete Inter-Area calculations, this can only be verified starting on R5:

```
R5#show ip route 150.1.6.6
Routing entry for 150.1.6.6/32  Known via "ospf 1", distance 110, metric 3, type inter area
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:42:12 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.4.4, 00:42:12 ago, via GigabitEthernet1.45

      Route metric is 3, traffic share count is 1
```

R5 says that it has a single paths to the destination 150.1.6.6 via the longest match 150.1.6.6/32. This is through VLAN 45, one of its area 0 adjacencies with R4. Because R5 sees this prefix as Inter-Area, it must ask the area 0 ABRs how they are calculating the path and then run SPF on the ABRs themselves to find the ultimate shortest path. This can be verified on R5 by first viewing the possible paths to the Network Summary LSAs (LSA 3) that the ABRs are advertising and then to the ABRs that are originating it via their Router LSAs (LSA 1):

```
R5#show ip ospf database summary 150.1.6.6

            OSPF Router with ID (150.1.5.5) (Process ID 1)
Summary Net Link States (Area 0)

  LS age: 1146
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.6.6 (summary Network Number)
Advertising Router: 150.1.1.1
  LS Seq Number: 80000002
  Checksum: 0xBF1
  Length: 28
  Network Mask: /32        MTID: 0 Metric: 2

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 772
```

https://t.me/learningnets

```
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.6.6 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 80000004
  Checksum: 0xDF15
  Length: 28
  Network Mask: /32        MTID: 0 Metric: 2




                Summary Net Link States (Area 3)


  LS age: 698
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 150.1.6.6 (summary Network Number)
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000005
  Checksum: 0xDA16
  Length: 28
  Network Mask: /32
        MTID: 0         Metric: 3
```

R5 sees two possible paths to the destination in the OSPF database. The first is in area 0 via the ABR 150.1.1.1 (R1), the second is in area 0 via the ABR 150.1.4.4 (R4). The Type-3 LSA in area 3 is actually generated by itself, thus it cannot be used for best path selection. Now R5 must find the shortest path to the two ABRs 150.1.1.1 and 150.1.4.4 in area 0. This is accomplished the same way it was in area 3, by finding out the cost to R5's adjacent neighbors, and then recursing over and over to find their cost and their adjacent neighbors' cost toward the ABRs. In this case, the topology is fairly simple because R5 is directly adjacent with both of these area 0 routers:

```
R5#show ip ospf database router self-originate


            OSPF Router with ID (150.1.5.5) (Process ID 1)


                Router Link States (Area 0)


  LS age: 1384
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 800005E8
```

https://t.me/learningnets

```
  Checksum: 0xDD41
  Length: 108
  Area Border Router
  Number of Links: 7

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.45.5
 (Link Data) Router Interface address: 155.1.45.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.4.4
 (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.1.1
 (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.2.2
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.3.3
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
<output omitted>
!
!R5#show ip ospf database network 155.1.45.5

            OSPF Router with ID (150.1.5.5) (Process ID 1)
```

https://t.me/learningnets

```
                Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1316
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.45.5 (address of Designated Router)
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000042
  Checksum: 0xE85E
  Length: 32
  Network Mask: /24 Attached Router: 150.1.5.5
 Attached Router: 150.1.4.4
```

Two points should be noted about the above output. First, the area 3 output that R5 would normally have originated is omitted, because this does not affect its path selection out toward area 1. Second, the output of the three area 0 learned routes indicates that the information is from a *Link connected to: another Router (point-to-point)*. This is because the OSPF network type point-to-multipoint non-broadcast, which is running over the DMVPN cloud does not support the DR/BDR election. Recall that this output in area 3 was different because R10 and R8 could only learn Type1 LSA information via the DR’s pseudonode from Type2 LSA. Network types point-to-multipoint, point-to-multipoint non-broadcast, and point-to-point do not support the DR/BDR election, and therefore do not require the Network LSA (LSA 2) to recurse toward the adjacent neighbor.

At this point we can see why R5 is installing a single path for R6's Loopback0, via the Ethernet link to R4. This is because R5's cost over the Ethernet link is 1, while the cost over the DMVPN cloud is 1000 for both R1 and R4. So the total cost for R6's Loopback0, as seen from R5 is 2 + 1 = 3. The cost to reach all ABR routers can also be identified from the following output:

https://t.me/learningnets

```
R5#show ip ospf border-routers


            OSPF Router with ID (150.1.5.5) (Process ID 1)


                Base Topology (MTID 0)

Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 150.1.1.1 [1000]
 via 155.1.0.1, Tunnel0, ABR, Area 0, SPF 1819i 150.1.2.2 [1000]
 via 155.1.0.2, Tunnel0, ABR, Area 0, SPF 1819i 150.1.3.3 [1000]
 via 155.1.0.3, Tunnel0, ABR, Area 0, SPF 1819i 150.1.4.4 [1]
 via 155.1.45.4, GigabitEthernet1.45, ABR, Area 0, SPF 1819
```

Because the goal of the section is to route through the DMVPN cloud via R1, we need to ensure that the OSPF cost via the Ethernet link is higher than the OSPF cost via the DMVPN cloud. We need to raise the Ethernet cost to be above 1000, thus the solution:

```
R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface gigabitEthernet1.45
R5(config-subif)#ip ospf cost 1001
!
!R5#show ip route 150.1.6.6
Routing entry for 150.1.6.6/32  Known via "ospf 1", distance 110, metric 1002, type inter area
  Last update from 155.1.0.1 on Tunnel0, 00:00:04 ago
  Routing Descriptor Blocks: * 155.1.0.4, from 150.1.4.4, 00:00:04 ago, via Tunnel0
      Route metric is 1002, traffic share count is 1
155.1.0.1, from 150.1.1.1, 00:00:04 ago, via Tunnel0
      Route metric is 1002, traffic share count is 1
!
!R10#show ip ospf database summary 150.1.6.6


            OSPF Router with ID (150.1.10.10) (Process ID 1)


                Summary Net Link States (Area 3)


  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 303
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
```

https://t.me/learningnets

```
Link State ID: 150.1.6.6 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000006
  Checksum: 0x302
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 1002
```

The end result now is that R5 installs two routes in the routing table, as the cost advertised by both ABRs in Type3 LSAs (R1 and R4) is the same with a value of 2, while R5's cost to both R1 and R4 equals 1000. As we cannot have a per-neighbor cost in the DMVPN cloud configured with the **`ip ospf cost`** command, we need to change the metric advertised by R1 and R4, so that R1 advertises a lower metric, thus the solution:

```
R4#configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet1.146
R4(config-subif)#ip ospf cost 2
 !
 !R5#show ip route 150.1.6.6
 Routing entry for 150.1.6.6/32  Known via "ospf 1", distance 110, metric 1002, type inter area
  Last update from 155.1.0.1 on Tunnel0, 00:03:58 ago
  Routing Descriptor Blocks: * 155.1.0.1, from 150.1.1.1, 00:03:58 ago, via Tunnel0
      Route metric is 1002, traffic share count is 1
 !
 !R5#show ip ospf database summary 150.1.6.6

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 522
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.6.6 (summary Network Number)
 Advertising Router: 150.1.1.1
  LS Seq Number: 80000003
  Checksum: 0x9F2
  Length: 28
  Network Mask: /32        MTID: 0 Metric: 2

  LS age: 141
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.6.6 (summary Network Number)
```

https://t.me/learningnets

```
Advertising Router: 150.1.4.4

  LS Seq Number: 80000006
  Checksum: 0xE50C
  Length: 28
  Network Mask: /32        MTID: 0 Metric: 3




                Summary Net Link States (Area 3)

  LS age: 76
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 150.1.6.6 (summary Network Number)
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000007
  Checksum: 0x103
  Length: 28
  Network Mask: /32
        MTID: 0         Metric: 1002
```

Verify with a traceroute, that traffic from R10 towards R6's Loopback0 is sent over the DMVPN cloud via R1:

```
R10#show ip route 150.1.6.6
Routing entry for 150.1.6.6/32  Known via "ospf 1", distance 110, metric 1004, type inter area
  Last update from 155.1.108.8 on GigabitEthernet1.108, 00:09:00 ago
  Routing Descriptor Blocks: * 155.1.108.8, from 150.1.5.5, 00:09:00 ago, via GigabitEthernet1.108
      Route metric is 1004, traffic share count is 1
!
!R10#traceroute 150.1.6.6
Type escape sequence to abort.
Tracing the route to 150.1.6.6
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.108.8 3 msec 2 msec 2 msec
  2 155.1.58.5 2 msec 2 msec 6 msec 3 155.1.0.1 3 msec 3 msec 4 msec

  4 155.1.146.6 54 msec *  4 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Path Selection with Bandwidth

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Using the interface-level `bandwidth` command so that:
  - Traffic from R6 destined to R8's Loopback0 uses VLAN 146 path via R1.

Configuration

```
R1:
interface Tunnel0
 bandwidth 100000
R4:

interface GigabitEthernet1.45
 bandwidth 10000
```

Verification

Like the previous section, it is first important to understand why the current path selection occurs before modifying it. In this case, without any configuration changes, R6 has one route to the destination 150.1.8.8 via the longest match 150.1.8.8/32 from R4:

```
R6#show ip route 150.1.8.8
```

https://t.me/learningnets

```
Routing entry for 150.1.8.8/32   Known via "ospf 1", distance 110, metric 4, type inter area
  Last update from 155.1.146.4 on GigabitEthernet1.146, 00:13:10 ago
  Routing Descriptor Blocks: * 155.1.146.4, from 150.1.4.4, 01:53:00 ago, via GigabitEthernet1.146

      Route metric is 4, traffic share count is 1
```

The metric of 4 is computed as the cost towards the ABR (R1 or R4), plus the ABR advertised cost for the Type3 LSA:

```
R6#show ip ospf database summary 150.1.8.8

            OSPF Router with ID (150.1.6.6) (Process ID 1)
 Summary Net Link States (Area 1)

  LS age: 156
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.1.1
  LS Seq Number: 80000004
  Checksum: 0x11F9
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 1002

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1247
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 8000000B
  Checksum: 0xB137
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 3


                Summary Net Link States (Area 2)

  LS age: 1321
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.3.3
  LS Seq Number: 80000002
  Checksum: 0xFA0E
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 1002
```

https://t.me/learningnets

```
!
!R6#show ip ospf database router self-originate

            OSPF Router with ID (150.1.6.6) (Process ID 1)

                Router Link States (Area 1)

  LS age: 1108
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.6.6
  Advertising Router: 150.1.6.6
  LS Seq Number: 80000025
  Checksum: 0x1791
  Length: 48
  Number of Links: 2

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.6.6
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.146.6
 (Link Data) Router Interface address: 155.1.146.6
      Number of MTID metrics: 0       TOS 0 Metrics: 1
<output omitted>
!
!R6#show ip ospf database network  155.1.146.6

            OSPF Router with ID (150.1.6.6) (Process ID 1)

                Net Link States (Area 1)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 155
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.146.6 (address of Designated Router) Advertising Router: 150.1.6.6
  LS Seq Number: 80000085
  Checksum: 0x29D2
  Length: 36 Network Mask: /24
 Attached Router: 150.1.6.6
 Attached Router: 150.1.1.1
```

https://t.me/learningnets

```
Attached Router: 150.1.4.4
```

R6 sees the Inter-Area Type3 LSA route via three ABRs, to R1 and R3 with a metric of 1002 and to R4 with a metric of 3. R6 must then recurse to LSA 1 to find the shortest Intra-Area paths to these ABRs. R6 says that the metric to reach the DR 155.1.146.6 for VLAN 146 is 1. Because we know from the topology diagram that the only neighbors on VLAN 146 are R1, R4, and R6, this implies that the metric to reach the ABRs R1 and R4 is 1 as well. The total cost via R1 would be 1002 + 1 = 1003 , via R3 would be 1002 + 2 = 1004 while via R4 would be 3 + 1 = 4. The cost to the ABRs can also be identified from the following output:

```
R6#show ip ospf border-routers

            OSPF Router with ID (150.1.6.6) (Process ID 1)



                Base Topology (MTID 0)

Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 150.1.1.1 [1]
 via 155.1.146.1, GigabitEthernet1.146, ABR, Area 1, SPF 249i 150.1.3.3 [2]
 via 155.1.67.7, GigabitEthernet1.67, ABR, Area 2, SPF 8i 150.1.4.4 [1]
 via 155.1.146.4, GigabitEthernet1.146, ABR, Area 1, SPF 249
```

Changing R6's metric/bandwidth to the DR of VLAN 146 would affect the metric to both ABRs R1 and R4. Although this will change the overall end-to-end cost, it will not change R6's path selection. Changing R5's area 3 path to the destination would affect the LSA 3 advertisement that R5 sends into area 0, but it would still result in the same LSA 3 being received by R6 via R1 and R4, just with a different metric than the default. Raising R4's metric of the LSA 3 sent into area 1, or lowering R1's metric of the LSA 3 sent into area 1, would cause R6 to route to R1. In this particular solution, R1's metric is lowered for the Type3 LSA. Because the OSPF cost value is derived from the bandwidth, changing the **`bandwidth`** keyword at the interface level changes the cost:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface tunnel0
R1(config-if)#bandwidth 100000
!
!R6#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32  Known via "ospf 1", distance 110, metric 4, type inter area
```

https://t.me/learningnets

```
  Last update from 155.1.146.1 on GigabitEthernet1.146, 00:00:19 ago
  Routing Descriptor Blocks: * 155.1.146.4, from 150.1.4.4, 02:08:39 ago, via GigabitEthernet1.146
      Route metric is 4, traffic share count is 1
155.1.146.1, from 150.1.1.1, 00:00:19 ago, via GigabitEthernet1.146
      Route metric is 4, traffic share count is 1
!
!R6#show ip ospf database summary 150.1.8.8

            OSPF Router with ID (150.1.6.6) (Process ID 1)
 Summary Net Link States (Area 1)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 41
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.1.1
  LS Seq Number: 80000005
  Checksum: 0xE410
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 3

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 71
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 8000000C
  Checksum: 0xAF38
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 3



                Summary Net Link States (Area 2)

  LS age: 115
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 150.1.8.8 (summary Network Number)
  Advertising Router: 150.1.3.3
  LS Seq Number: 80000003
  Checksum: 0xF80F
  Length: 28
  Network Mask: /32
```

https://t.me/learningnets

```
        MTID: 0         Metric: 1002
```

This did not completely fix the desired path, as now both paths via R1 and R4 have the same cost. We basically configured R1 to have the same cost via the DMVPN cloud as R4 via the VLAN 45 link. To obtain the correct path for the prefix, we change R4's cost on VLAN 45 link in order to change the metric for Type3 LSA advertised by R4. R4 now advertises a metric of 12, the end-to-end cost via R4 will now be 12 + 1 = 13, versus the end-to-end cost via R1 which is 3 + 1 = 4:

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.45
R4(config-subif)#bandwidth 10000
!
!R6#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32  Known via "ospf 1", distance 110, metric 4, type inter area
  Last update from 155.1.146.1 on GigabitEthernet1.146, 00:04:10 ago
  Routing Descriptor Blocks: * 155.1.146.1, from 150.1.1.1, 00:04:10 ago, via GigabitEthernet1.146
      Route metric is 4, traffic share count is 1
!
!R6#show ip ospf database summary 150.1.8.8

            OSPF Router with ID (150.1.6.6) (Process ID 1)
 Summary Net Link States (Area 1)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 270
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.1.1
  LS Seq Number: 80000005
  Checksum: 0xE410
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 3

  LS age: 45
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 8000000D
  Checksum: 0x8D5
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 12
```

https://t.me/learningnets

```
                Summary Net Link States (Area 2)

  LS age: 344
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 150.1.8.8 (summary Network Number)
  Advertising Router: 150.1.3.3
  LS Seq Number: 80000003
  Checksum: 0xF80F
  Length: 28
  Network Mask: /32
        MTID: 0         Metric: 1002
```

Verify with a traceroute, that traffic from R6 towards R8's Loopback0 is routed via R1:

```
R6#traceroute 150.1.8.8
Type escape sequence to abort.
Tracing the route to 150.1.8.8
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.1 12 msec 4 msec 5 msec

  2 155.1.0.5 2 msec 2 msec 3 msec
  3 155.1.58.8 5 msec *  4 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Path Selection with Per-Neighbor Cost

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure per-neighbor cost values on R5 so that cost to reach R1 is 50 and cost to reach R4 is 40.

Configuration

```
R5:

router ospf 1
 neighbor 155.1.0.1 cost 50
 neighbor 155.1.0.4 cost 40
```

Verification

OSPF cost calculation for a segment is based on the bandwidth value of the outgoing interface. In certain network topologies, this can be an issue if the underlying layer 2 network does not map directly to the bandwidth value of the interface connecting to it.

For example, in this topology, R5's single GigabitEthernet interface connecting to the DMVPN cloud has multiple VPNs. Although the physical GigabitEthernet interface has a bandwidth value of 1Gbps, the individual VPNs themselves are not

https://t.me/learningnets

provisioned at this rate. The bandwidth can be changed per interface or per sub-interface but that will affect all the VPNs. To have different OSPF cost per neighbor, the network type's point-to-multipoint and point-to-multipoint non-broadcast support the setting of the OSPF cost value on a per-neighbor basis. Before modifying the per neighbor cost, the output of the **`show ip ospf database router self-originate`** command shows the default cost to reach all neighbors in the DMVPN cloud, which equals to 1000:

```
R5#show ip ospf database router self-originate

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

  LS age: 1598
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 800005F0
  Checksum: 0x6DA9
  Length: 108
  Area Border Router
  Number of Links: 7

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.45.5
     (Link Data) Router Interface address: 155.1.45.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.4.4
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.3.3
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000
```

https://t.me/learningnets

```
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.2.2
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.1.1
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000



    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
<output omitted>
```

After changing the per-neighbor cost, R5's router LSA (LSA 1) is changed to reflect the cost values defined on a per-neighbor basis:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router ospf 1
R5(config-router)#neighbor 155.1.0.1 cost 50
R5(config-router)#neighbor 155.1.0.4 cost 40
!
!R5#show ip ospf database router self-originate


            OSPF Router with ID (150.1.5.5) (Process ID 1)


                Router Link States (Area 0)


  LS age: 44
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 800005F2
  Checksum: 0xBAD7
  Length: 108
  Area Border Router
  Number of Links: 7


    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
```

https://t.me/learningnets

```
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.45.5
     (Link Data) Router Interface address: 155.1.45.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.4.4
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 40
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.3.3
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.2.2
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.1.1
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 50


    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
<output omitted>
```

The fact that cost no longer reflects the interface bandwidth for all neighbors, is also visible in the following output:

```
R5#show ip ospf neighbor detail | include Neighbor|Cost
 Neighbor 150.1.4.4, interface address 155.1.45.4
    Neighbor priority is 1, State is FULL, 108 state changes
    Neighbor is up for 04:15:56 Neighbor 150.1.4.4, interface address 155.1.0.4
    Neighbor priority is 0 (configured 0), State is FULL, 6 state changes,
 Cost in topology Base with MTID 0 is 40
    Neighbor is up for 02:48:30
 Neighbor 150.1.3.3, interface address 155.1.0.3
    Neighbor priority is 0 (configured 0), State is FULL, 6 state changes
```

https://t.me/learningnets

```
    Neighbor is up for 02:48:30
 Neighbor 150.1.2.2, interface address 155.1.0.2
    Neighbor priority is 0 (configured 0), State is FULL, 6 state changes
    Neighbor is up for 02:48:30 Neighbor 150.1.1.1, interface address 155.1.0.1
    Neighbor priority is 0 (configured 0), State is FULL, 6 state changes,
Cost in topology Base with MTID 0 is 50

    Neighbor is up for 02:48:30
 Neighbor 150.1.8.8, interface address 155.1.58.8
    Neighbor priority is 1, State is FULL, 6 state changes
    Neighbor is up for 20:38:30
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

Discontiguous OSPF Areas with Virtual-Links

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure the network so that full reachability is maintained if R3's connection to R7 goes down.
  - Do not apply any configuration to R4 to solve this task.

Configuration

```
R1:
router ospf 1
 area 1 virtual-link 150.1.6.6
R6:

router ospf 1
 area 1 virtual-link 150.1.1.1
```

Verification

https://t.me/learningnets

Devices in OSPF area 2 have two exit points out the rest of the network, R3 via area 0, and R6 via area 1. However, although R6 has interfaces in multiple areas, it is technically not an ABR. This can be verified in the OSPF database of devices area 2, because only R3 can originate Summary Network LSAs (LSA 3) to describe Inter-Area destinations:

```
R9#show ip ospf border-routers

            OSPF Router with ID (150.1.9.9) (Process ID 1)



                Base Topology (MTID 0)

Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 150.1.3.3 [2]
 via 155.1.79.7, GigabitEthernet1.79, ABR, Area 2, SPF 7
!
!R9#show ip ospf database

            OSPF Router with ID (150.1.9.9) (Process ID 1)

                Router Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.3.3       150.1.3.3       1943        0x80000035 0x00C173 1
150.1.6.6       150.1.6.6       2008        0x80000034 0x001BD0 1
150.1.7.7       150.1.7.7       1807        0x80000036 0x00EAF4 5
150.1.9.9       150.1.9.9       1609        0x80000034 0x00262B 3

                Net Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.37.7      150.1.7.7       67          0x80000033 0x003C1A
155.1.67.7      150.1.7.7       67          0x80000033 0x0042EF
155.1.79.9      150.1.9.9       1609        0x80000032 0x00D248
 Summary Net Link States (Area 2)
 Link ID ADV Router
      Age         Seq#       Checksum 150.1.1.1 150.1.3.3
       1943        0x8000000D 0x00A17F 150.1.2.2 150.1.3.3
       454         0x80000013 0x008098 150.1.3.3 150.1.3.3
       1943        0x80000014 0x0002EB 150.1.4.4 150.1.3.3
       1943        0x80000015 0x0029D4 150.1.5.5 150.1.3.3
       454         0x80000013 0x000EF0 150.1.6.6 150.1.3.3
       1943        0x80000015 0x0009EF 150.1.8.8 150.1.3.3
       454         0x80000013 0x00D81F
```

https://t.me/learningnets

```
150.1.10.10 150.1.3.3
       454         0x80000013 0x00B83A 155.1.0.1 150.1.3.3
       1943        0x8000000D 0x0061BC 155.1.0.2 150.1.3.3
       454         0x80000013 0x004BCB 155.1.0.3 150.1.3.3
       963         0x80000015 0x00D516 155.1.0.4 150.1.3.3
       1943        0x80000015 0x000AF3 155.1.0.5 150.1.3.3
       454         0x80000013 0x00F906 155.1.5.0 150.1.3.3
       454         0x80000013 0x00FEFF 155.1.8.0 150.1.3.3
       454         0x80000013 0x00E713 155.1.10.0 150.1.3.3
       454         0x80000013 0x00DB1C 155.1.13.0 150.1.3.3
       1943        0x80000033 0x003290 155.1.23.0 150.1.3.3
       1943        0x80000033 0x00C3F4 155.1.45.0 150.1.3.3
       1943        0x80000015 0x004193 155.1.58.0 150.1.3.3
       454         0x80000013 0x00B514 155.1.108.0 150.1.3.3
       454         0x80000013 0x0097FE 155.1.146.0 150.1.3.3
       1943        0x80000015 0x00EF7E
!
!R9# show ip route 150.1.5.5
Routing entry for 150.1.5.5/32  Known via "ospf 1", distance 110, metric 1003, type inter area
  Last update from 155.1.79.7 on GigabitEthernet1.79, 10:04:27 ago
  Routing Descriptor Blocks: * 155.1.79.7, from 150.1.3.3, 10:04:27 ago, via GigabitEthernet1.79

      Route metric is 1003, traffic share count is 1
```

Based on the above output, we can infer that if area 2 loses its connection to R3, all Inter-Area connectivity to area 2 will be lost. This can be demonstrated by shutting down R3's connection to R7:

```
R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R3(config)#interface GigabitEthernet1.37
R3(config-subif)#shutdown
```

When the link to R3 is down, R7 issues a withdraw message, but the devices in area 2 cache the LSA information from R3 in the OSPF database until the LSA age expires. Although LSA Type3 will still be present in the OSPF database until expire, all routers know that R3 is no longer reachable in area 2 based on Type1 LSAs, thus all its advertised LSAs are not usable:

```
R9#show ip ospf database

            OSPF Router with ID (150.1.9.9) (Process ID 1)

                Router Link States (Area 2)
```

https://t.me/learningnets

```
Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.3.3       150.1.3.3       254         0x80000036 0x00BF74 1
150.1.6.6       150.1.6.6       296         0x80000035 0x0019D1 1
150.1.7.7       150.1.7.7       5           0x80000038 0x009913 5
150.1.9.9       150.1.9.9       1913        0x80000034 0x00262B 3

                Net Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.37.7      150.1.7.7       3604        0x80000034 0x003A1B
155.1.67.7      150.1.7.7       371         0x80000033 0x0042EF
155.1.79.9      150.1.9.9       1913        0x80000032 0x00D248
 Summary Net Link States (Area 2)
Link ID         ADV Router
      Age         Seq#       Checksum
150.1.1.1       150.1.3.3       254         0x8000000E 0x009F80
150.1.2.2       150.1.3.3       758         0x80000013 0x008098
150.1.3.3       150.1.3.3       254         0x80000015 0x00FFEC
150.1.4.4       150.1.3.3       254         0x80000016 0x0027D5150.1.5.5        150.1.3.3
       758         0x80000013 0x000EF0
150.1.6.6       150.1.3.3       254         0x80000016 0x0007F0
150.1.8.8       150.1.3.3       758         0x80000013 0x00D81F
150.1.10.10     150.1.3.3       758         0x80000013 0x00B83A
155.1.0.1       150.1.3.3       254         0x8000000E 0x005FBD
155.1.0.2       150.1.3.3       758         0x80000013 0x004BCB
155.1.0.3       150.1.3.3       1267        0x80000015 0x00D516
155.1.0.4       150.1.3.3       254         0x80000016 0x0008F4
155.1.0.5       150.1.3.3       758         0x80000013 0x00F906
155.1.5.0       150.1.3.3       758         0x80000013 0x00FEFF
155.1.8.0       150.1.3.3       758         0x80000013 0x00E713
155.1.10.0      150.1.3.3       758         0x80000013 0x00DB1C
155.1.13.0      150.1.3.3       254         0x80000034 0x003091
155.1.23.0      150.1.3.3       254         0x80000034 0x00C1F5
155.1.45.0      150.1.3.3       254         0x80000016 0x003F94
155.1.58.0      150.1.3.3       758         0x80000013 0x00B514
155.1.108.0     150.1.3.3       758         0x80000013 0x0097FE
155.1.146.0     150.1.3.3       254         0x80000016 0x00ED7F
!
!R9#show ip ospf border-routers

            OSPF Router with ID (150.1.9.9) (Process ID 1)


                Base Topology (MTID 0)
```

https://t.me/learningnets

```
Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route
R9#
!
!R9#show ip ospf database summary 150.1.5.5

            OSPF Router with ID (150.1.9.9) (Process ID 1)

                Summary Net Link States (Area 2)

  LS age: 800
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.5.5 (summary Network Number)
Advertising Router: 150.1.3.3
  LS Seq Number: 80000013
  Checksum: 0xEF0
  Length: 28 Network Mask: /32
        MTID: 0         Metric: 1001
!
!R9#show ip ospf database router adv-router 150.1.3.3

            OSPF Router with ID (150.1.9.9) (Process ID 1)

                Router Link States (Area 2)
Adv Router is not-reachable in topology Base with MTID 0
  LS age: 325
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.3.3
  Advertising Router: 150.1.3.3
  LS Seq Number: 80000036
  Checksum: 0xBF74
  Length: 36
  Area Border Router
  Number of Links: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.37.7
     (Link Data) Router Interface address: 155.1.37.3
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
!
```

https://t.me/learningnets

```
!R9#show ip route 150.1.5.5
% Subnet not in table
```

To resolve this problem, R6 must offer devices in area 2 an alternate path to area 0. This can be done by adding a new link to R6 that is in area 0, such as another Ethernet interface or a Tunnel interface, or by configuring an OSPF virtual-link.

An OSPF virtual-link allows the creation of an indirect area 0 adjacency. This adjacency can be used to repair breaks in the OSPF domain or solve traffic engineering requirements. In this particular case, R6 can virtual-link to the neighbors R1 or R4 who are in area 1, because they both have connections to area 0. R1 is chosen in this solution because the task instructs you not to configure R4. The first important point to note about the virtual-link is that the neighbor value specified in the virtual-link syntax is the router-id of the neighbor in the transit area. This means that if for some reason the router-id changes (for example, if a new higher Loopback interface is added) or the `router-id` command is changed, the virtual-link will fail.

Additionally, the neighbors forming adjacency over the virtual-link do not have to be directly connected; they simply need to know how to recurse toward each other's LSA 1 advertisements. This means that the traffic flow via the virtual-link should naturally follow the Intra-Area SPF calculation between the routers' LSA 1 advertisements. Verification of this can be seen as follows:

https://t.me/learningnets

```
R1#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#router ospf 1
R1(config-router)#area 1 virtual-link 150.1.6.6

!
!R6#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#router ospf 1
R6(config-router)#area 1 virtual-link 150.1.1.1

!
%OSPF-5-ADJCHG: Process 1, Nbr 150.1.1.1 on OSPF_VL0 from LOADING to FULL, Loading Done
!
!
R6#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.1.1         0   FULL/  -        00:00:29    155.1.146.1     OSPF_VL0

150.1.1.1         1   FULL/BDR        00:00:39    155.1.146.1     GigabitEthernet1.146
223.255.255.255   0   FULL/DROTHER    00:00:39    155.1.146.4     GigabitEthernet1.146
150.1.7.7         1   FULL/BDR        00:00:36    155.1.67.7      GigabitEthernet1.67
```

R1 and R6 form an adjacency over the virtual-link. **`show ip ospf interface`** indicates that the virtual-link is an area 0 interface:

```
R6#show ip ospf interface
OSPF_VL0 is up, line protocol is up
  Internet Address 155.1.146.6/24, Area 0, Attached via Not Attached
  Process ID 1, Router ID 150.1.6.6, Network Type VIRTUAL_LINK, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Configured as demand circuit Run as demand circuit
 DoNotAge LSA allowed
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:03
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can not be protected by per-prefix Loop-Free FastReroute
  Can not be used for per-prefix Loop-Free FastReroute repair paths
```

https://t.me/learningnets

```
  Index 1/4, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
 Adjacent with neighbor 150.1.1.1  (Hello suppressed)

  Suppress hello for 1 neighbor(s)
<output omitted>
```

The cost value of the virtual-link is based on R6's cost to reach the router 150.1.1.1 in area 1. This can be calculated via LSA 1 recursion. R6 now generates an LSA 1 for area 0, along with its previous LSA 1s in areas 1 and 2. Inside the area 0 LSA 1, it says that it is adjacent to R1 over a virtual-link. Inside the area 1 LSA 1, it says that it is the endpoint for the virtual-link and is adjacent with the DR 155.1.146.6 (itself). Asking the DR who it is adjacent with via LSA 2 it implies that R6 is on the same segment as R1 with a cost of 1. This metric of 1 is what the virtual-link inherits. Now because R6 is a true ABR, it should be generating LSA 3 into area 2 about area 0 (and the other areas) and generating LSA 3 into area 0 about area 2. Confirm that area 2 routers now see R6 as ABR and note that all Type 3 LSAs are now advertised by both R3 and R6:

```
R9#show ip ospf border-routers

            OSPF Router with ID (150.1.9.9) (Process ID 1)



                Base Topology (MTID 0)

Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route
 i 150.1.6.6 [2]
 via 155.1.79.7, GigabitEthernet1.79, ABR, Area 2, SPF 11
i 150.1.3.3 [2] via 155.1.79.7, GigabitEthernet1.79, ABR, Area 2, SPF 11
!
!R9#show ip ospf database

            OSPF Router with ID (150.1.9.9) (Process ID 1)

                Router Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.3.3       150.1.3.3       460         0x80000039 0x00B977 1
150.1.6.6       150.1.6.6       381         0x80000036 0x001ACE 1
150.1.7.7       150.1.7.7       459         0x80000039 0x00E4F7 5
```

https://t.me/learningnets

```
150.1.9.9       150.1.9.9       709         0x80000035 0x00242C 3

                Net Link States (Area 2)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.37.7      150.1.7.7       455         0x80000035 0x00381C
155.1.67.7      150.1.7.7       1156        0x80000033 0x0042EF
155.1.79.9      150.1.9.9       709         0x80000033 0x00D049
 Summary Net Link States (Area 2)
Link ID         ADV Router
      Age         Seq#       Checksum
150.1.1.1       150.1.3.3       1039        0x8000000E 0x009F80
150.1.1.1       150.1.6.6       371         0x80000001 0x0035C8
150.1.2.2       150.1.3.3       1543        0x80000013 0x008098
150.1.2.2       150.1.6.6       367         0x80000002 0x005CB1
150.1.3.3       150.1.3.3       1039        0x80000015 0x00FFEC
150.1.3.3       150.1.6.6       367         0x80000002 0x0047C4
150.1.4.4       150.1.3.3       1039        0x80000016 0x0027D5
150.1.4.4       150.1.6.6       366         0x80000002 0x00F303150.1.5.5       150.1.3.3
       1543        0x80000013 0x000EF0150.1.5.5       150.1.6.6
       366         0x80000002 0x00E80B
150.1.6.6       150.1.3.3       1039        0x80000016 0x0007F0
150.1.6.6       150.1.6.6       382         0x80000001 0x00C133
150.1.8.8       150.1.3.3       1543        0x80000013 0x00D81F
150.1.8.8       150.1.6.6       371         0x80000001 0x00B538
150.1.10.10     150.1.3.3       1543        0x80000013 0x00B83A
150.1.10.10     150.1.6.6       371         0x80000001 0x009553
155.1.0.1       150.1.3.3       1039        0x8000000E 0x005FBD
155.1.0.1       150.1.6.6       371         0x80000001 0x00F406
155.1.0.2       150.1.3.3       1543        0x80000013 0x004BCB
155.1.0.2       150.1.6.6       366         0x80000002 0x0027E4
155.1.0.3       150.1.3.3       22          0x80000016 0x00D317
155.1.0.3       150.1.6.6       366         0x80000002 0x001DED
155.1.0.4       150.1.3.3       1039        0x80000016 0x0008F4
155.1.0.4       150.1.6.6       366         0x80000002 0x00D422
155.1.0.5       150.1.3.3       1543        0x80000013 0x00F906
155.1.0.5       150.1.6.6       366         0x80000002 0x00D420
155.1.5.0       150.1.3.3       1543        0x80000013 0x00FEFF
155.1.5.0       150.1.6.6       371         0x80000001 0x00DB19
155.1.8.0       150.1.3.3       1543        0x80000013 0x00E713
155.1.8.0       150.1.6.6       371         0x80000001 0x00C42C
155.1.10.0      150.1.3.3       1543        0x80000013 0x00DB1C
155.1.10.0      150.1.6.6       371         0x80000001 0x00B835
155.1.13.0      150.1.3.3       1039        0x80000034 0x003091
155.1.13.0      150.1.6.6       371         0x80000001 0x007974
155.1.23.0      150.1.3.3       1039        0x80000034 0x00C1F5
```

https://t.me/learningnets

```
155.1.23.0      150.1.6.6       371         0x80000001 0x0049AD
155.1.45.0      150.1.3.3       1039        0x80000016 0x003F94
155.1.45.0      150.1.6.6       366         0x80000002 0x0016B6
155.1.58.0      150.1.3.3       1543        0x80000013 0x00B514
155.1.58.0      150.1.6.6       371         0x80000001 0x00922D
155.1.108.0     150.1.3.3       1543        0x80000013 0x0097FE
155.1.108.0     150.1.6.6       371         0x80000001 0x007418
155.1.146.0     150.1.3.3       1039        0x80000016 0x00ED7F
155.1.146.0     150.1.6.6       382         0x80000001 0x00B2B6
```

Note that in the above output, LSA 3s appear in area 2 from both R3 and R6, even though R3 is still unreachable. This is because R3's LSAs are still aging out, even though the router R3 itself is still unreachable. The result of the virtual-link is that traffic from area 2 to area 0, along with the other areas, now transits area 1 via the virtual-link:

```
R9# show ip route 150.1.5.5
Routing entry for 150.1.5.5/32
  Known via "ospf 1", distance 110, metric 5, type inter area
  Last update from 155.1.79.7 on GigabitEthernet1.79, 00:07:54 ago
  Routing Descriptor Blocks: * 155.1.79.7, from 150.1.6.6, 00:07:54 ago, via GigabitEthernet1.79
      Route metric is 5, traffic share count is 1
!
!R9#traceroute 150.1.5.5
Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.79.7 3 msec 1 msec 1 msec 2 155.1.67.6 3 msec 2 msec 2 msec
  3 155.1.146.4 2 msec 13 msec 5 msec 4 155.1.45.5 3 msec *  3 msec
```

When R3's connection to R7 comes back up, inter-area traffic from area 2 still uses R6 as the exit point because of the shorter metric advertised on Type3 LSAs. For example take R10's Loopback0. R7 has the same cost of 1 towards both ABRs, but R6 advertises the LSA with a mertic of 5, while R3 with a metric of 1003, thus the best path is via R6 with a total cost of 1 + 5 = 6, which is the metric from the routing table as well:

```
R7#show ip ospf border-routers

            OSPF Router with ID (150.1.7.7) (Process ID 1)


                Base Topology (MTID 0)
```

https://t.me/learningnets

```
Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 150.1.6.6 [1]
 via 155.1.67.6, GigabitEthernet1.67, ABR, Area 2, SPF 11i 150.1.3.3 [1]
 via 155.1.37.3, GigabitEthernet1.37, ABR, Area 2, SPF 11
!
!R7#show ip ospf database summary 150.1.10.10

            OSPF Router with ID (150.1.7.7) (Process ID 1)

                Summary Net Link States (Area 2)

  LS age: 2002
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)Link State ID: 150.1.10.10 (summary Network Number)
Advertising Router: 150.1.3.3
  LS Seq Number: 80000013
  Checksum: 0xB83A
  Length: 28Network Mask: /32
        MTID: 0Metric: 1003

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 831
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)Link State ID: 150.1.10.10 (summary Network Number)
Advertising Router: 150.1.6.6
  LS Seq Number: 80000001
  Checksum: 0x9553
  Length: 28Network Mask: /32
        MTID: 0Metric: 5
!
!R7#show ip route 150.1.10.10
Routing entry for 150.1.10.10/32  Known via "ospf 1", distance 110,metric 6, type inter area
  Last update from 155.1.67.6 on GigabitEthernet1.67, 00:13:37 ago
  Routing Descriptor Blocks: * 155.1.67.6, from 150.1.6.6, 00:13:37 ago, via GigabitEthernet1.67
      Route metric is 6, traffic share count is 1
!
!
R7#traceroute 150.1.10.10
Type escape sequence to abort.
Tracing the route to 150.1.10.10
VRF info: (vrf in name/id, vrf out name/id)1 155.1.67.6 20 msec 1 msec 3 msec
  2 155.1.146.4 7 msec 2 msec 1 msec
  3 155.1.45.5 3 msec 2 msec 3 msec
  4 155.1.58.8 3 msec 3 msec 4 msec
```

https://t.me/learningnets

```
5 155.1.108.10 10 msec *  46 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Path Selection with Non-Backbone Transit Areas

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure a virtual link between R1 and R6 in area 1.
- Modify the SPF calculation in the OSPF domain so that R4 cannot be used to reach area 0 by transiting area 1.
  - Do not change any cost values.
- Verify this by ensuring that traffic from R6 destined to the R8's Loopback0 is routed through R1.

Configuration

```
R1:
 router ospf 1
  area 1 virtual-link 150.1.6.6
  no capability transit
R6:

 router ospf 1
  area 1 virtual-link 150.1.1.1
  no capability transit
```

https://t.me/learningnets

Verification

Per RFC 2328 (OSPF Version 2) section 16.3, *Examining transit areas' summary-LSAs*, non-backbone (not area 0) areas can be used for inter-area transit if a shorter path can be found through them, and if the "TransitCapability parameter has been set to TRUE in Step 2 of the Dijkstra algorithm." In Cisco's IOS implementation, this flag is controlled with the `capability transit` routing process-level command, and is on by default.

The design case when the feature is used is very specific and has to do with a shorter Inter-Area path being found via a non-area 0 router as compared to the target router of a virtual-link. To understand this in detail, let's first see R6's path selection to 150.1.8.8/32 after virtual-link has been configured, but before disabling the transit capability:

```
R6#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32  Known via "ospf 1", distance 110, metric 4, type inter area
  Last update from 155.1.146.4 on GigabitEthernet1.146, 00:01:47 ago
  Routing Descriptor Blocks: * 155.1.146.4, from 150.1.5.5, 00:01:47 ago, via GigabitEthernet1.146

      Route metric is 4, traffic share count is 1
```

R6's virtual-link to area 0 is via R1. Normally, we would assume that R6 must route via the virtual-link path to R1 to reach Inter-Area destinations advertised by area 0. However, in this case R6 says that the Inter-Area route 150.1.8.8/32 is via the next-hop 155.1.146.4 (R4), but originated by the router 150.1.5.5 (R5). The below traceroute output indicates that R1 is not used in the transit path for this traffic:

```
R6#traceroute 150.1.8.8
Type escape sequence to abort.
Tracing the route to 150.1.8.8
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.4 12 msec 4 msec 4 msec

  2 155.1.45.5 6 msec 14 msec 5 msec
  3 155.1.58.8 52 msec *  6 msec
```

To determine why R6 chooses this path, we must first find out how R6 is learning the Inter-Area Type3 LSA prefix:

```
R6#show ip ospf database summary 150.1.8.8

            OSPF Router with ID (150.1.6.6) (Process ID 1)
```

https://t.me/learningnets

```
 Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1206 (DoNotAge)
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.8.8 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000047
  Checksum: 0x2289
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 2

<output omitted>
```

This output indicates that 150.1.5.5 (R5) is the originating ABR and advertises the prefix with a metric of 2. We must now figure out how to route toward the ABR R5 and what is the metric to the ABR:

```
R6#show ip ospf database router 150.1.5.5

            OSPF Router with ID (150.1.6.6) (Process ID 1)
 Router Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1206 (DoNotAge)
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000617
  Checksum: 0xAE41
  Length: 108
  Area Border Router
  Number of Links: 7

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.45.5
 (Link Data) Router Interface address: 155.1.45.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1
```

https://t.me/learningnets

```
 Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 150.1.1.1
(Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 150.1.4.4
(Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000


    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.3.3
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.2.2
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

R5's area 0 Router LSA (LSA 1) states that it is adjacent with R1 over the DMVPN network and R4 over the DMVPN and VLAN 45 links. Now we must determine what R1's and R4's costs are to R5. R1 says its metric to reach R5 is 1000 via the DMVPN link, while R4 says its metric to reach R5 is 1000 via the DMVPN link and 1 via VLAN 45 link:

```
R6#show ip ospf database router 150.1.1.1


            OSPF Router with ID (150.1.6.6) (Process ID 1)
 Router Link States (Area 0)


  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1 (DoNotAge)
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.1.1
  Advertising Router: 150.1.1.1
```

https://t.me/learningnets

```
  LS Seq Number: 800004F2
  Checksum: 0x2D74
  Length: 72
  Area Border Router
  Number of Links: 4

    Link connected to: a Virtual Link
     (Link ID) Neighboring Router ID: 150.1.6.6
     (Link Data) Router Interface address: 155.1.146.1
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.1.1
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.5.5
 (Link Data) Router Interface address: 155.1.0.1
      Number of MTID metrics: 0       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.1
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
<output omitted>
!
!R6#show ip ospf database router 150.1.4.4

            OSPF Router with ID (150.1.6.6) (Process ID 1)
 Router Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 868 (DoNotAge)
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.4.4
  Advertising Router: 150.1.4.4
  LS Seq Number: 800002AD
  Checksum: 0x505
  Length: 72
  Area Border Router
  Number of Links: 4
```

https://t.me/learningnets

```
    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.4.4
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.45.5
 (Link Data) Router Interface address: 155.1.45.4
      Number of MTID metrics: 0       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.5.5
 (Link Data) Router Interface address: 155.1.0.4
      Number of MTID metrics: 0       TOS 0 Metrics: 1000

<output omitted>
```

Now we must find what R6's metrics to R1 and R4 are. R6 has reachability to R1 via the virtual-link with a metric of 1. R6 has reachability to R4 via the DR of VLAN 146 in area 1 with a metric of 1:

```
R6#show ip ospf database router self-originate


            OSPF Router with ID (150.1.6.6) (Process ID 1)

                Router Link States (Area 0)

  LS age: 1193
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.6.6
  Advertising Router: 150.1.6.6
  LS Seq Number: 80000018
  Checksum: 0xC73B
  Length: 36
  Area Border Router
  Number of Links: 1
 Link connected to: a Virtual Link
 (Link ID) Neighboring Router ID: 150.1.1.1
     (Link Data) Router Interface address: 155.1.146.6
      Number of MTID metrics: 0       TOS 0 Metrics: 1


 Router Link States (Area 1)
```

https://t.me/learningnets

```
  LS age: 1398
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.6.6
  Advertising Router: 150.1.6.6
  LS Seq Number: 80000050
  Checksum: 0xCFA8
  Length: 48
  Area Border Router
  Virtual Link Endpoint
  Number of Links: 2

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.6.6
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.146.6
     (Link Data) Router Interface address: 155.1.146.6
      Number of MTID metrics: 0       TOS 0 Metrics: 1

<output omitted>
```

Now R6 must make its final determination on how to route. If R6 routes through R1 via the virtual-link, the metric is 1 to R1, 1000 from R1 to R5, and R5's metric of 2 to the final destination, for a total of 1003. If R6 routes through R4 via area 1, the metric is 1 to R4, 1 from R4 to R5, and R5's metric of 2 to the final destination, for a total of 4. Under normal conditions, R6 should choose to route through R1 because this is the route through area 0. However, if the **`capability transit`** feature is enabled, which it is by default, R6 can choose the shorter path through area 1 to R4, resulting in the final metric of 4 via R4 being installed in the routing table:

```
R6#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32  Known via "ospf 1", distance 110, metric 4, type inter area
  Last update from 155.1.146.4 on GigabitEthernet1.146, 00:28:39 ago
  Routing Descriptor Blocks: * 155.1.146.4, from 150.1.5.5, 00:28:39 ago, via GigabitEthernet1.146

      Route metric is 4, traffic share count is 1
```

https://t.me/learningnets

If area 1 is not allowed to be used as transit, by issuing the `no capability transit` , R6 cannot choose this path:

```
R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#router ospf 1
R6(config-router)#no capability transit
!
!R6#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32  Known via "ospf 1", distance 110, metric 1003, type inter area
  Last update from 155.1.146.1 on GigabitEthernet1.146, 00:00:06 ago
  Routing Descriptor Blocks: * 155.1.146.1, from 150.1.5.5, 00:00:06 ago, via GigabitEthernet1.146


      Route metric is 1003, traffic share count is 1
```

R6 now says the route is through R1 with a metric of 1003, which matches what we calculated from the OSPF database. The traceroute output is as follows:

```
R6#traceroute 150.1.8.8 numeric
Type escape sequence to abort.
Tracing the route to 150.1.8.8
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.1 1 msec 1 msec 1 msec
  2 155.1.0.5 2 msec 2 msec 1 msec 3 155.1.58.8 2 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Path Selection with Virtual-Links

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R2 with an IP address of 150.1.22.22/32 and advertise it into area 22.
- Configure the OSPF domain so that traffic from R7 going to R2's Loopback100 transits the VLAN 23 link between R3 and R2.

Configuration

```
R2:
interface Loopback100
 ip address 150.1.22.22 255.255.255.255
 ip ospf 1 area 22
!
router ospf 1
 area 5 virtual-link 150.1.3.3
R3:

router ospf 1
 area 5 virtual-link 150.1.2.2
```

Verification

https://t.me/learningnets

Let's see the default path for the traffic, which will follows OSPF design rules, thus for inter-area traffic we cannot transit a non-backbone area. For this reason R3 cannot route through area 2 in order to reach a non-directly connected area. Instead it routes via area 0 through R5, although it is an ABR:

```
R7#show ip route 150.1.22.22
Routing entry for 150.1.22.22/32  Known via "ospf 1", distance 110, metric 2002, type inter area
  Last update from 155.1.37.3 on GigabitEthernet1.37, 00:00:07 ago
  Routing Descriptor Blocks: * 155.1.37.3, from 150.1.3.3, 00:00:07 ago, via GigabitEthernet1.37
      Route metric is 2002, traffic share count is 1
!
!R7#traceroute 150.1.22.22
Type escape sequence to abort.
Tracing the route to 150.1.22.22
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.37.3 3 msec 1 msec 0 msec 2 155.1.0.5 2 msec 1 msec 1 msec


  3 155.1.0.2 2 msec *  3 msec
```

The goal of this section is to modify the path selection process so that area 2 exits via R3 and uses the link through area 5 (VLAN 23). Recall, though, that inter-area routing can only occur through area 0. Based on this design, however, the area 5 link can never be used to reach R2's Loopback100, regardless of the cost. This can be seen as R3's cost towards R2 is 1, while R3's cost towards R5 is 1000, thus the path through R5 has a higher end-to-end metric:

```
R3#show ip ospf database router 150.1.3.3


            OSPF Router with ID (150.1.3.3) (Process ID 1)
 Router Link States (Area 0)


  LS age: 949
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.3.3
  Advertising Router: 150.1.3.3
  LS Seq Number: 800004FE
  Checksum: 0x3239
  Length: 60
  Area Border Router
  Number of Links: 3


    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.3.3
     (Link Data) Network Mask: 255.255.255.255
```

https://t.me/learningnets

```
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.5.5
     (Link Data) Router Interface address: 155.1.0.3
      Number of MTID metrics: 0       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.3
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0

<output omitted> Router Link States (Area 5)

  LS age: 949
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.3.3
  Advertising Router: 150.1.3.3
  LS Seq Number: 8000004B
  Checksum: 0x3807
  Length: 36
  Area Border Router
  Number of Links: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.23.3
     (Link Data) Router Interface address: 155.1.23.3
      Number of MTID metrics: 0       TOS 0 Metrics: 1
```

R3 sets the cost of the link to R2 via area 5 to 1. Even though the cost is lower through this link than through area 0 to R5, the direct route to R2 cannot be used because inter-area routing can only occur through area 0, which is through R5. To utilize the link (VLAN23) between R2 and R3, a virtual-link must be created using area 5 as transit. For virtual-links to be stable even after a device reboot or OSPF process restart, make sure that R2 and R3 have OSPF router-id configured using the **`router-id`** command under OSPF process. Another way to resolve this could be to configure a tunnel between R2 and R3, and then configure OSPF area 0 over it; this is not the optimal or best solution:

```
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R2(config)#router ospf 1
R2(config-router)#area 5 virtual-link 150.1.3.3
```

https://t.me/learningnets

```
!
!R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R3(config)#router ospf 1
R3(config-router)#area 5 virtual-link 150.1.2.2
```

Before doing another trace from R9 toward R2's Loopback100, verify adjacency between R2 and R3 over virtual-link:

```
R2#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.3.3         0   FULL/  -        00:00:30    155.1.23.3      OSPF_VL0
150.1.5.5         0   FULL/  -        00:01:53    155.1.0.5       Tunnel0
150.1.3.3         1   FULL/DR         00:00:38    155.1.23.3      GigabitEthernet1.23
!
!R3#show ip ospf virtual-links
Virtual Link OSPF_VL0 to router 150.1.2.2 is up

  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 5, via interface GigabitEthernet1.23
 Topology-MTID    Cost    Disabled     Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:04
    Adjacency State FULL (Hello suppressed)
    Index 2/5, retransmission queue length 0, number of retransmission 0
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 0, maximum is 0
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

Now that R3 has an area 0 route to R2 via the virtual-link through area 5, and the cost of the area 5 link is 1, R3 chooses this as the shortest path toward R2 (ABR). Based on this, R7 chooses to use R3 as the exit point and the traffic is routed as desired toward R2 using VLAN 23 link, note the change in metric from 2002 to 3:

```
R7#show ip route 150.1.22.22
Routing entry for 150.1.22.22/32  Known via "ospf 1", distance 110, metric 3, type inter area
  Last update from 155.1.37.3 on GigabitEthernet1.37, 00:01:46 ago
  Routing Descriptor Blocks: * 155.1.37.3, from 150.1.3.3, 00:01:46 ago, via GigabitEthernet1.37
      Route metric is 3, traffic share count is 1
!
!R7#traceroute 150.1.22.22
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 150.1.22.22
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.37.3 3 msec 1 msec 1 msec

  2 155.1.23.2 2 msec *  2 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Demand Circuit

> You must load the initial configuration files for the section, **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure the OSPF demand circuit feature between R7 and R9 to reduce periodic OSPF hello transmission and paranoid update flooding.

Configuration

```
R9:

interface GigabitEthernet1.79
 ip ospf demand-circuit
```

Verification

Per RFC 1793, *Extending OSPF to Support Demand Circuits*, “OSPF Hellos and the refresh of OSPF routing information are suppressed on demand circuits, allowing the underlying data-link connections to be closed when not carrying application traffic.” This feature allows low-speed and pay-per-use links, such as analog dial and ISDN, to run OSPF without the need for periodic hellos and LSA flooding. Periodic hellos are only suppressed for *point-to-point* and *point-to-multipoint* OSPF network types. This feature is enabled with the interface-level command `ip ospf demand-circuit` and is negotiated as part of the neighbor adjacency establishment, thus only one OSPF router on the segment requires that the feature be enabled. If routers on the segment do not support it, it will just ignore the option in the HELLO

https://t.me/learningnets

packet, but OSPF neighbors will still be established. Note the difference before and after the feature is enabled, for example on R7.

```
R7#show ip ospf interface gigabitEthernet1.79
GigabitEthernet1.79 is up, line protocol is up
  Internet Address 155.1.79.7/24, Area 2, Attached via Network Statement
  Process ID 1, Router ID 150.1.7.7, Network Type BROADCAST
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 150.1.7.7, Interface address 155.1.79.7
  Backup Designated router (ID) 150.1.9.9, Interface address 155.1.79.9
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:06
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 4/4, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 21
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.9.9  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
!
!R7#show ip ospf interface gigabitEthernet1.79
GigabitEthernet1.79 is up, line protocol is up
  Internet Address 155.1.79.7/24, Area 2, Attached via Network Statement
  Process ID 1, Router ID 150.1.7.7, Network Type BROADCAST
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base Configured as demand circuit
 Run as demand circuit
 DoNotAge LSA allowed
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 150.1.9.9, Interface address 155.1.79.9
  Backup Designated router (ID) 150.1.7.7, Interface address 155.1.79.7
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40 Hello due in 00:00:05
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
```

https://t.me/learningnets

```
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 4/4, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 21
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.9.9  (Designated Router) Suppress hello for 0 neighbor(s)
```

The output above indicates that the Ethernet link between R7 and R9 is set to *Run as demand circuit*, that *DoNotAge LSA* is allowed, and that R9 is going to *Suppress hello for 0 neighbor*, because the OSPF network type is broadcast. This indicates that periodic hellos are still sent and the paranoid flooding of LSAs is disabled. Normally, when an LSA reaches an age of 30 minutes, it must be re-flooded, regardless of whether the network is stable and has not changed.

To verify that LSA flooding is stopped, note that R9 generated LSAs over the configured link have the **DNA** bit set. This is not visible in R9's OSPF database, however, because R9 may have other OSPF adjacencies with this feature not-configured, and therefore the DNA bit is set only when LSAs are sent over links configured as demand circuits.

```
R9#show ip ospf database router self-originate

            OSPF Router with ID (150.1.9.9) (Process ID 1)
 Router Link States (Area 2)
 LS age: 13
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.9.9 Advertising Router: 150.1.9.9
  LS Seq Number: 8000004E
  Checksum: 0xF145
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.9.9
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.79.9
     (Link Data) Router Interface address: 155.1.79.9
```

https://t.me/learningnets

```
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.9.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
!
!R7#show ip ospf database router 150.1.9.9

            OSPF Router with ID (150.1.7.7) (Process ID 1)
 Router Link States (Area 2)
 LS age: 1 (DoNotAge)
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.9.9 Advertising Router: 150.1.9.9

  LS Seq Number: 8000004E
  Checksum: 0xF145
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.9.9
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.79.9
     (Link Data) Router Interface address: 155.1.79.9
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.9.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
```

https://t.me/learningnets

To suppress HELLO packets, let's change the OSPF network type. The lack of periodic hello exchange will also be visible in the `show ip ospf neighbor` output, where the Dead Time field for the adjacency to R9 is null.

```
R9#show ip ospf neighbor
 Neighbor ID     Pri   State Dead Time
   Address         Interface 150.1.7.7         1   FULL/BDR 00:00:36
    155.1.79.7      GigabitEthernet1.79
 !
 !R9#configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#interface gigabitEthernet1.79
R9(config-subif)#ip ospf network point-to-point
 !
 !R7#configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface gigabitEthernet1.79
R7(config-subif)#ip ospf network point-to-point
 !
 !R9#show ip ospf neighbor
 Neighbor ID     Pri   State Dead Time
   Address         Interface 150.1.7.7         0   FULL/  -  -
        155.1.79.7      GigabitEthernet1.79
 !
 !R9#show ip ospf interface gigabitEthernet1.79
 GigabitEthernet1.79 is up, line protocol is up
   Internet Address 155.1.79.9/24, Area 2, Attached via Network Statement
   Process ID 1, Router ID 150.1.9.9, Network Type POINT_TO_POINT, Cost: 1
   Topology-MTID    Cost    Disabled    Shutdown      Topology Name
         0           1         no          no            Base
   Configured as demand circuit
   Run as demand circuit
   DoNotAge LSA allowed
   Transmit Delay is 1 sec, State POINT_TO_POINT
   Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
     oob-resync timeout 40
     Hello due in 00:00:08
   Supports Link-local Signaling (LLS)
   Cisco NSF helper support enabled
   IETF NSF helper support enabled
   Can be protected by per-prefix Loop-Free FastReroute
   Can be used for per-prefix Loop-Free FastReroute repair paths
   Index 2/2, flood queue length 0
   Next 0x0(0)/0x0(0)
   Last flood scan length is 1, maximum is 2
   Last flood scan time is 0 msec, maximum is 1 msec
   Neighbor Count is 1, Adjacent neighbor count is 1
 Adjacent with neighbor 150.1.7.7  (Hello suppressed)
```

https://t.me/learningnets

```
Suppress hello for 1 neighbor(s)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Flooding Reduction

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R5, R8, and R10 to stop periodic paranoid update LSA flooding in area 3.

Configuration

```
R5:
interface GigabitEthernet1.58
 ip ospf flood-reduction
R8:
interface GigabitEthernet1.58
 ip ospf flood-reduction
!
interface GigabitEthernet1.108
 ip ospf flood-reduction
R10:

interface GigabitEthernet1.108
 ip ospf flood-reduction
```

Verification

Per RFC 2328, *OSPF Version 2*, “an LSA's LS age is never incremented past the

https://t.me/learningnets

value MaxAge." When the Link State Age reaches MaxAge, "the router must attempt to flush the LSA... by reflooding the MaxAge LSA just as if it was a newly originated LSA".

In Cisco's IOS implementation of OSPF, the MaxAge value is 3600 seconds, or 60 minutes. To ensure the an LSA is not aged out, which means it will be flushed from the OSPF database, each LSA is reflooded after 30 minutes, regardless of whether the topology is stable or not. This periodic flooding behavior is commonly referred to as the "paranoid update." The `ip ospf flood-reduction` feature stops unnecessary LSA flooding by setting the DoNotAge (DNA) bit in the LSA, removing the requirement for the periodic refresh. This needs to be enabled on links with OSPF neighbors attached, as on the other links, as there are no neighbors, no LSAs are sent anyways.

In the output below, R10 has the (DNA) field next to all LSAs learned from area 3, which indicates that they do not need to be periodically flooded:

```
R10#show ip ospf database


            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Router Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.5.5       150.1.5.5       6 (DNA)
 0x8000004E 0x00F93A 2150.1.8.8       150.1.8.8       1 (DNA)
 0x80000050 0x00F540 4
150.1.10.10     150.1.10.10     119         0x8000004D 0x0045B1 3

                Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum155.1.58.8      150.1.8.8       1 (DNA)
 0x80000001 0x00EA7F155.1.108.8     150.1.8.8       1 (DNA)
 0x80000001 0x004AE3

                Summary Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum150.1.1.1       150.1.5.5       2 (DNA)
 0x80000001 0x006CA8150.1.2.2       150.1.5.5       2 (DNA)
 0x80000001 0x0057BB150.1.3.3       150.1.5.5       2 (DNA)
 0x80000001 0x0042CE150.1.4.4       150.1.5.5       2 (DNA)
 0x80000001 0x0003F6150.1.5.5       150.1.5.5       2 (DNA)
 0x80000001 0x00E315150.1.6.6       150.1.5.5       2 (DNA)
 0x80000001 0x00E212150.1.7.7       150.1.5.5       2 (DNA)
 0x80000001 0x00F710
```

https://t.me/learningnets

```
150.1.9.9       150.1.5.5       2 (DNA)
 0x80000001 0x00D72B 155.1.0.1       150.1.5.5       2 (DNA)
 0x80000001 0x002CE5 155.1.0.2       150.1.5.5       2 (DNA)
 0x80000001 0x0022EE 155.1.0.3       150.1.5.5       2 (DNA)
 0x80000001 0x0018F7 155.1.0.4       150.1.5.5       2 (DNA)
 0x80000001 0x00E316 155.1.0.5       150.1.5.5       2 (DNA)
 0x80000001 0x00CF2A 155.1.7.0       150.1.5.5       2 (DNA)
 0x80000001 0x00FC0D 155.1.9.0       150.1.5.5       2 (DNA)
 0x80000001 0x00F016 155.1.13.0      150.1.5.5       2 (DNA)
 0x80000001 0x00B054 155.1.23.0      150.1.5.5       2 (DNA)
 0x80000001 0x0042B8 155.1.37.0      150.1.5.5       2 (DNA)
 0x80000001 0x00A745 155.1.45.0      150.1.5.5       2 (DNA)
 0x80000001 0x001BB5 155.1.67.0      150.1.5.5       2 (DNA)
 0x80000001 0x006667 155.1.79.0      150.1.5.5       2 (DNA)
 0x80000001 0x00E1DF 155.1.146.0     150.1.5.5       2 (DNA)
 0x80000001 0x00C9A0
```

Verify that interfaces are configured for LSA flooding redution, so that all LSAs sent over these interfaces have the DNA bit set. See the difference with a regular OSPF interface:

```
R5#show ip ospf interface tunnel0
Tunnel0 is up, line protocol is up
  Internet Address 155.1.0.5/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 150.1.5.5, Network Type POINT_TO_MULTIPOINT, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no          no            Base
  Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 3, maximum is 16
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 4, Adjacent neighbor count is 4
    Adjacent with neighbor 150.1.1.1
    Adjacent with neighbor 150.1.4.4
    Adjacent with neighbor 150.1.3.3
    Adjacent with neighbor 150.1.2.2
```

https://t.me/learningnets

```
  Suppress hello for 0 neighbor(s)!
!
!R5#show ip ospf interface gigabitEthernet1.58
GigabitEthernet1.58 is up, line protocol is up
  Internet Address 155.1.58.5/24, Area 3, Attached via Network Statement
  Process ID 1, Router ID 150.1.5.5, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base Reduce LSA flooding.
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 150.1.8.8, Interface address 155.1.58.8
  Backup Designated router (ID) 150.1.5.5, Interface address 155.1.58.5
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 21, maximum is 24
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.8.8  (Designated Router)
  Suppress hello for 0 neighbor(s)
!
!R5#show ip ospf

 Routing Process "ospf 1" with ID 150.1.5.5
 Start time: 5w0d, Time elapsed: 1d16h
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101)
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 It is an area border router
 Router is not originating router-LSAs with maximum metric
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
```

https://t.me/learningnets

```
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 2. 2 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps Area BACKBONE(0)
       Number of interfaces in this area is 3 (1 loopback)
       Area has no authentication
       SPF algorithm last executed 00:05:26.777 ago
       SPF algorithm executed 2 times
       Area ranges are
       Number of LSA 30. Checksum Sum 0x1024BE
       Number of opaque link LSA 0. Checksum Sum 0x000000
       Number of DCbitless LSA 0
       Number of indication LSA 0 Number of DoNotAge LSA 0
       Flood list length 0
   Area 3
       Number of interfaces in this area is 2
       Area has no authentication
       SPF algorithm last executed 00:05:26.777 ago
       SPF algorithm executed 2 times
       Area ranges are
       Number of LSA 27. Checksum Sum 0x0FD72F
       Number of opaque link LSA 0. Checksum Sum 0x000000
       Number of DCbitless LSA 0
       Number of indication LSA 0 Number of DoNotAge LSA 4

       Flood list length 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Clear Text Authentication

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure clear-text OSPF authentication for all adjacencies in area 2 using the password **CLEARKEY**.
  - R7 should enable authentication on all interfaces in area 2 with a single command.
  - R3, R6, and R9 should only enable authentication on their links connecting to R7.

Configuration

```
R3:
interface GigabitEthernet1.37
 ip ospf authentication
 ip ospf authentication-key CLEARKEY
R6:
interface GigabitEthernet1.67
 ip ospf authentication
 ip ospf authentication-key CLEARKEY
R7:
interface GigabitEthernet1.37
 ip ospf authentication-key CLEARKEY
!
interface GigabitEthernet1.67
 ip ospf authentication-key CLEARKEY
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.79
 ip ospf authentication-key CLEARKEY
!
router ospf 1
 area 2 authentication
R9:

interface GigabitEthernet1.79
 ip ospf authentication
 ip ospf authentication-key CLEARKEY
```

Verification

OSPFv2 supports three types of authentication as defined in RFC 2328: type 0, or null authentication (no authentication), type 1, or clear text authentication, and type 2, or Keyed-MD5 authentication. As we'll see in following labs, type 2 also support HMAC-SHA authentication. The type of authentication can be configured at the interface level with the **`ip ospf authentication`** command or at the process level with the **`area [id] authentication`** command. The only difference between these commands is whether authentication is enabled on all interfaces in the area at the same time or on a per-link basis. In either case, the password must still be configured at the interface level with the **`ip ospf authentication-key`** or **`ip ospf message-digest-key`** commands.

Based on how authentication was enabled, you can verify this at the interface or protocol level:

```
R7#show ip ospf
 Routing Process "ospf 1" with ID 150.1.7.7
 Start time: 5w0d, Time elapsed: 1d23h
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101)
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 Router is not originating router-LSAs with maximum metric
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
```

https://t.me/learningnets

```
  LSA group pacing timer 240 secs
  Interface flood pacing timer 33 msecs
  Retransmission pacing timer 66 msecs
  Number of external LSA 0. Checksum Sum 0x000000
  Number of opaque AS LSA 0. Checksum Sum 0x000000
  Number of DCbitless external and opaque AS LSA 0
  Number of DoNotAge external and opaque AS LSA 0
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Number of areas transit capable is 0
  External flood list length 0
  IETF NSF helper support enabled
  Cisco NSF helper support enabled
  Reference bandwidth unit is 100 mbps Area 2
         Number of interfaces in this area is 5 (1 loopback) Area has simple password authentication
         SPF algorithm last executed 04:35:38.930 ago
         SPF algorithm executed 25 times
         Area ranges are
         Number of LSA 29. Checksum Sum 0x12D757
         Number of opaque link LSA 0. Checksum Sum 0x000000
         Number of DCbitless LSA 0
         Number of indication LSA 0
         Number of DoNotAge LSA 0
         Flood list length 0
!
!R7#show ip ospf interface gigabitEthernet1.37
GigabitEthernet1.37 is up, line protocol is up
  Internet Address 155.1.37.7/24, Area 2, Attached via Network Statement
  Process ID 1, Router ID 150.1.7.7, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 150.1.7.7, Interface address 155.1.37.7
  Backup Designated router (ID) 150.1.3.3, Interface address 155.1.37.3
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 21
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
```

https://t.me/learningnets

```
    Adjacent with neighbor 150.1.3.3  (Backup Designated Router)
  Suppress hello for 0 neighbor(s) Simple password authentication enabled
```

Pitfall

A failure in authentication can occur for two reasons: a mismatch in authentication type or a mismatch involving the authentication key. An authentication type mismatch occurs when one neighbor is configured with clear text while the other is running MD5, or one is running MD5 and the other is running null, etc. A key mismatch is simply when the routers are using different passwords. Failure in authentication type can be verified as follows:

```
R9#debug ip ospf adj
OSPF adjacency debugging is on
!R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#interface gigabitEthernet1.79
R9(config-subif)#ip ospf authentication message-digest
!
!OSPF-1 ADJ   Gi1.79: Rcv pkt from 155.1.79.7 :
Mismatched Authentication type. Input packet specified type 1, we use type 2
OSPF-1 ADJ   Gi1.79: Rcv pkt from 155.1.79.7 :
Mismatched Authentication type. Input packet specified type 1, we use type 2
```

The above output from **`debug ip ospf adj`** indicates a *Mismatch Authentication type*, where the inbound OSPF packet is using type 1 and the local router is using type 2. This indicates that the local router is trying to do MD5 authentication and the remote router is using clear text authentication.

https://t.me/learningnets

```
R9#debug ip ospf adj

OSPF adjacency debugging is on
!R9#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#interface gigabitEthernet1.79
R9(config-subif)#ip ospf authentication null

!
! OSPF-1 ADJ   Gi1.79: Rcv pkt from 155.1.79.7 :
Mismatched Authentication type. Input packet specified type 1, we use type 0
OSPF-1 ADJ   Gi1.79: Rcv pkt from 155.1.79.7 :
Mismatched Authentication type. Input packet specified type 1, we use type 0
```

The above output indicates that R9 is doing authentication type 0 which is no authentication, and the remote end is doing clear text authentication (type 1). A mismatch in the password itself can be seen as follows:

```
R9#debug ip ospf adj
OSPF adjacency debugging is on
!R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#interface gigabitEthernet1.79
R9(config-subif)#ip ospf authentication
R9(config-subif)#ip ospf authentication-key testkey
!
!OSPF-1 ADJ   Gi1.79: Rcv pkt from 155.1.79.7,  :Mismatched Authentication Key - Clear Text
OSPF-1 ADJ   Gi1.79: Rcv pkt from 155.1.79.7,  :Mismatched Authentication Key - Clear Text
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF MD5 Authentication

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure MD5-based OSPF authentication for all adjacencies in area 3, using the password **MD5KEY**.
  - R5 should enable MD5 authentication on all interfaces in area 3 with a single command.
  - All other devices in area 3 should enable MD5 authentication on a per-interface basis.

Configuration

```
R5:
router ospf 1
 area 3 authentication message-digest
!
interface GigabitEthernet1.58
 ip ospf message-digest-key 1 md5 MD5KEY
R8:
interface GigabitEthernet1.58
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 MD5KEY
!
interface GigabitEthernet1.108
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 MD5KEY
```

https://t.me/learningnets

```
R10:

interface GigabitEthernet1.108
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 MD5KEY
```

Verification

Before you configure MD5-based authentication, all the interfaces in area 3 must be identified. This can be verified by using the command **`show ip ospf interface brief`** on all the OSPF-enabled devices. For example, we can see the output on R5:

```
R5#show ip ospf interface brief
Interface     PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0           1     0               150.1.5.5/32       1     LOOP  0/0
Gi1.45        1     0               155.1.45.5/24      1     BDR   1/1
Tu0           1     0               155.1.0.5/24       1000  P2MP  4/4 Gi1.5          1      3
               155.1.5.5/24       1     DR    0/0 Gi1.58         1      3
               155.1.58.5/24      1     DR    1/1
```

In the above output, we can see that interfaces GigabitEthernet1.5 and GigabitEthernet1.58 are part of OSPF area 3. In this case we ignore GigabitEthernet1.5 because it is not being used for any OSPF neighborships. Like clear text authentication, MD5 authentication can be enabled on a per-link basis with the interface-level command **`ip ospf authentication message-digest`** , or for all links in the area with the process-level command **`area [id] authentication message-digest`** . For successful MD5 authentication, the authentication type, the password, and the key ID must match. Correctly configured authentication can be verified as follows:

```
R5#show ip ospf
 Routing Process "ospf 1" with ID 150.1.5.5
 Start time: 5w0d, Time elapsed: 1d23h
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101)
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 It is an area border router
 Router is not originating router-LSAs with maximum metric
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
```

https://t.me/learningnets

```
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 2. 2 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 0
 IETF NSF helper support enabled
 Cisco NSF helper support enabled
 Reference bandwidth unit is 100 mbps
    Area BACKBONE(0)
        Number of interfaces in this area is 3 (1 loopback)
        Area has no authentication
        SPF algorithm last executed 06:39:04.766 ago
        SPF algorithm executed 2 times
        Area ranges are
        Number of LSA 30. Checksum Sum 0x0FA3FF
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0 Area 3
        Number of interfaces in this area is 2 Area has message digest authentication
        SPF algorithm last executed 06:30:54.421 ago
        SPF algorithm executed 5 times
        Area ranges are
        Number of LSA 27. Checksum Sum 0x0D55A6
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
!
!R5#show ip ospf interface gigabitEthernet1.58
GigabitEthernet1.58 is up, line protocol is up
  Internet Address 155.1.58.5/24, Area 3, Attached via Network Statement
  Process ID 1, Router ID 150.1.5.5, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
```

https://t.me/learningnets

```
  Designated Router (ID) 150.1.8.8, Interface address 155.1.58.8
  Backup Designated router (ID) 150.1.5.5, Interface address 155.1.58.5
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 24
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.8.8  (Designated Router)
  Suppress hello for 0 neighbor(s) Cryptographic authentication enabled
Youngest key id is 1
```

Pitfall

Remember that a virtual-link is an interface in area 0. This means that if the **area 0 authentication [message-digest]** command is enabled, authentication is also enabled on the virtual-link. If MD5 authentication is enabled on the virtual-link with the **area 0 authentication message-digest** command, the password must still be assigned on the virtual-link interface itself with the **area <NR> virtual-link <RID> message-digest-key <NR> md5 <STRING>** command. Alternatively, authentication can be enabled at the virtual-link interface level with the **area <NR> virtual-link <RID> authentication message-digest** command, and the key is applied with the **area <NR> virtual-link <RID> message-digest-key <NR> md5 <STRING>** command. In some versions, these two commands are combined automatically in the running config to the single statement **area <NR> virtual-link <RID> authentication message-digest message-digest-key <NR> md5 <STRING>** , but the result of either syntax is the same. After authentication is enabled on the virtual-link, make sure to issue the **clear ip ospf process** command, because the virtual-link does not support periodic hellos. This means that if the authentication is wrong the virtual-link interface will not immediately go down, but if there is a change in the topology it won't actually be propagated across the virtual-link.

Failures in MD5 authentication can occur because of a type mismatch, a password mismatch, or a key ID mismatch, as verified below. This output indicates a

https://t.me/learningnets

password mismatch:

```
R10#debug ip ospf adj
OSPF adjacency debugging is on
!R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#interface gigabitEthernet1.108
R10(config-subif)#no ip ospf message-digest-key 1 md5 MD5KEY
R10(config-subif)#ip ospf message-digest-key 1 md5 WRONGKEY
!
!OSPF-1 ADJ   Gi1.108: Rcv pkt from 155.1.108.8 :Mismatched Authentication key - ID 1
OSPF-1 ADJ   Gi1.108: Rcv pkt from 155.1.108.8 :Mismatched Authentication key - ID 1
```

Following messages illustrate a key ID/number mismatch. R10 is using key ID of 2, but R8 is using key ID of 1:

```
R10#debug ip ospf adj
OSPF adjacency debugging is on
!R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#interface gigabitEthernet1.108
R10(config-subif)#no ip ospf message-digest-key 1 md5 MD5KEY
R10(config-subif)#ip ospf message-digest-key 2 md5 MD5KEY
!
!OSPF-1 ADJ   Gi1.108: Rcv pkt from 155.1.108.8 :
Mismatched Authentication Key - Invalid cryptographic authentication Key ID 1 on interface
OSPF-1 ADJ   Gi1.108: Rcv pkt from 155.1.108.8 :
Mismatched Authentication Key - Invalid cryptographic authentication Key ID 1 on interface
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Null Authentication

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Enable clear-text OSPF authentication for the adjacency between R7 and R9 as follows:
  - Use the string of **PASSWORD**.
  - Enable authentication at the area level on R7 and at the interface level on R9

Configuration

```
R7:
router ospf 1
 area 2 authentication
!
interface GigabitEthernet1.79
 ip ospf authentication-key PASSWORD
!
interface GigabitEthernet1.37
 ip ospf authentication null
!
interface GigabitEthernet1.67
 ip ospf authentication null
R9:

interface GigabitEthernet1.79
 ip ospf authentication
```

https://t.me/learningnets

```
ip ospf authentication-key PASSWORD
```

Verification

Type 0 authentication or Null authentication means that basically authentication is disabled, which is the default on all OSPF enabled interface. The use case for this authentication type is when for example you have globally configured clear-text or MD5/SHA authentication for one OSPF area, but want one or multiple interfaces in that area to actually use no authentication. Because the interface level configuration overrides the area level configuration, interfaces for which **`ip ospf authentication null`** is configured will require no OSPF authentication.

In this case, we are asked to enable OSPF authentication between R7 and R9. Enabling authentication for area 2 on R7, requires to disable authentication on R7's interfaces towards R3 and R6. Note that the other links in area 2 are not affected by this change:

```
R7#show ip ospf interface gigabitEthernet1.79
GigabitEthernet1.79 is up, line protocol is up
  Internet Address 155.1.79.7/24, Area 2, Attached via Network Statement
  Process ID 1, Router ID 150.1.7.7, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 150.1.9.9, Interface address 155.1.79.9
  Backup Designated router (ID) 150.1.7.7, Interface address 155.1.79.7
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 4/4, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 21
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.9.9  (Designated Router)
  Suppress hello for 0 neighbor(s) Simple password authentication enabled
!
!R7#show ip ospf interface gigabitEthernet1.67
```

https://t.me/learningnets

```
GigabitEthernet1.67 is up, line protocol is up
  Internet Address 155.1.67.7/24, Area 2, Attached via Network Statement
  Process ID 1, Router ID 150.1.7.7, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 150.1.7.7, Interface address 155.1.67.7
  Backup Designated router (ID) 150.1.6.6, Interface address 155.1.67.6
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:09
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 17
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.6.6  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
```

Verify that R7 is still OSPF neighbor with R3, R6 and R9:

```
R7#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.9.9          1   FULL
/DR         00:00:38    155.1.79.9      GigabitEthernet1.79 150.1.6.6          1   FULL
/BDR        00:00:32    155.1.67.6      GigabitEthernet1.67 150.1.3.3          1   FULL
/BDR        00:00:37    155.1.37.3      GigabitEthernet1.37
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF MD5 Authentication with Multiple Keys

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Enable OSPF MD5 authentication over the DMVPN cloud as follows:
  - Use a key ID of 1 with the string **KEYONE** between R1, R2 and R5.
  - Use a key ID of 2 with the string **KEYTWO** between R3, R4 and R5.

Configuration

```
R1 - R2:
interface Tunnel0
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 KEYONE
R3 - R4:
interface Tunnel0
 ip ospf authentication message-digest
 ip ospf message-digest-key 2 md5 KEYTWO
R5:

interface Tunnel0
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 KEYONE
 ip ospf message-digest-key 2 md5 KEYTWO
```

https://t.me/learningnets

Verification

OSPF supports multiple MD5 keys applied to a single interface to allow for key rotation. The normal application of this would be that all neighbors are configured with key 1. Key 2 is then added on all neighbors, and key 1 is removed afterwards. Because both keys are temporarily sent (baiscally R5 will duplicate all OSPF packets it needs to send over its Tunnel interface, sending the same packet authenticated with each configured key), there is no loss of adjacency while the old key is being removed. In this design, multiple keys are used to authenticate different neighbors on the same interface. Verify that authentication is enabled and routers are still OSPF neighbors:

```
R5#show ip ospf neighbor tunnel0

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.1.1          0   FULL/
  -        00:01:56    155.1.0.1       Tunnel0 150.1.4.4          0   FULL/
  -        00:01:56    155.1.0.4       Tunnel0 150.1.3.3          0   FULL/
  -        00:01:56    155.1.0.3       Tunnel0 150.1.2.2          0   FULL/
  -        00:01:56    155.1.0.2       Tunnel0
!
!R5#show ip ospf interface tunnel0
Tunnel0 is up, line protocol is up
  Internet Address 155.1.0.5/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 150.1.5.5, Network Type POINT_TO_MULTIPOINT, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no          no            Base
  Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:18
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 16
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 4, Adjacent neighbor count is 4
    Adjacent with neighbor 150.1.1.1
    Adjacent with neighbor 150.1.4.4
    Adjacent with neighbor 150.1.3.3
    Adjacent with neighbor 150.1.2.2
```

https://t.me/learningnets

```
  Suppress hello for 0 neighbor(s) Cryptographic authentication enabled
 Youngest key id is 2
 Rollover in progress, 2 neighbor(s) using the old key(s):
 key id 1 algorithm MD5
```

Verify that R5 sends OSPF packets authenticated with same keys and receives packets from R1 - R2 with ley ID of 1 and from R3 - R4 with key ID of 2:

```
R5#debug ip ospf adj
OSPF adjacency debugging is on
!R5#debug ip ospf packet
OSPF packet debugging is on
!
!OSPF-1 ADJ Tu0: Send with key 1
OSPF-1 ADJ Tu0: Send with key 2
!
!OSPF-1 PAK  : rcv. v:2 t:1 l:48 rid:150.1.1.1 aid:0.0.0.0 chk:0 aut:2 keyid:1
 seq:0x53974FDE from Tunnel0 OSPF-1 PAK  : rcv. v:2 t:1 l:48
 rid:150.1.2.2 aid:0.0.0.0 chk:0 aut:2 keyid:1
 seq:0x53974FDE from Tunnel0 OSPF-1 PAK  : rcv. v:2 t:1 l:48
 rid:150.1.3.3 aid:0.0.0.0 chk:0 aut:2 keyid:2
 seq:0x53974FDE from Tunnel0 OSPF-1 PAK  : rcv. v:2 t:1 l:48
 rid:150.1.4.4 aid:0.0.0.0 chk:0 aut:2 keyid:2
 seq:0x53973FA7 from Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Internal Summarization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R5 to advertise a summary route for VLAN 8 and VLAN 10 prefixes as they are sent into area 0.
- This summary should be as specific as possible while still encompassing all addresses in both subnets.

Configuration

```
R5:

router ospf 1
 area 3 range 155.1.8.0 255.255.252.0
```

Verification

Because devices in an OSPF area require the same copy of the database to compute correct SPF, filtering or summarization of routes in the database can only occur between areas or domains, not within an area. The below configuration illustrates how an Intra-Area Summary Network LSA (LSA 3) can be summarized as it is originated by an ABR.

Without any modification, R1 sees two separate routes and LSAs to reach the

https://t.me/learningnets

networks 155.1.8.0/24 and 155.1.10.0/24. Because they are both originated by R5 (150.1.5.5), it implies that only R5 can modify R1's view of these. By configuring the `area 3 range 155.1.8.0 255.255.252.0` command, R5 stops sending the specific LSAs from area 3 into area 0 and groups them into the single route 155.1.8.0/22. Verify the outputs on R1 before configuration changes:

```
R1#show ip ospf database summary 155.1.8.0 adv-router 150.1.5.5

            OSPF Router with ID (150.1.1.1) (Process ID 1)
 Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 96
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 155.1.8.0 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 8000001C
  Checksum: 0x8752
  Length: 28 Network Mask: /24
        MTID: 0         Metric: 2
!
!R1#show ip route 155.1.8.0
 Routing entry for 155.1.8.0/24
  Known via "ospf 1", distance 110, metric 1002, type inter area
  Last update from 155.1.0.5 on Tunnel0, 00:00:40 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:00:40 ago, via Tunnel0
      Route metric is 1002, traffic share count is 1
!
!R1#show ip ospf database summary 155.1.10.0 adv-router 150.1.5.5

            OSPF Router with ID (150.1.1.1) (Process ID 1)
 Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 125
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 155.1.10.0 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xB140
  Length: 28 Network Mask: /24
        MTID: 0         Metric: 3
!
!R1#show ip route 155.1.10.0
 Routing entry for 155.1.10.0/24
```

https://t.me/learningnets

```
  Known via "ospf 1", distance 110, metric 1003, type inter area
  Last update from 155.1.0.5 on Tunnel0, 00:00:57 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:00:57 ago, via Tunnel0

      Route metric is 1003, traffic share count is 1
```

Verify the OSPF database entries and routing table of R1 after configuration changes, now R3 advertises a single Type3 LSA for 155.1.8.0/22:

```
R1#show ip ospf database summary 155.1.8.0 adv-router 150.1.5.5

            OSPF Router with ID (150.1.1.1) (Process ID 1)
 Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 23
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 155.1.8.0 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 8000001D
  Checksum: 0x7665
  Length: 28 Network Mask: /22
        MTID: 0         Metric: 2
!
!R1#show ip route 155.1.8.0
Routing entry for 155.1.8.0/22
  Known via "ospf 1", distance 110, metric 1002, type inter area
  Last update from 155.1.0.5 on Tunnel0, 00:00:40 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:00:40 ago, via Tunnel0
      Route metric is 1002, traffic share count is 1
!
!R1#show ip ospf database summary 155.1.10.0 adv-router 150.1.5.5

            OSPF Router with ID (150.1.1.1) (Process ID 1) R1#
!
!R1#show ip route 155.1.10.0
Routing entry for 155.1.8.0/22
  Known via "ospf 1", distance 110, metric 1002, type inter area
  Last update from 155.1.0.5 on Tunnel0, 00:00:54 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:00:54 ago, via Tunnel0

      Route metric is 1002, traffic share count is 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Path Selection with Summarization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R4 to originate the summary route 155.1.146.0/23 so that traffic from R5 destined to VLAN 146 transits R1.
  - If R5's connection to DMVPN cloud is down, traffic for VLAN 146 should transit R4.

Configuration

```
R4:

router ospf 1
 area 1 range 155.1.146.0 255.255.254.0
```

Verification

https://t.me/learningnets

When a router does a routing lookup on a destination, it always chooses the longest match route for the path. This means that if the router is trying to reach the destination 1.2.3.4, it will choose the route 1.2.3.0/24 over 1.2.0.0/18, or 1.2.0.0/16 over 0.0.0.0/0. This principle can be used for traffic engineering purposes by selectively summarizing prefixes into the IGP domain, as seen below. Verify how R5 routes towards VLAN 146 before configuration changes, best path being via R4 due to the better metric towards the ABR (R4 against R1):

```
R4#show ip ospf database summary 155.1.146.0 adv-router 150.1.1.1

            OSPF Router with ID (150.1.4.4) (Process ID 1)

                Summary Net Link States (Area 0)

  LS age: 91
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 155.1.146.0 (summary Network Number)
 Advertising Router: 150.1.1.1
  LS Seq Number: 80000073
  Checksum: 0xFF1
  Length: 28 Network Mask: /24
        MTID: 0 Metric: 1
!
!R4#show ip ospf database summary 155.1.146.0 adv-router 150.1.4.4

            OSPF Router with ID (150.1.4.4) (Process ID 1)

                Summary Net Link States (Area 0)

  LS age: 56
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 155.1.146.0 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 8000007F
  Checksum: 0xCF1F
  Length: 28 Network Mask: /24
        MTID: 0 Metric: 1
!
!R5#show ip ospf border-routers

            OSPF Router with ID (150.1.5.5) (Process ID 1)


                Base Topology (MTID 0)
```

https://t.me/learningnets

```
Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 150.1.1.1 [1000]
 via 155.1.0.1, Tunnel0, ABR, Area 0, SPF 1840
i 150.1.2.2 [1000] via 155.1.0.2, Tunnel0, ABR, Area 0, SPF 1840
i 150.1.3.3 [1000] via 155.1.0.3, Tunnel0, ABR, Area 0, SPF 1840i 150.1.4.4 [1]
 via 155.1.45.4, GigabitEthernet1.45, ABR, Area 0, SPF 1840
!
!R5#show ip route 155.1.146.0
Routing entry for 155.1.146.0/24
  Known via "ospf 1", distance 110,metric 2, type inter area
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:02:42 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.4.4, 00:02:42 ago, via GigabitEthernet1.45
      Route metric is 2, traffic share count is 1
!
!R5#traceroute 155.1.146.6
Type escape sequence to abort.
Tracing the route to 155.1.146.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 3 msec 1 msec 2 msec

  2 155.1.146.6 3 msec *  5 msec
```

After applying the configuration changes, based on the longest route match, traffic will be routed thorugh R1 via the DMVPN cloud, so metrics are no longer compared between the path via R1 and path via R4:

```
R5#show ip ospf database summary 155.1.146.0 adv-router 150.1.1.1

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 486
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 155.1.146.0 (summary Network Number)
Advertising Router: 150.1.1.1
  LS Seq Number: 80000073
  Checksum: 0xFF1
  Length: 28 Network Mask: /24
        MTID: 0         Metric: 1
!
!R5#show ip ospf database summary 155.1.146.0 adv-router 150.1.4.4

            OSPF Router with ID (150.1.5.5) (Process ID 1)
```

https://t.me/learningnets

```
                Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 32
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 155.1.146.0 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 80000080
  Checksum: 0xC826
  Length: 28 Network Mask: /23
        MTID: 0         Metric: 1
!
!R5#show ip route 155.1.146.0
Routing entry for 155.1.146.0/24
  Known via "ospf 1", distance 110, metric 1001, type inter area
  Last update from 155.1.0.1 on Tunnel0, 00:00:44 ago
  Routing Descriptor Blocks: * 155.1.0.1, from 150.1.1.1, 00:00:44 ago, via Tunnel0
      Route metric is 1001, traffic share count is 1
!
!R5#show ip route 155.1.146.0 255.255.254.0 longer-prefixes
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      155.1.0.0/16 is variably subnetted, 24 subnets, 3 masks
O IA     155.1.146.0/23 [110/2] via 155.1.45.4, 00:01:10, GigabitEthernet1.45
O IA     155.1.146.0/24 [110/1001] via 155.1.0.1, 00:01:10, Tunnel0
!
!R5#traceroute 155.1.146.6
Type escape sequence to abort.
Tracing the route to 155.1.146.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 3 msec 1 msec 1 msec

  2 155.1.146.6 9 msec *  5 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF External Summarization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R9 as follows:
  - Create Loopback100 with IP address of 160.1.9.100/32 and Loopback200 with IP address of 160.1.9.200/32.
  - Redistribute these prefixes into OSPF and ensure all OSPF routers use a metric of 50.
  - Advertise a single /24 summary for these prefixes.
- Configure R10 as follows:
  - Create Loopback100 with IP address of 160.1.10.100/32 and Loopback200 with IP address of 160.1.10.200/32.
  - Redistribute these prefixes into OSPF with a cost of 100 and ensure all OSPF routers add the cost to reach R10 to the metric.
  - Advertise a /24 summary for these prefixes.

Configuration

```
R9:
interface Loopback100
 ip address 160.1.9.100 255.255.255.255
!
interface Loopback200
 ip address 160.1.9.200 255.255.255.255
```

https://t.me/learningnets

```
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100 Loopback200
!
router ospf 1
 redistribute connected metric 50 subnets route-map CONNECTED->OSPF
 summary-address 160.1.9.0 255.255.255.0
R10:

interface Loopback100
 ip address 160.1.10.100 255.255.255.255
!
interface Loopback200
 ip address 160.1.10.200 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100 Loopback200
!
router ospf 1
 redistribute connected metric 100 metric-type 1 subnets route-map CONNECTED->OSPF
 summary-address 160.1.10.0 255.255.255.0
```

Verification

External OSPF summarization is configured at the redistribution point between routing domains with the **`summary-address`** command. These summaries inherit their attributes from the subnets that make them up. For example, a summary comprised of External Type-1 routes will result in an External Type-1 summary. This means that on R10 in this configuration, the **`metric-type 1`** command is set at the time of redistribution instead of on the summary itself. External Type-2 OSPF routes, which are the default, do not install the end-to-end metric in the routing table. Instead, only the metric that was reported via the ASBR is installed. The actual routing path is determined by the addition of the reported metric and the metric toward the ASBR, which is called the forward metric:

Verify that redisstributed prefixes are summarized with correct metric and metric-type, and routers in the OSPF domain have IP reachability with the redistributed prefixes:

```
R1#show ip ospf database external

            OSPF Router with ID (150.1.1.1) (Process ID 1)

                Type-5 AS External Link States
```

https://t.me/learningnets

```
  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 207
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.9.0 (External Network Number )
Advertising Router: 150.1.9.9
  LS Seq Number: 80000001
  Checksum: 0xB479
  Length: 36 Network Mask: /24
Metric Type: 2 (Larger than any link state path)
        MTID: 0  Metric: 50
        Forward Address: 0.0.0.0
        External Route Tag: 0

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 201
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.10.0 (External Network Number )
Advertising Router: 150.1.10.10
  LS Seq Number: 80000001
  Checksum: 0xF6A
  Length: 36 Network Mask: /24
Metric Type: 1 (Comparable directly to link state metric)
        MTID: 0  Metric: 100
        Forward Address: 0.0.0.0
        External Route Tag: 0
!
!R1#show ip route ospf | i O E
O E2     160.1.9.0 [110/50]
 via 155.1.0.5, 00:03:01, Tunnel0 O E1     160.1.10.0 [110/1102]
 via 155.1.0.5, 00:02:51, Tunnel0
!
!R1#ping 160.1.9.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.9.100, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
!
!R1#ping 160.1.9.200
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.9.200, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
!
!R1#ping 160.1.10.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.10.100, timeout is 2 seconds:!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/6 ms
!
!R1#ping 160.1.10.200
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.10.200, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Verify that both R9 and R10 install routes to Null0 for the created summaries:

```
R9#show ip route 160.1.9.0
Routing entry for 160.1.9.0/24
  Known via "ospf 1", distance 254, metric 50, type intra area
  Routing Descriptor Blocks: * directly connected, via Null0
      Route metric is 50, traffic share count is 1
!
!R10#show ip route 160.1.10.0
Routing entry for 160.1.10.0/24
  Known via "ospf 1", distance 254, metric 100, type intra area
  Routing Descriptor Blocks: * directly connected, via Null0

      Route metric is 100, traffic share count is 1
```

Let's identify how the metric is computed for both external route types, starting with E2, from R5's perspective. Being a E2 route, the metric showing up in the routing table is the one which was set by the ASBR at redistribution:

```
R5#show ip ospf database external 160.1.9.0

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 625
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.9.0 (External Network Number )
Advertising Router: 150.1.9.9
  LS Seq Number: 80000001
  Checksum: 0xB479
  Length: 36 Network Mask: /24
Metric Type: 2 (Larger than any link state path)
        MTID: 0  Metric: 50
Forward Address: 0.0.0.0
```

https://t.me/learningnets

```
        External Route Tag: 0
!
!R5#show ip route 160.1.9.0
Routing entry for 160.1.9.0/24  Known via "ospf 1", distance 110,
metric 50, type extern 2, forward metric 1002

  Last update from 155.1.0.3 on Tunnel0, 00:08:53 ago
  Routing Descriptor Blocks:
  * 155.1.0.3, from 150.1.9.9, 00:08:53 ago, via Tunnel0
      Route metric is 50, traffic share count is 1
```

In the above output, R5 sees the summary 160.1.9.0/24 as an External Type-2 route originated by the ASBR 150.1.9.9. The *Forward Address: 0.0.0.0* field means that R5 must now compute the metric toward the advertising router, R9, and install this metric in the routing table as the forward metric. Specifically, this is calculated as follows: first, R5 needs to identify the metric towards the ABR generating the Type4 LSA, which in this case is R3 (because is the router with links in the same area as the ASBR, R9, and links in the same area as R5). The cost to reach R3 is 1000:

```
R5#show ip ospf database router adv-router 150.1.5.5

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

  LS age: 1924
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000643
  Checksum: 0x4084
  Length: 108
  Area Border Router
  Number of Links: 7

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.45.4
     (Link Data) Router Interface address: 155.1.45.5
```

https://t.me/learningnets

```
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.1.1
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.4.4
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.3.3
 (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1000


    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.2.2
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

Next, the metric to reach the ASBR is signaled through a Type4 LSA, generated by R3, thus R5 needs to know the cost from this LSA and add it to the cost to reach R3 in order to compute the forwarding metric of 1002. The routing table output for the specific routing lookup should now show a metric of 50 to the destination, but a forward metric of 1002 toward the ASBR. This forward metric is used to determine the path toward the exit point:

```
R5#show ip ospf database asbr-summary adv-router 150.1.3.3


            OSPF Router with ID (150.1.5.5) (Process ID 1)


                Summary ASB Link States (Area 0)
```

https://t.me/learningnets

```
  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1089
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(AS Boundary Router) Link State ID: 150.1.9.9 (AS Boundary Router address)
 Advertising Router: 150.1.3.3
  LS Seq Number: 80000002
  Checksum: 0xA34E
  Length: 28
  Network Mask: /0        MTID: 0 Metric: 2
!
!R5#show ip ospf border-routers

            OSPF Router with ID (150.1.5.5) (Process ID 1)



                Base Topology (MTID 0)

Internal Router Routing Table
Codes: i - Intra-area route, I - Inter-area route

i 150.1.1.1 [1000] via 155.1.0.1, Tunnel0, ABR, Area 0, SPF 1840
i 150.1.10.10 [2] via 155.1.58.8, GigabitEthernet1.58, ASBR, Area 3, SPF 28
i 150.1.2.2 [1000] via 155.1.0.2, Tunnel0, ABR, Area 0, SPF 1840
i 150.1.3.3 [1000] via 155.1.0.3, Tunnel0, ABR, Area 0, SPF 1840
i 150.1.4.4 [1] via 155.1.45.4, GigabitEthernet1.45, ABR, Area 0, SPF 1840
I 150.1.9.9 [1002] via 155.1.0.3
, Tunnel0, ASBR, Area 0, SPF 1840
!
!R5#show ip route 160.1.9.0
Routing entry for 160.1.9.0/24  Known via "ospf 1", distance 110, metric 50, type extern 2,
forward metric 1002
  Last update from 155.1.0.3 on Tunnel0, 00:51:02 ago
  Routing Descriptor Blocks:
  * 155.1.0.3, from 150.1.9.9, 00:51:02 ago, via Tunnel0
      Route metric is 50, traffic share count is 1
!
!R5#traceroute 160.1.9.100
Type escape sequence to abort.
Tracing the route to 160.1.9.100
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.3 4 msec 1 msec 1 msec

  2 155.1.37.7 6 msec 2 msec 1 msec
  3 155.1.79.9 2 msec *  3 msec
```

The calculation for External Type-1 OSPF routes does not distinguish in the routing

https://t.me/learningnets

table between the metric reported by the ASBR and the metric to the ASBR via the forward metric. Instead, External Type-1 routes represent the metric as one cumulative value of the reported metric and the metric to the ASBR:

```
R5#show ip route 160.1.10.0
Routing entry for 160.1.10.0/24  Known via "ospf 1", distance 110, metric 102, type extern 1

  Last update from 155.1.58.8 on GigabitEthernet1.58, 00:53:06 ago
  Routing Descriptor Blocks:
  * 155.1.58.8, from 150.1.10.10, 00:53:06 ago, via GigabitEthernet1.58
      Route metric is 102, traffic share count is 1
```

Let's calculate the metric of 102 from R5's perspective. Note that in this case, as R5 is member in the same area as the ASBR, R5 does not need any Type4 LSA to calculate the end-to-end metric, however R5 will generate a Type4 LSA into area 0 announcing its metric towards the ASBR which is R10:

```
R5#show ip ospf database asbr-summary adv-router 150.1.5.5

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary ASB Link States (Area 0)

  LS age: 1347
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(AS Boundary Router) Link State ID: 150.1.10.10 (AS Boundary Router address)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000002
  Checksum: 0x7477
  Length: 28
  Network Mask: /0        MTID: 0 Metric: 2



                Summary ASB Link States (Area 3)

  LS age: 1347
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(AS Boundary Router)
  Link State ID: 150.1.9.9 (AS Boundary Router address)
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000002
  Checksum: 0xBD44
  Length: 28
  Network Mask: /0
```

https://t.me/learningnets

```
        MTID: 0         Metric: 1002
```

From the above outputs, R5's metric of 102 to reach 160.1.10.0/24, is composed of its cost of 2 towards the ASBR and the ASBR reported cost of 100 at the redistribution point. Let's take a close look into this, first see the ASBR reported cost of 100:

```
R5#show ip ospf database external adv-router 150.1.10.10

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1475
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.10.0 (External Network Number )
 Advertising Router: 150.1.10.10
  LS Seq Number: 80000002
  Checksum: 0xD6B
  Length: 36 Network Mask: /24
        Metric Type: 1 (Comparable directly to link state metric)
        MTID: 0  Metric: 100
 Forward Address: 0.0.0.0

        External Route Tag: 0
```

As the ASBR is in the same area with R5, R5 needs to find the intra-area cost towards R10, which equals 2. First R5 sees that R10 is attached to VLAN 108, where the DR is R10, but R8 is attached to the segment as well:

```
R5#show ip ospf database router adv-router 150.1.10.10

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1499
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.10.10
  Advertising Router: 150.1.10.10
  LS Seq Number: 8000007B
```

https://t.me/learningnets

```
  Checksum: 0x1BA9
  Length: 60
  AS Boundary Router
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.10.10
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.108.10
 (Link Data) Router Interface address: 155.1.108.10
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.10.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
!
!R5#show ip ospf database network adv-router 150.1.10.10

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 99
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.108.10 (address of Designated Router) Advertising Router: 150.1.10.10
  LS Seq Number: 80000019
  Checksum: 0xDB34
  Length: 32 Network Mask: /24
 Attached Router: 150.1.10.10
 Attached Router: 150.1.8.8
```

Next R5 sees that R8 is connected to itself on VLAN 58, where the DR is R8:

```
R5#show ip ospf database router adv-router 150.1.8.8

            OSPF Router with ID (150.1.5.5) (Process ID 1)
```

https://t.me/learningnets

```
                Router Link States (Area 3)

  LS age: 129
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.8.8
  Advertising Router: 150.1.8.8
  LS Seq Number: 8000007F
  Checksum: 0xC341
  Length: 72
  Number of Links: 4

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.8.8
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 155.1.108.10
     (Link Data) Router Interface address: 155.1.108.8
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.58.8
 (Link Data) Router Interface address: 155.1.58.8
      Number of MTID metrics: 0       TOS 0 Metrics: 1

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.8.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
!
!R5#show ip ospf database network adv-router 150.1.8.8

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 408
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.58.8 (address of Designated Router)
```

https://t.me/learningnets

```
Advertising Router: 150.1.8.8

  LS Seq Number: 8000001E
  Checksum: 0xB09C
  Length: 32 Network Mask: /24
Attached Router: 150.1.8.8
Attached Router: 150.1.5.5
```

Basically R5 adds its cost on VLAN 58 to reach the DR, which equals 1 to the cost of R8's cost on VLAN 108 to reach the DR, which equals 1 as well, thus R5's cost to reach R10 equals 2. The metric from the routing table is composed of this value and the ASBR advertised cost for the prefix:

```
R5#show ip route 160.1.10.0
Routing entry for 160.1.10.0/24  Known via "ospf 1", distance 110, metric 102, type extern 1
  Last update from 155.1.58.8 on GigabitEthernet1.58, 01:09:08 ago
  Routing Descriptor Blocks: * 155.1.58.8, from 150.1.10.10, 01:09:08 ago, via GigabitEthernet1.58
      Route metric is 102, traffic share count is 1
!
!R5#traceroute 160.1.10.100
Type escape sequence to abort.
Tracing the route to 160.1.10.100
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.58.8 3 msec 2 msec 1 msec

  2 155.1.108.10 2 msec *  4 msec
```

Pitfall

Remember that the OSPF router-ID needs to be unique between routers within same area and recommended to be unique between all routers in the OSPF domain. The only case when it matters that a router from let's say area 3 does not have the same router-ID as a router in area 5, is when either of these two routers are ASBR's, thus perform redistribution in the OSPF domain. Because the Type5 LSA is flooded unmodified within the OSPF domain and contains the router-ID of the ASBR, this represents a potential problem. If a router reveives a Type5 LSA and the originator (routerd-ID of the ASBR) matches its own router-ID, it thinks it is receiving its own advertisement back. However, because it is actually not originating these routes, it sends a withdraw message back. The specifics behind this problem can be seen in section 13.4 “Receiving self-originated LSAs” of RFC 2328, OSPF Version 2. The end result is that the router cannot install the external

https://t.me/learningnets

routes in the routing table. Moreover, if both routers with same router-ID are ASBRs, all the routers in the topology will undertake a continous process of installing and withdrawing the external OSPF routes from OSPF database and routing table.

Let's simulate the failure by configuring both R9 and R10 with the same router-ID:

```
R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#router ospf 1
R9(config-router)#router-id 90.90.90.90
% OSPF: Reload or use "clear ip ospf process" command, for this to take effectR9#clear ip ospf process
Reset ALL OSPF processes? [no]: yes
!
!R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#router ospf 1
R10(config-router)#router-id 90.90.90.90
% OSPF: Reload or use "clear ip ospf process" command, for this to take effectR10#clear ip ospf process

Reset ALL OSPF processes? [no]: yes
```

The most visible sign of trouble is the following log messages on R9 and R10, basically each router will flush/invalidate the Type5 LSA of the other router and re-originate its own Type5 LSA which was flushed/invalidated as well by the remote router. This is visible on all routers in the OSPF domain as external routes will be removed and re-added in both the OSPF database and routing tables:

```
R9:
%OSPF-4-FLOOD_WAR: Process 1 flushes LSA ID 160.1.10.0 type-5 adv-rtr 90.90.90.90 in area 2
%OSPF-4-FLOOD_WAR: Process 1 re-originates LSA ID 160.1.9.0 type-5 adv-rtr 90.90.90.90 in area 2
!
!R10:

%OSPF-4-FLOOD_WAR: Process 1 flushes LSA ID 160.1.9.0 type-5 adv-rtr 90.90.90.90 in area 3
%OSPF-4-FLOOD_WAR: Process 1 re-originates LSA ID 160.1.10.0 type-5 adv-rtr 90.90.90.90 in area 3
```

Verify that both R9 and R10 will signal the flushing of external routes generated by remote router, by setting the maxage value for the LSA:

```
R9#show ip ospf database external

            OSPF Router with ID (90.90.90.90) (Process ID 1)

                Type-5 AS External Link States
```

https://t.me/learningnets

```
  LS age: 3
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.9.0 (External Network Number )
  Advertising Router: 90.90.90.90
  LS Seq Number: 8000006B
  Checksum: 0x12F1
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 50
        Forward Address: 0.0.0.0
        External Route Tag: 0
 Delete flag is set for this LSA
 LS age: MAXAGE(3603)
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.10.0 (External Network Number )
 Advertising Router: 90.90.90.90
  LS Seq Number: 8000006A
  Checksum: 0x1323
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 16777215
        Forward Address: 0.0.0.0
        External Route Tag: 0
!
!R10#show ip ospf database external

            OSPF Router with ID (90.90.90.90) (Process ID 1)

                Type-5 AS External Link States
 Delete flag is set for this LSA
 LS age: MAXAGE(3603)
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.9.0 (External Network Number )
 Advertising Router: 90.90.90.90

  LS Seq Number: 8000006E
  Checksum: 0x161D
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
```

https://t.me/learningnets

```
        MTID: 0
        Metric: 16777215
        Forward Address: 0.0.0.0
        External Route Tag: 0

    LS age: 2
    Options: (No TOS-capability, DC, Upward)
    LS Type: AS External Link
    Link State ID: 160.1.10.0 (External Network Number )
    Advertising Router: 90.90.90.90
    LS Seq Number: 8000006D
    Checksum: 0x75D9
    Length: 36
    Network Mask: /24
        Metric Type: 1 (Comparable directly to link state metric)
        MTID: 0
        Metric: 100
        Forward Address: 0.0.0.0
        External Route Tag: 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Stub Areas

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R4 and R6 with IP addressing in the format of 160.1.Y.Y/32, where **Y** is the router number.
  - Redistribute these prefixes into OSPF.
- Configure OSPF area 3 so that R5 filters external routes out as they are sent from area 0 to area 3.
  - Devices in area 3 should still have reachability to routes external to the OSPF domain.

Configuration

```
R5 , R8 , R10:
router ospf 1
 area 3 stub
R4:
interface Loopback100
 ip address 160.1.4.4 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

https://t.me/learningnets

```
R6:

interface Loopback100
 ip address 160.1.6.6 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

OSPF stub area types are used to filter information out of the OSPF database based on LSA Type. The stub flag is part of the OSPF adjacency formation, which implies that all devices in an area must agree on that parameter for adjacencies to establish. The four stub types that IOS supports are stub areas, totally stubby areas, not-so-stubby areas (NSSA), and not-so-totally-stubby areas. The first option, the stub area, is used to remove Type-5 External link states from the database and replace them with a default route. The logic behind this feature stems from how external lookups between areas occur in OSPF.

When an OSPF router redistributes a route into the domain, it originates a Type-5 External LSA representing the route and its attributes. Inside this LSA, the originating router sets the *advertising router* field to its local router-id and, generally, the *forward address* field to 0.0.0.0.

When an OSPF router in the same area as the originator looks up the Type-5 LSA, it looks at the forward address. If the forward address is set to 0.0.0.0, it means that the traffic should be sent toward the advertising router to reach the destination. To find out how to reach the advertising router, the advertising router's Type-1 Router LSA is consulted, and intra-area SPF is performed. This is similar to inter-area routing logic, because the router doing the lookup does not compute SPF to the final destination, only the intermediary advertising router. For external routing between areas, the logic is modified slightly.

When an Area Border Router receives a Type-5 External LSA from a device in its own area and passes it into a different area, a Type-4 ASBR Summary LSA is generated. The Type-4 LSA tells devices in the new area how to forward toward the ASBR, which in turn tells them how to forward toward the external route. For example, examine the following situation in this topology. R4 redistributes the directly connected route 160.1.4.4/32 into OSPF, originating a Type-5 External LSA:

https://t.me/learningnets

```
R4#show ip ospf database external 160.1.4.4


            OSPF Router with ID (150.1.4.4) (Process ID 1)

                Type-5 AS External Link States

  LS age: 36
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.4.4 (External Network Number ) Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0xD77F
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 0.0.0.0

        External Route Tag: 0
```

When R5 wants to reach the destination 160.1.4.4/32, it sees that the forward address is 0.0.0.0 and the advertising router is 150.1.4.4. R5 now does a Type-1 Router LSA lookup on 150.1.4.4 (R4):

```
R5#show ip ospf database router 150.1.4.4


            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 94
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.4.4
  Advertising Router: 150.1.4.4
  LS Seq Number: 800002E3
  Checksum: 0x884A
  Length: 72
  Area Border Router
  AS Boundary Router
  Number of Links: 4

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.4.4
```

https://t.me/learningnets

```
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.45.4
 (Link Data) Router Interface address: 155.1.45.4
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.5.5
 (Link Data) Router Interface address: 155.1.0.4

      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.4
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

From this R5 knows that R4 is directly adjacent via two paths, VLAN 45 with a metric of 1 and DMVPN cloud with a metric of 1000:

```
R5#show ip route 160.1.4.4
Routing entry for 160.1.4.4/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 1
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:03:10 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.4.4, 00:03:10 ago, via GigabitEthernet1.45

      Route metric is 20, traffic share count is 1
```

R5 installs the path via VLAN45 in the routing table with a metric of 20 from the Type-5 External LSA, and a forward metric of 1 to reach R4 via VLAN 45. If this route were redistributed as Type-1 External, as opposed to Type-2 External, the total metric would be 21 (the advertised metric plus the forward metric). Now R5 sends the Type-5 LSA from area 0 into area 3 to R8:

```
R8#show ip ospf database external 160.1.4.4

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Type-5 AS External Link States
```

https://t.me/learningnets

```
Routing Bit Set on this LSA in topology Base with MTID 0
LS age: 318
Options: (No TOS-capability, DC, Upward)
LS Type: AS External Link
Link State ID: 160.1.4.4 (External Network Number ) Advertising Router: 150.1.4.4
LS Seq Number: 80000001
Checksum: 0xD77F
Length: 36
Network Mask: /32
      Metric Type: 2 (Larger than any link state path)
      MTID: 0
      Metric: 20  Forward Address: 0.0.0.0

      External Route Tag: 0
```

R8, like R5, sees the Forward Address for the route as 0.0.0.0, meaning a lookup on 150.1.4.4 must be performed. The difference here, however, is that when R8 looks for a Type-1 Router LSA, none is found, as R4 is no in the same area as R8:

```
R8#show ip ospf database router 150.1.4.4

            OSPF Router with ID (150.1.8.8) (Process ID 1)R8#
```

This essentially means that the advertising router (R4) for the external route is not in the same area as R8. R8 now checks to see which ABRs are advertising reachability information about R4:

https://t.me/learningnets

```
R8#show ip ospf database asbr-summary 150.1.4.4


            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Summary ASB Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 387
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(AS Boundary Router)
  Link State ID: 150.1.4.4 (AS Boundary Router address) Advertising Router: 150.1.5.5

  LS Seq Number: 80000001
  Checksum: 0xEA0F
  Length: 28
  Network Mask: /0         MTID: 0 Metric: 1
```

R8 sees that 150.1.4.4 (R4) is known via the advertising router 150.1.5.5 (R5) with a metric of 1. An intra-area lookup is now performed on 150.1.5.5:

```
R8#show ip ospf database router 150.1.5.5

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Router Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1476
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000081
  Checksum: 0x936D
  Length: 48
  Area Border Router
  Number of Links: 2

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.5.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
```

https://t.me/learningnets

```
        TOS 0 Metrics: 1


    Link connected to: a Transit Network (Link ID) Designated Router address: 155.1.58.8
 (Link Data) Router Interface address: 155.1.58.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1
```

The recursion process continues until R8 sees that it is adjacent with the DR 155.1.58.8 as well, meaning that R5 is reachable out VLAN 58. Based on this, R8 knows that packets for 160.1.4.4/32 should go toward R5, which sends them toward R4, which in turn sends them to the final destination. The key point about this external lookup between areas, however, is that for all external destinations outside of area 3, R8 will see that R5 is the ABR that it must transit. This redundant information can be seen in the database view of area 3 as follows:

```
R8#show ip ospf database | b Type-5
                Type-5 AS External Link States


Link ID         ADV Router      Age         Seq#       Checksum Tag 160.1.4.4 150.1.4.4
       528         0x80000001 0x00D77F 0 160.1.6.6 150.1.6.6
       524         0x80000001 0x0093BB 0
```

R8 sees multiple Type-5 LSAs, reachable via R4 and R6. Recursion for both R4 and R6 points to R5, because R5 is the only ABR servicing area 3. In a design such as this, stub areas can be used to optimize the OSPF database by replacing the redundant Type-5 External and Type-4 ASBR routing information with default information. In this particular case, after configuring area 3 as stub, Type-5 External LSAs and Type-4 ASBR Summary LSAs no longer exist in the area 3 database:

```
R8#show ip ospf database external


            OSPF Router with ID (150.1.8.8) (Process ID 1) R8#
!
!R8#show ip ospf database asbr-summary


            OSPF Router with ID (150.1.8.8) (Process ID 1) R8#
```

This implies that R8 no longer has a specific route to 160.1.4.4/32, because of the deletion of the Type-4 and Type-5 LSAs:

https://t.me/learningnets

```
R8#show ip route 160.1.4.4
% Network not in table
```

What has been added to the database, however, is a default route as a Type-3 Summary LSA via the ABR:

```
R8#show ip ospf database summary 0.0.0.0

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Summary Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 104
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 0.0.0.0 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0x1D7F
  Length: 28 Network Mask: /0

        MTID: 0         Metric: 1
```

The result of this new default route is that area 3 maintains connectivity to the external routes by using the default route, but the size of the OSPF database and the routing table is much smaller:

```
R8#show ip cef 160.1.4.4
0.0.0.0/0
  nexthop 155.1.58.5 GigabitEthernet1.58
!
!R8#show ip cef 160.1.6.6
0.0.0.0/0
  nexthop 155.1.58.5 GigabitEthernet1.58
!
!R8#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 2, candidate default path, type inter area
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:02:12 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 150.1.5.5, 00:02:12 ago, via GigabitEthernet1.58
      Route metric is 2, traffic share count is 1
!
```

https://t.me/learningnets

```
!R8#traceroute 160.1.4.4

Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 6 msec 3 msec 3 msec 2 155.1.45.4 11 msec *  5 msec
!
!R8#traceroute 160.1.6.6
Type escape sequence to abort.
Tracing the route to 160.1.6.6
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 25 msec 2 msec 2 msec
  2 155.1.45.4 4 msec 2 msec 3 msec 3 155.1.146.6 3 msec *  5 msec
```

Verify that area 3 is configured as stub:

```
R5#show ip ospf | begin Area 3
    Area 3
        Number of interfaces in this area is 2 It is a stub area
 Generates stub default route with cost 1

        Area has no authentication
        SPF algorithm last executed 00:00:14.339 ago
        SPF algorithm executed 30 times
        Area ranges are
        Number of LSA 28. Checksum Sum 0x0B9268
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Totally Stubby Areas

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R4 and R6 with IP addressing in the format of 160.1.Y.Y/32, wherr **Y** is the router number.
  - Redistribute these prefixes into OSPF.
- Configure OSPF area 3 so that R5 filters inter-area and external routes out as they are sent from area 0 to area 3.
  - Devices in area 3 should still have reachability to routes external to the OSPF area 3.

Configuration

```
R5:
router ospf 1
 area 3 stub no-summary
R8 , R10:
router ospf 1
 area 3 stub
R4:
interface Loopback100
 ip address 160.1.4.4 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
```

https://t.me/learningnets

```
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
R6:

interface Loopback100
 ip address 160.1.6.6 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

In the previous task, we saw that with area 3 converted to a stub area, the size of the routing table and OSPF database was reduced with no negative impact on connectivity (review the previous task to for detailed information about stub areas). Specifically, R8's view of the topology was as follows (when area 3 is only configured as stub area):

```
R8#show ip ospf database


            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Router Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.5.5       150.1.5.5       180         0x80000084 0x00AB54 2
150.1.8.8       150.1.8.8       450         0x8000008A 0x00CB30 4
150.1.10.10     150.1.10.10     451         0x8000008C 0x0011A6 3

                Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.58.8      150.1.8.8       455         0x80000028 0x00BA8A
155.1.108.10    150.1.10.10     451         0x80000001 0x002AFF
 Summary Net Link States (Area 3)


Link ID         ADV Router      Age         Seq#       Checksum
0.0.0.0         150.1.5.5       464         0x80000001 0x001D7F
```

https://t.me/learningnets

```
150.1.1.1       150.1.5.5       464         0x80000033 0x0026BE
150.1.2.2       150.1.5.5       464         0x80000033 0x0011D1
150.1.3.3       150.1.5.5       464         0x80000033 0x00FBE4
150.1.4.4       150.1.5.5       464         0x80000033 0x00BC0D
150.1.5.5       150.1.5.5       464         0x80000033 0x009D2B
150.1.6.6       150.1.5.5       464         0x80000033 0x009C28
150.1.7.7       150.1.5.5       180         0x80000022 0x00D315
150.1.9.9       150.1.5.5       464         0x80000009 0x00E517
155.1.0.1       150.1.5.5       464         0x80000033 0x00E5FB
155.1.0.2       150.1.5.5       464         0x80000033 0x00DB05
155.1.0.3       150.1.5.5       464         0x80000033 0x00D10E
155.1.0.4       150.1.5.5       464         0x80000033 0x009D2C
155.1.0.5       150.1.5.5       464         0x80000033 0x008940
155.1.7.0       150.1.5.5       180         0x80000022 0x00D812
155.1.9.0       150.1.5.5       464         0x80000009 0x00FE02
155.1.13.0      150.1.5.5       464         0x80000033 0x006A6A
155.1.23.0      150.1.5.5       464         0x80000033 0x00FBCE
155.1.37.0      150.1.5.5       180         0x80000022 0x00834A
155.1.45.0      150.1.5.5       464         0x80000033 0x00D4CB
155.1.67.0      150.1.5.5       180         0x80000022 0x00426C
155.1.79.0      150.1.5.5       464         0x80000009 0x00EFCB
155.1.146.255   150.1.5.5       464         0x80000011 0x00C794
```

The output above indicates that the intra-area information from Type-1 Router LSAs and Type-2 Network LSAs still exists, along with the inter-area Type-3 Summary LSA information, but Type-4 ASBR Summary LSAs and Type-5 External LSAs have been removed (as seen in the previous task).

Recall that in the previous case for external routes we saw that every Type-4 ASBR Summary LSA inside of area 3 always recursed back to R5, because R5 was the only ABR connecting area 3 to area 0. By configuring area 3 as stub, this information was replaced with a default route that recursed to R5. The same connectivity resulted, but space was saved in the routing table and OSPF database. The next logical step in further optimizing the database is to summarize the redundant Type-3 Summary LSAs that represent the inter-area routes. This is where configuring the area as totally stubby can be advantageous.

Where a stub area optimizes the database by removing external routes and replacing it with a default route, a totally stubby area will optimize the database further by removing the inter-area and external routes, replacing them both with a default route. This is accomplished by telling the area border router not to inject Type-3 Summary Network LSAs from area 0, hence the **`no-summary`** argument used in conjunction with the **`stub`** command.

Note that only the ABR(s) connecting the stub area to area 0 need the **`no-summary`**

https://t.me/learningnets

argument on the stub command, because they are the only devices that are allowed to originate Type-3 LSAs. Although it won't break the OSPF design to add the command to other routers inside the totally stubby area, it is technically incorrect to configure the option this way. As long as all devices in the area agree on the stub flag in the first place, it is the ABR's duty to determine whether the area is totally stubby or not.

To verify the operation of the totally stubby area, view the changes to the database on the area 3 routers:

```
R8#show ip ospf database

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Router Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.5.5       150.1.5.5       314         0x80000084 0x00AB54 2
150.1.8.8       150.1.8.8       584         0x8000008A 0x00CB30 4
150.1.10.10     150.1.10.10     586         0x8000008C 0x0011A6 3

                Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.58.8      150.1.8.8       589         0x80000028 0x00BA8A
155.1.108.10    150.1.10.10     586         0x80000001 0x002AFF
 Summary Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
0.0.0.0         150.1.5.5       4           0x80000003 0x001981
```

Note the major change that has occurred: R8 no longer has specific inter-area routing information listed under the *Summary Net Link States* field, which represents the Type-3 Summary LSAs. Because R8 still has a default route via R5 (from the Type3 LSA generated by R5), connectivity with inter-area or external OSPF prfixes is not affected by this change. Verify connectivity with external OSPF prefixes:

```
R8#show ip route 160.1.4.4
% Network not in table
!
!R8#show ip cef 160.1.4.4
0.0.0.0/0
  nexthop 155.1.58.5 GigabitEthernet1.58
!
```

https://t.me/learningnets

```
!R8#show ip route 0.0.0.0

Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 2, candidate default path, type inter area
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:03:19 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 150.1.5.5, 00:03:19 ago, via GigabitEthernet1.58
      Route metric is 2, traffic share count is 1
!
!R8#traceroute 160.1.4.4
Type escape sequence to abort.
Tracing the route to 160.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 14 msec 2 msec 1 msec 2 155.1.45.4 3 msec *  2 msec
```

Verify connectivity with inter-area OSPF prefixes:

```
R8#show ip route 150.1.4.4
% Subnet not in table
!
!R8#show ip cef 150.1.4.4
0.0.0.0/0
  nexthop 155.1.58.5 GigabitEthernet1.58
!
!R8#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 2, candidate default path, type inter area
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:03:19 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 150.1.5.5, 00:03:19 ago, via GigabitEthernet1.58
      Route metric is 2, traffic share count is 1
!
!R8#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 16 msec 2 msec 1 msec 2 155.1.45.4 4 msec *  7 msec
```

Verify that area 3 is configured as totally stubby:

```
R8#show ip ospf | begin Area 3
    Area 3
        Number of interfaces in this area is 4 (1 loopback) It is a stub area
        Area has no authentication
        SPF algorithm last executed 00:15:03.196 ago
```

https://t.me/learningnets

```
        SPF algorithm executed 31 times
        Area ranges are
        Number of LSA 6. Checksum Sum 0x028734
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
!
!R5#show ip ospf | begin Area 3
    Area 3
        Number of interfaces in this area is 2 It is a stub area, no summary LSA in this area
 Generates stub default route with cost 1

        Area has no authentication
        SPF algorithm last executed 00:05:47.336 ago
        SPF algorithm executed 29 times
        Area ranges are
        Number of LSA 6. Checksum Sum 0x028734
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Not-So-Stubby Areas

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R6 and R8 with IP addressing in the format of 160.1.Y.Y/32, wherr **Y** is the router number.
  - Redistribute these prefixes into OSPF.
- Configure OSPF area 3 so that R5 filters external routes out as they are sent from area 0 to area 3.
  - Routers in area 3 should still be allowed to redistribute into OSPF.

Configuration

```
R5, R8 , R10:
 router ospf 1
  area 3 nssa
 R6:
 interface Loopback100
  ip address 160.1.6.6 255.255.255.255
 !
 route-map CONNECTED->OSPF permit 10
  match interface Loopback100
 !
 router ospf 1
  redistribute connected subnets route-map CONNECTED->OSPF
 R8:
```

https://t.me/learningnets

```
interface Loopback100
 ip address 160.1.8.8 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

The *OSPF Not-So-Stubby Area (NSSA) Option*, as defined in RFC 3101, extends the functionality of a stub area to allow the importing of a subset of external routes into the area. Recall that with the stub area, Type-5 External LSA information is suppressed from entering the database and is replaced with a default route originated by the ABR(s). Because all Type-5 LSAs are suppressed, this also implies that redistribution cannot occur within the area as well. This problem can be seen from the parser error generated when redistribution and stub areas are configured together:

```
R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#router ospf 1
R10(config-router)#area 3 stub
R10(config-router)#redistribute connected subnets
%OSPF-4-ASBR_WITHOUT_VALID_AREA: Router is currently an ASBR while having only one area which is a stub area
```

The OSPF NSSA option changes this behavior by allowing redistribution to occur *within* the stub area, while still blocking external routes from entering the area through the ABR(s). Specifically, this is implemented through the introduction of a new link-state advertisement type, the Type-7 NSSA External LSA.

Routes that are redistributed directly into the NSSA are generated as Type-7 NSSA External LSAs. Like Type-5 External LSAs, two subtypes of Type-7 NSSA External LSAs exist, type 1 (N1) and type 2 (N2). N1, similar to E1, considers the metric that the ASBR reports into the OSPF domain along with the metric needed to reach the ABSR. N2, similar to E2, separates the metric into the flat value that the ASBR reports into the OSPF domain, which is installed in the routing table, and the value needed to reach the ASBR, known as the forwarding metric.

From the output below, we can see that with the default redistribution values, R8 originates the Type-7 NSSA External LSAs as metric-type 2, with a metric value of

https://t.me/learningnets

20. The detailed output from R5's routing table indicates a metric of 20 reported in by R8, and a forward metric of 2, R5's metric to reach R8's Loopback:

```
R5#show ip route ospf | include N
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
O N2     160.1.8.8 [110/20] via 155.1.58.8, 00:08:26, GigabitEthernet1.58
!
!R5#show ip route 160.1.8.8
Routing entry for 160.1.8.8/32  Known via "ospf 1", distance 110,
metric 20, type NSSA extern 2, forward metric 2
  Last update from 155.1.58.8 on GigabitEthernet1.58, 00:08:39 ago
  Routing Descriptor Blocks:
  * 155.1.58.8, from 150.1.8.8, 00:08:39 ago, via GigabitEthernet1.58
      Route metric is 20, traffic share count is 1
!
!R5#show ip ospf database nssa-external

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-7 AS External Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 646
  Options: (No TOS-capability, Type 7/5 translation, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.8.8 (External Network Number )
Advertising Router: 150.1.8.8
  LS Seq Number: 80000001
  Checksum: 0x187D
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 150.1.8.8

        External Route Tag: 0
```

When the Type-7 NSSA External LSA is received by the ABR and is moved into area 0, the information contained in the Type-7 LSA is translated to a normal Type-5 External LSA. If multiple ABRs exist, only one of them performs the translation through an election process, which is discussed in depth in a later task. In this fashion, OSPF devices outside of the NSSA do not know that the NSSA exists, which is analogous to how a Confederation works in BGP.

Note that R5 receives the Type-7 NSSA External LSA with the forward address set

https://t.me/learningnets

to 150.1.8.8, which happens to be R8's router-ID. With the previous Type-5 external lookups, we saw the forward address set to 0.0.0.0, which meant to route toward the advertising router to reach the final destination. In this case, the forward address is non-zero, which causes the lookup to be performed toward 150.1.8.8. This is a subtle difference in the lookup process, and this particular case results in the same path selection even if the lookup had occurred on the advertising router (150.1.8.8) instead of the forward address (150.1.8.8). There can, however, be certain designs where there is a shorter path to the forward address than the advertising router's address, which is explored in a later task related to multiple exit points out of the NSSA. The result of the translation on R5 is that devices in area 0 see the routes as Type-5 External LSAs, not Type-7:

```
R1#show ip ospf database | begin Type-5
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag160.1.6.6        150.1.6.6
       909         0x80000001 0x0093BB 0
160.1.8.8       150.1.5.5       895         0x80000001 0x00D3D1 0
!
!R1#show ip ospf database external 160.1.8.8

            OSPF Router with ID (150.1.1.1) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 870
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.8.8 (External Network Number ) Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xD3D1
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 150.1.8.8

        External Route Tag: 0
```

R1 performs a lookup on the now Type-5 External LSA, and, like R5, sees the forward address set to 150.1.8.8. Again, note that the lookup process for this translated Type-7 LSA is performed differently than a normal inter-area Type-5 external LSA lookup, because R1 computes its metric toward 150.1.8.8, and not a

https://t.me/learningnets

Type-4 LSA describing the ASBR. Furthermore, note that R5 does not generate a Type-4 ASBR Summary LSA describing R8:

```
R5#show ip ospf database asbr-summary 150.1.8.8


            OSPF Router with ID (150.1.5.5) (Process ID 1)R5#
!
!R5#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32  Known via "ospf 1", distance 110, metric 2, type intra area
  Last update from 155.1.58.8 on GigabitEthernet1.58, 00:17:43 ago
  Routing Descriptor Blocks: * 155.1.58.8, from 150.1.8.8, 00:17:43 ago, via GigabitEthernet1.58


      Route metric is 2, traffic share count is 1
```

R1's metric to the forwarding address 150.1.8.8 is 1002 via R5. This is the value installed as the forward metric for the translated Type-7 LSA, with a metric of 20 from the Type-5 LSA itself:

```
R1#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32  Known via "ospf 1", distance 110, metric 1002, type inter area
  Last update from 155.1.0.5 on Tunnel0, 00:19:03 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:19:03 ago, via Tunnel0
      Route metric is 1002, traffic share count is 1
!
!R1#show ip route 160.1.8.8
Routing entry for 160.1.8.8/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 1002
  Last update from 155.1.0.5 on Tunnel0, 00:19:06 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:19:06 ago, via Tunnel0


      Route metric is 20, traffic share count is 1
```

Similar to the stub area, the NSSA flag must be agreed upon by all devices in the area, or adjacency cannot occur. This implies that the area is a normal area, a stub area, or an NSSA, but no combination of the three. Furthermore, like the stub area, Type-5 external LSAs are blocked from entering the NSSA area on the ABR(s), note that R6's Loopback0 is known in area 0, but not in area 3:

```
R5#show ip route 160.1.6.6
Routing entry for 160.1.6.6/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 2
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:20:55 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.6.6, 00:20:55 ago, via GigabitEthernet1.45
```

https://t.me/learningnets

```
      Route metric is 20, traffic share count is 1
!
!R8#show ip route 160.1.6.6
% Subnet not in table
```

Pitfall

The other key difference between stub and NSSA areas is how default routing works. The stub area removes external LSAs and replaces them with a default route. The totally stubby area extends this by replacing external LSAs and inter-area LSAs with a default route. However, *with the NSSA*, a default route is not automatically originated by the ABR. This means that devices within the NSSA will have reachability to their own area and to other areas, but not to destinations outside of the OSPF domain:

```
R8#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "ospf 1", distance 110, metric 3, type inter area
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:23:20 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 150.1.5.5, 00:23:20 ago, via GigabitEthernet1.58
      Route metric is 3, traffic share count is 1
!
!R8#traceroute 150.1.4.4
Type escape sequence to abort.
Tracing the route to 150.1.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 11 msec 2 msec 3 msec 2 155.1.45.4 17 msec *  6 msec
!
!R8#show ip cef 160.1.6.6
0.0.0.0/0 no route
!
!R8#traceroute 160.1.6.6 ttl 2 2
Type escape sequence to abort.
Tracing the route to 160.1.6.6
VRF info: (vrf in name/id, vrf out name/id) 2  *  *  *
```

Verify that area 3 is configured as NSSA:

```
R5#show ip ospf | begin Area 3
    Area 3
        Number of interfaces in this area is 2 It is a NSSA area
```

https://t.me/learningnets

```
Perform type-7/type-5 LSA translation

        Area has no authentication
        SPF algorithm last executed 00:39:47.094 ago
        SPF algorithm executed 36 times
        Area ranges are
        Number of LSA 28. Checksum Sum 0x0D7257
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Not-So-Stubby Areas and Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R6 and R8 with IP addressing in the format of 160.1.Y.Y/32, wherr **Y** is the router number.
  - Redistribute these prefixes into OSPF.
- Configure OSPF area 3 so that R5 filters external routes out as they are sent from area 0 to area 3.
  - Routers in area 3 should still be allowed to redistribute into OSPF.
  - Configure R5 to advertise a default route into area 3 with a cost of 500.

Configuration

```
R5:
router ospf 1
 area 3 nssa default-information-originate
 area 3 default-cost 500
R8 , R10:
router ospf 1
 area 3 nssa
R6:
interface Loopback100
 ip address 160.1.6.6 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
```

https://t.me/learningnets

```
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
R8:

interface Loopback100
 ip address 160.1.8.8 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

Unlike the stub area, totally-stubby area, and not-so-totally-stubby area, the ABR(s) of an NSSA do not automatically originate a default route. A default route can be originated as a Type-7 NSSA External LSA into the NSSA by adding the **`default-information-originate`** option onto the **`area [id] nssa`** statement. The cost that the ABR advertises for the default can be modified with the **`area [id] default-cost`** command. The default route is injected by the ABR as a regular Type-7 LSA with the type of N2 which uses the metric advertised by the ABR and the forwarding metric as the cost towards the ABR:

```
R8#show ip ospf database nssa-external 0.0.0.0

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Type-7 AS External Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 141
  Options: (No TOS-capability, No Type 7/5 translation, DC, Upward)
  LS Type: AS External Link
  Link State ID: 0.0.0.0 (External Network Number ) Advertising Router: 150.1.5.5
  LS Seq Number: 80000002
  Checksum: 0x713
  Length: 36
  Network Mask: /0 Metric Type: 2 (Larger than any link state path)
        MTID: 0  Metric: 500
 Forward Address: 0.0.0.0
```

https://t.me/learningnets

```
        External Route Tag: 0
!
!R8#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet  Known via "ospf 1", distance 110,
metric 500, candidate default path, type NSSA extern 2, forward metric 1
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:04:20 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 150.1.5.5, 00:04:20 ago, via GigabitEthernet1.58


      Route metric is 500, traffic share count is 1
```

Verify that now routers in area 3 have reachability with R6's redistributed Loopback100, although this is filtered from beign advertised into area 3:

```
R8#show ip cef 160.1.6.6
0.0.0.0/0
  nexthop 155.1.58.5 GigabitEthernet1.58
!
!R8#ping 160.1.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.6.6, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 3/6/14 ms
```

Verify that R5, the ABR is configured to inject a default route into the NSSA area:

```
R5#show ip ospf | begin Area 3
    Area 3
        Number of interfaces in this area is 2
        It is a NSSA area
        Perform type-7/type-5 LSA translation Generates NSSA default route with cost 500

        Area has no authentication
        SPF algorithm last executed 00:08:16.563 ago
        SPF algorithm executed 37 times
        Area ranges are
        Number of LSA 29. Checksum Sum 0x0D796A
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Not-So-Totally-Stubby Areas

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R6 and R8 with IP addressing in the format of 160.1.Y.Y/32, wherr **Y** is the router number.
  - Redistribute these prefixes into OSPF.
- Configure OSPF area 3 so that R5 filters inter-area and external routes out as they are sent from area 0 to area 3.
  - Routers in area 3 should still be allowed to redistribute into OSPF.

Configuration

```
R5:
router ospf 1
 area 3 nssa no-summary
R8 , R10:
router ospf 1
 area 3 nssa
R6:
interface Loopback100
 ip address 160.1.6.6 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
```

https://t.me/learningnets

```
 redistribute connected subnets route-map CONNECTED->OSPF
R8:

interface Loopback100
 ip address 160.1.8.8 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

The not-so-totally-stubby area is the combination of the totally-stubby area and the NSSA. Like the totally-stubby area, Type-3 Summary LSAs, Type-4 ASBR Summary LSAs, and Type-5 External LSAs are removed and replaced with a Type-3 Summary LSA default route. Like the NSSA, Type-7 NSSA External LSAs are allowed to be originated inside the area.

The combination of these two result in the blocking of all inter-area OSPF routes and routes external to the OSPF domain, replacing them with a default route, and allowing redistribution to occur. R8's routing table output indicates this because only intra-area, NSSA external, and an inter-area default route are installed. Note that the cost of the default route is derived from the intra-area cost to R5 plus the metric advertised by R5 for the Type3 LSA:

```
R8#show ip ospf database summary 0.0.0.0

            OSPF Router with ID (150.1.8.8) (Process ID 1)
 Summary Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 244
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 0.0.0.0 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xA4EF
  Length: 28
  Network Mask: /0        MTID: 0 Metric: 1
!
!R8#show ip route 0.0.0.0
```

https://t.me/learningnets

```
Routing entry for 0.0.0.0/0, supernet   Known via "ospf 1", distance 110,
metric 2, candidate default path, type inter area
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:05:52 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 150.1.5.5, 00:05:52 ago, via GigabitEthernet1.58

      Route metric is 2, traffic share count is 1
```

The database output indicates that the only Type-3 Summary LSA is the default route originated by the ABR, which is also visible in the routing table:

```
R8#show ip ospf database

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Router Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.5.5       150.1.5.5       1882        0x8000008A 0x002DC2 2
150.1.8.8       150.1.8.8       1659        0x80000092 0x000DF6 4

                Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.58.8      150.1.8.8       1659        0x8000002E 0x003601
 Summary Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
0.0.0.0         150.1.5.5       404         0x80000001 0x00A4EF

                Type-7 AS External Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Tag
160.1.8.8       150.1.8.8       1659        0x80000002 0x00167E 0
!
!R8#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override
```

https://t.me/learningnets

```
Gateway of last resort is 155.1.58.5 to network 0.0.0.0
O*IA  0.0.0.0/0 [110/2] via 155.1.58.5, 00:07:25, GigabitEthernet1.58


      155.1.0.0/16 is variably subnetted, 7 subnets, 2 masks
O        155.1.5.0/24 [110/2] via 155.1.58.5, 01:02:32, GigabitEthernet1.58
```

R8 does not have a longer match to 160.1.6.6, redistributed on R6, but it can use its default information to reach it:

```
R8#show ip route 160.1.6.6
% Subnet not in table
!
!R8#show ip cef 160.1.6.6
0.0.0.0/0
  nexthop 155.1.58.5 GigabitEthernet1.58
!
!R8#traceroute 160.1.6.6
Type escape sequence to abort.
Tracing the route to 160.1.6.6
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 7 msec 10 msec 2 msec
  2 155.1.45.4 7 msec 4 msec 4 msec 3 155.1.146.6 4 msec *  4 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Stub Areas with Multiple Exit Points

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Disable R6's connection to VLAN 67.
- Configure Loopback100 on R5 with IP address of 160.1.5.5/32 and redistribute it into OSPF.
- Configure area 1 as NSSA as follows:
  - all traffic from R6 destined to inter-area OSPF prefixes is routed through R1.
  - all traffic from R6 destined to external OSPF prefixes is routed through R4.
  - if either R1 or R4 looses its connection to area 1, traffic should be re-routed over the remaining ABR.

Configuration

```
R1:
router ospf 1
 area 1 nssa
 area 1 nssa default-information-originate
R4:
router ospf 1
 area 1 nssa no-summary
R6:
interface GigabitEthernet1.67
 shutdown
!
```

https://t.me/learningnets

```
router ospf 1
 area 1 nssa
R5:

interface Loopback100
 ip address 160.1.5.5 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

In addition to database filtering, stub areas can be used for inter-area traffic engineering. In this particular case, from R6's perspective, there are multiple exit points out of area 1, through R1 and R4. Based on task requirements, R4 advertises only a default route as a Type-3 Summary LSA, because of its not-so-totally-stubby configuration. R1 advertises all Type-3 Summary LSAs, plus a Type-7 NSSA External default route. For inter-area routing from devices in area 1, this means that the longest match learned from R1 will always be used, and for default routing the Type-3 default will be used from R4. The default preference through R4 for the default route occurs because OSPF always prefers routes in the sequence intra-area > inter-area > external > nssa-external. The result can be seen in R6's OSPF database as follows:

```
R6#show ip ospf database nssa-external 0.0.0.0

            OSPF Router with ID (150.1.6.6) (Process ID 1)

                Type-7 AS External Link States (Area 1)

  LS age: 454
  Options: (No TOS-capability, No Type 7/5 translation, DC, Upward)
  LS Type: AS External Link Link State ID: 0.0.0.0 (External Network Number )
 Advertising Router: 150.1.1.1
  LS Seq Number: 80000001
  Checksum: 0xAC6B
  Length: 36 Network Mask: /0
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 1
        Forward Address: 0.0.0.0
```

https://t.me/learningnets

```
        External Route Tag: 0
!
!R6#show ip ospf database summary 0.0.0.0

            OSPF Router with ID (150.1.6.6) (Process ID 1)

                Summary Net Link States (Area 1)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 460
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 0.0.0.0 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0xB1E4
  Length: 28 Network Mask: /0
        MTID: 0         Metric: 1
!
!R6#show ip ospf database | begin Summary
                Summary Net Link States (Area 1)
Link ID ADV Router
      Age         Seq#       Checksum 0.0.0.0 150.1.4.4
       482         0x80000001 0x00B1E4 150.1.1.1 150.1.1.1
       489         0x800000A5 0x00C895 150.1.2.2 150.1.1.1
       489         0x80000060 0x00A524 150.1.3.3 150.1.1.1
       489         0x8000005F 0x009236 150.1.4.4 150.1.1.1
       489         0x8000005F 0x00545D 150.1.5.5 150.1.1.1
       489         0x8000005F 0x00357B 150.1.7.7 150.1.1.1
       489         0x8000004B 0x007063 150.1.8.8 150.1.1.1
       141         0x8000002C 0x006676 150.1.9.9 150.1.1.1
       489         0x80000032 0x008265 155.1.0.1 150.1.1.1
       489         0x80000088 0x00C2B5 155.1.0.2 150.1.1.1
       489         0x8000005F 0x007256 155.1.0.3 150.1.1.1
       489         0x8000005F 0x00685F 155.1.0.4 150.1.1.1
       489         0x8000005F 0x00357C 155.1.0.5 150.1.1.1
       489         0x8000005F 0x002190 155.1.5.0 150.1.1.1
       489         0x8000005E 0x002889 155.1.7.0 150.1.1.1
       489         0x8000004B 0x007560 155.1.8.0 150.1.1.1
       141         0x8000002C 0x00756A 155.1.9.0 150.1.1.1
       489         0x80000032 0x009B50 155.1.13.0 150.1.1.1
       489         0x800000AA 0x000346 155.1.23.0 150.1.1.1
       489         0x8000005F 0x009220 155.1.37.0 150.1.1.1
       489         0x8000004B 0x002098 155.1.45.0 150.1.1.1
       489         0x8000005F 0x006C1C 155.1.58.0 150.1.1.1
       489         0x80000062 0x00D6A1 155.1.67.0 150.1.1.1
       489         0x8000004B 0x00DEBA
```

https://t.me/learningnets

```
155.1.79.0 150.1.1.1
       142         0x80000033 0x008A1B 155.1.108.0 150.1.1.1
       141         0x8000002C 0x002556
<output omitted>
```

The above data is reflected also in the routing table of R6, where the only inter-area route learned from R4 is the default route, which means all inter-area traffic is routed specifically towards R1 and traffic towards external prefixes is routed through R4:

```
R6#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 155.1.146.4 to network 0.0.0.0
 O*IA  0.0.0.0/0 [110/2] via 155.1.146.4, 00:01:45, GigabitEthernet1.146
      150.1.0.0/32 is subnetted, 9 subnets
O IA     150.1.1.1 [110/2] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     150.1.2.2 [110/2002] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     150.1.3.3 [110/2002] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     150.1.4.4 [110/1003] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     150.1.5.5 [110/1002] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     150.1.7.7 [110/2003] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     150.1.8.8 [110/1003] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     150.1.9.9 [110/2004] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
      155.1.0.0/16 is variably subnetted, 19 subnets, 2 masks
O IA     155.1.0.1/32 [110/1] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.0.2/32
           [110/2001] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.0.3/32
           [110/2001] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     155.1.0.4/32
           [110/1002] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.0.5/32
           [110/1001] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.5.0/24
           [110/1002] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.7.0/24
           [110/2003] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
```

https://t.me/learningnets

```
O IA     155.1.8.0/24
           [110/1003] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.9.0/24
           [110/2004] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     155.1.13.0/24 [110/2] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.23.0/24
           [110/2002] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     155.1.37.0/24
           [110/2002] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     155.1.45.0/24
           [110/1002] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.58.0/24
           [110/1002] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
O IA     155.1.67.0/24
           [110/2003] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     155.1.79.0/24
           [110/2003] via 155.1.146.1, 00:00:05, GigabitEthernet1.146
O IA     155.1.108.0/24
           [110/1003] via 155.1.146.1, 00:01:45, GigabitEthernet1.146
!
!R6#traceroute 150.1.5.5
Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.1 16 msec 3 msec 2 msec
  2 155.1.0.5 6 msec *  3 msec
!
!R6#traceroute 160.1.5.5
Type escape sequence to abort.
Tracing the route to 160.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.4 21 msec 13 msec 1 msec

  2 155.1.45.5 5 msec *  10 msec
```

If R4's link to OSPF area 1 is disabled, the deefault route through R1 will be installed in the database, and thus reachability to OSPF external prefixes will be maintained:

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet1.146
R4(config-subif)#shutdown
!
!R6#show ip route 160.1.5.5
% Subnet not in table
!
!R6#show ip route 0.0.0.0
```

https://t.me/learningnets

```
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 1, candidate default path,
type NSSA extern 2, forward metric 1
  Last update from 155.1.146.1 on GigabitEthernet1.146, 00:00:28 ago
  Routing Descriptor Blocks: * 155.1.146.1, from 150.1.1.1, 00:00:28 ago, via GigabitEthernet1.146
      Route metric is 1, traffic share count is 1
!
!R6#traceroute 160.1.5.5
Type escape sequence to abort.
Tracing the route to 160.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.1 13 msec 2 msec 1 msec


  2 155.1.0.5 3 msec *  2 msec
```

Likewise, if R4's connection to area 1 is functional but R4 looses its connection to area 1, all inter-area prefixes will be removed from the routing table and all inter-area and OSPF external traffic will be routed through the default route from R4:

https://t.me/learningnets

```
R6#show ip route ospf

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 155.1.146.4 to network 0.0.0.0
O*IA  0.0.0.0/0 [110/2] via 155.1.146.4, 00:00:15, GigabitEthernet1.146

!
!R6#traceroute 150.1.5.5

Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.4 33 msec 29 msec 4 msec

  2 155.1.45.5 5 msec *  2 msec
!
!R6#traceroute 160.1.5.5

Type escape sequence to abort.
Tracing the route to 160.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.146.4 86 msec 7 msec 25 msec

  2 155.1.45.5 11 msec *  110 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF NSSA Type-7 to Type-5 Translator Election

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Disable R6's connection to VLAN 67.
- Configure area 1 as NSSA.
- Configure Loopback100 on R6 with IP address of 160.1.6.6/32 and redistribute it into OSPF.
  - Ensure that only R1 advertises this route into area 0.

Configuration

```
R1:
router ospf 1
 router-id 150.1.11.11
!
clear ip ospf process
R1 , R4 , R6:
router ospf 1
 area 1 nssa
R6:

interface GigabitEthernet1.67
 shutdown
!
interface Loopback100
 ip address 160.1.6.6 255.255.255.255
```

https://t.me/learningnets

```
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

With OSPF Not-So-Stubby Areas, Type-7 NSSA External LSAs are translated to Type-5 External LSAs by the ABR connecting the NSSA to area 0. When multiple ABRs connect the NSSA to area 0, the ABR with the highest router-id is elected as the Type-7 to 5 translator, and is responsible for re-originating the Type-5 LSA into area 0. This election process is an optimization of the OSPF database and relates to how the Type-7 NSSA External route uses the forward address field to ensure optimal routing.

Recall that with normal external routes, only one Type-5 LSA is originated by the router performing the redistribution. When the route moves between areas, each ABR originates a Type-4 ASBR Summary LSA advertising its reachability to the ASBR. This means that for all Type-5 External LSA inter-area lookups, OSPF would require *Ext_Routes* + *Num_ABRs* + *Num_Routers* LSAs, where *Ext_Routes* is the number of Type-5 LSAs, *Num_ABRs* is the number of ABRs generating Type-4 ASBR summaries, and *Num_Routers* is the number of Type-1 LSAs from the routers in the local area.

Now with Type-7 LSAs, the situation becomes more complicated, because this information must be re-originated at the ABR level as the route moves into area 0. Let's suppose for the sake of argument that each ABR connecting the NSSA to area 0 *did* do a translation of Type-7 to 5. This would mean for all inter-area lookups on a Type-5 External LSAs that were translated from Type-7, there would be (*NSSA_Routes* * *Num_ABRs*) + *Num_ABRs* + *Num_Routers* LSAs, where *NSSA_Routes* is the number of Type-7 LSAs to start.

This operation would be highly redundant and inefficient, because each ABR would re-originate the same Type-5 LSA, each with the same forwarding address. To avoid this, only one ABR performs the Type-7 to 5 translation, but maintains the forward address field, essentially separating the relationship between the routing advertisement and the traffic flow. This principle can be illustrated as follows.

Before any router-ID modification in the OSPF domain, R5 performs a lookup on the Type-5 LSA for 160.1.6.6 that was translated from a Type-7 LSA. At this point, R1 has an OSPF Router-ID of 150.1.1.1 and R4 has 150.1.4.4. The advertising router

https://t.me/learningnets

that R5 sees is 150.1.4.4 (R4), because R4 won the translator election because of the higher RID. Note, however, that the forward address is set to 150.1.6.6 (R6). This means that R5 must figure out how to route toward 150.1.6.6:

```
R5#show ip ospf database external 160.1.6.6

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 27
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.6.6 (External Network Number )
 Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0xD4DA
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 150.1.6.6

        External Route Tag: 0
```

Because 150.1.6.6 does not belong to a device in the same area as R5, an inter-area lookup is performed on the Type-3 LSA. R5 finds that two ABRs are advertising the route to 150.1.6.6, 150.1.1.1 (R1) and 150.1.4.4 (R4), both with a metric of 2:

```
R5#show ip ospf database summary 150.1.6.6

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary Net Link States (Area 0)

  LS age: 557
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.6.6 (summary Network Number)
 Advertising Router: 150.1.1.1
  LS Seq Number: 8000000A
  Checksum: 0xFAF9
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 2

  Routing Bit Set on this LSA in topology Base with MTID 0
```

https://t.me/learningnets

```
  LS age: 703
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.6.6 (summary Network Number)
 Advertising Router: 150.1.4.4
  LS Seq Number: 8000000A
  Checksum: 0xD31B
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 2
```

R5 must now find the metric needed to reach these ABRs. R5 checks its locally originated Type-1 Router LSA and finds that 150.1.1.1 (R1) and 150.1.4.4 (R4) are directly attached, R1 with a metric of 1000 and R4 with a metric of 1:

```
R5#show ip ospf database router self-originate

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

  LS age: 1970
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000680
  Checksum: 0xCBB9
  Length: 108
  Area Border Router
  AS Boundary Router
  Number of Links: 7

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.45.4
 (Link Data) Router Interface address: 155.1.45.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.1.1
 (Link Data) Router Interface address: 155.1.0.5
```

https://t.me/learningnets

```
      Number of MTID metrics: 0        TOS 0 Metrics: 1000
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.4.4
 (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0        TOS 0 Metrics: 1000



    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.3.3
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.2.2
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
<output omitted>
```

This means that R5's intra-area cost to R4 is 1, and to R1 is 1000. Because both R1 and R4 reported a cost of 2 to the forward address 150.1.6.6, the total forward metric through R1 is 1000+2 = 1002, but is only 1+2 = 3 through R4. Therefore, the path through R4 installed with the default redistribution metric of 20 for the E2 route, and a forward metric of 3 through R4:

https://t.me/learningnets

```
R5#show ip route 160.1.6.6

Routing entry for 160.1.6.6/32   Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 3

  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:05:56 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.4.4, 00:05:56 ago, via GigabitEthernet1.45

      Route metric is 20, traffic share count is 1
!
!R5#traceroute 160.1.6.6

Type escape sequence to abort.
Tracing the route to 160.1.6.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 3 msec 1 msec 12 msec

  2 155.1.146.6 2 msec *  2 msec
```

This illustrates why a Type-5 External route that was translated from a Type-7 NSSA External route does not use a Type-4 ASBR Summary LSA, because the forward address lookup replaces the need for the ASBR Summary lookup. Because the forward address is preserved, only one router needs to do the translation, while the calculation of the final forwarding path stays independent. The Type-7 to 5 translator election can be modified by increasing R1's router-id to be higher than R4's:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#router ospf 1
R1(config-router)#router-id 150.1.11.11
!R1#clear ip ospf process
Reset ALL OSPF processes? [no]:yes
!
!R5#show ip ospf database external 160.1.6.6

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 75
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.6.6 (External Network Number ) Advertising Router: 150.1.11.11
```

https://t.me/learningnets

```
  LS Seq Number: 80000001
  Checksum: 0x7928
  Length: 36
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 150.1.6.6

        External Route Tag: 0
```

R5 now sees the advertising router as 150.1.11.11 (R1), because this is the highest router-id of the ABRs connecting the NSSA to area 0. Although the advertising router has changed, the forward address is still 150.1.6.6, which means that the traffic flow has not changed, although R1 is now the Type7-to-Type5 translator, the traffic path is still via R4 due to lowest cost towards the forward address via R4:

```
R5#show ip route 160.1.6.6
Routing entry for 160.1.6.6/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 3
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:01:11 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.11.11, 00:01:11 ago, via GigabitEthernet1.45
      Route metric is 20, traffic share count is 1
!
!R5#traceroute 160.1.6.6

Type escape sequence to abort.
Tracing the route to 160.1.6.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 21 msec 2 msec 1 msec

  2 155.1.146.6 39 msec *  7 msec
```

If R1 looses its links to area 0, immediately after OSPF neighbors are lost and OSPF routers from area 0 detect that R1 is no longer reachable via area 0, R4 will take over the role of Type7-to-Type5 translator and generate the LSA. Note that the old LSA generated by R1 will remain in the OSPF database until aged out, however will not be used in the SPF calculation as R1 is no longer reachable via area 0. Additionally, at the OSPF database level, OSPF will set the "Routing Bit" for the LSA originated by R4, in order to signal that this entry can be used for being installed in the routing table:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface gigabitEthernet1
R1(config-if)#shutdown
```

https://t.me/learningnets

```
!
!R5#show ip ospf database external 160.1.6.6

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States
 Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 162
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.6.6 (External Network Number ) Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0xD4DA
  Length: 36
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20
        Forward Address: 150.1.6.6
        External Route Tag: 0

  LS age: 521
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.6.6 (External Network Number ) Advertising Router: 150.1.11.11
  LS Seq Number: 80000001
  Checksum: 0x7928
  Length: 36
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20
        Forward Address: 150.1.6.6
        External Route Tag: 0
!
!R5#show ip ospf database router 150.1.11.11

            OSPF Router with ID (150.1.5.5) (Process ID 1)
 Router Link States (Area 0)
 Adv Router is not-reachable in topology Base with MTID 0

  LS age: 517
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.11.11
  Advertising Router: 150.1.11.11
```

https://t.me/learningnets

```
  LS Seq Number: 80000560
  Checksum: 0xC22C
  Length: 60
  Area Border Router
  AS Boundary Router
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.1.1
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.5.5
     (Link Data) Router Interface address: 155.1.0.1
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.1
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

Traffic destined towards R6's redistributed Loopback is still functional, via R4:

```
R5#show ip route 160.1.6.6
Routing entry for 160.1.6.6/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 3
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:03:26 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.4.4, 00:03:26 ago, via GigabitEthernet1.45
      Route metric is 20, traffic share count is 1
!
!R5#traceroute 160.1.6.6
Type escape sequence to abort.
Tracing the route to 160.1.6.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.45.4 5 msec 1 msec 1 msec

  2 155.1.146.6 2 msec *  2 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF NSSA Redistribution Filtering

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R5 and R8 with IP addressing in the format of 160.1.Y.Y/32, where Y is the router number.
  - Redistribute these prefixes into OSPF.
- Configure area 3 as an NSSA so R5 blocks all LSA types 3, 4, and 5 and replaces them with a default route.
- Modify area 3 so that R5's redistributed Loopback100 is injected into area 0 as LSA Type-5, but is not injected into area 3 as LSA Type-7.

Configuration

```
R5:
interface Loopback100
 ip address 160.1.5.5 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
 area 3 nssa no-redistribution no-summary
R8:
```

https://t.me/learningnets

```
interface Loopback100
 ip address 160.1.8.8 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
R8 , R10:

router ospf 1
 area 3 nssa
```

Verification

In certain NSSA designs, the ABR can be an ASBR at the same time. When routes are redistributed directly on the ABR, they are originated into area 0 as Type-5 External LSAs, and into the NSSA as Type-7 NSSA External LSAs. The origination as Type-7 into the NSSA may be unnecessary overhead if the ABR performing the redistribution is the only exit point out of the area. In this particular case, R5 is both an ABR and ASBR and is the only exit point for R8 and R10 to route packets into area 0.

By configuring the **`area 3 nssa no-summary`** option on R5, along with the **`area 3 nssa`** option on R8 and R10, the number of routes contained in the area 3 database are minimized, while still allowing redistribution on routers in area 3. Before R5 performs redistribution, the database in area 3 looks as follows:

```
R8#show ip ospf database


            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Router Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Link count
150.1.5.5       150.1.5.5       405         0x800000C1 0x00BEF9 2
150.1.8.8       150.1.8.8       5           0x800000CA 0x00D8D8 4
150.1.10.10     150.1.10.10     419         0x800000C5 0x002650 3

                Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.58.8      150.1.8.8       408         0x80000063 0x00CB36
```

https://t.me/learningnets

```
155.1.108.10    150.1.10.10     419         0x80000003 0x00AD72

                Summary Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
0.0.0.0         150.1.5.5       286         0x80000003 0x00A0F1

                Type-7 AS External Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum Tag
160.1.8.8       150.1.8.8       4           0x80000001 0x00187D 0
```

R8 knows about the three routers in the area via Type-1 Router LSAs, the two DRs in the area via Type-2 Network LSAs, an inter-area default route originated by R5 as a Type-3 Summary LSA and the external route generated by itself. This is essentially the minimal information needed in the database to perform intra-area SPF, use default routing to leave the area, and still allow redistribution. Next, R5 performs redistribution into OSPF:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router ospf 1
R5(config-router)#redistribute connected subnets route-map CONNECTED->OSPF
```

R5 originates the link 160.1.5.5/32 into area 3 as a Type-7 NSSA External LSA, as well as into area 0 as a Type-5 External LSA:

```
R5#show ip ospf database external 160.1.5.5

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States

  LS age: 40
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.5.5 (External Network Number )
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xB59D
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20
        Forward Address: 0.0.0.0
```

https://t.me/learningnets

```
        External Route Tag: 0
!
!R5#show ip ospf database nssa-external 160.1.5.5

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-7 AS External Link States (Area 3)

  LS age: 38
  Options: (No TOS-capability, No Type 7/5 translation, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.5.5 (External Network Number )
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0x99B7
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20
        Forward Address: 0.0.0.0
        External Route Tag: 0
!
!R1#show ip route 160.1.5.5
Routing entry for 160.1.5.5/32  Known via "ospf 1", distance 110,
 metric 20, type extern 2, forward metric 1000
  Last update from 155.1.0.5 on Tunnel0, 00:01:04 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:01:04 ago, via Tunnel0
      Route metric is 20, traffic share count is 1
!
!R8#show ip route 160.1.5.5
Routing entry for 160.1.5.5/32  Known via "ospf 1", distance 110,
 metric 20, type NSSA extern 2, forward metric 1
  Last update from 155.1.58.5 on GigabitEthernet1.58, 00:00:59 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 150.1.5.5, 00:00:59 ago, via GigabitEthernet1.58

      Route metric is 20, traffic share count is 1
```

https://t.me/learningnets

The problem with this design is that unnecessary information is now in the database of area 3. Because the area 3 routers already had a default route via R5, having specific reachability information about the network 160.1.5.5/32 is redundant. Therefore, this design is a good candidate for Type-7 LSA suppression on the ABR itself. By adding the **`no-redistribution`** keyword onto the **`area 3 nssa`** statement of R5, Type-7 LSAs are not generated for locally redistributed routes. This does not, however, prevent other devices inside the NSSA from performing redistribution, such as R8 or R10, just the ABR:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router ospf 1
R5(config-router)#area 3 nssa no-redistribution no-summary
```

Devices in area 3 no longer have a specific route to 160.1.5.5/32 as R5 no longer generates a Type7 LSA, but they can reach it using the default route. Also, a Type-7 NSSA External LSA still exists for 160.1.8.8/32:

```
R10#show ip ospf database nssa-external

            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Type-7 AS External Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 227
  Options: (No TOS-capability, Type 7/5 translation, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.8.8 (External Network Number )
 Advertising Router: 150.1.8.8
  LS Seq Number: 80000001
  Checksum: 0x187D
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20
        Forward Address: 150.1.8.8
        External Route Tag: 0
!
!R10#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
```

https://t.me/learningnets

```
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is 155.1.108.8 to network 0.0.0.0
O*IA  0.0.0.0/0 [110/3] via 155.1.108.8, 00:02:21, GigabitEthernet1.108
      150.1.0.0/32 is subnetted, 2 subnets
O        150.1.8.8 [110/2] via 155.1.108.8, 00:10:57, GigabitEthernet1.108
      155.1.0.0/16 is variably subnetted, 7 subnets, 2 masks
O        155.1.5.0/24 [110/3] via 155.1.108.8, 00:10:27, GigabitEthernet1.108
O        155.1.8.0/24 [110/2] via 155.1.108.8, 00:10:57, GigabitEthernet1.108
O        155.1.58.0/24 [110/2] via 155.1.108.8, 00:10:57, GigabitEthernet1.108
      160.1.0.0/32 is subnetted, 1 subnets
O N2     160.1.8.8 [110/20] via 155.1.108.8, 00:03:53, GigabitEthernet1.108
!
!R10#show ip cef 160.1.5.5
0.0.0.0/0
  nexthop 155.1.108.8 GigabitEthernet1.108
!
!R10#traceroute 160.1.5.5
Type escape sequence to abort.
Tracing the route to 160.1.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.108.8 4 msec 13 msec 2 msec 2 155.1.58.5 7 msec *  5 msec
```

Verify that devices in area 0 and beyond have a specific route for 160.1.5.5/32 as a Type-5 External LSA:

```
R1#show ip ospf database external 160.1.5.5

            OSPF Router with ID (150.1.1.1) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 649
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.5.5 (External Network Number )
Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xB59D
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
```

https://t.me/learningnets

```
Metric: 20
 Forward Address: 0.0.0.0
        External Route Tag: 0
!
!R1#show ip route 160.1.5.5
Routing entry for 160.1.5.5/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 1000
  Last update from 155.1.0.5 on Tunnel0, 00:11:00 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:11:00 ago, via Tunnel0
      Route metric is 20, traffic share count is 1
!
!R1#traceroute 160.1.5.5
Type escape sequence to abort.
Tracing the route to 160.1.5.5
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.5 3 msec *  1 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF LSA Type-3 Filtering

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure LSA Type-3 Filtering on R5 as follows:
  - Devices in area 0 do not have reachability information about subnet between R8 and R10 (VLAN 108 link) or R10's Loopback0 interface.
  - Devices in area 3 do not have reachability information about the Loopback0 interface of R1; this should not affect any new areas added to R5.

Configuration

R5:

```
ip prefix-list R1_LOOPBACK deny 150.1.1.1/32
ip prefix-list R1_LOOPBACK permit 0.0.0.0/0 le 32
!
ip prefix-list AREA_3_ROUTES deny 150.1.10.10/32
ip prefix-list AREA_3_ROUTES deny 155.1.108.0/24
ip prefix-list AREA_3_ROUTES permit 0.0.0.0/0 le 32
!
router ospf 1
 area 3 filter-list prefix R1_LOOPBACK in
 area 3 filter-list prefix AREA_3_ROUTES out
```

https://t.me/learningnets

Verification

LSA Type-3 Filtering, like stub areas, is used to remove LSAs from the database as advertisements move between areas. Unlike stub areas, however, Type-3 LSA Filtering can be used to permit or deny any arbitrary inter-area routes based on a prefix-list.

The filter-list syntax supports the arguments *in* and *out*, which are used to allow more control on ABRs that terminate multiple areas. In the case of R5, which terminates only area 0 and area 3, the syntax **`area 3 filter-list prefix AREA_3_ROUTES out`** has the same result as **`area 0 filter-list prefix AREA_3_ROUTES in`** . The syntax **`area 3 filter-list prefix R1_LOOPBACK in`** applies to prefixes leaving area 0 (and any other areas if configured on R5) going into area 3, whereas the syntax **`area 0 filter-list prefix R1_LOOPBACK out`** would apply to prefixes leaving area 0 going into area 3 and any other areas, if configured, on R5.

In other words, if **`area 0 filter-list prefix R1_LOOPBACK out`** were applied, R1's Loopback0 would not enter area 3 or any other area (if configured) on R5, but with **`area 3 filter-list prefix R1_LOOPBACK in`** applied, R1's does not enter area 3 only. This configuration can be verified by viewing the database and the routing table. Prior to filtering, R5 originated 150.1.1.1 into area 3 as a Type-3 Summary LSA and 150.1.10.10 into area 0 as a Type-3 LSA:

```
R8#show ip ospf database summary 150.1.1.1

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Summary Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.1.1 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000002
  Checksum: 0x6AA9
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 1001
!
!R4#show ip ospf database summary 150.1.10.10 adv-router 150.1.5.5

            OSPF Router with ID (150.1.4.4) (Process ID 1)

                Summary Net Link States (Area 0)
```

https://t.me/learningnets

```
  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 187
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.10.10 (summary Network Number)
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000002
  Checksum: 0x8C5F
  Length: 28 Network Mask: /32
        MTID: 0 Metric: 3
```

After the filter is applied, this LSAs are withdrawn from area 3 and area 0 by R5. As the LSA is flushed from the database, you may see it temporarely with *Delete flag is set for this LSA*

```
R8#show ip ospf database summary 150.1.1.1

            OSPF Router with ID (150.1.8.8) (Process ID 1) R8#
!
!R8#show ip ospf database | begin Summary
                Summary Net Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
150.1.2.2       150.1.5.5       100         0x80000002 0x0055BC
150.1.3.3       150.1.5.5       100         0x80000002 0x0040CF
150.1.4.4       150.1.5.5       100         0x80000002 0x0001F7
150.1.5.5       150.1.5.5       100         0x80000002 0x00E116
150.1.6.6       150.1.5.5       100         0x80000002 0x00E013
150.1.7.7       150.1.5.5       100         0x80000002 0x00F511
150.1.9.9       150.1.5.5       100         0x80000002 0x00D52C
155.1.0.1       150.1.5.5       100         0x80000002 0x002AE6
155.1.0.2       150.1.5.5       100         0x80000002 0x0020EF
155.1.0.3       150.1.5.5       100         0x80000002 0x0016F8
155.1.0.4       150.1.5.5       100         0x80000002 0x00E117
155.1.0.5       150.1.5.5       100         0x80000002 0x00CD2B
155.1.7.0       150.1.5.5       100         0x80000002 0x00FA0E
155.1.9.0       150.1.5.5       100         0x80000002 0x00EE17
155.1.13.0      150.1.5.5       100         0x80000002 0x00AE55
155.1.23.0      150.1.5.5       100         0x80000002 0x0040B9
155.1.37.0      150.1.5.5       100         0x80000002 0x00A546
155.1.45.0      150.1.5.5       100         0x80000002 0x0019B6
155.1.67.0      150.1.5.5       100         0x80000002 0x006468
155.1.79.0      150.1.5.5       100         0x80000002 0x00DFE0
155.1.146.0     150.1.5.5       100         0x80000002 0x00C7A1
```

https://t.me/learningnets

```
                Summary ASB Link States (Area 3)

Link ID         ADV Router      Age         Seq#       Checksum
150.1.6.6       150.1.5.5       100         0x80000002 0x00C82B
!
!R4#show ip ospf database summary 150.1.10.10 adv-router 150.1.5.5

            OSPF Router with ID (150.1.4.4) (Process ID 1)

                Summary Net Link States (Area 0)
 Delete flag is set for this LSA
 LS age: MAXAGE(3609)
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 150.1.10.10 (summary Network Number)
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000003
  Checksum: 0x6C81
  Length: 28
  Network Mask: /32        MTID: 0 Metric: 16777215
```

R5 knows how to reach R10's Loopback0, along with the link between R8 and R10, but devices in area 0 do not; likewise R5 knows how to reach R1's Loopback0, but devices in area 3 do not:

```
R5#show ip route | include 150.1.10|155.1.108|150.1.1
O        150.1.1.1 [110/1001] via 155.1.0.1, 01:00:01, Tunnel0
O        150.1.10.10 [110/3] via 155.1.58.8, 00:58:53, GigabitEthernet1.58
O        155.1.108.0/24 [110/2] via 155.1.58.8, 00:58:53, GigabitEthernet1.58
R4#show ip route | include 150.1.10|155.1.108
R4#
!
!R8#show ip route | include 155.1.1.1
R8#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Forwarding Address Suppression

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure area 3 as NSSA.
- Configure R5 to filter R8's Loopback0 from being advertised into area 0.
- Configure Loopback100 on R4 and R8 with IP addressing in the format of 160.1.Y.Y/32, where Y is the router number.
  - Redistribute these prefixes into OSPF.
- Modify R5's NSSA configuration so that devices outside of area 3 maintain connectivity to R8's redistributed Loopback100.

Configuration

```
R5 , R8 , R10:
router ospf 1
 area 3 nssa
R4:
interface Loopback100
 ip address 160.1.4.4 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

https://t.me/learningnets

```
R5:

ip prefix-list AREA_3_ROUTES deny 150.1.8.8/32
ip prefix-list AREA_3_ROUTES permit 0.0.0.0/0 le 32
!
router ospf 1
 area 3 filter-list prefix AREA_3_ROUTES out
 area 3 nssa translate type7 suppress-fa
R8:

interface Loopback100
 ip address 160.1.8.8 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

Recall that with OSPF database lookups on external routes, the *Forward Address* field determines who the next recursive lookup should be performed toward. With typical Type-5 External LSAs, such as R4's Loopback100 redistributes into OSPF, the forward address is normally set to 0.0.0.0. This means that the next lookup should be performed toward the *Advertising Router*. For example, when R4 redistributes its Loopback100 into OSPF, R5 performs a lookup on the Type-5 External LSA as follows:

```
R5#show ip ospf database external 160.1.4.4

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 488
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.4.4 (External Network Number ) Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0xD77F
  Length: 36
  Network Mask: /32
```

https://t.me/learningnets

```
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 0.0.0.0

        External Route Tag: 0
```

R5 sees the forward address field set to 0.0.0.0, which means a lookup should be performed on the advertising router 150.1.4.4. R5 sees that on VLAN 45 it is adjacent with the DR 155.1.45.5 (itself), with a cost of 1:

```
R5#show ip ospf database router self-originate

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)

  LS age: 592
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.5.5
  Advertising Router: 150.1.5.5
  LS Seq Number: 800006A9
  Checksum: 0x8FCB
  Length: 108
  Area Border Router
  AS Boundary Router
  Number of Links: 7

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.5.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Transit Network
 (Link ID) Designated Router address: 155.1.45.5
 (Link Data) Router Interface address: 155.1.45.5
      Number of MTID metrics: 0       TOS 0 Metrics: 1



    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.1.1
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000
```

https://t.me/learningnets

```
    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.4.4
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.3.3
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.2.2
     (Link Data) Router Interface address: 155.1.0.5
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.5
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
<output omitted>
```

R5 asks the DR who it is adjacent with, and finds that 150.1.4.4 is on the local segment:

```
R5#show ip ospf database network 155.1.45.5

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1458
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 155.1.45.5 (address of Designated Router)
  Advertising Router: 150.1.5.5
  LS Seq Number: 80000020
  Checksum: 0x2D3C
  Length: 32
  Network Mask: /24
        Attached Router: 150.1.5.5
```

https://t.me/learningnets

```
Attached Router: 150.1.4.4
```

This means that when R5 installs 160.1.4.4/32 in the routing table, the metric will be 20, from R4's default metric during redistribution, and a forward metric of 1 to reach R4:

```
R5#show ip route 160.1.4.4
Routing entry for 160.1.4.4/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 1
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:06:32 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.4.4, 00:06:32 ago, via GigabitEthernet1.45


      Route metric is 20, traffic share count is 1
```

Now let's compare this normal Type-5 External LSA lookup to a Type-5 External LSA that was translated from a Type-7 NSSA External LSA. In this design, R8 redistributes the route 160.1.8.8.32 into area 3 as a Type-7 NSSA External LSA, and R5 translates it into a Type-5 External LSA as it moves to area 0. R3 performs a lookup on the external route and sees the advertising router set to 150.1.5.5 (R5) and the forward address set to 150.1.8.8:

https://t.me/learningnets

```
R3#show ip ospf database external 160.1.8.8


            OSPF Router with ID (150.1.3.3) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 470
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.8.8 (External Network Number ) Advertising Router: 150.1.5.5

  LS Seq Number: 80000003
  Checksum: 0xCFD3
  Length: 36
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 150.1.8.8

        External Route Tag: 0
```

Because the forward address is non-zero, the next recursive lookup is performed toward 150.1.8.8, instead of the advertising router 150.1.5.5:

```
R3#show ip route 150.1.8.8
% Subnet not in table
```

The problem with this design, however, is that the prefix 150.1.8.8 was filtered out from being advertised into area 0, based on task request. The result is that recursion toward the external route fails, and it cannot be installed in the routing table.

```
R3#show ip route 160.1.8.8
% Subnet not in table
```

To resolve this problem, a very specific feature can be configured on R5, which is known as OSPF Forwarding Address Suppression in Translated Type-5 LSAs. This

https://t.me/learningnets

feature, configured by adding the `translate type7 suppress-fa` argument onto the `area 3 nssa` statement, instructs the ABR to not preserve the value in the forward address field as a Type-7 NSSA External LSA is translated into a Type-5 External LSA:

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router ospf 1
R5(config-router)#area 3 nssa translate type7 suppress-fa

!
!R3#show ip ospf database external 160.1.8.8

            OSPF Router with ID (150.1.3.3) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 15
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 160.1.8.8 (External Network Number ) Advertising Router: 150.1.5.5
  LS Seq Number: 80000004
  Checksum: 0x70D9
  Length: 36
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 0.0.0.0

        External Route Tag: 0
```

The result of this configuration is that R3 now sees the forward address for the Type-5 External LSA 160.1.8.8/32 set to 0.0.0.0, which means that a lookup must now be performed on the advertising router 150.1.5.5. In this particular case, R3 finds that 150.1.5.5 is directly connected with a metric of 1000:

```
R3#show ip ospf database router self-originate

            OSPF Router with ID (150.1.3.3) (Process ID 1)

                Router Link States (Area 0)

  LS age: 85
  Options: (No TOS-capability, DC)
  LS Type: Router Links
```

https://t.me/learningnets

```
  Link State ID: 150.1.3.3
  Advertising Router: 150.1.3.3
  LS Seq Number: 80000588
  Checksum: 0x1CC4
  Length: 60
  Area Border Router
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 150.1.3.3
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: another Router (point-to-point)
 (Link ID) Neighboring Router ID: 150.1.5.5
 (Link Data) Router Interface address: 155.1.0.3
      Number of MTID metrics: 0        TOS 0 Metrics: 1000



    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.3
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
<output omitted>
```

The final result is that the external route is installed with a metric of 20, which is derived from R8's default redistribution metric, plus a forward metric of 1000 to reach R5:

```
R3#show ip route 160.1.8.8
Routing entry for 160.1.8.8/32  Known via "ospf 1", distance 110,
metric 20, type extern 2, forward metric 1000
  Last update from 155.1.0.5 on Tunnel0, 00:02:33 ago
  Routing Descriptor Blocks: * 155.1.0.5, from 150.1.5.5, 00:02:33 ago, via Tunnel0


      Route metric is 20, traffic share count is 1
```

Although this feature fixes the problem introduced by the Type-3 LSA Filter, suboptimal routing may be introduced when there are multiple exit points out of the NSSA. As previously discussed, normally the Type-7 to 5 translator election and the forward address calculation are kept separate, which means the control plane advertisement of the route does not need to follow the traffic forwarding plane, but

https://t.me/learningnets

with forwarding address suppression enabled, the traffic will always flow through the Type-7 to 5 translator.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R6 with a static default route pointing to Null0.
  - Configure R6 to originate an external type-2 default route with a metric of 60 into OSPF, as long as the static default route is installed in the routing table.
  - Do not use a route-map to accomplish this.
- Configure R4 to originate an external type-1 default route with a metric of 40, regardless of whether it has a default route already installed in the routing table.

Configuration

```
R4:
router ospf 1
 default-information originate always metric 40 metric-type 1
R6:

ip route 0.0.0.0 0.0.0.0 Null0
!
router ospf 1
 default-information originate metric 60
```

https://t.me/learningnets

Verification

Default routing for non-stub areas in OSPF is accomplished through the origination of Type-5 External LSAs via the `default-information originate` command. Without any additional arguments, the OSPF process first checks to see if a default route is installed in the routing table. If a default route is already installed, such as via a static route or learned via BGP, the OSPF default route is originated. If the default route is not found, no origination occurs. This behavior is typically desirable in designs with multiple exit points out of the OSPF domain to upstream networks.

For example, imagine an OSPF network with exit points A and B out to the Internet. Both router A and B are running BGP with upstream peers, and learning a default route via BGP. As long as both devices maintain their upstream peerings, a default route can be advertised into OSPF. However, if A's link to the upstream neighbor is lost, and hence its default route via BGP is lost, its OSPF default route advertisement is withdrawn. The result of this design is that an individual exit point will only collect default traffic if they themselves have a default exit point to upstream networks. This behavior can be modified by adding the `always` argument to the `default-information originate` statement, which essentially skips the checking for a default route already being installed in the table.

The below view of the OSPF database on R1 indicates that both R4 and R6 are originating a default route. Without additional arguments on the command, the default route would have been advertised as a Type-2 External route with a metric of 20. The same route lookup logic is applied to these default routes as normal Type-5 External LSAs, where E1 is preferred over E2, and if multiple E2 routes exist with the same metric, the forward metrics are compared:

```
R1#show ip ospf database external 0.0.0.0

            OSPF Router with ID (150.1.1.1) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 105
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 0.0.0.0 (External Network Number )
 Advertising Router: 150.1.4.4
  LS Seq Number: 80000001
  Checksum: 0xB7B4
  Length: 36
  Network Mask: /0 Metric Type: 1 (Comparable directly to link state metric)
```

https://t.me/learningnets

```
        MTID: 0  Metric: 40
Forward Address: 0.0.0.0
        External Route Tag: 1

  LS age: 197
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 0.0.0.0 (External Network Number )
Advertising Router: 150.1.6.6
  LS Seq Number: 80000001
  Checksum: 0xE9E9
  Length: 36
  Network Mask: /0 Metric Type: 2 (Larger than any link state path)
        MTID: 0  Metric: 60
Forward Address: 0.0.0.0

        External Route Tag: 1
```

Verify that R1 prefers the E1 route, thus it installs it in the routing table:

```
R1#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet  Known via "ospf 1", distance 110, metric 41
, candidate default path  Tag 1, type extern 1
  Last update from 155.1.146.4 on GigabitEthernet1.146, 00:04:22 ago
  Routing Descriptor Blocks: * 155.1.146.4, from 150.1.4.4, 00:04:22 ago, via GigabitEthernet1.146

      Route metric is 41, traffic share count is 1
      Route tag 1
```

The conditional checking for the already installed default route on R6 can be verified as shown below. As long as R6 has the static default route installed in the routing table, a default route is originated into OSPF. When the route is removed, only R4 continues to originate default information:

```
R6#show ip route static
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 0.0.0.0 to network 0.0.0.0
```

https://t.me/learningnets

```
S*    0.0.0.0/0 is directly connected, Null0

!
!R6#show ip ospf database | begin External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         150.1.4.4       234         0x80000001 0x00B7B4 1
0.0.0.0         150.1.6.6       325         0x80000001 0x00E9E9 1
!
!R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R6(config)#no ip route 0.0.0.0 0.0.0.0 null0
!
!R6#show ip ospf database | begin External

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         150.1.4.4       335         0x80000001 0x00B7B4 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Conditional Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Confgure a Loopback66 on R6 with an IP address of 66.66.66.66/32.
- Configure R6 to originate a default route into OSPF, but only if the Loopbac66 prefix is in the routing table.

Configuration

R6:

```
interface Loopback66
 ip address 66.66.66.66 255.255.255.255
!
ip prefix-list LOOPBACK66 seq 5 permit 66.66.66.66/32
!
route-map TRACK_LOOPBACK66 permit 10
 match ip address prefix-list LOOPBACK66
!
router ospf 1
 default-information originate route-map TRACK_LOOPBACK66
```

Verification

https://t.me/learningnets

Conditional default-information origination in OSPF uses a route-map to check for the existence of a specific prefix in the IP routing table before the default route is originated. Recall from the previous section that the default condition is to check for a default route already installed in the IP routing table. In this example, the check for the existing default route is circumvented with **`route-map`** added to the **`default-information originate`** statement. When route-map option is used, the **`always`** keyword is no longer needed.

In this design, R6 is configured to check for the prefix 66.66.66.66/32 in the routing table. If the route exists, the default route is originated. As shown below, when this prefix is no longer in the routing table, the default advertisement is withdrawn:

```
R6#show ip ospf database external 0.0.0.0

            OSPF Router with ID (150.1.6.6) (Process ID 1)

                Type-5 AS External Link States

  LS age: 161
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 0.0.0.0 (External Network Number )
 Advertising Router: 150.1.6.6
  LS Seq Number: 80000003
  Checksum: 0x9577
  Length: 36 Network Mask: /0
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 1
        Forward Address: 0.0.0.0
        External Route Tag: 1
!
!R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface Loopback66
R6(config-if)#shutdown

!
!R6#show ip ospf database external 0.0.0.0

            OSPF Router with ID (150.1.6.6) (Process ID 1) R6#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Reliable Conditional Default Routing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure an IP SLA instance on R8 to check ICMP reachability to R10's VLAN 108 IP address every five seconds.
  - R8 should advertise a default route into OSPF, but only if the SLA monitoring reports its status as **OK**.

Configuration

R8:

```
ip sla 1
 icmp-echo 155.1.108.10
 frequency 5
!
ip sla schedule 1 life forever start-time now
track 1 ip sla 1 state
!
ip route 169.254.0.1 255.255.255.255 Null0 track 1
ip prefix-list PLACEHOLDER seq 5 permit 169.254.0.1/32
!
route-map TRACK_PLACEHOLDER permit 10
 match ip address prefix-list PLACEHOLDER
!
router ospf 1
```

https://t.me/learningnets

```
 default-information originate route-map TRACK_PLACEHOLDER
```

Verification

In the previous example, R6 was configured to track the status of Loopback66 interface. When the link was down and its IP prefix withdrawn from the routing table, the default route origination was withdrawn based on the conditional checking. In some designs, tracking an interface status directly is not a good indication of end-to-end reachability, because the interface could be UP/UP locally, but the circuit itself could be down.

Another example that is common in today's networks is with Metro Ethernet. Because the router's local Ethernet interface only tracks link status to its attached switch, end-to-end reachability cannot be inferred by checking this link status. In the output below, we see a case where R8 wants to originate a default route only when the link to R10 is viable. Because the prefix 155.1.108.0/24 is currently in the routing table, the default route is originated:

```
R8#show ip ospf database external 0.0.0.0

            OSPF Router with ID (150.1.8.8) (Process ID 1)

                Type-5 AS External Link States

  LS age: 65
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 0.0.0.0 (External Network Number )
 Advertising Router: 150.1.8.8
  LS Seq Number: 80000001
  Checksum: 0x7F8B
  Length: 36 Network Mask: /0
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 1
        Forward Address: 0.0.0.0
        External Route Tag: 1
!
!R10#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 1, candidate default path
  Tag 1, type extern 2, forward metric 1
  Last update from 155.1.108.8 on GigabitEthernet1.108, 00:01:13 ago
  Routing Descriptor Blocks: * 155.1.108.8, from 150.1.8.8, 00:01:13 ago, via GigabitEthernet1.108
```

https://t.me/learningnets

```
      Route metric is 1, traffic share count is 1
      Route tag 1
```

Now we simulate a case in which failure occurs in the network such that the other side is not reachable but the Link status on local device is UP/UP and subnet is in the routing table. In our case, we shutdown the interface GigabitEthernet1.108 on R10 and see that on R8, link status is still UP/UP and the 155.1.108.0/24 subnet is in the routing table. We configure an IP SLA instance which is instructed to ping 155.1.108.10 every five seconds. The SLA instance is called from a tracked object, which in turn is called from an arbitrary placeholder static route. As shown below, with the tracked object reporting the status code OK, the default route is originated, becuase the route-map condition is true:

```
R8#show track
Track 1
  IP SLA 1 state
  State is Up
    1 change, last change 00:03:28 Latest operation return code: OK
  Latest RTT (millisecs) 2
  Tracked by:
    Static IP Routing 0
!
!R8#show ip route 169.254.0.1
Routing entry for 169.254.0.1/32
  Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks: * directly connected, via Null0

      Route metric is 0, traffic share count is 1
```

Now we will shut down R10's interface GigabitEthernet1.108 and use IP SLA with enhanced object tracking to advertise/withdraw the default route into OSPF:

```
R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#interface GigabitEthernet1.108
R10(config-subif)#shutdown
```

R8 detects the failure of the SLA instance, and the tracked object transitions to down:

```
R8#show track
Track 1
  IP SLA 1 state
```

https://t.me/learningnets

```
  State is Down
    2 changes, last change 00:00:05 Latest operation return code: Timeout
  Tracked by:
    Static IP Routing 0
!%TRACK-6-STATE: 1 ip sla 1 state Up -> Down
```

The failure of the tracked object causes the static route to be withdrawn. Because the route-map condition for the default-information originate statement is looking for this prefix to be installed in the routing table, the default route is withdrawn when 169.254.0.1 is withdrawn:

```
R8#show ip route 169.254.0.1
% Network not in table
!
!R8#show ip ospf database external 0.0.0.0

            OSPF Router with ID (150.1.8.8) (Process ID 1)R8#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Filtering with Distribute-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure distribute-list filtering on R5, R8, and R10 so that these devices do not install routes to the Loopback0 networks of R1 and R2.

Configuration

```
R5 , R8 , R10:

router ospf 1
 distribute-list 1 in
!
access-list 1 deny 150.1.1.1 0.0.0.0
access-list 1 deny 150.1.2.2 0.0.0.0
access-list 1 permit any
```

Verification

Recall that to properly compute SPF, all routers within an OSPF area must agree on their view of the database. This implies that OSPF filtering in the database can be accomplished between areas, but not within an area. Inter-area filtering has been previously demonstrated with stub areas, and the Type-3 LSA Filter. Intra-area filtering can be accomplished in OSPF with an inbound distribute-list; however, this

https://t.me/learningnets

filtering only affects the local routing table, not the OSPF database.

Before applying distribute-list, all the routes of loopbacks are in the routing table of R5:

```
R5#show ip route ospf | i 150.1.
      150.1.0.0/32 is subnetted, 10 subnets
O        150.1.1.1 [110/1001] via 155.1.0.1, 00:50:45, Tunnel0
O IA     150.1.2.2 [110/1001] via 155.1.0.2, 00:50:45, Tunnel0

O        150.1.3.3 [110/1001] via 155.1.0.3, 00:50:45, Tunnel0
O        150.1.4.4 [110/2] via 155.1.45.4, 00:50:45, GigabitEthernet1.45
O IA     150.1.6.6 [110/3] via 155.1.45.4, 00:50:45, GigabitEthernet1.45
O IA     150.1.7.7 [110/1002] via 155.1.0.3, 00:50:45, Tunnel0
O        150.1.8.8 [110/2] via 155.1.58.8, 00:49:57, GigabitEthernet1.58
O IA     150.1.9.9 [110/1003] via 155.1.0.3, 00:50:45, Tunnel0
O        150.1.10.10 [110/3] via 155.1.58.8, 00:08:20, GigabitEthernet1.58
```

When the distribute-list has been applied on R5, the routes 150.1.2.2/32 and 150.1.1.1/32 no longer appear in the routing table:

```
R5#show ip route ospf | i 150.1.

      150.1.0.0/32 is subnetted, 8 subnets
O        150.1.3.3 [110/1001] via 155.1.0.3, 00:00:03, Tunnel0
O        150.1.4.4 [110/2] via 155.1.45.4, 00:00:03, GigabitEthernet1.45
O IA     150.1.6.6 [110/3] via 155.1.45.4, 00:00:03, GigabitEthernet1.45
O IA     150.1.7.7 [110/1002] via 155.1.0.3, 00:00:03, Tunnel0
O        150.1.8.8 [110/2] via 155.1.58.8, 00:00:03, GigabitEthernet1.58
O IA     150.1.9.9 [110/1003] via 155.1.0.3, 00:00:03, Tunnel0
O        150.1.10.10 [110/3] via 155.1.58.8, 00:00:03, GigabitEthernet1.58
```

However, information for these two prefixes still exists in the OSPF database of R5:

```
R5#show ip ospf database router 150.1.1.1

            OSPF Router with ID (150.1.5.5) (Process ID 1)
 Router Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1592
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.1.1
```

https://t.me/learningnets

```
  Advertising Router: 150.1.1.1
  LS Seq Number: 80000587
  Checksum: 0xC32E
  Length: 60
  Area Border Router
  Number of Links: 3
 Link connected to: a Stub Network
(Link ID) Network/subnet number: 150.1.1.1
(Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 150.1.5.5
     (Link Data) Router Interface address: 155.1.0.1
      Number of MTID metrics: 0
       TOS 0 Metrics: 1000

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.0.1
     (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 0
!
!R5#show ip ospf database summary 150.1.2.2

            OSPF Router with ID (150.1.5.5) (Process ID 1)
 Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1095
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network) Link State ID: 150.1.2.2 (summary Network Number)
 Advertising Router: 150.1.2.2
  LS Seq Number: 80000089
  Checksum: 0x3943
  Length: 28 Network Mask: /32

        MTID: 0         Metric: 1

  LS age: 1721
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 150.1.2.2 (summary Network Number)
  Advertising Router: 150.1.3.3
  LS Seq Number: 8000002C
```

https://t.me/learningnets

```
Checksum: 0xF0E5
Length: 28
Network Mask: /32
      MTID: 0         Metric: 2
```

Pitfall

This type of design can result in traffic black holes if not implemented carefully. If R5 were configured with the distribute-list, but R8 and R10 were not, traffic from R10 toward 150.1.1.1/32 would be sent to R8, from R8 to R5, and then black holed on R5. When implementing inbound distribute-list filtering, ensure that all routers still agree on the forwarding paths in the network.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Summarization and Discard Routes

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure distribute-list filtering on R3 and R5 so that these devices do not install routes to the Loopback0 networks of R7 and R9.
- Configure R7 to originate a default route into OSPF.
- Configure R3 to advertise the summary 150.1.0.0/20 into area 0.
  - Ensure that R3 and R5 can still reach the Loopback0 networks of R7 and R9.

Configuration

```
R3 , R5:
router ospf 1
 distribute-list 1 in
!
access-list 1 deny 150.1.7.7 0.0.0.0
access-list 1 deny 150.1.9.9 0.0.0.0
access-list 1 permit any
R3:
router ospf 1
 no discard-route internal
 area 2 range 150.1.0.0 255.255.240.0
R7:

router ospf 1
```

https://t.me/learningnets

```
 default-information originate always
```

Verification

When summarization is configured in OSPF, similar to EIGRP and BGP, a matching route to Null0 for the summary is installed locally in the routing table. This “discard” route is used to prevent the forwarding of traffic toward a shorter match, such as a default route, if no specific route toward the actual destination exists in the network.

The automatic origination of the discard route can be disabled with the no discard-route [internal | external], where *internal* refers to inter-area summarization performed with the area range command, and *external* refers to redistributed summarization performed with the summary-address command. The operation of the discard route can be illustrated as follows. R3 and R5 have the prefixes 150.1.7.7/32 and 150.1.9.9/32 filtered out of the routing table with a distribute-list. Additionally, R3 is originating the summary 150.1.0.0/20 into area 0, which encompasses addresses 150.1.0.0 through 150.1.15.255:

```
R5#show ip route | i 150.
      150.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
O IA     150.1.0.0/20 [110/1002] via 155.1.0.3, 00:03:04, Tunnel0

O        150.1.1.1/32 [110/1001] via 155.1.0.1, 00:08:43, Tunnel0
O        150.1.2.2/32 [110/1001] via 155.1.0.2, 00:08:43, Tunnel0
O        150.1.3.3/32 [110/1001] via 155.1.0.3, 00:08:43, Tunnel0
O        150.1.4.4/32 [110/2] via 155.1.45.4, 00:08:43, GigabitEthernet1.45
C        150.1.5.5/32 is directly connected, Loopback0
O IA     150.1.6.6/32 [110/3] via 155.1.45.4, 00:08:43, GigabitEthernet1.45
O        150.1.8.8/32 [110/2] via 155.1.58.8, 00:08:43, GigabitEthernet1.58
O        150.1.10.10/32 [110/3] via 155.1.58.8, 00:08:43, GigabitEthernet1.58
```

Reachability to the network 150.1.3.3 is obtained from R5, but reachability to 150.1.7.7 is not, as it's dropped by R3:

```
R5#ping 150.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.3.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
!
!R5#ping 150.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds:U.U.U
```

https://t.me/learningnets

```
Success rate is 0 percent (0/5)
```

This is because R5's longest match to 150.1.7.7 is the summary received from R3, while R3's longest match to 150.1.7.7 is the discard route via Null0, based on the configured summary. Although R3 does have a valid default path to 150.1.7.7 via R7, this cannot be used because /20 is a longer match than /0:

```
R3#show ip route 150.1.7.7
Routing entry for 150.1.0.0/20
  Known via "ospf 1", distance 110, metric 2, type intra area
  Routing Descriptor Blocks: * directly connected, via Null0
      Route metric is 2, traffic share count is 1
!
!R3#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 1, candidate default path
  Tag 1, type extern 2, forward metric 1
  Last update from 155.1.37.7 on GigabitEthernet1.37, 00:10:38 ago
  Routing Descriptor Blocks: * 155.1.37.7, from 150.1.7.7, 00:10:38 ago, via GigabitEthernet1.37


      Route metric is 1, traffic share count is 1
      Route tag 1
```

With the discard route removed on R3, the longest match to 150.1.7.7 is now 0.0.0.0/0:

https://t.me/learningnets

```
R3#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R3(config)#router ospf 1
R3(config-router)#no discard-route internal

!
!R3#show ip route 150.1.7.7
% Subnet not in table

!
!R3#show ip route 0.0.0.0

Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 1, candidate default path
  Tag 1, type extern 2, forward metric 1
  Last update from 155.1.37.7 on GigabitEthernet1.37, 00:00:29 ago
  Routing Descriptor Blocks: * 155.1.37.7, from 150.1.7.7, 00:00:29 ago, via GigabitEthernet1.37

      Route metric is 1, traffic share count is 1
      Route tag 1
```

The final result is that R5 uses the 150.1.0.0/20 prefix to route traffic for 150.1.7.7 toward R3, whereas R3 uses the 0.0.0.0/0 prefix to route the traffic toward R7:

```
R5#ping 150.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
!
!R5#traceroute 150.1.7.7
Type escape sequence to abort.
Tracing the route to 150.1.7.7
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.0.3 3 msec 2 msec 1 msec 2 155.1.37.7 6 msec *  2 msec
!
! R5#show ip route 150.1.7.7
Routing entry for 150.1.0.0/20
  Known via "ospf 1", distance 110, metric 1002, type inter area
  Last update from 155.1.0.3 on Tunnel0, 00:15:37 ago
  Routing Descriptor Blocks: * 155.1.0.3, from 150.1.3.3, 00:15:37 ago, via Tunnel0
      Route metric is 1002, traffic share count is 1
!
!R3#show ip cef 150.1.7.7
0.0.0.0/0
```

https://t.me/learningnets

```
nexthop 155.1.37.7 GigabitEthernet1.37
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Filtering with Administrative Distance

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure administrative distance filtering on R10 so that it does not install a route for VLAN 146 network.

Configuration

```
R10:

access-list 10 permit 155.1.146.0
!
router ospf 1
 distance 255 150.1.5.5 0.0.0.0 10
```

Verification

Like the other routing protocols, administrative distance can be changed on a per-prefix and per-neighbor basis in OSPF. One key difference, however, is that the address field in the `distance` command refers to the originator of the prefix into the area, not necessarily the neighbor from which you are learning the route. AD of 255 means the route cannot be installed in the routing table.

In the output below, we can see that R10 receives a Type-3 LSA for 155.1.146.0/24

https://t.me/learningnets

from R5, which is the ABR for area 3. As R5 is reachable through Type-1 LSAs, SPF can be calculated and route is installed in the routing table:

```
R10#show ip ospf database summary 155.1.146.0

            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Summary Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 300
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xF38B
  Length: 28
  Network Mask: /24        MTID: 0 Metric: 1001
!
!R10#show ip route 155.1.146.0
Routing entry for 155.1.146.0/24
  Known via "ospf 1", distance 110, metric 1003, type inter area
  Last update from 155.1.108.8 on GigabitEthernet1.108, 00:05:07 ago
  Routing Descriptor Blocks: * 155.1.108.8, from 150.1.5.5, 00:05:07 ago, via GigabitEthernet1.108

      Route metric is 1003, traffic share count is 1
```

After configuration is applied, R10 will no longer install the route in routing table, but it is not filtered from OSPF database:

```
R10#show ip ospf database summary 155.1.146.0

            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Summary Net Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 480
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xF38B
  Length: 28
  Network Mask: /24
```

https://t.me/learningnets

```
        MTID: 0 Metric: 1001


!
!R10#show ip route 155.1.146.0
                % Subnet not in table
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Filtering with Route-Maps

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Disable R5's VLAN 45 interface.
- Configure route-map filtering on R5 so that traffic destined to VLAN 146 is sent toward R1.

Configuration

https://t.me/learningnets

```
R5:

interface GigabitEthernet1.45
 shutdown
!
access-list 3 permit 155.1.146.0
access-list 4 permit 155.1.0.4
!
route-map DENY_VLAN146_FROM_R4 deny 10
 match ip address 3
 match ip next-hop 4
!
route-map DENY_VLAN146_FROM_R4 permit 20
!
router ospf 1
 distribute-list route-map DENY_VLAN146_FROM_R4 in
```

Verification

Referencing a route-map with a distribute-list in OSPF extends the filtering capability with additional match criteria. Specifically, the matching of interface (outgoing interface in the routing table), ip address, ip next-hop, ip route-source (router-id of the originating router), metric, route-type (intra-area, inter-area, etc.), and tagging are supported inside the route-map. The limitation of this feature is that only inbound filtering is supported, and the filter is still only local to the router's routing table; that is, the filter does not affect the OSPF database advertisements.

In the output below, we can see that initially, after disabling R5's VLAN 45 interface, R5 has two routes to the prefix 155.1.146.0/24, via R1 and R4, both reachable over the same Tunnel0 interface through the DMVPN cloud:

```
R5#show ip ospf database summary 155.1.146.0

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1955
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.1.1
```

https://t.me/learningnets

```
  LS Seq Number: 80000001
  Checksum: 0xF37F
  Length: 28
  Network Mask: /24        MTID: 0 Metric: 1

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1619
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.4.4
  LS Seq Number: 80000025
  Checksum: 0x84C4
  Length: 28
  Network Mask: /24        MTID: 0 Metric: 1
<output omitted>
!
!R5#show ip route 155.1.146.0
Routing entry for 155.1.146.0/24  Known via "ospf 1", distance 110, metric 1001, type inter area
  Last update from 155.1.0.1 on Tunnel0, 00:00:09 ago
  Routing Descriptor Blocks: * 155.1.0.4, from 150.1.4.4, 00:00:09 ago, via Tunnel0
      Route metric is 1001, traffic share count is 1
 155.1.0.1, from 150.1.1.1, 00:00:09 ago, via Tunnel0

      Route metric is 1001, traffic share count is 1
```

With a normal distribute-list filter referencing an access-list, there would be no way to distinguish between these two prefixes because they have they are one and the same, just two different paths in the network. After configuration is applied, the path via R4 is removed from the routing table, but OSPF database is not changed:

```
R5#show ip ospf database summary 155.1.146.0

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary Net Link States (Area 0)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1955
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.1.1
  LS Seq Number: 80000001
  Checksum: 0xF37F
  Length: 28
  Network Mask: /24
```

https://t.me/learningnets

```
        MTID: 0 Metric: 1



  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1619
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.4.4
  LS Seq Number: 80000025
  Checksum: 0x84C4
  Length: 28
  Network Mask: /24        MTID: 0 Metric: 1
<output omitted>
!
!R5#show ip route 155.1.146.0

Routing entry for 155.1.146.0/24
  Known via "ospf 1", distance 110, metric 1001, type inter area
  Last update from 155.1.0.1 on Tunnel0, 00:00:05 ago
  Routing Descriptor Blocks: * 155.1.0.1, from 150.1.1.1, 00:00:05 ago, via Tunnel0

      Route metric is 1001, traffic share count is 1
```

The result in this case is that traffic no longer follows the path via R4, and is sent only to R1:

```
R5#traceroute 155.1.146.6
Type escape sequence to abort.
Tracing the route to 155.1.146.6
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 3 msec 1 msec 1 msec

  2 155.1.146.6 3 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF NSSA ABR External Prefix Filtering

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure area 3 as NSSA.
- Configure Loopback100 on R10 with IP address 160.1.10.10/32 and redistribute it into OSPF.
- Configure summarization on R5 so that devices outside of area 3 do not have a route for R10's Loopback100.
  - This filter should not affect any other prefixes.

Configuration

```
R5 , R8 , R10:
router ospf 1
 area 3 nssa
R5:
router ospf 1
 summary-address 160.1.10.10 255.255.255.255 not-advertise
R10:

interface Loopback100
 ip address 160.1.10.10 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
```

https://t.me/learningnets

```
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
```

Verification

As previously discussed, only one ABR exit point out of an NSSA converts a Type-7 NSSA External LSA to a Type-5 External LSA, based on the translator election process. The NSSA translator can be configured to suppress the origination of the Type-5 External LSA into area 0 via the **`summary-address`** command. As shown below, R4 currently learns the prefix 160.1.10.10/32 as a Type-5 External route, originated by the NSSA translator R5.

```
R5#show ip ospf database external 160.1.10.10

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-5 AS External Link States

  LS age: 55
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.10.10 (External Network Number )
 Advertising Router: 150.1.5.5
  LS Seq Number: 80000001
  Checksum: 0xDFBD
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20
        Forward Address: 150.1.10.10
        External Route Tag: 0
!
!R4#show ip route 160.1.10.10
Routing entry for 160.1.10.10/32
  Known via "ospf 1", distance 110, metric 20, type extern 2, forward metric 4
  Last update from 155.1.45.5 on GigabitEthernet1.45, 00:00:00 ago
  Routing Descriptor Blocks: * 155.1.45.5, from 150.1.5.5, 00:00:00 ago, via GigabitEthernet1.45
      Route metric is 20, traffic share count is 1
!
!R4#traceroute 160.1.10.10
Type escape sequence to abort.
Tracing the route to 160.1.10.10
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.45.5 7 msec 2 msec 8 msec
  2 155.1.58.8 2 msec 4 msec 1 msec
```

https://t.me/learningnets

```
3 155.1.108.10 4 msec *  3 msec
```

To suppress the advertisement of this prefix into area 0, R5 configures a **summary-address** with an identical mask of the original NSSA external route, but adds the **not-advertise** argument to the summary. The main difference between this filtering technique and the previously seen distribute-lists and administrative distance filters is that the prefix is filtered out of the *database*, not just the routing table. This can be verified by R5's lack of Type-5 LSA information, as shown below:

```
R5#show ip ospf database external 160.1.10.10

            OSPF Router with ID (150.1.5.5) (Process ID 1)R5#

!
!R4#show ip route 160.1.10.10
% Network not in table
```

The Type-7 LSA originated by R10, is not filtered from the database, and thus R5 and all area 3 routers have reachability with R10's Loopback100:

```
R5#show ip ospf database nssa-external 160.1.10.10

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Type-7 AS External Link States (Area 3)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 270
  Options: (No TOS-capability, Type 7/5 translation, DC, Upward)
  LS Type: AS External Link Link State ID: 160.1.10.10 (External Network Number )
 Advertising Router: 150.1.10.10
  LS Seq Number: 80000001
  Checksum: 0xA7F
  Length: 36 Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20
        Forward Address: 150.1.10.10
        External Route Tag: 0
!
!R5#show ip route 160.1.10.10
```

https://t.me/learningnets

```
Routing entry for 160.1.10.10/32
  Known via "ospf 1", distance 110, metric 20, type NSSA extern 2, forward metric 3
  Last update from 155.1.58.8 on GigabitEthernet1.58, 00:05:10 ago
  Routing Descriptor Blocks: * 155.1.58.8, from 150.1.10.10, 00:05:10 ago, via GigabitEthernet1.58
      Route metric is 20, traffic share count is 1
!
!
R5#traceroute 160.1.10.10
Type escape sequence to abort.
Tracing the route to 160.1.10.10
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.8 3 msec 7 msec 3 msec 2 155.1.108.10 7 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Database Filtering

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R7 so that R9 cannot learn any OSPF routes from R7, but R7 can still learn OSPF routes from R9.
- Configure R5 so that R2 cannot learn from R5, but R5 can still learn OSPF routes from R2.

Configuration

```
R5:
router ospf 1
 neighbor 155.1.0.2 database-filter all out
R7:


interface GigabitEthernet1.79
 ip ospf database-filter all out
```

Verification

The OSPF command `database-filter all out` is similar in operation to the `passive-interface` command in RIPv2. This feature allows the formation of OSPF neighbors, because hello packets are not filtered out, but it stops the advertisements of all LSAs out the interface or to the neighbor in question. As shown below, R7 learns

https://t.me/learningnets

prefixes from R9, but R9 has no LSAs in the database, with the exception of locally originated ones:

```
R7#show ip ospf database router adv-router 150.1.9.9

            OSPF Router with ID (150.1.7.7) (Process ID 1)

                Router Link States (Area 2)

  LS age: 6
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 150.1.9.9
  Advertising Router: 150.1.9.9
  LS Seq Number: 80000003
  Checksum: 0x88F9
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network (Link ID) Network/subnet number: 150.1.9.9
 (Link Data) Network Mask: 255.255.255.255
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Transit Network (Link ID) Designated Router address: 155.1.79.9
 (Link Data) Router Interface address: 155.1.79.9
      Number of MTID metrics: 0
       TOS 0 Metrics: 1

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 155.1.9.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
!
!R7#show ip route ospf | i GigabitEthernet1.79
O        150.1.9.9 [110/2] via 155.1.79.9, 00:00:24, GigabitEthernet1.79
O        155.1.9.0/24 [110/2] via 155.1.79.9, 00:00:24, GigabitEthernet1.79
!
!R9#show ip ospf database

            OSPF Router with ID (150.1.9.9) (Process ID 1)

                Router Link States (Area 2)
 Link ID ADV Router
      Age         Seq#       Checksum Link count
```

https://t.me/learningnets

```
150.1.9.9 150.1.9.9
       43          0x80000003 0x0088F9 3

                Net Link States (Area 2)
Link ID ADV Router
      Age         Seq#       Checksum 155.1.79.9 150.1.9.9
       43          0x80000001 0x003517
!
!R9#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set
R9#
```

Likewise, R5 learns LSAs from R2, but R2 cannot learn any LSAs from R5:

```
R2#show ip ospf database adv-router 150.1.5.5

            OSPF Router with ID (150.1.2.2) (Process ID 1) R2#
!
!R2#show ip route ospf | i Tunnel
R2#
!
!R5#show ip ospf database adv-router 150.1.2.2

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Router Link States (Area 0)
Link ID ADV Router
      Age         Seq#       Checksum Link count 150.1.2.2 150.1.2.2
       106         0x80000008 0x00FE6F 3

                Summary Net Link States (Area 0)
Link ID ADV Router
      Age         Seq#       Checksum 155.1.23.0 150.1.2.2
       111         0x80000001 0x0035B7
```

https://t.me/learningnets

```
!
!R5#show ip route ospf | i Tunnel
O        150.1.1.1 [110/1001] via 155.1.0.1, 00:19:06, Tunnel0
O        150.1.2.2 [110/1001] via 155.1.0.2, 00:01:32, Tunnel0
O        150.1.3.3 [110/1001] via 155.1.0.3, 00:19:06, Tunnel0
O IA     150.1.7.7 [110/1002] via 155.1.0.3, 00:19:06, Tunnel0
O IA     150.1.9.9 [110/1003] via 155.1.0.3, 00:17:45, Tunnel0
O        155.1.0.1/32 [110/1000] via 155.1.0.1, 00:19:06, Tunnel0
O        155.1.0.2/32 [110/1000] via 155.1.0.2, 00:01:32, Tunnel0

O        155.1.0.3/32 [110/1000] via 155.1.0.3, 00:19:06, Tunnel0
O IA     155.1.7.0/24 [110/1002] via 155.1.0.3, 00:19:06, Tunnel0
O IA     155.1.9.0/24 [110/1003] via 155.1.0.3, 00:17:45, Tunnel0
O IA     155.1.13.0/24 [110/1001] via 155.1.0.3, 00:19:06, Tunnel0
                       [110/1001] via 155.1.0.1, 00:19:06, Tunnel0
O IA     155.1.23.0/24 [110/1001] via 155.1.0.3, 00:19:06, Tunnel0
                       [110/1001] via 155.1.0.2, 00:01:32, Tunnel0
O IA     155.1.37.0/24 [110/1001] via 155.1.0.3, 00:19:06, Tunnel0
O IA     155.1.67.0/24 [110/1002] via 155.1.0.3, 00:19:06, Tunnel0
O IA     155.1.79.0/24 [110/1002] via 155.1.0.3, 00:19:06, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Stub Router Advertisement

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R4 to advertise the maximum metric value to all neighbors inside its Type-1 Router LSA.

Configuration

```
R4:

router ospf 1
 max-metric router-lsa
```

Verification

The OSPF Stub Router Advertisement feature, not to be confused with OSPF stub areas, is used to prevent traffic black holes caused by device adds or removes to or from the OSPF topology. Essentially, this feature causes the router to advertise a maximum metric for non-stub destinations, making it the worst cost path to all destinations. The result is that upon initializing the OSPF process, transit traffic will not flow through the stub router unless it is the only possible path. When the routing domain is fully converged, the max metric value can be withdrawn, allowing normal forwarding to occur through the device.

https://t.me/learningnets

The `max-metric router-lsa` syntax unconditionally advertises the maximum metric until the command is removed, whereas the `max-metric router-lsa on-startup wait-for-bgp` option causes the router to advertise the maximum metric until BGP keepalives are received from all neighbors (keepalives in BGP indicate that convergence is complete), and the `max-metric router-lsa on-startup announce-time` controls how long the router should advertise the maximum metric after a reload.

In the output below, we can see that R5 learns the prefix 155.1.146.0/24 from R4 via its VLAN 45 interface, and installs it in its routing table with a metric of 2:

```
R5#show ip ospf database summary 155.1.146.0

            OSPF Router with ID (150.1.5.5) (Process ID 1)

                Summary Net Link States (Area 0)

  LS age: 1179
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number)
  Advertising Router: 150.1.1.1
  LS Seq Number: 8000000A
  Checksum: 0xE188
  Length: 28
  Network Mask: /24
        MTID: 0         Metric: 1

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 997
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.4.4
  LS Seq Number: 8000000A
  Checksum: 0xBAA9
  Length: 28 Network Mask: /24
        MTID: 0 Metric: 1
!
!R5#show ip route 155.1.146.0
Routing entry for 155.1.146.0/24  Known via "ospf 1", distance 110, metric 2, type inter area
  Last update from 155.1.45.4 on GigabitEthernet1.45, 05:16:48 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 150.1.4.4, 05:16:48 ago, via GigabitEthernet1.45

      Route metric is 2, traffic share count is 1
```

https://t.me/learningnets

R4 currently advertises a metric of 1 to its transit link of VLAN 146. After the max-metric router-lsa command is configured, R4 advertises the maximum cost of 65535 for VLAN 146, essentially making it the worst-cost path. The final result is that R5 no longer uses R4 to reach this destination:

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#router ospf 1
R4(config-router)#max-metric router-lsa


!
!R5#show ip ospf database summary 155.1.146.0


            OSPF Router with ID (150.1.5.5) (Process ID 1)


                Summary Net Link States (Area 0)


  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1303
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number)
  Advertising Router: 150.1.1.1
  LS Seq Number: 8000000A
  Checksum: 0xE188
  Length: 28
  Network Mask: /24
        MTID: 0         Metric: 1


  LS age: 1
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 155.1.146.0 (summary Network Number) Advertising Router: 150.1.4.4
  LS Seq Number: 8000000B
  Checksum: 0xAEB5
  Length: 28 Network Mask: /24
        MTID: 0 Metric: 65535
!
!R5#show ip route 155.1.146.0
Routing entry for 155.1.146.0/24  Known via "ospf 1", distance 110, metric 1001, type inter area
  Last update from 155.1.0.1 on Tunnel0, 00:00:28 ago
  Routing Descriptor Blocks: * 155.1.0.1, from 150.1.1.1, 00:00:28 ago, via Tunnel0
      Route metric is 1001, traffic share count is 1
!
!R5#traceroute 155.1.146.6
Type escape sequence to abort.
Tracing the route to 155.1.146.6
```

https://t.me/learningnets

```
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.1 8 msec 1 msec 1 msec

  2 155.1.146.6 5 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Interface Timers

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R5 and R8 to send OSPF hello packets every five seconds on VLAN 58, and wait for seven seconds before declaring a neighbor down.
- Configure R4 and R5 to send OSPF hello packets every 250ms on VLAN 45.

Configuration

https://t.me/learningnets

```
R4:

interface GigabitEthernet1.45
 ip ospf dead-interval minimal hello-multiplier 4
R5:

interface GigabitEthernet1.58
 ip ospf hello-interval 5
 ip ospf dead-interval 7
!
interface GigabitEthernet1.45
 ip ospf dead-interval minimal hello-multiplier 4
R8:

interface GigabitEthernet1.58
 ip ospf hello-interval 5
 ip ospf dead-interval 7
```

Verification

OSPF hello and dead timers must match for adjacency to occur. When the **`ip ospf hello-interval`** command is modified without the **`ip ospf dead-interval`** command, the dead timer is automatically set to be four times the configured hello. Default values for these timers are 10 seconds for hello interval and 40 seconds for dead interval, or 30 seconds for hello interval and 120 seconds for dead interval. The OSPF network-type dictates what values are used by default, lower values being active for *broadcast* and *point-to-point* and higher values for all other.

With the OSPF sub-second hello feature, the dead timer is set to 1 second, and the hello interval is set to 1000 ms/hello-multiplier. Per the above command **`ip ospf dead-interval minimal hello-multiplier 4`** , the dead interval is 1 second, and the hello interval is 250 ms (four times per second). When using minimal hello, the multiplier value does not need to match between neighbors, however there is no reason to configure it in asymmetric fashion. Verify the hello and dead intervals:

```
R5#show ip ospf interface gigabitEthernet 1.58
GigabitEthernet1.58 is up, line protocol is up
  Internet Address 155.1.58.5/24, Area 3, Attached via Network Statement
  Process ID 1, Router ID 150.1.5.5, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 150.1.8.8, Interface address 155.1.58.8
```

https://t.me/learningnets

```
  Backup Designated router (ID) 150.1.5.5, Interface address 155.1.58.5   Timer intervals configured,
Hello 5, Dead 7
, Wait 7, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 18, maximum is 20
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.8.8  (Designated Router)
  Suppress hello for 0 neighbor(s)
!
!R5#show ip ospf interface gigabitEthernet1.45
GigabitEthernet1.45 is up, line protocol is up
  Internet Address 155.1.45.5/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 150.1.5.5, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 150.1.5.5, Interface address 155.1.45.5
  Backup Designated router (ID) 150.1.4.4, Interface address 155.1.45.4   Timer intervals configured,
Hello 250 msec, Dead 1
, Wait 1, Retransmit 5
    oob-resync timeout 40
    Hello due in 21 msec
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 6, maximum is 9
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.4.4  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
```

Verify that R5 is still OSPF neighbor with R4 and R8 over the links with modifid

https://t.me/learningnets

values:

```
R5#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.4.4         1   FULL/BDR        840 msec    155.1.45.4      GigabitEthernet1.45
150.1.2.2         0   FULL/  -        00:01:34    155.1.0.2       Tunnel0
150.1.4.4         0   FULL/  -        00:01:34    155.1.0.4       Tunnel0
150.1.3.3         0   FULL/  -        00:01:34    155.1.0.3       Tunnel0
150.1.1.1         0   FULL/  -        00:01:34    155.1.0.1       Tunnel0
150.1.8.8         1   FULL/DR         00:00:03    155.1.58.8      GigabitEthernet1.58
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Global Timers

> You must load the initial configuration files for the section, **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Modify R4 and R5's OSPF timers as follows:
  - SPF throttling to start new re-calculation at least 100 ms after a new LSA arrives.
  - The second SPF calculation should occur no less than 1 second after the first one, and the maximum wait time should be no more than 10 seconds.
  - LSA pacing to wait at least 50 ms between consecutive link-state updates; LSA retransmissions should be paced at least 75 ms apart.
  - LSA throttling to generate subsequent LSAs after 10 ms, to wait at least 4 seconds to generate the next LSA, and no more than 6 seconds between generation of the same LSA.
  - LSA arrival throttling to wait 2 seconds between reception of the same LSA from a neighbor.
- Configure R4 and R5 to assume that LSA transmission takes 2 seconds on the point-to-point link between them.
  - LSA retransmission should occur if an acknowledgement is not received within 10 seconds over this link.

Configuration

```
R4:
router ospf 1
 timers throttle spf 100 1000 10000
```

https://t.me/learningnets

```
 timers pacing flood 50
 timers pacing retransmission 75
 timers throttle lsa 10 4000 6000
 timers lsa arrival 2000
!
interface GigabitEthernet1.45
 ip ospf transmit-delay 2
 ip ospf retransmit-interval 10
R5:

router ospf 1
 timers throttle spf 100 1000 10000
 timers pacing flood 50
 timers pacing retransmission 75
 timers throttle lsa 10 4000 6000
 timers lsa arrival 2000
!
interface GigabitEthernet1.45
 ip ospf transmit-delay 2
 ip ospf retransmit-interval 10
```

Verification

OSPF packet and SPF pacing/throttling timers control how fast OSPF responds to convergence events. In the majority of deployments, the default values should not need modification. Within the scope of the CCIE lab exam, determining which timers control which events should be self explanatory based on the usage guidelines of the commands in the OSPF command reference section of the documentation. These timers can be verified as follows.

```
R5#show ip ospf
 Routing Process "ospf 1" with ID 150.1.5.5
 Start time: 07:12:32.508, Time elapsed: 05:38:36.303
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101)
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 It is an area border router
 Router is not originating router-LSAs with maximum metric Initial SPF schedule delay 100 msecs
 Minimum hold time between two consecutive SPFs 1000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
```

https://t.me/learningnets

```
 Incremental-SPF disabled Initial LSA throttle delay 10 msecs
Minimum hold time for LSA throttle 4000 msecs
Maximum wait time for LSA throttle 6000 msecs
Minimum LSA arrival 2000 msecs
 LSA group pacing timer 240 secs Interface flood pacing timer 50 msecs
Retransmission pacing timer 75 msecs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 2. 2 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 0
 IETF NSF helper support enabled
 Cisco NSF helper support enabled
 Reference bandwidth unit is 100 mbps
<output omitted>
!
!R5#show ip ospf interface gigabitEthernet1.45
GigabitEthernet1.45 is up, line protocol is up
  Internet Address 155.1.45.5/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 150.1.5.5, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base Transmit Delay is 2 sec
, State DR, Priority 1
  Designated Router (ID) 150.1.5.5, Interface address 155.1.45.5
  Backup Designated router (ID) 150.1.4.4, Interface address 155.1.45.4
  Timer intervals configured, Hello 250 msec, Dead 1, Wait 1, Retransmit 10

    oob-resync timeout 40
    Hello due in 188 msec
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 8, maximum is 9
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.4.4  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF Resource Limiting

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure Loopback100 on R4 with IP address of 160.1.4.4/32 and redistribute it into OSPF.
- Configure R4 so that no more than 5000 LSAs can exist in the database.
  - No more than 500 of these routes should be originated through redistribution.

Configuration

R4:

```
interface Loopback100
 ip address 160.1.4.4 255.255.255.255
!
route-map CONNECTED->OSPF permit 10
 match interface Loopback100
!
router ospf 1
 redistribute connected subnets route-map CONNECTED->OSPF
 max-lsa 5000
 redistribute maximum-prefix 500
```

https://t.me/learningnets

Verification

LSA and redistributed prefix limiting in OSPF is used to prevent attacks or misconfigurations in the OSPF domain, which could interrupt normal forwarding, such as if the full Internet BGP table were accidentally redistributed into IGP. Verify the configured settings:

```
R4#show ip ospf
 Routing Process "ospf 1" with ID 150.1.4.4
 Start time: 07:12:23.280, Time elapsed: 05:51:14.036
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101) Maximum number of non self-generated LSA allowed 5000
    Current number of non self-generated LSA 55
    Threshold for warning message 75%
    Ignore-time 5 minutes, reset-time 10 minutes
    Ignore-count allowed 5, current ignore-count 0
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 It is an area border and autonomous system boundary router Redistributing External Routes from,
 connected, includes subnets in redistribution
 Maximum limit of redistributed prefixes 500

    Threshold for warning message 75%
 Router is not originating router-LSAs with maximum metric
    Unset reason: unconfigured
    Unset time: 12:36:57.639, Time elapsed: 00:26:39.677
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 2. 2 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 1
```

https://t.me/learningnets

```
 IETF NSF helper support enabled
 Cisco NSF helper support enabled
 Reference bandwidth unit is 100 mbps
<output omitted>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

Miscellaneous OSPF Features

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure R10 so that it does not generate a log message upon receipt of a Type-6 LSA advertisement.
- Configure R10 so that it does not account for the MTU value when establishing adjacencies on its GigabitEthernet1.108 interface.
- Configure R10 to reflect the following output:

```
R10#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface R8
                1   FULL/BDR        00:00:36 155.1.108.8      GigabitEthernet1.108
```

Configuration

```
R10:


ip host R8 150.1.8.8
ip ospf name-lookup
!
interface GigabitEthernet1.108
```

https://t.me/learningnets

```
 ip ospf mtu-ignore
!
router ospf 1
 ignore lsa mospf
```

Verification

Cisco's implementation of OSPFv2 does not support Multicast OSPF, which is advertised through LSA Type-6. Upon receipt of this LSA type from a non-Cisco OSPF router, a log message is generated. To disable this, issue the **`ignore lsa mospf`** command under the OSPF process.

A neighbor relationship cannot occur if two OSPF neighbors have different MTU values on their interfaces. If the MTU difference is by design, the interface-level command **`ip ospf mtu-ignore`** removes this requirement from the adjacency establishment.

The **`ip ospf name-lookup`** command performs DNS resolution on the OSPF router-id value in show commands to simplify the identification of neighbors:

```
R10#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
R8                1   FULL/BDR        00:00:36    155.1.108.8     GigabitEthernet1.108
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - OSPF

OSPF SHA Authentication

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic OSPF Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs OSPF Diagram** to complete this task.

Task

- Configure OSPF authentication between R7 and R9 as follows:
  - Use a key ID of 1 with the string **OSPFKEY**.
  - Use SHA-2 authentication algorithm with a 256 bits digest.

Configuration

```
R7 , R9:

key chain OSPF
 key 1
  key-string OSPFKEY
  cryptographic-algorithm hmac-sha-256
!
interface GigabitEthernet1.79
 ip ospf authentication key-chain OSPF
```

Verification

Type 2 OSPF authentication was extended to offer support for HMAC-SHA based authentication, and this is defined in RFC 5709. Cisco implemented support for all variants of HMAC-SHA although per the RFC, only HMAC-SHA-256 is mandatory.

https://t.me/learningnets

HMAC-SHA-1 uses the first version of SHA, while all the other methods use SHA-2, the second version of the function:

- HMAC-SHA-1 (160 bits digest)
- HMAC-SHA-256 (256 bits digest)
- HMAC-SHA-384 (384 bits digest)
- HMAC-SHA-512 (512 bits digest)

So far, IOS does not allow you to enable SHA authentication at the area level, only at the interface level. To configure SHA authentication, you first need to define the key ID and key string by using a key-chain, jut like for RIP or EIGRP, and afterwards apply the key-chain at the interface using command **`ip ospf authentication key-chain <NAME>`** . By using a key-chain, which allows defining several keys with specific lifetimes, the whole process of key rotation becomes simpled and more efficient, just like in EIGRP, as OSPF will always send OSPF packets authenticated with the first valid key from the key-chain, but received OSPF packets will be matched against all valid configured keys.

With the current IOS code, SHA authentication is not supported for virtual-links. In case of a mismatch between key ID, key string or authentication type, the debug messages are identical with the case when using MD5 authentication. With the addition of SHA to Type 2 authentication, this is now called a "Cryptographic Authentication" for both MD5 and SHA, which is also visible from the outputs. Verify that R7 and R9 use correctly use SHA-256 authentication:

```
R9#show ip ospf interface gigabitEthernet1.79
GigabitEthernet1.79 is up, line protocol is up
  Internet Address 155.1.79.9/24, Area 2, Attached via Network Statement
  Process ID 1, Router ID 150.1.9.9, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 150.1.9.9, Interface address 155.1.79.9
  Backup Designated router (ID) 150.1.7.7, Interface address 155.1.79.7
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:05
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 2
```

https://t.me/learningnets

```
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 150.1.7.7  (Backup Designated Router)
  Suppress hello for 0 neighbor(s) Cryptographic authentication enabled
 Sending SA: Key 1, Algorithm HMAC-SHA-256 - key chain OSPF
!
!R9#show key chain
Key-chain OSPF: key 1 -- text "OSPFKEY"

        accept lifetime (always valid) - (always valid) [valid now]
        send lifetime (always valid) - (always valid) [valid now]
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

Establishing iBGP Peerings

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial BGP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure BGP R1 - R8 routers using AS 100.
- Create a full mesh of iBGP peerings between these devices without using their Loopback0 interfaces.
- Advertise the Loopback0 prefixes of these devices into BGP.
- Ensure full IPv4 reachability for Loopback0 prefixes of R1 - R8 routers.

Configuration

```
R1:
router bgp 100
 network 150.1.1.1 mask 255.255.255.255
 neighbor 155.1.0.2 remote-as 100
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.4 remote-as 100
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.146.6 remote-as 100

R2:
router bgp 100
 network 150.1.2.2 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
```

https://t.me/learningnets

```
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.4 remote-as 100
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.37.7 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.146.6 remote-as 100
```

R3:

```
router bgp 100
 network 150.1.3.3 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.2 remote-as 100
 neighbor 155.1.0.4 remote-as 100
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.37.7 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.146.6 remote-as 100
```

R4:

```
router bgp 100
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.2 remote-as 100
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.146.6 remote-as 100
```

R5:

```
router bgp 100
 network 150.1.5.5 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.2 remote-as 100
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.4 remote-as 100
 neighbor 155.1.37.7 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.146.6 remote-as 100
```

R6:

```
router bgp 100
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.0.2 remote-as 100
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.5 remote-as 100
```

https://t.me/learningnets

```
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.146.4 remote-as 100
```

R7:

```
router bgp 100
 network 150.1.7.7 mask 255.255.255.255
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.23.2 remote-as 100
 neighbor 155.1.37.3 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.67.6 remote-as 100
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.146.4 remote-as 100
```

R8:

```
router bgp 100
 network 150.1.8.8 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.2 remote-as 100
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.4 remote-as 100
 neighbor 155.1.37.7 remote-as 100
 neighbor 155.1.58.5 remote-as 100
 neighbor 155.1.146.6 remote-as 100
```

Verification

The first step in any BGP configuration is always to establish peering relationships between the BGP speaking devices. Recall that because BGP does not have its own transport protocol, underlying IGP reachability must already be established to allow the TCP port 179 sessions to be successful between neighbors.

https://t.me/learningnets

BGP is a normal TCP application, which means that a TCP client initiates the session to the TCP server with a SYN packet going to the well-known port of 179. If the BGP server is configured to accept the session, a reply with SYN/ACK comes from port 179 back to the client, going to the high port number generated by the client. If both BGP peers attempt to establish the connection at the same time, RFC 4271 (A Border Gateway Protocol 4) defines a “BGP Connection Collision Detection” mechanism, in which essentially the session originated from the device with the higher BGP router-id is maintained, and the secondary session is dropped.

The below debug output shows the step-by-step formation of the iBGP peering between R1 and R2. Note that access-list 100 is used to filter the debug output and only show the output pertinent to the BGP session between R1 and R2:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access-list 100 permit tcp any host 155.1.0.2
R1(config)#access-list 100 permit tcp host 155.1.0.2 any
R1(config)#do debug ip packet detail 100
IP packet debugging is on (detailed) for access list 100R1(config)#router bgp 100
R1(config-router)#neighbor 155.1.0.2 remote-as 100
R1(config-router)# IP: s=155.1.0.1 (local), d=155.1.0.2
, len 44, local feature TCP src=11712, dst=179, seq=4030024239, ack=0, win=16384 SYN
```

R1 is configured with the neighbor statement pointing toward R2, and R2 is already configured with the neighbor statement pointing toward R1. A SYN is sent from R1 to initiate the session. Notice the source port of 11712 and the destination port of 179:

```
IP: s=155.1.0.2 (Tunnel0), d=155.1.0.1
, len 40, enqueue feature TCP src=179, dst=11712, seq=3286835183, ack=4030024297, win=16327 ACK SYN
,
```

Because R2 has a neighbor statement pointing back toward the address 155.1.0.1, a reply of ACK/SYN is received, with the source port of 179 and the destination port randomly generated by R1, 11712. These two steps indicate that R1 is the client and R2 is the server:

```
IP: s=155.1.0.1 (local), d=155.1.0.2
, len 40, local feature TCP src=11712, dst=179, seq=4030024441, ack=3286835384, win=16183 ACK
,
```

https://t.me/learningnets

R1 replies with ACK, completing the 3-way TCP handshake and opening the session for BGP attribute negotiation. Only after necessary parameters are correctly negotiated, such as remote-as numbers, authentication, and address-family support, will the BGP session actually be declared up:

```
    IP: s=155.1.0.2 (Tunnel0), d=155.1.0.1
, len 142, enqueue feature TCP src=179, dst=11712, seq=3286835282, ack=4030024339, win=16285 ACK PSH
<output omitted> %BGP-5-ADJCHANGE: neighbor 155.1.0.2 Up
```

The details of the peering negotiation between R1 and R2, such as the router-id and timers, can be seen below. The neighbor capability of “Address family IPv4 Unicast: advertised and received” means that by default, they can exchange IPv4 prefixes, but not other such as IPv6 Unicast or VPNv4/VPNv6. The details for IPv4 unicast address family show that one prefix has been received from the neighbor and one prefix has been advertised to the neighbor:

```
R1#show ip bgp neighbor 155.1.0.2
BGP neighbor is 155.1.0.2,  remote AS 100
, internal link BGP version 4, remote router ID 150.1.2.2
BGP state = Established, up for 00:23:43 Last read 00:00:44, last write 00:00:00,
hold time is 180, keepalive interval is 60 seconds
Neighbor sessions:
1 active, is not multisession capable (disabled)
Neighbor capabilities: Route refresh: advertised and received(new)
Four-octets ASN Capability: advertised and received
Address family IPv4 Unicast: advertised and received
Enhanced Refresh Capability: advertised and received
Multisession Capability:
Stateful switchover support enabled: NO for session 1
Message statistics:
InQ depth is 0
OutQ depth is 0

                      Sent       Rcvd
Opens:                   1          1
Notifications:           0          0
Updates:                 2          2
Keepalives:             28         27
Route Refresh:           0          0
Total:                  33         32
Default minimum time between advertisement runs is 0 seconds

For address family: IPv4 Unicast
Session: 155.1.0.2
```

https://t.me/learningnets

```
BGP table version 15, neighbor version 15/0
Output queue size : 0
Index 1, Advertise bit 0
1 update-group member
Slow-peer detection is disabled
Slow-peer split-update-group dynamic is disabled
Interface associated: (none) Sent       Rcvd
  Prefix activity:                ----       ----
  Prefixes Current:                1          1 (Consumes 120 bytes)

    Prefixes Total:                 9          1
    Implicit Withdraw:              8          0
    Explicit Withdraw:              0          0
    Used as bestpath:             n/a          1
    Used as multipath:            n/a          0

                                   Outbound    Inbound
  Local Policy Denied Prefixes:    --------    -------
    Bestpath from this peer:               8        n/a
    Bestpath from iBGP peer:              30        n/a
    Total:                                38          0
  Number of NLRIs in the update sent: max 1, min 0
```

The TTL of the outbound session is set to 255, because this is an iBGP session. This means that the iBGP neighbors need not be directly connected, as long as IGP reachability exists between them. EBGP sessions have a TTL of 1 by default, which means that neighbors must be directly connected, unless further configuration is applied. Also note the “Local port: 11712” and “Foreign port: 179.” These essentially mean the source and destination ports from R1’s perspective, which again enforces the notion that R1 is the client for this session and R2 is the server:

```
  Connections established 1; dropped 0
  Last reset never
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 155.1.0.1, Local port: 11712
Foreign host: 155.1.0.2, Foreign port: 179

<output omitted>
```

The following **`show ip bgp summary`** output shows a concise aggregation of all configured neighbors, with the important fields being the local AS, router-id, table size in both prefixes and memory, peer addresses, remote ASs, neighbor uptime,

https://t.me/learningnets

and number of prefixes learned:

```
R1#show ip bgp summary
BGP router identifier 150.1.1.1, local AS number 100
BGP table version is 15, main routing table version 15 8 network entries
 using 1984 bytes of memory
8 path entries using 960 bytes of memory
2/2 BGP path/bestpath attribute entries using 480 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory BGP using 3424 total bytes of memory
BGP activity 11/3 prefixes, 11/3 paths, scan interval 60 secs
 Neighbor        V  AS  MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
155.1.0.2        4 100       7       7       15    0    0 00:37:24   1
155.1.0.3        4 100       7       7       15    0    0 01:14:33   1
155.1.0.4        4 100       7       7       15    0    0 01:14:28   1
155.1.0.5        4 100       7       7       15    0    0 01:14:23   1
155.1.58.8       4 100       5       6       15    0    0 01:13:43   1
155.1.67.7       4 100       5       6       15    0    0 01:14:06   1
155.1.146.6      4 100       7       7       15    0    0 01:14:20   1
```

Note that for devices running multiple address-families, such as IPv4 unicast VPNv4/VPNv6, show commands are expressed as **`show bgp [AFI] [SAFI] [args]`**, such as the **`show bgp ipv4 unicast summary`** seen below. Although the output is the same as the above **`show ip bgp summary`**, it can quickly become hard to tell what output you are viewing in larger-scale BGP deployments without this logical separation:

```
R1#show bgp ipv4 unicast summary

BGP router identifier 150.1.1.1, local AS number 100
BGP table version is 15, main routing table version 15
8 network entries using 1984 bytes of memory
8 path entries using 960 bytes of memory
2/2 BGP path/bestpath attribute entries using 480 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 3424 total bytes of memory
BGP activity 11/3 prefixes, 11/3 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
155.1.0.2       4          100      50      51       15    0    0 00:39:56        1
155.1.0.3       4          100      88      91       15    0    0 01:17:06        1
155.1.0.4       4          100      88      89       15    0    0 01:17:00        1
155.1.0.5       4          100      88      91       15    0    0 01:16:55        1
```

https://t.me/learningnets

```
155.1.58.8      4          100      88      91       15    0    0 01:16:15        1
155.1.67.7      4          100      88      92       15    0    0 01:16:38        1
155.1.146.6     4          100      87      90       15    0    0 01:16:52        1
```

After the peering relationships are established, the actual BGP prefixes learned can be viewed in the BGP table with **`show ip bgp`** or **`show bgp ipv4 unicast .`** This output is crucial to completely understand, especially when multiple paths to the same destination exist:

```
R1#show ip bgp
BGP table version is 15, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  150.1.1.1/32     0.0.0.0                  0         32768 i
 *>i 150.1.2.2/32     155.1.0.2                0    100      0 i
 *>i 150.1.3.3/32     155.1.0.3                0    100      0 i
 *>i 150.1.4.4/32     155.1.0.4                0    100      0 i
 *>i 150.1.5.5/32     155.1.0.5                0    100      0 i
 *>i 150.1.6.6/32     155.1.146.6              0    100      0 i
 *>i 150.1.7.7/32     155.1.67.7               0    100      0 i
 *>i 150.1.8.8/32     155.1.58.8               0    100      0 i
!R1#show ip route bgp

      150.1.0.0/32 is subnetted, 8 subnets
B        150.1.2.2 [200/0] via 155.1.0.2, 00:42:08
B        150.1.3.3 [200/0] via 155.1.0.3, 01:18:11
B        150.1.4.4 [200/0] via 155.1.0.4, 01:18:07
B        150.1.5.5 [200/0] via 155.1.0.5, 01:18:01
B        150.1.6.6 [200/0] via 155.1.146.6, 01:17:54
B        150.1.7.7 [200/0] via 155.1.67.7, 01:17:43
B        150.1.8.8 [200/0] via 155.1.58.8, 01:17:43
```

One of the most important fields in the above outputs is the next-hop value. Note that this field is set to the peering address for the neighbor from which the route is learned. For example, the prefix 150.1.7.7/32 is via the next-hop 155.1.67.7. To reach this prefix, a recursive lookup must now be performed on the next-hop until an outgoing interface is found:

https://t.me/learningnets

```
R1#show ip route 150.1.7.7

Routing entry for 150.1.7.7/32
  Known via "bgp 100", distance 200, metric 0, type internal
  Last update from 155.1.67.7 01:18:46 ago
  Routing Descriptor Blocks:   * 155.1.67.7
, from 155.1.67.7, 01:18:46 ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
  MPLS label: none
!R1#show ip route 155.1.67.7
Routing entry for 155.1.67.0/24
  Known via "eigrp 100", distance 90, metric 3072, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.6 on GigabitEthernet1.146, 02:51:42 ago
  Routing Descriptor Blocks:   * 155.1.146.6
, from 155.1.146.6, 02:51:42 ago, via GigabitEthernet1.146
      Route metric is 3072, traffic share count is 1
      Total delay is 20 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R1#show ip route 155.1.146.6
Routing entry for 155.1.146.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks:   * directly connected, via GigabitEthernet1.146

      Route metric is 0, traffic share count is 1
```

In the above case, recursion toward 150.1.7.7/32 continues until the outgoing interface GigabitEthernet1.146 is found. Note that unless route-recursion toward the next-hop of a BGP prefix is successful, the route cannot be considered for best path selection, which also implies that it cannot be installed in the IP routing table or advertised to any other BGP peer. This issue will be explored in detail in the coming tasks.

https://t.me/learningnets

Also note that in this task, a full-mesh of iBGP peerings is established. This is because of the design requirement that an iBGP learned route cannot be advertised to another iBGP neighbor to prevent routing loops, unless exceptions such as route-reflection or confederation are implemented. The result is that the only routes advertised to the other BGP peers are the local Loopback0 prefixes, but not any of the routes learned from the other neighbors. This also implies that in this design, if any individual peering breaks, connectivity between prefixes advertised by those peers also breaks:

```
R1#show ip bgp neighbors 155.1.0.2 advertised-routes

BGP table version is 15, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>  150.1.1.1/32     0.0.0.0                  0         32768 i


Total number of prefixes 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

Establishing eBGP Peerings

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **eBGP with R9 to R10**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- R9 and R10 are configured for BGP in AS 54 and all other routers for BGP in AS 100.
- Configure EBGP peerings between R7 and R9, between R8 and R10 using their directly connected links.
- Advertise the directly connected links between R7 and R9, between R8 and R10 into EIGRP 100.
- Ensure full IPv4 reachability to all prefixes learned from AS 54, from all routers in AS 100 when sourcing traffic from Loopback0 interfaces.

Configuration

```
R7:
router eigrp 100
 network 155.1.79.0 0.0.0.255
!
router bgp 100
 neighbor 155.1.79.9 remote-as 54

R8:

router eigrp 100
 network 155.1.108.0 0.0.0.255
!
```

https://t.me/learningnets

```
router bgp 100
 neighbor 155.1.108.10 remote-as 54
```

Verification

From a configuration perspective, the only difference between an iBGP peering and an EBGP peering is that with an EBGP peering, the remote-as is different than the local-as. In practice, however, many important attributes differ between the two. As shown in the following output, R4 knows that the peering occurs on a directly connected external link and that the TTL should be 1 instead of 255. This implies that the neighbors must be directly connected for peering to be established, otherwise the TTL would be exceeded in transit and the TCP frames would be dropped:

```
R7#show ip bgp neighbors 155.1.79.9
BGP neighbor is 155.1.79.9, remote AS 54, external link
  BGP version 4, remote router ID 150.1.9.9
  BGP state = Established, up for 00:02:24
<output omitted>
!Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 1

Local host: 155.1.79.7, Local port: 17650
Foreign host: 155.1.79.9, Foreign port: 179
<output omitted>
```

Because both R9 and R10 run BGP in AS 54, the same view should be coming in from both peers. In the output below, R7 sees 10 prefixes learned from R9 and 11 prefixes learned from R8 (R8's Loopback0 plus the ten AS 54 prefixes from R10):

```
R7#show ip bgp summary
BGP router identifier 150.1.7.7, local AS number 100
BGP table version is 19, main routing table version 19
18 network entries using 4464 bytes of memory
28 path entries using 3360 bytes of memory
8/6 BGP path/bestpath attribute entries using 1920 bytes of memory
2 BGP AS-PATH entries using 64 bytes of memory
1 BGP community entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 9832 total bytes of memory
BGP activity 18/0 prefixes, 28/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
```

https://t.me/learningnets

```
155.1.0.5       4          100     879     887       19    0    0 13:19:05        1
155.1.23.2      4          100     885     886       19    0    0 13:19:11        1
155.1.37.3      4          100     884     887       19    0    0 13:19:15        1
155.1.58.8      4          100     883     885       19    0    0 13:18:59       11
155.1.67.6      4          100     885     885       19    0    0 13:19:16        1
155.1.79.9      4           54      13      11       19    0    0 00:04:42       10

155.1.146.1     4          100     887     886       19    0    0 13:19:15        1
155.1.146.4     4          100     883     889       19    0    0 13:19:10        1
```

Because duplicate routing information is learned, the BGP Bestpath Selection process must be run to choose one path over the other. The path that is active, known as the “best” path, can be seen from the greater-than sign (>) in the left column of the `show ip bgp` output:

```
R1#show ip bgp
BGP table version is 25, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 28.119.16.0/24   155.1.108.10             0    100      0 54 i *>
i                  155.1.79.9               0    100      0 54 i
 * i 28.119.17.0/24   155.1.108.10             0    100      0 54 i *>
i                  155.1.79.9               0    100      0 54 i
 * i 112.0.0.0        155.1.108.10             0    100      0 54 50 60 i *>
i                  155.1.79.9               0    100      0 54 50 60 i
 * i 113.0.0.0        155.1.108.10             0    100      0 54 50 60 i *>
i                  155.1.79.9               0    100      0 54 50 60 i
 * i 114.0.0.0        155.1.108.10             0    100      0 54 i *>
i                  155.1.79.9               0    100      0 54 i
 * i 115.0.0.0        155.1.108.10             0    100      0 54 i *>
i                  155.1.79.9               0    100      0 54 i
```

The best path is the only route that is installed in the routing table, and the only route that is advertised to other BGP neighbors. From the above output of R1, we can see that the best path for all AS 54 routes is via 155.1.79.9. Why this selection occurs can be seen from the detailed output below:

```
R1#show ip bgp 112.0.0.0 255.0.0.0
```

https://t.me/learningnets

```
BGP routing table entry for 112.0.0.0/8, version 18
Paths: (2 available, best #2, table default)
  Not advertised to any peer
  Refresh Epoch 1
  54 50 60
    155.1.108.10 (metric 3584) from 155.1.58.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  54 50 60 155.1.79.9 (metric 3328) from 155.1.67.7 (150.1.7.7
)      Origin IGP, metric 0, localpref 100, valid, internal, best

      rx pathid: 0, tx pathid: 0x0
```

Note the difference between the next-hop value and the neighbor from which the prefix is learned. The first highlighted value, 155.1.79.9, is the next-hop that R1 needs to be able to perform route-recursion toward to use the route. The next value, 155.1.67.7, is the BGP peer address that R1 uses to reach R7. The final address, 150.1.7.7, is the BGP router-id of R7. In this case, the route through R7 is chosen over R8's route because of the lower metric of 3328 toward the next-hop 155.1.79.9, as opposed to the metric 3584 toward 155.1.108.10. Bestpath selection will be discussed in detail in further tasks.

When R1 does its final lookup on 112.0.0.0/8, recursion continues toward the next-hop 155.1.79.9, resulting in two outgoing interfaces (ECMP), GigabitEthernet1.146 and GigabitEthernet1.13, toward 155.1.146.6 and 155.1.13.3, respectively:

```
R1#show ip route 112.0.0.0

Routing entry for 112.0.0.0/8
  Known via "bgp 100", distance 200, metric 0
  Tag 54, type internal
  Last update from 155.1.79.9 00:27:31 ago
  Routing Descriptor Blocks:   * 155.1.79.9
, from 155.1.67.7, 00:27:31 ago
      Route metric is 0, traffic share count is 1
      AS Hops 3
      Route tag 54
  MPLS label: none
!R1#show ip route 155.1.79.9
Routing entry for 155.1.79.0/24
  Known via "eigrp 100", distance 90, metric 3328, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.6 on GigabitEthernet1.146, 00:38:08 ago
  Routing Descriptor Blocks:
```

https://t.me/learningnets

```
  * 155.1.146.6
, from 155.1.146.6, 00:38:08 ago, via GigabitEthernet1.146
      Route metric is 3328, traffic share count is 1
      Total delay is 30 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2 155.1.13.3
, from 155.1.13.3, 00:38:08 ago, via GigabitEthernet1.13
      Route metric is 3328, traffic share count is 1
      Total delay is 30 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
!R1#show ip route 155.1.146.6
Routing entry for 155.1.146.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks:  * directly connected, via GigabitEthernet1.146
      Route metric is 0, traffic share count is 1
!R1#show ip route 155.1.13.3
Routing entry for 155.1.13.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks:  * directly connected, via GigabitEthernet1.13
      Route metric is 0, traffic share count is 1
!R1#traceroute 112.0.0.1 source Loopback 0

Type escape sequence to abort.
Tracing the route to 112.0.0.1
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.13.3 5 msec
    155.1.146.6 1 msec
    155.1.13.3 1 msec
  2 155.1.67.7 2 msec
    155.1.37.7 1 msec
    155.1.67.7 1 msec
  3 155.1.79.9 3 msec *  5 msec
```

Pitfall

Note that R7 and R8 did not update the next-hop values when advertising an EBGP learned route to their iBGP peers. If the iBGP peers do not have a route to the next-hop value in the prefix, bestpath selection fails and the route cannot be used. Fixes for this problem will be explored in depth in further tasks.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Update Source Modification

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Update Source**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Advertise Loopback0 prefixes of R4 and R5 into EIGRP 100.
- Modify the BGP peering between R4 and R5 so that if either the DMVPN Tunnel or VLAN 45 Ethernet link goes down, the BGP peering is not affected.

Configuration

https://t.me/learningnets

```
R4:

router eigrp 100
 network 150.1.4.4 0.0.0.0
!
router bgp 100
 neighbor 150.1.5.5 remote-as 100
 neighbor 150.1.5.5 update-source Loopback0

R5:

router eigrp 100
 network 150.1.5.5 0.0.0.0
!
router bgp 100
 neighbor 150.1.4.4 remote-as 100
 neighbor 150.1.4.4 update-source Loopback0
```

Verification

Because BGP peerings use TCP for transport, it is not a requirement that neighbors be directly connected. When neighbors are not directly connected, the choice of IP addresses used in peering can greatly affect the redundancy design of a BGP network. In the previous case, the peering between R4 and R5 was configured using their connected DMVPN Tunnel interface IP addresses. This implies that if the DMVPN link were to go down, the BGP peering would also go down, even if alternate routes still existed between the two devices. To fix this redundancy issue, the `update-source` for a BGP peering session can be changed on a per-neighbor basis.

Normally the IP source address used in a BGP packet is the IP address of the outgoing interface in the routing table. For example, before the above modifications, R5 used the address 155.1.0.4 to reach R4 for the BGP peering:

https://t.me/learningnets

```
R5#show ip route 155.1.0.4
                                Routing entry for 155.1.0.0/24

  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks:   * directly connected, via Tunnel0

  Route metric is 0, traffic share count is 1
!R5#show ip interface brief | include Tunnel0
           Tunnel0 155.1.0.5
      YES manual up                    up
```

Based on the fact that R5 routes out Tunnel0 to reach 155.1.0.4, the source address in the IP packet is 155.1.0.5. Observe what occurs with the BGP session when this interface is down:

```
R5(config)#interface Tunnel0
R5(config-if)#shutdown

%CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to down
%LINK-5-CHANGED: Interface Tunnel0, changed state to administratively down
!R5#show ip route 155.1.0.4
Routing entry for 155.1.0.0/24
Known via "eigrp 100", distance 90, metric 26880256, type internal
Redistributing via eigrp 100
Last update from 155.1.45.4 on GigabitEthernet1.45, 00:03:36 ago
Routing Descriptor Blocks: * 155.1.45.4, from 155.1.45.4, 00:03:36 ago, via GigabitEthernet1.45
  Route metric is 26880256, traffic share count is 1
  Total delay is 50010 microseconds, minimum bandwidth is 100 Kbit
  Reliability 255/255, minimum MTU 1400 bytes
  Loading 1/255, Hops 1
!
%BGP-3-NOTIFICATION: sent to neighbor 155.1.0.4 4/0 (hold time expired) 0 bytes
%BGP-5-NBR_RESET: Neighbor 155.1.0.4 reset (BGP Notification sent)
%BGP-5-ADJCHANGE: neighbor 155.1.0.4 Down BGP Notification sent

%BGP_SESSION-5-ADJCHANGE: neighbor 155.1.0.4 IPv4 Unicast topology base removed from session  BGP Notification sent
R5#
%BGP-3-NOTIFICATION: sent to neighbor 155.1.0.1 4/0 (hold time expired) 0 bytes
```

With a route to 155.1.0.4 pointing out of the GigabitEthernet1.45 interface, the BGP

https://t.me/learningnets

session is lost, even though the GigabitEthernet1.45 link could have been used for rerouting. The session is torn down because R4 is still using its DMVPN Tunnel interface and sourcing packets from 155.1.0.4 to get to R5 (155.1.0.5). However, R5 is using its GigabitEthernet1.45 interface and sourcing packets from 155.1.45.5 to get to R4 (155.1.0.4):

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#access-list 100 permit tcp host 155.1.45.5 host 155.1.0.4
R4(config)#access-list 100 permit tcp any host 155.1.45.5
R4#debug ip packet detail 100
IP packet debugging is on (detailed) for access list 100R4#debug ip bgp
IP: s=155.1.45.5 (GigabitEthernet1.45), d=155.1.0.4
, len 44, enqueue feature TCP src=32053, dst=179, seq=3906274821, ack=0, win=16384 SYN
, TCP Adjust MSS(5), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE IP: s=155.1.0.4 (local),
d=155.1.45.5
, len 40, local feature TCP src=179, dst=32053, seq=0, ack=3906274822, win=0 ACK RST
, feature skipped, Logical MN local(14), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
BGP: 155.1.0.5 active went from Active to Idle
BGP: nbr global 155.1.0.5 Active open failed
 - open timer running BGP: nbr global 155.1.0.5 Active open failed
 - open timer running
```

Based on the debug, we can see that the session attempts establishment, but R4 replies with ACK RST, refusing and closing the session. This is because R5's route to 150.1.4.4 is out the GigabitEthernet1.45 link, causing the source IP address to be 155.1.45.5. R4 has its neighbor statement pointing at 155.1.0.5, not 155.1.45.5, so the connection is refused.

The important thing to remember here is that *the TCP server of the BGP session must approve where the session is coming from*. If the SYN packet arrives from an address that is not specified in a neighbor statement, the connection is refused. To remedy this, the neighbor statement on R4 can be changed to point to 155.1.45.5 instead of 155.1.0.5:

https://t.me/learningnets

```
R4#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#router bgp 100
R4(config-router)#no neighbor 155.1.0.5
R4(config-router)#neighbor 155.1.45.5 remote-as 100

!R4#debug ip packet detail 100

IP packet debugging is on (detailed) for access list 100
via RIB IP: s=155.1.45.5 (GigabitEthernet1.45), d=155.1.0.4
, len 44, enqueue feature TCP src=12812, dst=179, seq=2285379679, ack=0, win=16384 SYN
,  IP: s=155.1.0.4 (local), d=155.1.45.5
, len 44, local feature TCP src=179, dst=12812, seq=438597729, ack=2285379680, win=16384 ACK SYN
,
```

Now R5 sends a SYN to 155.1.0.4 (R4's Tunnel 0 address), sourced from 155.1.45.5 (R5's GigabitEthernet1.45). Because R4 already has a neighbor statement for 155.1.45.5, SYN ACK is returned and the session opens:

```
<output omitted>%BGP-5-ADJCHANGE: neighbor 155.1.45.5 Up
```

If the peering between physical interfaces is reverted and R4 and R5 are configured via their Loopback0 addresses, this problem can be avoided. If one of the links between the neighbors goes down, the peering is simply rerouted based on the convergence of IGP:

```
R5(config-router)#no neighbor 155.1.0.4
R5(config-router)#neighbor 150.1.4.4 remote-as 100
R5(config-router)#neighbor 150.1.4.4 update-source Loopback0
!R4(config-router)#no neighbor 155.1.45.5
R4(config-router)#neighbor 150.1.5.5 remote-as 100
R4(config-router)#neighbor 150.1.5.5 update-source Loopback0
R5(config)#interface Tunnel0
R5(config-if)#no shutdown
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "eigrp 100", distance 90, metric 130816, type internal
  Redistributing via eigrp 100
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:01:17 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 155.1.45.4, 00:01:17 ago, via GigabitEthernet1.45
      Route metric is 130816, traffic share count is 1
      Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
```

https://t.me/learningnets

```
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface GigabitEthernet1.45
R5(config-if)#shutdown
!
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.45.4 (GigabitEthernet1.45) is down: interface down
R5#show ip route 150.1.4.4
Routing entry for 150.1.4.4/32
  Known via "eigrp 100", distance 90, metric 25984000, type internal
  Redistributing via eigrp 100
  Last update from 155.1.0.4 on Tunnel0, 00:02:12 ago
  Routing Descriptor Blocks: * 155.1.0.4, from 155.1.0.4, 00:02:12 ago, via Tunnel0

      Route metric is 25984000, traffic share count is 1
      Total delay is 15000 microseconds, minimum bandwidth is 100 Kbit
      Reliability 255/255, minimum MTU 1400 bytes
      Loading 1/255, Hops 1
```

In this scenario, R5 prefers to route over the GigabitEthernet1.45 interface to reach Loopback0 of R4 because of the EIGRP composite metric. The cost to reach R4's Loopback0 via R5's Tunnel interface is much higher. However, as soon as the GigabitEthernet1.45 interface of R5 is disabled, EIGRP convergence reroutes traffic via the next-best path toward R4. Observe that the BGP session does not go down.

Note that technically, only one neighbor needs to add the update-source command, as long as both agree on the destination of the peering. If R4 sets the update source to Loopback 0, but R5 does not, this ensures that R4 is always the TCP client and R5 is always the TCP server. In most designs, the update sources are modified on both sides for clarity.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

Multihop EBGP Peerings

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **eBGP Multihop**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a new Loopback1 interface on R8 with the IPv4 address 204.12.8.8/32, and advertise it into EIGRP.
- Configure an EBGP peering between R8 and R9 in AS 54 using this new interface as the source of the peering.

Configuration

https://t.me/learningnets

```
R8:

interface Loopback1
 ip address 204.12.8.8 255.255.255.255
!
router eigrp 100
 network 204.12.8.8 0.0.0.0
!
router bgp 100
 neighbor 155.1.79.9 remote-as 54
 neighbor 155.1.79.9 ebgp-multihop 255
 neighbor 155.1.79.9 update-source Loopback1

R9:

router bgp 54
 neighbor 204.12.8.8 remote-as 100
 neighbor 204.12.8.8 ebgp-multihop 255
```

Verification

As seen in previous output, the default TTL for EBGP peers is 1. This means that non-directly connected EBGP peers cannot be established, because the TTL will expire in transit. By issuing the **`ebgp-multihop [ttl]`** command, the TTL can be increased to support this type of design:

```
R8#show ip bgp summary | include 155.1.79.9
155.1.79.9    4    54      19      17       74    0    0 00:09:36 10
!R8#show ip bgp neighbors 155.1.79.9
BGP neighbor is 155.1.79.9,  remote AS 54, external link
  BGP version 4, remote router ID 212.18.3.1 BGP state = Established
, up for 00:02:16
  Last read 00:00:22, last write 00:00:51, hold time is 180, keepalive interval is 60 seconds
<output omitted> External BGP neighbor may be up to 255 hops away.
  Transport(tcp) path-mtu-discovery is enabled
Connection state is ESTAB, I/O status: 1, unread input bytes: 0 Mininum incoming TTL 0, Outgoing TTL 255

Local host: 204.12.8.8, Local port: 179
Foreign host: 155.1.79.9, Foreign port: 27742
<output omitted>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

Neighbor Disable-Connected-Check

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Disable Connected Check**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R2 and R5 in AS 100, and R4 in AS 200 as follows:
  - Configure an iBGP peering between R2 and R5.
  - Configure an EBGP peering between R2 and R10, which is in AS 54.
  - Configure an EBGP peering between R2 and R4.
  - Configure an EBGP peering between R4 and R5 so that the peering remains up as long as R4's connection to either DMVPN cloud or VLAN 45 is up.
  - Do not use `ebgp-multihop` option on the peering between R4 and R5.
- Advertise the 155.1.108.0/24 network into EIGRP on R8.
- Advertise the Loopback0 prefixes of R4 and R5 into EIGRP.
- Add a static route on R10 for 155.1.0.0/16 with a next hop of 155.1.108.8.
- Ensure that R2's GigabitEthernet1.23 interface is disabled.

Configuration

```
R2:
 router bgp 100
  neighbor 155.1.0.4 remote-as 200
  neighbor 155.1.0.5 remote-as 100
  neighbor 155.1.108.10 remote-as 54
  neighbor 155.1.108.10 ebgp-multihop 255
 !
 interface GigabitEthernet1.23
```

https://t.me/learningnets

```
 shutdown

R4:
router bgp 200
 neighbor 150.1.5.5 remote-as 100
 neighbor 150.1.5.5 disable-connected-check
 neighbor 150.1.5.5 update-source Loopback0
 neighbor 155.1.0.2 remote-as 100
!
router eigrp 100
 network 150.1.4.4 0.0.0.0

R5:
router bgp 100
 neighbor 150.1.4.4 remote-as 200
 neighbor 150.1.4.4 disable-connected-check
 neighbor 150.1.4.4 update-source Loopback0
 neighbor 155.1.0.2 remote-as 100
!
router eigrp 100
 network 150.1.5.5 0.0.0.0

R8:
router eigrp 100
 network 155.1.108.8 0.0.0.0

R10:

ip route 155.1.0.0 255.255.0.0 155.1.108.8
```

Verification

Recall that with default EBGP sessions, a TTL of one prevents non-directly connected neighbors from forming. Additionally, IOS prevents the initiation of non-directly connected EBGP sessions when the TTL is one (multihop is not configured), because it assumes that the TTL will expire in transit.

As previously seen, one way of resolving this problem is to simply increase the TTL between the peers. In designs where the peers are directly connected but the peering address is a Loopback instead of the connected interface between them, the `disable-connected-check` neighbor option may also be used.

Although similar in result to increasing the EBGP TTL, the difference between these features is that the `disable-connected-check` prevents cases in which the EBGP

https://t.me/learningnets

session between two devices is routed over another transit router.

For example, in this case, R4 and R5 peer with each other's Loopback 0 interfaces, but they do not increase the TTL:

```
R4#show ip bgp neighbor 150.1.5.5 | include TTL
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 1
!R4#show ip bgp summary
BGP router identifier 150.1.4.4, local AS number 200
BGP table version is 11, main routing table version 11

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
150.1.5.5       4          100      22      23       11    0    0 00:16:27        10

<output omitted>
```

Because the router does not decrement the TTL to a packet destined to itself, it technically only counts as one hop from R4 to R5's Loopback0. Now let's see what happens when R4's direct links to R5 are down, but a route still remains to R5's Loopback0.

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface Tunnel0
R4(config-if)#shutdown
!
%BGP-5-NBR_RESET: Neighbor 155.1.0.2 reset (Interface flap)
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.5 (Tunnel0) is down: interface down
%BGP-5-ADJCHANGE: neighbor 155.1.0.2 Down Interface flap
%BGP_SESSION-5-ADJCHANGE: neighbor 155.1.0.2 IPv4 Unicast topology base removed from session  Interface flap
%CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to down
%LINK-5-CHANGED: Interface Tunnel0, changed state to administratively down
!R4(config-if)#interface GigabitEthernet1.45
R4(config-subif)#shutdown
!
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.45.5 (GigabitEthernet1.45) is down: interface down
%BGP-3-NOTIFICATION: sent to neighbor 150.1.5.5 4/0 (hold time expired)
 0 bytes %BGP-5-NBR_RESET: Neighbor 150.1.5.5 reset (BGP Notification sent)
%BGP-5-ADJCHANGE: neighbor 150.1.5.5 Down BGP Notification sent
!R4#show ip route 150.1.5.5
Routing entry for 150.1.5.5/32
  Known via "eigrp 100", distance 90, metric 27008256, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.1 on GigabitEthernet1.146, 00:04:33 ago
  Routing Descriptor Blocks:
```

https://t.me/learningnets

```
* 155.1.146.1, from 155.1.146.1, 00:04:33 ago, via GigabitEthernet1.146

      Route metric is 27008256, traffic share count is 1
      Total delay is 55010 microseconds, minimum bandwidth is 100 Kbit
      Reliability 255/255, minimum MTU 1400 bytes
      Loading 1/255, Hops 2
```

Even though a route remains between R4 and R5's Loopback0 prefixes, the BGP peering is declared down. The reason can be seen in the BGP and ICMP debugs below:

```
R4#debug ip bgp
BGP debugging is on for address family: IPv4 UnicastR4#debug ip icmp
ICMP packet debugging is on
!ICMP: time exceeded rcvd from 155.1.146.1
ICMP: time exceeded rcvd from 155.1.146.1
ICMP: time exceeded rcvd from 155.1.146.1BGP: 150.1.5.5 open failed:Connection timed out
; remote host not responding
BGP: 150.1.5.5 Active open failed - tcb is not available, open active delayed 8192ms (35000ms max, 60% jitter)
```

Because the TTL of the EBGP packet is one, time exceeds as the packet transits through R1. Note that R4 and R5 continue to attempt setup of the BGP peering because the connected check is disabled. This design can be desirable in cases where you do not want to reroute the BGP session around network failures. Now let's see the difference if we had changed the TTL to support the multihop peering:

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface Tunnel0
R4(config-if)#no shutdown
R4(config-if)#interface GigabitEthernet1.45
R4(config-if)#no shutdown

!
%CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to up
%LINK-3-UPDOWN: Interface Tunnel0, changed state to up
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.5 (Tunnel0) is up: new adjacency
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.45.5 (GigabitEthernet1.45) is up: new adjacency
%BGP-5-NBR_RESET: Neighbor 150.1.5.5 active reset (BGP Notification sent)
%BGP-5-ADJCHANGE: neighbor 150.1.5.5 Up
%BGP-5-ADJCHANGE: neighbor 155.1.0.2 Up
!R4(config-if)#router bgp 100
R4(config-router)#no neighbor 150.1.5.5 disable-connected-check
 R4(config-router)#neighbor 150.1.5.5 ebgp-multihop
```

https://t.me/learningnets

```
!R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router bgp 200
R5(config-router)#no neighbor 150.1.4.4 disable-connected-check
 R5(config-router)#neighbor 150.1.4.4 ebgp-multihop
```

R4's links to R5 are brought back up, and the `disable-connected-check` command is replaced with the `ebgp-multihop [255]` command. The BGP peering comes up, and when R4's connected links to R5 are brought down again, the BGP peering continues to stay up:

```
R4#configure terrminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface Tunnel0
R4(config-if)#shutdown
%BGP-5-NBR_RESET: Neighbor 155.1.0.2 reset (Interface flap)
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.5 (Tunnel0) is down: interface down
%BGP-5-ADJCHANGE: neighbor 155.1.0.2 Down Interface flap
%BGP_SESSION-5-ADJCHANGE: neighbor 155.1.0.2 IPv4 Unicast topology base removed from session  Interface flap
%CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to down
%LINK-5-CHANGED: Interface Tunnel0, changed state to administratively down
!R4(config-if)#interface GigabitEthernet1.45
R4(config-subif)#shutdown

%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.45.5 (GigabitEthernet1.45) is down: interface down
```

The final result is that the BGP session between R4 and R5 is rerouted and stays up. This may not be wanted in a case where redundant paths are received from multiple neighbors, but only one physical link is used to reach both neighbors after a failure. If `disable-connected-check` is used instead of `ebgp-multihop` in such case, the peering would not stay up after rerouting and redundant paths would not be received.

Considering that the public Internet BGP table can grow to millions of paths and hundreds of thousands of prefixes, the separate views that the router must maintain can require an extremely large amount of memory and CPU resources. Therefore, in this particular design, choosing to disable the connected check instead of enabling a multihop peering can save resources while a failure scenario is in effect.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

Authenticating BGP Peerings

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Authenticating BGP Peerings**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R1 in AS 100 and R3 in AS 300 and establish an EBGP peering over their directly connected link.
- Authenticate this BGP peering with the password **CISCO**.

Configuration

```
R1:
router bgp 100
 neighbor 155.1.13.3 remote-as 300
 neighbor 155.1.13.3 password CISCO

R3:

router bgp 300
 neighbor 155.1.13.1 remote-as 100
 neighbor 155.1.13.1 password CISCO
```

Verification

BGP authentication is implemented through TCP Option 19, the MD5 hash. Configuration is very straightforward and requires only the additional neighbor statement with the password option. If BGP peering occurs, authentication is

https://t.me/learningnets

successful:

```
R3#show ip bgp neighbors 155.1.13.1 | include state|Flags
BGP state = Established
, up for 00:02:02
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Status Flags: passive open, gen tcbs Option Flags: nagle, path mtu capable, md5
, Retrans timeout
```

Authentication failure results in a log message and failure of the peering to establish:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#router bgp 100
R1(config-router)#neighbor 155.1.13.3 password WRONG
R1(config-router)#end
!R1#clear ip bgp *
```

Note log messages on R3 after clearing the BGP session:

```
%TCP-6-BADAUTH: Invalid MD5 digest from 155.1.13.1(179) to 155.1.13.3(27526) tableid - 0
%TCP-6-BADAUTH: Invalid MD5 digest from 155.1.13.1(32159) to 155.1.13.3(179) tableid - 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

iBGP Route Reflection

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **iBGP Route Reflection**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that there is no BGP configuration on R1 - R8 and:
  - Configure BGP on R1 - R8 using AS 100.
  - Configure IBGP peerings from R1 to all other AS 100 routers.
  - Advertise Loopback0 prefixes of all AS 100 devices into BGP.
- Configure EBGP peerings between R7 and R9, between R8 and R10 using their directly connected links:
  - R9 and R10 are in AS 54.
  - Advertise the link between R7 and R9 into EIGRP on R7.
  - Advertise the link between R8 and R10 into EIGRP on R8.
- Ensure full IPv4 reachability as follows:
  - To Loopback0 prefixes from all internal devices.
  - To all prefixes learned from AS 54, from all internal devices when sourcing traffic from Loopback0 interfaces.

Configuration

```
R1:
 router bgp 100
  network 150.1.1.1 mask 255.255.255.255
  neighbor 155.1.0.2 remote-as 100
  neighbor 155.1.0.3 remote-as 100
  neighbor 155.1.0.4 remote-as 100
```

https://t.me/learningnets

```
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.146.6 remote-as 100
 neighbor 155.1.0.2 route-reflector-client
 neighbor 155.1.0.3 route-reflector-client
 neighbor 155.1.0.4 route-reflector-client
 neighbor 155.1.0.5 route-reflector-client
 neighbor 155.1.58.8 route-reflector-client
 neighbor 155.1.67.7 route-reflector-client
 neighbor 155.1.146.6 route-reflector-client
```

R2:

```
router bgp 100
 network 150.1.2.2 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
```

R3:

```
router bgp 100
 network 150.1.3.3 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
```

R4:

```
router bgp 100
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
```

R5:

```
router bgp 100
 network 150.1.5.5 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
```

R6:

```
router bgp 100
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
```

R7:

```
router bgp 100
 network 150.1.7.7 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.79.9 remote-as 54
!
router eigrp 100
 network 155.1.79.0 0.0.0.255
```

https://t.me/learningnets

```
R8:

router bgp 100
 network 150.1.8.8 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.108.10 remote-as 54
!
router eigrp 100
 network 155.1.108.0 0.0.0.255
```

Verification

BGP route reflectors, as defined in RFC 2796, are used in large-scale iBGP deployments to reduce the need for [n*(n-1)/2] fully meshed peerings. Route reflectors accomplish this by creating an exception for passing advertisements between IBGP peers. Specifically, this is implemented as follows.

A route reflector can have three types of peers: EBGP peers, client peers, and non-client peers. EBGP peers are neighbors in a different AS number, including peers in different Sub-ASs in confederation. Client peers are iBGP neighbors that have the `route-reflector-client` statement configured. Non-client peers are normal iBGP peers that do not have the `route-reflector-client` statement configured. Routing advertisements sent from the route reflector must conform to the following three rules:

1. Routes learned from EBGP peers can be sent to other EBGP peers, clients, and non-clients.
2. Routes learned from client peers can be sent to EBGP peers, other client peers, and non-clients.
3. Routes learned from non-client peers can be sent to EBGP peers, and client peers, *but not other non-clients*.

In the simplest of route-reflection designs, a central peering point is chosen for all devices in the iBGP domain, and all peers of this device are defined as clients. In this particular example, R1 is configured in this manner. When R1 receives routes from its iBGP peers, they are tagged internally as being received from a client peer and are candidate to be advertised on to everyone. In the output below, we see R1 learning R2's Loopback0 prefix, with the RR-client attribute set:

```
R1#show ip bgp 150.1.2.2 255.255.255.255

BGP routing table entry for 150.1.2.2/32, version 3
Paths: (1 available, best #1, table default)
```

https://t.me/learningnets

```
  Advertised to update-groups:
     1
  Refresh Epoch 1  Local, (Received from a RR-client
)    155.1.0.2 from 155.1.0.2
 (150.1.2.2)
  Origin IGP, metric 0, localpref 100, valid, internal, best
  rx pathid: 0, tx pathid: 0x0
```

When a route is advertised, or “reflected,” from the route reflector to a client or non-client, BGP attributes such as the next-hop value are not updated:

```
R1#show ip bgp neighbors 155.1.0.3 advertised-routes

BGP table version is 19, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.79.9               0    100      0 54 i
 *>i 28.119.17.0/24   155.1.79.9               0    100      0 54 i
 *>i 112.0.0.0        155.1.79.9               0    100      0 54 50 60 i
 *>i 113.0.0.0        155.1.79.9               0    100      0 54 50 60 i
 *>i 114.0.0.0        155.1.79.9               0    100      0 54 i
 *>i 115.0.0.0        155.1.79.9               0    100      0 54 i
 *>i 116.0.0.0        155.1.79.9               0    100      0 54 i
 *>i 117.0.0.0        155.1.79.9               0    100      0 54 i
 *>i 118.0.0.0        155.1.79.9               0    100      0 54 i
 *>i 119.0.0.0        155.1.79.9               0    100      0 54 i
 *>  150.1.1.1/32     0.0.0.0                  0         32768 i
 *>i 150.1.2.2/32     155.1.0.2                0    100      0 i
 *>i 150.1.3.3/32     155.1.0.3                0    100      0 i
 *>i 150.1.4.4/32     155.1.0.4                0    100      0 i
 *>i 150.1.5.5/32     155.1.0.5                0    100      0 i
 *>i 150.1.6.6/32     155.1.146.6              0    100      0 i
 *>i 150.1.7.7/32     155.1.67.7               0    100      0 i
 *>i 150.1.8.8/32     155.1.58.8               0    100      0 i

Total number of prefixes 18
```

https://t.me/learningnets

Instead, two new attributes are added onto the reflected prefix, the Originator ID and the Cluster List:

```
R3#show ip bgp 150.1.2.2 255.255.255.255
BGP routing table entry for 150.1.2.2/32, version 7
Paths: (1 available, best #1, table default)
  Not advertised to any peer
  Refresh Epoch 1
  Local
    155.1.0.2 from 155.1.0.1 (150.1.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best
Originator: 150.1.2.2, Cluster list: 150.1.1.1


  rx pathid: 0, tx pathid: 0x0
```

Recall that previously, loop prevention in iBGP was achieved simply by not advertising routes learned from one iBGP neighbor to another. Because route-reflection violates this rule, new loop prevention must be implemented.

The first of these, the Originator ID, is set by the route reflector as the BGP router-id of the neighbor from which it learned the prefix. For the above prefix, this is the BGP router-id of R2, as seen in the parenthesis of the **`show ip bgp 150.1.2.2 255.255.255.255`** on R3. When any BGP speaker learns a route from an iBGP neighbor, and the Originator ID matches their own local router-id, the route is discarded. This is why it is essential that the BGP router-id value be unique throughout the entire routing domain, just like in OSPF and EIGRP.

The second new attribute, the Cluster List, contains the Cluster-IDs of the route reflectors that the route transited through in the network. Unless the **`bgp cluster-id`** command is manually configured under the BGP routing process, the value defaults to the router-id of the route reflector. In the above case, the Cluster List contains just the router-id of R1, 150.1.1.1.

This attribute is used to prevent loops *between* route-reflectors, when a hierarchical design called Clustering is implemented. A “cluster” in BGP is defined as a route-reflector and its clients, and will be explored in more detail in later tasks. When a route-reflector learns a route from an iBGP peer, and the Cluster List includes its own Cluster-ID, the route is discarded.

Note that because the other attributes in the prefix are not updated by the route reflector, the reflector is analogous to the DR in OSPF, and in many cases traffic does not physically transit through this device. Instead, the reflector is simply a central point for network control traffic, but not necessarily an aggregation point for traffic. This can be demonstrated in this design by the traffic flow between R2 and

https://t.me/learningnets

R7, as seen below:

```
R2#show ip bgp 150.1.7.7 255.255.255.255
BGP routing table entry for 150.1.7.7/32, version 8
Paths: (1 available, best #1, table default)
  Not advertised to any peer
  Refresh Epoch 1
  Local 155.1.67.7 (metric 3328) from 155.1.0.1 (150.1.1.1)

      Origin IGP, metric 0, localpref 100, valid, internal, best
  Originator: 150.1.7.7, Cluster list: 150.1.1.1
  rx pathid: 0, tx pathid: 0x0
```

R2 learns the prefix 150.1.7.7/32 from R1, but with a next-hop value of 155.1.67.7. Now a recursive lookup must be performed on 155.1.67.7 until the outgoing interface is found:

```
R2#show ip route 155.1.67.7
Routing entry for 155.1.67.0/24
  Known via "eigrp 100", distance 90, metric 3328, type internal
  Redistributing via eigrp 100
  Last update from 155.1.23.3 on GigabitEthernet1.23, 00:28:20 ago
  Routing Descriptor Blocks: * 155.1.23.3, from 155.1.23.3, 00:28:20 ago, via GigabitEthernet1.23
      Route metric is 3328, traffic share count is 1
      Total delay is 30 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
!R2#show ip route 155.1.23.3
Routing entry for 155.1.23.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks: * directly connected, via GigabitEthernet1.23

      Route metric is 0, traffic share count is 1
```

155.1.67.7 is known via IGP from R3, out GigabitEthernet1.23. The traceroute indicates that the traffic flow does not pass through the route reflector, even though the BGP control traffic did:

```
R2#traceroute 150.1.7.7



Type escape sequence to abort.
Tracing the route to 150.1.7.7
```

https://t.me/learningnets

```
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.23.3 3 msec 2 msec 1 msec   2 155.1.37.7 2 msec *  6 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

Large-Scale iBGP Route Reflection with Clusters

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Large Scale iBGP Route Reflection**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that there is no BGP configuration on R1 - R8.
- Advertise the link between R7 and R9 into EIGRP on R7.
- Advertise the link between R8 and R10 into EIGRP on R8.
- Configure BGP in AS 200 on R2 and in AS 100 on all other R1 - R8 routers.
- Configure a BGP cluster between R1, R4, and R6 as follows:
  - R1 should be the route-reflector, and peer with R4 and R6.
  - R4 and R6 should peer with R10 who is in AS 54; R10 is not preconfigured for this peering.
  - Use the cluster-id 150.1.1.1.
- Configure a BGP cluster between R3, R7, and R9 as follows:
  - R9 is preconfigured in AS 100.
  - R3 should be the route-reflector, and peer with R9 and R7.
  - Use the cluster-id 150.1.3.3.
- Configure a BGP cluster between R5 and R8 as follows:
  - R5 should be the route-reflector, and peer with R8.
  - Use the cluster-id 150.1.5.5.
- R1, R3, and R5 should all peer with each other in a full-mesh, but they should not propagate updates between clusters.
- Configure EBGP peerings between R2 and R3, between R2 and R5.
- Advertise the Loopback0 prefixes of R1 - R8 into BGP.
- Ensure full IPv4 reachability to Loopback0 prefixes from all internal devices, and to

https://t.me/learningnets

all prefixes learned from AS 54 from R1-R8 when sourcing traffic from the Loopback0 interfaces.

Configuration

```
R1:
router bgp 100
 bgp cluster-id 150.1.1.1
 network 150.1.1.1 mask 255.255.255.255
 neighbor 155.1.146.4 remote-as 100
 neighbor 155.1.146.6 remote-as 100
 neighbor 155.1.146.4 route-reflector-client
 neighbor 155.1.146.6 route-reflector-client
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.13.3 remote-as 100

R2:
router bgp 200
 network 150.1.2.2 mask 255.255.255.255
 neighbor 155.1.23.3 remote-as 100
 neighbor 155.1.0.5 remote-as 100

R3:
router bgp 100
 bgp cluster-id 150.1.3.3
 network 150.1.3.3 mask 255.255.255.255
 neighbor 155.1.37.7 remote-as 100
 neighbor 155.1.79.9 remote-as 100
 neighbor 155.1.37.7 route-reflector-client
 neighbor 155.1.79.9 route-reflector-client
 neighbor 155.1.13.1 remote-as 100
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.23.2 remote-as 200

R4:
router bgp 100
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.108.10 remote-as 54
 neighbor 155.1.108.10 ebgp-multihop 255

R5:
router bgp 100
 bgp cluster-id 150.1.5.5
```

https://t.me/learningnets

```
 network 150.1.5.5 mask 255.255.255.255
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.58.8 route-reflector-client
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.2 remote-as 200
```

R6:

```
router bgp 100
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.108.10 remote-as 54
 neighbor 155.1.108.10 ebgp-multihop 255
```

R7:

```
router bgp 100
 network 150.1.7.7 mask 255.255.255.255
 neighbor 155.1.37.3 remote-as 100
!
router eigrp 100
 network 155.1.79.0 0.0.0.255
```

R8:

```
router bgp 100
 network 150.1.8.8 mask 255.255.255.255
 neighbor 155.1.58.5 remote-as 100
!
router eigrp 100
 network 155.1.108.0 0.0.0.255
```

R10:

```
router bgp 54
 neighbor 155.1.146.4 remote-as 100
 neighbor 155.1.146.4 ebgp-multihop 255
 neighbor 155.1.146.6 remote-as 100
 neighbor 155.1.146.6 ebgp-multihop 255
```

Verification

The term “cluster” refers to a route-reflector and its clients, or multiple route-reflectors that service the same clients. Clustering is used to create a balance between the amount of BGP control traffic that must be maintained through peering

https://t.me/learningnets

and redundancy. Many large-scale service providers use clustering to create hierarchy in their iBGP designs by constraining clusters to different geographic regions, which reduces the number of long-haul BGP peerings that must occur.

In this particular example, three clusters are created. Each cluster is serviced by one route-reflector: R1, R3, or R5. Inside the cluster, all iBGP peers are configured as clients of the route reflector. Between clusters, however, the route reflectors are non-clients of each other. This design helps to limit redundant updating in the BGP control plane, but it sacrifices redundancy. To illustrate this, let's follow the path of an update through the iBGP domain and observe how different failure scenarios affect reachability to it:

```
R4#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 21
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  54 155.1.108.10 (metric 3328) from 155.1.108.10 (31.3.0.1)

      Origin IGP, localpref 100, valid, external, best
      rx pathid: 0, tx pathid: 0x0
```

R4 learns the prefix 112.0.0.0/8 from its EBGP peer, R10, and passes this route on to its iBGP peer, R1:

```
R1#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 29
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 1
  54, (Received from a RR-client) 155.1.108.10 (metric 3584) from 155.1.146.6 (150.1.6.6)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  54, (Received from a RR-client) 155.1.108.10 (metric 3584) from 155.1.146.4 (150.1.4.4)
      Origin IGP, metric 0, localpref 100, valid, internal, best

  rx pathid: 0, tx pathid: 0x0
```

R1 learns the route from R4 with a next-hop of 155.1.108.10. It also learns the identical route from R6 with a next-hop of 155.1.108.10. Because the router-id of R4

https://t.me/learningnets

is lower, this route is chosen as best and is candidate to be advertised. Also note that R1 says that these two prefixes are learned from route reflector clients.

Because these routes come from client peers, they are candidate to be advertised to EBGP peers, other clients, and non-clients. In this case, R1 advertises the path through R4 to its non-clients, R3 and R5:

```
R3#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 28
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 1
  54 155.1.108.10 (metric 3840) from 155.1.13.1 (150.1.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best
Originator: 150.1.4.4, Cluster list: 150.1.1.1
  rx pathid: 0, tx pathid: 0x0
!R5#show ip bgp 112.0.0.0
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1          3
  Refresh Epoch 1
  54 155.1.108.10 (metric 3072) from 155.1.0.1 (150.1.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best
Originator: 150.1.4.4, Cluster list: 150.1.1.1

  rx pathid: 0, tx pathid: 0x0
```

R3 and R5 learn the prefix from R1, with the new attributes Originator ID set to 150.1.4.4 (R4), and Cluster List including 150.1.1.1 (R1). Because from R3 and R5's perspective these prefixes were not learned from route reflector clients, they are only candidate to be advertised to EBGP peers and client peers. From R3, the result is that the prefix is advertised to R2, R7 and R9, but not to R5:

https://t.me/learningnets

```
R3#show ip bgp neighbors 155.1.23.2 advertised-routes | include 112.0.0.0
*>i 112.0.0.0        155.1.108.10             0    100      0 54 i

!R3#show ip bgp neighbors 155.1.37.7 advertised-routes | include 112.0.0.0
*>i 112.0.0.0        155.1.108.10             0    100      0 54 i

!R3#show ip bgp neighbors 155.1.79.9 advertised-routes | include 112.0.0.0
*>i 112.0.0.0        155.1.108.10             0    100      0 54 i

!R3#show ip bgp neighbors 155.1.0.5 advertised-routes | include 112.0.0.0
R3#
```

This is the behavior we should expect, because the inter-cluster peerings are non-client peerings. In essence, the only routes that are advertised between clusters are routes that came from within the cluster, or from EBGP peers.

Note that this behavior is the same in R5's cluster. R5 will only advertise this route to R2, its EBGP peer, and R8, its route-reflector client:

```
R7#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 28
Paths: (1 available, best #1, table default)
  Not advertised to any peer
  Refresh Epoch 1
  54 155.1.108.10 (metric 3840) from 155.1.37.3 (150.1.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal, best
Originator: 150.1.4.4, Cluster list: 150.1.3.3, 150.1.1.1

      rx pathid: 0, tx pathid: 0x0
```

R7 learns the route from R3, with the Originator ID still set to R4, but with the Cluster List now including both the Cluster-IDs of R3 and R1. Like AS-Path, the Cluster List is populated with the newest route reflector on the left. The other attributes, such as the next-hop value, have not been updated. This means that R7 must perform a recursive lookup toward 155.1.108.10:

```
R7#show ip route 155.1.108.10
Routing entry for 155.1.108.0/24
  Known via "eigrp 100", distance 90, metric 3840, type internal
  Redistributing via eigrp 100
  Last update from 155.1.67.6 on GigabitEthernet1.67, 00:36:40 ago
  Routing Descriptor Blocks:
```

https://t.me/learningnets

```
* 155.1.67.6, from 155.1.67.6, 00:36:40 ago, via GigabitEthernet1.67

      Route metric is 3840, traffic share count is 1
      Total delay is 50 microseconds, minimum bandwidth is 1000000 Kbit
  Reliability 255/255, minimum MTU 1500 bytes
  Loading 1/255, Hops 4
!R7#show ip route 155.1.67.6
Routing entry for 155.1.67.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks: * directly connected, via GigabitEthernet1.67

      Route metric is 0, traffic share count is 1
```

The outgoing interface is found, and a traceroute indicates the full end-to-end path. Note that R1 is not in the traffic path, even though it is in the BGP control traffic path. This is because an independent IGP lookup is performed toward the next-hop, which is unrelated to the path of the BGP peerings:

```
R7#traceroute 112.0.0.1 source loopback 0

R7#traceroute 112.0.0.1 source loopback 0
Type escape sequence to abort.
Tracing the route to 112.0.0.1
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.67.6 28 msec 9 msec 9 msec
  2 155.1.146.4 10 msec 16 msec 24 msec
  3 155.1.45.5 70 msec 42 msec 105 msec
  4 155.1.58.8 35 msec 52 msec 6 msec
  5 155.1.108.10 21 msec 31 msec 147 msec
  6 155.1.109.9 10 msec *  61 msec
```

At this point full IPv4 reachability should be obtained to all BGP learned prefixes when traffic is sourced from Loopback0 interfaces. Now let’s look at the case where a failure occurs on the DMVPN network between R1 and R5:

https://t.me/learningnets

```
R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface Tunnel0
R1(config-if)#no ip nhrp map 155.1.0.5 169.254.100.5
R1(config-if)#do clear ip nhrp
R1(config-if)#do clear ip bgp 155.1.0.5

<output omitted>R1#show ip bgp summary | include 155.1.0.5
155.1.0.5       4   100      73      70        0    0    0 00:10:03 Active
```

The NHRP map statement for R5 is withdrawn from R1, simulating a circuit failure. Shortly after, the BGP peer is declared down. Now look at the changes in the propagation of the prefix 112.0.0.0/8:

```
R1#show ip bgp 112.0.0.0

BGP routing table entry for 112.0.0.0/8, version 29
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 1
  54, (Received from a RR-client) 155.1.108.10 (metric 3584) from 155.1.146.6 (150.1.6.6)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  54, (Received from a RR-client) 155.1.108.10 (metric 3584) from 155.1.146.4 (150.1.4.4)
  Origin IGP, metric 0, localpref 100, valid, internal, best

  rx pathid: 0, tx pathid: 0x0
```

R1 still has the best route to 112.0.0.0/8 installed via R4, and advertises the prefix to R3. The advertisement to R5 cannot occur because the peering is down:

```
R3#show ip bgp 112.0.0.0
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 1
  54 155.1.108.10 (metric 3840) from 155.1.13.1 (150.1.1.1)

      Origin IGP, metric 0, localpref 100, valid, internal, best
      Originator: 150.1.4.4, Cluster list: 150.1.1.1
```

https://t.me/learningnets

```
      rx pathid: 0, tx pathid: 0x0
```

R3 receives the prefix from R1, but it cannot advertise it to R5 because both R1 and R5 are non-clients:

```
R3#show ip bgp neighbors 155.1.0.5 advertised-routes

BGP table version is 35, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  150.1.2.2/32     155.1.23.2               0             0 200 i
 *>  150.1.3.3/32     0.0.0.0                  0         32768 i
 *>i 150.1.7.7/32     155.1.37.7               0    100      0 i

Total number of prefixes 3
```

The final result is that R5 does not have the prefix installed, which implies that R8 likewise does not have the prefix:

```
R5#show ip bgp 112.0.0.0
% Network not in table
!R8#ping 112.0.0.1 source Loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 112.0.0.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.8.8 .....

Success rate is 0 percent (0/5)
```

Now let's modify the peerings between R1, R3, and R5, so that they are clients of each other, and see how this affects redundancy:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#router bgp 100
R1(config-router)#neighbor 155.1.0.5 route-reflector-client
R1(config-router)#neighbor 155.1.13.3 route-reflector-client
!R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R3(config)#router bgp 100
R3(config-router)#neighbor 155.1.0.5 route-reflector-client
R3(config-router)#neighbor 155.1.13.1 route-reflector-client
```

https://t.me/learningnets

```
!R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router bgp 100
R5(config-router)#neighbor 155.1.0.1 route-reflector-client
R5(config-router)#neighbor 155.1.0.3 route-reflector-client
```

Now that R1 is a client of R3, R3 can advertise prefixes received from R1 to R5, and vice versa. The resulting change is that R3 now advertises 112.0.0.0/8 to R5:

```
R3#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 81
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 2  54, (Received from a RR-client
) 155.1.108.10 (metric 3840) from 155.1.13.1 (150.1.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best      Originator: 150.1.4.4,
Cluster list: 150.1.1.1
      rx pathid: 0, tx pathid: 0x0
!R3#show ip bgp neighbors 155.1.0.5 advertised-routes | include 112.0.0.0
*>i 112.0.0.0        155.1.108.10              0     100       0 54 i
```

Likewise, R5 can now continue to propagate the prefix onto R8, and connectivity is restored:

https://t.me/learningnets

```
R8#show ip bgp 112.0.0.0

BGP routing table entry for 112.0.0.0/8, version 83
Paths: (1 available, best #1, table default)
  Not advertised to any peer
  Refresh Epoch 1
  54     155.1.108.10 from 155.1.58.5
 (150.1.5.5)
      Origin IGP, metric 0, localpref 100, valid, internal, best       Originator: 150.1.4.4,
Cluster list: 150.1.5.5, 150.1.3.3, 150.1.1.1

      rx pathid: 0, tx pathid: 0x0
!R8#ping 112.0.0.1 source Loopback0



Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 112.0.0.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.8.8 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/11/17 ms
```

So if, without client peerings between R1, R3, and R5, redundancy suffers, what is the disadvantage of configuring all the inter-cluster peers as clients of each other? The answer is route replication overhead.

With R5's connection to R1 working, let's look at the advertisement of 112.0.0.0/8 again. To start, R4 learns the prefix 112.0.0.0/8 from R10, and advertises it onto R1:

```
R4#show ip bgp neighbors 155.1.146.1 advertised-routes | include 112.0.0.0
 *> 112.0.0.0         155.1.108.10                          0 54 i
```

R1 reflects this route to both R3 and R5, as expected:

```
R1#show ip bgp neighbors 155.1.13.3 advertised-routes | include 112.0.0.0
 *> 112.0.0.0         155.1.108.10                          0 54 i
!R1#show ip bgp neighbors 155.1.0.5 advertised-routes | include 112.0.0.0
 *> 112.0.0.0         155.1.108.10                          0 54 i
```

https://t.me/learningnets

Because R1 is now a client of both R3 and R5, both R3 and R5 advertise the prefix to each other, as expected:

```
R3#show ip bgp neighbors 155.1.0.5 advertised-routes | include 112.0.0.0
 *> 112.0.0.0         155.1.108.10                           0 54 i
!R5#show ip bgp neighbors 155.1.0.3 advertised-routes | include 112.0.0.0
 *> 112.0.0.0         155.1.108.10                           0 54 i
```

Now, however, R3 and R5 each take the advertisement they are getting in from R1, and send it back to R1. This is where the advertisement feedback occurs:

```
R3#show ip bgp neighbors 155.1.13.1 advertised-routes | include 112.0.0.0
 *> 112.0.0.0         155.1.108.10                           0 54 i
!R5#show ip bgp neighbors 155.1.0.1 advertised-routes | include 112.0.0.0
 *> 112.0.0.0         155.1.108.10                           0 54 i
```

To see this update loop in action, limit the routes that R4 receives from R10 to just 112.0.0.0/8, and shut down R1's peering to R6. Next, request a route refresh with the `clear ip bgp` command while `debug ip bgp` is enabled:

https://t.me/learningnets

```
R4#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#ip prefix-list ONLY_112 permit 112.0.0.0/8
R4(config)#router bgp 100
R4(config-router)#neighbor 155.1.108.10 prefix-list ONLY_112 in
R4#clear ip bgp * in

!R1#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#router bgp 100
R1(config-router)#neighbor 155.1.146.6 shutdown

%BGP-5-ADJCHANGE: neighbor 155.1.146.6 Down Admin. shutdown
!R1#debug ip bgp updates

BGP updates debugging is on for address family: IPv4 UnicastR1#clear ip bgp 155.1.146.4

!
BGP(0): no valid path for 112.0.0.0/8 BGP(0): no valid path for 150.1.4.4/32
_
%BGP-5-ADJCHANGE: neighbor 155.1.146.4 Down User reset
%BGP_SESSION-5-ADJCHANGE: neighbor 155.1.146.4 IPv4 Unicast topology base removed from session  User reset
BGP: topo global:IPv4 Unicast:base Remove_fwdroute for 112.0.0.0/8
BGP: topo global:IPv4 Unicast:base Remove_fwdroute for 150.1.4.4/32
BGP(0): (base) 155.1.0.5 send unreachable (format) 112.0.0.0/8
BGP(0): (base) 155.1.0.5 send unreachable (format) 150.1.4.4/32
```

With R1's peering to R4 down, an UPDATE message is sent to withdraw the routes that were reachable via R4:

```
BGP: 155.1.0.5 Route Reflector cluster loop
; Received cluster-id 150.1.1.1
BGP(0): 155.1.0.5 rcv UPDATE w/ attr: nexthop 155.1.108.10, origin i, localpref 100, metric 0, originator 150.1.4.4,
BGPSSA ssacount is 0 BGP(0): 155.1.0.5 rcv UPDATE about 112.0.0.0/8 --
DENIED due to: CLUSTERLIST contains our own cluster ID
;
BGP: 155.1.0.5 Route Reflector cluster loop; Received cluster-id 150.1.1.1
BGP(0): 155.1.0.5 rcv UPDATE w/ attr: nexthop 155.1.146.4, o
R1#rigin i, localpref 100, metric 0, originator 150.1.4.4, clusterlist 150.1.5.5 150.1.3.3 150.1.1.1, merged path ,
BGPSSA ssacount is 0 BGP(0): 155.1.0.5 rcv UPDATE about 150.1.4.4/32 --
DENIED due to: CLUSTERLIST contains our own cluster ID
;
```

https://t.me/learningnets

```
BGP(0): 155.1.13.3 rcv UPDATE about 112.0.0.0/8 -- withdrawn
BGP(0): 155.1.13.3 rcv UPDATE about 150.1.4.4/32 -- withdrawn
BGP(0): 155.1.0.5 rcv UPDATE about 112.0.0.0/8 -- withdrawn
BGP(0): 155.1.0.5 rcv UPDATE about 150.1.4.4/32 -- withdrawn
```

When R1 issued a withdraw message for 112.0.0.0/8, it received looped updates back in from both R3 and R5. Ultimately these were blocked because R1 saw its own Cluster ID in the Cluster List. The same occurs when the peering to R4 comes back up:

```
%BGP-5-ADJCHANGE: neighbor 155.1.146.4 Up
BGP(0): (base) 155.1.0.5 send UPDATE (format) 150.1.2.2/32, next 155.1.0.2, metric 0, path 200
BGP(0): (base) 155.1.0.5 send UPDATE (format) 150.1.3.3/32, next 155.1.13.3, metric 0, path Local
BGP(0): (base) 155.1.0.5 send UPDATE (format) 150.1.7.7/32, next 155.1.37.7, metric 0, path Local
BGP(0): (base) 155.1.0.5 send UPDATE (format) 150.1.5.5/32, next 155.1.0.5, metric 0, path Local
BGP(0): (base) 155.1.0.5 send UPDATE (format) 150.1.8.8/32, next 155.1.58.8, metric 0, path Local
BGP(0): (base) 155.1.0.5 send UPDATE (format) 150.1.1.1/32, next 155.1.0.1, metric 0, path Local
BGP(0): 155.1.146.4
 rcvd UPDATE w/ attr: nexthop 155.1.108.10, origin i, localpref 100, metric 0, merged path 54, AS_PATH
BGP(0): 155.1.146.4 rcvd 112.0.0.0/8
BGP(0): 155.1.146.4
 rcvd UPDATE w/ attr: nexthop 155.1.146.4, origin i, localpref 100, metric 0 BGP(0): 155.1.146.4 rcvd
150.1.4.4/32
BGP(0): updgrp 2 - 155.1.146.4 updates replicated for neighbors: 155.1.0.5 155.1.13.3
```

R1 gets the routes 150.1.4.4/32 and 112.0.0.0/8 from R4, and replicates them to R3 and R5.

```
BGP: 155.1.13.3 Route Reflector cluster loop
; Received cluster-id 150.1.1.1
BGP(0): 155.1.13.3 rcv UPDATE w/ attr: nexthop 155.1.146.4, origin i, localpref 100, metric 0, originator 150.1.4.4,
BGP(0): 155.1.13.3 rcv UPDATE about 150.1.4.4/32 --
DENIED due to: CLUSTERLIST contains our own cluster ID
; BGP: 155.1.13.3 Route Reflector cluster loop
; Received cluster-id 150.1.1.1
BGP(0): 155.1.13.3 rcv UPDATE w/ attr: nexthop 155.1.108.10, origin i, localpref 100, metric 0, originator 150.1.4.4
BGP(0): 155.1.13.3 rcv UPDATE about 112.0.0.0/8 --
DENIED due to: CLUSTERLIST contains our own cluster ID
; BGP: 155.1.0.5 Route Reflector cluster loop
; Received cluster-id 150.1.1.1
BGP(0): 155.1.0.5 rcv UPDATE w/ attr: nexthop 155.1.146.4, origin i, localpref 100, metric 0, originator 150.1.4.4,
BGP(0): 155.1.0.5 rcv UPDATE about 150.1.4.4/32 --
DENIED due to: CLUSTERLIST contains our own cluster ID
;
```

https://t.me/learningnets

```
BGP: 155.1.0.5 Route Reflector cluster loop
; Received cluster-id 150.1.1.1
BGP(0): 155.1.0.5 rcv UPDATE w/ attr: nexthop 155.1.108.10, origin i, localpref 100, metric 0, originator 150.1.4.4,
BGP(0): 155.1.0.5 rcv UPDATE about 112.0.0.0/8 --DENIED due to: CLUSTERLIST contains our own cluster ID
;
```

When the peering to R4 comes back up, we can also see that another update loop occurs between R1, R3, and R5. Luckily, because the Cluster List contains R1's Cluster ID, the loop is broken.

Ultimately, the choice between making the inter-cluster peerings client or non-client depends on the redundancy design. We saw first that with them configured as non-client peerings, certain network failures could have caused traffic black holes, even though there were alternate viable paths to the destinations.

With the inter-cluster peerings configured as client peerings, each update message sent out results in a feedback loop of update messages received back in. With only a few prefixes in the BGP table, this may not seem like a big issue, but with the hundreds of thousands of prefixes in the Internet BGP table, these type of loops can quickly cause utilization problems.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

iBGP Confederation

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Confederation**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that there is no BGP configuration on R1-R9.
- Advertise the link between R8 and R10 into EIGRP on R8.
- Configure R9 to be part of the EIGRP 100 domain with R7.
- Configure a BGP Confederation Sub-AS between R1, R4, and R6 as follows:
  - Use the Sub-AS number 65146.
  - Use the Public AS number 100.
  - Configure full-mesh peerings between R1, R4, and R6.
  - R4 and R6 should EBGP peer with R10 who is in AS 54; R10 is already preconfigured for these peerings.
- Configure a BGP Confederation Sub-AS between R3, R7, and R9 as follows:
  - Use the Sub-AS number 65379.
  - Use the Public AS number 100.
  - Configure full-mesh peerings between R3, R7, and R9.
- Configure a BGP Confederation Sub-AS between R5 and R8 as follows:
  - Use the Sub-AS number 65508.
  - Use the Public AS number 100.
  - R5 should be a route-reflector and peer with R8.
- R1, R3, and R5 should all peer with each other in a full-mesh.
- Configure EBGP peerings between R2 and R3, between R2 and R5; R2 is in AS 200.
- Advertise the Loopback0 prefixes of R1-R9 into BGP.
- Ensure full IPv4 reachability to Loopback0 prefixes from all internal devices, and to

https://t.me/learningnets

all prefixes learned from AS 54 from all internal devices when sourcing traffic from the Loopback0 interfaces.

Configuration

```
R1:
router bgp 65146
 bgp confederation identifier 100
 bgp confederation peers 65379 65508
 network 150.1.1.1 mask 255.255.255.255
 neighbor 155.1.0.5 remote-as 65508
 neighbor 155.1.13.3 remote-as 65379
 neighbor 155.1.146.4 remote-as 65146
 neighbor 155.1.146.6 remote-as 65146

R2:
router bgp 200
 network 150.1.2.2 mask 255.255.255.255
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.23.3 remote-as 100

R3:
router bgp 65379
 bgp confederation identifier 100
 bgp confederation peers 65146 65508
 network 150.1.3.3 mask 255.255.255.255
 neighbor 155.1.0.5 remote-as 65508
 neighbor 155.1.13.1 remote-as 65146
 neighbor 155.1.23.2 remote-as 200
 neighbor 155.1.37.7 remote-as 65379
 neighbor 155.1.79.9 remote-as 65379

R4:
router bgp 65146
 bgp confederation identifier 100
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 65146
 neighbor 155.1.146.6 remote-as 65146
 neighbor 155.1.108.10 remote-as 54
 neighbor 155.1.108.10 ebgp-multihop 255

R5:
router bgp 65508
 bgp confederation identifier 100
```

https://t.me/learningnets

```
 bgp confederation peers 65146 65379
 network 150.1.5.5 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 65146
 neighbor 155.1.0.2 remote-as 200
 neighbor 155.1.0.3 remote-as 65379
 neighbor 155.1.58.8 remote-as 65508
 neighbor 155.1.58.8 route-reflector-client
```

R6:

```
router bgp 65146
 bgp confederation identifier 100
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.108.10 remote-as 54
 neighbor 155.1.108.10 ebgp-multihop 255
 neighbor 155.1.146.1 remote-as 65146
 neighbor 155.1.146.4 remote-as 65146
```

R7:

```
router bgp 65379
 bgp confederation identifier 100
 network 150.1.7.7 mask 255.255.255.255
 neighbor 155.1.37.3 remote-as 65379
 neighbor 155.1.79.9 remote-as 65379
!
router eigrp 100
 network 155.1.79.0 0.0.0.255
```

R8:

```
router bgp 65508
 bgp confederation identifier 100
 network 150.1.8.8 mask 255.255.255.255
 neighbor 155.1.58.5 remote-as 65508
!
router eigrp 100
 network 155.1.108.0 0.0.0.255
```

R9:

```
router bgp 65379
 bgp confederation identifier 100
 network 150.1.9.9 mask 255.255.255.255
 neighbor 155.1.37.3 remote-as 65379
 neighbor 155.1.79.7 remote-as 65379
!
router eigrp 100
```

https://t.me/learningnets

```
network 155.1.79.0 0.0.0.255
```

Verification

Defined in RFC 5065, *Autonomous System Confederations for BGP*, confederations, like route reflectors, are used to reduce the need for fully meshed iBGP peerings in large-scale deployments. In confederation, a public AS is split into smaller Sub Autonomous Systems (Sub-ASs), which exhibit a hybrid behavior of both iBGP and EBGP. Inside a Sub-AS, the requirement for either fully meshed iBGP peerings or route reflection still applies, but between Sub-ASs, EBGP advertisement rules apply.

First, the BGP process is initialized using the Sub-AS number, as opposed to the normal initialization with the public AS number. Sub-AS numbers are typically in the private AS range (64512 – 65535), but technically can be any valid number, private or not. Next, the **`bgp confederation identifier`** informs the router that it is part of a confederation, with the ID number being its public AS number.

Any neighbor whose remote-as matches either the local Sub-AS or a number listed in the **`bgp confederation peers`** statement is considered to be part of the confederation. In the latter case, these peers are considered “confederation EBGP peers.” Neighbors whose AS matches neither the local Sub-AS nor a confederation peer AS are considered normal EBGP neighbors.

The most notable difference between confederation implementations and route reflection or full mesh is the introduction of a new BGP attribute known as the AS_CONFED_SET. The confederation set, or simply *confed set*, is an unordered list of Sub-ASs that is prepended onto the normal AS-Path of a BGP prefix as it is passed between Sub-ASs. The confed set, however, is stripped and replaced by the confederation identifier when a prefix is advertised to a true EBGP peer. Take the following output from this example:

https://t.me/learningnets

```
R1#show ip bgp 150.1.6.6

BGP routing table entry for 150.1.6.6/32, version 7
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  Local     155.1.146.6 from 155.1.146.6
 (150.1.6.6)       Origin IGP, metric 0, localpref 100, valid, confed-internal
, best
      rx pathid: 0, tx pathid: 0x0
```

R1 learns the prefix 150.1.6.6 from R6 with a next-hop value of 155.1.146.6. Because both R1 and R6 are in the same Sub-AS of 65146, R1 tags this route as *confed-internal*, that is, coming from an iBGP peer. The AS-Path attribute of this prefix is not modified during R6's advertisement to R1, because they are iBGP peers. When R1 passes this route onto R3 or R5, who are both in different Sub-ASs, the confed set is populated with R1's Sub-AS number of 65146. This can be clearly seen as a separate denotation from the normal AS-Path information, as it is listed in parentheses:

```
R5#show ip bgp 150.1.6.6
BGP routing table entry for 150.1.6.6/32, version 9
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          2          3
  Refresh Epoch 1     ( 65379 65146
) 155.1.146.6
 (metric 3072) from 155.1.0.3 (150.1.3.3)      Origin IGP, metric 0, localpref 100, valid,
confed-external
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1 (65146)
 155.1.146.6
 (metric 3072) from 155.1.0.1 (150.1.1.1)      Origin IGP, metric 0, localpref 100, valid,
confed-external, best

  rx pathid: 0, tx pathid: 0x0
```

In the above case, R5 learns the prefix 150.1.6.6 from both R1 and R3. Because both of these neighbors are in different Sub-ASs, they are considered *confed-external* peers, or confederation EBGP peers. Note that the path through R3 contains both R1's Sub-AS and R3's Sub-AS, whereas the path through R1 only

https://t.me/learningnets

contains R1's Sub-AS.

Note that although prefixes are passed between Sub-ASs based on EBGP advertisement rules, the majority of attributes are left unchanged, with the most notable being that *the next-hop value is not modified*. As we can see in the output below, R9 sees the routes learned from AS 54 with a next-hop value of 155.1.108.10, which is the unmodified next-hop of the link between R8 and R10. Likewise, prefixes such as 150.1.8.8 and 150.1.5.5 have next-hop values of the originators into the public AS, not the neighbor from which R9 is learning them:

```
R9#show ip bgp
BGP table version is 12, local router ID is 150.1.9.9
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
   Network Next Hop
       Metric LocPrf Weight Path 28.119.16.0/24 155.1.108.10
            0    100        0 (65146) 54 i 28.119.17.0/24 155.1.108.10
            0    100        0 (65146) 54 i 150.1.1.1/32 155.1.13.1
             0    100       0 (65146) i 150.1.2.2/32 155.1.23.2
             0    100       0 200 i 150.1.3.3/32 155.1.37.3
             0    100       0 i 150.1.4.4/32 155.1.146.4
             0    100       0 (65146) i 150.1.5.5/32 155.1.0.5
             0    100       0 (65146 65508) i 150.1.6.6/32 155.1.146.6
             0    100       0 (65146) i 150.1.7.7/32 155.1.79.7
             0    100       0 i 150.1.8.8/32 155.1.58.8
             0    100       0 (65146 65508) i 150.1.9.9/32 0.0.0.0
            0    100    32768 i
```

When prefixes are advertised outside of the public AS, the confed set is stripped and replaced with the public AS number. In this manner, devices outside of the confederation do not know the confederation's internal routing topology:

```
R2#show ip bgp

BGP table version is 12, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
```

https://t.me/learningnets

```
 *   28.119.16.0/24   155.1.0.5                          0 100 54 i
 *>                   155.1.23.3                         0 100 54 i
 *   28.119.17.0/24   155.1.0.5                          0 100 54 i
 *>                   155.1.23.3                         0 100 54 i
 *   150.1.1.1/32     155.1.0.1                          0 100 i
 *>                   155.1.23.3                         0 100 i
 *>  150.1.2.2/32     0.0.0.0                  0         32768
 *   150.1.3.3/32     155.1.0.3                          0 100 i
 *>                   155.1.23.3               0         0 100 i
 *   150.1.4.4/32     155.1.0.5                          0 100 i
 *>                   155.1.23.3                         0 100 i
 *   150.1.5.5/32     155.1.23.3                         0 100 i
 *>                   155.1.0.5                0         0 100 i
 *   150.1.6.6/32     155.1.0.5                          0 100 i
 *>                   155.1.23.3                         0 100 i
 *   150.1.7.7/32     155.1.0.5                          0 100 i
 *>                   155.1.23.3                         0 100 i
 *   150.1.8.8/32     155.1.23.3                         0 100 i
 *>                   155.1.0.5                          0 100 i
 *   150.1.9.9/32     155.1.0.5                          0 100 i
 *>                   155.1.23.3                         0 100 i
```

From a bestpath selection point of view, the entire AS_CONFED_SET counts as only one AS. This can sometimes result in confusing path selections, such as R5's route to 150.1.7.0, shown below:

```
R5#show ip bgp 150.1.7.7
BGP routing table entry for 150.1.7.7/32, version 10
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     1          2          3
  Refresh Epoch 1  (65146 65379
)    155.1.37.7 (metric 3584) from 155.1.0.1 (150.1.1.1
)      Origin IGP, metric 0, localpref 100, valid, confed-external,best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1  (65379
)    155.1.37.7 (metric 3584) from 155.1.0.3 (150.1.3.3
)
      Origin IGP, metric 0, localpref 100, valid, confed-external
  rx pathid: 0, tx pathid: 0
```

Although R5's path through R3 has a shorter AS path, that is, only Sub-AS 65379 as opposed to both Sub-ASs 65146 and 65379, both of these confed sets are

https://t.me/learningnets

considered equal. In the case of this selection in particular, the bestpath is chosen as R1 because it has a lower router-id. Intra-Confederation bestpath selection is covered in later tasks, because there are some important exceptions such as this that must be noted.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Next-Hop Processing - Next-Hop-Self

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Next Hop Processing Next Hop Self**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that there is no BGP configuration on R1 - R8.
- Configure R2 in BGP AS 200.
- Configure BGP on all other R1 - R8 routers using AS 100.
- Configure iBGP peerings from R1 to all other devices in AS 100.
- Configure EBGP peerings between R7 and R9, between R8 and R10 using their directly connected links.
  - R9 and R10 are in AS 54 and are preconfigured to peer with R7 and R8.
- Configure EBGP peerings between R2 and R3, between R2 and R5.
- Advertise Loopback0 prefixes of R1 - R8 into BGP.
- Do not advertise the links between R7 and R9, between R8 and R10 into EIGRP.
- Use the `next-hop-self` command where necessary to ensure full IPv4 connectivity to the prefixes coming from AS 54.
- Ensure full reachability to Loopback0 prefixes from R1 - R8, and to all prefixes learned from AS 54 from R1 - R8 when sourcing traffic from the Loopback0 interfaces.

Configuration

```
R1:
router bgp 100
 network 150.1.1.1 mask 255.255.255.255
 neighbor 155.1.0.3 remote-as 100
```

https://t.me/learningnets

```
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.146.4 remote-as 100
 neighbor 155.1.146.6 remote-as 100
 neighbor 155.1.0.3 route-reflector-client
 neighbor 155.1.0.5 route-reflector-client
 neighbor 155.1.58.8 route-reflector-client
 neighbor 155.1.67.7 route-reflector-client
 neighbor 155.1.146.4 route-reflector-client
 neighbor 155.1.146.6 route-reflector-client
```

R2:

```
router bgp 200
 network 150.1.2.2 mask 255.255.255.255
 neighbor 155.1.23.3 remote-as 100
 neighbor 155.1.0.5 remote-as 100
```

R3:

```
router bgp 100
 network 150.1.3.3 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.23.2 remote-as 200
```

R4:

```
router bgp 100
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
```

R5:

```
router bgp 100
 network 150.1.5.5 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.2 remote-as 200
```

R6:

```
router bgp 100
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
```

R7:

```
router bgp 100
 network 150.1.7.7 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.146.1 next-hop-self
 neighbor 155.1.79.9 remote-as 54
```

https://t.me/learningnets

```
R8:

router bgp 100
 network 150.1.8.8 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.1 next-hop-self
 neighbor 155.1.108.10 remote-as 54
```

Verification

Recall from the *IP Routing* section how the route recursion process works. When the longest match route is found for the destination in question, the next-hop value of the prefix is checked. If the longest match to the next-hop value is a connected route, the outgoing interface is known, the Layer 2 address is found (depending on the media), and the frame is built for transmission. If the next-hop value is not via a connected interface, additional routing lookups (“recursive” lookups) must be performed until an outgoing interface is found.

With IGP routing, this process is usually transparent, because in the vast majority of cases routes are always learned from directly connected neighbors. For example, if an OSPF route is learned from neighbor A via interface X, it is safe to assume that interface X will be used to reach that destination. However, with BGP, a disconnect can occur between the neighbor from which prefixes are learned (the control plane) and the actual path that packets take toward the prefix (the forwarding/data plane). The main reason for this is that in many cases, BGP neighbors are not directly connected, but instead exchange BGP control plane information over additional hops in the network. This process can occur because IGP information provides reachability to establish the TCP transport inherent to the BGP control plane.

To ensure that this disconnect does not adversely affect the actual forwarding of traffic, the BGP process internally performs the route recursion process for all prefixes toward performing Bestpath selection. If route recursion is not successful (such as if the final outgoing interface cannot be found), the prefix cannot be considered for Bestpath selection. This implies that the prefix cannot be installed in the IP routing table, nor can it be advertised to any other BGP peers. In this particular example, this problem can be illustrated in AS 100 by the routes that are learned from AS 54 after disabling next-hop-self on R7:

```
R7(config)#router bgp 100
R7(config-router)#no neighbor 155.1.146.1 next-hop-self
!R1#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 31
Paths: (2 available, best #2, table default)
```

https://t.me/learningnets

```
  Advertised to update-groups:
     1
  Refresh Epoch 1
  54 50 60, (Received from a RR-client) 155.1.79.9 (inaccessible
) from 155.1.67.7 (150.1.7.7)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  54 50 60, (Received from a RR-client) 155.1.58.8
 (metric 3328) from 155.1.58.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
```

R7 and R8 both learn the prefix 112.0.0.0/8 from their EBGP peers in AS 54 and pass the route on to R1. Because the route is being advertised to an iBGP neighbor, the next-hop value is not normally updated. Recall that the next-hop value is only updated by default when prefixes are advertised to *true* EBGP peers, not iBGP peers or confederation EBGP peers. However, because the next-hop value is just another attribute of the prefix, like AS-Path or Community, it can be arbitrarily changed as the prefix is advertised or received.

In the above output we can see that on R1 the prefix learned from R8 has a next-hop value of 155.1.58.8, whereas the prefix from R7 has a next-hop value of 155.1.79.9. Normally, R8 would be reporting the next-hop value of 155.1.108.10 to R1, which is the next-hop it learned from R10, but the **`neighbor 155.1.146.1 next-hop-self`** command has been applied under R8's BGP process. This means that when a prefix is learned from an EBGP neighbor, and then advertised to the neighbor 155.1.146.1, the next-hop value is set to whatever local address is used for the peering toward 155.1.146.1. Because R7 does not have this option applied, the default next-hop that R9 reports (155.1.79.9) is retained as the next-hop attribute.

The final result of this is that R1 cannot use the route via R7 for Bestpath selection because the next-hop value is listed as "inaccessible." Inaccessible simply means that R1 does not have a route to the next-hop, which implies that successful route recursion cannot occur. Note that the links between R7 & R9 and R8 & R10 are not advertised into IGP. This is further reinforced by the output below:

```
R1#show ip route 155.1.79.9
% Subnet not in table
```

Essentially, there are only two solutions for the problem presented. Either R1 must learn a route to the next-hop 155.1.79.9 via either static or dynamic routing, or the

https://t.me/learningnets

next-hop attribute must be changed to something R1 already has a route to. By configuring the `next-hop-self` option on R7 in addition to R8, the latter solution is used:

https://t.me/learningnets

```
R7(config)#router bgp 100
R7(config-router)#neighbor 155.1.146.1 next-hop-self

!R1#show ip bgp

BGP table version is 40, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found


    Network          Next Hop            Metric LocPrf Weight Path  * i 28.119.16.0/24 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i  * i 28.119.17.0/24 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i  * i 112.0.0.0 155.1.58.8
               0    100      0 54 50 60 i  *>i 155.1.67.7
               0    100      0 54 50 60 i  * i 113.0.0.0 155.1.58.8
               0    100      0 54 50 60 i  *>i 155.1.67.7
               0    100      0 54 50 60 i  * i 114.0.0.0 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i  * i 115.0.0.0 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i  * i 116.0.0.0 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i  * i 117.0.0.0 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i  * i 118.0.0.0 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i  * i 119.0.0.0 155.1.58.8
               0    100      0 54 i  *>i 155.1.67.7
               0    100      0 54 i
 *>  150.1.1.1/32     0.0.0.0                  0         32768 i
 *>i 150.1.2.2/32     155.1.0.2                0    100      0 200 i
 * i                  155.1.23.2               0    100      0 200 i
 *>i 150.1.3.3/32     155.1.0.3                0    100      0 i
 *>i 150.1.4.4/32     155.1.146.4              0    100      0 i
 *>i 150.1.5.5/32     155.1.0.5                0    100      0 i
 *>i 150.1.6.6/32     155.1.146.6              0    100      0 i
 *>i 150.1.7.7/32     155.1.67.7               0    100      0 i
```

https://t.me/learningnets

```
 *>i 150.1.8.8/32     155.1.58.8               0    100      0 i
```

When R7 updates the next-hop value toward R1, R1 can use both prefixes via R7 and R9 for Bestpath selection. According to the output below, we can see that the prefix via R7 wins because of the lower metric to next-hop (3072 vs. 3328):

```
R1#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 41
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 2
  54 50 60, (Received from a RR-client)    155.1.67.7 (metric 3072)
 from 155.1.67.7 (150.1.7.7)      Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  54 50 60, (Received from a RR-client)    155.1.58.8 (metric 3328)
 from 155.1.58.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Next-Hop Processing - Manual Modification

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Next Hop Processing Manual Modification**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that there is no BGP configuration on R1-R8.
- Configure R2 in BGP AS 200.
- Configure BGP on R1 - R8 using AS 100.
- Configure iBGP peerings from R1 to all other devices in AS 100.
- Configure EBGP peerings between R7 and R9, between R8 and R10 using their directly connected links; R9 and R10 are in AS 54 and are preconfigured to peer with R7 and R8.
- Configure EBGP peerings between R2 and R3, between R2 and R5.
- Advertise Loopback0 prefixes of R1 - R8 into BGP.
- Do not advertise the links between R7 and R9, between R8 and R10 into EIGRP.
- Configure an outbound route-map on R7 and an inbound route-map on R1 to resolve any next-hop reachability issues for the routes learned from AS 54.
- Ensure full IPv4 reachability to Loopback0 prefixes from all internal devices, and to all prefixes learned from AS 54 from R1 - R8 when sourcing traffic from Loopback0 interfaces.

Configuration

R1:

```
route-map SET_NEXT_HOP_FROM_R8 permit 10
 set ip next-hop 155.1.58.8
!
```

https://t.me/learningnets

```
router bgp 100
 network 150.1.1.1 mask 255.255.255.255
 neighbor 155.1.0.3 remote-as 100
 neighbor 155.1.0.5 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.58.8 route-map SET_NEXT_HOP_FROM_R8 in
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.146.4 remote-as 100
 neighbor 155.1.146.6 remote-as 100
 neighbor 155.1.0.3 route-reflector-client
 neighbor 155.1.0.5 route-reflector-client
 neighbor 155.1.58.8 route-reflector-client
 neighbor 155.1.67.7 route-reflector-client
 neighbor 155.1.146.4 route-reflector-client
 neighbor 155.1.146.6 route-reflector-client
```

R2:

```
router bgp 200
 network 150.1.2.2 mask 255.255.255.255
 neighbor 155.1.23.3 remote-as 100
 neighbor 155.1.0.5 remote-as 100
```

R3:

```
router bgp 100
 network 150.1.3.3 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.23.2 remote-as 200
```

R4:

```
router bgp 100
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
```

R5:

```
router bgp 100
 network 150.1.5.5 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.2 remote-as 200
```

R6:

```
router bgp 100
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
```

R7:

https://t.me/learningnets

```
route-map SET_NEXT_HOP_TO_R1 permit 10
 set ip next-hop 155.1.67.7
!
router bgp 100
 network 150.1.7.7 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.146.1 route-map SET_NEXT_HOP_TO_R1 out
 neighbor 155.1.79.9 remote-as 54
```

R8:

```
router bgp 100
 network 150.1.8.8 mask 255.255.255.255
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.108.10 remote-as 54
```

Verification

Just as in the previous task, R7 and R8 learn prefixes from AS 54 and advertise them to their iBGP neighbor, R1. Normally, the next-hop value is not updated in this type of advertisement, which results in R1 seeing R7's prefix via R9's next-hop, and R8's prefix via R10's next-hop. If the next-hops are not changed, R1 would not have a route to either 155.1.79.9 or 155.1.108.10. Route recursion would fail, and the prefixes would not be considered for Bestpath selection. This can be seen in the following output, after removing the route-maps that are manually changing the next-hop. Note the lack of the ">" character in the status code on the left side, which normally denotes the Bestpath:

```
R1(config)#router bgp 100
R1(config-router)#no neighbor 155.1.58.8 route-map SET_NEXT_HOP_FROM_R8 in
R1(config-router)#do clear ip bgp 155.1.58.8 in
!R7(config)#router bgp 100
R7(config-router)#no neighbor 155.1.146.1 route-map SET_NEXT_HOP_TO_R1 out
R7(config-router)#do clear ip bgp 155.1.146.1 out
!R1#show ip bgp

BGP table version is 50, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*i 150.1.10.16/24   155.1.108.10             0    100      0 54 i
*i                  155.1.79.9               0    100      0 54 i
```

https://t.me/learningnets

```
<output omitted>
```

Just as in the previous case, there are two ways to resolve this problem. We must either give R1 a route toward 155.1.79.9 and 155.1.108.10, that is, through IGP or static routing, or change the next-hop to something R1 already has a route to. In the previous example, the latter solution was implemented using the **`next-hop-self`** command on R7 and R8 out toward R1. However, because the next-hop value is treated just like any other attribute of the prefix that can be modified, it is also possible to manually change the next-hop value of BGP prefixes arbitrarily by using a route-map.

In the output below, R7 applies an outbound route-map toward R1, which sets the next-hop value to 155.1.67.7. Also note the warning message, “Next hop address is our address”. This warning is meant to be used in policy routing implementations, to make sure that you are not trying to route a packet back to yourself. In our case, we are using the route-map for a BGP attribute change, so this warning can be ignored. Finally, note that the **`clear ip bgp`** command is used to send a triggered update from R7 to R1 via route refresh. This is necessary any time a manual attribute change is implemented in BGP:

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R7(config)#route-map SET_NEXT_HOP_TO_R1 permit 10
R7(config-route-map)#set ip next-hop 155.1.67.7
% Warning: Next hop address is our addressR7(config-route-map)#router bgp 100
R7(config-router)#neighbor 155.1.146.1 route-map SET_NEXT_HOP_TO_R1 out
!R7#clear ip bgp 155.1.146.1 out
```

Next, on R1, a similar route-map is used to change the next-hop value of routes learned from R8 to 155.1.58.8, followed by a route refresh request from R8 to apply the new policy:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#route-map SET_NEXT_HOP_FROM_R8 permit 10
R1(config-route-map)#set ip next-hop 155.1.58.8
R1(config-route-map)#router bgp 100
R1(config-router)#neighbor 155.1.58.8 route-map SET_NEXT_HOP_FROM_R8 in
!R1#clear ip bgp 155.1.58.8 in
```

https://t.me/learningnets

The final result is that prefixes that R7 sends to R1 appear with a next-hop of 155.1.67.7 in the BGP table of R1, whereas prefixes that R8 sends to R1 appear with a next-hop value of 155.1.58.8. This effect is essentially the same as using the **`next-hop-self`** feature, but it allows us more control over the BGP policy. In certain traffic engineering designs, it may be necessary to change the next-hop value of a prefix to an address completely unrelated to the neighbor that the prefix is learned from:

```
R1#show ip bgp
BGP table version is 40, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found


    Network          Next Hop            Metric LocPrf Weight Path * i 28.119.16.0/24155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i  * i 28.119.17.0/24155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i  * i 112.0.0.0155.1.58.8
               0    100      0 54 50 60 i  *>i155.1.67.7
               0    100      0 54 50 60 i  * i 113.0.0.0155.1.58.8
               0    100      0 54 50 60 i  *>i155.1.67.7
               0    100      0 54 50 60 i  * i 114.0.0.0155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i  * i 115.0.0.0155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i  * i 116.0.0.0155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i  * i 117.0.0.0155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i  * i 118.0.0.0155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i  * i 119.0.0.0155.1.58.8
               0    100      0 54 i  *>i155.1.67.7
               0    100      0 54 i
<output omitted>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

iBGP Synchronization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **iBGP Synchronization** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- IBGP is preconfigured on R2 in AS 200 and all other R1 - R8 in AS 100.
- EBGP peerings with AS 54 are preconfigured between R7 and R9, between R8 and R10.
- Enable iBGP synchronization on all devices in AS 100.
- Ensure full IPv4 reachability to Loopback0 prefixes of R1 - R8, and to all prefixes learned from AS 54 from R1 - R8 when sourcing traffic from Loopback0 interfaces.

Configuration

According to Cisco's BGP Bestpath Selection, "if BGP synchronization is enabled, there must be a match for the prefix in the IP routing table in order for an internal BGP (iBGP) path to be considered a valid path." In other words, for every iBGP route learned, there must be a matching IGP route already before the BGP route can be used. This rule was designed to prevent traffic black holes in legacy network designs where not all devices in the BGP transit path actually ran BGP.

With synchronization enabled, the assumption was that all routers in a transit network already run IGP. If these devices have IGP routes to all BGP destinations, it was safe to assume that traffic coming from other BGP ASs would not be blackholed, even if some of the internal routers were not running BGP. To implement this design, it was also assumed that some sort of BGP-to-IGP redistribution would occur.

The problem with this model, however, is that IGP simply cannot scale to the size of

https://t.me/learningnets

the current Internet BGP table. Instead, current best practices dictate that if a network is used for Internet transit, the routing table should be default free (that is, no 0.0.0.0/0 routes), and all devices should run BGP. Another common design is to use other tunneling mechanisms, such as MPLS, to limit the number of devices on which BGP must be run.

BGP synchronization is disabled by default as of IOS 12.2(8)T, and it is rarely, if ever, needed in a real implementation anymore. However, it is still important to understand the problem that synchronization was designed to prevent, and what the different resolutions for this problem are.

```
R1:
router eigrp 100
 network 150.1.1.1 0.0.0.0
!
router bgp 100
 synchronization

R2:
router eigrp 100
 network 150.1.2.2 0.0.0.0

R3:
router eigrp 100
 network 150.1.3.3 0.0.0.0
!
router bgp 100
 synchronization

R4:
router eigrp 100
 network 150.1.4.4 0.0.0.0
!
router bgp 100
 synchronization

R5:
router eigrp 100
 network 150.1.5.5 0.0.0.0
!
router bgp 100
 synchronization

R6:
```

https://t.me/learningnets

```
router eigrp 100
 network 150.1.6.6 0.0.0.0
!
router bgp 100
 synchronization
```

R7:

```
ip as-path access-list 1 permit ^54_
!
route-map BGP_TO_IGP permit 10
 match as-path 1
!
router eigrp 100
 redistribute bgp 100 metric 100000 1000 255 1 1500 route-map BGP_TO_IGP
 network 150.1.7.7 0.0.0.0
!
router bgp 100
 synchronization
```

R8:

```
ip as-path access-list 1 permit ^54_
!
route-map BGP_TO_IGP permit 10
 match as-path 1
!
router eigrp 100
 redistribute bgp 100 metric 100000 1000 255 1 1500 route-map BGP_TO_IGP
 network 150.1.8.8 0.0.0.0
!
router bgp 100
 synchronization
```

Verification

Pitfall

https://t.me/learningnets

Redistribution between IGP and BGP can be dangerous, causing traffic loops, black holes, and network instability as a result of memory and CPU resource exhaustion. In the previous configuration, only necessary routes are redistributed on R7 and R8 based on the AS-Path attribute. Route-map filtering should always be used when performing this type of redistribution to strictly control which prefixes are candidate for redistribution.

In the below output, BGP** `synchronization` ** is enabled with the synchronization command under the BGP process. Note that none of the iBGP-learned routes in the table (denoted by the “i” in the status code) are best routes. This is because the synchronization rule has not been met, as none of these routes have been redistributed into IGP in this example:

```
R1#show ip bgp
BGP table version is 90, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *i
 28.119.16.0/24   155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i *i
 28.119.17.0/24   155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i *i
 112.0.0.0        155.1.67.7               0    100      0 54 50 60 i *i
                  155.1.58.8               0    100      0 54 50 60 i *i
 113.0.0.0        155.1.67.7               0    100      0 54 50 60 i *i
                  155.1.58.8               0    100      0 54 50 60 i *i
 114.0.0.0        155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i *i
 115.0.0.0        155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i *i
 116.0.0.0        155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i *i
 117.0.0.0        155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i *i
 118.0.0.0        155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i *i
 119.0.0.0        155.1.67.7               0    100      0 54 i *i
                  155.1.58.8               0    100      0 54 i
 *>  150.1.1.1/32     0.0.0.0                  0         32768 i *i
 150.1.2.2/32     155.1.23.2               0    100      0 200 i
 *>i                  155.1.0.2                0    100      0 200 i
```

https://t.me/learningnets

```
 *i
 150.1.3.3/32     155.1.0.3                0    100      0 i  *i
 150.1.4.4/32     155.1.146.4              0    100      0 i  *i
 150.1.5.5/32     155.1.0.5                0    100      0 i  *i
 150.1.6.6/32     155.1.146.6              0    100      0 i  *i
 150.1.7.7/32     155.1.67.7               0    100      0 i  *i
 150.1.8.8/32     155.1.58.8               0    100      0 i
```

This can be further verified by the *not synchronized* output shown below. This means that synchronization is on, the route is learned from an iBGP neighbor, and there is no matching IGP route already in the routing table:

```
R1#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 83Paths: (2 available, no best path
)
  Not advertised to any peer
  Refresh Epoch 1
  54 50 60, (Received from a RR-client)
    155.1.67.7 (metric 3072) from 155.1.67.7 (150.1.7.7)
      Origin IGP, metric 0, localpref 100, valid, internal, not synchronized
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  54 50 60, (Received from a RR-client)
    155.1.58.8 (metric 3328) from 155.1.58.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal, not synchronized

      rx pathid: 0, tx pathid: 0
```

When BGP is redistributed into IGP, the synchronization rule is met, and the routes are installed as best paths, but the “r” in the status code denotes that RIB-failure now occurs:

```
R1#show ip bgp
BGP table version is 118, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path r>i
 28.119.16.0/24   155.1.67.7               0    100      0 54 i r i
                  155.1.58.8               0    100      0 54 i r>i
 28.119.17.0/24   155.1.67.7               0    100      0 54 i r i
                  155.1.58.8               0    100      0 54 i
```

https://t.me/learningnets

```
r>i
 112.0.0.0        155.1.67.7               0    100      0 54 50 60 i r i
                  155.1.58.8               0    100      0 54 50 60 i r>i
 113.0.0.0        155.1.67.7               0    100      0 54 50 60 i r i
                  155.1.58.8               0    100      0 54 50 60 i r>i
 114.0.0.0        155.1.67.7               0    100      0 54 i r i
                  155.1.58.8               0    100      0 54 i r>i
 115.0.0.0        155.1.67.7               0    100      0 54 i r i
                  155.1.58.8               0    100      0 54 i r>i
 116.0.0.0        155.1.67.7               0    100      0 54 i r i
                  155.1.58.8               0    100      0 54 i r>i
 117.0.0.0        155.1.67.7               0    100      0 54 i r i
                  155.1.58.8               0    100      0 54 i
<output omitted>
!R1#show ip bgp 28.119.17.0/24
BGP routing table entry for 28.119.17.0/24, version 110 Paths: (2 available, best #1, table default,
RIB-failure
(17))
  Advertised to update-groups:
     1
  Refresh Epoch 1
  54, (Received from a RR-client)
    155.1.67.7 (metric 3072) from 155.1.67.7 (150.1.7.7)
      Origin IGP, metric 0, localpref 100, valid, internal, synchronized, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  54, (Received from a RR-client)
    155.1.58.8 (metric 3328) from 155.1.58.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal, synchronized

      rx pathid: 0, tx pathid: 0
```

According to the output above on R1, 28.119.17.0/24 is now “synchronized” because there is a matching IGP route in the routing table. This means that the route can be used for Bestpath selection and can be advertised to other BGP neighbors.

The RIB-failure output is an informational message to let us know that although the BGP route is valid, it is not being installed in the routing table. This usually occurs when there is an identical match to a BGP route via an IGP route with a lower administrative distance. In the output below, we can see that the external EIGRP route with a distance of 170 prevents the iBGP route from being installed, which would normally have a distance of 200:

https://t.me/learningnets

```
R1#show ip route 28.119.17.0

Routing entry for 28.119.17.0/24 Known via "eigrp 100", distance 170
, metric 282112
  Tag 54, type external
  Redistributing via eigrp 100
  Last update from 155.1.146.6 on GigabitEthernet1.146, 00:04:04 ago
  Routing Descriptor Blocks: 155.1.146.6, from 155.1.146.6, 00:04:04 ago, via GigabitEthernet1.146
      Route metric is 282112, traffic share count is 1
      Total delay is 10020 microseconds, minimum bandwidth is 100000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
      Route tag 54 * 155.1.13.3, from 155.1.13.3, 00:04:04 ago, via GigabitEthernet1.13

      Route metric is 282112, traffic share count is 1
      Total delay is 10020 microseconds, minimum bandwidth is 100000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
      Route tag 54
```

RIB-failure by itself is not necessarily bad, but there are certain cases in which this disconnect between the BGP table and the routing table can cause traffic loops. By default, BGP routes that have RIB-failure *can* be advertised to other neighbors, because the command **`no bgp suppress-inactive`** is the default option under the routing process. To stop RIB-failure routes from being advertised, issue the **`bgp suppress-inactive`** command under the process.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP over GRE

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP over GRE**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R7 in AS 100.
- Configure R8 in AS 200, with an EBGP peering to R10 in AS 54 using the password **CISCO**.
  - R10 is preconfigured for this EBGP peering.
- Configure an EBGP peering between R7 and R8.
- Advertise the Loopback0 networks of R7 and R8 into BGP.
- Ensure that R7 and R8 can reach prefixes learned from AS 54 when sourcing traffic from their Loopback0 interfaces.
- Do not redistribute between BGP and IGP to accomplish this.

Configuration

Using automatic tunneling techniques along with BGP is the core of MPLS VPNs. In this case we'll use simple manual tunnels along with BGP to better understand possible effects. Two devices peer BGP (this could be eBGP or iBGP session) across a non-BGP-capable router cloud. This configuration means that any attempt to reach a BGP prefix across the non-BGP cloud would result in prefix blackholing. However, if we establish a direct tunnel between the BGP peers and force all packets go across the tunnel, the non-BGP devices will never notice those packets. Thus, the “unknown” addresses will be hidden from the “core” network, only appearing at the edge routers that know about them.

Notice the trick used in the solution. Although the “core” IP addresses are used for

https://t.me/learningnets

BGP peering, next-hops in BGP prefixes are modified to point to the tunnel endpoints. Alternatively, you could have peered directly off the tunnel endpoints or even used policy routing to divert packets to the tunnel interfaces:

```
R7:
interface Tunnel0
 ip address 10.0.0.7 255.255.255.0
 tunnel source 155.1.67.7
 tunnel destination 155.1.58.8
!
router bgp 100
 neighbor 155.1.58.8 remote-as 200
 neighbor 155.1.58.8 ebgp-multihop 255
 network 150.1.7.7 mask 255.255.255.255

R8:

interface Tunnel0
 ip address 10.0.0.8 255.255.255.0
 tunnel source 155.1.58.8
 tunnel destination 155.1.67.7
!
route-map SET_NEXT_HOP_TO_TUNNEL_OUT permit 10
 set ip next-hop 10.0.0.8
!
route-map SET_NEXT_HOP_TO_TUNNEL_IN permit 10
 set ip next-hop 10.0.0.7
!
router bgp 200
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.67.7 ebgp-multihop 255
 neighbor 155.1.67.7 route-map SET_NEXT_HOP_TO_TUNNEL_OUT out
 neighbor 155.1.67.7 route-map SET_NEXT_HOP_TO_TUNNEL_IN in
 neighbor 155.1.108.10 remote-as 54
 neighbor 155.1.108.10 password CISCO
 network 150.1.8.8 mask 255.255.255.255
```

Verification

Look at the BGP table in R7. Notice that prefixes learned from R8 have their next-hops pointing to the tunnel endpoint:

```
R7#show ip bgp
```

https://t.me/learningnets

```
BGP table version is 23, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>  28.119.16.0/24 10.0.0.8
                               0 200 54 i *>  28.119.17.0/24 10.0.0.8
                               0 200 54 i *>  112.0.0.0 10.0.0.8
                               0 200 54 50 60 i *>  113.0.0.0 10.0.0.8
                               0 200 54 50 60 i *>  114.0.0.0 10.0.0.8
                               0 200 54 i *>  115.0.0.0 10.0.0.8
                               0 200 54 i *>  116.0.0.0 10.0.0.8
                               0 200 54 i *>  117.0.0.0 10.0.0.8
                               0 200 54 i *>  118.0.0.0 10.0.0.8
                               0 200 54 i *>  119.0.0.0 10.0.0.8
                               0 200 54 i
 *>  150.1.7.7/32     0.0.0.0                  0         32768 i *>  150.1.8.8/32 10.0.0.8
                 0             0 200 i
```

Now ping any prefix learned from R8, such as 112.0.0.0/24. Source the packets off R7's Loopback0 interface, because this is the only R7 network known to the external AS 54 systems:

```
R7#ping 112.0.0.1 source lo0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 112.0.0.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.7.7 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 12/38/97 ms
```

Next, trace the route to the same prefix off R7's Loopback0 interface. Notice that the path taken by the packets goes across the tunnel interface, and any intermediate nodes between R7 and R8 do not appear in the output:

https://t.me/learningnets

```
R7#traceroute 112.0.0.1 source loopback 0


Type escape sequence to abort.
Tracing the route to 112.0.0.1
VRF info: (vrf in name/id, vrf out name/id) 1 10.0.0.8 20 msec 6 msec 10 msec

  2 155.1.108.10 53 msec 22 msec 31 msec
  3 155.1.109.9 32 msec *  13 msec
```

Now look at R8's BGP table and notice that the prefix learned from R7 has its next-hop pointing to 10.0.0.7 (the tunnel endpoint):

```
R8#show ip bgp neighbors 155.1.67.7 routes
BGP table version is 13, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>  150.1.7.7/32 10.0.0.7
                 0             0 100 i

Total number of prefixes 1
```

Repeat the traceroute test again, now targeting R7's Loopback0. Notice that the very first hop is the tunnel endpoint again. The transit nodes are unaware of the encapsulated packet's destination IP address. For this reason, none of the transit nodes need to have routing information about the tunneled packet's destination IP address (150.1.7.7/32):

```
R8#traceroute 150.1.7.7 source loopback 0


Type escape sequence to abort.
Tracing the route to 150.1.7.7
VRF info: (vrf in name/id, vrf out name/id) 1 10.0.0.7 22 msec *  10 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Redistribute Internal

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Redistribute Internal**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that there is no BGP configuration on R1 - R8.
- Configure R1, R3, R7, and R8 in AS 100.
- R7 and R8 should peer with R9 and R10, respectively, which are in AS 54.
- R9 and R10 are preconfigured for these EBGP peerings.
- R1 should peer with R3, R7, and R8 as a route reflector.
- Configure R7 and R8 to advertise the network 155.1.0.0/16 to AS 54.
- Configure BGP to IGP redistribution on R3 so that all internal devices have reachability to the prefixes learned from AS 54.

Configuration

As discussed previously, enabling BGP on all transit devices in the AS is one way to ensure that all routers have full external routing information. This solution is scalable, because it avoids feeding large BGP tables into IGP.

In some situations, you cannot enable BGP on all routers in your network. This may be the case of an enterprise network, in which only border routers peer eBGP with the ISP. In such situations, it is common to advertise a default route from the border routers toward the rest of the network.

There could be even more complicated scenarios, such as migrating your network or gradually enabling BGP on all devices. In situations like these, you may find that iBGP peers are separated by non-BGP cloud or that non-BGP speakers need BGP

https://t.me/learningnets

routes from the devices that learned them via iBGP (no eBGP!). So what's wrong with redistributing iBGP prefixes into IGP? As you remember, BGP uses AS_PATH attributes to detect routing loops. When exchanging iBGP routes, AS_PATH attributes are not prepended and thus the route loop prevention technique does not work. Because of that, feeding iBGP prefixes into an IGP may result in routing loops, because the "split-horizon" rules for BGP prefixes may be broken. To make this situation even worse, iBGP has the AD value that makes it less preferred than any IGP. Thus, iBGP prefixes redistributed into an IGP may preempt iBGP-learned prefixes on other iBGP speakers.

To prevent the above issues, iBGP-learned prefixes are not automatically redistributed into IGP when you issue the statement **`redistribute bgp`** under any IGP process on the router, only eBGP prefixes are redistributed. To make iBGP redistribution possible, you need an additional statement configured under the BGP process: **`bgp redistribute internal`** . Be very careful when enabling this feature, because you may quickly end up with routing loops, and try to avoid multiple points of iBGP to IGP redistribution.

In our task, we have to change R1's EIGRP External AD to avoid the effect of iBGP prefixes being preempted by IGP routes.

```
R1:
router eigrp 100
 distance eigrp 90 201
!
router bgp 100
 neighbor 155.1.67.7 remote-as 100
 neighbor 155.1.58.8 remote-as 100
 neighbor 155.1.13.3 remote-as 100
 neighbor 155.1.67.7 route-reflector-client
 neighbor 155.1.58.8 route-reflector-client
 neighbor 155.1.13.3 route-reflector-client

R3:
router eigrp 100
 redistribute bgp 100 metric 100000 1000 255 1 1500
!
router bgp 100
 neighbor 155.1.13.1 remote-as 100
 bgp redistribute-internal

R7:
router bgp 100
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.146.1 next-hop-self
```

https://t.me/learningnets

```
 neighbor 155.1.79.9 remote-as 54
 network 155.1.67.0 mask 255.255.255.0
 aggregate-address 155.1.0.0 255.255.0.0

R8:

router bgp 100
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.146.1 next-hop-self
 neighbor 155.1.108.10 remote-as 54
 network 155.1.58.0 mask 255.255.255.0
 aggregate-address 155.1.0.0 255.255.0.0
```

Verification

If you remove the **`bgp redistribute internal`** command from R3, none of the iBGP-learned prefixes will get redistributed into IGP:

```
R3#configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router bgp 100
R3(config-router)#no bgp redistribute-internal
!R3#clear ip route *
R3#show ip eigrp topology 112.0.0.0/8
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3)%Entry 112.0.0.0/8 not in topology table
```

112.0.0.0/8, an iBGP route, will not get redistributed unless **`bgp redistribute internal`** is used under the BGP process. Note that after adding the command, 112.0.0.0/8 appears in the EIGRP topology table via BGP to EIGRP redistribution:

```
R3#show ip eigrp topology 112.0.0.0/8
EIGRP-IPv4 Topology Entry for AS(100)/ID(150.1.3.3) for 112.0.0.0/8
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 281600
  Descriptor Blocks:  155.1.67.7, from Redistributed
, Send flag is 0x0
      Composite metric is (281600/0), route is External
      Vector metric:
        Minimum bandwidth is 100000 Kbit
    Total delay is 10000 microseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 0
```

https://t.me/learningnets

```
    Originating router is 150.1.3.3
  External data:
    AS number of route is 100 External protocol is BGP
, external metric is 0
    Administrator tag is 54 (0x00000036)
```

R1 has an AD fixup so that when R3 redistributes the iBGP routes into EIGRP, these external EIGRP routes with AD 170 will not be preferred on R1 over the same iBGP routes with AD 200:

```
R1#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 18
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  54 50 60, (Received from a RR-client) 155.1.58.8
 (metric 3328) from 155.1.58.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  54 50 60, (Received from a RR-client) 155.1.67.7
 (metric 3072) from 155.1.67.7 (150.1.7.7)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
R5#traceroute 112.0.0.1



Type escape sequence to abort.
Tracing the route to 112.0.0.1
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.0.3 20 msec 7 msec 13 msec
  2 155.1.37.7 4 msec 7 msec 8 msec
  3 155.1.79.9 114 msec *  22 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Peer Groups

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial BGP Base**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that there is no BGP configuration on R1 - R8.
- Configure BGP on R1 - R8 using AS 100.
- Configure iBGP peerings from R1 to all other devices in AS 100 using the peer group named **IBGP_PEERS**:
  - Be sure to use the Loopback0 of these devices to form the peerings.
- Configure EBGP peerings between R7 and R9, between R8 and R10 using their directly connected links:
  - R9 and R10 are in AS 54 and are preconfigured.
- Advertise the Loopback0 interfaces of all devices into both IGP and BGP.
- Ensure full IPv4 reachability to Loopback0 prefixes and all prefixes learned from AS 54 from R1 - R8 when sourcing traffic from Loopback0 interfaces.

Configuration

Large-scale BGP deployments often implement hundreds of BGP peers for a single router. The high number of peers creates certain scalability issues. For example, the configuration burden may become too intense to deal with and be error-prone. Or, numerous BGP peers may require the router CPU to prepare, batch, and replicate BGP updates (often hundreds of thousands of prefixes) individually for each peer. This puts great stress on router's CPUs.

BGP peer groups try to overcome these two issues by taking advantage of the fact that many peers often have similar BGP policy configuration. For example, they all

https://t.me/learningnets

have the same remote AS, outgoing route filters, route-reflection-properties, and so on. Based on this fact, many BGP peers sharing the same outgoing policy could be aggregated into a single BGP peer group. The peer group is essentially a template that specifies a particular BGP policy. You create a peer group by using the command **neighbor <PEER_GROUP_TAG> peer-group** and then applying all BGP settings to the named tag as if it were a regular BGP peer. When you're done with the peer group configuration, you assign BGP peers to the group using the command **neighbor <IP_Address> peer-group <PEER_GROUP_TAG> .**

Before IOS 12.0(24)S, peer groups was a way to not only optimize the router's configuration complexity, but also to optimize the CPU. The router's CPU usage is highly optimized when using the peer groups. All members of the peer group share the same policy, therefore allowing a batch of BGP prefixes to be prepared just once before being relayed to all peer group members. However, this optimization process means that you cannot tune outgoing BGP settings for every member of a peer group individually. For example, you cannot apply a different outgoing route-map to the neighbor that is a member of a BGP peer group. However, you are free to change any incoming policy settings, such as inbound filters.

After IOS 12.0(24)S, a new feature called BGP Dynamic Peer-Groups was introduced, which decouples update generation from peer-group configuration. One of the limitations of peer groups was that the network operator had to ensure that all members of the peer group had the same outbound policy. If any peer needed a different outbound policy, a separate peer group had to be created. The Dynamic Peer-Groups feature runs an algorithm that automatically places routers with similar outbound policy in the same update-group, and does not require any configuration from the network operator. See BGP Dynamic Update Peer-Groups.

R1:

```
router bgp 100
 neighbor IBGP_PEERS peer-group
 neighbor IBGP_PEERS remote-as 100
 neighbor IBGP_PEERS update-source Loopback0
 neighbor IBGP_PEERS route-reflector-client
 neighbor 150.1.2.2 peer-group IBGP_PEERS
 neighbor 150.1.3.3 peer-group IBGP_PEERS
 neighbor 150.1.4.4 peer-group IBGP_PEERS
 neighbor 150.1.5.5 peer-group IBGP_PEERS
 neighbor 150.1.6.6 peer-group IBGP_PEERS
 neighbor 150.1.7.7 peer-group IBGP_PEERS
 neighbor 150.1.8.8 peer-group IBGP_PEERS
 network 150.1.1.1 mask 255.255.255.255
!
router eigrp 100
```

https://t.me/learningnets

```
 network 150.1.1.1 0.0.0.0
```

R2:

```
router bgp 100
 neighbor 150.1.1.1 remote-as 100
 neighbor 150.1.1.1 update-source Loopback0
 network 150.1.2.2 mask 255.255.255.255
!
router eigrp 100
 network 150.1.2.2 0.0.0.0
```

R3:

```
router bgp 100
 neighbor 150.1.1.1 remote-as 100
 neighbor 150.1.1.1 update-source Loopback0
 network 150.1.3.3 mask 255.255.255.255
!
router eigrp 100
 network 150.1.3.3 0.0.0.0
```

R4:

```
router bgp 100
 neighbor 150.1.1.1 remote-as 100
 neighbor 150.1.1.1 update-source Loopback0
 network 150.1.4.4 mask 255.255.255.255
!
router eigrp 100
 network 150.1.4.4 0.0.0.0
```

R5:

```
router bgp 100
 neighbor 150.1.1.1 remote-as 100
 neighbor 150.1.1.1 update-source Loopback0
 network 150.1.5.5 mask 255.255.255.255
!
router eigrp 100
 network 150.1.5.5 0.0.0.0
```

R6:

```
router bgp 100
 neighbor 150.1.1.1 remote-as 100
 neighbor 150.1.1.1 update-source Loopback0
 network 150.1.6.6 mask 255.255.255.255
!
router eigrp 100
 network 150.1.6.6 0.0.0.0
```

https://t.me/learningnets

```
R7:
router bgp 100
 neighbor 150.1.1.1 remote-as 100
 neighbor 150.1.1.1 update-source Loopback0
 neighbor 150.1.1.1 next-hop-self
 neighbor 155.1.79.9 remote-as 54
 network 150.1.7.7 mask 255.255.255.255
!
router eigrp 100
 network 150.1.7.7 0.0.0.0

R8:

router bgp 100
 neighbor 150.1.1.1 remote-as 100
 neighbor 150.1.1.1 update-source Loopback0
 neighbor 150.1.1.1 next-hop-self
 neighbor 155.1.108.10 remote-as 54
 network 150.1.8.8 mask 255.255.255.255
!
router eigrp 100
 network 150.1.8.8 0.0.0.0
```

Verification

First, check the peer group configured in R1. You can see the peer group parameters by using the command **`show ip bgp peer-group`** . This command shows the group members in addition to displaying the group parameters:

```
R1#show ip bgp peer-group

BGP peer-group is IBGP_PEERS,  remote AS 100
  BGP version 4
  Neighbor sessions:
    0 active, is not multisession capable (disabled)
  Default minimum time between advertisement runs is 0 seconds

 For address family: IPv4 Unicast
  BGP neighbor is IBGP_PEERS, peer-group internal, members:
150.1.2.2 150.1.3.3 150.1.4.4 150.1.5.5 150.1.6.6 150.1.7.7 150.1.8.8

  Index 0, Advertise bit 0
  Route-Reflector Client
```

https://t.me/learningnets

```
  Interface associated: (none)
  Update messages formatted 0, replicated 0
  Number of NLRIs in the update sent: max 0, min 0
```

Now check the BGP table of R1. Make sure you received prefixes from both upstream peers:

https://t.me/learningnets

```
R1#show ip bgp

BGP table version is 29, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>i 28.119.17.0/24   150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>i 112.0.0.0        150.1.7.7               0    100      0 54 50 60 i
 * i                  150.1.8.8               0    100      0 54 50 60 i
 *>i 113.0.0.0        150.1.7.7               0    100      0 54 50 60 i
 * i                  150.1.8.8               0    100      0 54 50 60 i
 *>i 114.0.0.0        150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>i 115.0.0.0        150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>i 116.0.0.0        150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>i 117.0.0.0        150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>i 118.0.0.0        150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>i 119.0.0.0        150.1.7.7               0    100      0 54 i
 * i                  150.1.8.8               0    100      0 54 i
 *>  150.1.1.1/32     0.0.0.0                 0         32768 i
 r>i 150.1.2.2/32     150.1.2.2               0    100      0 i
 r>i 150.1.3.3/32     150.1.3.3               0    100      0 i
 r>i 150.1.4.4/32     150.1.4.4               0    100      0 i
 r>i 150.1.5.5/32     150.1.5.5               0    100      0 i
 r>i 150.1.6.6/32     150.1.6.6               0    100      0 i
 r>i 150.1.7.7/32     150.1.7.7               0    100      0 i
 r>i 150.1.8.8/32     150.1.8.8               0    100      0 i
```

Now you can test the connectivity by pinging the external prefixes off all routers' Loopback0 interfaces:

```
R1#ping 112.0.0.1 source loopback 0
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 112.0.0.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 7/16/33 ms
```

Having these peer groups essentially just simplified the configuration of R1. BGP Dynamic Update Peer-Groups were automatically formed on R1, and also on every BGP peer. Note that no configuration is needed for BGP Dynamic Update Peer-Groups to work, not even the manual peer groups that were used in this example. If the manual peer groups were to be removed, R1 would still run the dynamic algorithm and group all peers with the same outbound policy into the same update group:

```
R1#show ip bgp update-group

BGP version 4 update-group 1, internal, Address Family: IPv4 Unicast
 BGP Update version : 19/0, messages 0
 Route-Reflector Client
 Topology: global, highest version: 19, tail marker: 19
 Format state: Current working (OK, last not in list)
               Refresh blocked (not in list, last not in list)
 Update messages formatted 47, replicated 76, current 0, refresh 0, limit 1000
 Number of NLRIs in the update sent: max 6, min 0
 Minimum time between advertisement runs is 0 seconds
Has 7 members:
  150.1.2.2        150.1.3.3        150.1.4.4        150.1.5.5
  150.1.6.6        150.1.7.7        150.1.8.8
```

Because all of these devices have the same outbound policy, the Dynamic Update Peer-Group feature placed all of R1's peers in the same dynamic group. When R1 needs to build an update and send it to all of its peers, it will build and format the update for only one member of the dynamic update-group (the group leader). After it is done building the update for the leader, it will send it to all of the members of the dynamic update-group. If R1 did not have dynamic update-groups, it would have to build and format a new update for each one of its peers. In this case, R1 only had to build the update once, and then just replicate it out to all of its peers. Building and formatting the update is very CPU intensive. However, sending the update consumes far fewer CPU resources. This feature is extremely useful inside a service provider network where a route-reflector peers with hundreds of route-reflector clients. Instead of having to build a separate update for each client, the

https://t.me/learningnets

route-reflector just builds it once for the group leader and sends it out to all of its peers:

```
R1#show ip bgp replication

                                                                    Current    Next
Index  Members Leader
       MsgFmt    MsgRepl     Csize    Version Version 1        7 150.1.2.2
           47         76    0/1000         19/0
```

All other BGP speakers in the network also build dynamic update peer groups. Because they are only peering with R1, the only member (and leader) of the group will be R1:

```
R4#show ip bgp update-group

BGP version 4 update-group 1, internal, Address Family: IPv4 Unicast
  BGP Update version : 19/0, messages 0
  Topology: global, highest version: 19, tail marker: 19
  Format state: Current working (OK, last not in list)
                Refresh blocked (not in list, last not in list)
  Update messages formatted 1, replicated 1, current 0, refresh 0, limit 1000
  Number of NLRIs in the update sent: max 1, min 0
  Minimum time between advertisement runs is 0 seconds
  Has 1 member: 150.1.1.1

!R4#show ip bgp replication

                                                                    Current    Next
Index  Members          Leader       MsgFmt    MsgRepl     Csize    Version Version 1        1 150.1.1.1
           1          1    0/1000         19/0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Network Statement

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Full**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Remove the `next-hop-self` statement used in the initial configurations on R7 and R8 toward their route-reflector, R1.
- Ensure full IPv4 reachability from your internal network to all routes learned from AS 54 via the EBGP peering between R7 and R8 with R10.
  - Do not advertise any prefixes into IGP.

Configuration

Unlike the network statement used in IGP protocols configuration, the BGP version of the command is different. The basic command syntax is simple: `network <subnet> mask <netmask>` . It does not define a group of interfaces to enable the protocol, it only specifies the prefix in the IGP table (RIB) to be imported into BGP LocRIB. The term *LocRIB* stands for Local RIB and is another name for BGP table, which is separate from the routing table (RIB). For the prefix to be imported, it must exactly match the specification; it should have the same subnet number and network mask. For example, if you have interface Loopback0 with the IP address 150.1.1.1/24, the command would be `network 150.1.1.0 mask 255.255.255.0` , not `network 150.1.1.1 mask 255.255.255.255` . The second statement will not match any route in the IGP and therefore will not import any prefix. Notice that you may omit the mask specification if it matches the default mask for the IPv4 address class (such as 255.255.255.0 for class C).

When originating prefixes into BGP, it is common to use summarization to minimize the amount of information advertised. One way to achieve this is by creating a

https://t.me/learningnets

special static route that points to Null0 interface but encompasses all subnets of the particular AS. The static route is then advertised into the BGP table using the `network` command. For example:

```
ip route 150.0.0.0 255.252.0.0 Null0
!
router bgp 100
 network 150.0.0.0 mask 255.252.0.0
```

One of the special uses for the `network` command is demonstrated in this task. When peering with another AS, the common question is how to deal with the external next-hop. One way is to use the `next-hop-self` parameter when peering via iBGP or advertising the external link subnet into IGP. Another way is to advertise the link subnet into BGP and thus propagate it to all iBGP peers. All BGP routers will install it into their RIBs and perform a recursive lookup to find the actual next hop for every BGP prefix learned from the external AS.

```
R7:
router bgp 100
 no neighbor 150.1.1.1 next-hop-self
 network 155.1.79.0 mask 255.255.255.0

R8:

router bgp 100
 no neighbor 150.1.1.1 next-hop-self
 network 155.1.108.0 mask 255.255.255.0
```

Verification

Look at the BGP table of R1 and notice that both link subnets are there:

```
R1#show ip bgp

BGP table version is 51, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
```

https://t.me/learningnets

```
 *>i                  155.1.79.9               0    100      0 54 i
 * i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 *>i                  155.1.79.9               0    100      0 54 i
 * i 112.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i                  155.1.79.9               0    100      0 54 50 60 i
 * i 113.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i                  155.1.79.9               0    100      0 54 50 60 i
 * i 114.0.0.0        155.1.108.10             0    100      0 54 i
 *>i                  155.1.79.9               0    100      0 54 i
 * i 115.0.0.0        155.1.108.10             0    100      0 54 i
 *>i                  155.1.79.9               0    100      0 54 i
 * i 116.0.0.0        155.1.108.10             0    100      0 54 i
 *>i                  155.1.79.9               0    100      0 54 i
 * i 117.0.0.0        155.1.108.10             0    100      0 54 i
 *>i                  155.1.79.9               0    100      0 54 i
 * i 118.0.0.0        155.1.108.10             0    100      0 54 i
 *>i                  155.1.79.9               0    100      0 54 i
 * i 119.0.0.0        155.1.108.10             0    100      0 54 i
 *>i                  155.1.79.9               0    100      0 54 i
 *>  150.1.1.1/32     0.0.0.0                  0         32768 i
 r>i 150.1.2.2/32     150.1.2.2                0    100      0 i
 r>i 150.1.3.3/32     150.1.3.3                0    100      0 i
 r>i 150.1.4.4/32     150.1.4.4                0    100      0 i
 r>i 150.1.5.5/32     150.1.5.5                0    100      0 i
 r>i 150.1.6.6/32     150.1.6.6                0    100      0 i
 r>i 150.1.7.7/32     150.1.7.7                0    100      0 i
 r>i 150.1.8.8/32     150.1.8.8                0    100      0 i *>i 155.1.79.0/24    150.1.7.7
                0    100      0 i *>i 155.1.108.0/24   150.1.8.8
                0    100      0 i
```

Follow the route recursion for 112.0.0.0/8:

```
R1#show ip bgp 112.0.0.0
BGP routing table entry for 112.0.0.0/8, version 43
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  54 50 60, (Received from a RR-client) 155.1.108.10
 (metric 131328) from 150.1.8.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  54 50 60, (Received from a RR-client) 155.1.79.9
 (metric 131072) from 150.1.7.7 (150.1.7.7)
```

https://t.me/learningnets

```
      Origin IGP, metric 0, localpref 100, valid, internal, best
  rx pathid: 0, tx pathid: 0x0
!R1#show ip route 155.1.108.10
Routing entry for 155.1.108.0/24 Known via "bgp 100
", distance 200, metric 0, type internal
  Last update from 150.1.8.8 00:05:03 ago
  Routing Descriptor Blocks:  * 150.1.8.8
, from 150.1.8.8, 00:05:03 ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: none
!R1#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32 Known via "eigrp 100"
, distance 90, metric 131328, type internal
  Redistributing via eigrp 100
  Last update from 155.1.146.4 on GigabitEthernet1.146, 01:08:35 ago
  Routing Descriptor Blocks:  * 155.1.146.4
, from 155.1.146.4, 01:08:35 ago, via GigabitEthernet1.146
      Route metric is 131328, traffic share count is 1
      Total delay is 5030 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3
!R1#show ip route 155.1.146.4
Routing entry for 155.1.146.0/24 Known via "connected"
, distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks:  * directly connected, via GigabitEthernet1.146

      Route metric is 0, traffic share count is 1
```

Advertising the next-hop via BGP adds an extra step in the recursion process. However, the outgoing interface is found and connectivity can be properly checked. Repeat the below command on all BGP-enabled routers:

```
R1#ping 112.0.0.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 112.0.0.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 5/21/44 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Auto-Summary

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **BGP Full**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Ensure that no advertisements are being sent to R9 and R10 in AS 54 before you start this task.
- Configure R7 and R8 to originate classful auto-summaries for all of your internally assigned address spaces.
- R9 and R10 should not see any of the subnet advertisements that make up this summary.
- Ensure full reachability from R1 - R8 to all routes learned from AS 54.
- Do not use the `aggregate-address` command to accomplish this, and use different methods to originate routes at R7 and R8.

Configuration

BGP auto-summarization is the legacy feature that automatically summarizes network prefixes to their classful boundaries when the prefixes are advertised into BGP. The automatic summarization starts working when you enable it using the command `auto-summary` under BGP process configuration. It only applies in the following two cases:

1. A `network` command is configured with a *classful* subnet, such as `network 54.0.0.0` or `network 155.1.0.0` or `network 192.168.1.0` . In this case, the classful aggregate is installed into the BGP table if there is a prefix in the IGP table that is a *subnet* to the classful network. For example, if you advertise network 150.1.0.0, it would work if any of the prefixes (150.1.2.0/24 or 150.1.3.0/24, etc.) are in the IGP table. This is

https://t.me/learningnets

contrary to the regular exact match requirement imposed by the BGP network statements.

2. Prefixes are advertised into BGP using route *redistribution*. All redistributed networks are subject to auto-summarization; that is, only the major classful subnets are installed in the BGP table.

Because the feature is legacy, you won't see much use of it. However, it may become handy in some CCIE scenario that verifies your knowledge of BGP advertisement methods. This scenario uses both methods of route origination with prefix auto-summarization: classful network statement and route redistribution.

```
R1:
router bgp 100
 no network 150.1.1.1 mask 255.255.255.255

R2:
router bgp 100
 no network 150.1.2.2 mask 255.255.255.255

R3:
router bgp 100
 no network 150.1.3.3 mask 255.255.255.255

R4:
router bgp 100
 no network 150.1.4.4 mask 255.255.255.255

R5:
router bgp 100
 no network 150.1.5.5 mask 255.255.255.255

R6:
router bgp 100
 no network 150.1.6.6 mask 255.255.255.255

R7:
route-map CONNECTED_TO_BGP
 match interface Loopback0
 match interface GigabitEthernet1.79
!
router bgp 100
 no network 150.1.7.7 mask 255.255.255.255
 no network 155.1.79.0 mask 255.255.255.0
 auto-summary
```

https://t.me/learningnets

```
 redistribute connected route-map CONNECTED_TO_BGP

R8:

router bgp 100
 no network 150.1.8.8 mask 255.255.255.255
 no network 155.1.108.0 mask 255.255.255.0
 auto-summary
 network 150.1.0.0
 network 155.1.0.0
```

Verification

Start by checking the BGP tables of R7 and R8 for auto-summarized prefixes. Notice that the `regex ^$` filter is used to select the routes locally advertised in this AS. Notice the origin of “?”, which means the prefix has been redistributed into BGP. The prefix 155.1.0.0 even has BGP metric assigned based on the IGP metric (EIGRP metric):

```
R7#show ip bgp regexp ^$
BGP table version is 31, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 150.1.0.0        150.1.8.8                0    100      0 i *>                    0.0.0.0
                  0 32768 ?
 r>i 150.1.1.1/32     150.1.1.1                0    100      0 i
 * i 155.1.0.0        150.1.8.8                0    100      0 i  *>                   0.0.0.0
                  0 32768 ?
```

Now look up those prefixes in the BGP table. Notice that both prefixes appear as though they were NOT summarized in the classic BGP sense. That is, prefixes do not have any information about the aggregator or the atomic aggregate attribute. This is because summarization was performed on the IGP prefixes, not the BGP prefixes:

```
R7#show ip bgp 150.1.0.0
```

https://t.me/learningnets

```
BGP routing table entry for 150.1.0.0/16, version 27
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          3
  Refresh Epoch 2
  Local
    150.1.8.8 (metric 131584) from 150.1.1.1 (150.1.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal
      Originator: 150.1.8.8, Cluster list: 150.1.1.1
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (150.1.7.7)
      Origin incomplete, metric 0, localpref 100, weight 32768, valid, sourced, best
      rx pathid: 0, tx pathid: 0x0
!R4#show ip bgp 155.1.0.0

BGP routing table entry for 155.1.0.0/16, version 28
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          3
  Refresh Epoch 2
  Local
    150.1.8.8 (metric 131584) from 150.1.1.1 (150.1.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal
      Originator: 150.1.8.8, Cluster list: 150.1.1.1
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (150.1.7.7)
      Origin incomplete, metric 0, localpref 100, weight 32768, valid, sourced, best
     rx pathid: 0, tx pathid: 0x0
```

Now check the BGP table in R8 and confirm that auto-summarized prefixes are there. In this router, both prefixes have the origin of “i”, which means IGP:

```
R8#show ip bgp regexp ^$
BGP table version is 35, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
```

https://t.me/learningnets

```
*>  150.1.0.0        0.0.0.0                  0 32768 i

 r>i 150.1.1.1/32     150.1.1.1                0    100      0 i *>  155.1.0.0        0.0.0.0
                  0 32768 i
```

Now confirm that only the classful summaries for the internal subnets are advertised to the external BGP peers:

```
R7#show ip bgp neighbors 155.1.79.9 advertised-routes

BGP table version is 33, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>  150.1.0.0
        0.0.0.0                  0         32768 ? *>  155.1.0.0
        0.0.0.0                  0         32768 ?

Total number of prefixes 2
!R8#show ip bgp neighbors 204.12.1.254 advertised-routes

BGP table version is 37, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>  150.1.0.0
        0.0.0.0                  0         32768 i *>  155.1.0.0
        0.0.0.0                  0         32768 i

Total number of prefixes 2
```

Finally, perform the following connectivity check on all IBGP speakers to verify full reachability. You should ping any BGP prefix learned from the external BGP peers:

```
R1#ping 112.0.0.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 112.0.0.1, timeout is 2 seconds:
```

https://t.me/learningnets

```
Packet sent with a source address of 150.1.1.1 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 8/18/32 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - Weight

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Using the most influential attribute configure R7 as follows:
  - Traffic from AS 300 going to prefixes originated in AS 54 exits toward R3.
  - Traffic from AS 300 going to prefixes originated in AS 254 exits toward R6.

Configuration

BGP Bestpath selection is the core of the BGP routing process. This process replaces the shortest-path selection procedure found in traditional IGPs. The reason to use such a complicated procedure instead of the shortest path is that BGP prefixes cannot have a classic (additive) metric associated with them, as BGP in most cases is a multi-hop peering. The purpose of the BGP bestpath procedure is to select optimal paths based on *administrative* preferences while maintaining the following properties:

- **Routing Loop detection.** The best paths selected should form a loop-free topology. BGP implements this by filtering prefixes with the AS number matching the local AS in the AS_PATH attributes.
- **Deterministic path selection.** All BGP routers under the same conditions (such as all IBGP speakers configured similarly) must select the same best paths.
- **Routing table stability.** The best path selection procedure should not result in constant oscillating route insertion and removals.

https://t.me/learningnets

- **Information flooding minimization.** A BGP speaker only sends the best paths to its neighbors. This significantly reduces the amount of update flooding, saving bandwidth and CPU cycles.

Before we start with the best-path process description, recall that every BGP prefix has a set of attributes associated with it. The procedure uses those attributes when looking for optimal/best paths. Some of the attributes have more influence on the result than others, as we'll see later. Before the procedure runs, the bestpath process excludes some prefixes based on the following criteria:

- **No valid next-hop.** This is the most common cause for the prefix being ignored by the selection process. BGP prefixes carry their next-hop as a separate attribute (NEXT_HOP attribute). If the next-hop address is NOT reachable via IGP, the prefix is marked as invalid and is not considered. This usually happens with eBGP-learned prefixes when you forget to enter the command `next-hop-self` or advertise the link subnet into IGP/BGP.
- **BGP Synchronization enabled** and the prefix is not in the IGP table. The bestpath process will ignore this prefix. This is a legacy restriction, but you may occasionally run into it.
- **AS_PATH Loop.** Prefixes from the neighbor that has the local AS number in the AS_PATH attribute are dropped. This is the well-known BGP loop detection mechanism.

All eligible paths are then sorted, and prefixes for the same destination (subnet/mask) are grouped together (the actual implementation may differ, though, as a result of various optimizations). For every group, BGP must elect the best path. Here is a short outline of the steps performed by the selection process. Every step is tried if the previous one cannot reveal the best path:

1. Ignore invalid paths (no valid next-hop, not synchronized, looped).
2. Prefer path with the highest locally assigned weight value.
3. Prefer path with the highest Local Preference attribute value.
4. Prefer locally originated prefixes (originated via the network, aggregate-address, or redistribution commands).
5. Prefer path with the shortest AS_PATH attribute length.
6. Prefer path with the lowest Origin Type (value for the Origin code), where IGP < EGP < Incomplete.
7. Prefer path with the lowest MED attribute value (provided that the first AS in the list is the same).
8. Prefer external BGP paths over internal BGP paths.

https://t.me/learningnets

9. Prefer path with the smallest IGP metric to reach the NEXT_HOP IP address.
10. Prefer path originated from the router with the lowest BGP Router ID.

Before everything else, BGP prefers paths with the highest *weight* value. Weight is Cisco-specific and is not transported along with BGP prefixes/updates. This attribute is configured locally on the router, using the command **`neighbor <IP_Address> weight 1-65535`** . As mentioned, higher-weight values are preferred and the default weight is zero for learned prefixes. Thus, among two equal prefixes with different weights, the one with the highest weight is preferred. This attribute is commonly used in scenarios where the local router has multiple uplinks and you want to prefer one uplink over another. In addition to the **`neighbor`** command, the weight attribute could be set using inbound route-map associated with the neighbor, for example:

```
route-map SET_WEIGHT
 math ip address ACCCESS_LIST
 set weight 100
!
router bgp 100
 neighbor 204.12.1.254 route-map SET_WEIGHT in
```

This method could be used to change specific prefix preference without affecting any other subnets learned from the same peer. Remember that weight manipulations only affect the way that the traffic leaves the local router.

Notice that IOS routers assign the weight value of 32768 to all locally advertised prefixes—prefixes advertised using the **`network`** or **`aggregate-address`** commands or via route redistribution. This feature ensures that all locally originated prefixes are always preferred over the same prefixes learned from the peers.

The solution for this task uses AS-PATH access-lists to match all prefixes from AS 54 and 254. The regular expressions to match all networks originated from AS 54 and 254, respectively, are **`54$`** and **`254$`** . R7 peers with R6 and R3, and we manipulate weight values as follows:

- For prefixes originated from AS 254 and received from R6, we set the weight value to 1000. This makes R7 prefer paths to AS 254 via R6, as opposed to R3.
- For prefixes originate from AS 54 and received from R3, we set the weight value to 1000 as well. As a result, R7 prefers paths to AS 54 via R3, where it should prefer paths via R9 by default.

Always remember to create a “permit” entry at the end of your route-maps, or you may unintentionally filter non-matching networks.

https://t.me/learningnets

```
R7:

no ip as-path access-list 1
no ip as-path access-list 2
ip as-path access-list 1 permit _54$
ip as-path access-list 2 permit _254$
!
route-map FROM_R6 permit 10
 match as-path 2
 set weight 1000
!
route-map FROM_R6 permit 100

!
route-map FROM_R3 permit 10
 match as-path 1
 set weight 1000
!
route-map FROM_R3 permit 100
!
router bgp 300
 neighbor 155.1.67.6 route-map FROM_R6 in
 neighbor 155.1.37.3 route-map FROM_R3 in
```

Verification

Before you start any verification, use the command **`clear ip bgp * soft`** to force R7 to refresh the information learned from its peers. By default, when you apply any new policy it does not take effect unless the new updates are received/sent. Because BGP does not use periodic updates, you may need to force an update manually. The command performs a “soft-reset”; it does not tear down BGP sessions but simply sends out BGP updates and requests route-refresh from the peers. Compared to **`clear ip bgp *`** , it causes much less load on the router’s CPU and does not disrupt traffic routing:

```
R7#clear ip bgp * soft
```

Now let’s look through the BGP table in R7. Notice the use of regex-based filter **_54$** to select only the prefixes originated in AS 54. Look at the prefixes received from R3, they are selected as “best” and marked with the “>” sign, and the weight assigned is 1000. The “losing” prefixes have the weight value of 0 (the default):

https://t.me/learningnets

```
R7#show ip bgp regexp _54$

BGP table version is 36, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   28.119.16.0/24   155.1.67.6                             0 100 200 54 i
 *                    155.1.79.9                             0 54 i
 *>                   155.1.37.3                          1000 200 54 i
 *   28.119.17.0/24   155.1.67.6                             0 100 200 54 i
 *                    155.1.79.9                             0 54 i
 *>                   155.1.37.3                          1000 200 54 i

 *   114.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                          1000 200 54 i
 *                    155.1.79.9               0             0 54 i
 *   115.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                          1000 200 54 i
 *                    155.1.79.9               0             0 54 i
 *   116.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                          1000 200 54 i
 *                    155.1.79.9               0             0 54 i
 *   117.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                          1000 200 54 i
 *                    155.1.79.9               0             0 54 i
 *   118.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                          1000 200 54 i
 *                    155.1.79.9               0             0 54 i
 *   119.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                          1000 200 54 i
 *                    155.1.79.9               0             0 54 i
```

Repeat the same check with the prefixes originated in AS 254. Notice that now R6 is selected as the upstream peer to route to these subnets:

```
R7#show ip bgp regexp 254$
BGP table version is 36, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
```

https://t.me/learningnets

```
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>  51.51.51.51/32   155.1.67.6                          1000 100 200 254 ?
 *                   155.1.37.3                             0 200 254 ?
*>  192.10.1.0       155.1.67.6                          1000 100 200 254 ?

 *                   155.1.37.3                             0 200 254 ?
 *>  205.90.31.0     155.1.67.6                          1000 100 200 254 ?
 *                   155.1.37.3                             0 200 254 ?
 *>  220.20.3.0      155.1.67.6                          1000 100 200 254 ?
 *                   155.1.37.3                             0 200 254 ?
 *>  222.22.2.0      155.1.67.6                          1000 100 200 254 ?
 *                   155.1.37.3                             0 200 254 ?
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - Local Preference

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task on R7 is removed before starting.
- Use local-preference on R6 so that traffic from AS 100 going to AS 254 transits through AS 300.

Configuration

The next attribute that is compared among equal prefixes (assuming the weight for all prefixes is the same) is Local Preference (LP). This is a well-known discretionary attribute that influences the BGP best-path selection algorithm within the scope of a single AS. The numerically higher value of the LP attribute makes BGP prefer the path over others paths with lower LP. The Local Preference attribute is carried along with the prefix through the AS (relayed across iBGP sessions) but does not leave the AS boundaries. This AS-wide propagation ensures that all routers within the single AS perform deterministic path selection.

When you manipulate the local preference, you affect the way in which traffic leaves the local AS. Local Preference is typically modified at the border of the AS, at the point of the external connection. To set the local preference for prefixes received from a peer, you must construct a route-map that matches the prefixes you want to manipulate and use the respective set command:

https://t.me/learningnets

```
route-map SET_LP
 match ip address| match ip as-path ...
 set local-preference 1000
```

The maximum value for local preference is 2^32-1, because the attribute value is 32 bits in size. Remember that by default all iBGP-learned prefixes have the Local Preference value of 100 assigned to them. This is needed to give you some space for maneuvering when lowering the local preference for some prefixes. If you want to change the default local preference value, use the command **`bgp default local-preference <value>`** . Of course, this will only affect the local preference on the router where it is configured.

In this scenario, the default path from AS 100 to AS 254 is across AS 200, based on the AS_PATH length comparison (the step discussed in the following scenario). To affect the best-path selection, we configure R6 to assign local preference of 200 to AS 254 prefixes learned from R7. This will make all routers in AS 100 prefer the exit point via R7 to reach AS 254 prefixes. Similar to the previous scenarios, we use AS-PATH access-list to match the prefixes originated in AS 254:

```
R6:

no ip as-path access-list 1
ip as-path access-list 1 permit _254$
!
route-map FROM_R7 permit 10
 match as-path 1
 set local-preference 200
!
route-map FROM_R7 permit 100
!
router bgp 100
 neighbor 155.1.67.7 route-map FROM_R7 in
```

Verification

Refresh the routing information and check the BGP tables of the routers in AS 100. Look for paths originated from AS 254, using the regexp “254$”:

https://t.me/learningnets

```
R6#clear ip bgp * soft
R7#clear ip bgp * soft
```

All BGP routers in AS 100 prefer to reach AS 254 prefixes via R6 (look at the next-hop value). Notice that R1 and R4 learned alternative paths to AS 254 via R5 and R3, respectively, but they all select the paths with LP 200:

```
R6#show ip bgp regexp _254$

BGP table version is 73, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>  51.51.51.51/32   155.1.67.7                     200
      0 300 200 254 ?*>  192.10.1.0       155.1.67.7                     200
      0 300 200 254 ?*>  205.90.31.0      155.1.67.7                     200
      0 300 200 254 ?*>  220.20.3.0       155.1.67.7                     200
      0 300 200 254 ?*>  222.22.2.0       155.1.67.7                     200
      0 300 200 254 ?
!R1#show ip bgp regexp _254$
BGP table version is 46, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>i 51.51.51.51/32   155.1.67.7               0    200
      0 300 200 254 ?
 *                   155.1.13.3                           0 200 254 ?
r>i 192.10.1.0       155.1.67.7               0    200
      0 300 200 254 ?
 r                   155.1.13.3                           0 200 254 ?
*>i 205.90.31.0      155.1.67.7               0    200
      0 300 200 254 ?
 *                   155.1.13.3                           0 200 254 ?
*>i 220.20.3.0       155.1.67.7               0    200
      0 300 200 254 ?
```

https://t.me/learningnets

```
 *                    155.1.13.3                             0 200 254 ?
*>i 222.22.2.0       155.1.67.7               0    200
      0 300 200 254 ?
 *                    155.1.13.3                             0 200 254 ?
! R4#show ip bgp regexp _254$
BGP table version is 54, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   51.51.51.51/32   155.1.45.5                             0 200 254 ?
*>i                  155.1.67.7               0    200
      0 300 200 254 ?
 r   192.10.1.0       155.1.45.5                             0 200 254 ?
r>i                  155.1.67.7               0    200
      0 300 200 254 ?
 *   205.90.31.0      155.1.45.5                             0 200 254 ?
*>i                  155.1.67.7               0    200
      0 300 200 254 ?
 *   220.20.3.0       155.1.45.5                             0 200 254 ?
*>i                  155.1.67.7               0    200
      0 300 200 254 ?
 *   222.22.2.0       155.1.45.5                             0 200 254 ?
*>i                  155.1.67.7               0    200
      0 300 200 254 ?
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - AS-Path Prepending

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task on R6 is removed before starting.
- Using AS-Path Prepending, configure AS 200 so that traffic from AS 100 going to AS 254 transits AS 300.

Configuration

When two or multiple paths for the exact same prefix have equal weights and local preference values, BGP process will prefer the prefix originated locally, which is the prefix advertised via the `network` , `aggregate-address` , or `redistribution` commands. In IOS routers, this step is usually unnecessary, because the locally originated prefixes have a default weight value of 32768, overriding all other steps of the BGP bestpath selection process.

The next step in the process is selecting the prefix with the shortest AS_PATH. The length of this attribute is probably the best approximation of the classic IGP metric when mapping this concept to BGP. This could be directly compared to the hop count concept used in RIP. Here is the procedure for computing the AS_PATH length:

1. For every AS_SEQUENCE element, every AS in the list counts as one and the resulting sum is the number of entries in the path.

https://t.me/learningnets

2. If the prefix was a result of summarization, you may encounter the AS_SET elements in the AS_PATH. The length of AS_SET is assumed to be “1”. This is because summarization conceals topology information, and the summarized prefixes may have had different AS_PATH lengths.
3. If the prefix has been originated within the BGP Confederation, its AS_PATH attribute may contain any of BGP_CONFED_SEQUENCE or BGP_CONFED_SET sub-attributes. The second sub-attribute appears when you summarize prefixes inside confederation. Each of these elements does not count when computing the AS PATH length.

To use this step to influence best path selection, you should use the AS_PATH prepending procedure. This procedure applies only to eBGP sessions, which is when advertising prefixes to another AS and the local AS number is prepended in front of the AS_PATH attribute the number of times specified. The syntax to perform AS_PATH prepending is as follows, assuming that the local AS number is 100:

```
route-map PREPEND
 match ...
 set as-path prepend 100 100 100
!
router bgp 100
 neighbor 54.1.1.254 route-map PREPEND out
```

The example above applies the local AS number three times, instead of just one, to the prefixes matching the route-map criterion. Notice that even though you apply attribute manipulation in outbound direction, it affects the way that the external systems send traffic to your AS. Manipulating the AS_PATH length is the common way to influence the incoming traffic paths to the local AS and is widely used on the Internet. Usually, the prefixes advertised on the least-preferred inbound link have the local AS path number prepended three or more times. This ensures that any further manipulations will not make those prefixes preferred over the subnets advertised across the “primary” entry point. Remember that the remote AS may change your policy by applying the local preference attribute manipulations. However, that process will only affect path selection within the single AS, not globally on the Internet.

The AS_PATH comparison step could be disabled by issuing the command **`bgp bestpath as-path ignore`** . This command is hidden under BGP configuration CLI, so you must type it without using the “?” help sign.

```
R3:
```

https://t.me/learningnets

```
ip as-path access-list 1 permit _254$
!
route-map TO_R1 permit 10
 match as-path 1
 set as-path prepend 200 200 200
!
router bgp 200
 neighbor 155.1.13.1 route-map TO_R1 out
```

R5:

```
ip as-path access-list 1 permit _254$
!
route-map TO_R4 permit 10
 match as-path 1
 set as-path prepend 200 200 200
!
router bgp 200
 neighbor 155.1.45.4 route-map TO_R4 out
```

Verification

Clear the BGP session state on R3 and R5, and check the BGP tables in AS 100:

```
R5#clear ip bgp * soft
 R3#clear ip bgp * soft
```

Notice that R6 has only one set of prefixes toward AS 254, via AS 300. This is because R4 and R1 received the paths across AS 200 prepended and thus prefer R6's path via AS 300 a lower AS_PATH count. Because R1 and R4 select R6's path as best, they will not re-advertise that path back to R6. You can see the prepended paths in the BGP tables of R1 and R4:

```
R6#show ip bgp regexp _254$
BGP table version is 95, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  51.51.51.51/32   155.1.67.7                             0 300 200 254 ?
```

https://t.me/learningnets

```
 *>  192.10.1.0       155.1.67.7                             0 300 200 254 ?
 *>  205.90.31.0      155.1.67.7                             0 300 200 254 ?
 *>  220.20.3.0       155.1.67.7                             0 300 200 254 ?
 *>  222.22.2.0       155.1.67.7                             0 300 200 254 ?
!R4#show ip bgp regexp _254$
BGP table version is 92, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 51.51.51.51/32   155.1.67.7               0    100      0 300 200 254 ?
 *                    155.1.45.5                              0 200 200 200 200
 254 ?
 r>i 192.10.1.0       155.1.67.7               0    100      0 300 200 254 ?
 r                    155.1.45.5                              0 200 200 200 200
 254 ?
 *>i 205.90.31.0      155.1.67.7               0    100      0 300 200 254 ?
 *                    155.1.45.5                              0 200 200 200 200
 254 ?
 *>i 220.20.3.0       155.1.67.7               0    100      0 300 200 254 ?
 *                    155.1.45.5                              0 200 200 200 200
 254 ?
 *>i 222.22.2.0       155.1.67.7               0    100      0 300 200 254 ?
 *                    155.1.45.5                              0 200 200 200 200
 254 ?
!R1#show ip bgp regexp _254$
BGP table version is 90, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 51.51.51.51/32   155.1.67.7               0    100      0 300 200 254 ?
 *                    155.1.13.3                              0 200 200 200 200
 254 ?
 r>i 192.10.1.0       155.1.67.7               0    100      0 300 200 254 ?
 r                    155.1.13.3                              0 200 200 200 200
 254 ?
 *>i 205.90.31.0      155.1.67.7               0    100      0 300 200 254 ?
 *                    155.1.13.3                              0 200 200 200 200
 254 ?
```

https://t.me/learningnets

```
*>i 220.20.3.0       155.1.67.7               0    100      0 300 200 254 ?
*                    155.1.13.3                            0 200 200 200 200
254 ?
*>i 222.22.2.0       155.1.67.7               0    100      0 300 200 254 ?
*                    155.1.13.3                            0 200 200 200 200
254 ?
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - Origin Code

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task on R3 and R5 is removed before starting.
- Using the Origin Code attribute, configure AS 200 so that traffic from AS 100 going to AS 254 transits the link between R4 and R5.

Configuration

If several paths exist for exact same BGP prefix with equal weight, local preference, and AS_PATH length, the bestpath selection process compares route Origin Code. This attribute is well known and mandatory and is set by the prefix originator. The allowed values are as follows:

- **IGP** meaning that the route was originated using the `network` or `aggregate-address` commands. It appears as **i** in BGP table output.
- **EGP** meaning that the prefix was received from an EGP peer (legacy). You probably won't see this Origin value in any modern router, but it can be manually configured.
- **Incomplete** meaning that the source could not be determined. This value is assigned to the prefixes redistributed into BGP.

This attribute is rarely used to influence the path selection because of its inflexibility. Nevertheless, BGP always accounts for this step. Origin **IGP** is preferred over **EGP**, and the latter is preferred over **Incomplete**. This represents the level of trust that

https://t.me/learningnets

BGP assigns to various information sources. You can set the attribute using the route-map level command `set origin [igp|egp|incomplete]` . On latest IOS codes, changing the origin code to **EGP** is a hidden command.

In this task, we configure the border routers in AS 200 (R3 and R5) to impose different Origin attributes on the links between R1 and R3, and R4 and R5. The prefixes advertised to R1 have the Origin value of **Incomplete**, and the prefixes advertised to R4 have the Origin value of **IGP**.

```
R3:
no ip as-path access-list 1
ip as-path access-list 1 permit _254$
!
route-map TO_R1 permit 10
 match as-path 1
 set origin incomplete
!
route-map TO_R1 permit 100
!
router bgp 200
 neighbor 155.1.13.1 route-map TO_R1 out

R5:

no ip as-path access-list 1
ip as-path access-list 1 permit _254$
!
route-map TO_R4 permit 10
 match as-path 1
 set origin igp
!
route-map TO_R4 permit 100
!
router bgp 200
 neighbor 155.1.45.4 route-map TO_R4 out
```

Verification

Clear BGP sessions and check the BGP tables of R1 and R4. R1 prefers the paths learned from R4, because they have the origin code of **IGP**. R4 does not see any AS 254 paths learned from R1, because R1 suppresses their advertisement because it is using R4 as its best path:

https://t.me/learningnets

```
R5#clear ip bgp * soft
 R3#clear ip bgp * soft

!R1#show ip bgp regexp _254$
BGP table version is 112, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>i 51.51.51.51/32
155.1.45.5               0    100      0 200 254 i
 *                   155.1.13.3                             0 200 254?
 r>i 192.10.1.0 155.1.45.5               0    100      0 200 254 i
 r                   155.1.13.3                             0 200 254?
 *>i 205.90.31.0 155.1.45.5               0    100      0 200 254 i
 *                   155.1.13.3                             0 200 254?
 *>i 220.20.3.0 155.1.45.5               0    100      0 200 254 i
 *                   155.1.13.3                             0 200 254?
 *>i 222.22.2.0 155.1.45.5               0    100      0 200 254 i
 *                   155.1.13.3                             0 200 254?
!R4#show ip bgp regexp _254$

BGP table version is 129, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  51.51.51.51/32   155.1.45.5                             0 200 254 i
 *>  192.10.1.0       155.1.45.5                             0 200 254 i
 *>  205.90.31.0      155.1.45.5                             0 200 254 i
 *>  220.20.3.0       155.1.45.5                             0 200 254 i
 *>  222.22.2.0       155.1.45.5                             0 200 254 i
```

The situation at R6 is a bit more complex. It marks the paths via R4 as being optimal, because the paths received from R7 have longer AS_PATH length:

```
R6#show ip bgp regexp _254$
BGP table version is 108, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
```

https://t.me/learningnets

```
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>i 51.51.51.51/32
155.1.45.5               0    100      0 200 254
 i
 *                   155.1.67.7                             0 300 200 254 ? r>i 192.10.1.0
155.1.45.5               0    100      0 200 254
 i
 r                   155.1.67.7                             0 300 200 254 ? *>i 205.90.31.0
155.1.45.5               0    100      0 200 254
 i
 *                   155.1.67.7                             0 300 200 254 ? *>i 220.20.3.0
155.1.45.5               0    100      0 200 254
 i
 *                   155.1.67.7                             0 300 200 254 ? *>i 222.22.2.0
155.1.45.5               0    100      0 200 254
 i
 *                   155.1.67.7                             0 300 200 254 ?
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - MED

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task on R3 and R5 is removed before starting.
- Using MED, configure AS 200 so that traffic from AS 100 going to AS 54 transits the link between R4 and R5.
- Shutdown R6's BGP peering with R7.

Configuration

If several paths exist for exact same BGP prefix with equal weight, local preference, AS_PATH length, and Origin Code, BGP will compare the next-less-influential attribute which is MED or Multi Exit Discriminator. This attribute is optional and non-transitive, and its value is an unsigned 32 integer. It is often called “metric” and is used on eBGP peering links to select the entry point to the neighboring AS. The AS that originates the prefixes may attach different metric values to them on different exit points from the AS. This gives the neighboring AS a hint to select the best path based on the smallest metric value, because all other attributes usually match (weight, LP, AS_PATH length).

The MED attribute is non-transitive, so the receiving AS will not propagate it across its AS borders. However, the receiving AS may reset the metric value if it wants. Cisco BGP implementation automatically assigns the value of the MED attribute based on the IGP metric value for the locally originated prefixes. The explanation for

https://t.me/learningnets

this follows.

Most often, BGP prefixes are advertised by the border eBGP speakers. If there are redundant connections between two ASs, multiple BGP speakers may originate the same prefix, attaching the MED attribute copied from the IGP metric of the advertised prefix. The neighboring AS may then apply the bestpath selection for the peer's prefixes based on the smallest metric value. This makes sense if the following is true:

1. The peer (originator) AS uses a single IGP across its network; that is, metric values are numerically comparable.
2. MED values are only compared for the prefixes coming from the same AS.
3. Prefixes were originated from the peer's AS.

Based on these assumptions, the MED attribute simply gives the neighboring AS a hint to select the prefixes with the shortest IGP metric value in the peer's AS. This routing behavior is called "cold potato routing," in contrast to the "hot potato" routing model. The "hot potato" model is effectively in use when MED values are *not* considered during the bestpath selection. When MED values are filtered or are the same among all paths, the router will select the path with the lowest IGP cost for its next-hop (if the prefixes are of the same type, for example both learned via iBGP, comparing IGP costs is the next step in BGP bestpath selection process). This means selecting the path through the closest exit point, based on the IGP metrics of the local AS. This is analogous to a group of people trying to get rid of a "hot potato" as fast as possible. The "cold potato" routing model considers the metric hint information from the adjacent AS to make the final routing decision.

Often you may see the MED attribute used to influence the exit point selection for prefixes not originated in the local AS. This is perfectly legal, but you must assign some artificial metric values to these prefixes because they don't belong to the IGP from local AS. Remember that this sort of path attribute manipulation only affects the paths chosen by the directly connected neighboring AS. Finally, the MED attribute is optional and thus may not be present or attached to all updates/prefixes. By default, the BGP process assumes the MED value of zero for such prefixes, which will make them more preferred during the selection based on metric. If you want to change this behavior, you must enter the command `bgp bestpath med missing-as-worst` . This instructs the local router to impose the maximum MED value on the prefixes that do not carry the MED attribute, making these prefixes least preferable. The use of this logic was specified in earlier drafts of the BGP specification, whereas the most recent instructs to assign the MED value of zero to the prefixes lacking the metric attribute.

In our scenario, we configure AS 200 border routers (R3 and R5) peering with AS 100 to assign metric values for prefixes learned from AS 54. The metrics are

https://t.me/learningnets

assigned so that exit point through R3 is less preferred (higher metric). Notice that we configure both R3 and R5, whereas it is possible to configure just R3, because routes advertised by R5 will have the default MED value of 0. Note that R6's peering with AS 300 will be disabled for this example. R6 is peering with AS 300, which also has a peering with AS 54. MED is only compared by default when it is received from the same AS. If we did not disable R6's peering to AS 300, R6 would receive a MED of 0 from R7 in AS 300, and would also receive MED of 100 from R1. R6 would not compare these two MED values because they come from two different AS's.

```
R3:
no ip as-path access-list 1
ip as-path access-list 1 permit _54$
!
route-map TO_R1 permit 10
 match as-path 1
 set metric 1000
!
route-map TO_R1 permit 100
!
router bgp 200
 neighbor 155.1.13.1 route-map TO_R1 out

R5:
no ip as-path access-list 1
ip as-path access-list 1 permit _54$
!
no route-map TO_R4
route-map TO_R4 permit 10
 match as-path 1
 set metric 100
!
route-map TO_R4 permit 100
!
router bgp 200
 neighbor 155.1.45.4 route-map TO_R4 out

R6:

router bgp 100
 neighbor 155.1.67.7 shutdown
```

https://t.me/learningnets

Verification

Clear the BGP peering sessions softly and check the BGP tables of AS 100 routers. Notice that R4 only has paths to AS 54 prefixes via R5. R1 received the paths via R3 with a MED of 1000 and the same paths via R4 via a MED of 100. R1 prefers the paths via R4 and selects them as best. Because R6's peering to AS 300 is disabled, it receives the advertisement from its route-reflector, R1, with a next hop of R5. This is R1's best path out of the AS. AS 100 is correctly choosing the exit point with the lowest MED:

```
R3#clear ip bgp * soft
R5#clear ip bgp * soft
!R4#show ip bgp regex _54$
BGP table version is 150, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>  28.119.16.0/24 155.1.45.5
100
             0 200 54 i *>  28.119.17.0/24 155.1.45.5 100
             0 200 54 i *>  114.0.0.0 155.1.45.5 100
             0 200 54 i *>  115.0.0.0 155.1.45.5 100
             0 200 54 i *>  116.0.0.0 155.1.45.5 100
             0 200 54 i *>  117.0.0.0 155.1.45.5 100
             0 200 54 i *>  118.0.0.0 155.1.45.5 100
             0 200 54 i *>  119.0.0.0 155.1.45.5 100
             0 200 54 i
!R1#show ip bgp regexp _54$
BGP table version is 133, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>i 28.119.16.0/24   155.1.45.5             100
    100      0 200 54 i
 *                   155.1.13.3            1000             0 200 54 i
*>i 28.119.17.0/24   155.1.45.5             100
    100      0 200 54 i
```

https://t.me/learningnets

```
 *                    155.1.13.3            1000             0 200 54 i
*>i 114.0.0.0         155.1.45.5             100
    100      0 200 54 i
 *                    155.1.13.3            1000             0 200 54 i
*>i 115.0.0.0         155.1.45.5             100
    100      0 200 54 i
 *                    155.1.13.3            1000             0 200 54 i
*>i 116.0.0.0         155.1.45.5             100
    100      0 200 54 i
 *                    155.1.13.3            1000             0 200 54 i
*>i 117.0.0.0         155.1.45.5             100
    100      0 200 54 i
 *                    155.1.13.3            1000             0 200 54 i
*>i 118.0.0.0         155.1.45.5             100
    100      0 200 54 i
 *                    155.1.13.3            1000             0 200 54 i
*>i 119.0.0.0         155.1.45.5             100
    100      0 200 54 i
 *                    155.1.13.3            1000             0 200 54 i
!R6#show ip bgp regexp _54$
BGP table version is 124, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>i 28.119.16.0/24155.1.45.5
100
    100      0 200 54 i *>i 28.119.17.0/24155.1.45.5 100
    100      0 200 54 i *>i 114.0.0.0155.1.45.5 100
    100      0 200 54 i *>i 115.0.0.0155.1.45.5 100
    100      0 200 54 i *>i 116.0.0.0155.1.45.5 100
    100      0 200 54 i *>i 117.0.0.0155.1.45.5 100
    100      0 200 54 i *>i 118.0.0.0155.1.45.5 100
    100      0 200 54 i *>i 119.0.0.0155.1.45.5 100
    100      0 200 54 i
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - Always Compare MED

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task on R3, R5, and R6 is removed before starting.
- Ensure that R6's peering to R7 is not shut down.
- Create a new Loopback1 interface on both R6 and R7 with the IP address 1.2.3.4/32 and advertise it into BGP on both R6 and R7.
- Using just the MED attribute, configure the network so that traffic from AS 200 destined to Loopback1 is received by R7.

Configuration

As discussed previously, comparing automatically generated MED attributes makes sense only if all prefixes are originated from directly connected ASs. However, setting the artificial MED values is often used by an adjacent AS to hint the exit point to its peers. Now imagine a situation when the local AS (A) peers with two other ASs (B and C), and the peers have some sort of “backdoor” link between them, such as running an IGP on this link. Upon an agreement, the peers may decide to “share” their entry points, so that “A” may send traffic to a subset of “B”’s prefixes across “C” and vice versa. This could be done using the proper manipulation of MED attribute because the prefixes appear to be internal to both ASs, but classic BGP procedure does not compare MEDs for prefixes that were received from different ASs.

This limitation could be disabled by using command `bgp always-compare-med` under BGP configuration mode. This instructs BGP code to ignore the first AS# in the

https://t.me/learningnets

AS_PATH attributes when comparing MED values. Of course, this is only possible if the AS_PATH lengths are the same. Using MED for exit point selection to multiple adjacent ASs is becoming common, although it's against the idea of using MED as a reflection of the internal IGP cost for a prefix. Essentially, this procedure assumes that the adjacent systems have an agreement about MED values assignment, and the local AS accepts their policy.

Throughout the history of BGP implementation, it has become apparent that using MED in bestpath selection may sometimes result in unstable routing tables or routing loops. There are many different scenarios in which such issues may arise. Particularly, Cisco's BGP implementation suffered from non-deterministic MED-based path selection. What that means is that the result of best-path selection may depend on the order in which the local speaker receives BGP prefixes. Specifically, IOS code was ordering BGP prefixes based on their age and compared them in pairs, starting with the newer/latest received prefixes. This temporal dependency was intended to make routing tables more stable by giving more preference for older information. However, when the MED issues became apparent, Cisco implemented a workaround that removed temporal dependency in MED-based computations. The resulting procedure became deterministic and no longer depends on the order in which the prefixes are received. To enable this deterministic mode, use the command **`bgp bestpath deterministic-med`** . This feature is not enabled by default because many older IOSs still use the previous selection procedure, and combining those two in the same AS may result in a routing loop.

In this task, both AS 300 and AS 100 advertise the same prefix (artificially created). Border routers in AS 300 (R7) and AS 100 (R1 and R4) are configured to set metrics so that the exit point between R3 and R7 is used to reach this subnet. To make AS 200 account for metrics from different ASs, we enable the always-compared-med feature in all routers of AS 200.

R1:

```
ip prefix-list LOOPBACK1 permit 1.2.3.4/32
!
route-map TO_R3 permit 10
 match ip address prefix-list LOOPBACK1
 set metric 1000
!
route-map TO_R3 permit 100
!
router bgp 100
 neighbor 155.1.13.3 route-map TO_R3 out
```

R2:

https://t.me/learningnets

```
router bgp 200
 bgp always-compare-med
```

R3:

```
router bgp 200
 bgp always-compare-med
```

R4:

```
ip prefix-list LOOPBACK1 permit 1.2.3.4/32
!
route-map TO_R5 permit 10
 match ip address prefix-list LOOPBACK1
 set metric 1000
!
route-map TO_R5 permit 100
!
router bgp 100
 neighbor 155.1.45.5 route-map TO_R5 out
```

R5:

```
router bgp 200
 bgp always-compare-med
```

R6:

```
interface Loopback1
 ip address 1.2.3.4 255.255.255.255
!
router bgp 100
 network 1.2.3.4 mask 255.255.255.255
```

R7:

```
interface Loopback1
 ip address 1.2.3.4 255.255.255.255
!
ip prefix-list LOOPBACK1 permit 1.2.3.4/32
!
route-map TO_R3 permit 10
 match ip address prefix-list LOOPBACK1
 set metric 100

route-map TO_R3 permit 100
!
router bgp 300
 network 1.2.3.4 mask 255.255.255.255
 neighbor 155.1.37.3 route-map TO_R3 out
```

https://t.me/learningnets

```
R8:

router bgp 200
 bgp always-compare-med
```

Verification

Clear BGP session on all routers using the command `clear ip bgp * soft` . You may use the `send *` method on the access-server of your rack to accomplish this faster. Now check the BGP table for prefix 1.2.3.4 in R3. The prefix with the MED value of 100 is selected as best, even though paths were received from different ASs.

```
R3#show ip bgp 1.2.3.4
BGP routing table entry for 1.2.3.4/32, version 2
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     7          8          9
  Refresh Epoch 1 100
    155.1.13.1 from 155.1.13.1 (150.1.1.1)      Origin IGP, metric 1000
, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1 300
    155.1.37.7 from 155.1.37.7 (150.1.7.7)      Origin IGP, metric 100, localpref 100, valid, external,
best

      rx pathid: 0, tx pathid: 0x0
```

Check the same prefix in R5. The BGP table output here is similar to R3, and again the path through R7 is selected as the best. Notice the **(metric 3584)** field in the output. It has nothing to do with the MED value; it's the IGP cost to reach the next-hop for this prefix.

```
R5#show ip bgp 1.2.3.4
BGP routing table entry for 1.2.3.4/32, version 2
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     6          7
  Refresh Epoch 1 300
    155.1.37.7 (metric 3584) from 155.1.0.3 (150.1.3.3)      Origin IGP, metric 100
, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1 100
```

https://t.me/learningnets

```
    155.1.45.4 from 155.1.45.4 (150.1.4.4)       Origin IGP, metric 1000
, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
```

R2 received two paths from its route reflectors. Because both reflectors elected the path via R7 as best, both prefixes in R2's BGP table use R7 as the exit point out of the AS. R2 selected the second path based on the lowest originating router ID (we will discuss this selection step in a separate task).

```
R2#show ip bgp 1.2.3.4
BGP routing table entry for 1.2.3.4/32, version 2
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     3
  Refresh Epoch 1 300
    155.1.37.7 (metric 3072) from 155.1.0.5 (150.1.5.5)       Origin IGP, metric 100
, localpref 100, valid, internal
      Originator: 150.1.3.3, Cluster list: 150.1.5.5
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  300
    155.1.37.7 (metric 3072) from 155.1.23.3 (150.1.3.3)       Origin IGP, metric 100
, localpref 100, valid, internal, best

      rx pathid: 0, tx pathid: 0x0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - AS-Path Ignore

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task between AS 200, 300, and 100 is removed before starting.
- Ensure that traffic from AS 200 to AS 54 prefixes transit through AS 100.
- Do not use AS-PATH prepending to accomplish this.

Configuration

Ignoring AS_PATH length comparison is optional and could be enabled using a special hidden command. Using this feature in production is *NOT* recommended because it may severely affect routing table stability. However, if you need it just for some non-standard tweak of your BGP path selection, use the command `bgp bestpath as-path ignore` to activate this feature. BGP will automatically skip AS_PATH length comparison and proceed to comparing the Origin codes, MED attribute, and the IGP costs for NEXT_HOPs.

One case in which you may actually need this feature is BGP confederations. Remember that BGP_CONFED_SEQUENCE and BGP_CONFED_SET do not count when computing the AS_PATH length. Therefore, in BGP confederations you may see suboptimal paths being elected simply based on the AS_PATH attribute carried from the external ASs (internal path lengths are ignored). By disabling the AS_PATH comparison, you may force the local speakers to use the “hot potato” routing model, taking into account the IGP cost to reach the prefix NEXT_HOP, that

https://t.me/learningnets

is only provided if all prefixes have the same Origin Code and MED attribute values, which could be enforced by border BGP routers.

In this scenario, we modify the Origin Code attribute for paths injected into AS 200 through the peering connection R7-R3, R1-R3, R4-R5, and R8-R10. All routes in AS 200 have AS_PATH length comparison disabled and therefore prefer the paths to AS 54 learned from AS 100, even though those have longer AS_PATHs. R8 has a direct EBGP connection to AS 54. On R8's peering, we will set origin code inbound so that all routes originating in AS 54 and sent to R8 directly will have the Origin attribute set to incomplete. The same will happen on R7's peering with R3, the Origin attribute will be set to incomplete from R7 to R3 so that all routes originating in AS 54 that are sent from AS 300 to AS 200 will be marked as incomplete. This leaves both R1's and R4's peering with AS 200. On both of these peerings, we will set the Origin Code attribute to igp so that either one of these exit points is preferred over all of the other ones that have been marked as incomplete.

```
R1:
no ip as-path access-list 1
ip as-path access-list 1 permit _54$
!
route-map TO_R3 permit 10
 match as-path 1
 set origin igp
!
route-map TO_R3 permit 100
!
router bgp 100
 neighbor 155.1.13.3 route-map TO_R3 out

R2:
router bgp 200
 bgp bestpath as-path ignore

R3:
router bgp 200
 bgp bestpath as-path ignore

R4:
no ip as-path access-list 1
ip as-path access-list 1 permit _54$
!
route-map TO_R5 permit 10
 match as-path 1
 set origin igp
!
```

https://t.me/learningnets

```
route-map TO_R5 permit 100
!
router bgp 100
 neighbor 155.1.45.5 route-map TO_R5 out
```

R5:

```
router bgp 200
 bgp bestpath as-path ignore
```

R7:

```
no ip as-path access-list 1
ip as-path access-list 1 permit _54$
!
route-map TO_R3 permit 10
 match as-path 1
 set origin incomplete
!
route-map TO_R3 permit 100
!
router bgp 300
 neighbor 155.1.37.3 route-map TO_R3 out
```

R8:

```
no ip as-path access-list 1
ip as-path access-list 1 permit _54$
!
route-map TO_R10 permit 10
 match as-path 1
 set origin incomplete
!
route-map TO_R10 permit 100
!
router bgp 200
 bgp bestpath as-path ignore
 neighbor 155.1.108.10 route-map TO_R10 in
```

Verification

https://t.me/learningnets

```
R7#clear ip bgp * soft
R8#clear ip bgp * soft
R4#clear ip bgp * soft
R1#clear ip bgp * soft
```

After clearing the BGP sessions, inspect the BGP tables of R5 and R8. Notice that both routers select the paths with the longer AS_PATHs, based on their Origin Code. Notice that both devices have their next hops set to R1 and R4, respectively, both of AS 100's border routers with AS 200. This illustrates that AS 200 is correctly sending traffic to AS 100 even though their AS_PATH is longer:

```
R3#show ip bgp regexp _54$
BGP table version is 44, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 28.119.16.0/24   155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
 *                    155.1.37.7                               0 300 54 ?
 * i 28.119.17.0/24   155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
 *                    155.1.37.7                               0 300 54 ?
 * i 114.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
 *                    155.1.37.7                               0 300 54 ?
 * i 115.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
 *                    155.1.37.7                               0 300 54 ?
 * i 116.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
 *                    155.1.37.7                               0 300 54 ?
 * i 117.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
 *                    155.1.37.7                               0 300 54 ?
 * i 118.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
 *                    155.1.37.7                               0 300 54 ?
 * i 119.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 *>                   155.1.13.1                               0 100 300 54 i
```

https://t.me/learningnets

```
 *                    155.1.37.7                             0 300 54 ?
!R8#show ip bgp regexp _54$
BGP table version is 56, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.45.4               0    100      0 100 300 54 i
 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10             0             0 54 ?
 *>i 28.119.17.0/24   155.1.45.4               0    100      0 100 300 54 i
 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10             0             0 54 ?
 *>i 114.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10                           0 54 ?
 *>i 115.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10                           0 54 ?
 *>i 116.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10                           0 54 ?
 *>i 117.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10                           0 54 ?
 *>i 118.0.0.0        155.1.45.4               0    100      0 100 300 54 i
 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10                           0 54 ?
 *>i 119.0.0.0        155.1.45.4               0    100      0 100 300 54 i

 * i                  155.1.13.1               0    100      0 100 300 54 i
 *                    155.1.108.10                           0 54 ?
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - Router-IDs

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task is removed.
- Add a new Loopback1 on R6 and R4 of 1.2.3.4/32 and advertise it into BGP.
- Modify the BGP router-id in AS 100 as necessary so that R1 selects R6 as its best path to reach 1.2.3.4/32.

Configuration

The next few steps after the MED attribute processing include:

1. Select external paths over internal, because the information should be more actual. Here, external paths are prefixes learned via eBGP sessions on the router, whereas internal paths are learned via iBGP sessions.
2. Prefer the path with the minimum IGP metric to reach the NEXT_HOP IP address. This is a natural selection step, because it attempts to pick up the closest exit point based on the local AS IGP metrics. Remember that exit point selection based on MED (adjacent AS metrics) is preferred over this step. Routing based on the closest exit point in terms of local IGP metric is called “hot potato” routing. In contrast, routing based on the MED values (the metric values advertised by the adjacent AS) is called “cold potato” routing. If all prefixes have the same IGP cost to reach their NEXT_HOPs, the BGP process may consider inserting all of them into RIB,

https://t.me/learningnets

implementing equal-cost multipath load-balancing. This feature is enabled by using the command `maximum-paths [ibgp]` under BGP configuration mode. Specify the `ibgp` keyword if you want to load balance among the paths learned via iBGP.

3. Among the prefixes learned from different eBGP peers, prefer the oldest one (most stable) to minimize route flapping.
4. Use BGP Router IDs of the advertising (peering) routers as tie-breakers for the best-path selection process. The path advertised by the peer with the lowest router ID is preferred.

In this scenario, the default BGP Router IDs for R4 and R6 are based on their Loopback0 IP address value. This makes R1 prefer R4 over R6 as its best path for Loopback1 prefix, because all other criteria are the same. To change this, we configure R6 with an artificially lower Router ID value. Remember that changing a router's BGP router ID will hard-reset all active BGP sessions.

```
R4:
interface Loopback1
ip address 1.2.3.4 255.255.255.255
!
router bgp 100
 network 1.2.3.4 mask 255.255.255.255

R6:

interface Loopback1
ip address 1.2.3.4 255.255.255.255
!
router bgp 100
 bgp router-id 6.6.6.6
 network 1.2.3.4 mask 255.255.255.255
```

Verification

Check the BGP table of R1 before you change R6's router ID:

```
R1#show ip bgp 1.2.3.4$
BGP routing table entry for 1.2.3.4/32, version 63
Paths: (2 available, best #1, table default)
Flag: 0x840
  Advertised to update-groups: (Pending Update Generation)
     2          3
  Refresh Epoch 1
  Local, (Received from a RR-client)
```

https://t.me/learningnets

```
    155.1.146.4 from 155.1.146.4 (150.1.4.4
)       Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  Local, (Received from a RR-client)    155.1.146.6 from 155.1.146.6 (150.1.6.6
)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
```

The path learned via R4 is preferred. This is done based on the Router ID of R4, because all other attributes are equal (weight, LP, AP_PATH, Origin, MED, iBGP prefixes), including the IGP cost to reach the next-hops (directly connected in this case):

```
Rack1R1#show ip route 155.1.146.4
Routing entry for 155.1.146.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks: * directly connected, via GigabitEthernet1.146

  Route metric is 0, traffic share count is 1
```

Look at R1's BGP table again, after you have changed the router ID in R6. Now the path is preferred via R6, because it has the lowest Router ID:

```
R1#show ip bgp regexp _54$

BGP routing table entry for 1.2.3.4/32, version 64
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     2          3
  Refresh Epoch 1
  Local, (Received from a RR-client)    155.1.146.6 from 155.1.146.6 (6.6.6.6
)       Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  Local, (Received from a RR-client)    155.1.146.4 from 155.1.146.4 (150.1.4.4
)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
```

https://t.me/learningnets

Note that if a path contains route reflector attributes (Originator-ID, Cluster-ID), the originator ID is substituted for the router ID in the path selection process.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - DMZ Link Bandwidth

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Advertise Loopback0 interface of routers in AS 100 into BGP.
- Enable a new eBGP peering between R5 and R1 using their directly connected DMVPN Tunnel interface.
- Configure the DMVPN Tunnel interface on R5 with a bandwidth of 50 Mbps.
- Modify the configuration of AS 200 routers so that R5 load-balances traffic destined to AS 100 Loopback0 prefixes proportional to the bandwidth of the links connecting R5 to R4 and R5 to R1.

Configuration

As mentioned in the previous tasks, local BGP process may implement equal-cost load-balancing to the paths that:

- Have the same set of path attributes up to the MED (weight, Local Preference, Origin, MED).
- Are of the same type (both learned via iBGP or eBGP).
- Have the same IGP cost to reach their NEXT_HOP IP address.

https://t.me/learningnets

If the above conditions are met and `maximum-paths [ibgp]` is configured under the BGP process, BGP will install multiple equal-cost routes into the local RIB and use them for load-balancing. We call the above condition as load-balancing conditions for BGP.

BGP also implements the unique unequal-cost load balancing feature. As you remember, unequal-cost load balancing could not be implemented easily with any IGP. The protocol needs a way to ensure that all alternative paths are loop-free. So far only EIGRP support this feature, because all alternate unequal cost paths are guaranteed to be loop free by the virtue of feasible successor property. As for BGP, it ensures loop-free property for any routes learned via eBGP, based on the duplicate AS number detection. Thus, it is possible to implement unequal-cost load-balancing in BGP toward the prefixes learned from other ASs.

This feature is called DMZ Link Bandwidth in IOS. The rationale behind this name is that load-balancing is based on the bandwidth of the links connecting the border BGP peers to their neighbors. Here is how it works for a single router with multiple eBGP peering links:

1. You enable the feature on a border BGP router using the command `bgp dmzlink-bw`. With this command enabled, the BGP process will instruct the data plane to load-balance based on the bandwidth of the links used to connect to the external BGP peers. To select the links that are to be used for load-balancing, you configure the respective BGP peers using the command `neighbor <IP> dmzlink-bw`. The BGP process will consider the bandwidth on the links connecting to those peers when doing the unequal cost load-balancing. In Cisco terminology, those links are called the DMZ Links. The bandwidth is computed based on the `bandwidth` command configured on the respective interfaces, or based on the default administrative bandwidth.
2. You enable the classic BGP equal-cost load-balancing using the command `maximum-paths` under the local BGP process. Now, assuming that you received the same prefix from multiple peers and all paths satisfy the BGP load-balancing conditions defined above, the BGP process will insert them into RIB and assign load-balancing weights proportional to the interface bandwidth values (DMZ link bandwidth).

https://t.me/learningnets

This seems to be simple enough. Now what if you have multiple BGP border peers in your AS, each having just one uplink? Is it possible to implement an AS-wide load-balancing scheme based on the bandwidth of the upstream links? That is, it would be beneficial if every router learning multiple paths to the same prefix across iBGP links would load-balance toward them based on the bandwidth of the link where they were received. Cisco IOS allows for such implementation, using the following algorithm:

1. When the DMZ Link bandwidth feature is enabled in the border BGP routers for the specific peers, the interface bandwidth value is copied into a new extended community attribute associated with the prefixes received from those eBGP peers. Thus, every prefix received on the eBGP peering link will carry the link's bandwidth as a special extended community attribute, if the link is enabled for the DMZ Link bandwidth feature. Remember that you need two commands in the border peers: **`bgp dmzlink-bw`** and **`neighbor <IP> dmzlink-bw .`**
2. All BGP speakers in the AS should be configured to exchange extended communities across the iBGP peering links. This allows all internal BGP speakers to learn the bandwidth of the external link used to reach the prefixes. Use the command **`neighbor <IP> send-community extended`** to accomplish this.
3. Provided that an internal BGP speaker has both **`bgp maximum-path ibgp`** and **`bgp dmzlink-bw`** commands enabled and receives multiple paths to reach the same prefix, it performs load-balancing if the paths meet the BGP load-balancing conditions.
4. If all paths received carry the DMZ Link bandwidth extended community, the BGP process will perform unequal cost load-balancing proportional to the extended community attribute values.

In our scenario, R5 has two border routers in AS 100, R1, and R4 (after enabling the additional peering between R5 and R1). Our goal is to make R5 load balance for Loopback0 prfixes of AS 100 using both of its uplinks. We achieve this by configuring R5 with the dmzlink bandwidth feature on its uplinks to AS 100. At the same time, R5 is configured for eBGP multipathing and inserts both sets of paths into the local RIB.

```
R5:
router bgp 200
 maximum-path 4
 bgp dmzlink-bw
 neighbor 155.1.0.1 remote-as 100
 neighbor 155.1.0.1 dmzlink-bw
 neighbor 155.1.45.4 dmzlink-bw
!
interface Tunnel 0
```

https://t.me/learningnets

```
 bandwidth 50000

R4:
router bgp 100
 network 150.1.4.4 mask 255.255.255.255

R1:
router bgp 100
 neighbor 155.1.0.5 remote-as 200
 network 150.1.1.1 mask 255.255.255.255

R6:

router bgp 100
 network 150.1.6.6 mask 255.255.255.255
```

Verification

Take any Loopback0 prefix learned from AS 100 and look it up in the BGP table. Notice that there are two paths, both marked as “multipath.” That means that BGP is using them both, even though only the second path is elected as “best” by the BGP process. Notice the DMZ-Link Bw attribute values for both paths with their ratio being 125000/6250=20:

```
R5#show ip bgp 150.1.1.1
BGP routing table entry for 150.1.1.1/32, version 26
Paths: (3 available, best #1, table default)
Multipath: eBGP
  Advertised to update-groups:
     15         16         17
  Refresh Epoch 3
  100
    155.1.0.1 from 155.1.0.1 (150.1.1.1)      Origin IGP, metric 0, localpref 100, valid, external,
multipath, best
 DMZ-Link Bw 6250 kbytes
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 2
  100
    155.1.13.1 (metric 3328) from 155.1.0.3 (150.1.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 2
  100
    155.1.45.4 from 155.1.45.4 (150.1.4.4)
```

https://t.me/learningnets

```
      Origin IGP, localpref 100, valid, external, multipath(oldest)
 DMZ-Link 125000 kbytes

      rx pathid: 0, tx pathid: 0
```

Now look at the routing table entry for the same prefix. Notice that the share counters are 20:1, thus CEF hashing algorithm matched the exact ratio. That means that for approximately every 20 packets sent via R4, one packet is routed across R1 (although this proportion may be different with per-flow load balancing):

```
R5#show ip route 150.1.1.1
Routing entry for 150.1.1.1/32
  Known via "bgp 200", distance 20, metric 0
  Tag 100, type external
  Last update from 155.1.45.4 00:01:36 ago
  Routing Descriptor Blocks:
    155.1.45.4, from 155.1.45.4, 00:01:36 ago Route metric is 0, traffic share count is 20
      AS Hops 1
      Route tag 100
      MPLS label: none
  * 155.1.0.1, from 155.1.0.1, 00:01:36 ago Route metric is 0, traffic share count is 1

      AS Hops 1
      Route tag 100
      MPLS label: none
```

Now if we look at how CEF is programming this into the forwarding information base, more details will be uncovered:

```
R5#show ip cef 150.1.1.1 internal

 150.1.1.1/32, epoch 2, flags rib only nolabel, rib defined all labels, RIB[B], refcount 6, per-destination sharing
  sources: RIB
  feature space:
   IPRM: 0x00018000
   Broker: linked, distributed at 4th priority
  ifnums:
   Tunnel0(10): 155.1.0.1
   GigabitEthernet1.45(13): 155.1.45.4
  path 7FC45840ED60, path list 7FC464C0DC90, share 1/1, type recursive, for IPv4
  recursive via 155.1.0.1[IPv4:Default], fib 7FC466C7DE08, 1 terminal fib, v4:Default:155.1.0.1/32
    path 7FC45840F230, path list 7FC464C0D790, share 1/1, type adjacency prefix, for IPv4
    attached to Tunnel0, adjacency IP midchain out of Tunnel0, addr 155.1.0.1 7FC466986A80
  path 7FC45840F7B0, path list 7FC464C0DC90, share 20/20, type recursive, for IPv4
  recursive via 155.1.45.4[IPv4:Default], fib 7FC466C7EC08, 1 terminal fib, v4:Default:155.1.45.4/32
```

https://t.me/learningnets

```
    path 7FC45840F5A0, path list 7FC464C0DAB0, share 1/1, type adjacency prefix, for IPv4
    attached to GigabitEthernet1.45, adjacency IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
  output chain:
    loadinfo 7FC464124AB8, per-session, 2 choices, flags 0003, 7 locks
    flags: Per-session, for-rx-IPv4 16 hash buckets

< 0 > IP midchain out of Tunnel0, addr 155.1.0.1 7FC466986A80 IP adj out of GigabitEthernet1.100, addr 169.254.100.1
< 1 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 2 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 3 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 4 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 5 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 6 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 7 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 8 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
< 9 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
<10 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
<11 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
<12 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
<13 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
<14 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0
<15 > IP adj out of GigabitEthernet1.45, addr 155.1.45.4 7FC4669875C0

    Subblocks:
     None
```

Note that out of the 16 hash buckets available, the DMVPN Tunnel is only being used for one, and the GigabitEthernet1.45 interface is being used for the rest. As of IOS 12.2(4)T, load balancing can be achieved between an EBGP path and an IBGP path. As long as everything else is equal, BGP will load balance between the two paths. The **`maximum-paths eibgp`** command must be used. In our example, R5 has three paths total, two external paths that are being used for load balancing and an internal path from R3. If we enabled **`maximum-paths eibgp`** on R5, R5 would use all three paths for load balancing:

```
R5(config)#router bgp 200
R5(config-router)#no maximum-paths 4
R5(config-router)#maximum-paths eibgp 4
%BGP: This may cause traffic loop if not used properly (command accepted)
%BGP-4-MULTIPATH_LOOP: This may cause traffic loop if not used properly (command accepted)
!R5#show ip bgp 150.1.1.1
BGP routing table entry for 150.1.1.1/32, version 39
Paths: (3 available, best #1, table default) Multipath: eiBGP
  Advertised to update-groups:
     15          16          17
```

https://t.me/learningnets

```
  Refresh Epoch 6
  100
    155.1.0.1 from 155.1.0.1 (150.1.1.1)      Origin IGP, metric 0, localpref 100, valid, external,
multipath, best
      DMZ-Link Bw 6250 kbytes
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 5
  100
    155.1.13.1 (metric 3328) from 155.1.0.3 (150.1.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal, multipath
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 5
  100
    155.1.45.4 from 155.1.45.4 (150.1.4.4)      Origin IGP, localpref 100, valid, external,
multipath(oldest)
      DMZ-Link Bw 125000 kbytes
      rx pathid: 0, tx pathid: 0
!R5#show ip route 150.1.1.1
Routing entry for 150.1.1.1/32
  Known via "bgp 200", distance 20, metric 0
  Tag 100, type external
  Last update from 155.1.45.4 00:01:47 ago
  Routing Descriptor Blocks:
    155.1.45.4, from 155.1.45.4, 00:01:47 ago Route metric is 0, traffic share count is 20
      AS Hops 1
      Route tag 100
      MPLS label: none
    155.1.13.1, from 155.1.0.3, 00:01:47 ago Route metric is 0, traffic share count is 20
      AS Hops 1
      Route tag 100
      MPLS label: none
    * 155.1.0.1, from 155.1.0.1, 00:01:47 ago Route metric is 0, traffic share count is 1

    AS Hops 1
     Route tag 100
     MPLS label: none
```

Note that now all three routes are being used. As you noticed earlier from the log message, this command must be used carefully because it can introduce forwarding loops.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Bestpath Selection - Maximum AS Limit

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task is removed.
- Configure the routers in AS 200 to accept only the prefixes originated from directly connected AS's.
- Do not use filtering based on AS-PATH access-lists to accomplish this.

Configuration

BGP implementation in Cisco IOS has a special feature that looks similar to TTL scoping in IP networks. It is called BGP maximum AS limit and is enabled by using the BGP process command `bgp maxas-limit <N>` . This feature sets the maximum number of AS elements allowed in the AS_PATH attribute. The regular counting rules apply: AS_SET element counts as one, and AS_CONFED_* elements are ignored when counting. The default value for this limit is 75 AS elements, and this may be exceeded if AS_PATH prepending is used extensively.

The BGP process will generate log messages for prefixes that exceed the configured limit, similar to the message below:

```
%BGP-6-ASPATH: Long AS path 54 50 60 received from 155.1.23.3: BGP(0) Prefixes: 112.0.0.0/8 113.0.0.0/8
```

In our scenario, we configure R2, R3, R5, and R8 to accept only the prefixes with

https://t.me/learningnets

only one AS element in the AS_PATH attribute. This limits accepted prefixes to directly attached systems only – AS 54, AS 100, AS 254, and AS 200. For example, A 200 could not receive prefixes from AS 54 that pass through AS 300 or AS 100 before getting advertised to AS 200.

```
R2, R3, R5, R8:

router bgp 200
 bgp maxas-limit 1
```

Verification

Remember to clear the BGP sessions after you have applied this feature, because it applies only to the incoming prefixes. Then check the BGP tables of all routers in AS 200 and confirm that they only contain prefixes with an AS_PATH length of one or less:

```
R2#clear ip bgp * soft
R3#clear ip bgp * soft
R5#clear ip bgp * soft
R8#clear ip bgp * soft
!R2#show ip bgp

BGP table version is 160, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i -
              r RIB-failure, S Stale, m multipath, b backup-path, f R
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>  51.51.51.51/32   192.10.1.254             0             0 254 ?
 *>i 114.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 115.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 116.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 117.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
```

https://t.me/learningnets

```
 *>i 118.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 119.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 155.1.0.0        155.1.58.8               0    100      0 i
 * i                  155.1.58.8               0    100      0 i
 r>  192.10.1.0       192.10.1.254             0             0 254 ?
 *>  205.90.31.0      192.10.1.254             0             0 254 ?
 *>  220.20.3.0       192.10.1.254             0             0 254 ?
 *>  222.22.2.0       192.10.1.254             0             0 254 ?
!R5#show ip bgp

BGP table version is 56, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
 * i                  192.10.1.254             0    100      0 254 ?
 *>i 114.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 115.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 116.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 117.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 118.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 119.0.0.0        155.1.108.10             0    100      0 54 i
 * i                  155.1.108.10             0    100      0 54 i
 *>i 155.1.0.0        155.1.58.8               0    100      0 i
 * i                  155.1.58.8               0    100      0 i
 r>i 192.10.1.0       192.10.1.254             0    100      0 254 ?
 r i                  192.10.1.254             0    100      0 254 ?
 *>i 205.90.31.0      192.10.1.254             0    100      0 254 ?
 * i                  192.10.1.254             0    100      0 254 ?
 *>i 220.20.3.0       192.10.1.254             0    100      0 254 ?
 * i                  192.10.1.254             0    100      0 254 ?
```

https://t.me/learningnets

```
 *>i 222.22.2.0       192.10.1.254             0    100      0 254 ?
 * i                  192.10.1.254             0    100      0 254 ?
!R8#show ip bgp

BGP table version is 103, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  28.119.16.0/24   155.1.108.10             0             0 54 i
 *>  28.119.17.0/24   155.1.108.10             0             0 54 i
 * i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
 *>  114.0.0.0        155.1.108.10                           0 54 i
 *>  115.0.0.0        155.1.108.10                           0 54 i
 *>  116.0.0.0        155.1.108.10                           0 54 i
 *>  117.0.0.0        155.1.108.10                           0 54 i
 *>  118.0.0.0        155.1.108.10                           0 54 i
 *>  119.0.0.0        155.1.108.10                           0 54 i
 *>  155.1.0.0        0.0.0.0                           32768 i
 s>  155.1.58.0/24    0.0.0.0                  0        32768 i
 r i 192.10.1.0       192.10.1.254             0    100      0 254 ?
 r>i                  192.10.1.254             0    100      0 254 ?
 * i 205.90.31.0      192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
 * i 220.20.3.0       192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
 * i 222.22.2.0       192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Backdoor

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration to influence path selection from the previous task is removed (bgp maxas-limit).
- Shutdown the BGP peering between AS 100 and AS 300.
- Create a new Loopback1 interface in R7 with the IP address 150.1.77.77/24 and advertise it into BGP.
- Configure R1 and R4 so that they prefer to reach the new subnet via EIGRP, as opposed to eBGP.

Configuration

BGP prefixes learned from eBGP peers have the AD value of 20, the lowest among all dynamic routing protocols. This was done intentionally to prevent possible routing loops caused by redistribution of BGP routes into IGP. The local router always trusts the prefixes learned from external peers because they have passed the loop detection test.

In some cases, two autonomous systems may be connected by a “backdoor” link used to exchange routes between the ASs dynamically, by means of an IGP. This may be the result of a split or merger. In any case, the issue is that the backdoor link is usually preferred to exchange traffic between the two ASs. However, if both systems peer with another external AS, the border BGP routers will choose the prefix advertised via eBGP over the same prefix received across the backdoor link

https://t.me/learningnets

via IGP. For example, in our scenario, AS 300 and AS 100 run the EIGRP on the “backdoor” link connecting R6 and R7. When we shutdown the BGP peering session between the mentioned routers, the only way they can exchange routing information is by means of IGP. When the new prefix is advertised into BGP from R7, R1 and R4 will learn it via eBGP and EIGRP simultaneously. Based on the preferred AD for eBGP prefixes, R1 and R4 will choose suboptimal paths across AS 200, instead of the backdoor link.

To resolve this issue, you may change the AD of eBGP routes in R1 and R4, but this may increase the risk of routing loops. There is a special command in the BGP configuration mode used to explicitly change the distance of an eBGP prefix: **`network <subnet> mask <netmask> backdoor`** . Remember that the purpose of this command is to change the AD of a particular eBGP prefix from 20 to 200, not to advertise a new network. Thus, the command applies to non-local prefixes as well. When the command is entered, the eBGP speakers will prefer paths learned via IGP and utilize the backdoor link:

```
R7:
router bgp 300
 neighbor 155.1.67.6 shutdown
 network 150.1.77.0 mask 255.255.255.0
!
interface Loopback1
 ip address 150.1.77.77 255.255.255.0

R1, R4:

router bgp 100
 network 150.1.77.0 mask 255.255.255.0 backdoor
```

Verification

Check the route for the new prefix in R1’s RIB before you apply the backdoor configuration. Based on eBGP AD, the prefix learned via BGP is preferred:

```
R1#show ip route 150.1.77.0
Routing entry for 150.1.77.0/24 Known via "bgp 100", distance 20, metric 0

  Tag 200, type external
  Last update from 155.1.13.3 00:00:02 ago
  Routing Descriptor Blocks:
  * 155.1.13.3, from 155.1.13.3, 00:00:02 ago
      Route metric is 0, traffic share count is 1
      AS Hops 2
```

https://t.me/learningnets

```
      Route tag 200
      MPLS label: none
```

Apply the solution and see how the routing information has changed for the prefix. R1 now prefers the route learned via EIGRP over the eBGP path:

```
R1#sh ip route 150.1.77.0
Routing entry for 150.1.77.0/24 Known via "eigrp 100", distance 90, metric 131072, type internal

  Redistributing via eigrp 100
  Last update from 155.1.146.6 on GigabitEthernet1.146, 00:00:01 ago
  Routing Descriptor Blocks:
  * 155.1.146.6, from 155.1.146.6, 00:00:01 ago, via GigabitEthernet1.146
      Route metric is 131072, traffic share count is 1
      Total delay is 5020 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
```

Look at the BGP table entry for the same prefix. As you can see, the prefix is marked with “RIB Failure” state. This means that BGP was unable to insert the best path into RIB, because there is another prefix with a better AD:

```
R1#show ip bgp 150.1.77.0
BGP routing table entry for 150.1.77.0/24, version 131Paths: (2 available, best #2, table default,
RIB-failure(17) - next-hop mismatch)
  Advertised to update-groups:
     2
  Refresh Epoch 3
  200 300, (Received from a RR-client)
    155.1.45.5 (metric 3072) from 155.1.146.4 (150.1.4.4)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 3
  200 300
    155.1.13.3 from 155.1.13.3 (150.1.3.3)
      Origin IGP, localpref 100, valid, external, best
      rx pathid: 0, tx pathid: 0x0
!R1#show ip bgp rib-failure
  Network            Next Hop                      RIB-failure   RIB-NH Matches
 150.1.77.0/24      155.1.13.3Higher admin distance
              n/a
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Aggregation

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the BGP configuration from the previous task is removed.
- Configure R2 with four new Loopback interfaces with the IP addresses 10.0.0.1/24, 10.0.1.1/24, 10.0.2.1/24, and 10.0.3.1/24.
- Advertise an aggregate route for these networks that does not overlap any other IPv4 address space.

Configuration

Route aggregation is the key for information hiding. It is critical to BGP because of the tremendous amount of routing information passed on the Internet. There are three basic ways to do summarization in BGP:

- Create a summary prefix in IGP and advertise it into BGP using the `network` command. This is usually accomplished by creating a static route to Null0 in the routing table of the advertising router. This is a common way to advertise local prefixes into BGP. However, you cannot summarize external BGP prefixes using this method.
- Use auto-summarization. As discussed in another task, this method summarizes networks to their classful boundaries and only applies to redistributed prefixes or when using the classful `network` command. It is not used in modern networks.
- Summarize prefixes found in BGP tables by using the `aggregate-address` command.

https://t.me/learningnets

This is the most flexible way to do summarization, because it may be applied to any paths learned by the BGP speaker.

In the next few tasks, we will work with the last command. It has many options, and it allows manipulating BGP attributes when aggregating prefixes. In the simplest form, the syntax for the command is `aggregate-address <prefix> <mask>` . For the command to work, there must be a subnet in the BGP table that is encompassed by the summarized prefix. For example, if you issue the command `aggregate-address 192.168.0.0 255.255.0.0` , at least one subnet, such as 192.168.1.0/24, must be in the BGP table (Loc-RIB), but it does not necessarily need to be in the router's routing table (RIB).

If you do not specify any additional options to the command, it will create a new prefix in the BGP table with an empty AS_PATH. It will look like the new prefix was originated in the local AS. The new prefix will automatically have the weight value of 32768 and have a special attribute called ATOMIC_AGGREGATE assigned. The new attribute is informational and tells the other BGP speakers that this prefix is a result of route aggregation, and some information (like AS_PATH or other attributes) from the original prefixes may be missing. In addition to the ATOMIC_AGGREGATE attribute, BGP attaches another attribute called AGGREGATOR to the summarized prefix. This attribute specifies the AS number and the BGP router-ID of the aggregating router. Just like the ATOMIC_AGGREGATE, the new attribute is also informational.

For every aggregate, the BGP process will install an automatic static route to Null0 for the new prefix, to prevent routing loops. Remember that the original (specific) prefixes are *still advertised*, unlike in IGP, where summarization automatically suppresses more specific prefixes.

In our scenario, we must come up with the most effective summary prefix for the subnets. Using the classic summarization rules, we write all prefixes in binary format:

```
10.0.0.1=00000110.00000000.000000|00.00000001
10.0.1.1=00000110.00000000.000000|01.00000001
10.0.2.1=00000110.00000000.000000|10.00000001
10.0.3.1=00000110.00000000.000000|11.00000001
```

Based on that output, we select the maximum common part for all four prefixes, and shift the subnet prefix length by the number of bits stripped. The result is 10.0.0.0/22, or 10.0.0.0 255.255.252.0.

```
R2:
```

https://t.me/learningnets

```
interface Loopback100
 ip address 10.0.0.1 255.255.255.0
!
interface Loopback101
 ip address 10.0.1.1 255.255.255.0
!
interface Loopback102
 ip address 10.0.2.1 255.255.255.0
!
interface Loopback103
 ip address 10.0.3.1 255.255.255.0
!
router bgp 200
 network 10.0.0.0 mask 255.255.255.0
 network 10.0.1.0 mask 255.255.255.0
 network 10.0.2.0 mask 255.255.255.0
 network 10.0.3.0 mask 255.255.255.0
 aggregate-address 10.0.0.0 255.255.252.0
```

Verification

Check the summary prefix in the BGP table of R2. Notice the atomic-aggregate attribute on this prefix and the aggregator attribute value (aggregated by 200 150.1.2.2):

```
R2#show ip bgp 10.0.0.0/22
BGP routing table entry for 10.0.0.0/22, version 194
Paths: (1 available, best #1, table default)
Flag: 0x820
  Advertised to update-groups: (Pending Update Generation)
     1
  Refresh Epoch 1  Local, (aggregated by 200 150.1.2.2
)
    0.0.0.0 from 0.0.0.0 (150.1.2.2)      Origin IGP, localpref 100, weight 32768, valid, aggregated
, local, atomic-aggregate
, best
      rx pathid: 0, tx pathid: 0x0
```

Now check the summary route to Null0 installed in the routing table of R2:

```
R2#show ip route 10.0.0.0 255.255.252.0
Routing entry for 10.0.0.0/22  Known via "bgp 200", distance 200, metric 0, type locally generated
  Routing Descriptor Blocks: * directly connected, via Null0
```

https://t.me/learningnets

```
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: none
```

Confirm that the summary prefix did not suppress the more specific routes. Take R4, for instance:

```
R4#show ip bgp regexp _200$
BGP table version is 230, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
* i 10.0.0.0/24      155.1.13.3               0    100      0 200 i
 *>                  155.1.45.5                             0 200 i
* i 10.0.0.0/22      155.1.13.3               0    100      0 200 i

 *>                  155.1.45.5                             0 200 i
 * i 10.0.1.0/24     155.1.13.3               0    100      0 200 i
 *>                  155.1.45.5                             0 200 i
 * i 10.0.2.0/24     155.1.13.3               0    100      0 200 i
 *>                  155.1.45.5                             0 200 i
 * i 10.0.3.0/24     155.1.13.3               0    100      0 200 i
 *>                  155.1.45.5                             0 200 i
 *>  155.1.0.0       155.1.45.5                             0 200 i
 * i                 155.1.13.3               0    100      0 200 i
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Aggregation - Summary Only

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing With Aggregation**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- R2 has been preconfigured to inject a summary into BGP for its connected Loopback prefixes.
- Modify the aggregation configuration on R2 so that no other devices besides R2 can see the specific prefixes that make up the summary.

Configuration

As you learned in the previous scenario, BGP summarization using the `aggregate-address` command creates new prefixes in the BGP table but does not suppress the advertisement of the specific prefixes that make up the summary. To generate just the summary prefix, use the option `summary-only` after the `aggregate-address` command. The BGP process will automatically suppress advertisement of the prefixes in the BGP table encompassed by the new summary address. This is probably the most common use of the aggregation command, because usually only the summarized prefix should be advertised.

```
R2:

router bgp 200
 aggregate-address 10.0.0.0 255.255.252.0 summary-only
```

https://t.me/learningnets

Verification

Look at R2's BGP table. Notice that all specific prefixes are marked with "s," meaning "suppressed." They are not advertised to any peers, only the summary prefix is advertised:

```
R2#show ip bgp | include 10.0.

 s>  10.0.0.0/24      0.0.0.0                  0         32768 i
 *>  10.0.0.0/22      0.0.0.0                            32768 i
 s>  10.0.1.0/24      0.0.0.0                  0         32768 i
 s>  10.0.2.0/24      0.0.0.0                  0         32768 i
 s>  10.0.3.0/24      0.0.0.0                  0         32768 i
!R2#show ip bgp neighbors 155.1.0.5 advertised-routes

BGP table version is 198, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>   10.0.0.0/22      0.0.0.0                             32768 i

 *>  51.51.51.51/32   192.10.1.254             0             0 254 ?
 r>  192.10.1.0       192.10.1.254             0             0 254 ?
 *>  205.90.31.0      192.10.1.254             0             0 254 ?
 *>  220.20.3.0       192.10.1.254             0             0 254 ?
 *>  222.22.2.0       192.10.1.254             0             0 254 ?

Total number of prefixes 6
```

Connectivity is preserved nonetheless, because the summary prefix is enough to reach back to the new interfaces of R2:

https://t.me/learningnets

```
R2#ping 220.20.3.1 source Loopback1101


Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 220.20.3.1, timeout is 2 seconds:
Packet sent with a source address of 10.0.1.1 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/7/18 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Aggregation - Suppress Map

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing With Aggregation**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- R2 has been preconfigured to inject a summary into BGP for its connected Loopback prefixes.
- Modify the aggregation configuration on R2 so that R2 advertises 10.0.2.0/24 prefix in addition to the summary route, but everything else is suppressed.

Configuration

When you specify the `summary-only` keyword, all specific prefixes are suppressed. It is possible to suppress prefixes selectively, using a route-map associated via the parameter `suppress-map` . The prefixes permitted by this route-map are suppressed; prefixes denied by this route-map are NOT suppressed when performing summarization. For example:

```
ip prefix-list SUPPRESS_PREFIX 150.1.1.0/24
!
route-map SUPPRESS_MAP permit 10
 match ip address prefix-list SUPPRESS_PREFIX
!
router bgp 200
 aggregate-address 150.1.0.0 mask 255.255.0.0 suppress-map SUPPRESS_MAP
```

Imagine that prefixes 150.1.1.0/24, 150.1.2.0/24, and 150.1.3.0/24 are in the BGP

https://t.me/learningnets

table. The above command will produce the new summary prefix 150.1.0.0/16 and suppress only one prefix: 150.1.1.0/24. In our scenario, all prefixes are suppressed with the exception of 10.0.2.0/24. We use a prefix list to match the subnet and a special “deny” statement in the route map to exclude this prefix from suppression. Other prefixes match the “permit” entry in the end of the route-map and are suppressed.

```
R2:

ip prefix-list NET_2 permit 10.0.2.0/24
!
route-map SUPPRESS_MAP deny 10
 match ip address prefix-list NET_2
!
route-map SUPPRESS_MAP permit 100
!
router bgp 200
 aggregate-address 10.0.0.0 255.255.252.0 suppress-map SUPPRESS_MAP
```

Verification

Look at R2’s BGP table again. Now you can see that the prefix 10.0.2.0/24 is not suppressed. Furthermore, you can confirm that it is being advertised to R2’s peers:

```
R2#show ip bgp | include 10.0
 s>  10.0.0.0/24      0.0.0.0                  0         32768 i
*>  10.0.0.0/22      0.0.0.0                            32768 i
 s>  10.0.1.0/24      0.0.0.0                  0         32768 i
*>  10.0.2.0/24      0.0.0.0                  0         32768 i
 s>  10.0.3.0/24      0.0.0.0                  0         32768 i
!R2#show ip bgp neighbors 155.1.0.5 advertised-routes

BGP table version is 221, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.0.0.0/22      0.0.0.0                            32768 i
*>  10.0.2.0/24      0.0.0.0                  0         32768 i

 *>  51.51.51.51/32   192.10.1.254             0             0 254 ?
```

https://t.me/learningnets

```
 r>  192.10.1.0       192.10.1.254             0             0 254 ?
 *>  205.90.31.0      192.10.1.254             0             0 254 ?
 *>  220.20.3.0       192.10.1.254             0             0 254 ?
 *>  222.22.2.0       192.10.1.254             0             0 254 ?

Total number of prefixes 7
```

You can observe the new aggregate prefix generation by using the **`debug ip bgp`** command. Enable debugging and enter the aggregation command under the BGP process. Notice that the debug output shows the suppressed prefixes and explicitly states that 10.0.2.0/24 is not suppressed:

```
R2(config)#router bgp 200
R2(config-router)#do debug ip bgp
BGP debugging is on for address family: IPv4 Unicast
R2(config-router)#aggregate-address 10.0.0.0 255.255.252.0 suppress-map SUPPRESS_MAP
!
BGP: Sched timer-wheel running slow by 1 ticks
BGP(0): Aggregate processing for IPv4 UnicastBGP(0): For aggregate 10.0.0.0/22
BGP(0): 10.0.0.0/22 subtree has an entry 10.0.0.0/24
BGP(0): sub-prefix : 10.0.0.0/24
BGP(0): 10.0.0.0/22 subtree has an entry 10.0.0.0/24
BGP(0): 10.0.0.0/22 aggregate has 10.0.0.0/24 more-specific
BGP(0): 10.0.0.0/22 aggregate updated
BGP(0): 10.0.0.0/22 subtree has an entry 10.0.0.0/24
BGP(0): Found sub-prefix 10.0.0.0/24: suppressed
BGP(0): Found sub-prefix 10.0.0.0/22:
BGP(0): Found sub-prefix 10.0.1.0/24: suppressedBGP(0): Found sub-prefix 10.0.2.0/24: Not matched

BGP(0): Found sub-prefix 10.0.3.0/24: suppressed
BGP(0): Suppress sub-prefix 28.119.16.0/24 : out of range
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Aggregation - Unsuppress Map

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing With Aggregation**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- R2 has been preconfigured to inject a summary into BGP for its connected Loopback prefixes.
  - Remove the summarization configured on R2, but leave the network statements for all of the loopbacks.
- Using the summary-only feature, configure R3 and R5 to originate an aggregate route for these networks that does not overlap any other IPv4 address space.
- Using the unsuppress-map feature configure the network as follows:
  - Traffic from AS 100 going to the prefix 10.0.1.0/24 always transits AS 300 unless the link between R3 and R7 is down.
  - Traffic from AS 100 going to other subnets of the aggregate should use best path as selected by BGP.

Configuration

This scenario demonstrates one of the common uses for the `aggregate-address` command. Local networks are advertised into BGP and aggregated by the border BGP speakers.

It is often desirable to load-balance traffic ingress to the local AS, so that traffic to some subnets enters via one BGP peer and the other peer is used as the entry point for other subnets. Generally, to accomplish this, you need to advertise all specific prefixes on both uplinks and use AS_PATH prepending to modify prefixes'

https://t.me/learningnets

preference. This scheme implements load balancing and provides backup in case of any uplink failures.

However, it is possible to achieve the same goal using a different technique. It is based on the fact that classless routing always prefers the most specific prefix to reach the destinations. If there is a specific prefix in the routing table (for example, 10.0.1.0/24) and the summary one (for example, 10.0.0.0/22), the router will prefer /24 and use /22 only if the most specific prefix vanishes. Thus, by configuring the border BGP peers for advertising both the summary and selected specific prefixes, you may achieve the same load-balancing with the necessary level of redundancy.

To implement this technique, you may use the **`unsuppress-map`** BGP feature. This feature can only be configured on the router that performs prefix aggregation using the command **`aggregate-address`** with **`summary-only`** . The feature uses a special route-map that matches and permits the prefixes required to be unsuppressed and is applied only on a per-neighbor basis as follows:

```
router bgp 100
 neighbor 150.1.37.7 unsuppress-map UNSUPPRESS
```

When the aggregate route is advertised to the selected peer, all the suppressed prefixes found in the local BGP table are matched against the configured **`unsuppress-map`** . The matching prefixes are advertised in addition to the summary prefix. Other peers or the local BGP table are not affected by this configuration.

In this scenario, R3 and R5 perform prefix aggregation. R3 is configured to unsuppress and advertise one specific prefix: 10.0.1.0/24 to R7. When AS 100 receives these prefixes, it prefers to reach 10.0.1.0/24 via AS 300, because this is the way that explicit prefix (most specific) has traveled to reach AS 100.

```
R2:
router bgp 200
 no aggregate-address 10.0.0.0 255.255.252.0

R3:
ip prefix-list NET_1 permit 10.0.1.0/24
!
route-map UNSUPPRESS_MAP permit 10
 match ip address prefix-list NET_1
!
router bgp 200
 aggregate-address 10.0.0.0 255.255.252.0 summary-only
 neighbor 155.1.37.7 unsuppress-map UNSUPPRESS_MAP
```

https://t.me/learningnets

```
R5:

router bgp 200
 aggregate-address 10.0.0.0 255.255.252.0 summary-only
```

Verification

Start your verification by looking at the BGP tables of R3 and R5. Notice that all specific prefixes are suppressed:

```
R3#show ip bgp | include 10.0
 s>i 10.0.0.0/24      155.1.23.2               0    100      0 i
* i 10.0.0.0/22      155.1.0.5                0    100      0 i
 s>i 10.0.1.0/24      155.1.23.2               0    100      0 i
 s>i 10.0.2.0/24      155.1.23.2               0    100      0 i
 s>i 10.0.3.0/24      155.1.23.2               0    100      0 i
R5#show ip bgp | include 10.0
 s>i 10.0.0.0/24      155.1.0.2                0    100      0 i
*>  10.0.0.0/22      0.0.0.0                             32768 i

 s>i 10.0.1.0/24      155.1.0.2                0    100      0 i
 s>i 10.0.2.0/24      155.1.0.2                0    100      0 i
 s>i 10.0.3.0/24      155.1.0.2                0    100      0 i
```

Look at the routes that R3 advertises to R7. Notice that the prefix 10.0.1.0/24 is included, even though it is marked as “suppressed.” This means that the prefix was unsuppressed by the configured feature:

```
R3#show ip bgp neighbors 155.1.37.7 advertised-routes | include 10.0
 *>  10.0.0.0/22      0.0.0.0                             32768 i s>i 10.0.1.0/24
     155.1.23.2               0    100      0 i
```

Now take any router in AS 100. Look up the route 10.0.1.0/24 in the local routing table. Notice that it points to R7 and the number of AS hops is 2:

```
R6#show ip route 10.0.1.0
Routing entry for 10.0.1.0/24
  Known via "bgp 100", distance 20, metric 0
  Tag 300, type external
  Last update from 155.1.67.7 00:03:56 ago
  Routing Descriptor Blocks: * 155.1.67.7, from 155.1.67.7, 00:03:56 ago
      Route metric is 0, traffic share count is 1 AS Hops 2
```

https://t.me/learningnets

```
      Route tag 300
      MPLS label: none
```

At the same time, all unsuppressed prefixes are preferred via R1, because they are covered by the summary route and R1 is a more direct path than through R7:

```
R6#show ip route 10.0.2.0
Routing entry for 10.0.0.0/22
  Known via "bgp 100", distance 200, metric 0
  Tag 200, type internal
  Last update from 155.1.13.3 00:05:57 ago
  Routing Descriptor Blocks:
  * 155.1.13.3, from 155.1.146.1, 00:05:57 ago
      Route metric is 0, traffic share count is 1 AS Hops 1

      Route tag 200
      MPLS label: none
```

If you look at R6's BGP table, you will notice that the path to 10.0.0.0/22 is via R3, whereas the path to 10.0.1.0/24 is via R7:

```
R6#show ip bgp

BGP table version is 96, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>i 10.0.0.0/22      155.1.13.3
               0    100      0 200 i
 *                   155.1.67.7                             0 300 200 i *>  10.0.1.0/24      155.1.67.7
                            0 300 200 i
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Aggregation - AS-Set

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing With Aggregation**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that all summarization configuration on R3 and R5 from previous task is removed.
- Configure R5 to aggregate the subnets 112.0.0.0/24–119.0.0.0/24 into one prefix using the optimal prefix length.
- Ensure that the new summary prefix is not accepted by AS 54 peers.
- Do not use any filtering techniques to accomplish this.

Configuration

It is important to remember that aggregation hides information previously found in the specific prefixes. This includes all attributes, such as NEXT_HOP, AS_PATH, and so on. The new prefix appears to be originated from within the local AS where aggregation is perfpormed. This causes no problems if all specific prefixes belong to the local AS. However, when you summarize prefixes learned from other ASs, information hiding may result in the following dangerous consequences:

- Suboptimal routing, caused by loss of path information, such as AS_PATH, MED and so on.
- Routing loops, because removing the AS_PATH attribute and replacing it with an empty list prevents the BGP loop-detection mechanism from working properly.

The second issue is more dangerous. To prevent it, it is possible to insert a special

https://t.me/learningnets

new member into the AS_PATH of the newly created summary prefix. This element is called AS_SET and contains the AS numbers found in all AS_PATHs of the specific prefixes. This list of AS numbers is unordered, unlike the regular AS_SEQUENCE element. Its only use is for routing loop prevention; when BGP receives a prefix, it scans the AS_PATH attribute. If the local AS number is found in any of the AS_SET or AS_SEQUENCE elements, the prefix is dropped.

By default, the aggregated address in BGP will not include the AS-Set information. To force the use of this information, specify the as-set option as follows: **`aggregate-address <subnet> <mask> as-set`** . In our scenario, BGP router in AS 200 aggregates the prefixes learned from another ASs. If the as-set feature is not used, both R9 and R10 would have accepted the new summary prefix; when R9 or R10 loose the specifc route to prefix 113.0.0.0, for example, they would route to AS 200 for this prefix, while AS 200 never had it to begin with. The solution aggregates the prefixes using the following bitmap breakdown:

```
01110|000.00000000.00000000.00000000
01110|001.00000000.00000000.00000000
...
01110|111.00000000.00000000.00000000
```

This results in the summary prefix of 112.0.0.0/5, or the same as 112.0.0.0 248.0.0.0

```
R5:

router bgp 200
 aggregate-address 112.0.0.0 248.0.0.0 summary-only as-set
```

Verification

If you look at R5's BGP table, you see that prefix 112.0.0.0/5 has the AS_PATH attribute in the format {54,50,60}, listing all AS numbers found in prefixes received from R9 and R10:

```
R5#show ip bgp

BGP table version is 46, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

https://t.me/learningnets

```
     Network          Next Hop            Metric LocPrf Weight Path
 * i 10.0.0.0/24      155.1.23.2               0    100      0 i
 *>i                  155.1.0.2                0    100      0 i
 * i 10.0.1.0/24      155.1.23.2               0    100      0 i
 *>i                  155.1.0.2                0    100      0 i
 * i 10.0.2.0/24      155.1.23.2               0    100      0 i
 *>i                  155.1.0.2                0    100      0 i
 * i 10.0.3.0/24      155.1.23.2               0    100      0 i
 *>i                  155.1.0.2                0    100      0 i
 * i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 *>i                  155.1.108.10             0    100      0 54 i
 * i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 *>i                  155.1.108.10             0    100      0 54 i
 * i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
 s>i 112.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 s i                  155.1.108.10             0    100      0 54 50 60 i
 *>  112.0.0.0/5      0.0.0.0                       100  32768 {54,50,60} i

 <output omitted>
```

Look at the detailed information for the new BGP prefix and notice the aggregator and the AS_SET attributes:

```
R5#show ip bgp 112.0.0.0 248.0.0.0
BGP routing table entry for 112.0.0.0/5, version 34
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     18         19         20
  Refresh Epoch 1 {54,50,60}, (aggregated by 200 150.1.5.5)

    0.0.0.0 from 0.0.0.0 (150.1.5.5)
      Origin IGP, localpref 100, weight 32768, valid, aggregated, local, best
      rx pathid: 0, tx pathid: 0x0
```

When R9 and R10 receive this prefix, they will detect the loop and drop the update. Look at what R7 is sending to R9 (in AS 54):

https://t.me/learningnets

```
R7#show ip bgp neighbors 155.1.79.9 advertised-routes | include 112.0.0.0

 *>  112.0.0.0        155.1.79.9               0             0 54 50 60 i
 *>  112.0.0.0/5      155.1.37.3                             0 200 {54,50,60} i
```

When R9 receives these prefixes and sees its own AS, it automatically drops them because of BGP's loop prevention mechanics. If as-set had NOT been used, the 112.0.0.0/5 prefix would have been accepted by R9 (and R10):

```
R9(config)#access-list 100 permit ip 112.0.0.0 7.255.255.255 any

 !R9#debug ip bgp updates 100 in
 R9#clear ip bgp * soft

BGP(0): 155.1.79.7 rcv UPDATE about 112.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE about 113.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE about 114.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE about 115.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE about 116.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE about 117.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE about 118.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE about 119.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
BGP(0): 155.1.79.7 rcv UPDATE w/ attr: nexthop 155.1.79.7, origin i, aggregated by 200 150.1.5.5
, originator 0.0.0.0, merged path 300 200 {54,50,60}, AS_PATH
 , community , extended community , SSA attribute BGP(0): 155.1.79.7 rcv UPDATE about
112.0.0.0/5 -- DENIED due to: AS-PATH contains our own AS;
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Aggregation - Attribute-Map

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing With Aggregation**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that all summarization configuration on R5 is removed before starting this task.
- Configure R5 to aggregate the subnets 112.0.0.0/8–119.0.0.0/8 into one prefix using the optimal prefix length. Use the `as-set` command on this summary.
- Configure R8 to mark the prefix 112.0.0.0/8 received from R10 with the community value of “no-export”.
- Ensure that this community value propagates across AS 200.
- Configure R5 so that the summary prefix 112.0.0.0/5 is still advertised to AS 300 and AS 100.

Configuration

When you use the `as-set` parameter to the `aggregate-address` command, the resulting prefix will inherit “additive” attributes of the specific prefixes. This includes the AS_PATH attributes, condensed into AS_SET and the community attributes, which are grouped together from all prefixes. We will explore community signaling in further detail, but for now remember that any prefix bearing the community attribute value of “no-export” is not advertised to the adjacent ASs.

In our scenario, we have R8 tagging just one prefix—112.0.0.0/8 with the community value of “no-export”. However, when R5 aggregates all prefixes into one, the summary prefix inherits the “no-export” community from one of the specific

https://t.me/learningnets

routes. In effect, AS 200 speakers will not be able to advertise the summary prefix to the neighbors.

The solution to this problem is the use of the `attribute-map` parameter to the `aggregate-address` command. This parameter specifies the route-map that sets BGP attributes for the newly generated prefix. You may set any configuration BGP value, such as metric, origin, local-preference, and so on. However, in our case we are interested in setting the community attribute value for the summary. The route-map applies the `set community none` command and erases all communities for the new prefix. Naturally, all routers are configured to propagate communities across AS 200.

```
R2:
router bgp 200
 neighbor 155.1.0.5 send-community
 neighbor 155.1.23.3 send-community

R3:
router bgp 200
 neighbor 155.1.0.5 send-community
 neighbor 155.1.23.2 send-community
 neighbor 155.1.58.8 send-community

R8:
no ip prefix-list NET_112
ip prefix-list NET_112 permit 112.0.0.0/8
!
no route-map SET_COMMUNITY
route-map SET_COMMUNITY permit 10
 match ip address prefix-list NET_112
 set community no-export
!
route-map SET_COMMUNITY permit 100
!
router bgp 200
 neighbor 155.1.108.10 route-map SET_COMMUNITY in
 neighbor 155.1.58.5 send-community
 neighbor 155.1.23.3 send-community

R5:

route-map ATTR_MAP
 set community none
!
router bgp 200
 aggregate-address 112.0.0.0 248.0.0.0 summary-only as-set attribute-map ATTR_MAP
```

https://t.me/learningnets

```
 neighbor 155.1.58.8 send-community
 neighbor 155.1.0.2 send-community
 neighbor 155.1.0.3 send-community
```

Verification

Check the BGP table of R8 and confirm that the prefix 112.0.0.0/8 is marked with the community attribute “no-export”:

```
R8#show ip bgp 112.0.0.0 255.0.0.0

BGP routing table entry for 112.0.0.0/8, version 49 Paths: (1 available, best #1, table default,
not advertised to EBGP peer
)
  Advertised to update-groups:
     5
  Refresh Epoch 3
  54 50 60
    155.1.108.10 from 155.1.108.10 (31.3.0.1)
      Origin IGP, localpref 100, valid, external, best Community: no-export

      rx pathid: 0, tx pathid: 0x0
```

Check the BGP tables of R5 for the prefix 112.0.0.0/5 before you apply the solution for this task to R5; only use `as-set` on the aggregate-address command so that you can observe the community value on the aggregate. Notice that the summary prefix has the “no-export” community attached as well. This prevents the summary prefix from being advertised to AS 100 and AS 300. Confirm that other BGP speakers, such as R2, also have the summary prefix tagged with “no-export” community:

```
R5#show ip bgp 112.0.0.0 248.0.0.0
BGP routing table entry for 112.0.0.0/5, version 75 Paths: (1 available, best #1, table default,
not advertised to EBGP peer
)
  Advertised to update-groups:
     21         22
  Refresh Epoch 1
  {54,50,60}, (aggregated by 200 150.1.5.5)
    0.0.0.0 from 0.0.0.0 (150.1.5.5)
      Origin IGP, localpref 100, weight 32768, valid, aggregated, local, best Community: no-export
      rx pathid: 0, tx pathid: 0x0
!R2#show ip bgp 112.0.0.0 248.0.0.0
```

https://t.me/learningnets

```
BGP routing table entry for 112.0.0.0/5, version 80 Paths: (2 available, best #2, table default,
not advertised to EBGP peer
)
  Not advertised to any peer
  Refresh Epoch 2  {54,50,60}, (aggregated by 200 150.1.5.5
)
    155.1.0.5 from 155.1.23.3 (150.1.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal Community: no-export
      Originator: 150.1.5.5, Cluster list: 150.1.3.3
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 4  {54,50,60}, (aggregated by 200 150.1.5.5
)
    155.1.0.5 from 155.1.0.5 (150.1.5.5)
      Origin IGP, metric 0, localpref 100, valid, internal, best Community: no-export

      rx pathid: 0, tx pathid: 0x0
```

Apply the solution to R5 and check the BGP table entry for 112.0.0.0/5 again. Confirm that the summary prefix does not have any community values now, and R5 advertises it to any eBGP peer:

```
R5#show ip bgp 112.0.0.0 248.0.0.0
BGP routing table entry for 112.0.0.0/5, version 76
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     20         21         22
  Refresh Epoch 1
  {54,50,60}, (aggregated by 200 150.1.5.5)
    0.0.0.0 from 0.0.0.0 (150.1.5.5)
      Origin IGP, localpref 100, weight 32768, valid, aggregated, local, best
      rx pathid: 0, tx pathid: 0x0
!R5#show ip bgp neighbors 155.1.45.4 advertised-routes

BGP table version is 84, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 10.0.0.0/24      155.1.0.2                0    100      0 i
 *>i 10.0.1.0/24      155.1.0.2                0    100      0 i
 *>i 10.0.2.0/24      155.1.0.2                0    100      0 i
 *>i 10.0.3.0/24      155.1.0.2                0    100      0 i
```

https://t.me/learningnets

```
 *>i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 *>i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
*>  112.0.0.0/5      0.0.0.0                       100  32768 {54,50,60} i

 *>i 155.1.0.0        155.1.58.8               0    100      0 i
 r>i 192.10.1.0       192.10.1.254             0    100      0 254 ?
 *>i 205.90.31.0      192.10.1.254             0    100      0 254 ?
 *>i 220.20.3.0       192.10.1.254             0    100      0 254 ?
 *>i 222.22.2.0       192.10.1.254             0    100      0 254 ?

Total number of prefixes 13
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Aggregation - Advertise Map

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing With Aggregation**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Configure R2 with two new Loopback interfaces with the IPv4 addresses 222.22.0.1/24 and 222.22.1.1/24 and advertise them into BGP.
- Configure R7 with a new Loopback interface with the IPv4 address 222.22.3.1/24 and advertise it into BGP.
- Configure R4 and R6 to advertise the aggregate 222.22.0.0/22 into BGP:
  - Include as much of the original AS-Path information as possible while still allowing devices in AS 300 to install the aggregate in the BGP table.

Configuration

When using the as-set keyword with BGP aggregation, some of the specific prefix attributes got mixed together in the new prefix. Specifically, you should watch out for the resulting AS_SET and list of community attributes. In the previous task, you learned how to modify some of the aggregated prefix attributes. However, you cannot manipulate an important attribute such as AS_SET directly. Instead, you may specify the specific prefixes that will be used to make up the attribute list for the aggregate prefix. This is accomplished by using the `advertise-map` parameter to the `aggregate-address` command. The route-map used as advertise-map should permit specific prefixes to be used to compose the aggregate attributes, such as AS_SET. You can use only access-list, prefix-list, or as-path access-lists to match the specific prefixes. Information from the prefixes denied by the route-map is not used when constructing the resulting summary-prefix. You may use this method to remove the

https://t.me/learningnets

prefixes with unwanted BGP community attributes as well.

In our scenario, the AS_SET attribute for the summary route should be composed of AS numbers 200, 254, and 300. However, by using the advertise-map parameter, we filter out prefix originated in AS 300 and thus end up with AS_SET of {200,254}. This allows AS 300 accepting back the summary prefix.

```
R2:
interface Loopback220
 ip address 222.22.0.1 255.255.255.0
!
interface Loopback221
 ip address 222.22.1.1 255.255.255.0
!
router bgp 200
 network 222.22.0.0 mask 255.255.255.0
 network 222.22.1.0 mask 255.255.255.0

R7:
interface Loopback 223
 ip address 222.22.3.1 255.255.255.0
!
router bgp 300
 network 222.22.3.0 mask 255.255.255.0

R4, R6:

ip prefix-list AS300_PREFIX permit 222.22.3.0/24
!
route-map ADVERTISE_MAP deny 10
 match ip address prefix-list AS300_PREFIX
!
route-map ADVERTISE_MAP permit 100
!
router bgp 100
 aggregate-address 222.22.0.0 255.255.252.0 summary-only as-set advertise-map ADVERTISE_MAP
```

Verification

https://t.me/learningnets

Check the specific and the summary prefix at any BGP speaker that performs summarization. Notice the AS_PATH attributes of the specific prefixes, especially for the prefix 222.22.3.0/24. The AS_SET for the summary /22 does not include AS 300:

```
R6#show ip bgp

BGP table version is 52, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
<snip>
 *   220.20.3.0       155.1.67.7                             0 300 200 254 ?
 *>i                  155.1.13.3               0    100      0 200 254 ?
 s   222.22.0.0       155.1.67.7                             0 300 200 i
 s>i                  155.1.13.3               0    100      0 200 i
 *>  222.22.0.0/22    0.0.0.0                       100  32768 {200,254} ?
 * i                  155.1.146.4              0    100      0 {200,254} ?
 s   222.22.1.0       155.1.67.7                             0 300 200 i
 s>i                  155.1.13.3               0    100      0 200 i
 s   222.22.2.0       155.1.67.7                             0 300 200 254 ?
 s>i                  155.1.13.3               0    100      0 200 254 ?
s i 222.22.3.0       155.1.13.3               0    100      0 200 300 i
 s>                   155.1.67.7               0             0 300 i
```

Now check R7's BGP table and notice that the summary prefix is there:

```
R7#show ip bgp

BGP table version is 34, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 <snip>
 *   220.20.3.0       155.1.67.6                             0 100 200 254 ?
 *>                   155.1.37.3                             0 200 254 ?
 *>  222.22.0.0       155.1.37.3                             0 200 i
 *>  222.22.0.0/22    155.1.67.6               0             0 100 {200,254} ?
```

https://t.me/learningnets

```
*>  222.22.1.0       155.1.37.3                             0 200 i
*>  222.22.2.0       155.1.37.3                             0 200 254 ?
*>  222.22.3.0       0.0.0.0                  0         32768 i
```

If the summarization performed on R4 and R6 did not make use of the advertise-map filtering out AS 300, the as-set by itself would have included all ASs in the AS_SET, including AS 300, and R7 would have dropped this update.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Communities

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Configure AS 100 to set local-preference attribute to 200 for eBGP prefixes tagged with community value 100:200.
- Configure R5 in AS 200 to signal AS 100 to prefer path to the prefixes originated in AS 60 via R4.

Configuration

BGP Communities are optional transitive attributes used mainly to associate an administrative tag to a route. All prefixes with the same community essentially belong to the same group and share some property. Using communities allows for manipulation of BGP prefixes based on their “meaning” (customer prefixes, peer prefixes, upstream networks, etc.), not the network values. Even though the BGP community attribute is transitive, Cisco IOS routers do not pass it across BGP sessions by default. To start sending the community values to a particular peer, you must activate this feature by using the command `neighbor <IP> send-community` .

There could be many communities associated with a BGP prefix, and every community attribute has a 32-bit value. There are two formats to read a community value: raw, as a 32-bit number, and structured, as a pair `AS Number:16-bit value` . The second format makes it easier to interpret communities, because they naturally point to the AS that originated the tagging. Note that by default community values are displayed in 32-bit format, and to use the structured notation you must enter the

https://t.me/learningnets

global configuration command `ip bgp-community new-format`. Because there are no ASs numbered 0 or 65535, the BGP community ranges 0x0000:0x0000-0x0000:0xFFFF and 0xFFFF:0x0000-0xFFFF:0xFFFF are reserved and not available for public use. There are three well-known BGP community values from reserved range: NO_EXPORT (0xFFFF:0xFF01), NO_ADVERTISE (0xFFFF:0xFF02), and NO_EXPORT_SUBCONFED (0xFFFF:0xFF03) interpreted by all BGP speakers. We will discuss their use in separate tasks.

Other community values do not trigger any special BGP processing unless you configure your BGP speaker to do so. The most common use of communities is to signal a neighboring AS (because the attribute is transitive) some special sort of treatment to apply to the tagged prefixes. For example, imagine a company with its own AS being a customer of two ISPs in separate ASs. Every ISP may implement a policy, in which routes tagged with a community value, say AS#:101, are prepended with ISP's AS# when advertised upstream. This allows the customer to instruct the ISPs for prepending of specific prefixes and thus adjusting BGP bestpath selection in the upstream AS. More advanced uses include implementing community-based filtering, such as using communities to signal "advertise this prefix only to directly connected peers" or using communities to signal QoS policy. As you can see, communities could be used to implement almost any configurable policy. However, the use of community values should be agreed between two ASs, because semantic interpretation is left to the administrator.

To set a community value, use the route-map command `set community <value1> <value2> ... <valueN>` or `set community additive <value1> <value2> ... <valueN>`. The former command will impose a new set of community values on the prefix. The other command will add the specified communities to the list already attached to the path. To match a community value, you must configure a *community-list* and use it in a route-map later. There are two types of lists: standard and expanded. A list could be either numbered (1–99 for standard, 100–500 for expanded) or named. Standard community list entries permit or deny a community value, such as the following:

```
ip community-list 1 permit 100:10 100:20
ip community-list 1 deny no-export
```

For an entry to match, all mentioned community values must exist in the prefix. For example, the first line in the example above would match only prefixes with two community values: 100:10 and 100:20. Essentially, OR logic is implemented by setting multiple entries, and AND logic is implemented for values in a single line.

Expanded community lists allow the use of regular expressions for community matching. This is helpful when you need to match a range of community values. Before applying an expanded community-list, BGP engine will order the

https://t.me/learningnets

communities for every prefix numerically (because the communities are just 32 bit values) and remove the duplicates. This allows for deterministic construction of regular expressions. We will illustrate the use of the most common wildcard characters:

`^100:1_200:1` Match communities 100:1 and 200:1 in the beginning (^ anchor) of the community list. Here, "_" means the "space" separating different community values. Notice that the prefix may not have any community values ranged between 100:1 and 200:1 because of the community ordering. Also, the prefix may not have any community less than 100:1 because this is the first community in the list.

`300:2$` Match the community 300:2 at the end of the community list ($ anchor). The prefix may not have any community greater than 300:2 because this is the last value in the list.

`400:[2-9]_` Match a range of communities ([] specify a range) such as 400:1, 400:2,..., 400:9. The use of "_" at the end is important; without it, this pattern would also match 400:22, 400:2333, and so on. This may be one of the most useful wildcard constructs for matching community ranges.

`100:1.*_` Illustrates the use of two wildcard characters. The "." matches any digit (you may also use [0-9] in this context), and "*" means "repeat the previous pattern zero or more times." The pattern above would match 100:1, 100:2, 100:22, 100:11, and so on. You may use "+" instead of "*", which means "repeat one or more times." Thus, `100:1.+_` will not match against 100:1.

`100:([0-9]2)+_` Demonstrates the use of "()" for grouping. Here, "[0-9]2" is treated as a single group for the operator "+". Thus, the pattern would match 100:1212, 100:2222, 100:0202, and so on.

`100:1_|100:2_` Here we use the alternation symbol "|". The pattern would match either 100:1 or 100:2. You may use "|" with grouping such as 100:(12)|(22) that will match 100:12 or 100:22.

In spite of expanded community lists flexibility, you usually need only the standard lists, because the number of destination groups is usually limited. In our scenario, AS 100 advertises to its peers that the community 100:200 is treated as having local preference of 200 inside the AS. To implement this policy, we configure inbound route-maps on R1,R4 and R6 matching the community 100:200 and setting the local-preference to 200. R5 matches the prefixes originated in AS 60 and marks them with the community value of 100:200, signaling the preferred path.

```
R5:
no ip as-path access-list 1
ip as-path access-list 1 permit 60$
!
```

https://t.me/learningnets

```
route-map SET_COMMUNITY permit 10
 match as-path 1
 set community 100:200
!
route-map SET_COMMUNITY permit 100

router bgp 200
 neighbor 155.1.45.4 send-community
 neighbor 155.1.45.4 route-map SET_COMMUNITY out
```

R1:

```
ip community-list standard 100:200 permit 100:200
!
route-map SET_LOCAL_PREFERENCE permit 10
 match community 100:200
 set local-preference 200
!
route-map SET_LOCAL_PREFERENCE permit 100
!
router bgp 100
 neighbor 155.1.13.3 route-map SET_LOCAL_PREFERENCE in
```

R4:

```
ip community-list standard 100:200 permit 100:200
!
route-map SET_LOCAL_PREFERENCE permit 10
 match community 100:200
 set local-preference 200
!
route-map SET_LOCAL_PREFERENCE permit 100
!
router bgp 100
 neighbor 155.1.45.5 route-map SET_LOCAL_PREFERENCE in
```

R6:

```
ip community-list standard 100:200 permit 100:200
!
route-map SET_LOCAL_PREFERENCE permit 10
 match community 100:200
 set local-preference 200
!
route-map SET_LOCAL_PREFERENCE permit 100
!
router bgp 100
```

https://t.me/learningnets

```
neighbor 155.1.67.7 route-map SET_LOCAL_PREFERENCE in
```

Verification

Check the paths toward prefixes originated in AS 60 on R4. Notice the local preference of 200 associated with the path via R5:

```
R4#show ip bgp regexp 60$
BGP table version is 40, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>   112.0.0.0        155.1.45.5                      200      0 200 54 50 60 i
*>   113.0.0.0        155.1.45.5                      200      0 200 54 50 60 i
```

Because local preference is passed throughout the AS, all other routers in AS100 will use R4 as the exit point out of AS100 for all prefixes originated in AS 60. Check R6, for example:

```
R6#show ip bgp regexp 60$
BGP table version is 32, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   112.0.0.0        155.1.67.7                               0 300 54 50 60 i
*>i                   155.1.45.5               0     200      0 200 54 50 60 i
 *   113.0.0.0        155.1.67.7                               0 300 54 50 60 i
*>i                   155.1.45.5               0     200      0 200 54 50 60 i
```

https://t.me/learningnets

Look at the BGP attributes for 113.0.0.0/8 on R4. Notice the local preference and the community value. The community is presented in unstructured format. Configure the router to display the communities in structured format:

```
R4#show ip bgp 113.0.0.0

BGP routing table entry for 113.0.0.0/8, version 40
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 3
  200 54 50 60
    155.1.45.5 from 155.1.45.5 (150.1.5.5)      Origin IGP, localpref 200
, valid, external, best Community: 6553800
      rx pathid: 0, tx pathid: 0x0
!R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#ip bgp-community new-format

!R4#show ip bgp 113.0.0.0
BGP routing table entry for 113.0.0.0/8, version 40
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 3
  200 54 50 60
    155.1.45.5 from 155.1.45.5 (150.1.5.5)      Origin IGP, localpref 200
, valid, external, best Community: 100:200

      rx pathid: 0, tx pathid: 0x0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Communities - No-Advertise

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Configure R2 so that it does not advertise prefixes received from AS 254 to any peer.
- Do not use any sort of prefix filtering to accomplish this.

Configuration

The well-known NO_ADVERTISE BGP community signals the BGP speaker not to advertise the particular prefix to any BGP peer. This may be useful to limit the scope of routing information to just directly connected neighbors. In our scenario, we set the community attribute inbound on prefixes received from R10. This makes R2 think that the prefixes should not be advertised to any other peers.

```
R2:

route-map SET_COMMUNITY
 set community no-advertise
!
router bgp 200
 neighbor 192.10.1.254 route-map SET_COMMUNITY in
```

Verification

https://t.me/learningnets

Verify the BGP prefixes received from AS 254. Look at the detailed information for any of the prefixes. Confirm that community no-advertise prevents routes from being advertised to any peer:

```
R2#show ip bgp regexp _254$
BGP table version is 26, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>  51.51.51.51/32   192.10.1.254             0             0 254 ?
r>  192.10.1.0       192.10.1.254             0             0 254 ?
*>  205.90.31.0      192.10.1.254             0             0 254 ?
*>  220.20.3.0       192.10.1.254             0             0 254 ?
*>  222.22.2.0       192.10.1.254             0             0 254 ?
!R2#clear ip bgp * soft

!R2#show ip bgp 205.90.31.0
BGP routing table entry for 205.90.31.0/24, version 29Paths: (1 available, best #1, table default,
not advertised to any peer
)
  Not advertised to any peer
  Refresh Epoch 2
  254
    192.10.1.254 from 192.10.1.254 (31.3.0.1)
      Origin incomplete, metric 0, localpref 100, valid, external, best Community: no-advertise

      rx pathid: 0, tx pathid: 0x0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Communities - No-Export

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Configure R2 so that AS 254 prefixes received from R10 are constrained to stay within AS 200.

Configuration

The well-known NO_EXPORT community instructs the BGP speaker to advertise the prefix only across iBGP peering links. This restricts the prefix to remain within the boundaries of the local AS. One good use of this feature is prefix aggregation. Imagine that your local AS has multiple connections to some other AS. You advertise a summary of all your internal prefixes using the `aggregate-address` command out of all links. However, you want just the adjacent AS to select the best entry point based on the MED attribute. This could be accomplished by advertising the specific prefixes tagged with NO_EXPORT community along with the aggregates. The neighboring AS would be able to select the best path based on the specific information (MED) but will not advertise the specifics any further, thus containing the routing information.

In our scenario, we simply tag all prefixes received from R10 with the community NO_EXPORT. Because the community value must propagate to all peers, we enable sending community in all AS 200's BGP speakers. There is no need to send BGP communities to all speakers; just make sure the border speakers always receive the community-tagged routes.

https://t.me/learningnets

```
R2:

route-map SET_COMMUNITY permit 10
 no set community
 set community no-export
!
router bgp 200
 neighbor 155.1.23.3 send-community
 neighbor 155.1.0.5 send-community
 neighbor 192.10.1.254 route-map SET_COMMUNITY in

R3:
router bgp 200
 neighbor 155.1.0.5 send-community

R5:

router bgp 200
 neighbor 155.1.0.3 send-community
```

Verification

Find the prefixes learned from AS 254. Check these prefixes in the BGP tables of R3 and R5. Make sure they are tagged with no-export community. For example, we verify one prefix: 205.90.31.0/24. Recall that you may need to do a route-refresh to apply the new policy (adding communities) to the routes from AS 254:

```
R2#clear ip bgp * soft

!R2#show ip bgp regexp 254$
BGP table version is 36, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  51.51.51.51/32   192.10.1.254             0             0 254 ?
 r>  192.10.1.0       192.10.1.254             0             0 254 ?
 *>  205.90.31.0      192.10.1.254             0             0 254 ?
 *>  220.20.3.0       192.10.1.254             0             0 254 ?
 *>  222.22.2.0       192.10.1.254             0             0 254 ?
!R3#show ip bgp 205.90.31.0
```

https://t.me/learningnets

```
BGP routing table entry for 205.90.31.0/24, version 45 Paths: (2 available, best #1, table default,
not advertised to EBGP peer
)
  Advertised to update-groups:
     2          4
  Refresh Epoch 3
  254, (Received from a RR-client)
    192.10.1.254 (metric 2560000512) from 155.1.23.2 (150.1.2.2)
      Origin incomplete, metric 0, localpref 100, valid, internal, best Community: no-export
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 2
  254
    192.10.1.254 (metric 2560000512) from 155.1.0.5 (150.1.5.5)
      Origin incomplete, metric 0, localpref 100, valid, internal Community: no-export
      Originator: 150.1.2.2, Cluster list: 150.1.5.5
      rx pathid: 0, tx pathid: 0
!R5#show ip bgp 205.90.31.0
BGP routing table entry for 205.90.31.0/24, version 39 Paths: (2 available, best #2, table default,
not advertised to EBGP peer
)
  Advertised to update-groups:
     2          4
  Refresh Epoch 2
  254
    192.10.1.254 (metric 2560001280) from 155.1.0.3 (150.1.3.3)
      Origin incomplete, metric 0, localpref 100, valid, internal Community: no-export
      Originator: 150.1.2.2, Cluster list: 150.1.3.3
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 3
  254, (Received from a RR-client)
    192.10.1.254 (metric 2560001280) from 155.1.0.2 (150.1.2.2)
      Origin incomplete, metric 0, localpref 100, valid, internal, best Community: no-export

      rx pathid: 0, tx pathid: 0x0
```

Confirm that AS 254 prefixes are not among the prefixes advertised by AS 200 to other ASs. Check this by inspecting the advertised routes on AS 200 border peers:

```
R3#show ip bgp nei 155.1.13.1 advertised-routes

BGP table version is 47, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
```

https://t.me/learningnets

```
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 *>i 112.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 113.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 114.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 115.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 116.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 117.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 118.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 119.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 155.1.0.0        155.1.58.8               0    100      0 i

Total number of prefixes 11
!R3#show ip bgp nei 155.1.37.7 advertised-routes

BGP table version is 47, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 *>i 112.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 113.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 114.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 115.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 116.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 117.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 118.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 119.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 155.1.0.0        155.1.58.8               0    100      0 i

Total number of prefixes 11
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Communities - Local-AS

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Modify the routers in AS 100 so that R1 and R4 are in the same BGP sub-confederation, using AS 65014:
  - R6 should be in the sub-confederation 65006.
  - Keep AS 100 as the public AS that is used to peer with other external ASs.
- Advertise R4's Loopback0 prefix in BGP, but make sure that inside AS 100 only R1 receives it.

Configuration

The well-known community Local-AS, or NO_EXPORT_SUBCONFED in IETF RFC terms, serves the same purpose as the NO_EXPORT community, but within a sub-confederation boundaries. That is, prefixes tagged by this community are not advertised to external sub-confederation peers (that is, peers in other sub-confederations) AND to regular eBGP peers. In effect, the prefix is contained within a single sub-confederation. The use of Local-AS community is the same as of NO_EXPORT community, but only within the single confederation boundaries. For example, you may use it for routing optimization toward prefixes aggregated within a sub-confederation.

https://t.me/learningnets

In our example, R4 advertises its local Loopback0 subnet into BGP and tags it with the Local-AS community. This prevents the prefix from leaking out of AS 65014 boundaries.

```
R1:
no router bgp 100
router bgp 65014
 bgp confederation identifier 100
 bgp confederation peers 65006
 neighbor 155.1.13.3 remote-as 200
 neighbor 155.1.146.4 remote-as 65014
 neighbor 155.1.146.6 remote-as 65006

R4:
route-map SET_COMMUNITY
 set community local-as
!
no router bgp 100
router bgp 65014
 bgp confederation identifier 100
 neighbor 155.1.45.5 remote-as 200
 neighbor 155.1.146.1 remote-as 65014
 neighbor 155.1.146.1 send-community
 network 150.1.4.4 mask 255.255.255.255 route-map SET_COMMUNITY

R6:

no router bgp 100
router bgp 65006
 bgp confederation identifier 100
 bgp confederation peers 65014
 neighbor 155.1.67.7 remote-as 300
 neighbor 155.1.146.1 remote-as 65014
```

Verification

Verify that the prefix is tagged with the community value of Local-AS and is not advertised to eBGP peers on R4:

```
R4#show ip bgp 150.1.4.4
BGP routing table entry for 150.1.4.4/32, version 12Paths: (1 available, best #1, table default,
not advertised outside local AS
)
  Advertised to update-groups:
```

https://t.me/learningnets

```
    1
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (150.1.4.4)
      Origin IGP, metric 0, localpref 100, weight 32768, valid, sourced, local, best Community: local-AS

      rx pathid: 0, tx pathid: 0x0
```

Now confirm that the prefix gets into R1's BGP table and is not advertised to any other peer, such as R6 or another eBGP neighbor such as R3:

```
R1#show ip bgp 150.1.4.0
BGP routing table entry for 150.1.4.4/32, version 12 Paths: (1 available, best #1, table default,
not advertised outside local AS
, RIB-failure(17))
  Not advertised to any peer
  Refresh Epoch 1
  Local
    155.1.146.4 from 155.1.146.4 (150.1.4.4)
      Origin IGP, metric 0, localpref 100, valid, confed-internal, best Community: local-AS
      rx pathid: 0, tx pathid: 0x0
!R1#show ip bgp neighbors 155.1.146.6 advertised-routes

BGP table version is 13, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  28.119.16.0/24   155.1.13.3                             0 200 54 i
 *>  28.119.17.0/24   155.1.13.3                             0 200 54 i
 *>  112.0.0.0        155.1.13.3                             0 200 54 50 60 i
 *>  113.0.0.0        155.1.13.3                             0 200 54 50 60 i
 *>  114.0.0.0        155.1.13.3                             0 200 54 i
 *>  115.0.0.0        155.1.13.3                             0 200 54 i
 *>  116.0.0.0        155.1.13.3                             0 200 54 i
 *>  117.0.0.0        155.1.13.3                             0 200 54 i
 *>  118.0.0.0        155.1.13.3                             0 200 54 i
 *>  119.0.0.0        155.1.13.3                             0 200 54 i
 *>  155.1.0.0        155.1.13.3                             0 200 i

R1#show ip bgp neighbors 155.1.13.3 advertised-routes
```

https://t.me/learningnets

```
 Total number of prefixes 0
```

Check on R6:

```
R6#show ip bgp 150.1.4.0
% Network not in table
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Communities - Deleting

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Configure R2 to tag prefixes received from AS 254 with the community values “254:200”, “200:254”, and “200:123”.
- Configure AS 300 to add the community value 300:200 to the list of communities and send them to AS 100.
- Configure AS 300 to remove any communities attached by AS 200, (communities starting with “200:”).

Configuration

In complicated community signaling environments, it may become necessary to remove a subset of communities associated with a prefix. For example, when passing a transit prefix to another AS, you may want to remove community values attached by downstream AS, yet retain community values attached by the AS of origin. Or you may want to delete the no-export community for a prefix, while leaving other communities intact.

Deleting a subset of community list is possible using a special IOS feature. The configuration is as follows. First, you create a community access-list (standard or expanded). This list specifies the communities to be removed. It is flexible to use expanded access-lists to be able to remove community ranges—for example, by matching “200:[0-9]+_” you will erase any community set in AS 200. Then, you create a route-map to delete the communities using the following syntax: `set comm-list [<NAME>|<NUMBER>] delete` . You may set your own communities while

https://t.me/learningnets

deleting the others.

In our scenario, R7 is configured to remove any communities matching the pattern “200:[0-9]+” and attach its own community “300:200”. This is performed using a single route-map entry.

```
R2:
route-map SET_COMMUNITY 10
 no set community
 set community 200:254 254:200 200:123
!
router bgp 200
 neighbor 155.1.23.3 send-community
 neighbor 155.1.0.5 send-community
 neighbor 192.10.1.254 route-map SET_COMMUNITY in
!
ip bgp-community new-format

R3:
router bgp 200
 neighbor 155.1.37.7 send-community
!
ip bgp-community new-format

R6:
ip bgp-community new-format

R7:

ip community-list expanded AS200 permit 200:[0-9]+_
!
route-map RESET_COMMUNITY permit 10
 set community 300:200 additive
 set comm-list AS200 delete
!
router bgp 300
 neighbor 155.1.67.6 send-community
 neighbor 155.1.37.3 route-map RESET_COMMUNITY in
!
ip bgp-community new-format
```

Verification

https://t.me/learningnets

Trace any prefix from AS 254 through the BGP tables of R3, R7, and R6. Notice how the BGP communities associated with the prefix change every time.

```
R3#show ip bgp 205.90.31.0
BGP routing table entry for 205.90.31.0/24, version 50
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     1          2          4          5
  Refresh Epoch 5
  254, (Received from a RR-client)
    192.10.1.254 (metric 2560000512) from 155.1.23.2 (150.1.2.2)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
Community: 200:123 200:254 254:200
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 2
  254
    192.10.1.254 (metric 2560000512) from 155.1.0.5 (150.1.5.5)
      Origin incomplete, metric 0, localpref 100, valid, internal Community: 200:123 200:254 254:200

      Originator: 150.1.2.2, Cluster list: 150.1.5.5
      rx pathid: 0, tx pathid: 0
```

R7 deletes all community values starting with “200:”.

```
R7#show ip bgp 205.90.31.0
BGP routing table entry for 205.90.31.0/24, version 37
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 3
  200 254
    155.1.37.3 from 155.1.37.3 (150.1.3.3)
      Origin incomplete, localpref 100, valid, external, best Community: 254:200 300:200

      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  100 200 254
    155.1.67.6 from 155.1.67.6 (150.1.6.6)
      Origin incomplete, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
```

https://t.me/learningnets

The same set of communities is associated with the prefix in R6's BGP table. Notice that only the prefix learned via AS 300 is tagged.

```
R6#show ip bgp 205.90.31.0
BGP routing table entry for 205.90.31.0/24, version 15
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     2
  Refresh Epoch 2
  300 200 254
    155.1.67.7 from 155.1.67.7 (150.1.7.7)
      Origin incomplete, localpref 100, valid, external Community: 254:200 300:200

      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  (65014) 200 254
    155.1.13.3 (metric 3072) from 155.1.146.1 (150.1.1.1)
      Origin incomplete, metric 0, localpref 100, valid, confed-external, best
      rx pathid: 0, tx pathid: 0x0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Conditional Advertisement

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Configure R3 in such a way that AS 300 uses AS 100 to get to all prefixes learned from AS 254.
- If the link between R1 and R3 goes down, traffic from AS 300 to AS 254 should be rerouted directly to AS 200.

Configuration

Conditional advertisement allows the BGP speaker to advertise a set of BGP prefixes to a peer only if certain other prefixes are present or not present in the local BGP table. Thus, the existence (or non-existence) of those “trigger” prefixes is used as a condition to advertise selected prefixes to a peer. Conditional advertisements are helpful in situations with multiple uplinks to different ASs. For example, assume that your AS is connected to a pair of ISPs. One ISP charges you more, so you only want to use it in case of emergency with the primary ISP. Based on this, you want to advertise your local prefixes to the “expensive” ISP only if the primary fails. To detect the primary ISP failure, you track a certain prefix learned from the primary ISP. This allows for more sophisticated tracking of the connectivity issue, compared to simply tracking the interface state. When the tracked prefix is gone, all local prefixes are announced to the “expensive” ISP until the moment the tracked prefix appears again.

The syntax for conditional advertisement is as follows:** `neighbor <IP> advertise-map MAP1 {non-exist|exist-map} MAP2`

https://t.me/learningnets

**. The configuration involves defining two route-maps. One route-map (MAP1) selects the prefixes to be advertised to the peer. These prefixes must already exist in the local BGP table. The other route-map (MAP2) selects the prefixes to be tracked in the local BGP table. If this is a “non-exist” map, the condition is triggered when no prefixes in the BGP table match the route-map. If this is an “exist” map, the condition is triggered when there is a prefix in the BGP table matching the route-map. The BGP process performs condition verification every time the BGP scanner runs (60 seconds by default), so it may take some time after your configuration change before the conditional advertisement occurs.

In our scenario, we advertise the link connecting R1 and R3 into BGP. We then create a route-map matching this prefix. This route-map is used as a “non-exist” condition for the advertisement of AS 254 prefixes. The prefixes are selected using an AS_PATH access-list matching the regular expression “254$”.

```
R3:

ip as-path access-list 1 permit 254$
!
route-map ADVERTISE_MAP permit 10
 match as-path 1
!
ip prefix-list LINK_R1_R3 permit 155.1.13.0/24
!
route-map NON_EXIST_MAP permit 10
 match ip address prefix-list LINK_R1_R3
!
router bgp 200
 network 155.1.13.0 mask 255.255.255.0
 neighbor 155.1.37.7 advertise-map ADVERTISE_MAP non-exist-map NON_EXIST_MAP
```

Verification

Check the “normal” conditions, when the link between R1 and R3 is up. The prefix is in R3’s BGP table:

```
R3#show ip bgp 155.1.13.0
BGP routing table entry for 155.1.13.0/24, version 18
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     1          2          3          4
  Refresh Epoch 1
  Local 0.0.0.0 from 0.0.0.0 (150.1.3.3)
```

https://t.me/learningnets

```
      Origin IGP, metric 0, localpref 100, weight 32768, valid, sourced, local, best
      rx pathid: 0, tx pathid: 0x0
```

AS 254 prefixes are not advertised to R7, even though they are in the local BGP table:

```
R3#show ip bgp regexp 254$
BGP table version is 23, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
 r i 192.10.1.0       192.10.1.254             0    100      0 254 ?
 r>i                  192.10.1.254             0    100      0 254 ?
 * i 205.90.31.0      192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
 * i 220.20.3.0       192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
 * i 222.22.2.0       192.10.1.254             0    100      0 254 ?
 *>i                  192.10.1.254             0    100      0 254 ?
!R3#show ip bgp neighbors 155.1.37.7 advertised-routes

BGP table version is 23, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 *>i 112.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 113.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 114.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 115.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 116.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 117.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 118.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 119.0.0.0        155.1.108.10             0    100      0 54 i
```

https://t.me/learningnets

```
 *>i 155.1.0.0        155.1.58.8               0    100      0 i
 *>  155.1.13.0/24    0.0.0.0                  0         32768 i

Total number of prefixes 12
```

Check the state of the advertise map associated with R7. Notice that the status is “Withdraw,” meaning that prefixes matching the advertise-map are not advertised to R7:

```
R3#show ip bgp neighbors 155.1.37.7
BGP neighbor is 155.1.37.7,  remote AS 300, external link
<snip>  Condition-map NON_EXIST_MAP, Advertise-map ADVERTISE_MAP, status:Withdraw

<snip>
```

Disable the link connecting R1 and R3, and check the prefixes advertised to R7. Notice that AS254 prefixes are now advertised:

```
R1:
interface GigabitEthernet 1.13
 shutdown

R3:

interface GigabitEthernet 1.13
 shutdown
```

Note that in this scenario, we need to disable both sides of the link. The devices being used are not connected via a true point-to-point link, so if only one side is shutdown, the other side will stay up and will keep advertising the link into IGP. R3 will eventually learn about its own connected route via IGP and will pull it into the BGP table by virtue of the network command for 155.1.13.0/24. This will make the 155.1.13.0/24 network appear in the BGP table and the non-exist map will never trigger. Shutting down the link on both sides would not be necessary if the routers were connected via a true point-to-point link.

Ensure that 155.1.13.0/24 is not in the BGP table of R3 and that R7 is now receiving AS254 prefixes via R3:

```
R3#show ip bgp 155.1.13.0/24
% Network not in table
!R3#show ip bgp neighbors 155.1.37.7 advertised-routes

BGP table version is 65, local router ID is 150.1.3.3
```

https://t.me/learningnets

```
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.108.10             0    100      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    100      0 54 i
 *>i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
 *>i 112.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 113.0.0.0        155.1.108.10             0    100      0 54 50 60 i
 *>i 114.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 115.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 116.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 117.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 118.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 119.0.0.0        155.1.108.10             0    100      0 54 i
 *>i 155.1.0.0        155.1.58.8               0    100      0 i
r>i 192.10.1.0        192.10.1.254             0    100      0 254 ?
 *>i 205.90.31.0      192.10.1.254             0    100      0 254 ?
 *>i 220.20.3.0       192.10.1.254             0    100      0 254 ?
 *>i 222.22.2.0       192.10.1.254             0    100      0 254 ?

Total number of prefixes 16
!R7#show ip bgp regexp _254$
BGP table version is 73, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>   51.51.51.51/32   155.1.37.3                             0 200 254 ?
 *                    155.1.67.6                             0 100 200 254 ?
*>   192.10.1.0       155.1.37.3                             0 200 254 ?
 *                    155.1.67.6                             0 100 200 254 ?
*>   205.90.31.0      155.1.37.3                             0 200 254 ?
 *                    155.1.67.6                             0 100 200 254 ?
*>   220.20.3.0       155.1.37.3                             0 200 254 ?
 *                    155.1.67.6                             0 100 200 254 ?
*>   222.22.2.0       155.1.37.3                             0 200 254 ?
```

https://t.me/learningnets

```
 *                    155.1.67.6                          0 100 200 254 ?
```

Check the advertise-map associated with R7, and confirm that the status is now “Advertise”:

```
R3#show ip bgp neighbors 155.1.37.7

BGP neighbor is 155.1.37.7,  remote AS 300, external link
<snip>  Condition-map NON_EXIST_MAP, Advertise-map ADVERTISE_MAP, status:Advertise

<snip>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Conditional Route Injection

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that BGP Conditional Route Advertisement configuration from the previous task is removed and Ethernet link between R1 and R3 is up .
- Configure R2 with four new Loopback interfaces with the IPv4 addresses 10.0.0.1/24, 10.0.1.1/24, 10.0.2.1/24, and 10.0.3.1/24, and advertise it into BGP.
  - Configure R2 to originate an aggregate route for these networks that does not overlap any IPv4 address space.
  - Ensure that no other devices in the BGP network see the individual subnet routes of this aggregate.
- Configure BGP Conditional Route Injection on R7 and R8 as follows:
  - Traffic from AS 54 going to the subnet 10.0.1.0/24 enters via R7.
  - Traffic to the subnet 10.0.2.0/24 enters via R8.
- Do not allow the more specific routes to be advertised to R5 and R3 from R8 or to R3 and R6 from R7.

Configuration

Conditional Route Injection (CRI) is special feature that allows a BGP speaker to “de-aggregate” a particular prefix. As you know, aggregation is critical to large-scale routing for reducing routing table size and increasing stability. CRI operation is the opposite of aggregation, and its purpose is to create specific prefixes at the administrator’s discretion. This may be useful for optimizing routing by advertising

https://t.me/learningnets

some specific subnets of the aggregate across a certain path. CRI is similar to the BGP unsuppress-map feature, but it will work on any router, not just the one originating the aggregate prefix. However, because of the lack of information about the prefixes that were summarized, you must explicitly set the prefixes to be injected into the BGP table.

To configure CRI, you need two route-maps. The first route-map specifies the prefixes to be injected into the BGP table by means of the **set ip address prefix-list <MAP1>** command. The **le** and **ge** keywords in the prefix-list entries are ignored. In addition to setting the prefixes, you may also set other BGP attributes, such as Weight, Local Preference, Origin, Metric, Community list, and so on. The AS_PATH attribute is reset to an empty list, to reflect the fact that prefixes were originated in the local AS. By default, the new prefixes don't have a Local Preference value assigned and the Weight attribute is reset to zero (unlike 32768 for locally originated prefixes). This could be changed by setting these values manually. The second route-map defines the conditions that must be met for the new prefixes to be injected. This route-map must have two match statements. The first statement is **match ip address prefix-list <MAP2>** , and it matches the prefix list defining the aggregated prefix. The second statement is **match ip route-source prefix-list <NAME> .** This prefix-list should match the IP address of the BGP peer that advertised the aggregate to the local router. Remember that this is NOT the NEXT_HOP attribute of the aggregate prefix. It is the IP address used to establish the BGP session with a peer that sent the update to the local system. The two route-maps are then used as follows:

```
route bgp <AS#>
 bgp inject-map <MAP1> exist-map <MAP2>
```

The result is that prefixes matching MAP1 are injected in the local BGP table if the conditions specified by MAP2 have been met.

```
R2:
interface Loopback100
 ip address 10.0.0.1 255.255.255.0
!
interface Loopback101
 ip address 10.0.1.1 255.255.255.0
!
interface Loopback102
ip address 10.0.2.1 255.255.255.0
!
interface Loopback103
 ip address 10.0.3.1 255.255.255.0
```

https://t.me/learningnets

```
!
router bgp 200
 network 10.0.0.0 mask 255.255.255.0
 network 10.0.1.0 mask 255.255.255.0
 network 10.0.2.0 mask 255.255.255.0
 network 10.0.3.0 mask 255.255.255.0
 aggregate-address 10.0.0.0 255.255.252.0 summary-only
```

R7:

```
ip prefix-list INJECT_PREFIX permit 10.0.1.0/24
ip prefix-list AGGREGATE permit 10.0.0.0/22
ip prefix-list ROUTE_SOURCE permit 155.1.37.3/32
!
route-map INJECT_MAP permit 10
 set ip address prefix-list INJECT_PREFIX
 set origin igp
!
route-map EXIST_MAP permit 10
 match ip address prefix-list AGGREGATE
 match ip route-source ROUTE_SOURCE
!
route-map DENY_INJECT_PREFIX deny 10
 match ip address prefix-list INJECT_PREFIX
!
route-map DENY_INJECT_PREFIX permit 100
!
router bgp 300
 bgp inject-map INJECT_MAP exist-map EXIST_MAP
 neighbor 155.1.67.6 route-map DENY_INJECT_PREFIX out
 neighbor 155.1.37.3 route-map DENY_INJECT_PREFIX out
```

R8:

```
ip prefix-list INJECT_PREFIX permit 10.0.2.0/24
ip prefix-list AGGREGATE permit 10.0.0.0/22
ip prefix-list ROUTE_SOURCE permit 155.1.23.3/32
!
route-map INJECT_MAP permit 10
 set ip address prefix-list INJECT_PREFIX
 set origin igp
!
route-map EXIST_MAP permit 10
 match ip address prefix-list AGGREGATE
 match ip route-source ROUTE_SOURCE
!
route-map DENY_INJECT_PREFIX deny 10
```

https://t.me/learningnets

```
 match ip address prefix-list INJECT_PREFIX
!
route-map DENY_INJECT_PREFIX permit 100
!
router bgp 200
 bgp inject-map INJECT_MAP exist-map EXIST_MAP
 neighbor 155.1.58.5 route-map DENY_INJECT_PREFIX out
 neighbor 155.1.23.3 route-map DENY_INJECT_PREFIX out
```

Verification

Check R7 and R8's BGP routing table. Confirm that the aggregate prefix is there. Then check the paths injected into the BGP table. Notice that the NEXT_HOP attribute for these prefixes is taken from the aggregate prefix:

```
R7#show ip bgp 10.0.0.0 255.255.252.0
BGP routing table entry for 10.0.0.0/22, version 80
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 7
  100 200, (aggregated by 200 150.1.2.2)
    155.1.67.6 from 155.1.67.6 (150.1.6.6)      Origin IGP, localpref 100, valid, external,
atomic-aggregate
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 3
  200, (aggregated by 200 150.1.2.2) 155.1.37.3 from 155.1.37.3
 (150.1.3.3)      Origin IGP, localpref 100, valid, external, atomic-aggregate, best
      rx pathid: 0, tx pathid: 0x0
!R8#shoe ip bgp 10.0.0.0 255.255.252.0
BGP routing table entry for 10.0.0.0/22, version 171
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     2
  Refresh Epoch 5
  Local, (aggregated by 200 150.1.2.2) 155.1.23.2 (metric 3840) from 155.1.23.3
 (150.1.3.3)      Origin IGP, metric 0, localpref 100, valid, internal, atomic-aggregate, best
      Originator: 150.1.2.2, Cluster list: 150.1.3.3
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 9
  Local, (aggregated by 200 150.1.2.2)
    155.1.0.2 (metric 25856256) from 155.1.58.5 (150.1.5.5)
      Origin IGP, metric 0, localpref 100, valid, internal, atomic-aggregate
```

https://t.me/learningnets

```
      Originator: 150.1.2.2, Cluster list: 150.1.5.5
      rx pathid: 0, tx pathid: 0
!R7#show ip bgp injected-paths

BGP table version is 86, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   10.0.1.0/24      155.1.37.3                             0 i
*>                   155.1.67.6                              0 i
!R8#show ip bgp injected-paths
BGP table version is 177, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path * i 10.0.2.0/24
      155.1.0.2                             0 i *>i                  155.1.23.2
                            0 i
```

Check the advertised prefixes. For AS 54 to select the paths to 10.0.1.0/24 and 10.0.2.0/24 via R7 and R8, respectively:

```
R7#show ip bgp neighbors 155.1.79.9 advertised-routes | include 10.0.
 *>  10.0.0.0/22      155.1.37.3                             0 200 i
 *>  10.0.1.0/24      155.1.67.6                             0 i
!R8#show ip bgp neighbors 155.1.108.10 advertised-routes | include 10.0.

 *>i 10.0.0.0/22      155.1.23.2               0    100      0 i
 *>i 10.0.2.0/24      155.1.23.2                              0 i
```

Make sure the prefixes are not advertised to R3, R5, or R6:

```
R8#show ip bgp neighbors 155.1.58.5 advertised-routes

BGP table version is 177, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
```

https://t.me/learningnets

```
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  28.119.16.0/24   155.1.108.10             0             0 54 i
 *>  28.119.17.0/24   155.1.108.10             0             0 54 i
 *>  112.0.0.0        155.1.108.10                           0 54 50 60 i
 *>  113.0.0.0        155.1.108.10                           0 54 50 60 i
 *>  114.0.0.0        155.1.108.10                           0 54 i
 *>  115.0.0.0        155.1.108.10                           0 54 i
 *>  116.0.0.0        155.1.108.10                           0 54 i
 *>  117.0.0.0        155.1.108.10                           0 54 i
 *>  118.0.0.0        155.1.108.10                           0 54 i
 *>  119.0.0.0        155.1.108.10                           0 54 i
 *>  155.1.0.0        0.0.0.0                           32768 i

Total number of prefixes 11
!R8#show ip bgp neighbors 155.1.23.3 advertised-routes
BGP table version is 177, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  28.119.16.0/24   155.1.108.10             0             0 54 i
 *>  28.119.17.0/24   155.1.108.10             0             0 54 i
 *>  112.0.0.0        155.1.108.10                           0 54 50 60 i
 *>  113.0.0.0        155.1.108.10                           0 54 50 60 i
 *>  114.0.0.0        155.1.108.10                           0 54 i
 *>  115.0.0.0        155.1.108.10                           0 54 i
 *>  116.0.0.0        155.1.108.10                           0 54 i
 *>  117.0.0.0        155.1.108.10                           0 54 i
 *>  118.0.0.0        155.1.108.10                           0 54 i
 *>  119.0.0.0        155.1.108.10                           0 54 i
 *>  155.1.0.0        0.0.0.0                           32768 i

Total number of prefixes 11
!R7#sh ip bgp neighbors 155.1.67.6 advertised-routes
BGP table version is 86, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

https://t.me/learningnets

```
     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.0.0.0/22      155.1.37.3                             0 200 i
 *>  28.119.16.0/24   155.1.79.9                             0 54 i
 *>  28.119.17.0/24   155.1.79.9                             0 54 i
 *>  51.51.51.51/32   155.1.37.3                             0 200 254 ?
 *>  112.0.0.0        155.1.79.9               0             0 54 50 60 i
 *>  113.0.0.0        155.1.79.9               0             0 54 50 60 i
 *>  114.0.0.0        155.1.79.9               0             0 54 i
 *>  115.0.0.0        155.1.79.9               0             0 54 i
 *>  116.0.0.0        155.1.79.9               0             0 54 i
 *>  117.0.0.0        155.1.79.9               0             0 54 i
 *>  118.0.0.0        155.1.79.9               0             0 54 i
 *>  119.0.0.0        155.1.79.9               0             0 54 i
 *>  155.1.0.0        0.0.0.0                            32768 i
 *>  192.10.1.0       155.1.37.3                             0 200 254 ?
 *>  205.90.31.0      155.1.37.3                             0 200 254 ?
 *>  220.20.3.0       155.1.37.3                             0 200 254 ?
 *>  222.22.2.0       155.1.37.3                             0 200 254 ?

Total number of prefixes 17
!R7#show ip bgp neighbors 155.1.37.3 advertised-routes
BGP table version is 86, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.0.0.0/22      155.1.37.3                             0 200 i
 *>  28.119.16.0/24   155.1.79.9                             0 54 i
 *>  28.119.17.0/24   155.1.79.9                             0 54 i
 *>  51.51.51.51/32   155.1.37.3                             0 200 254 ?
 *>  112.0.0.0        155.1.79.9               0             0 54 50 60 i
 *>  113.0.0.0        155.1.79.9               0             0 54 50 60 i
 *>  114.0.0.0        155.1.79.9               0             0 54 i
 *>  115.0.0.0        155.1.79.9               0             0 54 i
 *>  116.0.0.0        155.1.79.9               0             0 54 i
 *>  117.0.0.0        155.1.79.9               0             0 54 i
 *>  118.0.0.0        155.1.79.9               0             0 54 i
 *>  119.0.0.0        155.1.79.9               0             0 54 i
 *>  155.1.0.0        0.0.0.0                            32768 i
 *>  192.10.1.0       155.1.37.3                             0 200 254 ?
 *>  205.90.31.0      155.1.37.3                             0 200 254 ?
 *>  220.20.3.0       155.1.37.3                             0 200 254 ?
```

https://t.me/learningnets

```
 *>  222.22.2.0       155.1.37.3                          0 200 254 ?

Total number of prefixes 17
!R7#debug ip bgp updates 155.1.79.9 out
R7#clear ip bgp 155.1.79.9 soft out
BGP(0): updating injected prefix 10.0.1.0/24, from source prefix 10.0.0.0/22
BGP(0): updating injected prefix 10.0.1.0/24, from source prefix 10.0.0.0/22

BGP(0): retaining  injected prefix 10.0.1.0/24, from source prefix 10.0.0.0/22
BGP(0): retaining  injected prefix 10.0.1.0/24, from source prefix 10.0.0.0/22
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Filtering with Prefix-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Configure a prefix-list on R2 so that it does not accept the prefix 222.22.2.0/24 from R10:
  - This prefix-list should be applied directly to the neighbor.
- Configure a prefix-list on R7 so that it does not accept any prefixes with a subnet mask greater than /22 from R9:
  - This prefix-list should be applied through a route-map to the neighbor.

Configuration

Prefix lists are the most preferable way to filter subnets in BGP based on their IP addressing information. Prefix list is an ordered sequence of entries in which each entry specifies either a single IP prefix or a range of prefixes. Prefix lists are stored in efficient data structures, allowing for very fast lookup and information retrieval. They have certain performance benefits over the standard and extended IOS access-lists when used for prefix filtering. Here is the syntax for a typical prefix-list entry:

```
ip prefix-list <NAME>seq <Num> {permit|deny} <Subnet>/<Prefix > [ge <Length1>] [le
 <Length2>]
```

Entries in a prefix list are processed sequentially, until the first match. As soon as

https://t.me/learningnets

the match is found, the processing is stopped and the associated action is performed. The **<Subnet>/<Prefix>** pair specifies the major subnet that all prefixes matching this entry should belong to. For example, this could be 192.168.0.0/16 or 172.16.8.0/24, any valid classless prefix. The modifiers **ge** and **le** are optional and used to specify a prefix range. Specifically, a prefix matches the entry if:

1. The prefix is a subnet of **<Subnet>/<Prefix>** . That is, the prefix subnet is a subset of and prefix-length is greater than or equal to `<Prefix>` .
2. The prefix length is less than or equal to **<Length2>** . That is, if the **le** modifier is used, the prefix length must be within the **[<Prefix>, <Length2>]** range. For example, with **192.168.0.0/16 le 24** , an example of valid prefix is 192.168.2.0/24 or 192.168.0.0/22, because both prefixes are subnets to 192.168.0.0/16 and have prefix-length less than or equal to 24. However, 192.168.2.128/25 will not match the above prefix-list entry.
3. The prefix length is greater than or equal to **<Length1>** but less than 32 if the **ge** modifier is used. That is, the prefix-length should be within the **[<Length1>,32]** range. It is obvious that **<Length1>** should be greater than or equal to **<Prefix>** . Take, for example, the prefix-list entry **172.16.3.0/24 ge 25** . It would match 172.16.3.128/25, 172.16.3.0/30, and 172.16.3.1/32, but not 172.16.3.0/24.

If both **le** and **ge** modifiers are in use, the resulting prefix-length range is between **<Length1>** and **<Length2>** inclusive. For example, 172.16.0.0/16 ge 24 le 30 would match 172.16.0.0/24, 172.16.3.0/24, 172.16.3.252/30, and so on.

Two common questions with prefix-lists are how to match the default route and how to match all prefixes. The entries are permit **0.0.0.0/0** and permit **0.0.0.0/0 le 32** , respectively. The first entry matches the prefix with the prefix-length of zero and the network part of 0.0.0.0. The second entry matches any subnet of 0.0.0.0/0, which encompasses the whole IPv4 address space. Prefix lists could be applied directly to a BGP peer using the command **neighbor <IP> prefix-list <NAME [in|out]** or a route-map matching the prefix-list. The latter is preferable because it allows for more flexible policy editing.

R2:

```
ip prefix-list BLOCK_222 deny 222.22.2.0/24
ip prefix-list BLOCK_222 permit 0.0.0.0/0 le 32
!
router bgp 200
 neighbor 192.10.1.254 prefix-list BLOCK_222 in
```

https://t.me/learningnets

```
R7:

ip prefix-list SHORTER_THAN_22 permit 0.0.0.0/0 le 22
!
route-map FROM_R9 permit 100
 match ip address prefix-list SHORTER_THAN_22
!
router bgp 300
 neighbor 155.1.79.9 route-map FROM_R9 in
```

Verification

Compare the BGP table of R2 before and after you apply the prefix list:

https://t.me/learningnets

```
R2#show ip bgp regexp 254$

BGP table version is 103, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  51.51.51.51/32   192.10.1.254             0             0 254 ?
 r>  192.10.1.0       192.10.1.254             0             0 254 ?
 *>  205.90.31.0      192.10.1.254             0             0 254 ?
 *>  220.20.3.0       192.10.1.254             0             0 254 ?
 *>  222.22.2.0       192.10.1.254             0             0 254 ?

!R2#show ip bgp regexp 254$

BGP table version is 104, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  51.51.51.51/32   192.10.1.254             0             0 254 ?
 r>  192.10.1.0       192.10.1.254             0             0 254 ?
 *>  205.90.31.0      192.10.1.254             0             0 254 ?
 *>  220.20.3.0       192.10.1.254             0             0 254 ?
```

Compare the BGP tables of R7 and R8. Notice that /24 prefixes are only learned by R8, and R7 filters them out:

```
R8#show ip bgp regexp _54$
BGP table version is 179, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  28.119.16.0/24   155.1.108.10             0             0 54 i
```

https://t.me/learningnets

```
*>  28.119.17.0/24   155.1.108.10              0              0 54 i

 *>  114.0.0.0        155.1.108.10                           0 54 i
 *>  115.0.0.0        155.1.108.10                           0 54 i
 *>  116.0.0.0        155.1.108.10                           0 54 i
 *>  117.0.0.0        155.1.108.10                           0 54 i
 *>  118.0.0.0        155.1.108.10                           0 54 i
 *>  119.0.0.0        155.1.108.10                           0 54 i
 *> 118.0.0.0        54.1.1.254               0             0 54 i
 * i                 204.12.1.254             0    100      0 54 i
 *> 119.0.0.0        54.1.1.254               0             0 54 i
 * i                 204.12.1.254             0    100      0 54 i
```

Notice that R8 is learning them directly from R10. R7 still learns the /24s, but they are not accepted from R9 because of the filter. The /24s in R7's BGP table are received from R3 and R6, not directly from R9. However, other shorter prefixes that were not filtered, such as 114.0.0.0, are received from R9:

```
R7#show ip bgp regexp _54$
BGP table version is 91, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *   28.119.16.0/24 155.1.67.6
                             0 100 200 54 i  *> 155.1.37.3
                             0 200 54 i  *   28.119.17.0/24 155.1.67.6
                             0 100 200 54 i  *> 155.1.37.3
                             0 200 54 i
 *   114.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
 *   115.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
 *   116.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
 *   117.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
 *   118.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
 *   119.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
!R7#debug ip bgp updates 155.1.79.9 in
R7#clear ip bgp 155.1.79.9 soft in
```

https://t.me/learningnets

```
!
BGP: nbr_topo global 155.1.79.9 IPv4 Unicast:base (0x7F64EA6239A0:1) rcvd Refresh Start-of-RIB
BGP: nbr_topo global 155.1.79.9 IPv4 Unicast:base (0x7F64EA6239A0:1) refresh_epoch is 9
BGP(0): 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, metric 0, merged path 54, AS_PATH , community
BGP(0): 155.1.79.9 rcvd 114.0.0.0/8...duplicate ignored
BGP(0): 155.1.79.9 rcvd 115.0.0.0/8...duplicate ignored
BGP(0): 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, merged path 54, AS_P
R7#ATH BGP(0): 155.1.79.9 rcvd 28.119.16.0/24 -- DENIED due to: route-map;
BGP(0): 155.1.79.9 rcvd 28.119.17.0/24 -- DENIED due to: route-map;
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Filtering with Standard Access-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the prefix filtering configuration from the previous task is removed.
- Configure a standard access-list on R2 so that it does not accept any prefix with the address 222.22.2.0 from R10:
  - This access-list should be applied directly to the neighbor.
- Configure a standard access-list on R7 so that it does not accept any prefixes with an even number in the first octet from R9:
  - This access-list should be applied through a route-map to the neighbor.

Configuration

Using a standard access-list for BGP filtering is not as performance-effective as using prefix lists. In addition, a standard access-list does not allow you to specify the prefix-length, only the subnet numbers. However, this approach offers some unique features not available with prefix-lists, such as selecting a range of subnet numbers.

Remember that a standard access-list selects a subnet based on the pre-defined number and a wildcard mask. The wildcard mask is a 32-bit value that specifies the bits in the subnet numbers to be ignored. If a bit in the wildcard mask is 0, the respective subnet bit value is “fixed.” If a bit in the wildcard mask is 1, the respective subnet bit value could be either 0 or 1. For example, take the combination 192.168.0.0 0.0.255.255. The leading 16 bits of the subnet number are fixed at their value of 192.168. However, the remaining 16 bits could take any value, because the

https://t.me/learningnets

wildcard mask bits are all ones. Thus, the construct would match 192.168.0.1, 192.168.2.0, 192.168.32.128, and so on.

Because the wildcard mask does not represent the prefix subnet mask, you may make it discontinuous. This allows for some “oddball” filtering, such as permitting odd/even prefixes. Because of the binary nature of the wildcard mask, the number of prefixes selected is always a power of 2: 2^0, 2^1, 2^2, and so on. For example, the combination 23.0.1.0 14.0.0.255 would match 8x256 subnets (3 bits set to 1 in the first octet and 8 bits set to 1 in the last octet), such as 23.0.1.64, 21.0.1.128, 17.0.1.32, and so on. The key is to transform the subnet number into binary format and apply the wildcard mask by walking over all possible combinations.

In this scenario, we create an access-list that matches all prefixes with the odd first octet. This means that the first octet must always have the lowest-significant bit set to one. This results in the corresponding wildcard bit set to 0 all the time. All other bits don’t matter, so we can set the remaining wildcard bits to ones. The resulting combination is 1.0.0.0 254.255.255.255.

To associate the access-list with a BGP peer, use the command **`neighbor <IP> distribute-list [in|out]`** . Note that you cannot use prefix-list-based and access-list-based filtering at the same time; that is, you cannot apply the** `distribute-list/prefix-list` ** commands at the same time for the same peer. However, you may freely mix those commands in a route-map.

```
R2:
ip access-list standard BLOCK_222
 deny 222.22.2.0
 permit any
!
router bgp 200
 neighbor 192.10.1.254 distribute-list BLOCK_222 in

R7:

ip access-list standard ODD_FIRST_OCTET
 permit 1.0.0.0 254.255.255.255
!
route-map FROM_R9 permit 100
 match ip address ODD_FIRST_OCTET
!
router bgp 300
 neighbor 155.1.79.9 route-map FROM_R9 in
```

https://t.me/learningnets

Verification

Check R7's BGP table. Notice that prefixes received from R9 all have odd first octet values:

```
R7#show ip bgp regexp _54$
BGP table version is 95, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   28.119.16.0/24   155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                             0 200 54 i
 *   28.119.17.0/24   155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                             0 200 54 i
 *   114.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                             0 200 54 i
 *   115.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
 *   116.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                             0 200 54 i
 *   117.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
 *   118.0.0.0        155.1.67.6                             0 100 200 54 i
 *>                   155.1.37.3                             0 200 54 i
     Network          Next Hop            Metric LocPrf Weight Path
 *   119.0.0.0        155.1.37.3                             0 200 54 i
 *>                   155.1.79.9               0             0 54 i
```

Notice all of the routes with an even first octet getting denied in the following debug:

```
R7#debug ip bgp updates 155.1.79.9 in
R7#clear ip bgp 155.1.79.9 soft in
!
BGP: nbr_topo global 155.1.79.9 IPv4 Unicast:base (0x7F64EA6239A0:1) rcvd Refresh Start-of-RIB
BGP: nbr_topo global 155.1.79.9 IPv4 Unicast:base (0x7F64EA6239A0:1) refresh_epoch is 11
BGP(0): 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, metric 0, merged path 54, AS_PATH , community
BGP(0): 155.1.79.9 rcvd 114.0.0.0/8 -- DENIED due to: route-map;
```

https://t.me/learningnets

```
BGP(0): 155.1.79.9 rcvd 115.0.0.0/8...duplicate ignored
BGP(0): 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, merged path 54, AS_PATH
BGP(0): 155.1.79.9 rcvd 28.119.16.0/24 -- DENIED due to: route-map;
BGP(0): 155.1.79.9 rcvd 28.119.17.0/24 -- DENIED due to: route-map;
BGP(0): 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, metric 0, merged path 54, AS_PATH
BGP(0): 155.1.79.9 rcvd 116.0.0.0/8 -- DENIED due to: route-map;
BGP(0): 155.1.79.9 rcvd 117.0.0.0/8...duplicate ignored BGP(0): 155.1.79.9
rcvd 118.0.0.0/8 -- DENIED due to: route-map;
BGP(0): 155.1.79.9 rcvd 119.0.0.0/8...duplicate ignored
BGP(0: 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, metric 0, merged path 54 50 60, AS_PATH
BGP(0): 155.1.79.9 rcvd 112.0.0.0/8 -- DENIED due to: route-map;

BGP(0): 155.1.79.9 rcvd 113.0.0.0/8...duplicate ignored
BGP: nbr_topo global 155.1.79.9 IPv4 Unicast:base (0x7F64EA6239A0:1) rcvd Refresh End-of-RIB
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Filtering with Extended Access-Lists

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** in order to complete this task.

Task

- Ensure that the prefix filtering configuration from the previous task is removed.
- Configure an extended access-list on R7 as follows:
  - It does not accept any prefixes with an even third octet and with a subnet mask greater than or equal to /22 from R9.
  - This list should apply directly to the neighbor.

Configuration

Extended access-lists add more functionality to BGP prefixes filtering. In addition to matching the subnet numbers, they also allow for subnet mask matching. A typical extended access-list entry in the format `permit {proto} <src-subnet> <src-mask> <dst-subnet> <dst-mask> [options]` is treated as follows. First, the protocol field and other options are ignored. Next, the `<src-subnet> <src-mask>` pair is used to build an expression for prefix subnet matching. The pair `<dst-subnet> <dst-mask>` is used as an expression to match prefixes subnet mask.

For example, the statement `permit ip 192.168.0.0 0.0.0.255 255.255.255.0 0.0.0.255` would match any prefix with the subnet number in the range 192.168.0.0-192.168.0.255 AND having the prefix length of /24 or greater. It is possible to use more sophisticated constructs based on the wildcard bits logic, but this usually makes the configuration hard to read and interpret. Here are more examples:

https://t.me/learningnets

`permit ip 10.0.0.0 0.0.0.0 255.255.0.0 0.0.0.0` - matches 10.0.0.0/16 - Only

`permit ip 10.0.0.0 0.0.0.0 255.255.255.0 0.0.0.0` - matches 10.0.0.0/24 - Only

`permit ip 10.1.1.0 0.0.0.0 255.255.255.0 0.0.0.0` - matches 10.1.1.0/24 - Only

`permit ip 10.0.0.0 0.0.255.0 255.255.255.0 0.0.0.0` - matches 10.0.X.0/24 - Any number in the third octet of the network with a /24 subnet mask

`permit ip 10.0.0.0 0.255.255.0 255.255.255.0 0.0.0.0` - matches 10.X.X.0/24 - Any number in the second and third octet of the network with a /24 subnet mask

`permit ip 10.0.0.0 0.255.255.255 255.255.255.240 0.0.0.0` - matches 10.X.X.X/28 - Any number in the second, third, and fourth octet of the network with a /28 subnet mask

`permit ip 10.0.0.0 0.255.255.255 255.255.255.0 0.0.0.255` - matches 10.X.X.X/24 to 10.X.X.X/32 - Any number in the second, third, and fourth octet of the network with a /24 to /32 subnet mask

`permit ip 10.0.0.0 0.255.255.255 255.255.255.128 0.0.0.127` - matches 10.X.X.X/25 to 10.X.X.X/32 - Any number in the second, third, and fourth octet of the network with a /25 to /32 subnet mask

In this scenario, we create a special entry that matches only the prefixes with the even third octet AND a mask length greater than or equal than 22. The second requirement is accomplished by translating the prefix length of 22 into binary and then into the decimal form: 255.255.252.0. Now we construct the wildcard mask that permits the remaining bit to take any value and end up with 255.255.252.0 0.0.3.255.

```
R7:

ip access-list extended EVEN_3RD_MASK_GT_22
 deny ip 0.0.0.0 255.255.254.255 255.255.252.0 0.0.3.255
 permit ip any any
!
router bgp 300
 neighbor 155.1.79.9 distribute-list EVEN_3RD_MASK_GT_22 in
```

Verification

Check the BGP table in R7. Notice that the prefix 28.119.16.0/24 is not being received from R9. This is the only prefix matching the access-list (third octet even and prefix-length of 24). 28.119.16.0/24 is still received from R6 and R3, however:

```
R7#show ip bgp regexp ^54
```

https://t.me/learningnets

```
BGP table version is 23, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*   28.119.16.0/24   155.1.67.6                               0 100 200 54 i
 *>                  155.1.37.3                               0 200 54 i
 *>  28.119.17.0/24   155.1.79.9                               0 54 i
 *                   155.1.37.3                               0 200 54 i
 *>  114.0.0.0        155.1.79.9               0             0 54 i
 *                   155.1.37.3                               0 200 54 i
 *>  115.0.0.0        155.1.79.9               0             0 54 i
 *                   155.1.37.3                               0 200 54 i
 *>  116.0.0.0        155.1.79.9               0             0 54 i
 *                   155.1.37.3                               0 200 54 i
 *>  117.0.0.0        155.1.79.9               0             0 54 i
 *                   155.1.37.3                               0 200 54 i
 *>  118.0.0.0        155.1.79.9               0             0 54 i
 *                   155.1.37.3                               0 200 54 i
 *>  119.0.0.0        155.1.79.9               0             0 54 i
 *                   155.1.37.3                               0 200 54 i
!R7#debug ip bgp updates 155.1.79.9 in
R7#clear ip bgp 155.1.79.9 soft in
!
BGP: nbr_topo global 155.1.79.9 IPv4 Unicast:base (0x7F64EE941D70:1) rcvd Refresh Start-of-RIB
BGP: nbr_topo global 155.1.79.9 IPv4 Unicast:base (0x7F64EE941D70:1) refresh_epoch is 2
BGP(0): 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, metric 0, merged path 54, AS_PATH , community
BGP(0): 155.1.79.9 rcvd 114.0.0.0/8...duplicate ignored
BGP(0): 155.1.79.9 rcvd 115.0.0.0/8...duplicate ignored
BGP(0): 155.1.79.9 rcvd UPDATE w/ attr: nexthop 155.1.79.9, origin i, merged path 54, AS_PATH
BGP(0): 155.1.79.9 rcvd 28.119.16.0/24 -- DENIED due to: distribute/prefix-list;
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Regular Expressions

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Create a new Loopback1 on each device R1–R6 with IPv4 addresses in the format Y.Y.Y.Y/24, where Y is your device number, and advertise it into BGP.
- Configure an AS-Path access-list on R7 so that AS 300 cannot be used as transit for AS 100:
  - This access-list should be applied directly to R6.
- Configure a local-preference modification on R8 and R3 as follows:
  - Traffic from AS 200 going to prefixes originated in AS 54 is always sent to R8.
  - Traffic to prefixes that transit AS 54 but were not originated in AS 54 is always sent to R3.
- Configure R3 so that routes learned from AS 254 are not advertised to R1.

Configuration

Filtering based on the AS_PATH attribute is done using BGP regular expressions. Regular expressions are matched against the AS_PATH strings. Remember that AS_PATH can be constructed of the following elements: AS_SET (unordered list of AS numbers), AS_SEQUENCE (ordered list of AS numbers), AS_CONFED_SET, and AS_CONFED_SEQUENCE, which are the same elements but consist of the confederation AS numbers. For the purpose of matching, the AS_PATH attribute is viewed as a string starting with the adjacent AS number on the leftmost position, and the originating AS number in the rightmost position. When matching the

https://t.me/learningnets

AS_SET attribute, enclose the AS numbers in curly brackets and separate them with commas; for example, {100,200,300}. When matching a confederation path, enclose the AS numbers in parentheses, using backslashes to escape the special meaning of the character: “\(100\)”.

We will discuss the most useful types of regexp patterns suitable for many “real-life” situations. You may read more about BGP regular expressions basics in our blog post Understanding BGP Regular Expressions. First, recall the basic regular expression meta-characters or modifiers:

1. “.” – any character
2. “?” – repeat the previous character one or zero times
3. “*” – repeat the previous character zero or any times
4. “+” – repeat the previous character one or more times
5. “^” – match the beginning of a string
6. “$” – match the end of a string
7. “[]” – range or elements
8. “_” – match the “space” separating AS numbers OR the end of the AS_PATH list

Other important regexp features include grouping and back-referencing. You can use parentheses to group AS numbers, such as (123 124 1+), and every group is assigned a number starting from left to right. For example, in the string “1 2 (3 4) 5 6 (7 8)”, the first group is assigned the number 1 and the second group number 2. You can later “recall” the grouping by using the commands \1, \2, and so on for the group numbers. For example, the string “(1 2) 3 \1” would match “1 2 3 1 2”. You may use the pipe character “|” in addition to the grouping characters for the concept of alternation. For example, (1 2)|(5 6) would match “1 2” or “5 6”. Now the practical examples:

“^$” - means an empty AS_PATH attribute, which identifies the prefixes advertised in the local AS.

“^254_” - means prefixes received from the directly adjacent AS 254. Note that using “_” is important, because there could be another adjacent AS with the number starting with 254.

“_254_” - prefixes transiting AS 254. The “_” characters are needed to clearly separate the AS number.

“_254$” - means prefixes originated in the AS 254. This expression matches the rightmost position in the string, meaning that the expression could be of arbitrary length.

https://t.me/learningnets

“^([0-9]+)_254” - routes from the AS 254 when it’s just “one-hop” away.

“^254_([0-9]+)” - prefixes from the clients of the directly connected AS 254.

“^(254_)+([0-9]+)” - prefixes from the clients of the adjacent AS 254, accounting for the fact that AS 254 may do AS_PATH prepending.

“^254_([0-9]+_)+” - prefixes from the clients of the adjacent AS 254, accounting for the fact that the clients may do AS_PATH prepending.

^\(65100\) - prefixes learned from the confederation peer 65100.

You configure BGP regular-expression using the IP AS-PATH access-lists: **`ip as-path access-list <N> {permit|deny} <Regexp>`** . This access-list might be applied as a filter-list to a peer using the syntax: **`neighbor <IP> filter-list <N> [in|out]`** . However, the best approach is to match AS_PATH access-lists under a route-map applied to the peer (**`match as-path`**), because this allows for flexible policy editing. Features are applied in the following order:

For inbound updates:

1. route-map
2. filter-list
3. prefix-list OR distribute-list

For outbound updates:

1. prefix-list OR distribute-list
2. filter-list
3. route-map

Remember that you may test regular expressions on the BGP table by using the commands **`show ip bgp regexp`** and **`show ip bgp quote-regexp`** . The latter command allows use of the “|” character to additionally filter the output.

```
R1:
interface Loopback1
 ip address 1.1.1.1 255.255.255.0
!
router bgp 100
 network 1.1.1.0 mask 255.255.255.0

R2:
interface Loopback1
 ip address 2.2.2.2 255.255.255.0
```

https://t.me/learningnets

```
!
router bgp 200
 network 2.2.2.0 mask 255.255.255.0
```

R3:

```
interface Loopback1
 ip address 3.3.3.3 255.255.255.0
!
ip as-path access-list 1 deny _54$
ip as-path access-list 1 permit _54_
!
ip as-path access-list 2 permit _254$
!
route-map FROM_R7 permit 10
 match as-path 1
 set local-preference 200
!
route-map FROM_R7 permit 100
!
route-map TO_R1 deny 10
 match as-path 2
!
route-map TO_R1 permit 100
!
router bgp 200
 network 3.3.3.0 mask 255.255.255.0
 neighbor 155.1.37.7 route-map FROM_R7 in
 neighbor 155.1.13.1 route-map TO_R1 out
```

R4:

```
interface Loopback1
 ip address 4.4.4.4 255.255.255.0
!
router bgp 100
 network 4.4.4.0 mask 255.255.255.0
```

R5:

```
interface Loopback1
 ip address 5.5.5.5 255.255.255.0
!
!
router bgp 200
 network 5.5.5.0 mask 255.255.255.0
```

R6:

https://t.me/learningnets

```
interface Loopback1
 ip address 6.6.6.6 255.255.255.0
!
router bgp 100
 network 6.6.6.0 mask 255.255.255.0
```

R7:

```
ip as-path access-list 1 permit ^$
!
route-map NO_TRANSIT permit 100
 match as-path 1
!
router bgp 300
 neighbor 155.1.67.6 route-map NO_TRANSIT out
```

R8:

```
ip as-path access-list 1 permit _54$
!
route-map FROM_R10 permit 10
 match as-path 1
 set local-preference 200
!
route-map FROM_R10 permit 100
!
router bgp 200
 neighbor 155.1.108.10 route-map FROM_R10 in
```

Verification

Look at R7's BGP table. Even though there are prefixes learned from AS 100, they are not being advertised to R6.

```
R7#show ip bgp regexp ^200
BGP table version is 27, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   1.1.1.0/24       155.1.37.3                             0 200 100 i
 *>  2.2.2.0/24       155.1.37.3                             0 200 i
```

https://t.me/learningnets

```
 *>  3.3.3.0/24       155.1.37.3               0             0 200 i
 *>  5.5.5.0/24       155.1.37.3                             0 200 i
 *   6.6.6.0/24       155.1.37.3                             0 200 100 i
 *>  8.8.8.0/24       155.1.37.3                             0 200 i
 *   28.119.16.0/24   155.1.37.3                             0 200 54 i
 *   28.119.17.0/24   155.1.37.3                             0 200 54 i
 *>  51.51.51.51/32   155.1.37.3                             0 200 254 ?
 *   114.0.0.0        155.1.37.3                             0 200 54 i
 *   115.0.0.0        155.1.37.3                             0 200 54 i
 *   116.0.0.0        155.1.37.3                             0 200 54 i
 *   117.0.0.0        155.1.37.3                             0 200 54 i
 *   118.0.0.0        155.1.37.3                             0 200 54 i
 *   119.0.0.0        155.1.37.3                             0 200 54 i
 *   155.1.0.0        155.1.37.3                             0 200 i
 *>  192.10.1.0       155.1.37.3                             0 200 254 ?
 *>  205.90.31.0      155.1.37.3                             0 200 254 ?
 *>  220.20.3.0       155.1.37.3                             0 200 254 ?
 *>  222.22.2.0       155.1.37.3                             0 200 254 ?
!R7#show ip bgp neighbors 155.1.67.6 advertised-routes

BGP table version is 27, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  155.1.0.0        0.0.0.0                             32768 i

Total number of prefixes 1
```

Check R2’s BGP table. Notice that prefixes transit across AS 54 have NEXT_HOP pointing to R3 (R7's next hop reachable via R3). At the same time, AS 54 originated prefixes have their NEXT_HOP pointing to R8 (R10's next hop reachable via R8).

```
R2#show ip bgp quote-regexp _54_

BGP table version is 53, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

https://t.me/learningnets

```
     Network          Next Hop            Metric LocPrf Weight Path
 *>i 28.119.16.0/24   155.1.108.10             0    200      0 54 i
 * i                  155.1.108.10             0    200      0 54 i
 *>i 28.119.17.0/24   155.1.108.10             0    200      0 54 i
 * i                  155.1.108.10             0    200      0 54 i
 *>i 112.0.0.0        155.1.37.7               0    200      0 300 54 50 60 i
 * i                  155.1.37.7               0    200      0 300 54 50 60 i
 *>i 113.0.0.0        155.1.37.7               0    200      0 300 54 50 60 i
 * i                  155.1.37.7               0    200      0 300 54 50 60 i
 * i 114.0.0.0        155.1.108.10             0    200      0 54 i
 *>i                  155.1.108.10             0    200      0 54 i
 * i 115.0.0.0        155.1.108.10             0    200      0 54 i
 *>i                  155.1.108.10             0    200      0 54 i
 * i 116.0.0.0        155.1.108.10             0    200      0 54 i
 *>i                  155.1.108.10             0    200      0 54 i
 * i 117.0.0.0        155.1.108.10             0    200      0 54 i
 *>i                  155.1.108.10             0    200      0 54 i
 * i 118.0.0.0        155.1.108.10             0    200      0 54 i
 *>i                  155.1.108.10             0    200      0 54 i
 * i 119.0.0.0        155.1.108.10             0    200      0 54 i
 *>i                  155.1.108.10             0    200      0 54 i
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Filtering with Maximum Prefix

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Configure R8 so that the peering session to R10 is torn down if R8 learns more than 20 BGP prefixes from that neighbor.
  - When 16 prefixes are received from R10, R8 should begin generating warning messages.
  - When down, the peering should attempt to restart after three minutes.
- Configure R7 so that if more than 20 prefixes are learned from R3, warning messages should be generated, but the peering session should NOT be terminated.

Configuration

Filtering based on maximum-prefix number is an important BGP security feature. The number of BGP prefixes on the Internet is many hundreds of thousands. It is possible to overwhelm a BGP speaker's table by injecting too many prefixes and putting too much stress on router's CPU or memory. Cisco IOS supports special feature that limit the maximum number of prefixes received over a BGP session. The command is `neighbor <IP> maximum-prefix <Number> [<Threshold%>] [warning-only]|[restart <minutes>]` .

By default, the router permanently shuts down the session that exceeds the maximum-prefix limit specified by the <Number> parameter. The <Threshold%> value specifies the percent value applied to <Number> to generate a warning syslog message. The default threshold is 75%. For example, if the prefix limit is 1000, the warning message is generated after 750 prefixes have been learned from this

https://t.me/learningnets

neighbor.

If you don't want the session to be shut down when the maximum prefix number has been reached, you can specify the `warning-only` keyword. This instructs the router to generate two warning messages: one for 75%*<Maximum> number of prefixes and another when the <Maximum> has been crossed. Another option is to use the `restart <minutes>` parameter. With this option set, the router will tear down the session when it reaches the maximum threshold but restore it after the number of `<minutes>` has passed.

```
R8:
router bgp 200
 neighbor 155.1.108.10 maximum-prefix 20 80 restart 3

R7:

router bgp 300
 neighbor 155.1.37.3 maximum-prefix 20 warning-only
```

Verification

You may get the following message on R7 because of an excessive number of prefixes received. Also, the commands below will display the current prefix-limit settings.

```
%BGP-4-MAXPFX: Number of prefixes received from 155.1.37.3 (afi 0) reaches 20, max 20
!R7#show ip bgp neighbors 155.1.37.3 | include Maximum|Thresh
 Maximum prefixes allowed 20 (warning-only)
    Threshold for warning message 75%
  Maximum output segment queue size: 50
!R8#show ip bgp neighbors 155.1.108.10 | include Maximum|rest
    Maximum prefixes allowed 20 Threshold for warning message 80%
, restart interval 3 min
  Maximum output segment queue size: 50
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Default Routing

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Ensure that all previous BGP configuration from the previous task is removed.
- Configure R2 to originate a default route to R3 and R5 via BGP.
- This default route should be withdrawn if R2's link to R10 goes down.

Configuration

BGP default routing could be helpful when a stub AS uses just one uplink or uses primary-backup uplinks scenario. The upstream eBGP peers could be configured to advertise just the default route information. You may inject a default route into BGP by using the `network 0.0.0.0 mask 0.0.0.0` command, as long as there is a default route in the RIB. However, this will advertise the default route to all BGP neighbors. To selectively generate a default route, use the command `neighbor <IP> default-originate [route-map <CONDITION>]` . Without the route-map parameter, this command will generate a default route and send it to the configured peer. It is not required to have a matching default route in the BGP table or the RIB.

You can make the default-route advertisement conditional by associating a route-map with the statement. In this case, the default route is advertised if the route-map match conditions are satisfied. You can match ip addresses with the route-map by using either standard, extended access-list or prefix-lists. When using extended ACLs, you can match both the prefix and the subnet mask range in the same way you specify that for BGP filtering. In our scenario, we are required to advertise the default route only if the interface connecting R2 to R10 is up. This is accomplished by matching the prefix corresponding to the interface subnet 192.10.1.0/24. For

https://t.me/learningnets

more advanced scenarios, you may create reliable static routes, tracking the next hop reachability and matching the reliable prefix within the conditional route-map.

```
R2:

ip prefix-list LINK_TO_R10 permit 192.10.1.0/24
!
route-map DEFAULT permit 10
 match ip address prefix-list LINK_TO_R10
!
router bgp 200
 neighbor 155.1.23.3 default-originate route-map DEFAULT
 neighbor 155.1.0.5 default-originate route-map DEFAULT
```

Verification

Make sure both R3 and R5 receive the default route from R2.

```
R3#show ip bgp 0.0.0.0
BGP routing table entry for 0.0.0.0/0, version 60
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          2          3
  Refresh Epoch 3
  Local
    155.1.0.2 from 155.1.0.5 (150.1.5.5)
      Origin IGP, metric 0, localpref 100, valid, internal Originator: 150.1.2.2
, Cluster list: 150.1.5.5
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 3
  Local, (Received from a RR-client)    155.1.23.2 from 155.1.23.2 ( 150.1.2.2
)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
!R5#show ip bgp 0.0.0.0
BGP routing table entry for 0.0.0.0/0, version 51
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     1          2
  Refresh Epoch 3
  Local
    155.1.23.2 (metric 3584) from 155.1.0.3 (150.1.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal Originator: 150.1.2.2
, Cluster list: 150.1.3.3
```

https://t.me/learningnets

```
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 3
  Local, (Received from a RR-client)    155.1.0.2 from 155.1.0.2 (150.1.2.2
)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
```

Now configure R2 to debug BGP updates for the prefix 0.0.0.0. Disable the interface connected to R10 and observe how R2 sends BGP WITHDRAW messages to R3 and R5.

```
R2(config)#access-list 99 permit 0.0.0.0
R2#debug ip bgp updates 99
BGP updates debugging is on for access list 99 for address family: IPv4 Unicast
!R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R2(config)#interface GigabitEthernet1.210
R2(config-if)#shutdown

!
%BGP_SESSION-5-ADJCHANGE: neighbor 192.10.1.254 IPv4 Unicast topology base removed from session  Interface flap
BGP: topo global:IPv4 Unicast:base Remove_fwdroute for 0.0.0.0
BGP(0): (base) 155.1.0.5 send unreachable (format) 0.0.0.0/0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Local AS

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- AS 100 is planning transition to the AS number 146. Configure R4 and R6 to use the new AS number; R1 should still use the old AS 100.
- Ensure that all BGP peering relationships are maintained, but do not modify the configurations of any routers other than R1, R4, and R6.
- Advertise R4 and R6's Loopback0 into BGP.

Configuration

The Hide Local Autonomous System feature could be useful when migrating an autonomous system to a different AS number. When the AS has multiple eBGP peering links, it may become time consuming to negotiate the AS number change with all peering partners. In this case, you may reconfigure the local BGP speakers to use the new AS number but advertise the old AS in BGP OPEN messages and BGP updates. This could be enforced on a per-eBGP peer basis using the command `neighbor <IP> local-as <OldAS> [no-prepend]` .

The `local-as <OldAS>` command instructs the local router to advertise the `<OldAS>` number in BGP OPEN messages instead of the AS number specified with the `router bgp <NewAS>` command. In addition, all BGP prefixes advertised to this eBGP peer would have the AS numbers <OldAS> <NewAS> preprended in front of every BGP update's AS_PATH attribute. Thus, the external system may continue with the local system using the old AS number. Also, the external system will see the updates coming from the <OldAS> looking like they first transited <NewAS>. This is

https://t.me/learningnets

necessary to avoid BGP routing loops.

If you specify the **`no-prepend`** keyword, any routes *received* from the eBGP peer will not have <OldAS> prepended upon reception. By default, the AS number specified with the `local-as` command (<OldAS>) is prepended to all updates received, to avoid potential routing loops. However, this may cause problems with partial transitions, when part of your AS is using the new AS number, and another part is still using the old AS number. The routers using the old number will reject such updates because the same AS number is present in AS_PATH.

In our scenario, only R4 and R6 have been reconfigured to use the new AS number 146. R1 is still using AS 100 and has been reconfigured to peer eBGP with R4 and R6. To make R1 accept prefixes coming from other ASs that are peering with R4 and R6, we use the **`no-prepend`** keyword when peering using the local-as feature with R5 and R7.

```
R1:
router bgp 100
 no neighbor 155.1.146.4 route-reflector-client
 no neighbor 155.1.146.6 route-reflector-client
 neighbor 155.1.146.4 remote-as 146
 neighbor 155.1.146.6 remote-as 146
 neighbor 155.1.13.3 remote-as 200

R4:
no router bgp 100
router bgp 146
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.45.5 remote-as 200
 neighbor 155.1.45.5 local-as 100 no-prepend

R6:

no router bgp 100
router bgp 146
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 100
 neighbor 155.1.67.7 remote-as 300
 neighbor 155.1.67.7 local-as 100 no-prepend
```

Verification

https://t.me/learningnets

Observe how the Local-AS feature appears in the respective command's output. Then check R1's BGP table and notice that routes learned from AS 54 appear as though they transited through AS 146.

```
R6#show ip bgp neighbors 155.1.67.7 | include local
BGP neighbor is 155.1.67.7,  remote AS 300,  local AS 100 no-prepend
, external link
!R1#show ip bgp regexp _54$

BGP table version is 105, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   28.119.16.0/24   155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
 *   28.119.17.0/24   155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
 *   114.0.0.0        155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
 *   115.0.0.0        155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
 *   116.0.0.0        155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
 *   117.0.0.0        155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
 *   118.0.0.0        155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
 *   119.0.0.0        155.1.146.6                           0 146 300 54 i
 *                    155.1.146.4                           0 146 200 54 i
 *>                   155.1.13.3                            0 200 54 i
```

Prefixes advertised via eBGP to R7 (`local-as no-prepend` peer) have AS_PATH prepended with "100 146". Remember that the `no-prepend` feature applies only to

https://t.me/learningnets

inbound learned routes. All externally advertised routes still have the local-as number prepended:

```
R7#show ip bgp regexp _146$
BGP table version is 72, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>  150.1.4.4/32     155.1.37.3                               0 200 100 146 i
*>  150.1.6.6/32     155.1.67.6               0               0 100 146 i
```

Disable the “no-prepend” feature in R6 and check the BGP routes learned from AS AS300. Note that now they have the AS number 100 prepended in front of their AS_PATH attribute.

```
R6:
router bgp 146
 neighbor 155.1.67.7 local-as 100

R6#show ip bgp regexp _54$
BGP table version is 26, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   28.119.16.0/24   155.1.67.7                             0 100 300 54 i
 *>                   155.1.146.1                            0 100 200 54 i
 *   28.119.17.0/24   155.1.67.7                             0 100 300 54 i
 *>                   155.1.146.1                            0 100 200 54 i
 *   114.0.0.0        155.1.67.7                             0 100 300 54 i
 *>                   155.1.146.1                            0 100 200 54 i
 *   115.0.0.0        155.1.67.7                             0 100 300 54 i
 *>                   155.1.146.1                            0 100 200 54 i
 *   116.0.0.0        155.1.67.7                             0 100 300 54 i
 *>                   155.1.146.1                            0 100 200 54 i
 *   117.0.0.0        155.1.67.7                             0 100 300 54 i
```

https://t.me/learningnets

```
 *>                    155.1.146.1                              0 100 200 54 i
 *   118.0.0.0         155.1.67.7                               0 100 300 54 i
 *>                    155.1.146.1                              0 100 200 54 i
 *   119.0.0.0         155.1.67.7                               0 100 300 54 i
 *>                    155.1.146.1                              0 100 200 54 i
```

The AS 100 attribute in the AS_PATH prevents the AS 54 originated routes learned via R6 from being accepted by R1. R1 shows only the prefixes learned via R4. Compare the previous output on R1 before the no-prepend command mas removed to the one below. Now R1 only received AS54 routes from its direct eBGP peering with R3, and from R4. The routes from R6 are dropped as they contain AS 100 in the AS_PATH.

```
R1#show ip bgp regexp _54$
BGP table version is 107, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   28.119.16.0/24   155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
 *   28.119.17.0/24   155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
 *   114.0.0.0        155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
 *   115.0.0.0        155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
 *   116.0.0.0        155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
 *   117.0.0.0        155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
 *   118.0.0.0        155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
 *   119.0.0.0        155.1.146.4                             0 146 200 54 i
 *>                   155.1.13.3                               0 200 54 i
!R1#debug ip bgp updates 155.1.146.6 in
R1#clear ip bgp 155.1.146.6 soft in
!
BGP: nbr_topo global 155.1.146.6 IPv4 Unicast:base (0x7F413B47FC60:1) rcvd Refresh Start-of-RIB
BGP: nbr_topo global 155.1.146.6 IPv4 Unicast:base (0x7F413B47FC60:1) refresh_epoch is 2
BGP(0): 155.1.146.6 rcv UPDATE w/ attr: nexthop 155.1.146.6
 origin i, originator 0.0.0.0, merged path 146 100
 300 54 50 60, AS_PATH , community , extended community , SSA attribute
```

https://t.me/learningnets

```
BGPSSA ssacount is 0 BGP(0):
155.1.146.6 rcv UPDATE about 112.0.0.0/8 -- DENIED due to: AS-PATH contains our own AS;
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Local AS Replace-AS/Dual-AS

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Ensure that all local-as BGP configurations from the previous task are removed.
- AS 100 is planning transition to the AS number 146. Configure R1, R4, and R6 to use the new AS number.
- Configure R1 as the route-reflector of R4 and R6.
- All external Autonomous Systems should be unaware of the new AS numbers used by these routers.
- R5 should peer with R4 using the AS number 146.
- Advertise R1, R4, and R6's Loopback0 into BGP.

Configuration

Remember that when configuring the hide local AS feature, the external peers see both the local-AS and the real AS number prepended in front of the AS_PATH. Sometimes it is desirable to completely hide the “real” AS number (the one configured via the `router bgp <RealAS>` command). To accomplish this, use the `no-prepend replace-as` parameters to the `local-as` command. This combination will replace the real AS number with the one specified in the `local-as` command. The respective neighbor will be tricked into thinking that all routers are received from the AS number configured with the `local-as` command, because this number will appear in the AS_PATH and BGP OPEN message. Remember that such replacement could lead to routing loops, if the original AS were partitioned using two AS numbers.

https://t.me/learningnets

With the replace-AS feature configured, the external peer could be configured to peer using the real AS number—for example, the AS number that would be used *after* migration. In this case, it is possible to configure the “hiding” peer to initiate/accept BGP sessions using both AS numbers (the real number and the local number). The external peer will accept the correct number and negotiate the BGP session. The “hiding” peer will then use the negotiated AS number to prepend the updates sent to the external peer.

```
R1:
no router bgp 100
router bgp 146
 network 150.1.1.1 mask 255.255.255.255
 neighbor 155.1.146.4 remote-as 146
 neighbor 155.1.146.6 remote-as 146
 neighbor 155.1.146.4 route-reflector-client
 neighbor 155.1.146.6 route-reflector-client
 neighbor 155.1.13.3 remote-as 200
 neighbor 155.1.13.3 local-as 100 no-prepend replace-as

R4:
no router bgp 100
router bgp 146
 network 150.1.4.4 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 146
 neighbor 155.1.45.5 remote-as 200
 neighbor 155.1.45.5 local-as 100 no-prepend replace-as dual-as

R5:
router bgp 200
 neighbor 155.1.45.4 remote-as 146

R6:

no router bgp 100
router bgp 146
 network 150.1.6.6 mask 255.255.255.255
 neighbor 155.1.146.1 remote-as 146
 neighbor 155.1.67.7 remote-as 300
 neighbor 155.1.67.7 local-as 100 no-prepend replace-as
```

Verification

Look at R3’s and R7’s prefixes received from AS 146. Notice that only AS 100 is

https://t.me/learningnets

prepended to those prefixes by AS 146 speakers, even though the real AS is 146.

```
R3#show ip bgp neighbors 155.1.13.1 routes

BGP table version is 106, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   112.0.0.0        155.1.13.1                             0 100
 300 54 50 60 i *   113.0.0.0        155.1.13.1                             0 100
 300 54 50 60 i *>  150.1.1.1/32     155.1.13.1               0             0 100
 i *>  150.1.4.4/32     155.1.13.1                             0 100
 i *>  150.1.6.6/32     155.1.13.1                             0 100
 i

Total number of prefixes 5
!R7#show ip bgp neighbors 155.1.67.6 routes

BGP table version is 81, local router ID is 150.1.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  150.1.1.1/32     155.1.67.6                             0 100
 i *>  150.1.4.4/32     155.1.67.6                             0 100
 i *>  150.1.6.6/32     155.1.67.6               0             0 100
 i

Total number of prefixes 3
```

At the same time, R5 peers with R4 via the Dual-AS feature. All prefixes received from R4 have AS_PATH prepended with AS 146, not AS 100.

```
R5#show ip bgp neighbors 155.1.45.4 routes

BGP table version is 95, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
```

https://t.me/learningnets

```
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   112.0.0.0        155.1.45.4                             0 146
 300 54 50 60 i *   113.0.0.0        155.1.45.4                             0 146
 300 54 50 60 i *>  150.1.1.1/32     155.1.45.4                             0 146
 i *>  150.1.4.4/32     155.1.45.4               0             0 146
 i *>  150.1.6.6/32     155.1.45.4                             0 146
 i

Total number of prefixes 5
```

Reconfigure R5 for peering using the AS number 100. Then check the routes received from R4. Notice that now AS_PATH is prepended with the AS number 100, not 146.

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#router bgp 200
R5(config-router)#neighbor 155.1.45.4 remote-as 100
!R5#show ip bgp neighbors 155.1.45.4 routes

BGP table version is 104, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   112.0.0.0        155.1.45.4                             0 100
 300 54 50 60 i *   113.0.0.0        155.1.45.4                             0 100
 300 54 50 60 i *>  150.1.1.1/32     155.1.45.4                             0 100
 i *>  150.1.4.4/32     155.1.45.4               0             0 100
 i *>  150.1.6.6/32     155.1.45.4                             0 100
 i

Total number of prefixes 5
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Remove Private AS

> You must load the initial configuration files for the section, **Basic Remove Private AS**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Reconfigure R7 to be in private AS 65089, and adjust the peering settings accordingly. Disable R7's peering with R9.
- Create and advertise Loopback1 interface in R7 with the IPv4 address 7.7.7.7/24.
- Configure AS 100 and AS 200 speakers to strip the private AS number when advertising the prefixes to AS 254 and AS 54.

Configuration

Private AS numbers in the range 64512–65535 are often assigned to small enterprises that use BGP to peer with their ISPs. Private AS numbers are similar to RFC 1918 IP addressing, which allows for consuming AS numbers on the Internet. However, private AS numbers should not appear on the public Internet, because many sites may originate the same number. Thus, the AS that provides upstream connection for the private site should remove the private AS numbers from the AS_PATH attribute.

The command to perform the AS_PATH stripping in IOS is `neighbor <IP> remove-private-as` . All BGP updates sent over this session are inspected to have a sequence of private AS numbers in the beginning of the AS_PATH. All private numbers are then removed, and the local AS number is prepended. In situations where the private AS sequence is not located in the beginning of the AS_PATH, the stripping will not work and the AS_PATH will remain unmodified.

https://t.me/learningnets

```
R2:

router bgp 200
 neighbor 192.10.1.254 remove-private-as

R3:
router bgp 200
 neighbor 155.1.37.7 remote-as 65089

R6:
router bgp 146
 neighbor 155.1.67.7 remote-as 65089

R7:
no router bgp 300
router bgp 65089
 neighbor 155.1.79.9 shutdown
 neighbor 155.1.67.6 remote-as 100
 neighbor 155.1.37.3 remote-as 200
 network 7.7.7.0 mask 255.255.255.0
!
interface Loopback1
 ip address 7.7.7.7 255.255.255.0

R8:

router bgp 200
 neighbor 155.1.108.10 remove-private-as
```

Verification

Check the paths advertised to R8 on R10. Notice the AS_PATH attribute for the prefix 7.7.7.0/24. This is the output of Adj-RIBs-Out, and the result of the private AS removal and local AS prepending is not yet shown.

```
R8#show ip bgp neighbors 155.1.108.10 advertised-routes

BGP table version is 145, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

https://t.me/learningnets

```
     Network          Next Hop            Metric LocPrf Weight Path
 *>i 7.7.7.0/24       155.1.37.7               0    100      0 65089 i

 *>i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
 *>i 150.1.1.1/32     155.1.45.4               0    100      0 100 i
 *>i 150.1.4.4/32     155.1.45.4               0    100      0 100 i
 *>i 150.1.6.6/32     155.1.45.4               0    100      0 100 i
 *>  155.1.0.0        0.0.0.0                             32768 i
 r>i 192.10.1.0       192.10.1.254             0    100      0 254 ?
 *>i 205.90.31.0      192.10.1.254             0    100      0 254 ?
 *>i 220.20.3.0       192.10.1.254             0    100      0 254 ?
 *>i 222.22.2.0       192.10.1.254             0    100      0 254 ?

Total number of prefixes 10
```

Now check R10's BGP table. Notice that the prefix 7.7.7.0/24 appears with the AS_PATH of 200; the private AS number has been removed.

```
R10#show ip bgp neighbors 155.1.108.8 routes

BGP table version is 104, local router ID is 31.3.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  7.7.7.0/24       155.1.108.8                              0 200 i

 *>  51.51.51.51/32   155.1.108.8                              0 200 254 ?
 *>  150.1.1.1/32     155.1.108.8                              0 200 100 i
 *>  150.1.4.4/32     155.1.108.8                              0 200 100 i
 *>  150.1.6.6/32     155.1.108.8                              0 200 100 i
 r>  155.1.0.0        155.1.108.8              0               0 200 i
 *>  192.10.1.0       155.1.108.8                              0 200 254 ?
 *>  205.90.31.0      155.1.108.8                              0 200 254 ?
 *>  220.20.3.0       155.1.108.8                              0 200 254 ?
 *>  222.22.2.0       155.1.108.8                              0 200 254 ?

Total number of prefixes 10
```

Shut down the BGP peering session between R3 and R7. This will make AS 200 accept the prefix from AS 146 (100) with the AS_PATH "100 65089". In real life, AS

https://t.me/learningnets

100 should have removed the private AS when advertising the prefix to AS 200. However, we intentionally left this misconfiguration.

```
R7:
neighbor 155.1.37.3 shutdown

R8#show ip bgp neighbors 155.1.108.10 advertised-routes

R8#show ip bgp neighbors 155.1.108.10 advertised-routes
BGP table version is 147, local router ID is 150.1.8.8
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 7.7.7.0/24       155.1.45.4               0    100      0 100 65089 i

 *>i 51.51.51.51/32   192.10.1.254             0    100      0 254 ?
 *>i 150.1.1.1/32     155.1.45.4               0    100      0 100 i
 *>i 150.1.4.4/32     155.1.45.4               0    100      0 100 i
 *>i 150.1.6.6/32     155.1.45.4               0    100      0 100 i
 *>  155.1.0.0        0.0.0.0                             32768 i
 r>i 192.10.1.0       192.10.1.254             0    100      0 254 ?
 *>i 205.90.31.0      192.10.1.254             0    100      0 254 ?
 *>i 220.20.3.0       192.10.1.254             0    100      0 254 ?
 *>i 222.22.2.0       192.10.1.254             0    100      0 254 ?

Total number of prefixes 10
```

Check R10's BGP table and notice that the private AS has not been stripped, because it was not the first AS number in the AS_PATH.

```
R10#show ip bgp neighbors 155.1.108.8 routes
BGP table version is 105, local router ID is 31.3.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  7.7.7.0/24       155.1.108.8                              0 200 100 65089 i
```

https://t.me/learningnets

```
 *>  51.51.51.51/32   155.1.108.8                          0 200 254 ?
 *>  150.1.1.1/32     155.1.108.8                          0 200 100 i
 *>  150.1.4.4/32     155.1.108.8                          0 200 100 i
 *>  150.1.6.6/32     155.1.108.8                          0 200 100 i
 r>  155.1.0.0        155.1.108.8              0           0 200 i
 *>  192.10.1.0       155.1.108.8                          0 200 254 ?
 *>  205.90.31.0      155.1.108.8                          0 200 254 ?
 *>  220.20.3.0       155.1.108.8                          0 200 254 ?
 *>  222.22.2.0       155.1.108.8                          0 200 254 ?

Total number of prefixes 10
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Dampening

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Ensure that all BGP configuration from the previous task is removed, and that all peerings that were shut down are re-established.
- Create a Loopback1 interface in R1 with the IPv4 address 1.1.1.1/24 and advertise it into BGP.
- Configure AS 200 routers to suppress advertisement of oscillating networks.
- After a prefix flaps two times in a row, the advertisement should resume in 5 minutes.

Configuration

Network instabilities (such as unreliable links) may cause BGP prefix flapping. A flap is a network prefix going up and down or vice-versa; in other words, a change in reachability state. Prefix flapping is dangerous to network stability, because it causes network withdraws and best-path re-computations. Moreover, a flapping prefix may cause recursive route withdraws, resulting in massive BGP table changes. Two methods for reducing the impact of network instabilities are summarization (information hiding) and prefix dampening. Summarization aggregates reachability information and hides flaps of the specific prefixes constituting a summary. Dampening is the process of suppressing a flapping prefix advertisement until the moment it becomes “stable.” This introduces some “inertial” mechanism to new prefix advertisement, delaying the change announcements for oscillating prefixes.

The idea of dampening is to suppress a prefix based on the number of flaps

https://t.me/learningnets

accounted and un-suppress the prefix only after an exponentially decaying timer expires. Every time a prefix flaps, it is assigned an additive penalty value, 1000 by default. If this is just an attribute change, such as Local-Preference or AS_PATH, the penalty is halved, 500 by default. If the prefix becomes unavailable after flapping at least once, the BGP process still keeps it in the table, marked in “history” state to account for further flaps and penalty accumulation. If the accumulated penalty value exceeds the *Suppress Limit*, which defaults to 2000, the BGP process will mark the route as *damped*. Even though the prefix could be currently “active” (flapped from down to up), BGP will not advertise it to any peers. Every five seconds the BGP process exponentially decreases the penalty value assigned to the prefix. The exponential decay process has one parameter, *Half-Life* time period, which specifies the amount of time needed to decrease the current penalty to the value twice smaller. That is, the decay process follows the equation **P(t) = P(0)/2^(t/Half_Life)**, with the default *Half_Life* time of 15 minutes. As soon as the penalty falls below the *Reuse Limit*, the router will unsuppress the prefix and start advertising it again. The decision to unsuppress a prefix is made every 10 seconds.

When facing tasks similar to the current scenario, you should understand that they assume some “ideal” conditions. That is, when the scenario says “flaps two times in a row,” you may assume that the flaps are immediately one after another. This results in an accumulated penalty of 2000 and the route being damped. We now need to find the Half_Life value that will make the router reuse the prefix in 5 minutes. We take the penalty evolution equation and write it as follows:

```
P(5) = P(0)/2^(5/Half_Life)
```

And substitute P(5)=750 (the reuse limit) and P(0)=2000 (Suppress Limit). The equation then becomes:

```
2000/750=2^(5/Half_Life)
```

From this equation, we can find the Half_Life value by taking logarithm of the both sides:

```
Half_Life=5*Ln(2)/Ln(200/75)=3.5 (approximately)
```

We may round the result up to 4 minutes. That is, the task could be accomplished by setting the Half_Life value to 4 minutes. Now, the BGP command to apply the dampening parameters is **`bgp dampening [<Half_Life> <ReuseLimit> <SuppressLimit> <MaximumSuppressTime>]`**. The last parameter, **`<MaximumSuppressTime>`**, specifies the time limit to keep the prefix damped if it keeps oscillating. The default value is 4xHalf_Life

https://t.me/learningnets

or 60 minutes. The router sets the maximum penalty value based on this timer using the formula **Max_Penalty = ReuseLimit *2^(MaximumSuppressTime/Half_Life)**. You may review the current dampening parameters (if enabled) by using the command `show ip bgp dampening parameters` .

```
R2, R3, R5, R8:
router bgp 200
 bgp dampening 4 750 2000 16

R1:

!
! We adjust the advertisement interval to minimize prefix batching
! and make R1 advertise prefix changes as soon as possible
!
router bgp 100
 network 1.1.1.0 mask 255.255.255.0
 neighbor 155.1.13.3 advertisement-interval 0
!
interface Loopback1
 ip address 1.1.1.1 255.255.255.0
```

Verification

Start by checking the BGP dampening parameters in any of the AS 200 routers. Next, go to R1 and shutdown/no shutdown Loopback1 interface a few times, emulating route flaps, enough to accumulate the suppress-limit penalty in AS 200 routers.

```
R3#show ip bgp dampening parameters

  dampening 4 750 2000 16 Half-life time       : 4  mins
      Decay Time       : 620 secs
   Max suppress penalty: 12000         Max suppress time: 16 mins
   Suppress penalty    :  2000         Reuse penalty    : 750
```

Inspect the dampened path and flap statistics in R3. Notice the character “d”, indicating that the prefixes have been damped.

```
R3#show ip bgp dampening dampened-paths

BGP table version is 50, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
```

https://t.me/learningnets

```
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

   Network          From             Reuse    Path*d 1.1.1.0/24        155.1.37.7
       00:06:50 300 100 i*d 1.1.1.0/24        155.1.13.1
       00:01:14 100 i
!R3#show ip bgp dampening flap-statistics

BGP table version is 51, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

    Network          From            Flaps Duration Reuse    Path d  1.1.1.0/24        155.1.37.7
      4     00:02:37 00:08:30 300 100 *d                    155.1.13.1
      7     00:02:37 00:00:44 100
```

Check R3's BGP table for the prefix 1.1.1.0/24. Notice that the prefix appears as damped and not advertised to any peer. If you check R2's BGP table after this, you will notice that the prefix is there but not received from R3. Because the advertisement intervals of AS 100 toward AS 54 are the default, this flap will be somewhat throttled and not cause the issue that we artificially introduced by setting the advertisement interval to 0 between R1 and R3.

```
R3#show ip bgp

BGP table version is 52, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path*d  1.1.1.0/24        155.1.37.7
                            0 300 100 i*d                    155.1.13.1
               0            0 100 i
!R3#show ip bgp 1.1.1.0
BGP routing table entry for 1.1.1.0/24, version 54
Paths: (4 available, best #1, table default)
  Advertised to update-groups:
     1          2          3
```

https://t.me/learningnets

```
  Refresh Epoch 1
  54 300 100, (Received from a RR-client)
    155.1.108.10 (metric 2560001280) from 155.1.58.8 (150.1.8.8)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  54 300 100
    155.1.108.10 (metric 2560001280) from 155.1.0.5 (150.1.5.5)
      Origin IGP, metric 0, localpref 100, valid, internal
      Originator: 150.1.8.8, Cluster list: 150.1.5.5
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1  300 100, (suppressed due to dampening
)
    155.1.37.7 from 155.1.37.7 (150.1.7.7)
      Origin IGP, localpref 100, valid, external Dampinfo: penalty 2737
, flapped 4 times in 00:03:35, reuse in 00:07:30
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1  100, (suppressed due to dampening
)
    155.1.13.1 from 155.1.13.1 (150.1.1.1)
      Origin IGP, metric 0, localpref 100, valid, external Dampinfo: penalty 4787
, flapped 7 times in 00:03:35, reuse in 00:10:40
      rx pathid: 0, tx pathid: 0
!R2#show ip bgp 1.1.1.0
BGP routing table entry for 1.1.1.0/24, version 52
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     1
  Refresh Epoch 1 54 300 100
    155.1.108.10 (metric 2560001536) from 155.1.0.5 (150.1.5.5)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      Originator: 150.1.8.8, Cluster list: 150.1.5.5
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1 54 300 100

    155.1.108.10 (metric 2560001536) from 155.1.23.3 (150.1.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal
      Originator: 150.1.8.8, Cluster list: 150.1.3.3
      rx pathid: 0, tx pathid: 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Dampening with Route-Map

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Configure Loopback1 interface in R1 with the IPv4 address 1.1.1.1/24 and advertise it into BGP.
- Configure Loopback1 interface on R7 with the IPv4 address of 7.7.7.7/24 and advertise it into BGP.
- Configure AS 200 routers to suppress advertisement of oscillating networks.
- After a prefix flaps two times in a row, the advertisement should resume in 5 minutes.
- Ensure that the dampening process applies only to AS 100 originated routes and does not affect any other prefixes.

Configuration

Sometimes it might be desirable to apply dampening only to a certain set of routes, such as to the prefixes originated from the “problematic” AS. Another good example might be a set of prefixes describing critical network resources that must always be available. To account for such situations, it is possible to use a route map with the BGP dampening command to select prefixes eligible for dampening. The command is `bgp dampening route-map <MAP_NAME>` , where the route-map may match IP addresses using access-lists, prefix-lists, and as-path lists. For every route-map entry you may set specific dampening parameters using the command `set dampening <Half_Life> <ReuseLimit> <SuppressLimit> <MaximumSuppressTime>` . For this task, we create an AS_PATH access-list matching the paths originated in AS 100 and set the dampening parameters using the Half-Life value of 4 minutes.

https://t.me/learningnets

```
R2, R3, R5, R8:

ip as-path access-list 100 permit _100$
!
route-map DAMPENING
 match as-path 100
 set dampening 4 750 2000 16
!
router bgp 200
 bgp dampening route-map DAMPENING

R1:
!
! We adjust the advertisement interval to minimize prefix batching
! and make R1 advertise prefix changes ASAP
!
router bgp 100
 network 1.1.1.0 mask 255.255.255.0
 neighbor 155.1.13.3 advertisement-interval 0
!
interface Loopback1
 ip address 1.1.1.1 255.255.255.0

R7:

router bgp 300
 network 7.7.7.0 mask 255.255.255.0
!
interface Loopback1
 ip address 7.7.7.7 255.255.255.0
```

Verification

Configure R3 for BGP dampening debugging. Then configure R7 for minimal advertisement interval and do a number of consecutive shutdowns/no shutdowns for its Loopback1 interface. After this, go to R1 and perform the same series of operations on Loopback1.

https://t.me/learningnets

```
R7(config)#router bgp 300
R7(config-router)#neighbor 155.1.37.3 advertisement-interval 0

! R3#debug ip bgp dampening

BGP dampening debugging is on for address family: IPv4 Unicast
```

Check BGP dampening settings in R3. Notice that the dampening settings apply only to the prefixes matching the route map. Then check the output for the debugging command activated above. Notice that the only prefix tracked by the dampening process is 1.1.1.0/24.

```
R3#show ip bgp dampening parameters

 dampening 4 750 2000 16 (route-map DAMPENING 10) Half-life time      : 4  mins
       Decay Time       : 620 secs
  Max suppress penalty: 12000         Max suppress time: 16 mins
  Suppress penalty    :  2000         Reuse penalty    : 750
 EvD: charge penalty 1000, new accum. penalty 1000, flap count 1
EvD: unsuppress item left in reuse timer array with penalty 1000
BGP(0): charge penalty for 1.1.1.0/24 path 100 with halflife-time 4 reuse/suppress 750/2000
BGP(0): flapped 1 times since 00:00:00. New penalty is 1000
EvD: charge penalty 1000, new accum. penalty 1000, flap count 1
EvD: unsuppress item left in reuse timer array with penalty 1000
BGP(0): charge penalty for 1.1.1.0/24 path 300 100 with halflife-time 4 reuse/suppress 750/20
BGP(0): flapped 1 times since 00:00:00. New penalty is 1000
EvD: accum. penalty 1000, not suppressed
EvD: accum. penalty decayed to 1000 after 4 second(s)
EvD: charge penalty 1000, new accum. penalty 2000, flap count 2
EvD: unsuppress item left in reuse timer array with penalty 2000
BGP(0): charge penalty for 1.1.1.0/24 path 100 with halflife-time 4 reuse/suppress 750/2000
BGP(0): flapped 2 times since 00:00:04. New penalty is 2000
EvD: accum. penalty 1971, not suppressed
EvD: accum. penalty decayed to 1942 after 8 second(s)
EvD: charge penalty 1000, new accum. penalty 2942, flap count 3
EvD: unsuppress item left in reuse timer array with penalty 2942
BGP(0): charge penalty for 1.1.1.0/24 path 100 with halflife-time 4 reuse/suppress 750/2000
BGP(0): flapped 3 times since 00:00:13. New penalty is 2942
EvD: accum. penalty decayed to 971 after 14 second(s)
EvD: accum. penalty decayed to 2942 after 1 second(s)
BGP(0): suppress 1.1.1.0/24 path 100 for 00:07:40 (penalty 2858)
halflife-time 4, reuse/suppress 750/2000
EvD: accum. penalty 2858, now suppressed with a reuse intervals of 46
EvD: accum. penalty 929, not suppressed
```

https://t.me/learningnets

```
EvD: accum. penalty decayed to 2776 after 13 second(s)
EvD: charge penalty 1000, new accum. penalty 3776, flap count 4
EvD: accum. penalty 3776, now suppressed with a reuse intervals of 56
BGP(0): charge penalty for 1.1.1.0/24 path 100 with halflife-time 4 reuse/suppress 750/2000

BGP(0): flapped 4 times since 00:00:38. New penalty is 3776
EvD: accum. penalty 3776, now suppressed with a reuse intervals of 56
EvD: accum. penalty decayed to 3776 after 3 second(s)
EvD: charge penalty 1000, new accum. penalty 4776, flap count 5
EvD: accum. penalty 4776, now suppressed with a reuse intervals of 64
BGP(0): charge penalty for 1.1.1.0/24 path 100 with halflife-time 4 reuse/suppress 750/2000
BGP(0): flapped 5 times since 00:00:41. New penalty is 4776
EvD: accum. penalty 4707, now suppressed with a reuse intervals of 64
EvD: accum. penalty decayed to 815 after 45 second(s)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Timers Tuning

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Ensure that the BGP configuration from the last task is removed, including the advertisement interval on R7.
- Configure R2's BGP process to process conditional route advertisement every 20 seconds.
- R2 should not batch routing updates to R10 and advertise them immediately.
- Configure R2 so that session de-activation happens within 15 seconds of no session activity.

Configuration

The BGP routing process is complicated and uses many data structures and periodically scheduled tasks. One of the most important BGP processes is "BGP scanner," which performs the following important functions:

1. Runs through all prefixes in BGP table and checks the validity and reachability of the BGP NEXT_HOP attribute.
2. Performs conditional advertisement and route injection.
3. Imports new routes into the BGP table from RIB (via network and redistribute commands).
4. Performs route dampening.

Later you will see that some of this periodic behavior became event driven in recent

https://t.me/learningnets

IOS releases (BGP next-hop trigger feature). However, for now what's important is that the BGP scanner runs every 60 seconds by default. You can change this interval by using the command `bgp scan-time <5-60>` . The shorter the interval, the better the routing convergence, but the more load is put on the router's CPU. You may check the BGP scanner runs by using the command `debug ip bgp events` .

Another important BGP process is "BGP I/O," which processes BGP UPDATE and KEEPALIVE messages. By default, BGP batches all new prefixes and delays the sending of an update packet to the peer until the next advertisement-interval timer expires. This interval is configured on a per-peer basis using the command `neighbor <IP> advertisement-interval <seconds>` . Decreasing this interval (the minimum value is zero) improves BGP convergence but generates more BGP traffic and puts more stress on the router's CPU.

The last important timer is the BGP session keepalive interval. Keepalives are important for validating BGP session health, to avoid routing black holes. BGP peers advertise the hold time interval when establishing the BGP peering session and send KEEPALIVE message to inform each other of their availability. Peers may advertise different hold-time intervals—it is only important that the peer receive the KEEPALIVE message until its hold-time interval expires. You can change the keepalive and hold-time periods on a per-process basis by using the command `timers bgp <keepalive> <holdtime>` . The default values are 60 and 180 for keepalive and holdtime intervals, respectively. Setting the keepalive interval too small results in faster peering deactivation detection, but it may lead to "false positives" and disruption of the BGP session by mistake and excessive route flapping. Remember that you must completely reset the BGP session for the new keepalive timers to take effect. If you want to observe the BGP keepalive exchange process, use the command `debug ip bgp keepalive` .

Read this article for a detailed explanation on BGP convergence: [Understanding BGP Convergence]

Configuration

```
R2:

router bgp 200
 timers bgp 5 15
 neighbor 192.10.1.254 advertisement-interval 0
 bgp scan-time 20
```

https://t.me/learningnets

Verification

Check the timer values using the show commands demonstrated below. Note that you must clear the BGP session between R2 and R10 for the new bgp timer values to be negotiated.

```
R2#show ip bgp summary | include scan
BGP activity 18/0 prefixes, 46/15 paths, scan interval 20 secs
!   R2#show ip bgp neighbors 192.10.1.254 | inc advertisement|keepal
   Last read 00:00:08, last write 00:00:48, hold time is 180, keepalive interval is 60 seconds
 Configured hold time is 15, keepalive interval is 5 seconds
   Default minimum time between advertisement runs is 30 seconds
 Minimum time between advertisement runs is 0 seconds
!R2#clear ip bgp 192.10.1.254
R2#show ip bgp neighbors 192.10.1.254 | inc advertisement|keepal
  Last read 00:00:00, last write 00:00:03, hold time is 15, keepalive interval is 5 seconds
 Configured hold time is 15, keepalive interval is 5 seconds
  Default minimum time between advertisement runs is 30 seconds
 Minimum time between advertisement runs is 0 seconds
```

Notice that after clearing the session, the 'configured timers' match the active timers on the session. Read this article for a detailed explanation on BGP convergence: Understanding BGP Convergence

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Fast Fallover

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Disable the BGP feature in R3 that allows for eBGP peering session deactivation when a physical interface goes down.
- Configure all of R3's BGP peering sessions for fast peering deactivation.

Configuration

It is common to have eBGP peers directly connected across a physical interface. With point-to-point links, this allows for fast external session deactivation based on the interface protocol state. That is, as soon as the interface connecting to an external peer signals protocols down, the BGP process may deactivate the peering session without waiting for the hold-down time to expire. This feature is enabled by default for eBGP sessions using the BGP process command `bgp fast-external-fallover` . Notice that this feature is only efficient when the peering session is established across the non-shared link; using it on NBMA and Ethernet interfaces might be inefficient.

A more advanced version of this feature is available in the recent IOS release. The new feature is called "fast peering session deactivation support" and could be configured on a per-neighbor basis using the command `neighbor <IP> fall-over` . This feature applies to both iBGP and eBGP (mostly multi-hop) sessions and does not depend on interface state. The IGP route used to reach the peer configured for fast session deactivation is monitored by the BGP process. When the IGP route disappears, the BGP session gets immediately torn down unless there is a backup IGP route to reach the peer. Thus, the new feature is event driven and allows for

https://t.me/learningnets

fast detection of peering issues even for iBGP (non-direct) sessions. Notice that this feature has the same limitation as the fast-external-fallover; if the peer is connected via a shared interface, the router may not detect the loss of the directly connected IGP network. In situations like this, you may use point-to-point NBMA subinterfaces or reliable static /32 routes on the shared NBMA or Ethernet interfaces. BFD (Bidirectional Forwarding Detection) can be tied to the **`neighbor <IP> fall-over`** to overcome connections over shared segments.

Remember that convergence improvements result in less stable topology. To minimize the impact of IGP instabilities on BGP tables, use event dampening technologies (such as ip dampening) and prefix summarization.

R3:

```
router bgp 200
 no bgp fast-external-fallover
 neighbor 155.1.0.5 fall-over
 neighbor 155.1.13.1 fall-over
 neighbor 155.1.23.2 fall-over
 neighbor 155.1.37.7 fall-over
 neighbor 155.1.58.8 fall-over
```

Verification

To test the fast fall-over feature, configure R7 and R8 to stop advertising the aggregate for 155.1.0.0/16. This is needed because R4 and R6 generate a summary prefix 155.1.0.0/16, which is injected into RIB and could be used as a backup route to reach any BGP next-hop in AS 200. Another way to accomplish this would be to use the **`bgp nexthop route-map <route-map>`** command on R3. This route-map can match a prefix-list that specifies a prefix length. For example, we could use '0.0.0.0/0 ge 24' on the prefix-list and this would make R3 only consider next-hops that are at least /24's. If the /24 IGP route is lost but the /16 aggregate is still in IGP, R3 would not use the aggregate because it does not meet the required length specified in the prefix-list.

https://t.me/learningnets

```
R7:

router bgp 300
 no aggregate-address 155.1.0.0 255.255.0.0

R8:

router bgp 200
 no aggregate-address 155.1.0.0 255.255.0.0
```

Verify that R3's iBGP peering session is enabled for fast fall-over. Then activate a BGP debugging command for RIB watching and shut down the link connecting R5 and R8. Recall that because these routers are not connected via a true point-to-point link, we must shut down both sides of the link to simulate a link down even. If we only shut down the link on R8, R5 will still advertise GigabitEthernet1.58 into IGP because its side will not go down.

```
R3#show ip bgp neighbors 155.1.58.8 | include [Ff]all
 Fall over configured for session
!R3#debug ip bgp rib-filter

BGP Rib filter debugging is on
!   R5, R8
interface GigabitEthernet1.58
 shutdown

 BGP RIB_RWATCH: (default:ipv4:base) T 155.1.58.0/24 EVENT RIB update DOWN
BGP RIB_RWATCH: (default:ipv4:base) N 155.1.58.0/24 QP Schedule query
BGP RIB_RWATCH: (default:ipv4:base) T 155.1.58.0/24 EVENT Query did not find route
BGP RIB_RWATCH: (default:ipv4:base) R 155.1.58.0/24  d=90 p=1 -> Tu0 155.1.0.5 base 26880256 Deleting
BGP RIB_RWATCH: Adding to client notification queue
BGP RIB_RWATCH: Adding to client notification queue
BGP RIB_RWATCH: (default:ipv4:base) W 155.1.58.8/32 c=0x7F8368ED1D08 C
R3#lient notified unreachable
 BGP RIB_RWATCH: (default:ipv4:base) W 155.1.58.8/32 c=0x7F8360DCC0B0 Client notified unreachable
%BGP-5-NBR_RESET: Neighbor 155.1.58.8 reset (Route to peer lost)
%BGP-5-ADJCHANGE: neighbor 155.1.58.8 Down Route to peer lost
%BGP_SESSION-5-ADJCHANGE: neighbor 155.1.58.8 IPv4 Unicast topology base removed from session  Route to peer lost
BGP RIB_RWATCH: (default:ipv4:base) T 28.119.16.0/24 EVENT RIB update MODIFY
BGP RIB_RWATCH: (default:ipv4:base) T 28.119.17.0/24 EVENT RIB
BGP RIB_RWATCH: (default:ipv4:base) T 112.0.0.0/8 EVENT RIB update MODIFY
BGP RIB_RWATCH: (default:ipv4:base) T 113.0.0.0/8 EVENT RIB update MODIFY
BGP RIB_RWATCH: (default:ipv4:base) T 114.0.0.0/8 EVENT RIB update MODIFY
BGP RIB_RWATCH: (default:ipv4:base) T 115.0.0.0/8 EVENT RIB update MODIFY
```

https://t.me/learningnets

```
BGP RIB_RWATCH: (default:ipv4:base) T 116.0.0.0/8 EVENT RIB update MODIFY
BGP RIB_RWATCH: (default:ipv4:base) T 117.0.0.0/8 EVENT RIB update MODIFY
BGP RIB_RWATCH: (default:ipv4:base) T 118.0.0.
BGP RIB_RWATCH: (default:ipv4:base) T 119.0.0.0/8 EVENT RIB update MODIFY
BGP RIB_RWATCH: (default:ipv4:base) T 155.1.58.0/24 EVENT RIB update UP

BGP RIB_RWATCH: (default:ipv4:base) N 155.1.58.0/24 Adding route
BGP RIB_RWATCH: (default:ipv4:base) R 155.1.58.0/24  d=0 p=0 -> Updating
BGP RIB_RWATCH: (default:ipv4:base) N 155.1.58.8/32 Adding internal track
```

Notice that the peering with R8 is torn down as soon as the route is lost. R5, however, did not tear down its session with R8! This is because R5 does not have fall-over configured toward R8, and fast-external fall-over mechanism only happens for eBGP sessions by default. R5 loses its EIGRP adjacency with R8, but it still has 155.1.58.0/24 in its BGP table. This is why toward the end of the debug from R3 you can observe the "EVENT RIB UP" and "Adding route". R3 installs the BGP route, but because there is no connectivity toward R8 due to the link being shut down, the session will never get established.

```
R5#show ip route 155.1.58.8

Routing entry for 155.1.58.0/24
  Known via "bgp 200", distance 200, metric 0, type internal
  Last update from 155.1.58.8 00:00:06 ago
  Routing Descriptor Blocks:
  * 155.1.58.8, from 155.1.58.8, 00:00:06 ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: none
```

R5 will tear down its session with R8 based on default BGP timers, 180 seconds. If R5 had 'fall-over' configured toward R8, R5's peering would have been terminated as soon as the link was shut down.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Outbound Route Filtering

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- R7 and R5 should filter out the prefixes 112.0.0.0/8 and 114.0.0.0/8 from being advertised to R6 and R4, respectively.
- The filtering configuration should be applied to routers R4 and R6.

Configuration

ORF, or outbound route filtering, is the technique that allows a BGP peer to “push” a filter to the remote neighbor. The neighbor then applies the prefix filter to the outbound updates sent to the peer that pushed the filter. This feature is particularly helpful in situations where BGP peers exchange large amounts of BGP information. Applying filtering outbound on the remote peer instead of inbound on the local peer significantly decreases the amount of routing information sent across the link. There are two types of ORF filters defined in IETF’s draft: prefix-list based and community based. Cisco IOS supports only the prefix-list ORFs.

In BGP terms, ORF is a special capability negotiated during the establishment of a BGP session. A peer may advertise its willingness to send, receive, or both send and receive the ORFs. You must enable this capability on peering routers before configuring ORFs. The command to enable the feature in the IOS routers is `neighbor <IP> capability orf prefix-list [send|receive|both]` . You must reset the BGP session to negotiate the new capabilities.

To configure and push an ORF, you must define a prefix list and apply it to the peer’s session using the command `neighbor <IP> prefix-list <NAME> in` . The list must

https://t.me/learningnets

be inbound, because this is the natural direction for ORF. If the session has ORF send capability enabled, the list will be pushed to the remote peer and installed as an outbound filter after you do a session refresh using the **`clear ip bgp <IP> soft in prefix-filter`** command. This command pushes the prefix list and requires route refresh (re-advertisement) from the peer.

```
R7:
router bgp 300
 neighbor 155.1.67.6 capability orf prefix-list both

R5:
router bgp 200
 neighbor 155.1.45.4 capability orf prefix-list both

R6:
ip prefix-list ORF deny 112.0.0.0/8
ip prefix-list ORF deny 114.0.0.0/8
ip prefix-list ORF permit 0.0.0.0/0 le 32
!
router bgp 100
 neighbor 155.1.67.7 capability orf prefix-list both
 neighbor 155.1.67.7 prefix-list ORF in

R4:

ip prefix-list ORF deny 112.0.0.0/8
ip prefix-list ORF deny 114.0.0.0/8
ip prefix-list ORF permit 0.0.0.0/0 le 32
!
router bgp 100
 neighbor 155.1.45.5 capability orf prefix-list both
 neighbor 155.1.45.5 prefix-list ORF in
```

Verification

To verify ORFs, issue the following “show” command on any of the routers “pushing” the filters, such as on R6. Notice that the new capability has been negotiated and the list sent.

```
R6#show ip bgp neighbors 155.1.67.7

BGP neighbor is 155.1.67.7,  remote AS 300, external link
<snip> For address family: IPv4 Unicast
```

https://t.me/learningnets

```
  Session: 155.1.67.7
  BGP table version 141, neighbor version 141/0
  Output queue size : 0
  Index 4, Advertise bit 0
  4 update-group member
  AF-dependant capabilities:
Outbound Route Filter (ORF) type (128) Prefix-list:
      Send-mode: advertised, received
      Receive-mode: advertised, received  Outbound Route Filter (ORF): sent;

  Incoming update prefix filter list is ORF
  Slow-peer detection is disabled
  Slow-peer split-update-group dynamic is disabled
  Interface associated: GigabitEthernet1.67
                                 Sent       Rcvd
  Prefix activity:               ----       ----
    Prefixes Current:               8         15 (Consumes 1800 bytes)
    Prefixes Total:                18         15
    Implicit Withdraw:              0          0
    Explicit Withdraw:             10          0
    Used as bestpath:             n/a         10
    Used as multipath:            n/a          0

                                   Outbound    Inbound
  Local Policy Denied Prefixes:    --------    -------
    Bestpath from this peer:             10        n/a
    Total:                               10          0
```

Get to the other side of the connection and check the prefix list received by R7. Notice the name for the list, constructed from the peer's IP address.

```
R7#show ip bgp neighbors 155.1.67.6 received prefix-filter

Address family: IPv4 Unicast
ip prefix-list 155.1.67.6: 3 entries
   seq 5 deny 112.0.0.0/8
   seq 10 deny 114.0.0.0/8
   seq 15 permit 0.0.0.0/0 le 32
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Soft Reconfiguration

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Configure R2 to accept all prefixes from R 10 irrespective of the configured inbound filters, and store them all locally.

Configuration

Until RFC2918 introduced the Route Refresh capability to BGP, it was impossible to signal a remote BGP peer to re-advertise the prefixes (Adj-RIB-Out) to the local peer. This feature is very helpful in situations where the local peer changes inbound filtering policy and needs the peer to re-advertise the routing information. The only way to accomplish the policy refresh was to tear down and re-establish the peering session, which could be very resource consuming and cause connectivity disruption. Before the Route Refresh capability became standardized, one workaround was to use the so-called *soft-reconfiguration* approach.

https://t.me/learningnets

When a local BGP speaker is configured to apply soft-reconfiguration to a peer using the command **`neighbor <IP> soft-reconfiguration inbound`** the speaker will accept ALL prefixes from the remote peer and store them in a separate memory buffer (of course, a session reset is required for this operation to initialize). The prefixes are then processed via the inbound filters and the resulting information imported into Adj-RIB-In and finally to the BGP table. Every time the local policy changes, there is no need to re-establish the peering session but simply apply the filters to the stored information. The penalty is the extra memory needed to store the routing information from the peer. Of course, this feature became deprecated with the introduction of Route Refresh capability.

```
R2:

router bgp 200
 neighbor 192.10.1.254 soft-reconfiguration inbound
```

Verification

Check the routes received and stored from R10.

```
R2#show ip bgp neighbors 192.10.1.254 received-routes

BGP table version is 101, local router ID is 150.1.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  51.51.51.51/32   192.10.1.254             0             0 254 ?
 r>  192.10.1.0       192.10.1.254             0             0 254 ?
 *>  205.90.31.0      192.10.1.254             0             0 254 ?
 *>  220.20.3.0       192.10.1.254             0             0 254 ?
 *>  222.22.2.0       192.10.1.254             0             0 254 ?

Total number of prefixes 5
```

Apply a prefix filter to the peering session with R10, filtering all possible prefixes. Apply soft reset to the peering session and check the Adj-RIB-In (routes received from R10 after filtering) with the total number of routes received from R10.

https://t.me/learningnets

```
R2(config)#ip prefix-list TEST deny 0.0.0.0/0 le 32
R2(config)#router bgp 200
R2(config-router)#neighbor 192.10.1.254 prefix-list TEST in

!R2#clear ip bgp 192.10.1.254 soft in
 R2#show ip bgp neighbors 192.10.1.254 routes



Total number of prefixes 0
!R2#show ip bgp neighbors 192.10.1.254 received-routes

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   51.51.51.51/32   192.10.1.254             0             0 254 ?
 *   192.10.1.0       192.10.1.254             0             0 254 ?
 *   205.90.31.0      192.10.1.254             0             0 254 ?
 *   220.20.3.0       192.10.1.254             0             0 254 ?
 *   222.22.2.0       192.10.1.254             0             0 254 ?

Total number of prefixes 5
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP Next-Hop Trigger

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Configure R3 to respond to BGP prefixes' next-hop changes within 30 seconds of IGP prefix change.
- For this scope, configure and advertise into BGP a new Loopback1 prefix on R8 with IPv4 address of 8.8.8.8/24.

Configuration

The BGP NEXT_HOP attribute is often used for recursive lookup through the IGP table to resolve the actual next-hop interface and router. Until IOS 12.3(14)T, BGP was accounting for IGP information changes only during periodic BGP scans with the interval defined by the command `bgp scan-time <seconds>` . With the default value of 60 seconds, it was impossible to react to IGP topology changes between the runs of the BGP scanner process.

Since IOS 12.3(14)T, the next-hop tracking behavior has changed from periodic to event driven. This behavior is enabled by default using the command `bgp nexthop trigger enable` . BGP process registers the NEXT_HOP attribute values with the RIB table watch process. As soon as any change that affects an existing NEXT_HOP occurs, the watch process notifies the BGP router process. If the change results in prefix withdrawn, the BGP process immediately removes the prefix. All other notification are delayed and batched until the time-interval specified by the command `bgp nexthop trigger delay <seconds>` expires. After this, a full BGP table walk occurs, performing best-path computations for all prefixes. The delay value

https://t.me/learningnets

should be tuned according to the IGP convergence speed to avoid unnecessary full table walks. That is, it is desirable to have IGP fully converged after an initial change (or sequence of change) until the full BGP walk has started.

```
R3:

router bgp 200
 bgp nexthop trigger delay 30
```

Verification

Advertise a new network 8.8.8.0/24 into BGP on R8. Then remove the NHRP mapping on R3 for R5, and clear the nhrp process on R3. This will lead to R3 and R5 losing EIGRP adjacency. Before this, enable the following debugging on R3.

`debug ip bgp event nexthop` :

```
R8:
interface Loopback1
 ip address 8.8.8.8 255.255.255.0
!
router bgp 200
 network 8.8.8.0 mask 255.255.255.0

R3:

interface Tunnel 0
 no ip nhrp map 155.1.0.5 169.254.100.5
```

Observe the debugging output on R3. Notice that the BGP process responds to IGP changes and schedules a BGP table walk in 30 seconds.

https://t.me/learningnets

```
R3#clear ip nhrp
R3#clear crypto sa

!%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.5 (Tunnel0) is down
: holding time expired
EvD: accum. penalty decayed to 0 after 232 second(s)
BGP(IPv4 Unicast): Nexthop modified, reuse in 00:00:19, 19000 , scheduling nexthop scan in 30 secs

EvD: accum. penalty decayed to 500 after 0 second(s)
BGP(IPv4 Unicast): Nexthop modified, reuse in 00:00:27, 27000 , timer already running
BGP: NHOP scanner event timer

BGP: Nexthop walk for IPv4 Unicast
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP TTL Security

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Configure R3 to accept TCP packets from eBGP peers only if they are no more than one hop away.

Configuration

The Generalized TTL Security Mechanism (GTSM) defined in RFC 3682 specifies a protection method against BGP session hijacking and resource exhaustion attacks. Generally, the BGP process listens on the TCP port 179 and accepts all TCP SYN packets destined to this port, unless they are filtered by an ACL. It is possible to generate a barrage of spoofed packets imitating a valid BGP session and inject false information (if the session is unauthenticated) or generate a TCP SYN-flooding attack.

GTSM utilizes the simple fact that every router on the path to the BGP speaker decrements the TTL field in IP packets by one. Based on this, it is possible to identify potentially spoofed packets by looking at their TTL field; the packets sent from “afar” will have the TTL field below some threshold. It is possible to define a “secure radius” in the number of hop counts to accept the incoming IP packets. For example, if all BGP peers are within 10 hops from the local BGP speaker, all incoming IP packets will have their TTL field set to no less than 245. This is because all IP packets start with TTL=255 and the field is decremented by every hop on the path. Thus, by accepting the IP packets with TTL greater than or equal to 245, it is possible to minimize the risk of spoofed packets reaching the BGP process. Notice that the usefulness of GTSM feature decreases as the diameter of eBGP Multihop

https://t.me/learningnets

session grows.

To configure the TTL security checks for a BGP peer, use the command **`neighbor <IP> ttl-security hops <hop-count>`**. This command applies to eBGP peering sessions only (either directly-connected or multihop) and specifies the number of hops the remote peer could be away from the local speaker. Remember that the internal BGP sessions are not protected, and therefore the internal network is assumed to be "trusted." All incoming TCP packets targeted at the BGP port with an IP TTL value below (255 - <hop-count>) are silently discarded by the router. In addition, the feature sets the TTL value for outgoing TCP/IP packets to 255 to make sure the remote peer will accept the local packets. The GTSM feature is mutually exclusive with the `ebgp-multihop BGP` feature. This is because the eBGP session by default sets TTL=1 in the outgoing IP packets and with the `multihop <n>` session parameter, the TTL value is set to <n>, which is not compatible with GTSM. Therefore, make sure you configure the GTSM feature on both sides of the peering link.

```
R1:
router bgp 100
 neighbor 155.1.13.3 ttl-security hops 1

R3:
router bgp 200
 neighbor 155.1.13.1 ttl-security hops 1
 neighbor 155.1.37.7 ttl-security hops 1

R7:

router bgp 300
 neighbor 155.1.37.3 ttl-security hops 1
```

Verification

Check the GTSM settings for eBGP peers on R3. Repeat the same verifications on R1 and R7:

```
R3#show ip bgp neighbors 155.1.13.1 | inc hop
  External BGP neighbor may be up to 1 hop away.
!   R3#show ip bgp neighbors 155.1.37.7 | inc hop
  External BGP neighbor may be up to 1 hop away.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - BGP

BGP AllowAS in

> You must load the initial configuration files for the section, **Basic BGP Routing**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs BGP Diagram** to complete this task.

Task

- Configure R2 and R8 to advertise networks 2.2.2.0/24 and 8.8.8.0/24 into BGP.
  - Configure new Loopback1 interfaces for this scope.
- Configure AS 200 border routers so that if AS 200 is partitioned, the remaining segments could transit AS 100 to recover connectivity.

Configuration

The BGP loop-prevention mechanism does not allow a BGP speaker to accept prefixes with the local AS number in the AS_PATH list. However, in some cases, it would be desirable to accept the routes originated in the same AS via another AS. There are two common scenarios:

1. The company's network is partitioned, and every partition connects to the Internet or ISP. Every network has its own set of prefixes but uses the same AS number. In this case, for the partitions to exchange prefixes, they must accept the NLRIs with the same AS number.
2. The company connects to an ISP and wants to use it as a transit path in case the company's network becomes segmented due to an emergency. In this case, the prefixes advertised to the ISP must be accepted back by the border peers.

Cisco IOS allows for accepting the prefixes with the local AS number from a specific peer using the command `neighbor <IP> allowas-in [<count>]` . Here, `<count>` is the number of the local AS number occurrences in the AS_PATH attribute, which

https://t.me/learningnets

defaults to one. This parameter serves a purpose similar to the hop-count limit in distance-vector protocol and implements the well-known count-to-infinity loop prevention technique.

To prevent routing loops with this feature, you should be careful when implementing prefix aggregation. Specifically, only one “partition” or border peer can implement summarization, or summarization should not be used at all. Otherwise, the upstream ASs will have trouble selecting the proper entry point to the AS partitions. Needless to mention, using the AllowAS in feature is highly NOT recommended and only advised as a last resort. In a Layer 3 MPLS-VPNs environment, this feature can be used safely by sites that connect into the same service provider and don't have a backdoor link. The sites can all share the same AS and configure **`neighbor <IP> allowas-in [<count>]`** toward their provider edge router. Alternatively, the service provider can use the **`neighbor <IP> as-override`** command if the customer sites of the Layer 3 MPLS VPN have routers that don't support allowas-in.

```
R2:
interface Loopback1
 ip address 2.2.2.2 255.255.255.0
!
router bgp 200
 network 2.2.2.0 mask 255.255.255.0

R3:
router bgp 200
 neighbor  155.1.13.1 allowas-in

R5:
router bgp 200
 neighbor 155.1.45.4 allowas-in

R8:

interface Loopback1
 ip address 8.8.8.8 255.255.255.0
!
router bgp 200
 network 8.8.8.0 mask 255.255.255.0
```

Verification

https://t.me/learningnets

Configure the routers so that AS 200 is split into two parts. To accomplish this, configure the routers as follows:

```
R3:
router eigrp 100
 passive-interface GigabitEthernet1.37
 passive-interface GigabitEthernet1.13

R5:

router eigrp 100
 passive-interface Tunnel0
 passive-interface GigabitEthernet1.45
!
interface Tunnel0
 shutdown
```

Check the BGP tables of R3 and R5 for the prefixes originated in AS 200 (note that you must give the BGP network time to converge). Notice that both R3 and R5 accept those prefixes because of the AllowAS in feature. Then trace the route from R2 to R8 between the two configured subnets and make sure connectivity is maintained.

```
R3#show ip bgp regexp _200$
BGP table version is 543, local router ID is 150.1.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  8.8.8.0/24       155.1.13.1                             0 100 200 i
!R5#show ip bgp regexp _200$
BGP table version is 22, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  2.2.2.0/24       155.1.45.4                             0 100 200 i
!R2#traceroute 8.8.8.8 source loopback1
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 8.8.8.8
VRF info: (vrf in name/id, vrf out name/id)

  1 155.1.23.3 13 msec 2 msec 2 msec
  2 155.1.13.1 9 msec 4 msec 20 msec
  3 155.1.146.4 23 msec 9 msec 6 msec
  4 155.1.45.5 11 msec 10 msec 10 msec 5 155.1.58.8 38 msec *  47 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM Dense Mode

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM dense-mode multicast delivery on the Ethernet path between R6 and R10.
  - Do not enable PIM on the DMVPN cloud.
- To test this configuration, configure R10's GigabitEthernet1.10 interface to join the multicast group **224.10.10.10**.
  - Make sure that R6 can ping this multicast group.

Configuration

Multicast traffic distribution uses the concept of "constrained" flooding. This means that packets destined to a particular multicast group are flooded only on the links that lead to the actual group subscribers. The constrained flooding procedure consists of two parts:

- Building a multicast distribution tree for a group—aka Shortest Path Tree or SPT.
- Flooding the actual multicast packets down the SPT, while performing Reverse Path Forwarding (RPF) to avoid loops.

We are going to work mostly with Protocol Independent Multicast (PIM) Dense Mode and Sparse Mode. PIM is a special signaling protocol that does not distribute any routing information on its own. Rather, PIM uses the unicast routing table to perform RPF checks. This procedure is the core of multicast routing. When the router receives a multicast packet, it looks at the source IP address for the packet.

https://t.me/learningnets

The router looks up the source IP address in the unicast routing table and determines the outgoing interface for this packet. If this outgoing interface does not match the interface where the packet was received, the router drops the packet. This behavior intends to eliminate potential packet loops, which may occur when routers flood multicast packets. Only the immediate upstream router (with respect to the source of the multicast stream) is allowed to send us multicast packets.

If the packet passes the RPF check, the router will determine the outgoing interface list (OIL) for this packet, that is, the branches of the multicast distribution tree. The OIL never includes the input interface; this is the well-known split horizon rule. The SPT should be signaled by PIM, based on the active subscribers across the network. However, PIM Dense Mode (PIM DM) does not use any explicit signaling to join a multicast distribution tree. Instead, it uses the inverse logic approach. All routers initially flood packets out of all their multicast-enabled interfaces. If the downstream router determines that it does not have any directly connected subscribers or other routers willing to receive multicast traffic, it sends a special PIM Prune message to the upstream router. The upstream router then excludes the particular interface from the OIL for the pruned group.

PIM DM works fine in small networks with a lot of subscribers. However, it has one inherent scalability limitation: excessive flooding. Every Pruned interface state expires in 3 minutes by default, and then flooding out of this interface resumes again, until the upstream does not receive another PIM Prune message on the interface. This is needed to ensure that no node in the network potentially misses multicast traffic. This periodic flood and prune behavior is what makes PIM Dense mode non-scalable.

The obvious benefit of PIM DM is its “plug-and-play” behavior. As soon as you configure PIM DM in your network, you can start sending multicast traffic, and the traffic is flooded to all interested subscribers.

```
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim dense-mode
!
interface GigabitEthernet1.45
 ip pim dense-mode
R6:
interface GigabitEthernet1.146
 ip pim dense-mode
R5:
ip multicast-routing distributed
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.45
 ip pim dense-mode
!
interface GigabitEthernet1.58
 ip pim dense-mode
R8:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim dense-mode
!
interface GigabitEthernet1.108
 ip pim dense-mode
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim dense-mode
!
interface GigabitEthernet1.10
 ip igmp join-group 224.10.10.10
 ip pim dense-mode
```

Verification

Start your basic verification by looking for PIM neighbors and PIM interfaces. Notice the PIM version and mode on every interface, and make sure they match the requirements in the tasks.

```
R5#show ip pim neighbor

PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode
155.1.45.4        GigabitEthernet1.45      00:02:15/00:01:27 v2    1 / S P G
155.1.58.8        GigabitEthernet1.58      00:02:05/00:01:37 v2    1 / DR S P G
!
!R5#show ip pim interface



Address          Interface                Ver/   Nbr    Query  DR     DR
```

https://t.me/learningnets

```
                                        Mode   Count  Intvl  Prior
155.1.45.5       GigabitEthernet1.45      v2/D   1      30     1      155.1.45.5
155.1.58.5       GigabitEthernet1.58      v2/D   1      30     1      155.1.58.8
```

Ping the group and confirm that the packets reach the destination. Note from configuration that on the client side, which is R6, multicast routing does not need to be enabled.

```
R6#ping 224.10.10.10 repeat 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:
...Reply to request 3 from 155.1.108.10, 11 ms
Reply to request 4 from 155.1.10.10, 70 ms
Reply to request 5 from 155.1.10.10, 87 ms

Reply to request 6 from 155.1.10.10, 160 ms
Reply to request 7 from 155.1.10.10, 288 ms
Reply to request 8 from 155.1.10.10, 112 ms
```

Now verify multicast route states created by the traffic flow. You should see (S,G) entries corresponding to the dense-mode SPT. Notice that (*,G) states have all potential interfaces within their OILs. Look for the RPF neighbor for each entry, and make sure they match the traffic flow. The RPF neighbor of 0.0.0.0 means that this router is connected to the source, and the RP of 0.0.0.0 confirms that no RP is used for this group, so we are using PIM dense mode.

```
R4#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
```

https://t.me/learningnets

```
(*, 224.10.10.10), 00:01:07/stopped, RP 0.0.0.0, flags: D

  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Dense, 00:01:07/stopped
    GigabitEthernet1.146, Forward/Dense, 00:01:07/stopped
(155.1.146.6, 224.10.10.10
), 00:01:07/00:01:52, flags: T
  Incoming interface: GigabitEthernet1.146, RPF nbr 155.1.146.6
  Outgoing interface list: GigabitEthernet1.45, Forward/Dense
, 00:01:07/stopped
!
!R5#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:02:28/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Dense, 00:02:28/stopped
    GigabitEthernet1.45, Forward/Dense, 00:02:28/stopped
(155.1.146.6, 224.10.10.10
), 00:02:28/00:00:31, flags: T
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list: GigabitEthernet1.58, Forward/Dense
, 00:02:28/stopped
!
!R8#show ip mroute 224.10.10.10

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
```

https://t.me/learningnets

```
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:03:32/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Dense, 00:03:32/stopped
    GigabitEthernet1.58, Forward/Dense, 00:03:32/stopped
(155.1.146.6, 224.10.10.10
), 00:03:32/00:02:27, flags: T
  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list: GigabitEthernet1.108, Forward/Dense
, 00:03:32/stopped
!
!R10#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:05:02/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.10, Forward/Dense, 00:05:02/stopped
    GigabitEthernet1.108, Forward/Dense, 00:05:02/stopped
```

https://t.me/learningnets

```
(155.1.146.6, 224.10.10.10
), 00:04:43/00:02:08, flags: LT
  Incoming interface: GigabitEthernet1.108, RPF nbr 155.1.108.8
  Outgoing interface list: GigabitEthernet1.10, Forward/Dense
, 00:04:43/stopped
```

Another way to check the RPF neighbor and RPF interface is by using the following command.

```
R5#show ip rpf 155.1.146.6
RPF information for ? (155.1.146.6) RPF interface: GigabitEthernet1.45
  RPF neighbor: ? (155.1.45.4)
  RPF route/mask: 155.1.146.0/24 RPF type: unicast (eigrp 100)

  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

As stated previously, the unicast routing table is used to look up the RPF interface for the source of the packet. In this case, the source was 155.1.146.6. This command looks at the RIB and sees that there is a route for 155.1.146.0/24 via EIGRP. The interface that is used for this route is GigabitEthernet1.45. Because this is where the multicast traffic is received on R5, the RPF check passes.

```
R5#show ip route 155.1.146.6
Routing entry for 155.1.146.0/24
 Known via "eigrp 100", distance 90, metric 3072, type internal
  Redistributing via eigrp 100, ospf 1
  Advertised by ospf 1 subnets
  Last update from 155.1.45.4 on GigabitEthernet1.45, 00:58:47 ago
  Routing Descriptor Blocks: * 155.1.45.4, from 155.1.45.4, 00:58:47 ago, via GigabitEthernet1.45

      Route metric is 3072, traffic share count is 1
      Total delay is 20 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
  Loading 1/255, Hops 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Multicast RPF Failure

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM dense-mode multicast delivery on the Ethernet path between R6 and R10.
  - Don't enable PIM on the GigabitEthernet1.45 link between R4 and R5 but instead enable it on the Tunnel10 connection between R1 and R5.
- To test this configuration, configure R10's GigabitEthernet1.10 interface to join the multicast group **224.10.10.10**.
  - Make sure that R6 can ping this multicast group.

Configuration

The RPF check procedure was discussed in the previous task. RPF failures occur in two general situations: packets being flooded out of the wrong interface (good, prevents looping) or unicast shortest paths not matching multicast distribution trees (bad, PIM-enabled interface does not follow the unicast routing table). Situations of the second type generally occur in the following cases:

- Duplicate paths to the multicast source exist, and PIM is not enabled on all links. In this situation, the router may receive multicast packets on an interface that is not on the shortest path to the source. As an extreme solution to this situation, PIM might be enabled on the links where IGP is not running.
- There are multiple routing protocols in the network, and the router might have paths to the source learned via different protocols. In this situation, the router may receive

https://t.me/learningnets

multicast packets on an interface running RIP, while thinking the shortest path to the source is via OSPF.

Warning! Use of static routes may the change a local router's perception of the shortest paths and force it to reject otherwise valid multicast packets.

Additionally, temporary RPF failures may happen in situations where the network is unstable and the routers keep changing their shortest path tables. The best way to find and isolate the RPF issue in your lab is probably to run the `debug ip mfib pak` command on all routers, starting from the one closest to the destination and moving upstream to the source. This command will show you process-switched multicast packets and signal any RPF issues. Note that you may need to enter the commands `no ip mfib cef input` and `no ip mfib cef output` on all interfaces where multicast packets are received; this will disable cef switching of multicast packets on the interfaces where it is specified.

The easiest way to deal with an RPF failure is to manually provide RPF information using the `ip mroute` command. This command is local to the router and specifies the RPF next-hop or interface for the given subnet. The syntax for the command is `ip mroute <SOURCE> <MASK> [RPF-IP-Address|<Interface-Name>] [distance]` . It is important to understand that this command does not specify any unicast forwarding rules. Instead, it creates an ordered table of entries to look up for RPF information. When a multicast packet is received, the router will first try to find a matching entry in the mroute table, to determine the RPF interface. Note that the mroute table is ordered in the same way in which you enter the ip mroute commands, so you should enter the most specific mroutes first. When the router finds RPF information in both the mroute table and the unicast routing table, it prefers mroute information, because it has a better distance (zero). However, connected routes still have preference over any other RPF source. You may change the mroute administrative distance to make it less preferred than the unicast routing table information.

```
R1:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim dense-mode
!
interface Tunnel 10
 ip pim dense-mode
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim dense-mode
```

https://t.me/learningnets

```
R6:

interface GigabitEthernet1.146
 ip pim dense-mode
R5:
ip multicast-routing distributed
!
interface Tunnel 10
 ip pim dense-mode
!
interface GigabitEthernet1.58
 ip pim dense-mode
!
ip mroute 0.0.0.0 0.0.0.0 Tunnel10
R8:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim dense-mode
!
interface GigabitEthernet1.108
 ip pim dense-mode
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim dense-mode
!
interface GigabitEthernet1.10
 ip igmp join-group 224.10.10.10
 ip pim dense-mode
```

Verification

Try pinging the group before you apply the mroute needed to correct the RPF lookup error.

https://t.me/learningnets

```
R6#ping 224.10.10.10 repeat 10000

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:..........
```

Check the multicast routing table on R5. Notice that the (S,G) state for our flow has no Incoming Interface field (it's Null); this means that the traffic is not arriving on the RPF interface.

```
R5#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.10.10.10), 00:00:34/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Dense, 00:00:34/stopped
    Tunnel10, Forward/Dense, 00:00:34/stopped
 (155.1.146.6, 224.10.10.10
), 00:00:34/00:02:25, flags: Incoming interface: Null, RPF nbr 0.0.0.0

  Outgoing interface list:
    Tunnel10, Forward/Dense, 00:00:34/stopped
    GigabitEthernet1.58, Forward/Dense, 00:00:34/stopped
```

You can easily see RPF failures when you enable multicast packet debugging.

https://t.me/learningnets

```
R5#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface Tunnel10
R5(config-if)#no ip mfib cef input
R5(config-if)#no ip mfib cef output
!
!R5#debug ip mfib pak
MFIB IPv4 pak debugging enabled for default IPv4 table
MFIBv4(0x0): Pkt (155.1.146.6,224.10.10.10
) from Tunnel10 (PS) Queued signalling for routing protocolMFIBv4(0x0): Pkt (155.1.146.6,224.10.10.10
) from Tunnel10 (PS)
Packet is marked as not to be forwarded by MFIB. Hence it will not be preserved - dropping
MFIBv4(0x0): Pkt (155.1.146.6,224.10.10.10
) from Tunnel10 (PS) Queued signalling for routing protocolMFIBv4(0x0): Pkt (155.1.146.6,224.10.10.10
) from Tunnel10 (PS)
Packet is marked as not to be forwarded by MFIB. Hence it will not be preserved - dropping
MFIBv4(0x0): Pkt (155.1.146.6,224.10.10.10
) from Tunnel10 (PS) Queued signalling for routing protocolMFIBv4(0x0): Pkt (155.1.146.6,224.10.10.10
) from Tunnel10 (PS)
Packet is marked as not to be forwarded by MFIB. Hence it will not be preserved - dropping
```

Now apply the static multicast route on R5. A static default mroute will be enough in this situation.

```
R5(config)#ip mroute 0.0.0.0 0.0.0.0 Tunnel10

!
!R6#ping 224.10.10.10 repeat 10000
Type escape sequence to abort.
Sending 10000, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:
.....................................................................
Reply to request 114 from 155.1.108.10, 99 ms
Reply to request 115 from 155.1.10.10, 79 ms

Reply to request 116 from 155.1.10.10, 41 ms
Reply to request 117 from 155.1.10.10, 28 ms
Reply to request 118 from 155.1.10.10, 13 ms
Reply to request 119 from 155.1.10.10, 21 ms
Reply to request 120 from 155.1.10.10, 21 ms
Reply to request 121 from 155.1.10.10, 64 ms
Reply to request 122 from 155.1.10.10, 14 ms
Reply to request 123 from 155.1.10.10, 11 ms
Reply to request 124 from 155.1.10.10, 17 ms
Reply to request 125 from 155.1.10.10, 7 ms
```

https://t.me/learningnets

```
Reply to request 126 from 155.1.10.10, 12 ms
```

Now the corresponding mroute state has a valid input interface and the OIL is not empty. Notice that the RPF neighbor information is taken from the mroute cache, per the output of the show command.

```
R5#show ip mroute


IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode


(*, 224.10.10.10), 00:09:11/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Dense, 00:09:11/stopped
    Tunnel10, Forward/Dense, 00:09:11/stopped
 (155.1.146.6, 224.10.10.10
), 00:03:09/00:02:50, flags: T Incoming interface: Tunnel10, RPF nbr 155.1.15.1, Mroute
  Outgoing interface list: GigabitEthernet1.58
, Forward/Dense, 00:03:09/stopped
```

Keep in mind that because we used a "default" mroute out of the Tunnel10 interface on R5, all RPF checks performed on R5 will now resolve to that interface. This may not be the desired behavior in certain scenarios. In this case, R5 is only performing an RPF check against the source of the multicast traffic, which is 155.1.146.6. However, if we join a group on R6 and send traffic to it from R10, R5 will perform an RPF check against 155.1.108.10 and it will fail because it will not be coming in the Tunnel10 interface.

https://t.me/learningnets

```
R5#show ip rpf 155.1.108.10

RPF information for ? (155.1.108.10) RPF interface: Tunnel10
RPF neighbor: ? (155.1.15.1)
  RPF route/mask: 0.0.0.0/0  RPF type: multicast (static
)
  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base
```

To fix this, we can just use a more specific mroute on R5 for 155.1.146.6.

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#no ip mroute 0.0.0.0 0.0.0.0 Tunnel 10
R5(config)#ip mroute 155.1.146.6 255.255.255.255 Tunnel 10
!
!R5#show ip rpf 155.1.108.10
RPF information for ? (155.1.108.10) RPF interface: GigabitEthernet1.58
RPF neighbor: ? (155.1.58.8)
  RPF route/mask: 155.1.108.0/24  RPF type: unicast (eigrp 100
)
  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM Sparse Mode

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10.
- Statically configure R5's Loopback0 as the Rendezvous Point.
- Configure R10's GigabitEthernet1.10 interface to join the group **224.10.10.10**, and ensure that R6 can send multicast packets to this group.

Configuration

PIM Sparse Mode is the scalable version of the multicast signaling protocol. Instead of flooding multicast traffic to all potential receivers, PIM SM builds explicit multicast distribution trees from the receivers to the sources. For this reason, this model does not consume large amounts of network resources like PIM DM's flood-and-prune behavior does.

To build an explicit tree to the source, receivers and sources should have a way to dynamically discover each other. This is facilitated by the use of Rendezvous Points, or RPs. An RP is a special router known to both sources and receivers. If a receiver is interested in traffic for a multicast group G, it first builds a multicast distribution tree toward the RP as the fictive "source." This is facilitated by PIM Join messages; the resulting tree is called a shared tree (also called "RP Tree" or "*,G Tree") and it is designated as (*,G). When a source appears in the network, the closest multicast router will contact the RP using PIM Register messages. The PIM Register messages are sent toward the RP across the unicast shortest path,

https://t.me/learningnets

encapsulated in a unicast packet. In fact, a dynamic Tunnel interface is created on all routers in the PIM network to encapsulate these register messages toward the RP. This interface is not configurable, but exists to solve some of the complexities of the unicast registration mechanism of PIM. PIM should be enabled along the shortest path to the RP, or the RPF check for the RP will fail and the registration process will not be completed.

When registration completes, the RP builds a new SPT toward the source using PIM Join messages and starts forwarding received multicast traffic down the (*,G) tree. When the receivers see traffic going down the (*,G) tree from the actual source, they initiate a PIM Join toward the source, building another SPT designated as the (S,G) tree. This new tree follows the optimal path to the source and removes the RP as a possible “bottleneck” of all multicast traffic flows. This process of switching from (*,G) to the (S,G) tree is called multicast SPT switchover. Finally, the receiving router will send a special Prune message toward the RP designated as (S, G, RPbit). This will prune back the duplicate traffic down the (*,G) tree from the RP.

Configuring the RP for multicast groups is a crucial part of PIM SM setup. In this scenario, we use a static RP configuration on every router using the command **`ip pim rp-address <IP> [<ACL>] [override]`** . The ACL parameter lists the groups that are mapped to this particular RP. You could have multiple RP’s using different group lists on every router, for the purpose of load-balancing. Later we will see ways of automatically disseminating RP information using Auto-RP and BSR protocols. But for now, remember that the **`override`** parameter will force the router to retain static information even if a different RP for the group is learned via Auto-RP. Contrary to dynamic routing protocols, static RPs are overridden by default by dynamically learned RPs using Auto-RP or BSR. The **`override`** parameter ensures that statically configured RPs are not overridden by Auto-RP or BSR learned RPs.

In previous versions of IOS, the SPT switch over had a parameter that could be used to influence how much traffic should be received down the (*,G) tree before switching over. In previous versions of IOS, this threshold could be changed by using the command **`ip pim spt-threshold [<Rate in Kbps>|infinity]`** . However, newer versions such as the one used in these demonstrations, only allows two parameters, 0 or infinity, essentially either always switching (0, the default) over as soon as the first packet is received or never switching over (infinity). By setting the rate to infinity, you can effectively disable the use of the shortest-path tree entirely, and all information will be received down the shared tree.

```
R4:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.146
```

https://t.me/learningnets

```
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
```

R6:

```
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.146
 ip pim sparse-mode
```

R5:

```
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface GigabitEthernet1.58
 ip pim sparse-mode
```

R8:

```
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
```

R10:

```
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip igmp join-group 224.10.10.10
 ip pim sparse-mode
```

Verification

https://t.me/learningnets

Before you source any traffic, verify that R10 has joined the shared tree toward the RP. Notice the (*,G) states and the RP address for the group.

```
R10#show ip mroute

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:10:11/00:02:49, RP 150.1.5.5
, flags: SJCL  Incoming interface: GigabitEthernet1.108, RPF nbr 155.1.108.8
  Outgoing interface list:
GigabitEthernet1.10, Forward/Sparse, 00:10:10/00:02:49
!
!R8#show ip mroute


IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
```

https://t.me/learningnets

```
(*, 224.10.10.10), 00:12:02/00:03:15, RP 150.1.5.5
, flags: S   Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5

  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse, 00:12:02/00:03:15
```

The RP itself shows the RPF neighbor of 0.0.0.0 for the (*,G) state, meaning it is the RPF neighbor. At the same time, R4 has no (*,G) state for the group 224.10.10.10 because it is not on the shared tree.

```
R5#show ip mroute

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:13:17/00:02:58, RP 150.1.5.5
, flags: S   Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:13:17/00:02:58
!
!R4#show ip mroute 224.10.10.10
Group 224.10.10.10 not found
```

When you start pinging from R6 and check mroute states on R4, you will notice that the (*,G) state for the group 224.10.10.10 is “prune” because there are no receivers for this group below R4. At the same time, there is an (S,G) state toward R6 that enables traffic flow from R6 down to R10.

```
R6#ping 224.10.10.10 repeat 100
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:
 Reply to request 0 from 155.1.108.10, 199 ms
Reply to request 0 from 155.1.10.10, 199 ms
Reply to request 1 from 155.1.10.10, 89 ms
Reply to request 2 from 155.1.10.10, 20 ms
Reply to request 3 from 155.1.10.10, 48 ms
!
!R4#show ip mroute

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.10.10.10), 00:00:07/stopped, RP 150.1.5.5, flags: SP
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.5
  Outgoing interface list: Null
 (155.1.146.6, 224.10.10.10
), 00:00:07/00:02:52, flags: T
  Incoming interface: GigabitEthernet1.146, RPF nbr 155.1.146.6
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Sparse, 00:00:07/00:03:24
```

When you look at the mroutes on R5, notice that there is a (*,224.10.10.10) state representing the shared tree and a (155.1.146.6,224.10.10.10) state representing the SPT built by the RP. This SPT is used to connect the multicast source to the shared tree.

```
R5#show ip mroute

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
```

https://t.me/learningnets

```
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 224.10.10.10
), 00:16:53/00:03:18, RP 150.1.5.5, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:16:53/00:03:18
 (155.1.146.6, 224.10.10.10
), 00:01:39/00:01:54, flags: T Incoming interface: GigabitEthernet1.45
, RPF nbr 155.1.45.4
  Outgoing interface list: GigabitEthernet1.58
, Forward/Sparse, 00:01:39/00:03:18
```

On R8, the SPT and shared tree overlap (follow the same paths), but in other scenarios they may differ, and this can result in RPF issues.

```
R8#show ip mroute


IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
```

https://t.me/learningnets

```
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 224.10.10.10
), 00:18:03/00:03:07, RP 150.1.5.5, flags: S Incoming interface: GigabitEthernet1.58
, RPF nbr 155.1.58.5
  Outgoing interface list: GigabitEthernet1.108
, Forward/Sparse, 00:18:03/00:03:07
 (155.1.146.6, 224.10.10.10
), 00:02:49/00:00:10, flags: T Incoming interface: GigabitEthernet1.58
, RPF nbr 155.1.58.5
  Outgoing interface list: GigabitEthernet1.108
, Forward/Sparse, 00:02:49/00:03:07
```

R10 has a similar output. Notice that on R10 the 'L' flag is set. This means that the router itself has joined the group.

```
R10#sh ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected, L - Local
, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 224.10.10.10
), 00:19:02/stopped, RP 150.1.5.5, flags: SJCL Incoming interface: GigabitEthernet1.108
, RPF nbr 155.1.108.8
  Outgoing interface list: GigabitEthernet1.10
, Forward/Sparse, 00:19:01/00:02:06
 (155.1.146.6, 224.10.10.10), 00:03:47/00:02:12, flags: L
JT Incoming interface: GigabitEthernet1.108
, RPF nbr 155.1.108.8
  Outgoing interface list: GigabitEthernet1.10
, Forward/Sparse, 00:03:47/00:02:06
```

When R6 sent the first multicast packet to 224.10.10.10, it was encapsulated inside

https://t.me/learningnets

of a PIM Register message toward the RP. There is a dynamically created Tunnel interface on all PIM Sparse-Mode routers that is used for this register message encapsulation. The PIM register Tunnel can also be observed from any router on the PIM Sparse Mode network, but we will look at it from R6 because this is where the registers were sent from.

```
R6#show ip pim tunnel

Tunnel0
  Type  : PIM Encap
  RP    : 150.1.5.5
  Source: 155.1.146.6
```

This Tunnel interface appears in "show ip interface brief," but it does not appear in the running configuration. As mentioned previously, this interface is not configurable.

```
R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface Tunnel0
%Tunnel0 used by PIM for Registering, configuration not allowed
```

We can see the configuration of this interface by using the following command.

```
R6#show derived-config interface tunnel0

Building configuration...

Derived configuration : 201 bytes
!
interface Tunnel0
 description Pim Register Tunnel (Encap) for RP 150.1.5.5
 ip unnumbered GigabitEthernet1.146
 tunnel source GigabitEthernet1.146
 tunnel destination 150.1.5.5
 tunnel tos 192
end
```

On the RP, two Tunnel interfaces are created, one for encapsulating the PIM Register messages and another one for decapsulating them.

```
R5#show ip pim tunnel

Tunnel1
```

https://t.me/learningnets

```
  Type  : PIM Encap

  RP    : 150.1.5.5*
  Source: 155.1.45.5
Tunnel2*   Type  : PIM Decap

  RP    : 150.1.5.5*
  Source: -
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM Sparse-Dense Mode

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable sparse-dense-mode multicast delivery on the Ethernet path between R6 and R10.
  - Do not enable PIM on the DMVPN link between R4 and R5.
- Statically configure R5's Loopback0 interface as the Rendezvous Point for groups in the range **224.0.0.0/8**.
- Configure R10's GigabitEthernet1.10 interface to join the groups **224.10.10.10** and **239.0.0.1**, and ensure that R6 can send multicast packets to both groups.

Configuration

PIM Sparse-Dense mode is a hybrid of the Sparse and Dense mode operations. However, it does not define any extension to the PIM protocol. Rather, when you apply the command `ip pim sparse-dense-mode` to an interface, the router will forward traffic for both sparse and dense multicast groups out of this interface. A "sparse" multicast group is the one that has an RP defined. This mode of operation is useful when you have a range of groups that you want to be delivered using the simple dense-mode. As we will see later, Auto-RP groups are a good example of dense-mode groups needed to be flooded along with sparse-groups.

Most of the time you don't have to define PIM SM/DM interfaces, unless you are using the Cisco proprietary Auto-RP protocol. The problem with PIM SM/DM is that any group that does not have an RP defined is treated like a dense-mode group. Consider a situation in which some routers lose RP information for their sparse-

https://t.me/learningnets

mode groups. For example, say some routers haven't received Auto-RP announcements for some time and the old information has expired. In this situation, these routers will switch all groups to dense-mode and start flooding the network with multicast traffic. To prevent this behavior, you should either use PIM SM only along with a standard RP information dissemination protocol, such as BSR, or issue the command **`no ip dm-fallback`** on all PIM SM/DM routers. This will prevent the DM fallback behavior and only allow forwarding for sparse-mode groups.

```
R4:
ip access-list standard SPARSE_GROUPS
 permit 224.0.0.0 0.255.255.255
!
ip multicast-routing distributed
ip pim rp-address 150.1.5.5 SPARSE_GROUPS
!
interface GigabitEthernet1.146
 ip pim sparse-dense-mode
!
interface GigabitEthernet1.45
 ip pim sparse-dense-mode
R6:
ip access-list standard SPARSE_GROUPS
 permit 224.0.0.0 0.255.255.255
!
ip multicast-routing distributed
ip pim rp-address 150.1.5.5 SPARSE_GROUPS
!
interface GigabitEthernet1.146
 ip pim sparse-dense-mode
R5:
ip access-list standard SPARSE_GROUPS
 permit 224.0.0.0 0.255.255.255
!
ip multicast-routing distributed
ip pim rp-address 150.1.5.5 SPARSE_GROUPS
!
interface GigabitEthernet1.45
 ip pim sparse-dense-mode
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
R8:
ip access-list standard SPARSE_GROUPS
 permit 224.0.0.0 0.255.255.255
!
```

https://t.me/learningnets

```
ip multicast-routing distributed
ip pim rp-address 150.1.5.5 SPARSE_GROUPS
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
R10:

ip access-list standard SPARSE_GROUPS
 permit 224.0.0.0 0.255.255.255
!
ip multicast-routing distributed
ip pim rp-address 150.1.5.5 SPARSE_GROUPS
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
!
interface GigabitEthernet1.10
 ip igmp join-group 224.10.10.10
 ip igmp join-group 239.0.0.1
 ip pim sparse-dense-mode
```

Verification

First, ping the group that has no RP from R6. Notice that all routers on the path to the receiver create (S,G) dense state entries. There is also a corresponding (*,G) state, but it has an RP value of 0.0.0.0 and has all PIM-enabled interfaces listed in the OIL.

```
R6#ping 239.0.0.1 repeat 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 239.0.0.1, timeout is 2 seconds:
...Reply to request 3 from 155.1.108.10, 20 ms
Reply to request 4 from 155.1.10.10, 45 ms
Reply to request 5 from 155.1.10.10, 6 ms
Reply to request 6 from 155.1.10.10, 31 ms
Reply to request 7 from 155.1.10.10, 57 ms
Reply to request 8 from 155.1.10.10, 69 ms
!
!R4#show ip mroute 239.0.0.1
```

https://t.me/learningnets

```
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 239.0.0.1), 00:00:56/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Sparse-Dense, 00:00:56/stopped
    GigabitEthernet1.146, Forward/Sparse-Dense, 00:00:56/stopped
(155.1.146.6, 239.0.0.1), 00:00:56/00:02:03, flags: T
  Incoming interface: GigabitEthernet1.146, RPF nbr 155.1.146.6
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Sparse-Dense, 00:00:56/stopped
!
!R5#show ip mroute 239.0.0.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 239.0.0.1), 00:01:56/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
```

https://t.me/learningnets

```
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:01:56/stopped
    GigabitEthernet1.45, Forward/Sparse-Dense, 00:01:56/stopped
 (155.1.146.6, 239.0.0.1), 00:01:56/00:01:03, flags: T
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:01:56/stopped
!
!R8#show ip mroute 239.0.0.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 239.0.0.1), 00:03:04/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse-Dense, 00:03:04/stopped
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:03:04/stopped
 (155.1.146.6, 239.0.0.1), 00:03:04/00:02:55, flags: T

  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse-Dense, 00:03:04/stopped
```

Now, again from R6, ping the group that has an RP configured. Notice that (*,G) now represents the shared tree and is used to forward the first packets from the sources. The SPT is built by the RP, and the router is connected to the receiver.

```
R6#ping 224.10.10.10 repeat 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:
 Reply to request 0 from 155.1.108.10, 35 ms
Reply to request 0 from 155.1.10.10, 35 ms
```

https://t.me/learningnets

```
Reply to request 1 from 155.1.10.10, 9 ms
Reply to request 2 from 155.1.10.10, 6 ms
Reply to request 3 from 155.1.10.10, 6 ms
Reply to request 4 from 155.1.10.10, 3 ms
!
!R5#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:05:33/stopped, RP 150.1.5.5, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:05:33/00:02:52
(155.1.146.6, 224.10.10.10), 00:00:11/00:03:22, flags: T
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:00:11/00:03:18
!
!R8#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
```

https://t.me/learningnets

```
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:07:06/00:03:15, RP 150.1.5.5, flags: S
  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse-Dense, 00:07:06/00:03:15
(155.1.146.6, 224.10.10.10), 00:01:44/00:01:15, flags: T

  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse-Dense, 00:01:44/00:03:15
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM Assert

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM dense-mode multicast delivery on the Ethernet path between R6 and R10.
  - Enable PIM dense-mode also on the Tunnel10 interface between R1 and R5.
- Ensure that R1 is always elected via PIM to flood traffic onto the shared VLAN 146 segment.
  - Use one static mroute to fix underlying RPF issues.
  - For testing purposes, join R6's Loopback0 to **239.6.6.6** multicast group and ensure that R10 can send packets to this group.

Configuration

If multiple multicast routers share a single segment, all of them could flood this segment with the same multicast traffic. For example, if both R1 and R4 receive the same multicast flow from their upstream neighbors, they will both send it to R6. From R6's point of view, both flows are the same with respect to RPF validation. Thus, traffic duplication occurs.

To avoid this situation, only one router is allowed to flood traffic on the shared segment. When a router detects that someone is sending traffic for the same source IP/destination group on the segment, where it has an active (S,G) state for the same group/source, it immediately originates a PIM Assert message. This message contains the source IP, the group, and the path cost to the source. The path cost is a touple (AD, Metric), where AD is the administrative distance of the routing protocol

https://t.me/learningnets

used to look up the source IP, and Metric is that same protocol's metric to reach the source. The router with the best (lowest) AD value on the segment wins the assertion. If the ADs are equal, metric is used as tie-breaker. If both the AD and the metric are the same, the router with the highest IP address wins. If another router on the segment receives the Assert message and determines that it loses the assertion, it will remove the (S,G) state on its interface and stop flooding traffic. However, if it sees itself as the winner, it will emit a superior PIM Assert message to inform the other router that it should stop flooding the traffic for this (S,G) pair.

Note that the PIM Assert procedure might be dangerous on NBMA interfaces. PIM will treat those interfaces as fully broadcast capable networks, even though not all nodes are capable of hearing each-other's broadcast messages. Imagine a hub-and-spoke DMVPN topology. If both the hub and the spoke start flooding the segment with the same multicast traffic, PIM Assert will occur. If for some reason the spoke should win, the hub will stop sending its multicast traffic. However, all traffic coming from the spoke to the hub is NOT relayed back to the other spokes sharing the same segment, based on the RPF rule. Thus, all other spokes will effectively stop receiving the multicast traffic. To avoid this situation, either use PIM NBMA mode (described in a separate task) or make sure that the hub always wins the PIM Assert procedure. PIM NBMA mode, however, is supported only in PIM Sparse Mode.

In our scenario, R1 and R4 run different IGPs. This is not a well-designed multicast solution, because you may want all routers to be under the same IGP. In our case, under normal circumstances, R4 would be the assert winner because EIGRP has a better AD (90) than OSPF (110). However, R1 needs a static mroute out its Tunnel10 interface to R5 to fix RPF issues. R1 will receive multicast traffic from 155.1.108.10 on its Tunnel10 interface, but its IGP route to get to that source is via its GigabitEthernet1.13 interface. Having this mroute on R1 will implicitly make R1 win the Assert procedure. R1 will report its AD to the source as 1 with a metric of 0, while R4 will report its AD to the source as 90 (EIGRP) with a metric of 3328.

If we wanted to make R4 the Assert winner, and at the same time fix the RPF issue, we could use the distance argument at the end of the mroute on R1 to something higher than 90.

R1:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim dense-mode
!
interface Tunnel 10
 ip pim dense-mode
!
ip mroute 155.1.108.10 255.255.255.255 Tunnel10
```

https://t.me/learningnets

R4:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim dense-mode
!
interface GigabitEthernet1.45
 ip pim dense-mode
```

R6:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim dense-mode
!
interface Loopback0
 ip pim dense-mode
 ip igmp join-group 239.6.6.6
```

R5:

```
ip multicast-routing distributed
!
interface Tunnel 10
 ip pim dense-mode
!
interface GigabitEthernet1.58
 ip pim dense-mode
!
interface GigabitEthernet1.45
 ip pim dense-mode
```

R8:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim dense-mode
!
interface GigabitEthernet1.108
 ip pim dense-mode
```

R10:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim dense-mode
!
interface GigabitEthernet1.10
```

https://t.me/learningnets

```
 ip pim dense-mode
```

Verification

Join the group 239.6.6.6 on R6's Loopback0 and ping this IP address from R10. The group will use simple dense-mode forwarding, and R1 will be elected as the assert winner on the VLAN 146 segment. Both R1 and R4 initially try to send the traffic to R6. The PIM Assert Process stops this parallel forwarding of traffic when one router becomes the PIM Forwarder. In this case, we “rigged” the election by assigning a better Administrative Distance to R1 via the use of the mroute. At the same time, we solved the underlying RPF issue that would have prevented R1 from receiving multicast traffic from R10 on its Tunnel10 interface:

```
R10#ping 239.6.6.6 repeat 1000
Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 239.6.6.6, timeout is 2 seconds:
...Reply to request 3 from 155.1.146.6, 30 ms
Reply to request 4 from 150.1.6.6, 18 ms

Reply to request 5 from 150.1.6.6, 10 ms
Reply to request 6 from 150.1.6.6, 30 ms
Reply to request 7 from 150.1.6.6, 20 ms
Reply to request 8 from 150.1.6.6, 17 ms
Reply to request 9 from 150.1.6.6, 12 ms
```

The mroute entry on R1 is marked with an “A” flag, which means “Assert Winner.”.

```
R1#show ip mroute 239.6.6.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group Outgoing interface flags: H - Hardware switched, A - Assert winner
, p - PIM Join
 Timers: Uptime/Expires
```

https://t.me/learningnets

```
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.6.6.6), 00:02:18/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel10, Forward/Dense, 00:02:18/stopped
    GigabitEthernet1.146, Forward/Dense, 00:02:18/stopped
 (155.1.108.10, 239.6.6.6
), 00:02:18/00:00:41, flags: T
  Incoming interface: Tunnel10, RPF nbr 155.1.15.5, Mroute
  Outgoing interface list: GigabitEthernet1.146, Forward/Dense, 00:02:18/stopped, A
```

If you were to enable the following debugging on R1 before sending pings from R10, you could catch the PIM Assert message exchange between R1 and R4. Notice that R1 wins because of the better AD.

```
R1#debug ip pim
!
!PIM(0): Received v2 Assert on GigabitEthernet1.146 from 155.1.146.4
PIM(0): Assert metric to source 155.1.108.10 is [90/3328]
PIM(0): We win, our metric [1/0]
PIM(0): Schedule to prune GigabitEthernet1.146
PIM(0): (155.1.108.10/32, 239.6.6.6) oif GigabitEthernet1.146 in Forward state PIM(0):
Send v2 Assert on GigabitEthernet1.146 for 239.6.6.6, source 155.1.108.10, metric [1/0]

PIM(0): Assert metric to source 155.1.108.10 is [1/0] PIM(0): We win, our metric [1/0]

PIM(0): (155.1.108.10/32, 239.6.6.6) oif GigabitEthernet1.146 in Forward state
```

R4 prunes the interface when it loses the assert procedure from the OIL for the group 239.6.6.6. What happens if the Assert Winner stops working? Unfortunately, the Assert “Loser” has no way of knowing that the Assert “Winner” has failed and will wait three (3) minutes before timing out its pruned interface. Thus, we face a worst case scenario of loss of traffic for three minutes should the PIM Assert winner go down right after winning the election.

```
R4#show ip mroute 239.6.6.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
```

https://t.me/learningnets

```
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.6.6.6), 00:03:32/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Dense, 00:03:32/stopped
    GigabitEthernet1.146, Forward/Dense, 00:03:32/stopped
(155.1.108.10, 239.6.6.6), 00:00:31/00:02:28, flags: PT
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.5
  Outgoing interface list: GigabitEthernet1.146, Prune/Dense, 00:00:31/00:02:28
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM Accept RP

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the path between R5 and R10.
- Configure R5's Loopback0 as the static Rendezvous Point.
- Ensure that R5 and R8 will accept (*,G) joins toward R5's Loopback0 only for groups **224.10.10.10** and **224.110.110.110**.

Configuration

This is a security feature used by IOS routers to prevent unwanted RPs or groups from becoming active in the PIM SM multicast domain. When you configure `ip pim accept-rp [rp-address | auto-rp] [access-list]` , the local router will only accept (*,G) Join/Prune messages toward the RP-address specified in the command. Additionally, if the access-list parameter is specified, the router will only accept Join/Prune messages for groups matching this access-list. Note that regular SPT joins are not affected by this configuration.

To implement thorough security using this feature, you must configure every router on all potential paths to the RP, or simply configure your RP. Performing this on the RP alone, however, will allow illegal joins across the network, but they will eventually be dropped at the RP.

```
R5:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
```

https://t.me/learningnets

```
ip access-list standard ALLOWED_GROUPS
 permit 224.10.10.10
 permit 224.110.110.110
!
ip pim accept-rp 150.1.5.5 ALLOWED_GROUPS
!
interface GigabitEthernet1.58
 ip pim sparse-mode
R8:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
ip access-list standard ALLOWED_GROUPS
 permit 224.10.10.10
 permit 224.110.110.110
!
ip pim accept-rp 150.1.5.5 ALLOWED_GROUPS
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
R10:

ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip pim sparse-mode
```

Verification

Join R10's GigabitEthernet1.10 interface to the groups 224.10.10.10 and 224.11.11.11. Observe the console output on R8.

```
R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#interface GigabitEthernet1.10
R10(config-subif)#ip igmp join-group 224.10.10.10
R10(config-subif)#ip igmp join-group 224.11.11.11
```

https://t.me/learningnets

```
!
!
R8#%PIM-6-INVALID_RP_JOIN:Received (*, 224.0.1.40) Join from 155.1.108.10 for invalid RP 150.1.5.5
%PIM-6-INVALID_RP_JOIN:Received (*, 224.11.11.11) Join from 155.1.108.10 for invalid RP 150.1.5.5
```

Notice that R8 rejected joins toward R5's Loopback0 for the invalid groups. If you are wondering what group 224.0.1.40 is, it's the Auto-RP discovery group, which all routers join by default. Because we happened to specify an RP for the group range covering this particular one (default), every router attempts to join the shared tree for it. Using the **`show ip mroute`** command, you can observe that only group 224.10.10.10 has an mroute state corresponding to the shared tree.

```
R8#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.10.10.10), 00:05:13/00:03:09, RP 150.1.5.5, flags: S
  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse, 00:05:13/00:03:09
!
!R8#show ip mroute 224.11.11.11
Group 224.11.11.11 not found
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM DR Election

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode on the LAN segment between R1, R4, and R6 (GigabitEthernet1.146), and between R4 and R5's GigabitEthernet1.45.
- Statically configure R5's Loopback0 interface as the Rendezvous Point.
- Ensure that R4 is responsible for multicast source registration with the RP for devices on the VLAN146 segment.

Configuration

A PIM Designated Router (DR) is elected on every multiple-access segment—that is, every segment where multiple routers share the same medium/subnet. This election process is based on the highest priority and highest IP address; the router with the numerically higher value wins the election. This process is *preemptive* and every new router with a better priority will preempt the previous DR.

One purpose of a DR is to signal multicast delivery trees using PIM messages when it sees interested receivers on the shared segment by means of IGMP. Another purpose is to register active sources on the segment with the regional RP. When the DR hears multicast packets on the segment, it will check to determine whether the destination group has an RP. If it does, the data packets are encapsulated into special PIM Register messages and sent to the RP. The RP will start forwarding them down the shared tree if there are any active subscribers. At the same time, the RP will build a shortest-path tree toward the DR and send a PIM Register-Stop message to the DR to inform it that regular forwarding may start now. After this, the

https://t.me/learningnets

multicast traffic is delivered over the SPT. Note that PIM Register messages are subject to RPF checks, as usual. If the Register message is received on a non-RPF interface, the check will fail.

```
R5:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.45
 ip pim sparse-mode
R1:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.146
 ip pim sparse-mode
R4:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.146
 ip pim sparse-mode
 ip pim dr-priority 100
!
interface GigabitEthernet1.45
 ip pim sparse-mode
R6:

ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.146
 ip pim sparse-mode
```

Verification

```
R4#show ip pim interface GigabitEthernet1.146 detail

GigabitEthernet1.146 is up, line protocol is up
  Internet address is 155.1.146.4/24
  Multicast switching: fast
  Multicast packets in/out: 1024/112
  Multicast TTL threshold: 0
  PIM: enabled
```

https://t.me/learningnets

```
    PIM version: 2, mode: sparse PIM DR: 155.1.146.4 (this system)
    PIM neighbor count: 2
    PIM Hello/Query interval: 30 seconds
    PIM Hello packets in/out: 4/4
    PIM State-Refresh processing: enabled
    PIM State-Refresh origination: disabled
    PIM NBMA mode: disabled
    PIM ATM multipoint signalling: disabled
    PIM domain border: disabled
    PIM neighbors rpf proxy capable: TRUE
    PIM BFD: disabled
    PIM Non-DR-Join: FALSE
  Multicast Tagswitching: disabled
!
!R1#show ip pim neighbor gigabitEthernet1.146
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode
155.1.146.6       GigabitEthernet1.146     00:04:25/00:01:18 v2    1 / S P G 155.1.146.4
       GigabitEthernet1.146     00:01:24/00:01:19 v2 100/ DR
 S P G
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM Accept Register

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10.
  - Do not run PIM on the DMVPN link between R4 and R5.
- Statically configure R5's Loopback0 interface as the Rendezvous Point.
- Configure R5 so that the only source allowed on the VLAN146 segment is R6.
- Your configuration should not affect sources on any other segments.

Configuration

This security feature is configured on the PIM SM Rendezvous Point and specifies the sources that are allowed to register with the RP. Remember that registration is performed by the PIM DR router, and every register message contains the original multicast packet, which includes the IP address of the multicast source and the group destination address. If the RP denies the registration, it sends a PIM Register-Stop to the DR immediately and never builds the SPT toward the source. The command to enable PIM Register message filtering is `ip pim accept-register [list <Extended-ACL | route-map <Route-Map>]` . The basis of filtering is an extended ACL in the format:

```
ip access-list REGISTER_FILTER
 permit ip <source-ip> <source-wildcard> <group-address> <group-wildcard>
```

https://t.me/learningnets

In these access-lists, we can match both sources and destination groups at the same time. For example, to permit the source 155.1.146.1 to send traffic to any group via the RP, use the entry **`permit ip host 155.1.146.1 224.0.0.0 15.255.255.255 .`** Be advised that if the RP is the DR for the same segment, this filtering will not work. In newer versions of IOS code, a route-map can no longer be used with the **`ip pim accept-register`** command. The only parameter that can be used to do filtering with this command is an access-list, such as how this task uses it. For our scenario, we create an access-list and apply it to the **`ip pim accept-register`** command. The access-list only permits R6 from the VLAN 146 segment, and allows all other sources.

```
R4:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
R6:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.146
 ip pim sparse-mode
R5:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
ip access-list extended VLAN146_SOURCES
 permit ip host 155.1.146.6 any
 deny ip 155.1.146.0 0.0.0.255 any
 permit ip any any
!
ip pim accept-register list VLAN146_SOURCES
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface GigabitEthernet1.58
 ip pim sparse-mode
R8:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
R10:

ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip pim sparse-mode
```

Verification

First, we join R10 to group 224.10.10.10 and then try pinging it from R6. Because the IP address of R6 is permitted to register with the RP, everything goes smoothly.

```
R10(config)#interface GigabitEthernet1.10
R10(config-subif)#ip igmp join-group 224.10.10.10
!
!R6#ping 224.10.10.10 repeat 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:
Reply to request 0 from 155.1.108.10, 78 ms
Reply to request 0 from 155.1.10.10, 78 ms

Reply to request 1 from 155.1.10.10, 18 ms
Reply to request 2 from 155.1.10.10, 14 ms
Reply to request 3 from 155.1.10.10, 7 ms
Reply to request 4 from 155.1.10.10, 114 ms
Reply to request 5 from 155.1.10.10, 67 ms
Reply to request 6 from 155.1.10.10, 15 ms
```

Now temporarily create interface GigabitEthernet1.146 on R7 and make R4 the DR for the LAN segment as follows.

```
R7(config)#ip multicast-routing distributed
R7(config)#ip pim rp-address 150.1.5.5
R7(config)#interface GigabitEthernet1.146
R7(config-subif)# encapsulation dot1Q 146
```

https://t.me/learningnets

```
!
!R4(config)#interface GigabitEthernet1.146
R4(config-subif)#ip pim dr-priority 100
```

Enable PIM debugging on R4 (the DR) using the command **`debug ip pim`**. Notice that R4 receives a Register-Stop message immediately after sending the first Register packet. However, this time the RP does not build an SPT toward the source.

```
R7#ping 224.10.10.10 repeat 100



Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:...
!
!R4#show logging

<snip>
PIM(0): Adding register encap tunnel (Tunnel1) as forwarding interface of (155.1.146.7, 224.10.10.10).
PIM(0): Received v2 Register-Stop on GigabitEthernet1.45 from 150.1.5.5

PIM(0):   for source 155.1.146.7, group 224.10.10.10
PIM(0): Removing register encap tunnel (Tunnel1) as forwarding interface of (155.1.146.7, 224.10.10.10).
PIM(0): Clear Registering flag to 150.1.5.5 for (155.1.146.7/32, 224.10.10.10)
```

You may discover more on R5, the RP. Notice the console message that warns you about an invalid source. Also, there is no SPT entry toward the new VLAN146 IP address of R7 in the mroute table of R5. Thus, the source is not connected to the shared tree.

```
R5#
%PIM-4-INVALID_SRC_REG: Received Register from 155.1.45.4 for (155.1.146.7, 224.10.10.10),
not willing to be RP
!
!R5#show ip mroute 224.10.10.10

IP Multicast Routing Table
<snip>

(*, 224.10.10.10), 00:43:51/stopped, RP 150.1.5.5, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:43:51/00:02:58
```

https://t.me/learningnets

```
(155.1.146.6, 224.10.10.10), 00:00:03/00:03:28, flags: T
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:00:03/00:03:26
```

In this lab, we avoided doing verification on R1 and R4 for the Accept-Register configuration. This was to avoid issues related to packets being flooded out all-PIM enabled interfaces on these devices. In this particular configuration, this situation leads to traffic being able hit R5 (the RP) and thus being able to flow down the shared tree.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Multicast Tunneling

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Configure PIM sparse-Mode between R5 and R10.
- Statically configure R5's Loopback0 interface as the Rendezvous Point.
- Without configuring multicast routing on the transit nodes, make sure that receivers on R9 may receive multicast feeds from R10.
  - Do not configure any additional IGP adjacencies to accomplish this task.
  - Additional interfaces can be configured.
  - For testing purposes, join R9's VLAN9 interface to multicast group **224.10.10.10**.

Configuration

Tunneling is a common technique used with multicast technologies to traverse non-multicast capable networks. This greatly simplifies configuration, but it has the major drawback of "hiding" the real network topology and thus often prevents effective multicast replication. When configuring a tunnel, you have an option of enabling or disabling the IGP on the tunnel interface. If you plan to use the tunnel for multicast traffic delivery, it is necessary to enable PIM on the same interface. Because of the following two factors, you may easily encounter RPF failure issues:

- If no IGP is running on the tunnel, you must provide static mroutes to correct any RPF checks.
- If IGP is running on the tunnel interface, the cost of traversing the tunnel might be

https://t.me/learningnets

higher than the cost of traversing the underlying network. It is generally not a good idea to run the same IGP on the underlying network and on the tunnel. You may need to adjust metrics or provide static mroutes to correct any resulting problems.

In our situation, we are not running an IGP on the tunnel, so we must provide static mroute statements to allow the endpoints to perform RPF checks correctly.

```
R8:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
R10:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.108
 ip pim sparse-mode
R5:
ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface Tunnel59
 ip unnumbered Loopback0
 tunnel source Loopback0
 tunnel destination 150.1.9.9
 ip pim sparse-mode
!
router ospf 1
 passive-interface Tunnel59
R9:

ip multicast-routing distributed
ip pim rp-address 150.1.5.5
!
interface GigabitEthernet1.9
 ip igmp join-group 224.10.10.10
 ip pim sparse-mode
```

https://t.me/learningnets

```
!
interface Tunnel59
 ip unnumbered Loopback0
 tunnel source Loopback0
 tunnel destination 150.1.5.5
 ip pim sparse-mode
!
router ospf 1
 passive-interface Tunnel59
!
ip mroute 0.0.0.0 0.0.0.0 Tunnel59
```

Verification

Note that enabling PIM on R9's VLAN 9 interface is mandatory; otherwise, IOS will not forward multicasts out of this interface. Notice that all RPF information on R9 is condensed into a single default mroute. It overrides all checks using the unicast routing table and allows R9 to receive multicasts successfully.

```
R9#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.10.10.10), 00:00:59/00:02:01, RP 150.1.5.5, flags: SJCL
  Incoming interface: Tunnel59, RPF nbr 150.1.5.5, Mroute
  Outgoing interface list:
    GigabitEthernet1.9, Forward/Sparse, 00:00:59/00:02:01
!
!R5#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
```

https://t.me/learningnets

```
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 224.10.10.10), 00:01:39/00:02:49, RP 150.1.5.5, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel59, Forward/Sparse, 00:01:39/00:02:49
!
!R10#ping 224.10.10.10

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:
 Reply to request 0 from 155.1.9.9, 95 ms
 Reply to request 0 from 155.1.9.9, 125 ms

Reply to request 0 from 155.1.9.9, 125 ms
Reply to request 1 from 155.1.9.9, 77 ms
Reply to request 1 from 155.1.9.9, 77 ms
Reply to request 1 from 155.1.9.9, 77 ms
Reply to request 2 from 155.1.9.9, 62 ms
Reply to request 2 from 155.1.9.9, 62 ms
Reply to request 3 from 155.1.9.9, 20 ms
Reply to request 4 from 155.1.9.9, 22 ms
Reply to request 5 from 155.1.9.9, 61 ms
Reply to request 6 from 155.1.9.9, 63 ms
Reply to request 7 from 155.1.9.9, 138 ms
Reply to request 8 from 155.1.9.9, 11 ms
```

We did not need to use an mroute on R5 because there is no RPF check performed by R5 of any address on R9. R9, however, does have to perform an RPF check, and because the only interface with PIM on it is Tunnel59, we can safely use a "default" mroute. If we put the receiver (igmp join) on R10 and sourced the multicast from R9, we would need to also add an mroute on R5 to fix the RPF failure. In that

https://t.me/learningnets

scenario, we would not want to use a "default" mroute of 0.0.0.0 0.0.0.0, but instead use a more specific route toward R9's address. If we added a "default" mroute on R5 pointing out of the Tunnel59 interface, all RPF lookups would resolve to the Tunnel59 interface. Although it would work for multicast sourced from R9, if R5 did an RPF check for any other device on the network, it would fail.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Auto-RP

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-dense-mode multicast delivery on the path between R5 and R10.
- Using a Cisco proprietary technology, configure R5 so that it advertises itself as the RP for all multicast groups.
  - Ensure that R5 also floods the information in the network.

Configuration

Auto-RP was the first protocol to automatically distribute RP information across the multicast domain when using PIMv1. Auto-RP is a Cisco proprietary protocol, and it was later “replaced” by the standards-based Bootstrap Router (BSR) protocol with the advent of PIMv2. However, all Cisco routers still support Auto-RP along with PIMv2 protocol.

https://t.me/learningnets

Auto-RP defines two new concepts: Candidate RP (cRP) and Mapping Agents (MA). cRP is any router that is willing to become an RP. You configure this router using the command `ip pim send-rp-announce <Interface> scope <TTL> [group-list <Std-ACL>] [interval <seconds>]` . The router will start sending UDP packets to the IP/Port 224.0.1.39/496 with the list of groups serviced by this particular RP. The cRP announcements are originated every 60 seconds by default, with the Time-to-Live field in the IP headers set to `<TTL>` , a form of administrative scoping. The interface that you specify in this command must be enabled for PIM and its IP address will be used as the RP's IP address. The list of groups is defined using the `<Std-ACL>` access-list. You configure this list using standard access-list syntax. For example:

```
access-list GROUPS permit 239.0.0.0 0.0.0.255
access-list GROUPS permit 232.0.2.0 0.0.0.255
access-list GROUPS deny 224.0.1.50 0.0.0.0
```

The Auto-RP code will convert wildcard masks into the prefix-lengths, so you cannot use discontinuous masks. Next, the "deny" statements are interpreted as a special type of *negative* group announcement. This means that all groups matching this range should be treated as "dense" mode groups. We will discuss how the routers interpret the mapping entries a bit later.

The cRP announcements are flooded across the network and reach special routers called Mapping Agents. You configure these routers using the command `ip pim send-rp-discovery <Interface> scope <TTL> interval <Seconds>` . Mapping Agents listen on the standard address 224.0.1.39 and collect all announcements from candidate RPs. After that, every mapping agent compiles a resulting list of Group to RP mappings and starts sending "RP discovery" messages to the special multicast address 224.0.1.40 port 496. The discovery messages contain an amalgam of all information learned by the mapping agents. Notice that if there are multiple MAs in the network, they will hear each other, and then all of them, except the one with the highest IP address, will cease sending discoveries. When building a discovery message, an MA will follow a few simple rules:

- If there are two announcements with the same group range but different RPs, the MA will select the announcement with the highest RP IP address.
- If there are two announcements where one group is a subset of another but the RPs are different, both will be sent.
- All other announcements are grouped together without any conflict resolution.

All regular routers join the multicast group 224.0.1.40 and listen to the discovery messages. Based on their content, they populate their Auto-RP cache and learn about Group-to-RP mappings. The cache contains both "negative" and "positive"

https://t.me/learningnets

entries. When looking for an RP, Auto-RP code will first scan through negative entries. If a match is found, the group is considered to be dense. Note that RP information for the negative entries will be effectively ignored. If the group is not found in the “negative” list, the code looks it up in the positive list. Because every group in the list is bound to a particular RP, there could be conflicts when multiple RPs try to service overlapping group ranges. The receiving router uses the longest-match rule to resolve all conflicts: if there are multiple matches, only the one with the longest prefix length is selected.

Note that “negative” statements could be defined at any cRP and affect all routers in the multicast region. For example, if a single group list contained the **`deny any`** statement, all groups would be treated as dense, even if there are “positive” entries. The last thing to discuss about Auto-RP is how the multicast groups 224.0.1.39 and 224.0.1.40 are propagated across the network. Because there is no explicit RP information for these groups, they must use dense mode forwarding. This requires the use of **`pim sparse-dense-mode`** on all interfaces within the multicast domain. As discussed before, this is not the safest thing to do in a large-scale network. You may want to use the **`no ip dm-fallback`** global command in such situations or use the Auto-RP Listener feature, discussed in a separate task.

Note that you may use PIM SM mode along with Auto-RP if you define a static RP value for the Auto-RP groups (224.0.1.39 and 224.0.1.40). This will require you to use the **`override`** option when defining the static RP. By default, Auto-RP announcements override a statically configured RP. If you want them to persist, use the **`override`** keyword along with the **`ip pim rp-address`** command.

You might be asking one question: Why is there a need for a Mapping Agent? Couldn’t candidate RPs just broadcast themselves to all routers and the latter learn/elect the best RPs directly? This is possible, but it might result in different routers electing different RPs for the same group ranges. Some routers may miss announcements of a particular candidate RP because of network outages or RP failures. Therefore, cRP information should be collected at one single point before being disseminated to the multicast routers.

R5:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
!
interface Loopback0
 ip pim sparse-dense-mode
!
ip pim send-rp-announce Loopback0 scope 10
ip pim send-rp-discovery loopback0 scope 10
```

https://t.me/learningnets

```
R8:

ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
```

Verification

Initial verification includes checking for RP to group mappings. In our case, there is just one RP, so all groups map to it. Make sure that you perform the verification on all routers, because some of them might reject discovery announcements because of RPF issues.

```
R5#show ip pim rp mapping

This system is an RP (Auto-RP)
This system is an RP-mapping agent (Loopback0)
Group(s) 224.0.0.0/4
RP 150.1.5.5 (?), v2v1

    Info source: 150.1.5.5 (?), elected via Auto-RP
         Uptime: 00:07:10, expires: 00:02:47
           Uptime: 00:00:34, expires: 00:02:24
```

If you are wondering what the (?) sign means, it is for the hostname of the RP. IOS will display the hostname only if the router is configured to do DNS name lookup but the IP address of the RP also has to resolve to a name.

We can see in the mroute table of R5 that there is an (S,G) entry for both 224.0.1.39 and 224.0.1.40. The source is R5's Loopback0 because this is the interface being

https://t.me/learningnets

used for the cRP and Mapping Agent roles.

https://t.me/learningnets

```
R5#show ip mroute

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.10.10.10), 00:06:40/00:02:43, RP 150.1.5.5, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:06:40/00:02:43

(*, 224.0.1.39), 00:09:23/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback0, Forward/Sparse-Dense, 00:07:40/stopped
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:09:23/stopped
 (150.1.5.5, 224.0.1.39
), 00:07:40/00:02:19, flags: LT
  Incoming interface: Loopback0, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:07:40/stopped

(*, 224.0.1.40), 00:09:23/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:07:32/stopped
    Loopback0, Forward/Sparse-Dense, 00:07:39/stopped
 (150.1.5.5, 224.0.1.40
), 00:06:41/00:02:17, flags: LT
  Incoming interface: Loopback0, RPF nbr 0.0.0.0
  Outgoing interface list:
```

https://t.me/learningnets

```
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:06:41/stopped
```

All other routers that received the Auto-RP Discovery messages and populated their Auto-RP cache should also have a (S,G) entry for 224.0.1.40.

```
R10#show ip pim rp mapping

PIM Group-to-RP Mappings
Group(s) 224.0.0.0/4
RP 150.1.5.5 (?), v2v1
    Info source: 150.1.5.5 (?), elected via Auto-RP
         Uptime: 00:10:45, expires: 00:02:08
!
!R10#show ip mroute 224.0.1.40
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 00:11:54/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse-Dense, 00:11:54/stopped
 (150.1.5.5, 224.0.1.40
), 00:11:11/00:02:43, flags: PLTX
  Incoming interface: GigabitEthernet1.108, RPF nbr 155.1.108.8
  Outgoing interface list: Null
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Auto-RP - Multiple Candidate RPs

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-dense-mode multicast delivery on the path between R5 and R10.
- Using a Cisco proprietary technology, configure the following:
  - R8 as RP for group ranges **224.0.0.0–231.255.255.255**
  - R10 as RP for group ranges **232.0.0.0–239.255.255.255**
  - R5 as the Mapping Agent
- If one RP fails, the other one should provide backup for the other's groups.
- The group **224.110.110.110** should be always switched in dense-mode.

Configuration

As discussed in the previous task, Auto-RP MAs amalgamate information learned from multiple candidate RPs and advertise RP discovery messages. You may want to use multiple RPs in cases where you want load-balancing, redundancy, or both:

- If your goal is load balancing, try to configure the group-mapping access-lists so that every RP services a range of groups.
- If your goal is redundancy, make both RPs service the same group ranges. The one with the highest IP address will be selected by the MA.
- If you want to achieve both load-balancing and redundancy, you may map RP1 to a specific group range, say 224.0.0.0-231.255.255.255, and permit 224.0.0.0 15.255.255.255 in the end of the respective ACL. Use the entry to permit the range 232.0.0.0–239.255.255.255 for RP2 along with the entry 224.0.0.0 15.255.255.255 in

https://t.me/learningnets

the end. This will ensure that RP1 and RP2 are used for specific group ranges, while RP1 is used for the rest of the groups when RP2 fails or vice versa. This is based on the longest match selection criteria used by the multicast routers and the fact that for overlapping ranges, the MA will only advertise the RP with the highest IP address.

Notice that the mapping list is compiled by the MA, but still the final RP selection is performed by the multicast router. Longest match criteria is an effective rule to allow for unique RP selection along with providing redundancy.

```
R5:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
!
interface Loopback0
 ip pim sparse-dense-mode
!
ip pim send-rp-discovery Loopback0 scope 10
R8:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
!
interface Loopback0
 ip pim sparse-dense-mode
!
ip access-list standard R8_GROUPS
 deny 224.110.110.110
 permit 224.0.0.0 7.255.255.255
 permit 224.0.0.0 15.255.255.255
!
ip pim send-rp-announce Loopback0 scope 10 group-list R8_GROUPS
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
!
```

https://t.me/learningnets

```
interface Loopback0
 ip pim sparse-dense-mode
!
ip access-list standard R10_GROUPS
 deny 224.110.110.110
 permit 232.0.0.0 7.255.255.255
 permit 224.0.0.0 15.255.255.255
!
ip pim send-rp-announce Loopback0 scope 10 group-list R10_GROUPS
```

Verification

Look at the RP mappings on R5. Notice how the MA elects the best candidate RP for a given range. The group 224.110.110.110 is negatively cached, and thus is always processes in dense-mode.

```
R5#show ip pim rp mapping

PIM Group-to-RP Mappings
This system is an RP-mapping agent

Group(s) 224.0.0.0/5
  RP 150.1.8.8 (?), v2v1    Info source: 150.1.8.8 (?), elected via Auto-RP
         Uptime: 00:00:53, expires: 00:02:05
Group(s) 224.0.0.0/4
  RP 150.1.10.10 (?), v2v1    Info source: 150.1.10.10 (?), elected via Auto-RP
         Uptime: 00:00:43, expires: 00:02:12
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), via Auto-RP
         Uptime: 00:00:53, expires: 00:02:03 Group(s) (-)224.110.110.110/32
  RP 150.1.10.10 (?), v2v1    Info source: 150.1.10.10 (?), elected via Auto-RP
         Uptime: 00:00:43, expires: 00:02:13
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), via Auto-RP
         Uptime: 00:00:53, expires: 00:02:05
Group(s) 232.0.0.0/5
  RP 150.1.10.10 (?), v2v1    Info source: 150.1.10.10 (?), elected via Auto-RP

         Uptime: 00:00:43, expires: 00:02:13
```

Enable PIM Sparse-Dense mode on the GigabitEthernet1.45 interface between R5 and R4 so that R4 gets the Auto-RP information.

https://t.me/learningnets

```
R4:

ip multicast-routing distributed
!
interface GigabitEthernet1.45
 ip pim sparse-dense-mode
R5:

interface GigabitEthernet1.45
 ip pim sparse-dense-mode
```

Check the Auto-RP cache of R4. Notice that it only has a single RP for every range. R10 is elected as the RP for all ranges except 224.0.0.0/5.

```
R4#show ip pim rp mapping

PIM Group-to-RP Mappings
Group(s) 224.0.0.0/5
  RP 150.1.8.8 (?), v2v1
    Info source: 155.1.45.5 (?), elected via Auto-RP
         Uptime: 00:00:20, expires: 00:02:37Group(s) 224.0.0.0/4
  RP 150.1.10.10 (?), v2v1
    Info source: 155.1.45.5 (?), elected via Auto-RP
         Uptime: 00:00:20, expires: 00:02:38Group(s) (-)224.110.110.110/32
  RP 150.1.10.10 (?), v2v1
    Info source: 155.1.45.5 (?), elected via Auto-RP
         Uptime: 00:00:20, expires: 00:02:38Group(s) 232.0.0.0/5

  RP 150.1.10.10 (?), v2v1
    Info source: 155.1.45.5 (?), elected via Auto-RP
         Uptime: 00:00:20, expires: 00:02:38
```

Now check to see whether the group 224.110.110.110 is forwarded using PIM Dense mode. Join 224.110.110.110 on R8's GigabitEthernet1.8 and ping it from R4.

```
R8#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R8(config)#interface GigabitEthernet1.8
R8(config-subif)#ip pim sparse-dense-mode
R8(config-subif)#ip igmp join-group 224.110.110.110
```

https://t.me/learningnets

Notice that the group has no RP in the mroute output, just as it should be for the dense group.

```
R4#ping 224.110.110.110 repeat 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.110.110.110, timeout is 2 seconds:
Reply to request 0 from 155.1.58.8, 66 ms
Reply to request 1 from 155.1.8.8, 40 ms
Reply to request 2 from 155.1.8.8, 8 ms
Reply to request 3 from 155.1.8.8, 47 ms
Reply to request 4 from 155.1.8.8, 13 ms
!
!R5#show ip mroute 224.110.110.110
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 224.110.110.110), 00:00:30/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Sparse-Dense, 00:00:30/stopped
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:00:30/stopped
(155.1.45.4, 224.110.110.110), 00:00:30/00:02:29, flags: T

  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse-Dense, 00:00:30/stopped
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Auto-RP - Filtering Candidate RPs

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-dense-mode multicast delivery on the path between R5 and R10.
- Using a Cisco proprietary technology, configure R8 and R10 as RPs.
  - Configure R5 as the Mapping Agent.
- The group **224.110.110.110** should always be switched in dense-mode.
- Configure R5 to filter announcements for the group **224.110.110.110** received from R10.

Configuration

The Auto-RP mapping agent allows for the filtering of incoming RP announcements sent by RP candidates. This type of filtering applies only to Auto-RP announcements received by listening to the 224.0.1.39 multicast IP address and thus is only effective on Mapping Agents. You configure a filtering statement using the following syntax: `ip pim rp-announce-filter [group-list <access-list> | rp-list <access-list>]` . All access-lists are either numbered or named standard ACLs. RP announcements are inspected, and if a match is found for the group and the RP IP address, the action is taken based on the “permit” or “deny” statement. If you omit the `rp-list` keyword, any announcements containing groups matching the `group-list` are matched. If you omit the `group-list` parameter, all updates from the RPs in `rp-list` are matched.

```
R5:
```

https://t.me/learningnets

```
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
!
interface Loopback0
 ip pim sparse-dense-mode
!
ip access-list standard RP_LIST
 permit 150.1.10.10
!
ip access-list standard GROUP_LIST
 deny 224.110.110.110
 permit any
!
ip pim send-rp-discovery Loopback0 scope 10
ip pim rp-announce-filter rp-list RP_LIST group-list GROUP_LIST
```

R8:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-dense-mode
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
!
interface Loopback0
 ip pim sparse-dense-mode
!
ip access-list standard R8_GROUPS
 deny 224.110.110.110
 permit any
!
ip pim send-rp-announce Loopback0 scope 10 group-list R8_GROUPS
```

R10:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-dense-mode
!
interface Loopback 0
 ip pim sparse-dense-mode
!
ip access-list standard R10_GROUPS
 deny 224.110.110.110
```

https://t.me/learningnets

```
 permit any
!
ip pim send-rp-announce Loopback0 scope 10 group-list R10_GROUPS
```

Verification

Use the command **`debug ip pim auto-rp`** on the MA to monitor the updates that are filtered.

```
R5#debug ip pim auto-rp

!
!
Auto-RP(0):  Build mapping (224.0.0.0/4, RP:150.1.10.10), PIMv2 v1, Auto-RP(0):
Build mapping (-224.110.110.110/32, RP:150.1.10.10), PIMv2 v1.
Auto-RP(0): Send RP-discovery packet of length 54 on GigabitEthernet1.58 (1 RP entries)
Auto-RP(0): Received RP-announce packet of length 54, from 150.1.10.10, RP_cnt 1, ht 10
Auto-RP(0): Filtered -224.110.110.110/32 for RP 150.1.10.10

!
!
Auto-RP(0): Mapping (224.110.110.110/32, RP:150.1.10.10) expired,
Auto-RP(0): Build RP-Discovery packet
Auto-RP(0):  Build mapping (224.0.0.0/4, RP:150.1.10.10), PIMv2 v1, Auto-RP(0):
Build mapping (-224.110.110.110/32, RP:150.1.8.8), PIMv2 v1.
Auto-RP(0): Send RP-discovery packet of length 60 on GigabitEthernet1.58 (2 RP entries)
Auto-RP(0): Received RP-announce packet of length 54, from 150.1.8.8, RP_cnt 1, ht 10
(0): pim_add_prm:: 224.110.110.110/255.255.255.255, rp=150.1.8.8, repl = 0, ver =3
```

Notice that the update from R10 for 224.110.110.110 is filtered, but R8's is not. Verify that the Auto-RP cache on the MA does not have the entry for 224.110.110.110 mapped to the RP-R10.

```
R5#show ip pim rp mapping

PIM Group-to-RP Mappings
This system is an RP-mapping agent

Group(s) 224.0.0.0/4
  RP 150.1.10.10 (?), v2v1
    Info source: 150.1.10.10 (?), elected via Auto-RP
         Uptime: 00:06:41, expires: 00:00:08
  RP 150.1.8.8 (?), v2v1
```

https://t.me/learningnets

```
    Info source: 150.1.8.8 (?), via Auto-RP
         Uptime: 00:06:50, expires: 00:00:07Group(s) (-)224.110.110.110/32
RP 150.1.8.8 (?), v2v1

    Info source: 150.1.8.8 (?), elected via Auto-RP
         Uptime: 00:06:50, expires: 00:00:07
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Auto-RP Listener

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R4 and R10.
- Using a Cisco proprietary technology, configure R8 and R10 as RPs.
  - Configure R5 as the Mapping Agent.
- Ensure that all devices still hear Auto-RP announcements.

Configuration

Auto-RP listener is another solution designed to allow using Auto-RP without risking any groups falling back to dense mode forwarding. This feature works in tandem with PIM sparse mode enabled on all interfaces (not PIM SM/DM). However, two Auto-RP multicast groups 224.0.1.39 and 224.0.1.40 are flooded in dense mode. No other groups are allowed to use dense mode and thus dangerous flooding fallback is eliminated.

```
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
ip pim autorp listener
R5:
```

https://t.me/learningnets

```
ip multicast-routing distributed
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim send-rp-discovery Loopback0 scope 10
ip pim autorp listener
```

R8:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim send-rp-announce Loopback0 scope 10
ip pim autorp listener
```

R10:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim send-rp-announce Loopback0 scope 10
ip pim autorp listener
```

Verification

https://t.me/learningnets

Make sure that all transit interfaces are configured for PIM SM. Repeat the following operation on all routers, and pay attention to the Ver/Mode field.

```
R5#show ip pim interface


Address          Interface                Ver/   Nbr    Query  DR     DR
                                          Mode   Count  Intvl  Prior
155.1.45.5       GigabitEthernet1.45v2/S
   1      30     1      155.1.45.5155.1.58.5       GigabitEthernet1.58v2/S
   1      30     1      155.1.58.8150.1.5.5        Loopback0v2/S
   0      30     1      150.1.5.5
!
!R5#show ip pim autorp
AutoRP Information:
  AutoRP is enabled.
  RP Discovery packet MTU is 1472.
  224.0.1.40 is joined on Loopback0. AutoRP groups over sparse mode interface is enabled



PIM AutoRP Statistics: Sent/Received
  RP Announce: 0/0, RP Discovery: 0/0
```

Ensure that all routers are still capable of learning Auto-RP information.

```
R4#show ip pim rp mapping


PIM Group-to-RP Mappings
Group(s) 224.0.0.0/4
RP 150.1.10.10 (?), v2v1
Info source: 150.1.5.5 (?), elected via Auto-RP


         Uptime: 00:02:04, expires: 00:02:51
```

Note that auto-rp listener does not change the behavior of Auto-RP, nor does it modify how both Auto-RP groups are flooded. This feature allows us to run the interfaces in sparse-mode. R5, the mapping-agent, still learns about both RPs and selects the best one.

```
R5#show ip pim rp mapping


PIM Group-to-RP Mappings
This system is an RP-mapping agent
```

https://t.me/learningnets

```
Group(s) 224.0.0.0/4
  RP 150.1.10.10 (?), v2v1
    Info source: 150.1.10.10 (?), elected via Auto-RP
         Uptime: 00:04:59, expires: 00:01:59
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), via Auto-RP
         Uptime: 00:05:06, expires: 00:02:53
```

Check that the Auto-RP groups (224.0.1.39 and 224.0.1.40) are still forwarded without any RP information. Observe that the output shows Forward/Sparse, but the actual forwarding uses dense mode.

```
R4#show ip mroute 224.0.1.40
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 00:06:16/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Sparse, 00:06:16/stopped
 (155.1.45.5, 224.0.1.40), 00:05:59/00:02:07, flags: PLT

  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.5
  Outgoing interface list: Null
```

Join 224.10.10.10 on R10 and ping it from R4 to ensure that multicast delivery is operational.

```
R10(config)#interface GigabitEthernet1.10
R10(config-subif)#ip pim sparse-mode
R10(config-subif)#ip igmp join-group 224.10.10.10
R10(config-subif)#end
```

https://t.me/learningnets

```
!
!R4#ping 224.10.10.10 rep 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.10.10.10, timeout is 2 seconds:
Reply to request 0 from 155.1.10.10, 55 ms
Reply to request 1 from 155.1.10.10, 23 ms
Reply to request 1 from 155.1.108.10, 23 ms
Reply to request 2 from 155.1.10.10, 38 ms
Reply to request 3 from 155.1.10.10, 5 ms
Reply to request 4 from 155.1.10.10, 7 ms
Reply to request 5 from 155.1.10.10, 32 ms
!
!R5#show ip mroute 224.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.10.10.10), 00:00:35/stopped, RP 150.1.10.10, flags: SPF
  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.8
  Outgoing interface list: Null
(155.1.45.4, 224.10.10.10), 00:00:35/00:02:58, flags: FT

  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:00:35/00:02:54
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Auto-RP and RP/MA Placement

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R3, R7, and R9.
- Enable PIM sparse-mode on the DMVPN connection between R2, R3, and R5.
- Using a Cisco proprietary technology, configure R7 the RP.
  - Configure R2 as the Mapping Agent.
- Provide a solution that allows R2 and R3 to properly hear Auto-RP messages.

Configuration

As discussed previously, PIM by default treats NBMA interfaces as if they were broadcast-capable. That is, it assumes that all neighbors on a WAN cloud can hear multicast packets sent by any neighbor. However, in non-fully meshed topologies such as hub-and-spoke networks, this is not the case. When a spoke sends multicast traffic over the NBMA segment, only the hub can hear it. The hub will not forward multicast back to other spokes, based on the split-horizon rule. This problem could be solved using PIM NBMA mode, but this only works with PIM Sparse mode.

Now the real issue is that the Auto-RP uses dense mode for RP information dissemination, so the PIM NBMA solution will not work. If a candidate RP or a Mapping Agent is placed behind the spoke node, RP information could be lost. Imagine that an MA is located behind the spoke node. Then any Auto-RP discovery messages it sends will only reach the hub node, but not the other spokes. This

https://t.me/learningnets

could also be the case, if the RP and the MA are both placed behind NBMA spokes. Therefore, when designing your multicast network, take care and place the MA behind the hub. RPs could still be located at spokes, as long as announcements can reach the hub.

Other solutions include the use of sub-interfaces on the hub router or creating tunnels between the hub and the spokes. Both solutions effectively break the split-horizon rule by receiving and sending multicast packets on different interfaces. Notice that you will need static mroutes if you don't run an IGP across the tunnel interfaces.

```
R2:
ip multicast-routing distributed
!
interface Loopback0
 ip pim sparse-mode
!
interface Tunnel0
 ip pim sparse-mode
!
interface Tunnel100
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 150.1.3.3
!
router ospf 1
 passive-interface Tunnel100
!
ip pim autorp listener
ip pim send-rp-discovery Loopback0 scope 10
ip mroute 150.1.7.7 255.255.255.255 Tunnel100
R3:
ip multicast-routing distributed
!
interface Tunnel0
 ip pim sparse-mode
!
interface Tunnel100
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 150.1.2.2
!
router ospf 1
```

https://t.me/learningnets

```
 passive-interface Tunnel100
!
interface GigabitEthernet1.37
 ip pim sparse-mode
!
ip pim autorp listener
ip mroute 150.1.2.2 255.255.255.255 Tunnel100
R5:
ip multicast-routing distributed
!
interface Tunnel0
 ip pim sparse-mode
R7:
ip multicast-routing distributed
!
interface GigabitEthernet1.37
 ip pim sparse-mode
!
interface GigabitEthernet1.79
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim send-rp-announce Loopback0 scope 10
ip pim autorp listener
R9:

ip multicast-routing distributed
!
interface GigabitEthernet1.79
 ip pim sparse-mode
!
ip pim autorp listener
```

Verification

First, make sure that R2 is fully functional as an MA. It should be receiving announcements from R7. Check the multicast routing table for the group 224.0.1.39.

```
R2#show ip mroute 224.0.1.39
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
```

https://t.me/learningnets

```
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.39), 00:25:08/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:25:08/stopped
    Tunnel0, Forward/Sparse, 00:25:08/stopped
    Tunnel100, Forward/Sparse, 00:23:40/stopped
 (150.1.7.7, 224.0.1.39
), 00:23:19/00:02:55, flags: LT Incoming interface: Tunnel100, RPF nbr 150.1.3.3, Mroute

  Outgoing interface list:
    Tunnel0, Forward/Sparse, 00:23:19/stopped
    Loopback0, Forward/Sparse, 00:23:19/stopped
```

Without the additional Tunnel 100 and the static mroute, R2 would not be able to hear these messages from R7. Not only do we have the issue the NBMA interface described above, but also R2 does not use the DMVPN interface to reach R7. This causes an RPF failure, and thus a static mroute is needed pointing out of the new Tunnel 100 interface. These two actions allow R2 to properly receive messages from R7 and punt them to the Auto-RP process.

Now that R2 receives the announcements from R7, R2 needs to send out the Auto-RP discovery messages to the rest of the PIM routers. Here we encounter the NBMA issue described above again, where R2 sends the discovery messages to R5 via the DMVPN Tunnel0 interface, but R5 cannot send them back out of the DMVPN Tunnel0 so that R3 and the rest of the PIM network can hear them. The solution is to use the new Tunnel 100 interface between R2 and R3, but the RPF issue must also be resolved. R3 needs to be able to receive the discover messages from R2 via the new Tunnel 100, so a static mroute is used here also. Check the mroute table of R3 for 224.0.1.40 and notice the incoming interface and RPF neighbor.

https://t.me/learningnets

```
R3#show ip mroute 224.0.1.40

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 00:37:50/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel0, Forward/Sparse, 00:35:59/stopped
    Tunnel100, Forward/Sparse, 00:33:30/stopped
    GigabitEthernet1.37, Forward/Sparse, 00:37:50/stopped
 (150.1.2.2, 224.0.1.40
), 00:22:43/00:02:15, flags: LT Incoming interface: Tunnel100, RPF nbr 150.1.2.2, Mroute

  Outgoing interface list:
    Tunnel0, Prune/Sparse, 00:19:38/00:02:15, A
GigabitEthernet1.37, Forward/Sparse, 00:22:43/stopped
```

To further verify the RPF details, use the following commands on R2 or R3. Note that R3 only needs to point out of the Tunnel 100 interface for R2's loopback0, the source of the discover messages. Also, R2 only needs to resolve R7's loopback, the source of the RP announcement messages, out of the Tunnel 100 interface. This is why exact static mroutes were used on each device.

```
R3#show ip rpf 150.1.2.2

RPF information for ? (150.1.2.2)  RPF interface: Tunnel100
  RPF neighbor: ? (150.1.2.2)
  RPF route/mask: 150.1.2.2/32  RPF type: multicast (static
)
  Doing distance-preferred lookups across tables
```

https://t.me/learningnets

```
  RPF topology: ipv4 multicast base
!
!R3#show ip route multicast 150.1.2.2



Routing Table: multicast
Routing entry for 150.1.2.2/32  Known via "static
", distance 1, metric 0
  Routing Descriptor Blocks:  *directly connected, via Tunnel100
      Route metric is 0, traffic share count is 1
!
!R3#show ip mfib 150.1.2.2 224.0.1.40

Entry Flags:    C - Directly Connected, S - Signal, IA - Inherit A flag,
                ET - Data Rate Exceeds Threshold, K - Keepalive
                DDE - Data Driven Event, HW - Hardware Installed
I/O Item Flags: IC - Internal Copy, NP - Not platform switched,
                NS - Negate Signalling, SP - Signal Present,
                A - Accept, F - Forward, RA - MRIB Accept, RF - MRIB Forward,
                MA - MFIB Accept
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kbits per second
Other counts:      Total/RPF failed/Other drops
I/O Item Counts:   FS Pkt Count/PS Pkt Count
Default  (150.1.2.2,224.0.1.40
) Flags: HW
   SW Forwarding: 0/0/0/0, Other: 0/0/0
   HW Forwarding:   0/0/0/0, Other: 0/0/0 Tunnel100
 Flags: A
   GigabitEthernet1.37 Flags: F IC NS
     Pkts: 0/0
```

Check the RP mapping tables on all PIM routers and make sure that they all have proper entries for the RP. This step also verifies that the Auto-RP messages are getting properly announced throughout the network.

```
R2#show ip pim rp mapping

PIM Group-to-RP Mappings
This system is an RP-mapping agent (Loopback0)

Group(s) 224.0.0.0/4
  RP 150.1.7.7 (?), v2v1
    Info source: 150.1.7.7 (?), elected via Auto-RP
     Uptime: 00:36:36, expires: 00:00:14
!
```

https://t.me/learningnets

```
!R3#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 150.1.7.7 (?), v2v1
    Info source: 150.1.2.2 (?), elected via Auto-RP
         Uptime: 00:23:26, expires: 00:02:24
!
!R5#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 150.1.7.7 (?), v2v1
    Info source: 150.1.2.2 (?), elected via Auto-RP
         Uptime: 00:34:52, expires: 00:02:54
!
!R7#show ip pim rp mapping

PIM Group-to-RP Mappings
This system is an RP (Auto-RP)

Group(s) 224.0.0.0/4
  RP 150.1.7.7 (?), v2v1
    Info source: 150.1.2.2 (?), elected via Auto-RP
     Uptime: 00:24:37, expires: 00:02:16
!
!R9#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 150.1.7.7 (?), v2v1
    Info source: 150.1.2.2 (?), elected via Auto-RP
         Uptime: 00:25:07, expires: 00:02:45
```

Join group 229.9.9.9 from R9 and send traffic to it from R3.

```
R9:
interface GigabitEthernet1.79
 ip igmp join-group 229.9.9.9

R3#ping 229.9.9.9 rep 100
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 229.9.9.9, timeout is 2 seconds:
Reply to request 0 from 155.1.79.9, 36 ms
Reply to request 0 from 155.1.79.9, 48 ms

Reply to request 0 from 155.1.79.9, 36 ms
Reply to request 1 from 155.1.79.9, 4 ms
Reply to request 1 from 155.1.79.9, 5 ms
Reply to request 1 from 155.1.79.9, 5 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Filtering Auto-RP Messages

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10.
- Using a Cisco proprietary technology, configure R5 as the RP and R6 as the Mapping Agent.
- Configure R6 so that R10 cannot hear Auto-RP discovery messages.

Configuration

Auto-RP uses special dense-mode groups to disseminate RP information, and the information is flooded across the multicast domain. The only way to control the scope of the information is by setting the TTL to the value that roughly represents the diameter of the multicast domain. However, the problem is that this does not enforce strict administrative limits on the information scope, and per-link flooding is not controlled. The other way to filter Auto-RP announcements is to use the `ip multicast boundary` command, discussed in a separate task.

```
R4:
ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface GigabitEthernet1.45
```

https://t.me/learningnets

```
 ip pim sparse-mode
```

R6:

```
ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim send-rp-discovery Loopback0 scope 3
```

R5:

```
ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim send-rp-announce Loopback0 scope 10
```

R8:

```
ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
```

R10:

```

ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.108
 ip pim sparse-mode
```

https://t.me/learningnets

Verification

Ensure that R8 still has entries in the Auto-RP cache and that R10 does not. Note that it may take as long as three minutes for old mappings to expire from the RP mapping table. You may clear them by using `clear ip pim rp-mapping` .

```
R10#show ip pim rp mapping
PIM Group-to-RP Mappings
R10#
!
!R8#show ip pim rp mapping
PIM Group-to-RP Mappings
Group(s) 224.0.0.0/4
RP 150.1.5.5 (?), v2v1

    Info source: 150.1.6.6 (?), elected via Auto-RP
         Uptime: 00:09:40, expires: 00:00:27
```

R10's mroute table does not have (S,G) state for 224.0.1.40 from 150.1.6.6.

https://t.me/learningnets

```
R10#show ip mroute 224.0.1.40

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 00:14:25/00:02:37, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse, 00:14:25/stopped
```

To inspect the TTL of the packets received on R8 from R6, we can use Flexible Netflow. We can see from this output that the packets from R6 sent to 224.0.1.40 are arriving to R8 with a TTL of 1.

```
R8#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R8(config)#flow record MAPPING_AGENT
R8(config-flow-record)#match ipv4 protocol
R8(config-flow-record)#match ipv4 source address
R8(config-flow-record)#match ipv4 destination address
R8(config-flow-record)#collect ipv4 ttl
R8(config-flow-record)#collect interface input
R8(config-flow-record)#exit
R8(config)#flow monitor MAPPING_AGENT
R8(config-flow-monitor)#record MAPPING_AGENT
R8(config-flow-monitor)#exit
R8(config)#interface GigabitEthernet1.58
R8(config-subif)#ip flow monitor MAPPING_AGENT input
!
!R8#show flow monitor MAPPING_AGENT cache format table
```

https://t.me/learningnets

```
  Cache type:                                Normal (Platform cache)
  Cache size:                                200000
  Current entries:                                4
  High Watermark:                                 6

  Flows added:                                   19
  Flows aged:                                    15
    - Inactive timeout    (    15 secs)         15

IPV4 SRC ADDR    IPV4 DST ADDR    IP PROT  intf input            ip ttl
===============  ===============  =======  ====================  ======
155.1.58.5       224.0.0.5             89  Gi1.58                     1
155.1.58.5       224.0.0.13           103  Gi1.58                     1
155.1.58.5       224.0.0.10            88  Gi1.58                     1
150.1.6.6        224.0.1.40            17  Gi1.58                     1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Multicast Boundary

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10.
- Using a Cisco proprietary technology, configure the following:
  - R8 as RP for group ranges **224.0.0.0–231.255.255.255**
  - R10 as RP for group ranges **232.0.0.0–239.255.255.255**
  - R6 as the Mapping Agent
- If one RP fails, the other one should provide backup for the other's groups.
- The group **224.110.110.110** should always be switched in dense-mode.
- Configure R5's connection to VLAN 58 (GigabitEthernet1.58) so that traffic to the group range **232.0.0.0/5** cannot reach R8.
  - Filter the Auto-RP messages to remove the information about this group range.

Configuration

https://t.me/learningnets

The multicast boundary feature allows for setting administrative borders for multicast traffic. This feature applies filtering to both the control plane traffic (IGMP, PIM, AutoRP) and the data plane (installing multicast route states out of the configured interface). It is much more flexible than using Auto-RP TTL scoping and allows the application of finer and more granular access control. Using this feature, you may contain multicast traffic within the boundaries of your administrative domain, without relying on TTL-based filtering.

When you apply the command `ip multicast boundary <access-list> [filter-autorp]` to an interface, the following filtering rules apply:

- If the access-list is a standard ACL, any ingress IGMP or PIM messages are inspected to see if the group being joined or tree being built has a match in the access-list. This might be an (S,G) or (*,G) join. Additionally, the interface is used as an outgoing interface to forward a group “G” only if the group matches the access-list.
- If the access-list is an extended ACL, it specifies both multicast sources and groups, using the format `permit ip <src-ip> <src-wildcard> <group-address> <group-mask>` . Any incoming PIM/IGMP messages are inspected, and if both the source and group are matched, they are permitted. At the same time, the interface could be used as outgoing for multicast traffic sourced off the IP addresses matching the extended access-list with the group matching the same access-list entry. If you want to match only (*,G) shared tree signaling, specify the source IP address of 0.0.0.0; this will affect PIM Join/Prune messages.

Remember that unicast PIM Register messages are not affected by the multicast-boundary configuration and must be filtered using the respective feature. If you have specified the `filter-autorp` keyword, the router will inspect any Auto-RP messages (announces or discovery) and filter away those not matching the access-list. Note that the access-list must be a *standard* ACL if you use Auto-RP filtering. For the Auto-RP group range to be permitted, the whole range must be covered by permit statements in the access-list. If any part of the range’s group is not permitted, the whole range is removed from the advertisement.

Starting with version 12.3(17)T, you may use `in` or `out` options with the multicast boundary command (this does not work with the `filter-autorp` option, though). When applied as an ingress filter, the command affects control plane traffic, IGMP/PIM Joins, and Auto-RP messages. When configured as an egress filter, it will control the interface being added to the OIL for multicast groups, allowing only the groups permitted by the access-list.

```
R4:
ip multicast-routing distributed
ip pim autorp listener
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
```

R6:

```
ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim send-rp-discovery Loopback0 scope 10
```

R5:

```
ip multicast-routing distributed
ip pim autorp listener
!
ip access-list standard PERMITTED_GROUPS
 deny 232.0.0.0 7.255.255.255
 permit any
!
interface GigabitEthernet1.58
 ip pim sparse-mode
 ip multicast boundary PERMITTED_GROUPS filter-autorp
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
```

R8:

```
ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
```

https://t.me/learningnets

```
!
ip access-list standard R8_GROUPS
 deny 224.110.110.110
 permit 224.0.0.0 7.255.255.255
 permit 224.0.0.0 15.255.255.255
!
ip pim send-rp-announce Loopback0 scope 10 group-list R8_GROUPS
R10:

ip multicast-routing distributed
ip pim autorp listener
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip access-list standard R10_GROUPS
 deny 224.110.110.110
 permit 232.0.0.0 7.255.255.255
 permit 224.0.0.0 15.255.255.255
!
ip pim send-rp-announce Loopback0 scope 10 group-list R10_GROUPS
```

Verification

Notice the following console message on R5. Because the range 224.0.0.0–239.255.255.255 advertised by both R8 and R10 *overlaps* with the denied group range, it is also filtered from Auto-RP announcements in addition to the explicitly denied range. This results in the following RP to group mapping in the MA. (Again, note that it may take a few moments for old mappings to expire from the RP Mapping list.)

```
R5#
%AUTORP-4-OVERLAP: AutoRP Announcement packet,
group 224.0.0.0 with mask 240.0.0.0 removed because of multicast boundary for 232.0.0.0 with mask 248.0.0.0
```

Look at the mappings on the Mapping Agent before applying the filter. Note that all of the group to RP mappings are present.

https://t.me/learningnets

```
R6#show ip pim rp mapping

PIM Group-to-RP Mappings
This system is an RP-mapping agent (Loopback0)

Group(s) 224.0.0.0/5
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), elected via Auto-RP
         Uptime: 00:00:39, expires: 00:00:11
Group(s) 224.0.0.0/4
  RP 150.1.10.10 (?), v2v1
    Info source: 150.1.10.10 (?), elected via Auto-RP
         Uptime: 00:00:40, expires: 00:00:14
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), via Auto-RP
         Uptime: 00:00:39, expires: 00:00:11
Group(s) (-)224.110.110.110/32
  RP 150.1.10.10 (?), v2v1
    Info source: 150.1.10.10 (?), elected via Auto-RP
         Uptime: 00:00:40, expires: 00:00:14
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), via Auto-RP
         Uptime: 00:00:39, expires: 00:00:10
Group(s) 232.0.0.0/5
  RP 150.1.10.10 (?), v2v1
    Info source: 150.1.10.10 (?), elected via Auto-RP
     Uptime: 00:00:40, expires: 00:00:15
```

Apply the filter and look at the cache again.

```
R6#show ip pim rp mapping

This system is an RP-mapping agent (Loopback0)
Group(s) 224.0.0.0/5

  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), elected via Auto-RP
         Uptime: 00:05:17, expires: 00:00:13
Group(s) (-)224.110.110.110/32
  RP 150.1.10.10 (?), v2v1
    Info source: 150.1.10.10 (?), elected via Auto-RP
         Uptime: 00:05:18, expires: 00:00:12
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.8.8 (?), via Auto-RP
         Uptime: 00:05:17, expires: 00:00:13
```

https://t.me/learningnets

```
          Uptime: 01:34:10, expires: 00:02:46
```

Based on the above output, R8 won't even be able to join the shared trees for the denied groups, because it does not know the RP for those groups.

```
R8#show ip pim rp mapping


PIM Group-to-RP Mappings
This system is an RP (Auto-RP)


Group(s) 224.0.0.0/5
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.6.6 (?), elected via Auto-RP
         Uptime: 00:17:56, expires: 00:00:13
Group(s) (-)224.110.110.110/32
  RP 150.1.10.10 (?), v2v1
    Info source: 150.1.6.6 (?), elected via Auto-RP
         Uptime: 00:17:57, expires: 00:00:13
```

Recall that even though R8 is an RP, it needs to learn about itself via the Mapping Agent's Discover Messages. These messages are being filtered by R5.

```
R8#show ip pim rp-hash 224.1.1.1
  RP 150.1.8.8 (?), v2v1
    Info source: 150.1.6.6 (?), elected via Auto-RP
         Uptime: 00:20:15, expires: 00:00:11
  PIMv2 Hash Value (mask 0.0.0.0)
    RP 150.1.8.8, via Auto-RP
!
!R8#show ip pim rp-hash 238.8.8.8
No RP available for this group
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

PIM Bootstrap Router

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R4 and R10.
  - Do not enable PIM over the DMVPN network between R4 and R5.
- Configure R5 to advertise itself as the RP for all multicast groups using the standards-based protocol.

Configuration

Bootstrap router, or BSR, is a standards-based solution available with PIMv2 that performs the same function as Auto-RP; that is, it disseminates RP information. Both protocols use the concept of a candidate RP. However, BSR does not use any dense-mode groups to distribute RP-to-group mapping information. Instead, the information is flooded using PIM messages, on a hop-by-hop basis. That is, when a router receives a candidate RP announcement inside a PIM message, it applies an RPF check, validating that the announcement was received on the interface that is on the shortest path to the RP. If the RPF check succeeds, the message is flooded out of all PIM-enabled interfaces.

To configure a candidate RP, use the command `ip pim rp-candidate <PIM-Enabled-Interface> [group-list <Standard-ACL>] [interval <Seconds>] [priority <0-255>]` . If you omit all arguments, the router will start advertising itself as the RP for all groups. You may specify a list of groups using the `group-list` argument. All entries in this list are “positive”, if you compare that to Auto-RP, and you cannot use “negative”

https://t.me/learningnets

groups. The priority value is used when the routers select the best RP for a given group, and lower values are preferred. The default priority is zero (which is against the standard, which specifies a value of 192), which is the highest possible value. You would very rarely want to change the priority value for a candidate RP, possibly only in cases when you want to gracefully take the RP out of service.

Where BSR differs from Auto-RP is the bootstrap router itself. This router performs a role similar to the Auto-RP MA, by listening to candidate RP announcements. However, unlike the Auto-RP MA, BSR does not elect the best RP for every group range. Instead, for every group range it builds a set of candidate RPs, including all routers that advertised their willingness to service this group range. This is called the *group range to RP set mapping*.

The resulting array of group range to RP set mappings is distributed by the BSR using PIM messages and the same flooding procedure described above. The command to configure a router as a BSR is **`ip pim bsr-candidate <Interface-Name> [hash-mask-length] [priority]`**. Ignore the **`hash-mask-length`** parameter for the moment, and notice the priority field. By default, the priority of zero is advertised in all BSR messages. The higher the priority value, the more preferred the BSR. The IP address of the interface used to source the BSR messages is used as a tie-breaker; if two priorities match, the higher IP is preferred. If there are multiple BSRs, they all listen to other potential BSR messages. If a BSR hears a message with a higher priority or IP address, it immediately stops its own BSR advertisements. This process ensures a unique BSR in the domain while maintaining some redundancy.

The bootstrap messages are received by every multicast router and used to populate their RP cache. Note that it’s up to the routers to select the best matching RP from the sets advertised by the BSR router. To facilitate RP load-balancing, routers may use a special hash function to select the best RP from a set that services the same group range.

```
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.45
 ip pim sparse-mode
R5:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
```

https://t.me/learningnets

```
interface Loopback0
 ip pim sparse-mode
!
ip pim rp-candidate Loopback0
ip pim bsr-candidate Loopback0
R8:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-mode
```

Verification

Enable PIM BSR debugging on R5 to see how BSR messages are originated. Notice that R5 sends both candidate RP and BSR router messages. After that, check the RP mappings on all multicast routers to ensure that they all received the BSR information.

```
R5#debug ip pim bsr

PIM-BSR debugging is onPIM-BSR(0): RP-set for 224.0.0.0/4
PIM-BSR(0):   RP(1) 150.1.5.5, holdtime 150 sec priority 0
PIM-BSR(0): Bootstrap message for 150.1.5.5 originated
PIM-BSR(0): Build v2 Candidate-RP advertisement for 150.1.5.5 priority 0, holdtime 150
PIM-BSR(0):  Candidate RP's group prefix 224.0.0.0/4
PIM-BSR(0): Send Candidate RP Advertisement to 150.1.5.5
PIM-BSR(0):  RP 150.1.5.5, 1 Group Prefixes, Priority 0, Holdtime 150
!
!R8#show ip pim bsr-router

PIMv2 Bootstrap information
  BSR address: 150.1.5.5 (?)
  Uptime:      00:07:04, BSR Priority: 0, Hash mask length: 0
  Expires:     00:01:09
!
```

https://t.me/learningnets

```
!R8#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 150.1.5.5 (?), v2
    Info source: 150.1.5.5 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:07:28, expires: 00:02:04
!
!R4#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 150.1.5.5 (?), v2
    Info source: 150.1.5.5 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:08:47, expires: 00:01:46
```

Always remember that BSR messages are subject to RPF checks. If your router does not receive BSR information, enable the command **`debug ip pim bsr`** to determine whether the locally received BSR packets are dropped because of RPF checks. For example, *temporarily* configure a static route on R4 for R5's Loopbac 0 to make R4 think it's reachable via **R1**.

Note the RPF interface before and after applying the static route.

https://t.me/learningnets

```
R4#show ip rpf 150.1.5.5

RPF information for ? (150.1.5.5) RPF interface: GigabitEthernet1.45
 RPF neighbor: ? (155.1.45.5)
 RPF route/mask: 150.1.5.5/32
   RPF type: unicast ( eigrp 100
)
  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base, originated from ipv4 unicast base

R4:

ip route 150.1.5.5 255.255.255.255 155.1.146.1

R4#show ip rpf 150.1.5.5
 failed, no route exists
```

The reason why this output shows "failed, no route exists" is because we don't have a PIM neighbor on the GigabitEthernet1.146 interface. If we enable PIM on this interface and also enable PIM on either R1 or R6's GigabitEthernet1.146 interface, we would see an RPF entry.

```
R1 or R6:
interface GigabitEthernet1.146
 ip pim sparse-mode
R4:

interface GigabitEthernet1.146
 ip pim sparse-mode
```

Now when we look at the output again, we see that R4 has an RPF entry.

https://t.me/learningnets

```
R4#show ip rpf 150.1.5.5

RPF information for ? (150.1.5.5) RPF interface: GigabitEthernet1.146
RPF neighbor: ? (155.1.146.1)
RPF route/mask: 150.1.5.5/32
  RPF type: unicast ( static
)
  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

However, in this scenario, we have given R4 a "false" RPF entry. R5 is receiving the BSR messages from R5 on its GigabitEthernet1.45 interface, not its GigabitEthernet1.146 interface. Enabling a PIM adjacency between R4 and R1 will not fix this RPF failure because of this. Enabling PIM on this interface was just done to illustrate why R4 saw the "failed, no route exists" message when looking at the RPF entry.

```
R4#debug ip pim bsr
PIM-BSR debugging is on
!
!PIM-BSR(0): bootstrap (150.1.5.5)
on non-RPF path GigabitEthernet1.45 (expected GigabitEthernet1.146) or from non-RPF neighbor 155.1.45.5 (expected 15
!
!R4#show ip pim rp mapping

PIM Group-to-RP Mappings
```

Another method of determining what is feeding the RPF entry is to use the following command. The unicast routing table is "replicated" to another data structure in IOS and used to feed the RPF table.

```
R4#show ip route multicast 150.1.5.5

Routing Table: multicast
Routing entry for 150.1.5.5/32 Known via "static", distance 1, metric 0,
replicated from topology(default)

  Routing Descriptor Blocks:
  * 155.1.146.1 (default)
      Route metric is 0, traffic share count is 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

BSR - Multiple RP Candidates

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R4 and R10.
  - Do not enable PIM over the DMVPN network between R4 and R5.
- Configure R8 and R10 to advertise themselves as RP candidates for all multicast groups using the standards-based protocol.
- R5 should distribute this information and instruct all routers to load-balance multicast groups between the two RPs.
  - Use the maximum possible hash mask length to evenly distribute the load across the RPs.

Configuration

As mentioned in the previous task, BSR protocol distributes multicast group ranges along with the RP-Set information. This allows the multicast routers to effectively load-balance among multiple RPs using a special procedure. This procedure is deterministic, and it ensures that for a given group ALL multicast routers will select the same RP. This is needed to make sure a source for a given group will not register to an RP that is not selected for this group. Here is how the procedure works:

Input: Group Address (G), RP-Set (R1, R2… Rn), Mask (distributed by BSR).

1. Among the routers in the RP-Set, select those that have the numerically

https://t.me/learningnets

lowest priority values. By default, all cRPs advertise a priority of zero, so they all are eligible. You may adjust the priority and take some of the RPs out of service gracefully.

2. For every RP IP address, calculate the hash function value:

   value1 = Hash(G&Mask, R1), value2 = Hash(G&Mask, R2) … valueN = Hash(G&Mask,Rn).

   Notice that the Group IP address is ANDed with the Mask value. The mask value is propagated by the BSR using the **`hash-mask-length`** parameter. Thus, only the first **`hash-mask-length`** bits of the Group are used to calculate the hash value. The default value is zero—that is, the group IP address is ignored when computing the hash value and all groups map to the same RP.

3. Select the RP with the highest hash function value for a given group. If the values are the same, select the RP with the highest IP address.

Using this “pseudo-random” selection procedure, the whole multicast address space is partitioned among different RPs. Every RP will get approximately 2\^[32-hash-mask-length] groups assigned, provided that there are enough RPs to evenly distribute the load. BSR protocol implements an automatic failover procedure. If any one of the RPs were to fail, the BSR would exempt it from bootstrap messages and automatic failover would occur. The failover delay is based on the RP/BSR advertisement intervals. Of course, if there are redundant BSRs, and the primary fails, the secondary would revive and take its role. In this task, odd/even groups are distributed among two RPs, both covering the same group ranges, based on the hash mask length of 31 bits. This ensures that the load is evenly distributed among the two RPs.

```
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.45
 ip pim sparse-mode
R5:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
```

https://t.me/learningnets

```
!
interface Loopback0
 ip pim sparse-mode
!
ip pim bsr-candidate Loopback0 31
R8:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim rp-candidate Loopback0
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim rp-candidate Loopback0
```

Verification

You can quickly find which RP will be selected for a given group by using the show **`ip pim rp-hash`** command. For example:

```
R4#show ip pim rp-hash 239.1.1.1
 RP 150.1.10.10 (?), v2
   Info source: 150.1.5.5 (?), via bootstrap, priority 0, holdtime 150
        Uptime: 00:01:06, expires: 00:02:18 PIMv2 Hash Value (mask 255.255.255.254
)   RP 150.1.10.10, via bootstrap, priority 0,hash value 989207280
```

https://t.me/learningnets

```
   RP 150.1.8.8, via bootstrap, priority 0, hash value 718054422
!
!R4#show ip pim rp-hash 239.1.1.2
RP 150.1.8.8 (?), v2
  Info source: 150.1.5.5 (?), via bootstrap, priority 0, holdtime 150
       Uptime: 00:02:19, expires: 00:02:10 PIMv2 Hash Value (mask 255.255.255.254
)
  RP 150.1.10.10, via bootstrap, priority 0, hash value 1093093598
  RP 150.1.8.8, via bootstrap, priority 0, hash value 1364246456
```

Notice the RP selected for the group and the hash function values. We can also check the control-plane by joining groups and checking the corresponding the (*,G) entries.

```
R5#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface GigabitEthernet1.45
R5(config-subif)#ip igmp join-group 224.1.1.1
R5(config-subif)#ip igmp join-group 224.1.1.2
R5(config-subif)#end
```

R8 has both groups, but only because it is in the path toward R10.

```
R8#show ip mroute

IP Multicast Routing Table
(*, 224.1.1.1), 00:07:09/00:03:12, RP 150.1.10.10, flags: S
  Incoming interface: GigabitEthernet1.108, RPF nbr 155.1.108.10
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:07:09/00:03:12
(*, 224.1.1.2), 00:07:06/00:03:15, RP 150.1.8.8, flags: S

  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:07:06/00:03:15

(*, 224.0.1.40), 00:13:07/00:02:55, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:13:05/00:02:47
```

https://t.me/learningnets

R10 only has the group for which it is serving as an RP.

```
R10#show ip mroute
IP Multicast Routing Table
(*, 224.1.1.1), 00:01:11/00:03:17, RP 150.1.10.10, flags: S

  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse, 00:01:11/00:03:17

(*, 224.0.1.40), 00:07:02/00:02:51, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Sparse, 00:07:00/00:02:51
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Filtering BSR Messages

> You must load the initial configuration files for the section, named **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the path between R5 and R10.
- Configure R5 as the RP and BSR.
- Configure your network so that R10 does not learn any RP information.

Configuration

As you may recall, filtering Auto-RP messages requires applying the administrative multicast boundary command. Filtering BSR messages is much easier, because they are carried inside PIM. When you apply the command `ip pim bsr-border` on a link, BSR messages are no longer flooded or received on that link. This means that any RP or BSR advertisements are filtered, effectively creating two isolated domains of RP information exchange.

```
R5:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
```

https://t.me/learningnets

```
!
ip pim bsr-candidate Loopback0
ip pim rp-candidate Loopback0
R8:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
 ip pim bsr-border
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-mode
```

Verification

Ensure that R10 did not learn any RP information via the BSR. Again, it may take a few moments for any previously existing entries to age out.

```
R10#show ip pim rp mapping
 PIM Group-to-RP Mappings
```

R8 is not affected by this filtering, however.

```
R8#show ip pim bsr-router
PIMv2 Bootstrap information BSR address: 150.1.5.5 (?)

  Uptime:      00:08:41, BSR Priority: 0, Hash mask length: 0
  Expires:     00:01:34
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Stub Multicast Routing & IGMP Helper

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10.
- Configure R4 as the RP and BSR.
- Configure R8 as the stub multicast router with R10 emulating a host on the stub segment.
  - Join R10's VLAN108 interface to group **239.1.1.7** for testing.

Configuration

Stub multicast routing is useful in situations where you have small remote sites connected to a centralized WAN cloud across low-bandwidth links that use low-end routers. Such remote sites never perform any transit function, and they may suffer from resource starvation caused by PIM DM's flood-and-prune behavior, or PIM SM RP announcements, RP cache growth, and mroute state proliferation.

Assume that R5 is the central router and R8 is the stub remote router with directly connected receivers. In the case where both routers are running PIM DM, periodic flood-and-prune behavior has the capacity to overwhelm the WAN link. If both routers were to run PIM SM, R8 would have to accept RP discovery messages, build its own RP cache, and maintain states for all multicast groups joined by local receivers, thus possibly over-taxing a small router's resources. Being a stub multicast router allows R8 to be exempted from PIM and IGMP message processing. Instead, R8 is configured to forward all IGMP messages received on its

https://t.me/learningnets

client-facing interface to R5. Thus, R8 never creates any group states. The command to achieve this is `ip igmp helper-address <Upstream-IP>` . As a result, R5 will track all IGMP states for any client on the segment connected to R8.

Next, R8 is configured with PIM DM on both the uplink (GigabitEthernet1.58 in our case) and the client-facing interfaces. This will allow the router to flood ANY multicast traffic received from upstream sources down to clients; this is based on default PIM DM behavior, because there is no downstream router to prune the tree. Finally, R5 is configured with PIM enabled on its downstream interface, but with a special neighbor filter applied, to make sure the upstream and the downstream routers never form a PIM adjacency. This allows the upstream router to flood multicast traffic downstream, because IOS will not send a multicast packet out of a non-PIM interface. At the same time, the downstream router will not be able to prune any group using PIM signaling. The command used to filter PIM neighbors is `ip pim neighbor-filter <ACL>` where the standard ACL permits or denies particular neighbors.

The next effect is that the upstream router (R5) builds a multicast distribution tree on behalf of the downstream router (R8) by virtue of the IGMP proxy function performed by the downstream router. At the same time, no excessive load is being put on the stub router or the stub link, effectively saving bandwidth and router resources. Essentially, R5 performs all multicast routing functions for R8, and the latter is only used as a “dumb” packet forwarder.

```
R6:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim sparse-mode
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim bsr-candidate Loopback0
ip pim rp-candidate Loopback0
R5:
```

https://t.me/learningnets

```
ip multicast-routing distributed
!
access-list 58 deny 155.1.58.8
access-list 58 permit any
!
interface GigabitEthernet1.58
 ip pim sparse-mode
 ip pim neighbor-filter 58
!
interface GigabitEthernet1.45
 ip pim sparse-mode
```

R8:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim dense-mode
!
interface GigabitEthernet1.108
 ip pim dense-mode
 ip igmp helper-address 155.1.58.5
```

We must enable PIM DM on R10's interface connected to R8 to activate multicast processing on this interface. A neighbor filter is used allow R10 to avoid becoming PIM neighbors with R8. R10 also joins group 239.1.1.7 for testing.

R10:

```
ip multicast-routing distributed
!
access-list 10 deny any
!
interface GigabitEthernet1.108
 ip igmp join-group 239.1.1.7
 ip pim dense-mode
 ip pim neighbor-filter 10
```

Verification

Check that R5 sees the group 239.1.1.7 as directly connected on its GigabitEthernet1.58 interface. Next, ping the group from R6.

https://t.me/learningnets

```
R5#show ip igmp groups

IGMP Connected Group Membership
Group Address    Interface                Uptime    Expires   Last Reporter   Group Accounted
239.1.1.7        GigabitEthernet1.58      00:01:01  00:02:32  155.1.58.8

<snip>
!
!R6#ping 239.1.1.7 repeat 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 239.1.1.7, timeout is 2 seconds:
Reply to request 0 from 155.1.108.10, 55 ms
Reply to request 1 from 155.1.108.10, 23 ms

Reply to request 2 from 155.1.108.10, 10 ms
Reply to request 3 from 155.1.108.10, 49 ms
```

Check the multicast route states on R5 and R8. Notice that R5 uses sparse-mode SPT to forward traffic down to R8. In turn, R8 simply floods the traffic using dense mode forwarding to the directly connected source.

```
R5#show ip mroute 239.1.1.7
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.1.1.7), 00:04:13/stopped, RP 150.1.4.4, flags: SJC
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:04:13/00:02:19
 (155.1.146.6, 239.1.1.7
), 00:00:54/00:02:05, flags: JT
```

https://t.me/learningnets

```
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list: GigabitEthernet1.58, Forward/Sparse
, 00:00:54/00:02:19
!
!R8#show ip mroute 239.1.1.7
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.1.1.7), 00:05:00/stopped, RP 150.1.4.4, flags: SJC
  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list:
    GigabitEthernet1.108, Forward/Dense, 00:05:00/stopped
 (155.1.146.6, 239.1.1.7
), 00:01:42/00:01:17, flags: JT
  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list: GigabitEthernet1.108, Forward/Dense
, 00:01:42/stopped
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

IGMP Filtering

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the path between R8 and R10.
- Configure R8 as the RP and BSR.
- Only permit R8 to accept IGMP joins for groups in the range **239.1.1.0/24** via its connection to R10.
  - Limit the number of concurrent IGMP states on the interface to 10.

Configuration

IGMP is the protocol used by multicast receivers to communicate their willingness to listen to a particular multicast group. When a host wants to join a multicast group, it sends an IGMP membership report message to the multicast address heard by all routers on the segment. This report contains the multicast group that the host wants to join. The multicast router may control groups allowed to be joined by the receivers. When you apply the command `ip igmp access-group <ACL>` to an interface, the router will filter all attempts to join groups not matching the access-list. Recall that you can accomplish this goal by using the command `ip multicast boundary` , but `ip igmp access-group` is more commonly used on the interfaces facing receivers. Notice that you can use either standard or extended access-lists with this command.

If you use a standard access-list, your configuration applies to IGMP v1, v2, and v3 receivers. The hosts are allowed to listen to the channels (multicast groups) matching an entry in the access-list. If you use an extended access-list, you may also selectively filter IGMPv3 reports. IGMPv3 allows receivers to join explicit

https://t.me/learningnets

sources along with the multicast group. That is, every IGMPv3 report contains the list of groups along with the multicast sources that the receiver wants to listen. The access-list entry will have the format **`permit ip <src-ip> <src-mask> <group-ip> <group-mask>`** . If you want to filter joins to any source, use the 0.0.0.0 255.255.255.255 wildcard pair. However, if you want to filter IGMPv2 or v1 joins that don't support explicit source specification, use the host IP address of 0.0.0.0 for the source.

Another useful feature is limiting the number of mroute states created for the interface as a result of IGMP reports. The same command **`ip igmp limit <N>`** can be applied globally and per-interface at the same time. In the first case, it limits the aggregate number of multicast groups joined by directly connected receivers on all multicast interfaces. When applied per-interface, it limits the number of different multicast groups that can be joined on this particular interface.

```
R8:
ip multicast-routing distributed
!
ip access-list standard IGMP_FILTER
 permit 239.1.1.0 0.0.0.255
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
 ip igmp access-group IGMP_FILTER
 ip igmp limit 10
!
interface Loopback0
 ip pim sparse-mode
!
ip pim bsr-candidate Loopback0
ip pim rp-candidate Loopback0
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip pim sparse-mode
```

https://t.me/learningnets

Verification

Check the IGMP settings on R8's interface. Notice that the maximum number of allowed IGMP states is 10.

```
R8#show ip igmp interface GigabitEthernet1.108
GigabitEthernet1.108 is up, line protocol is up
  Internet address is 155.1.108.8/24
  IGMP is enabled on interface
  Current IGMP host version is 2
  Current IGMP router version is 2
  IGMP query interval is 60 seconds
  IGMP configured query interval is 60 seconds
  IGMP querier timeout is 120 seconds
  IGMP configured querier timeout is 120 seconds
  IGMP max query response time is 10 seconds
  Last member query count is 2
  Last member query response interval is 1000 ms Inbound IGMP access group is IGMP_FILTER
  IGMP activity: 0 joins, 0 leaves Interface IGMP State Limit : 0 active out of 10 max
  Multicast routing is enabled on interface
  Multicast TTL threshold is 0
  Multicast designated router (DR) is 155.1.108.10  IGMP querying router is 155.1.108.8 (this system)

  No multicast groups joined by this system
```

Now configure R10 to join a group, such as 239.1.1.10.

```
R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#
R10(config)#interface GigabitEthernet1.108
R10(config-subif)#ip igmp join-group 239.1.1.10
```

Check R8 for this IGMP group and notice that the group is marked as "Group Accounted."

```
R8#show ip igmp groups

IGMP Connected Group Membership
Group Address    Interface                Uptime    Expires   Last Reporter   Group Accounted
239.1.1.10       GigabitEthernet1.108     00:01:29  00:02:51  155.1.108.10    Ac
224.0.1.40       GigabitEthernet1.58      00:03:10  00:02:54  155.1.58.8
!
```

https://t.me/learningnets

```
!R8#show ip igmp interface gigabitEthernet 1.108

GigabitEthernet1.108 is up, line protocol is up
  Internet address is 155.1.108.8/24
  IGMP is enabled on interface
  Current IGMP host version is 2
  Current IGMP router version is 2
  IGMP query interval is 60 seconds
  IGMP configured query interval is 60 seconds
  IGMP querier timeout is 120 seconds
  IGMP configured querier timeout is 120 seconds
  IGMP max query response time is 10 seconds
  Last member query count is 2
  Last member query response interval is 1000 ms
  Inbound IGMP access group is IGMP_FILTER
  IGMP activity: 1 joins, 0 leaves Interface IGMP State Limit : 1 active out of 10 max

  Multicast routing is enabled on interface
  Multicast TTL threshold is 0
  Multicast designated router (DR) is 155.1.108.10
  IGMP querying router is 155.1.108.8 (this system)
  No multicast groups joined by this system
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

IGMP Timers

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the VLAN 146 (GigabitEthernet1.146) segment between R1, R4, and R6.
  - Configure R6 as the BSR and RP.
  - Configure the designated IGMP querier on VLAN 146 so that failed multicast traffic receivers are detected and removed within 20 seconds.
  - Designated querier failures should be detected two times faster than by default.
  - Every active receiver should respond to general IGMP queries within 4 seconds.
- Enable PIM sparse-mode multicast delivery between R8 and R10.
  - Configure R8 as the static RP.
  - Configure R8 to remove multicast group state immediately after a corresponding IGMP leave report has been received for **239.1.1.0/24** groups.
  - For testing purposes, join R10's VLAN108 interface to the **239.1.1.10** group.

Configuration

IGMP reports are sent asynchronously, so they might be missed by the router. Therefore, one of the multicast-capable routers sharing the same segment will be elected as the designated IGMP querier, and it will periodically send IGMP Membership queries to all hosts on the segment. Every host receiving the Membership query should respond with an IGMP report containing all joined groups.

https://t.me/learningnets

The designated querier is the router with the lowest IP address on the segment, if its running IGMP version 2 or 3. Notice that the PIM DR on the segment is elected based on the highest IP address by default, so some functional decoupling and load-balancing may occur if there is more than one router on the segment. In addition to sending the periodic membership queries, the designated querier also builds PIM SM shared trees on behalf of the receivers signaling multicast group membership.

Periodic queries are sent at the interval defined by the interface-level command `ip igmp query-interval <seconds>` . If a non-designated multicast router does not hear the membership queries for more than the interval defined by the command `ip igmp querier-timeout <seconds>` , it will attempt to become a designated querier itself, assuming that the old one failed. If the latter command is not configured on the interface, the querier timeout equals twice the query-interval configured on the same interface. The default query-interval value is 60 seconds (although the RFC recommends 125 seconds).

When IGMPv1 is configured on an interface explicitly, the router times out the group state if there is no report for this group during three consecutive membership queries (180 seconds by default). This might result in very high leave latency, because IGMPv1 has no explicit group leave message. IGMPv2 significantly enhances the procedure of leaving a multicast group, as detailed below:

1. Every membership query message contains a special timer value, defined by the command `ip igmp query-max-response-time [time-in-seconds]` . Every host on the segment is supposed to send an IGMP report during the time-window defined by this interval. All hosts count to the interval value and randomly fire the IGMP report. If a host hears another report for the same group, it suppresses its own message. Thus, excessive report flooding is avoided. If there is no report for a given group during the query-max-response interval, the router removes the mroute state for this group.

2. Every host, willing to leave a group, may send out a special IGMP Leave report for this particular group. As soon as the router receives a leave report, it generates a special IGMP group-specific query for the multicast channel in question. This query is needed to check for any other hosts on the segment that still need this group. The number of special queries generated is defined by the command `ip igmp last-member-query-count <N>` , which is set to 2 by default. The queries are sent at the intervals specified by the command `ip igmp last-member-query-interval <milliseconds>` , which is 1000 ms by default. If there is no answer to the special queries, the group state is removed from the mroute table.

3.

https://t.me/learningnets

If there is just one receiver for a particular group on the segment (for example, if there is just one host connected to the router), the leave latency could be further reduced by configuring the **`ip igmp immediate-leave group-list <access-list>`** command. If there is an IGMP leave message received on the interface for a group matching the access-list, the mroute state is immediately removed, without any further delays. The access-list is a standard ACL that defines the groups eligible for this feature.

Notice that the explicit leave feature is also available for IGMPv3, with the extension to the multicast group source. That is, a host may leave a particular sender for the group, while continuing to listen to other senders. For our scenario, R1 is the designated querier for the segment. Because R4 has the next-lowest IP address, it is the backup querier. In real life, you would configure every router on the segment with the same settings, but this task's goal is to check your understanding of the querier election. For the immediate leave feature, we configure only the groups permitted by the IGMP access-group. You may configure it for all groups, because the resulting effect would be the same.

```
R1:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim sparse-mode
 ip igmp query-interval 20
 ip igmp querier-timeout 60
 ip igmp query-max-response-time 4
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip igmp querier-timeout 60
 ip pim sparse-mode
R6:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip igmp querier-timeout 60
 ip pim sparse-mode
!
interface Loopback 0
 ip pim sparse-mode
!
ip pim bsr-candidate Loopback0
ip pim rp-candidate Loopback0
```

https://t.me/learningnets

```
R8:

ip multicast-routing distributed
ip pim rp-address 150.1.8.8
!
ip access-list standard IMMEDIATE_LEAVE
 permit 239.1.1.0 0.0.0.255
!
interface GigabitEthernet1.108
 ip pim sparse-mode
 ip igmp immediate-leave group-list IMMEDIATE_LEAVE
R10:

ip multicast-routing distributed
ip pim rp-address 150.1.8.8
!
interface GigabitEthernet1.108
 ip pim sparse-mode
 ip igmp join-group 239.1.1.10
```

Verification

Verify timers using **`the show igmp interface`** command.

https://t.me/learningnets

```
R1#show ip igmp interface GigabitEthernet1.146

GigabitEthernet1.146 is up, line protocol is up
  Internet address is 155.1.146.1/24
  IGMP is enabled on interface
  Current IGMP host version is 2
  Current IGMP router version is 2 IGMP query interval is 20 seconds
IGMP configured query interval is 20 seconds
IGMP querier timeout is 60 seconds
IGMP configured querier timeout is 60 seconds
IGMP max query response time is 4 seconds

  Last member query count is 2
  Last member query response interval is 1000 ms
  Inbound IGMP access group is not set
  IGMP activity: 1 joins, 0 leaves
  Multicast routing is enabled on interface
  Multicast TTL threshold is 0
  Multicast designated router (DR) is 155.1.146.6  IGMP querying router is 155.1.146.1 (this system)

  Multicast groups joined by this system (number of users):
      224.0.1.40(1)
```

Enable IGMP debugging on R8, and unsubscribe R10 from the group 239.10.10.10. Notice that the group state is removed immediately, without any additional last member queries.

```
R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#interface GigabitEthernet1.108
R10(config-subif)#no ip igmp join-group 239.1.1.10
!
!R8#debug ip igmp
!
!IGMP(0): Received Leave from 155.1.108.10 (GigabitEthernet1.108) for 239.1.1.10
IGMP(0): Leave group 239.1.1.10 immediately on GigabitEthernet1.108
IGMP(0): Deleting 239.1.1.10 on GigabitEthernet1.108

IGMP(0): Received v2 Report on GigabitEthernet1.108 from 155.1.108.10 for 224.0.1.40
IGMP(0): Received Group record for group 224.0.1.40, mode 2 from 155.1.108.10 for 0 sources
IGMP(0): Updating EXCLUDE group timer for 224.0.1.40
IGMP(0): MRT Add/Update GigabitEthernet1.108 for (*,224.0.1.40) by 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Multicast Helper Map

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM dense-mode multicast delivery on the path Ethernet between R6 and R10.
  - Do not enable PIM over the DMVPN network between R4 and R5.
- When R6 sends broadcast UDP packets on port 5000, those packets should be transported across the network and broadcasted on VLAN108.
  - Use the group **239.1.1.100** to accomplish this task, and use DNS broadcasts for testing.

Configuration

The purpose of this feature is to allow forwarding of broadcast traffic across a multicast capable network. Generally, you need a single Layer 2 domain between two nodes to let them hear each other's broadcast packets. However, broadcast UDP packets can be relayed between two subnets using a special IOS feature known as the helper-address. There are two variations of this feature:

- Unicast helper (`ip helper-address`), which converts the broadcast destination address to the fixed unicast IP address. Most often this feature is used with DHCP to forward requests to the server.
- Multicast helper (`ip multicast helper-map`), which converts the broadcast destination to a fixed multicast address.

https://t.me/learningnets

The multicast helper-map feature allows scalable forwarding of broadcast traffic between disjoined segments. This is often needed to support legacy applications like stock tickers that use broadcast to deliver information to multiple sources simultaneously. To configure multicast helper, follow the steps outlined below:

Step 1: Set up a multicast network between the segments that should exchange broadcast packets. You should select a group to deliver the broadcast packets and decide which PIM mode to use. If you chose PIM SM, make sure the group you chose maps to an RP. Make sure multicasting works by joining an interface on the egress router (closest to the broadcast receiver) to the selected group, and ping this group from the ingress router (closest to the broadcast source).

Step 2: Enable broadcast forwarding on the ingress router—the one directly connected to the source. If there are multiple sources, you must configure all respective routers. Use the command `ip forward-protocol udp <port-number>` to enable forwarding of broadcast UDP packets sent to the specified port-number.

Step 3: Configure a multicast helper-map on the ingress routers to redirect broadcast packets to the selected multicast address. The syntax for this interface-level command is `ip multicast helper-map broadcast <mcast-address> <ACL>` . The access-list controls which broadcast packets are eligible to be converted into the multicast. For example, if you want to forward UDP packets destined to port 5000, use an access-list similar to the following: `access-list 100 permit udp any any eq 5000` . Note that the same UDP port must be enabled for broadcast forwarding at Step 2. All broadcast traffic received on the configured interface that matches the access-list is converted and sent to the specified multicast address. If the group is in sparse mode, the ingress router will register the source with the RP, per the usual procedure.

Step 4: Enable broadcast forwarding on the egress router, that is, the router directly connected to the destination subnet. Use the same command that you used in Step 2, `ip forward-protocol udp <port-number>` , to accomplish this. Next, enable multicast helper map on the egress router for all interfaces that may *receive* multicast traffic. Note that you should not configure the multicast-helper on the interface connected to the destination. Use the command `ip multicast helper-map <mcast-group> <directed-broadcast-IP> <ACL>` where mcast-group is the same group you used in Step 3 and directed-broadcast-IP is the broadcast subnet IP address on the segment that receives the broadcast traffic.

Step 5: Enable directed broadcasts on the interface connected to the receiving segment using the command `ip directed-broadcast` . This is needed to successfully send broadcasts out of this segment. By default, the broadcasts are sent to the address 255.255.255.255, irrespective of the `directed-broadcast-IP` configured in Step 4. If you want to use a different address, put the command `ip broadcast-address <IP>`

https://t.me/learningnets

on the same segment.

To test your configuration, you will need a broadcast packet source. You may use the IP SLA command to generate UDP packets to a segment broadcast address, but this might not work on some platforms/IOS versions. If that's the case, you may use either of the following two methods:

- Enable DNS name resolution, but do not configure a DNS server. After this, the router will broadcast for any DNS name entered in the command line using the address 255.255.255.255 out all interfaces. You will need to adjust the port in your access-lists to forward this broadcast traffic.
- Generate an extended `traceroute` command using parameters to trace to the broadcast destination, starting off the port number that is covered by your ACL.

```
R6:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim dense-mode
R4:
ip multicast-routing distributed
ip forward-protocol udp 5000
!
interface GigabitEthernet1.45
 ip pim dense-mode
!
ip access-list extended TRAFFIC
 permit udp any any eq 5000
 permit udp any any eq 53
!
interface GigabitEthernet1.146
 ip pim dense-mode
 ip pim dr-priority 100
 ip multicast helper-map broadcast 239.1.1.100 TRAFFIC
R5:
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim dense-mode
!
interface GigabitEthernet1.45
 ip pim dense-mode
R8:
```

https://t.me/learningnets

```
ip multicast-routing distributed
ip forward-protocol udp 5000
!
ip access-list extended TRAFFIC
 permit udp any any eq 5000
 permit udp any any eq 53
!
interface GigabitEthernet1.58
 ip pim dense-mode
 ip multicast helper-map 239.1.1.100 155.1.108.255 TRAFFIC
!
interface GigabitEthernet1.108
 ip pim dense-mode
 ip directed-broadcast
 ip broadcast-address 155.1.108.255
R10:

ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim dense-mode
```

Verification

For verification, we will use DNS broadcasts sent from R6. Configure R6 to resolve DNS names, but do not provide any DNS server.

```
R6:

ip domain lookup
```

Note that in previous versions of IOS the command, **`no ip mroute-cache`** was used on the interfaces to disable CEF switching of multicast packets. This command, along with **`debug ip mpacket`** , is now obsolete and no longer supported on newer versions of code such as the one being used for this example (15.4). The new syntax for disabling multicast CEF switching on the interfaces, along with debugging multicast forwarding, is shown below. Enable debugging on R4 and R8 (ingress and egress points) and start a broadcast traffic flow on R6.

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.146
R4(config-subif)#no ip mfib cef input
R4(config-subif)#no ip mfib cef output
R4(config-subif)#interface GigabitEthernet1.45
 R4(config-subif)#no ip mfib cef input
```

https://t.me/learningnets

```
!R4#debug ip mfib pak 239.1.1.100

MFIB IPv4 pak debugging enabled for group 239.1.1.100 for default IPv4 table
!R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#access-list 100 permit udp any any eq 53
!R4#debug ip packet detail 100
IP packet debugging is on (detailed) for access list 100

R8#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R8(config)#interface GigabitEthernet1.58
R8(config-subif)#no ip mfib cef input
R8(config-subif)#no ip mfib cef output
R8(config-subif)#interface GigabitEthernet1.108
 R8(config-subif)#no ip mfib cef input
R8(config-subif)#no ip mfib cef output
!R8#debug ip mfib pak 239.1.1.100
MFIB IPv4 pak debugging enabled for group 239.1.1.100 for default IPv4 table
!R8#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R8(config)#access-list 100 permit udp any any eq 53
!R8#debug ip packet detail 100

IP packet debugging is on (detailed) for access list 100
```

R4 accepts the broadcasts and converts them to multicast packets. Initially, the SPT is not built and some packets are lost, and the OIL for the (S,G) is empty. When the SPT is finally finished, everything will work smoothly.

https://t.me/learningnets

```
R4#

!
!
IP: s=155.1.146.6 (GigabitEthernet1.146), d=255.255.255.255, len 50, input feature
    UDP src=54185, dst=53, MCI Check(95), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
FIBipv4-packet-proc: route packet from GigabitEthernet1.146 src 155.1.146.6 dst 255.255.255.255
MFIBv4(0x0): Pkt (155.1.146.6,239.1.1.100
) from GigabitEthernet1.146 (PS) Queued signalling for routing protocol
R4#
IP: s=155.1.146.6 (GigabitEthernet1.146), d=255.255.255.255, len 50, input feature
    UDP src=54185, dst=53, MCI Check(95), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
FIBipv4-packet-proc: route packet from GigabitEthernet1.146 src 155.1.146.6 dst 255.255.255.255
FIBfwd-proc: Default:255.255.255.255/32 receive entryFIBipv4-packet-proc: packet routing failed
MFIBv4(0x0): Pkt (155.1.146.6,239.1.1.100) from GigabitEthernet1.146 (PS) accepted for forwarding
MFIBv4(0x0): Pkt (155.1.146.6,239.1.1.100) from GigabitEthernet1.146 (PS) sent on GigabitEthernet1.45
```

R8 receives the multicast packets and sends them as broadcasts to R10.

```
R8#
!
!
IP(0): s=155.1.146.6 (GigabitEthernet1.58) d=239.1.1.100 (GigabitEthernet1.108) id=0, ttl=252, prot=17, len=50(50),
IP: tableid=0, s=155.1.146.6 (GigabitEthernet1.58), d=155.1.37.255 (GigabitEthernet1.108), routed via RIB
!
!R8#show ip mroute 239.1.1.100
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
```

https://t.me/learningnets

```
(*, 239.1.1.100), 00:01:23/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Dense, 00:01:23/stopped
    GigabitEthernet1.108, Forward/Dense, 00:01:23/stopped
(155.1.146.6, 239.1.1.100
), 00:01:03/00:01:56, flags: PLTX
  Incoming interface: GigabitEthernet1.58, RPF nbr 155.1.58.5
  Outgoing interface list: GigabitEthernet1.108, Prune/Dense, 00:01:00/00:01:59
```

Check the mroute states on R4 and R5 to ensure that the traffic follows the SPT.

```
R4#show ip mroute 239.1.1.100
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.1.1.100), 00:00:26/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Dense, 00:00:26/stopped
    GigabitEthernet1.146, Forward/Dense, 00:00:26/stopped
(155.1.146.6, 239.1.1.100
), 00:00:26/00:02:32, flags: T
  Incoming interface: GigabitEthernet1.146, RPF nbr 155.1.146.6
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Dense, 00:00:26/stopped
!
!Rack1R5#show ip mroute 239.1.1.100
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
```

https://t.me/learningnets

```
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.1.1.100), 00:01:01/stopped, RP 0.0.0.0, flags: DC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Dense, 00:01:01/stopped
    GigabitEthernet1.58, Forward/Dense, 00:01:01/stopped
 (155.1.146.6, 239.1.1.100
), 00:01:00/00:01:59, flags: T
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Dense, 00:01:00/stopped
```

The reason we are able to do this testing with UDP traffic with destination port of 53 is because by default IOS has the following command enabled: **`ip forward-protocol udp domain`** . It does not appear in the running configuration, but it can be seen if we disable it. If this was not enabled by default, we would need to enable it on R4 and R8 for the testing to work.

```
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#no ip forward-protocol udp 53
!R4#show running-config | section forward-protocol
ip forward-protocol ndno ip forward-protocol udp domain

ip forward-protocol udp 5000
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Bidirectional PIM

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** in order to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10, including VLAN10.
  - Do not enable PIM over the DMVPN network netween R4 and R5.
- The group range **238.0.0.0/8** is used for network video-conferencing with many participants.
- Configure the network so that this group uses a single shared tree rooted on R5 for multicast traffic delivery.
  - Use BSR for this purpose.

Configuration

Bidirectional PIM or PIM BiDir is a special extension to the PIM SM concept that uses only the shared tree for multicast distribution. This mode of operation is useful in situations where most receivers are also senders at the same time. For example, this might be the case when you run videoconferencing. In this situation, in addition to joining the shared tree rooted at the RP, every receiver needs to join the shortest-path multicast distribution tree rooted at every other participant. If the number of participants is significant, the amount of multicast route states in the core of the network will grow at a quadric rate.

One special feature of PIM SM shared and shortest path trees is that they are unidirectional; traffic passes down from the root to the leaves of the tree. PIM BiDir

https://t.me/learningnets

uses a single distribution tree rooted at the RP for all sources and receivers at the same time. If there are multiple RPs, there could be many BiDir trees. Unlike the classic tree, traffic may flow up and down this tree. When a source sends multicast packets, they first flow up to the root of the tree (toward the RP) and then down to all receivers.

To build the bi-directional tree, PIM elects special designated forwarders (DFs) on every link in the network. A DF is elected based on the rules similar to the ones used in the PIM Assert procedure—that is, the router on the link that has the shortest metric to reach the RP is selected as the DF. Notice that a single router might be the DF on one link and a non-DF on another. After the elections, DF routers are the only routers that are allowed to forward traffic toward the RP—via the bi-directional tree (this is considered the “upstream” portion of the BiDir tree). Every router in the multicast domain creates a (*,G) state for each BiDir group, with the OIL built based on PIM Join messages received from its neighbors. This is the “downstream” portion of the BiDir tree. Any packet received on a valid RPF interface is forwarded based on the OIL. At the same time, the DF will forward a copy of these packets toward the RP (upstream through the shared tree), provided that the packet is not received on the interface pointing to the RP.

Notice that PIM BiDir does not utilize the source registration procedure, via PIM Register/Register-Stop messages. Every source connected to a PIM BiDir capable router may start sending at any time, and the packets will flow upward to the RP. After reaching the RP, packets are either dropped, if there are no receivers for this group (that is, the OIL for this (*,G) state is empty) or forwarded down the BiDir tree. There is no way for the RP to signal the source to stop sending traffic even if there are no receivers. This means that commands like “ip pim accept-register” covered in a later task will not work with PIM BiDir, because they rely on these “register-stop” messages to work.

Configuring PIM BiDir is relatively simple. You just need to enable BiDir PIM on all multicast routers by using the command `ip pim bidir-enable` and designate particular RP/Group combinations as bi-directional. You can do this in the following ways:

- Using a static RP configuration with the command **`ip pim rp-address <IP> <ACL> bidir .`**
- Using BSR or RP information dissemination, you may flag particular group/RP combinations as bi-directional using the syntax **`ip pim rp-candidate <interface> group-list <ACL> bidir .`**
- Using Auto-RP for RP information dissemination, you may flag particular group/RP combinations as bi-directional using the syntax **`ip pim send-rp-announce <interface> scope <TTL> group-list <ACL> bidir .`**

This is all you need to enable bi-directional PIM. However, remember to enable bi-

https://t.me/learningnets

directional mode on all routers in your network, or you might end up with routing loops.

```
R6:
ip multicast-routing distributed
ip pim bidir-enable
!
interface GigabitEthernet1.146
 ip pim sparse-mode
R4:
ip multicast-routing distributed
ip pim bidir-enable
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface GigabitEthernet1.146
 ip pim sparse-mode
R5:
ip multicast-routing distributed
ip pim bidir-enable
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface Loopback 0
 ip pim sparse-mode
!
ip access-list standard GROUP238
 permit 238.0.0.0 0.255.255.255
!
ip pim rp-candidate Loopback0 group-list GROUP238 bidir
ip pim bsr-candidate Loopback0
R8:
ip multicast-routing distributed
ip pim bidir-enable
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
```

https://t.me/learningnets

```
R10:

ip multicast-routing distributed
ip pim bidir-enable
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip pim sparse-mode
```

Verification

To verify, join R6 and R10 to the bi-directional group 238.1.1.1:

```
R6:
interface GigabitEthernet1.146
 ip igmp join-group 238.1.1.1
R10:

interface GigabitEthernet1.10
 ip igmp join-group 238.1.1.1
```

Ping this group from R5:

```
R5#ping 238.1.1.1 repeat 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 238.1.1.1, timeout is 2 seconds:
Reply to request 0 from 155.1.146.6, 15 ms
Reply to request 0 from 155.1.10.10, 38 ms

Reply to request 0 from 155.1.10.10, 34 ms
Reply to request 0 from 155.1.10.10, 34 ms
Reply to request 0 from 155.1.146.6, 15 ms
Reply to request 1 from 155.1.10.10, 18 ms
Reply to request 1 from 155.1.146.6, 27 ms
Reply to request 1 from 155.1.146.6, 27 ms
Reply to request 1 from 155.1.10.10, 18 ms
Reply to request 1 from 155.1.10.10, 18 ms
```

Check the mroute states on all routers. Notice that some interfaces are marked as Bidir-Upstream; these interfaces are used to send packets upward the root of the

https://t.me/learningnets

tree. The root of the tree (the RP) will have no upstream interfaces:

```
R6#show ip mroute 238.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 238.1.1.1
), 00:05:59/00:02:53, RP 150.1.5.5, flags: BPL Bidir-Upstream: GigabitEthernet1.146
, RPF nbr 155.1.146.4
  Outgoing interface list: GigabitEthernet1.146, Bidir-Upstream/Sparse, 00:05:59/stopped
!
!R4#show ip mroute 238.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 238.1.1.1
), 00:03:09/00:03:18, RP 150.1.5.5, flags: BC Bidir-Upstream: GigabitEthernet1.45, RPF nbr 155.1.45.5
  Outgoing interface list: GigabitEthernet1.146, Forward/Sparse, 00:03:09/00:03:18
    GigabitEthernet1.45, Bidir-Upstream/Sparse, 00:03:09/stopped
!
```

https://t.me/learningnets

```
!R5#show ip mroute 238.1.1.1

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 238.1.1.1
), 00:03:59/00:03:27, RP 150.1.5.5, flags: B Bidir-Upstream: Null, RPF nbr 0.0.0.0
  Outgoing interface list: GigabitEthernet1.58, Forward/Sparse, 00:03:52/00:02:33
 GigabitEthernet1.45, Forward/Sparse, 00:03:59/00:03:27
!
!R8#show ip mroute 238.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 238.1.1.1
), 00:04:35/00:02:50, RP 150.1.5.5, flags: B Bidir-Upstream: GigabitEthernet1.58
, RPF nbr 155.1.58.5
  Outgoing interface list: GigabitEthernet1.108, Forward/Sparse, 00:04:35/00:02:50
    GigabitEthernet1.58, Bidir-Upstream/Sparse, 00:04:35/stopped
!
```

https://t.me/learningnets

```
!R10#show ip mroute 238.1.1.1


IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 238.1.1.1
), 00:05:12/00:02:47, RP 150.1.5.5, flags: BCL Bidir-Upstream: GigabitEthernet1.108
, RPF nbr 155.1.108.8
  Outgoing interface list: GigabitEthernet1.10, Forward/Sparse, 00:05:12/00:02:47


    GigabitEthernet1.108, Bidir-Upstream/Sparse, 00:05:12/stopped
```

Note that since the RP is now a PIM BiDir RP, there is no source registration procedure via PIM Register/Register-Stop messages. In Sparse-Mode, PIM Register Tunnels were automatically created on all PIM routers in order to unicast the PIM register messages to the RP. These Tunnels are not created when PIM BiDir is in use. The RP does not need to decapsulate/encapsulate unicast register messages when running BiDir, so these dynamic tunnels are not needed:

https://t.me/learningnets

```
R10#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 238.0.0.0/8   RP 150.1.5.5 (?), v2, bidir

    Info source: 150.1.5.5 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:10:43, expires: 00:01:53
!
!R10#show ip pim tunnel
 R10#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Source Specific Multicast

> You must load the initial configuration files for the section, **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10.
  - Do not enable PIM over the DMVPN network between R4 and R5.
- Using the default multicast group range, enable PIM SSM functionality on your network.
  - R10 should join the multicast feed **(150.1.6.6,232.10.10.10)**.

Configuration

Classic multicast delivery technologies use IGMPv2 and PIM DM/SM and are known as “Any Source Multicast” or ASM. That is, receivers agree to accept traffic from any source. This is why Rendezvous Points are actually needed in PIM SM—to allow receivers to discover new sources. The core of PIM Source Specific Multicast (SSM) protocol is the use of IGMPv3 signaling by clients. This client-side protocol allows receivers to specify sources that they want to listen to explicitly. That is, the host may explicitly ask to join group G at source S. PIM SSM works in association with IGMPv3 in building shortest-path trees (SPT) only, toward the sources. There are no shared trees in PIM SSM and no RPs are used. Thus, there is no need to use auxiliary protocols such as BSR or Auto-RP to distribute RP information. Notice that source discovery is outside the scope of PIM SSM and IGMPv3 and must be accomplished via some other means, such as global directory services.

Configuring PIM SSM is relatively straight-forward, because it uses regular PIM

https://t.me/learningnets

messages. You need only specify the range of groups that are using SSM signaling with the command `ip pim ssm range {default|range <Standard-ACL>}` . The `default` keyword means that the range 232.0.0.0/8 will be used for SSM. For the groups in the SSM range, no shared trees are allowed and the (*,G) joins are dropped.

The second step in configuring PIM SSM is to enable IGMPv3 on the interfaces connected to the receivers capable of using this protocol. Without IGMPv3, there can be no use of PIM SSM because no other IGMP version allows sources to be selected by the receivers explicitly to build shortest-path trees.

```
R4:
ip multicast-routing distributed
ip pim ssm default
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
R6:
ip multicast-routing distributed
ip pim ssm default
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
R5:
ip multicast-routing distributed
ip pim ssm default
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface GigabitEthernet1.58
 ip pim sparse-mode
R8:
ip multicast-routing distributed
ip pim ssm default
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
```

https://t.me/learningnets

```
 ip pim sparse-mode
R10:

ip multicast-routing distributed
ip pim ssm default
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip pim sparse-mode
 ip igmp version 3
 ip igmp join 232.10.10.10 source 150.1.6.6
```

Verification

PIM SSM is generally easier to configure than ASM, because it does not require the complicated RP infrastructure. All you need to do is verify the SPT toward the explicit source.

```
R10#show ip igmp groups 232.10.10.10 detail

Flags: L - Local, U - User, SG - Static Group, VG - Virtual Group,
       SS - Static Source, VS - Virtual Source,
       Ac - Group accounted towards access control limit

Interface:      GigabitEthernet1.10
Group:          232.10.10.10 Flags:          L SSM
Uptime:         00:02:14
Group mode:     INCLUDE
Last reporter:  155.1.10.10
Group source list: (C - Cisco Src Report, U - URD, R - Remote, S - Static,
                    V - Virtual, M - SSM Mapping, L - Local,
                    Ac - Channel accounted towards access control limit)
  Source Address   Uptime    v3 Exp   CSR Exp   Fwd  Flags
150.1.6.6         00:02:14  00:02:50  stopped   Yes  RL
```

Notice that there is no (*,G) group created; only (S,G) state is created with SSM groups.

```
R10#show ip mroute 232.10.10.10 150.1.6.6
IP Multicast Routing Table Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group
, C - Connected,
```

https://t.me/learningnets

```
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(150.1.6.6, 232.10.10.10), 00:02:11/00:02:49, flags: sLTI
  Incoming interface: GigabitEthernet1.108, RPF nbr 155.1.108.8
  Outgoing interface list:
    GigabitEthernet1.10, Forward/Sparse, 00:02:10/00:02:49
!
!R5#show ip mroute 232.10.10.10 150.1.6.6
IP Multicast Routing TableFlags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group
, C - Connected,
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(150.1.6.6, 232.10.10.10), 00:04:24/00:02:59, flags: sT
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.4
  Outgoing interface list:
    GigabitEthernet1.58, Forward/Sparse, 00:04:24/00:02:59
!
!R8#show ip mroute 232.6.6.6
IP Multicast Routing TableFlags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group
, C - Connected,
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
```

https://t.me/learningnets

```
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(150.1.6.6, 232.10.10.10), 00:05:24/00:03:03, flags: sT

  Incoming interface: GigabitEthernet1.146, RPF nbr 155.1.146.6
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Sparse, 00:05:24/00:03:03
```

One notable difference between ASM and SSM multicast is that the IGMPv3 join triggers state throughout the PIM network all the way up to the source. In ASM, IGMP triggered state up to the RP, not directly to the source. This is why we are able to go to the source (R6 in our example) and look at the (S,G) entry created after joining the group on R10. R10 knows who the source is, so it does not need to go through an RP and can build the tree directly toward the source. This additional "knowledge" of the source is usually built into the host multicast application.

```
R6#show ip mroute 232.10.10.10
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
        L - Local, P - Pruned, R - RP-bit set, F - Register flag,
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode


(150.1.6.6, 232.10.10.10), 00:11:59/00:03:16, flags: sT
```

https://t.me/learningnets

```
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.146, Forward/Sparse, 00:11:59/00:03:16
!
!R6#ping  232.10.10.10 rep 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 232.10.10.10, timeout is 2 seconds:
Reply to request 0 from 155.1.10.10, 32 ms

Reply to request 1 from 155.1.10.10, 11 ms
Reply to request 2 from 155.1.10.10, 29 ms
```

If we source multicast packets destined to 232.10.10.10 from any interface other than R6's Loopback0 (150.1.6.6), R10 will not respond to these packets because it is only listening to multicast packets sourced from 150.1.6.6 and destined to 232.10.10.10.

```
R6#ping
Protocol [ip]: Target IP address: 232.10.10.10
Repeat count [1]: 10
Datagram size [100]:
Timeout in seconds [2]: Extended commands [n]: y
Interface [All]: GigabitEthernet1.146
Time to live [255]: Source address or interface: 155.1.146.6
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 232.10.10.10, timeout is 2 seconds:
Packet sent with a source address of 155.1.146.6
 ..........
!
!R6#ping
Protocol [ip]: Target IP address: 232.10.10.10
Repeat count [1]: 10
Datagram size [100]:
Timeout in seconds [2]: Extended commands [n]: y
Interface [All]: Loopback0
Time to live [255]: Source address or interface: 150.1.6.6
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
```

https://t.me/learningnets

```
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 232.10.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.6.6

 Reply to request 0 from 155.1.10.10, 39 ms
Reply to request 1 from 155.1.10.10, 71 ms
Reply to request 2 from 155.1.10.10, 20 ms
Reply to request 3 from 155.1.10.10, 35 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Multicast BGP Extension

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Inter Domain Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** in order to complete this task.

Task

- Routers R3, R6, R7 and R9 run BGP in AS 100, while routers R4, R5, R8 and R10 run BGP in AS 200.
  - Enable multicast exchange between AS100 and AS200 on both peering links.
- Multicast traffic should prefer to be routed across the Tunnel35 between R3 and R5, using the GigabitEthernet1.146 link as backup.

Configuration

A multicast BGP extension is commonly needed when you plan to exchange multicast traffic between two different administrative domains, such as different autonomous systems. To achieve this goal, you must fulfill the following tasks:

1. Enable PIM between the two domains, to allow signaling of shared and shortest-path trees between them. PIM SM is most often used for multicast traffic exchange between different domains. Each domain usually has its own set of RPs, so you should prevent BSR/Auto-RP information from leaking between the domains. You must exchange information about active sources between the RPs in every domain. Because the RPs are separated, one domain cannot easily learn about the sources in another

https://t.me/learningnets

domain. As we'll see later, a special protocol called MSDP is used for this purpose.

2. To facilitate multicast traffic forwarding, you must exchange information on routes toward the multicast sources in each domain, to allow routers performing RPF checks to do so correctly. PIM uses the unicast routing table to perform these RPF checks, so it may use routes learned via either IGP or BGP. However, BGP is most commonly used to exchange this routing information.

In some cases, you may want to apply different policies to unicast-specific routes exchanged via BGP, as well as to the information about multicast sources. This is possible thanks to Multi-Protocol BGP extensions. Using a special address family, you may exchange prefixes under the "multicast" address-family and apply a different policy to this information. These prefixes are interpreted in the same way as the mroute command information (static mroutes); they are used for RPF checks on the router that receives them. That is, if a prefix is learned via multicast BGP extension, it is assumed to have RPF neighbor toward the next-hop IP address found in the update. If needed, BGP performs recursive routing lookups for the next hop via the IGP routing table to find the immediate RPF neighbor. Unlike the mroute command, which is purely local, the information is propagated via BGP to every neighbor configured for the multicast address family.

Using separate policies for multicast inter-domain RPF information allows the use of different inter-domain links for unicast and multicast traffic. Or you may selectively filter out certain multicast sources from another domain while leaving unicast routes intact.

These tasks require us to enable BGP multicast extensions an all BGP routers. Notice the use of peer-groups under the multicast address family. This is used to ease the amount of configuration per peer. BGP allows configuration of policy per address family, which in our case is needed to propagate multicast RPF information through both domains, because route-reflection is configured separately per address family. Essentially we are able to re-use peer-groups that are being used for IPv4 Unicast address-family. Notice the use of AS-PATH prepending to designate the primary path. Multicast prefixes are subject to the same best-path selection procedure, so you may use the same methods of path manipulation that you used with unicast prefixes. Finally, PIM is activated on the links connecting the two autonomous systems. The PIM BSR border command is used to stop BSR information from leaking into AS 100.

R3:

https://t.me/learningnets

```
router bgp 100
 address-family ipv4 multicast
  neighbor 155.1.0.5 activate
  redistribute ospf 1
  neighbor 150.1.7.7 activate
  neighbor 150.1.7.7 next-hop-self
!
interface Tunnel35
 ip pim sparse-mode
```

R4:

```
route-map PREPEND
 set as-path prepend 200 200 200
!
router bgp 200
 address-family ipv4 multicast
  redistribute ospf 1
  neighbor 155.1.146.6 activate
  neighbor 155.1.146.6 route-map PREPEND out
  neighbor 150.1.5.5 activate
  neighbor 150.1.5.5 next-hop-self
!
interface GigabitEthernet1.146
 ip pim sparse-mode
 ip pim bsr-border
```

R5:

```
router bgp 200
 address-family ipv4 multicast
  neighbor 155.1.0.3 activate
  redistribute ospf 1
  neighbor IBGP route-reflector-client
  neighbor 150.1.4.4 peer-group IBGP
  neighbor 150.1.8.8 peer-group IBGP
  neighbor 150.1.10.10 peer-group IBGP
  neighbor IBGP next-hop-self
!
interface Tunnel35
 ip pim sparse-mode
 ip pim bsr-border
```

R6:

```
route-map PREPEND
 set as-path prepend 100 100 100
!
router bgp 100
 address-family ipv4 multicast
  neighbor 155.1.146.4 activate
  redistribute ospf 1
```

https://t.me/learningnets

```
  neighbor 155.1.146.4 route-map PREPEND out
  neighbor 150.1.7.7 activate
  neighbor 150.1.7.7 next-hop-self
!
interface GigabitEthernet1.146
 ip pim sparse-mode
R7:
router bgp 100
 address-family ipv4 multicast
  neighbor IBGP route-reflector-client
  neighbor 150.1.3.3 peer-group IBGP
  neighbor 150.1.6.6 peer-group IBGP
  neighbor 150.1.9.9 peer-group IBGP
R8:
router bgp 200
 address-family ipv4 multicast
  neighbor 150.1.5.5 activate
R9:
router bgp 100
 address-family ipv4 multicast
  neighbor 150.1.7.7 activate
R10:

router bgp 200
 address-family ipv4 multicast
  neighbor 150.1.5.5 activate
```

Verification

Use the regular show BGP commands to verify that the multicast address family is activated between the routers. Repeat it on every BGP router to make sure you didn't miss anything:

```
R6#show bgp ipv4 multicast summary

BGP router identifier 150.1.6.6, local AS number 100
BGP table version is 27, main routing table version 27
18 network entries using 4464 bytes of memory
32 path entries using 3712 bytes of memory
12/7 BGP path/bestpath attribute entries using 2880 bytes of memory
1 BGP rrinfo entries using 40 bytes of memory
2 BGP AS-PATH entries using 64 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
```

https://t.me/learningnets

```
BGP using 11160 total bytes of memory
BGP activity 37/0 prefixes, 67/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
150.1.7.7       4          100      85      79       27    0    0 00:51:45        16
155.1.146.4     4          200      83      72       27    0    0 00:51:58         9
```

Check the BGP tables on the border routers to make sure that best paths toward the multicast prefixes are across the Tunnel between R3 and R5:

https://t.me/learningnets

```
R4#show bgp ipv4 multicast regexp 100$

BGP table version is 21, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>i 150.1.3.3/32      150.1.5.5
                0    100      0 100 ?
 *                    155.1.146.6              3             0 100 100 100 100 ?
 *   150.1.6.6/32     155.1.146.6                            0 100 100 100 100 ?
*>i                  150.1.5.5
                3    100      0 100 ?
 *   150.1.7.7/32     155.1.146.6              2             0 100 100 100 100 ?
*>i                  150.1.5.5
                2    100      0 100 ?
 *   150.1.9.9/32     155.1.146.6              3             0 100 100 100 100 ?
*>i                  150.1.5.5
                3    100      0 100 ?
 *   155.1.7.0/24     155.1.146.6              2             0 100 100 100 100 ?
*>i                  150.1.5.5
                2    100      0 100 ?
 *   155.1.9.0/24     155.1.146.6              3             0 100 100 100 100 ?
*>i                  150.1.5.5
                3    100      0 100 ? *>i 155.1.37.0/24    150.1.5.5
                0    100      0 100 ?
 *                    155.1.146.6              2             0 100 100 100 100 ?
 *   155.1.67.0/24    155.1.146.6                            0 100 100 100 100 ?
*>i                  150.1.5.5
                2    100      0 100 ?
 *   155.1.79.0/24    155.1.146.6              2             0 100 100 100 100 ?
*>i                  150.1.5.5
                2    100      0 100 ?
```

Check the best paths in the other AS also:

```
R6#show bgp ipv4 multicast regexp 200$
BGP table version is 27, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
```

https://t.me/learningnets

```
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   150.1.4.4/32     155.1.146.4                            0 200 200 200 200 ?
*>i                   150.1.3.3
                2    100      0 200 ?*>i 150.1.5.5/32     150.1.3.3
                0    100      0 200 ?
 *                   155.1.146.4              2             0 200 200 200 200 ?
*>i 150.1.8.8/32     150.1.3.3
                2    100      0 200 ?
 *                   155.1.146.4              3             0 200 200 200 200 ?
*>i 150.1.10.10/32   150.1.3.3
                3    100      0 200 ?
 *                   155.1.146.4              4             0 200 200 200 200 ?
*>i 155.1.0.0/24     150.1.3.3
                0    100      0 200 ?
 *                   155.1.146.4           1001             0 200 200 200 200 ?
*>i 155.1.8.0/24     150.1.3.3
                2    100      0 200 ?
 *                   155.1.146.4              3             0 200 200 200 200 ?
*>i 155.1.10.0/24    150.1.3.3
                3    100      0 200 ?
 *                   155.1.146.4              4             0 200 200 200 200 ?
*>i 155.1.58.0/24    150.1.3.3
                0    100      0 200 ?
 *                   155.1.146.4              2             0 200 200 200 200 ?
*>i 155.1.108.0/24   150.1.3.3
                2    100      0 200 ?
 *                   155.1.146.4              3             0 200 200 200 200 ?
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

MSDP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Inter Domain Multicast MSDP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast MSDP Diagram** in order to complete this task.

Task

- Configure R5 and R8 as the RPs for AS 200 using BSR, use R10 as the BSR. Ensure that BSR border is configured on the link between the ASs.
- AS100 is pre-configured with a static RP (R7 is the RP for AS 100).
- Create an MSDP peering session between R7 & R8 and R5 & R7, sourcing it off the Loopback0 interfaces.

Configuration

When implementing inter-domain multicast using PIM SM, each domain usually has its own RP. To allow sources and receivers from different domains to locate each other, RPs need to exchange the information about their local active sources. After this information is exchanged between the RPs, all routers that joined the respective shared trees may build shortest-path trees toward the actual sources.

MSDP or Multicast Source Discovery Protocol is used to exchange multicast source information between RPs. It is configured as a TCP connection between the RPs and used to exchange the so-called Source Active (SA) messages. Note that all MSDP peerings are configured manually, using the command `ip msdp peer` at both endpoints. When a source in one PIM SM domain starts sending the multicast traffic, the respective DR will start the registration process with the local RP. When the local RP receives the PIM Register message, it replicates it to all of its MSDP

https://t.me/learningnets

neighbors as an SA message. The SA message contains the IP address of the multicast source as well as the destination group and the IP address of the RP sending the SA message. The latter is known as the MSDP ID and can be changed using the command **`ip msdp originator-id`** .

When any RP receives a new SA message, it determines whether there are local receivers that have joined the shared tree for the encapsulated group. If there are any, the message is forwarded down the tree, allowing the receivers to learn about the sources in another domain. After that, the receivers might join the SPT toward the source in the other domain. This is only possible if the source IP address is learned via BGP or some other inter-domain route exchange procedure. Until the moment there is an active source in a domain, the respective RP will forward periodic SA messages with an empty payload to refresh the active state for this group/source in all other domains.

MSDP allows us to connect RPs in an arbitrary meshed topology, to include loops. To prevent the SA messages from cycling the topology, as a result of these loops, every MSDP peer forwards SA messages only after they pass the RPF check. The RPF check is performed based on the RP IP address (originator-ID) inside the message and the IP address of the MSDP peer that relayed the message. If the MSDP peer is on the shortest path toward the originating RP, the message is accepted; otherwise it is dropped. This RPF check requires full routing information from other domains to discover routes to other RPs. If you have a stub multicast domain, lacking full BGP information, you may use the command **`ip msdp default-peer`** to identify the upstream RP that forwards SA messages. RPF checks are not applied to default peers, and all SA messages are accepted.

Our scenario is a bit tricky, because it has two RPs in AS 200. However, the BSR protocol ensures that all routers in AS 200 will select the same RP for a given group. Therefore, you only need to peer R7 with R8 and R5 via MSDP; there is no need to peer R8 with R5 via MSDP.

```
R5:
interface Loopback0
 ip pim sparse-mode
!
interface Tunnel35
 ip pim bsr-border
!
ip pim rp-candidate Loopback0
ip msdp peer 150.1.7.7 connect-source Loopback0 remote-as 100
R8:
interface Loopback0
 ip pim sparse-mode
!
```

https://t.me/learningnets

```
ip pim rp-candidate Loopback0
ip msdp peer 150.1.7.7 connect-source Loopback0 remote-as 100
R10:
interface Loopback0
 ip pim sparse-mode
!
ip pim bsr-candidate Loopback0
R6:
interface GigabitEthernet1.146
 ip pim bsr-border
R7:

ip msdp peer 150.1.5.5 connect-source Loopback0 remote-as 200
ip msdp peer 150.1.8.8 connect-source Loopback0 remote-as 200
```

Verification

We want to ensure that we have two multicast groups that map to different RPs inside AS 200. To make this happen, we must alter the “rp-hash” value used on R10:

```
R10(config)#no ip pim bsr-candidate loopback0 0
R10(config)#ip pim bsr-candidate loopback0 31
```

Now the groups 239.1.1.1 and 239.1.1.2 map to RPs R5 and R8, respectively:

https://t.me/learningnets

```
R4#show ip pim rp-hash 239.1.1.1
 RP 150.1.5.5 (?), v2

   Info source: 150.1.10.10 (?), via bootstrap, priority 0, holdtime 150
          Uptime: 00:53:54, expires: 00:01:30
   PIMv2 Hash Value (mask 255.255.255.254)    RP 150.1.5.5, via bootstrap, priority 0,
hash value 1362971077

   RP 150.1.8.8, via bootstrap, priority 0, hash value 718054422
!
!R4#show ip pim rp-hash 239.1.1.2
 RP 150.1.8.8 (?), v2

   Info source: 150.1.10.10 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:54:32, expires: 00:01:53
  PIMv2 Hash Value (mask 255.255.255.254)
   RP 150.1.5.5, via bootstrap, priority 0, hash value 443334807
   RP 150.1.8.8, via bootstrap, priority 0, hash value 1364246456
```

Configure speakers on both ASs to join these groups:

```
R4:
interface Loopback0
 ip pim sparse-mode
 ip igmp join-group 239.1.1.1
 ip igmp join-group 239.1.1.2
R9:

interface Loopback0
 ip pim sparse-mode
 ip igmp join-group 239.1.1.1
 ip igmp join-group 239.1.1.2
```

Initially, every router joins the shared tree in its own domain.

```
R9#show ip mroute 239.1.1.1
IP Multicast Routing Table
<snip>
 (*, 239.1.1.1), 00:00:17/00:02:42, RP 150.1.7.7
, flags: SJCL
  Incoming interface: GigabitEthernet1.79, RPF nbr 155.1.79.7
  Outgoing interface list:
```

https://t.me/learningnets

```
    Loopback0, Forward/Sparse, 00:00:17/00:02:42
!
!R9#show ip mroute 239.1.1.2
IP Multicast Routing Table
<snip>
(*, 239.1.1.2), 00:02:10/00:02:28, RP 150.1.7.7
, flags: SJCL
  Incoming interface: GigabitEthernet1.79, RPF nbr 155.1.79.7
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:02:10/00:02:28
!
!R4#show ip mroute 239.1.1.1
IP Multicast Routing Table
<snip>
(*, 239.1.1.1), 00:02:51/00:02:12, RP 150.1.5.5
, flags: SJCL
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.5
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:02:50/00:02:12
!
!R4#show ip mroute 239.1.1.2
IP Multicast Routing Table
<snip>
(*, 239.1.1.2), 00:03:12/00:02:51, RP 150.1.8.8
, flags: SJCL
  Incoming interface: GigabitEthernet1.45, RPF nbr 155.1.45.5
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:03:11/00:02:51
```

Enable MSDP debugging on R7 and start pinging group 239.1.1.1 from R10:

```
R7#debug ip msdp detail
R7#debug ip msdp detail
MSDP Detail debugging is on
R10#ping 239.1.1.1 repeat 1000

Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 239.1.1.1, timeout is 2 seconds:

Reply to request 0 from 155.1.45.4, 59 ms
Reply to request 0 from 150.1.9.9, 75 ms
Reply to request 0 from 150.1.4.4, 64 ms
Reply to request 0 from 150.1.9.9, 64 ms
Reply to request 0 from 155.1.79.9, 64 ms
Reply to request 0 from 150.1.4.4, 59 ms
```

https://t.me/learningnets

```
Reply to request 1 from 155.1.45.4, 69 ms
Reply to request 1 from 150.1.9.9, 93 ms
Reply to request 1 from 155.1.79.9, 93 ms
Reply to request 1 from 150.1.4.4, 69 ms
Reply to request 1 from 150.1.4.4, 69 ms
Reply to request 1 from 150.1.4.4, 69 ms
Reply to request 2 from 150.1.4.4, 29 ms
Reply to request 2 from 150.1.4.4, 66 ms
Reply to request 2 from 150.1.4.4, 66 ms
Reply to request 2 from 155.1.45.4, 66 ms
Reply to request 2 from 150.1.9.9, 45 ms
Reply to request 2 from 150.1.9.9, 29 ms
Reply to request 2 from 155.1.79.9, 29 ms
```

Notice that R7 received Source Active messages for the sources on R10. Because R10 uses all of its PIM-enabled interfaces to source multicast, there are multiple SA messages for every registered source. The actual source is registered with the RP located in AS 200:

```
R7#
!
!
MSDP(0): Received 120-byte TCP segment from 150.1.5.5
MSDP(0): Append 120 bytes to 0-byte msg 14 from 150.1.5.5, qs 1
MSDP(0): WAVL Insert SA Source 155.1.10.10 Group 239.1.1.1 RP 150.1.5.5 Successful
 MSDP(0): Forward decapsulated SA data for (155.1.10.10, 239.1.1.1) on GigabitEthernet1.79
MSDP(0): Received 120-byte TCP segment from 150.1.5.5
MSDP(0): Append 120 bytes to 0-byte msg 15 from 150.1.5.5, qs 1
MSDP(0): WAVL Insert SA Source 155.1.108.10 Group 239.1.1.1 RP 150.1.5.5 Successful
 MSDP(0)  Forward decapsulated SA data for (155.1.108.10, 239.1.1.1) on GigabitEthernet1.79

MSDP(0): Received 120-byte TCP segment from 150.1.5.5
MSDP(0): Append 120 bytes to 0-byte msg 16 from 150.1.5.5, qs 1
MSDP(0): WAVL Insert SA Source 150.1.10.10 Group 239.1.1.1 RP 150.1.5.5 Successful
MSDP(0): Forward decapsulated SA data for (150.1.10.10, 239.1.1.1) on GigabitEthernet1.79
```

Confirm that R9 has joined the SPTs toward the sources in different the AS. Notice that RPF information for these sources is taken from MBGP updates, not the unicast routing table:

```
R9#show ip mroute 239.1.1.1
IP Multicast Routing Table
<snip>
```

https://t.me/learningnets

```
(*, 239.1.1.1), 00:09:53/stopped, RP 150.1.7.7, flags: SJCL
  Incoming interface: GigabitEthernet1.79, RPF nbr 155.1.79.7
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:09:53/00:02:45
 (155.1.108.10, 239.1.1.1
), 00:02:01/00:00:58, flags: LJT  Incoming interface: GigabitEthernet1.79, RPF nbr 155.1.79.7,Mbgp
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:02:01/00:02:45
 (150.1.10.10, 239.1.1.1
), 00:02:21/00:00:38, flags: LJT  Incoming interface: GigabitEthernet1.79, RPF nbr 155.1.79.7,Mbgp
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:02:21/00:02:45
 (155.1.10.10, 239.1.1.1
), 00:02:25/00:00:34, flags: LJT  Incoming interface: GigabitEthernet1.79, RPF nbr 155.1.79.7,Mbgp

  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:02:25/00:02:45
```

Now make sure the SPTs are built across the Tunnel link, because this is the preferred path for multicast traffic. Use the **`show ip mroute`** command to accomplish this:

```
R7#show ip mroute 239.1.1.1
IP Multicast Routing Table
<snip>

(*, 239.1.1.1), 00:08:55/00:03:27, RP 150.1.7.7, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.79, Forward/Sparse, 00:08:55/00:03:27

(150.1.10.10, 239.1.1.1), 00:01:27/00:01:32, flags: MT  Incoming interface:
GigabitEthernet1.37, RPF nbr 155.1.37.3, Mbgp
  Outgoing interface list:
    GigabitEthernet1.79, Forward/Sparse, 00:01:27/00:03:27

(155.1.10.10, 239.1.1.1), 00:01:27/00:01:32, flags: MT  Incoming interface:
GigabitEthernet1.37, RPF nbr 155.1.37.3, Mbgp
  Outgoing interface list:
    GigabitEthernet1.79, Forward/Sparse, 00:01:27/00:03:27

(155.1.108.10, 239.1.1.1), 00:02:06/00:00:53, flags: T  Incoming interface:
GigabitEthernet1.37, RPF nbr 155.1.37.3, Mbgp
  Outgoing interface list:
    GigabitEthernet1.79, Forward/Sparse, 00:02:06/00:03:27
```

https://t.me/learningnets

```
!
!R3#show ip mroute 239.1.1.1
IP Multicast Routing Table
<snip>

(*, 239.1.1.1), 00:06:50/stopped, RP 150.1.7.7, flags: SP
  Incoming interface: GigabitEthernet1.37, RPF nbr 155.1.37.7
  Outgoing interface list: Null
 (155.1.10.10, 239.1.1.1
), 00:00:06/00:02:53, flags: T  Incoming interface:Tunnel35, RPF nbr 155.1.0.5, Mbgp
  Outgoing interface list:
    GigabitEthernet1.37, Forward/Sparse, 00:00:06/00:03:23
 (150.1.10.10, 239.1.1.1
), 00:03:11/00:03:18, flags: T  Incoming interface:Tunnel35, RPF nbr 155.1.0.5, Mbgp
  Outgoing interface list:
    GigabitEthernet1.37, Forward/Sparse, 00:03:11/00:03:14
 (155.1.108.10, 239.1.1.1
), 00:03:50/00:02:39, flags: T  Incoming interface:Tunnel35, RPF nbr 155.1.0.5, Mbgp

  Outgoing interface list:
    GigabitEthernet1.37, Forward/Sparse, 00:03:50/00:02:37
```

You may also use the `mtrace` command to trace the multicast delivery tree from the leaf to the root. The first parameter is the source address and the second parameter is the destination group. This command queries the neighbors for the upstream multicast path and tells you the method used for RPF check at every router. Notice that inside AS 100, the RPF checks are performed using MBGP:

```
R7#mtrace 150.1.10.10 239.1.1.1
Type escape sequence to abort.
Mtrace from 150.1.10.10 to 155.1.37.7 via group 239.1.1.1
From source (?) to destination (?)
Querying full reverse path...
 0  155.1.37.7-1  155.1.37.7 ==> 155.1.37.7 PIM_MT Reached RP/Core
 [150.1.10.10/32]-2  155.1.37.3 ==> 155.1.0.3 PIM_MT
  [150.1.10.10/32]-3  155.1.0.5 ==> 155.1.58.5 [AS 200] PIM Reached RP/Core
 [150.1.10.10/32]
-4  155.1.58.8 ==> 155.1.108.8 [AS 200] PIM  [150.1.10.10/32]
-5  155.1.108.10 ==> 150.1.10.10 [AS 200] PIM_MT  [150.1.10.10/32]
```

You may now repeat the tests for the group 239.1.1.2 and see that it works as well:

```
R10#ping 239.1.1.2 repeat 1000
Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 239.1.1.2
```

https://t.me/learningnets

```
Reply to request 0 from 155.1.45.4, 130 ms
Reply to request 0 from 150.1.9.9, 151 ms
Reply to request 0 from 150.1.9.9, 151 ms
Reply to request 0 from 155.1.79.9, 151 ms
Reply to request 0 from 150.1.4.4, 130 ms
Reply to request 0 from 150.1.4.4, 130 ms
Reply to request 1 from 155.1.45.4, 114 ms
Reply to request 1 from 150.1.9.9, 129 ms
Reply to request 1 from 155.1.79.9, 129 ms
Reply to request 1 from 150.1.4.4, 118 ms
Reply to request 1 from 150.1.4.4, 114 ms
Reply to request 1 from 150.1.4.4, 114 ms
Reply to request 2 from 155.1.45.4, 39 ms
Reply to request 2 from 150.1.9.9, 54 ms
Reply to request 2 from 150.1.9.9, 54 ms
Reply to request 2 from 155.1.79.9, 54 ms
Reply to request 2 from 150.1.4.4, 40 ms
Reply to request 2 from 150.1.4.4, 40 ms
Reply to request 2 from 150.1.4.4, 40 ms
!
!R9#mtrace 150.1.10.10 239.1.1.2
Type escape sequence to abort.
Mtrace from 150.1.10.10 to 155.1.79.9 via group 239.1.1.2
From source (?) to destination (?)
Querying full reverse path...
 0  155.1.79.9 -1 155.1.79.9 ==> 155.1.79.9 PIM_MT
  [150.1.10.10/32] -2 155.1.79.7 ==> 155.1.37.7 PIM_MT Reached RP/Core
 [150.1.10.10/32] -3 155.1.37.3 ==> 155.1.0.3 PIM_MT
  [150.1.10.10/32]
-4  155.1.0.5 ==> 155.1.58.5 [AS 200] PIM  [150.1.10.10/32]
-5  155.1.58.8 ==> 155.1.108.8 [AS 200] PIM Reached RP/Core [150.1.10.10/32]
-6  155.1.108.10 ==> 150.1.10.10 [AS 200] PIM_MT  [150.1.10.10/32]
```

Take a look at the MSDP cache on R7. The current active sources on the network are sending multicast traffic to 239.1.1.2. Recall that we stopped 239.1.1.1 before starting 239.1.1.2:

https://t.me/learningnets

```
R7#show ip msdp sa-cache

MSDP Source-Active Cache - 3 entries
(150.1.10.10, 239.1.1.2), RP 150.1.8.8, MBGP/AS 200, 00:16:42/00:05:28, Peer 150.1.8.8
(155.1.10.10, 239.1.1.2), RP 150.1.8.8, MBGP/AS 200, 00:16:44/00:05:28, Peer 150.1.8.8
(155.1.108.10, 239.1.1.2), RP 150.1.8.8, MBGP/AS 200, 00:16:44/00:05:28, Peer 150.1.8.8
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Anycast RP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Initial Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** in order to complete this task.

Task

- Enable PIM sparse-mode multicast delivery on the Ethernet path between R6 and R10.
  - Do not enable PIM over the DMVPN network netween R4 and R5.
- Configure R4 and R8 as cRPs and R5 as the BSR.
- Instead of relying on the BSR protocol to distribute the load between the RPs, implement a solution that hides both RPs behind the same RP IP address **150.1.100.100**.
  - The RPs should inform each other of the active sources registered with them.

Configuration

Anycast RP is a special RP redundancy scenario that allows using redundant RPs sharing the *same* IP address. Here, Anycast means that groups of RPs use the same IP address used by all multicast routers in the domain to build shared trees. However, the PIM Joins are being sent to the closest RP, based on the unicast routing table. Thus, different routers might join shared trees rooted at different RPs. At the same time, different DRs will pick up different physical RPs based on the anycast address to register their local sources.

To maintain consistent source information, MSDP sessions should be configured between the RPs. This will ensure that all routers joining different RPs will still have

https://t.me/learningnets

full information about all potential sources in the domain. Thus, the following are the guidelines to configure Anycast RP:

- Use the same IP address on all routers as the candidate RP IP address. Propagate this information via BSR or Auto-RP.
- Using different IP addresses on every router, source MSDP sessions and link all candidate RPs in a mesh. Note that you might need to manually specify the MSDP originator ID to be different on every RP, or the MSDP sessions won't come up.

```
R4:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim sparse-mode
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
interface Loopback100
 ip address 150.1.100.100 255.255.255.255
 ip pim sparse-mode
!
router eigrp 100
 network 150.1.100.100 0.0.0.0
!
ip msdp originator-id Loopback0
ip msdp peer 150.1.8.8 connect-source Loopback0
!
ip pim rp-candidate Loopback100
R6:
ip multicast-routing distributed
!
interface GigabitEthernet1.146
 ip pim sparse-mode
R5:
ip multicast-routing distributed
!
interface GigabitEthernet1.45
 ip pim sparse-mode
!
interface GigabitEthernet1.58
```

https://t.me/learningnets

```
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim bsr-candidate Loopback0
```

R8:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.58
 ip pim sparse-mode
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
interface Loopback100
 ip address 150.1.100.100 255.255.255.255
 ip pim sparse-mode
!
router eigrp 100
 network 150.1.100.100 0.0.0.0
!
ip msdp originator-id Loopback0
ip msdp peer 150.1.4.4 connect-source Loopback0
!
ip pim rp-candidate Loopback100
```

R10:

```
ip multicast-routing distributed
!
interface GigabitEthernet1.108
 ip pim sparse-mode
!
interface GigabitEthernet1.10
 ip pim sparse-mode
```

Verification

https://t.me/learningnets

Join receivers on R10 to the multicast group 239.1.1.1. Verify that R10 actually uses the Anycast RP IP address as its RP:

```
R10:
interface Loopback0
 ip pim sparse-mode
 ip igmp join-group 239.1.1.1
!
!R10#show ip mroute 239.1.1.1
IP Multicast Routing Table
<snip>
(*, 239.1.1.1), 00:00:11/00:02:48, RP 150.1.100.100
, flags: SJCL
  Incoming interface: GigabitEthernet1.108, RPF nbr 155.1.108.8
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:00:11/00:02:48
```

Source multicast from R6 and confirm that it actually reaches the receivers:

```
R6#ping 239.1.1.1 repeat 1000
Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 239.1.1.1, timeout is 2 seconds:
...Reply to request 3 from 155.1.108.10, 13 ms
Reply to request 3 from 150.1.10.10, 13 ms

Reply to request 4 from 150.1.10.10, 85 ms
Reply to request 4 from 150.1.10.10, 85 ms
Reply to request 4 from 155.1.108.10, 85 ms
Reply to request 5 from 150.1.10.10, 20 ms
Reply to request 5 from 150.1.10.10, 20 ms
Reply to request 6 from 150.1.10.10, 33 ms
Reply to request 6 from 150.1.10.10, 33 ms
```

Look at the SA caches of R8:

```
R8#show ip msdp sa-cache

MSDP Source-Active Cache - 2 entries
(150.1.6.6, 239.1.1.1), RP 150.1.4.4, AS ?,00:00:11/00:05:48, Peer 150.1.4.4
(155.1.146.6, 239.1.1.1), RP 150.1.4.4, AS ?,00:00:11/00:05:48, Peer 150.1.4.4
```

The closest RP to R6 is R4, so R4 receives the PIM Register message and sends an MSDP SA to R8. Since R8 as a receiver for 239.1.1.1 (R10 joined this group), it

https://t.me/learningnets

send the encapsulated packet to R10. R10 then builds the SPT tree back to R6 directly. Join 239.1.1.2 on R6 and send traffic to that group from R10:

```
R6:

interface Loopback0
 ip pim sparse-mode
 ip igmp join-group 239.1.1.2
```

Verify that R6 is using R4 as the RP by checking R4 for the (*,G) for 239.1.1.2:

```
R4#show ip mroute 239.1.1.2
<snip>
(*, 239.1.1.2), 00:01:06/00:03:23, RP 150.1.100.100
, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.146, Forward/Sparse, 00:01:06/00:03:23
```

Now source traffic from R10. R10 should register with R8, then R8 will send an MSDP SA message to R4, which will then send the encapsulated packet to R6. At this point, R6 will join the SPT towards R10 directly:

```
R10#ping 239.1.1.2 repeat 1000

Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 239.1.1.2, timeout is 2 seconds:
Reply to request 0 from 155.1.146.6, 74 ms
Reply to request 0 from 150.1.6.6, 74 ms

Reply to request 0 from 150.1.6.6, 74 ms
Reply to request 1 from 155.1.146.6, 17 ms
Reply to request 1 from 150.1.6.6, 17 ms
Reply to request 2 from 155.1.146.6, 18 ms
Reply to request 2 from 150.1.6.6, 18 ms
```

Check the MSDP cache on R4:

```
R4#show ip msdp sa-cache

MSDP Source-Active Cache - 3 entries
(150.1.10.10, 239.1.1.2), RP 150.1.8.8, AS ?,00:00:36/00:05:31, Peer 150.1.8.8
(155.1.10.10, 239.1.1.2), RP 150.1.8.8, AS ?,00:00:38/00:05:31, Peer 150.1.8.8
```

https://t.me/learningnets

```
(155.1.108.10, 239.1.1.2), RP 150.1.8.8, AS ?,00:00:38/00:05:31, Peer 150.1.8.8
```

This type of setup allows us to have RP redundancy/load sharing. If one RP fails, all multicast routers are already pointing to the Anycast address. As soon as the routing protocol converges, the multicast routers will be able to begin using the other RP. At the same time, while both RPs are up, multicast routers throughout the network will use whichever RP is closest (based on routing protocol metrics). Anycast is used for other networking applications. DNS servers across the globe (for example, Google's DNS servers), use Anycast by sharing the same IP address so that clients making DNS queries reach the closest DNS server from their location. Anycast is also used in Data Center fabrics leveraging FabricPath, where up to four FabricPath nodes share the same HSRP address vMAC/IP. Instead of only one of them being active and the rest standby/listen, all four devices in the HSRP group are able to respond to traffic sent to the HSRP vMAC (active-active), giving the fabric optimal layer-3 forwarding.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Catalyst IGMP Snooping

> You must load the initial configuration files for the section, **Catalyst Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Configure SW1's Fa0/20 interface as an access port for VLAN 146. Configure SW3's side of this link as a Layer-3 interface with the IP address **155.1.146.20/24**.
- Configure SW2's Fa0/20 interface as an access port for VLAN 146. Configure SW4's side of this link as a Layer-3 interface with the IP address **155.1.146.40/24**.
- Join the multicast group **239.1.1.100** on the Layer-3 interface of SW3 and SW4.
- Configure the SW1 and SW2 to assist in optimal traffic flooding across the switched topology for VLAN 146.
- Ensure that a switch stops flooding out of a given port as soon as it receives an IGMPv2 report.

Configuration

Multicast poses special difficulties in switched Ethernet networks. By default, IP Multicast addresses are mapped to Ethernet multicast MAC addresses according to the following procedure:

1. The multicast Ethernet MAC address range starts at 01:00:5E:00:00:00 and goes through 01:00:5E:7F:FF:FF. The highest bit of the 4th byte in these addresses is fixed at 0, so only low-order 23 bits can vary. This allocation is historical.
2. Based on this range, the low-order 23 bits of the multicast IP address are mapped into the low-order 23 bits of the MAC address.
3. The high order 4 bits of the Layer 3 IP address is fixed to 1110 to indicate the Class

https://t.me/learningnets

D address space between 224.0.0.0 and 239.255.255.255. Thus, 28 bits remain for IP Multicast group numbering.

Therefore, there are 2^5 multicast groups mapped to the same multicast MAC address. However, this is only a part of the puzzle. As you remember, Ethernet switches flood multicast traffic out of all ports by default. In heavily loaded networks, this might result in excessive bandwidth usage. To overcome this issue, it's possible for switches to listen to IGMP and PIM messages received on Layer 2 ports. Based on the information snooped from these messages, switches may selectively prune some ports from unneeded multicast traffic. This procedure is called IGMP snooping and allows for selective multicast flooding in switched networks.

Notice that to work effectively, IGMP snooping must be implemented on Layer 3 capable switches. This is because Layer 2 devices cannot distinguish IGMP and PIM packets from any other multicast frames and must process all multicast traffic at the CPU level. Thus, a Layer 2 switch's performance might be severely affected by intense multicast flows.

IGMP snooping is enabled by default on Catalyst multi-layer switches. Snooping is enabled globally and on a per-VLAN basis. If you want to disable IGMP snooping globally on all VLANs, use the command **`no ip igmp snooping`** . Use the command **`no ip igmp snooping vlan <VLAN-ID>`** to disable it for a single VLAN. Generally, switches automatically discover switchports connected to multicast-capable routers by listening to PIM and DVMRP messages, and flood all multicast groups on such ports. If you want to statically configure a port as connected to a multicast router, use the command **`ip igmp snooping vlan <vlan-id> mrouter interface <interface-id>`** .

If your switch has just one host connected to every Layer 2 switch-port, you may want to enable the IGMP Snooping Immediate Leave feature using the command **`ip igmp snooping vlan <vlan-id> immediate-leave`** . When you enable this feature, the switch will immediately remove a port from the flood list for a given group after it receives the IGMPv2 Leave message on an interface that is part of a particular group.

```
SW1:
ip igmp snooping vlan 146 immediate-leave
!
interface FastEthernet0/20
 switchport access vlan 146
SW2:
ip igmp snooping vlan 146 immediate-leave
!
interface FastEthernet0/20
 switchport access vlan 146
SW3:
```

https://t.me/learningnets

```
ip multicast-routing distributed
!
interface FastEthernet0/20
 no switchport
 ip address 155.1.146.20 255.255.255.0
 ip pim dense-mode
 ip igmp join-group 239.1.1.100
SW4:

ip multicast-routing distributed
!
interface FastEthernet0/20
 no switchport
 ip address 155.1.146.40 255.255.255.0
 ip pim dense-mode
 ip igmp join-group 239.1.1.100
```

Verification

First, check the IGMP Snooping general settings for VLAN 146. Notice that “IGMPv2 immediate leave” is enabled for this VLAN.

https://t.me/learningnets

```
SW1#show ip igmp snooping vlan 146

Global IGMP Snooping configuration:
-------------------------------------------IGMP snooping                  : Enabled

IGMPv3 snooping (minimal)    : Enabled
Report suppression           : Enabled
TCN solicit query            : Disabled
TCN flood query count        : 2
Robustness variable          : 2
Last member query count      : 2
Last member query interval   : 1000

Vlan 146:
--------IGMP snooping                          : Enabled

CAPWAP enabled                       : DisabledIGMPv2 immediate leave               : Enabled
Multicast router learning mode       : pim-dvmrp

CGMP interoperability mode           : IGMP_ONLY
Robustness variable                  : 2
Last member query count              : 2
Last member query interval           : 1000
```

Join SW3's Layer 3 FastEthernet0/20 interface to the IGMP group 239.1.1.100 and check that IGMP snooping actually processes the IGMP packets. From the command’s output on SW1, you can see that there are two ports in the flood list for the group, one for the receiver and the other for the other multicast router (SW4), across the trunk. The same output can be observed from SW2; it has two ports in its flood list, one for its "local" receiver and the other for SW3 across the trunk.

```
SW3:
interface FastEthernet 0/20
 ip igmp join-group 239.1.1.100

SW1#show ip igmp snooping groups vlan 146
Vlan      Group                    Type        Version     Port List
-----------------------------------------------------------------------
146       224.0.1.40               igmp        v2          Fa0/20, Fa0/24
146       239.1.1.100              igmp        v2          Fa0/20, Fa0/24
!
!SW1#show ip igmp snooping mrouter vlan 146
Vlan    ports
----    -----
```

https://t.me/learningnets

```
146     Fa0/20(dynamic), Fa0/24(dynamic)

!
!SW4#show ip igmp snooping groups vlan 146
Vlan      Group                    Type        Version     Port List
------------------------------------------------------------------------
146       224.0.1.40               igmp        v2          Fa0/20, Fa0/24
146       239.1.1.100              igmp        v2          Fa0/20, Fa0/24
!
!SW4#show ip igmp snooping mrouter vlan 146
Vlan    ports
----    -----146     Fa0/20(dynamic), Fa0/24(dynamic)
```

Note that FastEthernet0/24 on SW1 and SW2 is being used as a trunk to provide transport for VLAN 146. This can be verified by looking at spanning-tree.

```
SW1#show spanning-tree vlan 146

VLAN0146
  Spanning tree enabled protocol ieee
  Root ID    Priority    24722
             Address     001c.576d.4a00
             Cost        19
             Port        26 (FastEthernet0/24)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32914  (priority 32768 sys-id-ext 146)
             Address     000a.b832.3580
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Desg FWD 19        128.22   P2p
 Fa0/24              Root FWD 19        128.26   P2p

!
!SW2#show spanning-tree vlan 146

VLAN0146
  Spanning tree enabled protocol ieee
  Root ID    Priority    24722
             Address     001c.576d.4a00
             This bridge is the root
```

https://t.me/learningnets

```
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    24722  (priority 24576 sys-id-ext 146)
             Address     001c.576d.4a00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Desg FWD 19        128.22   P2p
Fa0/24              Desg FWD 19        128.26   P2p
```

Ping the multicast group from SW3.

```
SW3#ping 239.1.1.100 repeat 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 239.1.1.100, timeout is 2 seconds:
Reply to request 0 from 155.1.146.20, 1 ms
Reply to request 0 from 155.1.146.40, 1 ms

Reply to request 1 from 155.1.146.20, 1 ms
Reply to request 1 from 155.1.146.40, 1 ms
Reply to request 2 from 155.1.146.20, 1 ms
Reply to request 2 from 155.1.146.40, 1 ms
```

Multicast router information is learned via PIM messages. There are two multicast routers on VLAN 146—SW3 and SW4. Confirm that SW4 actually receives the IGMP report from SW3.

```
SW4#show ip igmp groups
IGMP Connected Group Membership
Group Address    Interface                Uptime    Expires   Last Reporter   Group Accounted
239.1.1.100      FastEthernet0/20         00:18:57  00:02:42  155.1.146.40

224.0.1.40       FastEthernet0/20         00:18:56  00:02:49  155.1.146.20
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Catalyst Multicast VLAN Registration

> You must load the initial configuration files for the section, **Catalyst Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Configure SW1's Fa0/20 interface as an access port for VLAN 146. Configure SW3's side of this link as a Layer 3 interface with the IP address **155.1.146.20/24**.
- Configure SW1's Fa0/22 interface as an access port for VLAN 58. Configure SW4's side of this link as a Layer 3 interface with the IP address **155.1.58.40/24**.
- Join the multicast group **239.1.1.100** on the Layer 3 interface of SW3 and SW4.
- Configure SW1 so that multicast traffic sent by SW3 on VLAN 146 is received by SW4 on VLAN 58.
- Use the configuration that allows SW3 to dynamically track sources in other VLANs.

Configuration

Multicast VLAN registration (MVR) is a special type of multicast traffic delivery suited to the access layer of metro Ethernet networks, especially for ring topologies. In these networks, it is common to allocate a VLAN based on a network's geographic region. In this configuration, when you send a video multicast feed to receivers on multiple VLANs, the feed will be replicated for every VLAN across the ring topology, causing bandwidth overutilization.

MVR uses a single dedicated VLAN across the whole ring to deliver a multicast feed to all receivers. The actual receivers reside in different VLANs, but the switches intercept their IGMP Join requests and pull the multicast feed from the MVR VLAN to the receiver VLAN. Thus, MVR allows multicast traffic to cross VLAN boundaries based on client IGMP reports. This function is similar to IGMP snooping, but the two

https://t.me/learningnets

work independently. You can have both features enabled on the switch, but MVR will only inspect IGMP reports for groups explicitly configured to be supported by MVR.

There are two modes for the MVR feature: dynamic and compatible. Before we look at the differences, notice that MVR and multicast routing are mutually exclusive. When MVR is enabled, the switch acts like a Layer 2 device with respect to multicast traffic, and multicast routing should remain disabled. However, the switch still inspects IGMP messages and may forward them to the port whether a multicast router is connected or not. Back to the modes: The first mode is called dynamic. This mode allows the multicast router connected to the switch ring to listen to the IGMP messages and dynamically create respective mroute states. That is, IGMP join messages from the clients are forwarded to the multicast source port, allowing the broadcast of unneeded channels to be stopped. The other mode, called compatible (the default in switches), inspects the IGMP messages but does not forward them to multicast router ports. This was the default mode of operation of the MVR feature in 3500XL switches. If you use compatible mode, make sure you configure static IGMP joins on the multicast router to feed the multicast streams down. The default mode assumes that the source constantly sends down all multicast channels, and does not process client IGMP joins.

Configure MVR by following these steps:

Step 1: Enable MVR globally using the commands `mvr` and `mvr group <mcast-group> <count>` . This will enable the MVR feature for the particular group or for the number of groups specified by the `<count>` argument and starting at `<mcast-group>` address. Note that you cannot have more than 256 addresses configured for MVR, and they should never alias. Here, aliasing means that two multicast IP addresses map to the same multicast MAC address (see the previous task). For example, addresses 228.1.1.1 and 230.1.1.1 would alias to the same MAC address and would be rejected.

Step 2: Define the MVR VLAN. This is the VLAN that will span multiple switches and carry the actual multicast traffic feed. This VLAN should be allowed on all trunks to permit multicast delivery. The command is `mvr vlan <vlan-id>` . You may also define the MVR mode by using the command `mvr mode [dynamic|compatible]` .

Step 3: Configure the source and receiver interfaces, using the interface-mode command `mvr type {source|receiver}` . Additionally, you may configure ports for immediate leave, using the command `mvr immediate` . When this command is enabled, any single IGMP leave would cause the switch to prune the port from receiving multicast feeds. This is good for the ports that have only one host connected.

Step 4: Optionally, configure static group joins using the command `mvr vlan <vlan-id> group <ip-address>`

https://t.me/learningnets

on receiver ports. This is similar to the static join command configured on a router, but it allows pulling multicast traffic from the MVR VLAN. Another optional command is **`mvr querytime <1/10 of second>`**. This command is similar to the **`igmp query-max-response-time`** configured on routers. The switch passively listens to IGMP general queries sent from multicast routers, and then starts the **`querytime`** timer, waiting for client replies. If no replies are received during the interval, the switch prunes the port from the list of output ports for the group. Note that you cannot configure trunk ports as MVR receivers.

```
SW1:
no ip multicast-routing distributed
mvr
mvr vlan 146
mvr group 239.1.1.100
mvr mode dynamic
!
interface FastEthernet 0/20
 switchport access vlan 146
 mvr type source
!
interface FastEthernet 0/22
 switchport access vlan 58
 mvr type receiver
SW3:
ip multicast-routing distributed
!
interface FastEthernet0/20
 no switchport
 ip address 155.1.146.20 255.255.255.0
 ip pim dense-mode
 ip igmp join-group 239.1.1.100
SW4:

ip multicast-routing distributed
!
interface FastEthernet0/22
 no switchport
 ip address 155.1.58.40 255.255.255.0
 ip pim dense-mode
 ip igmp join-group 239.1.1.100
```

https://t.me/learningnets

Verification

Check the MVR settings on SW1. Because the mode is dynamic, SW3 should receive IGMP Joins from SW4.

```
SW1#show mvr
 MVR Running: TRUE
MVR multicast VLAN: 146
MVR Max Multicast Groups: 256
MVR Current multicast groups: 1
MVR Global query response time: 5 (tenths of sec)MVR Mode: dynamic
!
!SW1#show mvr interface

Port      Type       Status           Immediate Leave
----      ----       ------           ---------------
Fa0/20    SOURCE     ACTIVE/UP        DISABLED
Fa0/22    RECEIVER   ACTIVE/UP        DISABLED
!
!SW1#show mvr members

MVR Group IP        Status         Member          Membership
------------------------------------------------------------
239.001.001.100     ACTIVE/UP      Fa0/20          Dynamic
239.001.001.100     ACTIVE/UP      Fa0/22          Dynamic
!
!SW3#show ip igmp groups

IGMP Connected Group Membership
Group Address    Interface                Uptime    Expires   Last Reporter   Group Accounted
239.1.1.100      FastEthernet0/20         00:05:32  00:02:19  155.1.58.40

224.0.1.40       FastEthernet0/20         00:05:31  00:02:05  155.1.146.20
```

Source the multicast from SW3 and verify that SW4 feed actually receives the multicast packets by logging the traffic with an ACL.

```
SW4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW4(config)#access-list 100 permit ip host 155.1.146.20 host 239.1.1.100 log-input
 SW4(config)#access-list 100 permit ip any any
SW4(config)#interface FastEthernet0/22
SW4(config-if)#ip access-group 100 in
```

https://t.me/learningnets

```
!
!SW3#ping 239.1.1.100 repeat 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 239.1.1.100, timeout is 2 seconds:...
!
!SW4#
%SEC-6-IPACCESSLOGDP: list 100 permitted icmp 155.1.146.20 (FastEthernet0/22 000a.b832.3580) ->
239.1.1.100
 (0/0), 1 packet
!
!SW4#show ip access-list
Extended IP access list 100 10 permit ip host 155.1.146.20 host 239.1.1.100 log-input (156 matches)

    20 permit ip any any (3 matches)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Multicast

Catalyst IGMP Profiles

> You must load the initial configuration files for the section, **Catalyst Multicast**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs Multicast Diagram** to complete this task.

Task

- Configure SW1's Fa0/20 interface as an access port for VLAN 146. Configure SW3's side of this link as a Layer-3 interface with the IP address **155.1.146.20/24**.
- Configure SW1 to permit only the IGMP reports for groups in ranges **232.0.0.0/8** and **239.0.0.0/8** from SW3.

Configuration

Catalyst switches allow filtering of IGMP messages sent by directly connected hosts to multicast routers. This feature is similar to the `ip igmp access-group` command used on routers, but this command applies to transit IGMP messages. This functionality is accomplished by using IGMP profiles. IGMP profiles have global numbers, and every profile defines a list of ranges plus the accompanying operation: “permit” or “deny” (default). When using "permit," the switch permits IGMP reports for the groups specified by the range and denies everything else. When using "deny," the switch denies the group range configured and permits everything else. To configure the profile, use the command:

https://t.me/learningnets

```
ip igmp profile <global-number>
 range low-address1 [high-address1]
 range low-address2 [high-address2]
 ...
 [permit|deny]
```

The profile applies ingress to Layer 2 ports only using the interface-level command **`ip igmp filter <number>`** and affects all IGMP reports sent by hosts connected to the particular port.

```
SW1:
ip igmp profile 1
  permit
  range 232.0.0.0 232.255.255.255
  range 239.0.0.0 239.255.255.255
!
interface FastEthernet 0/20
 switchport access vlan 146
 ip igmp filter 1
SW3:

ip multicast-routing distributed
!
interface FastEthernet0/20
 no switchport
 ip address 155.1.146.20 255.255.255.0
 ip pim dense-mode
```

Verification

Join SW3's Layer 3 FastEthernet0/20 interface to a couple of groups. One group should match the profile criteria, and the other should not. After this, verify what IGMP groups are seen on the switch, using the IGMP snooping function.

https://t.me/learningnets

SW3:

```
interface FastEthernet0/20
 ip igmp join-group 239.4.4.4
 ip igmp join-group 230.4.4.4
!
!SW1#show ip igmp snooping groups vlan 146

Vlan      Group                    Type        Version     Port List
---------------------------------------------------------------------
146       239.4.4.4                igmp        v2          Fa0/20
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Link-Local Addressing

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure link-local IPv6 addresses on the Ethernet link between R1, R4, and R6.
- Use the IPv6 addressing format **FE80::Y**, where *Y* is the router number.

Configuration

```
R1:
interface GigabitEthernet1.146
 ipv6 address fe80::1 link-local
R4:
interface GigabitEthernet1.146
 ipv6 address fe80::4 link-local
R6:

interface GigabitEthernet1.146
 ipv6 address fe80::6 link-local
```

Verification

https://t.me/learningnets

Link-local IPv6 addresses are significant only within the context of a single link. This means that packets with link-local addresses cannot be routed between interfaces, and link-local addresses may overlap as long as they exist on different interfaces. The address format for IPv6 link-local addresses is FE80::/10.

Packets with link-local sources or destinations are mostly used by the router's control plane protocols, such as IPv6 routing protocols. For broadcast segments, such as Ethernet, link-local reachability is implicit because of automatic resolution through ICMP Neighbor Discovery (ICMPv6). When pinging link-local addresses, it is necessary to tell the router which interface to send traffic out. Because the link-local address can exist on multiple interfaces, the router requires you to indicate the "Output Interface." You must specify the interface using the full interface name without spaces (for example, GigabitEthernet1.146).

```
R1#show ipv6 interface brief

GigabitEthernet1       [up/up]
unassigned
GigabitEthernet1.13    [up/up]
FE80::250:56FF:FE8D:7819
2001:155:1:13::1
GigabitEthernet1.100   [up/up]
FE80::250:56FF:FE8D:7819
2001:169:254:100::1GigabitEthernet1.146   [up/up]
FE80::1
2001:155:1:146::1
!
!R1#ping ipv6 fe80::4
Output Interface: GigabitEthernet1.146
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FE80::4, timeout is 2 seconds:
Packet sent with a source address of FE80::1%GigabitEthernet1.146!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/4/9 ms
!
!R1#ping ipv6 fe80::6
   Output Interface: GigabitEthernet1.146
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FE80::6, timeout is 2 seconds:
Packet sent with a source address of FE80::1%GigabitEthernet1.146!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 5/12/33 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Unique Local Addressing

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure ULA IPv6 addressing as follows:
  - Use **FC00:155:0:45::Y/64** for Ethernet link between R4 and R5.
  - Use **FC00:155:0:37::Y/64** for Ethernet link between R3 and R7.
  - In both cases, **Y** represents the router number.

Configuration

```
R4:
interface GigabitEthernet1.45
 ipv6 address FC00:155:0:45::4/64
R5:
interface GigabitEthernet1.45
 ipv6 address FC00:155:0:45::5/64
R3:
interface GigabitEthernet1.37
 ipv6 address FC00:155:0:37::3/64
R7:

interface GigabitEthernet1.37
 ipv6 address FC00:155:0:37::7/64
```

https://t.me/learningnets

Verification

Unique Local IPv6 Unicast Addressing (ULA), defined in RFC 4193, deprecates the previously used Site-Local (FEC0::/10) addressing. ULA addresses in IPv6 are synonymous with the RFC 1918 private addresses found in IPv4, that is, the 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16 prefixes. Considered private, ULA addresses are not publicly routable prefixes on the Internet. The format of the ULA is:

```
FC00 (7 bits) + Unique ID (41 bits) + Link ID (16 bits) + Interface ID (64 bits).
```

The randomly generated Unique ID helps avoid address collisions. Other than the addressing format, ULA addressing exhibits no other unique behavior when compared to normal publicly routable IPv6 addresses.

```
R3#show ipv6 interface brief

GigabitEthernet1       [up/up]
unassigned
GigabitEthernet1.13    [up/up]
FE80::250:56FF:FE8D:3357
2001:155:1:13::3
GigabitEthernet1.23    [administratively down/down]
FE80::250:56FF:FE8D:3357
2001:155:1:23::3
GigabitEthernet1.37    [up/up]
FE80::250:56FF:FE8D:3357
2001:155:1:37::3 FC00:155:0:37::3
!
!R4#show ipv6 interface brief

GigabitEthernet1       [up/up]
unassigned
GigabitEthernet1.45    [up/up]
FE80::250:56FF:FE8D:4949
2001:155:1:45::4 FC00:155:0:45::4
GigabitEthernet1.67    [deleted/down]
unassigned
GigabitEthernet1.100   [up/up]
FE80::250:56FF:FE8D:4949
2001:169:254:100::4
!
!R3#ping FC00:155:0:37::7
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FC00:155:0:37::7, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/13/41 ms
!
!R4#ping FC00:155:0:45::5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FC00:155:0:45::5, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 3/13/47 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Global Aggregatable Addressing

> You must load the initial configuration files for the section, **IPv6 Global Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure globally aggregatable IPv6 addresses as follows:
  - Use **2001:155:1:1234::Y/64** on the Tunnel interfaces of R1, R2, and R5.
  - Here, **Y** is your router number.

Configuration

```
R1:
interface Tunnel0
 ipv6 address 2001:155:1:1234::1/64
R2:
interface Tunnel0
 ipv6 address 2001:155:1:1234::2/64
b#R5:

interface Tunnel0
 ipv6 address 2001:155:1:1234::5/64
```

Verification

Globally Aggregatable IPv6 addresses, also called global unicast addresses, are publicly allocated and routable on the Internet. Global unicast addresses start with the binary prefix 001 (2000::/3) and therefore encompass the range 2000:: – 3FFF::.

https://t.me/learningnets

By design, the allocation of these addresses through Internet registries such as ARIN and APNIC are hierarchical, unlike IPv4 allocation. RIPE-450 *(IPv6 Address Allocation and Assignment Policy)* defines this hierarchical design methodology.

Although this range is extremely large (1/8th of the total IPv6 address space), generally only the prefix 2001::/16 is currently used for allocation. Other ranges, such as the 6to4 Tunnel range of 2002::/16, are generally used for special purposes. When assigned, the behavior of a global unicast address is identical to that of a ULA address, but as its name implies, it has global routing significance. As we have configured IPv6 global unicast address on the DMVPN tunnel interface, make sure to verify NHRP mapping before testing reachablity from both the spokes R1 and R2 to the Hub.

```
R1#show ipv6 interface brief | begin Tunnel0
    Tunnel0                [up/up]
    FE80::21E:BDFF:FEE7:1800 2001:155:1:1234::1
    !
    !R1#show ipv6 nhrp
     2001:155:1:1234::/64 via 2001:155:1:1234::5
    Tunnel0 created 01:18:13, never expire
    Type: static, Flags: used     NBMA address: 169.254.100.5

    !
    !R1#ping  2001:155:1:1234::5
    Type escape sequence to abort.
    Sending 5, 100-byte ICMP Echos to 2001:155:1:1234::5, timeout is 2 seconds: !!!!!
    Success rate is 100 percent (5/5), round-trip min/avg/max = 3/7/13 ms
    !
    !R2#show ipv6 interface brief | begin Tunnel0
    Tunnel0                [up/up]
    FE80::21E:BDFF:FEE7:1800 2001:155:1:1234::2
    !
    !R2#show ipv6 nhrp
                           2001:155:1:1234::/64 via 2001:155:1:1234::5
    Tunnel0 created 00:56:00, never expire
    Type: static, Flags: used     NBMA address: 169.254.100.5

    !
    !R2#ping 2001:155:1:1234::5
    Type escape sequence to abort.
    Sending 5, 100-byte ICMP Echos to 2001:155:1:1234::5, timeout is 2 seconds: !!!!!

    Success rate is 100 percent (5/5), round-trip min/avg/max = 2/4/6 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 EUI-64 Addressing

> You must load the initial configuration files for the section, **IPv6 Global Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure globally aggregatable IPv6 addresses as follows:
  - Use **2001:155:1:146::Y/64** on the VLAN 146 interfaces of R1, R4, and R6.
  - Here, **Y** is based on the interface MAC address.

Configuration

```
R1:
interface GigabitEthernet1.146
 ipv6 address 2001:155:1:146::/64 eui-64
R4:
interface GigabitEthernet1.146
 ipv6 address 2001:155:1:146::/64 eui-64
R6:

interface GigabitEthernet1.146
 ipv6 address 2001:155:1:146::/64 eui-64
```

Verification

Extended Unique Identifiers (EUIs) are 64-bit values assigned to physical interfaces. EUIs are similar in many respects to IEEE MAC addresses, in that they identify a physical interface, but EUIs are designed to be used universally, not just with the

https://t.me/learningnets

hardware addresses. IPv6 uses EUI-64 to construct a unique host address on a shared Ethernet segment automatically. When you use the **`eui-64`** keyword, the IOS uses the 48-bit hardware address of the Ethernet interface as the foundation to construct the unique 64-bit interface identifier of the IPv6 address.

Specifically, this is accomplished by inverting the 7th-most-significant bit of the Ethernet MAC address, called the Universal/Local (U/L) bit, and then inserting 16 bits of padding in the format FFFE in the middle. From the below output, we can see that the EUI-64-derived host address of R1's GigabitEthernet1.146 interface is 250:56FF:FE8D:7819. This means that the MAC address of R1's GigabitEthernet1 interface is 0050.568d.7819. Note that the link-local address of the interface automatically uses the format FE80::[EUI-64] unless manually modified.

```
R1#sh ipv6 interface brief
GigabitEthernet1       [up/up]
unassigned
GigabitEthernet1.13    [up/up]
FE80::250:56FF:FE8D:7819
2001:155:1:13::1
GigabitEthernet1.100   [up/up]
FE80::250:56FF:FE8D:7819
2001:169:254:100::1
GigabitEthernet1.146   [up/up]
FE80::1 2001:155:1:146::1
!
!R1#sh ipv6 interface GigabitEthernet 1.146
GigabitEthernet1.146 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::1
No Virtual link-local address(es):
 Global unicast address(es): 2001:155:1:146:250:56FF:FE8D:7819, subnet is 2001:155:1:146::/64 [EUI]

Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ICMP unreachables are sent
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds (using 30000)
ND advertised reachable time is 0 (unspecified)
ND advertised retransmit interval is 0 (unspecified)
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
ND advertised default router preference is Medium
```

https://t.me/learningnets

```
Hosts use stateless autoconfig for addresses.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Auto-Configuration

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Use the unique local address **FC00:1:1:58::5/64** for R5's VLAN 58 interface.
  - Configure an additional IPv6 address of **FC00:1:1:85::5/64** on the same interface of R5.
- Configure R5 to multicast both prefixes using ICMPv6 ND RAs, but only allow the hosts to use the second prefix for auto-configuration.
  - The valid and preferred lifetime for both prefixes should be set to 4 hours.
  - R5 should advertise itself as the default router on the segment every 40 seconds with a lifetime interval of 60 seconds.
- R8 should learn its IPv6 address on VLAN58 automatically and use R5 as its default gateway.

Configuration

```
R5:
ipv6 unicast-routing
!
interface GigabitEthernet1.58
 ipv6 address fc00:1:1:58::5/64
 ipv6 address fc00:1:1:85::5/64
 ipv6 nd prefix fc00:1:1:58::/64 14400 14400 no-autoconfig
 ipv6 nd prefix fc00:1:1:85::/64 14400 14400
 ipv6 nd ra-interval 40
 ipv6 nd ra-lifetime 60
```

https://t.me/learningnets

```
 no ipv6 nd suppress-ra
R8:

interface GigabitEthernet1.58
 ipv6 address autoconfig default
```

Verification

IPv6 has a special feature called auto-configuration. It replaces many functions served by DHCP in IPv4 networks (yet there is a special DHCPv6 edition of the DHCP protocol). With IPv6 auto-configuration, an IPv6 host may automatically learn the IPv6 prefixes assigned to the local segment, as well as determine the default routers on that segment. A special type of link-local IPv6 addressing and the ICMPv6 ND (Neighbor Discovery) protocol accomplish this.

Router Advertisements can be sent at a specified interval, or the feature can be suppressed completely ("suppress-ra" is enabled by default on Ethernet interfaces; this default behavior can be changed with the no ipv6 nd suppress-ra), depending on how you configure your Cisco router. Note that you must enable IPv6 unicast routing to send router advertisements. The following commands control the IPv6 RA announcements:

- **`ipv6 nd ra-interval`** specifies the periodic interval to send RAs.
- **`ipv6 nd ra-lifetime`** specifies the validity interval of the router's IPv6 address.
- **`ipv6 nd prefix`** manipulates the IPv6 network prefixes included into RA. By default, all prefixes are included.

You may adjust the interval that the prefix is valid and preferred with every command. Additionally, you may instruct the hosts not to use this prefix for auto-configuration by using the **`no-autoconfig`** keyword.

```
R8#show ipv6 interface GigabitEthernet1.58
GigabitEthernet1.58 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::250:56FF:FE8D:3558
No Virtual link-local address(es):
Stateless address autoconfig enabled
Global unicast address(es):
    2001:155:1:58::8, subnet is 2001:155:1:58::/64
    2001:155:1:58:250:56FF:FE8D:3558, subnet is 2001:155:1:58::/64 [EUI/CAL/PRE]
  valid lifetime 2591974 preferred lifetime 604774
FC00:1:1:85:250:56FF:FE8D:3558, subnet is FC00:1:1:85::/64 [EUI/CAL/PRE]
  valid lifetime 14374 preferred lifetime 14374
Joined group address(es):
```

https://t.me/learningnets

```
    FF02::1
    FF02::2
    FF02::1:FF00:8
    FF02::1:FF8D:3558
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
!
!R8#sh ipv6 route
IPv6 Routing Table - default - 12 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - Application ND  ::/0 [2/0]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
!
!    R5#debug ipv6 nd

ICMPv6-ND: (GigabitEthernet1.58) Sending solicited RA
ICMPv6-ND: (GigabitEthernet1.58,FE80::250:56FF:FE8D:74DD) send RA to FF02::1
ICMPv6-ND: (GigabitEthernet1.58,FE80::250:56FF:FE8D:74DD) Sending RA (60) to FF02::1
ICMPv6-ND:   MTU = 1500
ICMPv6-ND:   prefix 2001:155:1:58::/64 [LA] 2592000/604800
ICMPv6-ND:   prefix FC00:1:1:58::/64 [L] 14400/14400
ICMPv6-ND:   prefix FC00:1:1:85::/64 [LA] 14400/14400
!
!R5#show ipv6 interface GigabitEthernet1.58 prefix

IPv6 Prefix Advertisements GigabitEthernet1.58
Codes for 1st column:
   A - Address, P - Prefix-Advertisement, O - Pool
   U - Per-user prefix
Codes for 2nd column and above:
   D - Default
   N - Not advertised, C - Calendar

PD default [LA] Valid lifetime 2592000, preferred lifetime 604800
AD 2001:155:1:58::/64 [LA] Valid lifetime 2592000, preferred lifetime 604800 PA FC00:1:1:58::/64
 [L] Valid lifetime 14400, preferred lifetime 14400 PA FC00:1:1:85::/64
 [LA] Valid lifetime 14400, preferred lifetime 14400
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

RIPng

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Enable RIPng on the VLAN 146 segment connecting R1, R4, and R6 using the process name **RIPNG**.
- Create new Loopback100 interfaces on R1, R4, and R6 with the IPv6 addresses **FC00:1:1:Y::Y/64**, where *Y* is the router number.
  - Advertise Loopback100 interfaces into RIPng.

Configuration

```
R1:
ipv6 unicast-routing
!
interface GigabitEthernet1.146
 ipv6 rip RIPNG enable
!
interface Loopback100
 ipv6 address fc00:1:1:1::1/64
 ipv6 rip RIPNG enable
R4:
ipv6 unicast-routing
!
interface GigabitEthernet1.146
 ipv6 rip RIPNG enable
!
interface Loopback100
```

https://t.me/learningnets

```
 ipv6 address fc00:1:1:4::4/64
 ipv6 rip RIPNG enable
R6:

ipv6 unicast-routing
!
interface GigabitEthernet1.146
 ipv6 rip RIPNG enable
!
interface Loopback100
 ipv6 address fc00:1:1:6::6/64
 ipv6 rip RIPNG enable
```

Verification

To configure RIPng, simply enable it on the respective interfaces. Global RIPng parameters are configured in the routing process sub-configuration mode by issuing the **`ipv6 rip [pid]`** command in global configuration. With the exception of the ip keyword being replaced by ipv6, many of the routing verification commands are similar between IPv4 and IPv6.

```
R6#show ipv6 protocols

    IPv6 Routing Protocol is "connected"
    IPv6 Routing Protocol is "application"
    IPv6 Routing Protocol is "ND"
    IPv6 Routing Protocol is "rip RIPNG"
Interfaces: Loopback100
 GigabitEthernet1.146
Redistribution:
    None
!
!    R6#show ipv6 rip RIPNG
RIP process "RIPNG", port 521, multicast-group FF02::9, pid 503
 Administrative distance is 120. Maximum paths is 16
 Updates every 30 seconds, expire after 180
 Holddown lasts 0 seconds, garbage collect after 120
    Split horizon is on; poison reverse is off
    Default routes are not generated
    Periodic updates 9, trigger updates 2
    Full Advertisement 0, Delayed Events 0
Interfaces:
    Loopback100
    GigabitEthernet1.146
```

https://t.me/learningnets

```
Redistribution:
    None
!
!R6#show ipv6 route rip

IPv6 Routing Table - default - 10 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
        B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
        I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
        EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
        NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
        OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
        ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   FC00:1:1:1::/64 [120/2]
        via FE80::250:56FF:FE8D:7819, GigabitEthernet1.146R   FC00:1:1:4::/64 [120/2]
    via FE80::250:56FF:FE8D:4949, GigabitEthernet1.146
!
!R6#debug ipv6 rip
RIPng [default VRF]: a message has been received.
RIPng [Gi1.146, default VRF]: response received from FE80::250:56FF:FE8D:4949 for process "RIPNG".
   src=FE80::250:56FF:FE8D:4949 (GigabitEthernet1.146)
   dst=FF02::9
   sport=521, dport=521, length=52
   command=2, version=1, mbz=0, #rte=2 tag=0, metric=1, prefix=2001:155:1:146::/64
 tag=0, metric=1, prefix=FC00:1:1:4::/64
!
!
RIPng [default VRF]: a message has been received.
RIPng [Gi1.146, default VRF]: response received from FE80::250:56FF:FE8D:7819 for process "RIPNG".
   src=FE80::250:56FF:FE8D:7819 (GigabitEthernet1.146)
   dst=FF02::9
   sport=521, dport=521, length=52
   command=2, version=1, mbz=0, #rte=2 tag=0, metric=1, prefix=2001:155:1:146::/64
 tag=0, metric=1, prefix=FC00:1:1:1::/64
!
!
RIPng [Lo100, default VRF]: process "RIPNG" is sending a multicast update.
   src=FE80::21E:BDFF:FEE7:1800
   dst=FF02::9 (Loopback100)
   sport=521, dport=521, length=92
   command=2, version=1, mbz=0, #rte=4 tag=0, metric=1, prefix=2001:155:1:146::/64
 tag=0, metric=1, prefix=FC00:1:1:6::/64
 tag=0, metric=2, prefix=FC00:1:1:4::/64
 tag=0, metric=2, prefix=FC00:1:1:1::/64
!
!
RIPng [Gi1.146, default VRF]: process "RIPNG" is sending a multicast update.
```

https://t.me/learningnets

```
    src=FE80::250:56FF:FE8D:6E90
    dst=FF02::9 (GigabitEthernet1.146)
    sport=521, dport=521, length=52
    command=2, version=1, mbz=0, #rte=2 tag=0, metric=1, prefix=2001:155:1:146::/64
 tag=0, metric=1, prefix=FC00:1:1:6::/64
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

RIPng over NBMA

> You must load the initial configuration files for the section, **IPv6 NBMA Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Enable RIPng on the Tunnel interfaces of R1, R2, and R5 using the process name **RIPNG**.
- Ensure connectivity between Loopback0 prefixes of R1, R2 and R5.

Configuration

```
R1:
ipv6 unicast-routing
!
interface Tunnel0
 ipv6 rip RIPNG enable
!
interface Loopback0
 ipv6 rip RIPNG enable
R2:
ipv6 unicast-routing
!
interface Tunnel0
 ipv6 rip RIPNG enable
!
interface Loopback0
 ipv6 rip RIPNG enable
R5:
```

https://t.me/learningnets

```
ipv6 unicast-routing
!
ipv6 router rip RIPNG
 no split-horizon
!
interface Tunnel0
 ipv6 rip RIPNG enable
!
interface Loopback0
 ipv6 rip RIPNG enable
```

Verification

NBMA networks pose two potential issues to IPv6 routing. First, unmapped link-local IPv6 addresses can cause incorrect processing of routing updates. These updates will not be processed correctly because in NBMA networks, the next-hop will always be a link-local address.

Second, partial mesh connectivity might prevent some nodes from receiving routing updates. Commonly, this problem arises in Hub-and-Spoke topologies, where spokes do not receive routing information from other spokes. To handle this issue, disable the split-horizon function in RIPng. You can only disable split-horizon globally, under the routing process, not on a per-interface basis. Alternatively, you may want to use either default routing or summarization at the hub router to prevent the loss of reachability from taking place. In our case, we will simply disable split-horizon on R5.

```
R5#sh ipv6 route rip

IPv6 Routing Table - default - 24 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   2001:150:1:1::1/128 [120/2]
        via FE80::1, Tunnel0R   2001:150:2:2::2/128 [120/2]
        via FE80::2, Tunnel0
!
!R5#debug ipv6 rip

RIPng [default VRF]: a message has been received.RIPng [Tu0, default VRF]: response received from
FE80::1
 for process "RIPNG"
```

https://t.me/learningnets

```
   src=FE80::1 (Tunnel0)
   dst=FF02::9
   sport=521, dport=521, length=132
   command=2, version=1, mbz=0, #rte=6 tag=0, metric=1, prefix=2001:150:1:1::1/128
!
!
RIPng [default VRF]: a message has been received.RIPng [Tu0, default VRF]: response received from
FE80::2
 for process "RIPNG".
   src=FE80::2 (Tunnel0)
   dst=FF02::9
   sport=521, dport=521, length=52
   command=2, version=1, mbz=0, #rte=2 tag=0, metric=1, prefix=2001:150:2:2::2/128
!
!R1#show ipv6 route rip
IPv6 Routing Table - default - 16 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   2001:150:2:2::2/128 [120/3]
        via FE80::5, Tunnel0R   2001:150:5:5::5/128 [120/2]
        via FE80::5, Tunnel0
!
!R1#ping 2001:150:2:2::2
    Type escape sequence to abort.
    Sending 5, 100-byte ICMP Echos to 2001:150:2:2::2, timeout is 2 seconds: !!!!!
    Success rate is 100 percent (5/5), round-trip min/avg/max = 3/5/12 ms
!
!R1#ping 2001:150:5:5::5
    Type escape sequence to abort.
    Sending 5, 100-byte ICMP Echos to 2001:150:5:5::5, timeout is 2 seconds: !!!!!

    Success rate is 100 percent (5/5), round-trip min/avg/max = 7/9/14 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

RIPng Summarization

> You must load the initial configuration files for the section, **RIPng Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Create a new Loopback100 interface on R1 and R4 as follows:
  - Use IPv6 address **FC00:1:0:Y::Y/64**, where **Y** is the router number.
- Advertise Loopback100 prefixes into RIPng.
- Ensure that R5 advertises to R8 only the summary prefix for the Loopback100 prefixes of R1 and R4.

Configuration

```
R1:
interface Loopback100
 ipv6 address FC00:1:0:1::1/64
 ipv6 rip RIPNG enable
R4:
interface Loopback100
 ipv6 address FC00:1:0:4::4/64
 ipv6 rip RIPNG enable
R5:

interface GigabitEthernet1.58
 ipv6 rip RIPNG summary fc00:1::/61
```

https://t.me/learningnets

Verification

To summarize RIPng routes, use the interface-level command **`ipv6 rip [pid] summary`** . Use the standard rules of summarization, converting IPv6 prefixes from hex format into binary format. In our case, we have two prefixes:

```
FC00:1:0:1::/64
FC00:1:0:4::/64
```

The prefixes differ in the 4th position (remember that each block of the IPv6 address is 16 bits):

```
0000 0000 0000 0**001**
0000 0000 0000 0**100**
```

To summarize, move the prefix length 3 bit positions to the left. The resulting prefix is FC00:1::/61.

https://t.me/learningnets

```
R5#sh ipv6 route rip

IPv6 Routing Table - default - 21 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   FC00:1:0:1::/64 [120/2]

        via FE80::1, Tunnel0R   FC00:1:0:4::/64 [120/2]

        via FE80::2, Tunnel0
!
!R8#sh ipv6 route rip

IPv6 Routing Table - default - 20 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - Application
R   2001:155:1:1234::/64 [120/2]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58R   FC00:1::/61 [120/3]

        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

RIPng Prefix Filtering

> You must load the initial configuration files for the section, **RIPng Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R5 to filter R1's Loopback0 IPv6 prefix from entering the local routing table.
  - Permit any other IPv6 prefixes.

Configuration

```
R5:

ipv6 prefix-list FILTER_R1_LO0 deny  2001:150:1:1::1/128
ipv6 prefix-list FILTER_R1_LO0 permit ::/0 le 128
!
ipv6 router rip RIPNG
 distribute-list prefix-list FILTER_R1_LO0 in
```

Verification

RIPng uses IPv6 prefix-lists to filter routing updates. You apply prefix-lists either inbound or outbound under the RIPng process configuration mode. You may choose to associate an interface with the distribute-list, or apply it to all interfaces simultaneously by *not* specifying an interface. Before applying the filter:

```
R5#sh ipv6 route
```

https://t.me/learningnets

```
IPv6 Routing Table - default - 21 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   2001:150:1:1::1/128 [120/2]
        via FE80::1, Tunnel0R   2001:150:2:2::2/128 [120/2]

        via FE80::2, Tunnel0
```

After the filter is applied, it may take some time for the routes to flush out of the routing table. This process can be sped up by clearing the RIP process with the **`clear ipv6 rip [pid]`** command. The route will no longer appear after the table ages out the old prefix.

```
R5#sh ipv6 prefix-list

ipv6 prefix-list FILTER_R1_LO0: 2 entries
    seq 5 deny 2001:150:1:1::1/128
    seq 10 permit ::/0 le 128
!
!R5#sh ipv6 route rip
IPv6 Routing Table - default - 20 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   2001:150:2:2::2/128 [120/2]

        via FE80::2, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

RIPng Metric Manipulation

> You must load the initial configuration files for the section, **RIPng Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Enable RIPng on the Ethernet link between R4 and R5.
  - Ensure that R1 prefers the VLAN146 path to reach Loopback0 interface of R5.
  - If R1 loses its VLAN146 link, it should follow the DMVPN path.

Configuration

```
R4:
interface GigabitEthernet1.45
 ipv6 rip RIPNG enable
!R5:
interface GigabitEthernet1.45
 ipv6 rip RIPNG enable
! R1:

interface Tunnel0
 ipv6 rip RIPNG metric-offset 3
```

Verification

RIPng does not have granular metric manipulation mechanisms. As an example, we can only apply metric offsets to all routes received via a particular interface, thus

https://t.me/learningnets

allowing us to define the outgoing interface preference. Note that by specifying a metric offset that results in a total metric of 16 or higher, we can also *filter* routes using this feature. Before the metric-offset:

```
R1#sh ipv6 route 2001:150:5:5::5/128
Routing entry for 2001:150:5:5::5/128 Known via "rip RIPNG", distance 120, metric 2
    Route count is 1/1, share count 0
    Routing paths: FE80::5, Tunnel0

  Last updated 00:00:29 ago
```

After the metric offset is applied:

```
R1#sh ipv6 route 2001:150:5:5::5/128
Routing entry for 2001:150:5:5::5/128 Known via "rip RIPNG", distance 120, metric 3
    Route count is 1/1, share count 0
Routing paths: FE80::250:56FF:FE8D:4949, GigabitEthernet1.146

  Last updated 00:00:02 ago
```

You can traceroute the prefix to verify the current path:

```
R1#traceroute 2001:150:5:5::5
    Type escape sequence to abort.
    Tracing the route to 2001:150:5:5::5 1 2001:155:1:146::4 16 msec 6 msec 6 msec
 2 2001:155:1:45::5 9 msec 141 msec 7 msec
```

Verify fail-over shutting VLAN 146 interface down:

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface GigabitEthernet1.146
R1(config-subif)#shutdown
!
!R1#traceroute 2001:150:5:5::5
Type escape sequence to abort.
Tracing the route to 2001:150:5:5::5 1 2001:155:1:1234::5 14 msec 10 msec 9 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

RIPng Default Routing

> You must load the initial configuration files for the section, **RIPng Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R6 to advertise a default route on VLAN146 via RIPng with an initial metric of 5.

Configuration

```
R6:

interface GigabitEthernet1.146
 ipv6 rip RIPNG default-information originate metric 5
```

Verification

RIPng allows the sending of a default route out of any router interface. Additionally, you may specify the default route's metric. By adding the keyword `only`, you may further configure the router to send just the default route exclusively out the particular interface, and filter all other subnet advertisements.

```
R4#sh ipv6 route rip
IPv6 Routing Table - default - 20 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
```

https://t.me/learningnets

```
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   ::/0 [120/6]
        via FE80::250:56FF:FE8D:6E90, GigabitEthernet1.146
!
!R4#show ipv6 route 2001:150:6:6::6
Routing entry for ::/0Known via "rip RIPNG", distance 120, metric 6
Route count is 1/1, share count 0
Routing paths:FE80::250:56FF:FE8D:6E90, GigabitEthernet1.146
  Last updated 00:02:07 ago
!
!R4#ping 2001:150:6:6::6
 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to 2001:150:6:6::6, timeout is 2 seconds:!!!!!

 Success rate is 100 percent (5/5), round-trip min/avg/max = 3/29/89 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

EIGRPv6

> You must load the initial configuration files for the section, **EIGRPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Enable EIGRPv6 100 on the tunnel interface of R1, R2, and R5.
  - Advertise Loopback0 prefixes.
- Authenticate the EIGRPv6 adjacency over DMVPN using the password value **CISCO**.

Configuration

EIGRPv6 is an IPv6 routing protocol that was added to Cisco IOS starting with version 12.4T. It mimics the functionality and convergence algorithm of the classic IPv4 EIGRP protocol, and many configuration commands are similar to their IPv4 counterparts. Even the administrative distances for IPv6 EIGRP internal and external routes are the same. To enable the EIGRPv6 routing process, use the following global commands:

```
ipv6 router eigrp N
 no shutdown
```

https://t.me/learningnets

The IPv6 version of EIGRP uses AS numbers to identify routing processes. The second command is necessary to bring the process up; this feature is not found in other IPv6 routing protocols. After enabling the process, you may select the interfaces that you want to advertise into EIGRPv6 by using the command **`ipv6 eigrp N`** , where *N* is the AS number you configured on the routing process. Just like with IPv4 EIGRP, you may use the interface commands **`ipv6 hold-time`** and **`ipv6 hello-interval`** to change the EIGRPv6 interface timers.

Similar to IPv4 EIGRP, enabling authentication on the interface is done in three stages: create a key-chain, set the interface authentication mode, and apply the key-chain.

```
key chain EIGRPV6
 key 1
  key-string CISCO
!
interface GigabitEthernet1
 ipv6 eigrp 100
 ipv6 authentication mode eigrp 100 md5
 ipv6 authentication key-chain eigrp 100 EIGRPV6
```

Notice that EIGRPv6 is still a distance-vector protocol; this means that the classic split horizon rule still applies. When deployed in hub-and-spoke topologies such as on a Frame-Relay, DMVPN interface, you may want to use the command **`no ipv6 split-horizon eigrp 100`** to disable the split-horizon rule on a particular interface. This is unlike RIPng, which only allows disabling split-horizon globally. It is interesting to note that EIGRPv6 also supports the interface-level command **`no ipv6 next-hop-self eigrp`** . This command, used on the hub router, explicitly sets the next-hop field in the relayed EIGRPv6 updates to the spoke router's IP address. DMVPN deployments commonly use this feature.

```
R5:
ipv6 unicast-routing
ipv6 router eigrp 100
 no shutdown
!
key chain EIGRPV6
 key 1
  key-string CISCO
!
interface Tunnel0
 ipv6 eigrp 100
 ipv6 authentication mode eigrp 100 md5
 ipv6 authentication key-chain eigrp 100 EIGRPV6
```

https://t.me/learningnets

```
!
interface Loopback0
 ipv6 eigrp 100
R1:
ipv6 unicast-routing
ipv6 router eigrp 100
 no shutdown
 !
key chain EIGRPV6
 key 1
  key-string CISCO
!
interface Tunnel0
 ipv6 eigrp 100
 ipv6 authentication mode eigrp 100 md5
 ipv6 authentication key-chain eigrp 100 EIGRPV6
!
interface Loopback0
 ipv6 eigrp 100
R2:

ipv6 unicast-routing
ipv6 router eigrp 100
 no shutdown
!
key chain EIGRPV6
 key 1
  key-string CISCO
!
interface Tunnel0
 ipv6 eigrp 100
 ipv6 authentication mode eigrp 100 md5
 ipv6 authentication key-chain eigrp 100 EIGRPV6
!
interface Loopback0
 ipv6 eigrp 100
```

Verification

Start by checking the IPv6 EIGRP process settings and protocol adjacencies.

```
R5#sh ipv6 eigrp 100 interfaces
EIGRP-IPv6 Interfaces for AS(100)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface             Peers   Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Tu0                     2        0/0          0/0         40       9/2624        192          0
Lo0                     0        0/0          0/0          0        0/0            0          0
```

https://t.me/learningnets

```
!
!R5#sh ipv6 protocols

IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "application"
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "rip RIPNG"
Interfaces:
    Tunnel0
Redistribution:
    None
IPv6 Routing Protocol is "NHRP-IPv6"
IPv6 Routing Protocol is "eigrp 100" EIGRP-IPv6 Protocol for AS(100)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
NSF-aware route hold timer is 240
EIGRP NSF disabled
 NSF signal timer is 20s
 NSF converge timer is 120s
Router-ID: 150.1.5.5
Topology : 0 (base)
Active Timer: 3 min
Distance: internal 90 external 170
Maximum path: 16
Maximum hopcount 100
Maximum metric variance 1

Interfaces: Tunnel0
Loopback0
Redistribution:
    None
!
!    R5#show ipv6 eigrp neighbors

EIGRP-IPv6 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   Link-local address:     Tu0                      11 00:07:44   23  5000  0  12 FE80::2
0   Link-local address:     Tu0                      12 00:07:44   58  5000  0  32 FE80::1
```

Check for the IPv6 routes, and verify whether the Loopback0 networks are advertised.

```
R5#sh ipv6 route eigrp
```

https://t.me/learningnets

```
IPv6 Routing Table - default - 15 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - ApplicationD   2001:150:1:1::1/128 [90/27008000]
        via FE80::1, Tunnel0D   2001:150:2:2::2/128 [90/27008000]
        via FE80::2, Tunnel0
!
!R5#ping 2001:150:1:1::1 source Loopback0
 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to 2001:150:1:1::1, timeout is 2 seconds:
 Packet sent with a source address of 2001:150:5:5::5 !!!!!
 Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/7 ms
!
!R5#ping 2001:150:2:2::2 source Loopback0
 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to 2001:150:2:2::2, timeout is 2 seconds:
 Packet sent with a source address of 2001:150:5:5::5 !!!!!

 Success rate is 100 percent (5/5), round-trip min/avg/max = 2/18/63 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

EIGRPv6 Summarization

> You must load the initial configuration files for the section, **EIGRPv6 Basic**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure Loopback1 interface on R1 and R2 using addressing in the format of **FC00:1:1:Y::Y/64**, where *Y* is the router number.
  - After advertising these into EIGRPv6, configure R5 to send a summary route to R8 for both prefixes.
- Upon completing the task, make sure that both the prefixes are reachable from R8.

Configuration

Thanks to EIGRPv6's distance-vector behavior, you may summarize routing information at any point in the network. More specifically, you may configure the routing process to summarize prefixes advertised out any interface using the command `ipv6 summary-address eigrp N` . EIGRPv6 follows the same summarization rules we used in the previous RIPng scenarios. In our case, we have two prefixes:

```
FC00:1:1:1::/64 and FC00:1:1:2::/64
```

Breaking down 1 and 2 into binary digit notation (remember that every component is 16 bits):

```
0000 0000 0000 00**01** = 1
0000 0000 0000 00**10** = 2
```

https://t.me/learningnets

We can see that to remove the “dissimilar” parts, we must shift the prefix length two bits to the left. Thus, the summary prefix is:

```
FC00:1:1::/62
```

Remember to configure summarization on the exit interface connecting R5 to R8 so that no specific routing information leaks to R8. Note that unlike IPv4 EIGRP, you cannot associate a leak-map with the summary prefix and allow specific routing information to leak out, unless running IOS code 15.4(1)T minimum.

```
R1:
interface Loopback1
 ipv6 address FC00:1:1:1::1/64
 ipv6 eigrp 100
R2:
interface Loopback1
 ipv6 address FC00:1:1:2::2/64
 ipv6 eigrp 100
R5:

interface GigabitEthernet1.58
 ipv6 summary-address eigrp 100 FC00:1::/62
```

Verification

Check R5’s routing table for EIGRP routes, and notice that individual prefixes for R1’s and R2’s Loopback1 interfaces are present.

```
R5#sh ipv6 route eigrp

IPv6 Routing Table - default - 15 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - Application
D   2001:150:1:1::1/128 [90/27008000]
        via FE80::1, Tunnel0
D   2001:150:2:2::2/128 [90/27008000]
        via FE80::2, Tunnel0D   FC00:1:1:1::/64 [90/27008000]
```

https://t.me/learningnets

```
        via FE80::1, Tunnel0D   FC00:1:1:2::/64 [90/27008000]

        via FE80::2, Tunnel0
```

Check R8's routing table for EIGRP routes, and notice that individual prefixes for R1's and R2's Loopback1 interfaces are not present. Instead, it has summary route.

```
R8#sh ipv6  route eigrp

IPv6 Routing Table - default - 14 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - Application
D   2001:150:1:1::1/128 [90/27008256]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
D   2001:150:2:2::2/128 [90/27008256]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
D   2001:150:5:5::5/128 [90/130816]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58D   FC00:1:1::/62 [90/27008256]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
!
!        R8#ping  FC00:1:1:1::1
 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to FC00:1:1:1::1, timeout is 2 seconds:!!!!!
 Success rate is 100 percent (5/5), round-trip min/avg/max = 3/8/15 ms
!
!R8#ping FC00:1:1:2::2

 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to FC00:1:1:2::2, timeout is 2 seconds:!!!!!

 Success rate is 100 percent (5/5), round-trip min/avg/max = 4/13/24 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

EIGRPv6 Prefix Filtering

> You must load the initial configuration files for the section, **EIGRPv6 Basic**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R5 to filter IPv6 prefix corresponding to R1's Loopback0 from entering the local routing table.
  - Permit any other IPv6 prefixes.

Configuration

As with any other distance-vector protocol, you may filter EIGRPv6 routing information at any point in the network. Use the routing process-command `distribute-list prefix-list <NAME> in|out [Interface]` to filter incoming or outgoing updates, respectively. You may apply the list to an interface or configure it to filter in/out all directions by omitting the interface name. Notice that because of its distance-vector nature, a filtered prefix will not appear on any router that is learning the route from the filtering node solely.

```
R5:

 ipv6 prefix-list R1_LOOP0 seq 10 deny 2001:150:1:1::1/128
 ipv6 prefix-list R1_LOOP0 seq 20 permit ::/0 le 128
!
ipv6 router eigrp 100
 distribute-list prefix-list R1_LOOP0 in
```

https://t.me/learningnets

Verification

Verify that EIGRPv6 no longer learns about R1's Loopback0, and ensure that other prefixes are not affected. Before filtering applied:

```
R5#sh ipv6 route eigrp

IPv6 Routing Table - default - 18 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - Application   D   2001:150:1:1::1/128 [90/27008000]
        via FE80::1, Tunnel0D   2001:150:2:2::2/128 [90/27008000]

        via FE80::2, Tunnel0
```

After filtering applied:

```
R5#show ipv6 prefix-list

ipv6 prefix-list R1_LOOP0: 2 entries seq 10 deny 2001:150:1:1::1/128
    seq 20 permit ::/0 le 128
!
!R5#sh ipv6 route eigrp

IPv6 Routing Table - default - 17 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - ApplicationD   2001:150:2:2::2/128 [90/27008000]

        via FE80::2, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

EIGRPv6 Metric Manipulation

> You must load the initial configuration files for the section, **EIGRPv6 Basic**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R5 to use only delay value to reach the Loopback0 prefixes of R1 and R2.

Configuration

EIGRPv6 uses the same automatic metric computation as EIGRP for IPv4. Thus, you may tune interface bandwidth and delay to affect the metric for the incoming updates. You may change the K-values vector by using the routing process command `metric weight` . Where IPv4 EIGRP supports unequal-cost load balancing, IPv6 EIGRP can only implement equal-cost load balancing as a result of existing IPv6 CEF limitations. In the future, these limitations may be lifted. Right now, however, you can instruct EIGRPv6 to account for paths with unequal metrics, but it will only load balance across them equally. This is accomplished by using the routing-process `variance` command to allow the process to install the routes with unequal costs into the routing table. Just as usual, paths selected for load balancing will include any feasible successor path with a metric less than the minimal metric scaled by the variance parameter, up to the maximum number of allowed paths.

```
R1, R2 & R5:

ipv6 router eigrp 100
 metric weight 0 0 0 1 0 0
```

https://t.me/learningnets

Verification

Check the K-values settings.

```
R5#show ipv6 protocols

IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "application"
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "rip RIPNG"
Interfaces:
    Tunnel0
Redistribution:
    None
IPv6 Routing Protocol is "NHRP-IPv6"
IPv6 Routing Protocol is "eigrp 100"
EIGRP-IPv6 Protocol for AS(100)Metric weight K1=0, K2=0, K3=1, K4=0, K5=0

NSF-aware route hold timer is 240
EIGRP NSF disabled
    NSF signal timer is 20s
    NSF converge timer is 120s
Router-ID: 150.1.5.5
Topology : 0 (base)
    Active Timer: 3 min
Distance: internal 90 external 170
Maximum path: 16
    Maximum hopcount 100
    Maximum metric variance 1
```

Check the routing table after manipulating metric.

```
R5#sh ipv6  route eigrp

IPv6 Routing Table - default - 17 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - Application D   2001:150:1:1::1/128 [90/1408000
]
        via FE80::1, Tunnel0
```

https://t.me/learningnets

```
D   2001:150:2:2::2/128 [90/1408000
]
        via FE80::2, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

EIGRPv6 Default Routing

> You must load the initial configuration files for the section, **EIGRPv6 Basic**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R5 to advertise a default route into EIGRPv6 via its DMVPN interface.

Configuration

There are two ways to advertise a default route into an IPv6 EIGRP routing domain: by using redistribution or by using summarization. The first technique will originate the route as EIGRP an external route with an AD of 170 (one of the least preferred). The second technique will summarize all routes, because the default route encompasses all of them, and there is no leak-map in EGRPv6 yet to “unsuppress” certain prefixes. Thus, both techniques have drawbacks. Also notice that using redistribution allows for specifying the default route metric explicitly, whereas summarization computes the metric automatically.

In this task, we use the summarization approach and summarize all prefixes under ::/0 on R5’s tunnel interface.

```
R5:

interface Tunnel0
 ipv6 summary-address eigrp 100 ::/0 5
```

https://t.me/learningnets

Verification

Make sure that the default route propagates throughout the EIGRPv6 topology.

```
R1#sh ipv6 route eigrp

IPv6 Routing Table - default - 13 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - ApplicationD   ::/0 [90/26880256]
        via FE80::5, Tunnel0
!
!R2# show ipv6 route eigrp

IPv6 Routing Table - default - 11 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - ApplicationD   ::/0 [90/26880256]

        via FE80::5, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

OSPFv3

> You must load the initial configuration files for the section, **OSPFv3 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure OSPFv3 on VLAN146 interfaces of R1, R4, and R6 as follows:
  - Manually configure the router identifiers to match the IPv4 Loopback0 addresses.
  - Advertise Loopback0 prefixes into area 0.
  - Configure hello and dead intervals to 5 and 20 seconds.
  - Ensure that R1 will always be the DR for VLAN146 segment.

Configuration

```
R1:
ipv6 unicast-routing
!
ipv6 router ospf 1
 router-id 150.1.1.1
!
interface GigabitEthernet1.146
 ipv6 enable
 ipv6 ospf 1 area 0
 ipv6 ospf hello-interval 5
 ipv6 ospf dead-interval 20
!
interface Loopback0
 ipv6 ospf 1 area 0
R4:
```

https://t.me/learningnets

```
ipv6 unicast-routing
!
ipv6 router ospf 1
 router-id 150.1.4.4
!
interface GigabitEthernet1.146
 ipv6 enable
 ipv6 ospf 1 area 0
 ipv6 ospf priority 0
 ipv6 ospf hello-interval 5
 ipv6 ospf dead-interval 20
!
interface Loopback0
 ipv6 ospf 1 area 0
```

R6:

```
ipv6 unicast-routing
!
ipv6 router ospf 1
 router-id 150.1.6.6
!
interface GigabitEthernet1.146
 ipv6 enable
 ipv6 ospf 1 area 0
 ipv6 ospf priority 0
 ipv6 ospf hello-interval 5
 ipv6 ospf dead-interval 20
!
interface Loopback0
 ipv6 ospf 1 area 0
```

Verification

OSPFv3 for IPv6 retains many concepts of OSPFv2 for IPv4; they have practically the same set of LSA types plus the concept of areas, ABRs, area types, and so on. Even the router ID for OSPFv3 is the same 32-bit value, derived from the highest Loopback IPv4 address. Most OSPFv3 configurations are done at the interface level. For instance, instead of using the **network** command, under the OSPF process, you simply assign individual interfaces to the OSPFv3 process. Like RIPng, many of the OSPFv3 verification commands use the same format as their OSPFv2 counterparts, with the exception of the **ipv6** keyword in place of the **ip** keyword.

https://t.me/learningnets

```
R1#show ipv6 ospf neighbor


                OSPFv3 Router with ID (150.1.1.1) (Process ID 1)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
150.1.4.4         1   FULL/DROTHER    00:00:16    13              GigabitEthernet1.146
150.1.6.6         1   FULL/BDR        00:00:18    11              GigabitEthernet1.146
!
!R1#show ipv6 ospf 1
Routing Process "ospfv3 1" with ID 150.1.1.1
Supports NSSA (compatible with RFC 3101)
Event-log enabled, Maximum number of events: 1000, Mode: cyclic
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPFs 10000 msecs
Maximum wait time between two consecutive SPFs 10000 msecs
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Retransmission limit dc 24 non-dc 24
Number of external LSA 0. Checksum Sum 0x000000
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Graceful restart helper support enabled
Reference bandwidth unit is 100 mbps
RFC1583 compatibility enabled Area BACKBONE(0)
    Number of interfaces in this area is 2
    SPF algorithm executed 5 times
    Number of LSA 11. Checksum Sum 0x07AB7B
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0
!
!R1#show ipv6 ospf interface

 Loopback0 is up, line protocol is up
 Link Local Address FE80::21E:BDFF:FEE7:1800, Interface ID 9
 Area 0, Process ID 1, Instance ID 0, Router ID 150.1.1.1
 Network Type LOOPBACK, Cost: 1
 Loopback interface is treated as a stub Host
GigabitEthernet1.146 is up, line protocol is up
 Link Local Address FE80::250:56FF:FE8D:7819, Interface ID 12 Area 0, Process ID 1, Instance ID 0,
Router ID 150.1.1.1
```

https://t.me/learningnets

```
  Network Type BROADCAST, Cost: 1  Transmit Delay is 1 sec, State DR, Priority 1
 Designated Router (ID) 150.1.1.1, local address FE80::250:56FF:FE8D:7819
  Backup Designated router (ID) 150.1.6.6, local address FE80::250:56FF:FE8D:6E90
 Timer intervals configured, Hello 5, Dead 20, Wait 20, Retransmit 5
      Hello due in 00:00:03
  !
  !R1#show ipv6 route ospf
  IPv6 Routing Table - default - 10 entries
  Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - ApplicationO   2001:150:4:4::4/128 [110/1]
         via FE80::250:56FF:FE8D:4949, GigabitEthernet1.146O   2001:150:6:6::6/128 [110/1]

         via FE80::250:56FF:FE8D:6E90, GigabitEthernet1.146
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

OSPFv3 over NBMA

> You must load the initial configuration files for the section, **OSPFv3 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure OSPFv3 area 0 on the tunnel interfaces of R1, R2, and R5.
  - Do not modify the OSPFv3 network type or hello/dead intervals on R1 and R2.
- Upon completing the task, verify reachability between Loopback0 of R1 and R2.

Configuration

DMVPN is an NBMA design where Layer 2 broadcast does not pass through and eliminate Layer 2 broadcasts. We have R5 as hub, which connects multiple sites, (R1 and R2). By default, the tunnel interface considers itself a point-to-point interface. This can cause issues when connecting two spokes, so usually we configure *ip ospf network point-to-multipoint* in such a scenario. When we change the network type here, the behavior of OSPF timers is also changed. To address such issues, use the `ipv6 ospf hello-interval 10` interface-specific command.

```
R5:
ipv6 unicast-routing
!
interface Tunnel0
 ipv6 ospf 1 area 0
 ipv6 ospf network point-to-multipoint
 ipv6 ospf hello-interval 10
R1:
```

https://t.me/learningnets

```
ipv6 unicast-routing
!
interface Tunnel0
 ipv6 ospf 1 area 0
!
interface Loopback0
 ipv6 ospf 1 area 0
R2:

ipv6 unicast-routing
!
interface Tunnel0
 ipv6 ospf 1 area 0
!
interface Loopback0
 ipv6 ospf 1 area 0
```

Verification

The task wording prompts us to use the multi-point non-broadcast network type. Just like in OSPFv2, this network type supports multiple neighbors on the link, sends unicast updates, and does not perform the DR/BDR election. Multi-point network type is also a valid option. Note that in OSPFv3, you configure the neighbors at the interface level. As usual, properly mapping the link-local IPv6 addresses is required so that OSPF can exchange packets. Note the use of link-local IPv6 addresses with the **`neighbor`** command. With the multi-point network-type, you can also specify the neighbor cost, as with OSPFv2.

```
R5#sh ipv6 ospf neighbor



        OSPFv3 Router with ID (150.1.5.5) (Process ID 1)


Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
150.1.1.1         0   FULL/  -        00:00:39    15              Tunnel0
150.1.2.2         0   FULL/  -        00:00:33    12              Tunnel0
!
!R5#show ipv6 ospf interface Tunnel0
Tunnel0 is up, line protocol is up
    Link Local Address FE80::5, Interface ID 18
    Area 0, Process ID 1, Instance ID 0, Router ID 150.1.5.5
Network Type POINT_TO_MULTIPOINT, Cost: 1000Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
```

https://t.me/learningnets

```
Graceful restart helper support enabled
Index 1/1/1, flood queue length 0
Next 0x0(0)/0x0(0)/0x0(0)
Last flood scan length is 1, maximum is 7
Last flood scan time is 0 msec, maximum is 37 msec
Neighbor Count is 2, Adjacent neighbor count is 2 Adjacent with neighbor 150.1.1.1
Adjacent with neighbor 150.1.2.2
Suppress hello for 0 neighbor(s)
!
!R1#ping 2001:150:2:2::2 source Loopback0
 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to 2001:150:2:2::2, timeout is 2 seconds:
 Packet sent with a source address of 2001:150:1:1::1 !!!!!

 Success rate is 100 percent (5/5), round-trip min/avg/max = 8/12/28 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

OSPFv3 Virtual Links

> You must load the initial configuration files for the section, **OSPFv3 Basic**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure OSPFv3 in area 58 between R5 and R8.
- Configure a virtual-link through area 58.

Configuration

```
R5:
interface GigabitEthernet1.58
 ipv6 ospf 1 area 58
!
ipv6 router ospf 1
 area 58 virtual-link 150.1.8.8
R8:

interface GigabitEthernet1.58
 ipv6 ospf 1 area 58
!
ipv6 router ospf 1
 area 58 virtual-link 150.1.5.5
```

Verification

The concept of the virtual-link is the same in OSPFv3 as in OSPFv2. Using it, you

https://t.me/learningnets

may repair a discontiguous backbone, attach a non-zero area to the backbone across a transit area, or perform traffic engineering. We configure virtual-links under the global OSPFv3 process configuration mode. Check the virtual-link health and verify that you can see Area 0 link prefixes on R6.

```
R5#sh ipv6 ospf neighbor



        OSPFv3 Router with ID (150.1.5.5) (Process ID 1)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
150.1.8.8         0   FULL/  -            -        14              OSPFv3_VL0
150.1.1.1         0   FULL/  -        00:00:36    15              Tunnel0
150.1.2.2         0   FULL/  -        00:00:31    12              Tunnel0
150.1.8.8         1   FULL/BDR        00:00:35    11              GigabitEthernet1.58
!
!R5#show ipv6 ospf virtual-links



        OSPFv3 Router with ID (150.1.5.5) (Process ID 1)
Virtual Link OSPFv3_VL0 to router 150.1.8.8 is up
Interface ID 19, IPv6 address 2001:155:1:58::8Run as demand circuit
DoNotAge LSA allowed.
Transit area 58, via interface GigabitEthernet1.58, Cost of using 1
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Adjacency State FULL (Hello suppressed)
    Index 1/3/4, retransmission queue length 0, number of retransmission 6
    First 0x0(0)/0x0(0)/0x0(0) Next 0x0(0)/0x0(0)/0x0(0)
    Last retransmission scan length is 1, maximum is 1
    Last retransmission scan time is 0 msec, maximum is 0 msec
!
!R8#sh ipv6 route ospf

IPv6 Routing Table - default - 17 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - Application
O   2001:150:1:1::1/128 [110/1001]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
O   2001:150:2:2::2/128 [110/1001]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
```

https://t.me/learningnets

```
O   2001:150:5:5::5/128 [110/1]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
O   2001:155:1:1234::/64 [110/2001]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

OSPFv3 Summarization

> You must load the initial configuration files for the section, **OSPFv3 Basic**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure OSPFv3 area 58 between R5 and R8.
- Create a new Loopback1 interface on R1 and R2 as follows:
  - Use IPv6 address **FC00:1:0:Y::Y/64**, where *Y* is the router number.
  - Advertise Loopback1 prefixes into area 0.
  - Ensure that R5 advertises to R8 only the summary prefix for Loopback1 interfaces of R1 and R2.

Configuration

```
R1:
interface Loopback1
 ipv6 address FC00:1:0:1::1/64
 ipv6 ospf 1 area 0
 ipv6 ospf network point-to-point
R2:
interface Loopback1
 ipv6 address FC00:1:0:2::2/64
 ipv6 ospf 1 area 0
 ipv6 ospf network point-to-point
R5:
ipv6 router ospf 1
 area 0 range FC00:1::/62
!
interface GigabitEthernet1.58
```

https://t.me/learningnets

```
 ipv6 ospf 1 area 58
!R8:

interface GigabitEthernet1.58
 ipv6 ospf 1 area 58
```

Verification

Summarization concepts are the same for OSPFv3 as for OSPFv2. Even the commands remain the same; you just need to apply them under the OSPFv3 process context. When summarizing the prefixes fc00:1:0:1::/64 and fc00:1:0:2:/64, remember that 1 and 2 are in hex. Convert them to binary format using the Windows calculator application:

```
1 = 0000 0000 1000 00**01**
2 = 0000 0000 0000 10**10**
```

Remove the mismatched parts. The resulting prefix is fc00:1::/62.

```
R5#show ipv6 ospf neighbor


        OSPFv3 Router with ID (150.1.5.5) (Process ID 1)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
150.1.1.1         0   FULL/  -        00:00:37    15              Tunnel0
150.1.2.2         0   FULL/  -        00:00:32    12              Tunnel0
150.1.8.8         1   FULL/BDR        00:00:31    11              GigabitEthernet1.58
!
!R8#sh ipv6 route ospf
IPv6 Routing Table - default - 17 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
   B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
   I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
   EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
   NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
   OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
   ls - LISP site, ld - LISP dyn-EID, a - Application
OI  2001:150:1:1::1/128 [110/1001]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
OI  2001:150:2:2::2/128 [110/1001]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
OI  2001:150:5:5::5/128 [110/1]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
```

https://t.me/learningnets

```
OI  2001:155:1:1234::/64 [110/2001]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
OI  2001:155:1:1234::5/128 [110/1]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58OI  FC00:1::/62 [110/1002]
        via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
!
!R8#ping FC00:1:0:1::1
 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to FC00:1:0:1::1, timeout is 2 seconds:!!!!!
 Success rate is 100 percent (5/5), round-trip min/avg/max = 6/8/14 ms
!
!R8#ping FC00:1:0:2::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FC00:1:0:2::2, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 6/12/27 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Redistribution

> You must load the initial configuration files for the section, **IPv6 Redistribution Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Perform mutual route redistribution between OSPFv3, RIPng, and EIGRPv6 on R5.
- Upon completing this task, R8 should be able to reach Loopback0 network of R1, R2, R4, and R5.

Configuration

R5:

```
ipv6 router ospf 1
 redistribute rip RIPNG metric 8
 redistribute eigrp 100 metric 8
 redistribute connected metric 8
!
ipv6 router rip RIPNG
 redistribute eigrp 100 include-connected metric 8
 redistribute ospf 1 include-connected metric 8
!
ipv6 router eigrp 100
 redistribute rip RIPNG include-connected metric 1000 0 255 1 1500
 redistribute ospf 1 include-connected metric 1000 0 255 1 1500
```

https://t.me/learningnets

Verification

When redistributing between two IPv6 IGPs, remember that the command `redistribute <protocol>` will not redistribute the locally connected interfaces advertised into the source routing protocol. For example, on R6, if you use the `redistribute rip RIPNG` command, only the routes learned by R6 via RIPng will be redistributed (not the local Loopback0, which is being advertised into RIPng). To account for directly connected subnets, use the separate command `redistribute connected` or the `include-connected` keyword when performing redistributing. Because OSPFv3 has a better administrative distance than RIPng or EIGRPv6, adjust the OSPFv3 external distance on R6 to make sure that external OSPFv3 prefixes are not preferred over internal RIPng prefixes.

The rest of the redistribution rules match those used when redistributing between IPv4 routing protocols.

```
R8#sh ipv6  route eigrp

IPv6 Routing Table - default - 16 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - Application
EX  2001:150:1:1::1/128 [170/2560256]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
EX  2001:150:2:2::2/128 [170/2560256]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
EX  2001:150:4:4::4/128 [170/2560256]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
EX  2001:150:5:5::5/128 [170/2560256]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
EX  2001:155:1:45::/64 [170/2560256]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
EX  2001:155:1:1234::/64 [170/2560256]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.58
!
!R4#sh ipv6 route rip

IPv6 Routing Table - default - 15 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
```

https://t.me/learningnets

```
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - Application
R   2001:150:1:1::1/128 [120/9]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.45
R   2001:150:2:2::2/128 [120/9]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.45
R   2001:150:5:5::5/128 [120/9]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.45
R   2001:150:6:6::6/128 [120/9]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.45
R   2001:150:8:8::8/128 [120/9]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.45
R   2001:155:1:58::/64 [120/9]
    via FE80::250:56FF:FE8D:74DD, GigabitEthernet1.45
!
!R1#sh ipv6  route ospf

IPv6 Routing Table - default - 19 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - Application
O   2001:150:2:2::2/128 [110/2000]
    via FE80::5, Tunnel0
OE2 2001:150:4:4::4/128 [110/8]
    via FE80::5, Tunnel0
O   2001:150:5:5::5/128 [110/1000]
    via FE80::5, Tunnel0
O   2001:150:6:6::6/128 [110/1]
    via FE80::250:56FF:FE8D:6E90, GigabitEthernet1.146
OE2 2001:150:8:8::8/128 [110/8]
    via FE80::5, Tunnel0
OE2 2001:155:1:5::/64 [110/8]
    via FE80::5, Tunnel0
OE2 2001:155:1:45::/64 [110/8]
    via FE80::5, Tunnel0
OE2 2001:155:1:58::/64 [110/8]
    via FE80::5, Tunnel0
O   2001:155:1:1234::5/128 [110/1000]
```

https://t.me/learningnets

```
via FE80::5, Tunnel0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Filtering

> You must load the initial configuration files for the section, **IPv6 Redistribution Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R3's Tunnel0 interface so that only FTP and HTTP traffic sourced from VLAN37 or destined to VLAN37 can be routed through the interface.
- Allow DNS queries and responses sourced from VLAN37 or destined to VLAN37, and ensure that IPv6 routing will not be affected.
- Enable HTTP Server on R5.

Configuration

```
R3:
ipv6 access-list FILTER_OUT
 permit tcp 2001:155:1:37::7/64 any eq 80
 permit tcp 2001:155:1:37::/64 any range 20 21
 permit udp 2001:155:1:37::/64 any eq 53
!
ipv6 access-list FILTER_IN
 permit tcp any eq 80 2001:155:1:37::/64
 permit tcp any range 20 21 2001:155:1:37::/64
 permit udp any eq 53 2001:155:1:37::/64
 permit udp any any eq 520
!
interface Tunnel0
 ipv6 traffic-filter FILTER_OUT out
 ipv6 traffic-filter FILTER_IN in
R5:
```

https://t.me/learningnets

```
ip http server
```

Verification

Cisco IOS supports only extended IPv6 access-lists. The syntax remains the same, except the source/destination specifications. Instead of using subnets and wildcard masks, you specify IPv6 prefixes. Aside from that, the logic of the access-lists remains the same. The access-lists are applied as traffic-filters to the interfaces. Note that IPv6 access-lists do not have a keyword to permit RIPng explicitly, so you must use its protocol/port number, which is UDP port 520. Generate some traffic to test the access-lists from R7.

```
R7#ping 2001:155:1:1234::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:155:1:1234::3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/32/116 ms
!
!R7#ping 2001:150:5:5::5

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:150:5:5::5, timeout is 2 seconds:AAAAA

Success rate is 0 percent (0/5)
```

Check that RIPng is learning routes from R5 and R7.

```
R3#sh ipv6 route rip
IPv6 Routing Table - default - 15 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
    B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
    I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
    EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
    NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
    OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
    ls - LISP site, ld - LISP dyn-EID, a - ApplicationR   2001:150:5:5::5/128 [120/2]
        via FE80::5, Tunnel0R   2001:150:7:7::7/128 [120/2]

        via FE80::250:56FF:FE8D:392B, GigabitEthernet1.37
```

Verify that the HTTP filtering is working.

```
R7#telnet  2001:155:1:1234::5 80
                                Trying 2001:155:1:1234::5, 80 ... Open
```

https://t.me/learningnets

```
!
!R7#telnet  2001:155:1:1234::5 80 /source-interface Loopback0
Trying 2001:155:1:1234::5, 80 ... % Connection timed out; remote host not responding
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 MP-BGP

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure BGP in AS 100 on R1 and AS 500 on R5, form a BGP peering over the DMVPN cloud.
  - Create two new loopback interfaces on R1 with IPv6 addresses **2003:1:0:1::1/64** and **2003:1:0:11::11/61**, and advertise them into BGP.
- Ensure that R5 receives only an optimal summary route encompassing both IPv6 prefixes.

Configuration

```
R1:
interface Loopback101
 ipv6 address 2003:1:0:1::1/64
!
interface Loopback102
 ipv6 address 2003:1:0:11::11/61
!
router bgp 100
 address-family ipv6 unicast
  neighbor 2001:155:1:1234::5 remote-as 500
  neighbor 2001:155:1:1234::5 activate
  network 2003:1:0:1::/64
  network 2003:1:0:10::/61
  aggregate-address 2003:1::/59 summary-only
R5:
```

https://t.me/learningnets

```
router bgp 500
 address-family ipv6 unicast
  neighbor 2001:155:1:1234::1 remote-as 100
  neighbor 2001:155:1:1234::1 activate
```

Verification

Multi-protocol extensions for BGP adds support for the IPv6 address family, which essentially means that IPv6 prefixes and attributes can be exchanged over a normal TCP-based BGP peering. We configure IPv6 peers under the **`address-family ipv6 unicast`** configuration mode inside of BGP, and then they must be "activated," or enabled. Technically, this is the same process used in IPv4 peering, but the normal global BGP process automatically refers to the **`address-family ipv4 unicast`** , and the command **`bgp default ipv4-unicast`** is enabled by default, which means that all unicast IPv4 peers are automatically "activated."

The configuration of all IPv6-related options takes place under the address-family, such as the network statement, or applying attributes onto neighbors like route-maps, prefix-lists, filter-lists, and so on. Remember that all other BGP operations stay the same, such as advertisement rules, route reflection, confederation, best path selection, and so on. The only thing that has changed is the format of the prefix and next-hop advertised inside of the BGP update messages. Note that for EBGP peerings, the next-hop values recurse to link-local addresses in the routing table, so when peering over multi-point NBMA, we must map the link-local IPv6 addresses, as with the other IPv6 protocols we have discussed. Prefix summarization uses the same command used in IPv4 and the same prefix length calculations. In our case, take the mismatched parts of the prefixes (remember, 11 is in hex, so it is 17 in decimal).

```
0000 0000 000**0 0001**
0000 0000 000**1 0**
```

Note that the second prefix length is 61, not 64. To build the summary, shift the prefix length left by 5 bits. The resulting prefix is 2003:1::/59.

```
R1#show bgp ipv6 unicast summary

BGP router identifier 150.1.1.1, local AS number 100
BGP table version is 1, main routing table version 1
2 network entries using 544 bytes of memory
2 path entries using 288 bytes of memory
1/0 BGP path/bestpath attribute entries using 240 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
```

https://t.me/learningnets

```
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1072 total bytes of memory
BGP activity 2/0 prefixes, 2/0 paths, scan interval 60 secs

    Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:155:1:1234::5
                    4          500       2       2        1    0    0 00:00:38        0
!
!R1#show bgp ipv6 unicast
BGP table version is 6, local router ID is 150.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
            r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
            x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

    Network          Next Hop            Metric     LocPrf  Weight      Path*>  2003:1::/59
      ::                                32768 i
s>  2003:1:0:1::/64      ::                  0              32768 i
s>  2003:1:0:10::/61     ::                  0              32768 i
!
!R5#show bgp ipv6 unicast
BGP table version is 2, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
            r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
            x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

        Network          Next Hop            Metric LocPrf Weight Path*>  2003:1::/59
      2001:155:1:1234::1
                                            0             0 100 i
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 PIM and MLD

> You must load the initial configuration files for the section, **OSPFv3 Basic**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Enable IPv6 Multicast routing on R3, R5, and R7.
- Configure R3 to join the multicast group **FF08::7/128** on its VLAN37 interface.
- R3 should only accept MLD Reports for the group range **FF08::/64**.
  - Send the general queries every 10 seconds on R3's VLAN37 interface.

Configuration

In IPv4, there is a specific range of addresses reserved for multicast use (224.0.0.0 to 239.255.255.255). In IPv6, multicast addresses are all addresses inside the prefix FF00::/8. In other words, an IPv6 address is a multicast address if and only if the first byte is FF (255 in decimal). An IPv6 address is 128 bits long, and these bits are divided into groups that are used to signify specific things. After the first 8 bits (which define a multicast address), there are 4 bits each for flags and scopes. This leaves 112 bits to specify a group ID, which, because of the expansive size of the IPv6 address space, is massive compared to IPv4 addressing.

https://t.me/learningnets

Now we will to look at the flag bits. Currently, the first three of these four bits are unused (and subsequently set to 0). The fourth flag bit is called the transient bit. Its purpose is to indicate whether an address is permanent or temporary. If the address is permanently assigned (issued by IANA), this bit is set to zero. However, if the transient bit is a one, the address is temporary (or transient). Perfect examples of permanently assigned addresses would be FF02::2 (All Routers) or FF02::6 (OSPF DR Routers).

The remaining four bits of interest to us in a multicast address are known as the Scope ID bits. There are 16 different combinations than can be created using these 4-bits. Not all 16 available values are currently in use at this time, but seven of them are used to determine an address scope. As an example, if the address scope is 1110 or “global,” that address is valid across the entire Internet. These are the currently defined scope ID bits:

Decimal Value	Binary Value	Address Scope
0	0000	Reserved
1	0001	Node-Local Scope
2	0010	Link-Local Scope
5	0101	Site Local Scope
8	1000	Organization Local Scope
14	1110	Global Scope
15	1111	Reserved

It must be noted that enforcement of flags and flooding scopes does not take place in software in any way. In addition, flooding scope enforcement must be configured administratively using multicast filtering. The prefixes just use the field for ease of administrative management.

Most of IPv6 multicast functionality is similar to IPv4 multicast, with some

https://t.me/learningnets

modifications. Therefore, we highly recommend that you complete the *Multicast* section of this workbook before continuing further into IPv6 multicasting.

The following protocols should be understood thoroughly before continuing because they are the fundamental core of IPv6 multicasting:

PIMv2 for IPv6 is very similar to standard PIMv2 in its functionality, but it supports only Sparse-Mode operation. There is no Dense Mode flooding feature in IPv6 multicast protocol. Many of IPv4 PIMv2's commands and concepts transition to its IPv6 counterpart, including Designated Router, Assert, Rendezvous Point, etc. IPv6 multicast uses the unicast routing table (IPv6 RIB) for RPF checks against multicast sources. This means that multicast traffic propagation for IPv6 takes place in the same fashion as IPv4 multicasting.

As soon as you type the command `ipv6 multicast-routing`, PIMv6 becomes active on all IPv6-capable interfaces. You may disable this on an interface-by-interface basis by using the command `no ipv6 pim`. As usual, all PIM-enabled interfaces are capable of flooding multicast traffic, if there is a multicast tree built.

MLD (Multicast Listener Discovery protocol), based on ICMPv6, has replaced IGMP. IGMP is the membership signaling protocol used in IPv4 multicasting. MLDv1 translates to IGMPv2 and MLDv2 translates to IGMPv3, allowing for source-specific IPv6 multicast.

MLD supports three message types very similar to IGMP's messages: Query, Report, and Done, which is equivalent to IGMPv2's Leave. Configuring MLD is similar to configuring IGMP; you may define the maximum number of groups joined by the hosts on the interface using the command `ipv6 mld limit`, or join a group by using the command `ipv6 mld join-group`. You may also set various MLD query-related parameters using the commands `ipv6 mld query-interval`, `ipv6 mld query-timeout`, and `ipv6 mld query-max-response-time`. The MLD timers are the same as those used with IGMPv2 or IGMPv3.

Notice the use of an IPv6 access-list below to control the groups that a host may join. The access-list entry `[permit|deny] ipv6 <part1> <part2>` allows for both SSM and normal multicast group filtering. If you want to filter just MLDv1 joins, leave `<part1>` as "any"; this specifies any source. The parameter configured in `<part2>` is responsible for the multicast-group being joined. So it is possible with this one filter to permit or deny the specific multicast groups that can be joined by specific hosts.

```
R3, R5, and R7:
ipv6 multicast-routing
R3:


ipv6 access-list MLD_FILTER
```

https://t.me/learningnets

```
 permit ipv6 any ff08::/64
!
interface GigabitEthernet1.37
 ipv6 mld access-group MLD_FILTER
 ipv6 mld query-interval 10
 ipv6 mld join-group ff08::7
```

Verification

Verify PIM interfaces and neighbors on every router. Here we show just R3's status.

```
R3#show ipv6 pim interface

Interface          PIM    Nbr    Hello  DR
                     Count Intvl  Prior
Tunnel0             on     1      30      1

    Address: FE80::3
    DR     : FE80::5
Loopback0           on     0      30      1
    Address: FE80::21E:BDFF:FEE7:1800
    DR     : this system
Gi1.37              on     1      30      1

    Address: FE80::250:56FF:FE8D:3357
    DR     : FE80::250:56FF:FE8D:392B
GigabitEthernet1   off     0      30      1
    Address: ::
    DR     : not elected
GigabitEthernet2   off     0      30      1
    Address: ::
    DR     : not elected
GigabitEthernet3   off     0      30      1
    Address: ::
    DR     : not elected
Tunnel1            off     0      30      1
    Address: FE80::21E:BDFF:FEE7:1800
    DR     : not elected
Gi1.13              on     0      30      1
    Address: FE80::250:56FF:FE8D:3357
    DR     : this system
!
!R3#sh ipv6 pim neighbor

PIM Neighbor Table
Mode: B - Bidir Capable, G - GenID Capable
```

https://t.me/learningnets

```
Neighbor Address           Interface          Uptime    Expires  Mode DR pri
FE80::5 Tunnel0
            00:00:50  00:01:25 B G  DR 1  FE80::250:56FF:FE8D:392B Gi1.37
             00:00:38  00:01:37 B G  DR 1
!
!R3#sh ipv6 mld groups

MLD Connected Group Membership
Group Address                            Interface          Uptime    Expires
FF08::7                                  Gi1.37             00:00:58  never
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 PIM BSR

> You must load the initial configuration files for the section, **IPv6 Multicast Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R6 as the RP for the multicast domain and R4 as the BSR to propagate this information.
- R1, R4, and R6 should be able to ping any multicast group joined by R3.

Configuration

PIMv6 RP and BSR are very similar to their IPv4 counterparts. However, there are a number of small configuration differences. The command to configure a router to announce itself as an RP via the BSR protocol is `ipv6 pim bsr candidate rp <IPv6 Address>` (not the interface name). The command to configure a router to start distributing RP bootstrap information and participate in BSR elections is `ipv6 pim bsr candidate bsr <IPv6 Address>` . As usual, you can configure a single router as both an RP and a BSR. One additional feature of an IPv6 BSR is that you may statically program a BSR with a list of candidate RPs by using the command `ipv6 pim bsr announced rp <IPv6 Address>` and therefore eliminate the need for dynamic RP advertisements by the candidates.

The only significant difference between the IPv4 PIM and IPv6 PIM is the source registration process. When a source starts multicasting, the DR on the segment is supposed to register it with the RP. This is normally performed by sending PIM register messages to the RP and encapsulating the original data packet inside them. As soon as a multicast IPv6 router learns the RP address, it creates a tunnel interface connecting the router to the RP router. This tunnel is automatically enabled for multicasting, so multicast traffic sent down the shared tree is simply forwarded

https://t.me/learningnets

out the tunnel toward the RP. The tunnel is used only for the duration of the registration process, and then the receivers switch to the optimal path, which will not transit the tunnel. The use of this tunnel allows consistent multicast packet flooding behavior using the same code and simple tunneling of PIM registration messages. You may see a list of all currently active tunnels by using the command **`show ipv6 pim tunnel`** .

```
R6:
ipv6 pim bsr candidate rp 2001:150:1:6::6
!R4:

ipv6 pim bsr candidate bsr 2001:150:1:4::4
```

Verification

Use the commands below to display the BSR election information and the RP to group mappings. You need to know that all routers have learned the BSR's information before starting any tests.

```
R1#sh ipv6 pim bsr election

PIMv2 BSR information

BSR Election Information
Scope Range List: ff00::/8 BSR Address: 2001:150:4:4::4
    Uptime: 00:01:00, BSR Priority: 0, Hash mask length: 126
    RPF: FE80::250:56FF:FE8D:4949,Gi1.146
    BS Timer: 00:01:10
!
!R3#sh ipv6 pim bsr election

PIMv2 BSR information

BSR Election Information
Scope Range List: ff00::/8 BSR Address: 2001:150:4:4::4

    Uptime: 00:02:47, BSR Priority: 0, Hash mask length: 126
    RPF: FE80::250:56FF:FE8D:7819,Gi1.13
    BS Timer: 00:01:23
```

The following are the mappings of RPs to the multicast group ranges.

https://t.me/learningnets

```
R3#sh ipv6 pim range-list

Static SSM Exp: never Learnt from : ::
    FF33::/32 Up: 09:56:22
    FF34::/32 Up: 09:56:22
    FF35::/32 Up: 09:56:22
    FF36::/32 Up: 09:56:22
    FF37::/32 Up: 09:56:22
    FF38::/32 Up: 09:56:22
    FF39::/32 Up: 09:56:22
    FF3A::/32 Up: 09:56:22
    FF3B::/32 Up: 09:56:22
    FF3C::/32 Up: 09:56:22
    FF3D::/32 Up: 09:56:22
    FF3E::/32 Up: 09:56:22
    FF3F::/32 Up: 09:56:22    BSR SM RP:2001:150:6:6::6 Exp: 00:02:01 Learnt from :2001:150:4:4::4

        FF00::/8 Up: 00:04:29
```

Now make sure that R3 has the (*,G) state for the multicast group joined locally.

```
R3#sh ipv6  mroute

Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group,
    C - Connected, L - Local, I - Received Source Specific Host Report,
    P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set,
    J - Join SPT, Y - Joined MDT-data group,
    y - Sending to MDT-data group
    g - BGP signal originated, G - BGP Signal received,
    N - BGP Shared-Tree Prune received, n - BGP C-Mroute suppressed,
    q - BGP Src-Active originated, Q - BGP Src-Active received
    E - Extranet
Timers: Uptime/Expires
Interface state: Interface, State
(*, FF08::3), 00:19:55/00:03:15, RP 2001:150:6:6::6, flags: SCL

    Incoming interface: GigabitEthernet1.13
    RPF nbr: FE80::250:56FF:FE8D:7819
Immediate Outgoing interface list:
    GigabitEthernet1.37, Forward, 00:19:55/00:03:15
```

Ping the group from R1, R4, and R6 via its outgoing interface.

https://t.me/learningnets

```
R1#ping ff08::3

Output Interface: GigabitEthernet1.13
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FF08::3, timeout is 2 seconds:
Packet sent with a source address of 2001:155:1:13::1

Reply to request 0 received from 2001:155:1:37::3, 113 ms
Reply to request 1 received from 2001:155:1:37::3, 41 ms
Reply to request 2 received from 2001:155:1:37::3, 2 ms
Reply to request 3 received from 2001:155:1:37::3, 3 ms
Reply to request 4 received from 2001:155:1:37::3, 3 ms
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/32/113 ms
5 multicast replies and 0 errors.
!
!R4#ping ff08::3
Output Interface: GigabitEthernet1.146
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FF08::3, timeout is 2 seconds:
Packet sent with a source address of 2001:155:1:146::4

Reply to request 0 received from 2001:155:1:37::3, 54 ms
Reply to request 0 received from 2001:155:1:37::3, 89 ms
Reply to request 1 received from 2001:155:1:37::3, 6 ms
Reply to request 2 received from 2001:155:1:37::3, 14 ms
Reply to request 3 received from 2001:155:1:37::3, 4 ms
Reply to request 4 received from 2001:155:1:37::3, 35 ms
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/33/89 ms
6 multicast replies and 0 errors.
!
!R6#ping ff08::3
Output Interface: GigabitEthernet1.146
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FF08::3, timeout is 2 seconds:
Packet sent with a source address of 2001:155:1:146::6

Reply to request 0 received from 2001:155:1:37::3, 150 ms
Reply to request 1 received from 2001:155:1:37::3, 18 ms
Reply to request 2 received from 2001:155:1:37::3, 11 ms
Reply to request 3 received from 2001:155:1:37::3, 3 ms
Reply to request 4 received from 2001:155:1:37::3, 19 ms
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/40/150 ms
```

https://t.me/learningnets

```
5 multicast replies and 0 errors.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Embedded RP

> You must load the initial configuration files for the section, **IPv6 Multicast Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R3 to join the group **FF76:0640:2001:CC1E::8** on its VLAN 13 interface.
- Make sure that R6 can ping this group without any router in the domain learning the RP information via the BSR.

Configuration

One feature is available in IPv4 multicasting that is not available in IPv6 at this time. With Auto-RP technology, routers can be set up as candidate RPs for all or selected multicast groups. They advertise themselves via multicast to 224.0.1.39. Auto-RP is not currently available in IPv6 Multicasting. Additionally, there is nothing like Multicast Source Discovery Protocol (MSDP) in IPv6 Multicasting.

https://t.me/learningnets

IPv6 Embedded RP is one solution we can use when we are required to support a PIM SM model across multiple domains. Earlier we discussed the Flags bits in an IPv6 address. We explained that in normal situations, the first 3 Flags bits are always set to 000. When we use Embedded RP, this situation changes, and the second, third, and fourth bits are set to 1s. We explained the function of the fourth, or transient, bit earlier; now we are looking at the second and third bits. The second bit is the R bit, and when it is set to a value of one, it indicates that the 4-bit RP interface-ID is actually embedded in the group address. The third bit is the P bit; when it is set to 1, it indicates that the 64-bit RP prefix is now embedded in the multicast address. Now with the P, R, and T bits set to one, instead of relying on the BSR for information propagation, senders and receivers agree to embed the RP IPv6 address into the multicast IPv6 address itself. This address takes the form:

```
FF7x:0yLL:PPPP:PPPP:PPPP:PPPP:<32-bit-group-id>
```

Where *FF7* is the hexadecimal value of “1111 1111 0111”, *x* is the 4-bit scope, *y* is the 4-bit RP interface-id, *LL* is an 8-bit prefix length, and *PPPP:PPPP:PPPP:PPPP* is the 64-bit RP prefix. The actual RP address is constructed by taking the prefix, masking it with LL ones bits, and appending the RP Interface-ID, like so:

```
PPPP:PPPP:PPPP:PPPP::y/LL
```

The interface ID cannot be zero, so it is any value from 0 to F (remember, all numbers are in hex!). The router that is supposed to be the RP with this address must have a Loopback interface with this address advertised into the routing protocol so all routers in the multicast domain can reach it. In our case, we’ve picked up the group address FF76:0640:2001:CC1E::8, which corresponds to the RP address 2001:CC1E::6/64, which we advertise from R6.

```
R3:
interface GigabitEthernet1.37
 ipv6 mld join-group ff76:0640:2001:CC1E::8
R6:

interface Loopback1
 ipv6 address 2001:CC1E::6/128
 ipv6 ospf 1 area 0
```

Verification

https://t.me/learningnets

Check that R4 and R5 learn the RP-information embedded in the group address.

```
R3#sh ipv6 pim neighbor

PIM Neighbor Table
Mode: B - Bidir Capable, G - GenID Capable
Neighbor Address            Interface          Uptime    Expires  Mode DR pri
FE80::250:56FF:FE8D:7819    Gi1.13             02:17:20  00:01:43 B G  DR 1
!
!R3#sh ipv6  pim range-list

Static SSM Exp: never Learnt from : ::
 FF33::/32 Up: 13:01:12
 FF34::/32 Up: 13:01:12
 FF35::/32 Up: 13:01:12
 FF36::/32 Up: 13:01:12
 FF37::/32 Up: 13:01:12
 FF38::/32 Up: 13:01:12
 FF39::/32 Up: 13:01:12
 FF3A::/32 Up: 13:01:12
 FF3B::/32 Up: 13:01:12
 FF3C::/32 Up: 13:01:12
 FF3D::/32 Up: 13:01:12
 FF3E::/32 Up: 13:01:12
 FF3F::/32 Up: 13:01:12Embedded SM RP: 2001:CC1E::6 Exp: never Learnt from : ::
 FF76:640:2001:CC1E::/96 Up: 01:18:40
!
!R3#sh ipv6  mld groups

MLD Connected Group Membership
Group Address                           Interface          Uptime    Expires
FF76:640:2001:CC1E::3                   Gi1.37             00:07:23  never

!
!R1#sh ipv6  pim  range-list

Static SSM Exp: never Learnt from : ::
 FF33::/32 Up: 03:31:56
 FF34::/32 Up: 03:31:56
 FF35::/32 Up: 03:31:56
 FF36::/32 Up: 03:31:56
 FF37::/32 Up: 03:31:56
 FF38::/32 Up: 03:31:56
 FF39::/32 Up: 03:31:56
 FF3A::/32 Up: 03:31:56
 FF3B::/32 Up: 03:31:56
```

https://t.me/learningnets

```
 FF3C::/32 Up: 03:31:56
 FF3D::/32 Up: 03:31:56
 FF3E::/32 Up: 03:31:56
 FF3F::/32 Up: 03:31:56Embedded SM RP: 2001:CC1E::6 Exp: never Learnt from : ::

 FF76:640:2001:CC1E::/96 Up: 00:04:39
```

Now ping the group from R1 and make sure you get a response.

```
R1#ping FF76:640:2001:CC1E::8
Output Interface: GigabitEthernet1.13
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FF76:640:2001:CC1E::8, timeout is 2 seconds:
Packet sent with a source address of 2001:155:1:13::1

Reply to request 0 received from 2001:155:1:37::3, 99 ms
Reply to request 0 received from 2001:155:1:37::3, 99 ms
Reply to request 1 received from 2001:155:1:37::3, 3 ms
Reply to request 2 received from 2001:155:1:37::3, 2 ms
Reply to request 3 received from 2001:155:1:37::3, 22 ms
Reply to request 4 received from 2001:155:1:37::3, 6 ms
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/38/99 ms

 6 multicast replies and 0 errors.
```

Look at the multicast routing table of R1 and notice the (*,G) state created for the RP 2001:CC1E::6:.

```
R1#sh ipv6  mroute

Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group,
    C - Connected, L - Local, I - Received Source Specific Host Report,
    P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set,
    J - Join SPT, Y - Joined MDT-data group,
    y - Sending to MDT-data group
    g - BGP signal originated, G - BGP Signal received,
    N - BGP Shared-Tree Prune received, n - BGP C-Mroute suppressed,
    q - BGP Src-Active originated, Q - BGP Src-Active received
    E - Extranet
Timers: Uptime/Expires
Interface state: Interface, State
 (*, FF76:640:2001:CC1E::8), 00:07:20/00:03:15, RP 2001:CC1E::6
, flags: S
    Incoming interface: GigabitEthernet1.146
```

https://t.me/learningnets

```
RPF nbr: FE80::250:56FF:FE8D:6E90
    Immediate Outgoing interface list:
    GigabitEthernet1.13, Forward, 00:07:20/00:03:15
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 SSM

> You must load the initial configuration files for the section, **IPv6 Multicast Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R3 to join the group **FF36::8** on its VLAN37 interface.
  - Ensure that it only accepts multicast streams sourced from R6's IPv6 address on VLAN146.

Configuration

SSM, or Source Specific Multicast, is the simplest form of multicasting. The receiver specifies the list of sources from which it will accept multicast flows, as well as the group address in an MLD Report message (alternatively, there could be an "exclude" report, listing the sources that the host does not want to listen to, but this requires source discovery by using an RP and does not work with SSM). The multicast router builds shortest-path trees toward the source(s) specified in the report, and the senders may start transmitting. There is no need for RPs with SSM. In fact, groups within the IPv6 SSM range will never have an RP mapped to them. The special SSM group range is FF3x::/96. This allows for 2^{32} multicast groups per sender. You may filter the sources that a listener may join by using the MLD configuration command `ipv6 mld access-group` , where an access-list specifies the sources and the group ranges.

https://t.me/learningnets

```
R3:

interface GigabitEthernet1.37
 ipv6 mld join-group ff36::8 2001:155:1:146::6
```

Verification

Check the multicast routing table of R5 and R4, and confirm that there are (S,G) entries installed in each.

```
R3#sh ipv6  mroute

Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group,
    C - Connected, L - Local, I - Received Source Specific Host Report,
    P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set,
    J - Join SPT, Y - Joined MDT-data group,
    y - Sending to MDT-data group
    g - BGP signal originated, G - BGP Signal received,
    N - BGP Shared-Tree Prune received, n - BGP C-Mroute suppressed,
    q - BGP Src-Active originated, Q - BGP Src-Active received
    E - Extranet
Timers: Uptime/Expires
    Interface state: Interface, State
(2001:155:1:146::6, FF36::8), 00:00:08/00:03:21, flags: sLTI
Incoming interface: GigabitEthernet1.13
    RPF nbr: FE80::250:56FF:FE8D:7819
Immediate Outgoing interface list:
    GigabitEthernet1.37, Forward, 00:00:08/00:03:21
!
!R1#sh ipv6  mroute

Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group,
    C - Connected, L - Local, I - Received Source Specific Host Report,
    P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set,
    J - Join SPT, Y - Joined MDT-data group,
    y - Sending to MDT-data group
    g - BGP signal originated, G - BGP Signal received,
    N - BGP Shared-Tree Prune received, n - BGP C-Mroute suppressed,
    q - BGP Src-Active originated, Q - BGP Src-Active received
    E - Extranet
Timers: Uptime/Expires
```

https://t.me/learningnets

```
Interface state: Interface, State
(2001:155:1:146::6, FF36::8), 00:00:33/00:02:56, flags: sT

    Incoming interface: GigabitEthernet1.146
RPF nbr: FE80::250:56FF:FE8D:6E90
Immediate Outgoing interface list:
    GigabitEthernet1.13, Forward, 00:00:33/00:02:56
```

Now generate multicast packets from R6 and confirm that you are getting the responses.

```
R6#ping ipv6 FF36::8
Output Interface: GigabitEthernet1.146
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FF36::8, timeout is 2 seconds:
Packet sent with a source address of 2001:155:1:146::6

Reply to request 0 received from 2001:155:1:37::3, 37 ms
Reply to request 1 received from 2001:155:1:37::3, 10 ms
Reply to request 2 received from 2001:155:1:37::3, 31 ms
Reply to request 3 received from 2001:155:1:37::3, 4 ms
Reply to request 4 received from 2001:155:1:37::3, 11 ms
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/18/37 ms

    5 multicast replies and 0 errors.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

IPv6 Tunneling

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure R5 and R8 for IPv6 tunnel as follows:
  - Create Tunnel255 interface on R5 and R8 for IPv6.
  - Use the IPv6 addresses **2001:255:1:58::Y/64** for the tunnel interfaces between R5 and R8, where *Y* is the router number.
  - Configure static IPv6 routing as required.
  - Upon completing this task, R5 and R8 should be able to reach each other's Loopback0 interfaces.

Configuration

```
R5:
ipv6 unicast-routing
!
interface Tunnel255
 ipv6 address 2001:255:1:58::5/64
 tunnel source 155.1.58.5
 tunnel mode ipv6ip
 tunnel destination 155.1.58.8
!
ipv6 route ::/0 Tunnel 255
R8:


ipv6 unicast-routing
```

https://t.me/learningnets

```
!
interface Tunnel255
 ipv6 address 2001:255:1:58::8/64
 tunnel source 155.1.58.8
 tunnel mode ipv6ip
 tunnel destination 155.1.58.5
!
ipv6 route ::/0 Tunnel 255
```

Verification

Pitfall

> IPv6 IP tunnels use IP protocol number 41 for transport. This protocol number does not have a keyword shortcut in extended IP access-lists in IOS. Therefore, to permit or deny an IPv6IP tunnel, the syntax **`access-list 100 [permit|deny] 41 any any`** is required.

IPv6 can be transported across IPv4 clouds using various tunneling techniques. The most common are static point-to-point tunnels, using either GRE (generic routing encapsulation) or IPv6 in IPv4 encapsulation (a special protocol type designed to carry only IPv6 packets). Designed to carry a multi-protocol payload, GRE has a slightly larger overhead than IPv6 in IPv4 encapsulation.

```
R5#show interfaces Tunnel255

Tunnel255 is up, line protocol is up
Hardware is Tunnel
 MTU 17872 bytes, BW 100 Kbit/sec, DLY 50000 usec,
        reliability 255/255, txload 1/255, rxload 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

Automatic 6to4 Tunneling

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Using the IPv4 Loopback0 interfaces, create automatic 6to4 tunnels connecting R5 and R8.
- Create additional Loopback interfaces on both routers with the subnet number 1 and the prefix length of 64 under the respective 6to4 /48 prefix.
- Use static routing to obtain connectivity between the newly allocated subnets.

Configuration

```
R5:
interface Tunnel 58
 tunnel source Loopback0
 tunnel mode ipv6ip 6to4
 ipv6 address 2002:9601:505::5/64
!
ipv6 route 2002::/16 Tunnel58
!
interface Loopback58
 ipv6 address 2002:9601:505:1::5/64
R8:

interface Tunnel 58
 tunnel source Loopback0
 tunnel mode ipv6ip 6to4
 ipv6 address 2002:9601:808::8/64
```

https://t.me/learningnets

```
!
ipv6 route 2002::/16 Tunnel58
!
interface Loopback58
 ipv6 address 2002:9601:808:1::8/64
```

Verification

Automatic 6to4 tunnels are multipoint by design. The idea is to allow automatic routing across an IPv4 cloud based on a part of the destinations IPv6 address. Specifically, the format of the 6to4 IPv6 address is as follows:

```
2002 (16 bits):IPv4 address (32 bits):Subnet ID(16 bits):Interface ID (64 bits)
```

When a packet is routed across the 6to4 tunnel, the router extracts the IPv4 address embedded in the IPv6 address and uses it to build the IPv4 destination address of the tunnel header. The receiving router strips the header, extracts the IPv6 packet, and routes it based on the IPv6 routing table. As you can imagine, 6to4 subnets have some addressing restrictions. First, you must use the 16-bit prefix 2002, because it is the common reservation for all 6to4 deployments. Second, you must select the public IPv4 address used to create the /48 prefix. It is common to pick any interface of the border router and then allocate the /64 subnets to other devices on the network, as long as the address is publicly routable.

6to4 tunnels are a transition mechanism for hosts that do not have native IPv6 connectivity, allowing them to reach other nodes that have full connectivity to the IPv6 Internet. Because of the multipoint nature of the 6to4 tunnels, only static routing is possible with this technology. However, it is common to simply route the whole 2002::/16 prefix to the 6to4 tunnel. In our case, the use of Loopback 0 subnets results in the following IPv6 6to4 prefixes:

```
R5: 150.1.5.5 = 2002:9601:505::/48
R8: 150.1.8.8 = 2001:9601:808::/48
```

“9601” in hex corresponds to “150.1” in decimal, and on R5, “505” corresponds to “5.5”.

```
R5#show interface tunnel58
Tunnel58 is up, line protocol is up
    Hardware is Tunnel
MTU 17872 bytes, BW 100 Kbit/sec, DLY 50000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
```

https://t.me/learningnets

```
Encapsulation TUNNEL, loopback not set
    Keepalive not set Tunnel source 150.1.5.5 (Loopback0)
    Tunnel Subblocks:
        src-track:
        Tunnel58 source tracking subblock associated with Loopback0
      Set of tunnels with source Loopback0, 1 member (includes iterators), on interface <OK>
Tunnel protocol/transport IPv6 6to4
Tunnel TTL 255
Tunnel transport MTU 1480 bytes
Tunnel transmit bandwidth 8000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
Last input never, output never, output hang never
Last clearing of "show interface" counters 00:11:07
Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
    Queueing strategy: fifo
!
!R5#ping 2002:9601:808:1::8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2002:9601:808:1::8, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
!
!R8#ping 2002:9601:505:1::5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2002:9601:505:1::5, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 3/11/25 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPv6

ISATAP Tunneling

> You must load the initial configuration files for the section, **IPv6 Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs IPv6 Diagram** to complete this task.

Task

- Configure ISATAP tunnel on R5 and R8 using their Loopback0 as the source.
- Use the /64 IPv6 prefix 2001:1:0:58::/64 to allocate IPv6 addresses to the tunnel endpoints.
- Create additional IPv6 Loopback interfaces on both the routers with the IPv6 addresses **2001:1:0:Y::Y/64**, where **Y** is the router number.
- Use static routing to obtain full connectivity.

Configuration

```
R5:
interface Tunnel58
 ipv6 address 2001:1:0:58::/64 eui-64
 tunnel source Loopback0
 tunnel mode ipv6ip isatap
!
interface Loopback58
 ipv6 address 2001:1:0:5::5/64
!
ipv6 route 2001::/16 2001:1:0:58:0:5EFE:9601:808
R8:

interface Tunnel58
 ipv6 address 2001:1:0:58::/64 eui-64
 tunnel source Loopback0
```

https://t.me/learningnets

```
 tunnel mode ipv6ip isatap
!
interface Loopback58
 ipv6 address 2001:1:0:8::8/64
!
ipv6 route 2001::/16 2001:1:0:58:0:5EFE:9601:505
```

Verification

ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) is a technology that contrasts 6to4 automatic tunnels. Whereas 6to4 automatically generates a /48 prefix out of an IPv4 address, ISATAP uses the IPv4 domain as a “multi-access” media with IPv4 addresses being the rough equivalent of Ethernet MAC addresses. Specifically, ISATAP constructs the interface identifier (last 64 bits) of the IPv6 address based on the IPv4 address of a host using the EUI-64 rules. Thus, because you already have a /64 IPv6 prefix, you can allocate IPv6 addresses to tunnel endpoints performing simple manipulations over the IPv4 endpoint addresses. Specifically, the interface ID is constructed as follows:

```
EUI-64 = 0000 (16 bits) + 5EFE (16 bits) + IPv4 Address (32 bits).
```

Therefore, if you select the prefix 2001:1:0:58::/64, R5 will have the IP address:

```
2001:1:0:58:0:5efe:9601:0505/64
```

Where 9601 is for 150.1 and 0505 is for last two octets of the IP address. When configuring an IOS router for ISATAP, you can automatically generate interface IDs with the **`eui-64`** keyword. Unlike the 6to4 tunnels, ISATAP tunnels cannot automatically extract the destination. Therefore, you must use static routes that point to the exact IPv6 endpoint on the other end of the tunnel.

```
R5#show interfaces tunnel58
Tunnel58 is up, line protocol is up
Hardware is Tunnel
 MTU 17872 bytes, BW 100 Kbit/sec, DLY 50000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation TUNNEL, loopback not set
 Keepalive not set Tunnel source 150.1.5.5 (Loopback0)
Tunnel Subblocks:
    src-track:
     Tunnel58 source tracking subblock associated with Loopback0
      Set of tunnels with source Loopback0, 1 member (includes iterators), on interface <OK>
Tunnel protocol/transport IPv6 ISATAP
```

https://t.me/learningnets

```
Tunnel TTL 255
Tunnel transport MTU 1480 bytes
Tunnel transmit bandwidth 8000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
Last input never, output never, output hang never
Last clearing of "show interface" counters 00:13:30
Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
<snip>
!
!R5#show ipv6 interface tunnel58
Tunnel58 is up, line protocol is up
 IPv6 is enabled, link-local address is FE80::5EFE:9601:505
No Virtual link-local address(es):
Global unicast address(es): 2001:1:0:58:0:5EFE:9601:505
, subnet is 2001:1:0:58::/64 [EUI]
Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF01:505
MTU is 1480 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ICMP unreachables are sent
ND DAD is not supported
ND reachable time is 30000 milliseconds (using 30000)
ND advertised reachable time is 0 (unspecified)
ND advertised retransmit interval is 0 (unspecified)
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
ND advertised default router preference is Medium
ND RAs are suppressed (periodic)
Hosts use stateless autoconfig for addresses.
!
!R5#ping 2001:1:0:58:0:5EFE:9601:808
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:1:0:58:0:5EFE:9601:808, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/8/14 ms
!
!R5#ping 2001:1:0:8::8

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:1:0:8::8, timeout is 2 seconds:!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/7/23 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Classification and Marking

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure an outbound MQC policy on R4's Ethernet link to R5 according to the following requirements:
  - HTTP traffic from servers on VLAN 146 should be marked with an IP Precedence of 2.
  - VoIP packets with UDP ports in the destination range of 16384–32767, and a Layer 3 packet size of 60 bytes should be marked with DSCP EF.
  - ICMP packets larger than 1000 bytes should be marked with IP Precedence of 0.
  - All other packets that come from any of R4's links with an IP precedence of 0 should be remarked with an IP precedence of 1.
- Do not use an access-list to classify ICMP packets.

Configuration

R4:

```
ip access-list extended HTTP
 permit tcp 155.1.146.0 0.0.0.255 eq www any
!
ip access-list extended VOICE
 permit udp any any range 16384 32767
!
```

https://t.me/learningnets

```
class-map HTTP
 match access-group name HTTP
!
class-map match-all LARGE_ICMP
 match protocol icmp
 match packet length min 1001
!
class-map match-all VOICE
 match access-group name VOICE
 match packet length min 60 max 60
!
class-map match-all SCAVENGER
 match ip precedence 0
!
policy-map ETHERNET_LINK_TO_R5
 class VOICE
  set ip dscp ef
 class HTTP
  set ip precedence 2
 class LARGE_ICMP
  set ip precedence 0
 class SCAVENGER
  set ip precedence 1
!
interface GigabitEthernet1.45
 service-policy output ETHERNET_LINK_TO_R5
```

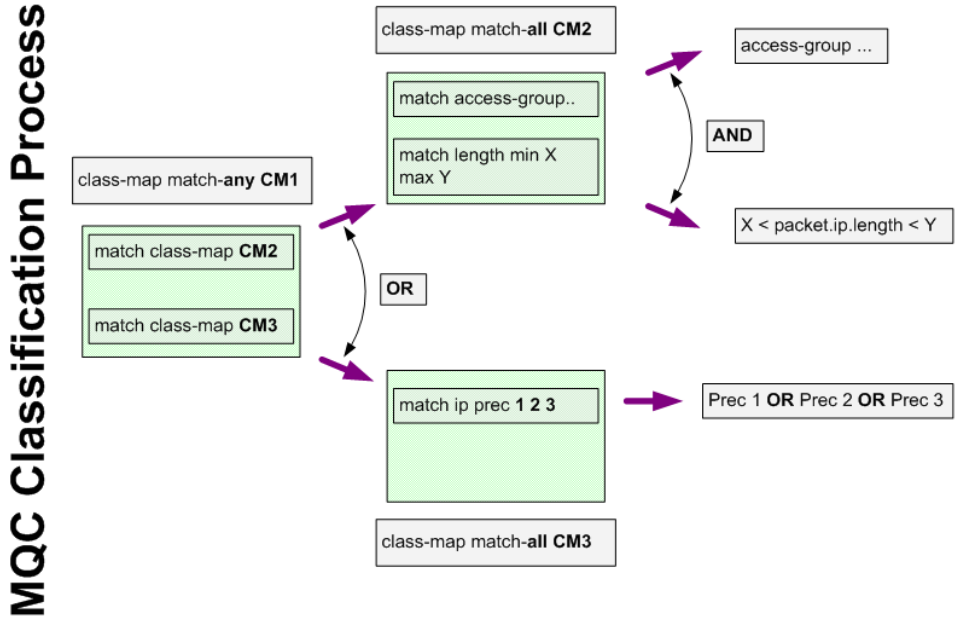
Verification

The Modular Quality of Service Command Line Interface (MQC), also known as Class-Based Weighted Fair Queueing (CBWFQ), unifies all IOS QoS features under a single interface. MQC allows the implementation of a full suite of QoS tools, including classification, congestion management, traffic metering, marking, traffic shaping, and link efficiency. The main advantage of using the MQC over the legacy methods is that multiple QoS features can be applied to the same interface in the same direction. For example, with legacy QoS, you can't apply custom queueing and priority queueing at the same time, but with MQC you can.

Classification in MQC uses case sensitive *class-maps* (not to be confused with a frame-relay map-class) to group criteria. Each class-map performs a logical AND (match-all) or a logical OR (match-any) on its criteria. In other words, in a match-all class-map, all matches must be TRUE for the class to be TRUE. Class-maps can be nested inside other maps to build complicated classification "AND-OR" logic gates. If

https://t.me/learningnets

multiple match criteria appear on the same line (for example, match ip dscp, or match ip precedence), they are treated as a logical OR match.



Different IOS versions and platforms support different matches in the class-map, but as a general rule the following classification criteria are supported:

- Named and numbered access-lists: allows matching of IP addresses, TCP/UDP ports, IP protocol numbers, etc.
- Layer 3 packet length
- Layer 2 addresses: source/destination MAC address, Frame-Relay DLCI, etc.
- Packet marking: Layer 2 CoS, Layer 3 DSCP/IP precedence, Frame Relay DE, ATM CLP, MPLS EXP, etc.
- Network-Based Application Recognition (NBAR)
- Inverse logical matching (logical NOT)

When you apply a logical NOT to a nested class-map or multiple criteria in a single line, *De Morgan's law* applies, where NOT (X AND Y) = (NOT X) OR (NOT Y), and NOT (X OR Y) = (NOT X) AND (NOT Y).

When classification is configured in a class-map, actions are defined for the different classes in a case-sensitive *policy-map*. A policy map is an *ordered* list of class-maps with their corresponding actions, similar to a route-map. The router matches packets entering/leaving the interface against all class-map entries in the respective input/output policy-map on the interface in a top-down fashion. This means that the first match in a class-map is used for classification, which implies that the order of the classes called in the policy-map is significant. The policy-map actions include marking, shaping, policing, assigning queue weight, compressing, etc. Any

https://t.me/learningnets

unclassified traffic in a policy-map falls into the *class-default* category, which is covered in depth, along with the policy-map actions, in the following sections.

Pitfall

Correct traffic flow classification within the class-map, and the correct order of operations in the policy-map, is important in the implementation of an MQC policy. In this task, you are asked to classify traffic flows from web servers in VLAN 146, which means that they will be using *source port* 80 in their responses to clients. Additionally, the SCAVENGER class-map, which matches IP Precedence 0 traffic, may overlap other traffic classes, such as the HTTP class, which makes it important that SCAVENGER is called last in the policy-map to match any un-classified traffic up to that point.

To verify this configuration, start by shutting down R5's DMVPN Tunnel. Next, enable the HTTP server service on R1 as well as HTTP authentication, and start transferring an IOS image from R1 to R8. Start an IP SLA jitter operation on R6 to source "voice-like" packets with the G.729 codec (60 bytes each), and finally send a large number of ICMP packets from R6 to R5, each larger than 1000 bytes. Also, be sure to tune down the load interval on R4's GigabitEthernet1 interface in order to get faster statistics.

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:

R4:
interface GigabitEthernet1
 load-interval 30

R5:
interface Tunnel0
 shutdown
!
ip sla responder

R6:
ip sla 1
 udp-jitter 155.1.45.5 16384 codec g729a control enable
 threshold 1000
 timeout 1000
 frequency 1
!
ip sla schedule 1 life forever start-time now
```

https://t.me/learningnets

```
R8#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
Accessing http://*****:*****@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg...
Loading http://***********@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg !!!!
!!!
```

```
    Class-map: SCAVENGER (match-all)
      0 packets, 0 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 0
      QoS Set
        ip precedence 1
          Marker statistics: Disabled

    Class-map: class-default (match-any)  3 packets, 234 bytes

      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Bandwidth Reservations and CBWFQ

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure an MQC policy on R4's to meet the following requirements for its Ethernet link to R5:
  - Shape the Ethernet link between R4 and R5 to 1.5 Mbps.
  - The traffic flows for the web servers in VLAN 146 and IP Precedence 0 packets should be guaranteed 32Kbps, each.
  - Limit the sizes of the FIFO queues for the HTTP and IP Precedence 0 traffic classes to 16 and 24 packets, respectively.
  - All other unmatched traffic in the policy should run WFQ.
  - Dynamic flows in this WFQ should start dropping when they reach 32 packets in length.

Configuration

R4:

```
ip access-list extended HTTP
 permit tcp 155.1.146.0 0.0.0.255 eq www any
!
class-map HTTP
 match access-group name HTTP
!
class-map match-all SCAVENGER
```

https://t.me/learningnets

```
 match ip precedence 0
!
policy-map ETHERNET_LINK_TO_R5
 class HTTP
  bandwidth 32
  queue-limit 16
 class SCAVENGER
  bandwidth 32
  queue-limit 24
 class class-default
  fair-queue
  queue-limit 32
!
policy-map PARENT_SHAPE
 class class-default
  shape average 1500000
   service-policy ETHERNET_LINK_TO_R5
!
interface GigabitEthernet1.45
 service-policy output PARENT_SHAPE
```

Before getting started with this section, it is important to take note of the current INE RSv5 topology in regards to QoS. The topology makes use of Ethernet sub-interfaces as means of interconnecting all of the devices using a single physical Ethernet link. Although this greatly simplifies the physical topology (or in our case virtual topology made up of CSR1000v routers), it imposes a challenge to the way QoS operates. Cisco IOS logical interfaces such as Ethernet Sub-Interfaces don't have a way to get the state of congestion that their underlying physical, or "main", interface may be experiencing. Due to this reason, a queuing policy applied directly to a sub-interface would have no way to know when the link is congested, and thus would never trigger. As such, Cisco IOS does not allow direct application of a policy-map that uses any sort of queuing policies directly to sub-interfaces.

A way to overcome this limitation is to apply a policy that shapes the rate of the sub-interface in order to create artificial congestion. Based on the defined shaped rate, child policies doing queuing can be attached to the sub-interface under the parent shaping policy. The applied queuing policies could then be based on the shaped rate and could be triggered when there is congestion, as defined by the rate of the shaper. This type of configuration is referred to as HQF, or Hirearchical Queuing Framework, and will be discussed in detail at a later section. Shaping will also be discussed in detail at a later section. For now just think of it as a way to assign an artificial congestion rate to the sub-interface so that the queuing policies that we define are able to work properly based on the shaped rate.

https://t.me/learningnets

In this exercise we are applying a parent shaping policy that shapes the rate of the Ethernet sub-interface to 1.5 Mbps. The queuing policies applied under this parent policy will guarantee 32 Kbps of bandwidth to each of the corresponding classes when there is congestion on the sub-interface *based on the 1.5 mbps, not on 1gbps*.

In production networks this type of setup is commonly used when a CE device connecting to a Metro-E circuit has to use Dot1q tags. The router or switch at the customer site uses an Ethernet sub-interface to connect to the provider with the Dot1q tag that the provider specifies. Commonly the physical link is either FastEthernet or GigabitEthernet, however the actual purchased bandwidth is less than the speed of the link. If the customer purchases a 10 Mbps link to the Metro-E network, then a way to ensure that the CE device can apply QoS policies based on 10 Mbps and not 100 Mbps or 1 Gpbs is to use a parent shaping policy of 10 Mbps (just like we are using in this example), and then apply the desired queuing policies underneath. Doing this has other benefits that will be discussed in the Shaping section.

If the provider does not require the use of Dot1q tags, then it would still be necesary to use a parent shaping policy in order to introduce the artificial congestion point. Although a physical interface can detect congestion, it will do so at the configured speed of the link (100 Mbps, 1 Gbps). Sub-interfaces on the other hand have no way to detect congestion, as mentioned previously.

Verification

Class-Based Weighted Fair Queueing (CBWFQ) is the MQC equivalent of a combination of the legacy interface level `fair-queue` command and the custom-queue. CBWFQ is used to reserve a *minimum* amount of bandwidth in the output queue for a particular traffic flow in the case of congestion. The `bandwidth` statement used in the policy-map for the class is used to compute the scheduling weight of the traffic flow, relative to the configured interface bandwidth.

https://t.me/learningnets

The logic of CBWFQ is that during congestion, a class with a bandwidth reservation of *ClassBandwidth* will have at least a *ClassBandwidth/InterfaceBandwidth* share of the total interface bandwidth. For this reason, it is important that the interface-level `bandwidth` statement is configured correctly, in respect to the access rate or guaranteed rate of the link in question whenever an MQC policy is applied. The `bandwidth` statement under the policy-map does not have a built-in policer, which means that if congestion is not occurring, the traffic flow matched by the class can use as much bandwidth as it wants. Based on this fact, the CBWFQ reservation only becomes active when congestion occurs. Configuration of this feature is much more straightforward than the legacy custom-queue, because the bandwidth reserved is not based on a relative ratio like it is in the legacy version.

The feature is deemed Class-Based Weighted Fair Queueing, because the WFQ flow is manually defined through the MQC class-map. The weighting value for the flow is then defined through the `bandwidth` value under the respective policy-map. When the policy-map is applied to the interface, the entire software queue changes to CBWFQ. The CLI will not allow you to assign a service-policy with CBWFQ weights unless the interface is using FIFO queuing, which implies that CBWFQ is not compatible with any of the legacy queuing methods, such as custom queueing or priority queueing. These must be disabled explicitly before applying the service-policy.

Each class configured with a `bandwidth` statement under the policy-map has its own dedicated FIFO queue in the interface's CBWFQ conversation pool. The depth of each FIFO queue can be changed on a per-class basis with the `queue-limit` command. The overall WFQ settings, such as the CDT and the total queue size, can be set using the `queue-limit` under class-default, and the `hold-queue <number> out` command at the interface level.

Recall that with legacy WFQ, flows are classified automatically. The scheduler shares the bandwidth in proportion to the flow's IP precedence value, normalized against the sum of other flows' precedences. Specifically, if there are *N* active flows, and flow *i* has IP Precedence IPP*(i)*, then any flow *k* is guaranteed a share as:

Share(k) = (IPP(k)+1)/(IPP(1)+IPP(2)…+IPP(N)+N)*100%

Note that each precedence value is shifted by 1 to ensure non-zero values. Because WFQ implements max-min sharing, any flow may claim unused interface bandwidth. Finally, the implementation uses a computation "weight" value based on the formula:

Weight(i) = 32384/(IPP(i)+1)

[Formula 1]

This allows for the flow with higher IP precedence to have lower computation

https://t.me/learningnets

weight, and larger share of bandwidth. Remember that computation weights are inversely proportional to the actual weights you use to compute the share of the bandwidth, because the bandwidth is shared in the proportion [1/Weight(1):1/(Weight(2)… :1/Weight(N)]. For example, WFQ assigns a weight of 4048 to a flow with an IP precedence of 7, which is the maximum IPP value. This is an important fact that will be referenced again later.

With the new MQC format, CBWFQ retains the above computations for *unclassified* traffic only. Traffic flows that are explicitly matched in a class-map and have a bandwidth reservation configured are treated differently. To start, this class uses a separate flow queue with its own limits. Second, this flow is assigned a computation weight based on the following formula:

Weight(i) = Const*InterfaceBandwidth/Bandwidth(i)

[Formula 2]

Here *Bandwidth(i)* is the value configured under the respective class using the `bandwidth` keyword, and *InterfaceBandwidth* is the value configured under the main interface using the `bandwidth` keyword, or in the case there is a parent shaping policy in use, the rate of the shaper (such as in this example). The value of the constant *Const* depends on the number of flow queues active in CBWFQ, and varies from 1 to 64. The implications of the above formula's calculations are as follows.

Almost any manually configured class has a weight value significantly lower than any automatically classified flow. Recall that IP Precedence 7, the best value, has a best possible weight of 4048 for a dynamic conversation. However, for a manually defined class, even in the worst case, *Const* equals 64 (with 32 flow-queues), the ratio of *interfaceBandwidth/Bandwidth* should be less than 0.02 to be comparable to 4048. This means that a class should have less than a 2% reservation of the interface's bandwidth, which is rare. Furthermore, because any user-defined class has its own separate FIFO queue, we can conclude that CBWFQ isolates it from congestive discard drops performed across dynamic WFQ queues. Instead, each class queue is FIFO with a tail-drop discipline by default.

The next question is how CBWFQ shares bandwidth between the user-defined classes. Looking at the weight calculation, we can assume that each class has a relative bandwidth share of

Share(i) = Bandwidth(i)/InterfaceBandwidth

Recall that by default, the sum of all *Bandwidth(i)* reservation is no more than 75% of *InterfaceBandwidth*, because of the `max-reserved-bandwidth` defaults. So what happens with the remaining 25%? This is where class-default comes in.

https://t.me/learningnets

As soon as the **bandwidth** keyword is specified under any user-defined class, the interface queue turns into CBWFQ. This means that any unmatched flows that fall back into class-default are scheduled using dynamic WFQ weights. This means that automatic classification occurs, along with precedence-based weight assignment and sharing of the single buffer space of WFQ. This behavior is default, even if you did not configure **fair-queue** under the class-default.

If you want to disable fair-queue for unclassified packets, an explicit **bandwidth** value for the class-default can be configured, which turns it into a single FIFO queue. For example, the following code snippet disables fair-queue for unclassified traffic, and gives it a static weight as relative to 96Kbps:

```
policy-map TEST
 class class-default
   bandwidth 96
```

Moreover, if you do not define any classes other than class-default, and class-default has a bandwidth value defined, the entire interface queue essentially becomes a FIFO queue.

In summary, the key point about CBWFQ is that it uses the same scheduling logic as the legacy WFQ, but user-configurable classes have a special low-weight, making them more important than any dynamic conversation. Thus, user-defined classes always get the guaranteed proportion of bandwidth, whereas flows using class-default may starve in the case of congestion.

For verification of this task, shut down R5's DMVPN link to force the traffic from VLAN 146 to the hosts behind R4 to take the path across the Ethernet link. Configure R1 as an HTTP server, and set R8 to transfer the IOS image from R1, causing the shaped Ethernet sub-interface to be congested. Then, generate a flow of ICMP packets from R6 to R5, simulating the SCAVENGER class traffic.

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:

R4:
interface GigabitEthernet1
 load-interval 30

R5:
interface Tunnel0
 shutdown
```

https://t.me/learningnets

```
R8#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
Accessing http://*****:*****@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg...
Loading http://***********@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg !!!!
!!!
```

```
    Queueing queue limit 16 packets
    (queue depth/total drops/no-buffer drops) 6/0/0
    (pkts output/bytes output) 319823/189954110 bandwidth 32 kbps


  Class-map: SCAVENGER (match-all)  3871380 packets, 317350160 bytes
    30 second offered rate 279000 bps, drop rate 0000 bps
    Match: ip precedence 0
    Queueing  queue limit 24 packets
    (queue depth/total drops/no-buffer drops) 1/9916/0
    (pkts output/bytes output) 3861464/302654792 bandwidth 32 kbps


  Class-map: class-default (match-any)
    17946 packets, 1420076 bytes
    30 second offered rate 0000 bps, drop rate 0000 bps
    Match: any
    Queueing queue limit 32 packets
    (queue depth/total drops/no-buffer drops/flowdrops) 0/0/0/0
    (pkts output/bytes output) 17946/1420076 Fair-queue: per-flow queue limit 8 packets
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Bandwidth Percent

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure an MQC policy on R4's to meet the following requirements for its Ethernet link to R5:
  - Shape the Ethernet link between R4 and R5 to 1.5 Mbps.
  - The traffic flows for the web servers in VLAN 146 and IP Precedence 0 packets should be guaranteed 25% of the bandwidth each.

Configuration

R4:

```
ip access-list extended HTTP
 permit tcp 155.1.146.0 0.0.0.255 eq www any
!
class-map HTTP
 match access-group name HTTP
!
class-map match-all SCAVENGER
 match ip precedence 0
!
policy-map ETHERNET_LINK_TO_R5
 class HTTP
  bandwidth percent 25
 class SCAVENGER
```

https://t.me/learningnets

```
  bandwidth percent 25
!
policy-map PARENT_SHAPE
 class class-default
  shape average 1500000
   service-policy ETHERNET_LINK_TO_R5
!
interface GigabitEthernet1.45
 service-policy output PARENT_SHAPE
```

Verification

Recall that CBWFQ does not use the absolute **`bandwidth`** value in a class to directly compute the scheduling weight, but instead it is computes relative to the interface's bandwidth, or the defined shaped rate, value as Weight(i) = Const*InterfaceBandwidth/Bandwidth(i). Because this value is in reality a relative reservation in the first place, the calculation can be simplified by ignoring the interface-level bandwidth or shaped rate and expressing the reservation as a percentage. In this manner with the **`bandwidth percent`** command, the weight calculation changes to:

Weight(i) = Const*100/Percent(i)

This implementation is useful in designs where a common template of reservation is applied to multiple links of differing bandwidth values. For example, a standard policy-map could be defined to reserve 20% of link bandwidth for HTTP flows, and then applied to both 100Mbps FastEthernet and 45Mbps DS3. The resulting CBWFQ weights would be the same, but the actual bandwidth value differs based on the underlying physical link bandwidth.

Like the Kbps reservation through the **`bandwidth`** command, the sum of all configured **`bandwidth percent`** values in all classes of a policy-map cannot exceed the total bandwidth of the interface, or the defined shaped rate. In addition, it is not possible to mix the **`bandwidth`** and **`bandwidth percent`** commands in the same policy-map; the units must be the same among classes.

Verification of this configuration is the same as the previous task. Shut down R5's DMVPN link to force the traffic from VLAN 146 to the hosts behind R4 to take the path across the Ethernet link. Configure R1 as an HTTP server, and set R8 to transfer the IOS image from R1, oversubscribing the shaped Ethernet sub-interface. Then, generate a flow of ICMP packets from R6 to R5, simulating the SCAVENGER class traffic.

```
R1:
```

https://t.me/learningnets

```
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:

R4:
interface GigabitEthernet1
 load-interval 30

R5:
interface Tunnel0
 shutdown

R8#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
Accessing http://*****:*****@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg...
Loading http://***********@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg !!!!
!!!
```

```
(queue depth/total drops/no-buffer drops) 0/9916/0
(pkts output/bytes output) 5516064/997058444 shape (average) cir 1500000, bc 6000, be 6000
target shape rate 1500000

Service-policy : ETHERNET_LINK_TO_R5

  Class-map: HTTP (match-all)  8549 packets, 5071918 bytes
    30 second offered rate 736000 bps, drop rate 0000 bps
    Match: access-group name HTTP
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 7/0/0
    (pkts output/bytes output) 8549/5071918 bandwidth 25% (375 kbps)

  Class-map: SCAVENGER (match-all)  10007 packets, 1392876 bytes
    30 second offered rate 192000 bps, drop rate 0000 bps
    Match: ip precedence 0
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 1/0/0
    (pkts output/bytes output) 10007/1392876 bandwidth 25% (375 kbps)


  Class-map: class-default (match-any)
    16 packets, 1292 bytes
    30 second offered rate 0000 bps, drop rate 0000 bps
    Match: any

    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 16/1292
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC LLQ and Remaining Bandwidth Reservations

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure an MQC policy on R4's to meet the following requirements for its Ethernet link to R5:
  - Shape the Ethernet link between R4 and R5 to 1.5 Mbps.
  - Configure Low Latency Queueing to allow for exactly one VoIP call to be prioritized.
  - Additional VoIP calls should be allowed only if the link is not congested, but they should not be prioritized.
  - Reserve 33% of the remaining bandwidth to HTTP traffic from servers on VLAN 146, and 33% of the remaining bandwidth to traffic marked with IP precedence 0.
  - Assume that VoIP calls are using the G.729 codec, which generates 50 packets per second with a Layer 3 size of 60 bytes.

Configuration

```
R4:

ip access-list extended HTTP
 permit tcp 155.1.146.0 0.0.0.255 eq www any
!
ip access-list extended VOICE
```

https://t.me/learningnets

```
 permit udp any any range 16384 32767
!
class-map HTTP
 match access-group name HTTP
!
class-map match-all VOICE
 match access-group name VOICE
 match packet length min 60 max 60
!
class-map match-all SCAVENGER
 match ip precedence 0

policy-map ETHERNET_LINK_TO_R5
 class VOICE
  priority 32
 class HTTP
  bandwidth remaining percent 33
 class SCAVENGER
  bandwidth remaining percent 33
!
policy-map PARENT_SHAPE
 class class-default
  shape average 1500000
   service-policy ETHERNET_LINK_TO_R5
!
interface GigabitEthernet1.45
 service-policy output PARENT_SHAPE
```

Verification

The Low Latency Queuing (LLQ) feature is the MQC equivalent of the legacy IP RTP Priority, but classification is not limited to just RTP packets. Like IP RTP Priority, LLQ uses the special conversation number 2^N+8, where N defines the number of dynamic conversations. For example, if there are 32 (2^5) dynamic queues, the LLQ conversation is number 40. This conversation has a weight value of 0, which means that it is always serviced first. To prevent starvation of other queues, the packets de-queued from the LLQ conversation are metered using a simple token bucket, with a configurable rate and burst size. Packets that exceed the token bucket are dropped (policed) during times of congestion; if there is no congestion, exceeding traffic is not dropped, but it is simply not prioritized. For these reasons, it is better to use the MQC LLQ feature in practical designs vs. the legacy priority-queue or IP RTP Priority, because the legacy priority-queue does not have a policer, IP RTP Priority cannot classify other traffic flows, and neither of them can be

https://t.me/learningnets

used in conjunction with other QoS features, such as a bandwidth reservation.

Multiple classes inside a single policy-map can use the priority keyword, but only a single priority queue exists. This design of multiple priority classes is used to ensure that one priority flow does not starve another priority flow. For example, assume that VoIP and Video traffic are grouped together in one class-map, which has a `priority 128` value defined under a policy-map. If Video traffic is using 128Kbps, VoIP traffic can potentially be dropped (policed), or at least not be guaranteed priority. This is because both traffic flows must contend for the same 128Kbps rate. However, if separate VoIP and Video class-maps are defined, each with a `priority 64` rate configured under the same policy-map, each flow is guaranteed priority up to 64Kbps. Although this still totals 128Kbps of priority, one class can't completely starve the other.

Note that the LLQ policer takes the layer 2 packet length into account, so in this case the Ethernet overhead of 18 bytes is added to the 60 bytes of layer 3 VoIP payload, which results in a value close to 32Kbps (78 bytes/packet * 50 packets/second * 8 bits/byte = 31200bps) being reserved. The burst value for the token bucket can be configured manually, or automatically calculated by the IOS itself. A burst size of 1 or 1.5 seconds should be enough, because this is probably the maximum duration of a single spoken word in VoIP. Refer to the **further learning** section for more information about computing LLQ sizes for VoIP calls:

When the LLQ priority reservation is configured, the CBWFQ algorithm subtracts the reserved bandwidth of the priority queue from the interface's available bandwidth or from the available rate based on the shaping policy. The remaining bandwidth can be used to create a relative bandwidth reservation for other classes in the CBWFQ. By this logic, instead of configuring absolute bandwidth values (either numerically or in interface bandwidth percents) for other reservations, you can just specify relative shares of the bandwidth that remains after the priority queue finished its run. This method is seen in the above configuration through the `bandwidth remaining percent` command.

For example, in this particular task, the link between R4 and R5 was configured with a shaping policy of 1.5 Mbps, meaning that 100% of 1.5 Mbps is reservable. With the 32Kbps reservation for VoIP in the LLQ, 1468Kbps is the remaining bandwidth. Because the HTTP and SCAVENGER classes reserve 33% of the remaining bandwidth, they are each allocated a minimum of 484Kbps ((1500-32) * .33 = 484).

To verify this configuration, generate three different traffic flows, as in the previous examples. First, shut down the DMVPN Tunnel interface of R5 to force the traffic from VLAN 146 to hosts behind R5 to take path across the Ethernet link. Configure R6 to send IP SLA jitter probes to R5, and R5 as an IP SLA Responder; these flow simulates the voice traffic in the LLQ. Next, configure R1 as an HTTP server, and

https://t.me/learningnets

set R8 to transfer the IOS image from R1, causing the shaped Ethernet sub-interface to be congested. Finally, generate a flow of ICMP packets from R6 to R5, simulating the SCAVENGER class traffic.

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
R4:
interface GigabitEthernet1
 load-interval 30
R5:
interface Tunnel0
 shutdown
!
ip sla responder

R6:
ip sla 1
 udp-jitter 155.1.45.5 16384 codec g729a control enable
 threshold 1000
 timeout 1000
 frequency 1
!
ip sla schedule 1 life forever start-time now

R8#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
Accessing http://*****:*****@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg...
Loading http://***********@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg !!!!
!!!
```

Check the statistics for policy map matches. Note that the priority queue for VOICE traffic has a 30 seconds offered of 30 kbps. Because the voice packets use the priority queue now, the remaining classes have 32kbps less bandwidth to share. The bandwidth measured for the HTTP class is 816Kbps. The SCAVENGER class is not using all of its available bandwidth, only 94kbps, allowing the HTTP class to use the remaining bandwidth. Note that the 33kbps of voice traffic will always be prioritized, even during periods of high congestion on the link. Also, if there is an increase of traffic in the SCAVENGER class, it will be able to reclaim some of its available bandwidth; it will take it back from the HTTP class, causing dropped packets in the HTTP class.

```
R4#show policy-map interface GigabitEthernet1.45
 GigabitEthernet1.45

  Service-policy output: PARENT_SHAPE

    Class-map: class-default (match-any)
      25787 packets, 5761441 bytes
      30 second offered rate 940000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 25786/5761273 shape (average) cir 1500000, bc 6000, be 6000
      target shape rate 1500000

      Service-policy : ETHERNET_LINK_TO_R5

        queue stats for all priority classes:
          Queueing
          queue limit 512 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 16562/1291836

        Class-map: VOICE (match-all)  16487 packets, 1285986 bytes
 30 second offered rate 30000 bps, drop rate 0000 bps
          Match: access-group name VOICE
          Match: packet length min 60 max 60 Priority: 33 kbps, burst bytes 1500, b/w exceed drops: 0



        Class-map: HTTP (match-all)  6890 packets, 4085377 bytes
 30 second offered rate 816000 bps, drop rate 0000 bps
          Match: access-group name HTTP
          Queueing
          queue limit 64 packets
```

https://t.me/learningnets

```
          (queue depth/total drops/no-buffer drops) 7/0/0
          (pkts output/bytes output) 6890/4085377 bandwidth remaining 33%

        Class-map: SCAVENGER (match-all)   2256 packets, 377888 bytes
 30 second offered rate 94000 bps, drop rate 0000 bps
          Match: ip precedence 0
          Queueing
          queue limit 64 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 2256/377888 bandwidth remaining 33%



        Class-map: class-default (match-any)
          78 packets, 6172 bytes
          30 second offered rate 0000 bps, drop rate 0000 bps
          Match: any

          queue limit 64 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 78/6172
```

Further Learning

Computing voice bandwidth is usually required for scenarios where you provision an LLQ based on the number of calls and VoIP codec used. To accomplish this, you need to account for codec payload rate, the Layer 3 overhead (RTP and UDP headers), and the Layer 2 overhead (Frame-Relay, Ethernet, HDLC, etc. headers). Accounting for Layer 2 overhead is important because the LLQ policer takes this overhead into account when enforcing the maximum rate.

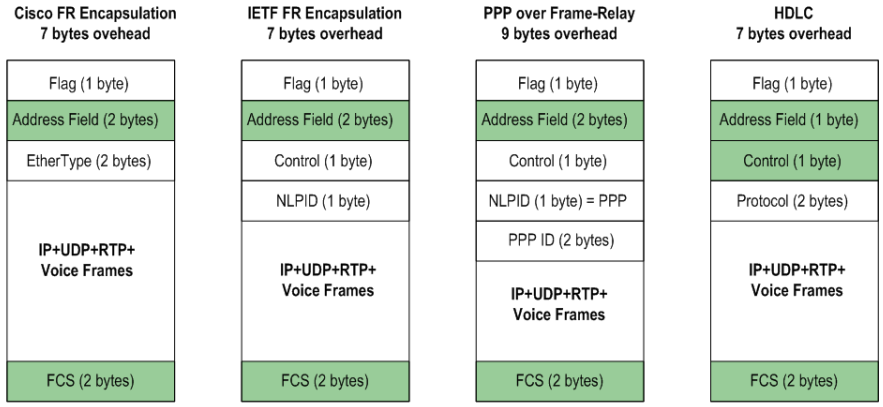
For this example, let’s consider two codecs for bandwidth computation, G.729 and G.711. By default, both codecs generate 50 VoIP packets per second, but the codec framing rate is 10ms (100 packets per second). Thus, each VoIP packet carries two frames with VoIP samples. The frame sizes are 10 bytes and 80 bytes for G.729 and G.711, respectively. Therefore, G.729 generates (10*2)*50*8=8000bps and G.711 generates (80*2)*50*8=64000bps of payload rate.

The RTP header size is 12 bytes, and the UDP header size is 8 bytes. A typical IP header (with no options) is 20 bytes. Therefore, the Layer 3 overhead is 40 bytes, if we don’t use header compression.

The following are the formats WAN frames commonly uses to transport voice

https://t.me/learningnets

(with or without FRF.12/MLP fragmentation—voice packets are never fragmented).



As we can see, both Cisco and IETF Frame-Relay encapsulations add 7 bytes of Layer 2 overhead to VoIP packets. The same is true for HDLC encapsulation (which is not very common but added here for completeness). PPP over Frame-Relay adds 9 bytes of overhead—the maximum overhead of all encapsulation types.

Using the information above, you can compute bandwidth usage for uncompressed voice traffic flow across any WAN connection. For example, let’s compute the bandwidth consumption for 3 G.729 calls across Frame-Relay link with FRF.12 fragmentation. First, FRF.12 does not fragment voice packets if set up properly, because the fragment size is greater than or equal to the VoIP packet. Thus, we assume 7 bytes of Layer 2 overhead for any of the Frame Relay encapsulation types. Next, the size of the payload + Layer 3 overhead is 20 bytes + 40 bytes = 60 bytes. Based on the 50pps rate, we get the bandwidth value of (20+40+7)*50*8=26800bps. If you want to use the G.711 codec, replace the 20 bytes payload with 160 bytes. The result is (160+40+7)*50*8=82800bps.

Another thing to consider is IP/RTP/UDP header compression. Cisco’s implementation reduces the total overhead of 40 bytes (12+8+20) down to 2 bytes (no UDP checksum). This limits the Layer 3 overhead to just 2 bytes. Based on this reduction from 40 to 2 bytes, we could compute the bandwidth usage for a G.729 call over MLPoFR with UDP header compression as (20+2+9)*50*8=12400bps. The same computation for compressed G.729 over Frame Relay with or without FRF.12 yields (20+2+7)*50*8=11600bps.

https://t.me/learningnets

For VoIP over Ethernet the Layer 2 overhead for is typically 18 bytes—14 bytes for the Ethernet header and 4 bytes for FCS (32 bits). If the frame carries a VLAN tag, add another 4 bytes, for 22 bytes of total overhead. Note that you typically see the G.711 codec used over LAN links.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC WRED

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure an MQC policy on R4's to meet the following requirements for its Ethernet link to R5:
  - Shape the Ethernet link between R4 and R5 to 768kbps.
  - Mark HTTP traffic from servers on VLAN 146 with IP Precedence 2.
  - Reserve 33% of the remaining bandwidth to HTTP traffic from servers on VLAN 146.
  - Configure the HTTP traffic class to use random detection for dropping, instead of tail drop.
  - Start dropping packets randomly when the average queue size is between 4 and 16.
  - Drop 1 of every 4 packets when the average queue size reaches the maximum threshold.

Configuration

```
R4:

ip access-list extended HTTP
 permit tcp 155.1.146.0 0.0.0.255 eq www any
!
class-map HTTP
```

https://t.me/learningnets

```
 match access-group name HTTP
!
policy-map ETHERNET_LINK_TO_R5
 class HTTP
  bandwidth remaining percent 33
  random-detect
  random-detect precedence 2 4 16 4
!
policy-map PARENT_SHAPE
 class class-default
  shape average 768000
   service-policy ETHERNET_LINK_TO_R5
!
policy-map MARK_HTTP
 class HTTP
  set precedence 2
!
interface GigabitEthernet1.45
 service-policy output PARENT_SHAPE
!
interface GigabitEthernet1.146
 service-policy input MARK_HTTP
```

Verification

CBWFQ supports three drop policies: classic tail-drop, which is the default for user-defined classes, Congestive Discard for WFQ, and Random Early Detection (RED). When you apply the **`random-detect`** command under a user-defined class, it automatically removes the MQC LLQ and Remaining Bandwidth Reservations (pending update) command and enforces RED as the drop policy. When using RED with CBWFQ, each flow is considered an individual FIFO queue. This is similar to flow-based WRED; the big improvement is the ability to use random drop per flow, not per whole queue.

Similar to legacy per-interface WRED, you can tune settings per IP precedence value or per DSCP. In our case, HTTP traffic uses IP precedence of 2, so we tune the RED settings for IP precedence 2.

To verify this configuration, generate two traffic flows similar to before. The goal is to see random drops occur under the HTTP traffic class. To do this, first shut down the DMVPN interface of R5 to force the traffic from VLAN146 to the hosts behind R5 to take path across the Ethernet link. Configure R1 as an HTTP server and set R8 to transfer the IOS image from R1, oversubscribing the shaped Ethernet sub-interface.

https://t.me/learningnets

Then, generate a flow of ICMP packets from R6 to R5, simulating traffic that will fall into class-default. As always, make sure to set the load-interface of R4's physical Ethernet link to 30.

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
R4:
interface GigabitEthernet1
 load-interval 30
R5:
interface Tunnel0
 shutdown
R8#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
Accessing http://*****:*****@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg...
Loading http://***********@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg !!!!!!!!!

R6#ping 155.1.45.5 repeat 1000000 size 150



Type escape sequence to abort.
Sending 1000000, 150-byte ICMP Echos to 155.1.45.5, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Check the statistics for the service policy. Note the long detailed output for the HTTP class. It demonstrates the number of random drops occurred in the queue for IP traffic marked with IP precedence of 2.

```
R4#show policy-map interface GigabitEthernet1.45
 GigabitEthernet1.45


  Service-policy output: PARENT_SHAPE


    Class-map: class-default (match-any)  54770 packets, 8856913 bytes
      5 minute offered rate 155000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/12/0
      (pkts output/bytes output) 54757/8849617
      shape (average) cir 768000, bc 3072, be 3072
```

https://t.me/learningnets

```
target shape rate 768000



      Service-policy : ETHERNET_LINK_TO_R5

        Class-map: HTTP (match-all)  7371 packets, 4367035 bytes
          5 minute offered rate 95000 bps, drop rate 0000 bps
          Match: access-group name HTTP
          Queueing
          queue limit 64 packets
          (queue depth/total drops/no-buffer drops) 7/12/0
          (pkts output/bytes output) 7359/4359907 bandwidth remaining 33%
Exp-weight-constant: 4 (1/16)
Mean queue depth: 6 packets
            class       Transmitted         Random drop      Tail drop          Minimum        Maximum     Mark
                     pkts/bytes            pkts/bytes       pkts/bytes          thresh         thresh     prob

            0               0/0               0/0              0/0                 16             32  1/10
            1               0/0               0/0              0/0                 18             32  1/10
2             7359/4359907        12/7128           0/0                  4             16  1/4
            3               0/0               0/0              0/0                 22             32  1/10
            4               0/0               0/0              0/0                 24             32  1/10
            5               0/0               0/0              0/0                 26             32  1/10
            6               0/0               0/0              0/0                 28             32  1/10
            7               0/0               0/0              0/0                 30             32  1/10

        Class-map: class-default (match-any)
          47398 packets, 4489710 bytes 5 minute offered rate 62000 bps, drop rate 0000 bps

          Match: any

          queue limit 64 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 47398/4489710
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Dynamic Flows and WRED

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure an MQC policy on R4's to meet the following requirements for its Ethernet link to R5:
  - Shape the Ethernet link between R4 and R5 to 768kbps.
  - Activate random drops for the unclassified traffic's dynamic flows.
  - Change the minimum and maximum RED thresholds for traffic with an IP precedence of one to 1 and 40, respectively.
  - As the queue depth grows close to the maximum threshold, the probability of packet discard should be 25%.

Configuration

R4:

```
policy-map ETHERNET_LINK_TO_R5
 class class-default
  fair-queue
  random-detect
  random-detect precedence 1 1 40 4
!
policy-map PARENT_SHAPE
 class class-default
  shape average 768000
```

https://t.me/learningnets

```
   service-policy ETHERNET_LINK_TO_R5
!
interface GigabitEthernet1.45
 service-policy output PARENT_SHAPE
```

Verification

There are two ways to enable WRED within class-default. The first is to configure a **`bandwidth`** reservation statement (turning the class's queue into a FIFO queue) and then enabling RED, and the second is to enable RED with WFQ. The second case activates RED dropping to replace Congestive Discard Threshold-based drops for dynamic flows. Think of this as a closer equivalent to flow-based RED, with automatic flow classification. All other WRED parameters remain the same as in the legacy case (averaging exponential factor, thresholds, etc.).

To verify this configuration, generate an ICMP traffic flow from R5 to R6, marking it for IP Precedence 1.

```
R5#ping
Protocol [ip]:
Target IP address: 155.1.146.6
Repeat count [5]: 10000
Datagram size [100]: 1500
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: Type of service [0]: 32

Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 10000, 1500-byte ICMP Echos to 155.1.146.6, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Verify the WRED statistics for class-default in your policy map. Note that no packets have been dropped because of the high maximum threshold (drop probability is inversely proportional to (max. thresh – min. thresh)). Also note matches for IP precedence 6 traffic, which corresponds to EIGRP packets sent between R4 and R5.

```
R4#show policy-map interface GigabitEthernet1.45
```

https://t.me/learningnets

```
GigabitEthernet1.45

 Service-policy output: PARENT_SHAPE

   Class-map: class-default (match-any)
     4844 packets, 7299940 bytes
     30 second offered rate 732000 bps, drop rate 0000 bps
     Match: any
     Queueing
     queue limit 64 packets
     (queue depth/total drops/no-buffer drops) 0/0/0
     (pkts output/bytes output) 4844/7299940
     shape (average) cir 768000, bc 3072, be 3072 target shape rate 768000

     Service-policy : ETHERNET_LINK_TO_R5

       Class-map: class-default (match-any)
         4844 packets, 7299940 bytes 30 second offered rate 732000 bps, drop rate 0000 bps
         Match: any
         Queueing
         queue limit 64 packets
         (queue depth/total drops/no-buffer drops/flowdrops) 1/0/0/0
         (pkts output/bytes output) 4844/7299940
         Fair-queue: per-flow queue limit 16 packets
           Exp-weight-constant: 4 (1/16)
           Mean queue depth: 0 packets
           class       Transmitted      Random drop      Tail/Flow drop Minimum Maximum Mark
                       pkts/bytes       pkts/bytes      pkts/bytes   thresh  thresh  prob

           0               3/222            0/0              0/0                 16            32  1/10
1            4807/7297026         0/0              0/0                  1            40  1/4

           2               0/0              0/0              0/0                 20            32  1/10
           3               0/0              0/0              0/0                 22            32  1/10
           4               0/0              0/0              0/0                 24            32  1/10
           5               0/0              0/0              0/0                 26            32  1/10
           6              33/2570           0/0              0/0                 28            32  1/10
           7               1/122            0/0              0/0                 30            32  1/10
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC WRED with ECN

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure an MQC policy on R4's to meet the following requirements for its Ethernet link to R5:
  - Shape the Ethernet link between R4 and R5 to 768kbps.
  - Mark HTTP traffic from servers on VLAN 146 with IP Precedence 2.
  - Reserve 33% of the remaining bandwidth to HTTP traffic from servers on VLAN 146.
  - Configure a policy on the HTTP traffic on R4 so that explicit congestion notification for TCP is used for RED dropping.

Configuration

R4:

```
ip access-list extended HTTP
 permit tcp 155.1.146.0 0.0.0.255 eq www any
!
class-map HTTP
 match access-group name HTTP
!
policy-map ETHERNET_LINK_TO_R5
 class HTTP
  bandwidth remaining percent 33
```

https://t.me/learningnets

```
  random-detect
  random-detect ecn
!
policy-map PARENT_SHAPE
 class class-default
  shape average 768000
   service-policy ETHERNET_LINK_TO_R5
!
policy-map MARK_HTTP
 class HTTP
  set precedence 2
!
interface GigabitEthernet1.45
 service-policy output PARENT_SHAPE
!
interface GigabitEthernet1.146
 service-policy input MARK_HTTP
```

Verification

TCP Explicit Congestion Notification (ECN), similar to BECN and FECN in Frame Relay, is used to signal the forthcoming of network congestion for TCP flows. Originally, TCP detected network congestion based on packet loss, timeouts, and duplicate acknowledgments. This was usually the result of full queues and unconditional packet drops. TCP ECN allows the network to signal the receiver of the flow that the network is close to dropping packets. It's then up to the TCP receiver to decide how to react to this notification; it usually signals the sender to slow the sending rate. The overall effect of TCP ECN is better performance, compared to simple packet drops and slow start, because it allows the sender to respond faster than slow start would and results in less time spent on the recovery from a packet loss.

TCP ECN works together with RED by changing the exceed action from random drop to ECN marking. Instead of randomly dropping a packet when the average queue depth grows above the minimum threshold, RED marks packet with the special ECN flag. This marking uses the two least-significant bits of the TOS bytes in the IP header. The Diff-Serv QoS model uses upper six bits of the TOS byte for DSCP/IP Precedence values. The lower two bits are known as ECN Capable Transport (ECT) and Congestion Experienced (CE). If an endpoint (sender or receiver) is ECN capable, it will set the ECT bit in packet headers to signal that it is capable of responding to congestion notifications. If RED is ECN enabled, and it's going to drop a packet randomly, the following actions apply.

https://t.me/learningnets

First, ECN RED checks to see that either the ECT or CE bits are set. If both of the bits are zero, the sending or receiving endpoint is not ECN capable, and the packet is randomly dropped. If either of the bits are set (ECT or CE), the RED procedure does not drop the packet, but instead sets the other bit and transmits the packet with both bits set. If both ECT and CE bits are set, the packet is simply transmitted. If the queue is full, RED drops the packets irrespective of their markings.

To verify this configuration, enable the TCP ECN feature on both R1 and R5. Enable the HTTP server service on R1, and configure R5 to transfer a file from R1. Shut down the DMVPN Tunnel interface of R5 to ensure that packets take path across the Ethernet link between R4 and R5. As always, make sure to set the load-interface of R4's physical Ethernet link to 30.

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
!
ip tcp ecn
R4:
interface GigabitEthernet1
 load-interval 30
R5:
ip tcp ecn
!
interface Tunnel 0
 shutdown

R5#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
```

Check the statistics for the service policy. Note the long detailed output for the HTTP class. It demonstrates the number of ECN marks occurred in the queue for IP traffic marked with IP precedence of 2:

```
R4#show policy-map interface GigabitEthernet1.45
 GigabitEthernet1.45

  Service-policy output: PARENT_SHAPE

    Class-map: class-default (match-any)
      52119 packets, 30858903 bytes
      30 second offered rate 764000 bps, drop rate 0000 bps
      Match: any
```

https://t.me/learningnets

```
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 7/0/0
    (pkts output/bytes output) 51874/30714401
    shape (average) cir 768000, bc 3072, be 3072
    target shape rate 768000

    Service-policy : ETHERNET_LINK_TO_R5

      Class-map: HTTP (match-all)
        51131 packets, 30355479 bytes
        30 second offered rate 763000 bps, drop rate 0000 bps
        Match: access-group name HTTP
        Queueing
        queue limit 64 packets
        (queue depth/total drops/no-buffer drops) 7/0/0
        (pkts output/bytes output) 28155/16713938
        bandwidth remaining 33%
          Exp-weight-constant: 4 (1/16)
          Mean queue depth: 6 packets
          class       Transmitted    ECN       Random drop      Tail drop          Minimum        Maximum     Mark
                  pkts/bytes     marked        pkts/bytes       pkts/bytes          thresh         thresh     prob

          0               0/0             0      0/0              0/0                 16            32  1/10
          1               0/0             0      0/0              0/0                 18            32  1/10
2          21214/12591176      2      0/0              0/0                 20            32  1/10

          3               0/0             0      0/0              0/0                 22            32  1/10
          4               0/0             0      0/0              0/0                 24            32  1/10
          5               0/0             0      0/0              0/0                 26            32  1/10
          6               0/0             0      0/0              0/0                 28            32  1/10
          7               0/0             0      0/0              0/0                 30            32  1/10

      Class-map: class-default (match-any)
        157 packets, 12386 bytes
        30 second offered rate 0000 bps, drop rate 0000 bps
        Match: any

    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 59/4646
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Class-Based Generic Traffic Shaping

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure MQC shaping on R6 to limit the sending rate on its link to VLAN 146 to 384Kbps.
- The link to VLAN 67 should be limited to 512Kbps.
- Use a burst interval (Tc) of 20ms.

Configuration

https://t.me/learningnets

```
R6:

policy-map SHAPE_VLAN146
 class class-default
   shape average 384000 7680
!
policy-map SHAPE_VLAN67
 class class-default
   shape average 512000 10240
!
interface GigabitEthernet1.146
  service-policy output SHAPE_VLAN146
!
interface GigabitEthernet1.67
  service-policy output SHAPE_VLAN67
```

Verification

The purpose of traffic shaping is to “format” an outbound packet flow so that it conforms to a traffic contract. The formatting process slows down the average bitrate and the packet flow structure, resulting in a traffic flow consisting of uniformly spaced traffic bursts. Service provider traffic contracts are usually verified using ingress policers, such as Frame Relay Traffic Policing. The burst values used in outbound customer edge shaping typically should match the inbound metering function used on the service provider edge.

An example when traffic shaping is needed is the case that a customer’s interface’s physical rate, the Access Rate (AR), is higher than the contracted rate guaranteed by the service provider. For example a customer buys an Ethernet circuit provisioned at 10Mbps, but the physical link to the provider is FastEthernet (100Mbps). Since the customer’s interface always serializes packets outbound at 100Mbps, and the service provider performs traffic policing/admission control inbound, shaping is needed on the customer side to slow the average output rate down to 10Mbps.

Another example case is when a remote site’s connection speed is lower than the local site’s. This means that the local site may overwhelm the remote site’s connection by sending packets faster than the remote site can accept. An example of this is in a WAN hub-and-spoke design, where the total sending rates of the spokes combined exceed the rate at which the hub can receive. By shaping the spoke sites down, the hub site is not oversubscribed.

https://t.me/learningnets

In both situations mentioned above, the physical AR (Access Rate) is greater than the desired output rate. To slow the rate down, the first task of the shaper is to meter the traffic coming into the output queue, and decide whether it exceeds the target average rate. The concept of metering is based on the fact that traffic leaves an interface in a serial manner (bit by bit, packet by packet), and that packets are usually grouped in bursts, separated by periods of interface silence.

While the router sends each burst at AR speed, the spacing between bursts makes the average rate less than the AR. The goal of metering is to mark those bursts that exceed (do not conform to) the desired average rate, called the Committed Information Rate (CIR).

Note that for the purpose of metering, the shaper's "CIR" value does not relate to the service provider's "CIR" value, where the latter actually means the "guaranteed" rate of the circuit based on the SLA. Instead, "CIR" in the context of shaping simply means the target average rate.

The metering function of traffic shaping uses what is known as a token bucket model to determine if traffic conforms to, or exceeds, the average rate. Every time a packet tries to be de-queued to the transmit ring, the metering function compares the size of the packet trying to leave to the amount of tokens, or credit, in the token bucket. If the size of the packet is less than or equal to the amount of credit, the packet conforms and is sent. If the size of packet is greater than the amount of credit in the token bucket, the packet exceeds, and is delayed.

The size of the token bucket is calculated by taking the desired average rate (CIR) in bits per second, and breaking it down into a smaller value of bursts in bits per interval in milliseconds. These values are expressed as Bc (Burst Committed) bits, and Tc (Time Committed) milliseconds. The size of the token bucket is Bc bits (or tokens), and the system refills the token bucket at a rate of Bc bits per Tc milliseconds. Essentially every Tc period, the token bucket is refilled with the Bc amount in bits. Think of the Bc bits as tokens going into the bucket every Tc interval. As mentioned previously, Tc is measured in milliseconds; multiple Tc intervals occur every second. The key point here is that Bc bits per Tc interval is the same value as CIR bits per second, but is simply expressed in smaller units.

https://t.me/learningnets

When a packet conforms to the average rate per interval, it is de-queued to the transmit ring, and the number of tokens deducted from the bucket is equal to the soze of the packet in bits. If packet exceeds because there are not enough tokens in the bucket, the shaping process delays the packet and holds it in the internal shaping queue. By this logic, even though traffic is always sent at the AR, the periods of delay incurred by non-conforming traffic in the shaping queue results in the overall average rate (CIR) being lower than the AR. This method is known the leaky bucket algorithm.

Since the shaper sends packets in bursts every Tc milliseconds, the outgoing traffic “shape” becomes uniform, meaning that packet bursts of almost the same size are separated by Tc intervals. The size of Tc is not manually configurable, however it is configured indirectly by configuring the CIR and the Bc values based on the formula Bc = CIR * Tc/1000. Note that most documentation for traffic shaping denotes this as Bc = CIR*Tc, which is true as long as Tc is expressed in seconds, not milliseconds, which it is not by default. This is why it has to be divided by 1000.

The minimal value for Tc is platform and version dependent, and is generally 10ms. Newer platforms, such as the CSR1000v used in these examples, can set the Tc as low as 4ms. This limits the scheduler “precision”, but also puts a limit on the CPU utilization. Most importantly though, the value of Tc sets the minimum delay between packet bursts. This generally limits the use of traffic shaping to WAN connections, since using it across a LAN connection may introduce unacceptably high delay.

One possible problem with the above calculation for Bc is the case that the packet trying to be de-queued is larger than Bc, which means that there would never be enough credits in the token bucket to send it. For example if a packet’s size is 1500 bytes, but the Bc is only 1000 bytes. To deal with this situation the shaper calculates a deficit counter (e.g. 1000-1500=-500) and adds this counter to the accumulated credit in the next round (next Tc interval). In effect this reduces the amount of traffic to send the next time around. With this method, although the scheduler sends a packet every time the accumulated credit is above zero, the accumulated credit will eventually drop to zero and the scheduler will have to wait 2 Tc intervals to accumulate credit and send the next packet. This may result in an average sending rate per Tc to be greater than or less than CIR, but the average rate over a longer period of time still never exceeds CIR. To avoid this problem altogether ensure that Bc is greater than the average packet size, which will achieve a smoother packet distribution. This is not always possible though, since there are cases when CIR value is too low. In the latter case, layer 2 fragmentation can be introduced.

The next problem case that the scheduler can run into is when it has no traffic to send during a time interval (e.g. a pause in the packet stream), but it has more than Bc bits to send in the following time interval. Based on the leaky token bucket

https://t.me/learningnets

algorithm, no more than Bc bytes can be sent per Tc interval, even if in previous intervals it did not send enough traffic. The result of this is that the shaper achieves less than the desired average rate. To resolve this problem, traffic shaping uses what is known as a dual leaky token bucket, with the first token bucket represented as Committed Burst (Bc) and the second token bucket as Excess Burst (Be).

The Excess Burst bucket is only filled in the case that the full Bc bucket was not emptied in the previous interval. The extra credits, or tokens, left over from the Bc bucket are then moved to the Be bucket before the Bc bucket is refilled. For example, if the Bc size is 10 bits, but only 8 bits were sent in the current interval, a credit of 2 bits can be moved to the Be bucket if space is available. During the next interval, the scheduler can now de-queue up to Bc+Be bits. If Bc capacity is again not used completely, the left over credits are moved to Be, up to its maximum size.

Like Bc, Be has a finite size defined which controls how much credit can be stored. The size of the Be bucket is constrained by the Access Rate of the physical link, since the packets are always serialized at this rate. Therefore the maximum Be value (maxBe) is equal to (AR-CIR)*Tc/1000 which implies that if the shaper sends Bc+maxBe per Tc, it is sending at the Access Rate. The Be value can be set lower than maxBe, but should never exceed maxBe. Note that since Be is only populated due to a lack of Bc being used, the average sending rate over time still never exceeds the CIR.

MQC-based traffic-shaping allows Generic Traffic Shaping to be applied to any traffic class. This makes traffic-shaping's configuration modular and unified, without special variants for encapsulations like Frame Relay. Class-based GTS is similar to legacy GTS, but allows shaping based on class-maps, as opposed to access-list classification with legacy GTS, thus allowing any MQC classification options, such as NBAR, to define shaped traffic. Additionally, MQC based shaping allows the shaping queue to be tuned, whereas legacy GTS supports just simple WFQ with no option to tune it.

With the command `shape average` under the policy-map class configuration, the CIR, Bc, and Be are defined. Because this task did not mention anything about Be, the default value is used. Bc can be calculated the same way as with legacy GTS, where Bc=CIR*Tc/1000. In our case, Tc is 20ms, so the Bc values are 7680 and 10240 bits respectively for each interface.

https://t.me/learningnets

In this scenario, we used class-based shaping to limit the sub-interfaces sending rate. This is a common use of GTS, and the effect is that each sub-interface now uses its own software queue, whereas by default, all sub-interfaces share the software queue of their main interface. This also allows the use of separate QoS policies per sub-interface, because of the ability to tune shaper’s queue (see the next task for more information on this).

To verify this configuration, generate three traffic flows across R6 to see the class-based GTS in action. Below this is accomplished with R6 sending IP SLA jitter probes to R4, R6 acting as an HTTP server for R4, and an ICMP flow being sent from R7 to R4. Set the load interval on R6's GigabitEthernet1 to 30 seconds.

https://t.me/learningnets

```
R4:

ip sla responder
R6:

username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
!
! Configure SLA probe for jitter with 60 byte packets

ip sla 1
 udp-jitter 155.1.146.4 16384 codec g729a control enable
 threshold 1000
 timeout 1000
 frequency 1
!
ip sla schedule 1 life forever start-time now
!
! Set load interval to 30 seconds
!
interface GigabitEthernet1
 load-interval 30

R4#copy http://admin:cisco@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:

Accessing http://*****:*****@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg...
Loading http://***********@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg !!!!!!!!!!!!
R7#ping 155.1.146.4 repeat 100000



Type escape sequence to abort.
Sending 100000, 100-byte ICMP Echos to 155.1.146.4, timeout is 2 seconds:
!!!!!!!!!!!
```

```
  Service-policy output: SHAPE_VLAN146

    Class-map: class-default (match-any)
      34935 packets, 6409549 bytes 30 second offered rate 375000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 9/0/0
      (pkts output/bytes output) 34935/6409549 shape (average) cir 384000, bc 7680, be 7680
target shape rate 384000
!
!R6#show policy-map interface gigabitEthernet1.67
 GigabitEthernet1.67

  Service-policy output: SHAPE_VLAN67

    Class-map: class-default (match-any)
      103199 packets, 12115056 bytes 30 second offered rate 13000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 103199/12115056 shape (average) cir 512000, bc 10240, be 10240
target shape rate 512000
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Class-Based GTS and CBWFQ

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure R6's connection to VLAN 146 as follows:
  - Configure MQC shaping on R6 to limit the sending rate on its link to VLAN 146 to 384Kbps.
  - Provide 32Kbps of priority treatment for small packets with a layer 3 size of 60 bytes (voice packets) using a 4000-byte burst value.
  - Guarantee the HTTP traffic 256Kbps of the shaper’s bandwidth.
  - Unclassified traffic should receive fair-queue treatment.

Configuration

R6:

```
class-map VOICE
 match packet length min 60 max 60
!
ip access-list extended HTTP
 permit tcp any eq 80 any
!
class-map HTTP
 match access-group name HTTP
!
policy-map CBWFQ
```

https://t.me/learningnets

```
 class VOICE
   priority 32 4000
 class HTTP
   bandwidth 256
 class class-default
   fair-queue
!
policy-map SHAPE_VLAN146
  class class-default
    shape average 384000 7680
    service-policy CBWFQ
!
interface GigabitEthernet1.146
  service-policy output SHAPE_VLAN146
```

Verification

The ability to configure the shaper's queue is one of the most powerful features of class-based shaping. Previously, GTS was able to use only WFQ, and the only parameter you could change was the WFQ buffer limit.

Using MQC syntax, you can "nest" another service policy inside a class configured for shaping. This nested service policy defines "sub-classes" and their respective parameters, including CBWFQ settings. The most prominent feature is the ability to configure an LLQ conversation, allowing optimal voice traffic handling when limiting traffic rate on sub-interfaces.

When you nest a CBWFQ configuration inside a shaper, the amount of bandwidth you can allocate to nested classes equals to the shaper average rate (384Kbps in this scenario). In a production environment, you may want to define a separate class for control plane traffic and guarantee it some bandwidth.

As mentioned in the previous task, the common use of shaping is limiting sub-interfaces' sending rates. By nesting a CBWFQ policy, you can define flexible, per-sub-interface queuing policies, comparable to the legacy features available for Virtual Circuit-based technologies such as ATM and Frame-Relay.

Note the priority queue bandwidth of 32Kbps. We get this bandwidth from the assumption of VoIP packet sizes of 60 bytes (G.729), plus 18 bytes overhead for an Ethernet header with 4 bytes of VLAN tag at 50 packets per second = (60+18)*50*8=31200bps. Remember that priority bandwidth must take layer 2 overhead in account.

To verify the configuration, configure three traffic sources. R6 should generate IP

https://t.me/learningnets

SLA jitter probes and R4 should respond. Set this probe to time out in 5 seconds, and use the ToS byte 224 (IP Precedence 7) for signaling. In addition to that, R6 should act as a web-server and R4 should transfer a large file from it. Finally, source an ICMP packet stream from R7 to R4.

```
R4:
ip sla responder
R6:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
!
! Configure SLA probe for jitter with 60 byte packets
! Note the ToS byte corresponding to IP Precedence of 7
!
ip sla 1
 udp-jitter 155.1.146.4 16384 codec g729a control enable
 threshold 1000
 timeout 1000
 frequency 1
 tos 224
!
ip sla schedule 1 life forever start-time now
!
! Set load interval to 30 seconds
!
interface GigabitEthernet1
 load-interval 30
R4#copy http://admin:cisco@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
Accessing http://*****:*****@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg...
Loading http://***********@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg !!!!!!!!! !!
R7#ping 155.1.146.4 repeat 100000



Type escape sequence to abort.
Sending 100000, 100-byte ICMP Echos to 155.1.146.4, timeout is 2 seconds:
!!!!!!!
```

```
Service-policy output: SHAPE_VLAN146

  Class-map: class-default (match-any)
    26437 packets, 4650651 bytes
    30 second offered rate 371000 bps, drop rate 0000 bps
    Match: any
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 26437/4650651
    shape (average) cir 384000, bc 7680, be 7680 target shape rate 384000

    Service-policy : CBWFQ

      queue stats for all priority classes:
        Queueing
        queue limit 512 packets
        (queue depth/total drops/no-buffer drops) 0/0/0
        (pkts output/bytes output) 3238/252564

      Class-map: VOICE (match-all)
        3238 packets, 252564 bytes 30 second offered rate 28000 bps, drop rate 0000 bps
        Match: packet length min 60 max 60 Priority: 32 kbps, burst bytes 4000, b/w exceed drops: 0



      Class-map: HTTP (match-all)
        1258 packets, 1809129 bytes 30 second offered rate 260000 bps, drop rate 0000 bps
        Match: access-group name HTTP
        Queueing
        queue limit 64 packets
        (queue depth/total drops/no-buffer drops) 3/0/0
        (pkts output/bytes output) 1258/1809129 bandwidth 256 kbps

      Class-map: class-default (match-any)
        21941 packets, 2588958 bytes 30 second offered rate 82000 bps, drop rate 0000 bps
        Match: any
        Queueing
        queue limit 64 packets
        (queue depth/total drops/no-buffer drops/flowdrops) 1/0/0/0
        (pkts output/bytes output) 21941/2588958 Fair-queue: per-flow queue limit 16 packets
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Single-Rate Three-Color Policer

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure R4 to meter incoming HTTP traffic on its link to VLAN 146 as follows:
  - If the traffic rate is less than 128Kbps, mark the packets with an IP precedence of one.
  - If the traffic exceeds 128Kbps, mark the packets with an IP precedence of zero.
  - Drop violating traffic.
- Ensure that the burst size is large enough to accommodate normal and excess burst durations of 200ms and 300ms at a rate of 128Kbps.

Configuration

R4:

```
ip access-list extended HTTP
 permit tcp any eq 80 any
!
class-map HTTP
 match access-group name HTTP
!
policy-map POLICE_VLAN146
 class HTTP
   police 128000 3200 4800
```

https://t.me/learningnets

```
    conform-action set-prec-transmit 1
    exceed-action set-prec-transmit 0
    violate-action drop
!
interface GigabitEthernet1.146
  service-policy input POLICE_VLAN146
```

Verification

To get a better understanding of how the Cisco IOS Policer works, we will look at how the legacy CAR (Committed Access Rate) first.

The Legacy Committed Access Rate (CAR) feature, or rate-limiting, was designed for two purposes - admission control (e.g. packet remarking), and traffic limiting (e.g. policing) at the network edge. Unlike traffic shaping, rate limiting can be configured as both an input and output feature, and it does not buffer (delay) exceeding traffic bursts.

Admission control through CAR is similar to how traffic-shaping's algorithm works, but with some key differences. Consider the situation in which a customer's edge device connects to the service provider via an interface with a physical Access Rate (AR) that is higher than the contracted rate (the provider's CIR). In order to enforce the contract, the provider needs to meter the input traffic's bitrate, and mark (dropping is also a form of marking) the packets based on the measured rate. Like in shaping the idea of metering is based on the fact that network traffic enters the interface in a serial manner (bit by bit, packet by packet), and that packets are usually grouped in bursts, separated by "islands" of networking interface silence. While the router receives each burst at AR, the spacing between bursts makes average input rate less than AR. The goal of metering is marking those bursts that conform or exceed the contracted average rate (CIR). For example, if two shortly separated bursts enter the router at AR speed, the average rate measured over those two bursts would be close to AR and much higher than CIR, and thus the second burst would most likely be exceeding.

To measure the average speed, metering uses a sliding "averaging time interval" (Tc), not to be confused with traffic shaping's Tc. The process considers a new incoming packet as conforming if the amount of traffic already received during the current Tc, plus the size of the new packet, is less than or equal to Committed Burst (Bc). The interval Tc is sliding in the sense that it moves across the packet line as packets enter the router. The larger the Tc value, the greater amount of averaging that is performed over the input packet rate.

Note that like in traffic shaping, Tc, Bc and CIR values are not independent, since

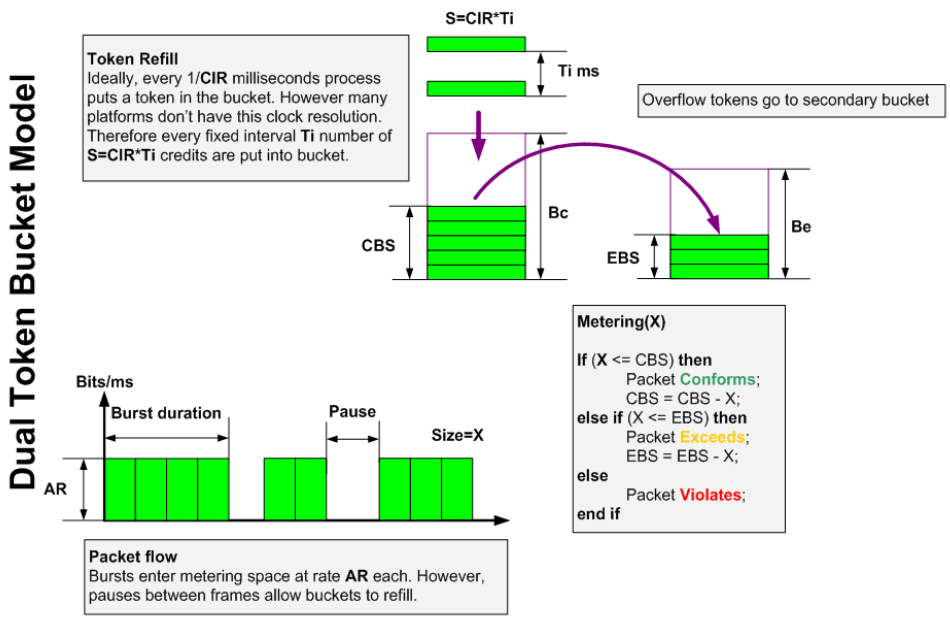
https://t.me/learningnets

Bc=CIR*Tc. CIR is enforced by the fact that during any interval Tc, the amount of conforming traffic is no more than the Bc, and thus the average CIR the provider enforces is Bc per Tc. The difference between shaping and policing however, is that shaping produces packet bursts uniformly separated by interval Tc, each burst of size Bc, while CAR uses a sliding Tc window to measure the current average traffic rate and mark every new packet accordingly.

A Single Rate Three Color Marker (Policer) or “srTCM” is the RFC-based implementation of the metering process. Compared to legacy CAR, the following has changed.

First, B_e is used as extra credit for periods of inactivity. The policing process uses a Dual Token Bucket model, but the implementation is different. “Three color” in this case means that incoming traffic is metered against CIR using the configured burst sizes, and the result is a marking using either of three colors: Green (Conform), Yellow (Exceed), Red (Violate).

As usual, the token bucket exhibits the concept of a sliding metering window over the packet flow. However, this time process uses a secondary, Excess Burst bucket.



The Excess Burst bucket accumulates extra credit, spilled over from the Normal Bucket when it overfills. This only happens during long periods of pause that exceed the averaging interval T_c. With just a single bucket, the process does not account for “extended” periods of silence. The Excess Burst bucket allows accumulating credits and using them when incoming burst exceeds the configured B_c value. Thus, B_e in this usage serves a purpose similar to what the Be functionality does with traffic-shaping. In fact, these two are complementary in the sense that srTCM’s B_c properly “recognizes” the excessive bursting of traffic shaping.

https://t.me/learningnets

In a classic token bucket model, a special timer fires every T_i interval, triggering the token refresh procedure. This method, however, limits the “resolution” of metering. At the same time, the amount of CPU operations needed to meter a packet is small. This is because the procedure uses only addition and subtraction operations, which consume little CPU cycles.

Cisco’s implementation of the token bucket used for srTCM uses a different algorithm. This algorithm has “unlimited” precision, but consumes more CPU cycles per-packet. The process keeps track of three variables:

CBS – current normal burst size
EBS – current excess burst size
T0 – last packet arrival time

Now imagine that a new packet of size “S” arrives at time “T”.

Step 1: The process computes accumulated credit:

Credit = CIR*(T-T0)

then adds this values to CBS:

if ((CBS + Cr) <= Bc) **then**
CBS = CBS + Credit;
else
CBS = Bc;
end if

Step 2: The process computes a new EBS size if there are enough credits:

If(CBS + Credit > Bc) **then**
EBS = CBS + Credit – Bc;
end

if (EBS > Be) **then**
EBS = Be;

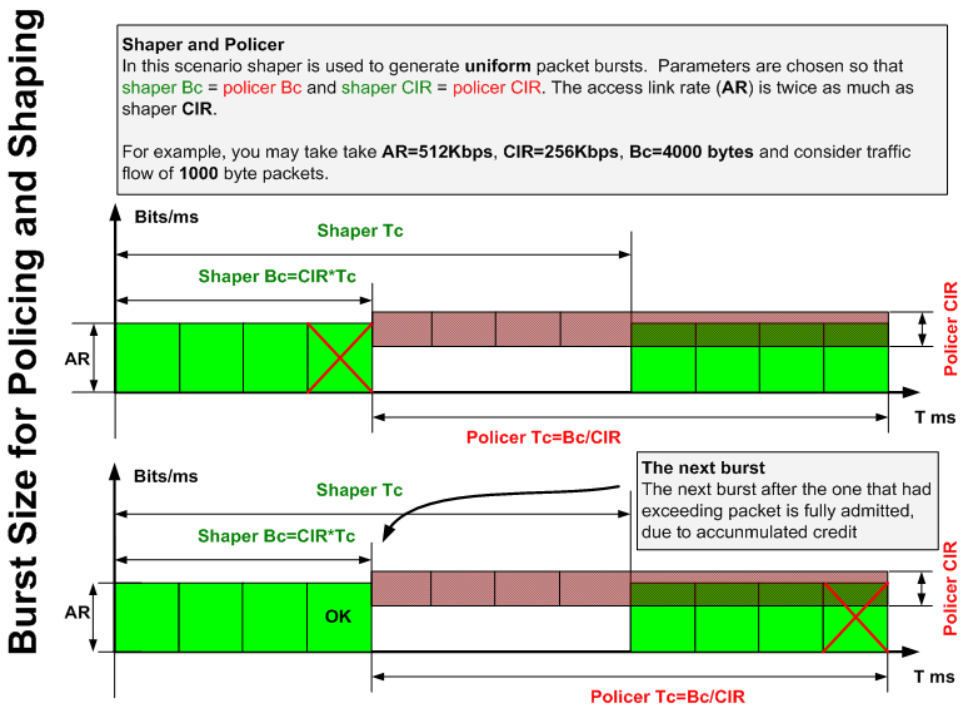
Step 3: The process compares packet size to accumulated credits and performs marking:

https://t.me/learningnets

```
if ( S < CBS ) then
    Packet Conforms;
    CBS = CBS – S;
else if (S < EBS) then
    Packet Exceeds;
    EBS = EBS – S;
else
    Packet Violates;
end if
```

For an optimal burst size selection, if you are using srTCM to perform traffic admission at the network edge, ensure that your policing burst sizes match at least one packet size more than the burst size configured at the customer side. This is needed because now the router meters traffic with extra precision, and the effect illustrated on the next figure (shaper Bc = policer Bc) may occur.

https://t.me/learningnets



Note that the policer may mark the beginning of the second burst as exceeding. This is because the policer B_C exactly equals shaper B_C. In this situation when the sliding window hits the beginning of the next burst, it has no capacity to accommodate another packet. The packet marked as exceeding does not count for the sliding window content, and thus the next burst will have all four packets admitted. For this ideal simulation, the policer B_C should be larger than traffic burst size by at least one packet. This brings us to the formula:

Policer B_C = Shaper Burst Size + Average Packet Size

In addition, it makes sense to configure burst size proportional to the average packet size as well, to ensure more precise matching. However, this is not necessary, especially with larger bursts.

In this particular example, the burst sizes are based on the time intervals as follows: Remember that the policer takes arguments Bc and Be is bytes, not bits such as the traffic-shaper.

Bc = 128000*0.2/8=3200 bytes
Be = 128000*0.3/8=4800 bytes

To test the policer, simulate an HTTP traffic flow from R1 to R4. At the same time, pre-shape the traffic rate from R1 to R4 to 128kbps. Set the shaper's burst size to 12800 bits (1600 bytes), which is half of the policer's burst:

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
```

https://t.me/learningnets

```
ip http path bootflash:
!
policy-map SHAPE_VLAN146
 class class-default
   shape average 128000 12800
!
interface GigabitEthernet1.146
  service-policy output SHAPE_VLAN146
R4:
interface GigabitEthernet1
  load-interval 30
R4#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
```

Check the policy-map statistics. Note the near perfect 128000 bits, with no exceeding packets:

```
R4#show policy-map interface gigabitEthernet 1.146
 GigabitEthernet1.146

  Service-policy input: POLICE_VLAN146

    Class-map: HTTP (match-all)
      1776 packets, 2565320 bytes 30 second offered rate 127000 bps, drop rate 0000 bps
      Match: access-group name HTTP
      police: cir 128000 bps, bc 3200 bytes, be 4800 bytes
conformed 1776 packets, 2565320 bytes; actions:
set-prec-transmit 1
        exceeded 0 packets, 0 bytes; actions:
          set-prec-transmit 0
        violated 0 packets, 0 bytes; actions:
          drop  conformed 127000 bps
, exceeded 0000 bps, violated 0000 bps

    Class-map: class-default (match-any)
      234 packets, 15908 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Hierarchical Policers

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure R4 to limit the aggregate rate of HTTP traffic entering the connection to VLAN 146 to 128Kbps.
- Transmit conforming packets, mark exceeding packets with an IP precedence of zero, and drop violating packets.
- For HTTP traffic flows from R1 and R6, limit the rate to 64Kbps for each flow.
- For these second-level policers, set the conform action to set the IP precedence to 1 and transmit, the exceed action to set the IP precedence to 0 and transmit, and the violate action to set the IP precedence to 0 and transmit.

Configuration

R4:

```
ip access-list extended HTTP
 permit tcp any eq 80 any
!
class-map HTTP
 match access-group name HTTP
!
ip access-list extended FROM_R1
 permit ip host 155.1.146.1 any
!
```

https://t.me/learningnets

```
ip access-list extended FROM_R6
 permit ip host 155.1.146.6 any
!
class-map FROM_R1
 match access-group name FROM_R1
!
class-map FROM_R6
 match access-group name FROM_R6
!
policy-map SUBRATE_POLICER
 class FROM_R1
  police 64000 3200 4800
   conform-action set-prec-transmit 1
   exceed-action set-prec-transmit 0
   violate-action set-prec-transmit 0
 class FROM_R6
  police 64000 3200 4800
   conform-action set-prec-transmit 1
   exceed-action set-prec-transmit 0
   violate-action set-prec-transmit 0
!
policy-map POLICE_VLAN146
 class HTTP
   police 128000 3200 4800
    conform-action transmit
    exceed-action set-prec-transmit 0
    violate-action drop
   service-policy SUBRATE_POLICER
!
interface GigabitEthernet1.146
  service-policy input POLICE_VLAN146
```

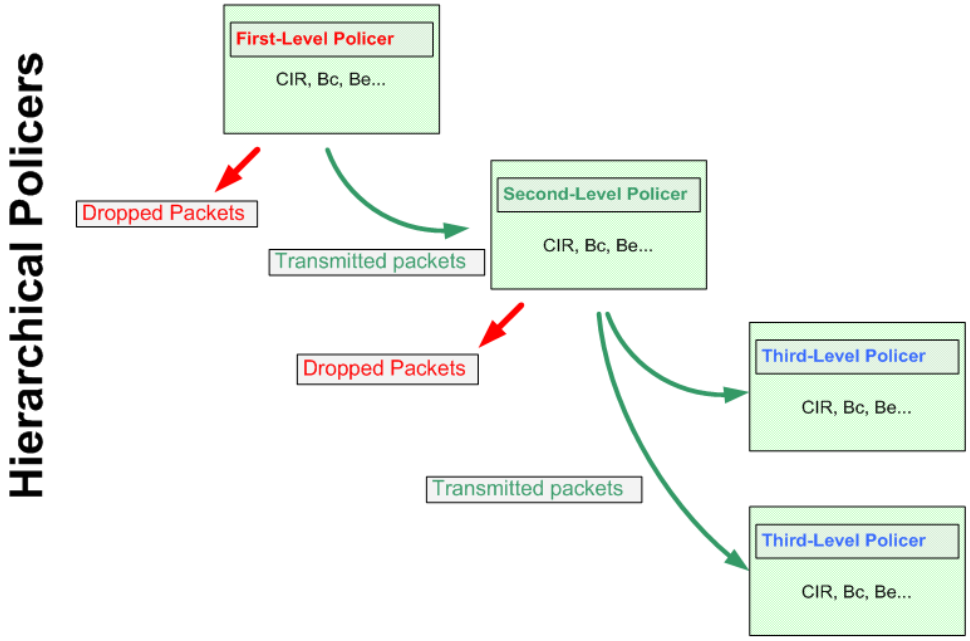
Verification

MQC allows the nesting of up to three levels of policing. The syntax uses nested service-policies, where the parent policy defines the aggregate rate, and the child policies define the sub-rates for each class. The effect is similar to the cascaded rate-limit statements used in legacy CAR; however, note the following differences.

First, with MQC hierarchical policing, the upper-level policers are applied before the nested policers. With CAR, you usually configure the “aggregate” statement in the end of the other “sub-rate” statements. Second, every policer could be a single-rate or two-rate policer. Furthermore, the MQC allows for a more natural and intuitive

https://t.me/learningnets

grouping of classes and policing statements.



Start verification by generating a single HTTP traffic flow from R1 down to R8 across R4. Shape this down to 128kbps on R1.

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
!
policy-map SHAPE_VLAN146
 class class-default
  shape average 128000 12800 0
!
interface GigabitEthernet1.146
  service-policy output SHAPE_VLAN146
R4:
interface GigabitEthernet1
 load-interval 30
R5:
interface Tunnel0
 shutdown
R8#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
```

https://t.me/learningnets

Note that traffic from R1 to R8 exceeds and violates the nested policer rate, and thus the second-level policer marks almost half of the traffic with IP Precedence 0:

```
R4#show policy-map interface GigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy input: POLICE_VLAN146

    Class-map: HTTP (match-all)
      6010 packets, 3503779 bytes 30 second offered rate 127000 bps, drop rate 0000 bps
      Match: access-group name HTTP
      police: cir 128000 bps, bc 3200 bytes, be 4800 bytes
        conformed 5487 packets, 3208581 bytes; actions:
          transmit
        exceeded 306 packets, 167898 bytes; actions:
          set-prec-transmit 0
        violated 217 packets, 127300 bytes; actions:
          drop  conformed 127000 bps
, exceeded 0000 bps, violated 0000 bps

      Service-policy : SUBRATE_POLICER
 Class-map: FROM_R1 (match-all)

          6010 packets, 3503779 bytes 30 second offered rate 127000 bps, drop rate 0000 bps
          Match: access-group name FROM_R1
          police: cir 64000 bps, bc 3200 bytes, be 4800 bytes
            conformed 2914 packets, 1680842 bytes; actions:
              set-prec-transmit 1
            exceeded 319 packets, 173956 bytes; actions:
              set-prec-transmit 0
            violated 2777 packets, 1648981 bytes; actions:
              set-prec-transmit 0 conformed 64000 bps, exceeded 0000 bps, violated 63000 bps



        Class-map: FROM_R6 (match-all)
          0 packets, 0 bytes
          30 second offered rate 0000 bps, drop rate 0000 bps
          Match: access-group name FROM_R6
          police:
              cir 64000 bps, bc 3200 bytes, be 4800 bytes
            conformed 0 packets, 0 bytes; actions:
              set-prec-transmit 1
            exceeded 0 packets, 0 bytes; actions:
              set-prec-transmit 0
            violated 0 packets, 0 bytes; actions:
              set-prec-transmit 0
```

https://t.me/learningnets

```
            conformed 0000 bps, exceeded 0000 bps, violated 0000 bps

        Class-map: class-default (match-any)
          0 packets, 0 bytes
          30 second offered rate 0000 bps, drop rate 0000 bps
          Match: any

    Class-map: class-default (match-any)
      304 packets, 24060 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

Now add another HTTP flow, from R6 down to R10 and across the Ethernet link between R4 and R5. Shape the R4-R5 Ethernet link down to 128kbps to aid in smoothing out both simultaneous flows, as well as the R6 connection to VLAN 146. Similar to how R1 is shaping:

```
R4:
policy-map SHAPE_R4_R5
 class class-default
  shape average 128000 12800 0
!
interface GigabitEthernet1.45
  service-policy output SHAPE_R4_R5

R6:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
!
policy-map SHAPE_VLAN146
 class class-default
  shape average 128000 12800 0
!
interface GigabitEthernet1.146
  service-policy output SHAPE_VLAN146
R10#copy http://admin:cisco@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
```

Verify the policer statistics again. Note that now both second-level policers meter input rates close to 64Kbps with no violating traffic:

```
R4#show policy-map interface GigabitEthernet1.146
```

https://t.me/learningnets

```
 GigabitEthernet1.146

  Service-policy input: POLICE_VLAN146

    Class-map: HTTP (match-all)
      29329 packets, 17259217 bytes 30 second offered rate 127000 bps, drop rate 0000 bps
      Match: access-group name HTTP
      police:
          cir 128000 bps, bc 3200 bytes, be 4800 bytes
        conformed 28435 packets, 16746201 bytes; actions:
          transmit
        exceeded 441 packets, 247108 bytes; actions:
          set-prec-transmit 0
        violated 453 packets, 265908 bytes; actions:
          drop  conformed 127000 bps
, exceeded 0000 bps, violated 0000 bps

      Service-policy : SUBRATE_POLICER

        Class-map: FROM_R1 (match-all)
          20270 packets, 11931119 bytes 30 second offered rate 64000 bps
, drop rate 0000 bps
          Match: access-group name FROM_R1
          police:
              cir 64000 bps, bc 3200 bytes, be 4800 bytes
            conformed 11912 packets, 6982583 bytes; actions:
              set-prec-transmit 1
            exceeded 409 packets, 227387 bytes; actions:
              set-prec-transmit 0
            violated 7949 packets, 4721149 bytes; actions:
              set-prec-transmit 0 conformed 64000 bps
, exceeded 0000 bps, violated 0000 bps

        Class-map: FROM_R6 (match-all)
          9059 packets, 5328098 bytes 30 second offered rate 63000 bps
, drop rate 0000 bps
          Match: access-group name FROM_R6
          police:
              cir 64000 bps, bc 3200 bytes, be 4800 bytes
            conformed 6387 packets, 3741070 bytes; actions:
              set-prec-transmit 1
            exceeded 108 packets, 64012 bytes; actions:
              set-prec-transmit 0
            violated 2564 packets, 1523016 bytes; actions:
              set-prec-transmit 0 conformed 63000 bps
, exceeded 0000 bps, violated 0000 bps
```

https://t.me/learningnets

```
      Class-map: class-default (match-any)
        0 packets, 0 bytes
        30 second offered rate 0000 bps, drop rate 0000 bps
        Match: any

    Class-map: class-default (match-any)
      689 packets, 54474 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: any
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Two-Rate Three-Color Policer

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Create a first-level policy-map applied to R4's connection to VLAN 146 matching on HTTP traffic.
  - Create two separate class-maps for HTTP traffic received from R1 and R6.
- Add a second-level policy-map to the first-level HTTP one that applies two-rate policers configured in the following way:
  - Use the CIR value of 64Kbps and PIR value of 128Kbps.
  - Use the values of CIR*400ms (64000 * 400/1000) and PIR*400ms (128000 * 400/1000) for normal and excess burst sizes.
  - Set the conform action to set IP precedence 1 and transmit, the exceed action to set IP precedence of 0 and transmit, and the violate action to drop.

Configuration

```
R4:

ip access-list extended HTTP
 permit tcp any eq 80 any
!
class-map HTTP
 match access-group name HTTP
!
ip access-list extended FROM_R1
```

https://t.me/learningnets

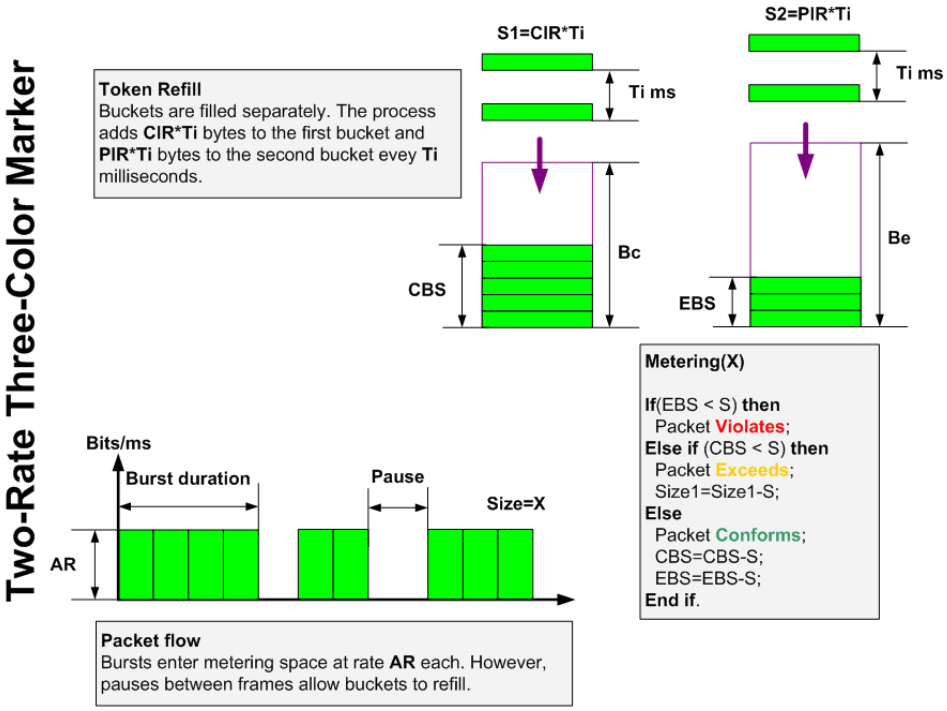
```
 permit ip host 155.1.146.1 any
!
ip access-list extended FROM_R6
 permit ip host 155.1.146.6 any
!
class-map FROM_R1
 match access-group name FROM_R1
!
class-map FROM_R6
 match access-group name FROM_R6
!
policy-map SUBRATE_POLICER
 class FROM_R1
  police cir 64000 bc 3200 pir 128000 be 6400
   conform-action set-prec-transmit 1
   exceed-action set-prec-transmit 0
   violate-action drop
 class FROM_R6
  police cir 64000 bc 3200 pir 128000 be 6400
   conform-action set-prec-transmit 1
   exceed-action set-prec-transmit 0
   violate-action drop
!
policy-map POLICE_VLAN146
 class HTTP
   service-policy SUBRATE_POLICER
!
interface GigabitEthernet1.146
  service-policy input POLICE_VLAN146
```

Verification

The two-rate policer implements an RFC-based procedure to meter traffic rate against two pre-configured rates, average and peak. The RFC names this procedure as a two-rate three-color marker, or “trTCM.” Three colors correspond to “Conform,” “Exceed,” and “Violate” actions of the policer.

Four configurable parameters, apply to the two-rate policer: CIR, PIR, B_c, B_e. This time, B_c bounds the average bursts, and B_e bounds the peak burst. Two-rate policer runs two token buckets at the same time. Each bucket has its own fill rate, as illustrated below.

https://t.me/learningnets



The process refills each bucket separately, at its own rate. The first bucket fills at rate CIR, and the second bucket fills at rate PIR. As usual, a special timer fires every T_i interval and triggers the token refresh procedure. This, however, limits the "resolution" of metering procedure.

The alternate implementation, used by Cisco with trTCM, is similar to the implementation of srTCM (Single Rate Three-Color Marker). The algorithm uses the following variables:

CBS – current normal burst size (initially set to B_c)
EBS – current excess burst size (initially set to B_e)
T0 – last packet arrival time

Now imagine that a new packet of size "S" arrives at time "T".

Step 1: The process computes the accumulated credit and new burst sizes:

Cr1 = CIR*(T-T0);
Cr2 = PIR*(T-T0);

then adds these values to CBS and EBS:

if ((CBS + Cr1) <= Bc) **then**
CBS = CBS + Cr1;
else
CBS = Bc;
end if

https://t.me/learningnets

if ((EBS + Cr2) <= Be) **then**
EBS = EBS + Cr2;
else
EBS = Be;
end if

Step 2: The process compares the packet size to accumulated credits:

if (S > EBS) **then**
    Packet **Violates**;
else if (S > CBS) **then**
    Packet **Exceeds**;
    EBS = EBS – S;
else
    Packet **Conforms**;
    EBS = EBS – S;
    CBS = CBS – S;
end if.

Effectively, the two-rate policer runs two sliding windows over the packet stream, metering against two bitrates using two separate averaging intervals. This procedure is common on Service Provider edges that offer oversubscription to customers, based on PIR and CIR rates.

Start verification by generating a single HTTP flow from R1 down to R8 across R4. Shape this down to 128kbps on R1.

```
R1:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
!
policy-map SHAPE_VLAN146
```

https://t.me/learningnets

```
 class class-default
  shape average 128000 12800 0
!
interface GigabitEthernet1.146
  service-policy output SHAPE_VLAN146
R4:
interface GigabitEthernet1
 load-interval 30
R5:
interface Tunnel0
 shutdown
R8#copy http://admin:cisco@155.1.146.1/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
```

Note that traffic from R1 to R8 exceeds the policer average rate but not the peak rate. Therefore, the policer marks half of the packets as exceeding. Because the traffic is TCP based, it adapts to the maximum allowable rate:

```
R4#show policy-map interface GigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy input: POLICE_VLAN146

    Class-map: HTTP (match-all)
      2099 packets, 1241124 bytes
      30 second offered rate 122000 bps Match: access-group name HTTP

      Service-policy : SUBRATE_POLICER

        Class-map: FROM_R1 (match-all)
          2099 packets, 1241124 bytes
          30 second offered rate 122000 bps, drop rate 0000 bps
          Match: access-group name FROM_R1
          police: cir 64000 bps, bc 3200 bytes
pir 128000 bps, be 6400 bytes
            conformed 1059 packets, 623393 bytes; actions:
              set-prec-transmit 1
            exceeded 1040 packets, 617731 bytes; actions:
              set-prec-transmit 0
            violated 0 packets, 0 bytes; actions:
              drop  conformed 61000 bps, exceeded 62000 bps, violated 0000 bps



        Class-map: FROM_R6 (match-all)
          0 packets, 0 bytes
```

https://t.me/learningnets

```
          30 second offered rate 0000 bps, drop rate 0000 bps
          Match: access-group name FROM_R6
          police:
              cir 64000 bps, bc 3200 bytes
              pir 128000 bps, be 6400 bytes
            conformed 0 packets, 0 bytes; actions:
              set-prec-transmit 1
            exceeded 0 packets, 0 bytes; actions:
              set-prec-transmit 0
            violated 0 packets, 0 bytes; actions:
              drop
            conformed 0000 bps, exceeded 0000 bps, violated 0000 bps

        Class-map: class-default (match-any)
          0 packets, 0 bytes
          30 second offered rate 0000 bps, drop rate 0000 bps
          Match: any

    Class-map: class-default (match-any)
      203 packets, 16054 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

Now add another HTTP flow, from R6 down to R10 and across the Ethernet link between R4 and R5. Shape the R4-R5 Ethernet link down to 128kbps to aid in smoothing out both simultaneous flows, as well as the R6 connection to VLAN 146. Similar to how R1 is shaping:

```
R4:
policy-map SHAPE_R4_R5
 class class-default
  shape average 128000 12800 0
!
interface GigabitEthernet1.45
  service-policy output SHAPE_R4_R5

R6:
username admin privilege 15 password cisco
ip http authentication local
ip http server
ip http path bootflash:
!
policy-map SHAPE_VLAN146
 class class-default
  shape average 128000 12800 0
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.146
  service-policy output SHAPE_VLAN146
R10#copy http://admin:cisco@155.1.146.6/csr1000v-mono-universalk9.03.11.01.S.154-1.S1-std.SPA.pkg null:
```

Verify the policer statistics again. Now both policers meter the same average rate, and there is no exceeding traffic:

```
R4#show policy-map interface GigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy input: POLICE_VLAN146

    Class-map: HTTP (match-all)
      8042 packets, 4766120 bytes 30 second offered rate 128000 bps
      Match: access-group name HTTP

      Service-policy : SUBRATE_POLICER

        Class-map: FROM_R1 (match-all)
          6627 packets, 3930756 bytes 30 second offered rate 63000 bps
, drop rate 0000 bps
          Match: access-group name FROM_R1
          police:
              cir 64000 bps, bc 3200 bytes
              pir 128000 bps, be 6400 bytes
            conformed 4019 packets, 2381633 bytes; actions:
              set-prec-transmit 1
            exceeded 2608 packets, 1549123 bytes; actions:
              set-prec-transmit 0
            violated 0 packets, 0 bytes; actions:
              drop  conformed 63000 bps, exceeded 0000 bps
, violated 0000 bps

        Class-map: FROM_R6 (match-all)
          1415 packets, 835364 bytes 30 second offered rate 64000 bps
, drop rate 0000 bps
          Match: access-group name FROM_R6
          police:
              cir 64000 bps, bc 3200 bytes
              pir 128000 bps, be 6400 bytes
            conformed 1415 packets, 835364 bytes; actions:
              set-prec-transmit 1
            exceeded 0 packets, 0 bytes; actions:
```

https://t.me/learningnets

```
              set-prec-transmit 0
            violated 0 packets, 0 bytes; actions:
              drop  conformed 64000 bps, exceeded 0000 bps
, violated 0000 bps

        Class-map: class-default (match-any)
          0 packets, 0 bytes
          30 second offered rate 0000 bps, drop rate 0000 bps
          Match: any

    Class-map: class-default (match-any)
      300 packets, 23708 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

MQC Percent-Based Policing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Use an inbound policer on R1's link to VLAN 146 that rate-limits the link to 1% of its original rate (1 Gbps).
- Use a burst value of 125ms.

Configuration

R1:

```
policy-map POLICE_VLAN146
 class class-default
   police rate percent 1 burst 125 ms
!
interface GigabitEthernet1.146
 service-policy input POLICE_VLAN146
```

Verification

Percent-based policing allows specifying the policer rate as a percentage of the interface speed. At the same time, you may configure burst sizes in milliseconds duration. The actual burst size will be *InterfaceSpeed * BurstDuration* based on the interface's speed. This method does not use "absolute" values but rather relative

https://t.me/learningnets

settings based on interface parameters. The drawback is a limited range of speed values, because CLI does not allow fractional percent values. Thus, you may only step by one percent configuring policed rates.

The metering procedure remains the same as with srTCM or trTCM, you just change the way to set configuration parameters.

To verify, send a stream of 1000-byte ICMP packets from R4 to R1.

```
R1:
interface GigabitEthernet1
 load-interval 30

R4#ping 155.1.146.1 size 1000 timeout 0 repeat 100000000
```

Check the policer statistics to view number of conforming and exceeding packets:

```
R1#show policy-map interface GigabitEthernet1.146
  GigabitEthernet1.146

   Service-policy input: POLICE_VLAN146

     Class-map: class-default (match-any)
       909928 packets, 385710196 bytes 5 minute offered rate 7080000 bps, drop rate 2097000 bps
       Match: any
       police: rate 1 %, burst 125 ms

           rate 10000000 bps, burst 156250 bytes
         conformed 806328 packets, 280246336 bytes; actions:
           transmit
         exceeded 103600 packets, 105463860 bytes; actions:
           drop
         conformed 4991000 bps, exceeded 2097000 bps
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

QoS Pre-Classify

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Create a GRE tunnel between R1 and R6, using their VLAN 146 Ethernet interfaces as the underlay.
- Route traffic between the Loopback0 subnets of R1 and R6 across the tunnel using static /32 routes.
- Limit the rate of traffic leaving the VLAN 146 interface of R6 to 10Mbps.
- Configure R6 to guarantee 3Mbps of priority bandwidth to IP traffic between the Loopback0 subnets of R1 and R6 marked with DSCP EF on the VLAN146 interface.

Configuration

```
R1:
interface Tunnel46
 tunnel source 155.1.146.1
 tunnel destination 155.1.146.6
 ip unnumbered GigabitEthernet1.146
!
router eigrp INE_RSv5
 !
 address-family ipv4 unicast autonomous-system 123
  !
  af-interface Tunnel46
   passive-interface
```

https://t.me/learningnets

```
!
ip route 150.1.6.6 255.255.255.255 Tunnel46
```

R6:

```
interface Tunnel46
 tunnel source 155.1.146.6
 tunnel destination 155.1.146.1
 ip unnumbered GigabitEthernet1.146
 qos pre-classify
!
!
router eigrp INE_RSv5
 !
 address-family ipv4 unicast autonomous-system 123
  !
  af-interface Tunnel46
   passive-interface
!
ip route 150.1.1.1 255.255.255.255 Tunnel46
!
ip access-list extended LOOPBACKS
 permit ip host 150.1.6.6 host 150.1.1.1
!
class-map LOOPBACKS_DSCP_EF
 match access-group name LOOPBACKS
 match dscp ef
!
policy-map LLQ
 class LOOPBACKS_DSCP_EF
 priority 3000
!
policy-map SHAPE_VLAN_146
 class class-default
  shape average 10000000
  service-policy LLQ
!
interface GigabitEthernet1.146
 service-policy output SHAPE_VLAN_146
```

Verification

From a QoS classification perspective, traffic routed inside tunnels, such as GRE or IPsec, may pose a serious problem. Imagine a situation in which multiple tunnels go

https://t.me/learningnets

across the same physical interface of a router in addition to non-tunneled traffic leaving the same interface. You also need to limit the aggregate outgoing traffic rate and share bandwidth between tunneled and non-tunneled traffic. If you apply a service policy to the physical interface, the policy will treat traffic inside the tunnel as a single flow, sourced off of the tunnel endpoints addresses. Because the classification takes into account only the top-most header, the system cannot distinguish traffic flows inside the tunnel. Several solutions exist to make the router aware of the “tunneled” traffic structure.

1. ToS reflection. When IOS encapsulates an IP packet with a tunnel header, it copies the ToS byte to the tunnel header. This makes it possible to distinguish traffic “classes” between the tunnel endpoints. However, specific flows remain “hidden” from the physical interface level policy, limiting the usefulness of flow-based schedulers like WFQ.

2. Applying service-policy to the tunnel interface. IOS allows applying GTS (Generic Traffic Shaping) and CBTS (Class Based Shaping) to GRE/IPIP tunnels. Using CBTS, you may shape traffic going across the tunnel and tune the shaper queue the way you want. This method has its limitations. First, not all tunnel technologies support CBTS. For example, you cannot apply CBTS inside an IPsec tunnel or apply it to Multipoint GRE tunnel. More seriously, even if the previous issue could be resolved (QoS for DMVPN Networks), you still cannot create one unified aggregate policy for all traffic. That is, you need to limit each tunnel up to its maximum rate and apply CBWFQ inside the tunnel. In addition, you have to configure some policy at the physical interface level, such as a bandwidth reservation for tunnel traffic flow. This makes the configuration process complicated.

3. QoS pre-classification. When you turn on this feature on a tunnel interface (GRE/mGRE, IPIP, IPsec, Virtual-Template), you no longer need to apply a service-policy inside the tunnel interface. Because of QoS pre-classification, the service-policy applied at the interface level can “see” the tunnel encapsulated packets as they cross the interface without any encapsulation. To do this, IOS keeps a temporary copy of the headers in memory. This copied header is taken before the tunnel encapsulation, or encryption. Th physical interface-level policy still accounts for tunnel header overhead, thus allowing for fair scheduling.

The QoS pre-classify feature is useful in scenarios similar to the described above, such as if you need to share the physical interface between tunneled and non-tunneled traffic. With pre-classification, you apply the service-policy to the physical

https://t.me/learningnets

interface and enable the feature on all tunnel interfaces, treating all traffic (from a QoS prospective) as though it is not encapsulated.

To verify this, generate packets from the Loopback0 of R6 to the Loopback0 of R1. This flow will be GRE encapsulated.

```
R6#show ip cef 150.1.1.1 detail

150.1.1.1/32, epoch 2, flags attached attached to Tunnel46

R6#ping

Protocol [ip]: Target IP address: 150.1.1.1
Repeat count [5]: 1000
Datagram size [100]: 1200
Timeout in seconds [2]: 1 Extended commands [n]: y
Source address or interface: 150.1.6.6
Type of service [0]: 184
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1000, 1200-byte ICMP Echos to 150.1.1.1, timeout is 1 seconds:
Packet sent with a source address of 150.1.6.6
!!!
```

```
      (pkts output/bytes output) 1071/1247626
      shape (average) cir 10000000, bc 40000, be 40000 target shape rate 10000000

      Service-policy : LLQ

        queue stats for all priority classes:
          Queueing
          queue limit 512 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 1000/1242000
 Class-map: LOOPBACKS_DSCP_EF (match-all)
   1000 packets, 1242000 bytes
          5 minute offered rate 26000 bps, drop rate 0000 bps
          Match: access-group name LOOPBACKS Match:  dscp ef (46)
 Priority: 3000 kbps
, burst bytes 75000, b/w exceed drops: 0


        Class-map: class-default (match-any)
          71 packets, 5626 bytes
          5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: any

          queue limit 64 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 71/5626
```

Note the size of the ICMP packets classified by the service-policy. A total of 1242000 bytes for 1000 packets means a size of 1242 bytes for each packet. Subtract 18 bytes for the tagged Ethernet frame, and the remaining 24 bytes are the for GRE encapsulation header. Note that the size of packets specified in the extended ping was 1200. This means that QoS pre-classification correctly accounts for the tunnel overhead.

To see what happens without QoS pre-classification, remove the command from the Tunnel46 interface of R6 and send another batch of pings. To get a fresh packet count, remove and re-apply the service policy from the GigabitEthernet1.146 interface on R6.

```
R6:
interface Tunnel46
 no qos pre-classify
!
interface GigabitEthernet1.146
 no service-policy output SHAPE_VLAN_146
```

https://t.me/learningnets

```
 service-policy output SHAPE_VLAN_146

R6#ping

Protocol [ip]: Target IP address: 150.1.1.1
Repeat count [5]: 1000
Datagram size [100]: 1200
Timeout in seconds [2]: 1Extended commands [n]: y
Source address or interface: 150.1.6.6
Type of service [0]: 184

Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1000, 1200-byte ICMP Echos to 150.1.1.1, timeout is 1 seconds:
Packet sent with a source address of 150.1.6.6
!!!
```

```
target shape rate 10000000

Service-policy : LLQ

  queue stats for all priority classes:
    Queueing
    queue limit 512 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 0/0

  Class-map: LOOPBACKS_DSCP_EF (match-all)  0 packets, 0 bytes
    5 minute offered rate 0000 bps, drop rate 0000 bps
    Match: access-group name LOOPBACKS
    Match:  dscp ef (46)
    Priority: 3000 kbps, burst bytes 75000, b/w exceed drops: 0


  Class-map: class-default (match-any)  1014 packets, 1243136 bytes

    5 minute offered rate 0000 bps, drop rate 0000 bps
    Match: any

    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 1014/1243136
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - QoS

Advanced HTTP Classification with NBAR

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **QoS Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** in order to complete this task.

Task

- Configure R6's VLAN146 interface so that HTTP transfers of files with extensions ".bin", ".text", and ".taxt" are limited to 256 Kbps.
- Use only one line to match the URL strings.
- Use the method most friendly for TCP connections.
- Do not apply this limit to the transfers of any other file types.

Configuration

R6:

```
class-map match-all EXTENSION
 match protocol http url "*.bin|*.t[ea]xt"
!
!
policy-map SHAPE
 class EXTENSION
  shape average 256000
!
interface GigabitEthernet1.146
  service-policy output SHAPE
```

https://t.me/learningnets

Verification

NBAR (Network Based Application Recognition) protocol classification allows deep inspection for some protocol types. One most notable feature is matching various fields inside HTTP headers. In recent IOS versions, this capability evolved into a complete Flexible Packet Matching feature that allows for comprehensive inspection of IP packets.

Most commonly, you will see HTTP inspection matching the URL field against certain strings.

The command to match URL is `match http protocol url <pattern> .`

The following is the list of the available wildcards used to construct the matching patterns:

"*" —Match any pattern, such as “aaa”, “abcd1234”. To match the substring “xyz” in the beginning of a string, use the pattern “xyz*”. To match “xyz” anywhere in the string, use the pattern "*xyz*". Use the pattern “*xyz” to match the substring in the end. Note that pattern “xyz” matches only the exact string “xyz”. You may also use complicated patterns “ab*cd*ef”, probably at the expense of some CPU penalty.

"?" —Match any single character. For example, pattern “???” matches “xyz”, “abc”, “efg”, “123”. You can mix “*” and “?” as in pattern “tes*.????”

"[ ]" —Match a range of characters. For example, “[abc]” will match any single character “a”, “b”, or “c”.

"|" —Alternative. Separate patterns with “|”to specify “OR” matching logic. For example, “xyz|abc” matches either full “xyz” or “abc” strings. You may mix “|” with other globs, like this: “*xyz*|*abc*|*pqr*”. Keep the overall limit of using the “|” symbol to a minimum, to make matching faster.

“()” —Grouping. Denotes the boundaries of a sub-pattern. For example, instead of “*.txt|*.bin”, you can write “*.(txt|bin)”.

All matching is case insensitive. Therefore, the pattern “text” matches “TEXT” as well. The engine matches your URL pattern against the directory path and the file name in the URL (NOT the server DNS name!). For example, if the URL is “http://www.cisco.com/pub/uploads/image.jpeg”, the URL matching will only match the “pub/uploads/image.jpeg” part of the URL. As a matter of fact, when you submit a request like the above URL, it translates into the following headers (there are actually more, but this is the bare minimum):

https://t.me/learningnets

```
GET /pub/uploads/image.jpeg HTTP/1.1
Host: www.cisco.com
```

When you apply a policy-map that contains a class with “match protocol” statement, the system starts NBAR classification engine on the interface. Any packet, ingress or egress, passes the NBAR inspection engine provided that it passes the basic filters such as matching the port number assigned to the protocol. You can change the port map by using the global command **`ip nbar port-map [protocol] .`**

When the engine sees a TCP SYN packet for the matching session, it starts the internal state machine, trying to parse the packet flow. Every new packet in the flow (in any direction) is inspected. Note that NBAR does not classify a flow instantly. It may take some packet exchange until the engine determines that the flow matches specified criteria. As soon as there is enough information about the flow to classify it, the engine “tags” the bi-directional flow with the corresponding class value and reports this decision to the policy-map. Note that this classification applies in both directions—that is, to the packets belonging to the flow and heading either ingress or egress.

For example, if a user starts a web sessions and opens a URL matching any of your NBAR criteria, the engine will classify the flow as soon as it sees the packet with the URL string. After this, both packet flows from the client to the server or from the server to the client would the respective class. The policy map could be applied in either direction of the interface.

The engine will remove the “tag” when it faces the flow “end” criteria: for example, it catches a TCP FIN or TCP RST flag. After this, the flow is no longer monitored by NBAR and reported as matching the respective class.

Consider the example below. In this scenario, GigabitEthernet1.37 is the outside interface, facing the internet. The user’s traffic flows out of this interface.

https://t.me/learningnets

```
class-map match-all TEST
 match protocol http url "*.(t?xt|ocx|ex[ea])
!
policy-map TEST
 class TEST
 police 8000 conform-action drop  exceed-action drop  violate-action drop
!
interface GigabitEthernet1.37
ip address 155.1.37.3 255.255.255.0
service-policy input TEST
```

However, as soon as the user opens the URL matching the class-map specification, the engine will classify the flow as matching the class “TEST”. After this, the policy will drop all returning packets (server to client) for this flow.

Note that even though this configuration works, it is not the recommended way of applying the URL filtering. This is because the actual GET requests will still get to the destination server and generate response traffic. In some situations, the client may send a FIN packet when the server generates the response. At this point, NBAR will report the flow as destroyed and the policy-map will not match the packets. If the server response is slow enough, it will actually bypass the ingress filter! Thus, it is always recommended to apply the NBAR classification and policy action in the direction matching the direction of the protocol commands (apply the URL filtering in the direction of GET commands).

We will use the following single-line expression for our task:

“*.bin|*.t[ea]xt”

This pattern matches either of two strings that end with “.bin” or “.text”, “.taxt”. You can also use “match-any” class map and match multiple URL strings on separate lines, resulting in the same effect as separating pattern with the symbol “|”.

You can match other fields in the HTTP headers. Specifically, you can match a pattern against Client and Server headers, using the expressions:

```
`match http protocol c-header-field <string>`
`match http protocol s-header-field <string>`
```

Those expressions match any of the client or server headers. Additionally, you can match the “Host” header field separately by using the command match **`http protocol host <string>`** .

https://t.me/learningnets

Finally, you can match MIME types to distinguish certain “types” of files being transferred (such as audio/video) by using the following command:

```
`match http protocol mime <MIME-Type>`
```

For example:

```
`match http protocol mime “image/jpeg”`
```

However, matching URIs is the most common use of advanced HTTP classification. To verify the configuration, set R7 as an HTTP server and create several file-names in the flash memory with extensions “.bin”, “.text”, “.taxt”, and “tsxt”.

```
R7:
 username admin privilege 15 password cisco
 ip http authentication local
 ip http server
 ip http path bootflash:
 R7#copy bootflash:csrlxc-cfg.log bootflash:config.bin
R7#copy bootflash:csrlxc-cfg.log bootflash:config.tsxt
R7#copy bootflash:csrlxc-cfg.log bootflash:config.taxt
R7#copy bootflash:csrlxc-cfg.log bootflash:config.text
```

Now we have four files to try. Using R4 as HTTP client, verify that “.bin”, “.text”, and “.taxt” match our configuration but “.tsxt” does not:

```
R4#copy http://admin:cisco@155.1.67.7/config.bin null:

Loading http://***********@155.1.67.7/config.bin !
1585 bytes copied in 0.060 secs (26417 bytes/sec)
```

Notice that there are 17 packets that were matched:

```
R6#show policy-map interface GigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy output: SHAPE

    Class-map: EXTENSION (match-all)   17 packets, 4598 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: protocol http url "*.bin|*.t[ea]xt"
      Queueing
```

https://t.me/learningnets

```
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0 (pkts output/bytes output) 17/4598
      shape (average) cir 256000, bc 1024, be 1024
      target shape rate 256000

    Class-map: class-default (match-any)
      26 packets, 1976 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
R4#copy http://admin:cisco@155.1.67.7/config.text null:

Loading http://***********@155.1.67.7/config.text !
1585 bytes copied in 0.070 secs (22643 bytes/sec)
```

Another 17 packets were matched, totaling 34:

```
R6#show policy-map interface GigabitEthernet1.146

GigabitEthernet1.146

 Service-policy output: SHAPE

   Class-map: EXTENSION (match-all)  34 packets, 9196 bytes
     5 minute offered rate 0000 bps, drop rate 0000 bps
     Match: protocol http url "*.bin|*.t[ea]xt"
     Queueing
     queue limit 64 packets
     (queue depth/total drops/no-buffer drops) 0/0/0
     (pkts output/bytes output) 34/9196
     shape (average) cir 256000, bc 1024, be 1024
     target shape rate 256000

   Class-map: class-default (match-any)
     55 packets, 4250 bytes
     5 minute offered rate 0000 bps, drop rate 0000 bps
     Match: any

R4#copy http://admin:cisco@155.1.67.7/config.taxt null:

Loading http://***********@155.1.67.7/config.taxt !
1585 bytes copied in 0.062 secs (25565 bytes/sec)
```

Yet again, another 17 packets. Now the counter is at 51:

https://t.me/learningnets

```
R6#show policy-map interface GigabitEthernet1.146

 GigabitEthernet1.146

  Service-policy output: SHAPE

    Class-map: EXTENSION (match-all)  51 packets, 13780 bytes
      5 minute offered rate 1000 bps, drop rate 0000 bps
      Match: protocol http url "*.bin|*.t[ea]xt"
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 51/13780
      shape (average) cir 256000, bc 1024, be 1024
      target shape rate 256000

    Class-map: class-default (match-any)
      84 packets, 6524 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any

R4#copy http://admin:cisco@155.1.67.7/config.tsxt null:

Loading http://***********@155.1.67.7/config.tsxt !
1585 bytes copied in 0.037 secs (42838 bytes/sec)
```

Note now that due to the policy, the .tsxt file was not matched and the packet counter did not increase (still 51):

```
R6#show policy-map interface GigabitEthernet1.146
GigabitEthernet1.146

 Service-policy output: SHAPE

   Class-map: EXTENSION (match-all)  51 packets, 13780 bytes

     5 minute offered rate 0000 bps, drop rate 0000 bps
     Match: protocol http url "*.bin|*.t[ea]xt"
     Queueing
     queue limit 64 packets
     (queue depth/total drops/no-buffer drops) 0/0/0
     (pkts output/bytes output) 51/13780
     shape (average) cir 256000, bc 1024, be 1024
     target shape rate 256000
```

https://t.me/learningnets

```
    Class-map: class-default (match-any)
      136 packets, 13824 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

AAA Authentication Lists

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 to use a TACACS+ server at the IP address 155.1.146.100 and encrypt communications using a key value of **CISCO**.
  - Source TACACS+ packets from the Loopback0 interface.
- Configure the router so that access to the console line uses local authentication.
  - Create a user named **ADMIN** with a password of **CISCO** in the local database for this.
- Ensure that the VTY lines authenticate users with TACACS+, and then fallback to line authentication with a password of **cisco**.
- Privilege mode authentication should first attempt TACACS+ and then fallback to local password of **cisco**.
- Customize the prompts for AAA user authentication, change the default banner and authentication failure message.

Configuration

R1:

```
enable secret cisco
username ADMIN password CISCO
!
aaa new-model
aaa authentication login CONSOLE local
aaa authentication login VTY group tacacs+ line
```

https://t.me/learningnets

```
aaa authentication enable default group tacacs+ enable
!
aaa authentication password-prompt "Please Enter Your Password:"
aaa authentication username-prompt "Please Enter Your ID:"
!
aaa authentication banner #
This system requires you to identify yourself.
#
aaa authentication fail-message #
Authentication Failed, Sorry.
#
!
ip tacacs source-interface loopback0
tacacs-server host 155.1.146.100
tacacs-server key CISCO
!
line con 0
 login authentication CONSOLE
!
line vty 0 4
 login authentication VTY
 password cisco
```

Verification

Cisco routers and switches may now use a “new” authentication model. The idea of the new model is to apply configurable lists of methods to authenticate, authorize, and/or perform accounting for certain processes. The processes that we are typically most interested in are:

- Login process. This is the procedure to obtain access to the router, usually requiring you to present your identity.
- Privileged mode access. To access a certain level of enable mode privilege, the system requires user authentication. You can configure local enable passwords for each level of access. You can also configure a named list of methods using the command **`aaa authentication <process> [default|<NAME>] <List of Methods>`** . You may apply this list to a certain subsystem, such as to terminal or console lines. Note that a default list applies to all subsystems that have no explicit AAA list applied to them.

As for the list of possible methods, we should be most concerned with the following:

- Local method. Uses the local user database with their passwords. You populate the

https://t.me/learningnets

database by using the `username` command.

- Local-case method. The same as the local method, but makes passwords case-sensitive.
- Line method. Uses the password configured on the line used to access the router. This includes VTY lines as well.
- Enable method. Uses the globally configured list of enable passwords associated with their levels.
- Group TACACS+ or RADIUS method. Uses the remote AAA servers group configured globally in the router.
- None method. Do not attempt to validate user identity, just allow access.

Note this important property of the AAA list: The router tries all methods in sequence, switching to the next one if the previous method fails or returns an error. Here, fail means authentication method failure, such as a non-responsive server, not the failure of the user account to be authenticated. This means that this list allows you to provide authentication redundancy. For example, consider a list that uses TACACS+ authentication first, and then switches to method none. If your TACACS+ server does not respond, the router tries the next method and allows access. However, if the AAA server returns an authentication failure message, the router terminates the list search process and denies access to the individual.

In this task configuration, you should verify the authentication process by trying to log in to the console line first:

```
R1 con0 is now available



Press RETURN to get started.



This system requires you to identify yourself.

Please Enter Your ID:ADMIN
Please Enter Your Password:CISCO
R1#
```

https://t.me/learningnets

Note that the privilege level of the console line has been set to 15 by the initial configurations, so the user logs directly to privileged mode. Enable AAA debugging and try logging in via a VTY line:

```
R1#debug aaa authentication

AAA Authentication debugging is on
!R1#telnet 150.1.1.1
Trying 150.1.1.1 ... Open

AAA/BIND(0000000D): Bind i/f  AAA/AUTHEN/LOGIN (0000000D): Pick method list 'VTY'

This system requires you to identify yourself.

Please Enter Your Password:cisco
AAA/AUTHEN/LINE(0000000D): GET_PASSWORD
```

The router attempts to contact the TACACS+ server, fails, and uses the second method, which is the line password:

```
AAA/AUTHEN/LINE(0000000D): PASS
AAA/AUTHOR (0000000D): Method=None for method list id=00000000. Skip author
R1>enable
```

Now try entering the wrong password for privileged mode authentication. Note that the system tries the TACACS+ method first, but it returns the ERROR message. Thus, the system tries to use the enable password for authentication, but the password entered does not match. Note the Status=FAI" response when you enter the wrong password:

```
AAA: parse name=tty66 idb type=-1 tty=-1
AAA: name=tty66 flags=0x11 type=5 shelf=0 slot=0 adapter=0 port=66 channel=0
AAA/MEMORY: create_user (0x85646E6C) user='NULL' ruser='NULL' ds0=0 port='tty66' rem_addr='150.1.1.1' authen_type=AS
AAA/AUTHEN/START (3199108865): port='tty2' list='' action=LOGIN service=ENABLE  AAA/AUTHEN/START (2250436155):
using "default" list
AAA/AUTHEN/START (2250436155):Method=tacacs+ (tacacs+)
TAC+: send AUTHEN/START packet ver=192 id=-2044531141 AAA/AUTHEN(2250436155):Status=ERROR
 AAA/AUTHEN/START (2250436155):Method=ENABLE
AAA/AUTHEN(2250436155): Status=GETPASS AAA/AUTHEN/CONT (2250436155): continue_login (user='(undef)')
 Please Enter Your Password: 12345

AAA/AUTHEN(2250436155): Status=GETPASS AAA/AUTHEN/CONT (2250436155):Method=ENABLE
```

https://t.me/learningnets

```
AAA/AUTHEN(2250436155): password incorrect AAA/AUTHEN(2250436155): Status=FAIL
AAA/MEMORY: free_user (0x85646E6C) user='NULL' ruser='NULL' port='tty2' rem_addr='150.1.1.1' authen_type=ASCII servi
% Access denied
```

Try authenticating again, entering the correct password this time. Note that the system tries the TACACS+ method again and then switches to the enable method. This time, passwords match and the reply is PASS. In the debugging output below, note that username is “null” or “undef” when you log in via the VTY line. This is because we used the line password for authentication and never supplied a username:

```
R1>enable

AAA: parse name=tty2 idb type=-1 tty=-1
AAA: name=tty66 flags=0x11 type=5 shelf=0 slot=0 adapter=0 port=66 channel=0
AAA/MEMORY: create_user (0x848DC170) user='NULL' ruser='NULL' ds0=0 port='tty66' rem_addr='150.1.1.1' authen_type=AS
AAA/AUTHEN/START (649338587): port='tty66' list='' action=LOGIN service=ENABLE
AAA/AUTHEN/START (649338587): using "default" list
AAA/AUTHEN/START (649338587): Method=tacacs+ (tacacs+)
TAC+: send AUTHEN/START packet ver=192 id=649338587 AAA/AUTHEN(649338587): Status=ERROR
Please Enter Your Password:cisco

AAA/AUTHEN/START (3711900598): Method=ENABLE
AAA/AUTHEN (3711900598): status = GETPASS AAA/AUTHEN/CONT (3711900598): continue_login (user='(undef)')
AAA/AUTHEN (3711900598): status = GETPASS
AAA/AUTHEN/CONT (3711900598): Method=ENABLE
AAA/AUTHEN (3711900598): status = PASS AAA/MEMORY: free_user (0x7F2665E41B68) user='NULL'
 ruser='NULL' port='tty2' rem_addr='150.1.1.1' authen_type=ASCII service=ENABLE priv=15 vrf= (id=0) R1#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

AAA Exec Authorization

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 to use a TACACS+ server at the IP address 155.1.146.100 and encrypt communications using a key value of **CISCO**.
  - Source TACACS+ packets from the Loopback0 interface.
- Configure the router so that access to VTY lines uses local authentication.
  - Create a user named **ADMIN** with a password of **CISCO** in the local database for this with a privilege level of 7.
- Authorize exec privilege levels using the remote TACACS+ server.
  - Ensure that if the TACACS+ server is unavailable, authenticated users logging in via VTY lines are placed in privilege level 15.
  - Use the VTY line settings to authorize any authenticated users if the TACACS+ server fails.

Configuration

R1:

```
ip tacacs source-interface loopback0
tacacs-server host 155.1.146.100
tacacs-server key CISCO
!
aaa new-model
aaa authentication login VTY local
aaa authorization exec VTY group tacacs+ if-authenticated
```

https://t.me/learningnets

```
!
username ADMIN privilege 7 password 0 CISCO
!
line vty 0 4
 privilege level 15
 authorization exec VTY
 login authentication VTY
```

Verification

Authorization is a procedure for granting certain rights to a process, or granting a permission to perform a certain action. The authorization procedure is only possible for authenticated entities. The identity of a subject is used to look up the policy and determine the permissions. This is why authentication always precedes authorization. In some cases, it is possible to grant some rights to unidentified subjects. The goal of exec authorization is to assign a privilege level (0–15) to a logged-in user. You configure an exec authorization list using the command **`aaa authorization exec [default|<NAME>] <Method List>`** . As with authentication, you can define a default list (which is used system wide) or apply a specific list per terminal line. Generally, there are three methods to obtain authorization information:

1. Consult a remote AAA server and download the user attributes. TACACS+ performs this procedure as a separate operation, but RADIUS has no explicit authorization state, and returns authorization information in authentication replies. Here is an example of using TACACS+ as the source of the required information: **`aaa authorization exec default group tacacs+`** .
2. Consult the local username database, looking for the privilege level assigned to the authenticated user: **`aaa authorization exec default local`** .
3. Use default settings, such as the default privilege level assigned to the terminal line, if the authorization configuration permits. This is commonly used when you disable authorization (method “none”) or authorize settings for any authenticated users (method “if-authenticated”). Note the difference between the method “none” and “if-authenticated” in the following example.

Scenario 1:

```
aaa authentication login default tacacs+ none
aaa authorization exec default none
!
line console 0
```

https://t.me/learningnets

```
 privilege level 15
```

Scenario 2:

```
aaa authentication login default tacacs+ none
aaa authorization exec default if-authenticated
!
line console 0
 privilege level 15
```

In the first case, if the TACACS+ server is not available, the router will allow incoming console connections without authentication. Because there is no exec authorization, the user will be granted the exec shell with privilege 15. In the second case, if the TACACS+ server is not available, the system grants access without authentication but fails authorization of exec shell. Thus, the difference between “none” and “if-authenticated” authorization cases is that the former always applies the desired authorization parameters without any verification. The latter requires the user to be authenticated, but does not consult the user database to check authorization attributes.

By default, exec authorization is set to “none,” so you may need to change it to accomplish your needs. Also, note that IOS routers by default do not authorize exec sessions on the console line. On the contrary, Catalyst IOS always authorizes the exec shell, even on the console line. Therefore, if you disable console authentication in the Catalyst switch, make sure that you never apply an AAA authorization list to the console (explicitly or using the default settings). You may enable console exec authorization in IOS routers using the command **`aaa authorization console`** .

Now enable AAA and TACACS debugging, try logging in via a VTY line:

```
R1#debug aaa authentication

AAA Authentication debugging is on
!R1#debug aaa authorization

AAA Authorization debugging is on
!R1#debug tacacs
TACACS access control debugging is on
!R1#telnet 150.1.1.1
Trying 150.1.1.1 ... Open

User Access Verification

Username:
AAA/BIND(00000016): Bind i/f  AAA/AUTHEN/LOGIN (00000016): Pick method list 'VTY' ADMIN
```

https://t.me/learningnets

```
Password: AAA/AUTHOR (0x16): Pick method list 'VTY'

TPLUS: Queuing AAA Authorization request 22 for processing
TPLUS(00000016) login timer started 1020 sec timeout
TPLUS: processing authorization request id 22
TPLUS: Protocol set to None .....Skipping
TPLUS: Sending AV service=shell
TPLUS: Sending AV cmd*TPLUS: Authorization request created for 22(ADMIN)
TPLUS: Using server 155.1.146.100
TPLUS(00000016)/0/NB_WAIT/7FC59FB112A0: Started 5 sec timeout
!
!TPLUS(00000016)/0/NB_WAIT/7FC59FB112A0: timed out
TPLUS(00000016)/0/NB_WAIT/7FC59FB112A0: timed out, clean up
TPLUS(00000016)/0/7FC59FB112A0: Processing the reply packet - PASS - PASS
AAA/AUTHOR/EXEC(00000016): processing AV cmd=AAA/AUTHOR/EXEC(00000016): Authorization successful
!R1#show privilege
Current privilege level is 15

AAA/AUTHOR: auth_need : user= 'ADMIN' ruser= 'R1'rem_addr= '150.1.1.1' priv= 1 list= '' AUTHOR-TYPE= 'commands'
```

The router attempts to contact the TACACS+ server for authorization, fails, and uses the second method, the "if-authenticated", thus it authorizes the user for shell access. The user is assigned a privilege-level of 7, but the line configuration with a privilege-level of 15 overrides the user settings.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

AAA Local Command Authorization

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure local username authentication for VTY lines of R1.
- Create a user named **ADMIN** with a password of **CISCO** in the local database of R1 with a privilege level of 7.
  - Ensure that the user ADMIN can use RIP debugging commands and can disable any currently active debugging using a single command.
  - The same user should be able to configure any interface IP settings and administratively enable or disable any of these interfaces.
  - Ensure that the user can see his permitted commands in his running configuration.

Configuration

R1:

```
aaa new-model
aaa authentication login VTY local
aaa authorization exec VTY local
!
username ADMIN privilege 7 password 0 CISCO
!
line vty 0 4
 login authentication VTY
 authorization exec VTY
```

https://t.me/learningnets

```
!
privilege exec level 7 configure terminal
privilege exec level 7 undebug all
privilege exec level 7 show running-config
privilege exec level 7 debug ip rip
!
privilege configure level 7 interface
!
privilege interface level 7 shutdown
privilege interface level 7 no shutdown
privilege interface all level 7 ip
```

Verification

IOS allows configuration of command authorization by using the local configuration database. Command authorization permits groups of users to use specific commands. IOS also supports remote command authorization with the TACACS+ protocol, but this is out of the scope of the CCIE Routing & Switching Lab Exam. Local command authorization uses the concept of privilege levels. There are sixteen levels supported, 0 to 15. Each level supports the commands found in all previous levels; for example, privilege 5 includes levels 0–5, and privilege 15 includes levels 0–15. By default, IOS has three pre-configured privilege levels:

- Level 0: Just a few basic commands, such as `enable`, `login`, and `exit`.
- Level 1: The default exec user level; has some show commands available, but no configuration commands.
- Level 15: The maximum privilege level, also known as privileged mode or enable mode; includes all the commands available in IOS.

To create custom command sets, you can perform one of the following:

- Assign some level-15 commands to level 1, effectively making them available to all users who may log in to the router (if they use the default privilege level settings). You may want to use this option if you need to allow all users to use some special features, such as certain debug commands.
- Re-assign some commands from level 1 to a higher level, thus disallowing all unprivileged users the use of this command. For example, you may want to disallow the use of the `show ip access-list` command for all default privilege users.
- Assign some level-15 commands to a new custom level, such as level 7. By doing this, you still make commands available to level 15, but do not allow any user with

https://t.me/learningnets

the default privilege to use them. After that, you can assign the custom privilege level to a specific user, allowing the use of some privileged commands to this particular user only.

To understand the command authorization process, remember that at all times that the IOS exec shell works in one of the interpreter modes. The two most well-known modes are “exec mode” and “global configuration mode.” The interpreter’s mode is shown by default through the router’s prompt, such as `router#` (exec) or `router(config)#` (global configuration). In addition, the shell contains many other interpreter modes, such as “interface configuration,” “vpdn configuration,” “ip extended ACL,” “map-class,” and so on. Each mode has its own subset of commands, which are only visible in the particular mode. To see how IOS performs authorization, you should understand the generic command syntax as well. Each command generally has the following structure:

```
`command sub-command [arguments] [argument-values] [options]`
```

Here, command is the first sub-string you send to the interpreter; for example, `ip` in the `ip address` command under interface configuration mode. The sub-command field may be present in some commands, such as `ip proxy-arp` , etc. The arguments cover all named parameters that may take values and that are mandatory. For example, in the `ip address` command, the `address` field is an argument and may take a value such as `1.2.3.4` . The options may cover various command attributes that are not mandatory. From the local command authorization standpoint, you can only match the “non-variable” or “mandatory” fields such as “command,” “sub-command,” and “arguments.” The system will automatically allow any argument values and options if the command that the user enters matches the configured pattern. The syntax to re-assign a particular command is as follows:

```
`privilege <mode> level <level><command>`
```

This command instructs the shell to assign the command matching the string “command” to the level specified by the “level” argument. Note that the match is performed against all mandatory parts of the command that a user enters in a particular exec mode. For example, if you assigned just the command `snmp-server` to level 7 but not the command `snmp-server host` , a user will not be able to configure the SNMP traps destination, because “host” is a mandatory (non-optional) part of the command. The following features help you configure local command authorization:

- When you enter commands as a shortcut, such as `privilege exec level 7 conf t` , the

https://t.me/learningnets

shell expands it to the full names, such as to `privilege exec level 7 configure terminal` .

- When you allow a compound command to a particular level, such as `privilege interface level 7 ip address` , the shell automatically adds another line allowing the first “sub-component” of the compound command, such as a `privilege interface level 7 ip` command.
- You can use the special keyword “all” to allow any sub-command or argument behind a specific keyword. For example, the command `privilege interface all level 7 ip` allows all “ip” subcommands in the interface configuration mode.

The last thing to remember is how the local command authorization interoperates with the `show running-config` . A special feature is that users may only see the command they are authorized to use. For example, if a user is authorized to use the `interface` command and all “ip” subcommands, the `show running-config` command will only display those specific commands. Verification for this task consists of checking the commands that you are allowed to use.

```
R1#telnet 150.1.1.1
Trying 150.1.1.1 ... Open

User Access Verification

Username: ADMIN
Password:
R1#show privilege
Current privilege level is 7
!
R1#?
Exec commands:
  access-enable    Create a temporary Access-List entry
  access-profile   Apply user-profile to interface
  call             Voice call
  clear            Reset functionsconfigure        Enter configuration mode
<snip>
!R1#debug ?
all     Enable all debugging
conn    Connection Manager information
ip      IP information
mcsa    mcsa debugging
ncia    Native Client Interface Architecture (NCIA) events
ospfv3  OSPFv3 debug commands
!R1#debug ip rip
RIP protocol debugging is on
!R1#undebug all
```

https://t.me/learningnets

```
All possible debugging has been turned off
!R1#configure ?
  terminal  Configure from the terminal
<cr>
!R1#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#?
Configure commands:
beep       Configure BEEP (Blocks Extensible Exchange Protocol)
call       Configure Call parameters
cts        Cisco Trusted Security commands
default    Set a command to its defaults
end        Exit from configure mode
exit       Exit from configure mode
help       Description of the interactive help system interface  Select an interface to configure
license    Configure license features
netconf    Configure NETCONF
no         Negate a command or set its defaults
pfr        Performance Routing configuration submodes
sasl       Configure SASL
sgi        Configure SGI
wsma       Configure Web Services Management Agents
!R1(config)#interface GigabitEthernet1.13
R1(config-subif)#?
Interface configuration commands:
default   Set a command to its defaults
exit      Exit from interface configuration mode ip         Interface Internet Protocol config commands
no        Negate a command or set its defaults
ospfv3    OSPFv3 interface commands shutdown  Shutdown the selected interface
!R1(config-if)#ip ?

Interface IP configuration subcommands:
access-group            Specify access control for packets
address                 Set the IP address of an interface
authentication          authentication subcommands
bandwidth-percent       Set EIGRP bandwidth limit
bgp                     BGP interface commands
broadcast-address       Set the broadcast address of an interface
cef                     Cisco Express Forwarding interface commands
<snip>
```

Inspect the running configuration contents. Note that you only see the `interface` and `ip` commands in the running configuration:

https://t.me/learningnets

R1#show running-config

```
R1#show running-config
Building configuration...

Current configuration : 833 bytes
!
! Last configuration change at 07:35:48 UTC Wed May 14 2014 by ADMIN
!
boot-start-marker
boot-end-marker
!
!

interface Loopback0
 ip address 150.1.1.1 255.255.255.255
!
interface Tunnel0
 ip address 155.1.0.1 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip nhrp authentication NHRPPASS
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp holdtime 300
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
!
 interface GigabitEthernet1
  no ip address
!
 interface GigabitEthernet1.13
  ip address 155.1.13.1 255.255.255.0
!
 interface GigabitEthernet1.100
  ip address 169.254.100.1 255.255.255.0
!
 interface GigabitEthernet1.146
  ip address 155.1.146.1 255.255.255.0
!
interface GigabitEthernet2
 no ip address
 shutdown
!
interface GigabitEthernet3
```

https://t.me/learningnets

```
 no ip address
 shutdown
!
!
end
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Traffic Filtering Using Standard Access-Lists

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Using the minimum number of ACL lines, allow IPv4 access to R1 from the Loopback0 subnets of R2 through R6 that have an even third octet.
  - Deny access from any other 150.1.0.0/16 subnets.
  - Any hosts from the 155.1.0.0/16 subnets should still be able to reach R1.
  - Apply the access-list inbound on all interfaces of R1.

Configuration

https://t.me/learningnets

```
R1:

access-list 1 permit 150.1.0.0 0.0.6.255
access-list 1 permit 155.1.0.0 0.0.255.255
!
interface GigabitEthernet1.13
 ip access-group 1 in
!
interface GigabitEthernet1.100
 ip access-group 1 in
!
interface GigabitEthernet1.146
 ip access-group 1 in
!
interface Tunnel0
 ip access-group 1 in
```

Verification

Standard ACLs only allow you to match source IP addresses based on “base” IP address and wildcard mask. Because of that “aggregate” behavior, standard ACLs are commonly configured at network nodes close to the “protected” object. One very common task is finding a required base IP address and wildcard mask pair based on a set of requirements. Let us look at two examples to illustrate how they could be solved.

Example 1 (the task)

Configure the IP/Wildcard Mask pair to match IP addresses in subnets 150.1.Y.0/24, where Y is an even number and is greater than or equal to 1 and less than or equal to 6. Note that the fourth octet of the subnets may take any value from 0 to 255, so we just set the corresponding part of the wildcard mask to 255 (all 1s).

Step 1: We start by finding the fixed part of the subnets’ range. Because Y changes from 1 to 6, only the last 3 bits of the third octet may vary. Thus, the topmost 6 bits of Y are fixed at zero. For the fixed part of the base IP address, we set wildcard bits to zeroes (meaning it does not change). Now write down all possible values that Y may take:

https://t.me/learningnets

2=00000010
4=00000100
6=00000110

Step 2: Find the part of the “bit-vector” that changes between different values. In this case, this part contains bits 1 and 2 (counting from the right and starting at zero). Create a mask vector, putting 1s in the positions where bit values in data vectors may change:

m=00000110=6

Essentially, this is the wildcard mask for the third octet.

Step 3: Account for the extra values covered by the mask. Note that the mask always covers 2^N values, where *N* is the number of 1s in the mask. In our case, we had three values, but the mask contains two non-zero bits, resulting in four possible values. Effectively, our wildcard mask covers one extra bit vector:

0=00000000

Strictly speaking, you may need a special ACL entry to deny the corresponding subnet. However, in our topology, there is no router with subnet 150.1.0.0/24, so you may ignore this extra value.

Step 4: Compute the “base” third octet value, taking the common part of all the bit vectors. Effectively, you may do so by performing a logical bitwise AND over all values:

2 AND 4 AND 6 = 0

Thus, the resulting IP and Wildcard Mask pair is:

150.1.0.0 0.0.6.255

The resulting ACL may look like:

deny 150.1.0.0 0.0.0.255
permit 155=0.1.0.0.6.255

Note that the first line is NOT necessary in our case.

Example 2

https://t.me/learningnets

Configure the IP/Wildcard Mask pair to match IP addresses in subnets 150.1.Y.0/24, where *Y* is an odd number and is greater than or equal to 1 and less than or equal to 6. Note that the fourth octet of the subnets may take any value from 0 to 255, so we just set the corresponding part of the wildcard mask to 255 (all 1s).

Step 1: We start by finding the fixed part of the subnets' range. Because Y changes from 1 to 6, only the last 3 bits of the third octet may vary. Thus, the topmost 6 bits of Y are fixed to zero. For the fixed part of the base IP address, we set wildcard bits to zeroes (meaning it does not change). Now write down all possible values that Y may take:

1=00000001
3=00000011
5=00000101

Step 2: Find the part of the "bit-vector" that changes between the different values. In this case, this part contains bits 1 and 2 (counting from the right and starting at zero). The zero bit is fixed to 1. Create a mask vector, putting 1s in the positions where bit values in data vectors may change:

m=00000110=6

Essentially, this is the wildcard mask for the third octet.

Step 3: Take into account the extra values covered by the mask. Note that the mask always covers 2^N values, where *N* is the number of 1s in the mask. In our case, we had three values, but the mask contains two non-zero bits, resulting in four possible values. Effectively, our wildcard mask covers one extra bit vector:

7=00000111

Strictly speaking, we need a special ACL entry to deny the corresponding subnet, because it is a part of our topology.

Step 4: Compute the "base" third octet value using the common part of all the bit vectors. Effectively, you may do so by performing a logical bitwise AND over all the values:

1 AND 3 AND 5 = 1

Thus the resulting IP and wildcard mask pair is:

150.1.1.0 0.0.6.255

And the resulting ACL looks like:

https://t.me/learningnets

deny 150.1.7.0 0.0.0.255
permit 150.1.1.0 0.0.6.255

In cases where octet values are widely separated, the resulting ACL may contain too many “deny” statements, and it may be more effective to use just the entries matching specific values. In our case, the resulting ACL needs to permit the 155.1.0.0/16 subnet as well. Note that standard ACLs do not allow you to specify the protocol values and effectively permit any IP protocol for matching sources. The general use of standard ACLs is to route filtering and prefix matching in routing protocol configuration. Note that standard ACLs may be named as well as numbered (1–99).

To verify your configuration, first make sure that the IP access list is actively applied to all interfaces:

```
R1#show ip interface GigabitEthernet1.146 | i access

 Outgoing Common access list is not set
 Outgoing access list is not set
 Inbound Common access list is not set  Inbound  access list is 1
 IP access violation accounting is disabled
!R1#show ip interface GigabitEthernet1.100 | i access

 Outgoing Common access list is not set
 Outgoing access list is not set
 Inbound Common access list is not set  Inbound  access list is 1
 IP access violation accounting is disabled
!R1#show ip interface GigabitEthernet1.13 | i access

 Outgoing Common access list is not set
 Outgoing access list is not set
 Inbound Common access list is not set  Inbound  access list is 1
 IP access violation accounting is disabled
!R1#show ip interface Tunnel0 | inc access

 Outgoing Common access list is not set
 Outgoing access list is not set
 Inbound Common access list is not set  Inbound  access list is 1

 IP access violation accounting is disabled
```

Verify that you can only reach R1 from the Loopbacks of R2, R4, and R6:

```
R2#ping 150.1.1.1 source loopback0
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.2.2 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 84/86/88 ms
!R3#ping 155.1.1.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.3.3 U.U.U
Success rate is 0 percent (0/5)
!R4#ping 150.1.1.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.4.4 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
!R5#ping 150.1.1.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.5.5 U.U.U
Success rate is 0 percent (0/5)
R6#ping 150.1.1.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.6.6 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
!R7#ping 150.1.1.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.7.7 U.U.U
Success rate is 0 percent (0/5)
!R8#ping 150.1.1.1 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.8.8 U.U.U

Success rate is 0 percent (0/5)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Traffic Filtering Using Extended Access-Lists

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure IPv4 filtering on R4's VLAN 45 and DMVPN links to achieve the following requirements.
- R4 allows any TCP traffic towards R5 if sourced from the Loopback0 subnets. Ensure that R4 allows returning TCP packets for those connections.
- VLAN 146 contains servers running FTP, WWW, and SMTP. R4 allows access to these applications from behind R5.
  - Ensure that you only permit active FTP responses from servers in VLAN 146.
- Permit the exchange of EIGRP routing updates.
- Allow IOS traceroute and ping commands to function across R4.
- Ensure that the Path MTU discovery procedure works successfully across your firewall.
- Deny and log all other traffic.

Configuration

R4:

```
ip access-list extended INBOUND
remark ==
remark == Active FTP uses TCP ports 20 and 21
remark ==
permit tcp any 155.1.146.0 0.0.0.255 range 20 21
remark ==
```

https://t.me/learningnets

```
remark == WWW and SMTP use port numbers 80 and 25
remark ==
permit tcp any 155.1.146.0 0.0.0.255 eq 80
permit tcp any 155.1.146.0 0.0.0.255 eq 25
remark ==
remark == Established TCP sessions are traffic returning back
remark ==
permit tcp any 150.1.0.0 0.0.255.255 established
remark ==
remark == Inbound traceroute UDP probes
remark ==
permit udp any any range 33434 33474
remark ==
remark == Backscatter ICMP traffic for outbound traceroute UDP probes
remark ==
permit icmp any any port-unreachable
permit icmp any any time-exceeded
remark ==
remark == ICMP message used by PMTU discovery: Packet too big and DF bit set
remark ==
permit icmp any any packet-too-big
remark ==
remark == Ping operation packets
remark ==
permit icmp any any echo
permit icmp any any echo-reply
remark ==
remark == EIGRP routing updates
remark ==
permit eigrp any any
remark ==
remark == Deny and log any other packets
remark ==
deny ip any any log
!
ip access-list extended OUTBOUND
remark ==
remark == Active FTP uses TCP ports 20 and 21
remark ==
permit tcp 155.1.146.0 0.0.0.255 range 20 21 any
remark ==
remark == WWW and SMTP use port numbers 80 and 25
remark ==
permit tcp 155.1.146.0 0.0.0.255 eq 80 any
permit tcp 155.1.146.0 0.0.0.255 eq 25 any
remark ==
```

https://t.me/learningnets

```
remark == TCP traffic from Loopback0 subnets
remark ==
permit tcp 150.1.0.0 0.0.255.255 any
remark ==
remark == Outbound traceroute UDP probes
remark ==
permit udp any any range 33434 33474
remark ==
remark == Backscatter ICMP traffic for inbound traceroute UDP probes
remark ==
permit icmp any any port-unreachable
permit icmp any any time-exceeded
remark ==
remark == ICMP message used by PMTU discovery: Packet too big and DF bit set
remark ==
permit icmp any any packet-too-big
remark ==
remark == Ping operation packets
remark ==
permit icmp any any echo
permit icmp any any echo-reply
remark ==
remark == Deny and log any other packets
remark ==
deny ip any any log
!
interface Tunnel0
 ip access-group INBOUND in
 ip access-group OUTBOUND out
!
interface GigabitEthernet1.45
 ip access-group INBOUND in
 ip access-group OUTBOUND out
```

Verification

Extended ACLs allow the matching of source/destination IP addresses and source/destination ports. You can also match ICMP message types, DSCP values, port ranges, TCP flags, and so on. Essentially, extended ACLs permit configuration of a very flexible stateless packet filter. Because of the stateless nature of the filtering, for each “outbound” rule you need a matching “inbound” rule, permitting the returning traffic. Thus, inbound and outbound extended access-lists usually mirror each other (for example, port numbers swapped between source and destination IP

https://t.me/learningnets

addresses). Extended ACLs allow limited emulation of stateful behavior for TCP connections. Specifically, by using the `established` keyword, you permit all TCP packets that have the “ACK” bit set. This does not include the initial “SYN-only” packet, and thus permits only the packets that are part of an established session.

To configure an extended ACL successfully, you need to know the details of application networking logic. At the very least, this includes the TCP/UDP port numbers used by the application. You may also need to know some additional details, such as how active FTP differs from passive FTP, how the traceroute utility works, and so on. Most of the common ports can be found by using the shell prompt and typing the question mark after `permit tcp any any eq .` You may also learn application ports by using the commands `show ip port-map` or `show ip nbar port-map .`

Another special feature is functionality of outbound access-lists. The router’s locally generated traffic is not subject to match against the outbound access-lists; only transit traffic matches outbound lists. Next we will discuss two applications commonly used in practice. The first one is FTP. According to the design, FTP uses two TCP connections, a control connection to send commands, and a data connection to transfer files. The control connection always uses a destination port 21. The data connection may work in one of two modes: active or passive.

In active mode, when a client issues a command requiring data transfer, the server instructs the client to listen on an ephemeral port (above 1024) and initiates a connection to the client sourced from port number 20. In this mode, the server connects to a client on a dynamic destination port. In passive mode, when the client needs to transfer a file, the server opens a new ephemeral port (above 1024) and reports it to the client. The client then connects to the ephemeral port and the data transfer begins. In this mode, the client connects to the server on a dynamic destination port. Note that port 20 is *not used* in passive mode.

https://t.me/learningnets

The next application to discuss is the traceroute utility. Different variants of the traceroute are utilized by various operating systems. IOS uses the so-called UNIX variant. In this variant, the client sends UDP probes with increasing TTL values starting at the TTL of 1. The client sends probes to incrementing UDP ports starting at 33434 for the first hop. The destination IP address in all packets is the target IP address. If the next hop is not the ultimate destination, the receiving device checks the TTL field. If the TTL field expires, the device sends an ICMP time exceeded message sourced from its own IP address. The source node learns the hop number based on the UDP port number, encapsulated as part of the payload in the ICMP response message. If the hop is the ultimate destination, it sends an ICMP port unreachable message for the port range selected specifically not to match any application. The source node learns the hop count based on the port number. By default, the traceroute utility only probes up to 30 hops, so the default UDP port range is 33434…33464.

Finally, the Path MTU Discovery (PMTUD) process relies on the ICMP message "packet too big," also known as "fragmentation required but DF bit set." Note that you can permit all ICMP unreachable messages by using the **`unreachable`** keyword, or permit any ICMP message selectively.

During the creation of access-lists, use remarks extensively in real-life configurations to make management and readability of function easier. In the lab exam, first try typing access-lists in Notepad; it is easier to spot mistakes that way. In the lab, it's also a good idea to add the additional **`deny ip any any log`** at the end of your access-list to catch any unnoticed traffic that you may have not permitted. In this verification, we will configure R6 to simulate some of the services, such as HTTP.

```
R6:

ip http server
```

Verify few of the services allowed through R4 from behind R5, like ping, traceroute and http:

```
R5#ping 155.1.146.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/17/20 ms
!
!R5#traceroute 155.1.146.6
Type escape sequence to abort.
Tracing the route to 155.1.146.6
```

https://t.me/learningnets

```
VRF info: (vrf in name/id, vrf out name/id)1 155.1.45.4 3 msec 2 msec 0 msec

2 155.1.146.6 2 msec *  4 msec
!
!R5#telnet 155.1.146.6 80
Trying 155.1.146.6, 80 ... Open
```

Verify that you can originate TCP connections only from a Loopback0 interface, but not from the 155.1.0.0/16 subnets. After this verification, ensure that traceroute and ping work from behind R4:

```
R6#telnet 150.1.5.5
Trying 150.1.5.5 ... % Destination unreachable; gateway or host down
!
!R6#telnet 150.1.5.5 /source loopback0
Trying 150.1.5.5 ... Open
R5#
!
!R6#ping 150.1.5.5

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.5.5, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/38/40 ms
!
!R6#traceroute 150.1.5.5

Type escape sequence to abort.
Tracing the route to 150.1.5.5

  1 155.1.146.4 0 msec 4 msec 0 msec 2 155.1.0.5 16 msec *  16 msec
```

Be sure to check your access-list match counters:

```
R4#show access-lists
Extended IP access list INBOUND
    10 permit tcp any 155.1.146.0 0.0.0.255 range ftp-data ftp
    20 permit tcp any 155.1.146.0 0.0.0.255 eq www (8 matches)
    30 permit tcp any 155.1.146.0 0.0.0.255 eq smtp
    40 permit tcp any 150.1.0.0 0.0.255.255 established (15 matches)
    50 permit udp any any range 33434 33474 (3 matches)
    60 permit icmp any any port-unreachable
    70 permit icmp any any time-exceeded
    80 permit icmp any any packet-too-big
```

https://t.me/learningnets

```
    90 permit icmp any any echo (5 matches)
    100 permit icmp any any echo-reply
    110 permit eigrp any any (2000 matches)
    120 deny ip any any log (24 matches)Extended IP access list OUTBOUND

    10 permit tcp 155.1.146.0 0.0.0.255 range ftp-data ftp any
    20 permit tcp 155.1.146.0 0.0.0.255 eq www any (7 matches)
    30 permit tcp 155.1.146.0 0.0.0.255 eq smtp any
    40 permit tcp 150.1.0.0 0.0.255.255 any (15 matches)
    50 permit udp any any range 33434 33474
    60 permit icmp any any port-unreachable (2 matches)
    70 permit icmp any any time-exceeded
    80 permit icmp any any packet-too-big
    90 permit icmp any any echo
    100 permit icmp any any echo-reply (5 matches)
    110 deny ip any any log (1 match)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Filtering Fragmented Packets

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Ensure that R3 prevents fragmented packets from reaching its local HTTP server.
  - Apply the filtering only on its Tunnel0 interface.

Configuration

```
R3:

ip access-list extended NO_FRAGMENTS
 permit eigrp any any
 deny ip any any fragments
 permit tcp any any eq 80
!
interface Tunnel0
 ip access-group NO_FRAGMENTS in
```

Verification

By design, the IP protocol allows packet fragmenting. This feature has been a known weakness for along time, exploited by various types of attacks (such as ping of death). Overlapping fragments, fragments exceeding the assembly buffer, and fragments arriving out of order are just a few examples of traffic that can severely degrade end system performance when an attacker sends them at high rates.

https://t.me/learningnets

Additionally, fragmented packets are often used to bypass IDS or firewall systems. The attacker splits a packet in such a way that the firewall or IDS system is not able to extract information about something like the port numbers. Advanced firewalls and IDS systems support packet stream reassembly, but this procedure may degrade overall security system performance.

Because of these reasons, it is a good practice to avoid traffic fragmentation in your network and protect your servers against fragmented packets. One solution would be to configure matching MTU values and enable the PMTU Discovery process. The IOS firewall can classify IP packets as one of the following:

- Non-fragmented packets or initial fragments. These packets have a fragment offset of zero and commonly contain some upper-level protocol payload (TCP) to allow the extraction of information about application ports.
- Non-initial fragments. These have a non-zero fragment offset and are the remaining parts of a fragmented packets. These packets do not have upper-level protocol port information, and the IOS firewall cannot match them against ACL entries configured with TCP/UDP port numbers. Instead of matching the port numbers, the firewall uses only Layer 3 information in an ACL entry (such as source/destination IP addresses) to match against the source/destination IP addresses in the packet.

This is why any non-initial fragment sourced from 1.1.1.1 to 2.2.2.2 matches the following ACL entry:

```
`permit tcp host 1.1.1.1 host 2.2.2.2 eq 80`
```

Even though it does not contain any port information, the firewall only matches the source and destination IP addresses for non-initial fragments. You can match non-initial fragments using the **`fragments`** keyword in your ACL entry. For example:

```
`deny ip any host 2.2.2.2 fragments`
`permit tcp host 1.1.1.1 host 2.2.2.2 eq 80`
```

This configuration ensures that only non-fragmented packets and initial fragments may reach port 80 of the target system. You can configure R5 so that outgoing TCP packets are fragmented. To accomplish this, ensure that PMTU Discovery is disabled on R8 and the MTU is set to a minimum value on the tunnel interface of R5. This will force R5 to fragment TCP/IP packets exceeding 68 bytes in size. You may need to remove the inbound ACL on R5 to permit returning traffic for the HTTP connection. After this configuration, try connecting to R3 on port 80 from R8 and download a file from the flash memory. The IP+TCP header size is 40 bytes and the

https://t.me/learningnets

payload size is most likely more than 28 bytes (the GET command and the URL), so the packets will surely exceed the IP MTU:

```
R3:
ip http server
ip http path flash:
ip http authentication local
username admin privilege 15 password 0 cisco
copy running-config flash:fragment.txt
!R8(config)#no ip tcp path-mtu-discovery

!R5(config)#interface Tunnel0
R5(config-if)#ip mtu 200
!R8#copy http://admin:cisco@155.1.0.3/fragment.txt null:
Accessing http://*****:*****@155.1.0.3/fragment.txt...
%Error opening http://*****:*****@155.1.0.3/fragment.txt (I/O error)
```

The connection times out, and by looking at the access-list statistics on R3 you can see exactly why. The reason is the packets are denied.

```
R3#show access-list
Extended IP access list NO_FRAGMENTS
    10 permit eigrp any any (16 matches) 20 deny ip any any fragments (4 matches)

    30 permit tcp any any eq www (3 matches)
```

Now change the MTU back to normal and try downloading again. Now R5 does not fragment the packets and the connection completes normally:

```
R5(config)#interface Tunnel0
R5(config-if)#no ip mtu 200
!R8#copy http://admin:cisco@155.1.0.3/fragment.txt null:
Accessing http://*****:*****@155.1.0.3/fragment.txt...
Loading http://***********@155.1.0.3/fragment.txt !2649 bytes copied in 0.040 secs (66225 bytes/sec)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Filtering Traffic with Time-Based Access Lists

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Restrict the users behind R8 from accessing R10 on weekends.
  - Additionally, restrict access to R10 from 6pm to 9am on the remaining days.
- Apply a single access-list for this.

Configuration

https://t.me/learningnets

```
R8:

time-range WEEKDAYS_EVES
 periodic weekdays 18:00 to 23:59
 periodic weekdays 0:00 to 8:59
!
time-range WEEKENDS
 periodic weekend 0:00 to 23:59
!
ip access-list extended OUTBOUND
 deny ip any any time-range WEEKDAYS_EVES
 deny ip any any time-range WEEKENDS
 permit ip any any
!
interface GigabitEthernet1.108
 ip access-group OUTBOUND out
```

Verification

Time-based ACLs allow the use of time ranges, bound to a specific entry. This allows the entry to be active only during the time-range specified. There are two types of time-ranges: absolute and periodic. Absolute time-ranges define non-repeating time intervals (such as from Jan 10 to Jan 12). Periodic time-ranges define recurring time-intervals that may repeat infinitely (such as every Wednesday from 6am to 11am). The latter type is based on the concept of weekdays;for example, you may define periods based on days of week (Mon, Tue, Fri, etc.) and specific ranges such as daily, weekends, weekdays (Mon–Fri). Absolute intervals define starting dates/times and ending dates/times.

It is important to note that time-ranges start with the first second of the first minute in the range and end with the last second of the last minute in the range. For example, the interval from 00:01 to 02:00 will last from 00:01:00 to 02:00:59 (hh:mm:ss). Therefore, if you want some interval to end at exactly 4:00pm, use the value “15:59” (note that IOS uses the 24-hour time format). The exact boundaries of time ranges may vary based on implementation.

In our scenario, we define two time-ranges. The first one covers weekends, and the second one covers the time interval from 6pm to 9am every weekday. As usual, remember that local traffic is not subject to an outbound ACL check. Therefore, try connecting from R3. At first, the connection fails because one of the time-ranges is active.

https://t.me/learningnets

```
R5#telnet 150.1.10.10

Trying 150.1.10.10 ... % Destination unreachable; gateway or host down
!R8#show clock
 07:22:17.456 UTC Sun May 18 2014
!R8#show access-lists

Extended IP access list OUTBOUND 10 deny ip any any time-range WEEKDAYS_EVES (inactive)
 20 deny ip any any time-range WEEKENDS (active) (1 match)

 30 permit ip any any
```

Change the time on R8 to during “business hours,” and try connecting again:

```
R8#show clock
14:00:59.761 UTC Mon May 19 2014
!R5#telnet 150.1.10.10

 Trying 150.1.10.10 ... Open
 R10#
```

The connection is now successful. Check the access-list counters again:

```
R8#show access-lists

Extended IP access list OUTBOUND 10 deny ip any any time-range WEEKDAYS_EVES (inactive) (5 matches)
 20 deny ip any any time-range WEEKENDS (inactive) (12 matches)
 30 permit ip any any (16 matches)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Traffic Filtering with Policy-Based Routing

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R6 to drop ICMP packets of 100 bytes in size entering the VLAN 146 interface.
- The router should not send ICMP unreachable notifications about dropped packets.

Configuration

R6:

```
ip access-list extended ICMP
 permit icmp any any
!
route-map DROP
 match ip address ICMP
 match length 100 100
 set interface Null0
!
interface GigabitEthernet1.146
 ip policy route-map DROP
!
interface Null0
 no ip unreachables
```

https://t.me/learningnets

Verification

Policy-Based Routing allows the implementation of select packet filtering based on various criteria. Among the most important are those based on:

- Any access-list (standard/extended)
- Packet size
- The packet ToS byte
- The output interface (allows sub-interface granularity)

The packet drop behavior with PBR is achieved byusing **`set interface Null0`** command. Based on the configuration of the Null0 interface, the router may generate an ICMP unreachable message. You can disable this feature to conserve router resources by using the command **`no ip unreachables`** under the Null0 interface. However, the router only sends unreachables for process-switched packets. For verification, send ICMP packets of different sizes from R7 to R4, or any ICMP packets that transit R6.

```
R7#ping 150.1.4.4 size 200
Type escape sequence to abort.
Sending 5, 200-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
!
!R7#ping 150.1.4.4 size 100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:.....

Success rate is 0 percent (0/5)
```

Verify that dropped packets matched the route-map:

```
R6#show route-map
route-map DROP, permit, sequence 10
  Match clauses:
    ip address (access-lists): ICMP
    length 100 100
  Set clauses:
    interface Null0 Policy routing matches: 5 packets, 590 bytes
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Preventing Packet Spoofing with uRPF

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Consider that R4's VLAN 146 link is a ISP connection providing Internet services.
  - Ensure that R4 does not accept packets with IP addresses of the internal subnets on its connection to the ISP.
  - Considering there is another connection to the same ISP in the network, account for possible asymmetric routing issues on R4.
- At the same time, R4 should only accept packets from legitimate subnets of 150.1.0.0/16 and 155.1.0.0/16 learned via IGP on its other links.
- Log all packets with spoofed sources.

Configuration

R4:

```
access-list 100 deny ip any any log
!
interface GigabitEthernet1.146
 ip verify unicast source reachable-via any 100
!
interface GigabitEthernet1.45
 ip verify unicast source reachable-via rx 100
!
interface GigabitEthernet1.100
 ip verify unicast source reachable-via rx 100
```

https://t.me/learningnets

```
!
interface Tunnel0
 ip verify unicast source reachable-via rx 100
```

Verification

uRPF, or Unicast Reverse Path Forwarding, is the concept of verifying the routing path for the source IP address found in an IP packet. Usually routers only use destination IP addresses to look up the next hop for an IP packet. Reverse-Path Forwarding, however, is used with multicast routing. From a security perspective, uRPF is useful for checking IP address spoofing conditions. Generally, packets arrive on the interfaces that are on the shortest path to the source of the packets. This is the natural result of the IGP routing protocol at work on the device. However, with IP spoofing attacks, a malicious user may inject packets with IP addresses not belonging to its segment or network area. From the router perspective, such packets may appear on the interfaces not on the shortest path to their source. You should note that packets may have spoofed sources and appear on the shortest-path interface in cases when you have just one connection to all destinations.

Based on its function, uRPF could be a very useful security feature to prevent spoofing attacks, especially on routers that have a diverse number of connections, such as a system at an Internet peering point. The feature has two general modes of operation:

- **Strict mode**. When you enable uRPF on an interface with the `ip verify unicast source reachable-via rx` command (or the legacy format `ip verify unicast reverse-path`), the router applies the uRPF check to the source IP addresses of incoming packets. The source IP address must match an explicit IP route in the routing table, and the next hop for this entry should point out of the interface from which the packet was received. Otherwise, the router will drop the packets.
- **Loose mode**. You enable the loose mode with the command `ip verify unicast source reachable-via any`. When a router receives a packet on the interface with loose uRPF enabled, it just checks that it has an IP route matching the source address in the packet. It does not matter whether the next hop for this route points out the receiving interface. The exception to this checking is that if the route is to Null0, the packet is dropped. This feature is useful in enterprises that have more than one ISP uplink and use asymmetric routing. With asymmetric routing, packets may take paths not allowed with strict uRPF.

uRPF does have additional features. The first one is uRPF exemptions and violation logging. With this feature, you may specify a standard or extended access-list as follows: `ip verify unicast source reachable-via [rx|any] <ACL-NUM>` The uRPF feature

https://t.me/learningnets

consults this access-list for packets *violating* the uRPF condition. If the ACL permits a packet, it is allowed to pass through. If the ACL denies the packet, the router drops it. You may use the `log` keyword to log the packets allowed or denied by the uRPF access-list.

The other two features are:

- Allowing the router to do a self-ping with strict uRPF enabled on the interfaces (that is, accepting packets sourced from the router on the interface from which they were sent).
- Allowing the use of a default route with strict uRPF. In this case, the router will match source IP addresses against the default-route entry (0.0.0.0/0) as well. This is useful when you want to accept all packets on your Internet connections, but feature protection from spoofed packets coming from your internal network.

To verify the feature, first make sure it has been applied to all interfaces in the correct functional mode:

```
R4#show ip interface GigabitEthernet1.146 | i verify
   IP verify source reachable-via ANY
, ACL 100
!R4#show ip interface GigabitEthernet1.45 | i verify
 IP verify source reachable-via RX
, ACL 100
!R4#show ip interface GigabitEthernet1.100 | i verify
 IP verify source reachable-via RX
, ACL 100
!R4#show ip interface Tunnel0 | i verify
             IP verify source reachable-via RX
, ACL 100
```

Generate some spoofed traffic from R5 (the inside network) across R4. Create a special Loopback interface on R5 to accomplish this:

```
R5:
interface Loopback1
 ip address 150.2.55.55 255.255.255.0
R4:


ip access-list log-update threshold 1
```

Observe the logging and statistics on R4:

https://t.me/learningnets

```
R5#ping 150.1.4.4 source loopback1 repeat 10

Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.2.55.55 ..........
Success rate is 0 percent (0/10)
!
!R4#show access-lists
Extended IP access list 100    10 deny ip any any log (10 matches)
!
!R4#show ip interface gigabitEthernet1.45 | i ver
  ICMP mask replies are never sent
  Router Discovery is disabled
  IP verify source reachable-via RX, ACL 100 10 verification drops
   0 suppressed verification drops
   0 verification drop-rate

R4:

%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list 100 denied icmp 150.2.55.55 -> 150.1.4.4 (8/0), 1 packet
```

Generate some spoofed traffic from R1 (the outside/ISP network) across R4. Create a special Loopback interface on R1 to accomplish this. In order to simulate assymetric traffic, configure R4 with a static route for R1's Loopback1 through VLAN 100 connection:

```
R1:
interface Loopback1
 ip address 150.2.11.11 255.255.255.0
R4:

ip route 150.2.11.11 255.255.255.255 169.254.100.1
```

If uRPF is correctly configured in loose mode, we should have IP connectivity between R1's Loopback1 and R4's Loopback0, also we should see suppressed

https://t.me/learningnets

verifications drops on R4's VLAN 146 interface for the traffic:

```
R1#ping 150.1.4.4 source loopback1 repeat 10
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.2.11.11 !!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 1/1/6 ms
!
!R4#show ip interface gigabitEthernet1.146 | i ver
  ICMP mask replies are never sent
  Router Discovery is disabled
  IP verify source reachable-via ANY, ACL 100
   0 verification drops 17 suppressed verification drops

   0 verification drop-rate
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Using NBAR for Content-Based Filtering (pending update)

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R6 to identify any HTTP downloads requested by users on and behind VLAN 146.
  - The files have extensions “.exe,” “.com,” and “.bin.”
  - Use a single pattern to match all the filenames at once.

Configuration

R6:

```
class-map match-all EXTENSION
 match protocol http url “*.bin|*.exe|*.com”
!
!
policy-map HTTP
 class EXTENSION
!
interface GigabitEthernet1.146
  service-policy output HTTP
```

Verification

https://t.me/learningnets

The NBAR (Network Based Application Recognition) protocol classification mechanism allows deep inspection for some protocol types. One notable feature is matching various fields inside HTTP headers. In recent IOS versions, this feature evolved into the complete Flexible Packet Matching (FPM) feature, which allows for comprehensive inspection of any IP packets.

Commonly you see HTTP inspection used for matching the URL field against certain patterns. The URL matching occurs for HTTP GET/PUT/POST requests. It is important to note that the NBAR engine classifies bi-directional traffic flows. That is, if you apply a policy map matching the URL pattern inbound on an interface, the policy map will classify both incoming HTTP requests as well as packets returning in server replies. This is because NBAR classification occurs independently of policy-map direction, and the classification decision applies to both parts of a bi-directional traffic flow.

The command to match a URL is: **`match http protocol url <pattern>`** . The URL matching feature uses special wildcard symbols:

“*” – matches any sequence of characters (non-empty)
“?”– matches any single character (you need to press Ctrl-V to enter “?”)
“|” – alternative, logical OR
“[]” – range (ex: [ab] matches either “a” or “b”)
“()” – grouping; delimit the logical end of a pattern; for example, you can use “*.(exe|bin)” as equivalent to “*.exe|*.bin”

All matching is case insensitive. The pattern “text” matches “TEXT” as well. The engine matches your URL pattern against the directory path and the file name in the URL. For example, if the URL string is “http://www.cisco.com/pub/uploads/image.jpeg”, the matching procedure will only use the “pub/uploads/image.jpeg” part of the URL. When you submit a request like the above URL, it translates into the following headers (there are actually more, but this is the bare minimum):

```
GET /pub/uploads/image.jpeg HTTP/1.1
Host: www.cisco.com
```

Thus, if you need to match a host name, you should use the **`match protocol http host`** statement. You can use the same wildcard characters to match the HTTP Host header. We will use the following single-line expression for our task:

```
“\*.bin|\*.exe|\*.com”
```

https://t.me/learningnets

This pattern matches files with extension “*.com,” “*.exe,” or “*.bin.” You can also use the “match-any” class map and match multiple URL strings on separate lines, resulting in the same effect as separating a pattern with the symbol “|”. The second interesting part of using this feature is the MQC **`drop`** operation. Under any MQC class, you can apply this operation to drop the matching packets. For verification of our task, simulate an HTTP server with R7 and create some files in the flash with extensions “exe,” “com,” and “bin.”:

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R7(config)#username admin privilege 15 password cisco
R7(config)#ip http authentication local
R7(config)#ip http server
R7(config)#ip http path flash:
!R7#copy running-config flash:config.exe
Destination filename [config.exe]?
1971 bytes copied in 0.162 secs (12167 bytes/sec)
!R7#copy running-config flash:config.com
Destination filename [config.com]?
1971 bytes copied in 0.100 secs (19710 bytes/sec)
!R7#copy running-config flash:config.bin

Destination filename [config.bin]?
1971 bytes copied in 0.082 secs (24037 bytes/sec)
```

Verify the policy-map counters before starting any downloads for the created files:

```
R6#show policy-map interface gigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy output: HTTP
 Class-map: EXTENSION (match-all)
 0 packets, 0 bytes

      5 minute offered rate 0000 bps
      Match: protocol http url "*.bin|*.exe|*.com"

    Class-map: class-default (match-any)
      2 packets, 176 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

Try downloading those files from R1 across R6 and verify the policy-map counters

https://t.me/learningnets

after each download:

```
R1#copy http://admin:cisco@150.1.7.7/config.exe null:
Accessing http://*****:*****@150.1.7.7/config.exe...
Loading http://***********@150.1.7.7/config.exe !1971 bytes copied in 0.052 secs (37904 bytes/sec)
!R6#show policy-map interface gigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy output: HTTP

    Class-map: EXTENSION (match-all)  18 packets, 5428 bytes
      5 minute offered rate 0000 bps
      Match: protocol http url "*.bin|*.exe|*.com"

    Class-map: class-default (match-any)
      52 packets, 4568 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
!R1#copy http://admin:cisco@150.1.7.7/config.com null:
Accessing http://*****:*****@150.1.7.7/config.com...
Loading http://***********@150.1.7.7/config.com !1971 bytes copied in 0.033 secs (59727 bytes/sec)
!R6#show policy-map interface gigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy output: HTTP

    Class-map: EXTENSION (match-all)  36 packets, 10856 bytes
      5 minute offered rate 1000 bps
      Match: protocol http url "*.bin|*.exe|*.com"

    Class-map: class-default (match-any)
      62 packets, 5396 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
!R1#copy http://admin:cisco@150.1.7.7/config.bin null:
Accessing http://*****:*****@150.1.7.7/config.bin...
Loading http://***********@150.1.7.7/config.bin !1971 bytes copied in 0.037 secs (53270 bytes/sec)
!R6#show policy-map interface gigabitEthernet1.146
 GigabitEthernet1.146

  Service-policy output: HTTP

    Class-map: EXTENSION (match-all)  55 packets, 16342 bytes

      5 minute offered rate 0000 bps
      Match: protocol http url "*.bin|*.exe|*.com"
```

https://t.me/learningnets

```
    Class-map: class-default (match-any)
      78 packets, 6752 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Packet Logging with Access-Lists

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R6 to log all ICMP packets entering its VLAN 146 interface.
  - Aggregate logging messages so that a log entry is generated after five access-list entry hits.
  - Reduce the burden on the router CPU by process switching only one packet per second.

Configuration

R6:

```
ip access-list log-update threshold 5
ip access-list logging interval 1000
!
ip access-list extended LOGGING
 permit icmp any any
 permit ip any any
!
interface GigabitEthernet1.146
 ip access-group LOGGING in
```

https://t.me/learningnets

Verification

You can configure an access-list entry with the `log` keyword, making it log the matching packets via syslog. Using the `log-input` keyword, you can also log the input interface and the source MAC address. There are two configuration “knobs” to be used with access-list based logging. The first one is the logging update-threshold. It specifies how many times a packet must hit the access-list entry to generate a logging message. By default, this value is zero, which means the router would generate an entry based on a periodic timeout of 5 minutes. This command serves the purpose of aggregating the hits on the entry and the format is:

```
`ip access-list log-update threshold <HIT-COUNT>`
```

When this value is non-zero, the router generates a log entry for every number of hits and also produces the periodic 5 minute logging message. The second knob relates to the behavior of packet logging. You should remember that every logged packet is process-switched, so a large packet flow may easily consume all your CPU resources. For this reason, you may want to rate-limit the amount of process-switched packets using the command:

```
`ip access-list logging interval <TIME-INTERVAL>`
```

This command allows only one packet per interval to be process-switched. All exceeding packets are not process-switched and thus are not accounted for logging purposes. By default, this feature is off and all packets are process-switched. Note that this interval does not apply to packets destined to the router itself because these are process-switched by default. Using this command limits the effect of packet logging on the CPU, but it may result in an unpredictable number of packets being logged. In reality, it is useful when you just need a general hint of packet matches, not the detailed statistics. You may also limit the amount of all syslog messages (including the ACL logging messages) using the generic syslog rate-limiting feature.

To verify, try sending a barrage of ping packets from R4 to R7 across R6.

```
R4#ping 150.1.7.7 repeat 1000

Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

https://t.me/learningnets

```
<snip>

R6
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 1 packet
```

Change the logging interval and set the log-update threshold to 1 packet:

```
R6:
ip access-list logging interval 1
ip access-list log-update threshold 1

R4#ping 150.1.7.7 repeat 1000

Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
<snip>

R6
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 2 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 75 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 77 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 48 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 57 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 73 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 80 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 79 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 74 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 83 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 70 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 72 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 82 packets
%FMANFP-6-IPACCESSLOGDP: F0: fman_fp_image:  list LOGGING permitted icmp 155.1.146.4 -> 150.1.7.7 (8/0), 89 packets
```

To get the true count of aggregate packet matches, disable the logging interval.
Now you see 1 logging entry with 50 packet counts:

```
R6:
no ip access-list logging interval

R4#ping  150.1.7.7 repeat 50
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 50, 1500-byte ICMP Echos to 150.1.7.7, timeout is 2 seconds:
!!!
```

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

VLAN Filtering for IP Traffic

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW2 and SW3 with IP addresses on VLAN 10 in the format of **160.1.10.Y** , where **Y** is the switch number.
- Configure SW1 to restrict traffic on VLAN 10 so that only TCP traffic is allowed.
  - Disallow any other traffic to cross VLAN 10.

Configuration

https://t.me/learningnets

```
SW1:

ip access-list extended ALLOWED_L3_TRAFFIC
 permit tcp any any
!
vlan access-map VLAN10_FILTER 10
 match ip address ALLOWED_L3_TRAFFIC
 action forward
!
vlan filter VLAN10_FILTER vlan-list 10
SW2:

interface Vlan10
 ip address 160.1.10.2 255.255.255.0
SW3:

interface Vlan10
 ip address 160.1.10.3 255.255.255.0
```

Verification

VLAN filters on the Catalyst switches apply to any traffic ingress on the VLAN they attach to. The concept of a VLAN filter is an extension of access-lists to a whole bridged domain. It is important to remember the following key features of VLAN filters:

- VLAN filters are ingress and therefore conflict with any other ingress filtering features, such as port access-lists, either IP or MAC-based, and ingress SVI access-lists. You can still use outbound IP access-lists on SVIs.
- VLAN filters are organized as a sequence of entries. Each entry matches either an IP or MAC access-list and you can specify an action, which is either forward or drop. Optionally, you can omit matching any access-list, and then the entry would matches all IP and non-IP traffic. If a packet does not match the entry, the next vlan-filter entry in sequence is attempted. Note that if a packet is denied in the access-list, it does not automatically mean it is going to be dropped. It just does not match the particular VLAN filter entry.
- VLAN filters distinguish between IP and non-IP traffic based on the access-list matched. By default, a VLAN filter permits both types of traffic, until you match an access-list in a vlan-filter entry. If there is a particular type of access-list in a vlan-

https://t.me/learningnets

filter (such as IP access-list), all non-matching traffic of this type is dropped by the filter. The other type of traffic (non-IP) can still pass the filter if there are no explicit access-lists of this type.

- VLAN filters apply to both local traffic (inside the VLAN) and transit traffic. Also remember to re-apply your vlan-filter after you change its configuration,because those filters are programmed in hardware.

This configuration can be verified as follows, note that TCP traffic is functional, but for example ICMP is not:

```
SW2#ping 160.1.10.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.10.3, timeout is 2 seconds:.....
Success rate is 0 percent (0/5)
!SW2#telnet 160.1.10.3
Trying 160.1.10.3 ... Open
!SW3>show users
    Line       User       Host(s)              Idle       Location
   0 con 0                idle                 00:01:31
*  1 vty 0                idle                 00:00:00 160.1.10.2


  Interface    User               Mode         Idle     Peer Address
```

Verify the VLAN filtering settings:

```
SW1#show vlan filter
VLAN MapVLAN10_FILTER
 is filtering VLANs: 10
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

VLAN Filtering for Non-IP Traffic

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW1 to allow only the following Layer 2 protocols across VLAN 10:
  - STP BPDUs (consider both ISL and 802.1q trunks)
  - CDP, VTP, and UDLD
  - ARP
- Disallow any other Layer 2 packets.

Configuration

SW1:

```
mac access-list extended ALLOWED_L2_TRAFFIC
 permit any any lsap 0x4242 0x0
 permit any any 0x010B 0x0
 permit any any 0x806 0x0
!
vlan access-map VLAN10_FILTER 10
 match mac address ALLOWED_L2_TRAFFIC
 action forward
!
vlan filter VLAN10_FILTER vlan-list 10
```

https://t.me/learningnets

Verification

When you perform Layer 2 filtering, you usually do so based on Ethertypes or LSAP (Link Service Access Point) values. LSAP value consists of two parts (one byte each): SSAP and DSAP (Source SAP) and (Destination SAP). You can apply Layer 2 filtering based on MAC addresses, but this is not very useful because MAC access-lists only match non-IP traffic (such as ARP, STP, VTP). The following are the most commonly seen Layer 2 protocols:

IEEE STP BPDUs: These BPDUs use 802.2 LLC encapsulation with SSAP/DSSP values of 0x42 or LSAP value of 0x4242. Cisco switches and routers run IEEE STP over access-ports and run it across the native VLAN of a 802.1q trunk. You can also see STP packets sentacross ISL trunks using the same LSAP value of 0x42. The Cisco implementation does not modify the STP packets sent over ISL trunks, it just adds the ISL encapsulation. **PVST+ SSTP BPDUs:** These BPDUs use 802.2 SNAP encapsulation (LSAP=0xAAAA) with SNAP PID (Protocol ID) value of 0x010B. You can match this Protocol ID value as the Ethertype number in MAC ACLs in the Catalyst Switches. Cisco switches send PVST+ BPDUs across 802.1q only trunks in addition to IEEE STP BPDUs. The PVST+ BPDUsappear on native as well as non-native VLANs of a trunk port. **ARP protocol:** This uses an Ethernet II frame format with the Ethertype value of 0x806. The protocols below use 802.2 SNAP encapsulation with the SNAP Protocol ID values listed here:

```
VTP: 0x2003
CDP: 0x2000
DTP: 0x2004
UDLD: 0x0111
```

All SNAP-encapsulated packets can be matched using an LSAP value of 0xAAAA. The above-mentioned packet types have no VLAN tag header, so you can filter them on the native VLAN of a trunk, which is usually VLAN 1.

Pitfall

Be very careful when filtering Layer 2 protocols,because you can block STP BPDUs and effectively introduce topology loops and broadcast traffic storms.

For verification, make sure that you can see SW1 correctly executing the STP protocol. To accomplish this, configure SW2 as a STP root bridge for VLAN 10 and make sure that SW1 receives BPDUs from the root bridge:

```
SW2:
```

https://t.me/learningnets

```
spanning-tree vlan 10 priority 0

SW1#clear spanning-tree counters
!SW1#show spanning-tree vlan 10 interface fastEthernet0/24 detail
 Port 26 (FastEthernet0/24) of VLAN0010 is root forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.26.
   Designated root has priority 10, address 000a.b832.3a80
   Designated bridge has priority 10, address 000a.b832.3a80
   Designated port id is 128.26, designated path cost 0
   Timers: message age 1, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default BPDU: sent 0, received 8
!SW1#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    10
             Address     000a.b832.3a80
             Cost        19
             Port        26 (FastEthernet0/24)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Desg FWD 19        128.22   P2p Fa0/24               Root FWD 19
        128.26   P2p
```

Verify that CDP packets are not dropped by SW1:

```
SW1#show cdp traffic
CDP counters :         Total packets output: 55, Input: 129
        Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
        No memory: 0, Invalid packet: 0,
        CDP version 1 advertisements output: 0, Input: 0
        CDP version 2 advertisements output: 55, Input: 129
!SW1#show cdp traffic
CDP counters :         Total packets output: 55, Input: 130
        Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
        No memory: 0, Invalid packet: 0,
        CDP version 1 advertisements output: 0, Input: 0
```

https://t.me/learningnets

```
        CDP version 2 advertisements output: 55, Input: 130
```

Configure SW2 and SW3 with IP addresses in VLAN 10 and test ICMP connectivity which will confirm that ARP is functional through VLAN 10:

```
SW2:
interface Vlan10
 ip address 160.1.10.2 255.255.255.0

SW3:
interface Vlan10
 ip address 160.1.10.3 255.255.255.0

SW3#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  160.1.10.3             -   0022.5627.1fc1  ARPA   Vlan10
!SW3#ping 160.1.10.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.10.2, timeout is 2 seconds: .!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/3/9 ms
!SW3#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  160.1.10.2             0   000a.b832.3ac1  ARPA   Vlan10

Internet  160.1.10.3             -   0022.5627.1fc1  ARPA   Vlan10
```

To ensure that ARP traffic transited SW1, verify the STP topology for VLAN 10:

```
SW1#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    10
             Address     000a.b832.3a80
             Cost        19
             Port        26 (FastEthernet0/24)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    4106   (priority 4096 sys-id-ext 10)
             Address     0013.605f.f000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
```

https://t.me/learningnets

```
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Desg FWD
 19        128.22   P2p Fa0/24              Root FWD
 19        128.26   P2p
!SW2#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    10
             Address     000a.b832.3a80
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    10     (priority 0 sys-id-ext 10)
             Address     000a.b832.3a80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/19              Desg FWD 19        128.21   P2p
Fa0/20              Desg FWD 19        128.22   P2p Fa0/24              Desg FWD
 19        128.26   P2p
!SW3#show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    10
             Address     000a.b832.3a80
             Cost        38
             Port        22 (FastEthernet0/20)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32778  (priority 32768 sys-id-ext 10)
             Address     0022.5627.1f80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/20              Root FWD
 19        128.22   P2p
Fa0/23              Altn BLK 19        128.25   P2p
```

https://t.me/learningnets

```
Fa0/24              Altn BLK 19        128.26   P2p
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Port Security

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW1 to guard against MAC address flooding attacks on its ports as follows:
  - Age the learned secure entries after 10 minutes of inactivity.
  - Recover disabled ports after 3 minutes.
- Configure Fa0/5 for access mode in VLAN 10 and:
  - Limit the number of MAC addresses learned simultaneously to one.
  - In case of a policy violation, apply the shutdown action for the port.
- Configure Fa0/6 for access mode in VLAN 20 and:
  - Retain the MAC address learned on the port in the switch configuration.
  - In case of policy violation drop offending packets and generate log records of the violation.
- Configure Fa0/7 for 802.1q trunk mode and:
  - Limit the number of MAC addresses learned simultaneously to two, one for VLAN 10 and one for VLAN 20.
  - In case of policy violation drop offending packets.

Configuration

```
SW1:

interface FastEthernet0/5
 switchport mode access
```

https://t.me/learningnets

```
 switchport access vlan 10
 switchport port-security
 switchport port-security aging time 10
 switchport port-security aging type inactivity
!
interface FastEthernet0/6
 switchport mode access
 switchport access vlan 20
 switchport port-security
 switchport port-security aging time 10
 switchport port-security aging type inactivity
 switchport port-security violation restrict
 switchport port-security mac-address sticky
!
interface FastEthernet0/7
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport port-security
 switchport port-security aging time 10
 switchport port-security aging type inactivity
 switchport port-security violation protect
 switchport port-security maximum 2
 switchport port-security maximum 1 vlan 10
 switchport port-security maximum 1 vlan 20
!
errdisable recovery cause psecure-violation
errdisable recovery interval 180
```

Verification

Port Security is a Layer 2 feature that enforces a limit on the number of MAC addresses allowed per a particular switch port. The two main purposes of this feature is preventing unauthorized connections on a shared port and thwarting MAC-address flooding attacks. The MAC address flooding attack consists of sending a barrage of packets with different MAC addresses, forcing the switch to overpopulate its MAC address table. This attack may result in switch performance degradation and privacy violation. The latter may occur in cases when the switch starts behaving like a hub, flooding frames out all ports. This occurs when the MAC address table overflows. Note that port-security works only on ports configured as static access or static trunks. The port-security feature does not work on dynamic ports. Port-security logic is outlined below.

On a port configured for port-security, the switch keeps a table of secure MAC

https://t.me/learningnets

address entries. The total number of entries allowed in this table is configurable using the command `switchport port-security maximum-address <N>` . On trunk ports, the above command specifies the maximum number of MAC addresses active on all VLANs at the same time—the aggregate limit. Note that the switch treats the same MAC address on different VLANs as two different MAC addresses. For trunk ports, you may additionally specify the maximum number of MACs per VLAN by using the command `switchport port-security maximum <N> vlan <VLAN>` . If the port is an access-port configured with access and voice VLANs, you can use commands to impose restrictions on just two VLANs, without ever mentioning their numbers:

```
switchport port-security maximum <N> vlan [access|voice] .
```

When a switch has learned all allowed addresses on the port or a VLAN and a frame with a new source MAC address arrives on the port, the switch may take any of the following actions:

- Shutdown the port (the default “shutdown” action).
- Silently discard the frame (if configured for the “protect” action).Protect mode is not generally recommended, especially for trunks ports, although we use it in this task. The problem is that as soon as any VLAN on a trunk reaches its MAC address limit, the port stops learning MAC addresses on any other VLAN. The worst thing about this mode is that the switch does not notify you about this via any logging message.
- Discard the frame and generate a syslog message or SNMP trap (if configured for the “restrict” action). You may need to configure SNMP destination hosts to send the actual traps.

The switch does not allow the same MAC address to appear on more than one secure port at the same time. Thus, if a switch has learned a MAC address on a secure port, it will not allow the same address to appear on other switch ports until the secure entry has expired. The switch ages out secure MAC address entries using a configurable timeout. You can set this timeout per-port using two commands:

```
`switchport port-security aging timeout <TIMEOUT\>`
`switchport port-security aging type {absolute|inactivity}`
```

https://t.me/learningnets

Together these commands define the aging behavior. Absolute aging means to age each entry starting at the time it has been learned. Inactivity aging starts aging time only after each frame is received. If no frames have been received during the timeout, the switch removes the secure entry and opens a slot for a different MAC address. If the port-security feature has shut down a port, the port can be restored to an operational state using the error-disable recovery procedure. You can set one global recovery timeout (used for all types of error-disable recovery) by using the command:

```
`errdisable recovery interval <seconds>`
```

To enable a particular type of recovery procedure, use the following command:

```
`errdisable recovery cause <cause>`
```

In addition to setting up dynamic learning of secure MAC addresses, you may configure static secure MAC address entries using the interface-level command:

```
`switchport port-security mac-address H.H.H`
```

These entries also count against the maximum number of allowed MAC addresses on an interface. You may configure a port to age static secure MAC address entries as well using the command:

```
`switchport port-security aging static`
```

This may be useful when you need to set up guaranteed access for a specific MAC address for some amount of time. The last port-security feature is known as sticky learning. It allows you to learn “static” secure MAC address entries. When a switch learns new MAC addresses on a port in sticky mode, it generates a configuration line for a corresponding static entry. This line appears in the running configuration, so you must save it to make the static entry truly permanent.

For verification in our task, display the port-security configuration for all ports configured. As no hosts are connected to these ports, we can't simulate a policy violation or test the sticky MAC feature:

```
SW1#show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
                (Count)       (Count)          (Count)
--------------------------------------------------------------------------
      Fa0/5              1            0                  0         Shutdown
```

https://t.me/learningnets

```
      Fa0/6              1            0                  0         Restrict
      Fa0/7              2            0                  0          Protect
-----------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 6144
!SW1#show port-security interface fastEthernet0/5
Port Security               : Enabled
Port Status                 : Secure-down Violation Mode              : Shutdown
Aging Time                  : 10 mins
Aging Type                  : Inactivity
SecureStatic Address Aging : Disabled Maximum MAC Addresses       : 1
Total MAC Addresses         : 0
Configured MAC Addresses    : 0
Sticky MAC Addresses        : 0
Last Source Address:Vlan    : 0000.0000.0000:0
Security Violation Count    : 0
!SW1#show port-security interface fastEthernet0/6
Port Security               : Enabled
Port Status                 : Secure-down Violation Mode              : Restrict
Aging Time                  : 10 mins
Aging Type                  : Inactivity
SecureStatic Address Aging : Disabled Maximum MAC Addresses       : 1
Total MAC Addresses         : 0
Configured MAC Addresses    : 0
Sticky MAC Addresses        : 0
Last Source Address:Vlan    : 0000.0000.0000:0
Security Violation Count    : 0
!SW1#show port-security interface fastEthernet0/7
Port Security               : Enabled
Port Status                 : Secure-down Violation Mode              : Protect
Aging Time                  : 10 mins
Aging Type                  : Inactivity
SecureStatic Address Aging : Disabled Maximum MAC Addresses       : 2
Total MAC Addresses         : 0
Configured MAC Addresses    : 0
Sticky MAC Addresses        : 0
Last Source Address:Vlan    : 0000.0000.0000:0
Security Violation Count    : 0
!SW1#show port-security interface fastEthernet0/7 vlan
Default maximum: not set, using 6144
VLAN  Maximum    Current    10          1
          0 20          1
          0
```

Verify that err-disable recovery for port-security is enabled with the requested

https://t.me/learningnets

interval:

```
SW1#show errdisable recovery
ErrDisable Reason            Timer Status
-----------------            --------------
arp-inspection               Disabled
bpduguard                    Disabled
channel-misconfig (STP)      Disabled
dhcp-rate-limit              Disabled
dtp-flap                     Disabled
gbic-invalid                 Disabled
inline-power                 Disabled
l2ptguard                    Disabled
link-flap                    Disabled
mac-limit                    Disabled
loopback                     Disabled
pagp-flap                    Disabled
port-mode-failure            Disabled
pppoe-ia-rate-limit          Disabled psecure-violation            Enabled
security-violation           Disabled
sfp-config-mismatch          Disabled
small-frame                  Disabled
storm-control                Disabled
udld                         Disabled
vmps                         Disabled
psp                          Disabled
Timer interval: 180 seconds


Interfaces that will be enabled at the next timeout:
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

HSRP and Port Security

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW1's Fa0/19 anf Fa0/23 for access mode in VLAN 10.
  - Enable port-security on these ports.
- Configure SW2's Fa0/23 with IP address of 160.1.10.2 and SW3's Fa0/19 with IP address of 160.1.10.3.
  - Enable HSRP between SW2 and SW3 using a VIP of 160.1.10.254.

Configuration

```
SW1:
interface range FastEthernet0/19 , FastEthernet0/23
 switchport mode access
 switchport access vlan 10
 switchport port-security
 switchport port-security maximum 2
 no shutdown
SW2:
interface FastEthernet0/23
 no switchport
 ip address 160.1.10.2 255.255.255.0
 standby ip 160.1.10.254
 no shutdown
SW3:


interface FastEthernet0/19
```

https://t.me/learningnets

```
 no switchport
 ip address 160.1.10.3 255.255.255.0
 standby ip 160.1.10.254
 no shutdown
```

Verification

Virtual router protocols (HSRP, VRRP, and GLBP) use virtual MAC addresses in addition to physical interface addresses. This requires small changes in port-security configurations. At the very least, you must increase the maximum MAC address count to permit additional entries in the secure MAC address table. HSRP offers another option that allows you to use only physical MAC-addresses, even for the virtual IP address, via command **`standby use-bia`** .

First, look at the output of the switch show commands and logging buffer contents before you change the MAC address limit on SW1. Note that the additional HSRP virtual MAC address causes both ports to be disabled:

```
SW1#
%PM-4-ERR_DISABLE:psecure-violation error detected on Fa0/23, putting Fa0/23 in err-disable state
%PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred,caused by MAC address 0000.0c07.ac00
 on port FastEthernet0/23.%PM-4-ERR_DISABLE:
psecure-violation error detected on Fa0/19, putting Fa0/19 in err-disable state
```

Because SW1's interfaces are err-disabled, HSRP is no longer functional because SW2 and SW3 interfaces are in the down state:

```
SW1#show interfaces status err-disabled

Port      Name               Status       Reason               Err-disabled Vlans
Fa0/19                       err-disabled psecure-violation
Fa0/23                       err-disabled psecure-violation
!SW2#show standby
FastEthernet0/23 - Group 0 State is Init (interface down)
    6 state changes, last state change 00:01:42
  Virtual IP address is 160.1.10.254
  Active virtual MAC address is unknown
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
  Preemption disabled
  Active router is unknown
  Standby router is unknown
  Priority 100 (default 100)
```

https://t.me/learningnets

```
  Group name is "hsrp-Fa0/23-0" (default)
!SW3#show standby
FastEthernet0/19 - Group 0 State is Init (interface down)

    6 state changes, last state change 00:00:56
  Virtual IP address is 160.1.10.254
  Active virtual MAC address is unknown
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
  Preemption disabled
  Active router is unknown
  Standby router is unknown
  Priority 100 (default 100)
  Group name is "hsrp-Fa0/19-0" (default)
```

Configure SW1 to allow two MAC addresses on its ports to SW2 and SW3, bounce the ports, now HSRP should be functional:

```
SW1#show port-security address
               Secure Mac Address Table
-----------------------------------------------------------------------------
Vlan    Mac Address       Type                          Ports   Remaining Age
                                                                   (mins)
----    -----------       ----                          -----   -------------
  10    0022.5627.1fc1    SecureDynamic                 Fa0/19       -
10    0000.0c07.ac00    SecureDynamic                 Fa0/23       -
  10    000a.b832.3ac1    SecureDynamic                 Fa0/23       -
-----------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 1
Max Addresses limit in System (excluding one mac per port) : 6144
!SW2#show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP Fa0/23      0    100
Active  local           160.1.10.3
      160.1.10.254
!SW3#show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP Fa0/19      0    100
Standby 160.1.10.2       local
           160.1.10.254
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

DHCP Snooping

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW3 with IP address of 160.1.10.3 on VLAN 10.
- Configure SW3 as a DHCP server and SW2 as a DHCP client on VLAN 10.
- Configure SW1 to prevent potential DHCP attacks in such a way that it dynamically accounts for future hosts added to the VLAN 10 segment.
  - SW1 should store the binding database in flash with the filename **dhcp-bindings.txt**, and use a 15 second delay between changes.
  - Limit the number of DHCP messages that SW1 can receive from the DHCP client to 10 per second.

Configuration

```
SW1:
 ip dhcp snooping
 ip dhcp snooping vlan 10
 ip dhcp snooping database flash:/dhcp-bindings.txt
 ip dhcp snooping database write-delay 15
 !
 interface FastEthernet0/24
  ip dhcp snooping limit rate 10
 !
 interface FastEthernet0/20
  ip dhcp snooping trust
SW2:
```

https://t.me/learningnets

```
interface Vlan10
 ip address dhcp
SW3:

interface Vlan10
 ip dhcp relay information trusted
 ip address 160.1.10.3 255.255.255.0
!
ip dhcp pool VLAN10
 network 160.1.10.0 /24
```

Verification

DHCP Snooping is a security feature that inspects DHCP packets transiting a Layer 2 switch. The switch itself does not participate in DHCP packet exchange but enforces DHCP packet flow integrity. Before we start with an overview of DHCP snooping, let us quickly recap different agent roles for IP address allocation procedures using DHCP.

https://t.me/learningnets

- A DHCP client discovers a DHCP server and requests an IP address as well as other configuration parameters using a DHCPDISCOVER packet. The client may then select an IP address using a DHCPREQUEST message. When the client no longer needs the IP address, it returns it back to the server using a DHCPRELEASE packet.
- A DHCP Relay acts as a middleman agent between the DHCP clients and the server. The primary function of a DHCP Relay is to forward messages from local clients to the remote DHCP Server. When a DHCP Relay receives a broadcast packet from a connected client, it forwards it as a unicast packet to the IP address of the server. In the Cisco IOS, you can specify the IP address of the server using the interface-level command `ip helper-address X.X.X.X` . The DHCP relay changes the "giaddr" field in DHCP packets from zero to the IP address of the interface that forwards the packet. The DHCP server later uses this IP address to respond with a DHCP packet. A Catalyst switch may act as a DHCP Relay if you configure it as a Layer 3 device with an IP address and helper-address on some interface.
- A DHCP Server responds to discover and request messages from DHCP clients with DHCPOFFER and DHCPACK/DHCPNAK messages. The former message contains the IP address for the client, and the latter two acknowledge or reject a request message. The server uses the "giaddr" field to respond back, provided that it is non-zero. A server may also explicitly request a client to release and IP address with a DHCPRELEASEREQUERY message.

Every DHCP message also carries the MAC address of the client explicitly stored in one of DHCP fields. This MAC address may identify a client to the server if the Client ID field is set in the packet. The primary goal of using DHCP Snooping is to enforce DHCP security. When you enable DHCP Snooping on a switch, it starts treating every port as connected to a DHCP client (such as a workstation, or some downstream switch). For such ports, also called "untrusted" ports, the switch applies DHCP message filtering, only accepting messages expected from DHCP Clients (DHCPREQUEST, DHCPDISCOVER, DHCPRELEASE). The switch allows any type of DHCP messages on trusted ports. You may explicitly configure a port as trusted by the DHCP Snooping process using the interface-level command `ip dhcp snooping trust` . Usually, you need this command on the ports connected to DHCP servers or uplink ports on access switches.

On any type of port, you can configure the command `ip dhcp snooping limit rate <pps>` to control the number of DHCP packets per second received on a particular port. If the port exceeds the threshold set, the switch will put the port into an error-disabled state. The port can be recovered from this state by using common error-disable recovery procedures. In addition, if an incoming DHCP message contains a non-zero "giaddr", meaning it has been relayed, then it must be received on a

https://t.me/learningnets

trusted port. Untrusted ports simply discard such messages.

In addition to filtering messages based on their types, DHCP snooping also populatesa special DHCP snooping table based on messages exchanged across untrusted ports. This table contains the IP addresses that were allocated to clients by the DHCP server and other information such as client MAC address, client port, VLAN number, and lease duration. The switch consults this table when it receives DHCP messages such as DHCPRELEASE to make sure that no one would spoof a DHCP message for another host. The switch also ensures that MAC addresses in a DHCP packet match the MAC address of the host sending the message. This feature could be disabled by using the command `no ip dhcp snooping verify mac-address` . Note that messages received on trusted ports do not create any DHCP Snooping binding entries.

You may instruct the switch to save the DHCP snooping database contents in flash memory or on an external server (such as via TFTP). This allows the switch to preserve DHCP binding mappings across reloads. The global command to specify the external storage is `ip dhcp snooping database` . You may also control the interval between database updates by using the command `ip dhcp snooping database write-interval <seconds>` .

To start the DHCP Snooping process, you must enable it globally as well as activate it per-VLAN. You need to do *both* for DHCP Snooping to start working. Note that by default, when DHCP Snooping is enabled, the switch also adds DHCP Information Option (or Option 82) to all received packets. The purpose of this option is to identify the device and port that the client connects to. In short, the option contains the ID of the switch (such as MAC address or hostname) that inserted this option and the port identifier. This can be an interface name that the DHCP Client connects to. This information allows the DHCP Server to allocate IP addresses based on additional information—for example, split a subnet between two port ranges in the switch that inserted the option.

One issue that may arise here is that the switch inserts the option but leaves the “giaddr” field at zero. Thus, a DHCP Server may assume that option has been formatted incorrectly, because a DHCP Relay is supposed to set the “giaddr” field to its own IP address. Thus, an IOS DHCP server will reject by default such DHCP messages. To overcome this issue, you may use one of the following methods:

- Instruct the IOS DHCP Server to accept DHCP messages with a zero “giaddr” by using the global command `ip dhcp relay information trust-all` or the interface-level command `ip dhcp relay information trusted` .
- Configure the DHCP Snooping feature in the switch not to insert Option 82. This is accomplished by using the command `no ip dhcp-snooping information option` .

https://t.me/learningnets

- Trust the port where you receive the original DHCP message. The DHCP Snooping feature does not insert any Information Option into the received packets.

Here, we verify the status of DHCP snooping on SW1. Note the limits set for the port connected to DHCP client:

```
SW1#show ip dhcp snooping
Switch DHCP snooping is enabled
DHCP snooping is configured on following VLANs:
10
DHCP snooping is operational on following VLANs:
10
Smartlog is configured on following VLANs:
none
Smartlog is operational on following VLANs:
none
DHCP snooping is configured on the following L3 Interfaces:

Insertion of option 82 is enabled
   circuit-id default format: vlan-mod-port
   remote-id: 0013.605f.f000 (MAC)
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Verification of giaddr field is enabled
DHCP snooping trust/rate is configured on the following Interfaces:

Interface                  Trusted    Allow option    Rate limit (pps)
-----------------------    -------    ------------    ----------------   FastEthernet0/20            yes
        yes             unlimited
  Custom circuit-ids:FastEthernet0/24           no          no               10

Interface                  Trusted    Allow option    Rate limit (pps)
-----------------------    -------    ------------    ----------------
  Custom circuit-ids:
```

Verify the DHCP snooping binding database and configuration:

```
SW1#show ip dhcp snooping database
Agent URL : flash:/dhcp-bindings.txt
Write delay Timer : 15 seconds
Abort Timer : 300 seconds

Agent Running : No
Delay Timer Expiry : Not Running
```

https://t.me/learningnets

```
Abort Timer Expiry : Not Running

Last Succeded Time : 00:02:05 UTC Mon Mar 1 1993
Last Failed Time : None
Last Failed Reason : No failure recorded.

Total Attempts       :        2   Startup Failures :        0
Successful Transfers :        2   Failed Transfers :        0
Successful Reads     :        1   Failed Reads     :        0
Successful Writes    :        1   Failed Writes    :        0
Media Failures       :        0
!SW1#show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  --------------------
00:0A:B8:32:3A:C1   160.1.10.1       86318       dhcp-snooping   10    FastEthernet0/24

Total number of bindings: 1
```

You can also check the DHCP Snooping database contents by using the command below, which shows the raw database contents, as they are stored in flash memory:

```
SW1#more flash:/dhcp-bindings.txt

2b91527d
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
160.1.10.1 10 000a.b832.3ac1 2B92A3ED Fa0/24                              3e0676ed
END
```

Verify the DHCP binding on the DHCP server side and the allocated IP address on the DHCP client side, test IP connectivity to confirm functionality:

```
SW3#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/              Lease expiration        Type
                    Hardware address/
                    User name
160.1.10.1          0063.6973.636f.2d30.    Mar 02 1993 12:01 AM    Automatic
                    3030.612e.6238.3332.
                    2e33.6163.312d.566c.
                    3130
!SW2#show dhcp lease
Temp IP addr: 160.1.10.1  for peer on Interface: Vlan10
Temp  sub net mask: 255.255.255.0
```

https://t.me/learningnets

```
DHCP Lease server: 160.1.10.3, state: 5 Bound

   DHCP transaction id: 2297
   Lease: 86400 secs,  Renewal: 43200 secs,  Rebind: 75600 secs
   Next timer fires after: 11:54:31
   Retry count: 0   Client-ID: cisco-000a.b832.3ac1-Vl10
   Client-ID hex dump: 636973636F2D303030612E623833322E
                       336163312D566C3130
   Hostname: SW2
!SW3#ping 160.1.10.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.10.1, timeout is 2 seconds:.!!!!

Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/8 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

DHCP Snooping and the Information Option

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW3 with IP address of 160.1.10.3 on VLAN 10.
- Configure SW3 as a DHCP server and SW2 as a DHCP client on VLAN 10.
- Ensure that SW1 inserts Option 82 in DHCP requests received from DHCP client.
- Configure SW1 to use the string **SWITCH1** for Option 82 Remote-ID and the string **SWITCH2** as the Circuit-ID for the port connected to DHCP client.
- SW1 should accept information options in DHCP packets even on untrusted ports.

Configuration

```
SW1:
ip dhcp snooping
ip dhcp snooping vlan 10
ip dhcp snooping information option format remote-id string SWITCH1
ip dhcp snooping information option allow-untrusted
!
interface FastEthernet0/24
 ip dhcp snooping vlan 10 information option format-type circuit-id string SWITCH2
!
interface FastEthernet0/20
 ip dhcp snooping trust
SW2:
interface Vlan10
 ip address dhcp
SW3:
```

https://t.me/learningnets

```
interface Vlan10
 ip dhcp relay information trusted
 ip address 160.1.10.3 255.255.255.0
!
ip dhcp pool VLAN10
 network 160.1.10.0 /24
```

Verification

DHCP Information Option (Option 82) is used in large Enterprise and Metro Ethernet deployments to enhance functionality of IP address allocation processes. The option itself has no strict format; rather, the RFC defines a very flexible structure for a relay agent to insert any information in this option. Two commonly defined and used sub-fields are Remote-ID and Circuit-ID. The first field identifies the remote device that inserted the option—that is, the DHCP relay that redirected the original request. The second field identifies the point of client's attachment, which is commonly a port name or some other port identifier.

Over time, Cisco has been changing the format of Option 82 used in their switches. The most recent code uses the hostname as the default remote ID, but this can be changed to any ASCII string. For every port, you can also set a per-VLAN circuit ID as another ASCII string. This allows the use of different IDs for different VLANs on a trunk port. The switch does not accept DHCP packets with a non-zero "giaddr" on untrusted ports. Remember, these are the ports facing DHCP clients. In addition, it does not accept DHCP packets with Option 82 on untrusted ports. However, the latter behavior can be modified by using the command **`ip dhcp snooping information option allow-untrusted`**, which allows the switch to receive DHCP packets with Option 82 even on untrusted ports.

However, packets with a "giaddr" of zero are still rejected by the switch even with this command active in the configuration. You must mark the respective port as "trusted" to accept such packets. When you trust a port for DHCP Snooping operations, the switch does not create a snooping entry, nor does it add any information option by itself for IP packets received on the trusted port.

For verification in our case, enable a DHCP packet content dump on SW3 to read the Option 82 contents. Note the Client-ID that contains the name of the VLAN 10 interface of SW2 and the Option 82 contents containing the two strings configured in SW1:

```
SW3(config)#access-list 100 permit udp any any eq bootps
SW3#debug ip packet 100 dump

IP: s=0.0.0.0 (Vlan10), d=255.255.255.255, len 343, rcvd 1
```

https://t.me/learningnets

```
06AB9EF0:              FFFF FFFFFFFF 000AB832        ........82
06AB9F00: 3AC10800 45000157 000C0000 FF11BA8A  :A..E..W......:.
06AB9F10: 00000000 FFFFFFFF 00440043 0143A392  .........D.C.C#.
06AB9F20: 01010600 0000123D 00008000 00000000  .......=........
06AB9F30: 00000000 00000000 00000000 000AB832  ..............82
06AB9F40: 3AC10000 00000000 00000000 00000000  :A..............
06AB9F50: 00000000 00000000 00000000 00000000  ................
06AB9F60: 00000000 00000000 00000000 00000000  ................
06AB9F70: 00000000 00000000 00000000 00000000  ................
06AB9F80: 00000000 00000000 00000000 00000000  ................
06AB9F90: 00000000 00000000 00000000 00000000  ................
06AB9FA0: 00000000 00000000 00000000 00000000  ................
06AB9FB0: 00000000 00000000 00000000 00000000  ................
06AB9FC0: 00000000 00000000 00000000 00000000  ................
06AB9FD0: 00000000 00000000 00000000 00000000  ................
06AB9FE0: 00000000 00000000 00000000 00000000  ................
06AB9FF0: 00000000 00000000 00000000 00000000  ................
06ABA000: 00000000 00000000 00000000 63825363  ............c.Sc
06ABA010: 35010139 0204803D 1A006369 73636F2D  5..9...=..cisco-
06ABA020: 30303061 2E623833 322E3361 63312D56  000a.b832.3ac1-V
06ABA030: 6C31300C 03535732 37080106 0F2C0321  l10..SW27....,.!
06ABA040: 962B5216 01090107 53574954 43483202  .+R.....SWITCH2
.06ABA050: 09010753 57495443 4831FF             ...SWITCH1
.
```

Now try simulating a DHCP request with a non-zero “giaddr” field across SW1. To accomplish this, configure DHCP snooping on SW2 for VLAN 10 and configure SW4 as DHCP client on VLAN 10, before activating **`ip dhcp snooping information option allow-untrusted`** on SW1. As the debugging output reveals, the switch does not accept those packets on untrusted ports. It would still accept the packets with a zero “giaddr”. These are the packets with a DHCP option inserted by some other Layer 2 device:

```
SW2:
ip dhcp snooping
ip dhcp snooping vlan 10
!
interface FastEthernet0/24
 ip dhcp snooping trust
SW4:
interface Vlan10
 ip address dhcp

SW1#debug ip dhcp snooping event
```

https://t.me/learningnets

```
DHCP Snooping Event debugging is on
%DHCP_SNOOPING-5-DHCP_SNOOPING_NONZERO_GIADDR: DHCP_SNOOPING
drop message with non-zero giaddr or option82 value on untrusted port
, message type: DHCPDISCOVER, MAC sa: 001a.a174.2541
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Dynamic ARP Inspection

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW1, SW2 and SW3 with IP addresses in the format of 160.1.20.Y on VLAN 20, where Y is the switch number.
- Configure SW1 to prevent ARP poisoning attacks on VLAN 20.
  - Without configuring trust ports on SW1, ensure it enforces ARP security for SW2 and SW3.
  - Log all ARP packets permitted by the ARP access-list, but limit their rate to four per ten seconds.
  - Additionally, log all packets matched against the DHCP snooping database.
  - Store, at maximum, 16 entries in the ARP logging buffer.
  - Enable all additional sanity checks for ARP packets.

Configuration

```
SW1:
interface Vlan 20
 ip address 160.1.20.1 255.255.255.0
!
ip dhcp snooping
ip dhcp snooping vlan 20
!
arp access-list ARP_VLAN20
 permit ip host 160.1.20.2 mac host 000a.b832.3ac1 log
 permit ip host 160.1.20.3 mac host 0022.5627.1fc1 log
```

https://t.me/learningnets

```
!
ip arp inspection vlan 20
ip arp inspection vlan 20 logging acl-match matchlog
ip arp inspection log-buffer entries 16
ip arp inspection log-buffer logs 4 interval 10
ip arp inspection validate src-mac dst-mac ip
!
ip arp inspection filter ARP_VLAN20 vlan 20
SW2:
interface Vlan 20
 ip address 160.1.20.2 255.255.255.0
SW3:

interface Vlan 20
 ip address 160.1.20.3 255.255.255.0
```

Verification

Dynamic ARP inspection is a security feature that fixes some well-known weaknesses of the ARP protocol. Generally, ARP operates on a broadcast Ethernet segment and allows any host to spoof a MAC address for any IP address on the segment. These attacks, commonly known as Man-in-the-Middle (MiM) attacks, cannot be prevented by using just port-security, access-lists, or other well-known features. ARP Inspection creates a special IP to MAC address binding table in the switch. This table is dynamically populated based on the DHCP snooping database contents. You can also add static entries to the database manually, using the configuration commands for ARP Inspection access-lists.

When the switch receives an ARP packet on an ARP-untrusted (the default state) port, it inspects the packet contents. Based on the IP to MAC address binding information in the packet, the switch permits the packet only if it matches the ARP Inspection table. This prevents ARP poisoning attacks. Note that implementing ARP Inspection may break some services, such as Proxy ARP. To resolve these issues, ARP Inspection allows you to configure some ports as trusted for ARP Inspection. On trusted ports, the switch does not inspect any ARP message. It is common to trust ARP messages on switch uplink ports, pointing toward the network core. The command to make a port trusted is `ip arp inspection trust` .

https://t.me/learningnets

The ARP Inspection feature may perform additional checks on ARP packets. It can check that destination MAC addresses inside ARP packet bodies match the destination MAC address in Ethernet frames for ARP responses. In addition to this, the switch checks source MAC addresses in ARP packets and Ethernet frames for ARP requests. You can enable these two checks by using the following commands:

```
ip arp inspection validate src-mac
ip arp inspection validate dst-mac
```

Additionally, you may enable IP address consistency checks for ARP packets by using the command **`ip arp inspection validate ip`** . This command checks source/destination IP addresses for consistency. Specifically, it ensures that no host tries to bind MAC addresses to IP addresses such as “255.255.255.255” or “0.0.0.0”. When enabled by default, the IP ARP Inspection feature builds all ARP mapping information based on the DHCP bindings table. If there are hosts on the segment not using DHCP for address allocation, you must configure ARP access-lists. The syntax to create and apply a filter is:

```
arp access-list <ACL_NAME>
 permit [request|response] ip <address><mask> mac <address><mask> [log]


ip arp inspection filter <ACL_NAME> vlan <VLAN_ID> [static]
```

For the access-list, you specify an IP address and MAC address binding. Note that you commonly do not match ARP requests or responses and usually use just a host IP to host MAC address mapping. The ARP Inspection feature first checks the ARP access-list for a given VLAN to see if a given ARP packet is legitimate. If there are no permit matches found for the given IP and MAC address pair, and there is NO explicit **`deny ip any mac any`** statement in the end of the access-list, the feature also checks the DHCP bindings database. However, if there is an explicit deny statement in the end of access-list or the access-list has been applied with the **`static`** keyword, ARP Inspection does not consult the DHCP Snooping database.

Logging of denied or allowed packets is somewhat complicated with ARP Inspection. First, you may specify a log keyword in the ARP access-list. Then you must enable ARP inspection using the following command:

```
`ip arp inspection vlan <VLAN_ID> logging acl-match {matchlog|none}`
```

https://t.me/learningnets

This will enable or disable (the default) logging of ARP packets matching ACL entries configured with the `log` keyword. In parallel with the above command, you may configure two additional commands:

```
`ip arp inspection vlan <VLAN_ID> logging dhcp-bindings [all|permit|none]`
`ip arp inspection vlan <VLAN_ID> logging arp-probe`
```

Those two respective commands enable logging of packets matching the DHCP bindings database and the logging of special ARP probe packets (packets with a source IP of 0.0.0.0). By default, the switch logs ARP packets denied by the DHCP snooping database. You may change this mode by disabling the logging of DHCP denied packets or enable logging of permitted packets. You may even log all packets matched against the DHCP snooping entries.

The switch accumulates logged ARP packets in a special internal buffer. You can regulate the size of this buffer using the following command:

```
`ip arp inspection log-buffer entries <N>`
```

The switch empties this buffer using a rate-limited procedure based on the following command settings:

```
`ip arp inspection log-buffer logs <number><interval>`
```

Based on these settings, the switch will generate at maximum *<number>* of syslog messages during *<interval>* time. If you set the *<interval>* to zero, the switch will generate a message for every ARP packet to be logged. The last security feature is ARP message rate-limiting. This feature is on by default on untrusted ports and disabled on trusted ports. On both types of ports, you can enable this feature by using the following command:

```
`ip arp inspection limit rate [<pps> [burst interval <seconds>]|none]`
```

This command will restrict the rate of received ARP packets to *<pps>* per *<seconds>* interval. When the port exceeds this rate, the switch will bring it to the error-disable state. By default, the rate limit on untrusted ports is 15 pps. The feature limits the aggregate rate on trunks or EtherChannels, so you may need to adjust it in real-life situations.

For our tasks, verify that ARP is enabled for the particular VLAN.

https://t.me/learningnets

```
SW1#show ip arp inspection vlan 20


Source Mac Validation      : Enabled
Destination Mac Validation : Enabled
IP Address Validation      : Enabled

 Vlan     Configuration    Operation   ACL Match          Static ACL
 ----     -------------    ---------   ---------          ----------
20      Enabled           Active      ARP_VLAN20          No

 Vlan     ACL Logging      DHCP Logging      Probe Logging
 ----     -----------      ------------      -------------20      Acl-Match         Deny                Off
```

Note that ARP inspection is enabled on VLAN 20 and ARP logging for ACL entries is active. By default, DHCP Logging is also enabled. Clear the ARP cache on SW1 and send ping packets to SW2 and SW3. These packets should be matched and logged by the ARP ACL:

```
SW1#clear arp-cache
!SW1#ping 160.1.20.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.20.2, timeout is 2 seconds:.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/4/9 ms
!SW1#ping 160.1.20.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.20.3, timeout is 2 seconds:.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/8 ms
!SW1#show logging | b ARP


%SW_DAI-6-ACL_PERMIT: 1 ARPs (Res) on Fa0/24, vlan 20.([000a.b832.3ac1/160.1.20.2/0013.605f.f041/160.1.20.1/18:27:41
%SW_DAI-6-ACL_PERMIT: 1 ARPs (Req) on Fa0/24, vlan 20.([000a.b832.3ac1/160.1.20.2/0000.0000.0000/160.1.20.1/18:27:42
%SW_DAI-6-ACL_PERMIT: 1 ARPs (Res) on Fa0/20, vlan 20.([0022.5627.1fc1/160.1.20.3/0013.605f.f041/160.1.20.1/18:27:50
%SW_DAI-6-ACL_PERMIT: 1 ARPs (Req) on Fa0/20, vlan 20.([0022.5627.1fc1/160.1.20.3/0000.0000.0000/160.1.20.1/18:27:51
```

Change the MAC address entry for SW2 on VLAN 20 interface and observe how the switch denies the violating ARP packets. As you can see, there is no DHCP snooping entry to match the new SW2 MAC address, so the ARP packets are dropped by the switch:

```
SW1:
arp access-list ARP_VLAN20
 no permit ip host 160.1.20.2 mac host 000a.b832.3ac1 log
```

https://t.me/learningnets

```
 permit ip host 160.1.20.2 mac host aaaa.bbbb.cccc log
SW1#clear arp-cache
!SW1#ping 160.1.20.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.20.2, timeout is 2 seconds:.....
Success rate is 0 percent (0/5)
!SW1#show logging | b Invalid

%SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Res) on Fa0/24, vlan 20.([000a.b832.3ac1/160.1.20.2/0013.605f.f041/160
%SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Res) on Fa0/24, vlan 20.([000a.b832.3ac1/160.1.20.2/0013.605f.f041/160
%SW_DAI-4-DHCP_SNOOPING_DENY: 2 Invalid ARPs (Res) on Fa0/24, vlan 20.([000a.b832.3ac1/160.1.20.2/0013.605f.f041/160
%SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Res) on Fa0/24, vlan 20.([000a.b832.3ac1/160.1.20.2/0013.605f.f041/160
%SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Res) on Fa0/24, vlan 20.([000a.b832.3ac1/160.1.20.2/0013.605f.f041/160
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

IP Source Guard

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW2 with IP address of 160.1.30.2 on its Fa0/23 interface and SW3 with IP address of 160.1.30.3 on its Fa0/19 interface.
- Configure SW1 to prevent IP address spoofing on its Fa0/19 and Fa0/23 interfaces.
  - Enable all these interfaces for access mode in VLAN 30.
  - Ensure that your solution accounts for dynamic IP addresses obtained via DHCP.
  - Enforce Layer 2 filtering for the MAC addresses corresponding to secured IP addresses at the same time.

Configuration

```
SW1:
interface range FastEthernet0/19 , FastEthernet0/23
 switchport mode access
 switchport access vlan 30
 ip verify source port-security
 switchport port-security
 no shutdown
!
ip dhcp snooping
ip dhcp snooping vlan 30
!
ip source binding 000a.b832.3ac1 vlan 30 160.1.30.2 interface FastEthernet0/23
```

https://t.me/learningnets

```
ip source binding 0022.5627.1fc1 vlan 30 160.1.30.3 interface FastEthernet0/19
SW2:
interface FastEthernet0/23
 no switchport
 ip address 160.1.30.2 255.255.255.0
 no shutdown
SW3:

interface FastEthernet0/19
 no switchport
 ip address 160.1.30.3 255.255.255.0
 no shutdown
```

Verification

IP Source Guard is a feature intended to prevent IP packet spoofing at Layer 2 (MiM attacks). When you enable IP Source Guard on a port, the switch applies a Layer 3 filter to this port, only accepting the packets with source IP addresses matching DHCP snooping bindings created for the port. Enabling DHCP snooping is a prerequisite, but you may bind an IP address to a port manually, even if the host does not use DHCP.

As soon as you enable IP Source Guard, the switch only permits IP packets that match the DHCP snooping database or static IP to MAC addresses and port bindings. The switch also allows ingress DHCP packets for hosts to obtain IP addresses. IP Source Guard relieves you from the need of applying any IP ingress filtering on individual ports to prevent IP address spoofing. Combined with Dynamic ARP Inspection and DHCP Snooping it allows much more secure environment than default Layer 2 deployments.

You may enable IP Source Guard per-interface using the command `ip verify-source [port-security]` . The `port-security` option requires that port-security be enabled on the switch port as well. With this option, the switch filters packets based onbothe source IP and MAC addresses, and the secure MAC address is taken from the DHCP snooping database or a static mapping entry. Note that you may enable IP Source Guard on a trunk port as well. In this case, DHCP snooping must be enabled on all trunked VLANs for filtering to work properly.

Note that we need to enable DHCP snooping on VLAN 20 to ensure proper filtering for DHCP assigned hosts.

Here we verify that IP Source Guard is active on the ports, also check the manual bindings:

https://t.me/learningnets

```
SW1#show ip verify source

Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan   Log
---------  -----------  -----------  ---------------  -----------------  ----   ---
Fa0/19     ip-mac       active       160.1.30.3       00:22:56:27:1F:C1  30     disabled
Fa0/23     ip-mac       active       160.1.30.2       00:0A:B8:32:3A:C1  30     disabled
!SW1#show ip source binding

MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  --------------------
00:22:56:27:1F:C1   160.1.30.3       infinite    static          30    FastEthernet0/19
00:0A:B8:32:3A:C1   160.1.30.2       infinite    static          30    FastEthernet0/23
Total number of bindings: 2
```

Verify IP connectivity between SW2 and SW3:

```
SW2#ping 160.1.30.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.30.3, timeout is 2 seconds:.!!!!

Success rate is 80 percent (4/5), round-trip min/avg/max = 1/3/9 ms
```

Ensure that filtering actually prevents IP address spoofing by changing the IP address of SW2:

```
SW2:
interface FastEthernet0/23
 ip address 160.1.30.22 255.255.255.0
SW2#ping 160.1.30.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.30.3, timeout is 2 seconds:.....

Success rate is 0 percent (0/5)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Using Catalyst Ingress Access-Lists

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW2 with IP address of 160.1.30.2 on its Fa0/23 interface and SW3 with IP address of 160.1.30.3 on its Fa0/19 interface.
- Configure SW1's Fa0/19 and Fa0/23 as access mode in VLAN 30.
- Allow SW3 to only send ARP and ICMP packets out of its VLAN 30 connection.

Configuration

```
SW1:
interface range FastEthernet0/19 , FastEthernet0/23
 switchport mode access
 switchport access vlan 30
 no shutdown
!
mac access-list extended ARP_ONLY
 permit any any 0x806 0x0
!
ip access-list extended ICMP_ONLY
 permit icmp any any
!
interface FastEthernet0/19
 ip access-group ICMP_ONLY in
 mac access-group ARP_ONLY in
SW2:
```

https://t.me/learningnets

```
interface FastEthernet0/23
 no switchport
 ip address 160.1.30.2 255.255.255.0
 no shutdown
SW3:

interface FastEthernet0/19
 no switchport
 ip address 160.1.30.3 255.255.255.0
 no shutdown
```

Verification

You may apply IP or MAC access lists ingress on any Layer 2 port. You cannot apply them outbound on Layer 2 ports, only on SVI interfaces. In addition, the use of Layer 2 access-lists is limited to Layer 2 ports only. Note one important difference: In the 3550 platform, MAC-based access-lists match only non-IP traffic such as ARP, STP, and IPv6. In the 3560 platform, they match only non-IP/IPv6 traffic such as ARP, STP, IPX, and so on. You cannot filter IP traffic based on MAC address unless you use the port-security feature. Be careful when applying MAC-based access-lists; you can filter management traffic such as STP BPDUs.

Verify that filtering is configured on SW1's Fa0/19 interface:

```
SW1#show mac access-group interface fastEthernet0/19
Interface FastEthernet0/19: Inbound access-list is ARP_ONLY
   Outbound access-list is not set
!SW1#show ip interface fastEthernet0/19
FastEthernet0/19 is up, line protocol is up Inbound  access list is ICMP_ONLY
```

Verify that the access-list permits only ICMP traffic, which also confirms that ARP is functional:

```
SW3#ping 160.1.30.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 160.1.30.2, timeout is 2 seconds:.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/8 ms
!SW3#telnet 160.1.30.2
Trying 160.1.30.2 ... % Connection timed out; remote host not responding
```

https://t.me/learningnets

Remove the IP access-list and confirm that telnet is functional now:

```
SW1:
interface fastEthernet0/19
 no ip access-group ICMP_ONLY in

SW3#telnet 160.1.30.2

Trying 160.1.30.2 ... Open
SW2>show users
    Line       User       Host(s)              Idle       Location
   0 con 0                idle                 00:00:09
*  1 vty 0                idle                 00:00:00 160.1.30.3


  Interface    User               Mode         Idle     Peer Address
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Controlling Terminal Line Access

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Only allow Telnet connections to R2 from the Loopback0 subnets.
- Authenticate remote users locally using the name **TELNET** and the password of **CISCO**.
- Once connected to R2, only allow this user to connect to R1.
  - Log any attempt to connect from R2 to any destination on port 80.

Configuration

R2:

```
access-list 99 permit 150.1.0.0 0.0.255.255
!
access-list 100 permit ip any host 150.1.1.1
access-list 100 permit ip any host 155.1.146.1
access-list 100 permit ip any host 155.1.0.1
access-list 100 permit ip any host 155.1.13.1
access-list 100 permit ip any host 155.1.146.1
access-list 100 deny tcp any any eq 80 log
!
username TELNET password CISCO
username TELNET access-class 100
!
line vty 0 4
 access-class 99 in
```

https://t.me/learningnets

```
 login local
```

Verification

You can apply standard or extended access-lists to any VTY line or username using the respective access-class. Based on the direction of the access-class command, it can control connections to the router (inbound), or from the router (outbound). When an access-list applies to a particular user, it controls the user's outgoing connections from the router.

Using an extended access-list may be useful when you want to filter connections to a particular port—for example, when you want to deny connections to any rotary group port like 700X, or filter outgoing connections to port 80. When you use an extended access-list to match an incoming connection, use "any" as the destination IP address. The router will actually use the destination IP address 0.0.0.0 to match against the access-list, but using "any" is just a safeguard measure.

By specifying the log keyword, you may register packets matching a particular line in the access-list. This may be useful if you want to trace connections going to a particular port.

Try connecting to R2, sourcing the connection off the various interfaces:

```
R5#telnet 150.1.2.2
Trying 150.1.2.2 ... % Connection refused by remote host
!R5#telnet 150.1.2.2 /source-interface loopback0
Trying 150.1.2.2 ... Open



User Access Verification


Username: TELNET
Password: R2>
```

Try connecting from R2 to different destinations, and ensure that you can only connect to R1:

https://t.me/learningnets

```
R2>telnet 150.1.3.3

Trying 150.1.3.3 ... % Connections to that host not permitted from this terminal

!R2>telnet 150.1.1.1

Trying 150.1.1.1 ... Open
R1>
```

Now check that the router logs outgoing connections to port 80:

```
R2>telnet 150.1.3.3 80
Trying 150.1.3.3, 80 ... % Connections to that host not permitted from this terminal
!R2#show logging | b denied
%SEC-6-IPACCESSLOGP: list 100 denied tcp 0.0.0.0(38402) -> 150.1.3.3(80), 1 packet
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

IOS Login Enhancements

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- After 3 unsuccessful attempts to log into R3 in 30 seconds, block all further attempts for 40 seconds.
- Ensure that sessions sourced from the R5 Loopback0 are exempted from this restriction.
- Log every successful login attempt and every third unsuccessful attempt.
- Enforce a delay of 2 seconds between successive login attempts.
- Create a local username **TEST** with the password of **TEST** for this task.

Configuration

https://t.me/learningnets

```
R3:

username TEST password TEST
access-list 99 permit 150.1.5.5
!
login block-for 40 attempts 3 within 30
login quiet-mode access-class 99
login on-failure log every 3
login on-success log
login delay 2
!
line vty 0 4
 login local
```

Verification

Login enhancements or IOS login block provides protection from brute-force dictionary attacks. This feature counts the number of failed login attempts during a time interval and prevents any further attempts for the duration of the block time interval. The command to set this behavior is:

```
`login block-for <seconds> attempts <tries> within <interval>`
```

This command will block any further login attempts for <*seconds*> if there were <*tries*> failed attempts during the <*interval*>. During this “quiet” interval, all login attempts are blocked with the exception of the hosts mentioned in the following access-list:

```
`login quiet-mode access-class {acl-name|acl-number}`
```

This is a special access-list that allows some important hosts to connect to the router even though it is blocking logins. Additional commands for this feature include:

```
`login on-failure log [every <login>]`
`login on-success log [every <login>]`
`login delay <seconds>`
```

The first command enables logging every failed <*login*> attempt and the second command enables logging every unsuccessful <*login*> attempt. For example, if <*login*> is 3, then every third attempt is logged. Without the parameter, these

https://t.me/learningnets

commands log every next attempt. The last command specifies the delay to enforce between every next login attempt to the router. Note that these features are configured independent of any AAA settings. Additionally, banners and messages are also set separately from this feature.

This feature does not work with line-based authentication, only with AAA or local database authentication. Try logging in to R3 and fail authentication three times. Note that the router has logged the third failed attempt.

```
R3#telnet 150.1.3.3
Trying 150.1.3.3 ... Open

User Access Verification

Username: TEST
Password:
% Login invalid

Username: TEST
Password:
% Login invalid

Username: TEST
Password:
% Login invalid

%SEC_LOGIN-4-LOGIN_FAILED: Login failed [user: TEST] [Source: UNKNOWN] [localport: 23] [Reason: Login Authenticatio
```

At the same time, the router will enter the quiet mode, blocking any further login attempts. After 40 seconds, the quiet mode expires.

```
%SEC_LOGIN-1-QUIET_MODE_ON: Still timeleft for watching failures is 6 secs, [user: TEST] [Source: UNKNOWN] [localpo
[Connection to 150.1.3.3 closed by foreign host]
R3#telnet 150.1.3.3
Trying 150.1.3.3 ... % Connection refused by remote host
```

Verify that the router is currently in quiet mode:

```
R3#show login
     A login delay of 2 seconds is applied.
     Quiet-Mode access list 99 is applied.
     All successful login is logged.
```

https://t.me/learningnets

```
     Every 3 failed login is logged.

     Router enabled to watch for login Attacks.
     If more than 3 login failures occur in 30 seconds or less,
     logins will be disabled for 40 seconds.
 Router presently in Quiet-Mode.
     Will remain in Quiet-Mode for 20 seconds.
     Restricted logins filtered by applied ACL 99.
!R3#show login failures

Information about last 50 login failure's with the device

Username        SourceIPAddr    lPort Count TimeStamp
TEST            150.1.3.3       23    3     23:49:51 UTC Sun Aug 3 2014
```

Once the router exists the quiet mode, a message similar with the following will be logged:

```
%SEC_LOGIN-5-QUIET_MODE_OFF: Quiet Mode is OFF, because block period timed out at 15:56:28 UTC Tue May 20 2014
```

Now try logging in correctly and see if this attempt gets logged:

```
R3#telnet 150.1.3.3
Trying 150.1.3.3 ... Open



User Access Verification

Username: TEST
Password: TESTR3>
!
%SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: TEST] [Source: UNKNOWN] [localport: 23] at 16:05:58 UTC Tue May 20
```

Test the exemption for the quiet period. First try logging in three times incorrectly to R3 from R5. While testing, note that the delay between each attempt is 2 seconds:

```
R5#telnet 150.1.3.3
Trying 150.1.3.3 ... Open



User Access Verification
```

https://t.me/learningnets

```
Username: TEST
Password:
% Login invalid

Username: TEST
Password:
% Login invalid

Username: TEST
Password:
% Login invalid

[Connection to 150.1.3.3 closed by foreign host]
!R5#telnet 150.1.3.3
Trying 150.1.3.3 ... % Connection refused by remote host
!R5#telnet 150.1.3.3 /source Loopback0
Trying 150.1.3.3 ... Open



User Access Verification

Username: TEST
Password:R3>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Role-Based CLI

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Create roles in R4 named **SUPER**, **DEBUG**, **INTERFACE1**, and **INTERFACE2**.
- The role **INTERFACE1** should have access to all IP configuration commands of interface GigabitEthernet1.146 only.
  - For role **INTERFACE2**, do the same with the exception that it applies to interface GigabitEthernet1.45.
- The role **DEBUG** should be able to use any debug and undebug commands, and it should be able to inspect the running configuration.
- The role **SUPER** should be a sum of all the other roles.
- Use the password **CISCO** to authenticate all roles.

Configuration

R4:

```
aaa new-model
enable password CISCO
!
exit
!
enable view
!
configure terminal
!
```

https://t.me/learningnets

```
parser view DEBUG
 secret CISCO
 commands exec include show running-config
 commands exec include all debug
 commands exec include all undebug
!
parser view INTERFACE1
 secret CISCO
 commands interface include all ip
 commands configure include interface
 commands exec include configure terminal
 commands configure include interface GigabitEthernet1.146
!
parser view INTERFACE2
 secret CISCO
 commands interface include all ip
 commands configure include interface
 commands exec include configure terminal
 commands configure include interface GigabitEthernet1.45
!
parser view SUPER superview
 secret CISCO
 view DEBUG
 view INTERFACE1
 view INTERFACE2
```

Verification

Role-based CLI enhances the local command authorization model. Instead of using hierarchical privilege levels, you may define user roles in Cisco IOS, called “views,” and assign command sets accessible to each view. You may further group views in hierarchies using the concept of a super-view that contains other views instead of a list of the allowed commands. Each view is password protected, and users are required to enter this password when switching to the view. Alternatively, a view may be associated with a user utilizing the local database or special external AAA attribute, specifying the role name. Note that enabling AAA with **`aaa new-model`** is mandatory for the role-based access-control to work.

One special “root” view always exists. By default, even when in privileged exec mode, you are not part of the root view. You must explicitly switch to the root view to be able to create other views. This is the only difference from the regular enable privilege mode. You can switch to the root view by using the command **`enable view`** . You can define a new view by using the command parser **`view <VIEW_NAME>`** . Under

https://t.me/learningnets

the view configuration mode, you define the password for the view using the command **secret <password>** and define commands accessible to the view by using a syntax similar to command-authorization:

```
`commands parser-mode [include|exclude|include-exclusive] [all] [interface interface-name] [command]`
```

Here, **include-exclusive** means that the commands included in this view are not accessible to other views. By using the **interface** keyword, you can limit the interfaces accessible to the role, provided that the user has access to the configure mode **interface** command. You can switch to the view by using the command **enable view <VIEW_NAME>** in the router CLI. If you need to define a superview, use the command **parser view <NAME> superview** and add views using the nested command **view <SUBVIEW_NAME>** . Note that all views, including superview, must have passwords defined to work.

To verify, try switching between the views in the router CLI. First try the view that has access only to one interface:

https://t.me/learningnets

```
R4#enable view INTERFACE1

Password: CISCO

PARSER-6-VIEW_SWITCH: successfully set to view 'INTERFACE1'.
!R4#show parser view
Current view is 'INTERFACE1'

!R4#?

Exec commands:configure  Enter configuration mode

do-exec    Mode-independent "do-exec" prefix support
enable     Turn on privileged commands
exit       Exit from the EXEC
show       Show running system information
!R4#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#?

Configure commands:
do-exec    To run exec commands in config mode
end        Exit from configure mode
exit       Exit from configure modeinterface  Select an interface to configure

!R4(config)#interface GigabitEthernet1.146
R4(config-subif)#?

Interface configuration commands:
  exit  Exit from interface configuration mode ip    Interface Internet Protocol config commands

!R4(config-subif)#interface GigabitEthernet1.45

                              ^% Invalid input detected at '^' marker.
```

Now try accessing the superview and observe which commands are available:

```
R4#enable view SUPER
Password: CISCO
%PARSER-6-VIEW_SWITCH: successfully set to view 'SUPER'.
!R4#show parser view
Current view is 'SUPER'
```

https://t.me/learningnets

```
!R4#?

Exec commands:
  Exec commands:
configure  Enter configuration mode
debug      Debugging functions (see also 'undebug')
do-exec    Mode-independent "do-exec" prefix support
enable     Turn on privileged commands
exit       Exit from the EXEC
show       Show running system information
undebug    Disable debugging functions (see also 'debug')
!
R4#
R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.146
R4(config-subif)#?
Interface configuration commands:
  exit  Exit from interface configuration mode ip    Interface Internet Protocol config commands
!R4(config-subif)#interface GigabitEthernet1.45
R4(config-subif)#?
Interface configuration commands:
  exit  Exit from interface configuration mode ip    Interface Internet Protocol config commands
!R4#show running-config

Building configuration...

Current configuration : 305 bytes
!
! Last configuration change at 07:15:12 UTC Mon May 19 2014
! NVRAM config last updated at 15:28:43 UTC Sun May 11 2014
!
!
!
!
interface GigabitEthernet1.45
 ip address 155.1.45.4 255.255.255.0
 ip nbar protocol-discovery
!
interface GigabitEthernet1.146
 ip address 155.1.146.4 255.255.255.0
!
!
end
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Controlling the ICMP Messages Rate

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- In the future, you plan to apply an input traffic filter on R4's VLAN 146 interface.
- Configure this interface not to respond with an ICMP message when the filter drops a packet.
- The rate of these messages sent out of all other interfaces should not exceed 100 per second.
- To ensure that PMTUD works correctly, increase the rate of ICMP messages used by this feature to 1000 per second on R4.

Configuration

```
R4:

interface GigabitEthernet1.146
 no ip unreachables
!
ip icmp rate-limit unreachable 10
ip icmp rate-limit unreachable DF 1
```

Verification

Because routers operate at Layer 3, they can generate various ICMP messages

https://t.me/learningnets

when processing IP packets. Most commonly, routers respond with ICMP unreachable messages when dropping packets for some reason. Most commonly, the reasons are as follows:

- Network/host/port unreachable—the router cannot route the packet to the destination.
- Administratively Prohibited—the filter configuration on the router drops the packet.
- Fragmentation required but DF-bit set—this message is critical in the PMTU discovery procedure.

There are three ways to control the generation of ICMP unreachable messages:

- Disable generation of ICMP unreachables out of the specified interface. The interface-level command is `no ip unreachables` . You may want to enable this command to make it harder to map your network for an outside attacker by hiding the information of unreachable/filtered hosts or segments.
- Control the rate of ICMP unreachable messages sent by the router. The global command is `ip icmp rate-limit unreachable <once per this ms>` . This command limits the total rate of all router-generated unreachable messages and prevents attacks designed to exhaust router resources.
- Control the rate of “packet-too-big” messages (DF bit set but fragmentation required). Because this message type is crucial to PMTUD, you may want to set a separate rate-limit for it by using the command `ip icmp rate-limit unreachable DF <once per this ms>` .

You can determine whether the unreachables are disabled on the interface by using the following show command:

```
R4#show ip interface gigabitEthernet1.146 | include unreach
ICMP unreachables are never sent
```

Try tracing the route to different R4 interfaces. Traceroute sends UDP packets towards special UDP ports numbers. Because the router is the ultimate traceroute destination, it should respond with ICMP unreachables. In the first case, all probes timed out, because unreachables are disabled. In the second case, all probes are successful, initially packets are sent through the DMPVN hub, as the spoke-to-spoke tunnel is established:

```
R1#traceroute 155.1.146.4 probe 5 timeout 1

Type escape sequence to abort.
Tracing the route to 155.1.146.4
```

https://t.me/learningnets

```
  1  *  *  *  *  *  2  *  *  *  *  *
!R1#traceroute 155.1.0.4 probe 5 timeout 1
Type escape sequence to abort.
Tracing the route to 155.1.0.4
VRF info: (vrf in name/id, vrf out name/id)
  1  *
    155.1.0.5 2 msec 1 msec 1 msec 1 msec
  2  *  *  *  *  *
  3  *  *  *  *  *
  4 155.1.0.4 2 msec *  2 msec *  2 msec
!R1#traceroute 155.1.0.4 probe 5 timeout 1
Type escape sequence to abort.
Tracing the route to 155.1.0.4
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.0.4 3 msec *  2 msec *  1 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

Control Plane Policing

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Limit the aggregate rate of ARP traffic toward R1's route processor to 100 packets per second.
- The routing control traffic marked with an IP Precedence value of 6 should be limited to 50 packets per second.
- Limit the rate of outgoing ICMP messages to 10 per second.

Configuration

R1:

```
class-map ARP
 match protocol arp
!
ip access-list extended ICMP
 permit icmp any any
!
class-map ICMP
 match access-group name ICMP
!
class-map ROUTING
 match ip precedence 6
!
policy-map CPP_INPUT
 class ARP
```

https://t.me/learningnets

```
  police rate 100 pps
 class ROUTING
  police rate 50 pps
!
policy-map CPP_OUTPUT
 class ICMP
  police rate 10 pps
!
control-plane
 service-policy input CPP_INPUT
 service-policy output CPP_OUTPUT
```

Verification

Control plane policing (CoPP) allows you to control the rate for traffic destined to or originated from the router process. This traffic includes all process-switched traffic, such as IP packets with options or packets logged by an access-list. Other types of traffic directed to the router process include routing updates, Telnet sessions, and any other traffic directed to the router itself. The router may generate outgoing traffic, including ICMP responses, returning Telnet session traffic, outgoing IGP/BGP updates, outgoing Telnet sessions, etc.

To control this traffic, you must define traffic classes using MQC syntax and apply a special control-plane policy-map. The only applicable policy-map actions are “drop” and “police.” For classification, you may use an IP access-list with DSCP/IP precedence matching. Additionally, you can match protocol ARP directly in the class-maps. Do not try to use NBAR to classify the control plane traffic, because it may have unpredictable results. For the “police” action, CoPP supports a unique packet-per-second rate-limiting feature. Using the command **`police rate <N> pps`** , you can specify the aggregate rate for a class in packets per second. You can optionally specify the burst size (amount of packets received instantly) if you need to fine-tune your configuration. Note that this feature only works with CoPP policy-maps.

To verify the CoPP feature, configure R4 to send ICMP packets to R1:

https://t.me/learningnets

```
R4#ping 155.1.146.1 repeat 1000 timeout 1

Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 155.1.146.1, timeout is 1 seconds:
!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!
!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!
.!!.!!.!!.!!.!!.!!.Success rate is 66 percent
 (106/159), round-trip min/avg/max = 1/1/10 ms
```

To understand this behavior, look at the **`show policy-map control-plane`** command output at R1. Note the default burst size of 10 pps which is two packets. This is why R1 only accepts two packets in a row and drops the third packet. It takes one second for the token bucket to completely refill and then permit two more packets:

```
R1#show policy-map control-plane output
 Control Plane

  Service-policy output: CPP_OUTPUT

    Class-map: ICMP (match-all)
      159 packets, 15900 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: access-group name ICMP
      police: rate 10 pps, burst 2 packets
conformed 106 packets, 10600 bytes; actions:
transmit
exceeded 53 packets, 5300 bytes; actions:
drop

        conformed 0 pps, exceeded 0 pps

    Class-map: class-default (match-any)
      279 packets, 21094 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

Also verify the policy-map applied in the input direction:

```
R1#show policy-map control-plane input
 Control Plane

  Service-policy input: CPP_INPUT
```

https://t.me/learningnets

```
 Class-map: ARP (match-all)

      159 packets, 15900 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: protocol arp
      police: rate 100 pps, burst 24 packets
 conformed 159 packets, 15900 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0 pps, exceeded 0 pps
 Class-map: ROUTING (match-all)
      264 packets, 22840 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 6
      police: rate 50 pps, burst 12 packets
 conformed 268 packets, 23180 bytes; actions:

          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 1 pps, exceeded 0 pps

    Class-map: class-default (match-any)
      825 packets, 75688 bytes
      5 minute offered rate 1000 bps, drop rate 0000 bps
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

IOS ACL Selective IP Option Drop

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R3 to drop any packets destined to the router with IP source route option (either loose or strict).
- Apply this protection to the link connected to R7.

Configuration

R3:

```
ip access-list extended DROP_IP_SOURCE_ROUTE
 deny ip any any option lsr
 deny ip any any option ssr
 permit ip any any
!
interface GigabitEthernet1.37
 ip access-group DROP_IP_SOURCE_ROUTE in
```

Verification

IP options are special extensions to the standard IP header (which is 20 bytes in length) that signal special processing requirements for the datagram. For example, IP options may instruct the router to record the route in the IP header or specify a source-route at the origin. Packets with IP options are process-switched by Cisco

https://t.me/learningnets

IOS routers and put a significant load on the router CPU. Thus, it is important to filter packets with IP options unless they are really needed. You may configure the router to silently discard all packets with IP options by using the command `ip options drop`. You may use access-lists to drop IP options selectively, using the ACL line syntax similar to the following: **`permit ip any any option <Option Name>`**.

You may find the list of IP options by using the command-line interface help features ("?" sign). The special **`any-option`** keyword selects any option. The access-list may be used in either a policy-map or as an access-group. Notice that only named extended access-lists are supported for this feature.

To verify IP option-based filtering, generate ICMP packets with the Loose source route option from R7. Notice the response packets from R3—the ICMP unreachable messages have reason "Received Packet has Options."

```
R7#ping
Protocol [ip]: Target IP address:155.1.37.3
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]: Loose, Strict, Record, Timestamp, Verbose[none]:Loose
Source route:155.1.37.3
Loose, Strict, Record, Timestamp, Verbose[LV]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.37.3, timeout is 2 seconds:
Packet has IP options:  Total option bytes= 7, padded length=8
 Loose source route: <*>
   (155.1.37.3)
Unreachable from 155.1.37.3.  Received packet has options
 Total option bytes= 7, padded length=8
 Loose source route: <*>
   (155.1.37.3)
Request 1 timed out
Unreachable from 155.1.37.3.  Received packet has options
 Total option bytes= 7, padded length=8
 Loose source route: <*>
   (155.1.37.3)

Request 3 timed out
Unreachable from 155.1.37.3.  Received packet has options
```

https://t.me/learningnets

```
 Total option bytes= 7, padded length=8
 Loose source route: <*>
   (155.1.37.3)
Success rate is 0 percent (0/5)
```

Check the access-list statistics on R3 after this:

```
R3#show ip access-lists DROP_IP_SOURCE_ROUTE
Extended IP access list DROP_IP_SOURCE_ROUTE 10 deny ip any any option lsr (5 matches)

    20 deny ip any any option ssr
    30 permit ip any any (164 matches)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Security

BGP Generic TTL Security Mechanism

> You must load the initial configuration files for the section, **Security Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure eBGP session between R5 and R6, and make both routers accept the peering only if it's no more than two hops away.
  - Use AS 100 on R5 and AS 200 on R6.
  - Use Loopback0 interfaces for the peering.

Configuration

```
R5:
router bgp 100
 neighbor 150.1.6.6 remote-as 200
 neighbor 150.1.6.6 ttl-security hops 2
 neighbor 150.1.6.6 update-source loopback0
R6:

router bgp 200
 neighbor 150.1.5.5 remote-as 100
 neighbor 150.1.5.5 ttl-security hops 2
 neighbor 150.1.5.5 update-source loopback0
```

Verification

The Generalized TTL Security Mechanism (GTSM) defined in RFC 3682 specifies a

https://t.me/learningnets

protection method against BGP session hijacking and resource exhaustion attacks. Generally the BGP process listens on the TCP port 179 and accepts all TCP SYN packets destined to this port, unless they are filtered by an ACL. It is possible to generate a barrage of spoofed packets imitating a valid BGP session and inject false information (if the session is unauthenticated) or generate a TCP SYN-flooding attack.

GTSM utilizes the simple fact that every router on the path to the BGP speaker decrements the TTL field in IP packets by one. Based on this, it is possible to identify potentially spoofed packets by looking at their TTL field; the packets sent from “afar” will have the TTL field below some threshold. It is possible to define a “secure radius” in the number of hop counts to accept the incoming IP packets. For example, if all BGP peers are within 10 hops from the local BGP speaker, all incoming IP packets will have their TTL field set to at no less than 245. This is because all IP packets start with TTL=255 and the field is decremented by every hop on the path. Thus, by accepting the IP packets with TTL greater than or equal to 245, it is possible to minimize the risk of spoofed packets reaching the BGP process. Notice that the usefulness of GTSM feature decreases as the diameter of eBGP Multihop session grows.

To configure the TTL security checks for a BGP peer, use the command **`neighbor <IP> ttl-security hops <hop-count>`** . This command applies to eBGP peering sessions only (either directly-connected or multihop) and specifies the number of hops the remote peer could be away from the local speaker. Remember that the internal BGP sessions are not protected, and therefore the internal network assumed to be “trusted.” All incoming TCP packets targeted at the BGP port with an IP TTL value below (255 - <hop-count>) are silently discarded by the router. In addition, the feature sets the TTL value for outgoing TCP/IP packets to 255 to make sure the remote peer will accept the local packets. The GTSM feature is mutually exclusive with the `ebgp-multihop BGP` feature. This is because the eBGP session by default sets TTL=1 in the outgoing IP packets and with the `multihop <n>` session parameter, the TTL value is set to <n>, which is not compatible with GTSM. Therefore, make sure you configure the GTSM feature on both sides of the peering link.

Check the TTL settings for every peer. Notice the minimum incoming TTL of 253 in the peer properties:

https://t.me/learningnets

```
R5#show ip bgp neighbors 150.1.6.6 | inc TTL
 Connection is ECN Disabled, Mininum incoming TTL 253
, Outgoing TTL 255
!R6#show ip bgp neighbors 150.1.5.5 | include TTL
 Connection is ECN Disabled, Mininum incoming TTL 253
, Outgoing TTL 255
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Exec Aliases

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure aliases on R1 to accomplish the following:
  - Issuing the command `ri` in exec mode should view the routing table contents, whereas `rb` should view the BGP table.
  - Issuing `iae ACL` in global configuration should create the extended access-list named ACL.
  - Issuing `so` and `si` should apply a policy-map to an interface either outbound or inbound, respectively.

Configuration

R1:

```
alias exec ri show ip route
alias exec rb show ip bgp
!
alias configure iae ip access-list extended
!
alias interface si service-policy input
alias interface so service-policy output
```

https://t.me/learningnets

Verification

Command aliases in IOS can be used as shortcuts to speed up and simplify repetitive configurations. Aliases can be configured for other parser modes, such as route-map mode or class-map mode, but the most common ones are exec, global configuration, and interface mode. These configurations can be verified as follows:

```
R1(config)#iae ACL
R1(config-ext-nacl)#exit
!R1(config)#interface GigabitEthernet1
R1(config-if)#so TEST
% policy map TEST not configured
!R1(config-if)#si TEST
% policy map TEST not configured
!R1#ri
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

  150.1.0.0/32 is subnetted, 3 subnets
C        150.1.1.1 is directly connected, Loopback0
D        150.1.3.3 [90/130816] via 155.1.13.3, 1d22h, GigabitEthernet1.13
D        150.1.6.6 [90/130816] via 155.1.146.6, 1d06h, GigabitEthernet1.146
     155.1.0.0/16 is variably subnetted, 8 subnets, 2 masks
C        155.1.0.0/24 is directly connected, Tunnel0
L        155.1.0.1/32 is directly connected, Tunnel0
C        155.1.13.0/24 is directly connected, GigabitEthernet1.13
L        155.1.13.1/32 is directly connected, GigabitEthernet1.13
D        155.1.37.0/24 [90/3072] via 155.1.13.3, 1d22h, GigabitEthernet1.13
D        155.1.67.0/24 [90/3072] via 155.1.146.6, 1d06h, GigabitEthernet1.146
C        155.1.146.0/24 is directly connected, GigabitEthernet1.146
L        155.1.146.1/32 is directly connected, GigabitEthernet1.146
      169.254.0.0/16 is variably subnetted, 2 subnets, 2 masks
C        169.254.100.0/24 is directly connected, GigabitEthernet1.100
L        169.254.100.1/32 is directly connected, GigabitEthernet1.100
!
!R1#ri 150.1.3.3
```

https://t.me/learningnets

```
Routing entry for 150.1.3.3/32
    Known via "eigrp 100", distance 90, metric 130816, type internal
    Redistributing via eigrp 100
    Last update from 155.1.13.3 on GigabitEthernet1.13, 1d22h ago
    Routing Descriptor Blocks:
    * 155.1.13.3, from 155.1.13.3, 1d22h ago, via GigabitEthernet1.13
    Route metric is 130816, traffic share count is 1
    Total delay is 5010 microseconds, minimum bandwidth is 1000000 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 1
!
!R1#rb
% BGP not active
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

System Message Logging

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Enable system message logging on R4 and R6 as follows:
  - Both routers should save debugging messages to their internal buffers up to 8192 bytes.
  - Debugging messages should be sent to the router consoles, but limited to 1 message per second.
  - Console log messages should not interrupt other command output.
  - Users logged in via telnet should only see informational level messages and above.

Configuration

```
R4:
 logging on
 logging buffered 8192 debugging
 logging console debugging
 logging rate-limit console all 1
 logging monitor informational
 !
 line console 0
  logging synchronous
R6:
```

https://t.me/learningnets

```
logging on
logging buffered 8192 debugging
logging console debugging
logging rate-limit console all 1
logging monitor informational
!
line console 0
 logging synchronous
```

Verification

Logging messages can be sent to multiple destinations simultaneously, including the console/terminal lines, the internal buffer, and a remote syslog server. Some platforms also allow storing logging messages in the local flash memory. For each destination, you can configure the message severity level, which controls which messages are logged. Logging at severity 7 (debugging) includes all messages 0 through 7. Logging at severity 4 includes all messages 0 through 4.

For syslog the logging facility is also available, which controls the format of the log message. Using different facilities for different devices—for example, local5 for all routers and local6 for all switches—can make sorting through log files on the syslog server itself more manageable. Synchronous logging on a terminal line makes the logging process wait for the terminal to finish printing a line before outputting its own message.

Logging rate-limiting shrinks the number of messages (actually, message lines) sent per second either to all destinations or to the console line specifically. Note that the EIGRP debugging output below only prints a limited number of messages to the console (because rate limiting is enabled), whereas the logging buffer contains full messages.

```
R4#debug ip eigrp 100
 R4#clear ip eigrp neighbors

%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.45.5 (GigabitEthernet1.45) is down: manually cleared
EIGRP-IPv4(100): table(default): route installed for 155.1.5.0/24 (90/26880256) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 150.1.10.10/32 (90/27008512) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 155.1.108.0/24 (90/26880512) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 150.1.8.8/32 (90/27008256) origin(155.1.0.5)
!
!R4#show logging

Syslog logging: enabled (0 messages dropped, 830 messages rate-limited, 0 flushes, 0 overruns, xml disabled, filteri
```

https://t.me/learningnets

```
No Active Message Discriminator.




No Inactive Message Discriminator.

 Console logging: level debugging
, 676 messages logged, xml disabled,
                      filtering disabled Monitor logging: level informational
, 0 messages logged, xml disabled,
                      filtering disabled Buffer logging:  level debugging
, 838 messages logged, xml disabled,
                     filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled

No active filter modules.

    Trap logging: level informational, 94 message lines logged
        Logging Source-Interface:       VRF Name:
 Log Buffer (8192 bytes):

7008768) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 155.1.13.0/24 (90/26880768) origin(155.1.0.5)
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.146.6 (GigabitEthernet1.146) is down: manually cleared
EIGRP-IPv4(100): table(default): route installed for 155.1.79.0/24 (90/26881024) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 150.1.7.7/32 (90/27008768) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 155.1.7.0/24 (90/26881024) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 155.1.67.0/24 (90/26880768) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 155.1.9.0/24 (90/26881280) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 150.1.6.6/32 (90/27008512) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 155.1.37.0/24 (90/26881024) origin(155.1.0.5)
EIGRP-IPv4(100): table(default): route installed for 150.1.9.9/32 (90/27009024) origin(155.1.0.5)
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.0.5 (Tunnel0) is down: manually cleared
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Syslog Logging

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 and R6 to log messages to syslog as follows:
  - Both R4 and R6 should log to the server 155.1.146.100.
  - Log all messages up to notifications.
  - R4 should use reliable transport on port 5000.
  - R6 should use the default transport.
  - Use ID of **ROUTER4** and **ROUTER6**, respectively, and the UNIX facility **LOCAL1**.
  - Messages should be sourced off of the routers' Loopback0 interfaces.
  - Set the message queue depth to 256.

Configuration

```
R4:
 logging queue-limit trap 256
 logging origin-id string ROUTER4
 logging facility local1
 logging trap notifications
 logging source-interface Loopback0
 logging host 155.1.146.100 transport tcp port 5000
R6:

 logging queue-limit trap 256
```

https://t.me/learningnets

```
logging origin-id string ROUTER6
logging facility local1
logging trap notifications
logging source-interface Loopback0
logging host 155.1.146.100
```

Verification

IOS supports both TCP and UDP to transport syslog messages. UDP is the default, but it does not have reliable delivery like TCP does. The UNIX facility LOCAL1 allows the syslog server to multiplex incoming messages. Multiple logging destinations can be configured using their own transport, but all servers share the same facility and trap level.

```
R6#show logging
Syslog logging: enabled (0 messages dropped, 1 messages rate-limited, 0 flushes, 0 overruns, xml disabled, filtering

No Active Message Discriminator.




No Inactive Message Discriminator.



    Console logging: level debugging, 842 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 843 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled

No active filter modules.
 Trap logging: level notifications
, 837 message lines logged Logging to 155.1.146.100  (udp port 514
, audit disabled,
               link up),
               0 message lines logged,
               0 message lines rate-limited,
               0 message lines dropped-by-MD,
               xml disabled, sequence number disabled
               filtering disabled
```

https://t.me/learningnets

```
Logging Source-Interface:
        VRF Name: Loopback0
```

To trace the packets sent to the syslog server, configure IP packet debugging for an access-list matching the syslog UDP packets. To generate a notification log, go to global configuration and then type **exit**.

```
R6(config)#access-list 100 permit udp any any eq 514
R6#debug ip packet detail 100
IP packet debugging is on (detailed) for access list 100
!
!R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#exit
%SYS-5-CONFIG_I: Configured from console by console
FIBipv4-packet-proc: route packet from (local) src 150.1.6.6 dst 155.1.146.100

FIBfwd-proc: Default:155.1.146.0/24 process level forwarding
FIBfwd-proc: depth 0 first_idx 0 paths 1 long 0(0)
FIBfwd-proc: try path 0 (of 1) v4-con-GigabitEthernet1.146 first short ext 0(-1)
FIBfwd-proc: v4-con-GigabitEthernet1.146 valid
FIBfwd-proc: GigabitEthernet1.146 no nh type 2  - deag
FIBfwd-proc: ip_pak_table 0 ip_nh_table 65535 if GigabitEthernet1.146 nh non
R6#e deag 1 chg_if 0 via fib 0 path type connected prefix
FIBfwd-proc: packet routed to GigabitEthernet1.146 p2p(0)
FIBipv4-packet-proc: packet routing succeeded
IP: tableid=0, s=150.1.6.6 (local), d=155.1.146.100 (GigabitEthernet1.146), routed via FIB
FIBfwd-proc: ip_pak_table 0 ip_nh_table 65535 if GigabitEthernet1.146 nh none uhp 1 deag 0 ttlexp 0
FIBfwd-proc: sending link IP ip_pak_table 0 ip_nh_table 65535 if GigabitEthernet1.146 nh none uhp 1 deag 0 chgif 0 t
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Logging Counting and Timestamps

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 and R6 to count error messages.
- R4 should insert uptime-based timestamps in log messages, whereas R6 should insert date-time based timestamps.
- R6 should add milliseconds-based timestamps to debugging messages only.
- Include year information in logging messages on R6 but not in debugging messages.
- Provide a solution to prevent tampering with stored syslog information.

Configuration

https://t.me/learningnets

```
R4:

service timestamp debug uptime
service timestamps log uptime
service sequence-numbers
logging count
R6:

service timestamps debug datetime msec
service timestamps log datetime year
service sequence-numbers
logging count
```

Verification

Timestamps can be configured separately for debugging and logging messages. They can be based on either the router's uptime or the current date and time. In the latter case, millisecond resolution can also be configured. In the output below, note that debugging message and logging output (such as system events) use different timestamps (logging messages have year information).

```
R6#debug ip eigrp 100
R6#clear ip eigrp neighbors
006984: *May  6 2014 18:08:52: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.146.4 (GigabitEthernet1.146) is dow
006987: *May  6 2014 18:08:57: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.146.4 (GigabitEthernet1.146) is up:
!
!R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#exit
007063: *May  6 2014 18:12:27: %SYS-5-CONFIG_I: Configured from console by console
```

Compare the above debugging output with the debugging output on R4, which has timestamps based on uptime.

https://t.me/learningnets

```
R4#debug ip eigrp 100
R4#clear ip eigrp neighbors

!
!
002138: 4d23h: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.45.5 (GigabitEthernet1.45) is up: new adjacency
002178: 4d23h: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 155.1.146.1 (GigabitEthernet1.146) is up: new adjacency
```

Note that all above debugging/logging lines have sequence numbers. This helps prevent tampering with the stored syslog message. Error message counting allows counting of all notification and above system messages, to provide statistics for quick analysis of the system history.

```
R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface GigabitEthernet1.146
R6(config-if)#shutdown
R6(config-if)#no shutdown
!
!R6#show logging count

Facility       Message Name                     Sev Occur      Last Time
==================================================================================
SYS            CONFIG_I                          5     3 *May  6 2014 18:21:32

-------------  ------------------------------  ----------------------------------
SYS TOTAL                                            3
DUAL           NBRCHANGE                         5    10 *May  6 2014 18:21:31

-------------  ------------------------------  ----------------------------------
DUAL TOTAL                                          10
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Logging to Flash Memory

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R7 to log notification messages in its internal flash memory.
- Create the directory syslog and store all the log files into it.
- Set the maximum file size to 32768 bytes.

Configuration

```
R7#
mkdir bootflash:/syslog
Create directory filename [syslog]?
Created dir bootflash:/syslog
R7:

logging persistent filesize 32768
logging persistent url bootflash:/syslog/
logging on
```

Verification

Verify that persistent logging has been configured:

```
R7#show logging
```

https://t.me/learningnets

```
Syslog logging: enabled (0 messages dropped, 7 messages rate-limited, 0 flushes, 0 overruns, xml disabled, filtering

No Active Message Discriminator.




No Inactive Message Discriminator.



    Console logging: level debugging, 856 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 862 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
Persistent logging: enabled, url bootflash:/syslog/, disk space 783561932 bytes, file size 262144 bytes, batch size



No active filter modules.

    Trap logging: level informational, 858 message lines logged
        Logging Source-Interface:       VRF Name:

Log Buffer (4096 bytes):
```

Verify that log files have been created in flash:

https://t.me/learningnets

```
R7#dir | i syslog
64897  drwx             4096  Aug 28 2014 04:52:00 +00:00  syslog

!
!R7#dir bootflash:/syslog/

Directory of bootflash:/syslog/
64898  -rw-              719  Aug 28 2014 04:51:26 +00:00  log_20140828-045126

64899  -rw-              203  Aug 28 2014 04:51:49 +00:00  log_20140828-045149
64900  -rw-              334  Aug 28 2014 04:52:00 +00:00  log_20140828-045200

7835619328 bytes total (5985144832 bytes free)
!
!R7#more bootflash:/syslog/log_20140828-045126

%SYS-5-LOG_CONFIG_CHANGE: Persistent logging: enabled, url bootflash:/syslog, disk space 783561932 bytes, file size
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Configuration Change Notification and Logging

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 to locally track changes made to its running configuration.
- Log these changes via syslog, but ensure that passwords in the configuration will not be sent across this communication channel.
- Set the queue size to 1000.

Configuration

```
R4:

archive
 log config
  logging enable
  logging size 1000
  notify syslog
  hidekeys
```

Verification

Change notification and logging is a simple alternative to AAA accounting. This feature allows you to track configuration changes along with the user who made them. The primary purpose is to log the changes to a remote syslog server for

https://t.me/learningnets

further archiving and analysis, but it is also possible to review the local changes queue.

```
R4#configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface gigabitEthernet 1.146
R4(config-subif)#shutdown
002368: 5d01h: %PARSER-5-CFGLOG_LOGGEDCMD: User:console  logged command:interface GigabitEthernet1.146
002369: 5d01h: %PARSER-5-CFGLOG_LOGGEDCMD: User:console  logged command:shutdown
R4(config-subif)#no shutdown
002372: 5d01h: %PARSER-5-CFGLOG_LOGGEDCMD: User:console  logged command:no shutdown
```

You may review the changes made by all users, or just a specified user. The statistics option displays the number of user sessions tracked and the memory usage:

https://t.me/learningnets

```
R4#show archive log config all

idx   sess          user@line      Logged command
1     1        console@console  |  logging enable
2     1        console@console  |  logging size 1000
3     1        console@console  |  notify syslog
4     1        console@console  |  hidekeys
5     1        console@console  |  exit
6     1        console@console  |   exit
7     2        console@console  |interface GigabitEthernet1.146

8     2        console@console  | shutdown
9     2        console@console  | no shutdown
!
!R4#show archive log config statistics

Config Log Session Info:
    Number of sessions being tracked: 1
    Memory being held: 3934 bytes
    Total memory allocated for session tracking: 3934 bytes
    Total memory freed from session tracking: 0 bytes

Config Log log-queue Info:
    Number of entries in the log-queue: 9
    Memory being held by the log-queue: 3616 bytes
    Total memory allocated for log entries: 3616 bytes
    Total memory freed from log entries: 0 bytes
```

You may also display the changes in a format suitable to direct the application to a running IOS router (for example, to replay the changes):

https://t.me/learningnets

```
R4#show archive log config all provisioning

 archive
 log config
  logging enable
  logging size 1000
  notify syslog
  hidekeys
  exit  interface GigabitEthernet1.146
  shutdown
  no shutdown
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Configuration Archive and Rollback

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R7 to store its configuration archive on the TFTP server 155.1.58.100 using **R7-config** as the prefix.
  - Archive the configuration every 24 hours.
  - The configuration should be archived every time the running config is saved to NVRAM.

Configuration

```
R7:

archive
 path tftp://155.1.58.100/R7-config
 write-memory
 time-period 1440
```

Verification

https://t.me/learningnets

A router can save its configuration snapshots to a network path or to a secondary non-volatile storage plugged into the router (such as disk0), but not to its primary flash memory storage. When saving the configuration to a network path, the maximum number of copies cannot be specified:

```
R7#show archive
The maximum archive configurations allowed is 10.
The next archive file will be named tftp://155.1.58.100/R7-config-<timestamp>-1
Archive #  Name 1        :File is being written, please wait <- Most Recent


2
3
4
5
6
7
8
9
10
```

Configuration archiving is very useful in combination with two other features: configuration rollback and contextual diff utility. To see how they work, manually create a copy of the running-configuration in flash, and then make some changes to the running configuration:

```
R7#copy running-config flash:/saved-config
Destination filename [/syslog/saved-config]?
2031 bytes copied in 0.173 secs (11740 bytes/sec)
!
!R7#configure terminal


Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.67
R7(config-if)#shutdown
R7(config-if)#ip address 155.1.76.7 255.255.255.0
```

Now list the differences between the two configurations, the saved one and the running one. The “+” sign means that the configuration is present in the second configuration but missing in the first file. The “-“ sign means the configuration line is present in the first file but missing in the second one.

```
R7#show archive config differences flash:saved-config


 !Contextual Config Diffs:
```

https://t.me/learningnets

```
 interface GigabitEthernet1.67
 +ip address 155.1.67.7 255.255.255.0
 interface GigabitEthernet1.67
 -ip address 155.1.76.7 255.255.255.0
 -shutdown
```

The below output displays the lines that must be added to the running configuration to make it look like the saved configuration:

```
R7#show archive config incremental-diffs flash:saved-config

!List of Commands:
interface GigabitEthernet1.67
 ip address 155.1.67.7 255.255.255.0
 end
```

Use the **`configure replace`** command to roll back the running configuration to the configuration stored in flash. This procedure will ensure that the configurations are identical, and it will even un-shutdown the interfaces.

```
R7#configure replace flash:saved-config force
Total number of passes: 1 Rollback Done
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Logging with Access-Lists

> You must load the initial configuration files for the section, **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 to log every EIGRP packet received on its VLAN 146 interface.
  - Generate a cumulative log entry for every two matched packets.
  - Send a logging message no more than once every 10 seconds.

Configuration

```
R4:

ip access-list extended LOGGING
 permit eigrp any any log
 permit ip any any
!
interface GigabitEthernet1.146
 ip access-group LOGGING in
!
ip access-list logging interval 10000
ip access-list log-update threshold 2
```

Verification

The `log-input` option logs the MAC address of the device that sent the packet, in

https://t.me/learningnets

addition to the packet information. The update threshold set the minimum number of hits needed to generate a logging message, and effectively aggregates the hits. The logging interval specifies the minimum interval of log message generation, essentially providing a rate-limiting capability.

In the output below, R1's connection to VLAN 146 is disabled to show only EIGRP packets that R4 receives from R6. Note that a log message is generated approximately every 20 seconds, as R6 is configured to send updates every 10 seconds, and the logging configuration aggregates the ACL hits in pairs.

```
R1#configure terminal
R1(config)#inteface GigabitEthernet1.146
R1(config-if)#shutdown
!
!R4#
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.6 GigabitEthernet1.146-> 224.0.0.10
2 packets
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.1 GigabitEthernet1.146-> 224.0.0.10
2 packets
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.6 GigabitEthernet1.146-> 224.0.0.10
2 packets
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.6 GigabitEthernet1.146-> 224.0.0.10
2 packets
```

Now change the update count to 1 so that every packet hit generates a message. Note that logging messages are generated every 10 seconds now.

```
R4(config)#ip access-list log-update threshold 1
!
!
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.6 GigabitEthernet1.146-> 224.0.0.10
1 packet
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.6 GigabitEthernet1.146-> 224.0.0.10
1 packet
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.6 GigabitEthernet1.146-> 224.0.0.10
1 packet
%FMANFP-6-IPACCESSLOGNP: F0: fman_fp_image:  list LOGGING permitted 88 155.1.146.6 GigabitEthernet1.146-> 224.0.0.10
1 packet
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

TCP Keepalives

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 so that it is capable of detecting dead TCP control-plane connections and closing them prematurely.

Configuration

```
R1:

service tcp-keepalives-out
service tcp-keepalives-in
```

Verification

The TCP keepalive feature is useful for probing idle connections to see if they are still active. In some cases, such as when the exec-timeout feature is disabled on VTY lines, this helps to prevent resource starvation by freeing connections that are no longer active.

To test this, initiate a connection from R6 to R1 and verify its parameters. Note the keepalive fields in the TCP Connection Block output.

```
R6#telnet 155.1.146.1
```

https://t.me/learningnets

```
Trying 155.1.146.1 ... OpenR1#

!
!
R1#show tcp brief
TCB       Local Address               Foreign Address             (state)
7F912D553CC8  155.1.146.1.23             155.1.146.1.12289           ESTAB
!
!R1#show tcp tcb 7F912D553CC8
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 155.1.146.1, Local port: 23
Foreign host: 155.1.146.1, Foreign port: 12289
Connection tableid (VRF): 0
Maximum output segment queue size: 20

Enqueued packets for retransmit: 0, input: 0  mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x60E75EC):
Timer          Starts    Wakeups            Next
Retrans             5          0             0x0
TimeWait            0          0             0x0
AckHold             4          0             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0
Linger              0          0             0x0
ProcessQ            0          0             0x0

iss: 1086457305  snduna: 1086457341  sndnxt: 1086457341
irs:  530036474  rcvnxt:  530036511

sndwnd:   4093  scale:      0  maxrcvwnd:   4128
rcvwnd:   4092  scale:      0  delrcvwnd:     36

SRTT: 487 ms, RTTO: 3168 ms, RTV: 2681 ms, KRTT: 0 ms
minRTT: 1 ms, maxRTT: 1000 ms, ACK hold: 200 ms
uptime: 69977 ms, Sent idletime: 69973 ms, Receive idletime: 69771 ms
Status Flags: passive open, active openOption Flags: keepalive running, Retrans timeout

IP Precedence value : 6

Datagrams (max data segment is 1460 bytes):
Rcvd: 13 (out of order: 0), with data: 6, total data bytes: 36
```

https://t.me/learningnets

```
Sent: 11 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 7, total data bytes: 35

 Packets received in fast path: 0, fast processed: 0, slow path: 0
 fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7F912CF679C0  FREE
```

Debug TCP transactions on R1, and then shutdown R6's VLAN 146 interface to simulate a dead connection, without disconnecting from the telnet session. Note that after four keepalives have been missed, R1 forcefully closes the idle connection:

```
R1#debug ip tcp transactions
TCP special event debugging is on
!R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface GigabitEthernet1.146
R6(config-if)#shutdown
!R1#
TCP1: keepalive timeout (1/4)
TCP1: keepalive timeout (2/4)
TCP1: keepalive timeout (3/4)
TCP1: keepalive timeout (4/4)TCP1: state was ESTAB -> CLOSED [23 -> 155.1.146.6(43722)]
TCB 0x7F912D553CC8 destroyed
!R1#show tcp tcb 7F912D553CC8
TCB 0x7F912D553CC8 not found
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Telnet Service Options

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R3 to source all telnet sessions from its Loopback0 interface, and to use a marking of IP Precedence 3 for these packets.
  - Do not display any telnet informational messages.
  - Create a hostname entry **R4** that points to the IP address of R4 on VLAN 146.
  - Ensure that a user typing **R4** in exec mode is connected without seeing the IP address of R4.
  - Display the message **Sorry, your connection failed** when a telnet connection to the above host fails.

Configuration

R3:

```
service telnet-zeroidle
ip telnet source-interface Loopback0
ip telnet tos 60
ip telnet quiet
ip telnet hidden addresses
!
no ip domain-lookup
ip host R4 155.1.146.4
```

https://t.me/learningnets

```
!
busy-message R4 # Sorry, your connection failed #
```

Verification

The telnet client in IOS can be tuned to provide additional “obfuscation” settings, which hides information from the user initiating the session. Note how the below changes when the “quiet” and “hidden address” features are disabled. With both features on, telnet never displays any progress or informational messages and does not report the IP address of the host it connects to:

```
R3#R4
Translating "R4"
R4#
!
!R3#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R3(config)#no ip telnet quiet
R3(config)#do telnet R4
Translating "R4"Trying R4 address #1 ... Open
R4#
!
!R3(config)#no ip telnet hidden address
R3(config)#do telnet R4

Translating "R4"Trying R4 (155.1.146.4)... Open
R4#
```

To verify that the TOS byte settings are correct, telnet to R3 from R4. Note that the TOS value is entered in HEX format in the configuration:

```
R4#telnet 150.1.3.3
Trying 150.1.3.3 ... Open
!R3#show tcp brief
TCB        Local Address               Foreign Address             (state)
7F7337565D18  150.1.3.3.23                R4.28754                    ESTAB
!
!R3#show tcp tcb 7F7337565D18

Stand-alone TCP connection to host 155.1.146.4
Connection state is FINWAIT2, I/O status: 1, unread input bytes: 0
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 150.1.3.3, Local port: 36029
Foreign host: 155.1.146.4, Foreign port: 23
```

https://t.me/learningnets

```
Connection tableid (VRF): 0
Maximum output segment queue size: 50

Enqueued packets for retransmit: 0, input: 0  mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x62B85E5):
Timer          Starts    Wakeups            Next
Retrans             9          0             0x0
TimeWait            1          0       0x62FC0CF
AckHold            12          1             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0
Linger              0          0             0x0
ProcessQ            0          0             0x0

iss: 2762911647  snduna: 2762911693  sndnxt: 2762911693
irs: 1523076971  rcvnxt: 1523077021

sndwnd:   4084  scale:      0  maxrcvwnd:   4128
rcvwnd:      0  scale:      0  delrcvwnd:      0

SRTT: 699 ms, RTTO: 2656 ms, RTV: 1957 ms, KRTT: 0 ms
minRTT: 2 ms, maxRTT: 1000 ms, ACK hold: 200 ms
uptime: 330134 ms, Sent idletime: 13924 ms, Receive idletime: 13924 ms
Status Flags: active open, App closed
Option Flags: idle user, Retrans timeout IP Precedence value : 3



Datagrams (max data segment is 536 bytes):
Rcvd: 33 (out of order: 0), with data: 20, total data bytes: 49
Sent: 31 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 7, total data bytes: 44

 Packets received in fast path: 0, fast processed: 0, slow path: 0
 fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7F733697BAE0  FREE
```

To verify the busy message settings, shutdown the interface connecting to VLAN 146 on R4, and then try connecting to R4 from R3. Note that the busy-message can be set globally as well as per VTY line, and it is always bound to a target hostname.

```
R4#configure terminal
```

https://t.me/learningnets

```
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.146
R4(config-if)#shutdown


!
!R3#R4
Translating "R4" Sorry, your connection failed
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Tuning Packet Buffers

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Enable automatic buffer tuning on R3.
- Configure R4 to double the permanent number of all public (not interface-specific) buffers.
- Ensure that there are at least 10 Large and Huge buffers always available.

Configuration

```
R3:
buffers tune automatic
!R4:

buffers small permanent 100
buffers middle permanent 50
buffers big permanent 100
buffers verybig permanent 20
buffers large permanent 10
buffers huge permanent 10
```

Verification

Buffers represent chunks of space in the router's I/O memory. Each interface may

https://t.me/learningnets

have its own private buffer pool, and all interfaces have access to public buffer pools. Buffers are used to store incoming packets, but because packets may vary in size, different groups of buffer sizes exist. Having dynamically sized buffers is inefficient in terms of speed and memory fragmentation. Having pre-allocated buffers may speed up incoming packet processing but consumes additional memory. Therefore, tuning the buffer space is a trade-off between speed and resource usage.

Sometimes it is necessary to tune buffers manually based on their historical demand. However, with recent IOS versions it is possible to enable automatic buffer tuning, which changes the number of pre-allocated buffers based on an adaptive learning algorithm. Note that for each buffer group it is possible to specify the upper and lower bounds of the buffer count, as well as the permanent and initial buffer allocation. Use the `show buffer` command before applying any new settings to observe the historical statistics and the current settings. A buffer “hit” means that a buffer was available for use when a packet arrived, and a “miss” means that the IOS had to allocate a new buffer on demand for the packet.

```
R4#show buffers

 Buffer elements:
 1076 in free list
 1406097 hits, 0 misses, 1019 created

 Public buffer pools: Small buffers, 104 bytes (total 202, permanent 100, peak 1200 @ 5d23h):
 201 in free list (200 min, 2500 max allowed)
 779229 hits, 0 misses, 0 trims, 102 created
 0 failures (0 no memory) Middle buffers, 600 bytes (total 105, permanent 50, peak 3446 @ 3d04h):)
 104 in free list (100 min, 2000 max allowed)
 1652991 hits, 1208 misses, 3614 trims, 3669 created
 0 failures (0 no memory) Big buffers, 1536 bytes (total 100, permanent 100, peak 910 @ 5d23h):
 100 in free list (50 min, 1800 max allowed)
 357532 hits, 10 misses, 10 trims, 10 created
 0 failures (0 no memory) VeryBig buffers, 4520 bytes (total 20, permanent 20, peak 101 @ 5d23h):
 20 in free list (0 min, 300 max allowed)
 103945 hits, 0 misses, 1 trims, 1 created
 0 failures (0 no memory) Large buffers, 5024 bytes (total 10, permanent 10, peak 101 @ 5d23h):
 10 in free list (0 min, 300 max allowed)
 1 hits, 0 misses, 1 trims, 1 created
 0 failures (0 no memory)
 VeryLarge buffers, 8248 bytes (total 100, permanent 100):
 100 in free list (0 min, 300 max allowed)
 371 hits, 0 misses, 0 trims, 0 created
 0 failures (0 no memory) Huge buffers, 18024 bytes (total 10, permanent 10, peak 21 @ 5d23h):
```

https://t.me/learningnets

```
10 in free list (0 min, 33 max allowed)
0 hits, 0 misses, 1 trims, 1 created
0 failures (0 no memory)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Terminal Line Settings

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R3 to allow only telnet connections to VTY lines 0 through 4 without using an access-list.
  - When a user is telnetted into R3 and mistypes a command in exec mode, R3 should not try to open a telnet session to the command as if it is a hostname.
  - Configure VTY line 0 to listen for telnet at port 3001.
  - When the virtual terminal line is busy, issue the output **Sorry, the line is already in use** to the connecting user.
  - Exec sessions on a VTY line should timeout after 2 minutes of inactivity; a user should not be able to hold the line busy for more than 5 minutes.
  - The terminal length should be no more than 20 lines.
  - IP netmasks should be displayed using hex numbers.
  - Allow a user to lock VTY terminal lines.
  - Sessions initiated from a VTY line should timeout in 1 minute.
- When the console line is idle, the user should see the output **Welcome to IOS**.
  - Allow no more than 1 session to be initiated from the console line.

Configuration

R3:

https://t.me/learningnets

```
line vty 0 4
 session-timeout 1
 exec-timeout 2 0
 lockable
 absolute-timeout 5
 ip netmask-format hexadecimal
 refuse-message # Sorry, the line is already in use #
 length 20
 transport preferred none
 transport input telnet
!
line vty 0
 rotary 1
!
line console 0
 session-limit 1
 vacant-message # Welcome to IOS #
```

Verification

IOS supports multiple timeout settings for terminal sessions, which can be divided as follows. *Absolute* limits the maximum amount of time the user can spend on the line. *Exec* limits the time an exec shell can be idle. *Session* is the maximum amount of time a session opened from terminal (such as telnet to other router) can be idle. A refuse message is displayed when someone tries to connect to a line already in use. The vacant message is displayed when the current line is idle (not in use—no exec shell started).

The transport input/output option specifies which protocols can be used to connect to/from the terminal line. The preferred transport protocol is used when no session protocol is specified at exec prompt and a station name is typed. By default, the preferred transport is telnet, which is why the router tries to telnet when a command is mistyped. The lock feature allows a user to lock the current terminal session and require a password to unlock it.

```
R4#show line vty 0
 Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns   Int
 *     2 VTY              -    -      1    -    -      6       0     0/0       -

Line 2, Location: "", Type: ""Length: 20 lines, Width: 80 columns
Baud rate (TX/RX) is 9600/9600
Status: PSI Enabled, Ready, Connected, Active, No Exit BannerCapabilities: Lockable
Modem state: Ready
Group codes:    0
```

https://t.me/learningnets

```
Special Chars: Escape  Hold  Stop  Start  Disconnect  Activation
             ^^x    none   -     -       none
Timeouts:      Idle EXEC    Idle Session   Modem Answer  Session   Dispatch
00:02:00       00:01:00                    00:05:00    not set
                         Idle Session Disconnect Warning
                           never
                         Login-sequence User Response
                          00:00:30
                         Autoselect Initial Wait
                           not set
Modem type is unknown.
Session limit is not set.
Time since activation: 00:39:08
Editing is enabled.
History is enabled, history size is 10.
DNS resolution in show commands is enabled
Full user help is disabled Allowed input transports are telnet.
Allowed output transports are lat pad telnet rlogin mop ssh nasi. Preferred transport is none.

Shell: disabled
Shell trace: off
No output characters are padded
No special data dispatching characters
```

The netmask format can be verified as shown below.

```
R3#telnet 150.1.3.3
Trying 150.1.3.3 ... Open
!
!R3#show interface GigabitEthernet1.37
Serial1/0 is up, line protocol is up
  Hardware is CD2430 in sync mode Internet address is 155.1.0.3 0xFFFFFF00

  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation FRAME-RELAY, loopback not set
```

Rotary groups allow bundling multiple lines into a pool and to give the option to access the pool using the dedicated TCP port number 3000+N, where *N* is the rotary group number. Those special port numbers can also be used as “backdoors” for telnet access. Note that the refuse message is displayed when a user attempts to connect to the busy line.

```
R3#telnet 150.1.3.3 3001
```

https://t.me/learningnets

```
Trying 150.1.3.3, 3001 ... Open
!
!R3#telnet 150.1.3.3 3001
Trying 150.1.3.3, 3001 ... Open
Sorry, the line is already in use



[Connection to 150.1.3.3 closed by foreign host]
```

The console session limit can be verified as follows:

```
R3#telnet 150.1.2.2
Trying 150.1.2.2 ... OpenR2#
!
!R3#telnet 150.1.5.5
Session limit exceeded
!
!R3#where

Conn Host                Address             Byte  Idle Conn Name
*  1 150.1.2.2           150.1.2.2              0     0 150.1.2.2
```

To verify session locking, log in in via telnet and issue the lock command:

```
R3#telnet 150.1.3.3
Trying 150.1.3.3 ... Open
!R3#lock
Password:
Again:

<snip>
Locked
```

Verify that setting the preferred transport to “none” disables automatic telnet sessions opening for hostnames entered in the CLI as follows:

https://t.me/learningnets

```
R3#telnet 150.1.3.3

Trying 150.1.3.3 ... Open
R3#hostname

    ^% Invalid input detected at '^' marker.
```

Check the vacant message for the console line as follows:

```
R3#exit

<snip>
Welcome to IOS
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

SNMPv2 Server

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SNMP on R4 with the SNMP contact set to **Default Contact** and the SNMP Location set to **Default Location**.
- Enable the SNMP server service using the community string value of **CISCO** so that remote hosts can modify the local MIB database.
  - Allow the system to be reloaded via SNMP.
  - Ensure that interface index numbers persist between reloads.
  - Only allow configuration transfers via TFTP to/from the host 155.1.146.100.

Configuration

https://t.me/learningnets

```
R4:

snmp-server community CISCO RW
snmp-server location Default Location
snmp-server contact Default Contact
snmp-server ifindex persist
snmp-server system-shutdown
!
access-list 98 permit 155.1.146.100
snmp-server tftp-server-list 98
```

Verification

By default, when you configure the SNMP agent with a community string, SNMPv2c is used. This version uses only the community string for authentication. Issuing the **`snmp-server community`** command is the minimal configuration needed to enable the devices to be polled via SNMP. The read-write string allows the management station to make changes to the managed device, as opposed to the read-only string. SNMP interface index (IfIndex) values are not the same between reloads by default. This may confuse some network management systems (NMS), because interfaces are identified by the IfIndex value in the SNMP Management Information Base (MIB). To resolve this, the SNMP IfIndex persistence feature may be helpful.

To allow the NMS to reload the managed device, read-write access must be allowed, and the **`snmp-server system-shutdown`** feature must be enabled. The TFTP server list specifies the acceptable addresses to download/upload the router's configuration when instructed via SNMP. In recent IOS versions, the command has been replaced with the command **`snmp-server file-transfer access-group`** . The IOS still accepts the old command, but converts it to the new format.

```
R4#show snmp

Chassis: 9YQUKL7BRNCContact: Default Contact
Location: Default Location
    0 SNMP packets input
    0 Bad SNMP version errors
    0 Unknown community name
    0 Illegal operation for community name supplied
    0 Encoding errors
    0 Number of requested variables
    0 Number of altered variables
    0 Get-request PDUs
    0 Get-next PDUs
```

https://t.me/learningnets

```
    0 Set-request PDUs
    0 Input queue packet drops (Maximum queue size 1000)
0 SNMP packets output
    0 Too big errors (Maximum packet size 1500)
    0 No such name errors
    0 Bad values errors
    0 General errors
    0 Response PDUs
    0 Trap PDUs
!
!R4#show snmp community

Community name: ILMI
Community Index: cisco0
Community SecurityName: ILMI
storage-type: read-only  active

Community name: CISCO
Community Index: cisco3
Community SecurityName: CISCO
storage-type: nonvolatile    active
!
!R4#show snmp mib ifmib ifindex

GigabitEthernet1.58: Ifindex = 8
Tunnel0: Ifindex = 10
GigabitEthernet1: Ifindex = 1
GigabitEthernet3: Ifindex = 3
GigabitEthernet1.45: Ifindex = 7
GigabitEthernet1.100: Ifindex = 9
GigabitEthernet1.146: Ifindex = 11
Loopback0: Ifindex = 5
Null0: Ifindex = 4
GigabitEthernet2: Ifindex = 2
GigabitEthernet1.5: Ifindex = 6
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

SNMPv2c Access Control

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Enable the SNMP server service using the community string value of **CISCO** so that remote hosts can modify the local MIB database.
  - Configure R4 to restrict read-write access to only hosts in VLAN 146.
  - Log any other SNMP access attempts using the community string **CISCO**.
- Allow any host to access R4's MIBs in read-only mode using the community string **PUBLIC**.
- Read-only access should be permitted to the "cisco" MIB subtree only.

Configuration

```
R4:

access-list 99 permit 155.1.146.0 0.0.0.255
access-list 99 deny any log
!
snmp-server community CISCO RW 99
snmp-server view ROVIEW cisco included
snmp-server community PUBLIC view ROVIEW ro
```

https://t.me/learningnets

Verification

It is highly advisable to limit access to managed SNMP devices by associating an access-list with the SNMP server community. By using the log keyword, it is possible to expose other hosts that attempt to access the router via SNMP. Additionally, SNMPv2 allows the restriction of access to certain MIB values using the concept of views. A view is partition of the whole MIB tree built by including or excluding various subtrees. Note that view definition may have multiple include/exclude statements, and the latter ones take preference.

```
R4#show snmp community

Community name: CISCO
Community Index: CISCO
Community SecurityName: CISCO storage-type: nonvolatile active access-list: 99



Community name: ILMI
Community Index: ILMI
Community SecurityName: ILMI
storage-type: read-only  active

Community name: PUBLIC
Community Index: PUBLIC
Community SecurityName: PUBLIC storage-type: nonvolatile         active
!
!R4#show snmp view
*ilmi system - included permanent active
*ilmi atmForumUni - included permanent active ROVIEW cisco - included nonvolatile active

cac_view pimMIB - included read-only active
cac_view msdpMIB - included read-only active
cac_view interfaces - included read-only active
cac_view ip - included read-only active
cac_view ospf - included read-only active
cac_view bgp - included read-only active
cac_view ifMIB - included read-only active
cac_view nhrpMIB - included read-only active
cac_view ipMRouteStdMIB - included read-only active
cac_view igmpStdMIB - included read-only active
cac_view ospfv3MIB - included read-only active
cac_view ipForward - included read-only active
cac_view ipTrafficStats - included read-only active
cac_view ospfTrap - included read-only active
```

https://t.me/learningnets

```
cac_view sysUpTime.0 - included read-only active
cac_view mplsLsrStdMIB - included read-only active
cac_view mplsLdpStdMIB - included read-only active
cac_view ciscoPingMIB - included read-only active
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

SNMP Traps and Informs

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 to send SNMP traps to the management host 155.1.146.100 and informs to the host 155.1.146.101.
  - Use the community value of **CISCO** with the above traps and informs.
  - Only send notifications about interfaces going up or down.
  - Ensure that no informs are being sent about R4's VLAN 146 interface changing its status.

Configuration

```
R4:

snmp-server enable traps snmp linkdown linkup
snmp-server host 155.1.146.101 inform version 2c CISCO
snmp-server host 155.1.146.100 CISCO
!
interface GigabitEthernet1.146
 no snmp trap link-status
```

Verification

SNMP traps are part of SNMPv1 and SNMPv2 specifications. Additionally, SNMPv2

https://t.me/learningnets

allows sending notifications as informs, which differ from traps in that they require acknowledgement from the NMS. Informs are kept in router local queue until they are acknowledged or timeout has expired. Informs make SNMP reliable even though the transport protocol is still UDP.

The traps that are to be sent can be configured either globally or on a per-host basis. In the latter case, the host settings override the global settings. Some traps, however, can only enabled globally, such as link up and down notifications. If the single command **`snmp-server enable traps`** is entered, all traps will be sent to all configured destinations, unless they are overridden on a per-host basis. Note that *it is required* to configure both the **`snmp-server enable traps`** and the **`snmp-server host`** commands to send notifications to a particular host. Informs sending is enabled by default, and the same notifications configured with the global **`snmp-server enable traps`** command are sent to NMS hosts, provided that they are configured for informs. The same host may receive informs and traps in parallel, although this configuration is uncommon.

```
R4#show snmp host
 Notification host: 155.1.146.101        udp-port: 162   type: inform
user: CISCO     security model: v2c
 Notification host: 155.1.146.100        udp-port: 162   type: trap
user: CISCO     security model: v1
```

To verify SNMP trapping, enable SNMP packet debugging and create a link down event:

```
R4#debug snmp packet
SNMP packet debugging is on
!
!R4#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.45
R4(config-if)#snmp trap link-status
R4(config-if)#shutdown
!
!SNMP: Inform request, reqid 17, errstat 0, erridx 0

 sysUpTime.0 = 2019043
 snmpTrapOID.0 = snmpTraps.3
 ifIndex.1 = 1
 ifDescr.1 = Gigabitethernet1.45
 ifType.1 = 6  lifEntry.20.1 = administratively down
Packet sent via UDP to155.1.146.101.162
SNMP: Queuing packet to 155.1.146.100
SNMP: V1 Trap, ent snmpTraps, addr 155.1.146.4, gentrap 2, spectrap 0
```

https://t.me/learningnets

```
 ifIndex.1 = 1
 ifDescr.1 = GigabitEthernet1.45
 ifType.1 = 6  lifEntry.20.1 = administratively down
Jul 18 02:24:36.590: SNMP: Packet sent via UDP to 155.1.146.100
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

CPU and Memory Thresholds

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 to monitor CPU usage every 5 seconds using the `process cpu` command, and to generate a rising threshold event every time the CPU usage hits 50%.
- Set the free memory low threshold to 1000 Kbytes, and reserve 512 Kbytes of memory for the notification process using the `memory` command.

Configuration

```
R4:

snmp-server enable traps cpu threshold
!
memory free low-watermark processor 1000
process cpu threshold type total rising 50 interval 5
memory reserve critical 512
```

Verification

CPU threshold notification allows sending SNMP traps and/or informs when the CPU usage crosses defined boundaries. A trap could be generated on the CPU value crossing the rising threshold (going above) or the falling threshold (going

https://t.me/learningnets

below). Additionally, the memory threshold notification feature will post a syslog message when the amount of free memory shrinks below a defined limit. To verify CPU threshold tracking, enable SNMP packet debugging and set the CPU rising threshold to a very low value. Then execute the `show run` command to cause a CPU spike.

```
R4#debug snmp packet
!R4#configure terminal
R4(config)#process cpu threshold type total rising 5 interval 5
R4#show run
Building configuration...

Current configuration : 2493 bytes
....

%SYS-1-CPURISINGTHRESHOLD: Threshold: Total CPU Utilization(Total/Intr): 8%/4%, Top 3 processes(Pid/Util):  3/2%, 8/
```

To verify free memory threshold notifications, determine the current amount of free processor memory, set the threshold to a high value, and then set it to a lower value:

```
R4#show memory summmary
            Head          Total(b)      Used(b)     Free(b)     Lowest(b)       Largest(b)
Processor  7F51FD3A7010   833974896   256056156 577918740
   521221784   276235252
lsmpi_io  7F51FCC9D1A8     6295128     6294304         824         824         412
`Critical   7F521E70DE28      524292          92      524200      524200      524200
!
!R4(config)#memory free low-watermark processor 58000
%SYS-4-FREEMEMLOW: Free Memory has dropped below low watermark
!
!R4(config)#memory free low-watermark processor 6000
%SYS-5-FREEMEMRECOVER: Free Memory has recovered above low watermark
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

SNMPv3

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Create two SNMP views named **NORMAL** and **RESTRICTED** on R6.
  - The **NORMAL** view should include the branch “iso”.
  - The **RESTRICTED** view should include the branch “ifEntry.*.n”, where **n** is the ifIndex of R6’s Null0 interface.
- Create an SNMP group named **NORMAL** with the read/write view assigned to **NORMAL** and a security model of “priv”.
  - Assign the user named **NORMAL** to this group, set the SHA1 password and encryption key to **CISCO**.
- Create an SNMP group named **RESTRICTED** with the read view assigned to **RESTRICTED** and a security model of “auth”.
  - Only users from VLAN 146 should be allowed in the SNMP group **RESTRICTED**.
  - Assign the user named **RESTRICTED** to this group using the security model that only requires authentication using the password **CISCO**.
- Create an SNMP group named **TRAP** with the security model “priv”.
  - Assign the user named **TRAP** to this group using the password and encryption key **CISCO**.
- Enable SNMP traps for LinkUp and LinkDown events only, and send them to the destination host 155.1.146.100 using the security model “priv” and the username **TRAP**.

https://t.me/learningnets

- Use SHA for authentication and 3DES for encryption.

Configuration

```
R6:

access-list 99 permit 155.1.146.0 0.0.0.255
!
snmp-server ifindex persist
snmp-server view NORMAL iso included
snmp-server view RESTRICTED ifEntry.*.3 included
!
snmp-server group NORMAL v3 priv read NORMAL write NORMAL
snmp-server group RESTRICTED v3 auth read RESTRICTED access 99
snmp-server group TRAP v3 priv
!
snmp-server user NORMAL NORMAL v3 auth sha CISCO priv 3des CISCO
snmp-server user RESTRICTED RESTRICTED v3 auth sha CISCO
snmp-server user TRAP TRAP v3 auth sha CISCO priv 3des CISCO
!
snmp-server enable traps snmp linkup linkdown
snmp-server host 155.1.146.100 traps version 3 priv TRAP
```

Verification

SNMPv3 extends the previous versions of SNMP by introducing a new security model that replaces the old community-based authentication system. SMNPv3 also provides for communication privacy by means of encryption. The new concepts for SNMPv3 are the user, group, and security level.

A group defines what access rights a set of users has. This access policy controls which SNMP objects (MIBs) can be accessed for reading and writing, or which SNMP objects can generate notifications to the members of a group. The policy is defined by associating a read, write, or notify view with the group. By using a notify view, a group determines the list of notifications that its users can receive. The group also defines the security *model* (SNMP version) and the security *level* (authentication and/or encryption) for its users.

If a group is defined without a *read* view, all objects are available to be read (implicit permit). Contrary to that, if a *write* or *notify* view is not defined, no write access is granted, and no objects can send notifications to members of the group (implicit deny). The notify view is usually not configured manually, and is auto-generated by

https://t.me/learningnets

the `snmp-server host` command when users in a group are bound to a notification target host. The security *models* are defined as SNMPv1, SNMPv2, and SNMPv3, and the security *levels* are defined as noAuthNoPriv, AuthNoPriv, and AuthPriv. noAuthNoPriv, the `noauth` keyword in the IOS, means no authentication and no encryption. AuthNoPriv, the `auth` keyword in the IOS, means authentication but no encryption. AuthPriv, the `priv` keyword in IOS, means authentication and encryption.

SNMPv3 can implement any of the three above security levels. SNMPv1 and SNMPv2 only support noAuthNoPriv. In the case that SNMPv3 uses noAuthNoPriv, the username serves as a replacement for the community string. All users sharing a group utilize the same security model, but the specific model settings (password and encryption key) are set per user. Note that SNMPv3 does not send passwords in clear-text, but instead uses MD5 or SHA1 hash-based authentication. For encryption, statically configured keys are used along with a single-DES (56-bit) symmetric cipher. This means that the same key should be configured on NMS for the particular user.

Note that SNMPv3 users do not appear in the running configuration for security reasons. The basic configuration of SNMPv3 can be verified as follows.

```
R6#show snmp user

User name: TRAP
Engine ID: 8000000903000050568D6E90
storage-type: nonvolatile        active
Authentication Protocol: SHA
Privacy Protocol: 3DESGroup-name: TRAP
User name: NORMAL
Engine ID: 8000000903000050568D6E90
storage-type: nonvolatile        active
Authentication Protocol: SHA
Privacy Protocol: 3DESGroup-name: NORMAL
User name: RESTRICTED
Engine ID: 8000000903000050568D6E90
storage-type: nonvolatile        active
Authentication Protocol: SHA
Privacy Protocol: NoneGroup-name: RESTRICTED
```

For the TRAP group, the notify view is auto-generated by the `snmp-server host` command, which bound the user (TRAP) and the group it belongs to (TRAP) to the list of notifications that are to be sent to the host.

https://t.me/learningnets

```
R6#show snmp group

groupname: ILMI                              security model:v1
contextname: <no context specified>          storage-type: permanent
readview : *ilmi                             writeview: *ilmi
notifyview: <no notifyview specified>
row status: active

groupname: ILMI                              security model:v2c
contextname: <no context specified>          storage-type: permanent
readview : *ilmi                             writeview: *ilmi
notifyview: <no notifyview specified>
row status: active
groupname: TRAP                              security model:v3 priv
contextname: <no context specified>          storage-type: nonvolatile
readview : v1default                         writeview: <no writeview specified>

notifyview: *tv.FFFFFFFF.FFFFFFFF.FFFFFFFF.F
row status: active
groupname: NORMAL                            security model:v3 priv
contextname: <no context specified>          storage-type: nonvolatile
readview : NORMAL                            writeview: NORMAL

notifyview: <no notifyview specified>
row status: active
groupname: RESTRICTED                        security model:v3 auth
contextname: <no context specified>          storage-type: nonvolatile
readview : RESTRICTED                        writeview: <no writeview specified>

notifyview: <no notifyview specified>        row status: active access-list: 99
!
!R6#show snmp view

*ilmi system - included permanent active
*ilmi atmForumUni - included permanent active NORMAL iso - included nonvolatile active
v1default iso - included permanent active
v1default internet.6.3.15 - excluded permanent active
v1default internet.6.3.16 - excluded permanent active
v1default internet.6.3.18 - excluded permanent active
v1default ciscoMgmt.394 - excluded permanent active
v1default ciscoMgmt.395 - excluded permanent active
v1default ciscoMgmt.399 - excluded permanent active
v1default ciscoMgmt.400 - excluded permanent active
RESTRICTED ifEntry.0.3 FF:EF included nonvolatile active
*tv.FFFFFFFF.FFFFFFFF.FFFFFFFF0F iso.2.840.10036 - included volatile active
```

https://t.me/learningnets

```
*tv.FFFFFFFF.FFFFFFFF.FFFFFFFF0F internet - included volatile active
```

To ensure that SNMP traps are being sent in encrypted and authenticated, enable SNMP packet debugging along with a low-level packet dump for outgoing SNMP traps as follows:

```
R6(config)#access-list 100 permit udp any any eq 162
R6#debug ip packet detail 100 dump
IP packet debugging is on (detailed) (dump) for access list 100
!
!R6#debug snmp packet
SNMP packet debugging is on
!
!R6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface loopback0
R6(config-if)#shutdown
!
!
SNMP Queuing packet to 155.1.146.100: SNMP: V2 Trap, reqid 23, errstat 0, erridx 0

 sysUpTime.0 = 2893642
 snmpTrapOID.0 = snmpTraps.3
 ifIndex.11 = 11
 ifDescr.11 = Loopback0
 ifType.11 = 24 lifEntry.20.11 = administratively down
SNMP: Packet sent via UDP to 155.1.146.100
IP: tableid=0, s=155.1.146.6 (local), d=155.1.146.100 (GigabitEthernet1.146), routed via RIB
IP: s=155.1.146.6 (local), d=155.1.146.100 (GigabitEthernet1.146), len 283, sending
UDP src=58135, dst=162
07802010:                   4500011B 00060000          E.......
07802020: FF11605E 9B019206 9B019264 E31700A2  ..`^.......dc.."
07802030: 01077D80 3081FC02 0103300D 02011202  ..}.0.|...0.....
07802040: 0205DC04 01030201 03043530 33040C80  ..\.......503...
07802050: 00000903 00001192 21DA8002 01180202  ........!Z......
07802060: 190C0404 54524150 040C3972 C05E1654  ....TRAP
..9r@^.T
07802070: 7D2249D0 11F10408 00000018 F1F81DCD  }"IP.q......qx.M
07802080: 0481B021 3372C60A 394B07E8 93FD35A3  ..0!3rF.9K.h.}5#
07802090: 584291FA 887D96B8 6894CCE6 58AAD5DD  XB.z.}.8h.LfX*U]
078020A0: 5F4EACE4 4B30FB5C 25B58A78 09A78EF0  _N,dK0{\%5.x.'.p
078020B0: 863CEE49 3745902A FEE6530C BAD247DC  .<nI7E.*~fS.:RG\
078020C0: DF94530F E5ED993C FC955C8C 3B42FC15  _.S.em.<|.\.;B|.
078020D0: D4C2B05B 8CF2869A FED2EC8E 38570FBB  TB0[.r..~Rl.8W.;
078020E0: C1454016 599591C2 D3FA07B1 0D048B26  AE@.Y..BSz.1...&
078020F0: EF4CBD1F 34B5E63A 0C05AA56 385B32D2  oL=.45f:..*V8[2R
```

https://t.me/learningnets

```
07802100: E5AB5AB6 F9F722F2 13C5610E 4ADE3FC3  e+Z6yw"r.Ea.J^?C
07802110: 73F78791 74AED319 5F86D648 74A04BE9  sw..t.S._.VHt Ki
07802120: 1CCDAB2E 0D022589 6BFBE813 E613B58C  .M+...%.k{h.f.5.
07802130: 739C66                               s.f
```

Change the security model for the destination host to “noAuth” and generate a trap message again. Note that now the message is not encrypted:

https://t.me/learningnets

```
R6#snmp-server host 155.1.146.100 version 3 noauth TRAP

!
!R6(config)#interface Loopback0
R6(config-if)#no shutdown

!
!
SNMP: Queuing packet to 155.1.146.100
SNMP: V2 Trap, reqid 24, errstat 0, erridx 0
 sysUpTime.0 = 2928473
 snmpTrapOID.0 = snmpTraps.4
 ifIndex.11 = 11
 ifDescr.11 = Loopback0
 ifType.11 = 24
 lifEntry.20.11 = upSNMP: Packet sent via UDP to 155.1.146.100
IP: tableid=0, s=155.1.146.6 (local), d=155.1.146.100 (GigabitEthernet1.146), routed via RIB
IP: s=155.1.146.6 (local), d=155.1.146.100 (GigabitEthernet1.146), len 238, sending
UDP src=58135, dst=162

079D76B0:                   450000EE 00070000          E..n....
079D76C0: FF11608A 9B019206 9B019264 E31700A2  ..`........dc.."
079D76D0: 00DA3F33 3081CF02 0103300D 02011302  .Z?30.O...0.....
079D76E0: 0205DC04 01000201 03042130 1F040C80  ..\.......!0....
079D76F0: 00000903 00001192 21DA8002 01180202  ........!Z......
079D7700: 1A680404 54524150 04000400 30819704  .h..TRAP
....0...
079D7710: 0C800000 09030000 119221DA 800400A7  ..........!Z...'
079D7720: 81840201 18020100 02010030 79300F06  ...........0y0..
079D7730: 082B0601 02010103 0043032C AF593017  .+.......C.,/Y0.
079D7740: 060A2B06 01060301 01040100 06092B06  ..+...........+.
079D7750: 01060301 01050430 0F060A2B 06010201  .......0...+....
079D7760: 02020101 0B02010B 3017060A 2B060102  ........0...+...
079D7770: 01020201 020B0409 4C6F6F70 6261636B  ........Loopback

079D7780: 30300F06 0A2B0601 02010202 01030B02  00...+.........
079D7790: 01183012 060C2B06 01040109 02020101  ..0...+........
079D77A0: 140B0402 7570                        ....up
<snip>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

SNMP MAC Address Notifications (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure SW1 to notify the NMS station at the IP address 155.X.146.100 about MAC addresses learned on ports connected to R1 and R5.
- Use the community value of CISCO to send notifications.
- Limit the rate of notifications to 1 per second.
- Report both added and removed MAC addresses.
- Store the latest 100 notification events in history buffer.

Configuration

https://t.me/learningnets

```
SW1:
interface FastEthernet0/1
 snmp trap mac-notification added
 snmp trap mac-notification removed
!
interface FastEthernet0/5
 snmp trap mac-notification added
 snmp trap mac-notification removed
!
snmp-server enable traps mac-notification
!
mac-address-table notification interval 2
mac-address-table notification history-size 100
mac-address-table notification
```

Verification

MAC Address table notifications allow the switches to report changes in the CAM table entries (adds and removes) for a particular link. This feature must be first enabled globally, and then configured on per-interface level. On trunk interfaces, all VLANs are affected by the configuration. In addition to reporting changes to an NMS, the switch can store recent events in the local buffer.

https://t.me/learningnets

```
Rack1SW1#clear mac-address-table dynamic
 Rack1SW1#show mac-address-table notification

MAC Notification Feature is Enabled on the switch
Interval between Notification Traps : 2 secs Number of MAC Addresses Added : 7
Number of MAC Addresses Removed : 7
Number of Notifications sent to NMS : 9

Maximum Number of entries configured in History Table : 100
Current History Table Length : 2 MAC Notification Traps are Enabled

History Table contents
---------------------- History Index 0, Entry Timestamp 3441456, Despatch Timestamp 3441456

MAC Changed Message :
Operation: Deleted Vlan: 58    MAC Addr: 0004.9a0b.62c1 Dot1dBasePort: 7
Operation: Deleted Vlan: 146   MAC Addr: 0011.2030.f900 Dot1dBasePort: 3
Operation: Added   Vlan: 58    MAC Addr: 0004.9a0b.62c1 Dot1dBasePort: 7      History Index 1,
Entry Timestamp 3441657, Despatch Timestamp 3441657

MAC Changed Message :
Operation: Added   Vlan: 146   MAC Addr: 0011.2030.f900 Dot1dBasePort: 3
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

SNMP Notifications of Syslog Messages

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 so that all debugging and higher priority level messages are sent via SNMP to an NMS host at the IP address 155.1.146.100.
- Set the syslog to SNMP buffer size to 100 messages.

Configuration

```
R1:

snmp-server enable traps syslog
snmp-server host 155.1.146.100 CISCO
!
logging history debugging
logging history size 100
```

Verification

SNMP logging of syslog messages allows the router to forward syslog messages to a remote NMS station using SNMP trap PDUs. Syslog fist sends the logs to a special history buffer, and then the SNMP agent replicates the messages as SNMP traps. This requires that “syslog” traps have been enabled globally or for that particular NMS. To verify this configuration, enable SNMP packet debugging and

https://t.me/learningnets

generate a syslog message by shutting down an interface.

```
R1#debug snmp packets

SNMP packet debugging is on
!R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#interface Tunnel0
R1(config-if)#shutdown
 R1(config-if)#no shutdown

!
!SNMP: Queuing packet to 155.1.146.100
SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 155.1.146.1, gentrap 6, spectrap 1
 clogHistoryEntry.2.1557 = LINK
 clogHistoryEntry.3.1557 = 4
 clogHistoryEntry.4.1557 = UPDOWN  clogHistoryEntry.5.1557 = Interface Tunnel0, changed state to up
 clogHistoryEntry.6.1557 = 14453399SNMP: Packet sent via UDP to 155.1.146.100
```

Verify SNMP configuration:

```
R1#show snmp
Chassis: 92HVO0PSL3U
0 SNMP packets input
    0 Bad SNMP version errors
    0 Unknown community name
    0 Illegal operation for community name supplied
    0 Encoding errors
    0 Number of requested variables
    0 Number of altered variables
    0 Get-request PDUs
    0 Get-next PDUs
    0 Set-request PDUs
    0 Input queue packet drops (Maximum queue size 1000)
4 SNMP packets output
    0 Too big errors (Maximum packet size 1500)
    0 No such name errors
    0 Bad values errors
    0 General errors
    0 Response PDUs
    4 Trap PDUs
SNMP logging: enabled
Logging to 155.1.146.100.162, 0/10, 4 sent, 0 dropped.
!
!R1#show logging history
Syslog History Table:100 maximum table entries,
```

https://t.me/learningnets

```
saving level debugging or higher
 105 messages ignored, 0 dropped, 0 recursion drops
 1530 table entries flushed
 SNMP notifications enabled, 9 notifications sent
   entry number 1551 : DUAL-5-NBRCHANGE
EIGRP-IPv4 100: Neighbor 155.1.0.5 (Tunnel0) is down: interface down
    timestamp: 14452905
   entry number 1552 : CRYPTO-6-ISAKMP_ON_OFF
    ISAKMP is OFF
    timestamp: 14452906
   entry number 1553 : LINEPROTO-5-UPDOWN Line protocol on Interface Tunnel0, changed state to down
    timestamp: 14453105
   entry number 1554 : LINK-5-CHANGED Interface Tunnel0, changed state to administratively down
    timestamp: 14453105
   entry number 1555 : CRYPTO-6-ISAKMP_ON_OFF
    ISAKMP is ON
    timestamp: 14453200
   entry number 1556 : LINEPROTO-5-UPDOWN Line protocol on Interface Tunnel0, changed state to up
    timestamp: 14453399
   entry number 1557 : LINK-3-UPDOWN Interface Tunnel0, changed state to up

    timestamp: 14453399
   entry number 1558 : SYS-5-CONFIG_I
    Configured from console by console
    timestamp: 14453486
   entry number 1559 : DUAL-5-NBRCHANGE
    EIGRP-IPv4 100: Neighbor 155.1.0.5 (Tunnel0) is up: new adjacency
    timestamp: 14453803
   entry number 1560 : CDP-4-DUPLEX_MISMATCH
    duplex mismatch discovered on GigabitEthernet1 (not half duplex), with SW1 FastEthernet0/1 (half duplex).
    timestamp: 14456618
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

CDP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 to send CDP v2 announcement every 10 seconds, and instruct other devices to hold the updates for 40 seconds.
- Ensure R4 runs CDP over its DMVPN link

Configuration

```
R4:

cdp advertise-v2
cdp timer 10
cdp holdtime 40
!
interface Tunnel0
 cdp enable
```

Verification

Cisco Discovery Protocol (CDP) is used to exchange basic device information and aid in troubleshooting. CDP can be enabled/disabled globally with the `[no] cdp run` command, and at the interface level with the `[no] cdp enable` .

https://t.me/learningnets

```
R4#show cdp

Global CDP information: Sending CDP packets every 10 seconds
 Sending a holdtime value of 40 seconds
 Sending CDPv2 advertisements is  enabled
!
!R4#show cdp interface

GigabitEthernet1 is up, line protocol is up
  Encapsulation 802.1Q Virtual LAN, Vlan ID  1.
  Sending CDP packets every 10 seconds
  Holdtime is 40 secondsTunnel0 is up, line protocol is up
  Encapsulation TUNNEL
  Sending CDP packets every 10 seconds
  Holdtime is 40 seconds

 cdp enabled interfaces : 2
 interfaces up          : 2
 interfaces down        : 0
 !
 !R4#show cdp traffic
CDP counters :
        Total packets output: 7282, Input: 72030
        Hdr syntax: 0, Chksum error: 0, Encaps failed: 6
        No memory: 0, Invalid packet: 0,
        CDP version 1 advertisements output: 0, Input: 0
CDP version 2 advertisements output: 7282, Input: 72030
```

Verify R4's CDP neighbors:

```
R4#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW1              Gig 1             155              S I   WS-C3560- Fas 0/1
R10              Gig 1             127              R I   CSR1000V  Gig 1
R2               Gig 1             147              R I   CSR1000V  Gig 1
R3               Gig 1             152              R I   CSR1000V  Gig 1
R1               Gig 1             145              R I   CSR1000V  Gig 1
R6               Gig 1             147              R I   CSR1000V  Gig 1
R7               Gig 1             143              R I   CSR1000V  Gig 1
R5               Gig 1             170              R I   CSR1000V  Gig 1
```

https://t.me/learningnets

```
R8                Gig 1             166               R I   CSR1000V  Gig 1
R9                Gig 1             172               R I   CSR1000V  Gig 1
Total cdp entries displayed : 10
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

HTTP Server and Client

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 to support the transfer of the configuration file stored in R6's flash using HTTP.
  - R4 should be authenticated with a username/password pair of **CISCO/CISCO** stored in the local database of R6.
- R6 should only allow HTTP connections at port 8080, from the subnet 150.1.0.0/16, and limit the maximum number of concurrent connections to two.
- Allow R4 to use secure HTTP connection on port 4043, but limit the security features to use only DES symmetric cipher.

Configuration

```
R6:
username CISCO privilege 15 password 0 CISCO
!
ip http server
ip http max-connections 2
ip http path flash:
ip http port 8080
!
access-list 80 permit 150.1.0.0 0.0.255.255
!
ip http access-class 80
```

https://t.me/learningnets

```
ip http authentication local
ip http secure-server
ip http secure-port 4043
ip http secure-ciphersuite des-cbc-sha
R4:

ip http client source-interface Loopback0
ip http client username CISCO
ip http client password CISCO
ip http client secure-ciphersuite des-cbc-sha
```

Verification

IOS HTTP server is enabled by default to allow for remote router management (SDM). The default authentication is via the enable password (any name and the enable password) and can be changed to use the local database if AAA is not configured. Note that the local user needs privilege level 15 to access the router via HTTP. Additional security settings include setting the access-class or changing the default port number.

Secure HTTP server enables the use of SSL to protect communications. Note that by default the secure server is disabled and automatically generates a server SSL X.509 certificate when enabled (this will also bring the SSH server up). When changing the server cipher-suite, make sure the client will accept it during the handshake. In addition to server functionality, IOS implements an HTTP client. Using this client, it's possible to transfer files to the local router from the remote server. The client is even capable of using a proxy server.

To verify the client/server, issue a copy http command from R4.

```
R4#copy http://155.1.146.6:8080/EIGRP_Initial null:

Accessing http://155.1.146.6:8080/EIGRP_Initial...
Loading http://155.1.146.6:8080/EIGRP_Initial !1610 bytes copied in 0.044 secs (36591 bytes/sec)
```

Try to connect to the HTTP server from a different source interface or using different credentials.

```
R4(config)#ip http client source-interface GigabitEthernet 1.146
!
!R4#copy http://155.1.146.6:8080/EIGRP_Initial null:
 copy http://155.1.146.6:8080/EIGRP_Initial null:
 Accessing http://155.1.146.6:8080/EIGRP_Initial...
```

https://t.me/learningnets

```
%Error opening http://155.1.146.6:8080/EIGRP_Initial (I/O error)

!
!R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#ip http client source-interface Loopback 0
R4(config)#ip http client username USER
R4(config)#exit
!
!R4#copy http://155.1.146.6:8080/EIGRP_Initial null:
Accessing http://155.1.146.6:8080/EIGRP_Initial...
%Error opening http://155.1.146.6:8080/EIGRP_Initial (Permission denied)
```

Verify the HTTP settings on the server. Note that “HTTP secure server client authentication: Disabled” means that the server will not ask for the client’s SSL certificate.

```
R6#show ip http server status
R6#show ip http server status
HTTP server status: Enabled
HTTP server port: 8080
HTTP server active supplementary listener ports:
HTTP server authentication method: localHTTP server access class: 80
HTTP server base path: flash:
HTTP server help root: Maximum number of concurrent server connections allowed: 2
Server idle time-out: 180 seconds
Server life time-out: 180 secondsMaximum number of requests allowed on a connection: 1
Server linger time : 60 seconds
HTTP server active session modules: ALL
HTTP secure server capability: Present
HTTP secure server status: Enabled
HTTP secure server port: 4043HTTP secure server ciphersuite: des-cbc-sha
 HTTP secure server client authentication: Disabled

HTTP secure server trustpoint:
HTTP secure server active session modules: ALL
```

Verify the HTTP client settings on R4 as follows.

```
R6#show ip http client all
 HTTP client status: Enabled
HTTP client application session modules:
Id                : 1
Application Name : HTTP CFS
```

https://t.me/learningnets

```
Version          : HTTP/1.0
Persistent       : persistent
Response-timeout : 0
Retries          : 0
Proxy            :

Id               : 2
Application Name : HTTP_CALL_HOME_AGEN
Version          : HTTP/1.1
Persistent       : persistent
Response-timeout : 30000
Retries          : 1
Proxy            :


HTTP client current connections:Persistent connection = enabled (default)
Connection establishment timeout = 10s (default)
Connection idle timeout = 30s (default)
Maximum number of connection establishment retries = 1 (default)
Maximum http client connections per host : 2

HTTP secure client capability: Present
HTTP secure client ciphersuite: 3des-ede-cbc-sha des-cbc-sha rc4-128-md5 rc4-128-sha
HTTP secure client trustpoint:

local-ipaddress:port  remote-ipaddress:port in-bytes   out-bytes



Total client connections : 0



HTTP client cache:
Maximum Memory size for cache    : 100000 bytes (default)
Maximum memory per cache entry   : 2000 bytes (default)
Memory used                      : 0 bytes
Memory Available                 : 100000 bytes
Cache Ager interval              : 5 minutes (default)
Total entries created            : 0
Id    Type    Url              Memory-size(Bytes)     Refcnt     Valid(Sec)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

FTP Client

> You must load the initial configuration files for the section, **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 as FTP client to initiate sessions using the username/password **CISCO/CISCO**.
  - When a transfer occurs, the ftp server should initiate the FTP data channel back to the client.
  - Source FTP connections off R4 Loopback0 interface.

Configuration

```
R4:

no ip ftp passive
ip ftp source-interface Loopback0
ip ftp username CISCO
ip ftp password CISCO
```

Verification

In addition to FTP client support, IOS may act as a simple FTP server. The server service does not support any authentication, and because of numerous vulnerabilities (such as bug ID CSCse29244, *IOS crash when transferring files via FTP*) Cisco has removed the FTP server functionality from recent IOS releases,

https://t.me/learningnets

promising to re-implement a more stable and advanced feature set at a later time. The FTP client feature can be used to save core dumps and transfer files from external FTP servers. For file transfers between IOS routers, the HTTP or TFTP protocols should be used. Because of the FTP server crashing IOS, it is impossible to properly verify the FTP server functionality.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

TFTP Server and Client

> You must load the initial configuration files for the section, **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R6 as TFTP server so that file named **cvac.log** can be downloaded from flash using the file name **test**.
  - Ensure that only R1's Loopback0 interface is allowed to connect and retrieve files via TFTP from R6.
  - Configure R1 to source TFTP packets off its Loopback0 interface.

Configuration

```
R6:
tftp-server flash:cvac.log alias test 10
!
access-list 10 permit 150.1.1.1
R1:

ip tftp source-interface Loopback0
```

Verification

https://t.me/learningnets

IOS TFTP server allows read-only access to specified files. An access-list can be used to limit the scope of users accessing the files. The alias feature provides abbreviated names for long filenames stored in the flash.

```
R6#debug tftp events
 TFTP Event debugging is on
!R1#copy tftp://150.1.6.6/test null:
Accessing tftp://150.1.6.6/EIGRP_Config...Loading test from 150.1.6.6 (via GigabitEthernet1.146): !
[OK - 1610 bytes]
1610 bytes copied in 0.085 secs (18941 bytes/sec)
!
!R6#
TFTP: Looking for test
TFTP: Opened bootflash:cvac.log, fd 0, size 0 for process 139
TFTP: Finished bootflash:cvac.log, time 00:00:00 for process 139
TFTP: Looking for test
TFTP: Opened bootflash:cvac.log, fd 0, size 0 for process 436
TFTP: Finished bootflash:cvac.log, time 00:00:00 for process 436
TFTP: Looking for test
TFTP: Opened bootflash:cvac.log, fd 0, size 0 for process 139
TFTP: Finished bootflash:cvac.log, time 00:00:00 for process 139
TFTP: Looking for test
TFTP: Opened bootflash:cvac.log, fd 0, size 0 for process 436
TFTP: Finished bootflash:cvac.log, time 00:00:00 for process 436
```

R4 cannot access the same file using TFTP.

```
R4#copy tftp://150.1.6.6/test null:
%Error opening tftp://150.1.6.6/EIGRP_Config (No such file or directory)
!
!R6#
TFTP: Looking for EIGRP_Config
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Remote Shell

> You must load the initial configuration files for the section, **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure the network so that R1 is to view the running configuration of R6 using remote shell.
- Source remote-shell connections off the Loopback0 interface of R1.
- Allow RCP file-copy from R6 to R1.

Configuration

```
R1:
ip rcmd remote-username RCP
ip rcmd source-interface Loopback0

R6:

ip rcmd rcp-enable
ip rcmd rsh-enable
!
ip rcmd remote-host R6 150.1.1.1 R1 enable
ip rcmd remote-host RCP 150.1.1.1 R1 enable
```

Verification

https://t.me/learningnets

IOS implementation includes the UNIX-line remote command execution environment (RSH). No password is required to access the local system, just an entry in the local “.rhosts”-like table. The table maps a remote station’s IP and username to the local username and privilege level (in a classic UNIX system, the privilege level is determined by the local username) using the following syntax

```
ip rcmp remote-host <local-name><remote-IP><remote-name><privilege> .
```

The local name in “.rhosts” table does not need to match any user in the local database, and commonly the target router’s hostname is used as the RCMD local username. When a remote host connects to the local router via remote-shell, the remote host’s IP address is looked up in “.rhosts” table, the target username specified by remote system is matched against <local-name>, and the source username specified by remote system is matched against <remote-name>. If all fields are matched, remote-shell access is granted. Without the “enable” privilege, only exec privilege level 1 commands are available to the remote user.

Additionally, remote users may use the remote copy (RCP) feature to transfer files from the router to the remote system (similar to HTTP and TFTP transfers). IOS may act as an rsh and rcp client itself, allowing for remote command execution and file transfers. The local hostname is always used as the source username, but the remote (target) username can be changed.

This configuration can be verified as follows.

https://t.me/learningnets

```
R6#debug ip tcp rcmd
RCMD transactions debugging is on

!
!R1#rsh 150.1.6.6 /user R6 show run interface GigabitEthernet1.146


Building configuration...

Current configuration : 136 bytes
!
interface GigabitEthernet1.146
 encapsulation dot1Q 146
 ip address 155.1.146.6 255.255.255.0
 ipv6 address 2001:155:1:146::6/64
 end


RCMD: [995 -> 150.1.6.6:514] send \0
RCMD: [995 -> 150.1.6.6:514] send R1\0
RCMD: [995 -> 150.1.6.6:514] send R6\0
RCMD: [995 -> 150.1.6.6:514] send show run interface GigabitEthernet1.146\0

RCMD: [995 <- 150.1.6.6:514] recv <OK>
```

Transfer a file from R6 to R1 using RCP.

```
R6#show flash:


System flash directory:
File  Length   Name/status 1   29631128  RIP_Routing_Initial

  2   1941     saved-config
  3   1036     r6i
R1#copy rcp://150.1.6.6/RIP_Routing_Initial null:
Source username [RCP]?
Accessing rcp://*****@150.1.6.6/RIP_Routing_Initial...!
1629 bytes copied in 0.863 secs (1888 bytes/sec)
!
!R6#

RCMD: [988 -> 150.1.6.6:514] send \0
RCMD: [988 -> 150.1.6.6:514] send R1\0
```

https://t.me/learningnets

```
RCMD: [988 -> 150.1.6.6:514] send RCP\0
RCMD: [988 -> 150.1.6.6:514] send rcp -f RIP_Routing_Initial\0
RCMD: [988 <- 150.1.6.6:514] recv <OK>
RCMD: [988 -> 150.1.6.6:514] send <OK>
RCP: [988 <- 150.1.6.6:514] recv C0644 1629 RIP_Routing_Initial
RCMD: [988 -> 150.1.6.6:514] send <OK>
RCMD: [988 -> 150.1.6.6:514] send <BAD,Write failed>\n
RCMD: [981 -> 150.1.6.6:514] send \0
RCMD: [981 -> 150.1.6.6:514] send R1\0
RCMD: [981 -> 150.1.6.6:514] send RCP\0
RCMD: [981 -> 150.1.6.6:514] send rcp -f RIP_Routing_Initial\0
RCMD: [981 <- 150.1.6.6:514] recv <OK>
RCMD: [981 -> 150.1.6.6:514] send <OK>
RCP: [981 <- 150.1.6.6:514] recv C0644 1629 RIP_Routing_Initial
RCMD: [981 -> 150.1.6.6:514] send <OK>
RCP: [981 <- 150.1.6.6:514] recv 1629 bytes
RCMD: [981 -> 150.1.6.6:514] send <OK>
RCP: [981 <- 150.1.6.6:514] recv <EOF>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

NTP

> You must load the initial configuration files for the section, **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 and R6 as authoritative NTP time sources with stratum 5.
  - R4 and R6 should synchronize with each other using Loopback0 interfaces.
- R5 should poll R4 and R6 for time updates using Loopback0 interface, but prefer R4.
- Configure R5 to broadcast NTP updates on its connection to VLAN 58, and for R8 to listen to NTP broadcasts on this link.
- Configure R6 to send multicast NTP packets to the address 239.1.1.1 on its connection to VLAN 67.
- R7 and R9 should be able to synchronize with R6 using these multicast packets.

Configuration

```
R4:
ntp master 5
ntp peer 150.1.6.6
ntp source Loopback0
R5:
ntp server 150.1.6.6
ntp server 150.1.4.4 prefer
ntp source Loopback0
!
interface GigabitEthernet1.58
 ntp broadcast
```

https://t.me/learningnets

```
R6:

ntp master 5
ntp peer 150.1.4.4
ntp source Loopback0
!
interface GigabitEthernet1.67
 ntp multicast 239.1.1.1
R7:
interface GigabitEthernet1.67
 ip pim dense-mode
 ntp multicast client 239.1.1.1
!
interface GigabitEthernet1.79
 ip pim dense-mode
!
ip multicast-routing distributed
R8:
interface GigabitEthernet1.58
 ntp broadcast client
R9:

interface GigabitEthernet1.79
 ip pim dense-mode
 ntp multicast client 239.1.1.1
!
ip multicast-routing distributed
```

Verification

NTP time distribution is based on a loop-less tree topology. In the root of the tree resides the absolute time source (server), which is physically connected to an atomic clock, radio clock, or some other highly accurate time source. This server is described as being in *stratum 1*, meaning that it is one hop away from the source of the time.

Every next router that pulls time down from stratum N is automatically assigned to stratum N+1. That router in turn may become a time source (server) for clients in stratum N+2. A router in stratum N will not accept time updates from stratum N+1, enforcing a tree-level hierarchy of client-server relationships and preventing time-synchronization loops. If a router has multiple servers, it will only select one for time synchronization (the one that is closer to NTP root and has the least offset from the currently running clock) and use others as backup. The preference may be adjusted

https://t.me/learningnets

using the `prefer` keyword, but only among candidates having other parameters equal (such as same stratum).

Two routers in the same stratum may also form peer-to-peer, symmetric relationships. In this mode, both peers update each other's clock. However, just one router in a pair may be configured for NTP peering (active peer); the other router will automatically convert into passive mode and will provide time updates to the dynamic peer. NTP is also capable of broadcasting or multicasting time updates on a shared interface, and clients may listen to NTP packets on the other side. This allows a single server to synchronize multiple hosts in unsolicited mode.

Remember that NTP may take considerable time to synchronize, especially if router clocks are far out of sync. To speed up this process, adjust the clocks manually to values close to the server, to accelerate convergence (such as when a router considers all servers as "insane"). Note that in the following show command's output, reference "127.127.7.1" means "self." Before beginning verifications, make sure to set all clocks to the same value manually. It may take some time still for both masters to synchronize and consider each other "sane."

```
R4#clock set 18:00:00 6 May 2014
R5#clock set 18:00:00 6 May 2014
R6#clock set 18:00:00 6 May 2014
R7#clock set 18:00:00 6 May 2014
R8#clock set 18:00:00 6 May 2014
R9#clock set 18:00:00 6 May 2014
R4#show ntp status
 Clock is synchronized, stratum 5, reference is 127.127.1.1

nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 198300 (1/100 of seconds), resolution is 4000
reference time is D713A250.EB852140 (18:02:56.920 UTC Tue May 6 2014)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 2.21 msec, peer dispersion is 1.20 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 16, last update was 2 sec ago.
!
!R4#show ntp associations

address         ref clock       st   when   poll reach  delay  offset   disp
*~127.127.1.1     .LOCL.           4      5     16     1  0.000   0.000 7937.9
~150.1.6.6       .INIT.          16      -    128     0  0.000   0.000 15937.
* sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
!
!R6#show ntp status
 Clock is synchronized, stratum 5, reference is 127.127.1.1
```

https://t.me/learningnets

```
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 245900 (1/100 of seconds), resolution is 4000
reference time is D713A510.66A7F0B8 (18:14:40.401 UTC Tue May 6 2014)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 2.24 msec, peer dispersion is 1.20 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 16, last update was 4 sec ago.
!
!R6#show ntp associations

address         ref clock       st   when   poll reach  delay  offset   disp
*~127.127.1.1     .LOCL.           4     15     16   377  0.000   0.000  1.204
~150.1.4.4       .INIT.          16      -   1024     0  0.000   0.000 15937.
* sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
!
!R5#show ntp status
Clock is synchronized, stratum 6, reference is 150.1.4.4

nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 2085400 (1/100 of seconds), resolution is 4000
reference time is D713EF5A.7126EAB0 (23:31:38.442 UTC Tue May 6 2014)
clock offset is 0.5000 msec, root delay is 1.00 msec
root dispersion is 16.86 msec, peer dispersion is 1.97 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is -0.000000002 s/s
system poll interval is 1024, last update was 750 sec ago.
!
!R5#show ntp associations

  address         ref clock       st   when   poll reach  delay  offset   disp
*~150.1.4.4       127.127.1.1      5    736   1024   377  1.000   0.500  1.974
x~150.1.6.6       127.127.1.1      5    480   1024   377  1.000 -5874.4  2.015
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
!
!R7#show ntp status
Clock is synchronized, stratum 6, reference is 155.1.67.6

nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 2098600 (1/100 of seconds), resolution is 4000
reference time is D713F2A3.E8B43BD8 (23:45:39.909 UTC Tue May 6 2014)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 6.92 msec, peer dispersion is 2.89 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000016 s/s
system poll interval is 64, last update was 52 sec ago.
!
!R7#show ip mroute 239.1.1.1
```

https://t.me/learningnets

```
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.1.1.1), 05:43:51/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.79, Forward/Dense, 05:43:51/stopped
    GigabitEthernet1.67, Forward/Dense, 05:43:51/stopped
(155.1.67.6, 239.1.1.1), 05:43:42/00:01:28, flags: LT
Incoming interface: GigabitEthernet1.67, RPF nbr 0.0.0.0
  Outgoing interface list: GigabitEthernet1.79, Forward/Dense, 05:43:42/stopped
!
!R9#show ntp status
Clock is synchronized, stratum 6, reference is 155.1.67.6

nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 2095400 (1/100 of seconds), resolution is 4000
reference time is D713F262.ACCCCEA8 (23:44:34.675 UTC Tue May 6 2014)
clock offset is 1.0000 msec, root delay is 2.00 msec
root dispersion is 7944.16 msec, peer dispersion is 3939.45 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 64, last update was 99 sec ago.
!
!R9#show ip mroute 239.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
```

https://t.me/learningnets

```
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.1.1.1), 00:03:37/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.79, Forward/Dense, 00:03:37/stopped
(155.1.67.6, 239.1.1.1), 00:02:53/00:00:06, flags: PLTX
Incoming interface: GigabitEthernet1.79, RPF nbr 155.1.79.7
Outgoing interface list: Null
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

NTP Authentication

> You must load the initial configuration files for the section, **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R4 and R6 as authoritative NTP time sources with stratum 5.
- Configure R4 and R6 to authenticate each other's peering session using the key **CISCO46**.
  - Use Loopback0 interfaces for this.
- R5 should authenticate NTP packets received from R4 and R6 using key values **CISCO4** and **CISCO6**, respectively.
  - Use Loopback0 interfaces for this.
- Configure R5 to broadcast NTP updates on its connection to VLAN 58, and for R8 to listen to NTP broadcasts on this link.
  - R8 should only accept NTP updates on VLAN 58 if they are authenticated using a key value of **CISCO58**.

Configuration

```
R4:
ntp master 5
ntp authenticate
ntp authentication-key 46 md5 CISCO46
ntp trusted-key 46
ntp peer 150.1.6.6 key 46 source loopback0
ntp authentication-key 4 md5 CISCO4
R5:
```

https://t.me/learningnets

```
ntp authenticate
ntp authentication-key 4 md5 CISCO4
ntp authentication-key 6 md5 CISCO6
ntp trusted-key 4
ntp trusted-key 6
ntp server 150.1.4.4 key 4 source loopback0
ntp server 150.1.6.6 key 6 source loopback0
ntp authentication-key 58 md5 CISCO58
!
interface GigabitEthernet1.58
 ntp broadcast key 58
R6:
ntp master 5
ntp authenticate
ntp authentication-key 46 md5 CISCO46
ntp trusted-key 46
ntp peer 150.1.4.4 key 46 source loopback0
ntp authentication-key 6 md5 CISCO6
R8:

ntp authenticate
ntp authentication-key 58 md5 CISCO58
ntp trusted-key 58
!
interface GigabitEthernet1.58
 ntp broadcast client
```

Verification

NTP authentication is based on two concepts. First, NTP packets that can update (change) the local clock must be authenticated. Second, authenticated NTP packets are signed using an HMAC MD5 signature, and carry the signing key index (number) inside.

Based on the above, only the router that updates its clock (such as client or peer) must be configured to require authentication. However, authentication keys must be configured on **both** sides (such as client and server) using the **same** key number (index). For example, if R1 is the NTP client and R2 is the NTP server, R1 may send authenticated (signed) NTP request to R2 with the key index 12. R2 will not verify the signature (it's not going to update its clock), but instead it simply looks up the key with index 12 in its local key chain. Using this key, the NTP response is signed and sent back to R1.

To make router authenticate incoming updates, you must explicitly enable

https://t.me/learningnets

authentication, configure a key with a key index, and ensure that the authenticating router trusts this key. There is no need to trust keys that are **not** used to authenticate updates (such as reply keys).

```
R4#show ntp associations detail  | inc auth
150.1.6.6 configured,authenticated
, selected, sane, valid, stratum 5
!
!R6#show ntp associations detail  | inc auth
150.1.4.4 configured,authenticated
, selected, sane, valid, stratum 5
!
!R5#show ntp associations detail | inc auth
150.1.4.4 configured,authenticated
, selected, sane, valid, stratum 5150.1.6.6 configured,authenticated
, our_master, sane, valid, stratum 5
!
!R8#show ntp associations detail | inc auth
155.1.58.5 dynamic,authenticated
, our_master, sane, valid, stratum 6
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

NTP Access Control

> You must load the initial configuration files for the section, **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure R1 and R4 as NTP servers with stratum 5.
  - Configure R6 as the NTP client; R4 will be the preferred NTP server.
- Configure R4 so that it serves the time to R6 only.
- Configure R6 so that only R4 can update its clock.
- Use Loopback0 for all NTP messages.

Configuration

```
R4:
ntp master 5
ntp source Loopback0
access-list 6 permit 150.1.6.6
!
ntp access-group serve-only 6
R1:
ntp master 5
ntp source Loopback0
R6:

access-list 4 permit 150.1.4.4
!
ntp source Loopback0
ntp access-group peer 4
```

https://t.me/learningnets

```
ntp server 150.1.4.4 prefer
ntp server 150.1.1.1
```

Verification

NTP access-control divides NTP messages into two categories, control messages and update/request messages. Control messages are those needed to extract specific management information, such as the peer status or set a management parameter. NTP control messages are not needed for proper time synchronization. NTP update/request messages are those messages needed for time synchronization.

NTP access control defines four levels: peer, serve, serve-only, and query-only. Peer permits NTP updates/requests to the host as well as control queries. This is the only access type that allows the host to be synchronized by others. Serve permits NTP requests but rejects NTP updates (does not change local clock). Control queries are also permitted. Serve-only permits NTP requests only. It rejects attempts to synchronize the local system and does not accept control queries. Query-only accepts NTP control queries. No response to NTP requests are sent, and no local system time synchronization with a remote system is allowed.

When these levels are associated with access-lists on a router, an incoming message source IP address is matched in the order listed above (peer, serve, serve-only, and query-only). The first match determines the type of access. If some access-groups are configured but not all, all other types of access from any other sources are implicitly denied. To verify the clock synchronization status, check the debugging and show command outputs as given below.

https://t.me/learningnets

```
R6#debug ntp all

NTP message sent to 150.1.1.1, from interface 'Loopback0' (150.1.6.6).
NTP message sent to 150.1.4.4, from interface 'Loopback0' (150.1.6.6).
NTP message received from 150.1.1.1 on interface 'Loopback0' (150.1.6.6).
NTP Core(DEBUG): ntp_receive: message received
NTP Core(NOTICE): ntp_receive: dropping message: RES_DONTSERVE restriction.

NTP message received from 150.1.4.4 on interface 'Loopback0' (150.1.6.6).
NTP Core(DEBUG): ntp_receive: message receivedNTP Core(DEBUG): ntp_receive: peer is 0x7F

R6#9EE3FF7130, next action is 1.
NTP Core(DEBUG): Peer becomes reachable, poll set to 6.

NTP Core(INFO): 150.1.4.4 8014 84 reachable
NTP Core(INFO): 150.1.4.4 962A 8A sys_peerNTP Core(NOTICE): Clock is synchronized.
```

Note that although R6 receives NTP packets from R1, it does not accept it because we have applied access-group only to accept the time from R4. We can verify the association and synchronization status from both servers.

```
R6#show ntp association detail
150.1.1.1 configured, ipv4, insane, invalid, unsynced, stratum 16
ref ID .INIT., time 00000000.00000000 (00:00:00.000 UTC Mon Jan 1 1900)
our mode client, peer mode unspec, our poll intvl 64, peer poll intvl 1024
root delay 0.00 msec, root disp 0.00, reach 0, sync dist 15937.83
delay 0.00 msec, offset 0.0000 msec, dispersion 15937.50, jitter 0.00 msec
precision 2**10, version 4
assoc id 10776, assoc name 150.1.1.1
assoc in packets 0, assoc out packets 3, assoc error packets 0
org time D758468A.F0625068 (19:37:46.939 UTC Fri Jun 27 2014)
rec time 00000000.00000000 (00:00:00.000 UTC Mon Jan 1 1900)
xmt time 00000000.00000000 (00:00:00.000 UTC Mon Jan 1 1900)
filtdelay =     0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
filtoffset =    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
filterror =  16000.0 16000.0 16000.0 16000.0 16000.0 16000.0 16000.0 16000.0
minpoll = 6, maxpoll = 10
!
!150.1.4.4 configured, ipv4, our_master, sane, valid, stratum 5
ref ID 127.127.1.1    , time D7584689.29FBE7E0 (19:37:45.164 UTC Fri Jun 27 2014)
our mode client, peer mode server, our poll intvl 64, peer poll intvl 64
root delay 0.00 msec, root disp 2.18, reach 1, sync dist 7942.96
delay 2.00 msec, offset 1.0000 msec, dispersion 7938.47, jitter 0.97 msec
```

https://t.me/learningnets

```
precision 2**10, version 4
assoc id 10777, assoc name 150.1.4.4
assoc in packets 1, assoc out packets 3, assoc error packets 0
org time 00000000.00000000 (00:00:00.000 UTC Mon Jan 1 1900)
rec time D758468A.F0E562D8 (19:37:46.941 UTC Fri Jun 27 2014)
xmt time D758468A.F0E562D8 (19:37:46.941 UTC Fri Jun 27 2014)
filtdelay =     2.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
filtoffset =    1.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
filterror =     1.95 16000.0 16000.0 16000.0 16000.0 16000.0 16000.0 16000.0
minpoll = 6, maxpoll = 10
!
!R6#show ntp status
Clock is synchronized, stratum 6, reference is 150.1.4.4

nominal freq is 250.0000 Hz, actual freq is 249.9999 Hz, precision is 2**10
ntp uptime is 488000 (1/100 of seconds), resolution is 4016
reference time is D7584964.F0A3D9A0 (19:49:56.940 UTC Fri Jun 27 2014)
clock offset is 0.5000 msec, root delay is 1.00 msec
root dispersion is 7.06 msec, peer dispersion is 2.96 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000164 s/s
system poll interval is 128, last update was 20 sec ago.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Auto-Install over LAN Interfaces using DHCP (pending update)

Load the "EIGRP Routing" initial configurations prior to starting.

Task

- Configure the network to support auto-install as follows:
- Upon booting, R4 should request an IP address for its GigabitEthernet1.146 interface via DHCP.
- R1 should forward this request onto R5.
- R5 should reply with the IP address 155.1.146.4/24, the default gateway 155.1.146.1, the DNS server 155.1.146.6, and the TFTP server “R8”.
- R4 should then ask R6 what its own hostname is, and download the config file “R4-confg” from R7 via TFTP.

Configuration

```
R6:
ip dns server
ip host R8 155.1.58.8
ip host R4 155.1.146.4

R1:
interface GigabitEthernet1.146
 ip helper-address 155.1.0.5

R5:
ip dhcp pool HOST_R4
host 155.1.146.4 255.255.255.0
   client-identifier 0063.6973.636f.2d30.3030.372e.6562.6465.2e35.3632.322d.4574.302f.31
   default-router 155.1.146.1
```

https://t.me/learningnets

```
    dns-server 155.1.146.6
    option 66 ascii "R8"

SW2:
tftp-server flash:Rack1R4-confg
```

Verification

AutoInstall tries to obtain an IP address and configuration by broadcasting DHCP/BOOTP/RARP requests over all LAN interfaces, using HDLC encapsulation on the first serial interface and the SLARP protocol to obtain an IP address, and by using Frame-Relay encapsulation on the first serial interface and the BOOTP protocol to obtain an IP address.

On regular Ethernet LAN segments, the AutoInstall process relies primarily on DHCP for obtaining an IP address and TFTP for downloading the configuration. The procedure is performed in the following stages.

First the router tries to obtain an IP address and subnet mask via DHCP or BOOTP by broadcasting DHCP/BOOTP requests out of all LAN interfaces. In addition to IP address information, the DHCP server can provide the default-router, DNS server list, TFTP server, and a boot-file name. In parallel to DHCP, IOS sends BOOTP queries, and may then learn TFTP server's name ("sname") or IP address ("siaddr") and boot-file name ("file") values from BOOTP.

Next the router tries to download its configuration using unicast transfer from the TFTP server learned via DHCP. If the DHCP/BOOTP server has specified the boot-file, the router tries to download this file as its configuration. Prior to version 12.1(5)T, IOS only tried broadcast requests to reach the TFTP server and ignored the boot-file name learned via DHCP/BOOTP using generic names as described later.

Next, for the destination IP address of the TFTP server, the router either uses Option 150 or tries to resolve Option 66 using the DNS server list supplied by DHCP. If a BOOTP response is available with the "sname" and "siaddr" fields, they could also be used. The order of preference is: "sname", "option 66", "option 150", and "siaddr". If the TFTP server's IP address cannot be obtained, the router broadcasts download requests to the TFTP port. If the TFTP server does not respond to three unicast requests, the router falls back to broadcasts.

If the DHCP/BOOTP server did not provide a boot-file name, the router tries to download a network-wide configuration file named "network-confg" and then "cisconet.cfg" from the TFTP server (using either unicasts if the IP address of the TFTP server is known, or broadcasts). The purpose of network-wide configuration

https://t.me/learningnets

file is to allow the router to determine its hostname.

If the network-wide configuration has been successfully downloaded, the router looks into the `ip host` commands trying to find a match to its newly obtained IP address and map it to a hostname. If the hostname is successfully found, the router requests the host-specific configuration file “<hostname>-confg” via TFTP, and if this fails it requests “<hostname>.cfg”. If the file has been downloaded, it is merged with the router’s configuration.

If the hostname has not been determined, the router tries to resolve its name via DNS (using either DNS servers or broadcasts) and downloads the respective configuration file. If the name lookup or host-specific configuration file download has failed, the router tries to download the generic configuration file named either “router-confg” or “router.cfg”.

To verify this, copy R4’s configuration to SW2 via TFTP, and then erase and reload R4.

```
R4#copy running-config flash:R4-confg
Destination filename [R4-confg]?
Erase flash: before copying? [confirm]n
Verifying checksum...  OK (0xC982)
1459 bytes copied in 8.033 secs (182 bytes/sec)
Rack1R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Rack1R4(config)#tftp-server flash:Rack1R4-confg
Rack1SW2#copy tftp: flash:
Address or name of remote host []?150.1.4.4

Source filename []? Rack1R4-confg
Destination filename [Rack1R4-confg]?
Accessing tftp://150.1.4.4/Rack1R4-confg...
Loading Rack1R6-confg from 150.1.4.4 (via Vlan58): !
[OK - 1459 bytes]
```

Now determine the Client-ID of R4’s FastEthernet 0/1 interface so we know what client-identifier to put in R5’s DHCP host pool. To accomplish this, temporarily set up the respective interface for DHCP IP address assignment and use `show dhcp lease` command.

```
Rack1R4#config t
Enter configuration commands, one per line.  End with CNTL/Z.Rack1R4(config)#interface FastEthernet0/1
Rack1R4(config-if)#ip address dhcp
Rack1R4(config-if)#endRack1R4#show dhcp lease
```

https://t.me/learningnets

```
Temp IP addr: 0.0.0.0  for peer on Interface: FastEthernet0/1
Temp  sub net mask: 0.0.0.0
   DHCP Lease server: 0.0.0.0, state: 1 Selecting
   DHCP transaction id: 14F4
   Lease: 0 secs,  Renewal: 0 secs,  Rebind: 0 secs
   Next timer fires after: 00:00:01   Retry count: 0   Client-ID: cisco-0007.ebde.5622-Et0/1
   Client-ID hex dump: 636973636F2D303030372E656264652E
 353632322D4574302F31
   Hostname: Rack1R4
Rack1R4#show interfaces FastEthernet 0/1
FastEthernet0/1 is up, line protocol is up   Hardware is AmdP2, address is 0007.ebde.5622
 (bia 0007.ebde.5622)
```

Cisco uses the MAC address of the interface along with interface short name to create a client-identifier. In the Cisco IOS DHCP server, the client-ID must be specified as hex value, starting with ‘00’ (such as0063.6973.636F.2D30.3030.372E.6562.6465.2E35.3632.322D.4574.302F.31).

When R4 is reset, the DHCP request is received by R5, and the TFTP request is received by SW2.

```
Rack1R4#show start
 startup-config is not present
Rack1R4#reload
Proceed with reload? [confirm]

SYS-5-RELOAD: Reload requested by console. Reload Reason: Reload Command.
Rack1R5#debug  ip dhcp server events
 Rack1R5#debug  ip dhcp server packet
Rack1SW2#debug tftp events

TFTP Event debugging is on
Rack1SW2#debug tftp packets

TFTP Packet debugging is on

Rack1R5# DHCPD: DHCPDISCOVER received from client
 0063.6973.636f.2d30.3030.372e.6562.6465.2e35.3632.322d.4574.302f.31
 through relay 155.1.146.1.
DHCPD: Seeing if there is an internally specified pool class:
  DHCPD: htype 1 chaddr 0007.ebde.5622
  DHCPD: circuit id 01f50000
DHCPD: Sending DHCPOFFER to client 0063.6973.636f.2d30.3030.372e.6562.6465.2e35.3632.322d.4574.302f.31 (155.1.146.4)
DHCPD: unicasting BOOTREPLY for client 0007.ebde.5622 to relay 155.1.146.1.
DHCPD: DHCPREQUEST received from client 0063.6973.636f.2d30.3030.372e.6562.6465.2e35.3632.322d.4574.302f.31.
DHCPD: Sending notification of ASSIGNMENT:
```

https://t.me/learningnets

```
 DHCPD: address 155.1.146.4 mask 255.255.255.0
  DHCPD: lease time remaining (secs) = 4294967295
DHCPD: No default domain to append - abort update
DHCPD: Sending DHCPACK to client 0063.6973.636f.2d30.3030.372e.6562.6465.2e35.3632.322d.4574.302f.31 (155.1.146.4).
DHCPD: unicasting BOOTREPLY for client 0007.ebde.5622 to relay 155.1.146.1.

Rack1SW2#TFTP: read request from host 155.1.146.4(56461) via Vlan58
TFTP: Opened flash:Rack1R4-confg, fd 0, size 1494
TFTP: Sending block 1 (retry 0), socket_id 0x2373918
TFTP: Received ACK for block 1, socket_id 0x2373918
TFTP: Sending block 2 (retry 0), socket_id 0x2373918
TFTP: Received ACK for block 2, socket_id 0x2373918
TFTP: Sending block 3 (retry 0), socket_id 0x2373918
TFTP: Received ACK for block 3, socket_id 0x2373918
TFTP: Finished flash:Rack1R4-confg, time 00:00:00
TFTP: Server request for port 54974, socket_id 0x3B89C38
TFTP: read request from host 155.1.146.4(54974) via Vlan58
TFTP: Opened flash:Rack1R4-confg, fd 0, size 1494
TFTP: Sending block 1 (retry 0), socket_id 0x3B89C38
TFTP: Received ACK for block 1, socket_id 0x3B89C38
TFTP: Sending block 2 (retry 0), socket_id 0x3B89C38
TFTP: Received ACK for block 2, socket_id 0x3B89C38
TFTP: Sending block 3 (retry 0), socket_id 0x3B89C38
TFTP: Received ACK for block 3, socket_id 0x3B89C38TFTP: Finished flash:Rack1R4-confg, time 00:00:00

R4:
32K bytes of NVRAM.
32768K bytes of processor board System flash (Read/Write)
Translating "Rack1SW2"...domain server (155.1.146.6) [OK]

%Error opening tftp://155.1.58.8/network-confg (No such file or directory)
%Error opening tftp://155.1.58.8/cisconet.cfg (No such file or directory)
Domain server mapped address 155.1.146.4 to Rack1R4
Loading rack1r4-confg from 155.1.58.8 (via FastEthernet0/1): !
[OK - 1358 bytes]Configuration mapped ip address 155.1.58.8 to Rack1SW2
 Slot is empty or does not support clock participate
 WIC slot is empty or does not support clock participate
%Error opening tftp://255.255.255.255/network-confg (Timed out)
%Error opening tftp://255.255.255.255/cisconet.cfg (Timed out)

         --- System Configuration Dialog ---

Would you like to enter the initial configuration dialog? [yes/no]: no

Press RETURN to get started!
```

https://t.me/learningnets

```
<snip> *Mar  1 00:01:15.215: AUTOINSTALL: FastEthernet0/1 is assigned 155.1.146.4
*Mar  1 00:01:15.215: AUTOINSTALL: Obtain tftp server name Rack1SW2resolved to 155.1.58.8
*Mar  1 00:01:18.288: AUTOINSTALL: Obtain default router (opt 3) 155.1.146.1

<snip>
Rack1R4#show ip int brief
Interface                  IP-Address      OK? Method
 Status          Protocol FastEthernet0/0            204.12.1.4      YES TFTP
   up                     up      Serial0/0                 unassigned      YES TFTP
   up                     up      Serial0/0.1               155.1.0.4       YES TFTP
   up                     up      FastEthernet0/1           155.1.146.4     YES TFTP
   up                     up      Serial0/1                 155.1.45.4      YES TFTP
   up                     up      Loopback0                 150.1.4.4       YES TFTP
   up                     up
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

Auto-Install over LAN Interfaces Using RARP (pending update)

Task

- Configure R1 to supply an IP address to R4 via RARP on VLAN 146 when it boots.
- SW2 should supply the network and host-specific configuration files to R4.

Configuration

```
R1:
!
! ARP entry for R4’s VLAN 146 interface
!
arp 155.1.146.4 0007.ebde.5622 arpa
!
inteface FastEthernet 0/0
 no shutdown
 ip rarp-server 155.1.146.1
 ip helper-address 155.1.58.8

SW2:
tftp-server flash:rack1r4-confg
tftp-server flash:network-confg
```

Verification

RARP is a simple alternative to using DHCP/BOOTP for IP address configuration on the LAN interface. A router sends out RARP requests for an IP address after it fails to obtain an address via DHCP/BOOTP. Another IOS router could be configured as RARP server, responding to a MAC-to-IP mapping request from a configured static

https://t.me/learningnets

ARP entry. After obtaining an IP address, the router may proceed with TFTP broadcasts for network- and host-specific configuration files. Another router on the same subnet could relay the requests to the TFTP server.

Follow the procedure below to create network-wide and route-specific configuration files and upload them to SW2. The files differ in that network-specific files contain hostname to IP address mapping for R4, as in the previous example.

```
Rack1R4#copy running-config flash:Rack1R4-confg
Destination filename [Rack1R4-confg]?
Erase flash: before copying? [confirm]n
Verifying checksum...  OK (0xC982)
1459 bytes copied in 8.033 secs (182 bytes/sec)
Rack1R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Rack1R4(config)#tftp-server flash:Rack1R4-confg
Rack1SW2#copy tftp: flash:
Address or name of remote host []?150.1.4.4
Source filename []? Rack1R4-confg
Destination filename [Rack1R4-confg]?
Accessing tftp://150.1.4.4/Rack1R4-confg...
Rack1R4(config)#no ip host Rack1R4
Rack1R4(config)#ip host Rack1R4 155.1.146.4
Rack1R4(config)#end
Rack1R4#copy running-config flash:Rack1R4-confg
Destination filename [network-confg]?
Erase flash: before copying? [confirm]n
Verifying checksum...  OK (0xC982)
1459 bytes copied in 8.033 secs (182 bytes/sec)
Rack1R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Rack1R4(config)#tftp-server flash:network-confg
Rack1SW2#copy tftp: flash:
Address or name of remote host []?150.1.4.4

Source filename []? network-confg
Destination filename [network-confg]?
Accessing tftp://150.1.4.4/network-confg...
```

To ensure that SW2 exports both configuration files via TFTP, erase R4's configuration and reload the router. Observe the debugging output on R1 and SW2 to see the auto-configuration process.

```
Rack1SW2#debug tftp events
```

https://t.me/learningnets

```
TFTP Event debugging is on
Rack1SW2#debug tftp packets

TFTP Packet debugging is on
Rack1R1#debug arp
ARP packet debugging is on

Rack1R1#RARP: Rcvd RARP req for 0007.ebde.5622
RARP: Sending reply out FastEthernet0/0
IP RARP: sent rev-rep src 155.1.146.1 000d.bc04.e340,
                 dst 155.1.146.4 0007.ebde.5622 FastEthernet0/0

Rack1SW2#
TFTP: read request from host 155.1.146.4(54757) via Vlan58
TFTP: Opened flash:network-confg, fd 0, size 1440
TFTP: Sending block 1 (retry 0), socket_id 0x3B89500
TFTP: Received ACK for block 1, socket_id 0x3B89500
TFTP: Sending block 2 (retry 0), socket_id 0x3B89500
TFTP: Received ACK for block 1, socket_id 0x3B89500
TFTP: Sending block 2 (retry 1), socket_id 0x3B89500
TFTP: Received ACK for block 2, socket_id 0x3B89500
TFTP: Sending block 3 (retry 0), socket_id 0x3B89500
TFTP: Received ACK for block 2, socket_id 0x3B89500
TFTP: Sending block 3 (retry 1), socket_id 0x3B89500
TFTP: Received ACK for block 3, socket_id 0x3B89500TFTP: Finished flash:network-confg, time 00:00:08
<snip>TFTP: Opened flash:Rack1R4-confg, fd 0, size 1494
TFTP: Sending block 1 (retry 0), socket_id 0x3989B80
TFTP: Received ACK for block 1, socket_id 0x3989B80
TFTP: Sending block 2 (retry 0), socket_id 0x3989B80
TFTP: Received ACK for block 2, socket_id 0x3989B80
TFTP: Sending block 3 (retry 0), socket_id 0x3989B80
TFTP: Received ACK for block 3, socket_id 0x3989B80TFTP: Finished flash:Rack1R4-confg, time 00:00:00

<snip>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

IOS Menus (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- When a user named OPERATOR telnets into R1 and authenticates with the password of CISCO, it should see the following menu:

  Operator Menu

  1 Display IP Routing Table
  2 Display Running Config
  3 Escape to Shell
  4 Disconnect

- Option 1 should show the routing table.
- Option 2 should show the running configuration.
- Option 3 should exit to the exec prompt.
- Option 4 should exit the telnet session.
- Clear the screen when the menu starts.

Configuration

```
R1:

menu OPERATOR title #
Operator Menu
```

https://t.me/learningnets

```
#
menu OPERATOR text 1 Display IP Routing Table
menu OPERATOR command 1 show ip route
menu OPERATOR text 2 Display Running Config
menu OPERATOR command 2 show run
menu OPERATOR text 3 Escape to Shell
menu OPERATOR command 3 menu-exit
menu OPERATOR text 4 Disconnect
menu OPERATOR command 4 exit
menu OPERATOR clear-screen
!
username OPERATOR autocommand menu OPERATOR
username OPERATOR password CISCO
username OPERATOR privilege 15
!
line vty 0 4
 login local
```

Verification

Menus allow simple automation of many network-monitoring tasks for unskilled/untrusted personnel. Each menu selection is associated with a command that is executed when a user chooses the selection. The command is executed with privileges of the local user who executed the menu command, or the privileges of the terminal line if the menu command is associated with a line. Therefore, if the menu uses privileged commands (such as level 15), ensure that they are available in the privilege level associated with the user/line.

The special menu command **menu-exit** allows a user to leave the menu but not disconnect from the line. When this command is executed, the user is put back into the exec shell.

```
R1#telnet 150.1.1.1

Trying 150.1.1.1 ... Open


User Access Verification

Username: OPERATOR
Password: CISCO

Operator Menu
```

https://t.me/learningnets

```
 1          Display IP Routing Table

    2          Display Running Config

    3          Escape to Shell

    4          Disconnect

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
   D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
   N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
   E1 - OSPF external type 1, E2 - OSPF external type 2
   i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
   ia - IS-IS inter area, * - candidate default, U - per-user static route
   o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
   a - application route
   + - replicated route, % - next hop override

Gateway of last resort is not set

     150.1.0.0/32 is subnetted, 5 subnets
C        150.1.1.1 is directly connected, Loopback0
D        150.1.3.3 [90/130816] via 155.1.13.3, 1d22h, GigabitEthernet1.13
D        150.1.6.6 [90/130816] via 155.1.146.6, 1d21h, GigabitEthernet1.146
D        150.1.7.7 [90/131072] via 155.1.146.6, 1d21h, GigabitEthernet1.146
                   [90/131072] via 155.1.13.3, 1d21h, GigabitEthernet1.13
D        150.1.9.9 [90/131328] via 155.1.146.6, 1d21h, GigabitEthernet1.146
               [90/131328] via 155.1.13.3, 1d21h, GigabitEthernet1.13
    155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks
C        155.1.0.0/24 is directly connected, Tunnel0
L        155.1.0.1/32 is directly connected, Tunnel0
D        155.1.7.0/24 [90/3328] via 155.1.146.6, 1d21h, GigabitEthernet1.146
                  [90/3328] via 155.1.13.3, 1d21h, GigabitEthernet1.13
D        155.1.9.0/24 [90/3584] via 155.1.146.6, 1d21h, GigabitEthernet1.146
                     [90/3584] via 155.1.13.3, 1d21h, GigabitEthernet1.13
C        155.1.13.0/24 is directly connected, GigabitEthernet1.13
L        155.1.13.1/32 is directly connected, GigabitEthernet1.13
D        155.1.37.0/24 [90/3072] via 155.1.13.3, 1d21h, GigabitEthernet1.13
D        155.1.67.0/24 [90/3072] via 155.1.146.6, 1d21h, GigabitEthernet1.146

<snip>

Operator Menu


    1          Display IP Routing Table
```

https://t.me/learningnets

```
 2          Display Running Config

    3          Escape to Shell

    4          Disconnect


Building configuration...

Current configuration : 3081 bytes
!
! Last configuration change at 14:38:53 UTC Sun May 11 2014
!
version 15.4
no service timestamps debug uptime
no service timestamps log uptime
no platform punt-keepalive disable-kernel-core
platform console serial
!
hostname R1
!
boot-start-marker
boot-end-marker
<snip>

Operator Menu


    1          Display IP Routing Table

    2          Display Running Config
 3          Escape to Shell


    4          Disconnect
R1#
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

IOS Banners (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Change the default IOS banners on R3 as follows:
  - All connecting users, with the exception to console users, should see the notification message “Welcome to IOS Router”.
  - Before the login prompt is displayed, users should see the message “Please authenticate yourself”.
  - Before a user can see the shell prompt, display “Hi, you are on the line <Line>, have a nice time at <Hostname>”, where <Line> is the line number the user is connected to and <Hostname> is the current router’s hostname.
  - Set the banner for reverse telnet connections to “This is a reverse telnet connection”.

Configuration

R3:

```
banner motd #
Welcome to IOS Router
#
banner login #
Please authenticate yourself
#
banner exec #
Hi, you are on the line $(line), have a nice time at $(hostname)
```

https://t.me/learningnets

```
#
banner incoming #
This is a reverse telnet connection
#
line console 0
 no motd-banner
 no exec-banner
```

Verification

For remote access and local console connections, the following types of banners can be displayed:

- MOTD banner – displayed first to everyone connecting to the router
- Login banner – displayed before the authentication prompt is presented to the user (if login is disabled, the banner is not displayed)
- Exec-banner – displayed before the exec prompt is displayed to the user
- Incoming-banner – displayed to users connecting via reverse telnet

Special tokens can be used for the banner messages such as $(line) or $(hostname) to create dynamic notifications. Some lines can be selectively exempted from viewing banner messages, such as the console is in this example.

```
R3#telnet 150.1.3.3
Trying 150.1.3.3 ... Open
Welcome to IOS Router
Please authenticate yourself



User Access Verification


Password: ciscoHi, you are on the line 130, have a nice time at R3



R3#
```

Login to R3 via the console and type **exit**. Login again to check the banners. Note that only the login banner is displayed.

https://t.me/learningnets

```
R3#exit

 R3 con0 is now available



Press RETURN to get started.
 Please authenticate yourself
```

Simulate a reverse telnet connection to R3's AUX port to verify the incoming-banner message. To do so, configure the AUX line to accept incoming telnet connections. In the below example, the AUX port is line 129.

```
R3:
line aux 0
 no modem inOut transport input telnet
!
!R3#show line

   R3#show line
  Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns   Int
  *     0 CTY              -    -      -    -    -      8       0     0/0       -
  1 AUX   9600/9600  -    -      -    -    -      0       0     0/0       -
  2 VTY              -    -      -    -    -     25       0     0/0       -
  3 VTY              -    -      -    -    -      5       0     0/0       -
  4 VTY              -    -      -    -    -      0       0     0/0       -
  5 VTY              -    -      -    -    -      0       0     0/0       -
  6 VTY              -    -      -    -    -      0       0     0/0       -

Line(s) not in async mode -or- with no hardware support:
1-128
!
!R3#telnet 150.1.3.3 6129
Trying 150.1.3.3, 6129 ... Open

Welcome to IOS Router
 This is a reverse telnet connection
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

KRON Command Schedule (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure a KRON occurrence on R4 that saves the running config to a remote TFTP server daily at 08:00.
- Use the server 155.1.146.100 and the filename **r3-config**.

Configuration

```
R3:

kron occurrence SAVE_DAILY at 8:00 recurring
policy-list SAVE_CONFIG
!
kron policy-list SAVE_CONFIG
cli show running-config | redirect tftp://155.1.146.100/r3-config
```

Verification

The KRON Command Scheduler allows exec commands or TCL scripts to run at a specified time. The KRON configuration consists of a policy-list containing exec commands, and a scheduler to execute the commands in the policy-list at a specified time or recurring interval.

```
R3#debug kron all
```

https://t.me/learningnets

```
All kron debug flags are on
!
!R3#clock set 07:59:59 Jan 1 2010
Clock Set Seen%SYS-6-CLOCKUPDATE: System clock has been updated from 18:26:31 UTC Sun May 11 2014 to
00:00:00 UTC Sun May 11 2014,
 configured from console by console.
R3#
Major 4, Minor 9
Clock diff more than 3hrs 115199990 Start timer for SAVE_DAILY, expires in 28799990 Msec
!
!R3#show kron schedule

Kron Occurrence Schedule SAVE_DAILY inactive, will run again in 0 days 00:00:05 at 8 :00 on
!
!R3#show kron schedule

Kron Occurrence Schedule SAVE_DAILY inactive, will run again in 0 days 07:56:00 at 8 :00 on



Major 4, Minor 9
Clock diff more than 3hrs 115199990
Start timer for SAVE_DAILY, expires in 28799990 Msec
%SYS-5-CONFIG_I: Configured from console by console
<snip>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

EEM Scripting: Interface Events (pending update)

Task

- Create an EEM applet in R5 that sends an email message to **noc@INE.com** from **r5@ine.com** when the Frame-Relay interface receive load exceeds 50% of interface bandwidth.
- Use the SMTP server at the IP address 155.X.146.100.
- After this, apply the access-list CRITICAL_TRAFFIC_ONLY ingress on R5's connection to the Frame-Relay interface.
- The access-list should be pre-configured and permit just the HTTP and DNS traffic responses.
- The interface bandwidth is 64Kbps.

Configuration

Embedded Event Manager is a feature of the Cisco IOS operating system that allows you to write handlers for various system events. The core of EEM is a special process known as an EEM server that acts as a middleware agent between event detectors and EEM event subscribers. There is a fixed number of event detectors that post an event when a programmed condition is met. They include the following:

- CLI event detector – detects various commands typed in CLI based on regular expression matching
- Syslog event detector – responds to various syslog stings, allowing for matching on regular expressions just like CLI detector
- Interface counter – responds to various interface's counters crossing threshold settings
- Counter – responds to the change of value of a generic system counter
- SNMP – monitors specified SNMP objects and posts an event upon the programmed

https://t.me/learningnets

condition met

- None –a special case of event detector triggered when a user issues the command `event manager run` to execute a named EEM script/applet
- Watchdog – generates periodic timer events and allows the EEM script to be run at repeating time intervals

The other parts of the EEM system—event subscribers—are defined and registered with the EEM server as either EEM applets or scripts. Applets are simple programs written using a very basic set of CLI commands that start with an `action` keyword. Scripts are special TCL scripts written to handle EEM events. The applets are easy to write and powerful enough to perform many functions including CLI commands, email sending, SNMP/Syslog message generation, and implementing basic program logic (such as branching or computations). The TCL scripts are more powerful but require more skills to develop. In this workbook, we will be working with EEM applets only.

Every EEM applet has a name and a detector condition defined to trigger the applet. The applet may access global variables defined using the command `event manager environment` or the parameters passed to them by an event detector. There are pre-defined environment variables accessible to the EEM scripts, specific to every event detector. You may find a full list by reading the document “Writing Embedded Event Manager Policies Using Cisco IOS CLI” in the Cisco IOS documentation set. You may also list the variables for an event detector using the command `show event manager detector <NAME> detailed` . The same document describes the most common actions used with EEM applets. For the detailed description of every `action` that could be used in an EEM applet, refer to the “IOS Network Management Command Reference” list and locate the action commands there.

Notice that every applet definition starts with a group of `event` commands that specify the event detector and the condition to trigger the applet. Normally there is only one event defined for an applet—if you define multiple events, you must further specify how they co-relate to each other (covered in a separate task). For the complete list of events supported by EEM with various event detectors, refer to the “IOS Network Management Command Reference” document and locate the `event` commands there.

```
R5:
!
! Notice that the access-list also permits routing traffic and
! the returning STMP packets to allow email sending.
!
ip access-list extended CRITICAL_TRAFFIC
 permit tcp any eq 80 any
 permit udp any eq 43 any
```

https://t.me/learningnets

```
 permit tcp any eq 25 any
 permit udp any any eq rip
!
!
!
no event manager applet INTERFACE_LOAD
event manager applet INTERFACE_LOAD
!
! Notice that the event defined below show take only a single line
! Use CLI help function to enter all the parameter values
!
! The condition specifies to monitor R5's FR interface "TXLoad"
! counter (range 1-255) and post an event when the value crosses
! 153 (60%*255).
!
 event tag 1 interface name Serial0/0/0 parameter rxload entry-op gt entry-val 153 entry-type value poll-interval 30

!
! The first action set applies the  access-list
!
 action 0.0 cli command "enable"
 action 1.0 cli command "conf t"
 action 2.0 cli command "interface Serial 0/0/0"
 action 3.0 cli command "ip access-group CRITICAL_TRAFFIC in"
 action 4.0 mail server "155.1.146.100" to "noc@INE.com" from "r5@INE.com" subject "Interface Alert" body "Interface
!
! We set the load interval to the minimum value to achieve faster
! response of the interface counters.
!
interface Serial 0/0/0
 bandwidth 64
 load-interval 30
```

Verification

We registered the applet with the "none" event detector to be able to run the applet from the CLI for "dry-run" testing purposes. Enable EEM debugging commands to track the CLI and E-Mail actions and run the applet manually first.

```
R5:
event manager applet INTERFACE_LOAD
 event tag 2.0 none
Rack1R5#debug event manager action cli
```

https://t.me/learningnets

```
Debug EEM action cli debugging is on
Rack1R5#debug event manager action mail

Debug EEM action mail debugging is on
Rack1R5#event manager run INTERFACE_LOAD

%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : CTL : cli_open called.
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5>
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack1R5>enable
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5#
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack1R5#conf t
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Enter configuration commands, one per line.  End with CNTL/Z
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5(config)#
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack1R5(config)#interface Serial 0/0/0
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5(config-if)#
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack1R5(config-if)#ip access-group CRITICAL_TRAFFIC in
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5(config-if)#
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(smtp_lib) : smtp_connect_attempt: 1
Rack1R5#
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(smtp_lib) : smtp_connect connect fail 260
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(smtp_lib) : fh_smtp_connect callback timer is awake
%HA_EM-3-FMPD_SMTP: Error occurred when sending mail to SMTP server: 155.1.146.100 : timeout error
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : CTL : cli_close called.
Rack1R5#
%SYS-5-CONFIG_I: Configured from console by vty0
Rack1R5#show ip interface serial 0/0/0 | include access
  Outgoing access list is not set
  Inbound  access list is CRITICAL_TRAFFIC
  IP access violation accounting is disable
R5
:
event manager applet INTERFACE_LOAD
 no event tag 2.0 none
```

Notice that the configuration portion of the script succeeds, whereas the SMTP connection cannot be established to send out an email, because the SMTP server does not exist. Now attempt a “real” test by simulating interface oversubscription. First, make sure the applet is registered with the event detector.

```
Rack1R5#show event manager policy registered event-type interface

No.  Class     Type    Event Type          Trap  Time Registered           Name
1    applet    user    interface           Off   Sat Dec 12 18:51:49 2009  INTERFACE_LOAD
 1: interface: name {Serial0/0/0} parameter {rxload} entry_op gt entry_val 153 entry_type value poll_interval 30.000
 maxrun 20.000
```

https://t.me/learningnets

```
 action 0.0 cli command "enable"
 action 1.0 cli command "conf t"
 action 2.0 cli command "interface Serial 0/0/0"
 action 3.0 cli command "ip access-group CRITICAL_TRAFFIC in"
 action 4.0 mail server "155.1.146.100" to "noc@INE.com" from "r5@INE.com" subject "Interface Alert" body "Interface
```

Flood R5's Frame-Relay interface.

```
Rack1R3#ping 155.1.0.5 repeat 1000000 size 1000 timeout 0


Type escape sequence to abort.
Sending 1000000, 1000-byte ICMP Echos to 155.1.0.5, timeout is 0 seconds:
.................................................................
.................................................................
.................................................................
.................................................................
.................................................................
```

Add one extra debugging command in R5 to track interface detector events, and notice how the applet is triggered.

```
Rack1R5#debug event manager detector interface

Debug EEM Interface Event Detector debugging is on

Rack1R5#fh_fd_intf_param_fetch:I/F=Serial0/0/0 rxload=251
fh_fd_intf_start_poll_timer: start_t=30000

fh_fd_intf_event_register: re=0x66BD3658, sid=17
fh_send_intf_fd_msg: sval=0
fh_fd_intf_process_async
fh_fd_intf_param_fetch:I/F=Serial0/0/0 rxload=251
intf_value_uint_compare:op1=251 op2=153 ret=TRUE
intf_entry_value_check:Returning TRUE
fh_fd_intf_event_match: re = 0x66BD3658, num_matches = 1

fh_fd_intf_start_poll_timer: start_t=30000
fh_fd_intf_process_poll_timer: update_t=0
fh_send_intf_fd_msg: msg_type=64
fh_send_intf_fd_msg: sval=0
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : CTL : cli_open called.
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5>
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack1R5>enable
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5#
```

https://t.me/learningnets

```
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack1R5#conf t
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Enter configuration commands, one per line.  End with CNTL/Z
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : %HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5(config-if)#
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : IN  : Rack1R5(config-if)#ip access-group CRITICAL_TRAFFIC in
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(cli_lib) : : OUT : Rack1R5(config-if)#
%HA_EM-6-LOG: INTERFACE_LOAD : DEBUG(smtp_lib) : smtp_connect_attempt: 1
fh_fd_intf_process_async
fh_fd_intf_param_fetch:I/F=Serial0/0/0 rxload=255
intf_exit_comb_check:Returning TRUE
intf_value_uint_compare:op1=255 op2=153 ret=TRUE
intf_entry_value_check:Returning TRUE
fh_fd_intf_event_match: re = 0x66BD3658, num_matches = 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

EEM Scripting: Syslog Events (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create an EEM script in R5 that applies the “no shutdown” command every time someone attempts to shutdown the interface GigabitEthernet1.
- Send an email containing the result of the command “show users” to the address admin@INE.com if this event occurs.
- Make sure that your script does not apply to any other interface.

Configuration

In this example, we deal with the Syslog event detector. When the interface has been shutdown, the following message will be posted to syslog:

```
%LINK-5-CHANGED: Interface GigabitEthernet1, changed state to administratively down
```

We create an event subscriber to intercept this message and apply the `no shutdown` command to the interface. Notice the regular expression used to match the Syslog messages. Also, pay attention to the CLI command flow; it should be exactly the same set you would use to configure something manually.

```
R5:

event manager applet INTERFACE_SHUTDOWN
event tag 1.0 syslog pattern "Interface GigabitEthernet1.*changed.*down"
action 1.0 cli command "enable"
```

https://t.me/learningnets

```
action 2.0 cli command "conf t"
action 3.0 cli command "interface GigabitEthernet1"
action 4.0 cli command "no shutdown"
action 5.0 cli command "end"
action 6.0 cli command "show users"
action 7.0 mail server "155.1.146.100" to "admin@INE.com" from "r5@INE.com" subject "Interface Shutdown Alert" body
```

Verification

Start by making sure your applet is registered with the EEM server.

```
R5#show event manager policy registered
No.  Class     Type    Event Type          Trap  Time Registered           Name
1    applet    user    syslog              Off   Thu May 8 18:53:33 2014   INTERFACE_SHUTDOWN
1.0: syslog: pattern {Interface GigabitEthernet1.*changed.*down}
maxrun 20.000
action 1.0 cli command "enable"
action 2.0 cli command "conf t"
action 3.0 cli command "interface GigabitEthernet1"
action 4.0 cli command "no shutdown"
action 5.0 cli command "end"
action 6.0 cli command "show users"
action 7.0 mail server "155.1.146.100" to "admin@INE.com" from "r5@INE.com" subject "Interface Shutdown Alert" body
```

Enable EEM action and detector debugging and shutdown the GigabitEthernet1 interface. Observe the event being detected and the applet triggered.

```
R5#debug event manager detector syslog
 R5#debug event manager action cli
      R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#interface serial GigabitEthernet1
R5(config-if)#shutdown



fh_fd_syslog_event_match: num_matches = 0
%LINK-5-CHANGED: Interface GigabitEthernet1, changed state to administratively down
syslog_pubinfo_enqueue: matched message="
%LINK-5-CHANGED: Interface GigabitEthernet1, changed state to administratively down"
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : CTL : cli_open called.
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : OUT : R5>
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : IN  : R5>enable
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : OUT : R5#
```

https://t.me/learningnets

```
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : IN  : R5#conf t
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : OUT : Enter configuration commands, one per line.  End with CN
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : OUT : R5(config)#
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : IN  : R5(config)#interface GigabitEthernet1
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : OUT : R5(config-if)#
%HA_EM-6-LOG: INTERFACE_SHUTDOWN : DEBUG(cli_lib) : : IN  : R5(config-if)#no shutdown
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

EEM Scripting: CLI Events (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 so that every time someone types the `show run` command, it does not display the usernames.

Configuration

In this example, we learn about the CLI event detector. This detector allows for monitoring certain CLI patterns and publishes an event if a match occurs. The event monitoring configuration could use the parameter sync set to either “yes” or “no” to define a synchronous or asynchronous applet. When the applet event condition is synchronous, the EEM server will hold the matched CLI command execution until the script terminates. The script should return an exit value in the variable `$_exit_status` , and this will determine whether the triggered command will run (status 1) or not (status 0). Asynchronous event handlers will let the CLI command execute and the event will be posted after that. The script cannot affect the command execution. Notice that asynchronous CLI events require a set of additional parameters, such as number of occurrences and the time window for the occurrences.

Another command we use in the script below is the puts action. It allows you to display arbitrary text on the console, provided that the script is synchronous.

```
R5:

no event manager applet SHOW_RUN_FILTER
```

https://t.me/learningnets

```
event manager applet SHOW_RUN_FILTER
 event tag 1.0 cli pattern "show run" sync yes
 action 1.0 cli command "enable"
 action 2.0 cli command "show run | exclude username"
 action 3.0 puts $_cli_result
!
! Exit status of 0 block execution of the original command
!
 action 4.0 set $_exit_status 0
```

Verification

Start by making sure your applet is registered with the EEM server.

```
R5#show event manager policy registered event-type cli

No.  Class     Type    Event Type   Trap  Time Registered           Name
1    applet    user    cl           Off   Thu May 8 19:12:21 2014   SHOW_RUN_FILTER

1.0: cli: pattern {show run} sync yes
maxrun 20.000
action 1.0 cli command "enable"
action 2.0 cli command "show run | exclude username"
action 3.0 puts "$_cli_result"
action 4.0 set $_exit_status "0"
```

Enable EEM action and detector debugging, and execute the command `show running-config` . Notice that the output will not contain the usernames, because the actual command has been substituted with a “filtered” version.

```
R5#debug event manager detector cli
R5#debug event manager action cli



Building configuration...

Current configuration : 4010 bytes
!
! Last configuration change at 19:12:21 UTC Thu May 8 2014
!
version 15.4
no service timestamps debug uptime
no service timestamps log uptime
no platform punt-keepalive disable-kernel-core
```

https://t.me/learningnets

```
platform console serial
!
hostname R5
!
boot-start-marker
boot-end-marker
!
check_eem_cli_policy_handler: num_matches = 1, response_code = 4

%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : CTL : cli_open called.
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : R5>
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : IN  : R5>enable
cli_history_entry_add: free_hist_list size=0, hist_list size=7
eem_no_scan flag set, skipping scan of command_string=enable
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT :
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : IN  : R5#show run | exclude username
cli_history_entry_add: free_hist_list size=0, hist_list size=7
eem_no_scan flag set, skipping scan of command_string=show running-config
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : Building configuration...
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT :
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : Current configuration : 4010 bytes
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : !
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : ! Last configuration change at 19:12:21 UTC Thu May 8 2014
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : !
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : version 15.4
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : no service timestamps debug uptime
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : no service timestamps log uptime
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT
no platform punt-keepalive disable-kernel-core
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : platform console serial
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : !
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : hostname R5
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : !
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : boot-start-marker
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : boot-end-marker
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : !
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : !
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : !
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : OUT : no aaa new-model
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : CTL : 20+ lines read from cli, debug output truncated
%HA_EM-6-LOG: SHOW_RUN_FILTER : DEBUG(cli_lib) : : CTL : cli_close called.
fh_fd_cli_publish_done: errc=0, publish_expired=0
sid=2, ptp=0x7F5322FAFCE8, connp=0x7F5312FA82A0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

EEM Scripting: Periodic Scheduling (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 to send a copy of the running configuration every 5 minutes to noc@INE.com.
- Log a syslog message “Configuration saved” every time the script does that.
- Use the same mail server as in the previous tasks.

Configuration

Event applets may be configured to respond to periodic or timed events—for example, to fire every time interval or start at a fixed time in the future. In this example, we create a script that registers with the watchdog event publisher and runs every 300 seconds to save and email the running configuration.

https://t.me/learningnets

```
R5:

no event manager applet SHOW_RUN_EVERY_5MIN
event manager applet SHOW_RUN_EVERY_5MIN
event tag 1.0 timer watchdog time 300
!
! We use write term as we intercepted show runin the previous task
!
 action 1.0 cli command "enable"
 action 2.0 cli command "write term"
 action 3.0 syslog msg "Configuration Saved"
 action 4.0 mail server "155.1.146.100" to "noc@INE.com" from "r5@INE.com" subject "Configuration" body " $_cli_resu
```

Verification

Start by making sure your applet is registered with the EEM server.

```
R5#show event manager policy registered event-type timer-watchdog

No.  Class     Type    Event Type          Trap  Time Registered           Name
1    applet    user    timer watchdog      Off   Thu May 8 19:48:18 2014   SHOW_RUN_EVERY_5MIN
1.0: timer watchdog: time 300.000
maxrun 20.000
action 1.0 cli command "enable"
action 2.0 cli command "write term"
action 3.0 syslog msg "Configuration Saved"
action 4.0 mail server "155.1.146.100" to "noc@INE.com" from "r5@INE.com" subject "Configuration" body " $_cli_resul
```

Enable EEM detector debugging and wait about five minutes for the event to be published.

```
R5#debug event manager detector timer

Debug EEM Timer Event Detector debugging is on

fh_fd_timer_process_async
fh_fd_timer_event_expire: re=0x66BD3018
fh_send_timer_fd_msg: msg_type=64
fh_send_timer_fd_msg: sval=0
!
!
```

https://t.me/learningnets

```
R5#%HA_EM-6-LOG: SHOW_RUN_EVERY_5MIN: Configuration Saved


R5#
!
!
%HA_EM-3-FMPD_SMTP: Error occurred when sending mail to SMTP server: 155.1.146.100 : timeout error
R5# _fd_timer_process_async
cron_tick: num_matches 0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - System Management

EEM Scripting: Advanced Features (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **System Management Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create an EEM applet in R5 that clears all interface counters every 3 minutes.
- Ensure that the Tunnel interfaces counters are never cleared, though.
- Account for any interfaces added to the router in the future.

Configuration

Below is an example of an advanced script that uses loops and conditional constructs. You may see the well-known foreach and if/else blocks being implemented for EEM applets. A fewother interesting action are:

- `info type interface-name` – pulls out the list of the interface names in the system matching a regular expression.
- `regexp PATTERN STRING` – performs regular expression match against a string. If there is a match, the variable $_regexp_result is set to “1”. This command takes extra arguments for extracting the matched substrings.
- `continue` – classic structured programming command; starts another iteration.
- `cli command COMMAND pattern PATTERN` – this syntax allows for the CLI commands that await a user’s response, such as a confirmation. The PATTERN is the regular expression matching the command prompt.

```
R5:

no event manager applet CLEAR_COUNTERS
```

https://t.me/learningnets

```
event manager applet CLEAR_COUNTERS
 event tag 1.0 timer watchdog time 180
 action 1.0 info type interface-names regexp "Gig"
 action 2.0 foreach _iface $_info_interface_names
 action 3.0 regexp ".*(Tunnel).*" $_iface
 action 4.0 if $_regexp_result eq "1"
 action 5.0 continue
 action 6.0 else
 action 7.0 cli command "enable"
 action 8.0 cli command "clear counters $_iface" pattern "confirm"
 action 9.0 cli command "y"
 action 9.1 end
 action 9.2 end
```

Verification

Complex scripts are better be debugged by associating with the “none” event detector. You may then enable various event manager debugging and run the script as many times as you want.

```
R5#sh inter GigabitEthernet1
GigabitEthernet1 is up, line protocol is up
Hardware is CSR vNIC, address is 0050.568d.74dd (bia 0050.568d.74dd)
MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
 reliability 255/255, txload 1/255, rxload 1/255
Encapsulation 802.1Q Virtual LAN, Vlan ID  1., loopback not set
Keepalive set (10 sec)
Auto Duplex, Auto Speed, link type is auto, media type is RJ45
output flow-control is unsupported, input flow-control is unsupported
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:07, output 00:00:39, output hang never
Last clearing of "show interface" counters 00:02:59
Input queue: 1/375/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 3000 bits/sec, 3 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec776 packets input, 84310 bytes, 0 no buffer
%CLEAR-5-COUNTERS: Clear counter on interface GigabitEthernet1 by  on vty0 (EEM:CLEAR_COUNTERS)
!
!R5#sh inter GigabitEthernet1
GigabitEthernet1 is up, line protocol is up
Hardware is CSR vNIC, address is 0050.568d.74dd (bia 0050.568d.74dd)
MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
 reliability 255/255, txload 1/255, rxload 1/255
```

https://t.me/learningnets

```
Encapsulation 802.1Q Virtual LAN, Vlan ID  1., loopback not set
Keepalive set (10 sec)
Auto Duplex, Auto Speed, link type is auto, media type is RJ45
output flow-control is unsupported, input flow-control is unsupported
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:00, output 00:00:49, output hang never
Last clearing of "show interface" counters 00:00:09
Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 2000 bits/sec, 3 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec 40 packets input, 4046 bytes, 0 no buffer
```

Notice that only the GigabitEthernet interface counters are cleared.

```
R5#sh inter Tunnel0
  5 minute input rate 1000 bits/sec, 1 packets/sec
  5 minute output rate 1000 bits/sec, 1 packets/sec 204196 packets input, 20125500 bytes, 0 no buffer
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Proxy ARP (pending update)

> You must load the initial configuration files for the section, **IP Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations. Reference the **Advanced Technology Labs With Addressing Diagram** to complete this task.

Task

- Configure interface VLAN 146 with the IP address 155.1.146.5/24 on R5, disable proxy ARP in that interface.
- Disable Proxy-ARP on R4's interface connected to VLAN 146.
- Test reachability from R1 to R5, and check the ARP cache of R1.

Configuration

```
R4:
interface GigabitEthernet1.146
 no ip proxy-arp
R5:
interface GigabitEthernet1.146
 encapsulation dot1q 146
 ip address 155.1.146.5 255.255.255.0
 no ip proxy-arp
R6:

interface FastEthernet0/0.146
 ip proxy-arp
 ip local-proxy-arp
```

https://t.me/learningnets

Verification

```
R1#ping 155.1.146.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.5, timeout is 2 seconds:.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 2/3/4 ms
!
!R1#sh ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  155.1.146.1             -   0050.568d.7819  ARPA   GigabitEthernet1.146
Internet  155.1.146.5             0 0050.568d.6e90
  ARPA   GigabitEthernet1.146   Internet  155.1.146.6             3 0050.568d.6e90
  ARPA   GigabitEthernet1.146
!
!R6#show interfaces GigabitEthernet 1 | include bia
Hardware is CSR vNIC, address is 0050.568d.6e90
 (bia 0050.568d.6e90)
```

The MAC addresses in R1's ARP cache for R5 and R6 are the same, because R6 performs Proxy-ARP for all of R1's requests.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP Server (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R6 as a DHCP server for hosts on VLAN 146, but configure it to ignore BOOTP packets.
- Store the DHCP binding in the flash memory of R6.
- Clients should use R4 and R6 as the redundant default gateways.
- DHCP clients should use the default-router IP addresses for DNS servers.
- Exclude the address range 155.1.146.100 - 155.1.146.254 from allocation.
- Lease the addresses for 12 hours.

Configuration

https://t.me/learningnets

```
R6:

ip dhcp pool VLAN146
 network 155.1.146.0 /24
 default-router 155.1.146.6 155.1.146.4
 dns-server 155.1.146.6 155.1.146.4
lease 0 12
!
no ip bootp server
!
ip dhcp excluded-address 155.1.146.100 155.1.146.254
!
ip dhcp database flash:/bindings
```

Verification

```
R6#show ip dhcp pool

Pool VLAN146 :
Utilization mark (high/low)    : 100 / 0
Subnet size (first/next)       : 0 / 0
Total addresses                : 254
Leased addresses               : 0Excluded addresses              : 155
Pending event                  : none
1 subnet is currently in the pool :
Current index        IP address range                    Leased/Excluded/Total
155.1.146.1          155.1.146.1      - 155.1.146.254     0     / 155   / 254
```

Storing the DHCP database in flash allows R6 to retain host allocations when the router reboots. We can verify the DHCP allocation on R1 removing static IP address configured on VLAN 146 interface.

```
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface GigabitEthernet 1.146
R1(config-subif)#no ip address R1(config-subif)#ip address dhcp
!
!R6#show ip dhcp binding

Bindings from all pools not associated with VRF:
IP address       Client-ID/              Lease expiration        Type       State      Interface
```

https://t.me/learningnets

```
            Hardware address/
            User name
155.1.146.1 0063.6973.636f.2d30.    Jul 23 2014 05:08 AM    Automatic  Active     GigabitEthernet1.146
            3035.302e.3536.3864.
            2e37.3831.392d.4769.
            312e.3134.36
!
!R6#show ip dhcp database
URL       : flash:/bindings
Read      : Never
Written   : Jul 22 2014 05:13 PM
Status    : Last write succeeded. Agent information is up-to-date.
Delay     : 300 seconds
Timeout   : 300 seconds
Failures  : 0 Successes: 1
```

Pitfall

Configure the excluded IP address ranges before configuring the DHCP pool itself. If the pool is configured first, clients may request and be allocated IP addresses faster than the exclusion can be applied.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP Client (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R6 as a DHCP server for hosts on VLAN 146.
- Exclude the address range 155.1.146.100 - 155.1.146.254 from allocation.
- Configure R1 to obtain its IP address for the link to VLAN 146 dynamically via DHCP.
- The DHCP client-id should be based off of the MAC address of the interface connecting to VLAN 146.

Configuration

```
R6:
ip dhcp pool VLAN146
network 155.1.146.0 /24
default-router 155.1.146.6
!
ip dhcp excluded-address 155.1.146.100 155.1.146.254
!R1:

interface GigabitEthernet1.146
 no ip address
 ip address dhcp client-id GigabitEthernet1
```

https://t.me/learningnets

Verification

Enable DHCP packet debugging on the client before requesting the IP address via DHCP.

```
R1#debug dhcp detail

DHCP: DHCP client process started: 10RAC: Starting DHCP discover on GigabitEthernet1.146

DHCP: Try 1 to acquire address for GigabitEthernet1.146
DHCP: allocate request
DHCP: zapping entry in DHC_PURGING state for Gi1.146
```

R1 sends a DHCP DISCOVER on the interface with a source IP address of 0.0.0.0. Note that the client-ID contains the MAC address of the GigabitEthernet interface.

```
DHCP: SDiscover attempt # 1 for entry:
Temp IP addr: 0.0.0.0  for peer on Interface: GigabitEthernet1.146
Temp  sub net mask: 0.0.0.0
DHCP Lease server: 0.0.0.0, state: 3 Selecting
DHCP transaction id: 10CD
Lease: 0 secs,  Renewal: 0 secs,  Rebind: 0 secs
Next timer fires after: 00:00:04
Retry count: 1   Client-ID: cisco-0050.568d.7819-Gi1.146
Client-ID hex dump: 636973636F2D303035302E353638642E
                    373831392D4769312E313436Hostname: R1
```

R1 receives the OFFER message from R6, as seen in the Server ID field. Additional DHCP options, such as DNS, are also seen here inside the offer.

```
DHCP: Received a BOOTREP pkt
DHCP: Scan: Message type: DHCP Offer
DHCP: Scan: Client ID: cisco-0050.568d.7819-Gi1.146
DHCP: Scan: Server ID Option: 155.1.146.6 = 9B019206
DHCP: Scan: Lease Time: 86400
DHCP: Scan: Renewal time: 43200
DHCP: Scan: Rebind time: 75600
DHCP: Scan: Subnet Address Option: 255.255.255.0
DHCP: Scan: Router Option: 155.1.146.6
DHCP: rcvd pkt source 155.1.146.6,  destination:  255.255.255.255
UDP  sport: 43,  dport: 44,  length: 319
DHCP op: 2, htype: 1, hlen: 6, hops: 0
```

https://t.me/learningnets

```
DHCP server identifier: 155.1.146.6
        xid: 10CD, secs: 0, flags: 8000
        client: 0.0.0.0, your: 155.1.146.1
        srvr:   0.0.0.0, gw: 0.0.0.0
        options block length: 71

DHCP Offer Message   Offered Address: 155.1.146.1
DHCP: Lease Seconds: 86400    Renewal secs:  43200    Rebind secs:   75600
DHCP: Server ID Option: 155.1.146.6DHCP: offer received from 155.1.146.6
```

R1 sends a DHCP REQUEST for the IP address it was offered.

```
DHCP: SRequest attempt # 1 for entry:

Temp IP addr: 155.1.146.1  for peer on Interface: GigabitEthernet1.146
Temp  sub net mask: 255.255.255.0
DHCP Lease server: 155.1.146.6, state: 4 Requesting
DHCP transaction id: 10CD
Lease: 86400 secs,  Renewal: 0 secs,  Rebind: 0 secs
Next timer fires after: 00:00:03
Retry count: 1   Client-ID: cisco-0050.568d.7819-Gi1.146
Client-ID hex dump: 636973636F2D303035302E353638642E
                    373831392D4769312E313436
Hostname: R1
DHCP: SRequest- Server ID option: 155.1.146.6
DHCP: SRequest- Requested IP addr option: 155.1.146.1
DHCP: SRequest placed lease len option: 86400
DHCP: SRequest: 311 bytes
DHCP: SRequest: 311 bytes
            B'cast on GigabitEthernet1.146 interface from 0.0.0.0
```

R1 receives a DHCP ACK message from the server, and the allocation is complete.

```
DHCP: Received a BOOTREP pkt

DHCP: Scan: Message type: DHCP Ack
DHCP: Scan: Client ID: cisco-0050.568d.7819-Gi1.146
DHCP: Scan: Server ID Option: 155.1.146.6 = 9B019206
DHCP: Scan: Lease Time: 86400
DHCP: Scan: Renewal time: 43200
DHCP: Scan: Rebind time: 75600
DHCP: Scan: Subnet Address Option: 255.255.255.0
DHCP: Scan: Router Option: 155.1.146.6
DHCP: rcvd pkt source: 155.1.146.6,  destination:  255.255.255.255
```

https://t.me/learningnets

```
UDP  sport: 43,  dport: 44,  length: 319
DHCP op: 2, htype: 1, hlen: 6, hops: 0
DHCP server identifier: 155.1.146.6
        xid: 10CD, secs: 0, flags: 8000
        client: 0.0.0.0, your: 155.1.146.1
        srvr:   0.0.0.0, gw: 0.0.0.0
        options block length: 71
```

The binding can be verified on the DHCP server, as seen below.

```
R6#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/              Lease expiration        Type       State      Interface
            Hardware address/
            User name
155.1.146.1  0063.6973.636f.2d30.    Jul 23 2014 05:27 PM    Automatic  Active     GigabitEthernet1.146

            3035.302e.3536.3864.
            2e37.3831.392d.4769.
            312e.3134.36
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP Relay (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R6 to allocate IP addresses via DHCP to clients on VLAN 58.
- The clients on VLAN 58 should use R5 as the default gateway.
- Configure R8's VLAN 58 interface to obtain an IP address via DHCP.
- Base the client-id on the MAC address of its VLAN 58 interface.
- Ensure avoidance of IP address conflicts with the existing hosts on VLAN 58.

Configuration

```
R6:
ip dhcp pool VLAN58
 network 155.1.58.0 /24
 default-router 155.1.58.5
!
ip dhcp excluded-address 155.1.58.5
R5:
interface GigabitEthernet1.58
 ip helper-address 155.1.146.6
R8:

interface GigabitEthernet1.58
 ip address dhcp client-id GigabitEthernet1
```

https://t.me/learningnets

Verification

DHCP relay acts as a proxy agent, forwarding all broadcast packets from the clients to the unicast IP address of the actual DHCP server. The server uses a special “giaddr” field in the DHCP packets that contains the IP address of the DHCP relay interface. This field allows the server to select the proper DHCP pool and then unicast a reply back to the DHCP relay agent.

To verify the configuration, enable DHCP debugging on the server and request a new IP address on R8.

```
R6#debug ip dhcp server packet
R6#debug ip dhcp server events
```

R6 receives a DHCP DISCOVER packet from the relay IP address (R5) with the Client-ID based on the MAC address of R8’s VLAN 58 interface.

```
DHCPD: DHCPDISCOVER received from client 0100.5056.8d35.58 through relay 155.1.58.5.
DHCPD: Sending notification of DISCOVER:
DHCPD: htype 1 chaddr 0050.568d.3558
DHCPD: remote id 020a00009b01920600000092 DHCPD: giaddr = 155.1.58.5

DHCPD: interface = GigabitEthernet1.146
```

R6 sends a DHCP OFFER packet to the IP address of the relay (R5).

```
DHCPD: assigned IP address 155.1.58.2 to client 0100.5056.8d35.58.
DHCPD: Sending DHCPOFFER to client 0100.5056.8d35.58 (155.1.58.2).
DHCPD: unicasting BOOTREPLY for client 0050.568d.3558 to relay 155.1.58.5.
```

R8 requests the IP address using a REQUEST packet.

```
DHCPD: DHCPREQUEST received from client 0100.5056.8d35.58.
DHCPD: DHCPREQUEST received on interface GigabitEthernet1.146.
DHCPD: Found previous binding
DHCPD: FSM state change INVALID
DHCPD: Client is connected via Relay DHCPD: Sending notification of ASSIGNMENT:

DHCPD: address 155.1.58.2 mask 255.255.255.0
DHCPD: htype 1 chaddr 0050.568d.3558
```

https://t.me/learningnets

```
DHCPD: lease time remaining (secs) = 86400
    DHCPD: interface = GigabitEthernet1.146
```

The server acknowledges the assignment by sending ACK to the relay.

```
DHCPD: Sending DHCPACK to client 0100.5056.8d35.58 (155.1.58.2).
DHCPD: unicasting BOOTREPLY for client 0050.568d.3558 to relay 155.1.58.5.
```

Allocation can be verified by checking the DHCP binding database on R6.

```
R6#show ip dhcp binding

Bindings from all pools not associated with VRF:
IP address      Client-ID/              Lease expiration        Type       State      Interface
            Hardware address/
            User name
155.1.58.2  0100.5056.8d35.58        May 03 2014 06:49 AM    Automatic  Active     Unknown
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP Host Pools (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R6 as a DHCP server for the hosts in VLAN 146.
- Configure R1 as the DHCP client with client-id of GigabitEthernet1 interface.
- Create a DHCP host pool on R6 corresponding to R1's VLAN 146 interface to lease R1 the IP address 155.1.146.1.

Configuration

```
R6:
ip dhcp pool VLAN146
 network 155.1.146.0 /24
 default-router 155.1.146.6
R1:

interface GigabitEthernet1.146
 ip address dhcp client-id GigabitEthernet1
```

To find the DHCP Client-ID corresponding to R1, view the binding database on R6. You can easily distinguish a Client-ID from the Hardware Address by the additional byte value "01" in the beginning of the string. You may need to release the IP address of R1 before adding a static host pool in R6. Once you find the hardware address, create a static binding for R1 using respective client-id.

https://t.me/learningnets

```
R6#show ip dhcp binding

                 Hardware address/
            User name
155.1.146.1  0100.5056.8d78.19       May 03 2014 07:08 AM    Automatic  Active     GigabitEthernet1.146

R6#configure terminal
ip dhcp pool R1_HOST client-id 0100.5056.8d78.19
 host 155.1.146.1 /24
```

Verification

Issue the `no ip address dhcp` command on R1's interface followed by `ip address dhcp client-id <Interface>` . Ensure that the **debug dhcp detail** has been enabled on R1 first.

```
R1#debug dhcp detail

DHCP: DHCP client process started: 10
RAC: Starting DHCP discover on GigabitEthernet1.146
DHCP: Try 1 to acquire address for GigabitEthernet1.146
ARP: flush cache for interface GigabitEthernet1.146, by 291D22C
ARP: flush cache for interface GigabitEthernet1.146, by 2920A87
DHCP: allocate request
DHCP: zapping entry in DHC_PURGING state for Gi1.146
DHCP: deleting entry 7F266FFBF488 155.1.146.1 from list
Temp IP addr: 155.1.146.1  for peer on Interface: GigabitEthernet1.146

    Temp  sub net mask: 255.255.255.0
DHCP Lease server: 155.1.146.6, state: 10 Purging
DHCP transaction id: 8EE
Lease: 86400 secs,  Renewal: 43200 secs,  Rebind: 75600 secs
Next timer fires after: 00:00:25
    Retry count: 0   Client-ID: 0050.568d.7819
    Client-ID hex dump: 0050568D7819
    Hostname: R1
```

In addition to allocating a static IP address to R1 per the host pool, the DHCP server (R6) also imports the other settings (DNS servers, default routers, etc.) from the more specific subnet-wide DHCP pool.

```
DHCP: Received a BOOTREP pkt
```

https://t.me/learningnets

```
DHCP: Scan: Message type: DHCP Offer
DHCP: Scan: Client ID:
DHCP: Scan: Server ID Option: 155.1.146.6 = 9B019206
DHCP: Scan: Lease Time: 86400
DHCP: Scan: Renewal time: 43200
DHCP: Scan: Rebind time: 75600
DHCP: Scan: Subnet Address Option: 255.255.255.0DHCP: Scan: Router Option: 155.1.146.6

DHCP: rcvd pkt source: 155.1.146.6,  destination:  255.255.255.255
UDP  sport: 43,  dport: 44,  length: 308
DHCP op: 2, htype: 1, hlen: 6, hops: 0
DHCP server identifier: 155.1.146.6
    xid: C48, secs: 0, flags: 8000
    client: 0.0.0.0, your: 155.1.146.2
    srvr:   0.0.0.0, gw: 0.0.0.0
    options block length: 60

DHCP Offer Message   Offered Address: 155.1.146.2
DHCP: Lease Seconds: 86400    Renewal secs:  43200    Rebind secs:   75600
DHCP: Server ID Option: 155.1.146.6
DHCP Client ID:
DHCP: offer received from 155.1.146.6
```

R1 requests the IP address offered from R6.

```
DHCP: SRequest attempt # 1 for entry:

    Temp IP addr: 155.1.146.2  for peer on Interface: GigabitEthernet1.146
    Temp  sub net mask: 255.255.255.0
DHCP Lease server: 155.1.146.6, state: 4 Requesting
DHCP transaction id: C48
Lease: 86400 secs,  Renewal: 0 secs,  Rebind: 0 secs
Next timer fires after: 00:00:03
Retry count: 1   Client-ID: 0050.568d.7819
    Client-ID hex dump: 0050568D7819
    Hostname: R1
```

The new IP address has been acknowledged by R6.

```
DHCP: Received a BOOTREP pkt

DHCP: Scan: Message type: DHCP Ack
DHCP: Scan: Client ID:
DHCP: Scan: Server ID Option: 155.1.146.6 = 9B019206
DHCP: Scan: Lease Time: 86400
```

https://t.me/learningnets

```
DHCP: Scan: Renewal time: 43200
DHCP: Scan: Rebind time: 75600
DHCP: Scan: Subnet Address Option: 255.255.255.0
DHCP: Scan: Router Option: 155.1.146.6
DHCP: rcvd pkt source: 155.1.146.6,  destination:  255.255.255.255
UDP  sport: 43,  dport: 44,  length: 308
DHCP op: 2, htype: 1, hlen: 6, hops: 0
DHCP server identifier: 155.1.146.6
    xid: C48, secs: 0, flags: 8000
    client: 0.0.0.0, your: 155.1.146.2
    srvr:   0.0.0.0, gw: 0.0.0.0
    options block length: 60
```

When you're done with R1, verify the binding database on R6.

```
R6#show ip dhcp binding

Bindings from all pools not associated with VRF:
IP address       Client-ID/              Lease expiration        Type       State      Interface
             Hardware address/
             User name
155.1.146.1  0100.5056.8d78.19       Infinite                Manual     Selecting  Unknown
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP On-Demand Pool (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **DHCP_PPP Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R3 to assign the IP address 155.1.13.1/24 to R1 using IPCP.
- Create a DHCP pool on R1 that imports the subnet, the netmask, and the DNS servers (R4 and R6) from R3 via IPCP.
- Ensure that EIGRP routing is still functional after this configuration is complete.

Configuration

```
R1:
interface Dialer1
 ppp ipcp dns request
 ppp ipcp mask request
!
ip dhcp pool ODAP_POOL
 import all
 origin ipcp
R3:

interface Virtual-Template1
 ppp ipcp mask 255.255.255.0
 ppp ipcp dns 155.1.146.4 155.1.146.6
```

https://t.me/learningnets

Verification

ODAP allows importing DHCP settings using various methods, one of them being IPCP (Internet Protocol Control Protocol for PPP). IPCP allows importing an IP address along with the subnet mask (while PPP ignores the mask, DHCP can use it) as well as DNS/WINS servers. This is useful in PPPoE deployments, such as with DSL, where the DSL CPE router gets an address assignment through DHCP from the provider, and forwards options such as DNS to the clients behind the CPE.

To verify this, check the DHCP pool settings and verify the imported parameters.

```
R1#show ip dhcp import

 Address Pool Name: ODAP_POOL Domain Name Server(s): 155.1.146.4 155.1.146.6
!
!R1#show ip dhcp pool

Pool ODAP_POOL :
    Utilization mark (high/low)    : 100 / 0
    Subnet size (first/next)       : 0 / 0
    Total addresses                : 254
    Leased addresses               : 0
    Excluded addresses             : 0
    Pending event                  : none
    1 subnet is currently in the pool :
    Current index        IP address range                    Leased/Excluded/Total
155.1.13.1           155.1.13.1        - 155.1.13.254      0      / 0      / 254
```

Verify EIGRP neighbors.

```
R1#show ip eigrp neighbors

EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                 (sec)         (ms)       Cnt Num
0   155.1.13.3              Di1                      13 00:07:46  372  2658  0  9
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP Proxy (pending update)

Task

- Configure R2 to request an IP address on its serial connection to R3 via IPCP.
- R3 should relay this request via DHCP to R6, and R6 should allocate the IP address 155.X.23.2 to R2.

Configuration

```
R1:
ip route 155.1.23.0 255.255.255.0 155.1.13.3

R2:
interface Serial 0/1
 encapsulation ppp
 ip address negotiated
 no peer neighbor-route

R3:
interface Serial 1/3
 encapsulation ppp
 ip address 155.1.23.3 255.255.255.0
 peer default ip address dhcp
 no peer neighbor-route
!
ip address-pool dhcp-proxy-client
ip dhcp-server 155.1.146.6

R6:
ip dhcp pool R2_SERIAL
 network 155.1.23.0 /24
!
```

https://t.me/learningnets

```
ip dhcp excluded-address 155.1.23.1
ip dhcp excluded-address 155.1.23.3 155.1.23.254
!
ip route 155.1.23.0 255.255.255.0 155.1.146.1
```

Verification

Pitfall

> The static route is needed in this exercise because R3's link to R2 isn't in the UP/UP state until IP is actually negotiated. This means that when the proxy request is received at the server, R6 does not have a route back to the relay to send the reply back.

DHCP Proxy is an address allocation solution for PPP. When a client requests an IP address via IPCP, the server may issue a DHCP request (act as a proxy) using its own IP address on the PPP interface as the "giaddr" field.

To verify the task, enable debugging of DHCP events in R3 and R6.

```
Rack1R3#debug dhcp detail
Rack1R6#debug ip dhcp server packet
Rack1R6#debug ip dhcp server event
```

Shut down and no-shutdown R3's connection to R2 and observe the debugging output. Ensure that R3 released the IP address before no-shutting the interface.

```
Rack1R3#
```

R3 sends a DHCP discover packet to the globally configured DHCP server as a proxy IPCP address request.

```
DHCP: proxy allocate request
DHCP: zapping entry in DHC_PURGING state for Se1/3
DHCP: deleting entry 64F20E80 155.1.23.2 from list
Temp IP addr: 155.1.23.2  for peer on Interface: Serial1/3
Temp  sub net mask: 255.255.255.0
   DHCP Lease server: 155.1.146.6, state: 9 Purging
   DHCP transaction id: 152B
   Lease: 86400 secs,  Renewal: 43200 secs,  Rebind: 75600 secs
   Next timer fires after: 00:00:21
```

https://t.me/learningnets

```
   Retry count: 0   Client-ID: cisco-155.1.23.3-Serial1/3
   Client-ID hex dump: 636973636F2D3135352E312E32332E33
                       2D53657269616C312F33
DHCP: new entry. add to queue, interface Serial1/3
DHCP: SDiscover attempt # 1 for entry:
Rack1R3(config-if)#Temp IP addr: 0.0.0.0  for peer on Interface: Serial1/3

Temp  sub net mask: 0.0.0.0
   DHCP Lease server: 0.0.0.0, state: 1 Selecting
   DHCP transaction id: 152C
   Lease: 0 secs,  Renewal: 0 secs,  Rebind: 0 secs
   Next timer fires after: 00:00:04
   Retry count: 1   Client-ID: cisco-155.1.23.3-Serial1/3
   Client-ID hex dump: 636973636F2D3135352E312E32332E33
                       2D53657269616C312F33
   Hostname: Rack1R3
DHCP: SDiscover: sending 292 byte length DHCP packet
DHCP: SDiscover 292 bytes
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1/3, changed state to up
```

R3 receives an OFFER from the DHCP server with the client IP address.

```
DHCP: XID MATCH in dhcpc_for_us()
DHCP: Received a BOOTREP pkt
DHCP: Scan: Message type: DHCP OfferDHCP: Scan:Server ID Option: 155.1.146.6
 = 9B019206
DHCP: Scan: Lease Time: 86400
DHCP: Scan: Renewal time: 43200
DHCP: Scan: Rebind time: 75600
DHCP: Scan: Subnet Address Option: 255.255.255.0
DHCP: rcvd pkt source: 155.1.146.6,  destination:  155.1.23.3
   UDP  sport: 43,  dport: 43
Rack1R3(config,  length: 308
   DHCP op: 2, htype: 1, hlen: 6, hops: 0
   DHCP server identifier: 155.1.146.6
        xid: 152C, secs: 0, flags: 0client: 0.0.0.0, your: 155.1.23.2

        srvr:   0.0.0.0, gw: 155.1.23.3
        options block length: 60

DHCP Offer Message   Offered Address: 155.1.23.2
DHCP: Lease Seconds: 86400    Renewal secs:  43200    Rebind secs:   75600
DHCP: Server ID Option: 155.1.146.6
DHCP: offer received from 155.1.146.6
DHCP: SRequest attempt # 1 for entry:
```

https://t.me/learningnets

```
Temp IP addr: 155.1.23.2  for peer on Interface: Serial1/3
Temp  sub net mask: 255.255.255.0
   DHCP Lease server: 155.1.146.6, state: 2 Requesting
   DHCP transaction id: 152C
   Lease: 86400 secs,  Renewal: 0 secs,  Rebind: 0 secs
   Next timer fires after: 00:00:03
   Retry count: 1   Client-ID: cisco-155.1.23.3-Serial1/3
   Client-ID hex dump: 636973636F2D3135352E312E32332E33
                       2D53657269616C312F33
   Hostname: Rack1R3
```

A request is sent for the IP address offered by the DHCP server.

```
DHCP: SRequest- Server ID option: 155.1.146.6 DHCP: SRequest-Requested IP addr option: 155.1.23.2

DHCP: SRequest placed lease len option: 86400
DHCP: SRequest: 310 bytes
DHCP: SRequest: 310 bytes
DHCP: XID MATCH in dhcpc_for_us()
DHCP: Received a BOOTREP pkt
DHCP: Scan: Message type: DHCP Ack
DHCP: Scan: Server ID Option: 155.1.146.6 = 9B019206
DHCP: Scan: Lease Time: 86400
DHCP: Scan: Renewal time: 43200
DHCP: Scan: Rebind time: 75600
DHCP: Scan: Subnet Address Option: 255.255.255.0
DHCP: rcvd pkt source: 155.1.146.6,  destination:  155.1.23.3
   UDP  sport: 43,  dport: 43,  length: 308
   DHCP op: 2, htype: 1, hlen: 6, hops: 0
   DHCP server identifier: 155.1.146.6
        xid: 152C, secs: 0, flags: 0
        client: 0.0.0.0, your: 155.1.23.2
        srvr:   0.0.0.0, gw: 155.1.23.3
        options block length: 60
```

The server ACKed the request and the IP address is now assigned to R2.

```
DHCP Ack Message
DHCP: Lease Seconds: 86400    Renewal secs:  43200    Rebind secs:   75600
DHCP: Server ID Option: 155.1.146.6
DHCP Proxy Client Pooling: ***Allocated IP address: 155.1.23.2
```

The debugging output at R6 is as follows. The server receives a DHCP DISCOVER

https://t.me/learningnets

message from the relay, R3.

```
Rack1R6#
DHCPD: DHCPDISCOVER received from client 0063.6973.636f.2d31.3535.2e31.2e32.332e.332d.5365.7269.616c.312f.33
through relay 155.1.23.3.

DHCPD: Seeing if there is an internally specified pool class:
  DHCPD: htype 1 chaddr 0007.eb49.04c1
  DHCPD: remote id 020a00009b01920600000092
  DHCPD: circuit id 00000000
DHCPD: Allocate an address without class information (155.1.23.0)
DHCPD: Adding binding to radix tree (155.1.23.2)
DHCPD: Adding binding to hash tree
DHCPD: assigned IP address 155.1.23.2 to client 0063.6973.636f.2d31.3535.2e31.2e32.332e.332d.5365.7269.616c.312f.33.
```

The server sends a DHCP OFFER message to the relay.

```
DHCPD: Sending DHCPOFFER to client 0063.6973.636f.2d31.3535.2e31.2e32.332e.332d.5365.7269.616c.312f.33 (155.1.23.2).
DHCPD: unicasting BOOTREPLY for client 0007.eb49.04c1 to relay 155.1.23.3.
```

The client requests the IP address and the server acknowledges the request.

https://t.me/learningnets

```
DHCPD: DHCPREQUEST received from client 0063.6973.636f.2d31.3535.2e31.2e32.332e.332d.5365.7269.616c.312f.33.
DHCPD: Sending notification of ASSIGNMENT:
 DHCPD: address 155.1.23.2 mask 255.255.255.0
  DHCPD: htype 1 chaddr 0007.eb49.04c1
  DHCPD: lease time remaining (secs) = 86400
DHCPD: No default domain to append - abort updateDHCPD: Sending DHCPACK to client
 0063.6973.636f.2d31.3535.2e31.2e32.332e.332d.5365.7269.616c.312f.33 (155.1.23.2).
DHCPD: unicasting BOOTREPLY for client 0007.eb49.04c1 to relay 155.1.23.3.
DHCPD: checking for expired leases.
DHCPD: checking for expired leases.
DHCPD: writing bindings to flash:/bindings.
Rack1R6#show ip dhcp binding
 Bindings from all pools not associated with VRF:
IP address          Client-ID/            Lease expiration        Type
Hardware address/
User name
155.1.23.2          0063.6973.636f.2d31.    Jul 09 2008 10:05 PM    Automatic
3535.2e31.2e32.332e.
332d.5365.7269.616c.
312f.33
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP Information Option (pending update)

Task

- Configure R5 to insert the DHCP information option field into DHCP requests forwarded to R6 from clients on VLAN 58.
- R5 should set the subscriber-id string for these requests to **VLAN58**.
- Configure R6 to allocate IP addresses based on Option 82 and assign a static IP address to R8 based on the information option contents.

Configuration

```
R5:
ip dhcp relay information option
!
interface GigabitEthernet1.58
 ip dhcp relay information option subscriber-id VLAN58


R6:
ip dhcp class TEST
   relay agent information
      relay-information hex 020c020a00009b013a05000000000606564c414e3538
!
ip dhcp pool VLAN58
   class TEST
     address range 155.1.58.8 155.1.58.8
```

Verification

The DHCP information option format may vary from vendor to vendor and even from platform to platform. Therefore, IOS matches a variable hex string against an

https://t.me/learningnets

information option (you can use wildcard “*” and bitmask to match Option 82 ranges—for example, remote-id with multiple subscriber-ids) to define a DHCP class.

DHCP debugging commands will not reveal the information option passed in relay packets. To determine the actual value, use the detailed IP packet dumping procedure as follows.

First, create an access-list that matches only DHCP packets.

```
R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R6(config)#access-list 100 permit udp any any eq bootps
```

Enable DHCP packet debugging using an access-list 100 and hidden **dump** option, and configure R8 to release/request an IP address via DHCP.

```
R6#debug ip packet detail 100 dump
IP packet debugging is on (detailed) (dump) for access list 100
R8#conf t
R8(config)#interface GigabitEthernet1.58
R8(config-if)#shutdown
R8(config-if)#no shutdown
```

Look at the packet dumps below. The two captured packets are DHCP DISCOVER and DHCP REQUEST packets sent from R8. Note that the information option (you can locate it by looking for ASCII subscriber-id sring “VLAN58”) is located at the very end of UDP packet.

The information option starts with a byte having decimal value 82 (hex value 0x52, information option number), followed by the total option length (for example, 0x16 = 22 bytes). After that, suboptions follow (such as remote-id, circuit-id). The suboptions are always TLVs—that is, type-length-value blocks, with type and length being 1-byte quantities. Cisco may add sub-TLVs into each suboption to distinguish remote-id type, etc.

Look at the byte-string extracted from the captured DHCP packet.

|5216|020C|020A|00009B013A0500000000|0606|564C414E35|

This is DHCP option 82 (0x52) of total length 22 (0x16). The first suboption is remote-id (0x02 – suboption type) of total length 12 (0xC). The remote-id suboption starts with yet another TLV (type 0x02 length 0x0A) defining remote-id type,

https://t.me/learningnets

followed by 10 bytes of remote-id value. The next suboption is type 0x06 and length 0x06, which represents ASCII value assigned to R5 VLAN58 interface.

However, in a DHCP class, you need to match either a complete Option 82 or its part, using either a wildcard “*” or bitmask. In our case, the whole content of Option 82 (starting after initial 0x52 0x16) is matched by the class statement.

```
R6#
<snip>
07C57820: 63825363 35010236 049B0192 06330400  c.Sc5..6.....3..
07C57830: 0151803A 040000A8 C03B0400 01275001  .Q.:...(@;...'P.07C57840: 04FFFFFF 0003049B 013A05
52 16020C02
  .........:.R....07C57850:0A00009B 013A0500 00000006 06564C41  .....:.......VLA
07C57860:4E3538FFN58
.

IP: tableid=0, s=155.1.58.5 (FastEthernet0/0.146), d=155.1.146.6 (FastEthernet0/0.146), routed via RIB
IP: s=155.1.58.5 (FastEthernet0/0.146), d=155.1.146.6 (FastEthernet0/0.146), len 350, rcvd 3
    UDP src=67, dst=67

07E28FE0:                         000F 24093CE0            ..$.<`
07E28FF0: 000D2846 8F210800 4500015E 002E0000  ..(F.!..E..^....
07E29000: FD11BA52 9B013A05 9B019206 00430043  }.:R..:......C.C
07E29010: 014A1983 01010601 000010F3 00008000  .J.........s....
07E29020: 00000000 00000000 00000000 9B013A05  ..............:.
07E29030: 001BD4D4 7A420000 00000000 00000000  ..TTzB..........
07E29040: 00000000 00000000 00000000 00000000  ................
07E29050: 00000000 00000000 00000000 00000000  ................
07E29060: 00000000 00000000 00000000 00000000  ................
07E29070: 00000000 00000000 00000000 00000000  ................
07E29080: 00000000 00000000 00000000 00000000  ................
07E29090: 00000000 00000000 00000000 00000000  ................
07E290A0: 00000000 00000000 00000000 00000000  ................
07E290B0: 00000000 00000000 00000000 00000000  ................
07E290C0: 00000000 00000000 00000000 00000000  ................
07E290D0: 00000000 00000000 00000000 00000000  ................
07E290E0: 00000000 00000000 00000000 00000000  ................
07E290F0: 00000000 00000000 00000000 00000000  ................
07E29100: 63825363 35010339 0204803D 0701001B  c.Sc5..9...=....
07E29110: D4D47A42 36049B01 92063204 9B013A08  TTzB6.....2...:.
07E29120: 33040001 51800C08 5261636B 31535732  3...Q...Rack1SW2 07E29130: 37080106 0F2C0321 962B3401 03
521602
  7....,.!.+4..R..07E29140:0C020A00 009B013A 05000000 00060656  .......:.......
V 07E29150:  4C414E35 38FF                        LAN58.
```

https://t.me/learningnets

```
IP: tableid=0, s=155.1.146.6 (local), d=155.1.58.5 (FastEthernet0/0.146), routed via FIB
IP: s=155.1.146.6 (local), d=155.1.58.5 (FastEthernet0/0.146), len 342, sending
    UDP src=67, dst=67

07E28970:                   45000156 001B0000          E..V....
07E28980: FF11B86D 9B019206 9B013A05 00430043  ..8m......:..C.C
07E28990: 0142546B 02010600 000010F3 00008000  .BTk.......s....
07E289A0: 00000000 9B013A08 00000000 9B013A05  ......:.......:.
07E289B0: 001BD4D4 7A420000 00000000 00000000  ..TTzB..........
07E289C0: 00000000 00000000 00000000 00000000  ................
07E289D0: 00000000 00000000 00000000 00000000  ................
07E289E0: 00000000 00000000 00000000 00000000  ................
07E289F0: 00000000 00000000 00000000 00000000  ................
07E28A00: 00000000 00000000 00000000 00000000  ................
07E28A10: 00000000 00000000 00000000 00000000  ................
07E28A20: 00000000 00000000 00000000 00000000  ................
07E28A30: 00000000 00000000 00000000 00000000  ................
07E28A40: 00000000 00000000 00000000 00000000  ................
07E28A50: 00000000 00000000 00000000 00000000  ................
07E28A60: 00000000 00000000 00000000 00000000  ................
07E28A70: 00000000 00000000 00000000 00000000  ................
07E28A80: 63825363 35010536 049B0192 06330400  c.Sc5..6.....3..
07E28A90: 0151803A 040000A8 C03B0400 0127500C  .Q.:...(@;...'P.
07E28AA0: 08526163 6B315357 320104FF FFFF0003  .Rack1SW2.......07E28AB0: 049B013A 05
521602 0C020A00 009B013A
  ...:.R.........:07E28AC0:05000000 00060656 4C414E35 38FF      .......VLAN58.
```

Observe how R6 allocates a DHCP address based on the class. Enable DHCP server debugging and let SW2 renew its IP address.

```
R6#debug ip dhcp server packet
 R6#debug ip dhcp server event
  R6#debug ip dhcp server class
R8#conf t
R8(config)#interface GigabitEthernet1.
R8(config-if)#shutdown
R8(config-if)#no shutdown
R6#

DHCPD: DHCPRELEASE message received from client 0100.1bd4.d47a.42 (155.1.58.8).
DHCPD: Sending notification of TERMINATION:
 DHCPD: address 155.1.58.8 mask 255.255.255.0
 DHCPD: reason flags: RELEASE
  DHCPD: htype 1 chaddr 001b.d4d4.7a42
```

https://t.me/learningnets

```
  DHCPD: lease time remaining (secs) = 84898
DHCPD: returned 155.1.58.8 to address pool VLAN58.
DHCPD: checking for expired leases.
DHCPD: DHCPRELEASE message received from client 0100.1bd4.d47a.42 (155.1.58.8).
DHCPD: Finding a relay for client 0100.1bd4.d47a.42 on interface FastEthernet0/0.146.
```

Note that remote-id and circuit-id shown in DHCP debugging output do not represent the information received from R5.

```
DHCPD: Seeing if there is an internally specified pool class:DHCPD: htype 1 chaddr 001b.d4d4.7a42
DHCPD: remote id 020a00009b01920600000092
DHCPD: circuit id 00000000
DHCPD: Sending notification of DISCOVER:
DHCPD: htype 1 chaddr 001b.d4d4.7a42
DHCPD: remote id 020a00009b01920600000092
DHCPD: circuit id 00000000

DHCPD: DHCPDISCOVER received from client 0100.1bd4.d47a.42 through relay 155.1.58.5.
DHCPD: Seeing if there is an internally specified pool class:
  DHCPD: htype 1 chaddr 001b.d4d4.7a42
  DHCPD: remote id 020a00009b01920600000092
  DHCPD: circuit id 00000000
```

The configured DHCP class matched the information option received, and therefore the configured address subrange is used for allocation.

https://t.me/learningnets

```
DHCPD: Class 'TEST' matched by default
DHCPD: Searching for a match to 'relay-information 020c020a00009b013a05000000000606564c414e3538' in class TEST
DHCPD: input pattern 'relay-information 020c020a00009b013a05000000000606564c414e3538' matches class TEST

DHCPD: input matches class TEST
DHCPD: Adding binding to radix tree (155.1.58.8)
DHCPD: Adding binding to hash tree
DHCPD: assigned IP address 155.1.58.8 to client 0100.1bd4.d47a.42.
DHCPD: Sending DHCPOFFER to client 0100.1bd4.d47a.42 (155.1.58.8).
DHCPD: unicasting BOOTREPLY for client 001b.d4d4.7a42 to relay 155.1.58.5.
DHCPD: DHCPREQUEST received from client 0100.1bd4.d47a.42.
DHCPD: Sending notification of ASSIGNMENT: DHCPD: address 155.1.58.8 mask 255.255.255.0

  DHCPD: htype 1 chaddr 001b.d4d4.7a42
  DHCPD: lease time remaining (secs) = 86400
DHCPD: No default domain to append - abort update
DHCPD: Sending DHCPACK to client 0100.1bd4.d47a.42 (155.1.58.8).
DHCPD: unicasting BOOTREPLY for client 001b.d4d4.7a42 to relay 155.1.58.5.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DHCP Authorized ARP (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **DHCP_Advanced Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R6 to update the ARP cache on its VLAN 146 interface with the information from the DHCP binding database.
- Ensure that R6 does not accept any ARP mappings on its VLAN 146 interface other than those learned from the DHCP binding database.

Configuration

R6:

```
ip dhcp pool VLAN146
 update arp
!
ip dhcp pool R1_HOST
 update arp
!
interface GigabitEthernet1.146
 arp authorize
!
! R4’s MAC address is statically mapped
!
arp 155.1.146.4 0050.568d.4949 arpa
```

https://t.me/learningnets

Verification

The Update ARP feature allows the DHCP process to populate the ARP cache with the DHCP-based entries. Authorized ARP enabled at interface level will disable any dynamic ARP learning on that interface. Therefore, a static ARP entry for R4 is needed to reach the router (which has a static IP address). Also, make sure of the client IP address as there is no static DHCP binding for R4 and it may choose any IP from the defined range.

```
R6#ping 155.1.146.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.4, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms
```

When the static ARP entry for R4 is removed, it can no longer be reached because ARP learning is disabled.

```
R6(config)#no arp 155.1.146.4 0050.568d.4949 arpa
R6(config)#do ping 155.1.146.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.4, timeout is 2 seconds:.....

Success rate is 0 percent (0/5)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IP SLA (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R6 to automatically monitor VLAN 8's availability using ping and telnet every one minute.

Configuration

```
R6:

ip sla 1
 icmp-echo 155.1.8.8
 timeout 1000
 frequency 1
!
ip sla schedule 1 life forever start-time now
!
ip sla 2
 Tcp-connect 155.1.8.8 23 control disable
 timeout 5000
!
ip sla schedule 2 life forever start-time now
```

Verification

https://t.me/learningnets

Basically, we can verify IP SLA by looking into the *return code* and *successes* counters. If there is anything wrong with reachability or configuration, *Number of failure* counter will be increased.

```
R6#show ip sla statistics 1
R6#show ip sla statistics
IPSLAs Latest Operation Statistics

IPSLA operation id: 1
        Latest RTT: 9 milliseconds
Latest operation start time: 18:27:12 UTC Tue Jul 22 2014 Latest operation return code: OK
Number of successes: 1
Number of failures: 0
Operation time to live: Forever

IPSLA operation id: 2
        Latest RTT: 4 milliseconds
Latest operation start time: 18:27:14 UTC Tue Jul 22 2014 Latest operation return code: OK
Number of successes: 1

Number of failures: 0
Operation time to live: Forever
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Object Tracking (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R6 to automatically monitor VLAN 8's availability using ping and telnet.
- Create two enhanced objects on R6 to track the state of both IP SLA operations.
- Create a third enhanced object on R6 that tracks the first two objects; this third object should only be "up" if both of the first two objects are "up."
- The first two objects should wait 10 seconds before reporting their state from up to down, or from down to up.

Configuration

R6:

```
ip sla 1
 icmp-echo 155.1.8.8
 timeout 1000
 frequency 1
!
ip sla schedule 1 life forever start-time now
!
ip sla 2
 Tcp-connect 155.1.8.8 23 control disable
 timeout 5000
!
ip sla schedule 2 life forever start-time now
!
```

https://t.me/learningnets

```
track 1 ip sla 1 reachability
    delay down 10 up 10
!
track 2 ip sla 2 reachability
    delay down 10 up 10
!
track 3 list boolean and
    object 1
    object 2
```

Verification

In this example, a Boolean AND operation is used to calculate the resulting state of the two member objects. Because AND only returns TRUE if all inputs are TRUE, the state of object 3 is only “up” if both objects 1 and 2 are “up.”

In addition to SLA operations, interface states or IP routes can also be tracked. For IP routes, it is possible to track routing metrics and see when they cross thresholds.

To verify this configuration, display state of the tracked objects as follows.

```
R6#show track

R6#show track
Track 1
    IP SLA 1 reachability Reachability is Up
        1 change, last change 00:04:03
        Delay up 10 secs, down 10 secs
    Latest operation return code: OK
    Latest RTT (millisecs) 7
    Tracked by:
        Track List 3
Track 2
    IP SLA 2 reachability Reachability is Up
        1 change, last change 00:02:35
        Delay up 10 secs, down 10 secs
    Latest operation return code: OK
    Latest RTT (millisecs) 8
    Tracked by:
        Track List 3
Track 3
    List boolean and Boolean AND is Up

      2 changes, last change 00:01:41
    object 1 Up
```

https://t.me/learningnets

```
    object 2 Up
```

Stop ICMP packets coming back from VLAN 8 and see how that effects tracked objects.

```
R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R6(config)#access-list 100 deny icmp any any
 R6(config)#access-list 100 permit ip any any
 R6(config)#interface GigabitEthernet1.146
R6(config-if)#ip access-group 100 in
R6#show track
R6#sh track
Track 1
    IP SLA 1 reachability Reachability is Down
    2 changes, last change 00:00:12
Delay up 10 secs, down 10 secs
Latest operation return code: Timeout
Tracked by:
    Track List 3
Track 2
    IP SLA 2 reachability Reachability is Up
    1 change, last change 00:07:49
Delay up 10 secs, down 10 secs
Latest operation return code: OK
Latest RTT (millisecs) 19
Tracked by:
    Track List 3
Track 3
    List boolean and Boolean AND is Down

    3 changes, last change 00:00:11
        object 1 Down
        object 2 Up
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

HSRP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure HSRP on R4 and R6 so that hosts on VLAN 146 can use the virtual IP address 155.1.146.254 as their default gateway.
- Ensure that R6 is the active physical gateway unless it loses connectivity to VLAN 146.
- Set the hello interval to 1 second and dead interval to 3 seconds for faster failed gateway detection.
- Authenticate HSRP message exchanges using an MD5 hash of the key **CISCO**.

Configuration

```
R6:
 !
 ! Using “preempt” is recommended to ensure deterministic
 ! primary gateway election
 !
 interface GigabitEthernet1.146
  standby 146 ip 155.1.146.254
  standby 146 timers 1 3
  standby 146 preempt
  standby 146 authentication md5 key-string CISCO
  standby 146 name VLAN146
  standby 146 priority 110
R4:
```

https://t.me/learningnets

```
!
! Using “preempt” is recommended to ensure deterministic
! primary gateway election
!
interface GigabitEthernet1.146
 standby 146 ip 155.1.146.254
 standby 146 timers 1 3
 standby 146 preempt
 standby 146 authentication md5 key-string CISCO
 standby 146 name VLAN146
```

Verification

Issue the `show standby` command to ensure that R6 is the active HSRP gateway, and to verify the backup gateway activation. Note the last octet of the virtual HSRP MAC address, which represents the group number (shown in hex).

```
R6#show standby

GigabitEthernet1.146 - Group 146
 State is Active
    2 state changes, last state change 00:01:09
Virtual IP address is 155.1.146.254 Active virtual MAC address is 0000.0c07.ac92 (MAC In Use)
Local virtual MAC address is 0000.0c07.ac92 (v1 default)
Hello time 1 sec, hold time 3 sec
Next hello sent in 0.864 secs
 Authentication MD5, key-string
Preemption enabled Active router is local
Standby router is 155.1.146.4, priority 100 (expires in 2.448 sec)
Priority 110 (configured 110)
    Group name is "VLAN146" (cfgd)
!
!R6(config)#interface GigabitEthernet1.146
R6(config-subif)#shutdown
!
!R4#show standby

GigabitEthernet1.146 - Group 146
State is Active
    2 state changes, last state change 00:00:03
Virtual IP address is 155.1.146.254 Active virtual MAC address is 0000.0c07.ac92 (MAC In Use)
Local virtual MAC address is 0000.0c07.ac92 (v1 default)
Hello time 1 sec, hold time 3 sec
Next hello sent in 0.832 secs
```

https://t.me/learningnets

```
Authentication MD5, key-string
Preemption enabled Active router is local

Standby router is unknown
Priority 100 (default 100)
    Group name is "VLAN146" (cfgd)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

VRRP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure VRRP on R4 and R6 so that hosts on VLAN 146 can use the virtual IP address 155.1.146.253 as their default gateway.
- Ensure that R4 is the active physical gateway unless it loses connectivity to VLAN 146.
- Set the hello interval to 3 seconds, and authenticate messages using the clear-text password **CISCO**.

Configuration

```
R6:
interface GigabitEthernet1.146
 vrrp 146 ip 155.1.146.253
 vrrp 146 timers advertise 3
 vrrp 146 authentication text CISCO
R4:

interface GigabitEthernet1.146
 vrrp 146 ip 155.1.146.253
 vrrp 146 timers advertise 3
 vrrp 146 authentication text CISCO
 vrrp 146 priority 110
```

https://t.me/learningnets

Verification

Verify the status of the virtual router and check the backup activation with the `show vrrp` command.

```
R6#show vrrp

GigabitEthernet1.146 - Group 146 State is Backup
Virtual IP address is 155.1.146.253
Virtual MAC address is 0000.5e00.0192
Advertisement interval is 3.000 sec
Preemption enabled
Priority is 100
Authentication text "CISCO" Master Router is 155.1.146.4, priority is 110

Master Advertisement interval is 3.000 sec
Master Down interval is 9.609 sec (expires in 9.342 sec)
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.146
R4(config-if)#shutdown
R6#show vrrp

GigabitEthernet1.146 - Group 146 State is Master
Virtual IP address is 155.1.146.253
Virtual MAC address is 0000.5e00.0192
Advertisement interval is 3.000 sec
Preemption enabled
Priority is 100
Authentication text "CISCO" Master Router is 155.1.146.6 (local), priority is 100

Master Advertisement interval is 3.000 sec
Master Down interval is 9.609 sec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

GLBP (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure GLBP for hosts on VLAN 146 so that first-hop redundancy and load balancing occur between R4 and R6.
- Ensure that R6 is elected as the Active Virtual Gateway (AVG), and both R4 and R6 are used as Active Virtual Forwarders (AVFs) for the virtual IP address 155.1.146.252.
- Use a 1-second hello interval and a 3-second hold time.
- Distribute the traffic load between R4 and R6 in 2:1 ratio.
- Authenticate GLBP UDP packet exchange using a secure method based on a password value of **CISCO**.

Configuration

```
R6:
interface GigabitEthernet1.146
 glbp 146 ip 155.1.146.252
 glbp 146 timers 1 3
 glbp 146 priority 110
 glbp 146 preempt
 glbp 146 weighting 10
 glbp 146 name VLAN146
glbp 146 authentication md5 key-string CISCO
glbp 146 load-balancing weighted
R4:
```

https://t.me/learningnets

```
interface GigabitEthernet1.146
 glbp 146 ip 155.1.146.252
 glbp 146 timers 1 3
 glbp 146 preempt
 glbp 146 weighting 20
 glbp 146 name VLAN146
glbp 146 authentication md5 key-string CISCO
glbp 146 load-balancing weighted
```

Verification

GLBP allows multiple active physical gateways to serve the same virtual gateway IP address. The AVG (active virtual gateway) responds to ARP requests sent to virtual gateway IP addresses and replies with different virtual MAC addresses that correspond to multiple active virtual forwarders (AVFs). Both the AVG and AVFs are redundant—that is, if a primary physical router representing the AVG or an AVF fails, another physical router will take its role.

To verify this, first check the status of the GLBP group and virtual forwarders state with the `show glbp` command.

```
R4#show glbp
 GigabitEthernet1.146 - Group 146
```

R4 is the STANDBY AVG, because R6 has a higher priority.

```
State is Standby

1 state change, last state change 00:00:31
Virtual IP address is 155.1.146.252
Hello time 1 sec, hold time 3 sec
Next hello sent in 0.864 secs
Redirect time 600 sec, forwarder time-out 14400 sec
Authentication MD5, key-string
Preemption enabled, min delay 0 sec
Active is 155.1.146.6, priority 110 (expires in 2.976 sec)
Standby is local
Priority 100 (default)
Weighting 20 (configured 20), thresholds: lower 1, upper 20
Load balancing: weighted
IP redundancy name is "VLAN146"
Group members:
0050.568d.4949 (155.1.146.4) local
```

https://t.me/learningnets

```
0050.568d.6e90 (155.1.146.6) authenticated
There are 2 forwarders (1 active)
```

There are two members in the GLBP group, and R6 (the other member) is authenticated.

```
    0007.0e9c.dac2 (155.1.146.4) local
    0012.019a.27c0 (155.1.146.6) authenticated
      There are 2 forwarders (1 active)
```

It is important to understand that each AVF is in fact a virtual gateway consisting of primary and backup physical gateways. From the output below, you can see that R4 is the primary physical gateway for Forwarder 1 and is in “Listen” (backup state) for Forwarder 2. However, for each AVF only the primary physical gateway would forwards packets; the other is listening to become active for that AVF if the primary fails. The load between AVFs is distributed based on the number of times the AVG will respond to ARP requests for the IP address of the virtual gateway with the virtual MAC of a particular AVF. The load is proportional to the weight assigned to an AVF if the load-balancing scheme is “weighted.”

```
Forwarder 1
State is Active
      1 state change, last state change 00:25:34
    MAC address is 0007.b400.9201 (default)
    Owner ID is 0007.0e9c.dac2
    Preemption enabled, min delay 30 sec Active is local, weighting 20
    Arp replies sent: 12 Forwarder 2
State is Listen
    MAC address is 0007.b400.9202 (learnt)
    Owner ID is 0012.019a.27c0
    Time to live: 14399.896 sec (maximum 14400 sec)
    Preemption enabled, min delay 30 sec Active is 155.1.146.6 (primary), weighting 10
 (expires in 2.892 sec)
R6#show glbp
GigabitEthernet1.146 - Group 146
State is Active
1 state change, last state change 00:08:39
Virtual IP address is 155.1.146.252
Hello time 1 sec, hold time 3 sec
Next hello sent in 0.128 secs
Redirect time 600 sec, forwarder time-out 14400 sec
Authentication MD5, key-string
Preemption enabled, min delay 0 sec
Active is local
```

https://t.me/learningnets

```
Standby is 155.1.146.4, priority 100 (expires in 2.624 sec)
Priority 110 (configured)
Weighting 10 (configured 10), thresholds: lower 1, upper 10
Load balancing: weighted
IP redundancy name is "VLAN146"
Group members:
0050.568d.4949 (155.1.146.4) authenticated
0050.568d.6e90 (155.1.146.6) local
There are 2 forwarders (1 active)Forwarder 1
    State is Active
    1 state change, last state change 00:08:35
    MAC address is 0007.b400.9201 (default)
    Owner ID is 0050.568d.6e90
    Redirection enabled
    Preemption enabled, min delay 30 sec
    Active is local, weighting 10Forwarder 2

    State is Listen
    MAC address is 0007.b400.9202 (learnt)
    Owner ID is 0050.568d.4949
    Redirection enabled, 599.648 sec remaining (maximum 600 sec)
    Time to live: 14399.648 sec (maximum 14400 sec)
    Preemption enabled, min delay 30 sec
    Active is 155.1.146.4 (primary), weighting 20 (expires in 2.976 sec)
```

The below output illustrates how the AVG responds with virtual MACs based on a client's (R1) ARP requests to the GLBP virtual IP address. Note that virtual MAC 0007.b400.9201 is AVF 1 (R4), and 0007.b400.9202 is AVF 2 (R6).

```
R1#ping 155.1.146.252

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.252, timeout is 2 seconds:
.....
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/4 ms
!
!R1#show ip arp 155.1.146.252
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  155.1.146.252          52 0007.b400.9201
  ARPA   GigabitEthernet1.146
!
!
R1#clear arp-cache
R1(config)#interface GigabitEthernet 1.146
R1(config-subif)#shut
R1(config-subif)#do clear arp-cache
R1(config-subif)#no shutR1(config-subif)#end
```

https://t.me/learningnets

```
!
!R1#ping 155.1.146.252

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.252, timeout is 2 seconds:
.....
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/4 ms
!
!R1#show ip arp 155.1.146.252
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  155.1.146.252           0  0007.b400.9202
  ARPA   GigabitEthernet1.146
```

If we repeat this process for a couple of time, the AVF 1 (R4) virtual MAC appears more often than the AVF 2 MAC address, following the configured 2:1 distribution ratio.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Router Redundancy and Object Tracking

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **Object_Tracking Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- R6 should be the active HSRP gateway as long as it can ping VLAN 7 and reach it through telnet, but it should give up the active role when it loses either telnet or ping reachability to VLAN 7.
- When VLAN7 goes down, R4 should take the active role.

Configuration

```
R6:

interface GigabitEthernet1.146
 standby 146 track 3 decrement 20
```

Verification

Originally, HSRP was only able to track an interface state (such as uplink interface) and decrease its priority when the interface failed. As illustrated in the *IP Routing* section of this workbook, the interface state of a link is not always a good indication of end-to-end connectivity out the link. Combining enhanced object tracking with first hop redundancy extends functionality to any application that can be tracked through IP SLA in addition to basic routing checks. Remember to enable HSRP preemption on the “secondary” router to allow the secondary overtake role of the primary router.

https://t.me/learningnets

The below output verifies how HSRP tracks the object on R6.

```
R6#show track 3
Track 3
    List boolean and Boolean AND is Up
        2 changes, last change 00:17:06
        object 1 Up
        object 2 Up
!
!R6#show standby
GigabitEthernet1.146 - Group 146 State is Active
        2 state changes, last state change 00:27:47
    Virtual IP address is 155.1.146.254
    Active virtual MAC address is 0000.0c07.ac92 (MAC In Use)
     Local virtual MAC address is 0000.0c07.ac92 (v1 default)
    Hello time 1 sec, hold time 3 sec
     Next hello sent in 0.672 secs
    Authentication MD5, key-string
    Preemption enabled Active router is local

    Standby router is 155.1.146.4, priority 100 (expires in 2.960 sec)
    Priority 110 (configured 110)
        Group name is "VLAN146" (cfgd)
```

Shutting the VLAN 7 interface down causes the enhanced objects to go down. This causes R6 to decrement its HSRP priority and stop forwarding packets for that HSRP group.

```
R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#interface GigabitEthernet1.7
R6(config-if)#shutdown

1 ip sla 1 reachability Up -> Down
2 ip sla 2 reachability Up -> Down 3 list boolean and Up -> Down
%HSRP-5-STATECHANGE: GigabitEthernet1.146 Grp 146 state Active -> Speak
%HSRP-5-STATECHANGE: GigabitEthernet1.146 Grp 146 state Speak -> Standby
!
!R6#show standby
GigabitEthernet1.146 - Group 146 State is Standby
    4 state changes, last state change 00:00:39
Virtual IP address is 155.1.146.254
Active virtual MAC address is 0000.0c07.ac92 (MAC Not In Use)
Local virtual MAC address is 0000.0c07.ac92 (v1 default)
Hello time 1 sec, hold time 3 sec
    Next hello sent in 0.368 secs
```

https://t.me/learningnets

```
Authentication MD5, key-string
Preemption enabled
Active router is 155.1.146.4, priority 100 (expires in 3.104 sec)Standby router is local

Priority 90 (configured 110)
    Track object 3 state Down decrement 20
Group name is "VLAN146" (cfgd)
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IRDP (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- R5 and R8 should advertise themselves as default gateways for hosts on VLAN58 using ICMP messages.
- R5 should be the preferred gateway on the segment.

Configuration

```
R5:
interface GigabitEthernet1.58
 ip irdp
 ip irdp address 155.1.58.5 1000
 ip irdp maxadvertinterval 20
 ip irdp minadvertinterval 10
R8:

interface GigabitEthernet1.58
 ip irdp
 ip irdp address 155.1.58.8 500
 ip irdp maxadvertinterval 20
 ip irdp minadvertinterval 10
```

Verification

https://t.me/learningnets

IRDP uses ICMP messages to advertise candidate default gateway on a segment. Multiple routers can advertise themselves at the same time, with different priorities.

To verify this configuration, disable IP routing on R7, and enable default gateway discovery via IRDP. Additionally, create a VLAN 58 interface on R7 and assign it an IP address of 155.1.58.7/24.

```
R7:
no ip routing
ip gdp irdp
!
interface GigabitEthernet1.58 ip address 155.1.58.7 255.255.255.0
 !
 !R7#show ip route
 Gateway         Using  Interval  Priority   Interface
 155.1.58.5      IRDP       22       1000     GigabitEthernet1.58

 Default gateway is 155.1.58.5

 Host               Gateway           Last Use    Total Uses  Interface
 ICMP redirect cache is empty
 !
 !R5#show ip irdp GigabitEthernet 1.58
 GigabitEthernet1.58 has router discovery enabled
 Advertisements will occur between every 10 and 20 seconds.
 Advertisements are sent with broadcasts.
 Advertisements are valid for 60 seconds.
 Default preference will be 0. Proxy for 155.1.58.5 with preference 1000.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Router ICMP Settings

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R1's VLAN146 interface to stop sending ICMP messages about discarded packets, ICMP messages to select a better next-hop, and ICMP messages reporting subnets mask.
- Rate-limit ICMP unreachables to 2 per second globally.

Configuration

```
R1:

interface GigabitEthernet1.146
 no ip redirects
 no ip unreachables
 no ip mask-reply
!
ip icmp rate-limit unreachable 500
```

Verification

https://t.me/learningnets

You can verify the rate limit of unreachables by using the `traceroute` command on R3. The destination router sends ICMP unreachable packets in response to UDP probes. Note that you cannot do traceroute from R6 or R4 because R1 has ICMP unreachables disabled on VLAN146 interface.

```
R3#traceroute 155.1.13.1

Type escape sequence to abort.
Tracing the route to 155.1.13.1
    VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.13.1 12 msec *  12 msec
```

Change the ICMP unreachables rate to 1 each every 10 ms and repeat the traceroute command from R3.

```
R1(config)#ip icmp rate-limit unreachable 10
R3#traceroute 155.1.13.1

Type escape sequence to abort.
Tracing the route to 155.1.13.1
    VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.13.1 16 msec 16 msec 16 msec
```

Verify the ICMP messages settings on R1's VLAN146 interface.

```
R1#show ip interface GigabitEthernet1.146
GigabitEthernet1.146 is up, line protocol is up
 Internet address is 155.1.146.1/24
 Broadcast address is 255.255.255.255
 Address determined by setup command
 MTU is 1500 bytes
 Helper address is not set
 Directed broadcast forwarding is disabled
 Multicast reserved groups joined: 224.0.0.10
 Outgoing Common access list is not set
 Outgoing access list is not set
 Inbound Common access list is not set
 Inbound  access list is not set
 Proxy ARP is enabled
 Local Proxy ARP is disabled
 Security level is default
 Split horizon is enabled
```

https://t.me/learningnets

```
ICMP redirects are never sent
 ICMP unreachables are never sent
ICMP mask replies are never sent


 <snip>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Basic NAT

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 to translate IP source addresses for packets from the subnet 155.1.8.0/24 going to VLAN 146 transiting VLAN 45. Use a dynamic pool from the point-to-point subnet in VLAN 45.
- Do not use any route-maps to accomplish this.

Configuration

```
R5:

ip nat pool VLAN45 155.1.45.6 155.1.45.254 prefix-length 24
!
ip access-list extended NAT_TRAFFIC
 permit ip 155.1.8.0 0.0.0.255 any
!
ip nat inside source list NAT_TRAFFIC pool VLAN45
!
interface GigabitEthernet1.58
 ip nat inside
!
interface GigabitEthernet1.45
 ip nat outside
```

https://t.me/learningnets

Verification

Source ICMP packets off the VLAN 8 interface of R8 to verify that NAT translations are created on R5.

```
R8#ping 155.1.146.4 source GigabitEthernet 1.8

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.4, timeout is 2 seconds:
Packet sent with a source address of 155.1.8.8
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/9/25 ms
!
!R5#show ip nat translations
Pro  Inside global         Inside local          Outside local         Outside global
---  155.1.45.6            155.1.8.8             ---                   ---
icmp 155.1.45.6:8          155.1.8.8:8           155.1.146.4:8         155.1.146.4:8
Total number of translations: 2
R4#debug ip packet
```

Note that with access-list-based NAT, the process creates one “parent” IP to IP address translation entry, which is used to multiplex all sessions for this particular inside host. Those IP to IP entries are called non-extendable, because they only have inside local and inside global IP address information. The one-to-one entries also permit outside hosts to connect to the inside using the temporary mapped IP address. Additionally, with non-extendable entries, an inside local address may have only one inside global translation. This effectively prevents multi-homed NAT when using access-lists only for NAT configuration.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

NAT Overload

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R8 to translate packets sourced from the Loopback 0 of R1, R2, and R3 using the 155.1.108.8 address.

Configuration

```
R8:

interface GigabitEthernet1.108
 ip nat outside
!
interface GigabitEthernet1.58
 ip nat inside
!
ip access-list extended LOOPBACKS
 permit ip 150.1.0.0 0.0.3.255 any
!
ip nat inside source list LOOPBACKS interface GigabitEthernet1.108 overload
```

Verification

NAT overloading is also known as PAT (Port Address Translation) because it uses the same global IP address for all local IP addresses, changing just the port

https://t.me/learningnets

numbers.

To verify the configuration, source traffic off R1's Loopback 0 interface to the networks behind R8.

```
R1# ping 155.1.10.10 source l0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/9/13 ms
!
!R2#ping 155.1.10.10 source l0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/42/139 ms
!
!R3#ping 155.1.10.10 source l0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/45/89 ms
```

Note that the "overload" NAT feature does not create any IP-to-IP (non-extendable) entries, but rather it multiplexes all translations in one global IP address, using inside/outside IP addresses and port numbers (IDs) as keys.

```
R8#sh ip nat translations

Pro  Inside global         Inside local          Outside local         Outside global
icmp 155.1.108.8:3         150.1.3.3:2           155.1.10.10:2         155.1.10.10:3
icmp 155.1.108.8:1         150.1.1.1:23          155.1.10.10:23        155.1.10.10:1
icmp 155.1.108.8:2         150.1.2.2:2           155.1.10.10:2         155.1.10.10:2

Total number of translations: 3
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

NAT with Route Maps

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R2 with route-map-based NAT to support multiple outside interfaces as follows.
  - Configure additional sub-interface for VLAN 2 on R2 and advertise the prefix into EIGRP, use 155.1.2.0/24 network.
  - Traffic from R2's Loopback0 network going out the point-to-point link to R3 should be translated to 155.1.23.200.
  - Traffic from R2's GigabitEthernet1.2 interface going out the point-to-point link to R3 should be translated to R2's interface IP address.

Configuration

R2:

```
interface GigabitEthernet1.2
 encapsulation dot1q 2
 ip address 155.1.2.2 255.255.255.0
!
interface GigabitEthernet1.23
 ip nat outside
!
ip access-list standard FROM_G2
 permit 155.1.2.2
!
```

https://t.me/learningnets

```
ip access-list standard FROM_LOOPBACK
 permit 150.1.2.2
!
route-map NAT_OUT_PPP_FROM_GIG2 deny 10
 match ip address FROM_LOOPBACK
 match interface GigabitEthernet1.23
!
route-map NAT_OUT_PPP_FROM_GIG2 permit 20
 match ip address FROM_G2
 match interface GigabitEthernet1.23
!
route-map NAT_OUT_PPP_FROM_LOOPBACK permit 10
 match ip address FROM_LOOPBACK
 match interface GigabitEthernet1.23
!
ip nat pool PPP_LOOPBACK_POOL 155.1.23.200 155.1.23.200 prefix-length 24
!
ip nat inside source route-map NAT_OUT_PPP_FROM_GIG2 interface GigabitEthernet1.23 overload
!
ip nat inside source route-map NAT_OUT_PPP_FROM_LOOPBACK pool PPP_LOOPBACK_POOL overload
```

Verification

To verify this configuration, telnet from R2 to destination out the point-to-point link while changing the source addresses.

https://t.me/learningnets

```
R2#telnet 155.1.23.3 /source-interface GigabitEthernet1.2

Trying 155.1.23.3 ... Open
R3#show users

Line       User       Host(s)              Idle       Location
0 con 0                idle                 00:02:21
*  2 vty 0                idle                 00:00:00 155.1.23.2



Interface    User               Mode         Idle     Peer Address
!
!R2#telnet 155.1.23.3 /source-interface Loopback 0

Trying 155.1.23.3 ... Open
R3#show users

Line       User       Host(s)              Idle       Location
0 con 0                idle                 00:01:31
*  2 vty 0                idle                 00:00:00 155.1.23.200



 Interface    User               Mode         Idle     Peer Address
```

When using route-maps for NAT configuration (not just simple access-lists), the NAT process creates “extendable” NAT entries (fully extended, with local/global IP addresses and port numbers). This allows for NAT multi-homing, because a particular inside local source IP address is not bound to just one inside global IP address. Also, reverse connections to inside hosts using temporary global IP address mappings are not possible with default route-map configuration.

```
R2#show ip nat translation
Pro  Inside global         Inside local          Outside local         Outside global
tcp  155.1.23.200:1024     150.1.2.2:44512       155.1.23.3:23         155.1.23.3:23
tcp  155.1.23.2:4096       155.1.2.2:54458       155.1.23.3:23         155.1.23.3:23

Total number of translations: 2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Static NAT

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 so that R4 can reach R8's VLAN 58 IP address using the address 155.1.45.8.
- Do not configure a static route to accomplish this task.

Configuration

```
R5:

interface GigabitEthernet1.45
 ip nat outside
!
interface GigabitEthernet1.58
 ip nat inside
!
ip nat inside source static 155.1.58.8 155.1.45.8
```

Verification

https://t.me/learningnets

When a packet arrives at the NAT “outside” interface, it is translated via the static NAT table before routing lookup ; this is why the static route is needed. When R4 sends a packet to the address 155.1.45.8, R5 first translate the address using NAT table. Without the static route, R5 can't route it to the exiting interface.

```
R4#telnet 155.1.45.8
Trying 155.1.45.8 ... Open
R8#show users
Line       User       Host(s)              Idle       Location
0 con 0                idle                 00:00:13
*  2 vty 0                idle                 00:00:00 155.1.45.4

Interface    User               Mode         Idle     Peer Address
!
!R5#show ip nat translations

Pro  Inside global         Inside local          Outside local         Outside global
---  155.1.45.8            155.1.58.8            ---                   ---
tcp  155.1.45.8:23         155.1.58.8:23         155.1.45.4:59257      155.1.45.4:59257

Total number of translations: 2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Static PAT

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 so that when R4 telnets to 155.1.45.44 at port 8023, they get connected to R8's VLAN 8 interface at port 23.
- Configure R5 so that when R4 telnets to 155.1.45.44 at port 10023, they get connected to R10's VLAN 10 interface at port 23.

Configuration

```
R5:

interface GigabitEthernet1.45
 ip nat outside
!
interface GigabitEthernet1.58
 ip nat inside
!
ip nat inside source static tcp 155.1.8.8 23 155.1.45.44 8023
ip nat inside source static tcp 155.1.10.10 23 155.1.45.44 10023
```

Verification

https://t.me/learningnets

Static PAT allows mapping ports on a global IP address to the ports on a local IP address. To verify this configuration, telnet from R4 to R5 at ports 8023 and 10023.

```
R4#telnet 155.1.45.44 8023



Trying 155.1.45.44, 8023 ... Open R8#

R4#telnet 155.1.45.44 10023

Trying 155.1.45.44, 10023 ... Open R10#
```

Note that the host responding to ICMP echo packets is R5, because it created the IP alias entry for the translated IP address.

```
R4#ping 155.1.45.44
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.45.44, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/9/16 ms
!
!R5#sh ip aliases

Address Type              IP Address      Port
Interface                 150.1.5.5
Interface                 155.1.0.5
Interface                 155.1.5.5
Interface                 155.1.45.5 Dynamic                    155.1.45.44

Interface                 155.1.58.5
Interface                 169.254.100.5
Interface                 192.168.1.2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Static NAT and IP Aliasing

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 so that it does not add a local alias for the static mapping of the IP address 155.1.45.44.

Configuration

```
R5:

ip nat inside source static tcp 155.1.8.8 23 155.1.45.44 8023 extendable no-alias
ip nat inside source static tcp 155.1.10.10 23 155.1.45.44 10023 extendable no-alias
```

Verification

The no-alias feature will prevent a router running NAT from installing a local IP alias entry. However, the router will still respond to ARP requests for the global translated IP address; it just won't terminate any connection on itself. To verify this, telnet to 155.1.45.44 at port 8023 from R4, and ping this address.

https://t.me/learningnets

```
R4#telnet 155.1.45.44 10023



Trying 155.1.45.44, 10023 ... OpenR10#
```

Now R5 does not respond to the ICMP echo packets because it has no local alias generated for the IP address 155.1.45.44.

```
R4#ping 155.1.45.44
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.45.44, timeout is 2 seconds:.....
Success rate is 0 percent (0/5)
R5#show ip aliases

Address Type             IP Address      Port
Interface                150.1.5.5
Interface                155.1.0.5
Interface                155.1.5.5
Interface                155.1.45.5
Interface                155.1.58.5
Interface                169.254.100.5
Interface                192.168.1.2
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Static Policy NAT

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure a static translation on R3 so that when traffic is received from R7's VLAN 7 IP address and it is going out the point-to-point link to R1, it is translated to the address 155.1.13.7.
- Configure a static translation on R3 so that when traffic is received from R7's VLAN 7 IP address and it is going out the point-to-point link to R2, it is translated to the address 155.1.23.7.

Configuration

R3:

```
route-map LINK_TO_R1
 match interface GigabitEthernet1.13
!
route-map LINK_TO_R2
 match interface GigabitEthernet1.23
!
ip nat inside source static 155.1.7.7 155.1.13.7 route-map LINK_TO_R1
ip nat inside source static 155.1.7.7 155.1.23.7 route-map LINK_TO_R2
!
interface GigabitEthernet1.37
 ip nat inside
!
interface GigabitEthernet1.13
```

https://t.me/learningnets

```
 ip nat outside
!
interface GigabitEthernet1.23
 ip nat outside
```

Verification

Static policy NAT (static NAT with route-maps) allows the use of mappings to different inside global IP addresses for the same inside local IP. The route-maps are used to classify traffic to be used with respective static NAT translation. Commonly, this type of mapping is used on multi-homed routers where the same inside local server address maps to different inside global IP addresses out to multiple ISPs.

To verify this configuration, telnet from R7 with a source address of VLAN 7 out to R1 and R2, and issue the `show users` command.

```
R7#telnet 155.1.13.1 /source-interface gigabitEthernet 1.7
Trying 155.1.13.1 ... Open
R1#
!
!R1#show users

Line       User       Host(s)              Idle       Location
0 con 0                idle                 05:40:35
*  2 vty 0                idle                 00:00:00 155.1.13.7

Interface    User               Mode         Idle     Peer Address

R1#
!
!R7#telnet 155.1.23.2 /source-interface gigabitEthernet 1.7
Trying 155.1.23.2 ... Open
!
!R2#show users

Line       User       Host(s)              Idle       Location
0 con 0                idle                 00:00:35
*  2 vty 0                idle                 00:00:00 155.1.23.7


Interface    User               Mode         Idle     Peer Address
```

Remove the static statements that call the route-maps and replace them with normal

https://t.me/learningnets

static extendable translations.

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#no ip nat inside source static 155.1.7.7 155.1.13.7 route-map LINK_TO_R1
R3(config)#no ip nat inside source static 155.1.7.7 155.1.23.7 route-map LINK_TO_R2
R3(config)#ip nat inside source static 155.1.7.7 155.1.13.7 extendable
 R3(config)#ip nat inside source static 155.1.7.7 155.1.23.7 extendable

R3(config)#end
R3#
```

Re-verify the operation of the translations by telneting from R7 to R1 and R2 again, while sourcing the packets from VLAN 7.

```
R7#telnet 155.1.13.1 /source-interface gigabitEthernet 1.7
Trying 155.1.13.1 ... Open
!
!R1#show users
Line       User       Host(s)              Idle       Location
0 con 0                idle                 06:24:02
*  2 vty 0                idle                 00:00:00 155.1.13.7

Interface    User               Mode         Idle     Peer Address
!
!R7#telnet 155.1.23.2 /source-interface gigabitEthernet 1.7
Trying 155.1.23.2 ... Open
!
!R2#show users
Line       User       Host(s)              Idle       Location
0 con 0                idle                 00:34:37
*  2 vty 0                idle                 00:00:00 155.1.13.7


Interface    User               Mode         Idle     Peer Address
```

Note that in the above output, both sessions are translated to the address 155.1.13.7, whereas in the previous output they were translated based on the exit interface when the route-maps were used for classification.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

NAT with Overlapping Subnets (pending update)

Task

- Create a new Loopback1 interface on R1 and R2 using the IP addresses 10.0.0.1/24 and 10.0.0.2/24, respectively.
- Advertise the 10.0.0.0/24 network into RIP on R2.
- Configure NAT on R1 so that when R2 telnets to the address 11.0.0.Z from its Loopback1 address, it is redirected to the address 10.0.0.Z attached to R1, where “Z” is any number.
- The host 10.0.0.Z on R1 should see the packets from R2 as if they were sourced from the network 22.0.0.0/24.

Configuration

```
R1:
interface Loopback 1
 ip address 10.0.0.1 255.255.255.0
 ip nat inside
!
ip nat pool R2_MASQUARADE 22.0.0.1 22.0.0.254 prefix-length 24
!
ip access-list extended R2_LOOPBACK1
 permit ip 10.0.0.0 0.0.0.255 any
!
ip nat outside source list R2_LOOPBACK1 pool R2_MASQUARADE
ip nat inside source static network 10.0.0.0 11.0.0.0 /24
!
ip route 11.0.0.0 255.255.255.0 Null 0
ip route 22.0.0.0 255.255.255.0 Serial 0/1
!
router rip
```

https://t.me/learningnets

```
 redistribute static

R2:
interface Loopback 1
 ip address 10.0.0.2 255.255.255.0
!
router rip
 network 10.0.0.0
```

Verification

This particular lab shows how to deal with overlapping subnets by configuring just one router. A better approach in a real-world design would be to create static network-to-network mappings on both R1 and R2 to mask their overlapping 10.0.0.0/24 segments, and never advertise the overlapping subnet in the first place.

In this case, R2 still advertises its subnet 10.0.0.0/24 to the rest of the network, but R1 hides its Loopback1 subnet using a static NAT rule. Based on this, R1 must be configured so that it never sees packets sourced from the 10.0.0.0/24 subnet on R2.

To verify the scenario, enable NAT debugging on R1 and source ping packets off the Loopback1 interface of R2. Note that the `ip nat inside` command on the Loopback1 interface of R1 is not necessary in this design, but it is needed in a real design where there are actual hosts behind the masqueraded segment.

```
Rack1R1#debug ip nat detailed

IP NAT detailed debugging is on
Rack1R1#
Rack1R2#ping 11.0.0.1 source loopback 1 repeat 2


Type escape sequence to abort.
Sending 2, 100-byte ICMP Echos to 11.0.0.1, timeout is 2 seconds:
Packet sent with a source address of 10.0.0.2
!!
Success rate is 100 percent (2/2), round-trip min/avg/max = 60/62/64 ms
```

Per the NAT outside rule, 10.0.0.2 is translated to 22.0.0.2 using the dynamic pool configured. Note that 11.0.0.1 translates to 10.0.0.1, thanks to the static subnet mapping. Eventually, R1 sees the original packet from 10.0.0.2 to 11.0.0.1 as a packet from 22.0.0.2 to 10.0.0.1.

https://t.me/learningnets

```
Rack1R1#
NAT*: o: icmp (10.0.0.2, 20) -> (11.0.0.1, 20) [127]
NAT*: o: icmp (10.0.0.2, 20) -> (11.0.0.1, 20) [127]
NAT*: s=10.0.0.2->22.0.0.2, d=11.0.0.1 [127]
NAT*: s=22.0.0.2, d=11.0.0.1->10.0.0.1 [127]
```

The response packet from 10.0.0.1 to 22.0.0.2 is translated using two rules. First, 10.0.0.1 translates back to 11.0.0.1, and second, 22.0.0.2 translates to 10.0.0.2. Note that before the translation occurs, R1 will try to look up a route for 22.0.0.2; this is why you need a static route on R1. The rest of the network does not need this static route because they know how to get to the 10.0.0.0/24 subnet on R2 because of the dynamic RIP route.

```
NAT: i: icmp (10.0.0.1, 20) -> (22.0.0.2, 20) [127]
NAT: s=10.0.0.1->11.0.0.1, d=22.0.0.2 [127]
NAT: s=11.0.0.1, d=22.0.0.2->10.0.0.2 [127]
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

TCP Load Distribution with NAT

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Disable the VLAN 45 interface on R5.
- Configure rotary NAT R5 so that when R8 telnets to the address 155.1.58.55, it is redirected to R1, R2, and R3 in an even distribution.

Configuration

R5:

```
interface GigabitEthernet1.45
shutdown
!
interface GigabitEthernet1.58
 ip nat outside
!
interface Tunnel0
 ip nat inside
!
ip nat pool ROTARY prefix-length 24 type rotary
 address 155.1.0.1 155.1.0.1
 address 155.1.0.2 155.1.0.2
 address 155.1.0.3 155.1.0.3
!
ip access-list extended LOAD_BALANCE
 permit tcp any host 155.1.58.55 eq telnet
```

https://t.me/learningnets

```
!
ip nat inside destination list LOAD_BALANCE pool ROTARY
!
ip alias 155.1.58.55 23
```

Verification

NAT supports simple load-distribution for TCP-based applications using the concept of rotary NAT pools. In this particular case, the inside destination translation checks traffic as it comes in the outside interface. If the traffic is matched byaccess-list LOAD_BALANCE, which equates to telnet traffic going to 155.1.58.55, the destination address is changed to 155.1.0.1, 155.1.0.2, and 155.1.0.3 in a round-robin (rotary) fashion per flow.

A more scalable approach to this same design is to use the IOS Server Load Balancing (SLB) feature and NAT.

To verify the configuration, telnet from R8 to R5 three times, and see how the destination changes each time.

```
R8#telnet 155.1.58.55
Trying 155.1.58.55 ... OpenR3#
!
!R8#telnet 155.1.58.55
Trying 155.1.58.55 ... OpenR2#
!
!R8#telnet 155.1.58.55
Trying 155.1.58.55 ... OpenR1#
!
!
[Connection to 155.1.58.55 closed by foreign host]R8#
!
!R5#sh ip nat translations
Pro  Inside global         Inside local          Outside local         Outside global
tcp  155.1.58.55:23        155.1.0.1:23          155.1.58.8:38699      155.1.58.8:38699
tcp  155.1.58.55:23        155.1.0.2:23          155.1.58.8:53196      155.1.58.8:53196
tcp  155.1.58.55:23        155.1.0.3:23          155.1.58.8:11279      155.1.58.8:11279
Total number of translations: 3
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Stateful NAT with HSRP (pending update)

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **EIGRP Routing Initial** , which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure Loopback57 on R5 and R7 using 150.1.57.57/32 prefix. Shut the VLAN 37 link down on R7.
- Configure HSRP on R4 and R6 with the group name “VLAN146” so that hosts on VLAN 146 can use the virtual IP address 155.1.146.254 as their default gateway.
- Configure R6 to be the active physical gateway unless the reachability to VLAN 7 goes down.
- Set the hello interval to 1 second and dead interval to 3 seconds for faster failed gateway detection.
- Authenticate HSRP message exchanges using an MD5 hash of the key **CISCO**.
- Create a single NAT pool on R4 and R6 with the IP addresses in the range 155.1.254.1–155.1.254.254.
- Configure dynamic NAT on R4 and R6 so that traffic from the R1 is port translated to the previously defined NAT pool.
- If the link of VLAN 67 fails, R4 should take over the "Active" role of HSRP and maintain the NAT table.

Configuration

```
R5:
interface loopback57
 ip address 150.1.57.57 255.255.255.255
!R7:
```

https://t.me/learningnets

```
interface loopback57
 ip address 150.1.57.57 255.255.255.255
```

R4:

```
interface GigabitEthernet1.146
 standby 146 ip 155.1.146.254
 standby 146 timers 1 3
 standby 146 preempt
 standby 146 authentication md5 key-string CISCO
 standby 146 name VLAN146
 ip nat inside
!
interface GigabitEthernet1.45
 ip nat outside
!
ip nat pool SHARED_POOL 155.1.254.1 155.1.254.254 prefix-length 24
!
ip route 155.1.254.0 255.255.255.0 Null 0
!
ip access-list standard NAT_LIST
 permit 155.1.146.0 0.0.0.255
!
ip nat inside source list NAT_LIST pool SHARED_POOL
```

R6:

```
track 10 ip route 150.1.57.57 255.255.255.255 reachability
!
interface GigabitEthernet1.146
 standby 146 ip 155.1.146.254
 standby 146 timers 1 3
 standby 146 preempt
 standby 146 authentication md5 key-string CISCO
 standby 146 name VLAN146
 standby 146 priority 110
 standby 146 track 10 decrement 20
 ip nat inside
!
interface GigabitEthernet1.67
 ip nat outside
!
ip nat pool SHARED_POOL 155.1.254.1 155.1.254.254 prefix-length 24
!
ip route 155.1.254.0 255.255.255.0 Null 0
!
ip access-list standard NAT_LIST
 permit 155.1.146.0 0.0.0.255
```

https://t.me/learningnets

```
!
ip nat inside source list NAT_LIST pool SHARED_POOL
```

Verification

The idea of stateful NAT is to allow two redundant exit points using the same NAT policy to exchange their NAT translation tables. This allows for stateful failover when the active forwarder fails.

To verify this, check the Stateful NAT status first.

```
Rack1R6#sh ip snat distributed verbose



Stateful NAT Connected Peers
 SNAT: Mode IP-REDUNDANCY ::ACTIVE
     : State READY
     : Local Address 155.1.146.6
     : Local NAT id 1
     : Peer Address 155.1.146.4
     : Peer NAT id 2
     : Mapping List 1
     : InMsgs 5, OutMsgs 0, tcb 0x84835C00, listener 0x0

User Access Verification
Rack1R4#sh ip snat distributed verbose



Stateful NAT Connected Peers
 SNAT: Mode IP-REDUNDANCY ::STANDBY

     : State READY
     : Local Address 155.1.146.4
     : Local NAT id 2
     : Peer Address 155.1.146.6
     : Peer NAT id 1
     : Mapping List 1
     : InMsgs 39, OutMsgs 0, tcb 0x6408D040, listener 0x0
```

Verify that SNAT peers change roles as R6 loses its active HSRP status.

```
Rack1R4#show standby


Ethernet0/1 - Group 146
```

https://t.me/learningnets

```
  State is Active
    8 state changes, last state change 00:01:29
  Virtual IP address is 155.1.146.254
  Active virtual MAC address is 0000.0c07.ac92
    Local virtual MAC address is 0000.0c07.ac92 (v1 default)
  Hello time 1 sec, hold time 3 sec
    Next hello sent in 0.576 secs
  Authentication MD5, key-string "CISCO"
  Preemption enabled
 Active router is local
  Standby router is 155.1.146.6, priority 90 (expires in 2.716 sec)
  Priority 100 (default 100)
  IP redundancy name is "VLAN146" (cfgd)
Rack1R4#show ip snat distributed verbose



Stateful NAT Connected Peers
 SNAT: Mode IP-REDUNDANCY ::ACTIVE


    : State READY
    : Local Address 155.1.146.4
    : Local NAT id 2
    : Peer Address 155.1.146.6
    : Peer NAT id 1
    : Mapping List 1
    : InMsgs 6, OutMsgs 0, tcb 0x6408D040, listener 0x0
```

Open a telnet session to a backbone network (BB1) and make sure that the translation appears in the NAT table of R4.

```
Rack1R3#telnet 112.0.0.1
Trying 112.0.0.1 ... Open
Rack1R4#show ip nat translations | inc 155.1.0.3


tcp 155.1.254.4:65148  155.1.0.3:65148    112.0.0.1:23       112.0.0.1:23
```

Ensure that R4 installed this NAT entry in R6’s translation table using SNAT.

https://t.me/learningnets

```
Rack1R6#show ip snat peer 155.1.146.4


Show NAT Entries created by peer: 155.1.146.4

Pro Inside global      Inside local       Outside local      Outside global
tcp 155.1.254.4:65148  155.1.0.3:65148    112.0.0.1:23       112.0.0.1:23
```

Make R6 the active router now by enabling its backbone connection and shutting down the backbone connection of R4. Ensure that R3 never loses its telnet connection, but just experiences a temporary delay.

```
Rack1R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.Rack1R6(config)#interface serial 0/0/0
Rack1R6(config-if)#no shutdown
Rack1R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.Rack1R4(config)#interfaceFastethernet 0/0
Rack1R4(config-if)#shutdown



Rack1AS>3
[Resuming connection 3 to R3 ... ]

RS.1.1.BB1>
RS.1.1.BB1>
```

Verify the NAT translation table of R6 and ensure that it is now identical to the table used by R4.

```
Rack1R6#show ip nat translations

Pro Inside global      Inside local       Outside local      Outside global
tcp 155.1.254.4:65148  155.1.0.3:65148    112.0.0.1:23       112.0.0.1:23
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Stateful NAT with Primary/Backup (Pending update)

Task

- Remove the HSRP configuration on R4 and R6.
- Modify the SNAT configuration on these devices so that it does not rely on HSRP status for tracking of the active SNAT router.
- R6 should be the primary and R4 should be the backup router.

Configuration

```
R6:
interface FastEthernet0/0.146
 no standby 146
!
ip nat Stateful id 1
 no redundancy VLAN146
     primary 155.1.146.6
         peer 155.1.146.4
         mapping-id 1

R4:
interface FastEthernet0/1
 no standby 146
!
ip nat Stateful id 2
 no redundancy VLAN146
     backup 155.1.146.4
         peer 155.1.146.6
         mapping-id 1
```

https://t.me/learningnets

Verification

Stateful NAT with Primary/Backup replaces the need for HSRP tracking by using a built-in SNAT keepalive. To verify the operation of this without HSRP, source a telnet connection from R3 to BB1, and ensure that R6 installs the translation entry in the NAT table of R4.

https://t.me/learningnets

```
Rack1R6#show ip snat distributed verbose


Stateful NAT Connected Peers
 SNAT: Mode PRIMARY
     : State READY

     : Local Address 155.1.146.6
     : Local NAT id 1
     : Peer Address 155.1.146.4
     : Peer NAT id 2
     : Mapping List 1
     : InMsgs 5, OutMsgs 0, tcb 0x857FE654, listener 0x0
Rack1R4#show ip snat distributed verbose


Stateful NAT Connected Peers
 SNAT: Mode BACKUP
     : State READY

     : Local Address 155.1.146.4
     : Local NAT id 2
     : Peer Address 155.1.146.6
     : Peer NAT id 1
     : Mapping List 1
     : InMsgs 15, OutMsgs 0, tcb 0x650FE6E0, listener 0x650FE22C
Rack1R6#show ip nat translations

Pro Inside global      Inside local       Outside local      Outside global
tcp 155.1.254.1:26887  155.1.146.1:26887  112.0.0.1:23       112.0.0.1:23
Rack1R4#show ip snat peer 155.1.146.6


Show NAT Entries created by peer: 155.1.146.4

Pro Inside global      Inside local       Outside local      Outside global
tcp 155.1.254.1:26887  155.1.146.1:26887  112.0.0.1:23       112.0.0.1:23
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

NAT Virtual Interface (pending update)

Task

- Remove all previous NAT configurations on R5.
- Configure R5 so that traffic sourced from SW2's VLAN 8 network is translated to the source address range 155.X.188.0/24.
- Do not use the `ip nat inside` or `ip nat outside` commands or static routing to accomplish this.

Configuration

https://t.me/learningnets

```
R5:
interface Serial 0/0/0
 ip nat enable
!
interface Serial 0/1/0
 ip nat enable
!
interface FastEthernet 0/0
 ip nat enable
!
ip access-list standard VLAN8
 permit 155.1.8.0 0.0.0.255
!
ip nat pool NVI_POOL 155.1.188.1 155.1.188.254 prefix 24 add-route
!
ip nat source list VLAN8 pool NVI_POOL
!
router rip
 redistribute static metric 1
```

Verification

To verify translation using the NVI, disable CEF on R5's interfaces (no ip route-cache), generate traffic from SW2's VLAN 8 interface, and issue the following debugs.

https://t.me/learningnets

```
Rack1R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Rack1R5(config)#access-list 100 permit icmp any any

Rack1R5(config)#end
Rack1R5#Rack1R5#debug ip packet detail 100

IP packet debugging is on (detailed) for access list 100Rack1R5#debug ip nat detail

IP NAT detailed debugging is on
Rack1SW2#ping 155.1.45.4 source vlan8 repeat 1



Rack1R5#
IP: tableid=0, s=155.1.8.8 (FastEthernet0/0), d=155.1.45.4 (Serial0/1/0), routed via FIB
NAT: i: icmp (155.1.8.8, 10) -> (155.1.45.4, 10) [41]
     NAT: s=155.1.8.8->155.1.188.2, d=155.1.45.4 [41]

IP: tableid=0, s=155.1.188.2 (FastEthernet0/0), d=155.1.45.4 (Serial0/1/0), routed via RIB
IP: s=155.1.188.2 (FastEthernet0/0), d=155.1.45.4 (Serial0/1/0), g=155.1.45.4, len 100, forward
    ICMP type=8, code=0
```

Note that the NAT direction is always “inside” for NVI-based NAT. This means that a routing lookup is always performed before the translation. After the routing decision is made, the packet source is translated, and then the packet is forwarded down the Serial interface.

```
IP: tableid=0, s=155.1.45.4 (Serial0/1/0), d=155.1.188.2 (NVI0)
, routed via RIBNAT: i: icmp (155.1.45.4, 10) -> (155.1.188.2, 10) [41]
     NAT: s=155.1.45.4, d=155.1.188.2->155.1.8.8 [41]

IP: tableid=0, s=155.1.45.4 (Serial0/1/0), d=155.1.8.8 (FastEthernet0/0), routed via RIB
```

When the return packet comes back, it is routed using the NVI-based static route that is automatic. Note that NAT direction is “inside” again; this is why the NVI route is needed in this situation. With normal NAT outside to inside translation (the return traffic flow in this case), the packet destination would first get untranslated, and then routed using the true destination.

https://t.me/learningnets

```
Rack1R5#show ip route static

     155.1.0.0/24 is subnetted, 17 subnetsS        155.1.188.0 [0/0] via 0.0.0.0, NVI0
```

Note that the NVI-based static route eliminates the need for a statically configured route to Null0, but it does not remove the need for advertising the static route into the dynamic routing domain with the redistribute static statement.

To verify translation, issue the `show ip nat nvi translations` command, not the normal `show ip nat translations` command.

```
Rack1R5#show ip nat translations
     Pro Inside global      Inside local       Outside local      Outside global
Rack1R5#show ip nat nvi translations
 Pro Source global      Source local       Destin  local      Destin  global
--- 155.1.188.2        155.1.8.8          ---                ---
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

NAT Default Interface

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Disable R5's tunnel interface to the DMVPN network.
- Configure R5 so that all connections destined to its point-to-point link to R4 are redirected to R8's Loopback 0 interface.
- All other inside to outside traffic on R5 should be dynamically overloaded to the point-to-point link's IP address.

Configuration

https://t.me/learningnets

```
R5:

interface GigabitEthernet1.58
 ip nat inside
!
interface GigabitEthernet1.45
 ip nat outside
!
interface Tunnel0
 shutdown
!
ip access-list standard ALL
 permit any
!
ip nat inside source list ALL interface GigabitEthernet1.45 overload
ip nat inside source static 150.1.8.8 interface GigabitEthernet1.45
```

Verification

The NAT Default Interface feature allows all traffic received on the outside interface that does not already match an existing dynamic translation to be statically forwarded to an inside host. This feature is useful for applications that use different outbound and inbound traffic flows.

To verify this configuration, telnet to any outside destination from the inside devices, and view the translation table on R5.

```
R8#telnet 155.1.45.4
Trying 155.1.45.4 ... Open
!
!R4>show users
Line       User       Host(s)              Idle       Location
0 con 0                idle                 01:05:57
*  2 vty 0                idle                 00:00:00 155.1.45.5

Interface    User               Mode         Idle     Peer Address
!
!R10#telnet 155.1.45.4
Trying 155.1.146.6 ... Open
!
!R4>show users
Line       User       Host(s)              Idle       Location
0 con 0                idle                 01:07:37
*  3 vty 1                idle                 00:00:00 155.1.45.5
```

https://t.me/learningnets

```
Interface    User               Mode         Idle     Peer Address
```

All inside traffic sent outbound is translated to R5's interface address 155.1.45.5.

```
R5#sh ip nat translations

Pro  Inside global         Inside local          Outside local         Outside global
---  155.1.45.5            150.1.8.8             ---                   ---
tcp  155.1.45.5:4096       155.1.58.8:57732      155.1.45.4:23         155.1.45.4:23
tcp  155.1.45.5:4097       155.1.108.10:27718    155.1.45.4:23         155.1.45.4:23

Total number of translations: 3
```

All outside traffic sent inbound is redirected to the default host of R8.

```
R4#telnet 155.1.45.5
Trying 155.1.45.5 ... Open
!
!R8#show users
Line       User       Host(s)              Idle       Location
0 con 0                150.1.1.1            00:01:29
*  2 vty 0                idle                 00:00:00 155.1.45.4

Interface    User               Mode         Idle     Peer Address
!
!R5#sh ip nat translations

Pro  Inside global         Inside local          Outside local         Outside global
---  155.1.45.5            150.1.8.8             ---                   ---
tcp  155.1.45.5:23         150.1.8.8:23          155.1.45.4:35072      155.1.45.4:35072
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Reversible NAT

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Shutdown the Tunnel0 interface on R5.
- Configure R5 so that traffic from R10 received on the VLAN 58 interface is translated to the pool 155.1.45.100–155.1.45.200 as it exits out the point-to-point link to R4.
- Use route-map-based NAT for this configuration to allow outside hosts to initiate connections back to hosts on the inside network after an inside-to-outside dynamic entry is created in the NAT table.

Configuration

R5:

```
interface Tunnel0
 shutdown
!
interface GigabitEthernet1.58
 ip nat inside
!
interface GigabitEthernet1.45
 ip nat outside
!
ip nat pool POOL 155.1.45.100 155.1.45.200 prefix-length 24
!
ip access-list standard INSIDE_HOSTS
 permit host 150.1.10.10
```

https://t.me/learningnets

```
 permit host 155.1.108.10
!
route-map CREATE_EXTENDABLE_ENTRIES
 match ip address INSIDE_HOSTS
 match interface GigabitEthernet1.45
!
ip nat inside source route-map CREATE_EXTENDABLE_ENTRIES pool POOL reversible
```

Verification

By default, when you use route-maps with NAT rules, extendable entries are created. This disallows an external user to open a reverse connection back to an inside host because no one-to-one mapping exists in the translation table. Reversible NAT allows creation of extendable entries along with reversible one-to-one mappings.

To verify this configuration, attempt to open a connection from the outside network to hosts on the inside network using the inside global NAT pool.

```
R6#telnet 155.1.45.101
Trying 155.1.45.101 ... % Connection timed out; remote host not responding
R6#telnet 155.1.45.102
Trying 155.1.45.102 ... % Connection timed out; remote host not responding
R6#telnet 155.1.45.103
Trying 155.1.45.103 ... % Connection timed out; remote host not responding
R6#telnet 155.1.45.104
Trying 155.1.45.104 ... % Connection timed out; remote host not responding
```

These connections are denied because R5 has no entries installed in the NAT table.

```
R5#show ip nat translations
```

Initiate a connection from the inside network to the outside network and verify R5's translation table.

```
R10#telnet 155.1.146.6
Trying 155.1.146.6 ... Open
R6#show users
    Line       User       Host(s)              Idle       Location
   0 con 0                 idle                 00:05:26
* 66 vty 0                 idle                 00:00:00 155.1.45.100
```

https://t.me/learningnets

```
Interface    User               Mode         Idle     Peer Address
```

Note that R5 creates both an extendable entry and a one-to-one entry. When this one-to-one entry is created, the connection is reversible, and hosts on the outside network can initiate connections to hosts on the inside.

```
R1#telnet 155.1.45.100
Trying 155.1.45.100 ... Open
R10>
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Static Extendable NAT

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 so that R10's Loopback0 address is reachable from the outside of the network using the IP addresses 155.1.45.201 and 155.1.45.202 at the same time.

Configuration

```
R5:

interface GigabitEthernet1.45
 ip nat outside
!
interface Tunnel0
 ip nat outside
!
interface GigabitEthernet1.58
 ip nat inside
!
ip nat inside source static 150.1.10.10 155.1.45.201 extendable
ip nat inside source static 150.1.10.10 155.1.45.202 extendable
```

Verification

Extendable static NAT allows you to configure multiple static mappings for the same

https://t.me/learningnets

local or global IP address. In this example, the same local IP address is mapped to multiple global IP addresses. The extendable keyword allows every new translation to be fully extended, without binding a local IP address to a fixed global IP.

To verify this, telnet to both of the global IP addresses mapped to the local IP addresses of R10.

```
R5#sh ip nat translations

Pro  Inside global         Inside local          Outside local         Outside global
---  155.1.45.201          150.1.10.10           ---                   ---
---  155.1.45.202          150.1.10.10           ---                   ---
Total number of translations: 2
```

Before traffic is initiated, R5 does not install extendable entries for the static mappings.

```
R6#telnet 155.1.45.201
Trying 155.1.45.201 ... OpenR10#exit

[Connection to 155.1.45.201 closed by foreign host]
!
!R6#telnet 155.1.45.202
Trying 155.1.45.202 ... Open
R10#
```

After traffic is sent, the translations are fully extended.

```
R5#show ip nat translations

Pro  Inside global         Inside local          Outside local         Outside global
---  155.1.45.201          150.1.10.10           ---                   ---
---  155.1.45.202          150.1.10.10           ---                   ---
tcp  155.1.45.201:23       150.1.10.10:23        155.1.146.6:32048     155.1.146.6:32048
tcp  155.1.45.202:23       150.1.10.10:23        155.1.146.6:20239     155.1.146.6:20239

Total number of translations: 4
```

The rule that appears first in the configuration is used to resolve the ambiguity when translating from one to many addresses. This can be verified by initiating connections from the inside out.

https://t.me/learningnets

```
R10#telnet 150.1.1.1 /source-interface Loopback0

Trying 150.1.1.1 ... Open
!
!R1#show users
Line       User       Host(s)              Idle       Location
0 con 0                idle                 07:39:39
*  2 vty 0                idle                 00:00:00 155.1.45.201

Interface    User               Mode         Idle     Peer Address
!
!R10#telnet 150.1.6.6 /source-interface Loopback0
Trying 150.1.6.6 ... Open
!
!R6#sh users
Line       User       Host(s)              Idle       Location
0 con 0                idle                 00:10:43
*  2 vty 0                idle                 00:00:00 155.1.45.201


Interface    User               Mode         Idle     Peer Address
```

Static policy NAT with route-maps can be used to explicitly specify which NAT rules to use and when they are used (such as per interface).

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IP Precedence Accounting (pending update)

Load the *RIP Routing* initial configurations prior to starting.

Task

- Configure R6 to account for the number of input and output packets based on their IP Precedence values on its connection to R7.

Configuration

```
R6:
interface GigabitEthernet1.67
 ip accounting precedence input
 ip accounting precedence output
```

Verification

Routing protocols by default use an IP precedence value of 6. ICMP ping uses a default IP precedence of 0. These packets can be used to test the precedence accounting.

https://t.me/learningnets

```
R6#ping 155.1.67.7


Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/32/33 ms
Rack1R6#
```

Extended ping can be used to change the IP precedence of the packets generated.

```
R6#ping
Protocol [ip]: Target IP address: 155.1.67.7

Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
```

The first three bits of the Type of Service (TOS) field are IP precedence. Therefore, a precedence of 3 in binary would be 01100000, or 96 in decimal. A shortcut for this is to take the precedence value and multiple it by 32 to get its decimal representation (for example, 3 X 32 = 96).

```
Type of service [0]: 96

Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/32/36 ms
```

To verify these statistics accumulated by the Serial interface, issue the `show interface [interface] precedence` command. Use the `clear counters` command to clear these statistics.

https://t.me/learningnets

```
Rack1R6#show interfaces GigabitEthernet1.67 precedence

GigabitEthernet1.67
  Input
    Precedence 0:  5 packets, 520 bytes
    Precedence 3:  5 packets, 520 bytes
    Precedence 6:  712 packets, 58455 bytes
  Output
    Precedence 0:  5 packets, 520 bytes
    Precedence 3:  5 packets, 520 bytes
    Precedence 6:  62 packets, 3634 bytes
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IP Output Packet Accounting (pending update)

Load the *RIP Routing* initial configurations prior to starting.

Task

- Configure R1 to account for the number of output packets on all physical interfaces.
- Only account for packets going to subnets of 155.1.0.0/16.
- Limit the accounting database size to 4096 entries.
- The number of records for packets not matching the account list should be set to 1.

Configuration

```
R1:
ip accounting-threshold 4096
ip accounting-transits 1
ip accounting-list 155.1.0.0 0.0.255.255
!
interface GigabitEthernet1.13
 ip accounting output-packets
!
interface GigabitEthernet1.146
 ip accounting output-packets
!
interface Tunnel0
 ip accounting output-packets
```

Verification

IP accounting for output packets keeps local statistics of source and destination pairs, packet counts, and byte counts for packets leaving a particular link. The

https://t.me/learningnets

accounting-list matches either sources or destinations of outgoing packets that should be accounted for. Transit-records are accounting records created for packets not matching any of the accounting lists.

Those records do not contain detailed information, but instead just a notification of the first packet in the flow. By default, if there is no accounting-list configured, the value of the transit records is ignored. The default number of transit records is 0, so when you enable an accounting-list, all non-matching packets are not accounted for.

To test this configuration, generate some traffic to populate the accounting database.

```
R3#ping 155.1.146.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/31/32 ms
R3#ping 150.1.6.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/31/32 ms
R3#ping 150.1.6.6 source loopback 0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:
Packet sent with a source address of 150.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/45/108 ms
R3#ping 150.1.4.4 source loopback 0



Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 150.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/29/32 ms
```

Issue the `show ip accounting` command to view the accounting database. In this case, we should see entries for flows that match the accounting-list (from or to the 155.1.0.0/16 subnet). Additionally, there should be just one entry for the packet exchange between R3's Loopback 0 and R6's Loopback 0. This is because the transit-record limit is set to 1. The second traffic flow from R3's Loopback 0 to R4's

https://t.me/learningnets

Loopback0 will exceed the transit-record limit and therefore will not be recorded.

```
R1#show ip accounting

   Source           Destination              Packets               Bytes
150.1.3.3        150.1.6.6                        1                 100

 150.1.6.6        155.1.13.3                       5                 500
 155.1.13.3       150.1.6.6                        5                 500
 155.1.146.4      155.1.13.3                       5                 500
 155.1.13.3       155.1.146.4                      5                 500
```

When you clear the active accounting database, it is copied to the “checkpoint” database, and accounting starts over. You can retrieve the checkpoint database separately, as seen below. Every time you clear the active database, it is copied to the checkpoint.

```
R1#clear ip accounting
 R1#show ip accounting
   Source           Destination              Packets               Bytes

Accounting data age is 0
R1#show ip accounting checkpoint

   Source           Destination              Packets               Bytes
 150.1.3.3        150.1.6.6                        1                 100
 150.1.6.6        155.1.13.3                       5                 500
 155.1.13.3       150.1.6.6                        5                 500
 155.1.146.4      155.1.13.3                       5                 500
 155.1.13.3       155.1.146.4                      5                 500

Accounting data age is 8
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IP Access Violation Accounting (pending update)

Load the *RIP Routing* initial configurations prior to starting.

Task

- Configure R6 to deny packets with an IP precedence of 4 coming from R7.
- Account for the packets denied by this policy.

Configuration

```
R6:
access-list 101 deny ip any any precedence 4
access-list 101 permit ip any any
!
interface GigabitEthernet1.67
 ip access-group 101 in
 ip accounting access-violations
```

Verification

To test this configuration, generate packets from R6 to R7 using a TOS of 128 (IP Precedence value of 4).

```
R6#sh ip accounting access-violations

   Source          Destination             Packets              Bytes   ACL

Accounting data age is 01:34
R6#ping
Protocol [ip]:
Target IP address: 155.1.67.7
```

https://t.me/learningnets

```
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: Type of service [0]: 128
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.7, timeout is 2 seconds:.....
Success rate is 0 percent (0/5)
R6#show ip accounting access-violations

   Source           Destination              Packets               Bytes   ACL
155.1.67.7         155.1.67.6                  5                  500    101



Accounting data age is 01:35
```

Pitfall

Access-violation accounting does not work with named access-lists.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

MAC Address Accounting (pending update)

Task

- Configure R1 to account for the MAC addresses in IP packets entering or leaving its connection to VLAN 146.

Configuration

```
R1:
interface GigabitEthernet1.146
 ip accounting mac-address input
 ip accounting mac-address output
```

Verification

MAC address accounting has fixed size of 512 entries for input and output frames. The below output shows source MAC addresses for input packets, and destination MAC addresses for output packets, as well as packet counts and their cumulative size.

https://t.me/learningnets

```
Rack1R1#show interfaces GigabitEthernet1.146 mac-accounting


GigabitEthernet1.146
  Input  (510 free)
    0006.d7bc.ed82(2  ):  21 packets, 8526 bytes, last: 6551ms ago
    0011.9221.da80(248):  21 packets, 9786 bytes, last: 4275ms ago
                  Total:  42 packets, 18312 bytes
  Output  (511 free)
    0100.5e00.0009(86 ):  21 packets, 6006 bytes, last: 1831ms ago
                  Total:  21 packets, 6006 bytes
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

TCP Optimization

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Modify R1's TCP configuration to meet the following requirements:
- Avoid the TCP "silly window syndrome."
- Enable high-performance TCP options.
- Set the TCP window size to twice the standard 16-bit maximum value.
- Limit the wait time for a TCP SYN response to the minimum.
- Enable the feature to avoid fragmentation with TCP sessions.
- Hold no more than 16 packets in the outgoing TCP queue.

Configuration

R1:

```
service nagle
ip tcp ecn
ip tcp selective-ack
ip tcp timestamp
ip tcp window-size 131070
ip tcp queuemax 16
ip tcp synwait-time 5
ip tcp path-mtu-discovery
```

https://t.me/learningnets

Verification

For more information on `service nagle`, refer to RFC 896, *Congestion Control in IP/TCP Internetworks*, and RFC 1323, *TCP Extensions for High Performance*.

To verify this configuration, enable the same set of options in R6 and telnet from R6 to R1. When the session is open, examine the TCP connection status on R6 and R1.

```
R6#telnet 150.1.1.1
Trying 150.1.1.1 ... Open
R1#
!
!R1#show tcp  brief all

TCB       Local Address               Foreign Address             (state)
7F266ECE2D08  150.1.1.1.23               155.1.146.6.60048           ESTAB
R1#
!
!R1#show tcp tcb 7F266ECE2D08

Connection state is ESTAB, I/O status: 1, unread input bytes: 1
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 150.1.1.1, Local port: 23
Foreign host: 155.1.146.6, Foreign port: 60048
Connection tableid (VRF): 0
Maximum output segment queue size: 16

Enqueued packets for retransmit: 1, input: 0  mis-ordered: 0 (0 bytes)

Event Timers (current time is 0xD5E1676):
Timer          Starts    Wakeups            Next
Retrans            37          0       0xD5E1A97
TimeWait            0          0             0x0
AckHold            20          1             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0
Linger              0          0             0x0
ProcessQ            0          0             0x0

iss: 1778091425  snduna: 1778091754  sndnxt: 1778091756
irs: 2376386617  rcvnxt: 2376386705
```

https://t.me/learningnets

```
sndwnd:   3800  scale:      0  maxrcvwnd: 131070
rcvwnd: 130983  scale:      1  delrcvwnd:     87

SRTT: 992 ms, RTTO: 1059 ms, RTV: 67 ms, KRTT: 0 ms
minRTT: 3 ms, maxRTT: 1000 ms, ACK hold: 200 ms
Status Flags: passive open, active open
Option Flags: nagle, Retrans timeout
IP Precedence value : 6

Datagrams (max data segment is 536 bytes):
Rcvd: 59 (out of order: 0), with data: 21, total data bytes: 87
Sent: 66 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 62, total data bytes: 3

Packets received in fast path: 0, fast processed: 0, slow path: 0
fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7F266D1EC460  FREE
!
!R1#show tcp brief all
TCB       Local Address               Foreign Address             (state)
7F266ECE2D08  150.1.1.1.23               155.1.146.6.60048           ESTAB
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IOS Small Services and Finger

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R1 to meet the following requirements:
  - Users who telnet or send UDP packets to R1 on port 7 should have any typed characters echoed back to them.
  - Users who telnet or send UDP packets to R1 on port 9 should have any typed characters thrown away.
  - Users who telnet to R1 at port 13 should be returned the date and time.
  - Users who telnet or send UDP packets to R1 on port 19 should be sent a stream of ASCII characters.
  - Users who telnet to R1 at port 79 should be sent a list of currently logged-in users.

Configuration

```
R1:

service udp-small-servers
service tcp-small-servers
ip finger
```

https://t.me/learningnets

Verification

Connect to the well-known TCP ports on R1 to verify the small services. Use the escape sequence Ctrl-Shift-6-6 X to escape the session and disconnect it.

```
R1#telnet 150.1.1.1 echo
Trying 150.1.1.1, 7 ... Open
!
!R1#telnet 150.1.1.1 discard
Trying 150.1.1.1, 9 ... Open
!
!R1#telnet 150.1.1.1 daytime
Trying 150.1.1.1, 13 ... Open
Sunday, May 4, 2014 09:22:06-UTC
!
!R1#telnet 150.1.1.1 chargen
Trying 150.1.1.1, 19 ... Open
 Trying 150.1.1.1, 19 ... Open
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefg
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefgh
"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghi
#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghij
$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijk
%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijkl
&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklm
'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmn
()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmno
)*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnop
!
!R1#telnet 150.1.1.1 finger
 Trying 150.1.1.1, 79 ... Open



Line       User       Host(s)              Idle       Location
0 con 0                idle                 00:04:25
*  2 vty 0                idle                 00:00:00 155.1.146.6

Interface    User               Mode         Idle     Peer Address

[Connection to 150.1.1.1 closed by foreign host]
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Directed Broadcasts and UDP Forwarding

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 so that if it receives packets for the destination address 155.1.58.255, they are forwarded out the link to R8 but the destination is not changed to 255.255.255.255.
- Configure R5 to listen to UDP broadcasts on the point-to-point link to R4, and forward them to the address 155.1.58.255.
- This forwarding should only occur for DNS packets.
- Verify this configuration by sending ICMP pings to the address 155.1.58.255 and DNS broadcast requests from R4.

Configuration

R5:

```
no ip forward-protocol udp bootps
no ip forward-protocol udp tftp
no ip forward-protocol udp time
no ip forward-protocol udp netbios-ns
no ip forward-protocol udp netbios-dgm
no ip forward-protocol udp tacacs
!
interface GigabitEthernet1.45
 ip helper-address 155.1.58.255
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.58
 ip broadcast-address 155.1.58.255
 ip directed-broadcast
```

Verification

A directed broadcast packet is a packet whose destination address matches the last IP address in a subnet assigned to the local device. For example, the network 10.0.0.0/8 has a directed broadcast address, or subnet broadcast address, of 10.255.255.255. The directed broadcast feature, which is disabled by default, allows remote devices to send a broadcast packet to a particular link by sending the packet to the directed broadcast address. Normally this feature is not allowed because of the security risks inherent to broadcasts, such as smurf and fraggle DoS attacks.

When the router receives a directed broadcast, and the `ip directed-broadcast` command is enabled on the outgoing interface where the address match occurs, the router changes the destination address in the packet from the directed broadcast address to the all subnet broadcast address of 255.255.255.255. This behavior can be overridden with the `ip broadcast-address` command, which it is in the above case.

The UDP forwarding feature, controlled by the `ip helper-address` , is used to take broadcast UDP packets and change the destination address to a unicast or directed broadcast address. By default, the router will forward UDP packets only for the following protocols: TACACS (not TACACS+), TFTP, BOOTP, Time, NetBIOS Name Server, and NetBIOS Datagram Services and DNS. In the above case, all of these protocols are disabled with the exception of DNS.

To verify this configuration, send packets from R4 to the broadcast IP address of VLAN 58. Enable IP packet debugging for ICMP packets on R5 and R8 to view the results.

```
R5, R8:
access-list 100 permit icmp any any
 !
 !R5#debug ip packet detail 100
R8#debug ip packet detail 100
 !
 !R4#ping 155.1.58.255
 !
 !
 Type escape sequence to abort.
 Sending 5, 100-byte ICMP Echos to 155.1.58.255, timeout is 2 seconds:!!!!!
 Success rate is 100 percent (5/5), round-trip min/avg/max = 32/33/36 ms
 !
```

https://t.me/learningnets

```
!
R5#
IP: s=155.1.45.4 (GigabitEthernet1.45), d=155.1.58.255, len 100, input feature
ICMP type=8, code=0, MCI Check(95), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
FIBipv4-packet-proc: route packet from GigabitEthernet1.45 src 155.1.45.4 dst 155.1.58.255
FIBfwd-proc: Default:155.1.58.255/32 receive entry
FIBipv4-packet-proc: packet routing failed
IP: tableid=0, s=155.1.45.4 (GigabitEthernet1.45), d=155.1.58.255 (GigabitEthernet1.58)
R5#, routed via RIB
IP: s=155.1.45.4 (GigabitEthernet1.45), d=155.1.58.255 (GigabitEthernet1.58), g=255.255.255.255, len 100, forward di
    ICMP type=8, code=0

IP: s=155.1.45.4 (GigabitEthernet1.45), d=155.1.58.255 (GigabitEthernet1.58), len 100, rcvd 5
    ICMP type=8, code=0
!
!
R8#
IP: s=155.1.45.4 (GigabitEthernet1.58), d=155.1.58.255, len 100, stop process pak for forus packet
ICMP type=8, code=0

IP: s=155.1.58.8 (local), d=155.1.45.4, len 100, local feature
ICMP type=0, code=0, feature skipped, Logical MN local(14), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALS
```

R5 receives the directed broadcast and forwards it on to VLAN 58. R8 receives the packet, processes the packet locally, and sends an ICMP echo reply back to R4. This behavior is considered a security risk and is exploited in the smurf and fraggle attacks. These attacks send ICMP or UDP echo packets to a directed broadcast address with a spoofed source. When the destination segment receives the broadcast, all hosts on the segment reply back to the spoofed source (the victim). If the link is a /24 subnet that is fully populated with 253 hosts, it means that for every one request sent from the attacker, 253 replies are sent back to the victim. To prevent this, the `ip directed-broadcast` command is simply disabled at the interface level.

To test the UDP forwarding, enable DNS name lookup on R4 without configuring a DNS server. This causes the router to send DNS requests to the all subnet broadcast address of 255.255.255.255 out all links. When R5 hears this packet, it changes the destination address to the helper, 155.1.58.255, and forwards it as a directed broadcast to VLAN 58.

In a real network design, this behavior can be useful for DNS and DHCP forwarding. If there are multiple DNS or DHCP servers on a segment, traffic can be load shared to the fastest server by sending the name or address request to both at the same

https://t.me/learningnets

time. Whichever server replies first will be used for that particular application.

```
R4#conf t
R4#ip domain-lookup
R4(config)#access-list 100 permit udp any any eq 53
R4#debug ip packet detail 100
!
!
R4#testname
Translating "testname"...domain server (255.255.255.255)
IP: s=155.1.45.4 (local), d=255.255.255.255 (GigabitEthernet1.45), len 54, sending broad/multicast
UDP src=61681, dst=53

IP: s=155.1.146.4 (local), d=255.255.255.255 (GigabitEthernet1.146), len 54, sending broad/multicast
UDP src=61681, dst=53
```

R8 receives the DNS request sent from R4 as a directed broadcast through R5.

```
IP: s=155.1.45.4 (GigabitEthernet1.58), d=155.1.58.255 (GigabitEthernet1.58), len 54, rcvd 3
UDP src=52715, dst=53
IP: s=155.1.45.4 (GigabitEthernet1.58), d=155.1.58.255, len 54, stop process pak for forus packet
UDP src=52715, dst=53
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

DRP Server Agent

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R1 to support reporting of information to a Distributed Director.
- Only allow connections from Directors from VLAN 146; authenticate the communication using the key value **CISCO**.

Configuration

```
R1:

ip drp access-group 99
ip drp authentication key-chain DRP
ip drp server
!
key chain DRP
 key 1
  key-string CISCO
!
access-list 99 permit 155.1.146.0 0.0.0.255
```

Verification

```
R1#show ip drp

Director Responder Protocol Agent is enabled
```

https://t.me/learningnets

```
0 director requests:
0 successful route table lookups
0 successful measured lookups
0 no route in table
0 nortt
0 DRP packet failures returned
Authentication is enabled, using "DRP" key-chain
Director requests filtered by access-list 99
rttprobe source port is     : 53
rttprobe destination port is: 53
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

NBAR Protocol Discovery

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R4 to collect protocol statistics on its connections to R5 using NBAR.
- Both routers should be able to classify connections to HTTP proxy ports 3128 and 8080.
- Change the SOCKS protocol mapping to port 2080.
- Create a new custom protocol mapping called TEST that matches the ASCII character “A” in the beginning of a TCP segment flowing to the destination port 3001.

Configuration

```
R4:

interface GigabitEthernet1.45
 ip nbar protocol-discovery
!
ip nbar custom HTTP_PROXY destination tcp 3128 8080
ip nbar port-map socks tcp 2080
ip nbar custom TEST 0 ascii A destination tcp 3001
```

Verification

NBAR protocol discovery is used for network traffic monitoring. It displays the protocols flowing across a particular interface along with their statistics. As shown in

https://t.me/learningnets

this example, you can also define new protocols by using port-mapping or a low-level byte string match.

```
R4#show ip nbar protocol-discovery

GigabitEthernet1.45

Last clearing of "show ip nbar protocol-discovery" counters 00:01:31


                        Input                    Output
                        -----                    ------
Protocol                 Packet Count             Packet Count
                        Byte Count               Byte Count
                        5min Bit Rate (bps)      5min Bit Rate (bps)
                        5min Max Bit Rate (bps)  5min Max Bit Rate (bps)
------------------------ ------------------------ ------------------------
eigrp                    7                        8
                        546                      624
                         0                        0
                         0                        0
Total                    7                        8
                        546                      624
                         0                        0
                         0                        0
!
!R5#ping 155.1.45.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.45.4, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/9/16 ms
!
!R4#show ip nbar protocol-discovery

GigabitEthernet1.45

Last clearing of "show ip nbar protocol-discovery" counters 00:03:58


                        Input                    Output
                        -----                    ------
Protocol                 Packet Count             Packet Count
                        Byte Count               Byte Count
                        5min Bit Rate (bps)      5min Bit Rate (bps)
                        5min Max Bit Rate (bps)  5min Max Bit Rate (bps)
------------------------ ------------------------ ------------------------
eigrp                    40                       40
```

https://t.me/learningnets

```
                          3120                     3120
                          0                        0
                          0                        0
ping                      10                       10
                          1180                     1180
                          0                        0
                          0                        0
ipv6-icmp                 1                        1
                         122                      122
                          0                        0
                          0                        0
```

The port values that NBAR uses for matching are as follows.

```
R4#sh ip nbar port-map

port-map HTTP_PROXY               tcp 3128 8080 port-map TEST                    tcp 3001

port-map active-directory         udp 389
port-map active-directory         tcp 443 445 139 389 135
port-map activesync               tcp 80
port-map adobe-connect            tcp 443 80
port-map airplay                  tcp 554 8554 80
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Netflow Ingress and Egress (pending update)

Task

- Configure R4 and R6 to export Netflow version 5 information to a host 155.1.146.100 in VLAN 146 using port 9999.
- R4 should collect statistics on incoming flows on all interfaces, and R6 should collect statistics on outgoing flows on all interfaces.
- Limit the size of the flow cache to 4096 entries and export the BGP AS of Origin if available.
- Collect additional information about ICMP message information as well as layer 2 VLAN numbers where applicable.

Configuration

```
R4:
ip flow-capture vlan-id
ip flow-capture icmp
ip flow-export version 5
ip flow-export destination 155.1.146.100 9999
ip flow-export version 5 origin-as
ip flow-cache entries 4096
!
interface GigabitEthernet1.146
 ip flow ingress
!
interface GigabitEthernet1.45
 ip flow ingress
!
interface Tunnel0
 ip flow ingress
!
```

https://t.me/learningnets

```
R6:
ip flow-capture vlan-id
ip flow-capture icmp
ip flow-export version 5
ip flow-export version 5 origin-as
ip flow-export destination 155.1.146.100 9999
ip flow-cache entries 4096
!
interface GigabitEthernet1.146
 ip flow egress
!
interface GigabitEthernet1.67
 ip flow egress
```

Verification

Netfow evolved from a route-caching (switching path) technology to a powerful statistics-gathering tool. Flow collection on interfaces can be enabled for either inbound (ingress) or outbound (egress) flow. The difference between them is that ingress flow monitoring will capture all flows entering the router, including the packets destined to the router itself. Egress flow collects packets switched across the router that are leaving via an egress interface, but it does not include packets originated from the router itself.

Egress flow collection has some performance penalties, but it is useful when you need to collect Netflow information on MPLS traffic, which leaves the router untagged. This is needed because ingress Netflow can’t look inside the MPLS tagged packets. You may distinguish egress flows in the show command output by recognizing a “*” sign near the destination interface. Another difference between ingress and egress Netflow is their interaction with other IOS features, such as NAT, rate-limiting, and encryption. Ingress Netflow is applied before ACLs and rate-limiting, whereas egress Netflow is applied after rate-limiting and encryption features.

Different versions of Netflow are available, with the most basic, widely available, and default version being 5. Version 8 supports aggregation caches, and version 9, which is incompatible with previous versions, allows for the most flexibility with choosing the export template format. Aggregation caches are also supported by version 9, so version 8 is becoming deprecated.

To verify this configuration, generate some flows and check verbose flow-cache output as follows.

https://t.me/learningnets

```
R1#telnet 212.18.1.1

Trying 212.18.1.1 ... Open

<snip>
R6#show ip cache verbose flow

IP packet size distribution (30 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .566 .000 .333 .033 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .066 .000 .000 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes 2 active, 4094 inactive, 4 added
  50 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 21640 bytes
  4 active, 1020 inactive, 8 added, 4 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
ICMP                 2      0.0         5   100      0.0       0.1      15.9
Total:               2      0.0         5   100      0.0       0.1      15.9

SrcIf          SrcIPaddress    DstIf          DstIPaddress    Pr TOS Flgs  Pkts
Port Msk AS                    Port MskAS    NextHop              B/Pk  Active
Fa0/0.146      155.1.146.1     Se0/0*         212.18.1.1      06 C0  1A      10
 2AF8 /24 0                     0017 /24 0     54.1.1.254             43     0.4
FFlags: 01
  VLAN id:       146                            000
Min plen:        40                            Max plen:        52
Se0/0          212.18.1.1      Fa0/0.146*     155.1.146.1     06 C0  1A      10
 0017 /24 0                     2AF8 /24 0     155.1.146.1           160     0.6
FFlags: 01
  VLAN id:       000                            146
Min plen:        40                            Max plen:       576
Rack1R1#ping 30.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/5/8 ms
```

https://t.me/learningnets

```
Rack1R4#show ip cache verbose flow

IP packet size distribution (190 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .015 .000 .105 .000 .052 .000 .000 .168 .000 .000 .000 .494 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .163 .000 .000 .000 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes 6 active, 4090 inactive, 21 added
  1389 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 21640 bytes
  12 active, 1012 inactive, 42 added, 21 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
Protocol         Total     Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows      /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-BGP              3       0.0         1    44      0.0       0.0      15.7
UDP-other           10       0.0         1   192      0.0       0.0      15.5
ICMP                 2       0.0         5   100      0.0       0.0      15.5
Total:              15       0.0         1   132      0.0       0.0      15.5

SrcIf          SrcIPaddress    DstIf          DstIPaddress    Pr TOS Flgs  Pkts
Port Msk AS                    Port MskAS    NextHop              B/Pk  Active
Et0/1          155.1.146.1     Et0/0          30.0.0.1        01 00  10       5
 0000 /24 0                     0800 /16 0     204.12.1.254          100     0.0
VLAN id:       000                            000
Min plen:      100                            Max plen:       100
ICMP type:       8                            ICMP code:        0

Se0/1/0        155.1.45.5      Null           224.0.0.9       11 C0  10      32
0208 /24 0                     0208 /24 0     0.0.0.0               412   284.7
VLAN id:       000                            000
Min plen:      412                            Max plen:       412

<snip>
```

https://t.me/learningnets

```
Et0/0          30.0.0.1        Et0/1          155.1.146.1     01 00  10        5
 0000 /16 0                      0000 /24 0     155.1.146.1           100     0.0
VLAN id:       000                            000
Min plen:      100                            Max plen:       100
ICMP type:       0                            ICMP code:        0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Netflow Top Talkers (pending update)

Task

- Configure R4 and R6 to collect information about the 10 most active ICMP flows.
- R4 should count packets, and R6 should count bytes.
- Match only TCP and ICMP flows coming from the subnet 155.X.0.0/16.

Configuration

```
R4:
ip flow-top-talkers
 top 10
 sort-by packets
 match source address 155.1.0.0 255.255.0.0
 match protocol 1

R6:
ip flow-top-talkers
 top 10
 sort-by bytes
 match source address 155.1.0.0 255.255.0.0
 match protocol 1
```

Verification

https://t.me/learningnets

The top talkers feature collects information about the most packet- or traffic-intense flows. The information is displayed only about the active, non-expired flows, so you may miss some short and bursty transmission. A flexible filtering system allows customization of the view per your requirements. The feature can be verified as follows.

```
Rack1R1#ping 212.18.1.1 repeat 50



Type escape sequence to abort.
Sending 50, 100-byte ICMP Echos to 212.18.1.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (50/50), round-trip min/avg/max = 32/35/44 ms
Rack1R1#ping 30.0.0.1 repeat 50

Type escape sequence to abort.
Sending 50, 100-byte ICMP Echos to 30.0.0.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (50/50), round-trip min/avg/max = 4/5/8 ms
Rack1R4#show ip flow top-talkers



SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Et0/1         155.1.146.1     Et0/0         30.0.0.1        01 0000 0800 50
1 of 10 top talkers shown. 1 of 7 flows matched.
Rack1R6#show ip flow top-talkers



SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP Bytes
Fa0/0.146     155.1.146.1     Se0/0*        212.18.1.1      01 0000 0800 5000

1 of 10 top talkers shown. 1 of 2 flows matched.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Netflow Aggregation Cache (pending update)

Task

- Create an aggregation cache that is destination-prefix-based on R4 and source-based on R6.
- Export the new information to 155.X.146.100 at port 9998 with Netflow version 8.
- Do not summarize addresses to more than a /8 prefix length.
- Store no more than 1024 aggregated entries.

Configuration

```
R4:
ip flow-aggregation cache destination-prefix
 cache entries 1024
 export destination 155.1.146.100 9998
 mask destination minimum 8
 enabled

R6:
ip flow-aggregation cache source-prefix
 cache entries 1024
 export destination 155.1.146.100 9998
 mask source minimum 8
 enabled
```

Verification

The Netflow aggregation cache is an additional flow cache used to store aggregated prefix information. A special Netflow packet format that was first supported in version 8, and more recently in version 9, is used to export the aggregated

https://t.me/learningnets

information to a collector.

The aggregation cache is based on the main (normal) flow cache, and flow aggregation is based on the routing table contents. For example, when a flow is exported from the main cache to the aggregation cache, the flow's source and/or destination IP addresses (based on the configured scheme) are logically ANDed bitwise with the network masks of its corresponding route in the routing table. The resulting "masked" (less-specific) flow entry is merged with the aggregation cache. This will result in multiple flow entries with source or destination IP addresses belonging to the same prefix in the IP routing table looking the same in aggregation cache, with the bytes and/or packet count being the sum of all specific flows. Additionally, the number of merged flows for each aggregated entry is stored in aggregation cache.

An additional feature, called the minimum aggregation mask, allows preserving a specified level of granularity even when widely scoped summary routes are installed in the routing table. For example, if the router has the prefix 10.0.0.0/8 in its routing table, you may still want to know detailed flow information about the subnets 10.X.X.0. By setting the minimum aggregation mask to /24, the aggregation cache will choose the longer of the two masks when creating an entry. If a flow source with the IP address 10.1.2.3 is about to be aggregated, the routing table mask of /8 is compared against the minimum aggregation mask of /24, and the result is that the router installs the aggregate entry as 10.1.2.0.

```
Rack1R1#ping 30.2.0.1



Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.2.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/6/8 ms
Rack1R1#ping 30.2.0.254 time 0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.2.0.254, timeout is 0 seconds:
.....
Success rate is 0 percent (0/5)
Rack1R1#ping 30.1.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.1.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/5/8 ms
Rack1R1#ping 30.0.0.1
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/5/8 ms
```

The above case is to be summarized based on destination addresses. On R1, all prefixes in the routing table for the subnets of 30.0.0.0 have a prefix length of 16. Because max(minimum-mask, routing-prefix) = max(8,16) = 16, the prefix length in the aggregation table is /16.

Note that prefix 30.2.0.0/16 has twice as many packets as the other because of the two pings to destinations in that range.

https://t.me/learningnets

```
Rack1R4#show ip route 30.0.0.0

Routing entry for 30.0.0.0/16, 4 known subnets
  Redistributing via rip

R       30.2.0.0 [120/1] via 204.12.1.254, 00:00:23, Ethernet0/0
R       30.3.0.0 [120/1] via 204.12.1.254, 00:00:23, Ethernet0/0
R       30.0.0.0 [120/1] via 204.12.1.254, 00:00:23, Ethernet0/0
R       30.1.0.0 [120/1] via 204.12.1.254, 00:00:23, Ethernet0/0
Rack1R4#show ip cache flow aggregation destination-prefix



IP Flow Switching Cache, 69636 bytes
  5 active, 1019 inactive, 150 added
  2336 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 5896 bytes
  5 active, 251 inactive, 150 added, 150 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added

Minimum destination mask is configured to /8

Dst If          Dst Prefix      Msk  AS    Flows  Pkts B/Pk  Active
Null            224.0.0.0       /24  0        1     1   192     0.0
Fa0/1           155.1.146.0     /24  0        3    15   100    14.7
Fa0/0           30.2.0.0        /16  0        2    10   100     8.2
Fa0/0           30.0.0.0        /16  0        1     5   100     0.0
Fa0/0           30.1.0.0        /16  0        1     5   100     0.0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Netflow Random Sampling (pending update)

Task

- Modify R6's Netflow configuration so that it only samples every tenth packet in each flow.

Configuration

```
R6:
flow-sampler-map SAMPLER
 mode random one-out-of 10
!
policy-map NETFLOW_MAP
 class class-default
   netflow-sampler SAMPLER
!
interface Serial 0/0/0
 no ip flow egress
 service-policy output NETFLOW_MAP
!
interface FastEthernet 0/0.146
 no ip flow egress
 service-policy output NETFLOW_MAP
!
interface FastEthernet 0/0.67
 no ip flow egress
 service-policy output NETFLOW_MAP
```

Verification

https://t.me/learningnets

Flow sampling allows you to lower the amount of Netflow information collected, which lowers the performance impact on the router collecting statistics. Commonly, random sampling is used for the purpose of capacity planning, where precise numbers are not required, but statistical information is required. The random flow sampler randomly picks every 1 of Nth packets for each flow and uses it to build flow information. When a flow sampler is set to sample 1 of 1, it essentially behaves in normal flow-collecting mode.

The application of the flow samplers is through an MQC policy-map. An outgoing service policy will implement egress Netflow, and an incoming service policy implements ingress Netflow. The interface-level netflow configuration will override the MQC settings, so always remove the `ip flow ingress` or `ip flow egress` commands when applying Netflow samplers.

When a sampler is applied in the class-default of a policy-map, it effectively applies to all traffic entering or leaving the respective interface, provided that traffic does not match any other classes in the policy-map.

```
Rack1R1#ping 212.18.2.1 repeat 500


Type escape sequence to abort.
Sending 500, 100-byte ICMP Echos to 212.18.2.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!
<snip>
Rack1R6#show ip cache flow


IP packet size distribution (1631 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .068 .000 .925 .001 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .003 .000 .000 .000 .000 .000 .000 .000 .000


IP Flow Switching Cache, 278544 bytes
  2 active, 4094 inactive, 28 added
  452 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 21640 bytes
  4 active, 1020 inactive, 56 added, 28 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
```

https://t.me/learningnets

```
--------          Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-Telnet           14      0.0         8    72      0.0       1.4       9.6
ICMP                 12      0.0       117   100      0.0       4.5      15.4
Total:               26      0.0        58    97      0.0       2.8      12.3

SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Fa0/0.146     155.1.146.1     Se0/0*        212.18.2.1      01 0000 0800    57
Se0/0/0       212.18.2.1      Fa0/0.146*    155.1.146.1     01 0000 0000    43
```

Note that the number of packets accounted is approximately 10 times less than the number of packets actually sent and received. As the number of packets grows, the relation will tend to be more precise, so for the purpose of statistical analysis, it is possible to simply scale the numeric values collected by 10 to get realistic results.

```
Rack1R6#show policy-map interface serial 0/0/0
 netflow-sampler: SAMPLER
 Serial0/0/0

  Service-policy output: NETFLOW_MAP

    Class-map: class-default (match-any)
      1503 packets, 287842 bytes
      5 minute offered rate 2000 bps, drop rate 0 bps
      Match: any
Rack1R6#show policy-map int fastEthernet 0/0.146
 netflow-sampler: SAMPLER

 FastEthernet0/0.146

  Service-policy output: NETFLOW_MAP

    Class-map: class-default (match-any)
      1066 packets, 136236 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

Netflow Input Filters (pending update)

Task

- Modify R6's Netflow sampling so that every packet is sampled for sources on VLAN 146.
- Other packets should still be randomly sampled.

Configuration

```
R6:
ip access-list extended VLAN146
 permit ip 155.1.146.0 0.0.0.255 any
 permit ip any 155.1.146.0 0.0.0.255
!
class-map VLAN146
 match access-group name VLAN146
!
flow-sampler-map NORMAL
 mode random one-out-of 1
!
policy-map NETFLOW_MAP
 class VLAN146
   netflow-sampler NORMAL
```

Verification

Netflow input filters allow you to attach a flow sampler to any class in a policy-map. In this example, class VLAN146 samples 1/1 packets, whereas class-default (everything else) still samples 1/10 packets.

https://t.me/learningnets

```
Rack1R1#ping 212.18.2.1 repeat 500


Type escape sequence to abort.
Sending 500, 100-byte ICMP Echos to 212.18.2.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!
Success rate is 100 percent (500/500), round-trip min/avg/max = 32/37/408 ms

Rack1R6#show ip cache flow
IP packet size distribution (2631 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .042 .000 .954 .001 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .002 .000 .000 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  2 active, 4094 inactive, 30 added
  528 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 21640 bytes
  4 active, 1020 inactive, 60 added, 30 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-Telnet          14      0.0         8    72      0.0       1.4       9.6
ICMP                14      0.0       107   100      0.0       6.5      15.4
Total:              28      0.0        58    97      0.0       3.9      12.5

SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Se0/0/0       212.18.2.1      Fa0/0.146*    155.1.146.1     01 0000 0000   500
 Fa0/0.146     155.1.146.1     Se0/0*        212.18.2.1      01 0000 0800   500
```

Generate traffic from a subnet that is classified by class-default in the policy-map.

https://t.me/learningnets

Note that this traffic is statistically sampled.

https://t.me/learningnets

```
Rack1R1#ping 212.18.2.1 repeat 500 source loopback 0


Type escape sequence to abort.
Sending 500, 100-byte ICMP Echos to 212.18.2.1, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!
Success rate is 100 percent (500/500), round-trip min/avg/max = 32/37/400 ms
Rack1R1#
Rack1R6#show ip cache flow

IP packet size distribution (2731 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .041 .000 .955 .001 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .002 .000 .000 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  2 active, 4094 inactive, 32 added
  594 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 21640 bytes
  4 active, 1020 inactive, 64 added, 32 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-Telnet          14      0.0         8    72      0.0       1.4       9.6
ICMP                16      0.0       156   100      0.0       8.1      15.4
Total:              30      0.0        87    98      0.0       4.9      12.7
```

https://t.me/learningnets

```
SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Fa0/0.146     150.1.1.1       Se0/0*        212.18.2.1      01 0000 0800    47
 Se0/0/0       212.18.2.1      Fa0/0.146*    150.1.1.1       01 0000 0000    53
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IOS Authoritative DNS Server (pending update)

Load the *EIGRP Routing* initial configuration prior to starting.

Task

- Configure R4 and R6 as primary DNS servers for the domain cisco.com.
- The primary name server should be called **ns.cisco.com** with a contact mailbox of **ccie.cisco.com**.
- Both R4 and R6 should resolve the names **R4.cisco.com** and **R6.cisco.com** to all connected IP addresses of the respective router.
- R1 should use both R4 and R6 as its DNS servers in a round-robin fashion.
- R1's DNS client should complete all unqualified domain-names with the name **cisco.com**.

Configuration

```
R1:
ip name-server 155.1.146.4 155.1.146.6
ip domain-lookup
ip domain name cisco.com
ip domain round-robin

R4:
ip dns server
ip dns primary cisco.com soa ns.cisco.comccie.cisco.com
!
ip host cisco.com ns 155.1.146.4
ip host cisco.com ns 155.1.146.6
!
ip host R4.cisco.com 150.1.4.4 155.1.146.4 155.1.45.4 204.12.1.4 155.1.0.4
ip host R6.cisco.com 150.1.6.6 155.1.146.6 155.1.67.6 54.1.1.6
```

https://t.me/learningnets

```
R6:
ip dns server
!
ip dns server
ip dns primary cisco.com soa ns.cisco.comccie.cisco.com
!
ip host cisco.com ns 155.1.146.4
ip host cisco.com ns 155.1.146.6
!
ip host R4.cisco.com 150.1.4.4 155.1.146.4 155.1.45.4 204.12.1.4 155.1.0.4
ip host R6.cisco.com 150.1.6.6 155.1.146.6 155.1.67.6 54.1.1.6
```

Verification

IOS supports basic DNS server resolution, in addition to SOA, MX, NS, and SRV resource records to act as an authoritative DNS server.

The IOS DNS client can load-balance between multiple A entries returned in a DNS response in a round-robin fashion. You can specify the default domain name to complete “unqualified” names (such as “R4”) to fully-qualified DNS names (“R4.cisco.com”).

```
R4#show ip dns primary

Primary for zone cisco.com:
  SOA information: Zone primary (MNAME): ns.cisco.com
 Zone contact (RNAME): ccie.cisco.com
  Refresh (seconds):    21600
  Retry (seconds):      900
  Expire (seconds):     7776000
  Minimum (seconds):    86400
R6#show ip dns primary

Primary for zone cisco.com:
  SOA information:
  Zone primary (MNAME): ns.cisco.com
  Zone contact (RNAME): ccie.cisco.com
  Refresh (seconds):    21600
  Retry (seconds):      900
  Expire (seconds):     7776000
  Minimum (seconds):    86400
R1#ping R4
Translating "R4"...domain server (155.1.146.4)
 [OK]
```

https://t.me/learningnets

```
Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to155.1.45.4
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 100/100/100 ms
R1#ping R4

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to204.12.1.4
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 100/100/100 ms
R1#ping R4

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to155.1.0.4
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 112/116/124 ms
R1#show hosts
Default domain is cisco.com
Name/address lookup uses domain service
Name servers are 155.1.146.4, 155.1.146.6

Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate
       temp - temporary, perm - permanent
       NA - Not Applicable None - Not defined
Host                      Port  Flags      Age Type Address(es)
R4.cisco.com              None  (temp, OK)  0   IP 155.1.0.4
150.1.4.4
155.1.146.4
155.1.45.4
204.12.1.4
R1#ping R6
Translating "R6"...domain server (155.1.146.4)
 [OK]

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to155.1.67.6
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 100/100/104 ms
R1#ping R6

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to54.1.1.6
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 100/100/104 ms
R1#ping R6
```

https://t.me/learningnets

```
Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 150.1.6.6
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 100/100/104 ms
R1#ping R6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.146.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
R1#show hosts
Default domain is cisco.com
Name/address lookup uses domain service
Name servers are 155.1.146.4, 155.1.146.6

Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate
       temp - temporary, perm - permanent
       NA - Not Applicable None - Not defined
Host                      Port  Flags      Age Type Address(es)
R6.cisco.com              None  (temp, OK)  0   IP 155.1.146.6
 155.1.67.6
 54.1.1.6
 150.1.6.6
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IOS Caching DNS Server (pending update)

Task

- Configure R3 as a DNS client of R1.
- When R1 receives a DNS query from R3, it should check its local cache.
- If the entry is not cached, a request should be sent from R1 to R4 or R6 on behalf of R3.

Configuration

```
R1:
ip name-server 155.1.146.4 155.1.146.6
ip domain-lookup
ip dns server

R3:
ip name-server 150.1.1.1
ip domain-lookup
```

Verification

To verify this configuration, issue ping commands from R3 using R4 and R6's fully qualified domain names.

```
R3#ping R4.cisco.com.
 Translating "R4.cisco.com."...domain server (150.1.1.1)
 [OK]

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to150.1.4.4
, timeout is 2 seconds:
!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 128/128/132 ms
R3#ping R6.cisco.com
Translating "R6.cisco.com"...domain server (150.1.1.1)
 [OK]

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to150.1.6.6
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 144/144/144 ms
```

Debug DNS server activities on R1 and send a DNS query from R3.

```
R1#debug domain

Domain Name System debugging is on
R3#ping R6.cisco.com

Translating "R6.cisco.com"...domain server (150.1.1.1) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 144/152/188 ms
R1#
DNS: Incoming UDP query (id#43)DNS:
Type 1 DNS query (id#43) for host 'R6.cisco.com' from 155.1.13.3(55196)
DNS:Re-sending DNS query (type 1, id#43) to 155.1.146.4
DNS: Incoming UDP query (id#43)
DNS: Type 1 response (id#43) for host <R6.cisco.com> from 155.1.146.4(53)
DOM: dom2cache: hostname is R6.cisco.com, RR type=1, class=1, ttl=10, n=4
DOM: dom2cache: hostname is R6.cisco.com, RR type=1, class=1, ttl=10, n=4
DOM: dom2cache: hostname is R6.cisco.com, RR type=1, class=1, ttl=10, n=4
DOM: dom2cache: hostname is R6.cisco.com, RR type=1, class=1, ttl=10, n=4
DNS: Forwarding back A response - no director required
DNS: Finished processing query (id#43) in 0.012 secsDNS:Forwarding back reply to 155.1.13.3/
55196
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IOS DNS Spoofing

Task

- Configure R5 as a DNS client of R4.
- Configure R8 as a DNS client of R5.
- If R5 loses connectivity to R4, it should respond to all DNS queries with the IP address of its Loopback0 interface.

Configuration

```
R5:
ip dns spoofing 150.1.5.5
ip name-server 155.1.4.4
ip domain lookup
ip dns server

R8:
ip domain lookup
ip name-server 155.1.58.5
```

Verification

A router acting as a DNS caching server with just one upstream interface will forward DNS requests to its upstream server as long as the connection is healthy. When the router loses reachability to the DNS servers, it will respond to all DNS queries with a pre-configured IP address—that is, with the IP address of an internal HTTP server to display an informational message.

```
R8#ping R4.cisco.com
Translating "R4.cisco.com"...domain server (155.1.58.5) [OK]
```

https://t.me/learningnets

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 42/45/51 ms
R8#ping TEST



Translating "TEST"...domain server (155.1.58.5)
% Unrecognized host or address, or protocol not running.
```

Shut down R5's connections to make sure it cannot reach the Loopback0 subnet of R4, and enable DNS debugging on R5.

```
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R5(config)#interface Tunnel0
R5(config-if)#shutdown
R5(config-if)#interface GigabitEthernet1.45
R5(config-if)#shutdown
R5#debug domain
```

Ping any name from R8 to see that it actually resolves to the IP address of R5's Loopback0 interface.

```
R8#ping TEST

Translating "TEST"...domain server (155.1.58.5) [OK]

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to150.1.5.5
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/8 ms
R8#ping R4.cisco.com

Translating "R4.cisco.com"...domain server (155.1.58.5) [OK]

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to150.1.5.5
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R8#ping ANYNAME



Translating "ANYNAME"...domain server (155.1.58.5) [OK]
```

https://t.me/learningnets

```
Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to150.1.5.5
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
```

Note that queries for the hostname of R5 return the IP address of the R5 interface that received the query.

```
R8#ping R5

Translating "R5"...domain server (155.1.58.5) [OK]

Type escape sequence to abort.Sending 5, 100-byte ICMP Echos to155.1.58.5
, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/9 ms
```

Observe the debugging output at R5. The router responds with the configured spoofing address to any query, with the exception of queries for its own hostname. For its own name, the router returns the IP address of the interface that received the query.

```
R5#
DNS: Incoming UDP query (id#7)
DNS: Type 1 DNS query (id#7) for host 'ANYNAME' from 155.1.58.8(52461)
DNS: No name-servers are accessible
DNS: Spoofing reply to query (id#7)
DNS: Finished processing query (id#7) in 0.000 secs
DNS: Incoming UDP query (id#8)DNS: Type 1 DNS query (id#8) for host'R5'
 from 155.1.58.8(53293)DNS: Query for my own hostname: Rack1R5

DNS: Spoofing reply to query (id#8)
DNS: Finished processing query (id#8) in 0.004 secs
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IP Services

IP Event Dampening

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **IP_Services Initial**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a tunnel interface between R6 and R7. Configure 169.254.67.0/24 network under the tunnel interface.
- Configure dampening on R6 so that after a reload, its connection via tunnel to R7 is not advertised into IGP for 30 seconds.
- Configure R6 so that when this connection flaps, it does not disappear for more than 60 seconds from the routing table, no matter how much penalty it accumulates.

Configuration

https://t.me/learningnets

```
R6:

interface Tunnel67
 ip address 169.254.67.6 255.255.255.0
 tunnel source GigabitEthernet1.67
 tunnel destination 155.1.67.7
!
interface Tunnel67
 dampening 30 1000 2000 60 restart 2000
R7:

interface Tunnel76
 ip address 169.254.67.7 255.255.255.0
 tunnel source GigabitEthernet1.67
 tunnel destination 155.1.67.6
```

Verification

To find the default IP event dampening values, configure the `dampening` command with no arguments at the interface level, and then issue the `show interface dampening` command.

Then, to suppress the interface after the router reloads, apply a restart penalty value. With a half-life time of 30 seconds and restart penalty of 2000, it means that 30 seconds after bootup the penalty will have decayed to 1000. If the reuse value is set to 1000, the end result is that the interface is installed in the routing table 30 seconds after bootup is complete.

https://t.me/learningnets

```
R6#show dampening interface

1 interface is configured with dampening.
No interface is being suppressed.
Features that are using interface dampening:
IP Routing
VRRP
!
!R6#show interfaces dampening

Tunnel67
Flaps Penalty    Supp ReuseTm   HalfL  ReuseV   SuppV  MaxSTm    MaxP Restart
    0       0   FALSE       0      30    1000    2000      60    4000    2000
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

VRF Lite

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS VRF Lite**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a new subinterface, GigabitEthernet1.76, between R6 and R7.
  - Use dot1q tag 76 and IP addressing in the format **155.1.76.Y/24**, where *Y* is the router number.
- On R6 and R7, configure GigabitEthernet1.67 so that it belongs to vrf **VPN_A** and GigabitEthernet1.76 belongs to vrf named **VPN_B**.
- Configure the following interfaces on R7:
  - Loopback101 with IP address of 172.16.7.7/24, assigned to vrf **VPN_A**.
  - Loopback102 with IP address of 192.168.7.7/24, assigned to vrf **VPN_B**.
- Configure default routes on R7 for both VRFs toward R6.
- Configure R6 so that from R7 you have IP connectivity between Loopback101 and Loopback102.

Configuration

VRF (Virtual Routing and Forwarding) tables are the fundamental building blocks needed to turn a single router into multiple virtual routers. VRFs create multiple virtual, isolated networks, where each technically has its own separate RIB (Routing Information Base) and FIB (Forwarding Information Base). These structures are also known as Routing Tables and CEF tables.

Any router interface can be assigned to a VRF using the command `ip vrf forwarding <VRF_NAME>` . Notice that this command will erase all existing IP addresses configured on the interface to avoid potential address duplication in the new routing

https://t.me/learningnets

table. After this configuration, all packets received on this interface are routed and forwarded using the associated VRF table. This concept is very similar to the way VLAN trunking works at Layer 2. Packets entering a specific VRF can only follow routes in that specific VRF's routing table. Additionally, very much like a Layer 2 trunk can span VLANS across multiple switches, VRFs can be extended across multiple devices as well.

You may extend VRFs beyond a single router by properly mapping the VRFs to the links connecting two routers. This results in parallel VPNs being run across multiple devices, also known as VRF Lite, the most basic way to build VPNs. This is the simplest way of creating non-overlapping VPNs in a network; however, this solution has poor scalability, because you must allocate a dedicated inter-router link for every VPN. Therefore, if you have two routers and 100 VPNs, you must provision 100 connections between the two routers, one for every VPN. The connection could be either a separate interface or some Layer 2 virtualization technique, such as Frame-Relay PVC or Ethernet VLAN.

By default, all interfaces (physical or sub-interfaces) are assigned to a VRF known as the **global**, which is the regular routing table used in non-VRF capable routers. To create a new VRF, issue the command `ip vrf <VRF_NAME>` , which opens the VRF configuration context mode. After this initial step, you must define a route distinguisher (RD) for this particular VRF using the command `rd X:Y` , where X and Y are 32-bit numbers. A Route Distinguisher is a special 64-bit prefix prepended to every route in the respective VRF routing table. This is done for the purpose of distinguishing the prefixes inside the router and avoiding collisions if two VRFs contain the same prefixes. The common format for an RD is the combination **ASN:NN**, where ASN is the autonomous system number and NN is the VRF number inside the router, or more globally, the VPN number within the ASN. Alternatively, you may use the format **IP-Address:NN**, where IP is the router's IP address and NN is the VRF name. The second format properly reflects the feature of RD being a local distinguisher, but using the format **ASN:NN** is more popular and common, because it easily associates a VRF with a particular VPN in the network.

https://t.me/learningnets

It is possible to associate static routes or dynamic routing protocol processes with the VRFs. In this lab, we work with static routing only. The syntax for a VRF-bound static route is **`ip route vrf <NAME> PREFIX MASK [interface] [next-hop]`**, where **`[next-hop]`** is an IP address resolvable through the VRF. It is possible to use static routes for inter-VRF communications. If you are using a static route with the **`[interface]`** specification, the interface could belong to any VRF. Note that with multi-access interfaces, you must also specify the next-hop associated with the interface subnet because Cisco IOS will install a CEF entry in the source VRF using the information provided and will not attempt to resolve the next-hop recursively. Remember that this trick only works with the non-recursive static routes that use directly connected interfaces.

R6:

```
ip vrf VPN_A
 rd 100:1
!
ip vrf VPN_B
 rd 100:2
!
interface GigabitEthernet1.67
 ip vrf forwarding VPN_A
 ip address 155.1.67.6 255.255.255.0
!
interface GigabitEthernet1.76
 encapsulation dot1q 76
 ip vrf forwarding VPN_B
 ip address 155.1.76.6 255.255.255.0
!
ip route vrf VPN_A 192.168.7.0 255.255.255.0 GigabitEthernet1.76 155.1.76.7
ip route vrf VPN_B 172.16.7.0 255.255.255.0 GigabitEthernet1.67 155.1.67.7
```

R7:

```
ip vrf VPN_A
 rd 100:1
!
ip vrf VPN_B
 rd 100:2
!
interface GigabitEthernet1.67
 ip vrf forwarding VPN_A
 ip address 155.1.67.7 255.255.255.0
!
interface GigabitEthernet1.76
 encapsulation dot1q 76
 ip vrf forwarding VPN_B
```

https://t.me/learningnets

```
 ip address 155.1.76.7 255.255.255.0
!
interface Loopback101
 ip vrf forwarding VPN_A
 ip address 172.16.7.7 255.255.255.0
!
interface Loopback102
 ip vrf forwarding VPN_B
 ip address 192.168.7.7 255.255.255.0
!
ip route vrf VPN_A 0.0.0.0 0.0.0.0 155.1.67.6
ip route vrf VPN_B 0.0.0.0 0.0.0.0 155.1.76.6
```

Verification

Start by checking the VRF interfaces and basic IPv4 connectivity. Notice that the verification commands now use the **vrf** argument to select the specific routing table.

```
R6#show ip vrf
   Name                              Default RD            Interfaces VPN_A
                            100:1 Gi1.67
 VPN_B                             100:2 Gi1.76
!
!R6#ping vrf VPN_A 155.1.67.7

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.7, timeout is 2 seconds: .!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 2/3/5 ms
!
!R6#ping vrf VPN_B 155.1.76.7

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.76.7, timeout is 2 seconds: !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 12/72/114 ms
```

Let’s look at inter-VRF connectivity in detail. First, check the CEF table for VRF **VPN_A** in R6.

```
R6#show ip route vrf VPN_A 192.168.7.0
Routing entry for 192.168.7.0/24
  Known via "static", distance 1, metric 0
  Routing Descriptor Blocks: * 155.1.76.7, via GigabitEthernet1.76
```

https://t.me/learningnets

```
      Route metric is 0, traffic share count is 1
!
!R6#show ip cef vrf VPN_A 192.168.7.0 detail

 192.168.7.0/24, epoch 0 nexthop 155.1.76.7 GigabitEthernet1.76
```

It appears accurate and complete. Now ping 172.16.7.7 from within VRF **VPN_B** in R7 and vice-versa.

```
R7#ping vrf VPN_B 172.16.7.7 source loopback102

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.7.7, timeout is 2 seconds:
Packet sent with a source address of 192.168.7.7 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/13/20 ms
!
!R7#ping vrf VPN_A 192.168.7.7 source loopback 101

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.7.7, timeout is 2 seconds:
Packet sent with a source address of 172.16.7.7 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 11/75/266 ms
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

MPLS LDP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS LDP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R4, R5, and R6 for MPLS label exchange using an IETF standard protocol.
  - Authenticate LDP peering sessions using an MD5 hash and the password value of **CISCO**.
  - Ensure that LDP authentication is mandatory.
  - Ensure that all routers use Loopback0 as the LDP Router ID.
- Use only a single command on R4 to enable LDP on all OSPF-enabled interfaces.
- Use the physical interface's IP addresses for the LDP session between R4 and R6.

Configuration

The problem with the VRF Lite feature is its lack of scalability. When your VPNs span multiple routers, you need a separate link for every VPN between every pair of routers involved. Even worse, you cannot auto-provision these links by using any standard signaling protocol.

A technique that allows for overcoming this limitation emerged from tag-switching technology. Originally developed as a way to accelerate IP packet switching, tag-switching ultimately evolved into an advanced dynamic tunneling protocol. The name finally chosen for this technology was Multi-Protocol Label Switching, or MPLS.

MPLS tunnels, known as LSPs (Label Switching Paths), are similar to Frame-Relay or ATM PVCs. As you know, Frame-Relay packets are switched based on the DLCI

https://t.me/learningnets

value from the Layer 2 header. This DLCI value is purely local to every switch and could be rewritten every time the packet is switched out. The DLCI value could be thought of as a local label used to switch the incoming packet. The switching decision is made based on the local table that maps incoming DLCIs to outgoing DLCIs, effectively the Frame Relay routing table. However, remember that Frame Relay is a Layer 2 protocol, and the next header following the Frame Relay header normally belongs to the standard IP, or any other Layer 3 protocol.

MPLS employs a similar principle: A stack of labels is inserted between the original Layer 2 and the Layer 3 header. Per the RFC standard, every MPLS label starts with a 20-bit value. However, the actual tag is 32 bits after the addition of all special and reserved fields. MPLS labeling acts as a shim-layer, introducing an extra set of virtual-paths deployed over the Layer 2 topology being used in the network. If a packet carries the stack of MPLS labels, every router performs switching based on the topmost label found in the stack. This is done using the Label FIB or LFIB, which maps incoming labels to their outgoing equivalents. Thus, MPLS-labeled packets are switched based on a Label Lookup/Switch instead of a lookup in the IP table for the destination prefix. The first router on the edge of an MPLS cloud is known as a LER or Label Edge Router and is responsible for inserting (pushing) the initial label. The routers inside the MPLS cloud are known as LSRs or Label Switching Routers; they swap labels found in the packets (perform pop and push operations) and switch packets further. The last LER along the LSP is responsible for popping the label and switching the packets further using traditional prefix lookup mechanics.

The fundamental core of MPLS centers on how the labels are assigned. A tunnel path is established for every IGP prefix found in the network. This allows us to replace packet switching based on destination prefix lookup with switching based on label swapping. Although it does not provide any real performance improvement, it allows for other important features, such as MPLS VPNs, which we will discuss in a separate task. Now let’s see how MPLS routers exchange their label bindings.

In Frame Relay, to provision a PVC, you must manually program DLCI mappings along the path. However, MPLS is a dynamic technology and uses a special protocol called Label Distribution Protocol (LDP) to exchange label values. By default, every router will generate a local label for every prefix found in its routing table and advertise it via LDP to its neighbors. This is the label that adjacent routers must use when switching for a prefix via the local router. LDP works similar to distance-vector protocols; it broadcasts all local prefixes with their respective labels. As soon as a local router learns the labels used by its neighbors for the same prefixes, it will program the LFIB with respective label values: incoming label (which has been locally generated) replaced with an outgoing label (used by the neighbor routers).

LDP is a complex protocol defined in RFC3036. We will not discuss all of its

https://t.me/learningnets

functionality in detail. In short, LDP works as follows:

1. To enable MPLS switching on an interface and start LDP on the same interface, you must enter the interface-level command `mpls ip` . If you have too many interfaces to enable MPLS on, you may use MPLS LDP auto configuration, which is available when you run OSPF as your IGP protocol. Under the OSPF process, enter the command `mpls ldp autoconfig` to activate LDP/MPLS switching on all interfaces running OSPF. After MPLS has been enabled on an interface, LDP will attempt to discover valid neighbors on the interface. Initially, LDP sends discovery UDP packets to the multicast address of 224.0.0.2 on port 646. This multicast address corresponds to all routers address on the local segment.

2. Upon hearing from the other LDP routers, LDP learns their LDP Router IDs, which is by default the highest Loopback IP addresses. You may change the Router-ID by using the command `mpls ldp router-id <interface> force` . If for some reason the Loopback IP addresses are unreachable, a TCP connection will not be established. If you want LDP to establish a TCP connection using the physical interface IP address, use the interface-level command `mpls ldp discovery transport-address interface` .

3. Using the Router ID IP addresses as sources, two routers that heard from each other establish a TCP transport connection using the destination port of 646. This connection could be authenticated using an MD5 hash TCP option. The hashing key is defined per-neighbor by using the command `mpls ldp neighbor <IP> password <password>` . The IP address here is the neighbor's LDP Router ID. To make the use of passwords mandatory, you need the global command `mpls ldp password required` .

4. After the transport connection has been established, the routers exchange prefix and label bindings, resulting in LFIB population. The exchange is performed over the TCP connection established previously.

After the LFIB databases have been populated, label switching may occur. Notice that it is important for all routers within the MPLS domain to have the same prefixes in their routing tables; otherwise, the label bindings will not match. This results in the inability to use route summarization with MPLS and traditional IGPs.

The predecessor to LDP was a Cisco proprietary protocol known as Tag Switching Protocol, or TDP. It was the default label exchange protocol on Cisco routers until recent IOS releases. If for some reason you are required to use the legacy and proprietary protocol, you may do so using the command `mpls label protocol tdp` in

https://t.me/learningnets

the global configuration mode.

```
R4:
mpls ip
!
mpls ldp router-id Loopback0 force
!
interface GigabitEthernet1.146
 mpls ldp discovery transport-address interface
!
router ospf 1
 mpls ldp autoconfig
!
mpls ldp password required
mpls ldp neighbor 150.1.5.5 password CISCO
mpls ldp neighbor 150.1.6.6 password CISCO
R6:
mpls ip
!
mpls ldp router-id Loopback0 force
!
interface GigabitEthernet1.146
 mpls ldp discovery transport-address interface
 mpls ip
!
mpls ldp password required
mpls ldp neighbor 150.1.4.4 password CISCO
R5:

mpls ip
!
mpls ldp router-id Loopback0 force
!
interface GigabitEthernet1.45
 mpls ip
!
interface Tunnel0
 mpls ip
!
mpls ldp password required
mpls ldp neighbor 150.1.4.4 password CISCO
```

https://t.me/learningnets

Verification

Check the MPLS LDP peering sessions first. Notice the IP addresses used for the TCP peering sessions. R4 and R6 use physical interface IP addresses to accomplish this.

```
R4#show mpls ldp neighbor
    Peer LDP Ident: 150.1.6.6:0; Local LDP Ident 150.1.4.4:0
TCP connection: 155.1.146.6.46069 - 155.1.146.4.646
        State: Oper; Msgs sent/rcvd: 13/12; Downstream
        Up time: 00:00:58
        LDP discovery sources:
          GigabitEthernet1.146, Src IP addr: 155.1.146.6
        Addresses bound to peer LDP Ident:
          155.1.146.6     150.1.6.6
    Peer LDP Ident: 150.1.5.5:0; Local LDP Ident 150.1.4.4:0
TCP connection: 150.1.5.5.24868 - 150.1.4.4.646

        State: Oper; Msgs sent/rcvd: 13/13; Downstream
        Up time: 00:00:53
        LDP discovery sources:
          GigabitEthernet1.45, Src IP addr: 155.1.45.5
          Tunnel0, Src IP addr: 155.1.0.5
        Addresses bound to peer LDP Ident:
          155.1.5.5       155.1.45.5      155.1.58.5      169.254.100.5
          150.1.5.5       155.1.0.5
```

Confirm LDP authentication for the neighbors.

```
R4#show mpls ldp neighbor password
    Peer LDP Ident: 150.1.6.6:0; Local LDP Ident 150.1.4.4:0
        TCP connection: 155.1.146.6.23226 - 155.1.146.4.646 Password: required, neighbor, in use
        State: Oper; Msgs sent/rcvd: 13/13
    Peer LDP Ident: 150.1.5.5:0; Local LDP Ident 150.1.4.4:0
        TCP connection: 150.1.5.5.14779 - 150.1.4.4.646 Password: required, neighbor, in use

        State: Oper; Msgs sent/rcvd: 13/13
```

https://t.me/learningnets

Check the MPLS forwarding tables of all three routers. Notice the labels assigned to the Loopback0 interfaces. For example, R6 advertises label 21 for 150.1.5.5/32 and receives label 16 for this prefix from R4. Many prefixes have the **Pop Label** action as a result of implicit-null binding.

```
R6#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
19         Pop Label  150.1.4.4/32     0             Gi1.146    155.1.146.4
21         16         150.1.5.5/32     0             Gi1.146    155.1.146.4
27         Pop Label  155.1.0.0/24     0             Gi1.146    155.1.146.4
28         18         155.1.5.0/24     0             Gi1.146    155.1.146.4
29         Pop Label  155.1.45.0/24    0             Gi1.146    155.1.146.4
30         19         155.1.58.0/24    0             Gi1.146    155.1.146.4
22         Pop Label  169.254.100.0/24 0             Gi1.146    155.1.146.4
!
!R4#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  150.1.5.5/32     0             Gi1.45     155.1.45.5
17         Pop Label  150.1.6.6/32     0             Gi1.146    155.1.146.6
18         Pop Label  155.1.5.0/24     0             Gi1.45     155.1.45.5
19         Pop Label  155.1.58.0/24    0             Gi1.45     155.1.45.5
!
!R5#show mpls forwarding-table

Label      Label      or Tunnel Id     Switched      interface
21         Pop Label  150.1.4.4/32     0             Gi1.45     155.1.45.4
22         17         150.1.6.6/32     0             Gi1.45     155.1.45.4
23         Pop Label  155.1.146.0/24   0             Gi1.45     155.1.45.4
```

Do a traceroute from R6 to R5 and notice the MPLS label popping in the output.

```
R6#traceroute 150.1.5.5

Type escape sequence to abort.
Tracing the route to 150.1.5.5
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.146.4 [MPLS: Label 16
 Exp 0] 16 msec 7 msec 7 msec
  2 169.254.100.5 12 msec *  3 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

MPLS Label Filtering

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS Label Filtering**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R4, R5, and R6 so that the only labels advertised by LDP are the ones for Loopback0 interfaces of the mentioned routers.

Configuration

By default, LDP will generate and advertise labels for every prefix found in the local routing table. If you want to change this behavior and generate labels only for specific prefixes, you may use an access-list to select the prefixes eligible for label generation.

```
R4, R5, R6:

access-list 10 permit 150.1.0.0 0.0.255.255
!
no mpls ldp advertise-labels
mpls ldp advertise-labels for 10
```

Verification

Check the MPLS forwarding tables in R4, R5, and R6, and notice that only the Loopback0 prefixes now have labels assigned.

https://t.me/learningnets

```
R6#show mpls forwarding-table

Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
19         Pop Label  150.1.4.4/32     0             Gi1.146    155.1.146.4
21         16         150.1.5.5/32     0             Gi1.146    155.1.146.4

27         No Label   155.1.0.0/24     0             Gi1.146    155.1.146.4
28         No Label   155.1.5.0/24     0             Gi1.146    155.1.146.4
29         No Label   155.1.45.0/24    0             Gi1.146    155.1.146.4
30         No Label   155.1.58.0/24    0             Gi1.146    155.1.146.4
!
!R4#show mpls forwarding-table

Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  150.1.5.5/32     138           Gi1.45     155.1.45.5
17         Pop Label  150.1.6.6/32     0             Gi1.146    155.1.146.6
18         No Label   155.1.5.0/24     0             Gi1.45     155.1.45.5
19         No Label   155.1.58.0/24    0             Gi1.45     155.1.45.5
!
!R5#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
21         Pop Label  150.1.4.4/32     0             Gi1.45     155.1.45.4
22         17         150.1.6.6/32     0             Gi1.45     155.1.45.4

23         No Label   155.1.146.0/24   0             Gi1.45     155.1.45.4
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

MP-BGP VPNv4

> You must load the initial configuration files for the section, **MPLS MP BGP VPNv4**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a vrf named **VPN_A** on R5 and assign GigabitEthernet1.58 to it.
- Create a vrf named **VPN_B** on R5 and assign GigabitEthernet1.5 to it.
- Enable VPN route exchange between R5 and R6, using R4 as the BGP route-reflector.
  - Make sure that IPv4 peering sessions are not activated by default.
- By the end of this task, you should be able to ping the connected interfaces inside the VPNs.
  - R6 was pre-configured with the VRFs.

Configuration

MPLS technology is a perfect candidate for a dynamic tunneling solution to resolve the scalability issue associated with VRF Lite. The purpose of MPLS VPNs is to establish a full-mesh of dynamic MPLS LSRs between the PE (Provider Edge) routers and use those for tunneling VPN packets across the network core. To select the proper VRF instance on the endpoint PE router, an additional label is needed that selects the proper FIB entry associated with the target VRF. This requires two labels in the MPLS stack; one label (the topmost) is the transport label, which is being swapped along the entire path between the PEs, and the other label (innermost) is the VPN label, which is used to select the proper outgoing VRF CEF entry.

When a tunneling solution was found, a way to distribute VPN routes between the sites was needed. You cannot normally establish IGP protocol adjacencies across

https://t.me/learningnets

MPLS LSRs because they are unidirectional. And even if a bi-directional tunneling solution such as mGRE were in use, establishing hundreds of adjacencies for OSPF across a network core would not work well from a scaling perspective. Because of these factors, BGP has been chosen as a universal prefix redistribution protocol.

To support these new features, BGP functionality has been enhanced to handle the VRF specific routes. A new special MP-BGP (multiprotocol BGP) address family named **VPNv4** (VPN IPv4) has been added to BGP along with a new NLRI format. Every VPNv4 prefix has the RD associated with it and the corresponding MPLS label, in addition to the normal BGP attributes. This allows for transporting different VPN routes together and performing best-path selection independently for each different RD. The VPNv4 address-family capability is activated per-neighbor using the respective address-family configuration. By default, when you create a new BGP neighbor using the command `neighbor <IP> remote-as <NR>` , the default IPv4 unicast address-family is activated for this neighbor. If for some reason you don't want this behavior and only need the VPNv4 prefixes to be sent, you may disable the default behavior via the command `no bgp default ipv4-unicast` .

There are special limitations for iBGP peering sessions that you want to enable for VPNv4 prefix exchange. First, they must be sourced from a Loopback interface, and second, this interface must have a /32 mask (this is not a strict requirement on all platforms). This is needed because the BGP peering IP address is used as the NEXT_HOP for the locally originated VPNv4 prefixes. When the remote BGP router receives those prefixes, it performs a recursive routing lookup for the NEXT_HOP value and finds a label in the LFIB. This label is used as the transport label in the receiving router. Effectively, the NEXT_HOP is used to build the tunnel or the transport LSP between the PEs. The VPN label is generated by the BGP process on the advertising router and directly corresponds to the local VRF route. The /32 restriction is needed to guarantee that the transport LSP terminates on the particular PE router, and not some shared network segment.

To inject a particular VRF's routes into BGP, you must activate the respective address-family under the BGP process and enable route redistribution (such as static or connected). All the respective routes belonging to that particular VRF will be injected into the BGP table with their RDs and have their VPN labels generated. The import process is a bit more complicated and is based on the concept of Route Targets.

https://t.me/learningnets

A Route Target is a BGP extended community attribute. These BGP attributes are transitive and encoded as 64-bit values (as opposed to normal 32-bit communities). They are used for enhanced tagging of VPNv4 prefixes. The need for route-target arises from the fact that you cannot just use Route Distinguishers for prefix importing/exporting, because routes with the same RD may eventually belong to multiple VRFs, when you share their routes.

Here is how route-target-based import works. By default, all prefixes redistributed from a VRF into a BGP process are tagged with the extended community **`X:Y`** specified under the VRF configuration via the command **`route-target export X:Y .`** You may specify as many export commands as you want to tag prefixes with multiple attributes. On the receiving side, the VRF will import the BGP VPNv4 prefixes with the route-targets matching the local command **`route-target import X:Y .`** The import process is based entirely on the route-targets, not the RDs. If the imported routes used to have RDs different from the one used by the local VRF, they are naturalized by having the RD changed to the local value. Theoretically, you may assign a route-target to every VPN site, and specify fine-tune import policies, to select the remote site routes accepted locally. Finally, notice that the use of the command **`route-target both X:Y`** means import and export statements at the same time.

```
R4:
router bgp 100
 no bgp default ipv4-unicast
 neighbor 150.1.5.5 remote-as 100
 neighbor 150.1.5.5 update-source Loopback0
 neighbor 150.1.6.6 remote-as 100
 neighbor 150.1.6.6 update-source Loopback0
!
address-family vpnv4 unicast
 neighbor 150.1.5.5 activate
 neighbor 150.1.6.6 activate
 neighbor 150.1.5.5 send-community extended
 neighbor 150.1.6.6 send-community extended
 neighbor 150.1.5.5 route-reflector-client
 neighbor 150.1.6.6 route-reflector-client
R5:
ip vrf VPN_A
 rd 100:1
 route-target both 100:1
!
ip vrf VPN_B
 rd 100:2
 route-target both 100:2
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.58
 ip vrf forwarding VPN_A
 ip address 155.1.58.5 255.255.255.0
!
interface GigabitEthernet1.5
 ip vrf forwarding VPN_B
 ip address 155.1.5.5 255.255.255.0
R6:
ip vrf VPN_A
 rd 100:1
 route-target both 100:1
!
ip vrf VPN_B
 rd 100:2
 route-target both 100:2
R5 , R6:

router bgp 100
 no bgp default ipv4-unicast
 neighbor 150.1.4.4 remote-as 100
 neighbor 150.1.4.4 update-source Loopback0
!
address-family vpnv4 unicast
 neighbor 150.1.4.4 activate
 neighbor 150.1.4.4 send-community extended
!
 address-family ipv4 vrf VPN_A
  redistribute connected
  redistribute static
!
 address-family ipv4 vrf VPN_B
  redistribute connected
  redistribute static
```

Verification

Check the PE route and look for BGP-learned prefixes in every VRF's routing table.

```
R5#show ip route vrf VPN_A

Routing Table: VPN_A
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
```

https://t.me/learningnets

```
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      155.1.0.0/16 is variably subnetted, 3 subnets, 2 masks
C        155.1.58.0/24 is directly connected, GigabitEthernet1.58
L        155.1.58.5/32 is directly connected, GigabitEthernet1.58
B        155.1.67.0/24 [200/0] via 150.1.6.6, 00:00:07
B     192.168.7.0/24 [200/0] via 150.1.6.6, 00:00:07
!
!R5#show ip route vrf VPN_B

Routing Table: VPN_B
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      155.1.0.0/16 is variably subnetted, 3 subnets, 2 masks
C        155.1.5.0/24 is directly connected, GigabitEthernet1.5
L        155.1.5.5/32 is directly connected, GigabitEthernet1.5
B        155.1.76.0/24 [200/0] via 150.1.6.6, 00:01:18
      172.16.0.0/24 is subnetted, 1 subnetsB        172.16.7.0 [200/0] via 150.1.6.6, 00:01:18
```

Now check the route-reflector BGP table for VPNv4 prefixes. Notice how prefixes are grouped based on the RD.

```
R4#show bgp vpnv4 unicast all

BGP table version is 7, local router ID is 150.1.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
```

https://t.me/learningnets

```
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path Route Distinguisher: 100:1
 *>i 155.1.58.0/24    150.1.5.5               0    100      0 ?
 *>i 155.1.67.0/24    150.1.6.6               0    100      0 ?
 *>i 192.168.7.0      150.1.6.6               0    100      0 ? Route Distinguisher: 100:2

 *>i 155.1.5.0/24     150.1.5.5               0    100      0 ?
 *>i 155.1.76.0/24    150.1.6.6               0    100      0 ?
 *>i 172.16.7.0/24    150.1.6.6               0    100      0 ?
```

Check the route-target values associated with the various prefixes in R4's VPNv4 BGP table.

```
R4#show bgp vpnv4 unicast all 155.1.76.0 255.255.255.0
BGP routing table entry for 100:2:155.1.76.0/24, version 6
Paths: (1 available, best #1, no table)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  Local, (Received from a RR-client)
    150.1.6.6 (metric 2) (via default) from 150.1.6.6 (150.1.6.6)
      Origin incomplete, metric 0, localpref 100, valid, internal, best Extended Community: RT:100:2
      mpls labels in/out nolabel/25
      rx pathid: 0, tx pathid: 0x0
!
!R4#show bgp vpnv4 unicast all 155.1.67.0 255.255.255.0
BGP routing table entry for 100:1:155.1.67.0/24, version 3
Paths: (1 available, best #1, no table)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  Local, (Received from a RR-client)
    150.1.6.6 (metric 2) (via default) from 150.1.6.6 (150.1.6.6)
      Origin incomplete, metric 0, localpref 100, valid, internal, best Extended Community: RT:100:1

      mpls labels in/out nolabel/23
      rx pathid: 0, tx pathid: 0x0
```

Check the end-to-end connectivity across each VPN using ping and traceroute. Notice the label stack used to tunnel the VPN packets.

https://t.me/learningnets

```
R5#ping vrf VPN_A 155.1.67.6


Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.67.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/9/20 ms
!
!R5#traceroute vrf VPN_A 155.1.67.6
Type escape sequence to abort.
Tracing the route to 155.1.67.6
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.45.4 [MPLS: Labels 17/23
 Exp 0] 18 msec 8 msec 23 msec
  2 155.1.67.6 21 msec *  5 msec
!
!R5#ping vrf VPN_B 155.1.76.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.76.6, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/6/10 ms
!
!R5#traceroute vrf VPN_B 155.1.76.7

Type escape sequence to abort.
Tracing the route to 155.1.76.7
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.45.4 [MPLS: Labels 17/25
 Exp 0] 62 msec 8 msec 9 msec
  2 155.1.76.6 14 msec 6 msec 7 msec
  3 155.1.76.7 9 msec *  13 msec
!
!R5#traceroute vrf VPN_B 155.1.76.6

Type escape sequence to abort.
Tracing the route to 155.1.76.6
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.45.4 [MPLS: Labels 17/25
 Exp 0] 59 msec 19 msec 19 msec
  2 155.1.76.6 12 msec *  4 msec
```

Check the MPLS labels (VPN labels) assigned to the VPN prefixes at the PE routers. There are two labels in the stack; one is the VPN label and the other is the transport label. You may find the VPN label in the BGP table, and the transport label can be found by looking up the VPNv4 BGP next-hop in the MPLS forwarding table.

```
R5#show ip bgp vpnv4 vrf VPN_A 155.1.67.0
BGP routing table entry for 100:1:155.1.67.0/24, version 8
Paths: (1 available, best #1, table VPN_A)
```

https://t.me/learningnets

```
  Not advertised to any peer
  Refresh Epoch 1
  Local
    150.1.6.6 (metric 3) (via default) from 150.1.4.4 (150.1.4.4)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      Extended Community: RT:100:1
      Originator: 150.1.6.6, Cluster list: 150.1.4.4 mpls labels in/out nolabel/23
      rx pathid: 0, tx pathid: 0x0
!
!R5#show mpls forwarding-table 150.1.6.6
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface               17
19         150.1.6.6/32     0
             Gi1.45     155.1.45.4
           No Label   150.1.6.6/32     0             Gi1.100    169.254.100.4
```

To tie it all together, we look at CEF. Because these routes are in VRFs, the show cef commands are used for each VRF. Looking at the different tables (BGP Table, RIB, LIB) is good for troubleshooting, but to see what is exactly programmed into the hardware for forwarding, we have to look at CEF.

```
R5#show ip cef vrf VPN_A 155.1.67.0 detail
 155.1.67.0/24, epoch 0, flags rib defined all labels recursive via 150.1.6.6 label 23
 nexthop 155.1.45.4 GigabitEthernet1.45 label 19
```

CEF shows us both labels that have to be used when encapsulating traffic toward destination 155.1.67.0 in **VPN_A**. The second label, 19, is the IGP label or transport label. This is the label that is switched throughout the MPLS network from PE to PE. Because the next-hop for 155.1.67.0 is 150.1.6.6, R5 needs a label for 150.1.6.6. This label can be seen with `show mpls forwarding-table 150.1.6.6` . The first label, 23, is the VPN label. This is what R6 advertised in its MP-BGP VPNv4 update, and it can be seen with `show bgp vpnv4 unicast vrf VPN_A 155.1.67.0` . Putting both of these labels together in the forwarding table is the basis of MPLS VPN.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

MP-BGP Prefix Filtering

> You must load the initial configuration files for the section, **MPLS MP BGP Prefix Filtering**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a new Loopback101 interface in R5's VRF **VPN_A** with the IP address **172.16.5.5/24**.
- Create a new Loopback102 interface in R6's VRF **VPN_B** with the IP address **192.168.6.6/24**.
- Configure the network to provide bi-directional connectivity between the two new subnets.
- Make sure that R6's **VPN_A** does not see the prefix 172.16.5.0/24 and R5's **VPN_B** does not see the prefix 192.168.6.0/24.
  - Use export/import maps for this task.

Configuration

As we found in the previous task, route-target tagging applies to all routes injected from VRF into a BGP process; sometimes we need more granular control over route tagging. To accomplish this, Cisco IOS has a feature known as export/import maps. They are configured under the VRF configuration context using the command `[import|export] map <ROUTE_MAP_NAME>` . The special export route-map associated with the VRF could match the prefixes based on the prefix-lists, access-lists, or extended-communities. All routes not permitted in an export route-map are not exported into the BGP process. The export route-map may also be used to set the extended-community attribute selectively, using the command `set extcommunity rt` . This allows for selective tagging of VPN routes.

The import map is used less often than the export map, but still has some good use

https://t.me/learningnets

cases. First, it allows controlling all routes imported into VRF from BGP based on prefix-lists, access-lists, or extended/standard communities (or any other applicable BGP attribute). This might be helpful when, for example, you want to import all other sites' routes with the exception of the default route. Notice that by default, all prefixes not permitted with the import-map are implicitly denied and not imported.

```
R5:
interface Loopback101
 ip vrf forwarding VPN_A
 ip address 172.16.5.5 255.255.255.0
!
ip prefix-list LO101 permit 172.16.5.0/24
!
route-map VPN_A_EXPORT permit 10
 match ip address prefix-list LO101
 set extcommunity rt 100:55
!
route-map VPN_A_EXPORT permit 20
 set extcommunity rt 100:1
!
ip vrf VPN_A
 export map VPN_A_EXPORT
 route-target import 100:66
R6:

interface Loopback102
 ip vrf forwarding VPN_B
 ip address 192.168.6.6 255.255.255.0
!
ip prefix-list LO102 permit 192.168.6.0/24
!
route-map VPN_B_EXPORT permit 10
 match ip address prefix-list LO102
 set extcommunity rt 100:66
!
route-map VPN_B_EXPORT permit 20
 set extcommunity rt 100:2
!
ip vrf VPN_B
 export map VPN_B_EXPORT
 route-target import 100:55
```

https://t.me/learningnets

Verification

Determine whether the prefixes 172.16.5.0/24 and 192.168.6.0/24 appear in R6's and R5's **VPN_A** and **VPN_B** routing tables.

```
R6#show ip route vrf VPN_B 172.16.5.0
Routing entry for 172.16.5.0/24
  Known via "bgp 100", distance 200, metric 0, type internal
  Last update from 150.1.5.5 00:02:05 ago
  Routing Descriptor Blocks: * 150.1.5.5 (default), from 150.1.4.4, 00:02:05 ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: 18
      MPLS Flags: MPLS Required
!
!R6#show ip route vrf VPN_A 172.16.5.0
% Network not in table
```

Perform the same checks on R5's VRFs for the prefix 192.168.6.0/24.

```
R5#show ip route vrf VPN_A 192.168.6.0
  Known via "bgp 100", distance 200, metric 0, type internal
  Last update from 150.1.6.6 00:02:46 ago
  Routing Descriptor Blocks: * 150.1.6.6 (default), from 150.1.4.4, 00:02:46 ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: 22
      MPLS Flags: MPLS Required
!
!R5#show ip route vrf VPN_B 192.168.6.0
% Network not in table
```

Check the route-target values associated with the new prefixes.

```
R4#show ip bgp vpnv4 rd 100:1 172.16.5.0
BGP routing table entry for 100:1:172.16.5.0/24, version 11
Paths: (1 available, best #1, no table)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  Local, (Received from a RR-client)
```

https://t.me/learningnets

```
    150.1.5.5 (metric 2) (via default) from 150.1.5.5 (150.1.5.5)
      Origin incomplete, metric 0, localpref 100, valid, internal, best Extended Community: RT:100:55
      mpls labels in/out nolabel/18
      rx pathid: 0, tx pathid: 0x0
!
!R4#show ip bgp vpnv4 rd 100:2 192.168.6.0
BBGP routing table entry for 100:2:192.168.6.0/24, version 10
PPaths: (1 available, best #1, no table)
   Advertised to update-groups:
      1
   Refresh Epoch 1
   Local, (Received from a RR-client)
     150.1.6.6 (metric 2) (via default) from 150.1.6.6 (150.1.6.6)
       Origin incomplete, metric 0, localpref 100, valid, internal, best Extended Community: RT:100:66

       mpls labels in/out nolabel/22
       rx pathid: 0, tx pathid: 0x0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

PE-CE Routing with RIP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS PE CE Routing with RIP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Use RIP as the PE-CE routing protocol for VRF named **VPN_B**, between R5 and R8, R6 and R7.
- Configure R5 so that the GigabitEthernet1.58 link is assigned to VRF **VPN_B**.
  - R8 should run RIP with R5, and R8 should advertise all of its interfaces into RIP.
- Ensure IP connectivity between VRF **VPN_B** prefixes.
  - Preserve the RIP metric values learned from CE routers.

Configuration

Even though MP-BGP is the carrier routing protocol for MPLS VPNs, a wide variety of PE-CE routing protocols could be used. The PE-CE routing protocols run independently at every site, relying on MP-BGP to exchange their routing information. The route exchange is performed via prefix redistribution between the MP-BGP process and the respective PE-CE routing protocol. Several features have been added to MP-BGP to preserve some valuable routing protocol information that otherwise might be lost during redistribution. These features vary from one PE-CE routing protocol to another.

We start with the simplest of all PE-CE protocols, RIPv2. With RIPv2 you may enable the respective address family under the process configuration mode using the command `address-family ipv4 vrf <VRF_NAME>` . You may then specify the network commands as usual. Notice that the timer settings, version, and auto-summary

https://t.me/learningnets

settings are global for all VRFs with this approach. When you need to redistribute the MP-BGP routes into RIP, use the command **`redistribute bgp <N> metric [X|transparent]`** under the respective address family. Here, N is the BGP process number (AS#) and X is the metric assigned to the RIP routes. If you are using the keyword **`transparent`**, the RIP metrics will be recovered from the BGP MED attribute, which in turn is copied from RIP metrics learned at the remote site. This allows for transparent preservation of RIPv2 metric values across the VPN and better path selection in case of backdoor links.

Naturally, if you want to inject the respective VRF's RIP routes into BGP, you simply redistribute them under the corresponding address-family. The MED value for the new VPNv4 prefixes will be initialized from the RIP metrics of the redistributed routes.

```
R5:
interface GigabitEthernet1.58
 ip vrf forwarding VPN_B
 ip address 155.1.58.5 255.255.255.0
!
router rip
 version 2
 address-family ipv4 vrf VPN_B
  redistribute bgp 100 metric transparent
  network 155.1.0.0
!
router bgp 100
 address-family ipv4 vrf VPN_B
  redistribute rip
R8:
router rip
 version 2
 network 150.1.0.0
 network 155.1.0.0
R6:
router rip
 version 2
 address-family ipv4 vrf VPN_B
  redistribute bgp 100 metric transparent
  network 155.1.0.0
  network 192.168.6.0
!
router bgp 100
 address-family ipv4 vrf VPN_B
  redistribute rip
R7:
```

https://t.me/learningnets

```
router rip
 version 2
 address-family ipv4 vrf VPN_B
  network 155.1.0.0
  network 192.168.7.0
```

Verification

Check the VPN_A routes on R7 and make sure you see the RIP prefixes advertised by R8. Note that although R7 is not running BGP, it still has its interface to the PE in a VRF. There is no issue with this configuration, but remember that the commands must include the VRF. On the other CE, R8, the interface toward the PE is in the global table, so the show commands do not include the VRF.

```
R7#show ip route vrf VPN_B rip
Routing Table: VPN_B
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

R     150.1.0.0/16 [120/2] via 155.1.76.6, 00:00:22, GigabitEthernet1.76
      155.1.0.0/16 is variably subnetted, 6 subnets, 2 masks
R        155.1.5.0/24 [120/1] via 155.1.76.6, 00:00:22, GigabitEthernet1.76
R        155.1.8.0/24 [120/2] via 155.1.76.6, 00:00:22, GigabitEthernet1.76
R        155.1.58.0/24 [120/1] via 155.1.76.6, 00:00:22, GigabitEthernet1.76
R        155.1.108.0/24 [120/2] via 155.1.76.6, 00:00:22, GigabitEthernet1.76

      172.16.0.0/24 is subnetted, 1 subnets
R        172.16.5.0 [120/1] via 155.1.76.6, 00:00:22, GigabitEthernet1.76
R     192.168.6.0/24 [120/1] via 155.1.76.6, 00:00:22, GigabitEthernet1.76
```

https://t.me/learningnets

Check the routing table for VPN_B on R8 and look for prefixes advertised by R7. Again, note that this time we are not using VRF commands as R8's connection to its PE is in the global table.

```
R8#show ip route rip

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      155.1.0.0/16 is variably subnetted, 8 subnets, 2 masks
R        155.1.5.0/24 [120/1] via 155.1.58.5, 00:00:07, GigabitEthernet1.58
R        155.1.76.0/24 [120/1] via 155.1.58.5, 00:00:07, GigabitEthernet1.58
R     192.168.7.0/24 [120/2] via 155.1.58.5, 00:00:07, GigabitEthernet1.58
```

Look at R6's BGP table to see the RIP prefixes being carried in BGP updates. Notice the metric field value.

```
R6#show bgp vpnv4 unicast vrf VPN_B

BGP table version is 27, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 100:2 (default for vrf VPN_B)
 *>i 150.1.0.0        150.1.5.5                1    100      0 ?
 *>i 155.1.5.0/24     150.1.5.5                0    100      0 ?
 *>i 155.1.8.0/24     150.1.5.5                1    100      0 ?
 *>i 155.1.58.0/24    150.1.5.5                0    100      0 ?
 *>  155.1.76.0/24    0.0.0.0                  0         32768 ?
 *>i 155.1.108.0/24   150.1.5.5                1    100      0 ?
 *>i 172.16.5.0/24    150.1.5.5                0    100      0 ?
```

https://t.me/learningnets

```
 *>  192.168.6.0      0.0.0.0                  0         32768 ?
 *>  192.168.7.0      155.1.76.7               1         32768 ?
```

Test end-to-end connectivity across VPN_B.

```
R7#traceroute vrf VPN_B 155.1.8.8 source loopback102

Type escape sequence to abort.
Tracing the route to 155.1.8.8
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.76.6 14 msec 11 msec 6 msec
  2 155.1.146.4 [MPLS: Labels 16/24 Exp 0] 10 msec 13 msec 38 msec
  3 155.1.58.5 [MPLS: Label 24 Exp 0] 28 msec 41 msec 8 msec 4 155.1.58.8 75 msec *  10 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

PE-CE Routing with OSPF

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS PE CE Routing with OSPF**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Use OSPF as the PE-CE routing protocol for VRF named **VPN_A**, between R5 and R8, R6 and R7.
  - Configure PE-CE OSPF adjacency in area 1.
- Use the same OSPF process ID on R5 and R6, but ensure that R7 and R8 can reach each other.
  - CE routers should see OSPF routes as external.
- Configure R5 so that the GigabitEthernet1.58 link is assigned to VRF **VPN_A**.
- Configure Loopback100 interface on R8 with the IP address 172.16.8.8/24, and make sure R7 sees only a /16 summary for this prefix.

Configuration

OSPF is a link-state routing protocol, which relies on building a network topology table prior to calculating the optimal routes. However, as we remember, MPLS VPNs rely on using MP-BGP to transport the routing information across the MPLS backbone. This means that no OSPF adjacencies can be established across the core. Note that OSPF uses distance-vector-like updates sent across Area 0 to redistribute routing information between different areas. This allows for interpreting the MP-BGP cloud as one “super area 0” that is used to link all OSPF areas at different sites. This special “virtual area” is called the OSPF super-backbone, and it is emulated by passing OSPF VRF routing information in MP-BGP updates. The use of the super-backbone allows us to avoid using area 0 at all, because the super-

https://t.me/learningnets

backbone performs the same function. Therefore, we can have non-zero OSPF areas at different VPN sites connecting via the MP-BGP mesh without the need for an area 0 at any site. However, it's perfectly okay to have area 0 at different sites along with non-zero areas attached to the super-backbone as well. The main design principle that should be followed is that all areas are connected to the super-backbone in a loopless start-like manner.

We briefly discussed BGP extended communities before. With OSPF, special extended communities are used to propagate additional OSPF prefix information. All OSPF routes redistributed into MP-BGP are treated pretty much like Type 3 summary LSAs, because they enter the super-backbone from other areas. When injected into BGP, OSPF prefixes have two extended-community attributes attached to them. One of the attributes is known as domain-id, which is equal to the OSPF process numbers on the local router OR explicitly configured using the command `domain-id` under the OSPF process. The purpose of this attribute is to identify OSPF processes belonging to different VPNs. It is assumed that you configured all OSPF processes within the same VPN using the same domain-id (such as the same process number). If for some reason you exchange routes between two different VPNs using different domain-ids, the OSPF process will interpret all such prefixes as if they are Type-5 External LSAs, effectively external routes.

The other extended community attribute is known as the OSPF route-type, which has three significant fields: source area, route-type, and option. They are usually depicted as triple X:Y:Z. Here, Y=2 for intra-area learned prefix, Y=3 for inter-area routes, Y=5 for external prefixes, and Y=7 for NSSA routes. The last value, Z, is the metric type for Y=5 or 7—it's either 1 for E1 or 2 for E2. Notice that all routes redistributed from BGP into OSPF appear like *inter-area* routes even if they belong to the same area number at different sites. This effect occurs because the LSAs cross the super-backbone and essentially are inter-area routes.

The last BGP attribute used to carry OSPF information is MED or metric, which copies the original route's metric from the routing table. The route-type attribute is needed to allow for proper OSPF best-path calculation when routes are inserted into the OSPF database based on redistribution from BGP. Notice that the routes traveling the MP-BGP cloud do not increment their metric unless you manually change the MED attribute for incoming BGP prefixes. This might be needed sometimes for proper OSPF best-path selection.

All the above mechanics are implemented automatically after you configure route redistribution between a VRF process and BGP. Notice that you create a separate OSPF process for every VRF using the command `router ospf X vrf NAME` , unlike RIP, where the whole process is shared among VRFs.

The fact that there is an additional backbone area could introduce the possibility of

https://t.me/learningnets

routing loops if for some reason the VPN sites are not connected in a star-like manner. To reduce this risk, OSPF implements some basic loop prevention rules.

First, all summary LSAs generated from the routes redistributed from BGP have a special “Down” bit set in the LSA headers. If a router receives a summary-LSA with the down bit set on an interface that belongs to a VRF, it simply drops this LSA. This is to prevent the case of routing loops for multi-homed sites, when a summary LSA is flooded across the CE site and delivered back to another PE. However, this feature may have an undesirable effect when you have a CE router configured with multiple VRFs. In this case, you may want to enter the OSPF process command **`capability vrf-lite`** on the CE router. This will disable the default loop-prevention capability. Be advised that some IOS versions do not support this feature (such as older IOSs or some Catalyst IOS revisions). If you have such a router configured for multi-VRF and experience route black holing, configure the PE routers with different domain-IDs; this will force all redistributed routes to become external and bypass the down-bit check. Note that in newer IOS versions, such as the one used in this example, the down bit is also included in Type-5 LSAs. **`capability vrf-lite`** would be needed in this case as well, and it is the reason why it is used in the solution.

The other feature is based on the route tagging. All routes redistributed via a particular PE will carry the OSPF route tag with the BGP AS number encoded inside. The receiving router that has VRFs enabled will compare the AS number in the tag with the local BGP AS number. If they match, this could mean the LSA has looped back to another PE connecting the same site to the MPLS backbone. If for some reason you encounter this issue and need to get rid of it, simply apply a proper tag to the redistributed routes—for example, using the OSPF command **`redistribute bgp N subnets tag Y .`**

```
R5:
interface GigabitEthernet1.58
 ip vrf forwarding VPN_A
 ip address 155.1.58.5 255.255.255.0
!
router ospf 100 vrf VPN_A
 domain-id 0.0.0.5
 redistribute bgp 100 subnets
 network 0.0.0.0 255.255.255.255 area 1
!
router bgp 100
 address-family ipv4 vrf VPN_A
  redistribute ospf 100 vrf VPN_A
R6:
router ospf 100 vrf VPN_A
 domain-id 0.0.0.6
```

https://t.me/learningnets

```
 redistribute bgp 100 subnets
 network 0.0.0.0 255.255.255.255 area 1
 summary-address 172.16.0.0 255.255.0.0
!
router bgp 100
 address-family ipv4 vrf VPN_A
  redistribute ospf 100 vrf VPN_A
R7:
router ospf 1 vrf VPN_A
 capability vrf-lite
 network 0.0.0.0 255.255.255.255 area 1
R8:

router ospf 1
 network 0.0.0.0 255.255.255.255 area 1
!
interface Loopback100
 ip address 172.16.8.8 255.255.255.0
```

Verification

Check the routing tables of R8 and R7 VRF_A. Notice that R7 sees the /16 summary for the 172.16.8.8/24 prefix (actually, it was /32 because of the OSPF loopback network type).

```
R8#show ip route ospf
      155.1.0.0/16 is variably subnetted, 7 subnets, 2 masks
O E2     155.1.67.0/24 [110/1] via 155.1.58.5, 00:02:34, GigabitEthernet1.58
      172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
O E2     172.16.0.0/16 [110/2] via 155.1.58.5, 00:02:29, GigabitEthernet1.58
O        172.16.5.5/32 [110/2] via 155.1.58.5, 00:02:34, GigabitEthernet1.58
O E2     172.16.7.7/32 [110/2] via 155.1.58.5, 00:02:34, GigabitEthernet1.58
O E2  192.168.6.0/24 [110/1] via 155.1.58.5, 00:02:34, GigabitEthernet1.58
O E2  192.168.7.0/24 [110/1] via 155.1.58.5, 00:02:34, GigabitEthernet1.58
!
!R7#show ip route vrf VPN_A ospf
Routing Table: VPN_A
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
```

https://t.me/learningnets

```
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 1 subnets
O E2     150.1.8.8 [110/2] via 155.1.67.6, 00:00:17, GigabitEthernet1.67
      155.1.0.0/16 is variably subnetted, 5 subnets, 2 masks
O E2     155.1.8.0/24 [110/2] via 155.1.67.6, 00:00:17, GigabitEthernet1.67
O E2     155.1.58.0/24 [110/1] via 155.1.67.6, 00:00:17, GigabitEthernet1.67
O E2     155.1.108.0/24 [110/2] via 155.1.67.6, 00:00:17, GigabitEthernet1.67
      172.16.0.0/16 is variably subnetted, 3 subnets, 3 masks
O E2     172.16.0.0/16 [110/2] via 155.1.67.6, 00:00:17, GigabitEthernet1.67
```

Verify that the OSPF processes on R5 and R6 are connected to the OSPF super-backbone. Notice that both routers are ASBRs because they redistribute BGP routes.

```
R5#show ip ospf 100
 Routing Process "ospf 100" with ID 172.16.5.5
   Domain ID type 0x0005, value 0.0.0.5
 Start time: 6d12h, Time elapsed: 01:03:25.902
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101) Connected to MPLS VPN Superbackbone, VRF VPN_A
 Event-log disabled It is an area border and autonomous system boundary router
 Redistributing External Routes from,
    bgp 100, includes subnets in redistribution
 Router is not originating router-LSAs with maximum metric
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 5. Checksum Sum 0x0120A4
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
```

https://t.me/learningnets

```
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 0
 IETF NSF helper support enabled
 Cisco NSF helper support enabled
 Reference bandwidth unit is 100 mbps
    Area 1
        Number of interfaces in this area is 2 (1 loopback)
        Area has no authentication
        SPF algorithm last executed 00:45:14.365 ago
        SPF algorithm executed 3 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x00F9FF
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
!
!R6#show ip ospf 100
 Routing Process "ospf 100" with ID 155.1.67.6
   Domain ID type 0x0005, value 0.0.0.6
 Start time: 6d15h, Time elapsed: 01:04:45.099
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101) Connected to MPLS VPN Superbackbone, VRF VPN_A
 Event-log disabled It is an area border and autonomous system boundary router

 Redistributing External Routes from,
    bgp 100, includes subnets in redistribution
 Router is not originating router-LSAs with maximum metric
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 5. Checksum Sum 0x0256E3
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
```

https://t.me/learningnets

```
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 0
 IETF NSF helper support enabled
 Cisco NSF helper support enabled
 Reference bandwidth unit is 100 mbps
    Area 1
        Number of interfaces in this area is 1
        Area has no authentication
        SPF algorithm last executed 00:02:41.782 ago
        SPF algorithm executed 6 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x01BC85
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

Check the extended community attributes associated with the OSPF prefixes carried from R5 to R6. Take any prefix, such as 172.16.8.8. Pay attention to the Domain ID encoded in 0x000000**05**0200 as 0x05 and the route-type attribute value of 0.0.0.1:2:0, which stands for intra-area route received from area 1.

```
R6#show bgp vpnv4 unicast vrf VPN_A 172.16.8.8
BGP routing table entry for 100:1:172.16.8.8/32, version 46
Paths: (1 available, best #1, table VPN_A)
  Not advertised to any peer
  Refresh Epoch 2
  Local
    150.1.5.5 (metric 3) (via default) from 150.1.4.4 (150.1.4.4)
      Origin incomplete, metric 2, localpref 100, valid, internal, best
      Extended Community: RT:100:1 OSPF DOMAIN ID:0x0005:0x000000050200
         OSPF RT:0.0.0.1:2:0
 OSPF ROUTER ID:172.16.5.5:0
      Originator: 150.1.5.5, Cluster list: 150.1.4.4
      mpls labels in/out nolabel/19
      rx pathid: 0, tx pathid: 0x0
```

Now you may want to check the OSPF database on R7 to see what type of LSA it has for the subnet 172.16.8.0/24.

```
R7#show ip ospf database external 172.16.8.8


            OSPF Router with ID (150.1.7.7) (Process ID 1)
```

https://t.me/learningnets

```
!
!R7#show ip ospf database external 172.16.0.0

            OSPF Router with ID (172.16.7.7) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 984
  Options: (No TOS-capability, DC, Upward) LS Type: AS External Link
  Link State ID: 172.16.0.0 (External Network Number )
  Advertising Router: 155.1.67.6
  LS Seq Number: 80000002
  Checksum: 0x30DE
  Length: 36 Network Mask: /16

        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 2
        Forward Address: 0.0.0.0
        External Route Tag: 0
```

Note that in this version of IOS, currently IOS-XE 3.11 Version 15.4(1)S1, the Type-5 LSAs contain an upward/downward bit as the RFC dictates. Previously this bit was not included in Type-5 LSAs, only in Type-3s. However, with this version of code, Cisco is including this bit. Because of this, R7 had to include the **`capability vrf-lite`**. If we did not include the **`capability vrf-lite`** on R7, all LSAs marked with the down bit would not make it to the routing table. For example, look at 150.1.8.8 external LSA on R7.

```
R7#show ip ospf database external 150.1.8.8

            OSPF Router with ID (172.16.7.7) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1343  Options: (No TOS-capability, DC, Downward
)
  LS Type: AS External Link
  Link State ID: 150.1.8.8 (External Network Number )
  Advertising Router: 155.1.67.6
  LS Seq Number: 80000002
  Checksum: 0x1F4F
  Length: 36
```

https://t.me/learningnets

```
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 2
        Forward Address: 0.0.0.0
        External Route Tag: 3489661028
```

Note that it contains the down bit. As previously stated, in the previous IOS version the down bit was only included on Type-3 LSAs. On newer versions of the code, such as the one we are running, Cisco is including the down bit on Type-5 LSAs. Verify end-to-end connectivity across the VPN.

```
R7#traceroute vrf VPN_A 172.16.8.8

Type escape sequence to abort.
Tracing the route to 172.16.8.8
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.67.6 51 msec 4 msec 3 msec
  2 155.1.146.4 [MPLS: Labels 16/19 Exp 0] 21 msec 18 msec 33 msec
  3 155.1.58.5 [MPLS: Label 19 Exp 0] 10 msec 17 msec 22 msec 4 155.1.58.8 21 msec *  27 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

OSPF Sham-Link

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS OSPF Sham Link**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 and R6, the PE routers, to run OSPF in VRF **VPN_A** with R7 and R8, in area 1.
- Provision a back-door link between R7 and R8 as follows:
  - Use interface GigabitEthernet1.78 with a dot1q tag of 78.
  - Use IP addressing in the format of **155.1.78.Y/24**, where *Y* is the router number.
- Ensure that R7 and R8 prefer the path across the MPLS core, instead of the backdoor link.

Configuration

As we learned in the previous scenario, OSPF prefixes are transported via the MPLS VPN core using MP-BGP and interpreted either as type-3 summary LSAs or type-5 external LSAs on the receiving end. This may pose some difficulties in the event that there is a backdoor link connecting two VPN sites directly. Many times the backdoor link is supposed to be used as a backup path (it's a slow leased line) and the MPLS VPN cloud should be used as the primary way between the two sites. However, if the link is in the same area as the PE/CE routers, the PE routers will prefer the path across the back-door link, because OSPF treats all paths across it as intra-area, and the prefixes received via MP-BGP are interpreted as inter-area.

This is a fundamental issue that requires the MPLS core to be a part of the same OSPF area to resolve the conflict. The solution is called an OSPF sham-link, which

https://t.me/learningnets

is a special tunnel similar to a virtual-link connecting two PE routers and configured in the same area as the PE routers. This link is used to establish an OSPF adjacency and for the exchange of LSAs. The LSAs are then loaded in the OSPF database and the sham-link is used for intra-area path computations. However, when the routes are being installed in the respective VRF RIB, the forwarding information is based on looking up the MP-BGP learned routes based on exact prefix-match. The corresponding VPN and transport labels are then used for actual packet forwarding across the MPLS core. Therefore, the information loaded across the sham-link is used only for SPF calculations and best-path selection; the actual forwarding is being done based on the information learned via MP-BGP.

Configuring sham-links is a bit different from configuring virtual-links. First and foremost, sham-links are sourced off actual interfaces configured in the respective VRF. Commonly, these are Loopback interfaces used as endpoints for the sham-link tunnel. Notice that the IP addresses for these interfaces should be advertised into the VRF routing table by means other than OSPF, most commonly via BGP. After you have the endpoints reachable across the MPLS VPN core, you may configure the sham-link using the command **`area 1 sham-link <SRC><DST> cost X .`** Here,<SRC> and <DST> are the IP addresses for the sham-link source and destination. The cost is the OSPF metric value associated with traversing the MPLS core. If the endpoints are reachable, OSPF will establish an adjacency on the link treating it as a point-to-point connection and re-calculate the shortest paths.

In the solution below, we change the OSPF network command to cover only the interfaces that should establish OSPF adjacencies. The sham-link's endpoints should not be advertised into OSPF. Also note that the OSPF summary command we configured in the previous task no longer has any effect. R7 and R8 see each other's routes as intra-area even though they are sent across the MPLS core network.

```
R5:
router ospf 100 vrf VPN_A
 area 1 sham-link 150.1.55.55 150.1.66.66 cost 1
 no network 0.0.0.0 255.255.255.255 area 1
 network 155.1.58.5 0.0.0.0 area 1
!
interface Loopback 200
 ip vrf forwarding VPN_A
 ip address 150.1.55.55 255.255.255.255
!
router bgp 100
 address-family ipv4 vrf VPN_A
  network 150.1.55.55 mask 255.255.255.255
R6:
```

https://t.me/learningnets

```
router ospf 100 vrf VPN_A
 area 1 sham-link 150.1.66.66 150.1.55.55 cost 1
 no network 0.0.0.0 255.255.255.255 area 1
 network 155.1.67.6 0.0.0.0 area 1
!
interface Loopback 200
 ip vrf forwarding VPN_A
 ip address 150.1.66.66 255.255.255.255
!
router bgp 100
 address-family ipv4 vrf VPN_A
  network 150.1.66.66 mask 255.255.255.255
R7:
interface GigabitEthernet1.78
 encapsulation dot1q 78
 ip address 155.1.78.7 255.255.255.0
 ip ospf cost 9999
!
router ospf 1
 network 0.0.0.0 255.255.255.255 area 1
R8:

interface GigabitEthernet1.78
 encapsulation dot1q 78
 ip address 155.1.78.8 255.255.255.0
 ip ospf cost 9999
!
router ospf 1
 network 0.0.0.0 255.255.255.255 area 1
```

Verification

Check the sham-link status.

```
R5#show ip ospf sham-links
Sham Link OSPF_SL0 to address 150.1.66.66 is up
Area 1 source address 150.1.55.55

  Run as demand circuit
  DoNotAge LSA allowed. Cost of using 1 State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40,
    Hello due in 00:00:01
    Adjacency State FULL (Hello suppressed)
    Index 2/2, retransmission queue length 0, number of retransmission 0
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
```

https://t.me/learningnets

```
    Last retransmission scan length is 0, maximum is 0
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

Confirm that the endpoint addresses were learned via BGP.

```
R5#sh ip route vrf VPN_A 150.1.66.66
Routing entry for 150.1.66.66/32 Known via "bgp 100", distance 200, metric 0, type internal
  Redistributing via ospf 100
  Advertised by ospf 100 subnets
  Last update from 150.1.6.6 00:02:03 ago
  Routing Descriptor Blocks:
  * 150.1.6.6 (default), from 150.1.4.4, 00:02:03 ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: 26
      MPLS Flags: MPLS Required
!
!R6#show ip route vrf VPN_A 150.1.55.55
Routing entry for 150.1.55.55/32 Known via "bgp 100", distance 200, metric 0, type internal

  Redistributing via ospf 100
  Advertised by ospf 100 subnets
  Last update from 150.1.5.5 00:03:10 ago
  Routing Descriptor Blocks:
  * 150.1.5.5 (default), from 150.1.4.4, 00:03:10 ago
      Route metric is 0, traffic share count is 1
      AS Hops 0
      MPLS label: 26
      MPLS Flags: MPLS Required
```

Check that both R7 and R8 prefer to reach each other across the MPLS core, and that all the prefixes are seen as OSPF intra-area.

```
R7#show ip route ospf
      150.1.0.0/32 is subnetted, 4 subnets
O        150.1.8.8 [110/4] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
O E2     150.1.55.55 [110/1] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
O E2     150.1.66.66 [110/1] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
      155.1.0.0/16 is variably subnetted, 13 subnets, 2 masks
O        155.1.8.0/24 [110/4] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
O        155.1.58.0/24 [110/3] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
O        155.1.108.0/24 [110/4] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O        172.16.8.8/32 [110/4] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
O E2  192.168.6.0/24 [110/1] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
```

https://t.me/learningnets

```
O E2  192.168.7.0/24 [110/1] via 155.1.67.6, 00:03:16, GigabitEthernet1.67
!
!R8#show ip route ospf

      150.1.0.0/32 is subnetted, 4 subnets
O        150.1.7.7 [110/4] via 155.1.58.5, 00:03:48, GigabitEthernet1.58
O E2     150.1.55.55 [110/1] via 155.1.58.5, 00:04:14, GigabitEthernet1.58
O E2     150.1.66.66 [110/1] via 155.1.58.5, 00:04:06, GigabitEthernet1.58
      155.1.0.0/16 is variably subnetted, 12 subnets, 2 masks
O        155.1.7.0/24 [110/4] via 155.1.58.5, 00:03:48, GigabitEthernet1.58
O        155.1.37.0/24 [110/4] via 155.1.58.5, 00:03:48, GigabitEthernet1.58
O        155.1.67.0/24 [110/3] via 155.1.58.5, 00:03:48, GigabitEthernet1.58
O        155.1.79.0/24 [110/4] via 155.1.58.5, 00:03:48, GigabitEthernet1.58
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O        172.16.7.7/32 [110/4] via 155.1.58.5, 00:04:01, GigabitEthernet1.58
O E2  192.168.6.0/24 [110/1] via 155.1.58.5, 01:23:38, GigabitEthernet1.58
O E2  192.168.7.0/24 [110/1] via 155.1.58.5, 01:23:38, GigabitEthernet1.58
```

Check the end-to-end connectivity between R7 and R8.

```
R7#traceroute 150.1.8.8

Type escape sequence to abort.
Tracing the route to 150.1.8.8
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.67.6 21 msec 11 msec 6 msec
  2 155.1.146.4 [MPLS: Labels 16/24 Exp 0] 36 msec 14 msec 54 msec
  3 155.1.58.5 [MPLS: Label 24 Exp 0] 43 msec 45 msec 9 msec 4 155.1.58.8 16 msec *  19 msec
```

Check the data-plane information for prefix 150.1.8.8 on R5 to confirm that it is being switched using the MPLS VPN path.

```
R6#show bgp vpnv4 unicast vrf VPN_A 150.1.8.8
BGP routing table entry for 100:1:150.1.8.8/32, version 62
Paths: (1 available, best #1, table VPN_A, RIB-failure(17))
  Not advertised to any peer
  Refresh Epoch 2
  Local
    150.1.5.5 (metric 3) (via default) from 150.1.4.4 (150.1.4.4)
      Origin incomplete, metric 2, localpref 100, valid, internal, best
      Extended Community: RT:100:1 OSPF DOMAIN ID:0x0005:0x000000640200
        OSPF RT:0.0.0.1:2:0 OSPF ROUTER ID:172.16.5.5:0
      Originator: 150.1.5.5, Cluster list: 150.1.4.4 mpls labels in/out nolabel/24
      rx pathid: 0, tx pathid: 0x0
!
```

https://t.me/learningnets

```
!R6#show mpls forwarding-table 150.1.5.5

Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
21         16         150.1.5.5/32     0             Gi1.146    155.1.146.4

!
!R6#show ip cef vrf VPN_A 150.1.8.8
 150.1.8.8/32, epoch 0, flags rib defined all labels recursive via 150.1.5.5 label 24
 nexthop 155.1.146.4 GigabitEthernet1.146 label 16
```

Confirm that the backdoor link works if the primary link fails. Shut down R7's connection to R6 and confirm that the prefixes are available via the backup path. Don't forget to "no shutdown" the interface when you're done with the verifications!

```
R7:
interface GigabitEthernet1.67
 shutdown
!
!R7#show ip route ospf
      150.1.0.0/32 is subnetted, 4 subnets
O        150.1.8.8 [110/10000] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
O E2     150.1.55.55 [110/1] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
O E2     150.1.66.66 [110/1] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
      155.1.0.0/16 is variably subnetted, 12 subnets, 2 masks O        155.1.8.0/24
 [110/10000] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
O        155.1.58.0/24
           [110/10000] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
O        155.1.67.0/24
           [110/10002] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
O        155.1.108.0/24
           [110/10000] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O        172.16.8.8/32
           [110/10000] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
O E2  192.168.6.0/24 [110/1] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
O E2  192.168.7.0/24 [110/1] via 155.1.78.8, 00:00:10, GigabitEthernet1.78
!
!R7#traceroute 150.1.8.8
Type escape sequence to abort.
Tracing the route to 150.1.8.8
VRF info: (vrf in name/id, vrf out name/id) 1 155.1.78.8 13 msec *  5 msec
!
!R7:
```

https://t.me/learningnets

```
interface GigabitEthernet1.67
 no shutdown
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

PE-CE Routing with EIGRP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS PE CE Routing with EIGRP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Replace OSPF with EIGRP on the PE-CE as the routing protocol used for **VPN_A**.
- Ensure that the backdoor link is used for backup and that the primary path between the two sites is across the MPLS VPN cloud.
- All EIGRP routers should be in the same AS.

Configuration

EIGRP is an advanced distance vector protocol that uses composite metric for best path selection. One issue with transporting EIGRP routes over MP-BGP is preserving the original metric values, the route type, the source AS#, and the remote Router ID. These all are encoded using special BGP extended-community attributes that allow the remote site to properly decode the incoming routing update information. Why not just carry the metric in the MED attribute? The problem is that the remote site may happen to have different K values and have a different resulting metric. This is not something you would see in real life, but still the implementation could account for this. Of course, the local EIGRP process will treat all prefixes originated in different remote AS# as external, per the normal EIGRP rules. Therefore, the AS# and the Router-ID information could be crucial to resolve potential routing loops in scenarios with backdoor links between the VPN sites.

Another special attribute used with EIGRP prefixes redistributed into MP-BGP is known as the “cost attribute.” Although this attribute was designed to have pretty wide use, the main idea is to change the BGP best-path selection process. The

https://t.me/learningnets

problem with MPLS VPN route redistribution is that the same route may enter the PE's BGP table using redistribution (learned from a CE router) and via a BGP update (learned from a remote site). Per the BGP best-path selection process, locally redistributed prefixes have a BGP weight of 32768, which make them always win the best-path selection process. Therefore, BGP will always choose the locally received update even if the remote site has better a EIGRP metric to reach the destination.

To resolve this issue, EIGRP prefixes redistributed into MP-BGP will have the Cost attribute value set to their composite metric. The BGP process will honor the Cost attribute value before ANY other best-path selection option if the attribute is present. The prefix with the lowest cost will immediately win the best-path selection and will be redistributed into the local EIGRP process. This process happens automatically and does not require any additional configuration. Notice that the Cost attribute is NOT needed with OSPF because the prefixes received via MP-BGP are treated as inter-area and are always less preferred compared to the same prefixes learned as intra-area from a CE device. When RIP is used for PE-CE routing, Cisco IOS does not implement the Cost attribute at all, so the locally redistributed prefixes will always get preferred over MP-BGP learned prefixes, preventing effective RIP deployment in scenarios with backdoor links. This is not considered to be a big issue today, because it is no longer common to use RIP for large-scale deployments and backup routing.

When configuring EIGRP for MPLS VPNs, the same process is shared among multiple VRFs. An address family must be configured per VRF under the routing process, and for every address-family you must configure an AS number using the command **`autonomous-system N`** . This command is mandatory to enable EIGRP for that particular VRF. After that, you need only enter the normal network statements as in any EIGRP configuration.

```
R5:
no router ospf 100
!
router eigrp 100
 address-family ipv4 vrf VPN_A
  autonomous-system 100
  network 155.1.58.5 0.0.0.0
  redistribute bgp 100 metric 1 1 1 1 1
!
router bgp 100
 address-family ipv4 vrf VPN_A
  redistribute eigrp 100
R6:
```

https://t.me/learningnets

```
no router ospf 100
!
router eigrp 100
 address-family ipv4 vrf VPN_A
  autonomous-system 100
  network 155.1.67.6 0.0.0.0
  redistribute bgp 100 metric 1 1 1 1 1
!
router bgp 100
 address-family ipv4 vrf VPN_A
  redistribute eigrp 100
```

R7:

```
no router ospf 1
!
router eigrp 100
 network 0.0.0.0 255.255.255.255
!
interface GigabitEthernet1.67
 no shutdown
!
interface GigabitEthernet1.78
 delay 10000
```

R8:

```
no router ospf 1
!
router eigrp 100
 network 0.0.0.0 255.255.255.255
!
interface GigabitEthernet1.78
 delay 10000
```

Verification

Check the EIGRP routes at R7 and R8. Notice that the primary path is chosen via the PE-routers.

```
R7#show ip route eigrp

      150.1.0.0/32 is subnetted, 4 subnets
D        150.1.8.8 [90/131072] via 155.1.67.6, 00:00:06, GigabitEthernet1.67
D EX     150.1.55.55
           [170/2560000512] via 155.1.67.6, 00:00:06, GigabitEthernet1.67
D EX     150.1.66.66
```

https://t.me/learningnets

```
            [170/2560000512] via 155.1.67.6, 00:00:06, GigabitEthernet1.67
      155.1.0.0/16 is variably subnetted, 13 subnets, 2 masks
D        155.1.8.0/24 [90/3328] via 155.1.67.6, 00:00:06, GigabitEthernet1.67
D        155.1.58.0/24 [90/3072] via 155.1.67.6, 00:00:06, GigabitEthernet1.67
D        155.1.108.0/24
            [90/3328] via 155.1.67.6, 00:00:06, GigabitEthernet1.67
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
D        172.16.8.0/24
            [90/131072] via 155.1.67.6, 00:00:06, GigabitEthernet1.67
D EX  192.168.6.0/24
            [170/2562560512] via 155.1.78.8, 00:00:06, GigabitEthernet1.78
D EX  192.168.7.0/24
            [170/2562560512] via 155.1.78.8, 00:00:06, GigabitEthernet1.78
```

Check the BGP prefixes for redistributed EIGRP routes on R5 and R6, and notice the extended communities used to carry the additional EIGRP information. Also, notice the Cost attribute that carries the original route's metric.

```
R5#show bgp vpnv4 unicast vrf VPN_A 155.1.79.0
BBGP routing table entry for 100:1:155.1.79.0/24, version 110
PPaths: (1 available, best #1, table VPN_A)
F  Not advertised to any peer
   Refresh Epoch 2
   Local
     150.1.6.6 (metric 3) (via default) from 150.1.4.4 (150.1.4.4)
       Origin incomplete, metric 3072, localpref 100, valid, internal, best
       Extended Community: RT:100:1 Cost:pre-bestpath:128:3072 0x8800:32768:0
 0x8801:100:512 0x8802:65281:2560 0x8803:65281:1500 0x8806:0:2886731527

       Originator: 150.1.6.6, Cluster list: 150.1.4.4
       mpls labels in/out nolabel/31
       rx pathid: 0, tx pathid: 0x0
```

Take another prefix, which is reachable both via the backdoor link and the MPLS VPN cloud. Notice that BGP has only the path across the MPLS VPN; this is because the CE router has accepted this path as well and suppressed advertising the backup path to the PE router. This is all because of the BGP cost attribute.

```
R5#show bgp vpnv4 unicast vrf VPN_A 150.1.8.8

BGP routing table entry for 100:1:150.1.8.8/32, version 94
Paths: (1 available, best #1, table VPN_A)
  Advertised to update-groups:
     1
```

https://t.me/learningnets

```
  Refresh Epoch 1
  Local
    155.1.58.8 (via vrf VPN_A) from 0.0.0.0 (150.1.5.5)
      Origin incomplete, metric 130816, localpref 100, weight 32768, valid, sourced, best
      Extended Community: RT:100:1 Cost:pre-bestpath:128:130816

0x8800:32768:0 0x8801:100:128256 0x8802:65281:2560 0x8803:65281:1500          0x8806:0:2886731784

      mpls labels in/out 36/nolabel
      rx pathid: 0, tx pathid: 0x0
```

Verify end-to-end connectivity and confirm that the backup path (backdoor link) works as well.

```
R7#traceroute 150.1.8.8

Type escape sequence to abort.
Tracing the route to 150.1.8.8
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.67.6 16 msec 5 msec 3 msec
  2 155.1.146.4 [MPLS: Labels 16/36 Exp 0] 21 msec 21 msec 15 msec
  3 155.1.58.5 [MPLS: Label 36 Exp 0] 13 msec 11 msec 11 msec 4 155.1.58.8 62 msec *  13 msec
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.67
R7(config-if)#shutdown
!
!R7#show ip route 150.1.8.8
Routing entry for 150.1.8.8/32
  Known via "eigrp 100", distance 90, metric 2690560, type internal
  Redistributing via eigrp 100
  Last update from 155.1.78.8 on GigabitEthernet1.78, 00:00:20 ago
  Routing Descriptor Blocks: * 155.1.78.8, from 155.1.78.8, 00:00:20 ago, via GigabitEthernet1.78
      Route metric is 2690560, traffic share count is 1
      Total delay is 105000 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
!
!R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.67
R7(config-if)#no shutdown
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

EIGRP Site-of-Origin

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS EIGRP Site of Origin**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure R5 and R6 to prevent temporary routing loops that may occur because of mutual redistribution between EIGRP and MP-BGP.
- Ensure that the path across the MPLS VPN core is used as the primary path between R7 and R8.

Configuration

With multi-homed scenarios similar to the ones we are using, BGP and EIGRP perform mutual redistribution at PE routers. This may potentially result in transient routing loops, when a prefix is withdrawn in MP-BGP at one PE router and it is not timely propagated into EIGRP. EIGRP could then feed the invalid information back to BGP at another PE router, keeping the false information circulating between the PE routers until it's eliminated by counting to infinity. The core of this problem is the mutual redistribution that allows the information learned from BGP at one site to re-enter BGP at another site. This problem is commonly resolved by tagging the prefixes from one protocol while redistributing them into another, and then blocking the prefixes from re-entering the domain of origin based on the tag.

Implementing this solution using EIGRP route tags and BGP communities could be cumbersome if done using regular methods. Therefore, Cisco IOS implements a special feature known as EIGRP Site-of-Origin (SoO), which uses an extended community appended to BGP and EIGRP routing updates (EIGRP's TLV format allows adding this information easily). The feature is configured at the interface level

https://t.me/learningnets

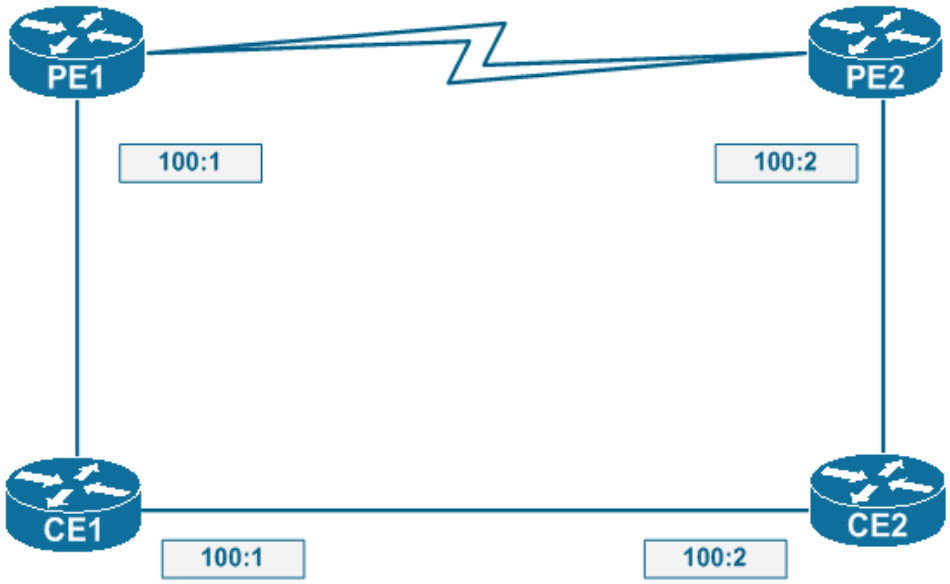
using the command `ip vrf sitemap <ROUTE_MAP>` , where ROUTE_MAP is a regular route-map that applies the command `set extcommunity soo ASN:XX` , where ASN:XX is the “VPN identifier” common for all PEs of the same multi-homed site.

All BGP prefixes redistributed into EIGRP and sent over the interface with the SoO set will have this extended community appended, but only if the community is not already present. If the update to be sent already has the same extended community value set, it is discarded as being redistributed back to the same site. The next thing that this feature does is apply the same extended community to all EIGRP routes received on the interface and redistributed into BGP. All these actions are performed by IOS automatically; you need only apply the route-map to the relevant interfaces.

There are two common ways of applying the VRF sitemap feature:

- The SoO is applied on the PE interfaces facing the CE routers. Every PE router uses the same SoO value. This prevents PE routers from learning MP-BGP originated routes from CE routers. However, the side-effect is that the MPLS core cannot be used as a backup path between the segments of the multi-homed sites if the backdoor link fails. This is because MP-BGP updates for in-site EIGRP prefixes carry the same SoO that is applied to the PE interfaces facing the CE routers and therefore cannot be redistributed back to the same site.
- If you need to preserve the path across the MPLS core network, you should use different SoO values at every PE router of a multi-homed site. However, this means that the MP-BGP information injected into EIGRP at one PE router will reach the other PEs without being blocked. To prevent this effect, an additional SoO interface should be configured on the CE routers with the backdoor link. Look at the figure below and notice the SoO settings on the respective PE/CE interfaces.

https://t.me/learningnets



In this case, if PE1 redistributes a prefix into MP-BGP, it will be tagged with 100:1. The prefix will reach PE2 and pass down to CE2. The prefix will finally be stopped at CE1 by SoO filtering and will never loop back to PE1 again. The same applies to prefixes redistributed into MP-BGP by PE2.

```
R5:
route-map EIGRP_SOO
 set extcommunity soo 100:15
!
interface GigabitEthernet1.58
 ip vrf sitemap EIGRP_SOO
R6:
route-map EIGRP_SOO
 set extcommunity soo 100:16
!
interface GigabitEthernet1.67
 no shutdown
 ip vrf sitemap EIGRP_SOO
R7:
route-map EIGRP_SOO
 set extcommunity soo 100:16
!
interface GigabitEthernet1.78
 ip vrf sitemap EIGRP_SOO
R8:

route-map EIGRP_SOO
 set extcommunity soo 100:15
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.78
 ip vrf sitemap EIGRP_SOO
```

Verification

Check the BGP prefixes for EIGRP routes and notice the SoO community values.

```
R6#show bgp vpnv4 unicast vrf VPN_A 150.1.8.8
BGP routing table entry for 100:1:150.1.8.8/32, version 183
Paths: (1 available, best #1, table VPN_A)
  Not advertised to any peer
  Refresh Epoch 2
  Local
    150.1.5.5 (metric 3) (via default) from 150.1.4.4 (150.1.4.4)
      Origin incomplete, metric 130816, localpref 100, valid, internal, best
Extended Community: SoO:100:15
 RT:100:1 Cost:pre-bestpath:128:130816
        0x8800:32768:0 0x8801:100:128256 0x8802:65281:2560 0x8803:65281:1500
        0x8806:0:2886731784
      Originator: 150.1.5.5, Cluster list: 150.1.4.4
      mpls labels in/out nolabel/20
      rx pathid: 0, tx pathid: 0x0
!
!R6#show bgp vpnv4 unicast vrf VPN_A 150.1.7.7
BGP routing table entry for 100:1:150.1.7.7/32, version 233
Paths: (1 available, best #1, table VPN_A)
  Advertised to update-groups:
     1
  Refresh Epoch 1
  Local
    155.1.67.7 (via vrf VPN_A) from 0.0.0.0 (150.1.6.6)
      Origin incomplete, metric 130816, localpref 100, weight 32768, valid, sourced, best
Extended Community: SoO:100:16
 RT:100:1 Cost:pre-bestpath:128:130816
        0x8800:32768:0 0x8801:100:128256 0x8802:65281:2560 0x8803:65281:1500
        0x8806:0:2886731527
      mpls labels in/out 24/nolabel
      rx pathid: 0, tx pathid: 0x0
```

Check the routing tables in the CE routers and confirm that primary paths are via the MPLS core.

```
R7#sh ip route eigrp
```

https://t.me/learningnets

```
      150.1.0.0/32 is subnetted, 4 subnets
D        150.1.8.8 [90/131072] via 155.1.67.6, 00:01:20, GigabitEthernet1.67
D EX     150.1.55.55
           [170/2560000512] via 155.1.67.6, 00:01:20, GigabitEthernet1.67
D EX     150.1.66.66
           [170/2560000512] via 155.1.67.6, 00:01:20, GigabitEthernet1.67
      155.1.0.0/16 is variably subnetted, 13 subnets, 2 masks
D        155.1.8.0/24 [90/3328] via 155.1.67.6, 00:01:20, GigabitEthernet1.67
D        155.1.58.0/24 [90/3072] via 155.1.67.6, 00:01:20, GigabitEthernet1.67
D        155.1.108.0/24
           [90/3328] via 155.1.67.6, 00:01:20, GigabitEthernet1.67
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
D        172.16.8.0/24
           [90/131072] via 155.1.67.6, 00:01:20, GigabitEthernet1.67
D EX  192.168.6.0/24
           [170/2562560512] via 155.1.78.8, 00:06:04, GigabitEthernet1.78
D EX  192.168.7.0/24
           [170/2562560512] via 155.1.78.8, 00:06:04, GigabitEthernet1.78
!
!R8#show ip route eigrp

      150.1.0.0/32 is subnetted, 4 subnets
D        150.1.7.7 [90/131072] via 155.1.58.5, 00:02:52, GigabitEthernet1.58
D EX     150.1.55.55
           [170/2560000512] via 155.1.58.5, 00:02:53, GigabitEthernet1.58
D EX     150.1.66.66
           [170/2560000512] via 155.1.58.5, 00:02:53, GigabitEthernet1.58
      155.1.0.0/16 is variably subnetted, 12 subnets, 2 masks
D        155.1.7.0/24 [90/3328] via 155.1.58.5, 00:02:52, GigabitEthernet1.58
D        155.1.37.0/24 [90/3328] via 155.1.58.5, 00:02:52, GigabitEthernet1.58
D        155.1.67.0/24 [90/3072] via 155.1.58.5, 00:02:56, GigabitEthernet1.58
D        155.1.79.0/24 [90/3328] via 155.1.58.5, 00:02:52, GigabitEthernet1.58
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
D        172.16.7.0/24
           [90/131072] via 155.1.58.5, 00:02:52, GigabitEthernet1.58
D EX  192.168.6.0/24
           [170/2560000512] via 155.1.58.5, 00:08:07, GigabitEthernet1.58
D EX  192.168.7.0/24
           [170/2560000512] via 155.1.58.5, 00:08:07, GigabitEthernet1.58
```

Shut down the PE-CE link at R7 and confirm that the backup link is still usable (don't forget to bring the link back up when you're done verifying!).

```
R7#configure terminal
```

https://t.me/learningnets

```
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.67
R7(config-if)#shutdown

!
!R7#show ip route eigrp

      150.1.0.0/32 is subnetted, 4 subnets
D        150.1.8.8 [90/2690560] via 155.1.78.8, 00:00:43, GigabitEthernet1.78
D EX     150.1.55.55
           [170/2562560512] via 155.1.78.8, 00:00:43, GigabitEthernet1.78
D EX     150.1.66.66
           [170/2562560512] via 155.1.78.8, 00:00:43, GigabitEthernet1.78
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks
D        155.1.8.0/24
           [90/2562816] via 155.1.78.8, 00:00:43, GigabitEthernet1.78
D        155.1.58.0/24
           [90/2562816] via 155.1.78.8, 00:00:43, GigabitEthernet1.78
D        155.1.108.0/24
           [90/2562816] via 155.1.78.8, 00:00:43, GigabitEthernet1.78
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
D        172.16.8.0/24
           [90/2690560] via 155.1.78.8, 00:00:43, GigabitEthernet1.78
D EX  192.168.6.0/24
           [170/2562560512] via 155.1.78.8, 00:09:37, GigabitEthernet1.78
D EX  192.168.7.0/24
           [170/2562560512] via 155.1.78.8, 00:09:37, GigabitEthernet1.78
!
!R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.67
R7(config-if)#no shutdown
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

PE-CE Routing with BGP

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS PE CE Routing with BGP**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Erase all EIGRP configurations for VPN_A on R5, R6, R7, and R8.
- Configure BGP AS 78 on R7 and R8, and peer the CE routers with R5 and R6.
- Advertise the Loopback0 interfaces of R7 and R8 into BGP and ensure end-to-end reachability.

Configuration

BGP seems to be a natural fit for PE-CE routing because of its perfect integration with core MP-BGP routing. Advantages include perfect scalability and precise route control. Slow convergence for intra-site routing is a disadvantage. However, as a balanced option, you may use an IGP for intra-site routing and BGP for PE-CE peering to get the best balance.

A common problem that arises from the use of BGP is that multiple sites may reuse the same AS number. Because of the BGP loop-prevention mechanism, this will filter BGP updates sent between the sites. There are two solutions to this problem: The first one is to configure the `allowas-in` option inbound for the CE peering session. The other is to configure the `as-override` option on the PE routers for the eBGP peering sessions with the CE routers. This option compares the remote-AS number with the AS number stored in the end of the AS_PATH attribute. If they match, the AS number in AS_PATH is replaced with the local PE router's AS number. The AS numbers used in the BGP peering sessions remain the same; only the AS_PATH attribute is changed when updates are relayed. Notice that the length

https://t.me/learningnets

of the AS_PATH attribute remains the same, which allows for proper best-path selection.

Configuring BGP for PE-CE routing requires only activating the respective VRF's address family under the global BGP process and configuring the BGP peering sessions under this VRF. There is no need to configure any redistribution in this case, because routes are propagated into the VPNv4 table automatically. Just as usual, prefix import and export is controlled by the route-target extended communities. If you want to, you may additionally redistribute other IGP protocols into BGP, or redistribute connected interfaces.

```
R5:
no router eigrp 100
!
interface GigabitEthernet1.58
 no ip vrf sitemap EIGRP_SOO
!
router bgp 100
 address-family ipv4 vrf VPN_A
 neighbor 155.1.58.8 remote-as 78
 neighbor 155.1.58.8 as-override
R6:
no router eigrp 100
!
interface GigabitEthernet1.67
 no shutdown
 no ip vrf sitemap EIGRP_SOO
!
router bgp 100
 address-family ipv4 vrf VPN_A
 neighbor 155.1.67.7 remote-as 78
 neighbor 155.1.67.7 as-override
R7:
no router eigrp 100
!
interface GigabitEthernet1.78
 no ip vrf sitemap EIGRP_SOO
!
interface GigabitEthernet1.67
 no shutdown
!
 router bgp 78
 neighbor 155.1.67.6 remote-as 100
 network 150.1.7.7 mask 255.255.255.255
R8:
```

https://t.me/learningnets

```
no router eigrp 100
!
interface GigabitEthernet1.78
 no ip vrf sitemap EIGRP_SOO
!
router bgp 78
 neighbor 155.1.58.5 remote-as 100
 network 150.1.8.8 mask 255.255.255.255
```

Verification

Check the BGP routes learned on R7. Look into the BGP table and confirm that the “source” AS 78 no longer appears in the AS_PATH and is replaced by the “core” AS 100. After this, verify end-to-end connectivity across the MPLS core.

```
R7#show ip route bgp
      150.1.0.0/32 is subnetted, 4 subnets
B        150.1.8.8 [20/0] via 155.1.67.6, 00:00:02
B        150.1.55.55 [20/0] via 155.1.67.6, 00:11:21
B        150.1.66.66 [20/0] via 155.1.67.6, 00:11:21
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks
B        155.1.58.0/24 [20/0] via 155.1.67.6, 00:11:21
B     192.168.7.0/24 [20/0] via 155.1.67.6, 00:11:21
!
!R7#show ip bgp
BGP table version is 8, local router ID is 172.16.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  150.1.7.7/32     0.0.0.0                  0         32768 i
 *>  150.1.8.8/32     155.1.67.6                              0 100 100 i
 *>  150.1.55.55/32   155.1.67.6                              0 100 i
 *>  150.1.66.66/32   155.1.67.6               0             0 100 i
 *>  155.1.58.0/24    155.1.67.6                              0 100 ?
 r>  155.1.67.0/24    155.1.67.6               0             0 100 ?
 *>  192.168.7.0      155.1.67.6               0             0 100 ?
!
!R7#traceroute 150.1.8.8 source loopback0

Type escape sequence to abort.
```

https://t.me/learningnets

```
Tracing the route to 150.1.8.8
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.67.6 [AS 100] 29 msec 3 msec 1 msec
  2 155.1.146.4 [MPLS: Labels 16/43 Exp 0] 43 msec 6 msec 14 msec
  3 155.1.58.5 [AS 100] [MPLS: Label 43 Exp 0] 8 msec 6 msec 9 msec
  4 155.1.58.8 [AS 100] 11 msec *  93 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

BGP SoO Attribute

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS BGP SOO Attribute**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure a backdoor BGP peering session between R7 and R8 using the inter-switch link configured previously.
- Ensure that this configuration does not result in routing loops caused by BGP's loop-prevention mechanism being disabled by the AS-Override feature.

Configuration

The use of the AS-Override feature allows circumventing the BGP loop prevention mechanism. However, the drawback is that this may result in routing-loops for multi-homed sites with backdoor links. If a site injects a prefix into MP-BGP, the prefix may re-enter the same site via another PE because AS-Override will replace the source AS.

To overcome this issue for multi-homed sites, BGP implements a SoO attribute similar to the one used for EIGRP. However, this time you configure the SoO value per-neighbor peering session in the PE routers. There are two basic methods of doing this:

- Setting the SoO attribute for incoming/outgoing prefixes on the peering session using the command `neighbor <IP> soo <VALUE>` . This command is available in IOS 12.4(11)T and higher and is a more elegant way of configuring the feature.
- Setting the SoO attribute using a route-map command `set extcommunity soo <VALUE>`

https://t.me/learningnets

and applying this route map inbound to the neighbor session via the command `neighbor <IP> route-map <NAME>in` .

BGP SoO works very similar to EIGRP SoO: If an incoming or outgoing update has the SoO value matching the locally configured one, the update is dropped. Otherwise, the update is tagged with the SoO extended community. Based on this similarity, you may quickly learn that using the same SoO at all PEs will prevent the MPLS VPN core from being used as a backup path if the backdoor link fails. The solution is, again, using different SoO values at PE routers and applying SoO at the CE routers connected to the backdoor link. However, in this scenario we simply configure the SoO on the PE routers.

```
R5:
router bgp 100
 address-family ipv4 vrf VPN_A
 neighbor 155.1.58.8 soo 100:1
R6:
router bgp 100
 address-family ipv4 vrf VPN_A
 neighbor 155.1.67.7 soo 100:1
R7:
router bgp 78
 neighbor 155.1.78.8 remote-as 78
R8:

router bgp 78
 neighbor 155.1.78.7 remote-as 78
```

Verification

Look into the BGP tables of the PE routers and confirm that BGP prefixes learned from the CEs are now tagged with the SoO attribute. Start with R6 and the prefix 150.1.8.8/32. Confirm that this prefix is not advertised to the CE router (R7).

```
R6#show bgp vpnv4 unicast vrf VPN_A 150.1.8.8
BGP routing table entry for 100:1:150.1.8.8/32, version 271
Paths: (2 available, best #1, table VPN_A)
  Advertised to update-groups:
     1
  Refresh Epoch 3
  78
    155.1.67.7 (via vrf VPN_A) from 155.1.67.7 (172.16.7.7)
      Origin IGP, localpref 100, valid, external, best      Extended Community: SoO:100:1
 RT:100:1
```

https://t.me/learningnets

```
      mpls labels in/out 31/nolabel
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 3
  78
    150.1.5.5 (metric 3) (via default) from 150.1.4.4 (150.1.4.4)
      Origin IGP, metric 0, localpref 100, valid, internal      Extended Community: SoO:100:1
 RT:100:1
      Originator: 150.1.5.5, Cluster list: 150.1.4.4
      mpls labels in/out 31/43
      rx pathid: 0, tx pathid: 0
!
!R6#sh bgp vpnv4 unicast vrf VPN_A neighbors 155.1.67.7 advertised-routes

BGP table version is 272, local router ID is 150.1.6.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 100:1 (default for vrf VPN_A)
 *>i 150.1.55.55/32   150.1.5.5                0    100      0 i
 *>  150.1.66.66/32   0.0.0.0                  0         32768 i
 *>i 155.1.58.0/24    150.1.5.5                0    100      0 ?
 *>  155.1.67.0/24    0.0.0.0                  0         32768 ?
 *>  192.168.7.0      155.1.76.7               0         32768 ?

Total number of prefixes 5
```

R5 has two prefixes in the BGP table for 150.1.7.7: one learned from R6 via R4 (the RR) and another learned from R8. Both are tagged with the SoO of 100:1. R5 does not advertise the prefix 150.1.7.7/32 to R8, in accordance with the rules of SoO filtering.

```
R5#show bgp vpnv4 unicast vrf VPN_A 150.1.7.7
BGP routing table entry for 100:1:150.1.7.7/32, version 201
Paths: (1 available, best #1, table VPN_A)
  Advertised to update-groups:
     1
  Refresh Epoch 3
  78
    155.1.58.8 (via vrf VPN_A) from 155.1.58.8 (172.16.8.8)
      Origin IGP, localpref 100, valid, external, best      Extended Community: SoO:100:1
 RT:100:1
```

https://t.me/learningnets

```
      mpls labels in/out 25/nolabel
      rx pathid: 0, tx pathid: 0x0
!
!R5#show bgp vpnv4 unicast vrf VPN_A neighbors 155.1.58.8 advertised-routes

BGP table version is 202, local router ID is 150.1.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 100:1 (default for vrf VPN_A)
 *>  150.1.55.55/32   0.0.0.0                  0         32768 i
 *>i 150.1.66.66/32   150.1.6.6                0    100      0 i
 *>  155.1.58.0/24    0.0.0.0                  0         32768 ?
 *>i 155.1.67.0/24    150.1.6.6                0    100      0 ?
 *>  172.16.5.0/24    0.0.0.0                  0         32768 ?
 *>i 192.168.6.0      150.1.6.6                0    100      0 ?
 *>i 192.168.7.0      150.1.6.6                0    100      0 ?

Total number of prefixes 7
```

We can also look under the neighbor for routes filtered outbound from SoO.

```
R5#show bgp vpnv4 unicast vrf VPN_A neighbors 155.1.58.8

<snip>
                                   Outbound    Inbound
  Local Policy Denied Prefixes:    --------    -------
    SOO loop:                             5        n/a
    Bestpath from this peer:              2        n/a
    Total:                                7          0
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

Internet Access

> You must load the initial configuration files for the section, **MPLS Internet Access**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a new subinterface between R6 and R10, GigabitEthernet1.106 with dot1q tag of 106.
  - Use IP addressing in the format **160.1.106.Y/24**, where *Y* is the router number.
- Enable BGP between R10 and R6 and configure R10 in AS 106.
  - On R10, advertise its Loopback0 interface into BGP.
- Configure R6 so that **VPN_A** customers may access R10's Loopback0.
  - You are allowed to use one default static route to accomplish this task.
  - Only the 150.X.0.0/16 subnets should be allowed to access the Internet using NAT.

Configuration

Internet access is a common service that must often be combined with MPLS VPN services provided by an SP. In many cases, the solution is to simply provide an additional PE-CE link that belongs to the global routing table in the PE, or the special VRF containing the Internet routes. In some situations, however, it is desirable to use a single point of attachment for both types of service. In this case, two problems must be addressed:

- Injecting the Internet routes into the respective VRF. In many situations, it is enough to inject just the default route. If the Internet access is provided via the global routing table, a special type of static route should be used. If the Internet access is provided

https://t.me/learningnets

via a special VPN, classic route export/import could be used.

- The Internet routing table should learn about the prefixes found in the VPN that needs VPN access. In many situations, VPNs use private IP addressing, so a kind of NAT mechanism is required to hide the source IP addresses and translate them into routable equivalents.

In this scenario, we will use the most common case, when Internet access is provided via the global routing table. We assume that the VPNs are using private IP addressing, which requires NAT to be configured when accessing the Internet. In our case, the “Internet” is emulated by R10's loopback. Here are the steps required to configure NAT-based Internet access via the global routing table:

1. Create a special default VRF route that resolves via the global routing table. The syntax is `ip route vrf <NAME> 0.0.0.0 0.0.0.0 <NEXT_HOP> global .` The last keyword means that even though the route is bound to a VRF, the next hop should be resolved using the global routing table, and FIB should be programmed accordingly. You may need to redistribute this route into MP-BGP to propagate it to all VPN sites.

2. Enable NAT on the Internet and VPN links. You must set up all VPN-facing links as NAT inside and the Internet-facing link as NAT outside. After this, create a global NAT address pool if needed. This address pool should be reachable via the global routing table.

3. This step is the core of NAT-based VPN Internet access. You must configure a source NAT translation rule that matches the VPN source IP addresses and specifies either the global pool or the global interface for address translation. This rule should end with the keyword `vrf <VRF_NAME> ,` which selects the source IP addresses only from the particular VRF. The ability to apply the NAT rules to IP addresses found in a particular VRF is a new feature of the NAT translation engine.

After these three steps have been configured, the VPN clients will use the injected default route to access all non-matched prefixes. The packets will be routed to the Internet-access router that has the NAT rule set up. After this, the source IP addresses will be changed to the addresses routable via the global table, and the respective NAT entries will be created. Even though the packets originally belong to the VRF, they will be switched using the global routing table and the NAT entries will serve as “entry points” back to the VRF for the returning packets.

Notice that our solution uses BGP, so we cannot simply redistribute the static

https://t.me/learningnets

default route. We use the BGP address-family command **default-information originate** to accomplish this task. This command requires the default route to be advertised into the local BGP table by using either redistribution or the **network** command. Only after this will the default route be propagated to all peers.

```
R6:
interface GigabitEthernet1.106
 encapsulation dot1Q 106
 ip address 160.1.106.6 255.255.255.0
!
ip route vrf VPN_A 0.0.0.0 0.0.0.0 GigabitEthernet 1.106 160.1.106.10 global
!
router bgp 100
 neighbor 160.1.106.10 remote-as 106
address-family ipv4
 neighbor 160.1.106.10 activate
address-family ipv4 vrf VPN_A
 default-information originate
 redistribute static
!
interface GigabitEthernet1.106
 ip nat outside
!
interface GigabitEthernet1.146
 ip nat inside
!
interface GigabitEthernet1.67
 ip nat inside
!
ip access-list standard VPN_PREFIXES
 permit 150.1.0.0 0.0.255.255
!
ip nat inside source list VPN_PREFIXES interface GigabitEthernet1.106 vrf VPN_A overload
R10:

interface GigabitEthernet1.106
 encapsulation dot1Q 106
 ip address 160.1.106.10 255.255.255.0
!
router bgp 106
 network 150.1.10.10 mask 255.255.255.255
 neighbor 160.1.106.6 remote-as 100
```

https://t.me/learningnets

Verification

Check the routing table of any of the CEs for the default route.

```
R7#show ip route bgp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 155.1.67.6 to network 0.0.0.0
B*    0.0.0.0/0 [20/0] via 155.1.67.6, 00:02:11

      150.1.0.0/32 is subnetted, 4 subnets
B        150.1.8.8 [200/0] via 155.1.78.8, 01:28:47
B        150.1.55.55 [20/0] via 155.1.67.6, 01:53:08
B        150.1.66.66 [20/0] via 155.1.67.6, 01:53:08
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks
B        155.1.58.0/24 [20/0] via 155.1.67.6, 01:53:08
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
B        172.16.5.0/24 [200/0] via 155.1.58.5, 01:28:47
B     192.168.6.0/24 [200/0] via 155.1.58.5, 01:28:47
B     192.168.7.0/24 [20/0] via 155.1.67.6, 01:53:08
```

Test connectivity to the “Internet” route learned from R10 from the CEs, and verify the associated NAT translations for each on R6.

```
R6#show ip route bgp
      150.1.0.0/32 is subnetted, 4 subnets
B        150.1.10.10 [20/0] via 160.1.106.10, 00:07:24
!
!R7#ping 150.1.10.10 source loopback0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.7.7 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/11/33 ms
!
```

https://t.me/learningnets

```
!R6#show ip nat translations

Pro  Inside global         Inside local          Outside local         Outside global
icmp 160.1.106.6:1         150.1.7.7:3           150.1.10.10:3         150.1.10.10:1
Total number of translations: 1
!
!R6#show ip nat translations verbose

Pro  Inside global         Inside local          Outside local         Outside global
icmp 160.1.106.6:1         150.1.7.7:3           150.1.10.10:3         150.1.10.10:1
  create: 05/10/14 20:03:13, use: 05/10/14 20:03:13, timeout: 00:00:32
  Map-Id(In): 1
  Appl type: none   Mac-Address: 0000.0000.0000 Input-IDB: GigabitEthernet1.67
 VRF: VPN_A
,  entry-id: 0xeb9e8a90, use_count:1

Total number of translations: 1
!
!R8#ping 150.1.10.10 source loopback 0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.8.8 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/13/18 ms
!
!R6#show ip nat translations verbose

Pro  Inside global         Inside local          Outside local         Outside global
icmp 160.1.106.6:1         150.1.8.8:2           150.1.10.10:2         150.1.10.10:1
  create: 05/10/14 20:04:57, use: 05/10/14 20:04:57, timeout: 00:00:41
  Map-Id(In): 1
  Appl type: none   Mac-Address: 0000.0000.0000 Input-IDB: GigabitEthernet1.146
 VRF: VPN_A
,  entry-id: 0xeb9e8b60, use_count:1

Total number of translations: 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - MPLS

MPLS VPN Performance Tuning

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, named **MPLS VPN Performance Tuning**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure the PE and routers to minimize the amount of time needed to propagate topology changes between CE-sites.

Configuration

MPLS VPNs utilize MP-BGP as the transport for IGP routing updates between VPN sites. BGP was designed as a highly scalable protocol; this, however, limits its ability to quickly propagate changes in the network topology. One of the reasons for this is the goal to maintain network stability, which, of course, slows convergence. Thus, it is not uncommon to see network changes at one site take a minute or even more to propagate to other sites, even with “fast” protocols such as OSPF and EIGRP. The following factors affect the amount of time it takes to propagate a topology change from one PE to another across the MP-BGP core:

- The time it takes for the IGP update to be redistributed into BGP. In the past, this was limited by the `bgp scan-interval` general scanning interval, but recent IOS versions made this feature event driven. This makes IGP-to-BGP route redistribution almost instant.
- The time it takes the local BGP speaker and other BGP speakers to propagate updates to their peers. This time is controlled by the timer known as the peer advertisement-interval. This interval specifies the periodic “batching” of events: BGP waits for the timer to expire and accumulates the updates instead of sending every

https://t.me/learningnets

one immediately. The purpose of this is to prevent a constant load on the peer's CPU for best-path re-calculations. In versions of the code prior to 12.0(32)S, the default interval was 5 seconds and could be set as low as 0 seconds using the command `neighbor <IP> advertisement-interval` . In recent versions of the code, the advertisement-interval is set to 0 for iBGP peers and VPNV4 eBGP peers (PE-CE).

- The time it takes the PE router's BGP process to import the MP-BGP VPNv4 prefixes into the local VRF table. This timer is controlled by the command `bgp scan-time import <5-60>` , which should be entered under the VPNv4 address-family configuration. The default value is 15 seconds, and setting it to 5 seconds may significantly decrease the update propagation time. This method of speeding up the convergence has also been deprecated, though. A new enhanced way of importing routes into the VRF exists called "Event-Based VPN Import." Instead of waiting for the 5-15 seconds that the scan time would take causing delay, the paths can be immediately imported as soon as a change is detected, allowing the PE routers to propagate the change to CE routers without having to wait for the scan-time to run. This feature gives control via a configurable policy as to which paths to import.

Configuration

```
! Note that the timers are set to 0 by default in new versions of code! R4:
router bgp 100
 address-family vpnv4 unicast
  neighbor 150.1.5.5 advertisement-interval 0
  neighbor 150.1.6.6 advertisement-interval 0

! Note that the timers are set to 0 by default in new versions of code!     R5 & R6:

router bgp 100
 address-family vpnv4 unicast
  neighbor 150.1.4.4 advertisement-interval 0
 address-family ipv4 unicast vrf VPN_A
  import path selection all
```

Verification

Check the optimized BGP timers on one of the routers. The other routers' configuration is verified similarly. Note that this is the default value on the version of code that is being used for these examples.

```
R5#show bgp vpnv4 unicast all neighbors 150.1.4.4  | inc adv
```

https://t.me/learningnets

```
    Route refresh: advertised and received(new)
    New ASN Capability: advertised and received
    Address family VPNv4 Unicast: advertised and received
  Default minimum time between advertisement runs is 0 seconds
```

To validate the enhanced import process, run the following debugging command on R5 and shut down R8's Loopback0. Notice that this is not timer based but purely event driven.

```
R5#debug ip bgp vpnv4 unicast import events

BGP events debugging is on

BGP VPN-IMP: nbr topo freed: Exported path deleted, cleanup scheduled.
BGP: IMP Process work queue.
BGP: tbl VPNv4 Unicast:base IMP Incremental export.
BGP: IMP Work queue processed.
BGP: IMP Process work queue.
BGP: tbl VPNv4 Unicast:base IMP Incremental export.
BGP: IMP Work queue processed.
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

IPsec VPNs with Crypto Maps

> You must load the initial configuration files for the section, **IPsec VPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure routing on R7 - R10 as follows:
  - Configure two IPv4 static default routes on R7, one toward R3 and one toward R6.
  - Configure a static default route on R8 toward R5.
  - Configure OSPF area 0 between R7 and R9, and advertise all links on R9 into OSPF.
  - Configure OSPF area 0 between R8 and R10, and advertise all links on R10 into OSPF.
  - Configure R7 and R8 to originate default routes into OSPF area 0.

- Configure an IPsec tunnel between R7 and R8 as follows:
  - Use an ISAKMP Policy with the following options:
    - Pre-Shared Key: **CISCO**
    - Encryption: AES 256 bit
    - Hash: SHA 512 bit
    - Diffie-Hellman Group: 24
  - Use a Crypto Map named **R7_TO_R8** with the following options:
    - R7 and R8 will be the tunnel endpoints.
    - Traffic from 150.1.9.9/32 and 155.1.9.0/24 going to 150.1.10.10/32 and 155.1.10.0/24 should be protected by the IPsec tunnel, and vice-versa.
    - Encrypt the traffic using 192-bit AES.

https://t.me/learningnets

    - Authenticate the traffic using 384-bit SHA.
  - Ensure that if R7's links to either R3 or R6 go down, the tunnel remains up and is re-routed via any available alternate path.

Configuration

```
R7:
crypto isakmp policy 10
 encr aes 256
 hash sha512
 authentication pre-share
 group 24
!
crypto isakmp key CISCO address 155.1.58.8
!
crypto ipsec transform-set ESP-AES-192-SHA-384 esp-aes 192 esp-sha384-hmac
 mode tunnel
!
ip access-list extended R9_TO_R10
 permit ip host 150.1.9.9 host 150.1.10.10
 permit ip host 150.1.9.9 155.1.10.0 0.0.0.255
 permit ip 155.1.9.0 0.0.0.255 host 150.1.10.10
 permit ip 155.1.9.0 0.0.0.255 155.1.10.0 0.0.0.255
!
crypto map R7_TO_R8 local-address Loopback0
!
crypto map R7_TO_R8 10 ipsec-isakmp
 set peer 155.1.58.8
 set transform-set ESP-AES-192-SHA-384
 match address R9_TO_R10
!
interface GigabitEthernet1.37
 crypto map R7_TO_R8
!
interface GigabitEthernet1.67
 crypto map R7_TO_R8
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
!
router ospf 1
 default-information originate
!
ip route 0.0.0.0 0.0.0.0 155.1.37.3
```

https://t.me/learningnets

```
ip route 0.0.0.0 0.0.0.0 155.1.67.6
```

R8:

```
crypto isakmp policy 10
 encr aes 256
 hash sha512
 authentication pre-share
 group 24
!
crypto isakmp key CISCO address 150.1.7.7
!
crypto ipsec transform-set ESP-AES-192-SHA-384 esp-aes 192 esp-sha384-hmac
 mode tunnel
!
ip access-list extended R10_TO_R9
 permit ip host 150.1.10.10 host 150.1.9.9
 permit ip host 150.1.10.10 155.1.9.0 0.0.0.255
 permit ip 155.1.10.0 0.0.0.255 host 150.1.9.9
 permit ip 155.1.10.0 0.0.0.255 155.1.9.0 0.0.0.255
!
crypto map R7_TO_R8 10 ipsec-isakmp
 set peer 150.1.7.7
 set transform-set ESP-AES-192-SHA-384
 match address R10_TO_R9
!
interface GigabitEthernet1.58
 crypto map R7_TO_R8
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
!
router ospf 1
 default-information originate
!
ip route 0.0.0.0 0.0.0.0 155.1.58.5
```

R9:

```
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.9
 ip ospf 1 area 0
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
```

https://t.me/learningnets

```
R10:

interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.10
 ip ospf 1 area 0
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
```

Verification

The first requirement of an IPsec tunnel is that routing reachability be obtained between the tunnel endpoints, and end to end between the hosts whose traffic will actually be sent over the tunnel. In this design, reachability between tunnel endpoints is obtained by EIGRP that is pre-configured in the main transit network.

```
R7#show ip route 155.1.58.8
Routing entry for 155.1.58.0/24 Known via "eigrp 100"
, distance 90, metric 3584, type internal
  Redistributing via eigrp 100
  Last update from 155.1.67.6 on GigabitEthernet1.67, 01:08:59 ago
  Routing Descriptor Blocks: * 155.1.67.6, from 155.1.67.6, 01:08:59 ago, via GigabitEthernet1.67
      Route metric is 3584, traffic share count is 1
      Total delay is 40 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 3
!R8#show ip route 150.1.7.7
Routing entry for 150.1.7.7/32 Known via "eigrp 100"
, distance 90, metric 131584, type internal
  Redistributing via eigrp 100
  Last update from 155.1.58.5 on GigabitEthernet1.58, 01:08:06 ago
  Routing Descriptor Blocks: * 155.1.58.5, from 155.1.58.5, 01:08:06 ago, via GigabitEthernet1.58

      Route metric is 131584, traffic share count is 1
      Total delay is 5040 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 4
```

Because dynamic routing is not supported over a Crypto Map-based IPsec tunnel, a static default route is used to provide end-to-end reachability for traffic sent inside the IPsec tunnel. This default route is then advertised to the devices in the private

https://t.me/learningnets

network (R9 and R10) through IGP. Additionally, because R7 has multiple exit points to the transit network, it is using ECMP for the default routes toward R3 and R6.

```
R7#show ip route 155.1.10.10
% Subnet not in table
!R7#show ip cef 155.1.10.10
  0.0.0.0/0
nexthop 155.1.37.3 GigabitEthernet1.37
nexthop 155.1.67.6 GigabitEthernet1.67
!  R8#show ip route 155.1.9.9
% Subnet not in table
!R8#show ip cef 155.1.9.9
0.0.0.0/0
nexthop 155.1.58.5 GigabitEthernet1.58
!R9#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 155.1.79.7 to network 0.0.0.0
O*E2  0.0.0.0/0
 [110/1] via 155.1.79.7, 01:12:38, GigabitEthernet1.79
!R10#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 155.1.108.8 to network 0.0.0.0
O*E2  0.0.0.0/0
 [110/1] via 155.1.108.8, 01:12:56, GigabitEthernet1.108
```

https://t.me/learningnets

Crypto Maps are dial-on-demand IPsec tunnels, so until interesting traffic is sent out the tunnel, both IPsec Phase 1 (ISAKMP) and IPsec Phase 2 Security Associations (SAs) should be down.

```
R7#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status

IPv6 Crypto ISAKMP SA
!R7#show crypto ipsec sa

interface: GigabitEthernet1.67
    Crypto map tag: R7_TO_R8, local addr 150.1.7.7

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (150.1.9.9/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (150.1.10.10/255.255.255.255/0/0)
   current_peer 155.1.58.8 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 150.1.7.7, remote crypto endpt.: 155.1.58.8
     plaintext mtu 1500, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1.67
 current outbound spi: 0x0(0)
     PFS (Y/N): N, DH group: none
 inbound esp sas:

     inbound ah sas:

     inbound pcp sas:
 outbound esp sas:



     outbound ah sas:

     outbound pcp sas:
```

https://t.me/learningnets

Next, R9 initiates traffic over the tunnel, which causes IPsec negotiation to start between R7 and R8.

```
R9#ping 150.1.10.10 source 150.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.10.10, timeout is 2 seconds:
Packet sent with a source address of 150.1.9.9 .!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 8/18/46 ms
!R7#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status155.1.58.8      150.1.7.7QM_IDLE
           1002 ACTIVE

IPv6 Crypto ISAKMP SA
!R7#show crypto ipsec sa

interface: GigabitEthernet1.67Crypto map tag: R7_TO_R8, local addr 150.1.7.7

   protected vrf: (none)   local  ident (addr/mask/prot/port): (150.1.9.9/255.255.255.255/0/0
)   remote ident (addr/mask/prot/port): (150.1.10.10/255.255.255.255/0/0
)current_peer 155.1.58.8
 port 500
     PERMIT, flags={origin_is_acl,}#pkts encaps: 4, #pkts encrypt: 4, #pkts digest: 4
#pkts decaps: 4, #pkts decrypt: 4, #pkts verify: 4
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 150.1.7.7, remote crypto endpt.: 155.1.58.8
     plaintext mtu 1422, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1.67
     current outbound spi: 0x3DDBFE89(1037827721)
     PFS (Y/N): N, DH group: none
 inbound esp sas:
spi: 0x9EE9266E(2666079854)
transform: esp-192-aes esp-sha384-hmac
 ,
        in use settings ={Tunnel, }
        conn id: 2009, flow_id: CSR:9, sibling_flags FFFFFFFF80004048, crypto map: R7_TO_R8
        sa timing: remaining key lifetime (k/sec): (4607999/3557)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     inbound ah sas:
```

https://t.me/learningnets

```
     inbound pcp sas:
  outbound esp sas:
 spi: 0x3DDBFE89(1037827721)
 transform: esp-192-aes esp-sha384-hmac
 ,
        in use settings ={Tunnel, }
        conn id: 2010, flow_id: CSR:10, sibling_flags FFFFFFFF80004048, crypto map: R7_TO_R8
        sa timing: remaining key lifetime (k/sec): (4607999/3557)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

<snip>
```

Note that there is one set of IPsec SAs for each Proxy ACL entry. This means that the SAs will not be negotiated until the traffic is actually sent between the pair of source and destinations defined by each entry of the Proxy ACL. This is one of the scalability limitations of Crypto Map-based configurations; the number of SAs does not scale linearly with the number of tunnel endpoints, as it would in either a GRE over IPsec or IPsec VTI configuration.

```
R8#show crypto ipsec sa | include ident|esp|spi
   local  ident (addr/mask/prot/port): (150.1.10.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (150.1.9.9/255.255.255.255/0/0)
     current outbound spi: 0x9EE9266E(2666079854)
     inbound esp sas:
      spi: 0x3DDBFE89(1037827721)
        transform: esp-192-aes esp-sha384-hmac ,
     outbound esp sas:
      spi: 0x9EE9266E(2666079854)
        transform: esp-192-aes esp-sha384-hmac ,
   local  ident (addr/mask/prot/port): (155.1.10.0/255.255.255.0/0/0)
   remote ident (addr/mask/prot/port): (155.1.9.0/255.255.255.0/0/0)
     current outbound spi: 0x0(0)
     inbound esp sas:
     outbound esp sas:
   local  ident (addr/mask/prot/port): (150.1.10.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (155.1.9.0/255.255.255.0/0/0)
     current outbound spi: 0x0(0)
     inbound esp sas:
     outbound esp sas:   local  ident (addr/mask/prot/port): (155.1.10.0
/255.255.255.0/0/0)   remote ident (addr/mask/prot/port): (150.1.9.9
/255.255.255.255/0/0)
```

https://t.me/learningnets

```
current outbound spi: 0x0(0)
 inbound esp sas:
outbound esp sas:
```

When traffic is actually initiated, additional SAs can form.

```
R10#ping 150.1.9.9 source 155.1.10.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 155.1.10.10 .!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 20/23/24 ms
!R8#show crypto ipsec sa | include ident|esp|spi
   local  ident (addr/mask/prot/port): (150.1.10.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (150.1.9.9/255.255.255.255/0/0)
     current outbound spi: 0x9EE9266E(2666079854)
     inbound esp sas:
      spi: 0x3DDBFE89(1037827721)
        transform: esp-192-aes esp-sha384-hmac ,
     outbound esp sas:
      spi: 0x9EE9266E(2666079854)
        transform: esp-192-aes esp-sha384-hmac ,
   local  ident (addr/mask/prot/port): (155.1.10.0/255.255.255.0/0/0)
   remote ident (addr/mask/prot/port): (155.1.9.0/255.255.255.0/0/0)
     current outbound spi: 0x0(0)
     inbound esp sas:
     outbound esp sas:
   local  ident (addr/mask/prot/port): (150.1.10.10/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (155.1.9.0/255.255.255.0/0/0)
     current outbound spi: 0x0(0)
     inbound esp sas:
     outbound esp sas:   local  ident (addr/mask/prot/port): (155.1.10.0
/255.255.255.0/0/0)   remote ident (addr/mask/prot/port): (150.1.9.9
/255.255.255.255/0/0)current outbound spi: 0x2DFBB8D9(771471577)
 inbound esp sas:
 spi: 0xAA2C811C(2855043356)
 transform: esp-192-aes esp-sha384-hmac
 , outbound esp sas:
 spi: 0x2DFBB8D9(771471577)
 transform: esp-192-aes esp-sha384-hmac
 ,
```

This task also asks that R7's IPsec tunnel be re-routed around a network failure. Because R7 has multiple connections to the transit network (one to R3 and one to

https://t.me/learningnets

R6), the IPsec tunnel should be sourced off an interface independent of these links, such as its Loopback0. This is accomplished by applying the crypto map to all outgoing links, but by setting the crypto map source to be the Loopback0 address.

```
R7#show crypto map

        Interfaces using crypto map NiStTeSt1:
Crypto Map: "R7_TO_R8" idb:Loopback0 local address: 150.1.7.7

Crypto Map IPv4 "R7_TO_R8" 10 ipsec-isakmp
        Peer = 155.1.58.8
        Extended IP access list R9_TO_R10
            access-list R9_TO_R10 permit ip host 150.1.9.9 host 150.1.10.10
            access-list R9_TO_R10 permit ip host 150.1.9.9 155.1.10.0 0.0.0.255
            access-list R9_TO_R10 permit ip 155.1.9.0 0.0.0.255 host 150.1.10.10
            access-list R9_TO_R10 permit ip 155.1.9.0 0.0.0.255 155.1.10.0 0.0.0.255
        Current peer: 155.1.58.8
        Security association lifetime: 4608000 kilobytes/3600 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): N
        Mixed-mode : Disabled
        Transform sets={
                ESP-AES-192-SHA-384:  { esp-192-aes esp-sha384-hmac  } ,
        } Interfaces using crypto map R7_TO_R8:
GigabitEthernet1.67
GigabitEthernet1.37
```

Traffic over the IPsec tunnel should now have convergence that is a function of IGP from R7 to R3 and from R7 to R6. This can be demonstrated as follows.

```
R10#ping 150.1.9.9 source 150.1.10.10 repeat 100000
Type escape sequence to abort.
Sending 100000, 100-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 150.1.10.10
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
<snip>
R7# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface GigabitEthernet1.37
R7(config-subif)#shutdown
%DUAL-5-NBRCHANGE: EIGRP-IPv4 100:Neighbor 155.1.37.3 (GigabitEthernet1.37) is down
: interface down
!R10#
```

https://t.me/learningnets

```
!!!
```

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

GRE over IPsec with Crypto Maps

> You must load the initial configuration files for the section, **IPsec VPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF area 0 between R7 and R9, and advertise all links on R9 into OSPF.
- Configure OSPF area 0 between R8 and R10, and advertise all links on R10 into OSPF.
- Configure a GRE tunnel between R7 and R8 as follows:
  - Source the tunnel from Loopback0 addresses.
  - Use the IP subnet 169.254.78.0/24, with host addresses **.7** and **.8** respectively.
  - Enable OSPF area 0 on the tunnel.

- Configure the above GRE tunnel inside an IPsec tunnel between R7 and R8 as follows:
  - Use an ISAKMP Policy with the following options:
    - Pre-Shared Key: **CISCO**
    - Encryption: 3DES
    - Hash: MD5
    - Diffie-Hellman Group: 5
  - Use a Crypto Map named **GRE_OVER_IPSEC** with the following options:
    - R7 and R8's Loopbacks will be the tunnel endpoints.
    - GRE Traffic from R7 to R8 and vice-versa should be sent inside the IPsec tunnel.
    - Encrypt the traffic using 128-bit AES.
    - Authenticate the traffic using SHA-1.

https://t.me/learningnets

    - Use ESP Transport mode to save additional encapsulation overhead.
  - To prevent the tunnel endpoints from having to do IPsec fragmentation, configure the GRE tunnel IP MTU to 1400 bytes, and set them to adjust the TCP MSS accordingly.
  - Ensure that if R7's links to either R3 or R6 go down, the tunnel remains up and is re-routed via any available alternate path.

Configuration

```
R7:
interface Tunnel0
 ip address 169.254.78.7 255.255.255.0
 ip mtu 1400
 ip tcp adjust-mss 1360
 ip ospf 1 area 0
 tunnel source Loopback0
 tunnel destination 150.1.8.8
!
crypto isakmp policy 10
 encr 3des
 hash md5
 authentication pre-share
 group 5
!
crypto isakmp key CISCO address 150.1.8.8
!
crypto ipsec transform-set ESP-AES-128-SHA-1 esp-aes esp-sha-hmac
 mode transport
!
ip access-list extended GRE_FROM_R7_TO_R8
 permit gre host 150.1.7.7 host 150.1.8.8
!
crypto map GRE_OVER_IPSEC local-address Loopback0
!
crypto map GRE_OVER_IPSEC 10 ipsec-isakmp
 set peer 150.1.8.8
 set transform-set ESP-AES-128-SHA-1
 match address GRE_FROM_R7_TO_R8
!
interface GigabitEthernet1.37
 crypto map GRE_OVER_IPSEC
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.67
 crypto map GRE_OVER_IPSEC
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
```

R8:

```
interface Tunnel0
 ip address 169.254.78.8 255.255.255.0
 ip mtu 1400
 ip tcp adjust-mss 1360
 ip ospf 1 area 0
 tunnel source Loopback0
 tunnel destination 150.1.7.7
!
crypto isakmp policy 10
 encr 3des
 hash md5
 authentication pre-share
 group 5
!
crypto isakmp key CISCO address 150.1.7.7
!
crypto ipsec transform-set ESP-AES-128-SHA-1 esp-aes esp-sha-hmac
 mode transport
!
crypto map GRE_OVER_IPSEC local-address Loopback0
!
ip access-list extended GRE_FROM_R8_TO_R7
 permit gre host 150.1.8.8 host 150.1.7.7
!
crypto map GRE_OVER_IPSEC 10 ipsec-isakmp
 set peer 150.1.7.7
 set transform-set ESP-AES-128-SHA-1
 match address GRE_FROM_R8_TO_R7
!
interface GigabitEthernet1.58
 crypto map GRE_OVER_IPSEC
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
```

R9:

```
interface Loopback0
 ip ospf 1 area 0
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.9
 ip ospf 1 area 0
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
```

R10:

```
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.10
 ip ospf 1 area 0
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
```

Verification

The advantage of combining GRE and IPsec tunnels together is the support for dynamic routing over the IPsec tunnel. The GRE tunnel is treated just like any other point-to-point link from a routing point of view, as seen below.

```
R8#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
150.1.10.10       1   FULL/DR         00:00:36    155.1.108.10    GigabitEthernet1.108
150.1.7.7         0   FULL/  -        00:00:35    169.254.78.7    Tunnel0
!R8#show ip ospf interface tunnel0
Tunnel0 is up, line protocol is up
  Internet Address 169.254.78.8/24, Area 0, Attached via Interface Enable
  Process ID 1, Router ID 150.1.8.8, Network Type POINT_TO_POINT
, Cost: 1000
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1000      no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can not be protected by per-prefix Loop-Free FastReroute
```

https://t.me/learningnets

```
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec Neighbor Count is 1, Adjacent neighbor count is 1
 Adjacent with neighbor 150.1.7.7
  Suppress hello for 0 neighbor(s)
!R8#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 10 subnets O        150.1.9.9 [110/1002] via 169.254.78.7, 00:22:38,
Tunnel0
O        150.1.10.10 [110/2] via 155.1.108.10, 00:22:50, GigabitEthernet1.108
      155.1.0.0/16 is variably subnetted, 18 subnets, 2 masks
O        155.1.9.0/24 [110/1002] via 169.254.78.7, 00:22:38, Tunnel0
O        155.1.10.0/24
           [110/2] via 155.1.108.10, 00:22:50, GigabitEthernet1.108
O        155.1.79.0/24 [110/1001] via 169.254.78.7, 00:22:38, Tunnel0
```

From the devices behind the tunnel endpoints, for example on R10, this removes the requirement of doing static routing or default routing to reach destinations over the IPsec tunnel. From R10's point of view, the remote site is simply comprised of other routers in the same OSPF area.

```
R10#show ip ospf database

            OSPF Router with ID (150.1.10.10) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count 150.1.7.7
       150.1.7.7       894         0x80000008 0x003E28 3 150.1.8.8
       150.1.8.8       1101        0x80000006 0x00A781 3 150.1.9.9
       150.1.9.9       978         0x80000004 0x0086FA 3 150.1.10.10
       150.1.10.10     980         0x80000004 0x00F14C 3
```

https://t.me/learningnets

```
                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
155.1.79.9      150.1.9.9       978         0x80000003 0x003119
155.1.108.10    150.1.10.10     980         0x80000003 0x00081E
!R10#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 2 subnetsO        150.1.9.9
 [110/1003] via 155.1.108.8, 01:22:27, GigabitEthernet1.108
      155.1.0.0/16 is variably subnetted, 6 subnets, 2 masks
O        155.1.9.0/24
           [110/1003] via 155.1.108.8, 01:22:27, GigabitEthernet1.108
O        155.1.79.0/24
           [110/1002] via 155.1.108.8, 01:22:27, GigabitEthernet1.108
      169.254.0.0/24 is subnetted, 1 subnets
O        169.254.78.0
           [110/1001] via 155.1.108.8, 01:22:49, GigabitEthernet1.108
!           R10#ping 150.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/26/63 ms
!R10#traceroute 150.1.9.9
Type escape sequence to abort.
Tracing the route to 150.1.9.9
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.108.8 21 msec 14 msec 5 msec   2 169.254.78.7
 20 msec 35 msec 10 msec
  3 155.1.79.9 59 msec *  24 msec
```

The difference, though, is that traffic is encrypted after it is GRE encapsulated between the tunnel endpoints. Note in the below output that only one set of IPsec SAs is needed, because the Proxy ACL used needs only a single entry to specify all

https://t.me/learningnets

GRE traffic (IP protocol 47). This means that from a scalability point of view, the IPsec tunnel endpoints will be able to scale the control plane better, as there are fewer SAs to maintain vs. the previous crypto map-based configuration. Additionally, note that the ESP transform set is running in Transport mode, which means that an extra GRE IP header is not needed, only the additional ESP IP header.

```
R8#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status150.1.8.8       150.1.7.7QM_IDLE
           1004 ACTIVE
150.1.7.7       150.1.8.8       QM_IDLE           1003 ACTIVE

IPv6 Crypto ISAKMP SA
!R8#show crypto ipsec sa

interface: GigabitEthernet1.58
    Crypto map tag: GRE_OVER_IPSEC, local addr 150.1.8.8

   protected vrf: (none)   local  ident (addr/mask/prot/port): (150.1.8.8/255.255.255.255/47/0
)   remote ident (addr/mask/prot/port): (150.1.7.7/255.255.255.255/47/0
)
   current_peer 150.1.7.7 port 500
     PERMIT, flags={origin_is_acl,}#pkts encaps: 571, #pkts encrypt: 571, #pkts digest: 571
#pkts decaps: 569, #pkts decrypt: 569, #pkts verify: 569
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 150.1.8.8, remote crypto endpt.: 150.1.7.7
     plaintext mtu 1458, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1.58
     current outbound spi: 0x10DC5497(282875031)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0x31FB9561(838571361)transform: esp-aes esp-sha-hmac
 ,        in use settings ={Transport
, }
        conn id: 2017, flow_id: CSR:17, sibling_flags FFFFFFFF80000008, crypto map: GRE_OVER_IPSEC
        sa timing: remaining key lifetime (k/sec): (4607984/1940)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     inbound ah sas:
```

https://t.me/learningnets

```
     inbound pcp sas:

     outbound esp sas:
      spi: 0x10DC5497(282875031) transform: esp-aes esp-sha-hmac
 ,         in use settings ={Transport
, }
        conn id: 2018, flow_id: CSR:18, sibling_flags FFFFFFFF80000008, crypto map: GRE_OVER_IPSEC
        sa timing: remaining key lifetime (k/sec): (4607984/1940)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:
```

Setting the IP MTU on the tunnel interface is used to attempt to offload fragmentation to the end host, and ideally stop the router from having to do fragmentation after IPsec encryption. Because the Don't Fragment (DF) bit is not copied by default from the original IP header to the inner GRE payload and further to the outer ESP header, hosts running Path MTU Discovery (PMTUD) over a GRE over IPsec tunnel will mistakenly think that the end-to-end path MTU is larger than it is and only account for the GRE encapsulation, not both the ESP and GRE encapsulation. By lowering the IP MTU to account for both ESP and GRE, the router will now generate ICMP Unreachable at a lower value when fragmentation is needed, as seen below.

```
R9#ping 155.1.10.10 df-bit size 1400

Type escape sequence to abort.
Sending 5, 1400-byte ICMP Echos to 155.1.10.10, timeout is 2 seconds:
Packet sent with the DF bit set!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/18/37 ms
!R1#debug ip icmp
ICMP packet debugging is on
!R9#ping 155.1.10.10 df-bit size 1401
Type escape sequence to abort.
Sending 5, 1401-byte ICMP Echos to 155.1.10.10, timeout is 2 seconds:
Packet sent with the DF bit setM.M.M
Success rate is 0 percent (0/5)ICMP: dst (155.1.79.9)
frag. needed and DF set unreachable rcv from 155.1.79.7 mtu:1400

ICMP: dst (155.1.79.9) frag. needed and DF set unreachable rcv from 155.1.79.7 mtu:1400
```

https://t.me/learningnets

```
ICMP: dst (155.1.79.9) frag. needed and DF set unreachable rcv from 155.1.79.7 mtu:1400
```

The TCP Adjust MSS feature is used to have the router edit the payload of a TCP three-way handshake if the MSS exceeds the configured value. At a maximum, the MSS should be the IP MTU minus 40 bytes (20 bytes for the IP header, 20 bytes for the TCP header). Below we can see that when devices using a larger MSS have a TCP session through the GRE over IPsec tunnel, the routers edit the value down to the configured 1360.

```
R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#ip tcp mss 1440
!R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#ip tcp mss 1440
!R9#telnet 150.1.10.10
Trying 150.1.10.10 ... Open
!R10#show tcp brief
TCB       Local Address               Foreign Address             (state) 7F48AC597558
  150.1.10.10.23             155.1.79.9.56053            ESTAB
!R10#show tcp tcb 7F48AC597558
Connection state is ESTAB, I/O status: 1, unread input bytes: 1
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 150.1.10.10, Local port: 23
Foreign host: 155.1.79.9, Foreign port: 56053
Connection tableid (VRF): 0
Maximum output segment queue size: 20

Enqueued packets for retransmit: 1, input: 0  mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x4FC084E):
Timer          Starts    Wakeups            Next
Retrans            41          0       0x4FC0C59
TimeWait            0          0             0x0
AckHold            38          1             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0
Linger              0          0             0x0
ProcessQ            0          0             0x0

iss:  268290563  snduna:  268290812  sndnxt:  268290814
irs: 4026563317  rcvnxt: 4026563401

sndwnd:   3880  scale:      0  maxrcvwnd:   4128
```

https://t.me/learningnets

```
rcvwnd:   4045  scale:      0  delrcvwnd:     83

SRTT: 995 ms, RTTO: 1035 ms, RTV: 40 ms, KRTT: 0 ms
minRTT: 14 ms, maxRTT: 1000 ms, ACK hold: 200 ms
Status Flags: passive open, active open
Option Flags: Retrans timeout
IP Precedence value : 6
Datagrams (max data segment is 1360 bytes
):
Rcvd: 69 (out of order: 0), with data: 39, total data bytes: 83
Sent: 61 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 57, total data bytes: 2

 Packets received in fast path: 0, fast processed: 0, slow path: 0
 fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7F48A9E0F520  FREE
```

If we remove the adjust MSS configuration on the GRE tunnels, the MSS of this session will be 1440, as the routers have requested. Sessions that actually use this large value will be fragmented by the GRE tunnel endpoints, which is undesirable.

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface Tunnel0
R7(config-if)#no ip tcp adjust-mss
!R8#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R8(config)#interface Tunnel0
R8(config-if)#no ip tcp adjust-mss
!R9#telnet 150.1.10.10
Trying 150.1.10.10 ... Open
!R10#show tcp brief
TCB       Local Address               Foreign Address             (state)
7F48AC597558  150.1.10.10.23             155.1.79.9.56053            TIMEWAIT 7F48AB770880
  150.1.10.10.23             155.1.79.9.11178 ESTAB
!R10#show tcp tcb 7F48AB770880
Connection state is ESTAB, I/O status: 1, unread input bytes: 1
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 150.1.10.10, Local port: 23
Foreign host: 155.1.79.9, Foreign port: 11178
Connection tableid (VRF): 0
Maximum output segment queue size: 20

Enqueued packets for retransmit: 1, input: 0  mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x4FE58F8):
Timer          Starts    Wakeups            Next
Retrans            34          0       0x4FE5D35
```

https://t.me/learningnets

```
TimeWait            0          0             0x0
AckHold            29          0             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0
Linger              0          0             0x0
ProcessQ            0          0             0x0

iss:   50880082  snduna:   50880389  sndnxt:   50880391
irs: 3317604324  rcvnxt: 3317604398

sndwnd:   3822  scale:      0  maxrcvwnd:   4128
rcvwnd:   4055  scale:      0  delrcvwnd:     73

SRTT: 988 ms, RTTO: 1087 ms, RTV: 99 ms, KRTT: 0 ms
minRTT: 13 ms, maxRTT: 1000 ms, ACK hold: 200 ms
Status Flags: passive open, active open
Option Flags: Retrans timeout
IP Precedence value : 6
 Datagrams (max data segment is 1440 bytes
):
Rcvd: 56 (out of order: 0), with data: 31, total data bytes: 73
Sent: 52 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 48, total data bytes: 3

 Packets received in fast path: 0, fast processed: 0, slow path: 0
 fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7F48A9E0F460  FREE
```

Finally, because the Crypto Map is sourced from the routers' Loopback interfaces, IGP should be able to re-route the payload traffic around a failure of either R7's link to R3 or R7's link to R6.

```
R8#traceroute 150.1.7.7
Type escape sequence to abort.
Tracing the route to 150.1.7.7
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.58.5 11 msec 4 msec 3 msec
  2 155.1.45.4 4 msec 6 msec 5 msec
  3 155.1.146.6 6 msec 14 msec 9 msec   4 155.1.67.7
 6 msec *  5 msec.
!R10#ping 150.1.9.9 repeat 1000
Type escape sequence to abort.
Sending 1000, 100-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:
```

https://t.me/learningnets

```
!!!
```

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

GRE over IPsec with Crypto Profiles

> You must load the initial configuration files for the section, **IPsec VPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF area 0 between R7 and R9, and advertise all links on R9 into OSPF.
- Configure OSPF area 0 between R8 and R10, and advertise all links on R10 into OSPF.
- Configure a GRE tunnel between R7 and R8 as follows:
  - Source the tunnel from Loopback0 addresses.
  - Use the IP subnet 169.254.78.0/24, with host addresses **.7** and **.8**, respectively.
  - Enable OSPF area 0 on the tunnel.

- Configure the above GRE tunnel inside an IPsec tunnel between R7 and R8 as follows:
  - Use an ISAKMP Policy with the following options:
    - Pre-Shared Key: **CISCO**
    - Encryption: 3DES
    - Hash: MD5
    - Diffie-Hellman Group: 5
  - Use a Crypto IPsec Profile named **GRE_OVER_IPSEC_PROFILE** with the following options:
    - Encrypt the traffic using 128-bit AES.
    - Authenticate the traffic using SHA-1.
    - Use ESP Transport mode to save additional encapsulation overhead.

https://t.me/learningnets

- To prevent the tunnel endpoints from having to do IPsec fragmentation, configure the GRE tunnel's IP MTU to 1400 bytes, and set them to adjust the TCP MSS accordingly.
- Ensure that if R7's links to either R3 or R6 go down, the tunnel remains up and is re-routed via any available alternate path.

Configuration

```
R7:
crypto isakmp policy 10
 encr 3des
 hash md5
 authentication pre-share
 group 5
!
crypto isakmp key CISCO address 150.1.8.8
!
crypto ipsec transform-set ESP-AES-128-SHA-1 esp-aes esp-sha-hmac
 mode transport
!
crypto ipsec profile GRE_OVER_IPSEC_PROFILE
 set transform-set ESP-AES-128-SHA-1
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
!
interface Tunnel0
 ip address 169.254.78.7 255.255.255.0
 ip mtu 1400
 ip tcp adjust-mss 1360
 ip ospf 1 area 0
 tunnel source Loopback0
 tunnel destination 150.1.8.8
 tunnel protection ipsec profile GRE_OVER_IPSEC_PROFILE

R8:
crypto isakmp policy 10
 encr 3des
 hash md5
 authentication pre-share
 group 5
!
crypto isakmp key CISCO address 150.1.7.7
```

https://t.me/learningnets

```
!
crypto ipsec transform-set ESP-AES-128-SHA-1 esp-aes esp-sha-hmac
 mode transport
!
crypto ipsec profile GRE_OVER_IPSEC_PROFILE
 set transform-set ESP-AES-128-SHA-1
!
interface Tunnel0
 ip address 169.254.78.8 255.255.255.0
 ip mtu 1400
 ip tcp adjust-mss 1360
 ip ospf 1 area 0
 tunnel source Loopback0
 tunnel destination 150.1.7.7
 tunnel protection ipsec profile GRE_OVER_IPSEC_PROFILE
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
```

R9:

```
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.9
 ip ospf 1 area 0
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
```

R10:

```
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.10
 ip ospf 1 area 0
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
```

Verification

This example is similar to the GRE over IPsec with Crypto Maps task, but instead uses a Crypto IPsec Profile. Functionally these two tasks are identical, but the

https://t.me/learningnets

above configuration is simplified compared to the Crypto Map-based one. In both cases the IPsec Phase 1 (ISAKMP) negotiation is identical.

```
R7#show crypto isakmp policy

Global IKE policy
Protection suite of priority 10 encryption algorithm:   Three key triple DES
hash algorithm:          Message Digest 5
authentication method:   Pre-Shared Key
Diffie-Hellman group:    #5 (1536 bit)
        lifetime:                86400 seconds, no volume limit
!R7#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status 150.1.7.7       150.1.8.8 QM_IDLE
           1005 ACTIVE
150.1.8.8       150.1.7.7       QM_IDLE           1006 ACTIVE
```

For IPsec Phase 2, the IPsec Transform Set is called from a Crypto IPsec Profile.

```
R7#show crypto ipsec profile
IPSEC profile GRE_OVER_IPSEC_PROFILE
        Security association lifetime: 4608000 kilobytes/3600 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): N
        Mixed-mode : Disabled Transform sets
 ={ ESP-AES-128-SHA-1:  { esp-aes esp-sha-hmac  }
  ,
        }
```

The IPsec Profile is then applied to the Tunnel interface. Because the Tunnel already specifies the source and destination, there is no need to set the peer address as in the Crypto Map-based configuration. Additionally, no Proxy ACL is needed, because all GRE traffic between the tunnel endpoints is subject to the IPsec encryption and authentication. These details can be seen from the IPsec Security Association (SA).

```
R7#show crypto ipsec sa

interface: Tunnel0 Crypto map tag: Tunnel0-head-0, local addr 150.1.7.7

   protected vrf: (none)   local  ident (addr/mask/prot/port): (150.1.7.7/255.255.255.255/47/0
)   remote ident (addr/mask/prot/port): (150.1.8.8/255.255.255.255/47/0
)
   current_peer 150.1.8.8 port 500
```

https://t.me/learningnets

```
     PERMIT, flags={origin_is_acl,} #pkts encaps: 144, #pkts encrypt: 144, #pkts digest: 144
#pkts decaps: 143, #pkts decrypt: 143, #pkts verify: 143
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 150.1.7.7, remote crypto endpt.: 150.1.8.8
     plaintext mtu 1458, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1.67
     current outbound spi: 0xAAA867CD(2863163341)
     PFS (Y/N): N, DH group: none
inbound esp sas
:
      spi: 0x4B7DCD67(1266535783) transform: esp-aes esp-sha-hmac
 ,         in use settings ={Transport
, }
        conn id: 2031, flow_id: CSR:31, sibling_flags FFFFFFFF80000008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/2361)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)
      spi: 0x7A84016A(2055471466)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Transport, }
        conn id: 2033, flow_id: CSR:33, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4607988/2363)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:
outbound esp sas
:
      spi: 0x39A52035(967122997) transform: esp-aes esp-sha-hmac
 ,         in use settings ={Transport
, }
        conn id: 2032, flow_id: CSR:32, sibling_flags FFFFFFFF80000008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/2361)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)
```

https://t.me/learningnets

```
      spi: 0xAAA867CD(2863163341)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Transport, }
        conn id: 2034, flow_id: CSR:34, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4607988/2363)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:
```

From a data plane point of view, the two configurations are the same.

```
R10#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 2 subnetsO        150.1.9.9
 [110/1003] via 155.1.108.8, 00:21:22, GigabitEthernet1.108
      155.1.0.0/16 is variably subnetted, 6 subnets, 2 masks
O        155.1.9.0/24
           [110/1003] via 155.1.108.8, 00:21:22, GigabitEthernet1.108
O        155.1.79.0/24
           [110/1002] via 155.1.108.8, 00:21:22, GigabitEthernet1.108
      169.254.0.0/24 is subnetted, 1 subnets
O        169.254.78.0
           [110/1001] via 155.1.108.8, 00:21:22, GigabitEthernet1.108
!R10#traceroute 150.1.9.9
Type escape sequence to abort.
Tracing the route to 150.1.9.9
VRF info: (vrf in name/id, vrf out name/id)
  1 155.1.108.8 174 msec 4 msec 5 msec  2 169.254.78.7
 15 msec 122 msec 55 msec
```

https://t.me/learningnets

```
  3 155.1.79.9 19 msec *  101 msec
!R10#ping 150.1.9.9 df-bit size 1400
Type escape sequence to abort.
Sending 5, 1400-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:
Packet sent with the DF bit set!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 15/30/48 ms
!R10#ping 150.1.9.9 df-bit size 1401
Type escape sequence to abort.
Sending 5, 1401-byte ICMP Echos to 150.1.9.9, timeout is 2 seconds:
Packet sent with the DF bit setM.M.M
Success rate is 0 percent (0/5)
!R9#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R9(config)#ip tcp mss 1440
!R10#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R10(config)#ip tcp mss 1440
!R10#telnet 150.1.9.9
Trying 150.1.9.9 ... Open
!R9#show tcp brief
TCB       Local Address               Foreign Address             (state)7F641E68A188
  150.1.9.9.23               155.1.108.10.31728          ESTAB
!R9#show tcp tcb 7F641E68A188
Connection state is ESTAB, I/O status: 1, unread input bytes: 1
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 150.1.9.9, Local port: 23
Foreign host: 155.1.108.10, Foreign port: 31728
Connection tableid (VRF): 0
Maximum output segment queue size: 20

Enqueued packets for retransmit: 6, input: 0  mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x5F6B338):
Timer          Starts    Wakeups            Next
Retrans            48          0       0x5F6B6B6
TimeWait            0          0             0x0
AckHold            39          0             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0
Linger              0          0             0x0
ProcessQ            0          0             0x0

iss: 3418857448  snduna: 3418861416  sndnxt: 3418861423
irs: 1196060148  rcvnxt: 1196060238
```

https://t.me/learningnets

```
sndwnd:   3565  scale:      0  maxrcvwnd:   4128
rcvwnd:   4039  scale:      0  delrcvwnd:     89

SRTT: 998 ms, RTTO: 1014 ms, RTV: 16 ms, KRTT: 0 ms
minRTT: 12 ms, maxRTT: 1000 ms, ACK hold: 200 ms
Status Flags: passive open, active open
Option Flags: Retrans timeout
IP Precedence value : 6
 Datagrams (max data segment is 1360 bytes
):
Rcvd: 76 (out of order: 0), with data: 41, total data bytes: 89
Sent: 74 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 70, total data bytes: 3

 Packets received in fast path: 0, fast processed: 0, slow path: 0
 fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7F6424CAA460  FREE
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

IPsec Virtual Tunnel Interfaces (VTIs)

> You must load the initial configuration files for the section, **IPsec VPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Configure OSPF area 0 between R7 and R9, and advertise all links on R9 into OSPF.
- Configure OSPF area 0 between R8 and R10, and advertise all links on R10 into OSPF.
- Configure an IPsec VTI based tunnel between R7 and R8 as follows:
  - Use an ISAKMP Policy with the following options:
    - Pre-Shared Key: **CISCO**
    - Encryption: 192 Bit AES
    - Hash: 384 Bit SHA
    - Diffie-Hellman Group: 3072 Bit
  - Use a Crypto IPsec Profile named **VTI_PROFILE** with the following options:
    - Encrypt the traffic using 3DES.
    - Authenticate the traffic using MD5.
  - Source the tunnel from Loopback0 addresses.
  - Use the IP subnet 169.254.78.0/24, with host addresses **.7** and **.8**, respectively.
  - Enable OSPF area 0 on the tunnel.
  - Configure the R7 and R8 to adjust the TCP MSS of sessions transiting the tunnel accordingly.
  - Ensure that if R7's links to either R3 or R6 go down, the tunnel remains up and is re-routed via any available alternate path.

https://t.me/learningnets

Configuration

```
R7:
crypto isakmp policy 10
 encr aes 192
 hash sha384
 authentication pre-share
 group 15
!
crypto isakmp key CISCO address 150.1.8.8
!
crypto ipsec transform-set ESP-3DES-ESP-MD5 esp-3des esp-md5-hmac
 mode tunnel
!
crypto ipsec profile VTI_PROFILE
 set transform-set ESP-3DES-ESP-MD5
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
!
interface Tunnel0
 ip address 169.254.78.7 255.255.255.0
 ip tcp adjust-mss 1406
 ip ospf 1 area 0
 tunnel source Loopback0
 tunnel destination 150.1.8.8
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile VTI_PROFILE

R8:
crypto isakmp policy 10
 encr aes 192
 hash sha384
 authentication pre-share
 group 15
!
crypto isakmp key CISCO address 150.1.7.7
!
crypto ipsec transform-set ESP-3DES-ESP-MD5 esp-3des esp-md5-hmac
 mode tunnel
!
crypto ipsec profile VTI_PROFILE
 set transform-set ESP-3DES-ESP-MD5
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
!
interface Tunnel0
 ip address 169.254.78.8 255.255.255.0
 ip tcp adjust-mss 1406
 ip ospf 1 area 0
 tunnel source Loopback0
 tunnel destination 150.1.7.7
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile VTI_PROFILE
```

R9:

```
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.9
 ip ospf 1 area 0
!
interface GigabitEthernet1.79
 ip ospf 1 area 0
```

R10:

```
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.10
 ip ospf 1 area 0
!
interface GigabitEthernet1.108
 ip ospf 1 area 0
```

Verification

https://t.me/learningnets

An IPsec Virtual Tunnel Interface (VTI) is a tunnel where the payload is directly encapsulated in ESP without the need of another transport header. A VTI behaves very similar to a GRE tunnel, except the encapsulation overhead is lower (24 bytes lower that GRE over IPsec with ESP in Tunnel Mode). One key point to remember though is that since the payload is directly encapsulated into IPsec, which is an IP-only encapsulation, other non-IP payloads are not supported. This means that non-IP protocols like the IS-IS routing protocol could not run over an IPv4 VTI.

The configuration of a VTI is identical to a GRE over IPsec tunnel with a Crypto IPsec Profile, except that the tunnel mode is set to IPsec IPv4 or IPsec IPv6. Phase 1 negotiation happens in the same manner as any other IPsec LAN-to-LAN tunnel. The difference is that in Phase 2, the Proxy Identities (what would normally be configured as the Proxy ACL) are automatically negotiated as IP any any. This reduces the number of IPsec SAs that are needed to maintain the tunnel, and adds scalability to the control plane. Additionally, the IPsec Transform Set must run in Tunnel Mode, because there is no other transport header such as GRE that adds additional IP encapsulation.

```
R7#show crypto isakmp policy

Global IKE policy
Protection suite of priority 10
encryption algorithm:   AES - Advanced Encryption Standard (192 bit keys)
. hash algorithm:          Secure Hash Standard 2 (384 bit)
authentication method:  Pre-Shared Key
Diffie-Hellman group:   #15 (3072 bit)
        lifetime:               86400 seconds, no volume limit
!R7#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status150.1.8.8       150.1.7.7QM_IDLE
           1011 ACTIVE

IPv6 Crypto ISAKMP SA
!R7#show crypto ipsec sa

interface: Tunnel0Crypto map tag: Tunnel0-head-0, local addr 150.1.7.7

   protected vrf: (none)   local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0
)   remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0
)
   current_peer 150.1.8.8 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 827, #pkts encrypt: 827, #pkts digest: 827
    #pkts decaps: 865, #pkts decrypt: 865, #pkts verify: 865
    #pkts compressed: 0, #pkts decompressed: 0
```

https://t.me/learningnets

```
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 150.1.7.7, remote crypto endpt.: 150.1.8.8
     plaintext mtu 1446, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1.67
     current outbound spi: 0x606A94A9(1617597609)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0x7FC0CDE6(2143342054) transform: esp-3des esp-md5-hmac
 ,         in use settings ={Tunnel
, }
        conn id: 2035, flow_id: CSR:35, sibling_flags FFFFFFFF80000048, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/673)
        IV size: 8 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)
      spi: 0x48F728B(76509835)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Tunnel, }
        conn id: 2037, flow_id: CSR:37, sibling_flags FFFFFFFF80004048, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4607915/678)
        IV size: 8 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x58D816D1(1490556625) transform: esp-3des esp-md5-hmac
 ,         in use settings ={Tunnel
, }
        conn id: 2036, flow_id: CSR:36, sibling_flags FFFFFFFF80000048, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/673)
        IV size: 8 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)
      spi: 0x606A94A9(1617597609)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Tunnel, }
```

https://t.me/learningnets

```
        conn id: 2038, flow_id: CSR:38, sibling_flags FFFFFFFF80004048, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4607920/678)
        IV size: 8 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     outbound ah sas:


     outbound pcp sas:
```

One interesting difference about a VTI-based tunnel vs. a GRE over IPsec tunnel is that the tunnel interface can now automatically account for the ESP header in its MTU. This can be seen in the output below.

```
R7# show interface tunnel0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 169.254.78.7/24
  MTU 17838 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 150.1.7.7 (Loopback0), destination 150.1.8.8
   Tunnel Subblocks:
       src-track:
          Tunnel0 source tracking subblock associated with Loopback0
           Set of tunnels with source Loopback0, 1 member (includes iterators), on interface <OK>
Tunnel protocol/transport IPSEC/IP
  Tunnel TTL 255 Tunnel transport MTU 1446 bytes

  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "VTI_PROFILE")
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 00:55:00
  Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     898 packets input, 89829 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
```

https://t.me/learningnets

```
     860 packets output, 84291 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
```

If the tunnel were changed back to normal GRE encapsulation, we would see this max payload size change. Note that the below MTU is incorrect because it does not account for the ESP overhead as the VTI does.

```
R7#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.R7(config)#interface Tunnel0
R7(config-if)#no tunnel mode
      R7(config-if)#do show interface Tunnel0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 169.254.78.7/24
  MTU 17868 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 150.1.7.7 (Loopback0), destination 150.1.8.8
   Tunnel Subblocks:
      src-track:
         Tunnel0 source tracking subblock associated with Loopback0
          Set of tunnels with source Loopback0, 1 member (includes iterators), on interface <OK>
Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255, Fast tunneling enabled Tunnel transport MTU 1476 bytes

  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "VTI_PROFILE")
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 00:56:04
  Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     906 packets input, 90408 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     869 packets output, 84950 bytes, 0 underruns
```

https://t.me/learningnets

```
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
```

The result of this is that you do not need to issue the **`ip mtu`** command at a VTI-based tunnel interface, because the tunnel automatically accounts for its own overhead. It is, however, still a good idea to issue the **`ip tcp adjust-mss`** to force the router to edit TCP 3-way handshakes that use an MSS larger than the tunnel can support. In this case, because the VTI usable payload is 1446 bytes, the TCP MSS should be no larger than 1406 (1446 minus 20 bytes for IP overhead and minus 20 bytes for TCP overhead).

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

DMVPN without IPsec

> You must load the initial configuration files for the section, **DMVPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a DMVPN network between R1 - R5 as follows:
  - R1 - R4 are the DMVPN spokes.
  - R5 is the DMVPN Hub, and the NHRP Next-Hop Server (NHS).
  - Create interface Tunnel0 as a multipoint GRE tunnel.
  - Source the tunnel from the routers' GigabitEthernet1.100 interface.
  - Use IP addressing in the format 155.1.0.Y/24, where *Y* is the router number.
  - Use an NHRP network ID of **1**.
  - Use an NHRP authentication string of **NHRPAUTH**.
  - Use GRE tunnel key of **2**.
  - Ensure that the spokes can send multicast traffic to the hub, and vice versa.
- Configure IGP routing over the DMVPN tunnel as follows:
  - Enable RIPv2 on the DMVPN tunnel on R1 - R5.
  - All links should be passive interfaces except the DMVPN tunnel.
  - Advertise the routers' Loopback0 networks into RIP.
  - Configure R5 to advertise a default route via RIPv2 to the DMVPN spokes.
- When complete, ensure that R1 - R5 can reach each other's Loopback0 networks over the DMVPN network.

Configuration

```
R1:
interface Tunnel0
 ip address 155.1.0.1 255.255.255.0
```

https://t.me/learningnets

```
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
```

R2:

```
interface Tunnel0
 ip address 155.1.0.2 255.255.255.0
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
```

R3:

```
interface Tunnel0
 ip address 155.1.0.3 255.255.255.0
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
```

https://t.me/learningnets

```
 ip nhrp nhs 155.1.0.5
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
```

R4:

```
interface Tunnel0
 ip address 155.1.0.4 255.255.255.0
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
```

R5:

```
interface Tunnel0
 ip address 155.1.0.5 255.255.255.0
 ip nhrp authentication NHRPAUTH
 ip nhrp map multicast dynamic
 ip nhrp network-id 1
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 no shutdown
```

https://t.me/learningnets

```
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
 default-information originate
```

Verification

Dynamic Multipoint VPN (DMVPN) is a multipoint GRE-based tunneling technology that behaves in many ways like a legacy Frame Relay or ATM hub-and-spoke network. DMVPN consists of one or more hub routers that are configured as Next-Hop Resolution Protocol (NHRP) Next-Hop Servers (NHS). Next-Hop Servers, or simply hubs, are used to create mappings between the public IP address used for the tunnel source, called the NBMA address, and the private IP address used inside of the tunnel, which is simply called the tunnel address. This NHRP mapping is loosely analogous to how Frame Relay and ATM maintained Layer 2 circuit to Layer 3 address mappings through protocols such as Inverse ARP.

The goal of the DMVPN network is to allow any-to-any communication over the private tunnel network, while at the same time not requiring that a full mesh of point-to-point tunnels be formed. Instead, tunnels can be formed on-demand based on the particular destination of traffic. This allows DMVPN designs to achieve very large scalability, because a full mesh of n*(n-1)/2 tunnels is not required.

From a configuration point of view, there are two roles in the DMVPN network, the hub(s) and the spoke(s). The hub and spokes must agree on certain parameters, such as NHRP authentication, GRE tunnel key number, and whether multicast will be supported. The spokes maintain a manual/static mapping of the hub's tunnel address to NBMA address, while the hub dynamically learns about the spokes through NHRP messages. From a verification point of view, this should be one of the first steps in checking a DMVPN configuration, which is to ensure that the spokes have registered with the hub.

```
R5#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
=====================================================================
```

https://t.me/learningnets

```
Interface: Tunnel0, IPv4 NHRP Details
Type:Hub, NHRP Peers:4,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----      1 169.254.100.1          155.1.0.1
    UP 00:29:24     D      1 169.254.100.2          155.1.0.2
    UP 00:29:17     D      1 169.254.100.3          155.1.0.3
    UP 00:29:09     D      1 169.254.100.4          155.1.0.4
    UP 00:29:26     D
!R5#show ip nhrp
 155.1.0.1/32 via 155.1.0.1
   Tunnel0 created 00:32:02, expire 01:27:57 Type: dynamic
, Flags: unique registered used nhop NBMA address: 169.254.100.1
155.1.0.2/32 via 155.1.0.2
   Tunnel0 created 00:31:55, expire 01:28:04 Type: dynamic
, Flags: unique registered used nhop NBMA address: 169.254.100.2
 155.1.0.3/32 via 155.1.0.3
   Tunnel0 created 00:31:47, expire 01:28:12 Type: dynamic
, Flags: unique registered used nhop NBMA address: 169.254.100.3
155.1.0.4/32 via 155.1.0.4
   Tunnel0 created 00:32:04, expire 01:27:55 Type: dynamic
, Flags: unique registered used nhop NBMA address: 169.254.100.4
```

The spokes, however, should have this resolution configured statically. This is analogous to a legacy `frame-relay map` command.

```
R1#show run int tunnel0 | in map
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
!R1#show ip nhrp
155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 00:36:11, never expire Type: static
, Flags: used NBMA address: 169.254.100.5
```

From a data plane encapsulation point of view, R5 knows that the outer IPv4 address of the GRE tunnel going toward 155.1.0.1 should be 169.254.100.1, because this information is provided by NHRP. What this allows DMVPN configurations to do is to not have to manually specify the `tunnel destination` , which is why this design is termed a "Dynamic Multipoint" tunnel. At this point, the hub and spokes should have IPv4 reachability to each other, as the NHRP mappings have

https://t.me/learningnets

been successfully formed.

```
R5#ping 155.1.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/9/26 ms
!R5#ping 155.1.0.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/21/87 ms
!R5#ping 155.1.0.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.3, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/8 ms
!R5#ping 155.1.0.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 155.1.0.4, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 6/13/26 ms
```

Note that although DMVPN is a multipoint tunnel, it is not a **multicast** tunnel. This means that the GRE source IP address and the GRE destination IP address are always unicast. Multicast is, however, supported inside of the tunnel, but essentially as a replicated unicast, similar to how multicast was supported over legacy Frame Relay or ATM PVCs. Multicast support is added to DMVPN by the spokes manually mapping multicast to the hub's NBMA address with the `ip nhrp map multicast` command, whereas the hub maps multicast as `dynamic`. This is analogous to the `broadcast` keyword at the end of a Frame Relay or ATM mapping statement.

The implication of how multicast support works over DMVPN can be seen from how routing protocols behave over DMVPN. Specifically, from an IGP routing point of view, the spokes only learn routes from the hub. This is because, like in Frame Relay or ATM hub-and-spoke networks, multicasts cannot be directly replicated between the spokes. In other words, if you were to enable EIGRP over DMVPN, the spokes would only become EIGRP adjacent with the hub, simply because the spokes do not see each other’s hello packets.

In this specific design, RIPv2 routing is used, with split-horizon enabled on the hub's tunnel interface (the default behavior). However, because the hub is advertising a default route to all the spokes, they do not need the specific routes about each other. This is one simple design technique that can help to scale DMVPN networks.

```
R1#show ip route rip
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
```

https://t.me/learningnets

```
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is 155.1.0.5 to network 0.0.0.0
R*    0.0.0.0/0 [120/1] via 155.1.0.5, 00:00:10, Tunnel0
      150.1.0.0/32 is subnetted, 2 subnets
R        150.1.5.5 [120/1] via 155.1.0.5, 00:00:10, Tunnel0
      155.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
R        155.1.5.0/24 [120/1] via 155.1.0.5, 00:00:10, Tunnel0
R        155.1.45.0/24 [120/1] via 155.1.0.5, 00:00:10, Tunnel0
R        155.1.58.0/24 [120/1] via 155.1.0.5, 00:00:10, Tunnel0
!R5#show ip route rip
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnetsR        150.1.1.1 [120/1] via 155.1.0.1, 00:00:13, Tunnel0
R        150.1.2.2 [120/1] via 155.1.0.2, 00:00:03, Tunnel0
R        150.1.3.3 [120/1] via 155.1.0.3, 00:00:09, Tunnel0
R        150.1.4.4 [120/1] via 155.1.0.4, 00:00:21, Tunnel0

      155.1.0.0/16 is variably subnetted, 12 subnets, 2 masks
R        155.1.13.0/24 [120/1] via 155.1.0.3, 00:00:09, Tunnel0
                       [120/1] via 155.1.0.1, 00:00:13, Tunnel0
R        155.1.23.0/24 [120/1] via 155.1.0.3, 00:00:09, Tunnel0
                       [120/1] via 155.1.0.2, 00:00:03, Tunnel0
R        155.1.37.0/24 [120/1] via 155.1.0.3, 00:00:09, Tunnel0
R        155.1.146.0/24 [120/1] via 155.1.0.4, 00:00:21, Tunnel0
                        [120/1] via 155.1.0.1, 00:00:13, Tunnel0
```

From a traffic forwarding point of view, this network now behaves just like legacy

https://t.me/learningnets

Frame Relay hub-and-spoke, where traffic goes from spoke to hub to spoke.

```
R1#show ip cef 150.1.2.2
0.0.0.0/0
 nexthop 155.1.0.5 Tunnel0
!R1#ping 150.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/14/45 ms
!R1#traceroute 150.1.2.2
Type escape sequence to abort.
Tracing the route to 150.1.2.2
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.0.5
 3 msec 6 msec 15 msec  2 155.1.0.2
 10 msec *  15 msec
```

This type of design is called "DMVPN Phase 1," because traffic cannot be directly exchanged between the spokes. Technically, "DMVPN Phase 1" uses point-to-point GRE tunnels on the spokes because there is no spoke-to-spoke dynamic tunnel negotiated, as we'll see in following labs. Depending on the traffic patterns of the network (for example, if there is a lot of spoke-to-spoke traffic), this design may not be desirable. In later tasks we will see how this problem is solved with both "DMVPN Phase 2" and "DMVPN Phase 3" configurations.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

DMVPN with IPsec

> You must load the initial configuration files for the section, **DMVPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a DMVPN network between R1 - R5 as follows:
  - R1 - R4 are the DMVPN spokes.
  - R5 is the DMVPN Hub, and the NHRP Next-Hop Server (NHS).
  - Create interface Tunnel0 as a multipoint GRE tunnel.
  - Source the tunnel from the routers' GigabitEthernet1.100 interface.
  - Use IP addressing in the format 155.1.0.Y/24, where *Y* is the router number.
  - Use an NHRP network ID of **1**.
  - Use an NHRP authentication string of **NHRPAUTH**.
  - Use GRE tunnel key of **2**.
  - Ensure that the spokes can send multicast traffic to the hub, and vice versa.
  - To prevent the tunnel endpoints from having to do IPsec fragmentation, configure the GRE tunnel's IP MTU to 1400 bytes, and set them to adjust the TCP MSS accordingly.
- Configure IGP routing over the DMVPN tunnel as follows:
  - Enable RIPv2 on the DMVPN tunnel on R1 - R5.
  - All links should be passive interfaces except the DMVPN tunnel.
  - Advertise the routers' Loopback0 networks into RIP.
  - Configure R5 to advertise a default route via RIPv2 to the DMVPN spokes.
- Configure IPsec over the DMVPN tunnels as follows:
  - Use an ISAKMP Policy with the following options:
    - Pre-Shared Key: **DMVPN_PSK**
    - Encryption: AES 128 Bit

https://t.me/learningnets

        - Hash: SHA 256 Bit
        - Diffie-Hellman Group: 16
        - R5 should use a single Pre-Shared Key for all DMVPN peers.
    - Use a Crypto IPsec Profile named **DMVPN_PROFILE** with the following options:
        - Encrypt the traffic using AES 256 Bit.
        - Authenticate the traffic using SHA 512 Bit.
        - Use ESP Transport mode to save additional encapsulation overhead.
- When complete, ensure that R1 - R5 can reach each other's Loopback0 networks over the DMVPN network.

Configuration

```
R1:
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.1 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
```

https://t.me/learningnets

```
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
```

R2:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.2 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
```

https://t.me/learningnets

```
 no passive-interface Tunnel0
```

R3:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.3 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
```

R4:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
```

https://t.me/learningnets

```
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.4 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
```

R5:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
```

https://t.me/learningnets

```
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.5 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map multicast dynamic
 ip nhrp network-id 1
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router rip
 version 2
 no auto-summary
 network 150.1.0.0
 network 155.1.0.0
 passive-interface default
 no passive-interface Tunnel0
 default-information originate
```

Verification

This example illustrates a more common implementation of DMVPN, where IPsec is used on top of the GRE encapsulation to provide encryption. Note that technically, DMVPN does not imply encryption, because DMVPN is an overlay tunneling technique, not an encryption technique. However, in real-world deployments it is commonly assumed that DMVPN includes IPsec.

DMVPN with IPsec uses the Crypto Profile-based configurations similar to IPsec VTI tunnels. This is because using a crypto map would not scale, because of its need to `set peer`, while DMVPN determines the tunnel peer dynamically based on the final destination of traffic inside the tunnel, hence the name *Dynamic Multipoint VPN*.

In DMVPN Phase 1 with IPsec, the spokes only form IPsec tunnels with the hub. Therefore, the first verification technique for DMVPN with IPsec is to check that the hub has both IPsec Phase 1 (ISAKMP) Security Associations (SAs) with the spokes, as well as IPsec phase 2 SAs, as seen below.

```
R5#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
```

https://t.me/learningnets

```
169.254.100.5 169.254.100.4    QM_IDLE
          1007 ACTIVE 169.254.100.5 169.254.100.2    QM_IDLE
          1005 ACTIVE 169.254.100.5 169.254.100.3    QM_IDLE
          1006 ACTIVE 169.254.100.5 169.254.100.1    QM_IDLE
          1008 ACTIVE
!   R5#show crypto ipsec sa peer 169.254.100.1

interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 169.254.100.5

   protected vrf: (none)   local  ident (addr/mask/prot/port): (169.254.100.5/255.255.255.255/47/0
)   remote ident (addr/mask/prot/port): (169.254.100.1/255.255.255.255/47/0
)
   current_peer 169.254.100.1 port 500
     PERMIT, flags={origin_is_acl,} #pkts encaps: 152, #pkts encrypt: 152, #pkts digest: 152
 #pkts decaps: 154, #pkts decrypt: 154, #pkts verify: 154
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 169.254.100.5, remote crypto endpt.: 169.254.100.1
     plaintext mtu 1442, path mtu 1500, ip mtu 1500, ip mtu idb (none)
     current outbound spi: 0x8E09A92D(2382997805)
     PFS (Y/N): N, DH group: none
 inbound esp sas
:
      spi: 0xB6F043A2(3069199266) transform: esp-256-aes esp-sha512-hmac
 ,         in use settings ={Transport
, }
        conn id: 2153, flow_id: CSR:153, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4607997/2854)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:
 outbound esp sas
:
      spi: 0x8E09A92D(2382997805) transform: esp-256-aes esp-sha512-hmac
 ,         in use settings ={Transport
, }
        conn id: 2154, flow_id: CSR:154, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
```

https://t.me/learningnets

```
        sa timing: remaining key lifetime (k/sec): (4607996/2854)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

    outbound ah sas:

    outbound pcp sas:
```

The IPsec Proxy Identities (Proxy ACL) include the exact tunnel source and destination, as well as GRE (IP protocol #47) as the payload. This is essentially the same as the previous GRE over IPsec examples. Also note that the IPsec Transform Set for ESP is set to Transport Mode instead of Tunnel Mode. Like previous GRE over IPsec examples, this saves an additional 20 bytes of encapsulation overhead, because having both an inner GRE header and outer ESP header with the same IP source and destination addresses pairs would be redundant. There is one corner case DMVPN design where IPsec ESP Tunnel Mode would be required, which is called Dual Tier Headend, but it is not a common deployment model.

Only when the IPsec negotiation is complete can the hub and spokes communicate with NHRP, as the NHRP runs inside the GRE tunnel.

```
R5#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel0, IPv4 NHRP Details
Type:Hub, NHRP Peers:4,
 # Ent  Peer NBMA Addr Peer Tunnel Add
 State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----     1 169.254.100.1 155.1.0.1
    UP 02:34:19     D     1 169.254.100.2 155.1.0.2
    UP 05:31:48     D     1 169.254.100.3 155.1.0.3
    UP 05:31:45     D     1 169.254.100.4 155.1.0.4
    UP 05:31:42     D
```

In this example the DMVPN hub, R5, uses a wildcard ISAKMP Pre-Shared Key.

https://t.me/learningnets

This allows it to authenticate all of the spokes without manually maintaining a separate password for each of them. In a real deployment, this is a bad design choice, because if one spoke is compromised, the entire network is compromised. A better and more scalable design choice for this deployment is to use PKI with certificates, which means that spokes can have their certificates automatically revoked on the Certificate Authority if there is a problem.

```
R5#show crypto isakmp key
Keyring      Hostname/Address                          Preshared Key
default 0.0.0.0          [0.0.0.0]
                    DMVPN_PSK
```

Because this network is configured for DMVPN Phase 1, all spoke-to-spoke traffic must transit the hub. Technically, "DMVPN Phase 1" uses point-to-point GRE tunnels on the spokes because there is no spoke-to-spoke dynamic tunnel negotiated, as we'll see in following labs. Additionally, this means that IPsec tunnels are not formed on-demand between spokes. For networks with a large amount of spoke to spoke traffic, DMVPN Phase 1 is not a good design choice, because the hub must maintain not only the control plane but also the data plane.

```
R1#show ip route rip
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 155.1.0.5 to network 0.0.0.0
R*    0.0.0.0/0 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
      150.1.0.0/32 is subnetted, 2 subnets
R        150.1.5.5 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
      155.1.0.0/16 is variably subnetted, 9 subnets, 2 masks
R        155.1.5.0/24 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.45.0/24 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
R        155.1.58.0/24 [120/1] via 155.1.0.5, 00:00:13, Tunnel0
!R1#show ip cef 150.1.3.3
0.0.0.0/0 nexthop 155.1.0.5 Tunnel0
!R1#traceroute 150.1.3.3
Type escape sequence to abort.
Tracing the route to 150.1.3.3
```

https://t.me/learningnets

```
VRF info: (vrf in name/id, vrf out name/id)   1 155.1.0.5
 12 msec 3 msec 1 msec   2 155.1.0.3
 5 msec *  45 msec
!R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status 169.254.100.5   169.254.100.1   QM_IDLE
           1003 ACTIVE


IPv6 Crypto ISAKMP SA
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

DMVPN Phase 1 with EIGRP

> You must load the initial configuration files for the section, **DMVPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a DMVPN Phase 1 network between R1 - R5 as follows:
  - R1 - R4 are the DMVPN spokes.
  - R5 is the DMVPN Hub, and the NHRP Next-Hop Server (NHS).
  - Source the tunnel from the routers' GigabitEthernet1.100 interface.
  - Use IP addressing in the format 155.1.0.Y/24, where *Y* is the router number.
  - Use an NHRP network ID of **1**.
  - Use an NHRP authentication string of **NHRPAUTH**.
  - Use GRE tunnel key of **2**.
  - Ensure that the spokes can send multicast traffic to the hub, and vice versa.
  - To prevent the tunnel endpoints from having to do IPsec fragmentation, configure the GRE tunnel's IP MTU to 1400 bytes, and set them to adjust the TCP MSS accordingly.
- Configure IGP routing over the DMVPN tunnel as follows:
  - Enable EIGRP in Multi-AF mode using AS 100 on the DMVPN tunnel on R1 - R5.
  - All links should be passive interfaces except the DMVPN tunnel.
  - Advertise the routers' Loopback0 networks into EIGRP.
  - Disable split-horizon for EIGRP on the DMVPN tunnel interface of R5.
- Configure IPsec over the DMVPN tunnels as follows:
  - Use an ISAKMP Policy with the following options:
    - Pre-Shared Key: **DMVPN_PSK**
    - Encryption: AES 128 Bit

https://t.me/learningnets

        - Hash: SHA 256 Bit
        - Diffie-Hellman Group: 16
        - R5 should use a single Pre-Shared Key for all DMVPN peers.
    - Use a Crypto IPsec Profile named **DMVPN_PROFILE** with the following options:
        - Encrypt the traffic using AES 256 Bit.
        - Authenticate the traffic using SHA 512 Bit.
        - Use ESP Transport mode to save additional encapsulation overhead.
- When complete, ensure that R1 - R5 can reach each other's Loopback0 networks over the DMVPN network.

Configuration

```
R1:
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.1 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
```

https://t.me/learningnets

```
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R2:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.2 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
```

https://t.me/learningnets

```
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R3:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.3 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
```

https://t.me/learningnets

```
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R4:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.4 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
```

https://t.me/learningnets

```
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R5:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.5 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
```

https://t.me/learningnets

```
 ip nhrp map multicast dynamic
 ip nhrp network-id 1
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
   no split-horizon
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

Verification

Because EIGRP is a Distance Vector-based protocol like RIP, it uses the Split Horizon principle as part of its loop prevention. This means to advertise routes between spokes, split horizon must be disabled on the hub. In DMVPN Phase 1 with EIGRP as an IGP, the DMVPN hub forms both IPsec tunnels and EIGRP adjacencies with the spokes. Like the previous RIPv2 over DMVPN example, spokes do not form IGP adjacencies nor do they form direct IPsec tunnels, in this case because spokes are using point-to-point GRE tunnels. All spoke-to-spoke traffic will always transit through the hub.

```
R1#show eigrp address-family ipv4 100 neighbors
EIGRP-IPv4 VR(DMVPN) Address-Family Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num0 155.1.0.5
Tu0
                                                   10 00:03:10   70  1398  0  46
```

https://t.me/learningnets

```
!R5#show eigrp address-family ipv4 100 neighbors

EIGRP-IPv4 VR(DMVPN) Address-Family Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num 3 155.1.0.3
Tu0
                      12 00:03:10   34  1398  0  38 2 155.1.0.2 Tu0
                      11 00:03:10   15  1398  0  30 1 155.1.0.4 Tu0
                      11 00:03:10   20  1398  0  34 0 155.1.0.1 Tu0
                      14 00:03:10   12  1398  0  37
!R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status 169.254.100.5   169.254.100.1 QM_IDLE
           1003 ACTIVE
R5#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status 169.254.100.5 169.254.100.4   QM_IDLE
           1013 ACTIVE 169.254.100.5 169.254.100.2   QM_IDLE
           1014 ACTIVE 169.254.100.5 169.254.100.3   QM_IDLE
           1015 ACTIVE 169.254.100.5 169.254.100.1   QM_IDLE
           1012 ACTIVE
!R1#show ip route eigrp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets
D        150.1.2.2 [90/102400640] via 155.1.0.5, 00:03:43, Tunnel0
D        150.1.3.3 [90/102400640] via 155.1.0.5, 00:03:43, Tunnel0
D        150.1.4.4 [90/102400640] via 155.1.0.5, 00:03:43, Tunnel0
D        150.1.5.5 [90/76800640] via 155.1.0.5, 00:03:43, Tunnel0
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks
D        155.1.5.0/24 [90/76805120] via 155.1.0.5, 00:03:43, Tunnel0
D        155.1.23.0/24 [90/102405120] via 155.1.0.5, 00:03:43, Tunnel0
D        155.1.37.0/24 [90/102405120] via 155.1.0.5, 00:03:43, Tunnel0
D        155.1.45.0/24 [90/76805120] via 155.1.0.5, 00:03:43, Tunnel0
D        155.1.58.0/24 [90/76805120] via 155.1.0.5, 00:03:43, Tunnel0
!R5#show ip route eigrp
```

https://t.me/learningnets

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets
D        150.1.1.1 [90/76800640] via 155.1.0.1, 00:03:42, Tunnel0
D        150.1.2.2 [90/76800640] via 155.1.0.2, 00:03:42, Tunnel0
D        150.1.3.3 [90/76800640] via 155.1.0.3, 00:03:42, Tunnel0
D        150.1.4.4 [90/76800640] via 155.1.0.4, 00:03:42, Tunnel0
      155.1.0.0/16 is variably subnetted, 12 subnets, 2 masks
D        155.1.13.0/24 [90/76805120] via 155.1.0.3, 00:03:42, Tunnel0
                       [90/76805120] via 155.1.0.1, 00:03:42, Tunnel0
D        155.1.23.0/24 [90/76805120] via 155.1.0.3, 00:03:42, Tunnel0
                       [90/76805120] via 155.1.0.2, 00:03:42, Tunnel0
D        155.1.37.0/24 [90/76805120] via 155.1.0.3, 00:03:42, Tunnel0
D        155.1.146.0/24 [90/76805120] via 155.1.0.4, 00:03:42, Tunnel0
                        [90/76805120] via 155.1.0.1, 00:03:42, Tunnel0
!R1#traceroute 150.1.2.2
Type escape sequence to abort.
Tracing the route to 150.1.2.2
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.0.5
 7 msec 4 msec 5 msec
  2 155.1.0.2 4 msec *  9 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

DMVPN Phase 1 with OSPF

> You must load the initial configuration files for the section, **DMVPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a DMVPN Phase 1 network between R1 - R5 as follows:
  - R1 - R4 are the DMVPN spokes.
  - R5 is the DMVPN Hub, and the NHRP Next-Hop Server (NHS).
  - Source the tunnel from the routers' GigabitEthernet1.100 interface.
  - Use IP addressing in the format 155.1.0.Y/24, where *Y* is the router number.
  - Use an NHRP network ID of **1**.
  - Use an NHRP authentication string of **NHRPAUTH**.
  - Use GRE tunnel key of **2**.
  - Ensure that the spokes can send multicast traffic to the hub, and vice versa.
  - To prevent the tunnel endpoints from having to do IPsec fragmentation, configure the GRE tunnel's IP MTU to 1400 bytes, and set them to adjust the TCP MSS accordingly.
- Configure IGP routing over the DMVPN tunnel as follows:
  - Enable OSPF area 0 on the DMVPN tunnel on R1 - R5.
  - Advertise the routers' Loopback0 networks into OSPF.
  - Use OSPF network type point-to-multipoint on R5's DMVPN tunnel.
  - Ensure that the OSPF hello and dead intervals agree between the DMVPN hub and spokes.
- Configure IPsec over the DMVPN tunnels as follows:
  - Use an ISAKMP Policy with the following options:
    - Pre-Shared Key: **DMVPN_PSK**
    - Encryption: AES 128 Bit

https://t.me/learningnets

        - Hash: SHA 256 Bit
        - Diffie-Hellman Group: 16
        - R5 should use a single Pre-Shared Key for all DMVPN peers.
    - Use a Crypto IPsec Profile named **DMVPN_PROFILE** with the following options:
        - Encrypt the traffic using AES 256 Bit.
        - Authenticate the traffic using SHA 512 Bit.
        - Use ESP Transport mode to save additional encapsulation overhead.
- When complete, ensure that R1 - R5 can reach each other's Loopback0 networks over the DMVPN network.

Configuration

```
R1:
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.1 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
```

https://t.me/learningnets

```
 no shutdown
!
router ospf 1
 network 150.1.0.0 0.0.255.255 area 0
 network 155.1.0.0 0.0.0.255 area 0
```

R2:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.2 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router ospf 1
 network 150.1.0.0 0.0.255.255 area 0
 network 155.1.0.0 0.0.0.255 area 0
```

R3:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
```

https://t.me/learningnets

```
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.3 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router ospf 1
 network 150.1.0.0 0.0.255.255 area 0
 network 155.1.0.0 0.0.0.255 area 0
```

R4:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 169.254.100.5
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.4 255.255.255.0
 ip mtu 1400
```

https://t.me/learningnets

```
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel destination 169.254.100.5
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router ospf 1
 network 150.1.0.0 0.0.255.255 area 0
 network 155.1.0.0 0.0.0.255 area 0
```

R5:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.5 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map multicast dynamic
 ip nhrp network-id 1
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 ip ospf network point-to-multipoint
 ip ospf hello-interval 10
 no shutdown
```

https://t.me/learningnets

```
!
router ospf 1
 network 150.1.0.0 0.0.255.255 area 0
 network 155.1.0.0 0.0.0.255 area 0
```

Verification

When OSPF is configured over GRE tunnel interfaces, the OSPF network type defaults to *point-to-point*. This is not supported in a DMVPN design, because the hub must maintain multiple adjacencies on the same interface, one for each remote spoke. In DMVPN Phase 1 with OSPF, the OSPF network type is set to point-to-multipoint on the hub at a minimum. With the hub being OSPF network type point-to-multipoint and the spokes being OSPF network type point-to-point, adjacency is supported, as long as the timer values match. There are essentially three options in this design.

- Set the DMVPN hub to OSPF network type point-to-multipoint and adjust its timers to match the spokes.
- Set the DMVPN hub to OSPF network type point-to-multipoint and adjust the spokes' timers to match the hub.
- Set the DMVPN hub and spokes to OSPF network type point-to-multipoint.

The end result of any of these three options is the same, in which the hub and spokes form adjacency, as seen below:

```
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.5.5         0 FULL/  -
        00:00:32    155.1.0.5       Tunnel0
!R5#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface 150.1.3.3         0 FULL/  -
        00:00:37    155.1.0.3       Tunnel0 150.1.2.2         0 FULL/  -
        00:00:35    155.1.0.2       Tunnel0 150.1.1.1         0 FULL/  -
        00:00:33    155.1.0.1       Tunnel0 150.1.4.4         0 FULL/  -
        00:00:32    155.1.0.4       Tunnel0
```

IPsec control plane and data plane traffic patterns are identical in this design as in RIP or EIGRP over DMVPN Phase 1. The spokes only form IPsec tunnels with the hub, and all spoke-to-spoke traffic must transit the hub.

```
R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status 169.254.100.5   169.254.100.1
  QM_IDLE           1003 ACTIVE
```

https://t.me/learningnets

```
R5#show crypto isakmp sa

IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status169.254.100.5   169.254.100.4
   QM_IDLE           1013 ACTIVE169.254.100.5   169.254.100.2
   QM_IDLE           1014 ACTIVE169.254.100.5   169.254.100.3
   QM_IDLE           1015 ACTIVE169.254.100.5   169.254.100.1
   QM_IDLE           1012 ACTIVE
!R1#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnetsO150.1.2.2 [110/2001]via 155.1.0.5, 00:25:27,Tunnel0
O        150.1.3.3 [110/2001] via 155.1.0.5, 00:25:27, Tunnel0
O        150.1.4.4 [110/2001] via 155.1.0.5, 00:25:27, Tunnel0
O        150.1.5.5 [110/1001] via 155.1.0.5, 00:25:37, Tunnel0
      155.1.0.0/16 is variably subnetted, 7 subnets, 2 masks
O        155.1.0.5/32 [110/1000] via 155.1.0.5, 00:25:37, Tunnel0
!R5#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnets
O        150.1.1.1 [110/1001] via 155.1.0.1, 00:25:57, Tunnel0O150.1.2.2 [110/1001]via 155.1.0.2
, 00:25:57,Tunnel0
O        150.1.3.3 [110/1001] via 155.1.0.3, 00:25:57, Tunnel0
O        150.1.4.4 [110/1001] via 155.1.0.4, 00:25:47, Tunnel0
!R1#traceroute 150.1.2.2
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 150.1.2.2
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.0.5
 3 msec 4 msec 4 msec
  2 155.1.0.2 7 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - IPSec VPN

DMVPN Phase 2 with EIGRP

> You must load the initial configuration files for the section, **DMVPN**, which can be found in CCIE R&S v5 Topology Diagrams & Initial Configurations.

Task

- Create a DMVPN Phase 2 network between R1 - R5 as follows:
  - R1 - R4 are the DMVPN spokes.
  - R5 is the DMVPN Hub, and the NHRP Next-Hop Server (NHS).
  - Source the tunnel from the routers' GigabitEthernet1.100 interface.
  - Use IP addressing in the format 155.1.0.Y/24, where **Y** is the router number.
  - Use an NHRP network ID of **1**.
  - Use an NHRP authentication string of **NHRPAUTH**.
  - Use GRE tunnel key of **2**.
  - Ensure that the spokes can send multicast traffic to the hub, and vice versa.
  - To prevent the tunnel endpoints from having to do IPsec fragmentation, configure the GRE tunnel's IP MTU to 1400 bytes, and set them to adjust the TCP MSS accordingly.
- Configure IGP routing over the DMVPN tunnel as follows:
  - Enable EIGRP in Multi-AF mode using AS 100 on the DMVPN tunnel on R1 - R5.
  - All links should be passive interfaces except the DMVPN tunnel.
  - Advertise the routers' Loopback0 networks into EIGRP.
  - Disable split-horizon for EIGRP on the DMVPN tunnel interface of R5.
  - Configure R5 to not modify the next-hop value of EIGRP routes that it advertises between spokes.
- Configure IPsec over the DMVPN tunnels as follows:
  - Use an ISAKMP Policy with the following options:

https://t.me/learningnets

        - Pre-Shared Key: **DMVPN_PSK**
        - Encryption: AES 128 Bit
        - Hash: SHA 256 Bit
        - Diffie-Hellman Group: 16
        - Use a single wildcard Pre-Shared Key for all DMVPN peers.
    - Use a Crypto IPsec Profile named **DMVPN_PROFILE** with the following options:
        - Encrypt the traffic using AES 256 Bit.
        - Authenticate the traffic using SHA 512 Bit.
        - Use ESP Transport mode to save additional encapsulation overhead.
- When complete, ensure that R1 - R5 can reach each other's Loopback0 networks over the DMVPN network.
- Additionally, ensure that spoke-to-spoke traffic does not transit the hub after initial NHRP mappings are formed.

Configuration

```
R1:
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.1 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
```

https://t.me/learningnets

```
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R2:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.2 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
```

https://t.me/learningnets

```
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R3:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.3 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
```

https://t.me/learningnets

```
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R4:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 ip address 155.1.0.4 255.255.255.0
 ip mtu 1400
```

https://t.me/learningnets

```
 ip nhrp authentication NHRPAUTH
 ip nhrp map 155.1.0.5 169.254.100.5
 ip nhrp map multicast 169.254.100.5
 ip nhrp network-id 1
 ip nhrp nhs 155.1.0.5
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

R5:

```
crypto isakmp policy 10
 encr aes 128
 hash sha256
 authentication pre-share
 group 16
!
crypto isakmp key DMVPN_PSK address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
```

https://t.me/learningnets

```
interface Tunnel0
 ip address 155.1.0.5 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPAUTH
 ip nhrp map multicast dynamic
 ip nhrp network-id 1
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.100
 tunnel mode gre multipoint
 tunnel key 2
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
!
router eigrp DMVPN
 !
 address-family ipv4 unicast autonomous-system 100
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel0
   no next-hop-self
   no passive-interface
   no split-horizon
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 150.1.0.0
  network 155.1.0.0
 exit-address-family
```

Verification

DMVPN Phase 2 was the first optimization for DMVPN designs, which required frequent spoke-to-spoke traffic patterns. Recall that in DMVPN Phase 1, all spoke-to-spoke traffic was always tunneled through the hub, before being tunneled back to the remote spokes. This design creates a bottleneck or choke-point in the network, because the hub is not only responsible for maintaining the IPsec and NHRP control plane for all spokes, but also the spoke-to-spoke forwarding plane. To resolve this problem, DMVPN Phase 2 allows the spokes to form on-demand GRE and IPsec tunnels, which can then be used for direct data plane exchange without packets

https://t.me/learningnets

having transit the hub.

To accomplish this, DMVPN Phase 2 requires that spokes learn about all specific routes in the topology, and that the next-hop value of spoke-to-spoke routes not be modified to the hub's tunnel address. This requirement implies two key points about DMVPN Phase 2: first, that the hub cannot summarize routes between the spokes, and second, that the routing protocol used must support maintaining the next-hop of the spokes' routes.

In the case of EIGRP, the second of these requirements is met by issuing the `no ip next-hop-self eigrp` command under the link level for Classic EIGRP, or the `no next-hop-self` under the tunnel's af-interface when EIGRP is in Multi-AF mode. The end result is that next-hop values are preserved for spoke to hub to spoke advertisements, as seen below.

```
R1#show ip route eigrp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      150.1.0.0/32 is subnetted, 5 subnetsD 150.1.2.2 [90/102400640] via 155.1.0.2
, 00:01:06, Tunnel0D 150.1.3.3 [90/102400640] via 155.1.0.3
, 00:01:06, Tunnel0D 150.1.4.4 [90/102400640] via 155.1.0.4
, 00:01:06, Tunnel0D 150.1.5.5 [90/76800640] via 155.1.0.5
, 00:01:06, Tunnel0
      155.1.0.0/16 is variably subnetted, 11 subnets, 2 masks
D        155.1.5.0/24 [90/76805120] via 155.1.0.5, 00:01:06, Tunnel0
D        155.1.23.0/24 [90/102405120] via 155.1.0.2, 00:01:06, Tunnel0
D        155.1.37.0/24 [90/102405120] via 155.1.0.3, 00:01:06, Tunnel0
D        155.1.45.0/24 [90/76805120] via 155.1.0.5, 00:01:06, Tunnel0
D        155.1.58.0/24 [90/76805120] via 155.1.0.5, 00:01:06, Tunnel0
```

The preservation of the routing next-hop in DMVPN Phase 2 means that the spokes must now perform NHRP resolutions for each other's addresses. Previously, the spokes only did static NHRP resolution for the hub, whereas the hub did dynamic resolution to the spokes. Furthermore, because the NHRP information is exchanged

https://t.me/learningnets

inside the GRE tunnel, which is inside an IPsec tunnel, it means that the spokes must first negotiate new ISAKMP SAs and IPsec SAs for their on-demand tunnels. This can be seen below before and after the NHRP and IPsec control plane processes are finished. Initially, the spoke has only NHRP and ISAKMP/IPsec entries for the hub.

```
R1#show ip nhrp
155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 01:03:51, never expire
   Type: static, Flags: used
   NBMA address: 169.254.100.5
!R1#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==============================================================================

Interface: Tunnel0, IPv4 NHRP Details
Type:Spoke, NHRP Peers:1,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1 169.254.100.5         155.1.0.5    UP 01:03:53     S
!R1#show crypto isakmp sa

IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
169.254.100.5   169.254.100.1   QM_IDLE           1017 ACTIVE

IPv6 Crypto ISAKMP SA
```

As spoke-to-spoke traffic is generated, NHRP and ISAKMP/IPsec tunnels processes start.

```
R1#show ip cef 150.1.2.2
150.1.2.2/32 nexthop 155.1.0.2 Tunnel0
!R1#ping 150.1.2.2 source 150.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 150.1.2.2, timeout is 2 seconds:
Packet sent with a source address of 150.1.1.1 .!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 3/11/30 ms
!R1#show ip nhrp
155.1.0.2/32 via 155.1.0.2
```

https://t.me/learningnets

```
   Tunnel0 created 00:00:06, expire 01:59:54    Type: dynamic
, Flags: router nhop    NBMA address: 169.254.100.2

155.1.0.5/32 via 155.1.0.5
   Tunnel0 created 01:04:21, never expire
   Type: static, Flags: used
   NBMA address: 169.254.100.5
!R1#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==============================================================================

Interface: Tunnel0, IPv4 NHRP Details
Type:Spoke, NHRP Peers:2,
 # Ent Peer NBMA Addr Peer Tunnel Add
 State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----     1 169.254.100.2         155.1.0.2
    UP 00:00:09     D
     1 169.254.100.5         155.1.0.5    UP 01:04:23     S
!R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status 169.254.100.1   169.254.100.2
   QM_IDLE           1018 ACTIVE
169.254.100.5   169.254.100.1   QM_IDLE           1017 ACTIVE

IPv6 Crypto ISAKMP SA
!R1#show crypto ipsec sa peer 169.254.100.2

interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 169.254.100.1

   protected vrf: (none)   local  ident (addr/mask/prot/port): (169.254.100.1/255.255.255.255/47
/0)   remote ident (addr/mask/prot/port): (169.254.100.2/255.255.255.255/47
/0)
   current_peer 169.254.100.2 port 500
     PERMIT, flags={origin_is_acl,} #pkts encaps: 4, #pkts encrypt: 4, #pkts digest: 4
 #pkts decaps: 5, #pkts decrypt: 5, #pkts verify: 5
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 169.254.100.1, remote crypto endpt.: 169.254.100.2
```

https://t.me/learningnets

```
     plaintext mtu 1442, path mtu 1500, ip mtu 1500, ip mtu idb (none)
     current outbound spi: 0xB6C12572(3066111346)
     PFS (Y/N): N, DH group: none
 inbound esp sas
:
      spi: 0x38B12992(951134610)        transform:esp-256-aes esp-sha512-hmac
 ,        in use settings ={Transport
, }
        conn id: 2183, flow_id: CSR:183, sibling_flags FFFFFFFF80000008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/3577)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)
      spi: 0x45D4E7C(73223804)
        transform: esp-256-aes esp-sha512-hmac ,
        in use settings ={Transport, }
        conn id: 2185, flow_id: CSR:185, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4607999/3577)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:
 outbound esp sas
:
      spi: 0xC348883B(3276310587)
        transform: esp-256-aes esp-sha512-hmac ,
        in use settings ={Transport, }
        conn id: 2184, flow_id: CSR:184, sibling_flags FFFFFFFF80000008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/3577)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
        Status: ACTIVE(ACTIVE)
      spi: 0xB6C12572(3066111346)        transform:esp-256-aes esp-sha512-hmac
 ,        in use settings ={Transport
, }
        conn id: 2186, flow_id: CSR:186, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4607999/3577)
        IV size: 16 bytes
        replay detection support: Y
        ecn bit support: N status: off
```

https://t.me/learningnets

```
        Status: ACTIVE(ACTIVE)

    outbound ah sas:

    outbound pcp sas:
```

Now that a direct IPsec tunnel is formed between R1 and R2, their spoke-to-spoke traffic need not flow through the hub, as seen below.

```
R1#traceroute 150.1.2.2 source 150.1.1.1
Type escape sequence to abort.
Tracing the route to 150.1.2.2
VRF info: (vrf in name/id, vrf out name/id)  1 155.1.0.2
 204 msec *  15 msec
```

Note that even though spokes now have a direct tunnel formed between them, they still do not form a routing adjacency. This is because the IGP multicast hello messages are only exchanged between the hub and spokes, not directly spoke to spoke.

```
R1#show ip eigrp neighbors
EIGRP-IPv4 VR(DMVPN) Address-Family Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
0   155.1.0.5               Tu0                      10 00:32:03  146  1398  0  85
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Foundation Labs

Foundation Lab 1 Tasks

> Load the **Foundation Lab 1** initial configurations prior to starting. Initial configs and the diagram for this task can be found under the **Resources** tab on the right. The diagram is optimized for 1080p fullscreen.

Troubleshooting

1.1 Faults

- There is 1 fault with the initial configurations that need to be resolved.
- All information (IP addressing, interface numbering, etc.) in the diagrams is correct.

LAN Switching

2.1 - EtherChannel

- Configure SW1's links to SW2, SW3, and SW4 as three separate Port Channels numbered 12, 13, and 14 respectively.
- SW1 should use LACP in active mode, while SW2, SW3, and SW4 should use LACP in passive mode.

2.2 - Trunking

- Configure the Port Channels from SW1 to SW2, SW3, and SW4 as static 802.1q trunk links.
- Configure SW1's port Fa0/1 as a static 802.1q trunk link.

https://t.me/learningnets

2.3 - VTP

- Configure SW1 as a VTP server in the domain *CCIE*.
- SW2, SW3, and SW4 should be VTP clients.
- Configure VLANs 21, 22, 121, 122, 124, 221, 222, and 239 on SW1.
- Once complete, SW2, SW3, and SW4 should learn these VLANs via VTP.

2.4 - Spanning-Tree

- Enable Per-VLAN Rapid Spanning Tree on SW1, SW2, SW3, and SW4.
- Configure SW1 as the RPVST root bridge for all configured VLANs.
- Ensure that no topology change notifications are sent throughout the spanning-tree domain if SW1's FastEthernet0/1 interface goes down/up.

WAN Technologies

3.1 PPPoE

- Configure a PPPoE connection between R1 and R3.
- Establish the PPPoE session over the GigabitEthernet1.13 interface.
- Configure R3 as the server and R1 as the client.
- Remove the IPv4 address from GigabitEthernet1.13 and apply it on the PPPoE interfaces.
- Leave the IPv6 address configured on the GigabitEthernet1.13 interface.

3.2 PPPoE Authentication

- Configure PPP CHAP authentication over the PPPoE session between R1 and R3.
- The routers should use their hostname and a password of 'RSv5' to authenticate each other.

Interior Gateway Protocol Routing

4.1 - IPv4 EIGRP

https://t.me/learningnets

- Configure EIGRP AS 1 on R1, R2, R5 - R10, and SW1 - SW4.
- Enable EIGRP on all links in the 10.1.0.0/16 network.
- Redistribute the devices Loopback0 networks into EIGRP.

4.2 - IPv4 EIGRP Authentication

- Configure EIGRP authentication between R1, R2, SW1, and SW2.
- All of SW1's neighbors should use MD5 key 0 and password 'SW!_Key'
- All of SW2's neighbors should use MD5 key 5 and password 'SW@_Key'

4.3 - IPv4 EIGRP Optimization

- Configure R5 so that GigabitEthernet1.35 never sends hellos, even if a network statement is ever entered for the '100.1.35.0/24' network.
- Configure R6 and R8 to detect a failure between their link in less than 5 seconds.

Exterior Gateway Protocol Routing

5.1 - EBGP

- R3 is in BGP AS 3, while R4 is in AS 4.
- R3 and R4 are preconfigured for peerings according to the table below.
- Configure BGP on the devices in the network per the table below.
- Form both IPv4 Unicast and IPv6 Unicast peerings between devices.
- Redistribute the devices IPv4 Loopback0 addresses into BGP.
- Ensure reachability between R1, R2, R5, R6, R9, and R10.

Device	BGP ASN	Peers
R1	65012	R3
R2	65012	R4
R5	65056	R3

https://t.me/learningnets

Device	BGP ASN	Peers
R6	65056	R4
R9	65012	R3
R10	65012	R3

5.2 - IBGP

- Configure BGP on the devices in the network per the table below.
- Ensure that the IBGP sessions in AS 65012 are able to withstand a single link failure.
- R1, R2, R5, and R6 must be able to receive routes from other AS's if their corresponding EBGP neighbor goes down.

Device	BGP ASN	Peers
R1	65012	SW1
R2	65012	SW1
R5	65056	R7
R6	65056	R7

5.3 BGP Authentication

- Secure R9 and R10's IPv6 EBGP session with R3 by using an md5 password of 'BGP_PASS!'

5.4 BGP Aggregation

- Configure 4 new loopbacks on R9 - Loopback 150, 151, 160, and 163. The

https://t.me/learningnets

addresses for these new loopbacks should be 119.150.0.9/32, 119.151.0.9/32, 119.160.0.9/32, and 119.163.0.9/32 respectively.
- Advertise and aggregate these networks into BGP using a single summary that is most optimal.
- Ensure that the subnets of the aggregate are not seen by other BGP speakers.

5.5 BGP Path Selection

- Configure AS 65012 so that traffic towards AS 65056 prefers to transit AS 4 instead of AS 3.

DMVPN

6.1 - IPv4 over DMVPN

- Configure a DMVPN network between R1, R5, R9, and R10.
- R1 should be the hub, and R5, R9, and R10 the spokes.
- Source the tunnel from the routers' Loopback0 interface.
- Use the IP addresses 10.1.0.Y/24 inside the tunnel, where *Y* is the router's number.
- Use NHRP network ID 1.
- Use GRE tunnel key 2.
- Use the NHRP authentication key *DMVPN*.
- Enable DMVPN Phase 3 support.

6.2 - EIGRP over DMVPN

- Configure R1 to advertise just a default route to the DMVPN spokes via EIGRP.
- Once complete, SW3 and SW4 should have reachability to each other, but their traffic should not forward through the DMVPN hub (R1).

6.3 - IPsec over DMVPN

- Configure IPsec over the DMVPN network using the following parameters.
  - Use the following ISAKMP Policy:
    - Pre-Shared Key: **CCIE_DMVPN**
    - Encryption: AES 192 Bit
    - Hash: SHA 256 Bit

https://t.me/learningnets

        - Diffie-Hellman Group: 5
    - Use the following IPsec Profile:
        - IPsec Encapsulation: ESP Transport Mode
        - Encryption: AES 256 Bit
        - Hash: SHA 512 Bit
- Once complete, R1 should form IPsec tunnels with the DMVPN spokes.
- Additionally, the spokes should be able to form on-demand IPsec tunnels with each other for forwarding spoke-to-spoke traffic.

IPv6 Routing

7.1 - IPv6 EIGRP

- Configure EIGRPv6 AS1 between R1, R2, R5-R8, SW1, and SW2. Advertise the Loopback0 networks of these devices natively into EIGRPv6.

7.2 - IPv6 Redistribution

- Redistribute between EIGRPv6 and MP-BGP on R1 and R5.
- Ensure reachability between the IPv6 Loopback0 networks of the devices running EIGRPv6.

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Foundation Labs

Foundation Lab 1 Solutions

1.1 - Troubleshooting

The IPv4 address of the GigabitEthernet1.12 link on R1 has the wrong IP address. This will prevent EIGRP from forming an adjacency with R2 in a later task. One of the first things that you should do when beginning a lab is ensuring that basic IP connectivity is working between your directly connected devices. Pinging the broadcast address (255.255.255.255) and matching the responses with the diagram is a quick way to accomplish this check. The RSv5 CCIE lab has many more devices than the RSv4 lab. In a larger topology it makes more sense to do this in blocks instead of doing it all at once. We can do our topology all at once since it is relatively small. This check should be done after the Layer 2 portion is complete.

```
R1:
interface GigabitEthernet1.12
 ip address 10.1.12.1 255.255.255.0
```

2.1 - EtherChannel

```
SW1:
interface FastEthernet0/19
 channel-group 13 mode active
!
interface FastEthernet0/20
 channel-group 13 mode active
!
interface FastEthernet0/21
 channel-group 14 mode active
!
interface FastEthernet0/22
 channel-group 14 mode active
!
interface FastEthernet0/23
```

https://t.me/learningnets

```
 channel-group 12 mode active
!
interface FastEthernet0/24
 channel-group 12 mode active

SW2:
interface FastEthernet0/23
 channel-group 12 mode passive
!
interface FastEthernet0/24
 channel-group 12 mode passive

SW3:
interface FastEthernet0/19
 channel-group 13 mode passive
!
interface FastEthernet0/20
 channel-group 13 mode passive

SW4:
interface FastEthernet0/21
 channel-group 14 mode passive
!
interface FastEthernet0/22
 channel-group 14 mode passive
```

2.1 - EtherChannel Verification

```
SW1#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 3
Number of aggregators:           3
```

https://t.me/learningnets

```
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------12     Po12(SU)LACP
Fa0/23(P)   Fa0/24(P)
   13     Po13(SU)LACP Fa0/19(P)   Fa0/20(P)
   14     Po14(SU)LACP Fa0/21(P)   Fa0/22(P)
```

2.2 - Trunking

```
SW1:
interface Port-channel12
 switchport trunk encapsulation dot1q
 switchport mode trunk
!
interface Port-channel13
 switchport trunk encapsulation dot1q
 switchport mode trunk
!
interface Port-channel14
 switchport trunk encapsulation dot1q
 switchport mode trunk
!
interface FastEthernet0/1
 switchport trunk encapsulation dot1q
 switchport mode trunk

SW2:
interface Port-channel12
 switchport trunk encapsulation dot1q
 switchport mode trunk

SW3:
interface Port-channel13
 switchport trunk encapsulation dot1q
 switchport mode trunk

SW4:
interface Port-channel14
 switchport trunk encapsulation dot1q
 switchport mode trunk
```

https://t.me/learningnets

2.2 - Trunking Verification

```
SW1#show interface trunk

Port        Mode             Encapsulation  Status        Native vlan Fa0/1 on 802.1q trunking
      1 Po12 on 802.1q trunking
      1 Po13 on 802.1q trunking
      1 Po14 on 802.1q trunking
      1

Port        Vlans allowed on trunk
Fa0/1       1-4094
Po12        1-4094
Po13        1-4094
Po14        1-4094

Port        Vlans allowed and active in management domain
Fa0/1       1
Po12        1
Po13        1
Po14        1

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/1       1
Po12        1
Po13        1
Po14        1
SW2#show interface trunk

Port        Mode             Encapsulation  Status        Native vlan Po12
on               802.1q         trunking
      1

Port        Vlans allowed on trunk
Po12        1-4094

Port        Vlans allowed and active in management domain
Po12        1

Port        Vlans in spanning tree forwarding state and not pruned
Po12        1
SW3#show interface trunk
```

https://t.me/learningnets

```
Port        Mode             Encapsulation  Status        Native vlan Po13
on               802.1q          trunking
      1

Port        Vlans allowed on trunk
Po13        1-4094

Port        Vlans allowed and active in management domain
Po13        1

Port        Vlans in spanning tree forwarding state and not pruned
Po13        1
SW4#show interface trunk

Port        Mode             Encapsulation  Status        Native vlan Po14
on               802.1q          trunking
      1

Port        Vlans allowed on trunk
Po14        1-4094

Port        Vlans allowed and active in management domain
Po14        1

Port        Vlans in spanning tree forwarding state and not pruned
Po14        1
```

2.3 - VTP

https://t.me/learningnets

```
SW1:
vtp domain CCIE
vtp mode server
!
vlan 21,22,121,122,124,221,222,239

SW2:
vtp mode client

SW3:
vtp mode client

SW4:
vtp mode client
```

2.3 - VTP Verification

```
SW1#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1VTP Domain Name                 :CCIE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 000a.b832.3580
Configuration last modified by 10.1.21.21 at 3-1-93 00:12:41
Local updater ID is 10.1.21.21 on interface Vl21 (lowest numbered VLAN interface found)

Feature VLAN:
--------------VTP Operating Mode                :Server
Maximum VLANs supported locally   : 1005Number of existing VLANs          :13
Configuration Revision            :3
MD5 digest                        : 0x4C 0x06 0xAF 0x7F 0x47 0x71 0x2A 0x3A
                                    0x97 0x3B 0x26 0x19 0xA1 0x8B 0x2C 0xC1
SW2#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1VTP Domain Name                 :CCIE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 001c.576d.4a00
Configuration last modified by 10.1.21.21 at 3-1-93 00:12:41

Feature VLAN:
--------------
```

https://t.me/learningnets

```
VTP Operating Mode                :Client

Maximum VLANs supported locally   : 1005 Number of existing VLANs          :13
Configuration Revision            :3
MD5 digest                        : 0x4C 0x06 0xAF 0x7F 0x47 0x71 0x2A 0x3A
                                    0x97 0x3B 0x26 0x19 0xA1 0x8B 0x2C 0xC1
SW1#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/2, Fa0/3, Fa0/4, Fa0/5
                                                Fa0/6, Fa0/7, Fa0/8, Fa0/9
                                                Fa0/10, Fa0/11, Fa0/12, Fa0/13
                                                Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                                Fa0/18, Gi0/1, Gi0/2
21   VLAN0021
                         active
22   VLAN0022
                         active
121  VLAN0121
                         active
122  VLAN0122
                         active
124  VLAN0124
                         active
221  VLAN0221
                         active
222  VLAN0222
                         active
239  VLAN0239
                         active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW2#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22, Gi0/1, Gi0/2
21   VLAN0021
                         active
22   VLAN0022
                         active
121  VLAN0121
                         active
122  VLAN0122
                         active
124  VLAN0124
                         active
221  VLAN0221
                         active
222  VLAN0222
                         active
239  VLAN0239
                         active
1002 fddi-default                     act/unsup
```

https://t.me/learningnets

```
1003 token-ring-default                act/unsup
1004 fddinet-default                   act/unsup
1005 trnet-default                     act/unsup
```

2.4 - Spanning-Tree

Note that even though FastEthernet0/1 is configured as a trunk, it can still be configured as a spanning-tree edge port. The 'trunk' keyword has to be added at the end of the portfast command in order to enable this feature. This feature is useful in the exact design that this lab is using - a trunk towards a server.

```
SW1:
spanning-tree mode rapid-pvst
spanning-tree vlan 1-4094 priority 0
!
interface FastEthernet0/1
 spanning-tree portfast trunk

SW2:
spanning-tree mode rapid-pvst

SW3:
spanning-tree mode rapid-pvst

SW4:
spanning-tree mode rapid-pvst
```

2.4 - Spanning-Tree Verification

```
SW1#show spanning-tree vlan 21

VLAN0021 Spanning tree enabled protocol rstp
  Root ID    Priority    21
             Address     000a.b832.3580 This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    21     (priority 0 sys-id-ext 21)
             Address     000a.b832.3580
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
```

https://t.me/learningnets

```
------------------- ---- --- --------- -------- --------------------------------Fa0/1 Desg FWD
 19        128.3    P2p Po12 Desg FWD
 12        128.152  P2p Po13 Desg FWD
 12        128.160  P2p Po14 Desg FWD
 12        128.168  P2p

SW1#show spanning-tree interface f0/1 portfast
 VLAN0001 enabled
VLAN0021 enabled
VLAN0022 enabled
VLAN0121 enabled
VLAN0122 enabled
VLAN0124 enabled
VLAN0221 enabled
VLAN0222 enabled
VLAN0239 enabled

SW1#show spanning-tree vlan 239

VLAN0239
  Spanning tree enabled protocol rstp
  Root ID    Priority    239
             Address     0018.ba8b.7b80
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    239    (priority 0 sys-id-ext 239)
             Address     0018.ba8b.7b80
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/1               Desg FWD 19        128.3    P2p Edge

Po12                Desg FWD 12        128.144  P2p
Po13                Desg FWD 12        128.152  P2p
Po14                Desg FWD 12        128.160  P2p
```

3.1 - PPPoE

```
R1:
!
interface GigabitEthernet1.13
```

https://t.me/learningnets

```
 no ip address 100.1.13.1 255.255.255.0
 pppoe-client dial-pool-number 13
!
interface Dialer13
 encapsulation ppp
 dialer pool 13
 ip address 100.1.13.1 255.255.255.0


R3:
!
interface GigabitEthernet1.13
 no ip address 100.1.13.3 255.255.255.0
 pppoe enable group global
!
bba-group pppoe global
 virtual-template 13
!
interface Virtual-Template13
 ip address 100.1.13.3 255.255.255.0
```

3.1 - PPPoE Verification

```
R1#show ppp all

Interface/ID OPEN+ Nego* Fail-     Stage    Peer Address    Peer Name
------------ --------------------- -------- --------------- --------------------
Vi1          LCP+ IPCP+            LocalT   100.1.13.3

R1#show pppoe session

     1 client session

Uniq ID  PPPoE  RemMAC          Port                    VT  VA         State
           SID  LocMAC                                      VA-st      Type
    N/A      2  0050.568d.0396 Gi1.13 Di13 Vi1 UP

                0050.568d.6533                              UP

R3#show pppoe session

     1 session  in LOCALLY_TERMINATED (PTA) State
     1 session  total
```

https://t.me/learningnets

```
Uniq ID  PPPoE  RemMAC          Port                    VT  VA         State
           SID  LocMAC                                     VA-st      Type
      1      2  0050.568d.6533 Gi1.13 13  Vi1.1
      PTA                  0050.568d.0396 VLAN:  13 UP


R3#ping 100.1.13.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.13.1, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

3.2 - PPPoE Authentication

```
R1:
username R3 password RSv5
!
interface Dialer13
 ppp authentication chap


R3:
username R1 password RSv5
!
interface Virtual-Template13
 ppp authentication chap
```

3.2 - PPPoE Authentication Verification

```
R1#debug ppp authentication

PPP authentication debugging is onR1#clear ppp all
R1#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to down
%DIALER-6-UNBIND: Interface Vi1 unbound from profile Di13
R1#
%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to down
R1#
%DIALER-6-BIND: Interface Vi1 bound to profile Di13
%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
Vi1 PPP: Using dialer call direction
Vi1 PPP: Treating connection as a callout
```

https://t.me/learningnets

```
Vi1 PPP: Session handle[BF000003] Session id[3] Vi1 CHAP: O CHALLENGE id 1 len 23 from "R1"
Vi1 CHAP: I CHALLENGE id 1 len 23 from "R3"
Vi1 PPP: Sent CHAP SENDAUTH Request
Vi1 PPP: Received SENDAUTH Response PASS
Vi1 CHAP: Using hostname from configured hostname
Vi1 CHAP: Using password from AAA
Vi1 CHAP: O RESPONSE id 1 len 23 from "R1" Vi1 CHAP: I SUCCESS id 1 len 4
Vi1 CHAP: I RESPONSE id 1 len 23 from "R3"
Vi1 PPP: Sent CHAP LOGIN Request
Vi1 PPP: Received LOGIN Response PASS Vi1 CHAP: O SUCCESS id 1 len 4

%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
```

4.1 - IPv4 EIGRP

```
R1, R2, R5 - R10, SW1 - SW4:
ip routing
!
router eigrp 1
 network 10.1.0.0 0.0.255.255
 redistribute connected metric 1000000 1 255 1 1500 route-map LOOPBACK
!
route-map LOOPBACK permit 10
 match interface Loopback0
```

2.1 - IPv4 EIGRP Verification

```
R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num 2 10.1.12.2
               Gi1.12                   12 00:02:06    9   100 0
  9 1 10.1.121.21             Gi1.121                  12 00:02:14    9   100 0
  8 0 10.1.122.22             Gi1.122                  11 00:02:14    7   100 0
  8

R1#show ip route eigrp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
```

https://t.me/learningnets

```
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      2.0.0.0/32 is subnetted, 1 subnets
 EX     2.2.2.2 [170/3072] via 10.1.12.2, 00:02:08, GigabitEthernet1.12
      10.0.0.0/8 is variably subnetted, 10 subnets, 2 masks
        10.1.21.0/24
           [90/3072] via 10.1.121.21, 00:02:03, GigabitEthernet1.121
        10.1.22.0/24
           [90/3072] via 10.1.122.22, 00:02:03, GigabitEthernet1.122
        10.1.221.0/24
           [90/3072] via 10.1.121.21, 00:02:08, GigabitEthernet1.121
           [90/3072] via 10.1.12.2, 00:02:08, GigabitEthernet1.12
        10.1.222.0/24
           [90/3072] via 10.1.122.22, 00:02:08, GigabitEthernet1.122
           [90/3072] via 10.1.12.2, 00:02:08, GigabitEthernet1.12
      21.0.0.0/32 is subnetted, 1 subnets
 EX     21.21.21.21
           [170/3072] via 10.1.121.21, 00:02:03, GigabitEthernet1.121
      22.0.0.0/32 is subnetted, 1 subnets
 EX     22.22.22.22
           [170/3072] via 10.1.122.22, 00:02:03, GigabitEthernet1.122
```

4.1 - IPv4 EIGRP Authentication

```
R1:
key chain SW1_KEY
 key 0
   key-string SW!_Key
!
key chain SW2_KEY
 key 5
   key-string SW@_Key
!
interface GigabitEthernet1.121
 ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 SW1_KEY
!
interface GigabitEthernet1.122
 ip authentication mode eigrp 1 md5
```

https://t.me/learningnets

```
 ip authentication key-chain eigrp 1 SW2_KEY
```

```
R2:
key chain SW1_KEY
 key 0
   key-string SW!_Key
!
key chain SW2_KEY
 key 5
   key-string SW@_Key
!
interface GigabitEthernet1.221
 ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 SW1_KEY
!
interface GigabitEthernet1.222
 ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 SW2_KEY
```

```
SW1:
key chain SW1_KEY
 key 0
   key-string SW!_Key
!
interface Vlan121
 ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 SW1_KEY
!
interface Vlan221
 ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 SW1_KEY
```

```
SW2:
key chain SW2_KEY
 key 5
   key-string SW@_Key
!
interface Vlan122
 ip authentication mode eigrp 1 md5
 ip authentication key-chain eigrp 1 SW2_KEY
!
interface Vlan222
 ip authentication mode eigrp 1 md5
```

https://t.me/learningnets

```
ip authentication key-chain eigrp 1 SW2_KEY
```

4.2 - IPv4 EIGRP Authentication Verification

```
R1#show ip eigrp neighbors

EIGRP-IPv4 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
2   10.1.122.22             Gi1.122                  10 00:02:19    9   100  0  35
1   10.1.121.21             Gi1.121                  14 00:04:00   10   100  0  27
0   10.1.12.2               Gi1.12                   12 00:14:54    1   100  0  42

R1#show ip eigrp interfaces detail Gi1.121

EIGRP-IPv4 Interfaces for AS(1)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi1.121                  1        0/0       0/0          10       0/0           50           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 12/0
  Hello's sent/expedited: 183/5
  Un/reliable mcasts: 0/11  Un/reliable ucasts: 13/23
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 19  Out-of-sequence rcvd: 1
  Topology-ids on interface - 0  Authentication mode is md5,  key-chain is "SW1_KEY"

R1#show ip eigrp interfaces detail Gi1.122

EIGRP-IPv4 Interfaces for AS(1)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi1.122                  1
        0/0       0/0           9       0/0           50           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 12/0
  Hello's sent/expedited: 205/4
  Un/reliable mcasts: 0/12  Un/reliable ucasts: 14/6
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 2  Out-of-sequence rcvd: 1
```

https://t.me/learningnets

```
  Topology-ids on interface - 0  Authentication mode is md5,  key-chain is "SW2_KEY"
```

4.2 - IPv4 EIGRP Optimization

```
R1:
router eigrp 1
 passive-interface GigabitEthernet1.35

R6:
interface GigabitEthernet1.68
 ip hello-interval eigrp 1 2
 ip hold-time eigrp 1 5

R8:
interface GigabitEthernet1.68
 ip hello-interval eigrp 1 2
 ip hold-time eigrp 1 5
```

4.2 - IPv4 EIGRP Optimization Verification

```
R5(config)#router eigrp 1
R5(config-router)#network 100.1.35.5 0.0.0.0
R5(config-router)#end

R5#show ip protocols | begin "eigrp 1"
Routing Protocol is "eigrp 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  Redistributing: connected
  EIGRP-IPv4 Protocol for AS(1)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
    Soft SIA disabled
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 5.5.5.5
    Topology : 0 (base)
      Active Timer: 3 min
```

https://t.me/learningnets

```
      Distance: internal 90 external 170
      Maximum path: 4
      Maximum hopcount 100
      Maximum metric variance 1

  Automatic Summarization: disabled
  Maximum path: 4
  Routing for Networks:
    10.1.0.0/16
    100.1.35.5/32 Passive Interface(s):
 GigabitEthernet1.35
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.1.57.7             90      00:30:35
    10.1.56.6             90      00:30:37
  Distance: internal 90 external 170

R5(config)#router eigrp 1
R5(config-router)#no network 100.1.35.5 0.0.0.0
R5(config-router)#end

R6#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num1
10.1.68.8                Gi1.68 4
 00:31:56    1   100  0  6
0   10.1.56.5               Gi1.56                   13 00:31:58 1020  5000  0  12

R6#show ip eigrp interfaces detail Gi1.68
EIGRP-IPv4 Interfaces for AS(1)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi1.68                   1        0/0       0/0           1       0/0           50           0
Hello-interval is 2, Hold-time is 5
```

5.1 - EBGP

```
R1:
router bgp 65012
 bgp log-neighbor-changes
 neighbor 2001:100:1:13::3 remote-as 3
 neighbor 100.1.13.3 remote-as 3
```

https://t.me/learningnets

```
 !
 address-family ipv4
  redistribute connected route-map LOOPBACK
  no neighbor 2001:100:1:13::3 activate
  neighbor 100.1.13.3 activate
 exit-address-family
 !
 address-family ipv6
  neighbor 2001:100:1:13::3 activate
 exit-address-family
!
route-map LOOPBACK permit 10
 match interface Loopback0

R2:
router bgp 65012
 neighbor 2001:200:1:24::4 remote-as 4
 neighbor 200.1.24.4 remote-as 4
 !
 address-family ipv4
  redistribute connected route-map LOOPBACK
  no neighbor 2001:200:1:24::4 activate
  neighbor 200.1.24.4 activate
 exit-address-family
 !
 address-family ipv6
  neighbor 2001:200:1:24::4 activate
 exit-address-family
!
route-map LOOPBACK permit 10
 match interface Loopback0

R5:
router bgp 65056
 neighbor 2001:100:1:35::3 remote-as 3
 neighbor 100.1.35.3 remote-as 3
 !
 address-family ipv4
  redistribute connected route-map LOOPBACK
  no neighbor 2001:100:1:35::3 activate
  neighbor 100.1.35.3 activate
 exit-address-family
 !
 address-family ipv6
  neighbor 2001:100:1:35::3 activate
 exit-address-family
```

https://t.me/learningnets

```
!
route-map LOOPBACK permit 10
 match interface Loopback0

R6:
router bgp 65056
 neighbor 2001:200:1:46::4 remote-as 4
 neighbor 200.1.46.4 remote-as 4
 !
 address-family ipv4
  redistribute connected route-map LOOPBACK
  no neighbor 2001:200:1:46::4 activate
  neighbor 200.1.46.4 activate
 exit-address-family
 !
 address-family ipv6
  neighbor 2001:200:1:46::4 activate
 exit-address-family
!
route-map LOOPBACK permit 10
 match interface Loopback0

R9:
router bgp 65009
 neighbor 2001:100:1:39::3 remote-as 3
 neighbor 100.1.39.3 remote-as 3
 !
 address-family ipv4
  redistribute connected route-map LOOPBACK
  no neighbor 2001:100:1:39::3 activate
  neighbor 100.1.39.3 activate
 exit-address-family
 !
 address-family ipv6
  neighbor 2001:100:1:39::3 activate
 exit-address-family
!
route-map LOOPBACK permit 10
 match interface Loopback0

R10:
router bgp 65010
 neighbor 2001:100:1:103::3 remote-as 3
 neighbor 100.1.103.3 remote-as 3
 !
 address-family ipv4
```

https://t.me/learningnets

```
  redistribute connected route-map LOOPBACK
  no neighbor 2001:100:1:103::3 activate
  neighbor 100.1.103.3 activate
 exit-address-family
 !
 address-family ipv6
  neighbor 2001:100:1:103::3 activate
 exit-address-family
!
route-map LOOPBACK permit 10
 match interface Loopback0
```

5.1 - EBGP Verification

```
R1#show bgp ipv4 unicast summary

BGP router identifier 1.1.1.1, local AS number 65012
BGP table version is 10, main routing table version 10
9 network entries using 2232 bytes of memory
9 path entries using 1080 bytes of memory
5/5 BGP path/bestpath attribute entries using 1200 bytes of memory
4 BGP AS-PATH entries using 144 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 4656 total bytes of memory
BGP activity 18/0 prefixes, 18/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
100.1.13.3      4            3      10       7       10    0    0 00:02:30        9
R1#show bgp ipv4 unicast

BGP table version is 12, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  1.1.1.1/32       0.0.0.0                  0         32768 ?
 *>  3.3.3.3/32       100.1.13.3               0             0 3 ?
 *>  5.5.5.5/32       100.1.13.3                             0 3 65056 ?
 *>  9.9.9.9/32       100.1.13.3                             0 3 65009 ?
 *>  10.10.10.10/32   100.1.13.3                             0 3 65010 ?
```

https://t.me/learningnets

```
 r>  100.1.13.0/24    100.1.13.3               0             0 3 ?
 r>  100.1.13.1/32    100.1.13.3               0             0 3 ?
 *>  100.1.35.0/24    100.1.13.3               0             0 3 ?
 *>  100.1.39.0/24    100.1.13.3               0             0 3 ?
 *>  100.1.103.0/24   100.1.13.3               0             0 3 ?
```

Note that although the IPv6 BGP sessions are established at this moment, the only routes being advertised over IPv6 are the connected interfaces of R3 and R4 (part of the pre-configs).

```
R1#show bgp ipv6 unicast summary
BGP router identifier 1.1.1.1, local AS number 65012
BGP table version is 10, main routing table version 10
9 network entries using 2448 bytes of memory
9 path entries using 1296 bytes of memory
5/5 BGP path/bestpath attribute entries using 1200 bytes of memory
4 BGP AS-PATH entries using 144 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 5088 total bytes of memory
BGP activity 18/0 prefixes, 18/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:100:1:13::3
                4            3      11       6       10    0    0 00:02:40        5
R1#show bgp ipv6 unicast

BGP table version is 11, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  2001:3:3:3::3/128
                      2001:100:1:13::3
                                                 0             0 3 ?
 r>  2001:100:1:13::/64
                      2001:100:1:13::3
                                                 0             0 3 ?
 *>  2001:100:1:35::/64
                      2001:100:1:13::3
                                                 0             0 3 ?
 *>  2001:100:1:39::/64
```

https://t.me/learningnets

```
                      2001:100:1:13::3
                                                    0             0 3 ?
 *>  2001:100:1:103::/64
                      2001:100:1:13::3
```

5.2 - IBGP

```
R1:
router bgp 65012
 bgp log-neighbor-changes
 neighbor 22.22.22.22 remote-as 65012
 neighbor 22.22.22.22 update-source loopback0
 !
 address-family ipv4
  neighbor 22.22.22.22 activate
  neighbor 22.22.22.22 next-hop-self

R2:
router bgp 65012
 bgp log-neighbor-changes
 neighbor 22.22.22.22 remote-as 65012
 neighbor 22.22.22.22 update-source loopback0
 !
 address-family ipv4
  neighbor 22.22.22.22 activate
  neighbor 22.22.22.22 next-hop-self

R5:
router bgp 65056
 bgp log-neighbor-changes
 neighbor 7.7.7.7 remote-as 65056
 neighbor 7.7.7.7 update-source loopback0
 !
 address-family ipv4
  neighbor 7.7.7.7 activate
  neighbor 7.7.7.7 next-hop-self

R6:
router bgp 65056
 bgp log-neighbor-changes
 neighbor 7.7.7.7 remote-as 65056
 neighbor 7.7.7.7 update-source loopback0
 !
 address-family ipv4
```

https://t.me/learningnets

```
  neighbor 7.7.7.7 activate
  neighbor 7.7.7.7 next-hop-self

R7:
router bgp 65056
 bgp log-neighbor-changes
 neighbor 5.5.5.5 remote-as 65056
 neighbor 5.5.5.5 update-source loopback0
 neighbor 6.6.6.6 remote-as 65056
 neighbor 6.6.6.6 update-source loopback0

 !
 address-family ipv4
  neighbor 5.5.5.5 activate
  neighbor 6.6.6.6 activate
  neighbor 5.5.5.5 route-reflector-client
  neighbor 6.6.6.6 route-reflector-client

SW2:
router bgp 65012
 bgp log-neighbor-changes
 neighbor 1.1.1.1 remote-as 65012
 neighbor 1.1.1.1 update-source loopback0
 neighbor 2.2.2.2 remote-as 65012
 neighbor 2.2.2.2 update-source loopback0

 !
 address-family ipv4
  neighbor 1.1.1.1 activate
  neighbor 2.2.2.2 activate
  neighbor 1.1.1.1 route-reflector-client
  neighbor 2.2.2.2 route-reflector-client
```

5.2 - IBGP Verification

The IBGP peerings in AS 65012 have to be configured off the Loopback0 interfaces to withstand a single link failure. The peerings in AS 65056 could be done off the physical interfaces, but they were configured off the loopbacks to be consistent. In order to account for an EBGP peer going down on the edge routers in AS 65012 and 65056, SW2 and R7 must be configured a route-reflectors. Additionally, SW2 and R7 must have a valid next-hop for the routes sent to them. We could either advertise the links used for EBGP peering into EIGRP, or set the next-hop to self. Note that this was not explicitly asked for, but it must be done in order to have a

https://t.me/learningnets

working solution.

```
SW2#show ip bgp summary

BGP router identifier 22.22.22.22, local AS number 65012
BGP table version is 5, main routing table version 5
15 network entries using 1755 bytes of memory
17 path entries using 884 bytes of memory
8/1 BGP path/bestpath attribute entries using 1120 bytes of memory
6 BGP AS-PATH entries using 144 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 3903 total bytes of memory
BGP activity 15/0 prefixes, 17/0 paths, scan interval 60 secs

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1.1.1.1         4 65012      19      13        5    0    0 00:08:34       11
2.2.2.2         4 65012      17      13        5    0    0 00:08:28        6
```

The IBGP session will remain UP even after shutting down the sub-interface towards SW2 - EIGRP re-routes around the failure.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#int g1.121
R1(config-subif)#shut
R1(config-subif)#en%DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 10.1.121.21
(GigabitEthernet1.121) is down: interface down
R1(config-subif)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console
R1#show ip bgp summ | b Ne
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
22.22.22.22     4        65012      15      22       58    0    0 00:10:55        1
100.1.13.3      4            3    1611    1596       58    0    0 23:54:39       10
R1#ping 22.22.22.22 source loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 22.22.22.22, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#int g1.121
R1(config-subif)#no shu
```

https://t.me/learningnets

```
R1(config-subif)#end
R1#
%SYS-5-CONFIG_I: Configured from console by console
%DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 10.1.121.21(GigabitEthernet1.121) is up: new adjacency
```

R1 should still receive routes from other AS's even if its EBGP peer (R3) goes down.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#router bgp 65012
R1(config-router)#neighbor 100.1.13.3 shutdown
%BGP-3-NOTIFICATION: sent to neighbor 100.1.13.3 6/2 (Administrative Shutdown)
 0 bytes
R1(config-router)#end
R1#
%BGP-5-NBR_RESET: Neighbor 100.1.13.3 reset (Admin. shutdown)
%BGP-5-ADJCHANGE: neighbor 100.1.13.3 Down Admin. shutdown
%BGP_SESSION-5-ADJCHANGE: neighbor 100.1.13.3 IPv4 Unicast topology base removed from session  Admin. shutdown
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show bgp ipv4 unicast
BGP table version is 43, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  1.1.1.1/32       0.0.0.0                  0         32768 ?
 r>i 2.2.2.2/32       2.2.2.2                  0    100      0 ?
 *>i 3.3.3.3/32       2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 4.4.4.4/32       2.2.2.2                  0    100      0 4 ?
 *>i 5.5.5.5/32       2.2.2.2                  0    100      0 4 65056 ?
 *>i 6.6.6.6/32       2.2.2.2                  0    100      0 4 65056 ?
 *>i 9.9.9.9/32       2.2.2.2                  0    100      0 4 65056 3 65009 ?
 *>i 10.10.10.10/32   2.2.2.2                  0    100      0 4 65056 3 65010 ?
 r>i 100.1.13.0/24    2.2.2.2                  0    100      0 4 65056 3 ?
 r>i 100.1.13.1/32    2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 100.1.35.0/24    2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 100.1.39.0/24    2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 100.1.103.0/24   2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 200.1.24.0       2.2.2.2                  0    100      0 4 ?
 *>i 200.1.46.0       2.2.2.2                  0    100      0 4 ?
```

https://t.me/learningnets

```
R1#show ip bgp neighbors 22.22.22.22 routes

BGP table version is 43, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 r>i 2.2.2.2/32       2.2.2.2                  0    100      0 ?
 *>i 3.3.3.3/32       2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 4.4.4.4/32       2.2.2.2                  0    100      0 4 ?
 *>i 5.5.5.5/32       2.2.2.2                  0    100      0 4 65056 ?
 *>i 6.6.6.6/32       2.2.2.2                  0    100      0 4 65056 ?
 *>i 9.9.9.9/32       2.2.2.2                  0    100      0 4 65056 3 65009 ?
 *>i 10.10.10.10/32   2.2.2.2                  0    100      0 4 65056 3 65010 ?
 r>i 100.1.13.0/24    2.2.2.2                  0    100      0 4 65056 3 ?
 r>i 100.1.13.1/32    2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 100.1.35.0/24    2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 100.1.39.0/24    2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 100.1.103.0/24   2.2.2.2                  0    100      0 4 65056 3 ?
 *>i 200.1.24.0       2.2.2.2                  0    100      0 4 ?
 *>i 200.1.46.0       2.2.2.2                  0    100      0 4 ?

Total number of prefixes 14
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#router bgp 65012
R1(config-router)#no neighbor 100.1.13.3 shutdown
R1(config-router)#
R1(config-router)#end%BGP-5-ADJCHANGE: neighbor 100.1.13.3 Up

R1#
```

5.3 - BGP Authentication

```
R3:
router bgp 3
 neighbor 2001:100:1:39::9 password BGP_PASS!
 neighbor 2001:100:1:103::10 password BGP_PASS!



R9:
```

https://t.me/learningnets

```
router bgp 65009
 neighbor 2001:100:1:39::3 password BGP_PASS!



R10:
router bgp 65010
 neighbor 2001:100:1:103::3 password BGP_PASS!
```

5.3 - BGP Authentication Verification

```
R3#show bgp ipv6 unicast neighbors 2001:100:1:39::9 | inc md5
Option Flags: nagle, path mtu capable, md5
, Retrans timeout

R3#show bgp ipv6 unicast neighbors 2001:100:1:103::10 | inc md5
Option Flags: nagle, path mtu capable, md5
, Retrans timeout
```

5.4 - BGP Aggregation

https://t.me/learningnets

```
R9:
interface Loopback150
 ip address 119.150.0.9 255.255.255.255
!
interface Loopback151
 ip address 119.151.0.9 255.255.255.255
!
interface Loopback160
 ip address 119.160.0.9 255.255.255.255
!
interface Loopback163
 ip address 119.163.0.9 255.255.255.255
!
route-map LOOPBACK permit 20
 match interface Loopback150 Loopback151 Loopback160 Loopback163
!
router bgp 65009
 !
 address-family ipv4
  aggregate-address 119.128.0.0 255.192.0.0 summary-only
!
```

5.4 - BGP Aggregation Verification

The single most optimal summary that encompasses these 4 networks is 119.128.0.0/10 - Usable IP addresses: 119.128.0.1 - 119.191.255.254. Note that the 'summary-only' keyword has to be used in order to suppress the subnets of the aggregate.

```
R9#show bg ipv4 unicast | include 119
 *>  119.128.0.0/10   0.0.0.0                          32768 i s>  119.150.0.9/32
   0.0.0.0                  0         32768 ? s>  119.151.0.9/32
   0.0.0.0                  0         32768 ? s>  119.160.0.9/32
   0.0.0.0                  0         32768 ? s>  119.163.0.9/32
   0.0.0.0                  0         32768 ?

R3#show bgp ipv4 unicast | include 119

 *>  119.128.0.0/10   100.1.39.9               0             0 65009 i
```

https://t.me/learningnets

5.5 - BGP Path Selection

```
R2:
ip as-path access-list 1 permit _65056$
!
route-map PATH_SELECTION_TOWARDS_65056 permit 10
 match as-path 1
 set local-preference 200
route-map PATH_SELECTION_TOWARDS_65056 permit 20
!
router bgp 65012
 address-family ipv4
  neighbor 200.1.24.4 route-map PATH_SELECTION_TOWARDS_65056 in
```

5.5 - BGP Path Selection Verification

Before any changes are applied, each edge router in AS 65012 is selecting its directly connected EBGP path as a best path towards AS 65056 destinations. R1 routes towards R3 in AS 3, and R2 routes towards R4 in AS 4.

```
R1#show bgp ipv4 unicast regexp _65056$
BGP table version is 192, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path *>  5.5.5.5/32 100.1.13.3
                            0 3 65056 ? *>  6.6.6.6/32 100.1.13.3
                            0 3 65056 ?

R2#show bgp ipv4 unicast regexp _65056$
BGP table version is 163, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 5.5.5.5/32       1.1.1.1                  0    100      0 3 65056 ?
```

https://t.me/learningnets

```
*>200.1.24.4
                                0 4 65056 ?
* i 6.6.6.6/32       1.1.1.1                  0    100      0 3 65056 ? *>200.1.24.4
                                0 4 65056 ?
```

AS 65012 should prefer to exit via R2 and transit AS 4 after the path selection changes.

https://t.me/learningnets

```
R1#show bgp ipv4 unicast regexp _65056$

BGP table version is 194, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path  *>i 5.5.5.5/32 2.2.2.2
                  0    200      0 4 65056 ?
 *                    100.1.13.3                           0 3 65056 ?  *>i 6.6.6.6/32 2.2.2.2
                  0    200      0 4 65056 ?
 *                    100.1.13.3                           0 3 65056 ?

R2#show bgp ipv4 unicast regexp _65056$

BGP table version is 165, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path  *>  5.5.5.5/32 200.1.24.4
                    200      0 4 65056 ?  *>  6.6.6.6/32 200.1.24.4
                    200      0 4 65056 ?

R1#traceroute 5.5.5.5 source loopback 0

Type escape sequence to abort.
Tracing the route to 5.5.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.12.2 3 msec 1 msec 1 msec
  2 200.1.24.4 [AS 4] 1 msec 6 msec 2 msec
  3 200.1.46.6 [AS 4] 1 msec 1 msec 1 msec
  4 10.1.56.5 7 msec
```

Note that we could have made the changes on R1 instead of R2 by using the same as-path access-list to match on the routes and dropping the local preference to anything lower than the default (50 for example).

https://t.me/learningnets

6.1 - IPv4 over DMVPN

```
R1:
interface Tunnel0
 ip address 10.1.0.1 255.255.255.0
 ip nhrp authentication DMVPN
 ip nhrp map multicast dynamic
 ip nhrp network-id 1
 ip nhrp redirect
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2

R5:
interface Tunnel0
 ip address 10.1.0.5 255.255.255.0
 ip nhrp authentication DMVPN
 ip nhrp map 10.1.0.1 1.1.1.1
 ip nhrp map multicast 1.1.1.1
 ip nhrp network-id 1
 ip nhrp nhs 10.1.0.1
 ip nhrp shortcut
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2

R9:
interface Tunnel0
 ip address 10.1.0.9 255.255.255.0
 ip nhrp authentication DMVPN
 ip nhrp map 10.1.0.1 1.1.1.1
 ip nhrp map multicast 1.1.1.1
 ip nhrp network-id 1
 ip nhrp nhs 10.1.0.1
 ip nhrp shortcut
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2

R10:
interface Tunnel0
 ip address 10.1.0.10 255.255.255.0
 ip nhrp authentication DMVPN
```

https://t.me/learningnets

```
 ip nhrp map 10.1.0.1 1.1.1.1
 ip nhrp map multicast 1.1.1.1
 ip nhrp network-id 1
 ip nhrp nhs 10.1.0.1
 ip nhrp shortcut
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2
```

6.1 - IPv4 over DMVPN Verification

```
R1#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==============================================================================

Interface: Tunnel0, IPv4 NHRP Details
Type:Hub, NHRP Peers:3,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1 5.5.5.5                10.1.0.5    UP 00:01:44     D
     1 9.9.9.9                10.1.0.9    UP 00:01:11     D
     1 10.10.10.10           10.1.0.10    UP 00:01:00     D
R1#show ip nhrp

10.1.0.5/32 via 10.1.0.5
   Tunnel0 created 00:02:03, expire 01:57:56
   Type: dynamic, Flags: unique registered used nhop
   NBMA address: 5.5.5.5
10.1.0.9/32 via 10.1.0.9
   Tunnel0 created 00:01:31, expire 01:58:28
   Type: dynamic, Flags: unique registered used nhop
   NBMA address: 9.9.9.9
10.1.0.10/32 via 10.1.0.10
   Tunnel0 created 00:01:20, expire 01:58:39
   Type: dynamic, Flags: unique registered used nhop
   NBMA address: 10.10.10.10
```

https://t.me/learningnets

6.2 - EIGRP over DMVPN

```
R1:
interface Tunnel0
 ip summary-address eigrp 1 0.0.0.0 0.0.0.0
```

6.2 - EIGRP over DMVPN Verification

```
R10#show ip route eigrp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 10.1.0.1 to network 0.0.0.0

D*    0.0.0.0/0 [90/26880256] via 10.1.0.1, 00:01:59, Tunnel0
      24.0.0.0/32 is subnetted, 1 subnets
D EX     24.24.24.24
           [170/3072] via 10.1.124.24, 00:16:08, GigabitEthernet1.124
R10#show ip route 23.23.23.23
% Subnet not in table
R10#show ip cef 23.23.23.23
0.0.0.0/0
  nexthop 10.1.0.1 Tunnel0
SW4#traceroute 23.23.23.23

Type escape sequence to abort.
Tracing the route to 23.23.23.23

  1 10.1.124.10 0 msec 8 msec 8 msec  2 10.1.0.1
 9 msec 8 msec 84 msec  3 10.1.0.9
 17 msec 42 msec 16 msec
  4 10.1.239.23 26 msec *  42 msec
SW4#traceroute 23.23.23.23
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 23.23.23.23

  1 10.1.124.10 0 msec 8 msec 0 msec  2 10.1.0.9
 9 msec 8 msec 9 msec
  3 10.1.239.23 16 msec *  17 msec
R10#show ip route 23.23.23.23
Routing entry for 23.23.23.23/32
  Known via "nhrp", distance 250, metric 1
  Last update from 10.1.0.9 on Tunnel0, 00:00:26 ago
  Routing Descriptor Blocks:
  * 10.1.0.9, from 10.1.0.9, 00:00:26 ago, via Tunnel0
      Route metric is 1, traffic share count is 1
      MPLS label: none
R10#show ip cef 23.23.23.23

23.23.23.23/32
  nexthop 10.1.0.9 Tunnel0
R10#show ip route nhrp

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 10.1.0.1 to network 0.0.0.0

      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
H        10.1.0.9/32 is directly connected, 00:01:28, Tunnel0
      23.0.0.0/32 is subnetted, 1 subnets
H        23.23.23.23 [250/1] via 10.1.0.9, 00:01:28, Tunnel0
```

6.3 - IPsec over DMVPN

```
R1, R5, R9, R10:
crypto isakmp policy 10
 encr aes 192
 hash sha256
```

https://t.me/learningnets

```
 authentication pre-share
 group 5
!
crypto isakmp key CCIE_DMVPN address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-256-SHA-512 esp-aes 256 esp-sha512-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-256-SHA-512
!
interface Tunnel0
 tunnel protection ipsec profile DMVPN_PROFILE
```

6.3 - IPsec over DMVPN Verification

```
R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
1.1.1.1         9.9.9.9         QM_IDLE           1027 ACTIVE
1.1.1.1         5.5.5.5         QM_IDLE           1025 ACTIVE
1.1.1.1         10.10.10.10     QM_IDLE           1029 ACTIVE
10.10.10.10     1.1.1.1         QM_IDLE           1030 ACTIVE
9.9.9.9         1.1.1.1         QM_IDLE           1028 ACTIVE
5.5.5.5         1.1.1.1         QM_IDLE           1026 ACTIVE
R1#show crypto ipsec sa



interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 1.1.1.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (1.1.1.1/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (10.10.10.10/255.255.255.255/47/0)
   current_peer 10.10.10.10 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 25, #pkts encrypt: 25, #pkts digest: 25
    #pkts decaps: 25, #pkts decrypt: 25, #pkts verify: 25
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 1.1.1.1, remote crypto endpt.: 10.10.10.10
```

https://t.me/learningnets

```
plaintext mtu 1442, path mtu 1500, ip mtu 1500, ip mtu idb (none)
current outbound spi: 0x7DB9D716(2109331222)
PFS (Y/N): N, DH group: none

inbound esp sas:
 spi: 0x1F687366(526938982)
   transform: esp-256-aes esp-sha512-hmac ,
   in use settings ={Transport, }
   conn id: 2241, flow_id: CSR:241, sibling_flags FFFFFFFF80000008, crypto map: Tunnel0-head-0
   sa timing: remaining key lifetime (k/sec): (4608000/3511)
   IV size: 16 bytes
   replay detection support: Y
   ecn bit support: N status: off
   Status: ACTIVE(ACTIVE)
 spi: 0x4D5751A7(1297568167)
   transform: esp-256-aes esp-sha512-hmac ,
   in use settings ={Transport, }
   conn id: 2243, flow_id: CSR:243, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
   sa timing: remaining key lifetime (k/sec): (4607998/3518)
   IV size: 16 bytes
   replay detection support: Y
   ecn bit support: N status: off
   Status: ACTIVE(ACTIVE)

inbound ah sas:

inbound pcp sas:

outbound esp sas:
 spi: 0x1C731E61(477306465)
   transform: esp-256-aes esp-sha512-hmac ,
   in use settings ={Transport, }
   conn id: 2242, flow_id: CSR:242, sibling_flags FFFFFFFF80000008, crypto map: Tunnel0-head-0
   sa timing: remaining key lifetime (k/sec): (4608000/3511)
   IV size: 16 bytes
   replay detection support: Y
   ecn bit support: N status: off
   Status: ACTIVE(ACTIVE)
 spi: 0x7DB9D716(2109331222)
   transform: esp-256-aes esp-sha512-hmac ,
   in use settings ={Transport, }
   conn id: 2244, flow_id: CSR:244, sibling_flags FFFFFFFF80004008, crypto map: Tunnel0-head-0
   sa timing: remaining key lifetime (k/sec): (4607998/3518)
   IV size: 16 bytes
   replay detection support: Y
   ecn bit support: N status: off
```

https://t.me/learningnets

```
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:
R7#traceroute 23.23.23.23
Type escape sequence to abort.
Tracing the route to 23.23.23.23
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.57.5 63 msec 3 msec 7 msec  2 10.1.0.1
 12 msec 8 msec 18 msec  3 10.1.0.9
 15 msec 10 msec 40 msec
  4 10.1.239.23 107 msec *  13 msec
R7#traceroute 23.23.23.23
Type escape sequence to abort.
Tracing the route to 23.23.23.23
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.57.5 70 msec 2 msec 4 msec  2 10.1.0.9
 7 msec 6 msec 4 msec
  3 10.1.239.23 204 msec *  74 msec
R5#show crypto ipsec sa peer 9.9.9.9 | include 47|esp|Transport|encap|decap
   local  ident (addr/mask/prot/port): (5.5.5.5/255.255.255.255/47/0
)   remote ident (addr/mask/prot/port): (9.9.9.9/255.255.255.255/47/0
)    #pkts encaps: 8
, #pkts encrypt: 8, #pkts digest: 8    #pkts decaps: 7
, #pkts decrypt: 7, #pkts verify: 7 inbound esp sas
:        transform: esp-256-aes esp-sha512-hmac
 ,        in use settings ={Transport
, }
        transform: esp-256-aes esp-sha512-hmac ,
        in use settings ={Transport, }
        transform: esp-256-aes esp-sha512-hmac ,
        in use settings ={Transport, } outbound esp sas
:        transform: esp-256-aes esp-sha512-hmac
 ,        in use settings ={Transport
, }
        transform: esp-256-aes esp-sha512-hmac ,
        in use settings ={Transport, }
        transform: esp-256-aes esp-sha512-hmac ,
        in use settings ={Transport, }
```

7.1 - IPv6 EIGRP

https://t.me/learningnets

```
R1:
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
interface GigabitEthernet1.121
 ipv6 eigrp 1
!
interface GigabitEthernet1.122
 ipv6 eigrp 1
!
interface GigabitEthernet1.12
 ipv6 eigrp 1


R2:
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
interface GigabitEthernet1.221
 ipv6 eigrp 1
!
interface GigabitEthernet1.222
 ipv6 eigrp 1
!
interface GigabitEthernet1.12
 ipv6 eigrp 1


R5:
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
interface GigabitEthernet1.57
 ipv6 eigrp 1
!
interface GigabitEthernet1.56
 ipv6 eigrp 1
```

https://t.me/learningnets

```
R6:
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
interface GigabitEthernet1.68
 ipv6 eigrp 1
!
interface GigabitEthernet1.56
 ipv6 eigrp 1


R7:
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
interface GigabitEthernet1.57
 ipv6 eigrp 1


R8:
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
interface GigabitEthernet1.68
 ipv6 eigrp 1


SW1:
ipv6 unicast-routing
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
```

https://t.me/learningnets

```
interface vlan 121
 ipv6 eigrp 1
!
interface vlan 221
 ipv6 eigrp 1


SW2:
ipv6 unicast-routing
ipv6 router eigrp 1
 no shut
!
interface Loopback0
 ipv6 eigrp 1
!
interface vlan 122
 ipv6 eigrp 1
!
interface vlan 222
 ipv6 eigrp 1
```

6.3 - IPv6 EIGRP Verification

```
R1#show ipv6 eigrp neighbors

EIGRP-IPv6 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
2   Link-local address:     Gi1.12                   12 00:00:04    3   100  0  11
    FE80::250:56FF:FE8D:6C80
1   Link-local address:     Gi1.122                  12 00:01:53   13   100  0  20
    FE80::214:A9FF:FEF2:D7C2
0   Link-local address:     Gi1.121                  13 00:09:15    9   100  0  22
    FE80::218:BAFF:FE8B:7BC2

R1#show ipv6 route eigrp

IPv6 Routing Table - default - 19 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
```

https://t.me/learningnets

```
        la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid
        a - Application
D   2001:2:2:2::2/128 [90/130816]
     via FE80::250:56FF:FE8D:6C80, GigabitEthernet1.12
D   2001:10:1:221::/64 [90/3072]
     via FE80::218:BAFF:FE8B:7BC2, GigabitEthernet1.121
     via FE80::250:56FF:FE8D:6C80, GigabitEthernet1.12
D   2001:10:1:222::/64 [90/3072]
     via FE80::214:A9FF:FEF2:D7C2, GigabitEthernet1.122
     via FE80::250:56FF:FE8D:6C80, GigabitEthernet1.12
D   2001:21:21:21::21/128 [90/130816]
     via FE80::218:BAFF:FE8B:7BC2, GigabitEthernet1.121
D   2001:22:22:22::22/128 [90/130816]
     via FE80::214:A9FF:FEF2:D7C2, GigabitEthernet1.122

R1#ping 2001:2:2:2::2 source loopback 0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:2:2:2::2, timeout is 2 seconds:
Packet sent with a source address of 2001:1:1:1::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

7.2 - IPv6 Redistribution

```
R1:
router bgp 65012
 address-family ipv6
  redistribute eigrp 1 include-connected
!
ipv6 router eigrp 1
 redistribute bgp 65012 metric 1000000 10 255 1 1500


R5:
router bgp 65056
 address-family ipv6
  redistribute eigrp 1 include-connected
!
ipv6 router eigrp 1
 redistribute bgp 65056 metric 1000000 10 255 1 1500
```

https://t.me/learningnets

7.2 - IPv6 Redistribution Verification

Mutual redistribution between EIGRPv6 and MP-BGP on R1 and R5 will allow both 'islands' of EIGRPv6 AS 1 to have reachability.

```
R5#show ipv6 eigrp topology

EIGRP-IPv6 Topology Table for AS(1)/ID(5.5.5.5)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 2001:1:1:1::1/128, 1 successors, FD is 5120
        via Redistributed (5120/0)
P 2001:5:5:5::5/128, 1 successors, FD is 128256
        via Connected, Loopback0
P 2001:10:1:12::/64, 1 successors, FD is 5120        via Redistributed
 (5120/0)
P 2001:10:1:222::/64, 1 successors, FD is 5120        via Redistributed
 (5120/0)
P 2001:8:8:8::8/128, 1 successors, FD is 131072
        via FE80::250:56FF:FE8D:3BB2 (131072/130816), GigabitEthernet1.56
P 2001:100:1:39::/64, 1 successors, FD is 5120        via Redistributed
 (5120/0)
P 2001:100:1:103::/64, 1 successors, FD is 5120        via Redistributed
 (5120/0)
P 2001:22:22:22::22/128, 1 successors, FD is 5120        via Redistributed
 (5120/0)
<output omitted>

R8#show ipv6 route eigrp
IPv6 Routing Table - default - 22 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid
       a - Application
EX  2001:1:1:1::1/128 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:2:2:2::2/128 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:3:3:3::3/128 [170/5632]
```

https://t.me/learningnets

```
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
D   2001:5:5:5::5/128 [90/131072]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
D   2001:6:6:6::6/128 [90/130816]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
D   2001:7:7:7::7/128 [90/131328]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:10:1:12::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
D   2001:10:1:56::/64 [90/3072]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
D   2001:10:1:57::/64 [90/3328]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:10:1:121::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:10:1:122::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:10:1:221::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:10:1:222::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:21:21:21::21/128 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:22:22:22::22/128 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:100:1:13::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:100:1:39::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
EX  2001:100:1:103::/64 [170/5632]
     via FE80::250:56FF:FE8D:3BB2, GigabitEthernet1.68
```

R8#traceroute

```
Protocol [ip]: ipv6 Target IPv6 address: 2001:22:22:22::22
Source address: 2001:8:8:8::8
Insert source routing header? [no]:
Numeric display? [no]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Priority [0]:
Port Number [0]:
Type escape sequence to abort. Tracing the route to 2001:22:22:22::22
```

https://t.me/learningnets

```
 1 2001:10:1:68::6 3 msec 1 msec 1 msec  2 2001:10:1:56::5
1 msec 1 msec 1 msec  3 2001:100:1:35::3
10 msec 14 msec 14 msec  4 2001:100:1:13::1
14 msec 15 msec 15 msec
 5 2001:10:1:122::22 14 msec 14 msec 14 msec
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Redistribution

Redistribution Case 1

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, which can be found under the **Resources** tab.

Redistribution Case 1

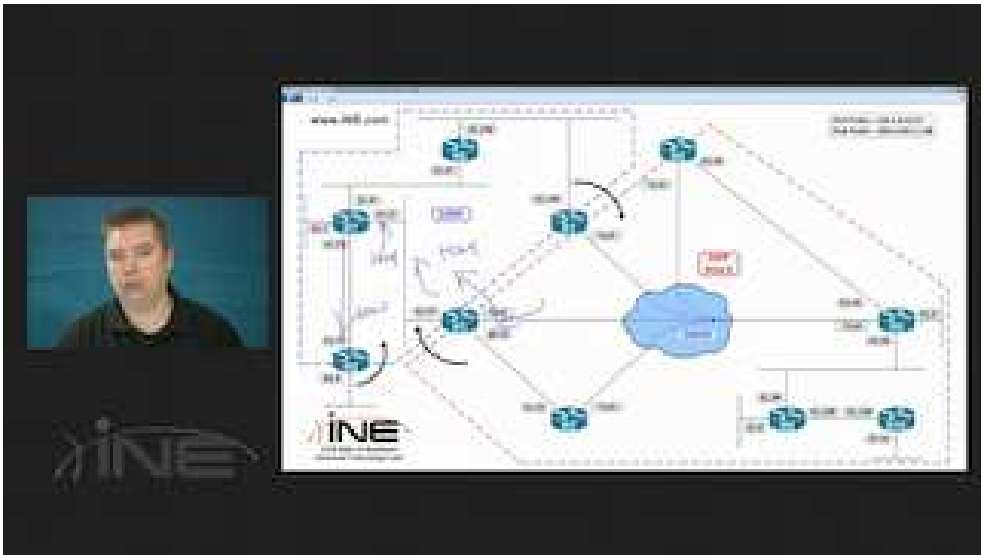
- The network is pre-configured as follows:
  - R1, R2, R3, R5, R8, and R10 run OSPF area 0.
  - R1, R3, R6, R7, and R9 run EIGRP AS 100.
- Refer to the diagram for specific protocol boundaries.
- Redistribution is pre-configured as seen below.

```
R1:
router ospf 1
 redistribute eigrp 100 subnets

R3:
router eigrp 100
 redistribute ospf 1 metric 1000000 1 255 1 1500

R9:
router eigrp 100
 redistribute connected
```

- Describe the problem in the topology, and note the possible solutions.

Video Solution

https://t.me/learningnets



https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Redistribution

Redistribution Case 2

> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, which can be found under the **Resources** tab.

Redistribution Case 2

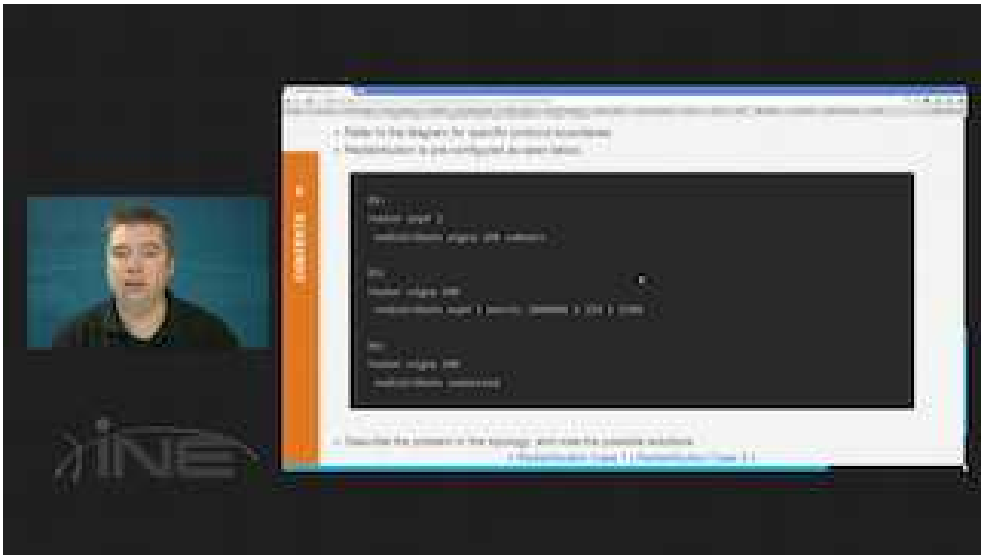
- The network is pre-configured as follows:
  - R1, R2, R3, R4, R5, R8, and R10 run OSPF area 0.
  - R1, R3, R6, R7, and R9 run EIGRP AS 100.
- Refer to the diagram for specific protocol boundaries.
- Redistribution is pre-configured as seen below.

```
R1:
router ospf 1
 redistribute eigrp 100 subnets

R3:
router eigrp 100
 redistribute ospf 1 metric 1000000 1 255 1 1500

R9:
router eigrp 100
 redistribute connected
```

- Describe the problem in the topology, and note the possible solutions.

Video Solution

https://t.me/learningnets



https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Technology Labs - Redistribution

Redistribution Case 3

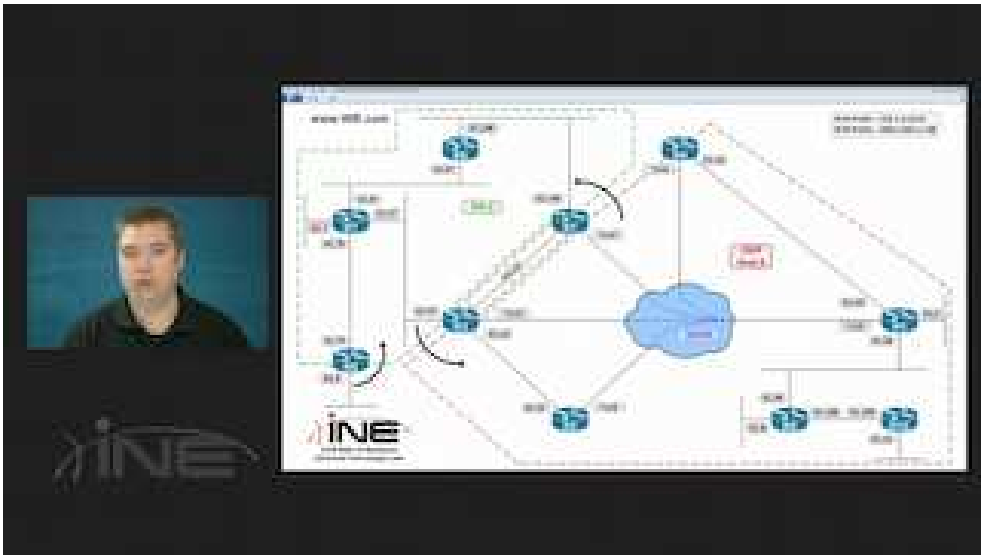
> **A Note On Section Initial Configuration Files**: You must load the initial configuration files for the section, which can be found under the **Resources** tab.

Redistribution Case 3

- The network is pre-configured as follows:
  - R1, R2, R3, R4, R5, R8, and R10 run OSPF area 0.
  - R1, R3, R6, R7, and R9 run RIPv2.
- Refer to the diagram for specific protocol boundaries.
- Redistribution is pre-configured as follows:
  - R1 redistribute OSPF into RIPv2
  - R3 redistributes RIPv2 into OSPF
  - R9 redistributes connected into RIPv2
- Describe the problem in the topology, and note the possible solutions.

Video Solution

https://t.me/learningnets



https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Troubleshooting Labs

Troubleshooting Lab 1 Tasks

> Diagrams and initial configs for this lab are located in the Resources section in the upper-right portion of this page.

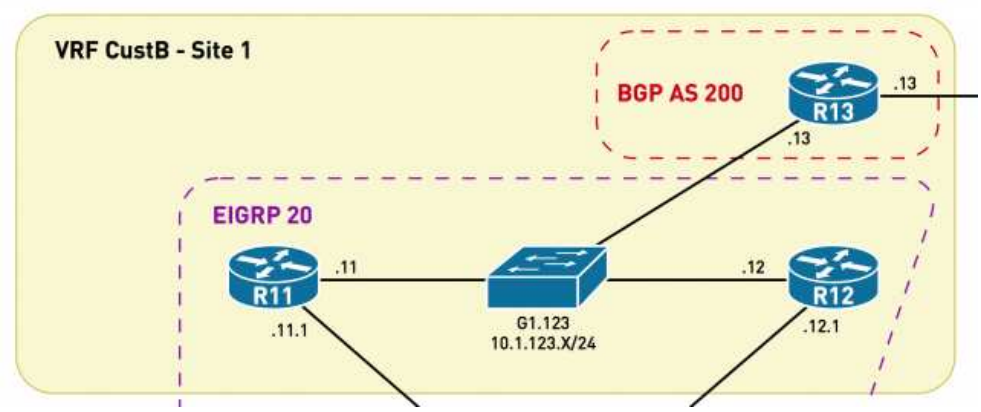
Difficulty Rating (10 highest): 8

Lab Overview

- Do not change the following configuration on any device:
  - Hostname
  - Enable password
  - Console or VTY configuration
- Use the password **cisco** for any authentication.
- Points are awarded for *finding and resolving* faults in the topology. An inserted fault is an introduced break for a scenario that was previously working. Depending on the scenario, fixing inserted faults could require one or multiple command lines on the same or multiple devices.
- The resolution of one incident MAY depend on the resolution of previous incidents. The dependency will not be visible if incidents are resolved in sequence.
- There are NO physical faults in the network.
- Do not change any routing protocol boundaries. Refer to the provided diagram.
- Do not add new interfaces or IP addresses.
- Do not remove any feature configured to resolve a ticket; you must *resolve* the issue, rather than remove the configuration.
- Static default routes are NOT permitted unless preconfigured.
- Routes to null0 that are generated as a result of a dynamic routing protocol solution are permitted.
- Routers do not need to ping themselves when verifying reachability.
- Tunneling and policy-based routing is not permitted unless preconfigured.

https://t.me/learningnets

Ticket 1

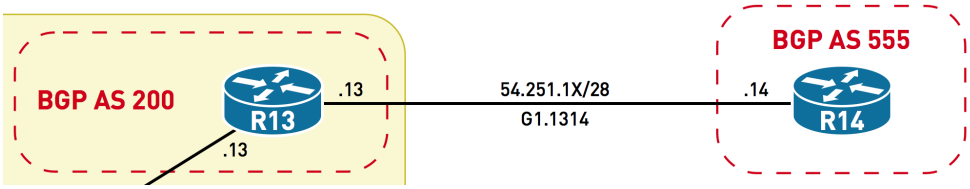


R13 is not forming IPv4 iBGP peerings with R11 and R12. Fix the network so that these IPv4 iBGP peerings are established via the BGP Dynamic Peers feature. Match the output below.

```
R13#show ip bgp summary | be Neig
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
*10.1.123.11    4          200       8      12       40    0    0 00:03:42
        1 *10.1.123.12    4          200       8      12       40    0    0 00:03:41
        1
54.251.1.14     4          555       9      11       40    0    0 00:03:40       21
* Dynamically created based on a listen range command

Dynamically created neighbors: 2, Subnet ranges: 1
```

Score: 2 Points

Ticket 2



R13 is generating a few BGP aggregates, but 30.9.0.0/16 is not being advertised to its peers. Fix the network so that R13 can send this aggregate to its peers. Match the output below.

Note that this verification can be done using any of R13's neighbors, including R14.

```
R13#show ip bgp neighbors 10.1.123.11 advertised-routes
```

https://t.me/learningnets

```
BGP table version is 40, local router ID is 122.1.1.13
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
r>   30.9.0.0/16      0.0.0.0                      100  32768 555 ?

 *>  30.10.9.4/32     54.251.1.14              0             0 555 ?
 *>  30.10.9.5/32     54.251.1.14              0             0 555 ?
 *>  30.10.9.6/32     54.251.1.14              0             0 555 ?
 *>  40.7.5.3/32      54.251.1.14              0             0 555 ?
 *>  40.7.7.2/32      54.251.1.14              0             0 555 ?
 *>  40.8.7.4/32      54.251.1.14              0             0 555 ?
 *>  40.8.7.5/32      54.251.1.14              0             0 555 ?
 *>  40.8.7.6/32      54.251.1.14              0             0 555 ?
 *>  54.251.1.0/28    0.0.0.0                  0         32768 i
 r>  60.7.7.0/25      0.0.0.0                      100  32768 555 ?
 *>  77.9.9.7/32      54.251.1.14              0             0 555 ?
 *>  77.9.9.8/32      54.251.1.14              0             0 555 ?
 *>  77.9.9.9/32      54.251.1.14              0             0 555 ?
 --More--
```

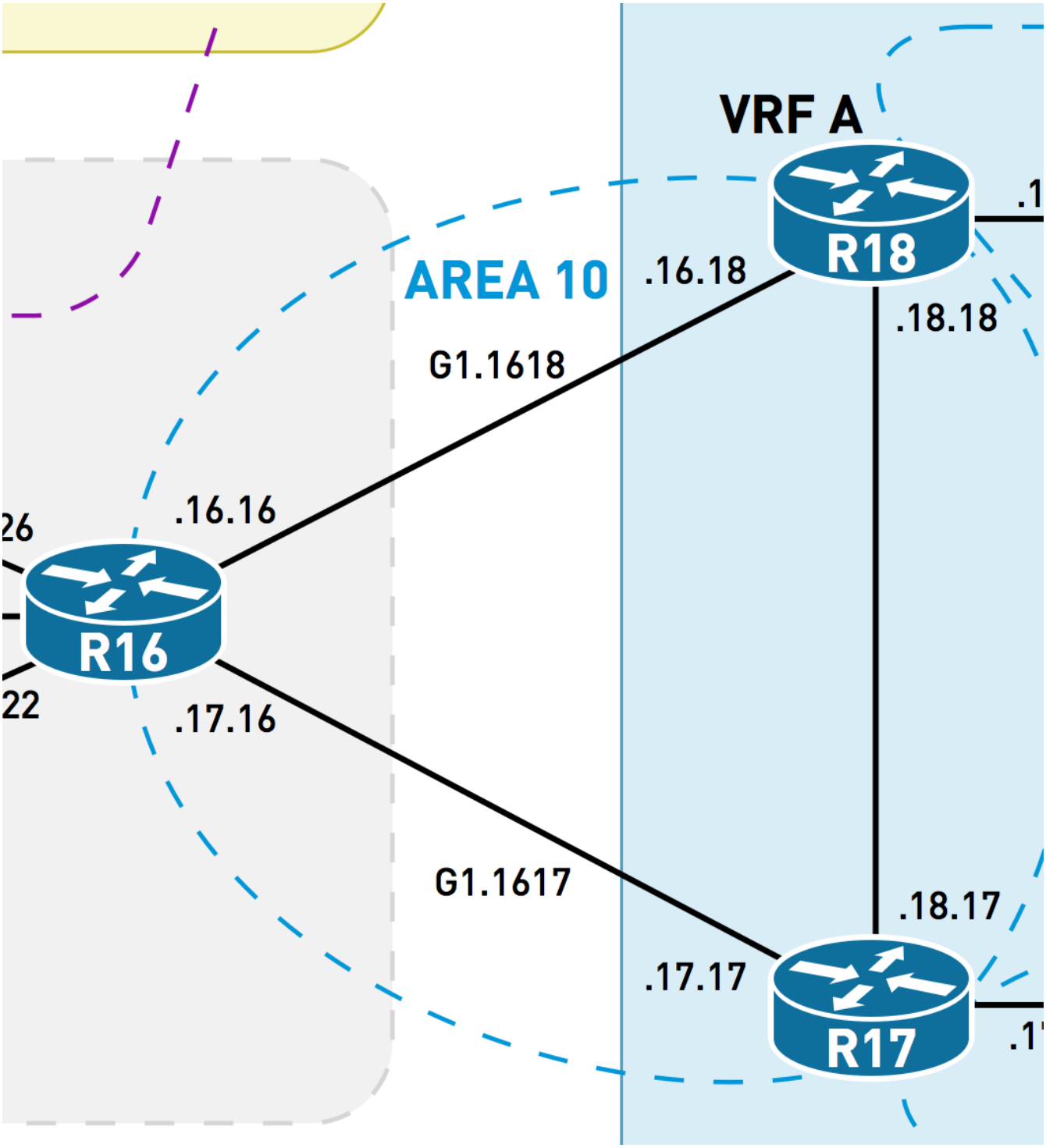
Restrictions:

- Do not modify configurations on R14.

Score: 2 Points

Ticket 3

https://t.me/learningnets



R17 needs to have 100% LFA coverage for its Gi1.1617 interface. Fix the OSPF network so that R17 can achieve 100% LFA coverage for prefixes using its Gi1.1617 interface. Match the output below.

```
R17#show ip ospf fast-reroute prefix-summary


            OSPF Router with ID (122.1.1.17) (Process ID 22)
                    Base Topology (MTID 0)
```

https://t.me/learningnets

```
<output snipped>

Area 10:

Interface        Protected    Primary paths    Protected paths Percent protected
                              All  High   Low   All  High   Low    All High  Low
Gi1.1718               Yes     6     3     3     2     0     2    33%   0%  66%
Gi1.1617               Yes    25    17     8    25    17     8   100% 100% 100%


Area total:                   31    20    11    27    17    10    87%  85%  90%
```

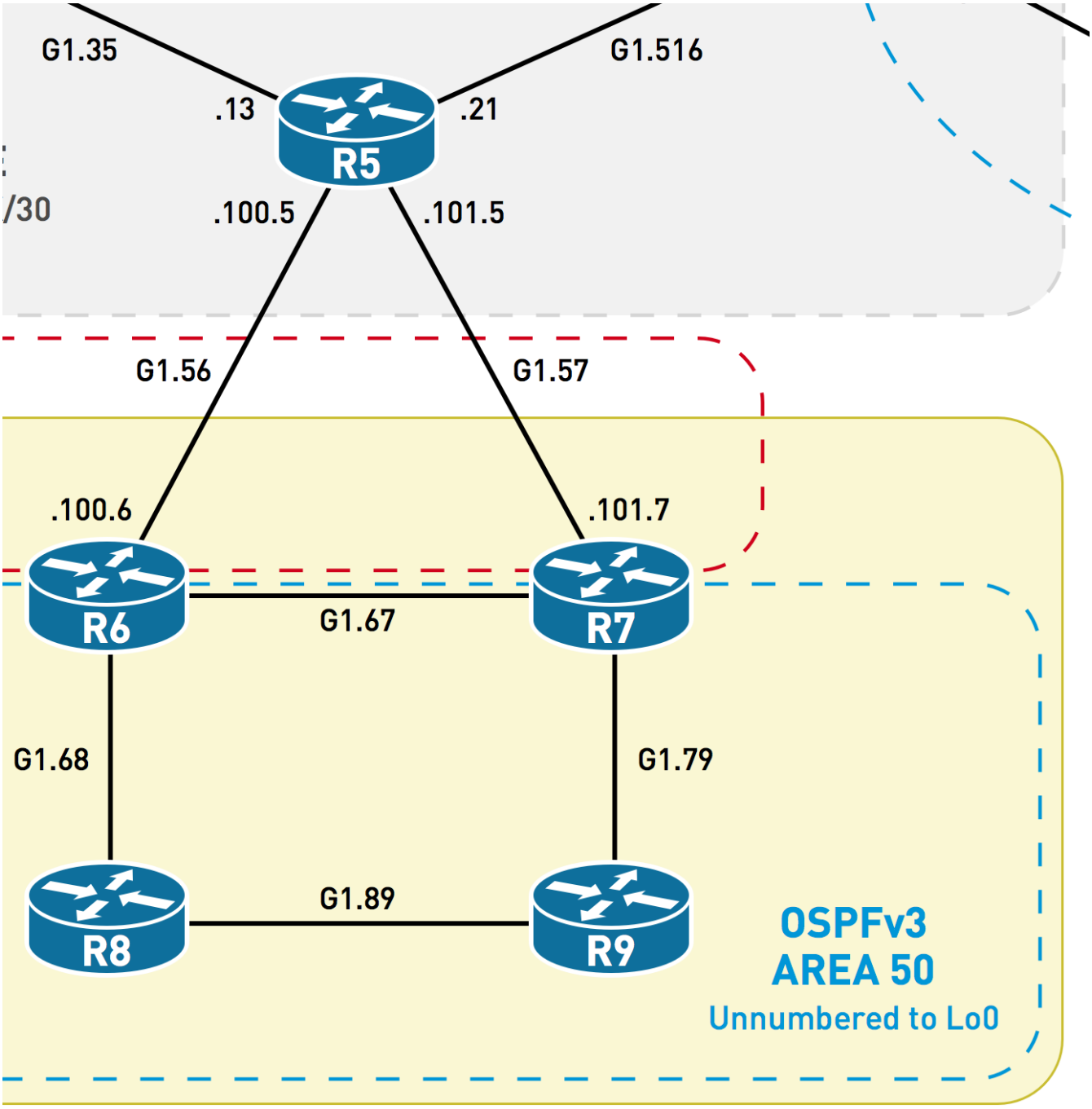
Restrictions:

- Do not change the OSPF area boundries.

Score: 2 Points

Ticket 4

https://t.me/learningnets



R9 cannot ping R20's loopback 122.1.1.20/32. Fix the network so that R9 can ping 122.1.1.20. Match the output below.

```
R9#ping 122.1.1.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 122.1.1.20, timeout is 2 seconds:!!!!!

Success rate is 100 percent (5/5),  min/avg/max = 1/8/20 ms
```

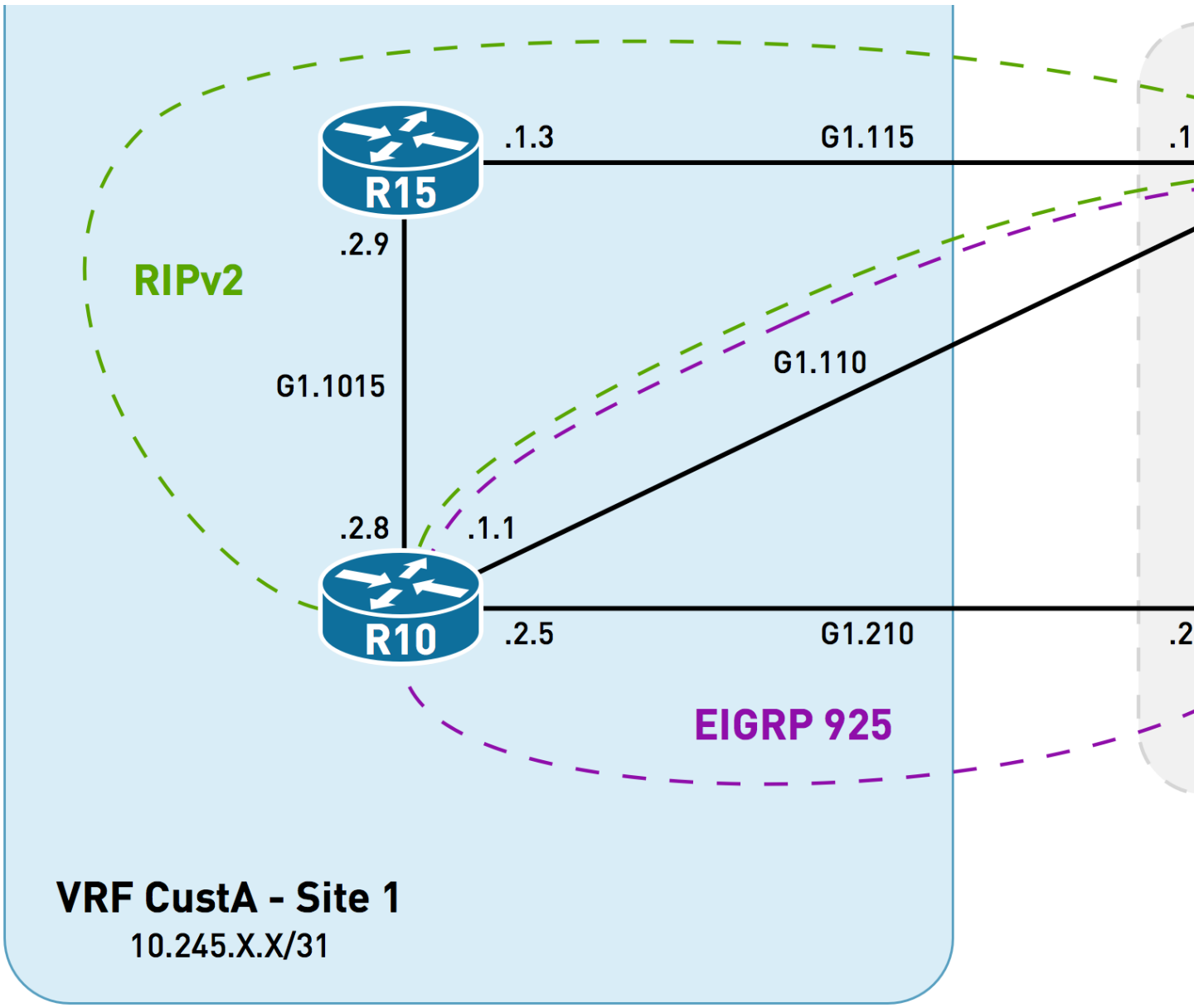
https://t.me/learningnets

Restrictions:

- Do not modify any access-lists or make any interface-level changes.
- Do not change routing protocol metrics (link, area). This does NOT include metric set on redistribution.
- Make a maximum of two changes in the network to solve this ticket.

Score: 2 Points

Ticket 5



A customer is complaining that PE router R2 is preferring to reach R10's IPv6

https://t.me/learningnets

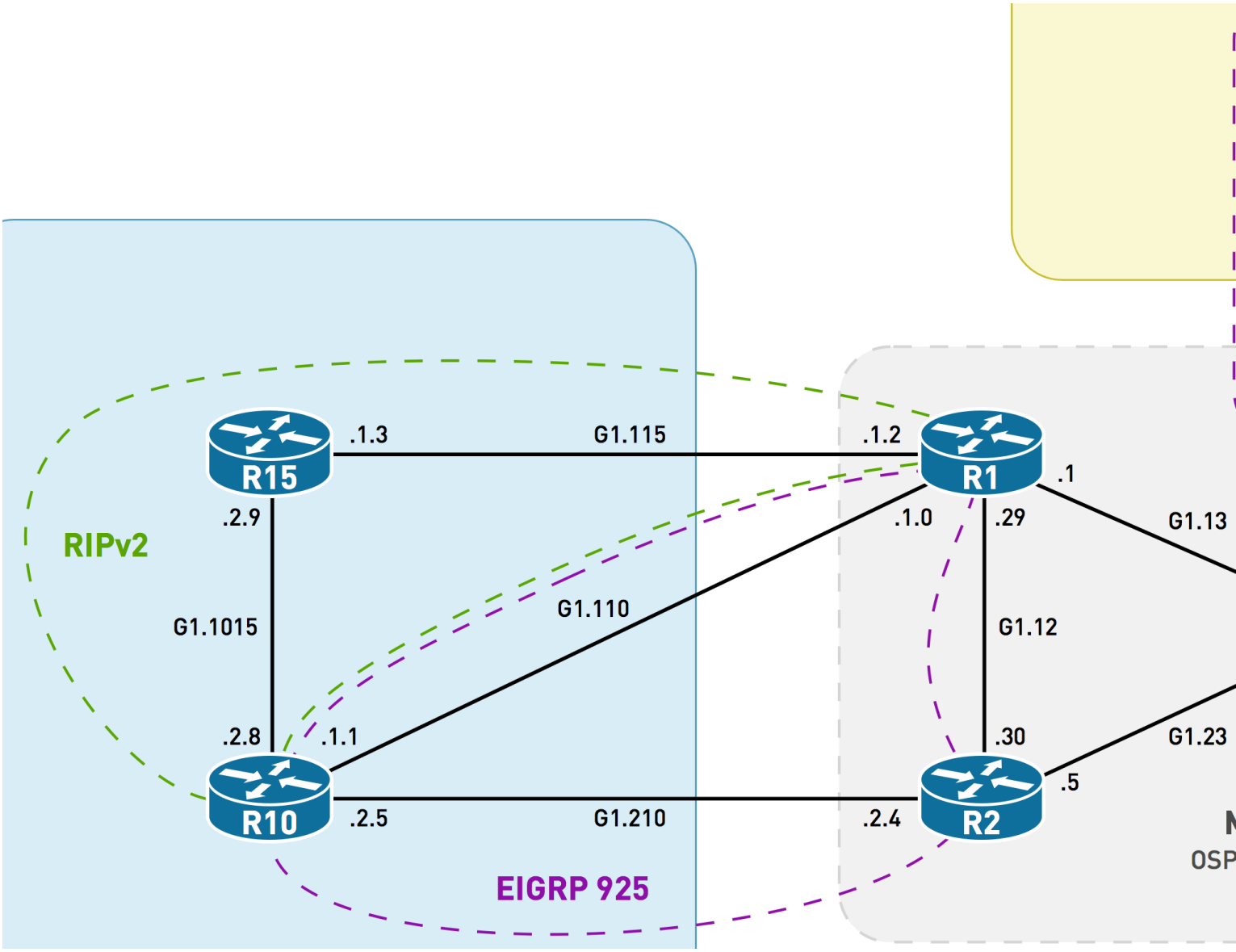
G1.1015, 2001:10:245:2::8/127, via the MPLS cloud. Fix the network so that R2 prefers to reach 2001:10:245:2::8/127 via the PE-CE connection toward R10. Match the output below.

```
R2#show ipv6 route vrf CustA 2001:10:245:2::8/127
Routing entry for 2001:10:245:2::8/127 Known via "eigrp 925"
, distance 90, metric 15360, type internal
  Redistributing via bgp 101
  Route count is 1/1, share count 0
  Routing paths:    FE80::250:56FF:FE8D:6E29, GigabitEthernet1.210

      Last updated 00:01:21 ago
```

Score: 2 Points

Ticket 6

https://t.me/learningnets



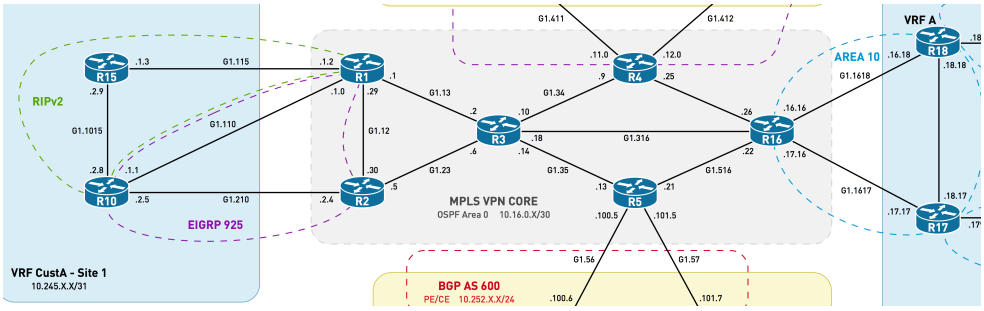
R10 cannot reach R11's loopback0, 122.1.1.11. Fix the network so that R10 can reach R11's loopback0 while sourcing packets from R10's loopback0. Match the output below.

```
R10#ping 122.1.1.11 source loopback 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 122.1.1.11, timeout is 2 seconds:
Packet sent with a source address of 122.1.1.10 !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/20 ms
```

Score: 3 Points

https://t.me/learningnets

Ticket 7



R17 cannot receive a multicast stream for group 227.7.7.7, which it has joined on its Loopback0 interface. The source of this stream is R15's Loopback0. Fix the network so that R17 can receive the multicast flow sent from R15's Loopback0. Match the output below.

https://t.me/learningnets

```
R15#ping

Protocol [ip]:  Target IP address:227.7.7.7
Repeat count [1]:100

Datagram size [100]:
Timeout in seconds [2]:  Extended commands [n]:Y
Interface [All]:Loopback0

Time to live [255]:  Source address or interface:122.1.1.15

Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 227.7.7.7, timeout is 2 seconds:
Packet sent with a source address of 122.1.1.15
.
Reply to request 1 from 172.23.18.17, 8 msReply to request 1 from 122.1.1.17, 8 ms..
Reply to request 4 from 122.1.1.17, 7 ms
Reply to request 5 from 122.1.1.17, 8 ms
Reply to request 6 from 122.1.1.17, 4 ms
Reply to request 7 from 122.1.1.17, 4 ms
Reply to request 8 from 122.1.1.17, 5 ms
Reply to request 9 from 122.1.1.17, 6 ms
Reply to request 10 from 122.1.1.17, 3 ms
```

Restrictions:

- You may use one static mroute as part of your solution to solve this ticket.
- Do not add additional RPs or MSDP peerings.

Hints:

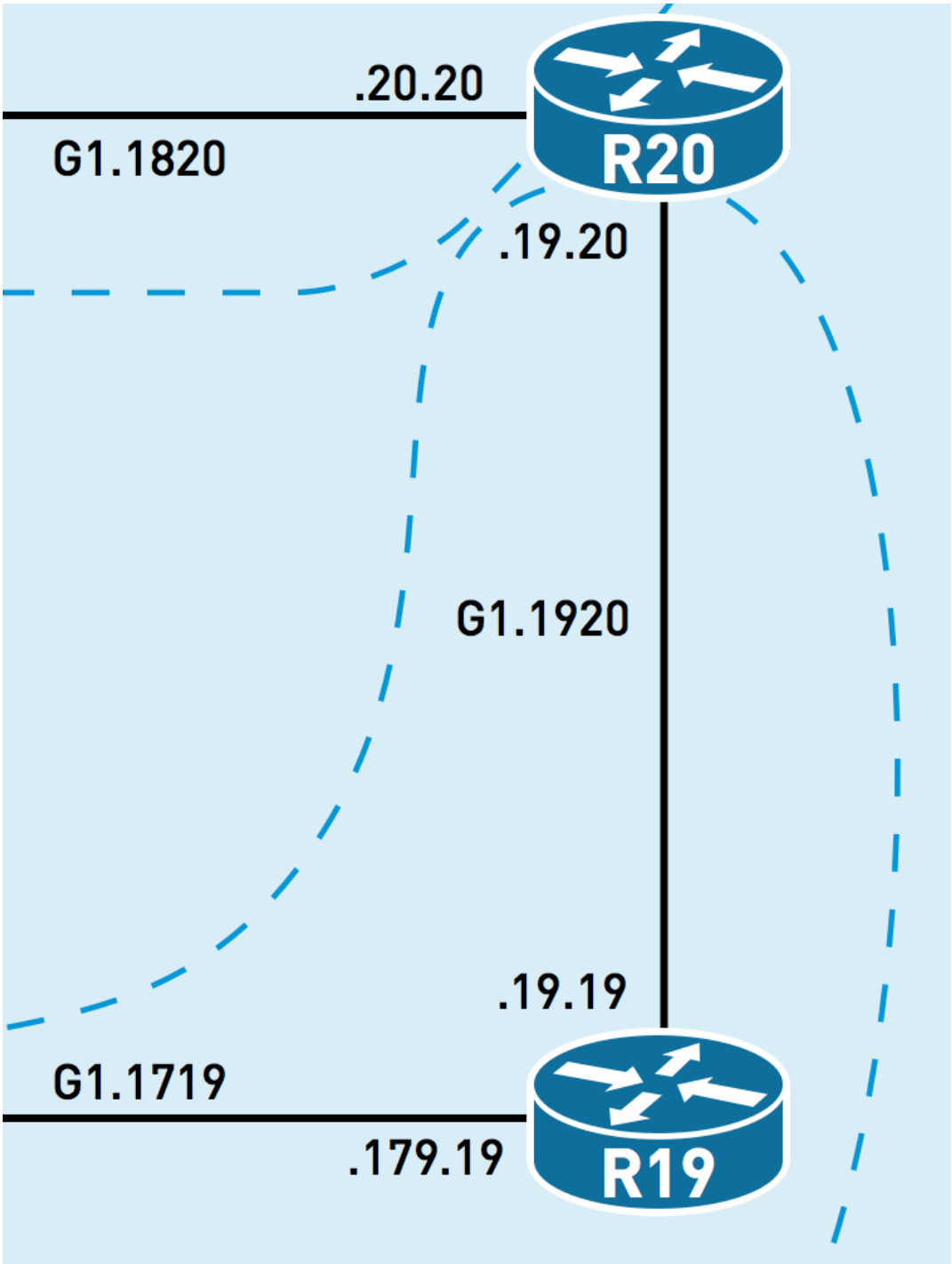
- R10 and R18 are using a GRE Tunnel (Tunnel 10) to tunnel multicast packets over the MPLS network. The MPLS network is not involved in the transmission of this multicast flow.

https://t.me/learningnets

- R18 has all of its interfaces in VRF A, including Tunnel 10. Corresponding "show" commands and other configuration commands must reference this VRF accordingly.

Score: 3 Points

Ticket 8



R19 cannot traceroute to R20's loopback via its directly connected link. Fix the network so that R19 can traceroute to R20's loopback via IPv4 and IPv6. Match the

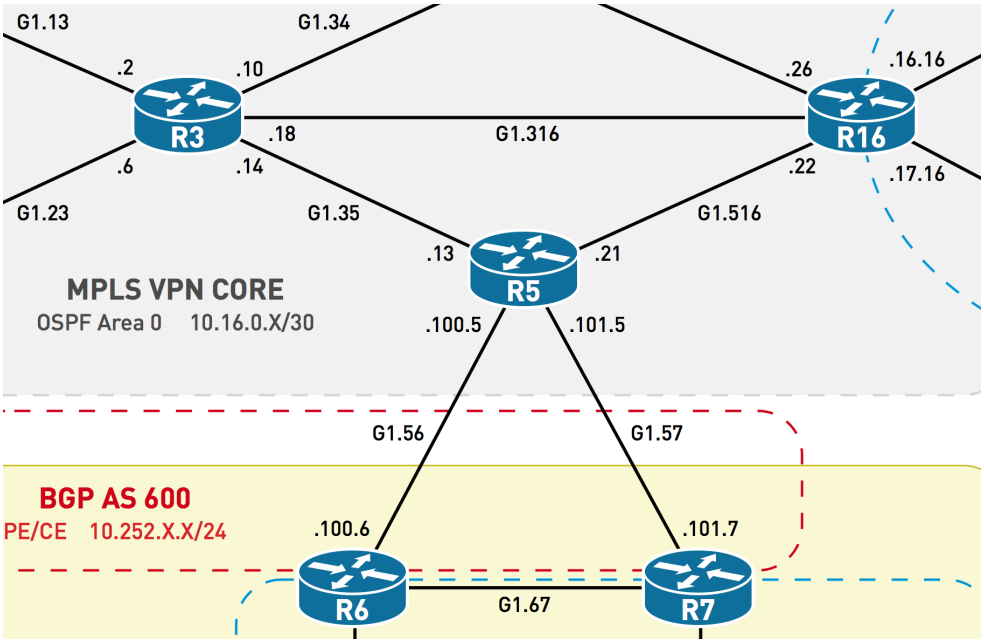
https://t.me/learningnets

output below.

```
R19#traceroute 122.1.1.20
Type escape sequence to abort.
Tracing the route to 122.1.1.20
VRF info: (vrf in name/id, vrf out name/id) 1 172.23.19.20 3 msec 1 msec 1 msec

R19#traceroute 2001:122:1:1::20
Type escape sequence to abort.
Tracing the route to 2001:122:1:1::20
 1 2004:172:23:20::20 3 msec 2 msec 1 msec
```

Score: 2 Points

Ticket 9



Ensure that R6 can traceroute to R10's loopback 0 when sourcing traffic from R6's loopback0, using the most optimal path through the MPLS cloud (lowest number of hops). While resolving this ticket, ensure that the BFD session between R5 and R3 is up. Match the output below.

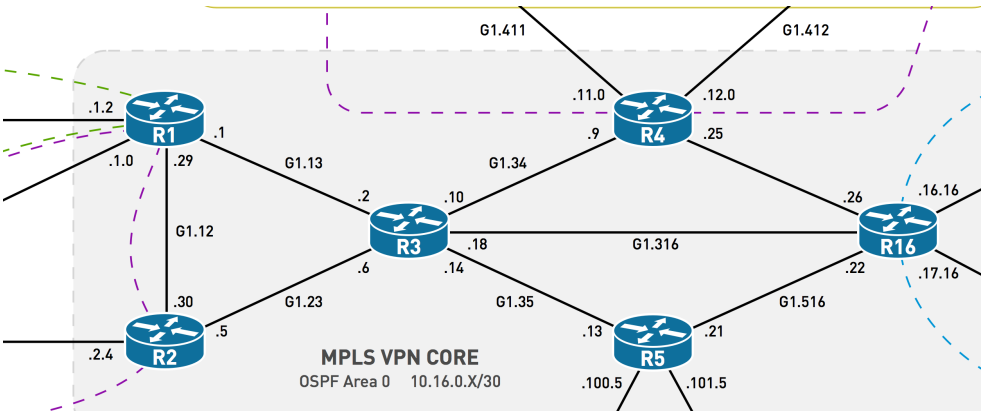
```
R6#traceroute 122.1.1.10 sou lo0
Type escape sequence to abort.
Tracing the route to 122.1.1.10
VRF info: (vrf in name/id, vrf out name/id)
  1 12.252.100.5 2 msec 2 msec 1 msec
  2 10.16.0.14 [MPLS: Labels 18/26 Exp 0] 2 msec 2 msec 2 msec
```

https://t.me/learningnets

```
  3 10.245.1.0 [AS 101] [MPLS: Label 26 Exp 0] 2 msec 2 msec 2 msec
  4 10.245.1.1 [AS 101] 2 msec 2 msec 2 msec

R3#show bfd neighbors


IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
10.16.0.1                            4393/4102       Up        Up        Gi1.13
10.16.0.5                            4383/4099       Up        Up        Gi1.23
10.16.0.9                            4394/4102       Up        Up        Gi1.34
10.16.0.13                           8881/8627       Up        Up        Gi1.35

10.16.0.17                           4399/4103       Up        Up        Gi1.316
```

Score: 2 Points

Ticket 10



The Site 2 Cust-A customer is paying the SP for additional redundancy, but PE R16 is having BGP convergence issues and is not fulfilling the SLA. Fix the network so that R16 has a pre-installed backup path in the RIB/FIB for prefixes originated by R10 and R15. Do your testing with the 10.245.2.8/31 prefix, the link between R10 and R15. A "repair" path should appear in the FIB of R16 as illustrated below.

https://t.me/learningnets

```
R16#show ip cef vrf CustA 10.245.2.8/31 detail

10.245.2.8/31, epoch 0, flags rib defined all labels
  recursive via 122.1.1.1 label 24
    nexthop 10.16.0.18 GigabitEthernet1.316 label 18 recursive via 122.1.1.2 label 35, repair

    nexthop 10.16.0.18 GigabitEthernet1.316 label 17
```

Score: 2 Points

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Troubleshooting Labs

Troubleshooting Lab 1 Solutions

Ticket 1 Faults

- The peers that are supposed to be formed dynamically have static neighbor statements.
- The limit of dynamic peers has been set to 1 on R13.
- R12 is peering with the wrong address (an address that does not exist).

Ticket 1 Solutions

```
R13:
router bgp 200
 bgp listen limit 2
 no neighbor 10.1.123.11 peer-group AS_200
 no neighbor 10.1.123.12 peer-group AS_200

R12:
router bgp 200
 no neighbor 10.1.123.113 remote-as 200
 neighbor 10.1.123.13 remote-as 200
!
address-family ipv4
 neighbor 10.1.123.13 activate
 neighbor 10.1.123.13 send-community
 neighbor 10.1.123.13 default-originate route-map default
```

Ticket 2 Faults

- R14 is sending 30.9.10.2/32 tagged with no-advertise community. as-set on R13 in causing this community to be inherited.

https://t.me/learningnets

Ticket 2 Solutions

```
R13:
route-map STRIP_OFF_COMMUNITIES permit 10
 set community none
 !
router bgp 200
 !
 address-family ipv4
 aggregate-address 30.9.0.0 255.255.0.0 as-set summary-only attribute-map STRIP_OFF_COMMUNITIES
```

Ticket 3 Faults

- R17 is only using LFA for high-priority prefixes in a single area.
- R18's link to R16 has been changed to a cost of 2.

Ticket 3 Solutions

```
R17:
router ospf 22
 fast-reroute per-prefix enable prefix-priority low

R18:
interface GigabitEthernet1.1618
 ip ospf cost 1
```

Ticket 4 Faults

- R9 has a service-policy dropping all traffic to R20's Loopback0 - need to route around this.
- R9 has a static route to null0 for the Forwarding Address the Type-7 from R6.

Ticket 4 Solutions

```
R7:
router ospfv3 50
 area 50 nssa no-redistribution
```

https://t.me/learningnets

```
R9:
ip route 122.1.1.6 255.255.255.255 Null0 255
```

Ticket 5 Faults

- R2 has control-plane policing, which is dropping all IPv6 protocol 88 traffic.
- R10 has an incorrect hmac-sha-256 password for the EIGRPv6 peering.

Ticket 5 Solutions

```
R2:
ipv6 access-list default
 no seq 10
 seq 10 deny 88 any any
!
router eigrp CustA
 address-family ipv6 unicast vrf CustA autonomous-system 925
 !
  af-interface GigabitEthernet1.210
  !
    no authentication mode hmac-sha-256 v6P4SS!
    authentication mode hmac-sha-256 v6PASS!
```

Ticket 6 Faults

- The RR does not have R4 activated for VPNv4, or have R4 as an RR client.
- R4 has TDP configured as the label protocol on its interface.
- There is an LDP password misconfiguration between R3 and R4.
- R4 has a table map blocking R10's loopback from being installed in the RIB.
- R4 is not importing the correct RT values for CustA, missing RT import 100:100.

Ticket 6 Solutions

```
R3:
router bgp 101
 address-family vpnv4
 neighbor 122.1.1.4 activate
 neighbor 122.1.1.4 inherit peer-policy RR
```

https://t.me/learningnets

```
R4:
vrf definition CustB
 route-target import 100:100
!
interface GigabitEthernet1.34
 mpls label protocol ldp
!
mpls ldp neighbor 122.1.1.3 password LDP!PASS
!
route-map ldpadj permit 10
```

Ticket 7 Faults

- IPSec Phase I and Phase II parameters between R10 and R18 are mismatched.
- MSDP authentication is missing on R10.
- MSDP connect-source is misconfigured on R10 - must use Tunnel10 as the source.
- Static mroute for R15's Loopback0 is missing on R18 - needed to pass RPF check.

Ticket 7 Solutions

```
R10:
crypto isakmp policy 18
 hash sha512
!
crypto ipsec transform-set ENTERPRISE_ENCRYPTION esp-des esp-sha-hmac
!
ip msdp peer 110.18.10.18 connect-source Tunnel10
!
ip msdp password peer 110.18.10.18 MsDpP@sS!

R18:
ip mroute vrf A 122.1.1.15 255.255.255.255 Tunnel10
```

Ticket 8 Faults

- PPPoE session is down because of a CHAP authentication issue.
- IPCP address not received by R19 because of a pool name misconfiguration on R20.
- OSPF network type is misconfigured on R20's virtual-template.

https://t.me/learningnets

- Distribute list on R19 is blocking 2001:122:1:1::20/128 from being installed.

Ticket 8 Solutions

```
R19:
username R20 password 0 CCIE
!
no ipv6 prefix-list tunnelme seq 5 deny 2001:122:1:1::20/128
ipv6 prefix-list tunnelme seq 5 per 2001:122:1:1::20/128

R20:
interface Virtual-Template1
 ip ospf network point-to-point
 peer default ip address pool default

#clear PPP session on R19/20 as last step#

clear ppp all
```

Ticket 9 Faults

- EEM scripts are bouncing G1.35 on R5.
- BFD authentication issue between R3 and R5.

Ticket 9 Solutions

https://t.me/learningnets

```
R5:
event manager applet noname
 event none
!
event manager applet noname1
 event none
!
bfd-template single-hop default
 authentication meticulous-sha-1 keychain bfddefault
!
key chain bfddefault
 key 0
   key-string CC!E
```

Ticket 10 Faults

- PIC configuration missing on R10.
- Export map on R2 is attaching an incorrect route-target to 10.245.2.8/31.

Ticket 10 Solutions

```
R16:
router bgp 101
 address-family ipv4 vrf CustA
 bgp additional-paths install

R2:
route-map vrfcusta permit 10
 set extcommunity rt 100:100
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Troubleshooting Labs

Troubleshooting Lab 2 Solutions

Ticket 1 Faults

- R19, a spoke at the Berlin site, has the wrong group-id configured under the Tunnel.
- R1, a hub at HQ, has an incorrect policy applied to the Tunnel.

Ticket 1 Solutions

```
R19:
interface Tunnel0
 no ip nhrp group Eur0pe
 ip nhrp group Europe

R1:
interface Tunnel0
 no ip nhrp map group Europe service-policy output Europe
 ip nhrp map group Europe service-policy output Europe_parent
```

Ticket 2 Faults

- R1 and R2 have the prefix-list referenced by the leak-map misconfigured.
- R7 has a weight of 32769 configured toward the PE.
- R7 has an offset-list configured, bumping up the metric of routes received from the hubs. This may seem harmless depending on the order in which the tickets are solved.

Ticket 2 Solutions

```
R1:
no ip prefix-list server-subnets seq 10 permit 180.10.153.0/24
ip prefix-list server-subnets seq 10 permit 180.10.153.0/24 le 32
```

https://t.me/learningnets

```
R2:
no ip prefix-list server-subnets seq 10 permit 180.10.153.0/24
ip prefix-list server-subnets seq 10 permit 180.10.153.0/24 le 32

R7:
router bgp 65004
 address-family ipv4
  neighbor 180.10.147.1 weight 4000
!
ip access-list standard 4
 no 10
 10 deny 180.10.153.0
!

R7#clear ip bgp 180.10.147.1 all | clear ip router *
```

Ticket 3 Faults

- Tokyo and the Colo site are using the same BGP ASN.
- R17 has a static route in the VRF ColoSite table to 180.10.200.100/32 to null0.

Ticket 3 Solutions

```
R9:
router bgp 65003
 address-family ipv4
  neighbor 180.10.119.1 allowas-in

R17:
ip route vrf Colo-Site 180.10.200.100 255.255.255.255 Null0 255
```

Ticket 4 Faults

https://t.me/learningnets

- The prefix list referenced by the route-map on R5 is not matching the correct prefix in the redistribution of RIP into OSPF.
- R5 has no usable address to set as the Forwarding Address.
- R2 has a static route pointing out of G1.210 for both of R5's possible Forwarding Addresses.

Ticket 4 Solutions

```
R5:
no ip prefix-list rip-subs seq 5 permit 180.1.100.100/32
ip prefix-list rip-subs seq 5 permit 180.10.100.100/32
!
interface Loopback0
 no ip ospf 2 area 34
 ip ospf 1 area 34

R4:
router ospf 1
 area 34 nssa translate type7 suppress-fa
```

Ticket 5 Faults

- R15 has a misconfigured network statement for 192.168.100.15/32.
- R13 and R15 have the same router-id, derived from their Anycast Loopback2.
- R19 has a manual "neighbor" statement toward R13.

Ticket 5 Solutions

```
R13:
router eigrp ABC_COMPANY
 address-family ipv4 unicast autonomous-system 56
 !
  eigrp router-id 172.16.1.13
  !
  neighbor 180.10.139.19 GigabitEthernet1.1319

R15:
router eigrp ABC_COMPANY
 address-family ipv4 unicast autonomous-system 56
 !
```

https://t.me/learningnets

```
eigrp router-id 172.16.1.15
!
no network 192.168.15.1 0.0.0.0
network 192.168.100.15 0.0.0.0
```

Ticket 6 Faults

- R10, "the internet," is NATing R6's underlay address to an IP that is not advertised in BGP - 30.9.0.1. R10 has Loopback44, which is already advertised into BGP and has "ip nat outside" already configured (hint).
- R12 and R2 have incorrect IPSec Phase I parameters for the spoke-to-spoke tunnel to form.
  - It is not necessary to also fix the Phase I parameters of R2 for this task; only one hub in necessary for this ticket to work.

Ticket 6 Solutions

```
R10:
no ip nat inside source static 44.2.5.5 30.9.0.1
ip nat inside source static 44.2.5.5 44.6.6.6

R2:
no  crypto isakmp key CORP_KEY address 44.2.5.5
crypto isakmp key CORP_KEY address 44.6.6.6

R12:
no  crypto isakmp key CORP_KEY address 44.2.5.5
crypto isakmp key CORP_KEY address 44.6.6.6
```

Ticket 7 Faults

- R5 is setting the `stub` flag in the OSPFv3 NSSA area.
- R16, the BGP RR at HQ, is missing the `route-reflector-client` keyword under the template policy.
- R1 has a misconfigured network statement.

Ticket 7 Solutions

https://t.me/learningnets

```
R1:
router bgp 6618605
 !
 address-family ipv6
  no network 2001:192:168:10::1/128
  network 2001:192:168:1::1/128

R5:
router ospfv3 1
 !
 address-family ipv6 unicast
  no area 34 stub
  area 34 nssa

R16:
router bgp 6618605
 template peer-policy IPv6_iBGP_POLICY
  route-reflector-client
```

Ticket 8 Faults

- R14 is not injecting the BGP routes into the NSSA.
- R14 is doing the Type-7 to Type-5 translation because of the Loopback1 in area 55.

Ticket 8 Solutions

```
R19:
router ospf 51 vrf HQ
 no area 51 nssa
 area 51 nssa
!
interface Loopback1
 no ip ospf 51 area 55
 ip ospf 51 area 51
```

Ticket 9 Faults

- IPSec Phase I and Phase II parameters are mismatched between R20 and the hubs.
- R1, the point of redistribution between OSPFv3 and EIGRPv6, has a misconfigured prefix-list that is filtering R4's Loopback1 network.

https://t.me/learningnets

- The seed metric is not configured on R1's OSPFv3 to EIGRPv6 redistribution.

Ticket 9 Solutions

```
R20:
crypto isakmp policy 10
 encr aes
!
crypto ipsec transform-set CORP_TRANS esp-aes esp-sha-hmac

R1:
no ipv6 prefix-list permit-out seq 10 permit 2001:172:16:1::4/127
ipv6 prefix-list permit-out seq 10 permit 2001:172:16:1::4/128
!
route-map permit-out permit 10
 set metric 100000 100 255 1 1500
```

Ticket 10 Faults

- R16, the RP/BSR in the HQ network, has control-plane policing that drops all PIM traffic.
- R1 does not have PIM enabled on the Loopback0.

Ticket 10 Solutions

```
R1:
interface Loopback0
 ip pim sparse-mode

R16:
ip access-list extended default
 no 10
 10 deny pim any any
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Advanced Troubleshooting Labs

Troubleshooting Lab 2 Tasks

> Diagrams and initial configs for this lab are located in the Resources section in the upper-right portion of this page.

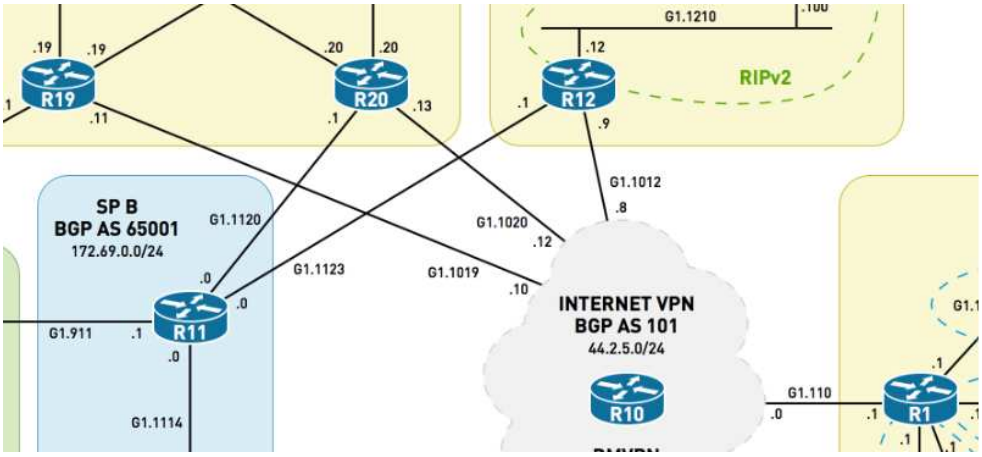
Difficulty Rating (10 highest): 7

Lab Overview

- Do not change the following configuration on any device:
  - Hostname
  - Enable password
  - Console or VTY configuration
- Use the password **cisco** for any authentication.
- Points are awarded for *finding and resolving* faults in the topology. An inserted fault is an introduced break for a scenario that was previously working. Depending on the scenario, fixing inserted faults could require one or multiple command lines on the same or multiple devices.
- The resolution of one incident MAY depend on the resolution of previous incident(s).The dependency will not be visible if incidents are resolved in sequence.
- There are NO physical faults in the network.
- Do not change any routing protocol boundaries. Refer to the provided diagram.
- Do not add new interfaces or IP addresses.
- Do not remove any feature configured to resolve a ticket; you must *resolve* the issue, rather than remove the configuration.
- Static default routes are NOT permitted unless preconfigured.
- Routes to null0 that are generated as a result of a dynamic routing protocol solution are permitted.
- Routers do not need to ping themselves when verifying reachability.
- Tunneling and policy-based routing is not permitted unless preconfigured.

https://t.me/learningnets

Ticket 1



R1 is unable to apply proper QoS policies towards R19. Fix the network to match the output below:

```
R1#show policy-map multipoint tunnel 0 output | begin 44.2.5.11

 Interface Tunnel0 <--> 44.2.5.11

   Service-policy output: Europe_parent

     Class-map: class-default (match-any)
       52 packets, 7281 bytes
       5 minute offered rate 0000 bps, drop rate 0000 bps
       Match: any
       Queueing
       queue limit 64 packets
       (queue depth/total drops/no-buffer drops) 0/0/0
       (pkts output/bytes output) 52/11592
       shape (average) cir 3000000, bc 12000, be 12000
       target shape rate 3000000

       Service-policy : Europe

         queue stats for all priority classes:
           Queueing
           queue limit 512 packets
           (queue depth/total drops/no-buffer drops) 0/0/0
           (pkts output/bytes output) 0/0

         Class-map: Europe_voice (match-all)
           0 packets, 0 bytes
```

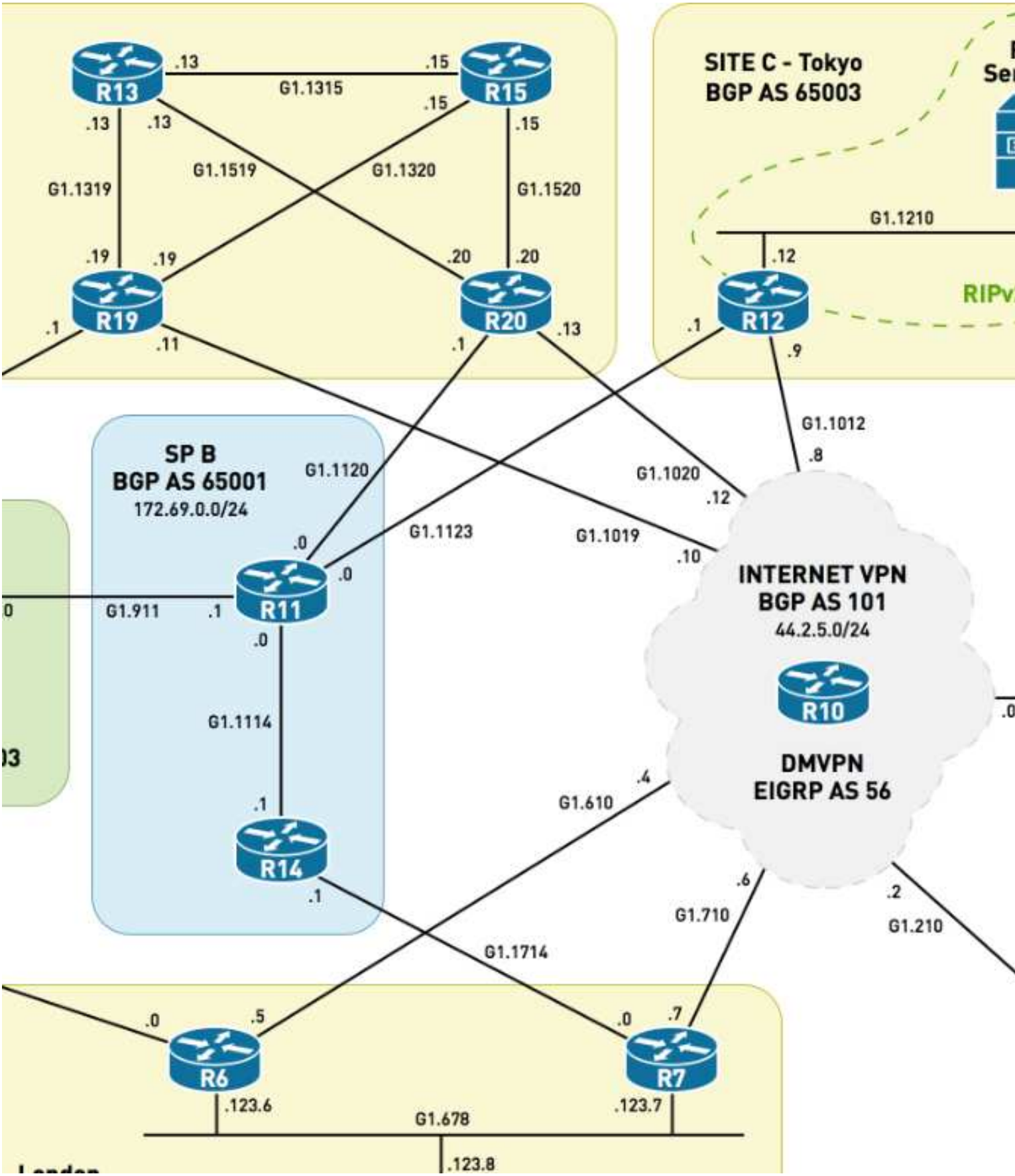
https://t.me/learningnets

```
    5 minute offered rate 0000 bps, drop rate 0000 bps
    Match: access-group 100
    Priority: 1000 kbps, burst bytes 25000, b/w exceed drops: 0


  Class-map: Europe_Routing (match-all)
    15 packets, 1320 bytes
    5 minute offered rate 0000 bps, drop rate 0000 bps
    Match: ip precedence 6
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 15/2550
    bandwidth 20% (600 kbps)

    <output omitted>
```

Score: 2 Points

Ticket 2

https://t.me/learningnets



https://t.me/learningnets

R7 is not installing routes 180.10.153.0/26 via the Internet VPN. Fix the network to match the output below:

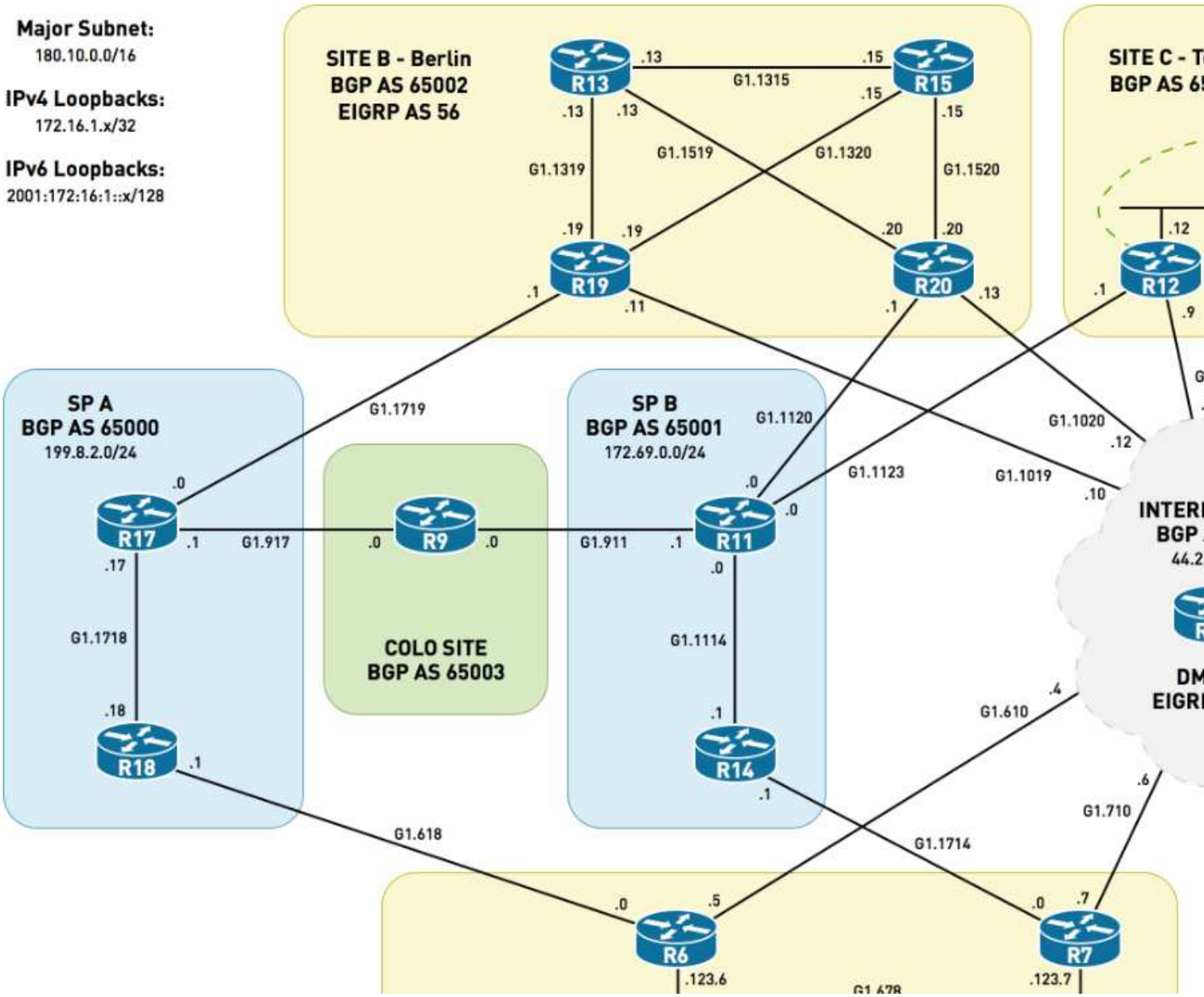
```
R7#show ip cef 180.10.153.0 detail
180.10.153.0/26, epoch 2, per-destination sharing
  nexthop 180.10.254.19 Tunnel0
  nexthop 180.10.254.19 Tunnel0
```

More than two paths may appear in this output depending on the order in which the tickets are solved. At least two paths must show in the output to earn points for this ticket. Having more than two paths in the output will not negatively effect the resolution of this ticket.

Score: 2 Points

Ticket 3

https://t.me/learningnets



R9 is a router strategically placed by the global network in a "Colo" location connected to both SP A and SP B. The role of R9 is to provide redundancy to single connected sites in the global network. If R7's link to SP B fails, traffic from London to Tokyo should not transit Berlin - it should transit the Colo instead.

R7 takes the direct path through SP B to reach Server2 at Tokyo, 180.10.200.100, before a failure.

```
R7#traceroute 180.10.200.100

Type escape sequence to abort.
Tracing the route to 180.10.200.100
VRF info: (vrf in name/id, vrf out name/id)
  1 180.10.147.1 4 msec 2 msec 1 msec
  2 180.10.121.0 [AS 65003] [MPLS: Label 33 Exp 0] 2 msec 1 msec 2 msec
  3 180.10.121.1 [AS 65003] 1 msec 2 msec 1 msec
```

https://t.me/learningnets

```
  4 180.10.120.100 [AS 65003] 2 msec *  3 msec
```

Simulate a failure on R7 by shutting down its link to SP B, GigabitEthernet1.714. R7 should begin transiting the Colo site to reach the Tokyo Server2 after this failure. Fix the network so that you can match the output below where R7 transits the Colo to reach the Tokyo server during a failure. Re-enable the interface after testing.

```
R7:
interface GigabitEthernet1.714
 shutdown

R7#traceroute 180.10.200.100
Type escape sequence to abort.
Tracing the route to 180.10.200.100
VRF info: (vrf in name/id, vrf out name/id)
  1 180.10.123.6 3 msec 1 msec 1 msec
  2 180.10.186.1 2 msec 1 msec 1 msec  3 180.10.179.1
 [MPLS: Label 28 Exp 0] 4 msec 2 msec 2 msec  4 180.10.179.0
2 msec 2 msec 1 msec  5 180.10.119.1
 2 msec 2 msec 2 msec
  6 180.10.121.1 2 msec 3 msec 2 msec
  7 180.10.120.100 5 msec *  3 msec
```

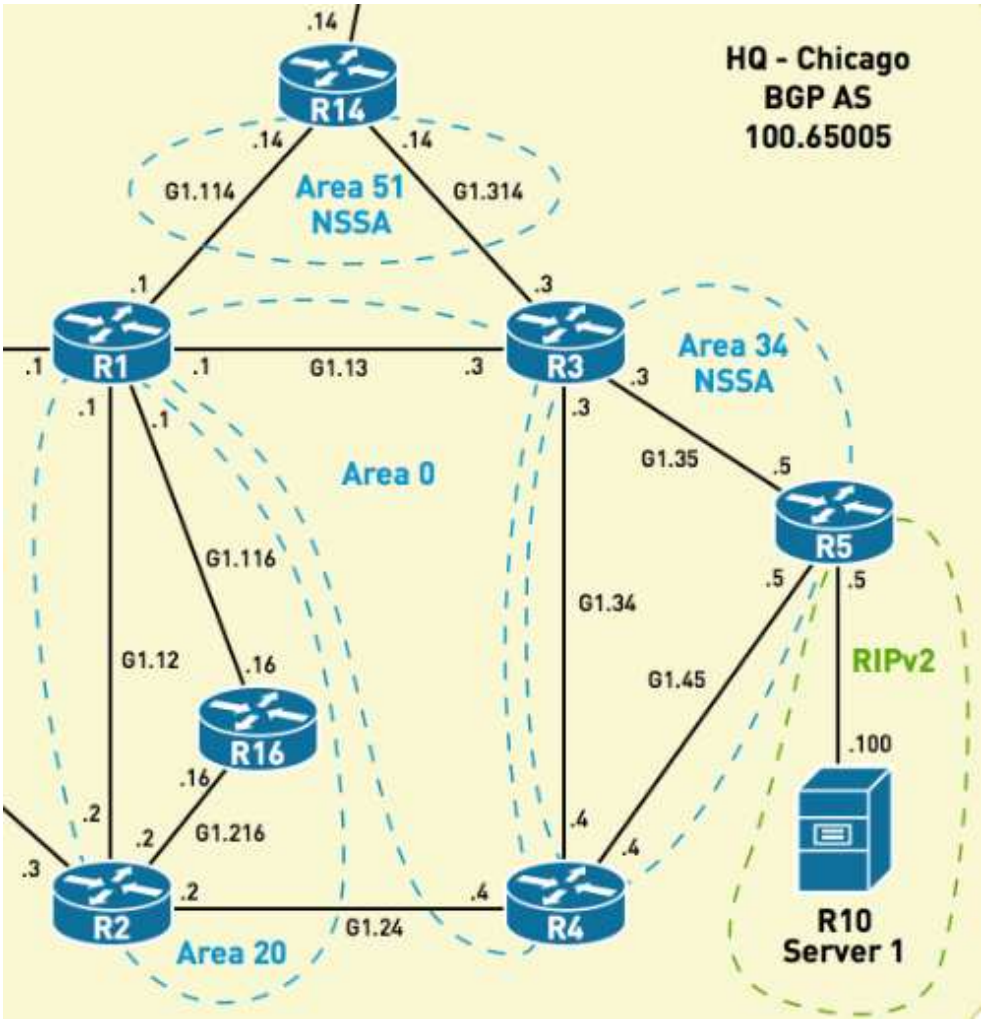
DONT FORGET!

```
R7:
interface GigabitEthernet1.714
 no shutdown
```

Score: 3 Points

Ticket 4

https://t.me/learningnets

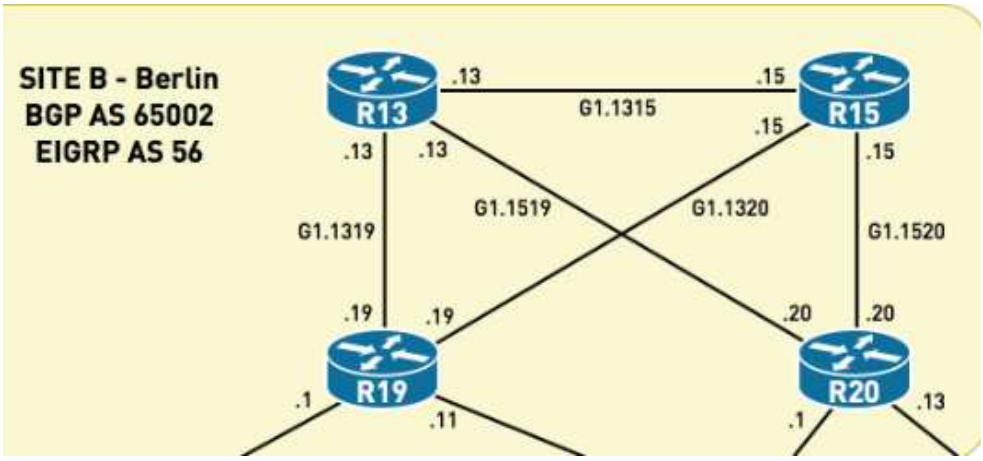


R2 is unable to reach Server1, 180.10.100.100. Fix the network to match the output below:

```
R2#ping 180.10.100.100

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 180.10.100.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

Score: 2 Points

Ticket 5



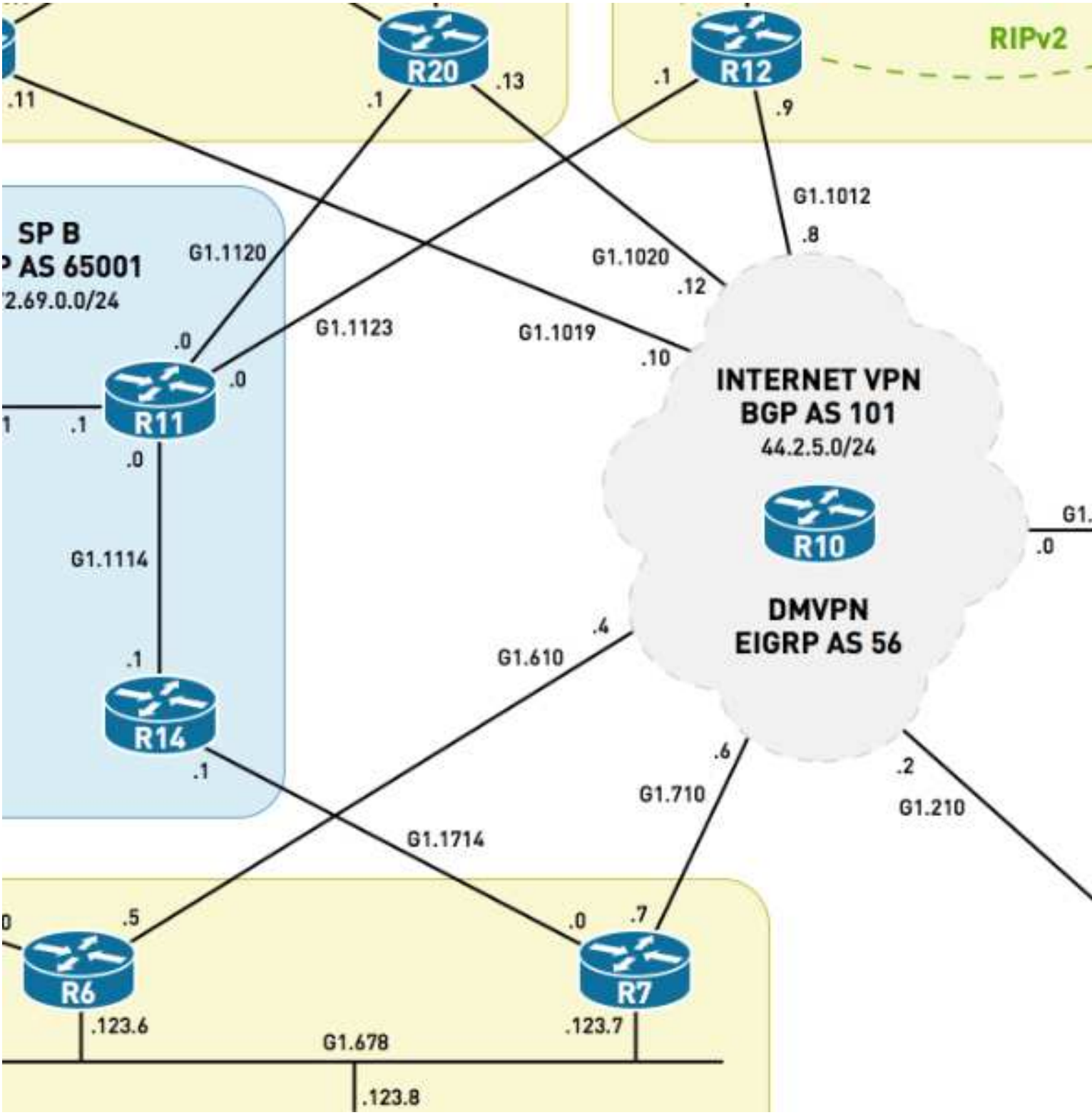
R19 must take the following path to reach R15's Loopback1. Fix the network to match the output below:

```
R19#traceroute 192.168.100.15

Type escape sequence to abort.
Tracing the route to 192.168.100.15
VRF info: (vrf in name/id, vrf out name/id)
  1 180.10.139.13 2 msec 1 msec 0 msec
  2 180.10.153.15 2 msec *  2 msec
```

Score: 2 Points

Ticket 6

https://t.me/learningnets



R6 is unable to reach R12 via the Internet VPN DMVPN Network. Fix the network to match the output below:

```
R6#traceroute 180.10.254.12

Type escape sequence to abort.
Tracing the route to 180.10.254.12
VRF info: (vrf in name/id, vrf out name/id)
```

https://t.me/learningnets

```
 1 180.10.254.12 4 msec
  *  2 msec
```

Restrictions:

- Do not remove any interface-level commands on any device.

Hints:

A traceroute from R1/R2 or R12 to R6's physical address should yield the following output:

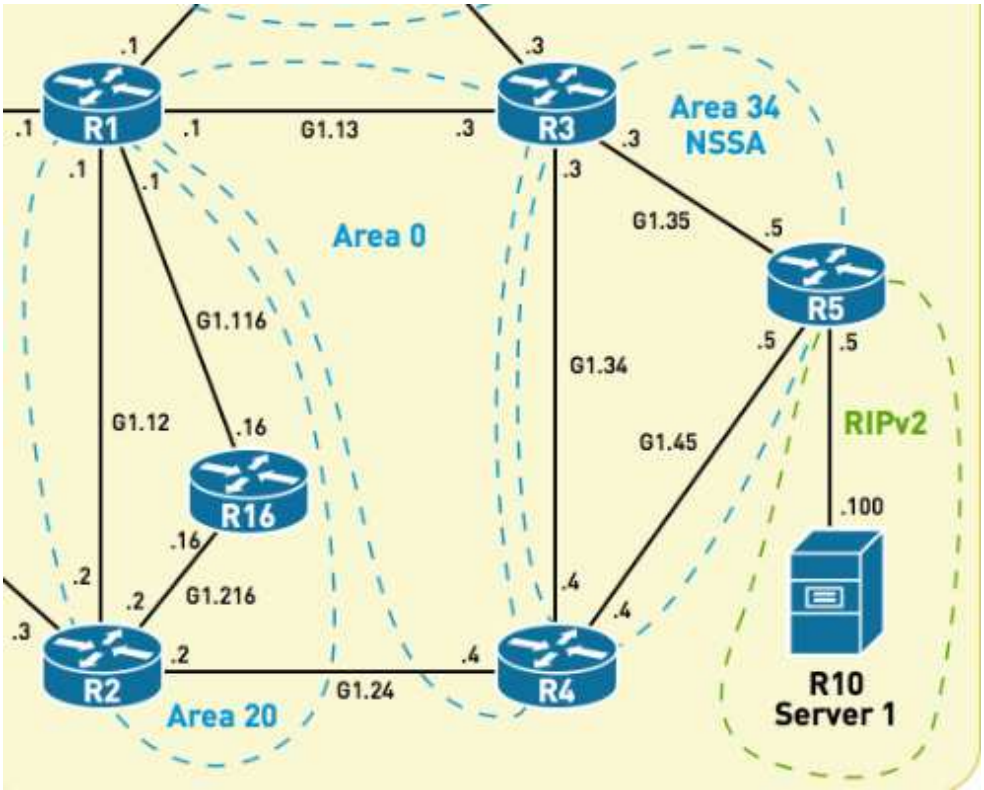
```
R2#traceroute  44.2.5.5

Type escape sequence to abort.
Tracing the route to 44.2.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 44.2.5.2 [AS 101] 3 msec 2 msec 1 msec
  2 44.6.6.6 [AS 101] 1 msec *  1 msec
```

Note the difference in the last addresses from the trace output as compared to the NBMA address.

Score: 3 Points

Ticket 7

https://t.me/learningnets



R5 is unable to ping R1's Loopback2 interface, 2001:192:168:1::1. Fix the network to match the output below:

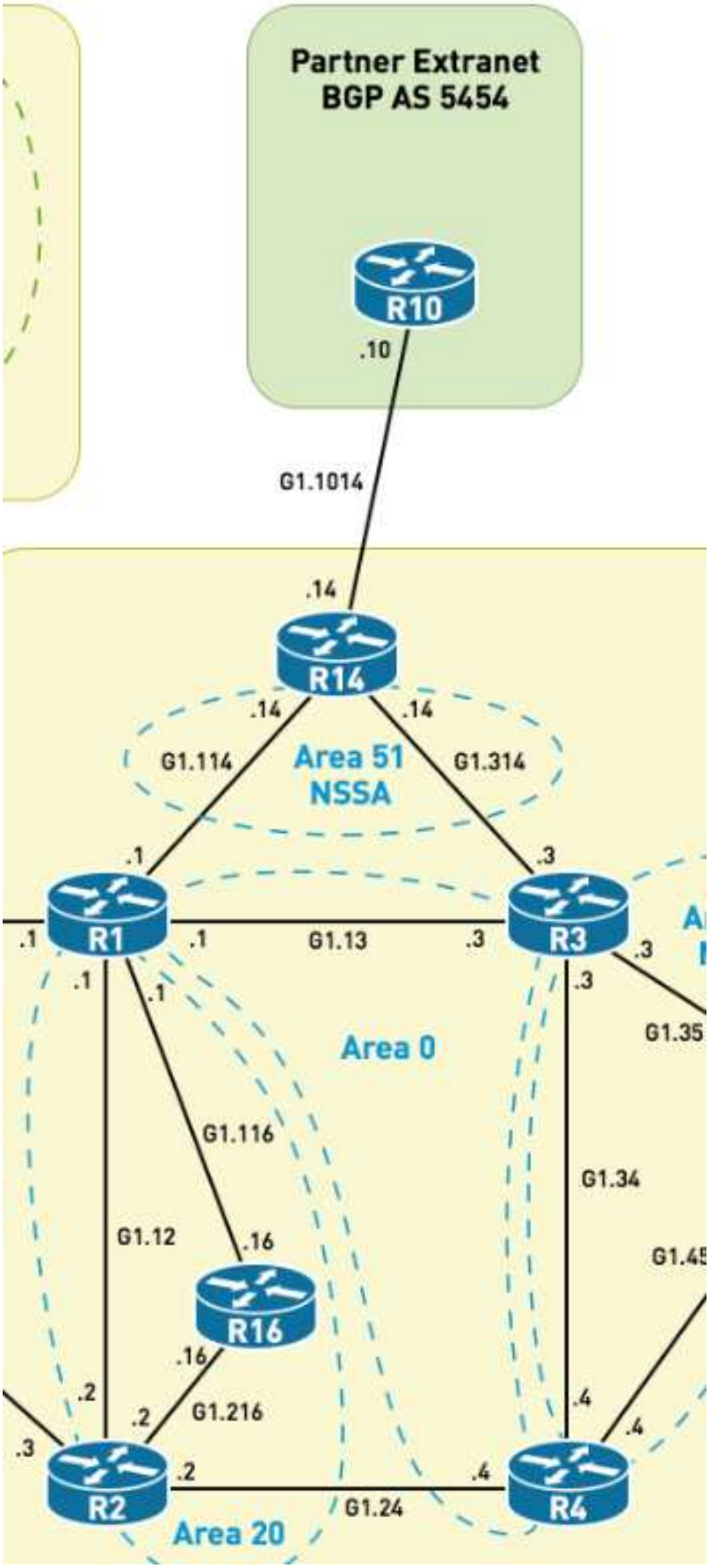
IPv6 IGP routing protocols are congruent with the IPv4 IGPs. BGP is used at HQ to advertise the IPv6 Loopback2 networks of each router.

```
R5#ping  2001:192:168:1::1 source 2001:192:168:1::5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:192:168:1::1, timeout is 2 seconds:
Packet sent with a source address of 2001:192:168:1::5!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

Score: 2 Points

Ticket 8

https://t.me/learningnets



HQ in Chicago connects to a Partner Extranet Network. R14 is connected to this network and is in charge of exchanging routes with them via BGP. Users at HQ are

https://t.me/learningnets

complaining that they cannot reach some of the partner extranet services that are on the 30.0.0.0/8 networks advertised into HQ by R14. Fix the network so that the following outputs are matched from R4.

```
R4#show ip route 30.0.0.0

Routing entry for 30.0.0.0/16, 4 known subnets
O E2     30.0.0.0 [110/1] via 180.10.34.3, 00:01:19, GigabitEthernet1.34
O E2     30.1.0.0 [110/1] via 180.10.34.3, 00:01:19, GigabitEthernet1.34
O E2     30.2.0.0 [110/1] via 180.10.34.3, 00:01:19, GigabitEthernet1.34
O E2     30.3.0.0 [110/1] via 180.10.34.3, 00:01:19, GigabitEthernet1.34

R4#ping 30.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.0.0.1, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms

R4#traceroute 30.0.0.1

Type escape sequence to abort.
Tracing the route to 30.0.0.1
VRF info: (vrf in name/id, vrf out name/id)
  1 180.10.34.3 3 msec 1 msec 1 msec
  2 180.10.143.14 1 msec 2 msec 0 msec
  3 54.1.254.10 3 msec *  3 msec
```

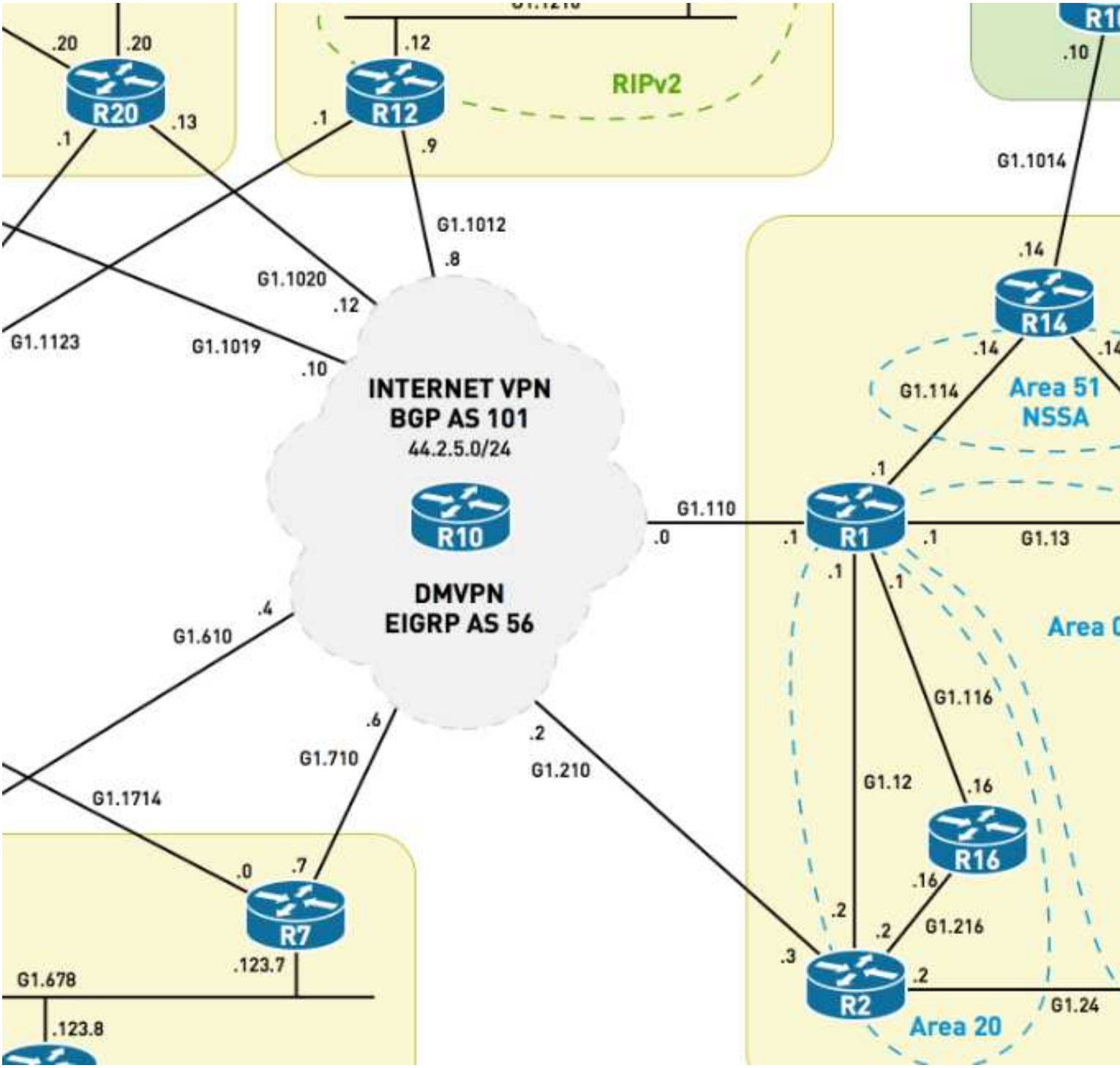
Hints:

- R14 has its interfaces facing HQ and the Partner Extranet in a VRF.

Score: 2 Points

Ticket 9

https://t.me/learningnets



R20 in Berlin is unable to reach R4's IPv6 loopback. Fix the network to match the output below:

IPv6 IGP routing protocols are congruent with the IPv4 IGPs.

```
R20#traceroute
 Protocol [ip]: ipv6
Target IPv6 address: 2001:172:16:1::4
```

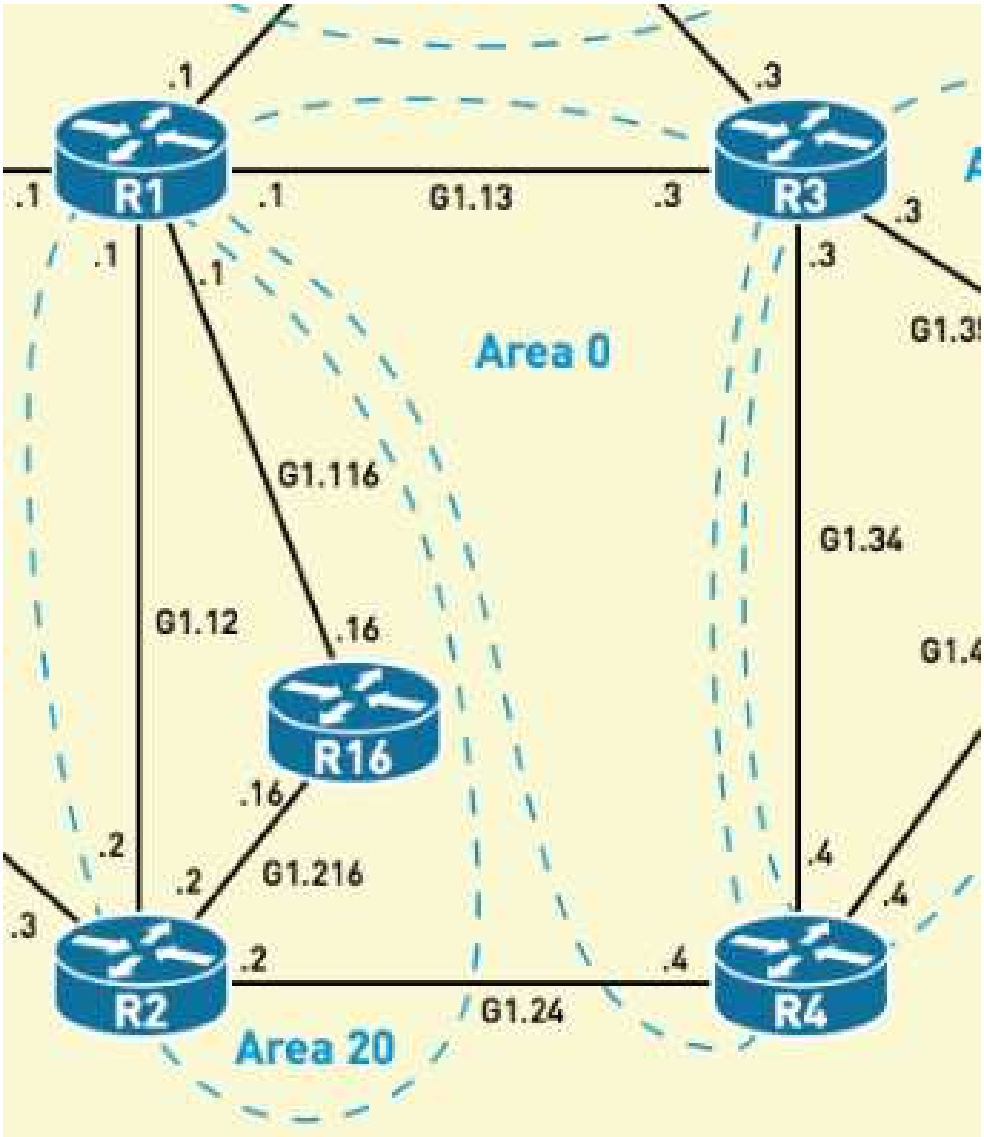
https://t.me/learningnets

```
Source address: 2001:172:16:1::20

Insert source routing header? [no]:
Numeric display? [no]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Priority [0]:
Port Number [0]:
Type escape sequence to abort.
Tracing the route to 2001:172:16:1::4

  1 2001:180:10:254::1 [AS 65000] 2 msec 2 msec 11 msec
  2 2001:180:10:13::3 [AS 65000] 3 msec 1 msec 7 msec
  3 2001:180:10:34::4 [AS 65000] 8 msec 2 msec 2 msec
```

Score: 3 Points

Ticket 10

https://t.me/learningnets



R1 is unable to receive multicast traffic for group 224.100.1.1 from R3. Fix the network to match the output below:

```
R3#ping
Protocol [ip]: Target IP address: 224.100.1.1
Repeat count [1]: 10

Datagram size [100]:
Timeout in seconds [2]: Extended commands [n]: y
Interface [All]: Loopback0
Time to live [255]: Source address or interface: 172.16.1.3
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
```

https://t.me/learningnets

```
Sending 10, 100-byte ICMP Echos to 224.100.1.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.1.3
.
Reply to request 1 from 180.10.116.1, 4 ms Reply to request 1 from 172.16.1.1, 4 ms
Reply to request 2 from 172.16.1.1, 4 ms

Reply to request 3 from 172.16.1.1, 2 ms
Reply to request 4 from 172.16.1.1, 3 ms
Reply to request 5 from 172.16.1.1, 2 ms
```

Restrictions:

- Do not add static mroutes or statically set the RP on any device.

Score: 2 Points

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Full-Scale Labs

Full-Scale Lab 1 Tasks

Diagrams and initial configs for this lab are located in the Resources section in the upper-right portion of this page.

1. IGP Core Routing
2. IGP Site Routing
3. MPLS
4. DMVPN
5. Multicast
6. IPv6
7. Optimizations
8. Security

Difficulty Rating (10 highest): 7

Lab Overview

The following scenario is a practice lab exam designed to test your skills at configuring Cisco networking devices. Specifically, this scenario is designed to assist you in your preparation for Cisco Systems' CCIE Routing & Switching Lab Exam. However, remember that in addition to being designed as a simulation of the actual CCIE lab exam, this practice lab should be used as a learning tool. Instead of rushing through the lab to complete all the configuration steps, take the time to research the networking technology in question and gain a deeper understanding of the principles behind its operation.

Lab Instructions

https://t.me/learningnets

Before starting, ensure that the initial configuration scripts for this lab have been applied. For a current copy of these scripts, see the INE Members site at http://members.INE.com. If you have any questions related to the scenario solutions, visit our CCIE support forum at http://IEOC.com.

Refer to the attached diagrams for interface and protocol assignments. Any reference to *X* in an IP address refers to your router number.

Upon completion, all devices should have full IP reachability to networks in the routing domain as specified by each task.

Lab Do's and Don'ts

- Do not change or add any IP addresses from the initial configuration unless otherwise specified or required for troubleshooting.
- If additional IP addresses are needed but not specifically permitted by the task, use IP unnumbered.
- Do not change any interface encapsulations unless otherwise specified.
- Do not change the console, AUX, and VTY passwords or access methods unless otherwise specified.
- Do not use any static routes, default routes, default networks, or policy routing unless otherwise specified.
- Save your configurations often.

Grading

This practice lab consists of various sections totaling 71 points. A score of 57 points is required to pass the exam. A section must work 100% with the requirements given to be awarded the points for that section. No partial credit is awarded. If a section has multiple possible solutions, choose the solution that best meets the requirements.

Point Values

The point values for each section are as follows:

Section	Point Value
IGP Core Routing	18

https://t.me/learningnets

Section	Point Value
IGP Site Routing	5
MPLS	13
DMVPN	9
Multicast	6
IPv6	9
Optimizations	7
Security	4

GOOD LUCK!

---

1. IGP Core Routing

1.1 OSPF - Core Routing

- Configure OSPF in the core network between R1, R2, R3, R4, and R16 as follows:
  - Configure the Loopback0 interface as the router-id and use process ID 100. All possible OSPF configurations should be done under the interface, not the routing process.
  - Configure the link between R3 and R4 in Area 0.
  - Configure the link between R1 and R2 in Area 10.
  - Configure the link between R1 and R3 in Area 20.
  - Configure the link between R2 and R4 in Area 30.
  - Configure the links between R1 and R16, as well as R2 and R16 in Area 121.

https://t.me/learningnets

Points: 3

1.2 OSPF - Core Routing

- Configure Area 121 to block Type-5 LSAs from entering the area, but allow R16 to receive Type-3 LSAs from other areas. Don't use routing filters to accomplish this.
- Ensure that default-routes generated by OSPF are only advertised into Area 121.
- Configure R1's link to R17 in Area 117 and ensure that no neighbors are formed on this link.
- The link between R2 and R3, as well as the link between R1 and R4, should be redistributed into OSPF as E1 with a metric of 1000.
- Advertise the Loopback0 interface of R1, R2, R3, and R4 into Area 0.
- Redistribute the Loopback0 interface of R16 into OSPF.

Points: 3

1.3 OSPF - Core Routing

- R2 and R4 should not send OSPF multicast packets to each other and should use a hello and dead interval that is the default value of a LAN interface.
- Configure R1 and R3 to generate /32 host routes for their link, and use a hello and dead interval that is the default value of a LAN interface.

Points: 2

1.4 RIP - Core Routing

- Configure RIP between R3, R4, R5, and R6 as follows:
  - Ensure that RIP does not send updates on interfaces that are part of other routing domains.
  - Ensure that only RIPv2 updates are sent between all peers.
  - R4 and R6 should receive RIP updates from each other using the link's broadcast address.
  - Advertise the Loopback0 interface of R5 into RIP.
  - Redistribute the Loopback0 interface of R6 into RIP with a metric of 6.

https://t.me/learningnets

Points: 2

1.5 EIGRP - Core Routing

- Configure an EIGRP instance with Autonomous System 56 between R1, R5, R7, and R8 as follows:
  - Ensure that EIGRP can accurately account for links greater than 1 Gbps.
  - Modify the EIGRP composite metric calculation so that only delay is considered when DUAL is computed within the EIGRP domain.
  - R7 and R8 should never install any routes with a hop count greater than 10. Do not use any manual filters such as route-maps to accomplish this.
  - Use 256-bit SHA authentication with the password **CC!E_!nE** to authenticate all of the routers in the EIGRP domain. Use the smallest number of configuration lines to accomplish this task.
  - Advertise the Loopback0 Interfaces of R7 and R8 into EIGRP.

Points: 3

1.6 Redistribution - Core Routing

- Perform mutual redistribution between RIP and OSPF on R4 and R3.
- Perform mutual redistribution between EIGRP and OSPF on R1 and between EIGRP and RIP on R5.
- Ensure full and stable reachability between all the routers in the OSPF, RIP, and EIGRP domains - R1-R8, and R16.

Points 5

---

2. Site Routing

2.1 EIGRP - Site Routing

- Configure EIGRP AS 789 between R10, R11, and R12 as follows:
  - Advertise the Loopback0 networks of R10, R11, and R12 into EIGRP.
  - Modify the EIGRP computed metric scaling factor, for routes entered into the RIB, to 12 less than the default on R10.

https://t.me/learningnets

- R10, R11, and R12 must be able to detect a link failure in the LAN segment to which they are connected to in less than 500 msec.

Points: 3

2.2 OSPF - Site Routing

- Configure OSPF Area 5 between R11, R12, R13, R14, and R15 as follows:
  - Use process-id 5 and set the router-id to each router's Loopback0 address.
  - Advertise the Loopback0 interfaces of R13, R14, and R15 into Area 5.
  - Configure the network so that R13 can reach R10's loopback0 via R11, and so that R14 can reach it via R12. Don't modify the link cost/metrics to accomplish this task.
  - Ensure reachability between the Loopback0 networks of all devices in Site A.

Points: 2

---

3. MPLS

3.1 LDP

- Configure LDP on all core routers, R1-R8 and R16, as follows:
  - Use the fewest configuration commands in the OSPF domain (on R1-R4) to enable LDP on all interfaces running OSPF.
  - R3 and R4 should only advertise labels for the Loopback0 networks of R2, R6, R7, and R8.
  - R6 must require password authentication for all of its LDP peers. Use the password **CC!E_!nE** for all of R6's LDP peers.

Points: 3

3.2 VRF Provisioning

R9 and R10 are CE routers and are members of VPN_CCIE. R9 is multihomed to two PE routers, R7 and R8, whereas R10 has a single uplink toward its PE, R2.

-

https://t.me/learningnets

Create VRFs on PE routers R2, R7, and R8 to provision this VPN_CCIE as follows:
  - R2:
    - RD 65066:200
    - VRF Name: VPN_CCIE
  - R7:
    - RD 65066:700
    - VRF Name: VPN_CCIE
  - R8:
    - RD 65006:800
    - VRF Name: VPN_CCIE
- Configure a route-target policy that allows Site A and Site B to freely exchange routes. R9 should be able to receive Site A routes via both of its PE/CE links.
- The VRFs configured should have native support for IPv6.

Points: 3

3.3 VPNv4 BGP

- Configure BGP AS 65006 in the Core network as follows:
  - R4 is the VPNv4 Route Reflector.
  - PE routers R2, R7, and R8 are the VPNv4 clients of R4.
  - Ensure that no other address family is active between these peers.
  - To comply with the security policy, configure a BGP MD5 password of **CC!E_!nE** on all peerings.
  - R4 should use a single peer-group for all iBGP VPNv4 peerings.
  - Use the Loopback0 interface on these devices to establish the peerings.

Points: 3

3.4 PE/CE Routing

- Configure EBGP PE/CE routing between R9 and both of its PEs as follows:
  - Use BGP AS 65100 on R9.
  - Establish the EBGP session using the PE/CE link as the update-source.
  - Use network statements on R9 to advertise both PE/CE links and R9's loopback0 network into BGP.
- Configure EBGP PE/CE routing between R10 and R2 as follows:
  - Use BGP AS 65100 on R10.

https://t.me/learningnets

- Ensure that R9 has full and stable reachability to the Loopback0 networks of all routers behind R10 when sourcing traffic from R9's Loopback0 interface.

Points: 4

---

4. DMVPN

R6 and R18 have been preconfigured to peer via IPv4 BGP with the local Internet Service Provider. The ISP is providing transit services for R18 to connect to the rest of the core network via the Internet.

The following is an overview of the desired state of the DMVPN Network. Follow the design below while working on the DMVPN sections.

- *R18 – Hub*
- *R16 – Spoke*
- *R17 – Spoke*
- *Provide secure connectivity between the servers at each site.*

4.1 DMVPN Underlay Connectivity

- Configure the network so that DMVPN routers R16, R17, and R18 have connectivity between their underlay (or "NBMA") interfaces as follows:
  - Ensure that only necessary routes are leaked between the Core network and the Internet.
  - You may use a single static default (0.0.0.0/0) route to achieve underlay connectivity, but do not redistribute this route.

Points: 2

4.2 DMVPN Overlay Connectivity

- Configure the DMVPN Network between R16, R17, and R18 following the design outlined at the beginning of the DMVPN section, as follows:
  - Use Interface Tunnel 100 on R16, R17, and R18 to establish DMVPN connectivity.
  - Use network 172.100.123.X/24 on the tunnel. Set the fourth octet to match the router name (that is, R18 should use 172.100.123.18/24).

https://t.me/learningnets

- Set the Network-ID to 100, and the NHRP Authentication to “NHRPKEY”.
- Use the following Phase 1 parameters:
  - AES 192 Encryption
  - SHA 256 Hash
  - Pre-Shared Key Authentication
    - Use key “DmvPn!23”
    - No wildcard keys
  - DH Group 5
- Use the following Phase 2 parameters:
  - Phase 2 must support encryption and authentication.
  - Use SHA for hashing.
- Ensure that the DMVPN network does not add unnecessary overhead in the data plane.
- Set the IP MTU of the Tunnel interface on all DMVPN routers to 1400 bytes.
- Configure the network so that Maximum Segment Size of TCP traffic sent from servers behind Hub/Spokes is adjusted appropriately based on the IP MTU of 1400 as it transits the DMVPN network.

Points: 3

4.3 DMVPN Routing

The servers on this network are VRFs configured on R15 named ‘server1’, ‘server2’, and ‘server3’, accordingly. To run show commands from each server, log in to R15 and change to the corresponding VRF context by using the following command: **R15#routing-context vrf server<1-3>** The prompt will change to the VRF context selected, such as **R15%server2#**. From this prompt, all show/ping/traceroute commands are executed from the VRF instead of from the global routing table.

- Configure EIGRP AS 123 in the DMVPN sites as follows:
  - Advertise the Loopback0 Networks of R17, R18, R19, and R20 into EIGRP.
  - Server 2 and Server 3 have been pre-configured to run EIGRP on their Ethernet links toward R17 and R16, respectively. Ensure that R16 and R17 establish an EIGRP adjacency with the server on their site.
  - Server 1 has been pre-configured with a default-gateway of 172.27.192.254. Configure the network so that R20 responds to this IP address as long as it

https://t.me/learningnets

has reachability to R18's Loopback0 network. R19 should begin responding to the IP address as soon as R20 loses reachability to R18's Loopback0. Ensure that R20 is able to begin responding to the IP address if reachability to R18's Loopback0 is restored. Note that the Interface status on R20's sub-interface is not a valid indicator of reachability for this scenario.
    - The protocol used to accomplish this task must use md5 with authentication key “SERVER_VIP”.
  - Ensure that R19 and R20 don’t become EIGRP neighbors.
  - Configure the DMVPN Network so that it is possible to configure summarization from the Hub toward the Spokes in the future, yet allow for direct spoke-to-spoke communications.
  - By the end of this section, all servers must be able to communicate with each other. Additionally, R16 and R17 should be able to send traffic to each other without going through the hub.

Points: 4

---

5. Multicast

5.1 PIM - Core

- Configure PIM in the Core network as follows:
  - Enable PIM on LAN interfaces on the Core routers.
  - To gain redundancy in the PIM network, configure R3 and R4 as a single logical RP.
    - Create a new Loopback (Loopback100) on R3 and R4 with an IP address of 192.122.3.100/32 to accomplish this task.
    - Advertise this new Loopback into the network.
    - You may introduce an additional TCP-based protocol, if needed, between R3 and R4.
  - Configure R8 to disseminate RP information throughout the PIM network using a protocol that is part of the PIMv2 Standard.
  - Ensure that no messages advertising RP information are leaked to the Internet. Use a single command to accomplish this task.

https://t.me/learningnets

Points: 4

5.2 Multicast Data Plane

- Configure R6 to join group 226.10.6.6 on its Loopback0 interface.
- Ensure that R6 can receive multicast traffic sourced by R17.
  - Test the multicast data-plane using ICMP.

Points: 2

---

6. IPv6

6.1 IPv6 IGPs - OSPFv3 and EIGRPv6

Configure IPv6 IGPs in Site A following the IPv4 IGP structure - EIGRP between R10-R12, and OSPFv3 between R11-R15.

- Configure EIGRPv6 in Site A between R10, R11, and R12 as follows:
  - Use the same autonomous system and routing instance used for IPv4.
  - Follow the same advertisements for IPv6 as configured in IPv4.
- Configure OSPFv3 in Site A between R11, R12, R13, R14, and R15 as follows:
  - Use the version of OSPFv3 that allows both IPv4 and IPv6 address families to be advertised in the same database.
  - Follow the same advertisements for IPv6 as configured in IPv4.
  - Use process ID 5 and advertise all OSPFv3 interfaces into Area 5.
- Ensure reachability of all Loopback0 IPv6 networks between the two routing domains.

Points: 3

6.2 IPv6 IGPs - RIPng

- Configure RIPng instance “CCIE” on core routers R3, R4, R5, and R6 as follows:
  - Advertise the Loopback0 IPv6 networks of these routers into RIPng.
  - Ensure reachability of the Loopback0 IPv6 networks between all of the RIPng routers.

https://t.me/learningnets

Points: 2

6.3 MP-BGP - IPv6

The ISP at AS 65123 is now offering IPv6 Internet connectivity and has been pre-configured to peer with R6 via IPv6 BGP.

- Configure the network using BGP AS 65006 as follows:
  - Configure R6 to peer with the ISP over IPv6. The ISP's edge router is using an IPv6 address of 2001:202:4:60::1.
  - The ISP is expecting the EBGP peering to come from AS 65600. Ensure that AS 65006 does not appear on routes sent to or received from the ISP.
  - Configure R6 as an IPv6 BGP Route Reflector and R3, R4, and R5 as the Route Reflector Clients.
  - Use the IPv6 Loopback0 addresses for the iBGP peerings.
  - Advertise the Loopback0 networks of routers R3, R4, R5, and R6 into BGP.
  - Ensure that the iBGP peers of R6 can reach the IPv6 routes received from the ISP – test with ICMP.
  - R6 should not advertise the ISP-facing IPv6 link (2001:202:4:60::/127) into the iBGP/RIPng domain.

Points: 4

---

7. Optimizations

7.1 Fast Reroute

- Configure the OSPF domain between R1-R4 and R16 so that all external routes that have a second best-path get pre-installed in the FIB.

Points: 2

7.2 Routing Policy

- Configure R7 and R8 with the following EBGP routing policy toward R9:
  - Set a local preference of 110 to routes sent to the PE routers with

https://t.me/learningnets

community value 65100:110.
    - Set a local preference of 90 to routes sent to the PE routers with community value 65100:90.
    - Set a community value of 'no-advertise' to routes sent to the PE routers with community value 65100:999 to stop further propagation (black hole the route).

Points: 3

7.3 Query Boundaries

- Configure the EIGRP DMVPN network as follows:
    - Ensure that R18 does not query spoke routers R16 and R17 when a route goes active during DUAL calculation.
    - Ensure that Site E and Site F can redistribute routes into EIGRP in the future without any modifications to the EIGRP settings.

Points: 2

---

8. Security

8.1 Remote Shell

- Configure SSH on R7 and R8 as follows:
    - Ensure that SSHv2 is the only remote shell protocol allowed on the VTY lines of R7 and R8; don't use an ACL to accomplish this task.
    - R7 and R8 should only accept SSHv2 connections on port 4022.
        - Ensure that this filter is active for all routing tables - traffic coming in on interfaces in VRFs must also be subject to these restrictions.
    - Account for any packet filters that may be introduced in the network at a later time that would only allow SSH traffic when sourced from the Loopback0 networks of R7 and R8.
    - Configure RSA key modulus size that is 1024 larger than the minimum required to run SSHv2.
    - Configure the routers so that local user accounts can log in via SSH.
    - Use a domain-name of **ine.ccie.lab**.

https://t.me/learningnets

Points: 2

8.2 Control Plane Security

- Configure the PIM RP(s) on the Core network to police the inbound PIM control-plane traffic to 256 Kbps.
- Protect the BGP control plane on PE router R2 by limiting the number of routes that R10 can send into the VPN to 1000.
- Generate a warning message if R10 sends 800 routes, and restart the peer every 5 minutes if the maximum is reached.

Points: 2

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Full-Scale Labs

Full-Scale Lab 1 Solutions

Task 1.1

This task is asking for basic OSPF configuration. This configuration should be done under the interface instead of the routing process, as stated in the task. We can accomplish this by using "ip ospf area". Some process-specific configuration can only be configured under the routing process, but assigning an interface to an area via the interface command instead of the "network" command is more intuitive.

Task 1.1 Solution

```
R1:
router ospf 100
 router-id 192.122.3.1
!
interface GigabitEthernet1.12
 ip ospf 100 area 10
!
interface GigabitEthernet1.13
 ip ospf 100 area 20
!
interface GigabitEthernet1.116
 ip ospf 100 area 121

R2:
router ospf 100
 router-id 192.122.3.2
!
interface GigabitEthernet1.12
 ip ospf 100 area 10
!
interface GigabitEthernet1.24
 ip ospf 100 area 30
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.216
 ip ospf 100 area 121

R3:
router ospf 100
 router-id 192.122.3.3
!
interface GigabitEthernet1.34
 ip ospf 100 area 0
!
interface GigabitEthernet1.13
 ip ospf 100 area 20

R4:
router ospf 100
 router-id 192.122.3.4
!
interface GigabitEthernet1.34
 ip ospf 100 area 0
!
interface GigabitEthernet1.24
 ip ospf 100 area 30

R16:
router ospf 100
 router-id 192.122.3.16
!
interface GigabitEthernet1.116
 ip ospf 100 area 121
!
interface GigabitEthernet1.216
 ip ospf 100 area 121
```

Task 1.1 Verification

To verify, look at the OSPF adjacencies formed between the routers.

https://t.me/learningnets

```
R1#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
192.122.3.2       1   FULL/DR         00:00:34    10.0.12.2       GigabitEthernet1.12
192.122.3.3       1   FULL/DR         00:00:38    10.0.13.2       GigabitEthernet1.13
192.122.3.16      1   FULL/DR         00:00:36    89.211.116.16   GigabitEthernet1.116
```

Ensure that the interfaces are assigned to the correct area.

```
R1#show ip ospf interface brief


Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C Gi1.12 100    10
            10.0.12.1/30       1     BDR   1/1 Gi1.13 100    20
            10.0.13.1/30       1     BDR   1/1 Gi1.116 100    121
            89.211.116.1/25    1     BDR   1/1
```

Pitfall

Area 10 and 121 are unable to receive any routing information from the other areas because of the current OSPF area design. At a minimum, two virtual-links must be added between R3 and R1 and between R4 and R2. The virtual links will be configured on the following task, where we are asked to advertise the Loopback0 interface of R1-R4 into Area 0.

Is it critical to read through the lab guide before starting the tasks to catch pitfalls and dependencies like this one. A quick Layer 3 diagram outlining the IGP design can help you find these issues early.

Task 1.2

This task has multiple goals. First, Area 121 must not allow Type-5 LSA, but R16 still needs to receive Type-3 LSAs. We can accomplish this without using manual filtering by using the "stubinness" of OSPF areas. The core network is used as transit for VPNs later in the lab, so full reachability is required in later tasks. R16 must still be able to find a way out of the area, and it must perform redistribution of its Loopback0. A perfect fit for all of these requirements is Area type NSSA. It will allow us to filter Type-5 but allow Type-3, and R16 will be able to redistribute into the area.

A default route must be advertised to R16 via one or both of its ABRs. This default route will ensure that R16 has reachability for all external routes. The task also

https://t.me/learningnets

specifies that any default routes generated by OSPF must only be advertised into Area 121 (another hint for using NSSA in Area 121); this can be accomplished by clearing the "P Bit" on the default route. There are multiple ways to do this, but the one best suited for this task is use of the `nssa-only` keyword.

The virtual-links configured here fix the disjoint OSPF Area design used by this lab. Area 10 and 121 will be able to exchange Type-3 LSAs, and also allow R1-R4 to advertise their Loopback0 into Area 0.

Route-maps are used for matching on the interface to be redistributed and setting the required metric-type / metric to grant flexibility during redistribution in later tasks. Having these route-maps in place ensures that our redistribution is as explicit as possible, and it gives us more granularity later when redistribution is done between OSPF, RIPv2, and EIGRP.

Task 1.2 Solution

```
R1:
router ospf 100
 area 20 virtual-link 192.122.3.3
 passive-interface GigabitEthernet1.117
 area 121 nssa default-information-originate
 redistribute connected subnets route-map CONNECTED_INTO_OSPF
!
interface GigabitEthernet1.117
 ip ospf 100 area 117
!
interface Loopback0
 ip ospf 100 area 0
!
route-map CONNECTED_INTO_OSPF permit 10
 match interface GigabitEthernet1.14
 set metric 1000
 set metric-type type-1

R2:
router ospf 100
 area 30 virtual-link 192.122.3.4
 area 121 nssa default-information-originate
 redistribute connected subnets route-map CONNECTED_INTO_OSPF
!
interface Loopback0
 ip ospf 100 area 0
!
```

https://t.me/learningnets

```
route-map CONNECTED_INTO_OSPF permit 10
 match interface GigabitEthernet1.23
 set metric 1000
 set metric-type type-1

R3:
router ospf 100
 area 20 virtual-link 192.122.3.1
 redistribute connected subnets route-map CONNECTED_INTO_OSPF
!
interface Loopback0
 ip ospf 100 area 0
!
route-map CONNECTED_INTO_OSPF permit 10
 match interface GigabitEthernet1.23
 set metric 1000
 set metric-type type-1

R4:
router ospf 100
 area 30 virtual-link 192.122.3.2
 redistribute connected subnets route-map CONNECTED_INTO_OSPF
!
interface Loopback0
 ip ospf 100 area 0
!
route-map CONNECTED_INTO_OSPF permit 10
 match interface GigabitEthernet1.14
 set metric 1000
 set metric-type type-1

R16:
router ospf 100
 area 121 nssa
 redistribute connected subnets route-map CONNECTED_INTO_OSPF
!
route-map CONNECTED_INTO_OSPF permit 10
 match interface Loopback0
```

Task 1.2 Verification

We can start by verifying in Area 121:

R16 has Area 121 properly configured as an NSSA and has two interfaces in the

https://t.me/learningnets

area.

```
R16#show ip ospf | section Area 121
 Area 121
        Number of interfaces in this area is 2 It is a NSSA area
```

Both R1 and R2 are adjacent with R16, so we can deduce that R1 and R2 are properly configured for NSSA in Area 121.

```
R16#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface 192.122.3.2
       1   FULL/BDR        00:00:31    202.4.216.2     GigabitEthernet1.216 192.122.3.1
       1   FULL/BDR        00:00:38    89.211.116.1    GigabitEthernet1.116
```

Redistribution of R16's Loopback0 into OSPF is also working as expected. We can look at the other areas to ensure they are also receiving this route, but by running a ping script later, the same thing can be accomplished.

https://t.me/learningnets

```
R16#show ip ospf database nssa-external 192.122.3.16


            OSPF Router with ID (192.122.3.16) (Process ID 100)

                Type-7 AS External Link States (Area 121)

  LS age: 2027   Options: (No TOS-capability, Type 7/5 translation
, DC, Upward)
  LS Type: AS External Link Link State ID: 192.122.3.16
 (External Network Number )
  Advertising Router: 192.122.3.16
  LS Seq Number: 80000001
  Checksum: 0xC47D
  Length: 36
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 202.4.216.16

        External Route Tag: 0
```

A healthy-looking routing table on R16 is displayed below. Note that the default-route is present, verifying that R1 and R2 injected the default into the NSSA. Both of the R2-R4 and R2-R3 links are also properly redistributed as E1 and with a metric of 1000. The four loopback0 networks of R1-R4 are also shown, further verifying the injection of the Loopback0 interfaces into Area 0, and the working state of the virtual-links. R16 has served as an excellent point of verification so far.

```
R16#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 202.4.216.2 to network 0.0.0.0

O*N2  0.0.0.0/0 [110/1] via 202.4.216.2, 00:35:12, GigabitEthernet1.216
                [110/1] via 89.211.116.1, 00:35:17, GigabitEthernet1.116
```

https://t.me/learningnets

```
      10.0.0.0/30 is subnetted, 6 subnets
O IA     10.0.12.0 [110/2] via 202.4.216.2, 00:35:12, GigabitEthernet1.216
                   [110/2] via 89.211.116.1, 00:35:12, GigabitEthernet1.116
O IA     10.0.13.0 [110/2] via 89.211.116.1, 00:35:12, GigabitEthernet1.116
O N1     10.0.14.0 [110/1001] via 89.211.116.1, 00:37:38, GigabitEthernet1.116
O N1     10.0.23.0 [110/1001] via 202.4.216.2, 00:40:48, GigabitEthernet1.216
O IA     10.0.24.0 [110/2] via 202.4.216.2, 00:35:12, GigabitEthernet1.216
O IA     10.0.34.0 [110/3] via 202.4.216.2, 00:33:32, GigabitEthernet1.216
                   [110/3] via 89.211.116.1, 00:32:29, GigabitEthernet1.116
      89.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O IA     89.211.117.0/25
           [110/2] via 89.211.116.1, 00:35:12, GigabitEthernet1.116
      192.122.3.0/32 is subnetted, 5 subnets
O IA     192.122.3.1 [110/2] via 89.211.116.1, 00:35:12, GigabitEthernet1.116
O IA     192.122.3.2 [110/2] via 202.4.216.2, 00:35:12, GigabitEthernet1.216
O IA     192.122.3.3 [110/3] via 89.211.116.1, 00:32:29, GigabitEthernet1.116
O IA     192.122.3.4 [110/3] via 202.4.216.2, 00:33:32, GigabitEthernet1.216
```

We must also verify that the default route has the "P Bit" cleared.

```
R16#show ip ospf database nssa-external 0.0.0.0

            OSPF Router with ID (192.122.3.16) (Process ID 100)

                Type-7 AS External Link States (Area 121)

  LS age: 229  Options: (No TOS-capability, No Type 7/5 translation
, DC, Upward)
  LS Type: AS External Link
  Link State ID: 0.0.0.0 (External Network Number )
  Advertising Router: 192.122.3.1
  LS Seq Number: 80000002
  Checksum: 0x551C
  Length: 36
  Network Mask: /0
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 1
        Forward Address: 0.0.0.0
        External Route Tag: 0
<output snipped>
```

This ensures that this Type-7 LSA will not be eligible to be translated into Type-5 if

https://t.me/learningnets

received any another ABR. However, in our network, R1 and R2 are the only NSSA ABRs. We we must verify that both R1 and R2 are not also advertising the default directly into other areas. The output below shows that the only defaults in the database are the Type-7 LSAs advertised by R1 and R2, verifying this task. Note that when an ABR injects a default route into OSPF via the 'area nssa default-information-originate' command, this Type-7 will have the "P-Bit' cleared and will not be leaked to other areas outside of the NSSA. The ABRs also do not require a default route in the RIB in order to originate the 0.0.0.0 Type-7 LSA.

```
R1#show ip ospf database | include Type-7|Type-5|Summary|0.0.0.0
                Summary Net Link States (Area 0)
                Summary ASB Link States (Area 0)
                Summary Net Link States (Area 10)
                Summary ASB Link States (Area 10)
                Summary Net Link States (Area 20)
                Summary ASB Link States (Area 20)
                Summary Net Link States (Area 117)
                Summary ASB Link States (Area 117)
                Summary Net Link States (Area 121) Type-7 AS External Link States (Area 121)
0.0.0.0         192.122.3.1     1775        0x80000043 0x00D25D 0
0.0.0.0         192.122.3.2     1927        0x80000043 0x00CC62 0

                Type-5 AS External Link States
R1#
```

Task 1.3

Changing the OSPF network type on the links to either 'non-broadcast' or 'point-to-multipoint non-broadcast' will disable the default multicast nature of OSPF (224.0.0.5 and 224.0.0.6). We will change the link between R2 and R4 to 'non-broadcast', and we will add manual neighbor statements to maintain the adjacency.

There are two network types in OSPF that generate /32 host routes to "fix" network designs where the Layer-2 network does not match the Layer-3 network—for example, a hub and spoke Layer-2 network (DMVPN, Frame-Relay) where all end points share the same subnet. According to Layer-3, all of the devices in such network are in the same broadcast domain and should be able to send packets directly to each other. However, the Layer-2 network indicates otherwise: devices cannot directly communicate between each other and must pass through the hub first (DLCIs from spokes to hub only - partial mesh). To overcome this challenge, OSPF network type point-to-multipoint generates /32 host routes for each of the end

https://t.me/learningnets

points. Additionally, the hub changes the next-hop of the routes received from spokes to itself when advertising them back out to other spokes.

Task 1.3 Solution

```
R1:
interface GigabitEthernet1.13
 ip ospf network point-to-multipoint
 ip ospf hello-interval 10

R2:
router ospf 100
 neighbor 10.0.24.2
!
interface GigabitEthernet1.24
 ip ospf network non-broadcast
 ip ospf hello-interval 10

R3:
interface GigabitEthernet1.13
 ip ospf network point-to-multipoint
 ip ospf hello-interval 10

R4:
router ospf 100
 neighbor 10.0.24.1
!
interface GigabitEthernet1.24
 ip ospf network non-broadcast
 ip ospf hello-interval 10
```

Task 1.3 Verification

Start by verifying that all of our neighbors are up after this change.

```
R1#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
192.122.3.3       0   FULL/  -           -        10.0.13.2       OSPF_VL0
192.122.3.2       1   FULL/DR         00:00:32    10.0.12.2       GigabitEthernet1.12
192.122.3.3       0   FULL/  -        00:00:37    10.0.13.2       GigabitEthernet1.13
```

https://t.me/learningnets

```
192.122.3.16      1   FULL/DR         00:00:33    89.211.116.16   GigabitEthernet1.116

R2#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
192.122.3.4       0   FULL/  -           -        10.0.24.2       OSPF_VL0
192.122.3.1       1   FULL/BDR        00:00:33    10.0.12.1       GigabitEthernet1.12
192.122.3.4       1   FULL/BDR        00:00:34    10.0.24.2       GigabitEthernet1.24
192.122.3.16      1   FULL/DR         00:00:38    202.4.216.16    GigabitEthernet1.216

R3#show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address         Interface
192.122.3.1       0   FULL/  -           -        10.0.13.1       OSPF_VL0
192.122.3.4       1   FULL/BDR        00:00:30    10.0.34.2       GigabitEthernet1.34
192.122.3.1       0   FULL/  -        00:00:30    10.0.13.1       GigabitEthernet1.13

R4#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
192.122.3.2       0   FULL/  -           -        10.0.24.1       OSPF_VL0
192.122.3.3       1   FULL/DR         00:00:37    10.0.34.1       GigabitEthernet1.34
192.122.3.2       1   FULL/DR         00:00:37    10.0.24.1       GigabitEthernet1.24
```

The interfaces have been changed to the desired network types, and the hello/dead intervals are set back to default LAN settings.

```
R1#show ip ospf interface GigabitEthernet1.13
GigabitEthernet1.13 is up, line protocol is up
  Internet Address 10.0.13.1/30, Area 20, Attached via Interface Enable
  Process ID 100, Router ID 192.122.3.1, Network Type POINT_TO_MULTIPOINT
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no             Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
```

https://t.me/learningnets

```
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 8
  Last flood scan time is 0 msec, maximum is 1 msec Neighbor Count is 1, Adjacent neighbor count is 1
 Adjacent with neighbor 192.122.3.3
  Suppress hello for 0 neighbor(s)

R2#show ip ospf interface GigabitEthernet1.24
GigabitEthernet1.24 is up, line protocol is up
  Internet Address 10.0.24.1/30, Area 30, Attached via Interface Enable
  Process ID 100, Router ID 192.122.3.2, Network Type NON_BROADCAST
, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.122.3.2, Interface address 10.0.24.1
  Backup Designated router (ID) 192.122.3.4, Interface address 10.0.24.2
 Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Index 1/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 10
  Last flood scan time is 0 msec, maximum is 1 msec Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 192.122.3.4  (Backup Designated Router)

  Suppress hello for 0 neighbor(s)
```

We can further verify that OSPF is not using multicast on the interfaces we configured as "non-broadcast" by doing a quick debug.

https://t.me/learningnets

```
R2#debug ip ospf hello

OSPF hello debugging is on
R2# *Sep 20 15:05:59.703: OSPF-100 HELLO Gi1.24: Send hello to 10.0.24.2 area 30 from 10.0.24.1

R2# *Sep 20 15:06:00.826: OSPF-100 HELLO Gi1.12: Send hello to 224.0.0.5 area 10 from 10.0.12.2
```

R1 and R3 should have created /32 host routes for each other.

```
R1#show ip route 10.0.13.2
Routing entry for 10.0.13.2/32
  Known via "ospf 100", distance 110, metric 1, type intra area
  Last update from 10.0.13.2 on GigabitEthernet1.13, 00:08:25 ago
  Routing Descriptor Blocks:  * 10.0.13.2, from 192.122.3.3, 00:08:25 ago, via GigabitEthernet1.13
      Route metric is 1, traffic share count is 1
R3#show ip route 10.0.13.1
Routing entry for 10.0.13.1/32
  Known via "ospf 100", distance 110, metric 1, type intra area
  Last update from 10.0.13.1 on GigabitEthernet1.13, 00:15:12 ago
  Routing Descriptor Blocks:  * 10.0.13.1, from 192.122.3.1, 00:15:12 ago, via GigabitEthernet1.13

      Route metric is 1, traffic share count is 1
```

The point-to-multipoint link is represented in the database as a point-to-point link. The IP network of the link is advertised as a stub network, with a Link-ID containing the actual prefix and the Link-Data containing the /32 mask. Note that if this link were configured as network type point-to-point, the Link-Data of the stub network would contain the mask configured on the link, not /32.

```
R3#show ip ospf database router self-originate | begin Area 20
                Router Link States (Area 20)

  LS age: 14
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 192.122.3.3
  Advertising Router: 192.122.3.3
  LS Seq Number: 8000004E
  Checksum: 0xA13E
  Length: 48
  Area Border Router
  AS Boundary Router
```

https://t.me/learningnets

```
  Number of Links: 2
 Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 192.122.3.1
     (Link Data) Router Interface address: 10.0.13.2
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
 Link connected to: a Stub Network
     (Link ID) Network/subnet number:10.0.13.2
     (Link Data)Network Mask: 255.255.255.255

      Number of MTID metrics: 0
       TOS 0 Metrics: 0
```

Now we should have full reachability within the OSPF domain, and we should be meeting all of the requirements. Before moving to the next section dealing with another IGP, we will run a ping-script that will verify reachability within the OSPF domain. If there are any issues, it would be wise to solve them before moving to the next section. We will run the script from three points in the OSPF domain; this should be enough to indicate whether we have full reachability within the OSPF domain.

```
R16#tclsh
R16(tcl)#proc ping_script {} {
+>foreach i {
+>192.122.3.1
+>192.122.3.2
+>192.122.3.3
+>192.122.3.4
+>192.122.3.16
+>} { ping $i source loopback0 }
+>}
R16(tcl)#R16(tcl)#ping_script
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.1, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.16
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.2, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.16
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.3, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.16
```

https://t.me/learningnets

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/11 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.4, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.16
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/8/11 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.16, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.16
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
R16(tcl)#

R1#tclsh
R1(tcl)#proc ping_script {} {
+>(tcl)#foreach i {
+>(tcl)#192.122.3.1
+>(tcl)#192.122.3.2
+>(tcl)#192.122.3.3
+>(tcl)#192.122.3.4
+>(tcl)#192.122.3.16
+>(tcl)#} { ping $i source loopback0 }
+>(tcl)#}R1(tcl)#ping_script
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.1, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.2, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.3, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/9/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.4, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/18/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.16, timeout is 2 seconds:
```

https://t.me/learningnets

```
Packet sent with a source address of 192.122.3.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
R1(tcl)#

R4#tclsh
R4(tcl)#proc ping_script {} {
+>(tcl)#foreach i {
+>(tcl)#192.122.3.1
+>(tcl)#192.122.3.2
+>(tcl)#192.122.3.3
+>(tcl)#192.122.3.4
+>(tcl)#192.122.3.16
+>(tcl)#} { ping $i source loopback0 }
+>(tcl)#}R4(tcl)#ping_script

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.1, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.2, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.3, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.4, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.16, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/8/10 ms
R4(tcl)#
```

https://t.me/learningnets

Task 1.4

In this task we enable RIPv2 on R3, R4, R5, and R6. Passive-interfaces have to be used to ensure that no updates are sent on interfaces that are not part of the RIP domain. The 'network 10.0.0.0' statement will include ALL links on the routers that fall within that network range, even the links that are part of the OSPF domain. Unlike OSPF and EIGRP, RIPv2 does not support the use of wildcard masks to specify which interface to run the protocol on.

R6 has to redistribute the Loopback0 interface into RIP with a metric of 6; remember this for later redistribution tasks.

Task 1.4 Solution

```
R3:
router rip
 version 2
 passive-interface default
 no passive-interface GigabitEthernet1.35
 no passive-interface GigabitEthernet1.36
 network 10.0.0.0
 no auto-summary

R4:
router rip
 version 2
 passive-interface default
 no passive-interface GigabitEthernet1.45
 no passive-interface GigabitEthernet1.46
 network 10.0.0.0
 no auto-summary
!
interface GigabitEthernet1.46
 ip rip v2-broadcast
 ip broadcast-address 10.0.46.3

R5:
router rip
 version 2
 passive-interface default
 no passive-interface GigabitEthernet1.35
 no passive-interface GigabitEthernet1.45
```

https://t.me/learningnets

```
 no passive-interface GigabitEthernet1.56
 network 10.0.0.0
 network 192.122.3.0
 no auto-summary

R6:
router rip
 version 2
 redistribute connected route-map CONNECTED_INTO_RIP
 passive-interface default
 no passive-interface GigabitEthernet1.36
 no passive-interface GigabitEthernet1.46
 no passive-interface GigabitEthernet1.56
 network 10.0.0.0
 no auto-summary
!
interface GigabitEthernet1.46
 ip rip v2-broadcast
 ip broadcast-address 10.0.46.3
!
route-map CONNECTED_INTO_RIP permit 10
 match interface Loopback0
 set metric 6
```

Task 1.4 Verification

This is a straightforward IGP section (easy points). Paying attention to detail in simple tasks like this is important. The use of passive-interface default ensures that only those interfaces that we explicitly set as "no passive" AND are part of the RIP domain will send RIP updates. The task also requires ensuring that you run RIPv2; the simplest way to accomplish this by using the `version 2` command at the routing process level.

Verification of most of these requirements can be observed in the `show ip protocols` command. Here we see that `version 2` is the only active version and that we are only running RIPv2 on the interfaces we explicitly configured as "no passive." This output is similar in R4, R5, and R6, the only difference being the active interfaces.

```
R3#show ip protocols | begin "rip"
Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 27 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
```

https://t.me/learningnets

```
  Redistributing: rip Default version control: send version 2, receive version 2
Interface             Send  Recv
  Triggered RIP  Key-chain GigabitEthernet1.35    2      2
                                   GigabitEthernet1.36    2      2

  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    10.0.0.0 Passive Interface(s):
GigabitEthernet1
GigabitEthernet1.13
GigabitEthernet1.23
GigabitEthernet1.34
GigabitEthernet2
Loopback0

  Routing Information Sources:
    Gateway         Distance      Last Update
    Gateway         Distance      Last Update
    10.0.34.2            120      00:13:44
    10.0.35.2            120      00:00:15
    10.0.36.2            120      00:00:26
  Distance: (default is 120)
```

A quick debug on R4 and R6 verifies that RIP is using the link's broadcast address instead of 224.0.0.9. Note that the link between R4 and R6 uses 10.0.46.0/30; the broadcast address of this link is 10.0.46.3. Had we not configured the `ip broadcast-address 10.0.46.3` on both R4 and R6, RIP would fall back to using 255.255.255.255 if configured with `ip rip v2-broadcast` .

```
R4#debug ip rip
 *Sep 20 16:10:59.896: RIP: sending v2 update to 10.0.46.3 via GigabitEthernet1.46
 (10.0.46.1)
*Sep 20 16:10:59.896: RIP: build update entries
*Sep 20 16:10:59.896:   10.0.14.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:10:59.896:   10.0.24.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:10:59.896:   10.0.34.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:10:59.896:   10.0.35.0/30 via 0.0.0.0, metric 2, tag 0
*Sep 20 16:10:59.896:   10.0.45.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:10:59.896:   192.122.3.5/32 via 0.0.0.0, metric 2, tag 0
R4# *Sep 20 16:11:19.103: RIP: sending v2 update to 224.0.0.9 via GigabitEthernet1.45
 (10.0.45.1)
*Sep 20 16:11:19.103: RIP: build update entries
*Sep 20 16:11:19.103:   10.0.14.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:11:19.103:   10.0.24.0/30 via 0.0.0.0, metric 1, tag 0
```

https://t.me/learningnets

```
*Sep 20 16:11:19.103:   10.0.34.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:11:19.103:   10.0.36.0/30 via 0.0.0.0, metric 2, tag 0
*Sep 20 16:11:19.103:   10.0.46.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:11:19.103:   192.122.3.6/
```

Notice what happens when the manually configured broadcast address is removed.

```
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.46
R4(config-subif)#no ip broadcast-address 10.0.46.3
R4(config-subif)#end
R4#debug ip rip
<output snip>*Sep 20 16:16:30.968: RIP: sending v2 update to 255.255.255.255 via GigabitEthernet1.46
 (10.0.46.1)
*Sep 20 16:16:30.968: RIP: build update entries
*Sep 20 16:16:30.969:   10.0.14.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:16:30.969:   10.0.24.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:16:30.969:   10.0.34.0/30 via 0.0.0.0, metric 1, tag 0
*Sep 20 16:16:30.969:   10.0.35.0/30 via 0.0.0.0, metric 2, tag 0
*Sep 20 16:16:30.969:   10.0.45.0/30 via 0.0.0.0, metric 1, tag 0

R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R4(config)#interface GigabitEthernet1.46
    R4(config-subif)#ip broadcast-address 10.0.46.3
   R4(config-subif)#end
```

R5 and R6's loopback0 network is advertised into RIP. Note that R6's loopback has a metric of 6, which was required by the task.

```
R3#show ip route rip

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set
```

https://t.me/learningnets

```
      10.0.0.0/8 is variably subnetted, 17 subnets, 2 masks
R        10.0.45.0/30 [120/1] via 10.0.35.2, 00:00:00, GigabitEthernet1.35
R        10.0.46.0/30 [120/1] via 10.0.36.2, 00:00:09, GigabitEthernet1.36
R        10.0.56.0/30 [120/1] via 10.0.36.2, 00:00:09, GigabitEthernet1.36
                      [120/1] via 10.0.35.2, 00:00:00, GigabitEthernet1.35
      192.122.3.0/32 is subnetted, 7 subnets
R        192.122.3.5 [120/1] via 10.0.35.2, 00:00:00, GigabitEthernet1.35
R        192.122.3.6 [120/6] via 10.0.36.2, 00:00:09, GigabitEthernet1.36
```

Task 1.5

EIGRP has native support for "Wide Metrics" when configured in Named Mode/Multi-AF Mode. Prior to wide metrics, EIGRP reported the delay of 1-Gbps links and greater (10 Gbps, 20 Gbps port-channels, etc.) as 10 microseconds. This limitation leads to suboptimal routing in networks with links greater than 1 Gbps (most data center and aggregation networks today). Wide metrics modifies the metric calculation by reporting the delay, now referred to as "latency," in picoseconds instead of microseconds. Another important change is that the computed metric is now stored as a 64-bit value, compared to the legacy metric, which only scaled up to 32 bits. The larger 64-bit metric can be observed in the EIGRP topology table in the Feasible Distance and Reported Distance fields. The Cisco IOS Routing Table only supports metric values up to 32 bits, so to cope with this RIB limitation, EIGRP uses a 'rib-scale' factor that defaults to 128. When installing EIGRP routes into the RIB, the 64-bit computed metric is divided by the rib-scale factor to "fit" the metric into the RIB.

Another requirement in this task states that R7 and R8 should never install routes with a hop count greater than 10. EIGRP keeps track of the "hop-count" each time a route is passed from peer to peer. One of the great features of EIGRP is the amount of extra metadata that each route can carry. Hop-count, external protocol, tag, BGP AS number, and Originating router are some of the extra fields that EIGRP keeps for each route. This gives network admins great flexibility for matching routes when doing tasks such as redistribution or route filtering. In this particular task, we can configure EIGRP at the routing process level to not accept routes that have passed through more than 10 routers and thus have a hop count greater than 10.

Note that the requirements in this lab did not specify what the EIGRP instance name should be configured as. However, other labs may ask for a specific name.

https://t.me/learningnets

Task 1.5 Solution

```
R1:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 56
  !
  af-interface default
   authentication mode hmac-sha-256 CC!E_!nE
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 136.5.17.0 0.0.0.255
  network 136.5.18.0 0.0.0.255
  metric weights 0 0 0 1 0 0 0
 exit-address-family

R5:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 56
  !
  af-interface default
   authentication mode hmac-sha-256 CC!E_!nE
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 136.6.58.0 0.0.0.255
  metric weights 0 0 0 1 0 0 0
 exit-address-family

R7:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 56
  !
  af-interface default
   authentication mode hmac-sha-256 CC!E_!nE
  exit-af-interface
  !
  topology base
```

https://t.me/learningnets

```
   metric maximum-hops 10
  exit-af-topology
  network 136.5.17.0 0.0.0.255
  network 172.30.78.0 0.0.0.255
  network 192.122.3.7 0.0.0.0
  metric weights 0 0 0 1 0 0 0
 exit-address-family

R8:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 56
  !
  af-interface default
   authentication mode hmac-sha-256 CC!E_!nE
  exit-af-interface
  !
  topology base
   metric maximum-hops 10
  exit-af-topology
  network 136.6.58.0 0.0.0.255
  network 136.5.18.0 0.0.0.255
  network 172.30.78.0 0.0.0.255
  network 192.122.3.8 0.0.0.0
  metric weights 0 0 0 1 0 0 0
 exit-address-family
```

Task 1.5 Verification

The `show ip protocols` command yields relevant output for the verification of this task. We can verify that delay, or latency, is the only vector being considered for the EIGRP metric calculation, K3. The "rib-scale" mentioned previously, as well as the 64-bit metric, show that we are using Wide Metrics and can properly account for links greater than 1 Gbps. The maximum hop count variable is set to 10 as required for R7 and R8. We can also quickly verify that the networks we included in EIGRP via the network statements are present in this output.

```
R7#sh ip protocols | begin "eigrp 56"
Routing Protocol is "eigrp 56"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 VR(INE_CCIE) Address-Family Protocol for AS(56)
```

https://t.me/learningnets

```
Metric weight K1=0, K2=0, K3=1, K4=0, K5=0 K6=0
 Metric rib-scale 128
Metric version 64bit
    Soft SIA disabled
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 192.122.3.7
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4 Maximum hopcount 10
      Maximum metric variance 1
      Total Prefix Count: 5
      Total Redist Count: 0

  Automatic Summarization: disabled
  Maximum path: 4 Routing for Networks:
136.5.17.0/24
172.30.78.0/24
192.122.3.7/32
  Routing Information Sources:
    Gateway         Distance      Last Update
    136.5.17.1            90      02:05:04
    172.30.78.8           90      02:05:04
  Distance: internal 90 external 170

R7#sho ip eigrp neighbors

EIGRP-IPv4 VR(INE_CCIE) Address-Family Neighbors for AS(56)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   136.5.17.1              Gi1.17                   11 02:16:30    1   100  0  6
0   172.30.78.8             Gi1.78                   12 02:18:03    1   100  0  13
```

R8 has very similar output, with the exception of the networks that are included in the protocol.

```
R8#sh ip protocols | begin "eigrp 56"
Routing Protocol is "eigrp 56"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
```

https://t.me/learningnets

```
  EIGRP-IPv4 VR(INE_CCIE) Address-Family Protocol for AS(56)
Metric weight K1=0, K2=0, K3=1, K4=0, K5=0 K6=0
Metric rib-scale 128
Metric version 64bit
    Soft SIA disabled
    NSF-aware route hold timer is 240
  EIGRP NSF disabled
     NSF signal timer is 20s
     NSF converge timer is 120s
    Router-ID: 192.122.3.8
    Topology : 0 (base)
      Active Timer: 3 min
      Distance: internal 90 external 170
      Maximum path: 4 Maximum hopcount 10
      Maximum metric variance 1
      Total Prefix Count: 5
      Total Redist Count: 0

  Automatic Summarization: disabled
  Maximum path: 4 Routing for Networks:
136.5.18.0/24
136.6.58.0/24
172.30.78.0/24
192.122.3.8/32
  Routing Information Sources:
    Gateway         Distance      Last Update
    136.5.18.1            90      02:13:00
    136.6.58.5            90      02:13:00
    172.30.78.7           90      02:13:00
  Distance: internal 90 external 170

R8#show ip eigrp neighbors

EIGRP-IPv4 VR(INE_CCIE) Address-Family Neighbors for AS(56)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
2   136.5.18.1              Gi1.18                   13 02:17:33   64   384  0  15
0   172.30.78.7             Gi1.78                   13 02:17:33    1   100  0  15
0   172.30.78.7             Gi1.78                   13 02:17:33    1   100  0  15
```

To verify that our authentication is properly applied on the interfaces, we can look at the detailed EIGRP interface output.

```
R7#show ip eigrp interfaces detail
```

https://t.me/learningnets

```
EIGRP-IPv4 VR(INE_CCIE) Address-Family Interfaces for AS(56)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes Gi1.78
                   1        0/0       0/0           1       0/0           50           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 7/0
  Hello's sent/expedited: 1867/4
  Un/reliable mcasts: 0/7  Un/reliable ucasts: 9/6
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 3  Out-of-sequence rcvd: 1
  Topology-ids on interface - 0  Authentication mode is HMAC-SHA-256
, key-chain is not set Gi1.17
                   1        0/0       0/0           1       0/0           50           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 4/0
  Hello's sent/expedited: 1821/2
  Un/reliable mcasts: 0/4  Un/reliable ucasts: 5/2
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 1  Out-of-sequence rcvd: 1
  Topology-ids on interface - 0  Authentication mode is HMAC-SHA-256
, key-chain is not set
Lo0                        0        0/0       0/0           0       0/0            0           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 0/0
  Hello's sent/expedited: 0/1
  Un/reliable mcasts: 0/0  Un/reliable ucasts: 0/0
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is HMAC-SHA-256, key-chain is not set
```

R1 and R5 should have a route via EIGRP for the Loopback0 of both R7 and R8, as well as all other links participating in EIGRP that R5 and R1 are not directly connected to.

```
R5#show ip route eigrp

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
```

https://t.me/learningnets

```
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      136.5.0.0/24 is subnetted, 2 subnets
D        136.5.17.0 [90/15360] via 136.6.58.8, 00:03:22, GigabitEthernet1.58
D        136.5.18.0 [90/10240] via 136.6.58.8, 00:03:24, GigabitEthernet1.58
      172.30.0.0/27 is subnetted, 1 subnets
D        172.30.78.0 [90/10240] via 136.6.58.8, 2d04h, GigabitEthernet1.58
      192.122.3.0/32 is subnetted, 9 subnets
D        192.122.3.7 [90/10880] via 136.6.58.8, 2d04h, GigabitEthernet1.58
D        192.122.3.8 [90/5760] via 136.6.58.8, 2d04h, GigabitEthernet1.58
```

Task 1.6

So far we have configured three separate IGP zones that make up the core network in this lab's topology. However, there is no reachability between the zones because of the lack of redistribution between the protocols. There are multiple points where the routing protocols meet, and the task calls for performing mutual redistribution in all of these meeting spots.

Tip

Before starting the redistribution section, you should draw a Layer-3 IGP diagram showing where all of the protocols meet. Then, following the task requirements, mark all of the places where redistribution is required. You should be able to see problem areas such as loops or conflicting configuration from earlier tasks. If you realize that this task could take a significant portion of the time you have left, either because there are far too many issues in the redistribution scenario or because you cannot devise a quick plan of action, the next best thing to do is to get full reachability without satisfying the redistribution requirements. For example, if the task asks for mutual redistribution between two IGP zones on two separate routers (for example, RIP and OSPF on R3 and R4), you can instead just do mutual redistribution on one of the routers. Doing this will ensure that you have full reachability without the possibility of a loop or other issues. The downside is

https://t.me/learningnets

that this will not satisfy the requirements of the task and points will be lost for the redistribution section; however, it allowed you to obtain full reachability. In the case of our network, if full reachability is not established in the core, many other future tasks will simply not work (multicast, MPLS, DMVPN, BGP, etc.). Keep this in mind as your backout plan. It is far better to lose points in the redistribution section and be able to move on and complete the rest of the lab than to spend a large portion of time on redistribution and not be able to finish the lab. This could can mean the difference between a pass or fail. To summarize:

- Draw a quick IGP diagram with all desired points of redistribution.
- Identify problem areas such as loops or previous conflicting configurations.
- Devise a plan of action and identify tools to prevent issues.
- If the plan fails or you are running out of time, fall back to the "backout plan."
- The backout plan should consist of only performing redistribution where it is absolutely necessary to obtain full reachability; all points from the redistribution section will be lost, but you will be able to continue the lab with a solid core network.

We will focus on redistribution between RIP and OSPF first. What problems can you see? Here is an example of potential issues: R3 and R4 both know how to reach R6's Loopback0 (192.122.3.6/32) via RIP. If R3 redistributes this route into OSPF, R4 will receive the OSPF advertisement and will by default prefer the newly learned OSPF route over the RIP route (AD of 110 vs. 120). Nothing bad has happened until now. At this point, R4 has to route into the OSPF domain to reach a route in the RIP domain; this is suboptimal routing, and it allows for the possibility of a routing loop. Now R4 takes this OSPF route and redistributes it BACK into RIP with a metric of 1 (not good!). Recall that R6's Loopback was redistributed into RIP with a metric of 6 in an earlier task. We have just introduced a packet forwarding routing loop. R5 will receive this new advertisement from R4 and will prefer it over the direct advertisement from R6 (metric of 1 vs. 6). R5 will pass the route to R3, which previously had the original R6 route with a metric of 6. R3 will now prefer the R5 advertisement over the R6 advertisement. What happens when R2 tries to send traffic destined to 192.122.3.6? It will look something like this: R2-R3-R5-R4-R3-R5-R4-R3-R5-R4-R3-R5...

Notice that this was a simple example where redistribution was only done from RIP to OSPF on R3 (one way), and from OSPF into RIP on R4 (one way). The situation can get worse quickly if we don't properly control the redistribution.

https://t.me/learningnets

Although there are no clear steps and tools that you can use every time when doing redistribution, there are a few guidelines to consider that will help most of the time:

- NEVER redistribute a route back to its protocol of origin. In the previous example, the loop formed because we redistributed a RIP route BACK into RIP. Although this will not cause a loop every time, its best to avoid it to prevent potential issues.
- Use metadata in the routes for matching; this is much simpler than having to match routes by prefix. You can match metadata such as tags, metric-type, metric, origin protocol, and route source.
- Pay close attention to routes in distance vector protocols that have been introduced into the protocol via redistribution elsewhere in the network. This can be observed again in the example above. R6 redistributed the Loopback0 into RIP (a route that came from redistribution elsewhere in the network) and set a high metric. If instead of RIP we were running EIGRP, this same issue would have been observed. The route redistributed elsewhere in the network would have had an AD of 170, causing the same looping issue that we discussed in the previous example.
- Protect the protocol with the higher AD. When doing mutual redistribution between two protocols, the protocol with the higher AD is usually the one where the problem occurs. Think of the example described above where RIP and OSPF are redistributed at two meeting points (R3 and R4). Because RIP has an AD of 120 and OSPF an AD of 110, when R3 redistributed the RIP route into OSPF, R4 naturally installed the OSPF route instead of the "original" RIP route. This would also occur if we replaced RIP with EIGRP. If R6 redistributed its Loopback0 into EIGRP, it would have an AD of 170. When this External EIGRP route would get redistributed into OSPF by R3, R4 would receive it via OSPF and prefer the OSPF route over the EIGRP route (110 vs. 170).

Some of the tools that we can use to meet the task requirements of redistributing RIP and OSPF on R3 and R4 are tags and distribute lists. We will tag all routes sent into each protocol and then prevent them from being installed in the RIB when the routes are "fed back." For example, if R3 tags the RIP routes with Tag 3120 when redistributing RIP into OSPF and R4 has an inbound distribute-list blocking these tagged routes from being installed in the RIB, R4 will never be able to send these tagged routes BACK into RIP because it does not have them installed in the RIB via OSPF.

We can take a similar approach with the EIGRP and OSPF redistribution on R1 and R5. When R1 redistributes EIGRP routes into OSPF, they will be tagged with 190. These EIGRP routes do not have to be sent into the RIP domain, because R5 will be redistributing EIGRP into RIP. So when R3 or R4 receives routes from R1 with

https://t.me/learningnets

tagged with 190, they will install them in the RIB, but we will deny them from being redistributed into RIP by using a route-map in our redistribution statement. The same approach will be taken for R5: R5 will tag EIGRP routes redistributed into RIP with tag 590. R3 and R4 will receive the advertisement via RIP, but will not redistribute the routes into OSPF because we will block them with a route-map in the redistribution statement. Doing this filtering at both RIP and EIGRP domains will prevent the case in which an EIGRP route is somehow sent BACK into EIGRP.

Note that R3 and R4 will route through the OSPF domain to get to the EIGRP domain (AD of 110 vs 120).

Pitfall

A problem area that may be missed is the connected redistribution done in an earlier OSPF task. R1, R3, and R5 are redistributing a connected interface into OSPF with a metric-type of E1 and a metric of 1000. This does not pose a looping problem, but it does mean that we now have to manually add other interfaces into the route-map used in the connected redistribution. When explicit connected redistribution of specific interfaces into a protocol (in our case OSPF) is done on Cisco IOS, and another protocol (in our case EIGRP/RIP) is also being redistributed into OSPF, the connected interfaces of EIGRP/RIP must be manually accounted in the connected redistribution into OSPF. Notice that tags are also used in this solution; the connected interface really belongs to the protocol being redistributed, so to prevent issues discussed previously, these connected interfaces get the same tags as the "source" protocol.

Before staring, take a look at the RIB of each of the border devices - R1, R3, R4, and R5 - (`show ip route ospf` , `show ip router rip` , etc.). Ideally, the RIB of these devices will not change. For example, routes originally installed in the RIB via RIP should remain installed via RIP after redistribution.

Task 1.6 Solution

```
R1:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 56
  !
  topology base
   redistribute ospf 100 metric 1000000 100 255 1 1500
!
router ospf 100
 redistribute eigrp 56 subnets route-map EIGRP_INTO_OSPF_REDISTRIBUTION
!
route-map EIGRP_INTO_OSPF_REDISTRIBUTION permit 10
 set tag 190
```

https://t.me/learningnets

```
!
route-map CONNECTED_INTO_OSPF permit 20
 match interface GigabitEthernet1.17 GigabitEthernet1.18
 set tag 190

R3:
router ospf 100
 redistribute rip subnets route-map RIP_INTO_OSPF_REDISTRIBUTION
 distribute-list route-map BLOCK_RIP_ROUTES_DISTRIBUTE_LIST_IN in
!
router rip
 redistribute ospf 100 metric 1 route-map OSPF_INTO_RIP_REDISTRIBUTION
!
route-map OSPF_INTO_RIP_REDISTRIBUTION deny 10
 match tag 190
route-map OSPF_INTO_RIP_REDISTRIBUTION permit 20
 set tag 3110
!
route-map RIP_INTO_OSPF_REDISTRIBUTION deny 10
 match tag 590
route-map RIP_INTO_OSPF_REDISTRIBUTION permit 20
 set tag 3120
!
route-map BLOCK_RIP_ROUTES_DISTRIBUTE_LIST_IN deny 10
 match tag 4120
route-map BLOCK_RIP_ROUTES_DISTRIBUTE_LIST_IN permit 20
!
route-map CONNECTED_INTO_OSPF permit 20
 match interface GigabitEthernet1.35 GigabitEthernet1.36
 set tag 3120

R4:
router ospf 100
 redistribute rip subnets route-map RIP_INTO_OSPF_REDISTRIBUTION
 distribute-list route-map BLOCK_RIP_ROUTES_DISTRIBUTE_LIST_IN in
!
router rip
 redistribute ospf 100 metric 1 route-map OSPF_INTO_RIP_REDISTRIBUTION
!
route-map OSPF_INTO_RIP_REDISTRIBUTION deny 10
 match tag 190
route-map OSPF_INTO_RIP_REDISTRIBUTION permit 20
 set tag 4110
!
route-map RIP_INTO_OSPF_REDISTRIBUTION deny 10
 match tag 590
```

https://t.me/learningnets

```
route-map RIP_INTO_OSPF_REDISTRIBUTION permit 20
 set tag 4120
!
route-map BLOCK_RIP_ROUTES_DISTRIBUTE_LIST_IN deny 10
 match tag 3120
route-map BLOCK_RIP_ROUTES_DISTRIBUTE_LIST_IN permit 20
!
route-map CONNECTED_INTO_OSPF permit 20
 match interface GigabitEthernet1.45 GigabitEthernet1.46
 set tag 4120

R5:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 56
  !
  topology base
   redistribute rip metric 1000000 100 255 1 1500
  exit-af-topology
 exit-address-family
!
router rip
 redistribute eigrp 56 metric 1 route-map EIGRP_INTO_RIP_REDISTRIBUTION
!
route-map EIGRP_INTO_RIP_REDISTRIBUTION permit 10
 set tag 590
```

Task 1.6 Verification

The routing tables of all of the core devices should be stable at this point. We have our redistribution in place with proper filtering, and we have accounted for the connected interfaces in the EIGRP and RIP domains that needed to be manually added to the route-maps on R1, R3, and R4.

Let's check R3 and R4. The only E2 routes here should be the EIGRP routes along with R16's Loopback0. We are not routing into the OSPF domain to get to routes in the RIP domain.

```
R3#show ip route ospf | begin Gateway
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 17 subnets, 2 masks
O IA     10.0.12.0/30 [110/2] via 10.0.13.1, 2d00h, GigabitEthernet1.13
O        10.0.13.1/32 [110/1] via 10.0.13.1, 2d00h, GigabitEthernet1.13
O E1     10.0.14.0/30 [110/1001] via 10.0.34.2, 2d00h, GigabitEthernet1.34
```

https://t.me/learningnets

```
                      [110/1001] via 10.0.13.1, 2d00h, GigabitEthernet1.13
O IA     10.0.24.0/30 [110/2] via 10.0.34.2, 2d00h, GigabitEthernet1.34
      89.0.0.0/25 is subnetted, 2 subnets
O IA     89.211.116.0 [110/2] via 10.0.13.1, 2d00h, GigabitEthernet1.13
O IA     89.211.117.0 [110/2] via 10.0.13.1, 2d00h, GigabitEthernet1.13
      136.5.0.0/24 is subnetted, 2 subnets
O E2     136.5.17.0 [110/20] via 10.0.13.1, 00:02:40, GigabitEthernet1.13
O E2     136.5.18.0 [110/20] via 10.0.13.1, 00:02:40, GigabitEthernet1.13
      136.6.0.0/24 is subnetted, 1 subnets
O E2     136.6.58.0 [110/20] via 10.0.13.1, 2d00h, GigabitEthernet1.13
      172.30.0.0/27 is subnetted, 1 subnets
O E2     172.30.78.0 [110/20] via 10.0.13.1, 2d00h, GigabitEthernet1.13
      192.122.3.0/32 is subnetted, 9 subnets
O        192.122.3.1 [110/2] via 10.0.13.1, 2d00h, GigabitEthernet1.13
O        192.122.3.2 [110/3] via 10.0.34.2, 2d00h, GigabitEthernet1.34
O        192.122.3.4 [110/2] via 10.0.34.2, 2d00h, GigabitEthernet1.34
O E2     192.122.3.7 [110/20] via 10.0.13.1, 2d00h, GigabitEthernet1.13
O E2     192.122.3.8 [110/20] via 10.0.13.1, 2d00h, GigabitEthernet1.13
O E2     192.122.3.16 [110/20] via 10.0.34.2, 2d00h, GigabitEthernet1.34
                      [110/20] via 10.0.13.1, 2d00h, GigabitEthernet1.13
      202.4.216.0/26 is subnetted, 1 subnets
O IA     202.4.216.0 [110/3] via 10.0.34.2, 2d00h, GigabitEthernet1.34
                     [110/3] via 10.0.13.1, 2d00h, GigabitEthernet1.13

R4#show ip route ospf | begin Gateway
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 18 subnets, 2 masks
O IA     10.0.12.0/30 [110/2] via 10.0.24.1, 2d00h, GigabitEthernet1.24
O IA     10.0.13.1/32 [110/2] via 10.0.34.1, 2d00h, GigabitEthernet1.34
O IA     10.0.13.2/32 [110/1] via 10.0.34.1, 2d00h, GigabitEthernet1.34
O E1     10.0.23.0/30 [110/1001] via 10.0.34.1, 2d00h, GigabitEthernet1.34
                      [110/1001] via 10.0.24.1, 2d00h, GigabitEthernet1.24
      89.0.0.0/25 is subnetted, 2 subnets
O IA     89.211.116.0 [110/3] via 10.0.34.1, 2d00h, GigabitEthernet1.34
                      [110/3] via 10.0.24.1, 2d00h, GigabitEthernet1.24
O IA     89.211.117.0 [110/3] via 10.0.34.1, 2d00h, GigabitEthernet1.34
      136.5.0.0/24 is subnetted, 2 subnets
O E2     136.5.17.0 [110/20] via 10.0.34.1, 00:03:48, GigabitEthernet1.34
O E2     136.5.18.0 [110/20] via 10.0.34.1, 00:03:48, GigabitEthernet1.34
      136.6.0.0/24 is subnetted, 1 subnets
O E2     136.6.58.0 [110/20] via 10.0.34.1, 2d00h, GigabitEthernet1.34
      172.30.0.0/27 is subnetted, 1 subnets
O E2     172.30.78.0 [110/20] via 10.0.34.1, 2d00h, GigabitEthernet1.34
      192.122.3.0/32 is subnetted, 9 subnets
O        192.122.3.1 [110/3] via 10.0.34.1, 2d00h, GigabitEthernet1.34
```

https://t.me/learningnets

```
O        192.122.3.2 [110/2] via 10.0.24.1, 2d00h, GigabitEthernet1.24
O        192.122.3.3 [110/2] via 10.0.34.1, 2d00h, GigabitEthernet1.34
O E2     192.122.3.7 [110/20] via 10.0.34.1, 2d00h, GigabitEthernet1.34
O E2     192.122.3.8 [110/20] via 10.0.34.1, 2d00h, GigabitEthernet1.34
O E2     192.122.3.16 [110/20] via 10.0.24.1, 2d00h, GigabitEthernet1.24

      202.4.216.0/26 is subnetted, 1 subnets
O IA     202.4.216.0 [110/2] via 10.0.24.1, 2d00h, GigabitEthernet1.24
```

Now the same thing but for RIP: we should only see native RIP routes on R3 and R4. All EIGRP routes are preferred via OSPF.

```
R3#show ip route rip | begin Gateway

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 17 subnets, 2 masks
R        10.0.45.0/30 [120/1] via 10.0.35.2, 00:00:02, GigabitEthernet1.35
R        10.0.46.0/30 [120/1] via 10.0.36.2, 00:00:14, GigabitEthernet1.36
R        10.0.56.0/30 [120/1] via 10.0.36.2, 00:00:14, GigabitEthernet1.36
                      [120/1] via 10.0.35.2, 00:00:02, GigabitEthernet1.35
      192.122.3.0/32 is subnetted, 9 subnets
R        192.122.3.5 [120/1] via 10.0.35.2, 00:00:02, GigabitEthernet1.35
R        192.122.3.6 [120/6] via 10.0.36.2, 00:00:14, GigabitEthernet1.36

R4#show ip route rip | begin Gateway

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 18 subnets, 2 masks
R        10.0.13.0/30 [120/2] via 10.0.46.2, 00:00:04, GigabitEthernet1.46
                      [120/2] via 10.0.45.2, 00:00:02, GigabitEthernet1.45
R        10.0.35.0/30 [120/1] via 10.0.45.2, 00:00:02, GigabitEthernet1.45
R        10.0.36.0/30 [120/1] via 10.0.46.2, 00:00:04, GigabitEthernet1.46
R        10.0.56.0/30 [120/1] via 10.0.46.2, 00:00:04, GigabitEthernet1.46
                      [120/1] via 10.0.45.2, 00:00:02, GigabitEthernet1.45
      192.122.3.0/32 is subnetted, 9 subnets
R        192.122.3.5 [120/1] via 10.0.45.2, 00:00:02, GigabitEthernet1.45
R        192.122.3.6 [120/6] via 10.0.46.2, 00:00:04, GigabitEthernet1.46
```

https://t.me/learningnets

The EIGRP routers that meet with other protocols should only have native EIGRP routes (no externals). R1 routes via the OSPF domain to get to RIP routes, and R5 routes via the RIP domain to get to the OSPF routes. The EIGRP External AD of 170 makes both R1 and R5 prefer to route via OSPF or RIP.

```
R1#show ip route eigrp | be Gateway
Gateway of last resort is not set

      136.6.0.0/24 is subnetted, 1 subnets
D        136.6.58.0 [90/10240] via 136.5.18.8, 00:13:42, GigabitEthernet1.18
      172.30.0.0/27 is subnetted, 1 subnets
D        172.30.78.0 [90/10240] via 136.5.18.8, 00:13:42, GigabitEthernet1.18
                     [90/10240] via 136.5.17.7, 00:13:42, GigabitEthernet1.17
      192.122.3.0/32 is subnetted, 9 subnets
D        192.122.3.7 [90/5760] via 136.5.17.7, 00:13:42, GigabitEthernet1.17
D        192.122.3.8 [90/5760] via 136.5.18.8, 00:13:42, GigabitEthernet1.18

R5#show ip route eigrp | be Gateway

Gateway of last resort is not set

      136.5.0.0/24 is subnetted, 2 subnets
D        136.5.17.0 [90/15360] via 136.6.58.8, 00:14:00, GigabitEthernet1.58
D        136.5.18.0 [90/10240] via 136.6.58.8, 00:14:02, GigabitEthernet1.58
      172.30.0.0/27 is subnetted, 1 subnets
D        172.30.78.0 [90/10240] via 136.6.58.8, 2d04h, GigabitEthernet1.58
      192.122.3.0/32 is subnetted, 9 subnets
D        192.122.3.7 [90/10880] via 136.6.58.8, 2d04h, GigabitEthernet1.58
D        192.122.3.8 [90/5760] via 136.6.58.8, 2d04h, GigabitEthernet1.58
```

Now we can test full and stable reachability as required using a ping-script. Doing this will verify our redistribution. Note that the following output only shows R7, but this script should be run from different sections of the network to verify that the configuration until now is solid. For example, good points to test would be R16, R6, R3, and R4.

```
R7#tclsh
 R7(tcl)#proc ping-script {} {
+>(tcl)#foreach i {
+>(tcl)#192.122.3.1
+>(tcl)#192.122.3.2
+>(tcl)#192.122.3.3
+>(tcl)#192.122.3.4
+>(tcl)#192.122.3.5
```

https://t.me/learningnets

```
+>(tcl)#192.122.3.6
+>(tcl)#192.122.3.7
+>(tcl)#192.122.3.8
+>(tcl)#192.122.3.16
+>(tcl)#10.0.12.1
+>(tcl)#10.0.13.1
+>(tcl)#10.0.13.2
+>(tcl)#10.0.14.1
+>(tcl)#10.0.23.1
+>(tcl)#10.0.24.1
+>(tcl)#10.0.34.1
+>(tcl)#10.0.35.1
+>(tcl)#10.0.36.1
+>(tcl)#10.0.45.1
+>(tcl)#10.0.46.1
+>(tcl)#10.0.56.1
+>(tcl)#89.211.116.16
+>(tcl)#89.211.117.1
+>(tcl)#136.5.18.8
+>(tcl)#136.6.58.5
+>(tcl)#202.4.216.2
+>(tcl)#136.5.17.7
+>(tcl)#172.30.78.8
+>(tcl)#} { ping $i }
+>(tcl)#}
R7(tcl)#
R7(tcl)#
R7(tcl)#
R7(tcl)#ping-script
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/14/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 15/18/23 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 10/18/27 ms
Type escape sequence to abort.
```

https://t.me/learningnets

```
Sending 5, 100-byte ICMP Echos to 192.122.3.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/18/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.16, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/18/28 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.12.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.13.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/9/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.13.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/18/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.14.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.23.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/11/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.24.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/18/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.34.1, timeout is 2 seconds:
!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 19/19/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.35.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/18/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.36.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/18/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.45.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/18/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.46.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/18/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.56.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 89.211.116.16, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/18/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 89.211.117.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/11 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 136.5.18.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/11/17 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 136.6.58.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 202.4.216.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/11/18 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 136.5.17.7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
Type escape sequence to abort.
```

https://t.me/learningnets

```
Sending 5, 100-byte ICMP Echos to 172.30.78.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 msR7(tcl)#
```

Task 2.1

Moving away from the core network to Site A, we are asked to configure EIGRP between R10, R11, and R12 and ensure that these nodes can detect a link failure in less than 500 msecs. Additionally, the EIGRP metric scaling factor for routes entered into the RIB (rib-scaling factor from the previous EIGRP section) must be modified to 12 less than the default on R10.

Because these devices are running in a LAN segment and therefore are connected to some form on Layer 2 device between them, they will not be able to detect a link going down based on the loss of light signal/electrical pulses. If there were only two devices connected back to back via a fiber or cross over Ethernet cable, they would be able to detect the link going down almost immediately. For LAN segments, BFD is the only other option for detecting a link failing in a sub-second fashion. Tuning down the hello timers on routing protocols should be avoided.

To identify the default rib-scaling factor, we can just look at the `show ip protocols` output. Additionally, we can find this default value by looking at the EIGRP topology table.

Note that the requirement of modifying the rib-scaling factor implies that we must use Wide Metrics. This means that EIGRP must be configured in Named Mode.

Task 2.1 Solution

```
R10:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 789
  !
  af-interface GigabitEthernet1.102
   bfd
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 172.19.102.0 0.0.0.255
  network 192.122.3.10 0.0.0.0
```

https://t.me/learningnets

```
  metric rib-scale 116
 exit-address-family
!
interface GigabitEthernet1.102
 bfd interval 150 min_rx 150 multiplier 3

R11:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 789
  !
  af-interface GigabitEthernet1.102
   bfd
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 172.19.102.0 0.0.0.255
  network 192.122.3.11 0.0.0.0
 exit-address-family
!
interface GigabitEthernet1.102
 bfd interval 150 min_rx 150 multiplier 3

R12:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 789
  !
  af-interface GigabitEthernet1.102
   bfd
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 172.19.102.0 0.0.0.255
  network 192.122.3.12 0.0.0.0
 exit-address-family
!
interface GigabitEthernet1.102
 bfd interval 150 min_rx 150 multiplier 3
```

https://t.me/learningnets

Task 2.1 Verification

To verify this task, we can first look at the EIGRP/BFD neighbors.

```
R10#show ip eigrp neighbors

EIGRP-IPv4 VR(INE_CCIE) Address-Family Neighbors for AS(789)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   172.19.102.12           Gi1.102                  11 00:06:54    1   100  0  13
0   172.19.102.11           Gi1.102                  12 00:06:58    1   100  0  13

R10#show bfd neighbors



IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
172.19.102.11                        4098/4098       Up        Up        Gi1.102
172.19.102.12                        4097/4098       Up        Up        Gi1.102
```

The configured BFD interval is 150 msec with a multiplier of 3. This means that these routers will be able to detect a link failure and begin converging around it in 450 msec, which satisfies the requirement.

```
R10#show bfd neighbors ipv4 172.19.102.11 details



IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
172.19.102.11                        4098/4098Up         Up
        Gi1.102
Session state is UP and using echo function with 150 ms interval.
Session Host: Software
OurAddr: 172.19.102.10
Handle: 2
Local Diag: 0, Demand mode: 0, Poll bit: 0MinTxInt: 1000000, MinRxInt: 1000000,Multiplier: 3
Received MinRxInt: 1000000, ReceivedMultiplier: 3
Holddown (hits): 0(0), Hello (hits): 1000(532)
Rx Count: 537, Rx Interval (ms) min/max/avg: 1/1002/872 last: 6 ms ago
Tx Count: 534, Tx Interval (ms) min/max/avg: 1/1001/875 last: 782 ms ago
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: EIGRP CEF
Uptime: 00:07:47
```

https://t.me/learningnets

```
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 0
             Multiplier: 3               - Length: 24
             My Discr.: 4098             - Your Discr.: 4098
             Min tx interval: 1000000    - Min rx interval: 1000000 Min Echo interval: 150000
```

The rib-scale can be verified by looking at R11 or R12.

```
R11#show ip protocols | include rib

    Metric rib-scale 128
```

The default is 128, so we can subtract 12 and configure R10 with a rib-scale of 116. As mentioned previously, this can also be seen in the EIGRP topology.

```
R11#show ip eigrp topology 192.122.3.12/32
EIGRP-IPv4 VR(INE_CCIE) Topology Entry for AS(789)/ID(192.122.3.11) for 192.122.3.12/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 1392640, RIB is 10880
  Descriptor Blocks:
  172.19.102.12 (GigabitEthernet1.102), from 172.19.102.12, Send flag is 0x0      Composite metric is (
1392640
/163840), route is Internal
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 11250000 picoseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 192.122.3.12
```

The FD for this route is 1392640, but the metric entered into the RIB is 10880. 1392640/10880 = 128.

```
R11#show  ip route 192.122.3.12
Routing entry for 192.122.3.12/32  Known via "eigrp 789", distance 90, metric 10880
, type internal
  Redistributing via eigrp 789
  Last update from 172.19.102.12 on GigabitEthernet1.102, 00:08:30 ago
  Routing Descriptor Blocks:
```

https://t.me/learningnets

```
  * 172.19.102.12, from 172.19.102.12, 00:08:30 ago, via GigabitEthernet1.102
      Route metric is 10880, traffic share count is 1
      Total delay is 11 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

Task 2.2

Continuing with Site A, OSPF must be configured between the rest of the routers in the site. The only task that stands out is the requirement for R13 to reach R10's Loopback0 via R11, and for R14 to reach R10's Loopback0 via R12. We cannot change link costs or metrics to accomplish this. This requirement calls for doing redistribution between EIGRP and OSPF on both R11 and R12.

Task 2.2 Solution

```
R11:
router ospf 5
 router-id 192.122.3.11
 redistribute eigrp 789 subnets
!
interface GigabitEthernet1.113
 ip ospf 5 area 5
!
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 789
  !
  topology base
   redistribute ospf 5 metric 1000000 100 255 1 1500
  exit-af-topology
 exit-address-family

R12:
router ospf 5
 router-id 192.122.3.12
 redistribute eigrp 789 subnets
!
interface GigabitEthernet1.124
 ip ospf 5 area 5
!
router eigrp INE_CCIE
```

https://t.me/learningnets

```
 !
 address-family ipv4 unicast autonomous-system 789
  !
  topology base
   redistribute ospf 5 metric 1000000 100 255 1 1500
  exit-af-topology
 exit-address-family

R13:
router ospf 5
 router-id 192.122.3.13
 redistribute eigrp 789 subnets
!
interface GigabitEthernet1.113
 ip ospf 5 area 5
!
interface GigabitEthernet1.134
 ip ospf 5 area 5
!
interface Loopback0
 ip ospf 5 area 5

R14:
router ospf 5
 router-id 192.122.3.14
 redistribute eigrp 789 subnets
!
interface GigabitEthernet1.124
 ip ospf 5 area 5
!
interface GigabitEthernet1.134
 ip ospf 5 area 5
!
interface Loopback0
 ip ospf 5 area 5

R15:
router ospf 5
 router-id 192.122.3.15
 redistribute eigrp 789 subnets
!
interface GigabitEthernet1.134
 ip ospf 5 area 5
!
interface Loopback0
```

https://t.me/learningnets

```
ip ospf 5 area 5
```

Task 2.2 Verification

Note that the redistribution in this scenario did not require special attention, even though it is mutually being done on both R11 and R12. This is possible because of EIGRP's high default external AD of 170 and low default internal AD of 90.

After the redistribution, R13 and R14 are routing toward R11 and R12, respectively, to reach R10's Loopback0.

https://t.me/learningnets

```
R13#show ip route 192.122.3.10

Routing entry for 192.122.3.10/32
  Known via "ospf 5", distance 110, metric 20, type extern 2, forward metric 1
  Last update from 172.19.113.11 on GigabitEthernet1.113, 00:09:22 ago
  Routing Descriptor Blocks:   *
172.19.113.11, from 192.122.3.11, 00:09:22 ago, via GigabitEthernet1.113

      Route metric is 20, traffic share count is 1
R13#show ip cef 192.122.3.10 detail

192.122.3.10/32, epoch 2 nexthop 172.19.113.11 GigabitEthernet1.113

R13#traceroute 192.122.3.10

Type escape sequence to abort.
Tracing the route to 192.122.3.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.19.113.11 1 msec 2 msec 0 msec
  2 172.19.102.10 1 msec *  2 msec
R14#show ip route 192.122.3.10

Routing entry for 192.122.3.10/32
  Known via "ospf 5", distance 110, metric 20, type extern 2, forward metric 1
  Last update from 172.19.124.12 on GigabitEthernet1.124, 00:10:33 ago
  Routing Descriptor Blocks:   *
172.19.124.12, from 192.122.3.12, 00:10:33 ago, via GigabitEthernet1.124

      Route metric is 20, traffic share count is 1
R14#show ip cef 192.122.3.10 detail

192.122.3.10/32, epoch 2
  nexthop 172.19.124.12 GigabitEthernet1.124
R14#traceroute 192.122.3.10

Type escape sequence to abort.
Tracing the route to 192.122.3.10
VRF info: (vrf in name/id, vrf out name/id)
  1 172.19.124.12 1 msec 3 msec 0 msec
  2 172.19.102.10 2 msec
```

https://t.me/learningnets

Now we need to verify full reachability between the Loopback0 networks of all devices in Site A.

```
R15#tclsh
 R15(tcl)#proc ping-script {} {
+>foreach i {
+>192.122.3.10
+>192.122.3.11
+>192.122.3.12
+>192.122.3.13
+>192.122.3.14
+>192.122.3.15
+>} { ping $i source lo0 }
+>}
R15(tcl)#ping-script
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.10, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/24 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.11, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/10/14 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.12, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 10/15/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.13, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.14, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.15, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/5 msR15(tcl)#
```

https://t.me/learningnets

```
R10#tclsh
 R10(tcl)#proc ping-script {} {
+>foreach i {
+>192.122.3.10
+>192.122.3.11
+>192.122.3.12
+>192.122.3.13
+>192.122.3.14
+>192.122.3.15
+>} { ping $i source lo0 }
+>}
R10(tcl)#ping-script
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.10, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.10
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.11, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.10
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.12, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.10
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/8 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.13, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.10
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/10/15 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.14, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.10
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/13/16 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.15, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.10
!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 msR10(tcl)#
```

Task 3.1

Returning to the core network, we are tasked with setting up LDP on all of the IGP zones. The core network will be used as transit for MPLS VPNs as well as DMVPNs later, so ensuring that our labeling protocol is set up correctly is critical for other technologies overlayed on top layer.

- OSPF and ISIS have a "shortcut" for configuring LDP. Using this shortcut is what the requirement is referring to. Instead of going under all interfaces running the IGP and configuring `mpls ip` , we can simply add one command under the routing process to automatically enable LDP on all interfaces running OSPF: `mpls ldp autoconfig <area id>` .
- R3 and R4 must perform some label filtering so that they only advertise labels for the Loopback0 networks of R2, R6, R7, and R8.
- R6 must require authentication for all LDP peers.

Pitfall

Unlike IGPs and BGP, LDP uses the router-id to establish the TCP session. Later in this lab we are asked to configure new loopbacks on several devices. If we do not hard-code the LDP router-id to something we know for certain will be reachable, we could inadvertently break LDP in a future task. Configuring manual router-ids for LDP is a best practice, similar to IGP/BGP.

Task 3.1 Solutions

```
R1:
mpls ldp router-id Loopback0 force
!
interface GigabitEthernet1.17
 mpls ip
interface GigabitEthernet1.18
 mpls ip
!
router ospf 100
 mpls ldp autoconfig

R2:
mpls ldp router-id Loopback0 force
```

https://t.me/learningnets

```
!
router ospf 100
 mpls ldp autoconfig

R3:
mpls ldp router-id Loopback0 force
no mpls ldp advertise-labels
mpls ldp advertise-labels for 10
mpls ldp neighbor 192.122.3.6 password CC!E_!nE
!
interface GigabitEthernet1.35
 mpls ip
interface GigabitEthernet1.36
 mpls ip
!
router ospf 100
 mpls ldp autoconfig
!
access-list 10 permit 192.122.3.2
access-list 10 permit 192.122.3.6
access-list 10 permit 192.122.3.7
access-list 10 permit 192.122.3.8

R4:
mpls ldp router-id Loopback0 force
no mpls ldp advertise-labels
mpls ldp advertise-labels for 10
mpls ldp neighbor 192.122.3.6 password CC!E_!nE
!
interface GigabitEthernet1.45
 mpls ip
interface GigabitEthernet1.46
 mpls ip
!
router ospf 100
 mpls ldp autoconfig
!
access-list 10 permit 192.122.3.2
access-list 10 permit 192.122.3.6
access-list 10 permit 192.122.3.7
access-list 10 permit 192.122.3.8

R5:
mpls ldp router-id Loopback0 force
mpls ldp neighbor 192.122.3.6 password CC!E_!nE
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.35
 mpls ip
interface GigabitEthernet1.45
 mpls ip
interface GigabitEthernet1.56
 mpls ip

R6:
mpls ldp router-id Loopback0 force
mpls ldp password required
mpls ldp neighbor 192.122.3.3 password CC!E_!nE
mpls ldp neighbor 192.122.3.4 password CC!E_!nE
mpls ldp neighbor 192.122.3.5 password CC!E_!nE
!
interface GigabitEthernet1.36
 mpls ip
interface GigabitEthernet1.46
 mpls ip
interface GigabitEthernet1.56
 mpls ip

R7:
mpls ldp router-id Loopback0 force
!
interface GigabitEthernet1.17
 mpls ip
interface GigabitEthernet1.78
 mpls ip

R8:
mpls ldp router-id Loopback0 force
!
interface GigabitEthernet1.18
 mpls ip
interface GigabitEthernet1.58
 mpls ip
interface GigabitEthernet1.78
 mpls ip


R16:
mpls ldp router-id Loopback0 force
!
router ospf 100
```

https://t.me/learningnets

```
mpls ldp autoconfig
```

Task 3.1 Verification

There are a few useful commands that will help us verify this section. To start, we can look at the interfaces enabled for MPLS on the routers in the OSPF domain. Note that these routers have the `mpls ldp autoconfig` command. Interfaces running IGP on all devices in the core network should be running LDP; peers should be established with all of the directly connected neighbors.

```
R2#show mpls interfaces

Interface              IP            Tunnel   BGP Static Operational
GigabitEthernet1.12    Yes (ldp)     No       No  No     Yes
GigabitEthernet1.24    Yes (ldp)     No       No  No     Yes
GigabitEthernet1.216   Yes (ldp)     No       No  No     Yes

R2#show mpls ldp neighbor | include Peer LDP Ident:
    Peer LDP Ident: 192.122.3.1:0; Local LDP Ident 192.122.3.2:0
    Peer LDP Ident: 192.122.3.16:0; Local LDP Ident 192.122.3.2:0
    Peer LDP Ident: 192.122.3.4:0; Local LDP Ident 192.122.3.2:0

R3#show mpls interfaces

Interface              IP            Tunnel   BGP Static Operational
GigabitEthernet1.13    Yes (ldp)     No       No  No     Yes
GigabitEthernet1.34    Yes (ldp)     No       No  No     Yes
GigabitEthernet1.35    Yes (ldp)     No       No  No     Yes
GigabitEthernet1.36    Yes (ldp)     No       No  No     Yes

R3#show mpls ldp neighbor | include Peer LDP Ident:

    Peer LDP Ident: 192.122.3.1:0; Local LDP Ident 192.122.3.3:0
    Peer LDP Ident: 192.122.3.4:0; Local LDP Ident 192.122.3.3:0
    Peer LDP Ident: 192.122.3.5:0; Local LDP Ident 192.122.3.3:0
    Peer LDP Ident: 192.122.3.6:0; Local LDP Ident 192.122.3.3:0
```

To verify that both R3 and R4 are only advertising labels for the Loopback0 networks of R2, R6, R7, and R8, we must look at the LIB and LFIB of a few devices.

https://t.me/learningnets

The following output shows R3's LIB. All of the FECs that R3 will advertise are marked by "Advert acl(s): Prefix acl 10". Although this output does not show the actual label mapping, it does indicate that there is an ACL in place doing filtering.

https://t.me/learningnets

```
R3#show mpls ldp bindings advertisement-acls

Advertisement spec:
        Prefix acl = 10

  lib entry: 10.0.12.0/30, rev 63
  lib entry: 10.0.13.0/30, rev 64
  lib entry: 10.0.13.1/32, rev 65
  lib entry: 10.0.13.2/32, rev 66
        no local binding
  lib entry: 10.0.14.0/30, rev 67
  lib entry: 10.0.23.0/30, rev 68
  lib entry: 10.0.24.0/30, rev 69
  lib entry: 10.0.34.0/30, rev 70
  lib entry: 10.0.35.0/30, rev 71
  lib entry: 10.0.36.0/30, rev 72
  lib entry: 10.0.45.0/30, rev 73
  lib entry: 10.0.46.0/30, rev 74
  lib entry: 10.0.56.0/30, rev 75
  lib entry: 89.211.116.0/25, rev 76
  lib entry: 89.211.117.0/25, rev 77
  lib entry: 136.5.17.0/24, rev 78
  lib entry: 136.5.18.0/24, rev 79
  lib entry: 136.6.58.0/24, rev 80
  lib entry: 169.254.254.0/24, rev 81
  lib entry: 172.30.78.0/27, rev 82
  lib entry: 192.122.3.1/32, rev 83
  lib entry: 192.122.3.2/32, rev 95          Advert acl(s): Prefix acl 10

  lib entry: 192.122.3.3/32, rev 85
  lib entry: 192.122.3.4/32, rev 86
  lib entry: 192.122.3.5/32, rev 87
  lib entry: 192.122.3.6/32, rev 96          Advert acl(s): Prefix acl 10

  lib entry: 192.122.3.7/32, rev 97          Advert acl(s): Prefix acl 10

  lib entry: 192.122.3.8/32, rev 98          Advert acl(s): Prefix acl 10

  lib entry: 192.122.3.16/32, rev 91
  lib entry: 192.168.1.0/24, rev 92
  lib entry: 202.4.60.0/31, rev 100
        no local binding
```

https://t.me/learningnets

```
  lib entry: 202.4.216.0/26, rev 94
```

Let's pick a route for which we should NOT advertise a label. R16's Loopback0 can be an example. R3 has this route installed in the RIB via OSPF and has two next hops, R4 and R1.

```
R3#show ip route 192.122.3.16

Routing entry for 192.122.3.16/32
  Known via "ospf 100", distance 110, metric 20, type extern 2, forward metric 3
  Redistributing via rip
  Advertised by rip metric 1 route-map OSPF_INTO_RIP_REDISTRIBUTION
  Last update from 10.0.34.2 on GigabitEthernet1.34, 2d02h ago
  Routing Descriptor Blocks:
    10.0.34.2, from 192.122.3.2, 2d02h ago, via GigabitEthernet1.34
      Route metric is 20, traffic share count is 1
  * 10.0.13.1, from 192.122.3.2, 2d02h ago, via GigabitEthernet1.13
      Route metric is 20, traffic share count is 1
```

In normal circumstances, we should have a label from both R1 and R4 installed in the LFIB. However, because R4 is only advertising labels for the four loopbacks mentioned above, we should only have a label installed in the LFIB from R1.

```
R3#show mpls forwarding-table 192.122.3.16

Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
36         34         192.122.3.16/32  0             Gi1.13     10.0.13.1   No Label
   192.122.3.16/32  0 Gi1.34
     10.0.34.2

R3#show ip cef 192.122.3.16  detail

192.122.3.16/32, epoch 2, per-destination sharing
  local label info: global/36
  nexthop 10.0.13.1 GigabitEthernet1.13 label 34 nexthop 10.0.34.2 GigabitEthernet1.34
```

To further verify, we can look at the LIB of R3 and see that all of our LDP peers advertised a label for this FEC (because of Cisco's implementation of LDP that uses downstream unsolicited with liberal label retention mode), with the exception of R4.

```
R3#show mpls ldp bindings 192.122.3.16 32
```

https://t.me/learningnets

```
lib entry: 192.122.3.16/32, rev 91
      local binding:  label: 36
      remote binding: lsr: 192.122.3.1:0, label: 34
      remote binding: lsr: 192.122.3.5:0, label: 38
      remote binding: lsr: 192.122.3.6:0, label: 39
```

To verify the dataplane, we can run an MPLS traceroute from R2 to R7/R8.

https://t.me/learningnets

```
R2#traceroute mpls ipv4 192.122.3.7 255.255.255.255

more work needed here to demux the tfs subtlv and to display the right output

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
  0 10.0.12.2 MRU 1500 [Labels: 32 Exp: 0]
L 1 10.0.12.1 MRU 1500 [Labels: implicit-null Exp: 0] 5 ms
! 2 136.5.17.7 6 ms

R2#traceroute mpls ipv4 192.122.3.8 255.255.255.255

more work needed here to demux the tfs subtlv and to display the right output

Codes: '!' - success, 'Q' - request not sent, '.' - timeout,
  'L' - labeled output interface, 'B' - unlabeled output interface,
  'D' - DS Map mismatch, 'F' - no FEC mapping, 'f' - FEC mismatch,
  'M' - malformed request, 'm' - unsupported tlvs, 'N' - no label entry,
  'P' - no rx intf label prot, 'p' - premature termination of LSP,
  'R' - transit router, 'I' - unknown upstream index,
  'l' - Label switched with FEC change, 'd' - see DDMAP for return code,
  'X' - unknown return code, 'x' - return code 0

Type escape sequence to abort.
  0 10.0.12.2 MRU 1500 [Labels: 33 Exp: 0]
L 1 10.0.12.1 MRU 1500 [Labels: implicit-null Exp: 0] 10 ms
! 2 136.5.18.8 3 ms
```

The last requirement was to ensure that R6 required authentication with all of its LDP peers. This was accomplished by using the `mpls ldp password required` command on R6. This command takes an optional ACL, but in our case it is not necessary because we must require it for all LDP peers.

https://t.me/learningnets

All three LDP sessions are "up" and appear as "MD5 on" and "Password: required" and "in use."

```
R6#show mpls ldp neighbor detail

    Peer LDP Ident: 192.122.3.3:0; Local LDP Ident 192.122.3.6:0
TCP connection: 192.122.3.3.646 - 192.122.3.6.50938; MD5 on
 Password: required, neighbor, in use
        State: Oper; Msgs sent/rcvd: 69/42; Downstream; Last TIB rev sent 64
        Up time: 00:31:01; UID: 14; Peer Id 3
        LDP discovery sources:
          GigabitEthernet1.36; Src IP addr: 10.0.36.1
            holdtime: 15000 ms, hello interval: 5000 ms
        Addresses bound to peer LDP Ident:
          169.254.254.109 10.0.13.2       10.0.23.2       10.0.34.1
          10.0.35.1       10.0.36.1       192.168.1.3     192.122.3.3
    <output snipped>
    Peer LDP Ident: 192.122.3.4:0; Local LDP Ident 192.122.3.6:0
TCP connection: 192.122.3.4.646 - 192.122.3.6.39803; MD5 on
 Password: required, neighbor, in use
        State: Oper; Msgs sent/rcvd: 70/42; Downstream; Last TIB rev sent 64
        Up time: 00:30:57; UID: 15; Peer Id 0
        LDP discovery sources:
          GigabitEthernet1.46; Src IP addr: 10.0.46.1
            holdtime: 15000 ms, hello interval: 5000 ms
        Addresses bound to peer LDP Ident:
          169.254.254.107 10.0.14.2       10.0.24.2       10.0.34.2
          10.0.45.1       10.0.46.1       192.168.1.4     192.122.3.4
    <output snipped>
    Peer LDP Ident: 192.122.3.5:0; Local LDP Ident 192.122.3.6:0
TCP connection: 192.122.3.5.646 - 192.122.3.6.39021; MD5 on
 Password: required, neighbor, in use

        State: Oper; Msgs sent/rcvd: 70/69; Downstream; Last TIB rev sent 64
        Up time: 00:30:57; UID: 16; Peer Id 1
        LDP discovery sources:
          GigabitEthernet1.56; Src IP addr: 10.0.56.1
            holdtime: 15000 ms, hello interval: 5000 ms
        Addresses bound to peer LDP Ident:
          169.254.254.100 10.0.35.2       10.0.45.2       10.0.56.1
          136.6.58.5      192.168.1.5     192.122.3.5
    <output snipped>
```

https://t.me/learningnets

Task 3.2

This task requires us to construct the VRFs for the VPN between Site A and Site B on PE routers R2, R7, and R8. Everything we need is given to us with the exception of the route-target policy. The requirements ask for both VPN routes to be able to exchange routes. R9 should be able to receive routes from Site A via both R7 and R8. This is a very straightforward "full mesh" route policy, where the customer routes are not filtered in any way and are freely imported into the VRF. For this scenario, we will import and export the same route-target on all PE devices.

The last requirement is to support native IPv6 inside of the VRF. This can only be done by using the newer VRF syntax in Cisco IOS. Instead of "ip vrf ABC," the new syntax is "vrf definition ABC." Likewise, under the interface, instead of "ip vrf ABC," the new syntax is "vrf forwarding ABC." This new kind of VRF follows the same rules as the "legacy" VRF, but it has support for multiple address families. As long as the IPv6 address family is configured under the VRF's definition, the interface belonging to that VRF can run IPv6 natively. With legacy VRFs, a PE router's IPv4 address configured on an VRF interface was placed on the VRF table, but the IPv6 address was placed in the global table.

Note: Just as with legacy VRFs, the IPv4 AND IPv6 address is removed from an interface as soon as it is placed into a VRF. Remember to re-address the links with both IPv4 and IPv6 as they were configured in the initial configurations.

Task 3.2 Solutions

```
R2:
vrf definition VPN_CCIE
 rd 65066:200
 route-target export 100:100
 route-target import 100:100
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
!
interface GigabitEthernet1.210
 vrf forwarding VPN_CCIE
 ip address 202.4.210.2 255.255.255.0
 ipv6 address 2001:202:4:210::2/64
```

https://t.me/learningnets

```
R7:
vrf definition VPN_CCIE
 rd 65066:700
 route-target export 100:100
 route-target import 100:100
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
!
interface GigabitEthernet1.79
 vrf forwarding VPN_CCIE
 ip address 172.30.79.7 255.255.255.224
 ipv6 address 2001:172:30:79::7/120

R8:
vrf definition VPN_CCIE
 rd 65006:800
 route-target export 100:100
 route-target import 100:100
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
!
interface GigabitEthernet1.89
 vrf forwarding VPN_CCIE
 ip address 172.30.89.8 255.255.255.224
 ipv6 address 2001:172:30:89::8/120
```

Task 3.2 Verification

To verify this task, we can ping the CE devices from the VRF on both IPv4 and IPv6.

```
R2#ping vrf VPN_CCIE 202.4.210.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 202.4.210.10, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
```

https://t.me/learningnets

```
R2#ping vrf VPN_CCIE 2001:202:4:210::10

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:202:4:210::10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/20 ms
R2#
R7#ping vrf VPN_CCIE 172.30.79.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.79.9, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
R7#ping vrf VPN_CCIE 2001:172:30:79::9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:172:30:79::9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/21 ms
R7#
R8#ping vrf VPN_CCIE 172.30.89.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.89.9, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
R8#ping vrf VPN_CCIE 2001:172:30:89::9

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:172:30:89::9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/16 ms
R8#
```

Task 3.3

The task requirements in this section call for a BGP VPNv4 setup between the PE routers and a Route Reflector, R4. We are asked to configure BGP authentication on the peerings and to ensure that no other address family is enabled between these peers. R4 must have all of its RR Clients in a single peer-group.

Task 3.3 Solutions

```
R2:
router bgp 65006
```

https://t.me/learningnets

```
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 192.122.3.4 remote-as 65006
 neighbor 192.122.3.4 update-source Loopback0
 neighbor 192.122.3.4 password CC!E_!nE
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 192.122.3.4 activate
```

R4:

```
router bgp 65006
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor VPNv4_iBGP peer-group
 neighbor VPNv4_iBGP remote-as 65006
 neighbor VPNv4_iBGP password CC!E_!nE
 neighbor VPNv4_iBGP update-source Loopback0
 neighbor 192.122.3.2 peer-group VPNv4_iBGP
 neighbor 192.122.3.7 peer-group VPNv4_iBGP
 neighbor 192.122.3.8 peer-group VPNv4_iBGP
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor VPNv4_iBGP send-community extended
  neighbor VPNv4_iBGP route-reflector-client
  neighbor 192.122.3.2 activate
  neighbor 192.122.3.7 activate
  neighbor 192.122.3.8 activate
 exit-address-family
```

R7:

```
router bgp 65006
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 192.122.3.4 remote-as 65006
 neighbor 192.122.3.4 update-source Loopback0
 neighbor 192.122.3.4 password CC!E_!nE
 !
 address-family ipv4
 exit-address-family
 !
```

https://t.me/learningnets

```
 address-family vpnv4
  neighbor 192.122.3.4 activate

R8:
router bgp 65006
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 192.122.3.4 remote-as 65006
 neighbor 192.122.3.4 update-source Loopback0
 neighbor 192.122.3.4 password CC!E_!nE
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 192.122.3.4 activate
```

Task 3.3 Verification

Check the state of the peerings to verify this task. Note that no prefixes have been advertised yet.

```
R4#show bgp vpnv4 unicast all summary
BGP router identifier 192.122.3.4, local AS number 65006
BGP table version is 1, main routing table version 1
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
  State/PfxRcd 192.122.3.2     4        65006       5       6        1    0    0 00:02:20
        0 192.122.3.7     4        65006       5       4        1    0    0 00:02:09
        0 192.122.3.8     4        65006       5       4        1    0    0 00:02:11
        0

R4#show ip bgp peer-group
 BGP peer-group is VPNv4_iBGP,  remote AS 65006
  BGP version 4
  Neighbor sessions:
    0 active, is not multisession capable (disabled)
  Default minimum time between advertisement runs is 0 seconds
 For address family: VPNv4 Unicast
  BGP neighbor is VPNv4_iBGP, peer-group internal, members: 192.122.3.2 192.122.3.7 192.122.3.8

  Index 0, Advertise bit 0 Route-Reflector Client

  Interface associated: (none)
  Update messages formatted 0, replicated 0
```

https://t.me/learningnets

```
  Number of NLRIs in the update sent: max 0, min 0
```

Check one of the peers to see some other detailed output.

```
R4#show bgp vpnv4 unicast all neighbors 192.122.3.2

BGP neighbor is 192.122.3.2,  remote AS 65006, internal link
Member of peer-group VPNv4_iBGP for session parameters
  BGP version 4, remote router ID 192.122.3.2 BGP state = Established
, up for 00:06:00
  Last read 00:00:20, last write 00:00:36, hold time is 180, keepalive interval is 60 seconds
  Neighbor sessions:
    1 active, is not multisession capable (disabled)
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    Four-octets ASN Capability: advertised and received
Address family VPNv4 Unicast: advertised and received
    Enhanced Refresh Capability: advertised and received
    Multisession Capability:
    Stateful switchover support enabled: NO for session 1

<output snipped>
 For address family: VPNv4 Unicast
  Session: 192.122.3.2
  BGP table version 1, neighbor version 1/0
  Output queue size : 0
  Index 1, Advertise bit 0 Route-Reflector Client
  1 update-group member VPNv4_iBGP peer-group member
  Slow-peer detection is disabled
  Slow-peer split-update-group dynamic is disabled
  Interface associated: (none)
                                Sent       Rcvd
<output snipped>
 Connection state is ESTAB, I/O status: 1, unread input bytes: 0
           Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 192.122.3.4, Local port: 35658
Foreign host: 192.122.3.2, Foreign port: 179
Connection tableid (VRF): 0
Maximum output segment queue size: 50

<output snipped>

SRTT: 699 ms, RTTO: 2656 ms, RTV: 1957 ms, KRTT: 0 ms
minRTT: 2 ms, maxRTT: 1000 ms, ACK hold: 200 ms
uptime: 362937 ms, Sent idletime: 20097 ms, Receive idletime: 20297 ms
```

https://t.me/learningnets

```
Status Flags: active open Option Flags: nagle, path mtu capable, md5


IP Precedence value : 6
```

Another important verification step is to ensure that none of these devices are running another address family besides VPNv4. We accomplished this by entering the `no bgp default ipv4-unicast` command, although we could have also deactivated each peer manually under `address-family ipv4 unicast` .

After this task, the lab network is ready for some VPNs. Routing and labeling are properly configured in the core; we verified LSPs between our PEs. The VRFs are provisioned with the required route-target policies on the PEs, and now we have configured VPNv4 BGP between the PEs and the RR. All of the verification steps taken thus far will save time later if something goes wrong.

Task 3.4

The ultimate goal of this task is to establish reachability between Site A and Site B. The task specifies that the PE/CE protocol used by both sites be eBGP, and both sites have been allocated AS 65100. The eBGP PE/CE peerings should be configured using the directly connected links, and R9 must advertise its networks using `network` statements. By the end of this task, R9 must have full reachability to the Loopback0 networks of Site A.

Do you see any issues?

The first thing you should notice is that both sites use the same AS. This will prevent the sites from learning each other's routes due to BGP's loop prevention mechanism. There are two simple ways to handle this: use `as-override` on the PEs eBGP session toward the CEs, or use `allow-as in` on the CEs eBGP sessiong toward the PEs. The task did not specify, so the solution for this task will utilize `as-override` .

Another issue that will happen when R10 redistributes the BGP routes into EIGRP: In the best case scenario, we will end up with sub-optimal routing; in the worst case scenario, we will end up with a routing loop.

The BGP routes that R10 will redistribute into EIGRP will be seen as EX with an AD of 170. R11 is redistributing EIGRP into OSPF (configured in a previous task), so R12 will quickly learn those routes via OSPF and prefer that advertisement. R12 is also redistributing OSPF into EIGRP, so R12 will advertise that "OSPF" route BACK into EIGRP (not good!). The metric that we use on R10 when redistributing BGP into EIGRP will decide whether we end up with only sub-optimal routing (R12 routing

https://t.me/learningnets

into the OSPF domain to reach R9) or a routing loop. If R12's redistribution metric is better than R10's, R11 will start routing toward R12 to get to R9 (loop). If R12's metric is worse than R10's, the R11 will continue routing in the correct direction toward R10, but R12 will route sub-optimally into the OSPF domain.

There are multiple ways to solve this issue. An approach that we did not explore in earlier tasks is the use of summarization. If R11 and R12 summarize R9's networks as they advertise them into OSPF, the EIGRP domain will continue to route optimally using the longest match routes and the more specific routes will not be fed BACK into EIGRP. In this lab, using summarization is not prohibited. However, you may encounter other cases where you are not allowed to use it. It is critical that you understand the tools available at your disposal when solving these types of routing issues.

Task 3.4 Solutions

```
R2:
router bgp 65006
 address-family ipv4 vrf VPN_CCIE
  neighbor 202.4.210.10 remote-as 65100
  neighbor 202.4.210.10 activate
  neighbor 202.4.210.10 as-override
 exit-address-family

R7:
router bgp 65006
 address-family ipv4 vrf VPN_CCIE
  neighbor 172.30.79.9 remote-as 65100
  neighbor 172.30.79.9 activate
  neighbor 172.30.79.9 as-override
 exit-address-family

R8:
router bgp 65006
 address-family ipv4 vrf VPN_CCIE
  neighbor 172.30.89.9 remote-as 65100
  neighbor 172.30.89.9 activate
  neighbor 172.30.89.9 as-override
 exit-address-family

R9:
router bgp 65100
 bgp log-neighbor-changes
 network 172.30.79.0 mask 255.255.255.224
```

https://t.me/learningnets

```
 network 172.30.89.0 mask 255.255.255.224
 network 192.122.3.9 mask 255.255.255.255
 neighbor 172.30.79.7 remote-as 65006
 neighbor 172.30.89.8 remote-as 65006

R10:
router bgp 65100
 bgp log-neighbor-changes
 redistribute eigrp 789
 neighbor 202.4.210.2 remote-as 65006
!
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 789
  !
  topology base
   redistribute bgp 65100 metric 1000000 100 255 1 1500
  exit-af-topology
 exit-address-family

R11:
router ospf 5
 summary-address 172.30.0.0 255.255.0.0
 summary-address 192.122.3.8 255.255.255.254

R12:
router ospf 5
 summary-address 172.30.0.0 255.255.0.0
 summary-address 192.122.3.8 255.255.255.254
```

Task 3.4 Verification

To begin verification, look at R9 and make sure that it is receiving all of Site A's routes. Both R7 and R8 should be advertising those routes to R9 (route-target policy requirement). The output here also tells us that we overcame the BGP AS PATH loop prevention mechanism, and that redistribution on R10 worked. Note that the task did not specifically say to do redistribution; we could have done network statements on R10 for all of Site A's Loopback0 networks. However, we would have had to also take care of advertising the Site B routes into EIGRP. Doing redistribution was the easiest way to accomplish reachability.

```
R9#show bgp ipv4 unicast

BGP table version is 24, local router ID is 192.122.3.9
```

https://t.me/learningnets

```
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   172.19.102.0/25  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   172.19.113.0/25  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   172.19.124.0/25  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   172.19.134.0/25  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   172.30.0.0       172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *>  172.30.79.0/27   0.0.0.0                  0         32768 i
 *>  172.30.89.0/27   0.0.0.0                  0         32768 i
 *   192.122.3.8/31   172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *>  192.122.3.9/32   0.0.0.0                  0         32768 i
 *   192.122.3.10/32  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   192.122.3.11/32  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   192.122.3.12/32  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   192.122.3.13/32  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   192.122.3.14/32  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
 *   192.122.3.15/32  172.30.89.8                             0 65006 65006 ?
 *>                   172.30.79.7                             0 65006 65006 ?
```

R10's should see the same prefixes.

```
R10#show bgp ipv4 unicast

BGP table version is 20, local router ID is 192.122.3.10
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

https://t.me/learningnets

```
     Network          Next Hop            Metric LocPrf Weight Path
 *>  172.19.102.0/25  0.0.0.0                  0         32768 ?
 *>  172.19.113.0/25  172.19.102.11       576264         32768 ?
 *>  172.19.124.0/25  172.19.102.11       576264         32768 ?
 *>  172.19.134.0/25  172.19.102.11       576264         32768 ?
 *>  172.30.0.0       172.19.102.12       576264         32768 ?
 *>  172.30.79.0/27   202.4.210.2                           0 65006 65006 i
 *>  172.30.89.0/27   202.4.210.2                           0 65006 65006 i
 *>  192.122.3.8/31   172.19.102.11       576264         32768 ?
 *>  192.122.3.9/32   202.4.210.2                           0 65006 65006 i
 *>  192.122.3.10/32  0.0.0.0                  0         32768 ?
 *>  192.122.3.11/32  172.19.102.11        12005         32768 ?
 *>  192.122.3.12/32  172.19.102.12        12005         32768 ?
 *>  192.122.3.13/32  172.19.102.11       576264         32768 ?
 *>  192.122.3.14/32  172.19.102.11       576264         32768 ?
 *>  192.122.3.15/32  172.19.102.11       576264         32768 ?
```

R10 should be redistributing these BGP routes into EIGRP; let's verify that.

```
R10#show ip route bgp | begin Gateway
Gateway of last resort is not set

      172.30.0.0/16 is variably subnetted, 3 subnets, 2 masks
B        172.30.79.0/27 [20/0] via 202.4.210.2, 00:25:49
B        172.30.89.0/27 [20/0] via 202.4.210.2, 00:27:59
      192.122.3.0/24 is variably subnetted, 8 subnets, 2 masks
B        192.122.3.9/32 [20/0] via 202.4.210.2, 00:25:49

R10#show ip eigrp topology 192.122.3.9/32
EIGRP-IPv4 VR(INE_CCIE) Topology Entry for AS(789)/ID(192.122.3.10) for 192.122.3.9/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 66191360
  Descriptor Blocks:  202.4.210.2, from Redistributed
, Send flag is 0x0
      Composite metric is (66191360/0), route is External
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 1000000000 picoseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 0 Originating router is 192.122.3.10
      External data: AS number of route is 65100
 External protocol is BGP
, external metric is 0
```

https://t.me/learningnets

```
        Administrator tag is 65006 (0x0000FDEE)
```

R11 and R12 were configured to do mutual redistribution between EIGRP and OSPF, so these routes should have been redistributed into OSPF. Because of the routing issues that this would have caused, we used summarization into the OSPF domain on R11 and R12 for these routes.

```
R11#show ip ospf database external 172.30.0.0 self-originate



            OSPF Router with ID (192.122.3.11) (Process ID 5)


                Type-5 AS External Link States


  LS age: 1541
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link Link State ID: 172.30.0.0 (External Network Number )
  Advertising Router: 192.122.3.11
  LS Seq Number: 80000001
  Checksum: 0xC7C4
  Length: 36
  Network Mask: /16
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 20  Forward Address: 0.0.0.0


        External Route Tag: 0
```

R15 should see two equal cost routes for these two summaries.

```
R15#show ip route 172.30.0.0
Routing entry for 172.30.0.0/16
  Known via "ospf 5", distance 110, metric 20, type extern 2
, forward metric 2
  Last update from 172.19.134.14 on GigabitEthernet1.134, 00:26:48 ago
  Routing Descriptor Blocks:
    172.19.134.14, from 192.122.3.12, 00:26:48 ago, via GigabitEthernet1.134
      Route metric is 20, traffic share count is 1
  * 172.19.134.13, from 192.122.3.11, 00:26:57 ago, via GigabitEthernet1.134
      Route metric is 20, traffic share count is 1
R15#show ip route 192.122.3.9
Routing entry for 192.122.3.8/31
  Known via "ospf 5", distance 110, metric 20, type extern 2
, forward metric 2
  Last update from 172.19.134.13 on GigabitEthernet1.134, 00:26:15 ago
```

https://t.me/learningnets

```
  Routing Descriptor Blocks:
  * 172.19.134.14, from 192.122.3.12, 00:26:22 ago, via GigabitEthernet1.134
      Route metric is 20, traffic share count is 1
    172.19.134.13, from 192.122.3.11, 00:26:15 ago, via GigabitEthernet1.134
      Route metric is 20, traffic share count is 1
```

We can use our ping-script for reachability testing. R9 should be able to reach all Loopback0 networks in Site A when sourcing traffic from its Loopback0.

```
R9#tclsh
 R9(tcl)#proc ping-script {} {
+>(tcl)#foreach i {
+>(tcl)#192.122.3.9
+>(tcl)#192.122.3.10
+>(tcl)#192.122.3.11
+>(tcl)#192.122.3.12
+>(tcl)#192.122.3.13
+>(tcl)#192.122.3.14
+>(tcl)#192.122.3.15
+>(tcl)#} { ping $i source lo0 }
+>(tcl)#}R9(tcl)#ping-script

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.9, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.10, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/21 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.11, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 19/19/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.12, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 19/19/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.13, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.9
```

https://t.me/learningnets

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 19/19/21 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.14, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/19/20 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.122.3.15, timeout is 2 seconds:
Packet sent with a source address of 192.122.3.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/20/26 ms
#bR9(tcl)##b
```

Let's delve a little deeper into the packet forwarding in the core network by looking at the FIB of one of the PEs and verify the label stack.

```
R2#show ip cef vrf VPN_CCIE 192.122.3.9/32 detail

192.122.3.9/32, epoch 0, flags [rib defined all labels]  recursive via 192.122.3.7 label 44
    nexthop 10.0.12.1 GigabitEthernet1.12 label 32
```

R2 received VPN label 44 from R7, and it is using label 32 to get toward R7's Loopback0 (LSP toward R7).

```
R2#show bgp vpnv4 unicast vrf VPN_CCIE 192.122.3.9/32
 BGP routing table entry for 65066:200:192.122.3.9/32
, version 21
Paths: (2 available, best #2, table VPN_CCIE)
  Advertised to update-groups:
     3
  Refresh Epoch 1
  65100, imported path from 65006:800:192.122.3.9/32 (global)
    192.122.3.8 (metric 20) (via default) from 192.122.3.4 (192.122.3.4)
      Origin IGP, metric 0, localpref 100, valid, internal
      Extended Community: RT:100:100
      Originator: 192.122.3.8, Cluster list: 192.122.3.4
      mpls labels in/out nolabel/43
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  65100, imported path from 65066:700:192.122.3.9/32 (global) 192.122.3.7
 (metric 20) (via default) from 192.122.3.4 (192.122.3.4)
      Origin IGP, metric 0, localpref 100, valid, internal, best
```

https://t.me/learningnets

```
      Extended Community: RT:100:100
      Originator: 192.122.3.7, Cluster list: 192.122.3.4 mpls labels in/out nolabel/44
      rx pathid: 0, tx pathid: 0x0

R2#show mpls forwarding-table 192.122.3.7 32

Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
37         32         192.122.3.7/32   0             Gi1.12     10.0.12.1
```

Task 4.1

In this section, we are tasked with pre-staging the network for DMVPN connectivity between Site C, Site E, and Site F. The task instructs us to provide R16, R17, and R18 with "underlay" connectivity, or connectivity between their "NBMA" addresses in DMVPN terminology. R18 is the edge router for Site C and connects to the local ISP using BGP. R6 is the exit point out of the core network and into the Internet and also peers BGP with the ISP. BGP peerings to the ISP have been pre-configured on R6 and R18.

The requirements of this task are to leak the fewest number of routes as possible to establish underlay connectivity between the DMVPN routers. Additionally, we are allowed to use one static default route somewhere in the network.

To satisfy the requirements, the following actions must be executed:

- Advertise the network used on R18's Internet uplink into BGP, 202.4.180.0/31.
- Advertise one of the links that connects R16 into the OSPF domain into BGP on R6. This lab does not specify which link to use as the source of the DMVPN tunnel, so the one used for the solution will be GigabitEthernet1.116. Note that you can pick either one; advertise whichever link you select and use that interface as the source of the DMVPN tunnel.
- Advertise R17's link to R1 into BGP on R6.
- Advertise R18's BGP route into RIP on R6. Redistribution from prior tasks will take care of advertising it to the rest of the core.
- Add a static default route on R17. This is needed because R17 does not run a routing protocol with the core network. This is the only place where a static route is needed throughout the lab, and is thus permitted by the guidelines.

Prefix-lists will be used to prevent any other route from being leaked in or out of the core network on R6. Currently there are no other routes being advertised, but a later

https://t.me/learningnets

task may introduce more networks and break the requirements of this task to leak the fewest number of routes possible.

Note that as a result of this configuration, only R16 and R17 will be able to reach R18. Although all routers in the core (including R6) have the route to R18, the ISP and R18 have no route back to any router in the core besides R16 and R17. This is the desired state: explicit and controlled route advertisements.

Task 4.1 Solutions

```
R17:
ip route 0.0.0.0 0.0.0.0 GigabitEthernet1.117 89.211.117.1

R18:
router bgp 65456
 network 202.4.180.0 mask 255.255.255.254

R6:
router bgp 65006
 network 89.211.116.0 mask 255.255.255.128
 network 89.211.117.0 mask 255.255.255.128
!
router rip
 redistribute bgp 65006 metric 1 route-map BGP_INTO_RIP_REDISTRIBUTION
!
ip prefix-list R18_UNDERLAY_CONNECTIVITY seq 5 permit 202.4.180.0/31
!
route-map BGP_INTO_RIP_REDISTRIBUTION permit 10
 match ip address prefix-list R18_UNDERLAY_CONNECTIVITY
```

Task 4.1 Verification

The verification of this task is simple: verifying reachability between the DMVPN routers and ensuring that no other routes have been leaked.

R16 can reach R18 sourcing traffic from its GigabitEthernet1.116, but not from its GigabitEthernet1.216 interface.

```
R16#ping 202.4.180.0 source GigabitEthernet1.116
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 202.4.180.0, timeout is 2 seconds:
Packet sent with a source address of 89.211.116.16
!!!!!
```

https://t.me/learningnets

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/5/16 ms
R16#ping 202.4.180.0 source GigabitEthernet1.216
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 202.4.180.0, timeout is 2 seconds:
Packet sent with a source address of 202.4.216.16
.....
Success rate is 0 percent (0/5)
R16#traceroute 202.4.180.0 source GigabitEthernet1.116

Type escape sequence to abort.
Tracing the route to 202.4.180.0
VRF info: (vrf in name/id, vrf out name/id)
  1 202.4.216.2 3 msec
    89.211.116.1 1 msec
    202.4.216.2 1 msec
  2 10.0.13.2 2 msec
    10.0.24.2 1 msec
    10.0.13.2 1 msec
  3 10.0.46.2 1 msec
    10.0.36.2 1 msec
    10.0.46.2 1 msec
  4 202.4.60.1 1 msec 2 msec 1 msec
  5 202.4.180.0 2 msec *  2 msec
```

R17 can also reach R18.

```
R17#ping 202.4.180.0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 202.4.180.0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/7/21 ms
R17#traceroute 202.4.180.0

Type escape sequence to abort.
Tracing the route to 202.4.180.0
VRF info: (vrf in name/id, vrf out name/id)
  1 89.211.117.1 4 msec 2 msec 0 msec
  2 10.0.13.2 1 msec 1 msec 1 msec
  3 10.0.36.2 1 msec 2 msec 5 msec
  4 202.4.60.1 10 msec 8 msec 10 msec
  5 202.4.180.0 10 msec *  3 msec
```

R6 only has one BGP route from R18, which in turn is being advertised into RIP

https://t.me/learningnets

(verified by successful reachability from the DMVPN Spoke routers) and is explicitly advertising only the needed network from R16 and R17 via network statements.

```
R6#show bgp ipv4 unicast

BGP table version is 5, local router ID is 192.122.3.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
*>   89.211.116.0/25  10.0.36.1                1         32768 i
*>   89.211.117.0/25  10.0.36.1                1         32768 i
 *>  202.4.180.0/31   202.4.60.1                             0 65123 65456 i

R6#show ip route bgp | begin Gate
Gateway of last resort is not set

      202.4.180.0/31 is subnetted, 1 subnets
B        202.4.180.0 [20/0] via 202.4.60.1, 00:16:27

R18#show ip route bgp | begin Gateway

Gateway of last resort is not set

      89.0.0.0/25 is subnetted, 2 subnets
B        89.211.116.0 [20/0] via 202.4.180.1, 00:22:13
B        89.211.117.0 [20/0] via 202.4.180.1, 00:21:01
      202.4.216.0/26 is subnetted, 1 subnets
```

Task 4.2

This section leverages the underlay network built in the previous task to overlay the DMVPN network between Site C, Site E, and Site F. We have been given some requirements outlining the characteristics of this VPN:

- Use interface Tunnel 100 on the routers, with network ID 100 and NHRP Key of â€œNHRPKEYâ€•.
- The IP subnet that should be configured on the DMVPN Tunnel is 172.100.123.X/24, where *X* is the router number.

https://t.me/learningnets

- The Phase I negotiation is required to use AES192 for encryption, SHA256 for hashing, Pre Shared Key of â€œDmvPn!23â€• with no wildcards, and Diffie Hellman group 5.
- Phase II requires both authentication and encryption, hashing should use SHA
- Unnecessary dataplane overhead should be avoided.
- Adjust the Tunnel MTU to 1400 and set an appropriate Maximum Segment Size.

For the Phase I exchange, we are asked to not use wild card keys; this means that we need to hard-code the keys on each device with the full address of all peers. R16 could potentially use either one of its NBMA networks when establishing Phase I with the peers. For this reason, we must include both of the R16 addresses when configuring the ISAKAMP keys on the peers.

The Phase II requirements dictate that we must use authentication and encryption. This means that we should use ESP instead of AH (IP Protocol 50 instead of 51). AH does not meet the requirements because it only supports authentication.

To avoid unnecessary dataplane overhead, Transport mode must be used instead of Tunnel mode. Tunnel mode is the default when configuring the transform set, which adds one additional IP header that is unnecessary in this case because of the existing GRE encapsulation. We must also set the MSS to 1360 (40 bytes less than the 1400 MTU of the tunnel). 1360 is the maximum size of a segment that can be transmitted over the DMVPN network without being fragmented.

Task 4.2 Solutions

```
R16:
crypto isakmp policy 10
 encr aes 192
 hash sha256
 authentication pre-share
 group 5
!
crypto isakmp key DmvPn!23 address 89.211.117.17
crypto isakmp key DmvPn!23 address 202.4.180.0


!
crypto ipsec transform-set TRANSFORM_SET esp-aes esp-sha-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set TRANSFORM_SET
!
```

https://t.me/learningnets

```
interface Tunnel100
 ip address 172.100.123.16 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPKEY
 ip nhrp map 172.100.123.18 202.4.180.0
 ip nhrp map multicast 202.4.180.0
 ip nhrp network-id 100
 ip nhrp nhs 172.100.123.18
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.116
 tunnel mode gre multipoint
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown

R17:
crypto isakmp policy 10
 encr aes 192
 hash sha256
 authentication pre-share
 group 5
!
crypto isakmp key DmvPn!23 address 89.211.116.16
crypto isakmp key DmvPn!23 address 202.4.180.0

!
crypto ipsec transform-set TRANSFORM_SET esp-aes esp-sha-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set TRANSFORM_SET
!
interface Tunnel100
 ip address 172.100.123.17 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPKEY
 ip nhrp map 172.100.123.18 202.4.180.0
 ip nhrp map multicast 202.4.180.0
 ip nhrp network-id 100
 ip nhrp nhs 172.100.123.18
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.117
 tunnel mode gre multipoint
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown

R18:
```

https://t.me/learningnets

```
crypto isakmp policy 10
 encr aes 192
 hash sha256
 authentication pre-share
 group 5
!
crypto isakmp key DmvPn!23 address 89.211.116.16
crypto isakmp key DmvPn!23 address 89.211.117.17
!
crypto ipsec transform-set TRANSFORM_SET esp-aes esp-sha-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set TRANSFORM_SET
!
interface Tunnel100
 ip address 172.100.123.18 255.255.255.0
 ip mtu 1400
 ip nhrp authentication NHRPKEY
 ip nhrp map multicast dynamic
 ip nhrp network-id 100
 ip nhrp nhs 172.100.123.18
 ip tcp adjust-mss 1360
 tunnel source GigabitEthernet1.180
 tunnel mode gre multipoint
 tunnel protection ipsec profile DMVPN_PROFILE
 no shutdown
```

Task 4.2 Verification

We will look at the effects of our configuration in the control and data plane to verify this task.

Let's look at the hub, R18. Phase I was established properly with both spokes. We can see this by looking at each spoke's NBMA address and the "QM_IDLE" state next to each one. This verifies that Phase I negotiation was successful and that the exchange moved on to Phase II.

```
R18#show crypto isakmp sa

IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status 202.4.180.0 89.211.117.17   QM_IDLE
           1004 ACTIVE 202.4.180.0 89.211.116.16   QM_IDLE
           1003 ACTIVE
```

https://t.me/learningnets

```
IPv6 Crypto ISAKMP SA


R18#
```

Let's verify Phase II by looking at just one of these peers from R18's perspective. This output is lengthy, but it contains a good amount of useful information. To start, it shows that we are encapsulating, encrypting, and de-crypting IP Protocol 47 (GRE) traffic to and from R17. The packet counts for all of these fields are increasing, meaning that the data-plane is working bidirectionally. We can also see that there is an inbound and outbound SPI, used by IPSec in the data-plane. R18 will be able to identify that an encrypted packet is from R17 by looking for SPI 0xE3627F02. Similarly, when R18 sends encrypted traffic to R17, it adds an SPI of 0xE3627F02 to the packets so that R17 can "mux" and identify to which tunnel they belong. This can be thought of as the VPN label in MPLS VPNS. The output below also validates that we are using Transport mode instead of Tunnel mode and thus avoiding extra overhead from the additional IP header that Tunnel mode uses.

```
R18#show crypto ipsec sa peer 89.211.117.17



interface: Tunnel100
    Crypto map tag: Tunnel100-head-0, local addr 202.4.180.0


   protected vrf: (none)   local  ident (addr/mask/prot/port): (202.4.180.0/255.255.255.255/47/0
)   remote ident (addr/mask/prot/port): (89.211.117.17/255.255.255.255/47/0
)
   current_peer 89.211.117.17 port 500
     PERMIT, flags={origin_is_acl,}#pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 3
#pkts decaps: 3, #pkts decrypt: 3, #pkts verify: 3
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0


     local crypto endpt.: 202.4.180.0, remote crypto endpt.: 89.211.117.17
     plaintext mtu 1458, path mtu 1500, ip mtu 1500, ip mtu idb (none)
     current outbound spi: 0xB892D39D(3096630173)
     PFS (Y/N): N, DH group: none
 inbound esp sas:
 spi: 0xE3627F02(3814883074)
        transform: esp-aes esp-sha-hmac
 , in use settings ={Transport, }
        conn id: 2013, flow_id: CSR:13, sibling_flags FFFFFFFF80000008, crypto map: Tunnel100-head-0
        sa timing: remaining key lifetime (k/sec): (4607999/3326)
        IV size: 16 bytes
```

https://t.me/learningnets

```
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:
 outbound esp sas:
 spi: 0xB892D39D(3096630173)
        transform: esp-aes esp-sha-hmac
 , in use settings ={Transport, }

        conn id: 2014, flow_id: CSR:14, sibling_flags FFFFFFFF80000008, crypto map: Tunnel100-head-0
        sa timing: remaining key lifetime (k/sec): (4607999/3326)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:
```

The DMVPN output below also shows that the DMVPN control-plane is operational. We have now validated both the crypto and NHRP portions on R18.

```
R18#show dmvpn

Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel100, IPv4 NHRP Details Type:Hub, NHRP Peers:2
,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
1 89.211.116.16    172.100.123.16    UP 00:17:58     D
 1 89.211.117.17    172.100.123.17    UP 00:31:56     D


R18#show ip interface Tunnel100 | inc MTU|MSS
```

https://t.me/learningnets

```
  MTU is 1400 bytes
  Input features: MCI Check, TCP Adjust MSS
  Output features: TCP Adjust MSS
```

All three of the DMVPN routers should be able to ping between their tunnel addresses. R16 and R17 will form a dynamic spoke-to-spoke tunnel to send traffic to each other.

```
R16#show dmvpn
 Legend: Attrb -->S - Static
, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel100, IPv4 NHRP Details Type:Spoke, NHRP Peers:1
,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----     1 202.4.180.0 172.100.123.18 UP
 00:24:33 S

R16#ping 172.100.123.17
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.100.123.17, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/10/30 ms

R16#show dmvpn
         Legend: Attrb --> S - Static, D - Dynamic
, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel100, IPv4 NHRP Details
Type:Spoke, NHRP Peers:2,
```

https://t.me/learningnets

```
 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----     1 89.211.117.17 172.100.123.17
    UP 00:00:05 D
     1 202.4.180.0      172.100.123.18    UP 00:24:52     S
R16#
```

Task 4.3

Now we must implement a routing design for the DMVPN network. The requirements of this task ask for EIGRP AS 123 on all sites; R16 and R17 need to form an adjacency with the Server3 and Server2, respectively. R19 and R20 need to run EIGRP with R18 but not between each other. We will advertise this link as passive to meet the full reachability requirements. Server1 has been pre-configured with a default gateway of 172.27.192.254, and we are asked to configure an FHRP between R19 and R20 to provide this virtual IP. The task did not specify which FHRP to use, but it did specify that it must use authentication. We will use HSRPv2 with R20 as the primary and use IP SLA with object tracking to track reachability to R18's Loopback0.

The DMVPN network must be configured in DMVPN Phase III so that direct spoke-to-spoke communication is not broken if summarization from hub to spokes is introduced (a default route is also considered summarization). An additional tweak that is necessary in the current network state is to disable split horizon on R18. Without it, R16 will not learn R17's routes and vice-versa.

Server2 and Server3 have been pre-configured with EIGRP. As soon as we start the EIGRP process on R16 and R17, we should see the adjacency with the servers come up.

Task 4.3 Solutions

```
R16:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 123
  !
  topology base
  exit-af-topology
  network 172.23.160.0 0.0.0.255
  network 172.100.123.0 0.0.0.255
```

https://t.me/learningnets

```
 exit-address-family
!
interface Tunnel100
 ip nhrp shortcut

R17:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 123
  !
  topology base
  exit-af-topology
  network 172.25.170.0 0.0.0.255
  network 172.100.123.0 0.0.0.255
  network 192.122.3.17 0.0.0.0
 exit-address-family
!
interface Tunnel100
 ip nhrp shortcut

R18:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 123
  !
  af-interface Tunnel100
   no split-horizon
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 172.27.181.0 0.0.0.255
  network 172.27.182.0 0.0.0.255
  network 172.100.123.0 0.0.0.255
  network 192.122.3.18 0.0.0.0
 exit-address-family
!
interface Tunnel100
 ip nhrp redirect

R19:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 123
  !
  af-interface GigabitEthernet1.192
```

https://t.me/learningnets

```
   passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 172.27.181.0 0.0.0.255
  network 172.27.192.0 0.0.0.255
  network 192.122.3.19 0.0.0.0
 exit-address-family
!
interface GigabitEthernet1.192
 standby version 2
 standby 192 ip 172.27.192.254
 standby 192 preempt
 standby 192 authentication md5 key-string SERVER_VIP
```

R20:

```
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 123
  !
  af-interface GigabitEthernet1.192
   passive-interface
  exit-af-interface
  !
  topology base
  exit-af-topology
  network 172.27.182.0 0.0.0.255
  network 172.27.192.0 0.0.0.255
  network 192.122.3.20 0.0.0.0
 exit-address-family
!
interface GigabitEthernet1.192
 standby version 2
 standby 192 ip 172.27.192.254
 standby 192 priority 120
 standby 192 preempt
 standby 192 authentication md5 key-string SERVER_VIP
 standby 192 track 192 decrement 50
!
track 192 ip sla 10 state
!
ip sla 10
 icmp-echo 192.122.3.18
 frequency 10
!
```

https://t.me/learningnets

```
ip sla schedule 10 life forever start-time now
```

Task 4.3 Verification

Look at the RIBs of the devices in the DMVPN network to verify this task. R16 is receiving all routes, which validates proper routing in the DMVPN cloud. R16 should be able to send traffic directly to R17 without going through the hub.

```
R16#show ip route eigrp | begin Gate

Gateway of last resort is 202.4.216.2 to network 0.0.0.0

      172.25.0.0/24 is subnetted, 1 subnets
D        172.25.170.0 [90/102405120] via 172.100.123.18, 00:42:16, Tunnel100
      172.27.0.0/24 is subnetted, 3 subnets
D        172.27.181.0 [90/76805120] via 172.100.123.18, 00:43:17, Tunnel100
D        172.27.182.0 [90/76805120] via 172.100.123.18, 00:43:09, Tunnel100
D        172.27.192.0 [90/76810240] via 172.100.123.18, 00:38:09, Tunnel100
      192.122.3.0/32 is subnetted, 11 subnets
D        192.122.3.17 [90/102400640] via 172.100.123.18, 00:42:16, Tunnel100
D        192.122.3.18 [90/76800640] via 172.100.123.18, 00:43:36, Tunnel100
D        192.122.3.19 [90/76805760] via 172.100.123.18, 00:38:17, Tunnel100
D        192.122.3.20 [90/76805760] via 172.100.123.18, 00:37:24, Tunnel100
```

The first time R16 sends packets to R17, they are routed through the hub. After this initial flow, the spoke-to-spoke tunnel along with the NHRP shortcut are built built and packets sent afterward should flow directly to R17.

```
R16#traceroute 192.122.3.17
Type escape sequence to abort.
Tracing the route to 192.122.3.17
VRF info: (vrf in name/id, vrf out name/id)
  1 172.100.123.18 3 msec 2 msec 2 msec
  2 172.100.123.17 9 msec *  2 msec

R16#traceroute 192.122.3.17

Type escape sequence to abort.
Tracing the route to 192.122.3.17
VRF info: (vrf in name/id, vrf out name/id)
  1 172.100.123.17 4 msec *  2 msec
```

https://t.me/learningnets

The next-hop override can be seen for R17's loopback.

https://t.me/learningnets

```
R16#show ip route eigrp

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route       + - replicated route, % - next hop override


Gateway of last resort is 202.4.216.2 to network 0.0.0.0

      172.25.0.0/24 is subnetted, 1 subnets
D        172.25.170.0 [90/102405120] via 172.100.123.18, 00:50:45, Tunnel100
      172.27.0.0/24 is subnetted, 3 subnets
D        172.27.181.0 [90/76805120] via 172.100.123.18, 00:51:46, Tunnel100
D        172.27.182.0 [90/76805120] via 172.100.123.18, 00:51:38, Tunnel100
D        172.27.192.0 [90/76810240] via 172.100.123.18, 00:46:38, Tunnel100
      192.122.3.0/32 is subnetted, 11 subnets
D   %    192.122.3.17 [90/102400640] via 172.100.123.18, 00:50:45, Tunnel100

D        192.122.3.18 [90/76800640] via 172.100.123.18, 00:52:05, Tunnel100
D        192.122.3.19 [90/76805760] via 172.100.123.18, 00:46:46, Tunnel100
D        192.122.3.20 [90/76805760] via 172.100.123.18, 00:45:53, Tunnel100

R16#show ip route nhrp

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP
, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 202.4.216.2 to network 0.0.0.0
```

https://t.me/learningnets

```
      172.100.0.0/16 is variably subnetted, 3 subnets, 2 masks
H        172.100.123.17/32 is directly connected, 00:04:20, Tunnel100
```

We can check the FIB to see that the next-hop is correctly set to R17's Tunnel address.

```
R16#show ip cef 192.122.3.17 detail
 192.122.3.17/32
, epoch 2, flags [rib only nolabel, rib defined all labels]
  local label info: global/37 nexthop 172.100.123.17 Tunnel100
```

These steps validate that we have spoke-to-spoke connectivity. R16 should be checked as well, because if `ip nhrp shortcut` is not entered on R17, this shortcut will only work one way.

Note that we did not have to utilize Third Party Next-Hops to achieve this behavior. R18 is changing the next-hop to itself when advertising R17 routes to R16, which is one of the benefits of using DMVPN Phase III. If the requirement asked for direct spoke-to-spoke connectivity but restricted the use of Phase III, R18 would have had to leave the next-hops unmodified. This is referred to as *DMVPN Phase II*, which has the downside of preventing summarization on the hub. If the hub performs summarization in a Phase II network, the direct spoke-to-spoke traffic is not possible and thus must all transit the hub.

The servers on this network are VRFs configured on R15. The easiest way to do verification from these servers is to log in to R15 and change into the server's vrf routing-context. Let's verify server-to-server connectivity.

Note that the prompt looks different after changing the routing context. It changes from R15# to R15%server1#. The benefit of doing the verification this way is that we do not have to append vrf to all of the verification commands, such as show/ping/traceroute.

```
R15#routing-context vrf server1


R15%server1#traceroute 172.23.160.100
Type escape sequence to abort.
Tracing the route to 172.23.160.100
VRF info: (vrf in name/id, vrf out name/id)
  1 172.27.192.20 3 msec 1 msec 6 msec
  2 172.27.182.18 1 msec 2 msec 1 msec
  3 172.100.123.16 2 msec 2 msec 11 msec
  4 172.23.160.100 31 msec *  3 msec
```

https://t.me/learningnets

```
R15%server1#traceroute 172.25.170.100

Type escape sequence to abort.
Tracing the route to 172.25.170.100
VRF info: (vrf in name/id, vrf out name/id)
  1 172.27.192.20 3 msec 1 msec 6 msec
  2 172.27.182.18 1 msec 1 msec 2 msec
  3 172.100.123.17 2 msec 3 msec 12 msec
  4 172.25.170.100 31 msec *  3 msecR15%server1#
```

Now check connectivity from Server2 to Server3.

```
R15%server1#exit

R15#routing-context vrf server2

R15%server2#show ip route eigrp | begin Gateway
Gateway of last resort is not set

      172.23.0.0/24 is subnetted, 1 subnets
D        172.23.160.0
           [90/102410240] via 172.25.170.17, 01:12:14, GigabitEthernet1.170
      172.27.0.0/24 is subnetted, 3 subnets
D        172.27.181.0
           [90/76810240] via 172.25.170.17, 01:13:15, GigabitEthernet1.170
D        172.27.182.0
           [90/76810240] via 172.25.170.17, 01:13:06, GigabitEthernet1.170
D        172.27.192.0
           [90/76815360] via 172.25.170.17, 01:08:07, GigabitEthernet1.170
      172.100.0.0/24 is subnetted, 1 subnets
D        172.100.123.0
           [90/76805120] via 172.25.170.17, 01:18:28, GigabitEthernet1.170
      192.122.3.0/32 is subnetted, 4 subnets
D        192.122.3.17
           [90/10880] via 172.25.170.17, 01:18:28, GigabitEthernet1.170
D        192.122.3.18
           [90/76805760] via 172.25.170.17, 01:13:33, GigabitEthernet1.170
D        192.122.3.19
           [90/76810880] via 172.25.170.17, 01:08:15, GigabitEthernet1.170
D        192.122.3.20
           [90/76810880] via 172.25.170.17, 01:07:21, GigabitEthernet1.170

R15%server2#traceroute 172.23.160.100
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 172.23.160.100
VRF info: (vrf in name/id, vrf out name/id)
  1 172.25.170.17 3 msec 1 msec 0 msec
  2 172.100.123.16 3 msec 8 msec 55 msec
  3 172.23.160.100 15 msec *  2 msec
R15%server2#R15%server2#exit
R15#
```

Server1 has connectivity to the rest of the DMVPN network, which means that the FHRP configuration between R19 and R20 is working properly. Let's go a littler deeper into that to verify that the requirements are met.

```
R20#show standby
GigabitEthernet1.192 - Group 192 (version 2
) State is Active
    2 state changes, last state change 01:05:23 Virtual IP address is 172.27.192.254
  Active virtual MAC address is 0000.0c9f.f0c0 (MAC In Use)
    Local virtual MAC address is 0000.0c9f.f0c0 (v2 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.384 secs Authentication MD5, key-string
 Preemption enabled
 Active router is local
 Standby router is 172.27.192.19
, priority 100 (expires in 8.960 sec)
  Priority 120 (configured 120) Track object 192 state Up decrement 50
  Group name is "hsrp-Gi1.192-192" (default)
R20#show ip sla statistics 10

IPSLAs Latest Operation Statistics

IPSLA operation id: 10
        Latest RTT: 1 milliseconds
Latest operation start time: 01:18:37 UTC Thu Sep 25 2014 Latest operation return code: OK
Number of successes: 19
Number of failures: 0
Operation time to live: Forever
R20#show track 192

Track 192 IP SLA 10 state
 State is Up
    1 change, last change 01:03:08
  Latest operation return code: OK
  Latest RTT (millisecs) 1
```

https://t.me/learningnets

```
Tracked by:      HSRP GigabitEthernet1.192 192
```

We will shut down the Loopback0 on R18 to test failover. This will cause the track object state to go down and HSRP to decrement its priority.

```
R20#debug standby events terse

HSRP Events debugging is on
    (protocol, neighbor, redundancy, track, ha, arp, interface)

R18#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R18(config)#interface loopback0
R18(config-if)#shutdown
R18(config-if)#end
R18#
*Sep 25 01:59:42.348: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to down
*Sep 25 01:59:42.348: %LINK-5-CHANGED: Interface Loopback0, changed state to administratively down
R18#


R20#*Sep 25 01:59:57.433: %TRACK-6-STATE: 192 ip sla 10 state Up -> Down
R20#*Sep 25 01:59:57.434: HSRP: Gi1.192 Grp 192 Track 192 object changed, state Up -> Down
*Sep 25 01:59:57.434: HSRP: Gi1.192 Grp 192 Priority 120 -> 70
*Sep 25 01:59:57.585: HSRP: Gi1.192 Grp 192
Active: j/Coup rcvd from higher pri router (100/172.27.192.19)
*Sep 25 01:59:57.585: HSRP: Gi1.192 Grp 192 Active router is 172.27.192.19, was local
*Sep 25 01:59:57.585: HSRP: Gi1.192 Nbr 172.27.192.19 active for group 192
*Sep 25 01:59:57.585: HSRP: Gi1.192 Nbr 172.27.192.19 no longer standby for group 192 (Active)
*Sep 25 01:59:57.585: HSRP: Gi1.192 Grp 192 Active -> Speak
*Sep 25 01:59:57.585: %HSRP-5-STATECHANGE: GigabitEthernet1.192 Grp 192 state Active -> Speak
*Sep 25 01:59:57.586: HSRP: Peer not present
*Sep 25 01:59:57.586: HSRP: Gi1.192 Grp 192 Redundancy "hsrp-Gi1.192-192" state Active -> Speak
*Sep 25 01:59:57.586: HSRP: Gi1.192 Grp 192 Removed 172.27.192.254 from ARP
*Sep 25 01:59:57.589: HSRP: Peer not present
*Sep 25 01:59:57.589: HSRP: Gi1.192 IP Redundancy "hsrp-Gi1.192-192" update, Active -> Speak
*Sep 25 02:00:08.088: HSRP: Gi1.192 Grp 192 Speak: d/Standby timer expired (unknown)
*Sep 25 02:00:08.088: HSRP: Gi1.192 Grp 192 Standby router is local
*Sep 25 02:00:08.088: HSRP: Gi1.192 Grp 192 Speak -> Standby*Sep 25 02:00:08.088:
%HSRP-5-STATECHANGE: GigabitEthernet1.192 Grp 192 state Speak -> Standby
```

R20 should take the active role back after the track object state comes back up by no shutting the Loopback0 on R18. This verifies the additional requirements for the

https://t.me/learningnets

FHRP configuration on this portion of the network.

```
R18#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R18(config)#interface loopback0
R18(config-if)#no shutdown
R18(config-if)#end
R18#
*Sep 25 02:03:42.834: %SYS-5-CONFIG_I: Configured from console by console
*Sep 25 02:03:43.608: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up



R20# *Sep 25 02:03:52.448: %TRACK-6-STATE: 192 ip sla 10 state Down -> Up
 *Sep 25 02:03:52.448: HSRP: Gi1.192 Grp 192 Track 192 object changed, state Down -> Up
*Sep 25 02:03:52.449: HSRP: Gi1.192 Grp 192 Priority 70 -> 120
R20# *Sep 25 02:03:53.598: HSRP: Gi1.192 Grp 192
Standby: h/Hello rcvd from lower pri Active router (100/172.27.192.19)
*Sep 25 02:03:53.598: HSRP: Gi1.192 Grp 192 Active router is local, was 172.27.192.19
*Sep 25 02:03:53.598: HSRP: Gi1.192 Nbr 172.27.192.19 no longer active for group 192 (Standby)
*Sep 25 02:03:53.598: HSRP: Gi1.192 Nbr 172.27.192.19 Was active or standby - start passive holddown
*Sep 25 02:03:53.598: HSRP: Gi1.192 Grp 192 Standby router is unknown, was local
*Sep 25 02:03:53.598: HSRP: Gi1.192 Grp 192 Standby -> Active
*Sep 25 02:03:53.598: %HSRP-5-STATECHANGE: GigabitEthernet1.192 Grp 192 state Standby -> Active
*Sep 25 02:03:53.598: HSRP: Peer not present
*Sep 25 02:03:53.598: HSRP: Gi1.192 Grp 192 Redundancy "hsrp-Gi1.192-192" state Standby -> Active
*Sep 25 02:03:53.598: HSRP: Gi1.192 Grp 192 Added 172.27.192.254 to ARP (0000.0c9f.f0c0)
*Sep 25 02:03:53.601: HSRP: Gi1.192 IP Redundancy "hsrp-Gi1.192-192" standby, local -> unknown
*Sep 25 02:03:53.601: HSRP: Gi1.192 IP Redundancy "hsrp-Gi1.192-192" update, Standby -> Active
*Sep 25 02:03:56.596: HSRP: Gi1.192 IP Redundancy "hsrp-Gi1.192-192" update, Active -> Active
*Sep 25 02:04:05.009: HSRP: Gi1.192 Grp 192 Standby router is 172.27.192.19
*Sep 25 02:04:05.009: HSRP: Gi1.192 Nbr 172.27.192.19 is no longer passive
*Sep 25 02:04:05.009: HSRP: Gi1.192 Nbr 172.27.192.19 standby for group 192
```

Task 5.1

The goal of this task is to configure PIM in the Core network along with a protocol to advertise the RP. We are asked to configure redundancy in the PIM network by configuring R3 and R4 as a logical RP. This is referred to as "Anycast RP"; the other RP can seamlessly take over as soon as the IGP converges. Because they are sharing the same IP address, other PIM routers in the network will not know that the RP failed. They still see a route to the Loopback via the remaining RP.

To accomplish this, both RPs must "sync" their state. If there is a multicast source

https://t.me/learningnets

that is registered to R3, R3 must tell R4 about this registration so that clients that want to join this group can join the shared tree toward either R3 or R4. If R3 does not inform R4 about this active source, clients that have joined the shared tree on R4 will not be able build a shortest path tree toward the source and thus will not receive the multicast stream. This is, of course, because R4 does not know about the source if R3 does not inform it!

The requirements ask to configure R8 using a protocol that is part of the PIMv2 standard to disseminate RP information. This means that we must use BSR instead of Auto-RP. Although Auto-RP has been implemented by other vendors besides Cisco, it is still considered "proprietary." R8 will be configured as the BSR and R3 and R4 as the RP.

To filter BSR messages from leaking toward the internet, we will use the `ip pim bsr-border` command on R6. This will satisfy the requirement with only a single command.

Task 5.1 Solutions

```
R1:
ip multicast-routing distributed
!
interface GigabitEthernet1.12
 ip pim sparse-mode
interface GigabitEthernet1.13
 ip pim sparse-mode
interface GigabitEthernet1.14
 ip pim sparse-mode
interface GigabitEthernet1.17
 ip pim sparse-mode
interface GigabitEthernet1.18
 ip pim sparse-mode
interface GigabitEthernet1.116
 ip pim sparse-mode
interface GigabitEthernet1.117
 ip pim sparse-mode

R2:
ip multicast-routing distributed
!
interface GigabitEthernet1.12
 ip pim sparse-mode
interface GigabitEthernet1.23
 ip pim sparse-mode
```

https://t.me/learningnets

```
interface GigabitEthernet1.24
 ip pim sparse-mode
interface GigabitEthernet1.240
 ip pim sparse-mode
interface GigabitEthernet1.216
 ip pim sparse-mode
```

R3:

```
ip multicast-routing distributed
!
interface Loopback0
 ip pim sparse-mode
!
interface Loopback100
 ip address 192.122.3.100 255.255.255.255
 ip ospf 100 area 0
 ip pim sparse-mode
!
ip pim rp-candidate loopback 100
!
ip msdp peer 192.122.3.4 connect-source loopback 0
!
interface GigabitEthernet1.13
 ip pim sparse-mode
interface GigabitEthernet1.23
 ip pim sparse-mode
interface GigabitEthernet1.34
 ip pim sparse-mode
interface GigabitEthernet1.35
 ip pim sparse-mode
interface GigabitEthernet1.36
 ip pim sparse-mode
```

R4:

```
ip multicast-routing distributed
!
interface Loopback0
 ip pim sparse-mode
!
interface Loopback100
 ip address 192.122.3.100 255.255.255.255
 ip ospf 100 area 0
 ip pim sparse-mode
!
ip pim rp-candidate loopback 100
!
```

https://t.me/learningnets

```
ip msdp peer 192.122.3.3 connect-source loopback 0
!
interface GigabitEthernet1.14
 ip pim sparse-mode
interface GigabitEthernet1.24
 ip pim sparse-mode
interface GigabitEthernet1.34
 ip pim sparse-mode
interface GigabitEthernet1.45
 ip pim sparse-mode
interface GigabitEthernet1.46
 ip pim sparse-mode

R5:
ip multicast-routing distributed
!
interface GigabitEthernet1.35
 ip pim sparse-mode
interface GigabitEthernet1.45
 ip pim sparse-mode
interface GigabitEthernet1.56
 ip pim sparse-mode
interface GigabitEthernet1.58
 ip pim sparse-mode

R6:
ip multicast-routing distributed
!
interface GigabitEthernet1.36
 ip pim sparse-mode
interface GigabitEthernet1.46
 ip pim sparse-mode
interface GigabitEthernet1.56
 ip pim sparse-mode
interface GigabitEthernet1.60
 ip pim sparse-mode
 ip pim bsr-border

R7:
ip multicast-routing distributed
!
interface GigabitEthernet1.17
 ip pim sparse-mode
interface GigabitEthernet1.78
 ip pim sparse-mode
interface GigabitEthernet1.79
```

https://t.me/learningnets

```
 ip pim sparse-mode

R8:
ip multicast-routing distributed
!
interface GigabitEthernet1.18
 ip pim sparse-mode
interface GigabitEthernet1.58
 ip pim sparse-mode
interface GigabitEthernet1.78
 ip pim sparse-mode
interface GigabitEthernet1.89
 ip pim sparse-mode
!
interface Loopback0
 ip pim sparse-mode
!
ip pim bsr-candidate loopback 0
```

Task 5.1 Verification

To verify that our PIM adjacencies are up, we can look at the following output on the routers in the multicast domain.

```
R3#show ip pim interface


Address          Interface                Ver/   Nbr    Query  DR         DR
                                          Mode   Count  Intvl  Prior
10.0.13.2        GigabitEthernet1.13      v2/S   1      30     1          10.0.13.2
10.0.34.1        GigabitEthernet1.34      v2/S   1      30     1          10.0.34.2
10.0.35.1        GigabitEthernet1.35      v2/S   1      30     1          10.0.35.2
10.0.36.1        GigabitEthernet1.36      v2/S   1      30     1          10.0.36.2
192.122.3.100    Loopback100              v2/S   0      30     1          192.122.3.100
R3#
R3#show ip pim neighbor

PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable,
      L - DR Load-balancing Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode
10.0.13.1         GigabitEthernet1.13      00:22:39/00:01:42 v2    1 / S P G
```

https://t.me/learningnets

```
10.0.34.2         GigabitEthernet1.34      00:22:30/00:01:22 v2    1 / DR S P G
10.0.35.2         GigabitEthernet1.35      00:22:39/00:01:44 v2    1 / DR S P G
10.0.36.2         GigabitEthernet1.36      00:22:38/00:01:43 v2    1 / DR S P G
```

Both R3 and R4 advertised the new Loopback100 into OSPF. Let's verify that routers in the core are seeing both advertisements. From R2, we can see that R3 and R4 are advertising the Loopback into OSPF.

```
R2#show ip ospf database router 192.122.3.3 | sec 192.122.3.100
     (Link ID) Network/subnet number: 192.122.3.100
R2#show ip ospf database router 192.122.3.4 | sec 192.122.3.100

     (Link ID) Network/subnet number: 192.122.3.100
```

R5 in the RIP domain has the route with two next hops, R3 and R4.

```
R5#show ip route 192.122.3.100
Routing entry for 192.122.3.100/32
  Known via "rip", distance 120, metric 1
  Tag 3110
  Redistributing via rip, eigrp 56
  Advertised by eigrp 56 metric 1000000 100 255 1 1500
  Last update from 10.0.35.1 on GigabitEthernet1.35, 00:00:06 ago
  Routing Descriptor Blocks: 10.0.45.1, from 10.0.45.1, 00:00:12 ago, via GigabitEthernet1.45
      Route metric is 1, traffic share count is 1
      Route tag 4110   *10.0.35.1, from 10.0.35.1, 00:00:06 ago, via GigabitEthernet1.35

      Route metric is 1, traffic share count is 1
      Route tag 3110
```

R8 was configured as the BSR; all routers in the multicast network should have proper RP mappings pointing to 192.122.3.100, including R3 and R4.

```
R6#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4 RP 192.122.3.100 (?), v2
 Info source: 192.122.3.8 (?), via bootstrap
, priority 0, holdtime 150
         Uptime: 00:34:07, expires: 00:02:23

R3#show ip pim rp mapping
```

https://t.me/learningnets

```
PIM Group-to-RP Mappings This system is a candidate RP (v2)



Group(s) 224.0.0.0/4 RP 192.122.3.100 (?), v2
 Info source: 192.122.3.8 (?), via bootstrap
, priority 0, holdtime 150
         Uptime: 00:34:07, expires: 00:02:23

R4#show ip pim rp mapping

PIM Group-to-RP Mappings This system is a candidate RP (v2)

Group(s) 224.0.0.0/4 RP 192.122.3.100 (?), v2
 Info source: 192.122.3.8 (?), via bootstrap
, priority 0, holdtime 150
         Uptime: 00:34:07, expires: 00:02:23
```

Now verify that the MSDP session between R3 and R4 is operational.

https://t.me/learningnets

```
R3#show ip msdp peer
 MSDP Peer 192.122.3.4
 (?), AS ?
  Connection status: State: Up, Resets: 0, Connection source: Loopback0 (192.122.3.3)


    Uptime(Downtime): 00:33:19, Messages sent/received: 34/34
    Output messages discarded: 0
    Connection and counters cleared 00:34:19 ago
  SA Filtering:
    Input (S,G) filter: none, route-map: none
    Input RP filter: none, route-map: none
    Output (S,G) filter: none, route-map: none
    Output RP filter: none, route-map: none
  SA-Requests:
    Input filter: none
  Peer ttl threshold: 0
  SAs learned from this peer: 0
  Number of connection transitions to Established state: 1
    Input queue size: 0, Output queue size: 0
  MD5 signature protection on MSDP TCP connection: not enabled
  Message counters:
    RPF Failure count: 0
    SA Messages in/out: 0/0
    SA Requests in: 0
    SA Responses out: 0
    Data Packets in/out: 0/0
```

Note that the Loopback0 was used as the source for this session. We could have used the connected interface as well, but the task did not specify. If the task had asked for the MSDP session to be able to withstand a link failure, we would have had to use their Loopback0 for peerings.

There is currently no multicast state in the network; no sources or clients have done any signaling. Let's verify this by joining a group on R5 and sending traffic to it from R16.

https://t.me/learningnets

```
R5#conf t

Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#interface gig1.58
R5(config-subif)#ip igmp join-group 225.5.5.5
R5(config-subif)#end
R5#
```

This IGMP Membership Report on R5 is sent to the RP as a PIM Join. However, because R3 and R4 are running Anycast-RP, only one of them should have the state.

```
R3#show ip mroute 225.5.5.5
Group 225.5.5.5 not found
R4#show ip mroute 225.5.5.5

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 225.5.5.5), 00:03:41/00:02:46, RP 192.122.3.100, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.45, Forward/Sparse, 00:03:41/00:02:46
```

https://t.me/learningnets

This verifies that we can signal a shared tree to one of the RPs. Let's now send traffic to a different group from R16 and verify the registration process as well as MSDP. Note that the ping performed on R16 is an extended ping forcing the ping to exit out of a specific interface.

```
R16#ping
Protocol [ip]:  Target IP address:224.16.16.16
Repeat count [1]:
Datagram size [100]:
Timeout in seconds [2]: Extended commands [n]:y
Interface [All]:GigabitEthernet1.116
Time to live [255]: Source address or interface:89.211.116.16

Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.16.16.16, timeout is 2 seconds:
Packet sent with a source address of 89.211.116.16
.
```

This single multicast packet sent by R16 will be received encapsulated in a PIM Register by one of the RPs. The RP that receives it will install multicast state, and then will send an MSDP SA message to its MSDP peers.

R4 did not receive the register:

```
R4#show ip mroute 224.16.16.16Group 224.16.16.16 not found
```

R3 received the register and installed multicast state. Note that the "A" flag is set, which means that this state is "candidate for MSDP Advertisement."

```
R3#show ip mroute 224.16.16.16
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running,A - Candidate for MSDP Advertisement
,
```

https://t.me/learningnets

```
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.16.16.16), 00:00:26/stopped, RP 192.122.3.100, flags: SP
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null
 (89.211.116.16, 224.16.16.16), 00:00:26/00:02:33, flags:PA

  Incoming interface: GigabitEthernet1.34, RPF nbr 10.0.34.1
  Outgoing interface list: Null
```

R3 sends an MSDP SA to R4 for this source/group.

```
R4#show ip msdp sa-cache MSDP Source-Active Cache - 1 entries
(89.211.116.16, 224.16.16.16),RP 192.122.3.100, AS ?,00:00:36/00:05:59,Peer 192.122.3.4
```

R4 does not have any interested receivers for this group, so it has no need to install multicast state for this SA. However, if R4 had an interested receiver (joined on the shared tree for this group), R4 would have installed the state and forwarded the multicast packet to the receiver. The receiver at that point would join the source tree directly.

This verifies that our multicast control plane is working properly. In the next section we will have to verify that mulitcast packets flow end to end from R17 toward R6.

Task 5.2

Now we have to use the multicast network built in the previous task to allow R6 to receive traffic for group 226.10.6.6 sourced from R17.

Task 5.2 Solutions

https://t.me/learningnets

```
R1:
interface GigabitEthernet1.117
 ip pim sparse-mode

R6:
interface Loopback0
 ip pim sparse-mode
 ip igmp join-group 226.10.6.6
```

Task 5.2 Verification

Following verification strategy similar to the previous task, we will check that R6 joined the shared tree to one of the RPs.

In this case, R6 joined the shared tree to R3 instead of R4.

```
R3#show ip mroute 226.10.6.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
 (*, 226.10.6.6
), 00:00:11/00:03:19, RP 192.122.3.100, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.36, Forward/Sparse, 00:00:11/00:03:19

R4#show ip mroute 226.10.6.6

Group 226.10.6.6 not found
```

https://t.me/learningnets

Now test the multicast dataplane by sending traffic to this group from R17. Note that R17 is not running PIM or multicast-routing and is acting as a host on the network. R1 must be running PIM on its interface to R17 and will assume the role of the DR for the segment. R1 will receive the multicast stream on this PIM-enabled interface and will send a PIM Register to the Anycast RP.

Verify that R1 has assumed the role of DR for this interface.

```
R1#show ip pim interface


Address          Interface                Ver/   Nbr    Query  DR         DR
                                          Mode   Count  Intvl  Prior
10.0.12.1        GigabitEthernet1.12      v2/S   1      30     1          10.0.12.2
10.0.13.1        GigabitEthernet1.13      v2/S   1      30     1          10.0.13.2
136.5.17.1       GigabitEthernet1.17      v2/S   1      30     1          136.5.17.7
136.5.18.1       GigabitEthernet1.18      v2/S   1      30     1          136.5.18.8
89.211.116.1     GigabitEthernet1.116     v2/S   1      30     1          89.211.116.16
89.211.117.1 GigabitEthernet1.117     v2/S   0      30     1 89.211.117.1
```

If we do not enable PIM on this interface, R1 will not be able to process the multicast stream and engage the multicast code to install multicast state and send the PIM Register.

```
R17#ping
Protocol [ip]: Target IP address: 226.10.6.6
               Repeat count [1]: 50
Datagram size [100]:
Timeout in seconds [2]: Extended commands [n]: y
Interface [All]: GigabitEthernet1.117

Time to live [255]: Source address or interface: 89.211.117.17

Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 50, 100-byte ICMP Echos to 226.10.6.6, timeout is 2 seconds:
Packet sent with a source address of 89.211.117.17

Reply to request 0 from 10.0.36.2, 39 ms
```

https://t.me/learningnets

```
Reply to request 0 from 192.122.3.6, 39 ms
Reply to request 1 from 192.122.3.6, 6 ms
Reply to request 2 from 192.122.3.6, 2 ms
Reply to request 3 from 192.122.3.6, 2 ms
Reply to request 4 from 192.122.3.6, 3 ms
```

Note that R3 received the Register instead of R4. For this reason, R4 did not create multicast state. R3 was the root shared tree and also received the Registe; MSDP was not involved in setting up this tree. R4 could have also received the Register, but because of the routing in the network, R1 prefers to route toward R3 to get to 192.122.3.100.

https://t.me/learningnets

```
R3#show ip mroute 226.10.6.6

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 226.10.6.6), 00:11:52/00:03:24, RP 192.122.3.100, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.36, Forward/Sparse, 00:11:52/00:03:24
 (89.211.117.17, 226.10.6.6
), 00:00:30/00:03:03, flags: TA
  Incoming interface: GigabitEthernet1.13, RPF nbr 10.0.13.1
  Outgoing interface list:
    GigabitEthernet1.36, Forward/Sparse, 00:00:30/00:03:24


R4#show ip mroute 226.10.6.6

Group 226.10.6.6 not found
```

R4 received the MSDP SA from R3 and can service any 226.10.6.6 sources in the network.

https://t.me/learningnets

```
R4#show ip msdp sa-cache

MSDP Source-Active Cache - 1 entries (89.211.117.17, 226.10.6.6), RP 192.122.3.100
, BGP/AS 0, 00:04:27/00:03:53, Peer 192.122.3.3
```

To verify that we can use either one of the RP routers, let's source traffic to 226.10.6.6 from R2. R2 is routing toward R4 to reach 192.122.3.3.

```
R2#show ip cef 192.122.3.3
192.122.3.3/32
  nexthop 10.0.24.2 GigabitEthernet1.24
R2#ping
Protocol [ip]: Target IP address: 226.10.6.6
Repeat count [1]: 50
Datagram size [100]:
Timeout in seconds [2]: Extended commands [n]: y
Interface [All]: GigabitEthernet1.24
Time to live [255]: Source address or interface: 10.0.24.1

Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 50, 100-byte ICMP Echos to 226.10.6.6, timeout is 2 seconds:
Packet sent with a source address of 10.0.24.1
.
Reply to request 1 from 10.0.36.2, 3 ms
Reply to request 1 from 192.122.3.6, 3 ms
Reply to request 2 from 192.122.3.6, 3 ms
Reply to request 3 from 192.122.3.6, 2 ms
```

R4 receives the PIM Register and sends an MSDP SA to R3. Because R3 is the root of the shared tree, it sends the multicast packets to R6. When R6 receives the multicast down the shared tree, it leaves the shared tree and joins the source tree directly to R2.

```
R4#show ip mroute 226.10.6.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
```

https://t.me/learningnets

```
        L - Local, P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set
, J - Join SPT, M - MSDP created entry, E - Extranet,       X - Proxy Join Timer Running,
A - Candidate for MSDP Advertisement
,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 226.10.6.6), 00:00:43/stopped, RP 192.122.3.100, flags: SP
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null
 (10.0.24.1, 226.10.6.6), 00:00:43/00:02:16, flags: TA

  Incoming interface: GigabitEthernet1.24, RPF nbr 10.0.24.1
  Outgoing interface list:
    GigabitEthernet1.34, Forward/Sparse, 00:00:43/00:02:46
```

R3 creates MSDP state for the SA it received.

```
R3#show ip msdp sa-cache

MSDP Source-Active Cache - 1 entries(10.0.24.1, 226.10.6.6),RP 192.122.3.100
, AS ?,00:02:50/00:05:52, Peer192.122.3.4
```

It then creates multicast state with the 'M' bit set, an MSDP created entry.

```
R3#show ip mroute 226.10.6.6
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
        L - Local, P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT,
M - MSDP created entry
, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
```

https://t.me/learningnets

```
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 226.10.6.6), 00:07:49/stopped, RP 192.122.3.100, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.36, Forward/Sparse, 00:07:49/00:02:33
(10.0.24.1, 226.10.6.6, 00:00:10/00:02:48, flags:MT

  Incoming interface: GigabitEthernet1.34, RPF nbr 10.0.34.2
  Outgoing interface list:
    GigabitEthernet1.36, Forward/Sparse, 00:00:10/00:03:20
```

Task 6.1

It is now time to enable IPv6 for Site A. The task requires us to enable IPv6 IGP protocols matching the IPv4 design of Site A. For EIGRPv6, we are asked to use the same instance and autonomous system (named mode). Note that in the IPv4 IGP task dealing with Site A, we were asked to modify the "rib-scale" factor. This was the only clue for using named mode in that section. If we did not have that clue and configured legacy EIGRP, we would have had to go back and change all of the EIGRP configuration! It is crucial that you read the entire task and notice dependencies like this one. We would only be able to run IPv4 and IPv6 EIGRP in the same instance by using named mode (Multi-AF Mode).

There are two ways to configure OSPFv3 on Cisco IOS. The older method has the syntax of "ipv6 router ospf " and "ipv6 ospf". This form of OSPFv3 is for IPv6 only. Cisco IOS has a newer implementation of OSPFv3 known as "Multi-AF OSPFv3." Multi-AF OSPFv3 can carry both IPv4 and IPv6 NLRIs in the database. The SPF tree is built using the standard Dijkstra algorithm, but nodes can advertise their links containing IPv4 and IPv6 NLRIs using Type-9 LSAs. The Type-9 LSAs reference the Type-1 and Type-2 LSAs used to build the SPT.

After configuring both of these IGPs independently, we need to perform redistribution to advertise the routes between the two domains. This task does not require for routing between the IGP domains in any particular way; it just asks for

https://t.me/learningnets

reachability, so we can do the redistribution at one of the ASBRs instead of both.

Note that EIGRPv6 automatically adds all IPv6-enabled interfaces to the process. This excludes interfaces that are members of other routing tables (in a VRF instead of the global table). To remove a link from the process, we must add the `shutdown` command under the EIGRP address-family IPv6 instance. This is needed on R11 and R12 so that their links into the OSPFv3 domain are not natively advertised into EIGRP.

Task 6.1 Solutions

```
R10:
router eigrp INE_CCIE
 !
 address-family ipv6 unicast autonomous-system 789
  !
  topology base
  exit-af-topology
 exit-address-family

R11:
router eigrp INE_CCIE
 !
 address-family ipv6 unicast autonomous-system 789
  !
  af-interface GigabitEthernet1.113
   shutdown
  exit-af-interface
  !
  topology base
   redistribute ospf 5 metric 1000000 100 255 1 1500
  exit-af-topology
 exit-address-family
!
router ospfv3 5
 router-id 192.122.3.11
 !
 address-family ipv6 unicast
  redistribute eigrp 789 include-connected
 exit-address-family
!
interface GigabitEthernet1.113
 ospfv3 5 ipv6 area 5
```

https://t.me/learningnets

```
R12:
router eigrp INE_CCIE
 !
 address-family ipv6 unicast autonomous-system 789
  !
  af-interface GigabitEthernet1.124
   shutdown
  exit-af-interface
  !
  topology base
  exit-af-topology
 exit-address-family
!
router ospfv3 5
 router-id 192.122.3.12
!
interface GigabitEthernet1.124
 ospfv3 5 ipv6 area 5

R13:
router ospfv3 5
 router-id 192.122.3.13
!
interface Loopback0
 ospfv3 5 ipv6 area 5
!
interface GigabitEthernet1.134
 ospfv3 5 ipv6 area 5
!
interface GigabitEthernet1.113
 ospfv3 5 ipv6 area 5

R14:
router ospfv3 5
 router-id 192.122.3.14
!
interface Loopback0
 ospfv3 5 ipv6 area 5
!
interface GigabitEthernet1.124
 ospfv3 5 ipv6 area 5
!
interface GigabitEthernet1.134
 ospfv3 5 ipv6 area 5

R15:
```

https://t.me/learningnets

```
router ospfv3 5
 router-id 192.122.3.15
!
interface Loopback0
 ospfv3 5 ipv6 area 5
!
interface GigabitEthernet1.134
 ospfv3 5 ipv6 area 5
```

Task 6.1 Verification

We can verify this task by looking at the adjacencies and active IGP interfaces on the devices. Note the newer "Multi-AF" syntax that is used for both of the protocols.

```
R15#show ospfv3 neighbor

          OSPFv3 5 address-family ipv6 (router-id 192.122.3.15)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
192.122.3.13      1   FULL/DROTHER    00:00:38    11              GigabitEthernet1.134
192.122.3.14      1   FULL/BDR        00:00:34    11              GigabitEthernet1.134

R10#show eigrp address-family ipv6 neighbors

EIGRP-IPv6 VR(INE_CCIE) Address-Family Neighbors for AS(789)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   Link-local address:     Gi1.102                  10 00:18:21    2   100  0  7
    FE80::250:56FF:FE8D:1455
0   Link-local address:     Gi1.102                  13 00:18:40    1   100  0  6
    FE80::250:56FF:FE8D:501B
```

R11 and R12 should not include their interfaces facing the OSPFv3 domain in EIGRP. These interfaces must be redistributed in, accomplished by the `include-connected` command in redistribution. IPv6 IGP protocols do not include connected interfaces when doing redistribution by default, unlike IPv4 IGPs.

```
R11#show ospfv3 ipv6 interface  brief
Interface    PID   Area            AF         Cost  State Nbrs F/C
Gi1.113      5     5               ipv6       1     BDR   1/1


R11#show eigrp address-family ipv6 interfaces
```

https://t.me/learningnets

```
EIGRP-IPv6 VR(INE_CCIE) Address-Family Interfaces for AS(789)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi1.102                  2        0/0       0/0         513       0/0         2556           0
Lo0                      0        0/0       0/0           0       0/0            0           0
```

Next check the routing tables, topology/database entries, and reachability.

```
R10#show ipv6 route eigrp

IPv6 Routing Table - default - 14 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid
       a - Application
D   ::192:122:3:11/128 [90/10880]
     via FE80::250:56FF:FE8D:501B, GigabitEthernet1.102
D   ::192:122:3:12/128 [90/10880]
     via FE80::250:56FF:FE8D:1455, GigabitEthernet1.102
EX  ::192:122:3:13/128 [170/522240]
     via FE80::250:56FF:FE8D:501B, GigabitEthernet1.102
EX  ::192:122:3:14/128 [170/522240]
     via FE80::250:56FF:FE8D:501B, GigabitEthernet1.102
EX  ::192:122:3:15/128 [170/522240]
     via FE80::250:56FF:FE8D:501B, GigabitEthernet1.102
EX  2001:172:19:113::/64 [170/522240]
     via FE80::250:56FF:FE8D:501B, GigabitEthernet1.102
EX  2001:172:19:124::/64 [170/522240]
     via FE80::250:56FF:FE8D:501B, GigabitEthernet1.102
EX  2001:172:19:134::/64 [170/522240]
     via FE80::250:56FF:FE8D:501B, GigabitEthernet1.102
R15#show ipv6 route ospf

IPv6 Routing Table - default - 13 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid
```

https://t.me/learningnets

```
        a - Application
OE2 ::192:122:3:10/128 [110/20]
     via FE80::250:56FF:FE8D:683D, GigabitEthernet1.134
OE2 ::192:122:3:11/128 [110/20]
     via FE80::250:56FF:FE8D:683D, GigabitEthernet1.134
OE2 ::192:122:3:12/128 [110/20]
     via FE80::250:56FF:FE8D:683D, GigabitEthernet1.134
O   ::192:122:3:13/128 [110/1]
     via FE80::250:56FF:FE8D:683D, GigabitEthernet1.134
O   ::192:122:3:14/128 [110/1]
     via FE80::250:56FF:FE8D:4671, GigabitEthernet1.134
OE2 2001:172:19:102::/64 [110/20]
     via FE80::250:56FF:FE8D:683D, GigabitEthernet1.134
O   2001:172:19:113::/64 [110/2]
     via FE80::250:56FF:FE8D:683D, GigabitEthernet1.134
O   2001:172:19:124::/64 [110/2]
     via FE80::250:56FF:FE8D:4671, GigabitEthernet1.134
OE2 2001:202:4:210::/64 [110/20]
     via FE80::250:56FF:FE8D:683D, GigabitEthernet1.134
```

This is a Type-5 LSA; notice that it is very similar to OSPFv2. R11 is the advertising router because this is our point of redistribution.

```
R15#show ospfv3 database external ::192:122:3:10/128


          OSPFv3 5 address-family ipv6 (router-id 192.122.3.15)


                Type-5 AS External Link States


  LS age: 1150  LS Type: AS External Link
  Link State ID: 0 Advertising Router: 192.122.3.11
  LS Seq Number: 80000001
  Checksum: 0x540D
  Length: 44 Prefix Address: ::192:122:3:10
 Prefix Length: 128
, Options: None Metric Type: 2
 (Larger than any link state path) Metric: 20
```

The EIGRPv6 topology table is also very similar to the IPv6 counterpart.

```
R10#show eigrp address-family ipv6 topology ::192:122:3:15/128
EIGRP-IPv6 VR(INE_CCIE) Topology Entry for AS(789)/ID(192.122.3.10) for ::192:122:3:15/128
```

https://t.me/learningnets

```
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 66846720, RIB is 522240

  Descriptor Blocks:
  FE80::250:56FF:FE8D:501B (GigabitEthernet1.102), from FE80::250:56FF:FE8D:501B, Send flag is 0x0
      Composite metric is (66846720/66191360), route is External
      Vector metric:
        Minimum bandwidth is 1000000 Kbit
        Total delay is 1010000000 picoseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
      External data: Originating router is 192.122.3.11

        AS number of route is 5 External protocol is OSPF
, external metric is 2
        Administrator tag is 0 (0x00000000)
```

To test reachability, we will use our ping-script.

```
R15#tclsh
 R15(tcl)#proc ping-script {} {
+>foreach i {
+>::192:122:3:10
+>::192:122:3:11
+>::192:122:3:12
+>::192:122:3:13
+>::192:122:3:14
+>::192:122:3:15
+>} { ping $i source lo0 }
+>}R15(tcl)#ping-script

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:10, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:11, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 18/18/19 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:12, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:15
```

https://t.me/learningnets

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/9/10 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:13, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/8/12 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:14, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/6 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:15, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:15
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R15(tcl)#
```

Task 6.2

Continuing with IPv6 IGPs, we move to the core network and configure RIPng according to the requirements. The task asks you to configure RIPng between R3, R4, R5, and R6, and advertise the Loopback0 networks of all of these devices into the RIPng process.

Task 6.2 Solutions

```
R3:
interface Loopback0
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.34
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.35
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.36
 ipv6 rip CCIE enable

R4:
```

https://t.me/learningnets

```
interface Loopback0
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.34
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.45
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.46
 ipv6 rip CCIE enable

R5:
interface Loopback0
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.35
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.45
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.56
 ipv6 rip CCIE enable

R6:
interface Loopback0
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.36
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.46
 ipv6 rip CCIE enable
!
interface GigabitEthernet1.56
 ipv6 rip CCIE enable
```

Task 6.2 Verification

The verification here is quite simple. Look at the RIB and check for reachability between the loopbacks with a ping script.

https://t.me/learningnets

```
R6#show ipv6 route rip

IPv6 Routing Table - default - 16 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid
       a - Application
R   ::192:122:3:3/128 [120/2]
     via FE80::250:56FF:FE8D:632A, GigabitEthernet1.36
R   ::192:122:3:4/128 [120/2]
     via FE80::250:56FF:FE8D:113D, GigabitEthernet1.46
R   ::192:122:3:5/128 [120/2]
     via FE80::250:56FF:FE8D:74AA, GigabitEthernet1.56
R   2001:10:0:34::/126 [120/2]
     via FE80::250:56FF:FE8D:632A, GigabitEthernet1.36
     via FE80::250:56FF:FE8D:113D, GigabitEthernet1.46
R   2001:10:0:35::/126 [120/2]
     via FE80::250:56FF:FE8D:632A, GigabitEthernet1.36
     via FE80::250:56FF:FE8D:74AA, GigabitEthernet1.56
R   2001:10:0:45::/126 [120/2]
     via FE80::250:56FF:FE8D:113D, GigabitEthernet1.46
     via FE80::250:56FF:FE8D:74AA, GigabitEthernet1.56
R3#tclsh
 R3(tcl)#proc ping-script {} {
+>(tcl)#foreach i {
+>(tcl)#::192:122:3:3
+>(tcl)#::192:122:3:4
+>(tcl)#::192:122:3:5
+>(tcl)#::192:122:3:6
+>(tcl)#} { ping $i source lo0 }
+>(tcl)#}R3(tcl)#ping-script
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:3, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:4, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/8 ms
Type escape sequence to abort.
```

https://t.me/learningnets

```
Sending 5, 100-byte ICMP Echos to ::192:122:3:5, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to ::192:122:3:6, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/8/10 ms
R3(tcl)#
R4#show ipv6 rip database

RIP process "CCIE", local RIB
 ::192:122:3:3/128, metric 2, installed
     GigabitEthernet1.34/FE80::250:56FF:FE8D:632A, expires in 168 secs
 ::192:122:3:5/128, metric 2, installed
     GigabitEthernet1.45/FE80::250:56FF:FE8D:74AA, expires in 157 secs
 ::192:122:3:6/128, metric 2, installed
     GigabitEthernet1.46/FE80::250:56FF:FE8D:6442, expires in 178 secs
 2001:10:0:34::/126, metric 2
     GigabitEthernet1.34/FE80::250:56FF:FE8D:632A, expires in 168 secs
 2001:10:0:35::/126, metric 2, installed
     GigabitEthernet1.45/FE80::250:56FF:FE8D:74AA, expires in 157 secs
     GigabitEthernet1.34/FE80::250:56FF:FE8D:632A, expires in 168 secs
 2001:10:0:36::/126, metric 2, installed
     GigabitEthernet1.46/FE80::250:56FF:FE8D:6442, expires in 178 secs
     GigabitEthernet1.34/FE80::250:56FF:FE8D:632A, expires in 168 secs
 2001:10:0:45::/126, metric 2
     GigabitEthernet1.45/FE80::250:56FF:FE8D:74AA, expires in 157 secs
 2001:10:0:46::/126, metric 2
     GigabitEthernet1.46/FE80::250:56FF:FE8D:6442, expires in 178 secs
 2001:10:0:56::/126, metric 2, installed
     GigabitEthernet1.46/FE80::250:56FF:FE8D:6442, expires in 178 secs
     GigabitEthernet1.45/FE80::250:56FF:FE8D:74AA, expires in 157 secs
```

Task 6.3

The final IPv6 task asks us to configure BGP between the Core network's RIPng domain and the Internet. The ISP Edge router has been pre-configured for this peering but is expecting the eBGP session to come from AS 65600 instead of 65006. The same tools available for IPv4 BGP are also available for IPv6 BGP (MP-BGP); we can use the local-as feature to accomplish this task. Additionally, we are asked to ensure that AS 65006 does not appear on any of the updates sent or

https://t.me/learningnets

received to and from the ISP. The `no-prepend replace-as` feature set will be used to accomplish this. Note that when using the `local as` feature, the AS_PATH for routes sent to the peer will show up as REAL_AS LOCAL_AS. With the `no-prepend` feature, the LOCAL_AS is not prepended to routes RECEIVED from the peer. Routes sent to the eBGP peer still show REAL_AS LOCAL_AS. With the `replace-as` feature, the REAL_AS is completely removed on routes sent to and received from the eBGP peer.

We are also asked to establish a new BGP session for IPv6. It is possible to advertise IPv6 NLRIs over an IPv4 session, but this causes many interesting next-hop issues. The route lookup process fails when it sees an IPv6 route with an IPv4 next hop, so route-maps must be used to change the next-hop to an IPv6 address. In newer version of code, there is a command that will take care of doing this conversion automatically: `bgp default ipv6-nexthop .`

The core network must be configured to run iBGP with R6 as the RR. All IPv6 BGP routes received from the Internet are passed on to the iBGP peers so that they also have reachability. Because of the restriction that we cannot advertise R6's IPv6 link connecting to the internet into RIPng or BGP, we must set "next-hop-self" on R6's iBGP peerings. The task also asks us to advertise the Loopback0 networks of routers R3, R4, R5, and R6 into BGP. We can use R6 as our central advertisement point. The goal of advertising these routes to the ISP is to give the iBGP routers reachability to the Internet destinations.

Task 6.3 Solutions

```
R3:
router bgp 65006
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor ::192:122:3:6 remote-as 65006
 neighbor ::192:122:3:6 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
  neighbor ::192:122:3:6 activate
 exit-address-family

R4:
router bgp 65006
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
```

https://t.me/learningnets

```
 neighbor ::192:122:3:6 remote-as 65006
 neighbor ::192:122:3:6 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
  neighbor ::192:122:3:6 activate
 exit-address-family
```

R5:

```
router bgp 65006
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor ::192:122:3:6 remote-as 65006
 neighbor ::192:122:3:6 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
  neighbor ::192:122:3:6 activate
 exit-address-family
```

R6:

```
router bgp 65006
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor IPv6_BGP peer-group
 neighbor IPv6_BGP remote-as 65006
 neighbor IPv6_BGP update-source Loopback0
 neighbor ::192:122:3:3 peer-group IPv6_BGP
 neighbor ::192:122:3:4 peer-group IPv6_BGP
 neighbor ::192:122:3:5 peer-group IPv6_BGP
 neighbor 2001:202:4:60::1 remote-as 65123
 neighbor 2001:202:4:60::1 local-as 65600 no-prepend replace-as
 !
 address-family ipv6
  network ::192:122:3:3/128
  network ::192:122:3:4/128
  network ::192:122:3:5/128
  network ::192:122:3:6/128
  neighbor IPv6_BGP route-reflector-client
  neighbor IPv6_BGP next-hop-self
  neighbor ::192:122:3:3 activate
  neighbor ::192:122:3:4 activate
```

https://t.me/learningnets

```
  neighbor ::192:122:3:5 activate
  neighbor 2001:202:4:60::1 activate
 exit-address-family
```

Task 6.3 Verification

Let's check our BGP peerings on R6.

```
R6#show bgp ipv6 unicast summary

BGP router identifier 192.122.3.6, local AS number 65006
BGP table version is 168, main routing table version 168
9 network entries using 2448 bytes of memory
9 path entries using 1296 bytes of memory
3/3 BGP path/bestpath attribute entries using 744 bytes of memory
2 BGP AS-PATH entries using 64 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 4552 total bytes of memory
BGP activity 20/8 prefixes, 36/24 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
::192:122:3:3   4        65006      18      27      168    0    0 00:12:51        0
::192:122:3:4   4        65006      18      28      168    0    0 00:12:42        0
::192:122:3:5   4        65006      17      29      168    0    0 00:12:34        0
2001:202:4:60::1
                4        65123       8       9      168    0    0 00:03:02        5
```

The iBGP peers should be receiving the five routes advertised from the ISP. Note that the next-hop for all of these routes shows up as R6's loopback. If we did not change the next-hop to self on R6, we would not have reachability for these routes. In fact, the iBGP peers would not be able to install them in the RIB/FIB, so they would not appear as ">" in the BGP table. Routes received from an eBGP peer and passed onto iBGP peers do not get their next-hops changed automatically, so the next hop for all of these routes would be the IPv6 address of the ISP router. R6 is not allowed to advertise this route into the iBGP or RIPng domain; route recursion would fail.

```
R4#show bgp ipv6 unicast regexp ^65123

BGP table version is 35, local router ID is 192.122.3.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
```

https://t.me/learningnets

```
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 2004:4:2:2::1/128
                       ::192:122:3:6            0    100      0 65123 ?
 *>i 2004:144:4:4::100/128
                       ::192:122:3:6            0    100      0 65123 ?
 *>i 2004:180::/32    ::192:122:3:6            0    100      0 65123 ?
 *>i 2006::/16        ::192:122:3:6            0    100      0 65123 ?
 *>i 2008:8:8:8::8/128
                       ::192:122:3:6            0    100      0 65123 ?
```

Now check that the routes R6 is advertising to the ISP do not have the REAL_AS anywhere in the AS_PATH. The ISP router is set up as a VRF on R15. We can log in to R15 and check the BGP table there. Note that in other labs you may not have access to these "external devices."

```
R15#show bgp vpnv6 unicast vrf internet

BGP table version is 181, local router ID is 192.122.3.15
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 65123:65123 (default for vrf internet)
 *>  ::192:122:3:3/128                       2001:202:4:60::          2             0 65600
 i
 *>  ::192:122:3:4/128                       2001:202:4:60::          2             0 65600
 i
 *>  ::192:122:3:5/128                       2001:202:4:60::          2             0 65600
 i
 *>  ::192:122:3:6/128                       2001:202:4:60::          0             0 65600
 i
 *>  2004:4:2:2::1/128
                       ::                        0         32768 ?
 *>  2004:144:4:4::100/128
                       ::                        0         32768 ?
 *>  2004:180::/32    ::                        0         32768 ?
 *>  2006::/16        ::                        0         32768 ?
 *>  2008:8:8:8::8/128
```

https://t.me/learningnets

```
                        ::                        0         32768 ?
```

If we remove the `replace-as` keyword on R6, the ISP router will see REAL_AS LOCAL_AS.

```
R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R6(config)#router bgp 65006
R6(config-router)#neighbor 2001:202:4:60::1 local-as 65600 no-prepend
R6(config-router)#
*Sep 26 23:01:59.377: %BGP-5-NBR_RESET: Neighbor 2001:202:4:60::1 reset (Local AS change)
*Sep 26 23:01:59.378: %BGP-5-ADJCHANGE: neighbor 2001:202:4:60::1 Down Local AS change
*Sep 26 23:01:59.378: %BGP_SESSION-5-ADJCHANGE: neighbor 2001:202:4:60::1 IPv6 Unicast topology base removed from se
*Sep 26 23:02:00.189: %BGP-5-ADJCHANGE: neighbor 2001:202:4:60::1 Up R6(config-router)#end
R15#show bgp vpnv6 unicast vrf internet

BGP table version is 189, local router ID is 192.122.3.15
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 65123:65123 (default for vrf internet)
 *>  ::192:122:3:3/128                       2001:202:4:60::          2             0 65600 65006
 i
 *>  ::192:122:3:4/128
                       2001:202:4:60::          2             0 65600 65006
 i
 *>  ::192:122:3:5/128
                       2001:202:4:60::          2             0 65600 65006
 i
 *>  ::192:122:3:6/128
                       2001:202:4:60::          0             0 65600 65006
 i
 *>  2004:4:2:2::1/128
                       ::                       0         32768 ?
 *>  2004:144:4:4::100/128
                       ::                       0         32768 ?
 *>  2004:180::/32    ::                       0         32768 ?
 *>  2006::/16        ::                       0         32768 ?
 *>  2008:8:8:8::8/128
                       ::                       0         32768 ?
R6(config)#router bgp 65006

R6(config-router)#neighbor 2001:202:4:60::1 local-as 65600 no-prepend replace-as
```

https://t.me/learningnets

```
*Sep 26 23:03:37.740: %BGP-5-NBR_RESET: Neighbor 2001:202:4:60::1 reset (Local AS change)
*Sep 26 23:03:37.741: %BGP-5-ADJCHANGE: neighbor 2001:202:4:60::1 Down Local AS change
*Sep 26 23:03:37.741: %BGP_SESSION-5-ADJCHANGE: neighbor 2001:202:4:60::1 IPv6 Unicast topology base removed from se
*Sep 26 23:03:38.508: %BGP-5-ADJCHANGE: neighbor 2001:202:4:60::1 Up R6(config-router)#end

R6#
```

Now we need to test reachability to the IPv6 Internet destinations from the iBGP routers.

```
R3#ping 2004:144:4:4::100 source loopback 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2004:144:4:4::100, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/21 ms

R4#ping  2004:4:2:2::1 source loopback 0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2004:4:2:2::1, timeout is 2 seconds:
Packet sent with a source address of ::192:122:3:4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/19 ms
```

Task 7.1

All of the routing tasks up to this point are completed. It is now time to add some optimizations to the protocols following the requirements. This task asks us to configure the OSPF domain in the Core network so that second routes that have a second-best path get installed in the FIB. This is referring to a newer feature in OSPF called LFA, Loop Free Alternates. EIGRP has been able to pre-calculate the "next best path" since the very beginning of the protocol. EIGRP calls it the "Feasible Successor." Unlike EIGRP, each OSPF node knows about the entire database - all of the links and adjacencies within a single area.

The SPF algorithm was modified so that after a node runs SPF for the first time calculates the SPT, a second SPF run is done, called rSPF, from the point of view of the node's directly connected neighbors. This is possible in OSPF because each node knows about the entire database. After running rSPF, the node would be able to algorithmically calculate if any of the nodes neighbors are loop free for a particular prefix. LFA takes the results of this calculation and installs them into the

https://t.me/learningnets

RIB/FIB.

In the event of a failure in the best path, the OSPF node can immediately begin using the backup path that was installed in the RIB/FIB without running any sort of calculation.

LFA can be enabled per area, as well as per prefix priority. For this lab, we will enable LFA for all areas and for prefix priority low. This ensures that external routes are protected. /32 host routes are considered high priority prefixes, so if LFA is set to protect "high" prefixes, only /32 host routes will be protected and nothing else.

Note that the requirement of this section did not say that all external routes MUST be protected, only external routes that have backup paths. There are cases in which LFA is not able to protect a prefix—if, for example, the node only has one neighbor, or the rLFA calculation determined that there was no Loop Free Alternate.

Task 7.1 Solutions

```
R1:
router ospf 100
 fast-reroute per-prefix enable prefix-priority low

R2:
router ospf 100
 fast-reroute per-prefix enable prefix-priority low

R3:
router ospf 100
 fast-reroute per-prefix enable prefix-priority low

R4:
router ospf 100
 fast-reroute per-prefix enable prefix-priority low

R16:
router ospf 100
 fast-reroute per-prefix enable prefix-priority low
```

Task 7.1 Verification

Look at prefix 10.0.12.0/30 on R4 for verification.

```
R4#show ip route 10.0.12.0
```

https://t.me/learningnets

```
Routing entry for 10.0.12.0/30
  Known via "ospf 100", distance 110, metric 2, type inter area
  Redistributing via rip
  Advertised by rip metric 1 route-map OSPF_INTO_RIP_REDISTRIBUTION
  Last update from 10.0.24.1 on GigabitEthernet1.24, 00:12:00 ago
  Routing Descriptor Blocks:
  * 10.0.24.1, from 192.122.3.2, 00:12:00 ago, via GigabitEthernet1.24
      Route metric is 2, traffic share count is 1 Repair Path: 10.0.34.1, via GigabitEthernet1.34
```

Recall that the links between R1-R4 and R2-R3 are redistributed as connected routes and thus no adjacency forms here. If these links were running as normal links in the area, R4 would have two ECMP paths toward 10.0.12.0.

Without LFA, R4 would only have a single path towars 10.0.12.0—through R2. However, LFA computed that R3 was a loop-free path and pre-installed this into the FIB. If the link between R2-R4 fails, R4 will immediately begin sending traffic toward 10.0.12.0 toward R3. It is guaranteed algorithmically that R3 will not loop the packets back to R4.

Under normal conditions, R3 routes toward R1 to get to 10.0.12.0. However, R3 is also using R4 as an LFA. Note that LFA protects against a single link failure. If both the R2-R4 link and the R1-R3 link were to fail at the exact same time, there would be a microloop between R3 and R4.

```
R3#show ip cef 10.0.12.0
10.0.12.0/30 nexthop 10.0.13.1 GigabitEthernet1.13
 label [implicit-null|none]
    repair: attached-nexthop 10.0.34.2 GigabitEthernet1.34
```

We can check the percentages of prefixes that are protected by LFA on any of the nodes in the OSPF domain with LFA enabled. This output is shown per area; R16 is only running on area 121.

```
R16#show ip ospf fast-reroute prefix-summary


            OSPF Router with ID (192.122.3.16) (Process ID 100)
                    Base Topology (MTID 0)


Area 121:


Interface        Protected    Primary paths    Protected paths Percent protected
```

https://t.me/learningnets

```
                                 All  High   Low   All  High   Low    All High  Low
Gi1.216                 Yes      8     3     5     5     2     3    62%  66%  60%
Gi1.116                 Yes     16     7     9     6     3     3    37%  42%  33%

Area total:                     24    10    14    11     5     6    45%  50%  42%

Process total:                  24    10    14    11     5     6    45%  50%  42%
```

Task 7.2

In this task we are asked to set up a routing policy on PE devices R7 and R8. This type of routing policy is outlined in RFC 1998, and most Service Providers have such implementations, allowing customers to signal to the provider via BGP communities which routing policy should be applied to the routes.

The communities will be utilized to signal inbound routing preference (via local-preference), or to stop advertising a route altogether (via no-advertise). This setup gives Site B the flexibility of simply attaching a community to a route and having the provider take a predefined action on the signaled community. The benefit of this type of setup is that Site B will not have to worry about using AS_PATH prepending or other BGP policy mechanisms to influence inbound routing.

The `send-community` string must be added to the neighbor statement to allow communities to be exchanged between the peers. Then the route-map is applied in the inbound direction toward the peer so that the PE can take the appropriate actions based on the communities that are sent from the CE. Note that a blank permit statement must be present at the bottom of the route-map so that all routes are permitted even if they are not tagged with communities.

The `ip bgp-community new-format` command is also added on the PEs so that the communities are displayed in AA:NN format instead of the 32-bit format. For example, community 65100:90 would be displayed as `4266393690` in show commands without this command present. This command does not change the functionality of communities, it simply allows us to view them in the more readable AA:NN format.

Task 7.2 Solutions

```
R7:
ip bgp-community new-format
!
ip community-list 90 permit 65100:90
```

https://t.me/learningnets

```
ip community-list 99 permit 65100:999
ip community-list 110 permit 65100:110
!
route-map RFC1998_INBOUND_SITE_B permit 10
 match community 90
 set local-preference 90
route-map RFC1998_INBOUND_SITE_B permit 20
 match community 110
 set local-preference 110
route-map RFC1998_INBOUND_SITE_B permit 30
 match community 99
 set community no-advertise
route-map RFC1998_INBOUND_SITE_B permit 1000
!
router bgp 65006
 address-family ipv4 vrf VPN_CCIE
  neighbor 172.30.79.9 send-community both
  neighbor 172.30.79.9 route-map RFC1998_INBOUND_SITE_B in
 exit-address-family
```

R8:

```
ip bgp-community new-format
!
ip community-list 90 permit 65100:90
ip community-list 99 permit 65100:999
ip community-list 110 permit 65100:110
!
route-map RFC1998_INBOUND_SITE_B permit 10
 match community 90
 set local-preference 90
route-map RFC1998_INBOUND_SITE_B permit 20
 match community 110
 set local-preference 110
route-map RFC1998_INBOUND_SITE_B permit 30
 match community 99
 set community no-advertise
route-map RFC1998_INBOUND_SITE_B permit 1000
!
router bgp 65006
 address-family ipv4 vrf VPN_CCIE
  neighbor 172.30.89.9 send-community both
  neighbor 172.30.89.9 route-map RFC1998_INBOUND_SITE_B in
 exit-address-family
```

R9:

```
router bgp 65100
```

https://t.me/learningnets

```
 neighbor 172.30.79.7 send-community
 neighbor 172.30.89.8 send-community
```

Task 7.2 Verification

To verify this task, we will tag some routes on R9 with some of these communities and observe how the PEs react to them.

Currently R2 prefers to route toward PE R7 to get to R9's Loopback0.

```
R2#show bgp vpnv4 unicast vrf VPN_CCIE 192.122.3.9/32
          BGP routing table entry for 65066:200:192.122.3.9/32
, version 65
Paths: (2 available, best #1, table VPN_CCIE)
  Advertised to update-groups:
     3
  Refresh Epoch 1
  65100, imported path from 65066:700:192.122.3.9/32 (global) 192.122.3.7
 (metric 20) (via default) from 192.122.3.4 (192.122.3.4)      Origin IGP, metric 0, localpref 100
, valid, internal, best
      Extended Community: RT:100:100
      Originator: 192.122.3.7, Cluster list: 192.122.3.4
      mpls labels in/out nolabel/46
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  65100, imported path from 65006:800:192.122.3.9/32 (global) 192.122.3.8
 (metric 20) (via default) from 192.122.3.4 (192.122.3.4)
      Origin IGP, metric 0, localpref 100, valid, internal
      Extended Community: RT:100:100
      Originator: 192.122.3.8, Cluster list: 192.122.3.4
      mpls labels in/out nolabel/45
      rx pathid: 0, tx pathid: 0
```

Currently R7 prefers using the direct eBGP link to reach R9's Loopback0.

```
R7#show bgp vpnv4 unicast vrf VPN_CCIE 192.122.3.9/32
 BGP routing table entry for 65066:700:192.122.3.9/32
, version 19
Paths: (2 available, best #2, table VPN_CCIE)
  Advertised to update-groups:
     2
  Refresh Epoch 2
  65100, imported path from 65006:800:192.122.3.9/32 (global) 192.122.3.8
 (metric 5760) (via default) from 192.122.3.4 (192.122.3.4)
```

https://t.me/learningnets

```
      Origin IGP, metric 0, localpref 100
, valid, internal
      Extended Community: RT:100:100
      Originator: 192.122.3.8, Cluster list: 192.122.3.4
      mpls labels in/out 46/45
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  65100 172.30.79.9 (via vrf VPN_CCIE
) from 172.30.79.9 (192.122.3.9)      Origin IGP, metric 0, localpref 100, valid, external, best

      Extended Community: RT:100:100
      mpls labels in/out 46/nolabel
  rx pathid: 0, tx pathid: 0x0
```

If we signal community 65100:110 to R8, the MPLS cloud should begin preferring R8 as the entry point into Site B to reach R9's Loopback0. R8 will begin setting local-pref of 110 for this route and thus the iBGP VPNv4 domain will prefer this advertisement over the one from R7 with a local-preference of 100. In fact, R7 will also begin routing toward R8 to get to R9's Loopback0.

```
R9#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R9(config)#ip prefix-list R9_LOOPBACK0 seq 5 permit 192.122.3.9/32
R9(config)#route-map RFC_1998_POLICY permit 10
R9(config-route-map)#match ip address prefix-list R9_LOOPBACK0
R9(config-route-map)#set community 65100:110
R9(config-route-map)#route-map RFC_1998_POLICY permit 100
R9(config-route-map)#R9(config-route-map)#router bgp 65100
R9(config-router)#neighbor 172.30.89.8 route-map RFC_1998_POLICY out

R9(config-router)#R9(config-router)#do clear bgp ipv4 unicast * soft out
R9(config-router)#end

R9#
```

R2 now only sees one path though R8; before it had an the advertisement from both R7 and R8.

```
R2#show bgp vpnv4 unicast vrf VPN_CCIE 192.122.3.9/32
 BGP routing table entry for 65066:200:192.122.3.9/32
, version 70
Paths: (1 available, best #1, table VPN_CCIE)
  Advertised to update-groups:
     3
```

https://t.me/learningnets

```
  Refresh Epoch 1
  65100, imported path from 65006:800:192.122.3.9/32 (global) 192.122.3.8
 (metric 20) (via default) from 192.122.3.4 (192.122.3.4)      Origin IGP, metric 0, localpref 110
, valid, internal, best

      Extended Community: RT:100:100
      Originator: 192.122.3.8, Cluster list: 192.122.3.4
      mpls labels in/out nolabel/45
      rx pathid: 0, tx pathid: 0x0
```

R7 now prefers routing through R8 instead of via its direct connection to the CE.

```
R7#show bgp vpnv4 unicast vrf VPN_CCIE 192.122.3.9/32
 BGP routing table entry for 65066:700:192.122.3.9/32
, version 33
Paths: (2 available, best #1, table VPN_CCIE)
  Advertised to update-groups:
     3
  Refresh Epoch 2
  65100, imported path from 65006:800:192.122.3.9/32 (global) 192.122.3.8
 (metric 5760) (via default) from 192.122.3.4 (192.122.3.4)      Origin IGP, metric 0, localpref 110
, valid, internal, best
      Extended Community: RT:100:100
      Originator: 192.122.3.8, Cluster list: 192.122.3.4
      mpls labels in/out nolabel/45
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 2
  65100
    172.30.79.9 (via vrf VPN_CCIE) from 172.30.79.9 (192.122.3.9)      Origin IGP, metric 0,
 ocalpref 100, valid, external

      Extended Community: RT:100:100
      rx pathid: 0, tx pathid: 0
```

The reason R2 only sees one path is because both PEs are now making the same routing decision. R7 and R8 have the same "best path" (through R8); before the policy change, each PE was selecting a different path to reach R9's Loopback0. This single best path is advertised to the RR and then toward the rest of the iBGP peers.

Task 7.3

https://t.me/learningnets

This asks us to configure the network so that R18 does not send queries to R16 and R17 when a route goes active in the DUAL calculation. We also must ensure that R16 and R17 can redistribute routes in the future without making any modifications.

This requirement is referring to the Stubs feature of EIGRP. Both R16 and R17 are non transit nodes, meaning that R18 does not route through them to get to other destinations. The connected interface of R16 and R17 are the only networks that R18 ever has to reach. These routers can be thought of as "stubs" or "ends" of the network; there are no other networks behind them. The issue, though, is that when a route goes active in DUAL, an EIGRP node will query all of its peers for that route (even if the route that went active was a connected route on the node sending the queries!). If we know that R16 and R17 have no other networks behind them besides what they are advertising, then it makes little sense to query them for routes that they do not have.

Configuring them as Stubs ensures that R18 will not query them when routes go active. For example, if 192.122.3.20/20 (the loopback of R20) goes active, R18 will not query R16 and R17 as they have been marked as Stubs.

There are multiple options to the Stub feature that allow control over what is advertised by the Stub node. By default, the Stub node will only advertise connected interface and summary routes. The "redistributed" option allows the Stub node to advertise redistributed routes into EIGRP. Without this option, any routes redistributed from other protocols into EIGRP will not be advertised to R18.

Pitfall

The servers will lose reachability to each other as soon as we configure the Stubs feature on R16 and R17. The only routes that server2 and server3 will have are the connected interfaces that their corresponding upstream router has (Tunnel100 and Loopback0 on R17, and Tunnel100 on R16). To account for this, a possible solution is to include the `summary` keyword in the stub configuration and advertise summaries for the 172.0.0.0/8 space toward the servers from R16 and R17.

Task 7.3 Solutions

```
R16:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 123
  !
  af-interface GigabitEthernet1.160
   summary-address 172.0.0.0 255.0.0.0
  exit-af-interface
  !
```

https://t.me/learningnets

```
  topology base
  eigrp stub connected summary redistributed
 exit-address-family

R17:
router eigrp INE_CCIE
 !
 address-family ipv4 unicast autonomous-system 123
  !
  af-interface GigabitEthernet1.170
   summary-address 172.0.0.0 255.0.0.0
  exit-af-interface
  !
  topology base
  eigrp stub connected summary redistributed
 exit-address-family
```

Task 7.3 Verification

Verification can be seen from R18's detailed output below. All peers on this interface have been detected as Stubs.

```
R18#show ip eigrp interfaces detail tunnel100

EIGRP-IPv4 VR(INE_CCIE) Address-Family Interfaces for AS(123)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Tu100 2
        0/0       0/0           8       6/233        502           0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is disabled
  Next xmit serial <none>
  Packetized sent/expedited: 41/1
  Hello's sent/expedited: 38593/10
  Un/reliable mcasts: 0/0  Un/reliable ucasts: 48/64
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 1
  Retransmissions sent: 0  Out-of-sequence rcvd: 0 Interface has all stub peers

  Topology-ids on interface - 0
  Authentication mode is not set
```

Note the "Suppressing Queries" output on both R16 and R17.

https://t.me/learningnets

```
R18#show ip eigrp neighbors detail

EIGRP-IPv4 VR(INE_CCIE) Address-Family Neighbors for AS(123)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num1172.100.123.17
Tu100
                    14 00:03:33    8  1398  0  51
   Version 17.0/2.0, Retrans: 0, Retries: 0, Prefixes: 2
   Topology-ids from peer - 0  Stub Peer Advertising (CONNECTED SUMMARY REDISTRIBUTED  ) Routes
 Suppressing queries
0172.100.123.16 Tu100
                    11 00:03:55    9  1398  0  63
   Version 17.0/2.0, Retrans: 0, Retries: 0, Prefixes: 1
   Topology-ids from peer - 0  Stub Peer Advertising (CONNECTED SUMMARY REDISTRIBUTED  ) Routes
 Suppressing queries

3   172.27.182.20           Gi1.182                  11 2d01h       1   100  0  17
   Version 17.0/2.0, Retrans: 2, Retries: 0, Prefixes: 2
   Topology-ids from peer - 0
2   172.27.181.19           Gi1.181                  11 2d01h       1   100  0  19
   Version 17.0/2.0, Retrans: 2, Retries: 0, Prefixes: 2
   Topology-ids from peer - 0
Max Nbrs: 0, Current Nbrs: 0
```

Let's redistribute a route into EIGRP on R16 and verify that it is advertised.

```
R16#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R16(config)#ip route 16.16.16.16 255.255.255.255 null 0
R16(config)#router eigrp INE_CCIE
R16(config-router)#address-family ipv4 unicast autonomous-system 123
R16(config-router-af)#topology base
R16(config-router-af-topology)#redistribute static
R16(config-router-af-topology)#end
R16#
R16#show ip eigrp topology 16.16.16.16/32
EIGRP-IPv4 VR(INE_CCIE) Topology Entry for AS(123)/ID(192.122.3.16) for16.16.16.16/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 131072
  Descriptor Blocks:  0.0.0.0,from Rstatic
, Send flag is 0x0
      Composite metric is (131072/0), route is External
      Vector metric:
        Minimum bandwidth is 10000000 Kbit
        Total delay is 1000000 picoseconds
        Reliability is 0/255
```

https://t.me/learningnets

```
        Load is 0/255
        Minimum MTU is 1500
        Hop count is 0 Originating router is 192.122.3.16
      External data:
        AS number of route is 0 External protocol is Static
, external metric is 0
        Administrator tag is 0 (0x00000000)
R18#show ip route 16.16.16.16
Routing entry for 16.16.16.16/32
  Known via "eigrp 123", distance 170, metric 76800512, type external
  Redistributing via eigrp 123  Last update from 172.100.123.16 on Tunnel100
, 00:03:03 ago
  Routing Descriptor Blocks:  * 172.100.123.16
, from 172.100.123.16, 00:03:03 ago, via Tunnel100
      Route metric is 76800512, traffic share count is 1
      Total delay is 50001 microseconds, minimum bandwidth is 100 Kbit
      Reliability 255/255, minimum MTU 1400 bytes
      Loading 1/255, Hops 1
R16(config)#no ip route 16.16.16.16 255.255.255.255 null 0
R16(config)#router eigrp INE_CCIE
R16(config-router)#address-family ipv4 unicast autonomous-system 123
R16(config-router-af)#topology base
R16(config-router-af-topology)#no redistribute static
R16(config-router-af-topology)#end
R16#
```

Now verify that the servers still have connectivity.

```
R15%server3#show ip route eigrp


Routing Table: server3
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set
 D     172.0.0.0/8 [90/15360] via 172.23.160.16, 00:11:39, GigabitEthernet1.160
R15%server3#
```

https://t.me/learningnets

```
R15%server3#ping 172.25.170.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.25.170.100, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/14/29 ms
R15%server3#trace 172.25.170.100
Type escape sequence to abort.
Tracing the route to 172.25.170.100
VRF info: (vrf in name/id, vrf out name/id)
  1 172.23.160.16 3 msec 1 msec 0 msec
  2 172.100.123.17 2 msec 1 msec 2 msec
  3 172.25.170.100 11 msec *  2 msec
R15%server3#ping 172.27.192.100

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.27.192.100, timeout is 2 seconds:!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/35/55 ms
R15%server3#
R15%server3#trace 172.27.192.100

Type escape sequence to abort.
Tracing the route to 172.27.192.100
VRF info: (vrf in name/id, vrf out name/id)
  1 172.23.160.16 3 msec 1 msec 1 msec
  2 172.100.123.18 2 msec 2 msec 11 msec
  3 172.27.182.20 26 msec 24 msec 36 msec
  4 172.27.192.100 29 msec *  3 msec
```

Task 8.1

In this task we are asked ensure that SSHv2 is the only remote shell protocol allowed on R7 and R8. Both of these devices should only accept incoming SSHv2 sessions to port 4022 and should account for packet filters that may be introduced in the network in the near future.

The VTY lines will be restricted to only allow SSH by using the `transport input ssh` command. This meets the requirement of restricting remote access to R7 and R8 to SSH only without using an ACL.

A rotary must be set up on the routers to make them listen for incoming SSH sessions on port 4022. An access-list will be used to ensure that only port 4022 can be used for SSH access. This will be applied via the `access-class <ACL-NAME> vrf-also` command under the VTY lines. The `vrf-also` keyword ensures that this filter is also

https://t.me/learningnets

applied to traffic coming in on interfaces that are members of VRFs instead of the global table. Without this ACL, the router will allow incoming SSH sessions on both port 22 and 4022.

We must set the `ip ssh source-interface` to the Loopback0 on R7 and R8 to account for packet filters that would only allow traffic when sourced from the Loopback0 networks of the devices.

The minimum RSA key modulus required to enable SSHv2 is 768; R7 and R8 must use a modulus of 1792 to meet the requirements of this task.

Task 8.1 Solutions

```
R7:
ip domain-name ine.ccie.lab
!
crypto key generate rsa modulus 1792
!
ip ssh version 2
ip ssh source-interface loopback 0
ip ssh port 4022 rotary 78
!
ip access-list extended SSHv2_PORT_4022
 permit tcp any any eq 4022
!
line vty 0 98
 access-class SSHv2_PORT_4022 in vrf-also
 login local
 rotary 78
 transport input ssh

R8:
ip domain-name ine.ccie.lab
!
crypto key generate rsa modulus 1792
!
ip ssh version 2
ip ssh source-interface loopback 0
ip ssh port 4022 rotary 78
!
ip access-list extended SSHv2_PORT_4022
 permit tcp any any eq 4022
!
line vty 0 98
 access-class SSHv2_PORT_4022 in vrf-also
```

https://t.me/learningnets

```
 login local
 rotary 78
 transport input ssh
```

Task 8.1 Verification

SSHv2 is enabled on R7 and R8.

```
R8#show ip ssh
 SSH Enabled - version 2.0

Authentication methods:publickey,keyboard-interactive,password
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
IOS Keys in SECSH format(ssh-rsa, base64 encoded):
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgQCUq+XP295bbmZ4jJZ1pB+vJrxqcL7mF491YVgNzuaU
yFt0cHzzftp3EudjqmqrZPxUIZfYSFEt0SasCfepRxHFl+8ng6Wm3yPns/06NCmJsUb915S9gXjAowm6
bFnYHL1nN+fhfkrJbwWOwjKnYq2BEopnyJK/K4eK/nMwkA3GAQ==
```

Let's try to SSH to R7 using the default port 22 and verify that the session is not established.

```
R1#ssh -l cisco -v 2 -p 22 192.122.3.7

% Connection refused by remote host
```

If we change the port to 4022, R7 should accept the session.

```
R1#ssh -l cisco -v 2 -p 4022 192.122.3.7

Password: cisco

R7#
R7#clear line vty 0

[confirm]
 [OK]
R7#
```

Now try to access R7 using SSHv1.

https://t.me/learningnets

```
R1#ssh -l cisco -v 1 -p 4022 192.122.3.7

[Connection to 192.122.3.7 aborted: error status 0]
R1#
```

Let's try from a CE router to verify that these same restrictions still apply when traffic is coming in to an interface participating in a VRF.

```
R9#ssh -l cisco -v 2 -p 22 172.30.89.8
% Connection refused by remote host

R9#
R9#ssh -l cisco -v 2 -p 4022 172.30.89.8

Password: cisco

R8#
R8#clear line vty 0

[confirm]
 [OK]
R8#
```

Now verify that the source-interface policy is working by SSHing to R1 from R7.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.R1(config)#ip domain-name ine.ccie.lab
R1(config)#crypto key generate rsa modulus 1792
The name for the keys will be: R1.ine.ccie.lab

% The key modulus size is 1792 bits
% Generating 1792 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)
R1(config)#line vty 0 98
R1(config-line)#login local
R1(config-line)#end

R1#
```

Note that the source address displayed from the output on R1 is the Loopback0 of R7. Normally R7 would have selected the IP address of the interface used to route towards 192.122.3.1.

https://t.me/learningnets

```
R7#ssh -l cisco 192.122.3.1

Password: cisco
R1#show tcp brief

TCB       Local Address               Foreign Address             (state)
7FB7CDF34068  192.122.3.1.646            192.122.3.3.53044           ESTAB
7FB7CE120C88  192.122.3.1.646            192.122.3.7.41827           ESTAB
7FB7CDB96458  192.122.3.1.646            192.122.3.16.46403          ESTAB
7FB7CE131380  192.122.3.1.646            192.122.3.8.54406           ESTAB 7FB7CE130278 192.122.3.1
.22 192.122.3.7
.43592           ESTAB
7FB7CDD656C0  192.122.3.1.646            192.122.3.2.19991           ESTAB
R1#
R1#show tcp tcb 7FB7CE130278

Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255 Local host: 192.122.3.1,
 Local port: 22
 Foreign host: 192.122.3.7, Foreign port: 43592
Connection tableid (VRF): 0
Maximum output segment queue size: 20
R1#clear line vty 0

[confirm]
 [OK]
R1#
```

Task 8.2

We are asked to protect the control-plane of devices on the network in this task. It is important to always keep the device's control-plane in mind, specially when dealing with devices that are software based (ISRs, 7200s). These devices use their main CPU for both control and data plane function;- the same CPU is used to forward data traffic through the device as well as maintain routing protocol data structures, management counters, logging, and management functions. If any of these processes causes the CPU of a device to be maxed out at 100% utilization for a prolonged period of time, all other functions will be starved of timely cycles and will suffer. Routing protocol peers can begin flapping, remote access to the device can be lost, or the device can simply crash.

Control-plane policing is one of the tools that can be leveraged to protect the control plane of the device. Sseveral platforms, such as the Cisco Nexus, come with built in

https://t.me/learningnets

control-plane policing profiles and give the device "out of the box" protection. This task calls for limiting the amount of PIM control-plane traffic to 256 kps. We can utilize the QoS MQC syntax to write a policy that matches and polices PIM traffic, and then apply it inbound (for traffic received) to the control-plane interface.

A maximum number of allowed prefixes can be defined for almost all routing protocols, to protect the device against bad behaved routing peers. This design is typical in Service Provider PE routers that are used to peer with customers in a L3VPN service. Defining a hard limit mitigates the case in which a customer leaks a very large number of routes to the provider, wasting resources on the PE device and potentially crashing the PE router. Hardware resources are limited on hardware-based platforms (7600, ASR9K, Nexus); these devices use ASICs to forward data traffic instead of the central CPU. Hardware-based platforms push down the routing information into a limited resource called TCAM. This allows hardware platforms to forward packets at a much faster rate than any software-based platform; the CPU is not involved in data forwarding through the device and all "routing lookups" are performed against the line card's TCAM (done in hardware). What if a customer leaked the entire Internet routing table to a PE device? What if this PE router was servicing hundreds of customers? The PE would have to run BGP best path against all of those routes, and then install all of the best paths into the FIB (wasting resources). The PE device will crash if TCAM resources are exhausted, so it is important to protect the control plane by limiting the number of routes that customers can send in as well.

Task 8.2 Solutions

```
R2:
router bgp 65006
 !
 address-family ipv4 vrf VPN_CCIE
  neighbor 202.4.210.10 maximum-prefix 1000 80 restart 5
 exit-address-family

R3:
ip access-list extended PIM
 permit pim any any
!
class-map match-all PIM
 match access-group name PIM
!
policy-map COPP_POLICE_IN
 class PIM
  police 256000
```

https://t.me/learningnets

```
!
control-plane
 service-policy input COPP_POLICE_IN

R4:
ip access-list extended PIM
 permit pim any any
!
class-map match-all PIM
 match access-group name PIM
!
policy-map COPP_POLICE_IN
 class PIM
  police 256000
!
control-plane
 service-policy input COPP_POLICE_IN
```

Task 8.2 Verification

We can look at a couple of show commands to verify that the control plane protection is in effect.

https://t.me/learningnets

```
R3#show policy-map control-plane input

 Control Plane

  Service-policy input: COPP_POLICE_IN

    Class-map: PIM (match-all)  16851 packets, 1267356 bytes

      5 minute offered rate 0000 bps, drop rate 0000 bps Match: access-group name PIM

      police: cir 256000 bps
, bc 8000 bytes onformed 16851 packets
, 1267356 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0000 bps, exceeded 0000 bps

    Class-map: class-default (match-any)
      147940 packets, 11516875 bytes
      5 minute offered rate 1000 bps, drop rate 0000 bps
      Match: any
```

Note that in our lab network there will never be 256 Kbps worth of PIM control-plane traffic.

```
R2#show bgp vpnv4 unicast vrf VPN_CCIE neighbors
 BGP neighbor is 202.4.210.10, vrf VPN_CCIE, remote AS 65100
, external link
  BGP version 4, remote router ID 192.122.3.10
  BGP state = Established, up for 6d00h
  Last read 00:00:48, last write 00:00:12, hold time is 180, keepalive interval is 60 seconds
  Neighbor sessions:
    1 active, is not multisession capable (disabled)
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
    Enhanced Refresh Capability: advertised and received
    Multisession Capability:
    Stateful switchover support enabled: NO for session 1
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
```

https://t.me/learningnets

```
                         Sent       Rcvd
    Opens:                  1          1
    Notifications:          0          0
    Updates:               12        179
    Keepalives:          9558       9470
    Route Refresh:          0          0
    Total:               9571       9650
  Default minimum time between advertisement runs is 0 seconds

 For address family: VPNv4 Unicast
  Translates address family IPv4 Unicast for VRF VPN_CCIE
  Session: 202.4.210.10
  BGP table version 71, neighbor version 71/0
  Output queue size : 0
  Index 3, Advertise bit 1
  3 update-group member
  Overrides the neighbor AS with my AS before sending updates
  Slow-peer detection is disabled
  Slow-peer split-update-group dynamic is disabled
  Interface associated: GigabitEthernet1.210
                                Sent       Rcvd
  Prefix activity:              ----       ---- Prefixes Current:                3          12
 (Consumes 1440 bytes)
    Prefixes Total:               18        362
    Implicit Withdraw:            11        350
    Explicit Withdraw:             4          0
    Used as bestpath:            n/a         12
    Used as multipath:           n/a          0

                                   Outbound    Inbound
  Local Policy Denied Prefixes:    --------    ------- Bestpath from this peer:              1
2        n/a
    Total:                                12          0 Maximum prefixes allowed 1000
 Threshold for warning message 80%, restart interval 5 min

  Number of NLRIs in the update sent: max 3, min 0
  Last detected as dynamic slow peer: never
  Dynamic slow peer recovered: never
  Refresh Epoch: 1
```

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Full-Scale Labs

Full-Scale Lab 2 Tasks

Diagrams and initial configs for this lab are located in the Resources section in the upper-right portion of this page.

1. LAN Switching
2. Core Routing
3. MPLS
4. Site Routing
5. BGP
6. DMVPN
7. Multicast
8. Infrastructure Security
9. Infrastructure Services

Difficulty Rating (10 highest): 6

Lab Overview

The following scenario is a practice lab exam designed to test your skills at configuring Cisco networking devices. Specifically, this scenario is designed to assist you in your preparation for Cisco's CCIE Routing & Switching Lab Exam. However, remember that in addition to being designed as a simulation of the actual CCIE lab exam, this practice lab should be used as a learning tool. Instead of rushing through the lab to complete all the configuration steps, take the time to research the networking technology in question and gain a deeper understanding of the principles behind its operation.

Lab Instructions

Before starting, ensure that the initial configuration scripts for this lab have been applied. These scripts can be found in the Resources section in the upper-right portion of this page. These scripts can also be automatically loaded for you via the

https://t.me/learningnets

Rack Control Panel if you are using INE's rack rentals.

Refer to the attached diagrams for interface, addressing, and protocol assignments. Upon completion of the scenario, all devices should have full IP reachability to networks in the routing domain as specified by each task.

If you have any questions related to the scenario solutions, visit our online community at http://IEOC.com.

Lab Do’s and Don’ts

- Do not change or add any IP addresses from the initial configuration unless otherwise specified or required for troubleshooting.
- If additional IP addresses are needed but not specifically permitted by the task, use IP unnumbered.
- Do not change any interface encapsulations unless otherwise specified.
- Do not change the console, AUX, and VTY passwords or access methods unless otherwise specified.
- Do not use any static routes, default routes, default networks, or policy routing unless otherwise specified.
- Save your configurations often.

Grading

This practice lab consists of various sections totaling 100 points. A score of 80 points is required to pass the exam. A section must work 100% with the requirements given to be awarded the points for that section. No partial credit is awarded. If a section has multiple possible solutions, choose the solution that best meets the requirements.

Point Values

The point values for each section are as follows:

Section	Point Value
LAN Switching	12

https://t.me/learningnets

Section	Point Value
Core Routing	12
MPLS	12
Site Routing	8
BGP	12
DMVPN	12
Multicast	12
Infrastructure Security	8
Infrastructure Services	12

GOOD LUCK!

---

1. LAN Switching

1.1 EtherChannel

- Configure the following links as EtherChannels:
  - SW1 to SW2's links
  - SW2 to SW3's links
  - SW2 to SW4's links
  - SW3 to SW4's links
- All of these links should be 802.1q trunks, and use LACP for link aggregation negotiation.
- Disable the links from SW1 to SW3 and from SW1 to SW4.

https://t.me/learningnets

Points: 3

1.2 Trunking

- Configure SW1's port Fa0/1 as an 802.1q trunk link as follows:
  - Do not send DTP negotiation requests.
  - Limit the port to forward the minimum number of VLANs necessary.
  - If the link flaps it should not generate an STP Topology Change Notification.

Points: 3

1.3 VLANs

- Create VLANs 227, 228, 1623, and 1723 on SW4.
- Do not manually create these VLANs on any other switches.
- Once complete SW2 should be able to ping R7 and R8, and SW3 should be able to ping R16 and R17.

Points: 3

1.4 Spanning-Tree

- Modify the switches to use the minimum number of STP instances necessary.
- Modify the switches so that layer 2 traffic from R16 to SW3 forwards from SW1 to SW2 to SW4 to SW3. Do not use any interface level commands to accomplish this.

Points: 3

---

2. Core Routing

2.1 Core Routing

- Configure OSPFv3 on the VPN Core routers (R1-R6) as follows:
  - Enable OSPFv3 area 0 for both IPv4 and IPv6 on all core facing links.
  - Advertise both the IPv4 and IPv6 Loopback0 interfaces into OSPFv3 area 0 as internal routes, but do not send hellos out these interfaces.

https://t.me/learningnets

- One complete the VPN Core routers should all have both IPv4 and IPv6 connectivity to and from their Loopback0 networks.

Points: 3

2.2 Core Security

- Configure the VPN Core routers with OSPF security as follows:
  - Authenticate adjacencies for both IPv4 and IPv6 AFIs.
  - Use the 512-bit SHA authentication algorithm.
  - Use the authentication key “CISCO”.
  - Any future links added to area 0 should require this authentication algorithm and key.

Points 3

2.3 Core Resource Optimization

- Optimize the OSPFv3 routing database on the VPN Core routers as follows:
  - Do not advertise the IPv4 transit links into the OSPFv3 database.
  - Do not advertise unnecessary Network LSAs into the OSPFv3 database.

Points: 3

2.4 Core Convergence Optimization

- Optimize link failure detection on the VPN Core routers as follows:
  - Use Bidirectional Forwarding Detection on all core facing interfaces for OSPFv3 dead neighbor detection.
  - BFD keepalives should be sent every 250ms with a dead interval of 750ms.

Points: 3

---

3. MPLS

3.1 PE to CE OSPFv2 Routing

https://t.me/learningnets

- Enable VRF aware OSPFv2 area 0 on R1, R2, R4, and R5 for their links to Sites A, B, and C per the diagram.
- In Sites A and C, enable OSPF area 0 on all links, including the Loopback0 interfaces.
- In Site B, enable OSPF area 0 on R9's link to R4, between R9, R10, & R11, and advertise R9, R10, & R11's Loopback0 interfaces into area 0.
- In Site B, enable OSPF area 1 between R10, R11, & R12, and advertise R12's Loopback0 into area 1.

Points: 4

3.2 MPLS Label Distribution

- Enable LDP on all core facing interfaces on the VPN core routers.

Points: 4

3.3 MPLS VPNv4 Routing

- VRFs and BGP have been pre-configured on the VPN core routers.
- Modify the configuration on R1, R2, R4, and R5 so that VPN Sites A, B, and C have full IPv4 reachability to each other.
- Do not add any additional BGP peerings to accomplish this.
- Traffic from Sites B and C to Site A should be load balanced over the VPN core routers to both R1 and R2.

Points: 4

---

4. Site Routing

4.1 OSPF Inter-Area Traffic Engineering

https://t.me/learningnets

- Configure OSPF in Site B so that R12 has only a default route in its routing table for all destinations.
- R11 should be R12's preferred exit point out of area 1, but R10 should be used if R11 fails.
- Do not use any interface level commands to accomplish this.

Points: 4

4.2 OSPF Intra-Area Traffic Engineering

- Configure OSPF in Site C so that R15 prefers to use R13 to reach all inter-Site destinations.
- Return traffic back to R15 should prefer to enter Site C via R5's link to R14.
- Do not use any default routing or route filtering to accomplish this.

Points: 4

---

5. BGP

5.1 BGP Peering

- BGP is preconfigured as follows:
  - R2 and R3 peer EBGP with ISP A (R15).
  - R6 peers EBGP with ISP B (R12).
  - ISP A peers EBGP with to both R16 & R17 in Site W, and R18 in Site X.
  - ISP B peers EBGP with R18 in Site X, R19 in Site Y, and R20 in Site Z.
- Configure R16, R17, R18, R19, and R20 to support these peerings using the BGP ASN information in the diagram.
- Advertise their Loopback0 interfaces into BGP.
- Once complete, R16 – R20 should have IPv4 reachability to their Loopback0 networks.
- Do not modify the configuration of ISP A (R15) or ISP B (R12).
- Do not add any additional BGP peerings on the VPN Core routers to accomplish this.
- Do not advertise any of the 169.254.0.0 transit links between BGP peers into IGP or BGP to accomplish this.

https://t.me/learningnets

Points: 4

5.2 BGP Transit Filtering

- Configure R18 so that transit traffic between ISP A and ISP B is denied.
- Use an AS-Path based ACL to accomplish this.

Points: 4

5.3 BGP Traffic Engineering

- Configure R18 so that ISP B is used only as a backup for ISP A for both inbound and outbound traffic.

Points: 4

---

6. DMVPN

6.1 DMVPN

- Configure two DMVPN tunnels for Sites X, Y, and Z to reach Site W. The first tunnel will use R16 as the hub, the second will use R17.
- The first tunnel to R16 should be configured as follows:
  - Use interface Tunnel 1.
  - Source the tunnel from the Loopback0 interface.
  - Use addresses 183.100.1.X/24, where X is the device number.
  - Use the NHRP Network-ID 1.
  - Use the NHRP authentication string “DMVPN1”
- The second tunnel to R17 should be configured as follows:
  - Use interface Tunnel 2.
  - Source the tunnel from the Loopback0 interface.
  - Use addresses 183.100.2.X/24, where X is the device number.
  - Use the NHRP Network-ID 2.
  - Use the NHRP authentication string “DMVPN2”

https://t.me/learningnets

Points: 4

6.2 DMVPN over IPsec

- Configure both DMVPN tunnels with IPsec encryption using the following parameters:
  - IPsec Phase 1 encryption – 3DES
  - IPsec Phase 1 hash – SHA-1
  - IPsec Phase 1 password – “DMVPNPASS”
  - IPsec Phase 1 DH group – 5
  - IPsec Phase 2 encapsulation – ESP transport mode
  - IPsec Phase 2 encryption – AES 128-bit
  - IPsec Phase 2 hash – MD5
- Use the minimum amount of ISAKMP keys needed to accomplish this.

Points: 4

6.3 Routing over DMVPN

- Configure the DMVPN so that traffic to Site W prefers to enter via the tunnel to R16.
- Traffic between Sites X, Y, and Z should be able to route directly between them without having to transit through Site W first. Do not use the ip nhrp shortcut command to accomplish this.
- Once complete, IP reachability should be established between all sites.

Points: 4

---

7. Multicast

7.1 Multicast over GRE

- Site B has an IPv4 multicast based application on VLAN 102 that must be accessible by both Sites A and C, but their MPLS L3VPN Service Provider does not support multicast transport. As a workaround, configure inter-site PIM Sparse Mode adjacencies as follows:
  - R9 to R7

https://t.me/learningnets

    - R9 to R8
    - R9 to R15
- Additionally, enable PIM Sparse Mode on the following links:
    - R7 to SW2
    - R8 to SW2
    - VLAN 109 between R9, R10, and R11
    - VLAN 102 between R10, R11, and R12
    - VLAN 100 between R13, R14, and R15
- Do not add any additional IP addresses to accomplish this.

Points: 4

7.2 Multicast RP Distribution

- Configure R9 to send RP announcements using the group addresses 224.0.1.39 and 224.0.1.40 using its Loopback0 interface every 5 seconds.
- Ensure that Site A, B, and C multicast routers are able to learn about the RP information, and that they are able to receive multicast streams from servers on VLAN 102. Use the minimum amount of configuration statements necessary to accomplish this.
- Do not modify the PIM mode to accomplish this.

Points: 4

7.3 Multicast Traffic Engineering

- The multicast based application will be located on VLAN 102.
- To test this, configure R13, R14, and SW2 to generate IGMP Report messages for the (*,G) pair (*,224.1.2.3).
- R12 should be able to ping this multicast address and get an ICMP Echo-Reply from R13, R14, and SW2.
- Configure the network so that traffic from the application server to Site A and Site C prefers to route through R10 over R11. If R10 is unavailable then traffic should route through R11 to reach the remote sites.
- Do not use static multicast routes or multicast BGP accomplish this.

Points: 4

https://t.me/learningnets

8. Infrastructure Security

8.1 Denial of Service Tracking

- Network administrators in Site B have been getting complaints from users that an internal web server with the IP address 172.30.102.100 is sometimes unreachable. After further investigation, it was found that the server has been having periods of abnormally high CPU usage, which potentially might indicate that it is under DoS attack.
- To help gather more information, configure R9 to log all HTTP SYN packets received on the link to the VPN Core which are destined to the server.
- These logs should only include public address sources, not private ones.
- Logs should also include the MAC address of the device that forwarded the packet onto the segment.

Points: 3

8.2 Syslog

- Configure R9 to redirect these log messages to the internal logging server located at 172.30.102.200.
- For added accounting security, configure R9 to include sequence numbers in the log messages, and timestamp them to the millisecond with the local datetime.
- To reduce CPU load from the logging process, R9 should only generate a log message for every 10 packets that match the logging filter.

Points: 2

8.3 Traffic Filtering

https://t.me/learningnets

- Further investigation of log messages has indicated that a large volume of traffic to the affected server is originating from spoofed source addresses.
- To mitigate this type of attack, configure R9 so that traffic received from their VPN provider is dropped if it is originated from source addresses that have not been legitimately advertised into IGP.
- Do not use any access-lists to accomplish this filtering.

Points: 3

---

9. Infrastructure Services

9.1 NTP

- After implementing syslog logging, your NOC engineers have noticed inconsistent timestamps on your device logs. To resolve this problem, you have decided to maintain consistent time by implementing Network Time Protocol.
- Configure NTP on the devices in Sites A, B, and C as follows:
  - R8, R9, and R13 should be NTP stratum 3 time sources.
  - These devices should all synchronize their time with each other.
  - Devices in Site A should get time from R8.
  - Devices in Site B should get time from R9.
  - Devices in Site C should get time from R13.

Points: 3

9.2 NTP Security

- Configure the following policy to ensure security of the NTP communication:
  - R8, R9, and R13 should only be allowed to synchronize with each other's Loopback0 addresses.
  - R8, R9, and R13 should only serve time to the Loopback0 interfaces of the devices within their site.
  - All NTP communication should be MD5 authenticated with the password 'NTPK3Y'

Points: 3

https://t.me/learningnets

9.3 QoS

- R9's link to the VPN provider is via a sub-rate metro Ethernet circuit. The physical interface is GigE, but the VPN provider drops all traffic above 50Mbps.
- Configure an outbound QoS policy on R9 to account for this.
- Traffic in excess of 50Mbps should be preferred to be delayed first, then dropped as a last resort.

Points: 3

9.4 Sub-Rate QoS

- Configure R9 with the following policy to manage its sub-rate circuit:
  - VoIP traffic (DSCP EF) should be given 10Mbps of priority.
  - VoIP signalling (DSCP CS3) should be guaranteed 1Mbps of bandwidth.
  - Transactional (AF21) and Bulk (AF11) Data should be guaranteed 30Mbps of bandwidth, and should not be subject to Tail Drop.
  - All other traffic should have best-effort forwarding.

Points: 3

https://t.me/learningnets

CCIE Routing & Switching v5 Workbook - CCIE R&S v5 Full-Scale Labs

Full-Scale Lab 2 Solutions

This document is currently in beta release. Additional verifications and breakdowns will be added shortly.

Task 1.1 EtherChannel

```
SW1:
interface FastEthernet0/19
 shutdown
!
interface FastEthernet0/20
 shutdown
!
interface FastEthernet0/21
 shutdown
!
interface FastEthernet0/22
 shutdown
!
interface FastEthernet0/23
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 12 mode active
 no shutdown
!
interface FastEthernet0/24
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 12 mode active
 no shutdown

SW2:
interface FastEthernet0/19
 switchport trunk encapsulation dot1q
 switchport mode trunk
```

https://t.me/learningnets

```
 channel-group 24 mode active
 no shutdown
!
interface FastEthernet0/20
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 24 mode active
 no shutdown
!
interface FastEthernet0/21
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 23 mode active
 no shutdown
!
interface FastEthernet0/22
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 23 mode active
 no shutdown
!
interface FastEthernet0/23
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 12 mode active
 no shutdown
!
interface FastEthernet0/24
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 12 mode active
 no shutdown

SW3:
interface FastEthernet0/19
 shutdown
!
interface FastEthernet0/20
 shutdown
!
interface FastEthernet0/21
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 23 mode active
 no shutdown
!
```

https://t.me/learningnets

```
interface FastEthernet0/22
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 23 mode active
 no shutdown
!
interface FastEthernet0/23
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 34 mode active
 no shutdown
!
interface FastEthernet0/24
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 34 mode active
 no shutdown
```

SW4:

```
interface FastEthernet0/19
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 24 mode active
 no shutdown
!
interface FastEthernet0/20
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 24 mode active
 no shutdown
!
interface FastEthernet0/21
 shutdown
!
interface FastEthernet0/22
 shutdown
!
interface FastEthernet0/23
 switchport trunk encapsulation dot1q
 switchport mode trunk
 channel-group 34 mode active
 no shutdown
!
interface FastEthernet0/24
 switchport trunk encapsulation dot1q
 switchport mode trunk
```

https://t.me/learningnets

```
 channel-group 34 mode active
 no shutdown
```

Task 1.1 Verification

```
SW2#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port



Number of channel-groups in use: 3
Number of aggregators:           3

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------12     Po12(SU)
LACP      Fa0/23(P)   Fa0/24(P)
   23     Po23(SU) LACP      Fa0/21(P)   Fa0/22(P)
   24     Po24(SU) LACP      Fa0/19(P)   Fa0/20(P)

SW2#show interfaces trunk



Port        Mode             Encapsulation  Status        Native vlan Po12        on
802.1q         trunking
      1 Po23        on 802.1q         trunking
      1 Po24        on 802.1q         trunking
      1

Port        Vlans allowed on trunk
Po12        1-4094
Po23        1-4094
Po24        1-4094

Port        Vlans allowed and active in management domain
Po12        1
Po23        1
```

https://t.me/learningnets

```
Po24        1

Port        Vlans in spanning tree forwarding state and not pruned
Po12        1
Po23        1
Po24        none
```

Task 1.2 Trunking

```
SW1:
interface FastEthernet0/1
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 227,228,1623,1723
 switchport mode trunk
 switchport nonegotiate
 spanning-tree portfast trunk
 no shutdown
```

Task 1.2 Verification

```
SW1#show interfaces trunk


Port        Mode             Encapsulation  Status        Native vlan Fa0/1       on
802.1q          trunking
      1
Po12        on               802.1q         trunking      1

Port        Vlans allowed on trunk Fa0/1 227-228,1623,1723
Po12        1-4094

Port        Vlans allowed and active in management domain
Fa0/1       227-228,1623,1723
Po12        1,227-228,1623,1723

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/1       227-228,1623,1723
Po12        1,227-228,1623,1723
SW1#show interfaces Fa0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: trunk
```

https://t.me/learningnets

```
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1qNegotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: noneTrunking VLANs Enabled: 227,228,1623,1723
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
SW1#show spanning-tree interface Fa0/1

Vlan                 Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
VLAN0227             Desg FWD 19        128.3P2p Edge
 VLAN0228             Desg FWD 19        128.3P2p Edge
 VLAN1623             Desg FWD 19        128.3P2p Edge
 VLAN1723             Desg FWD 19        128.3P2p Edge
```

Task 1.3 VLANs

```
SW1 - SW3:
vtp domain INE
vtp version 3
vtp mode client vlan

SW4:
SW4#config tSW4(config)#vtp domain INE
SW4(config)#vtp version 3SW4(config)#vtp mode server vlan
```

https://t.me/learningnets

```
Setting device to VTP Server mode for VLANS.
SW4(config)#endSW4#vtp primary vlan

This system is becoming primary server for feature vlan
No conflicting VTP3 devices found.
Do you want to continue? [confirm]
%SW_VLAN-4-VTP_PRIMARY_SERVER_CHG: 001c.576d.3d00 has become the primary server for the VLAN VTP feature

SW4#config t
Enter configuration commands, one per line.  End with CNTL/Z.SW4(config)#vlan 227,228,1623,1723

SW4(config-vlan)#end
SW4#
```

Task 1.3 Verification

```
SW1#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/2, Fa0/3, Fa0/4, Fa0/5
                                                Fa0/6, Fa0/7, Fa0/8, Fa0/9
                                                Fa0/10, Fa0/11, Fa0/12, Fa0/13
                                                Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                                Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                                Fa0/22, Gi0/1, Gi0/2 227  VLAN0227
                        active    228  VLAN0228
                        active
1002 fddi-default                     act/unsup
1003 trcrf-default                    act/unsup
1004 fddinet-default                  act/unsup
1005 trbrf-default                    act/unsup 1623 VLAN1623
                        active    1723 VLAN1723
                        active
SW4#show vtp status
VTP Version capable             : 1 to 3VTP version running             : 3
VTP Domain Name                 : INE
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : 0017.5937.4f00

Feature VLAN:
--------------VTP Operating Mode                : Primary Server
```

https://t.me/learningnets

```
Number of existing VLANs          : 7
Number of existing extended VLANs : 2
Configuration Revision            : 2
Primary ID                        : 0017.5937.4f00
Primary Description               : SW4
MD5 digest                        : 0xE4 0xB4 0x69 0xC4 0x5B 0xD1 0x6D 0x9A
                                    0x87 0xCE 0xA9 0xC3 0xDD 0x31 0x9D 0x79



Feature MST:
--------------
VTP Operating Mode                : Transparent



Feature UNKNOWN:
--------------
VTP Operating Mode                : Transparent
```

Task 1.4 Spanning-Tree

```
SW1 - SW3:
spanning-tree mode mst

SW4:
spanning-tree mode mst
spanning-tree mst 0 priority 0
```

Task 1.4 Verification

Before modifying the root bridge placement, SW3 uses the Port-Channel23 to SW2 to reach R16's MAC address, as seen below:

```
R16#show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  169.254.160.0           -   0050.568d.7905  ARPA   GigabitEthernet1.160
Internet  169.254.160.1         111   0050.568d.7156  ARPA   GigabitEthernet1.160
Internet  183.16.23.16            -   0050.568d.7905
  ARPA   GigabitEthernet1.1623
Internet  183.16.23.23           95   001d.45cc.05c1  ARPA   GigabitEthernet1.1623
SW3#show mac address-table dynamic address 0050.568d.7905
          Mac Address Table
-------------------------------------------
```

https://t.me/learningnets

```
Vlan    Mac Address       Type        Ports
----    -----------       --------    -----1623    0050.568d.7905    DYNAMIC     Po23

Total Mac Addresses for this criterion: 1
```

After SW4 is set as the root bridge, R16's MAC address must be learned on the root port to SW4.

```
SW4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.SW4(config)#spanning-tree mst 0 priority 0
SW4(config)#end
SW3#show spanning-tree mst 0

##### MST0    vlans mapped:   1-4094
Bridge        address 001d.45cc.0580  priority      32768 (32768 sysid 0)
Root          address 001c.576d.3d00  priority      0     (0 sysid 0)
              port    Po34            path cost     0
Regional Root address 001c.576d.3d00  priority      0     (0 sysid 0)
                                      internal cost 100000    rem hops 19
Operational   hello time 2 , forward delay 15, max age 20, txholdcount 6
Configured    hello time 2 , forward delay 15, max age 20, max hops    20

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Po23             Altn BLK 100000    128.232  P2p Po34              Root FWD
 100000    128.320  P2p
SW3#show mac address-table dynamic address 0050.568d.7905
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----1623    0050.568d.7905    DYNAMIC     Po34

Total Mac Addresses for this criterion: 1
```

Task 2.1 Core Routing

```
R1:
interface Loopback0
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.12
```

https://t.me/learningnets

```
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.14
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.15
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
router ospfv3 1
 address-family ipv4 unicast
  passive-interface Loopback0
 exit-address-family
 !
 address-family ipv6 unicast
  passive-interface Loopback0
 exit-address-family
```

R2:

```
interface Loopback0
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.12
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.23
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.24
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.25
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.26
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
router ospfv3 1
```

https://t.me/learningnets

```
 address-family ipv4 unicast
  passive-interface Loopback0
 exit-address-family
 !
 address-family ipv6 unicast
  passive-interface Loopback0
 exit-address-family
```

R3:

```
interface Loopback0
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.23
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.35
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.36
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
router ospfv3 1
 address-family ipv4 unicast
  passive-interface Loopback0
 exit-address-family
 !
 address-family ipv6 unicast
  passive-interface Loopback0
 exit-address-family
```

R4:

```
interface Loopback0
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.14
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.24
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.45
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
router ospfv3 1
 address-family ipv4 unicast
  passive-interface Loopback0
 exit-address-family
 !
 address-family ipv6 unicast
  passive-interface Loopback0
 exit-address-family
```

R5:

```
interface Loopback0
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.15
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.25
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.35
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.45
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.56
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
router ospfv3 1
 address-family ipv4 unicast
  passive-interface Loopback0
 exit-address-family
 !
 address-family ipv6 unicast
  passive-interface Loopback0
```

https://t.me/learningnets

```
 exit-address-family

R6:
interface Loopback0
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.26
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.36
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
interface GigabitEthernet1.56
 ospfv3 1 ipv6 area 0
 ospfv3 1 ipv4 area 0
!
router ospfv3 1
 address-family ipv4 unicast
  passive-interface Loopback0
 exit-address-family
 !
 address-family ipv6 unicast
  passive-interface Loopback0
 exit-address-family
```

Task 2.1 Verification

```
R2#show ospfv3 neighbor
 OSPFv3 1 address-family ipv4
 (router-id 10.255.255.2)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface 10.255.255.6
      1   FULL/DR         00:00:38    10              GigabitEthernet1.26 10.255.255.5
      1   FULL/DR         00:00:33    11              GigabitEthernet1.25 10.255.255.4
      1   FULL/DR         00:00:37    11              GigabitEthernet1.24 10.255.255.3
      1   FULL/DR         00:00:39    10              GigabitEthernet1.23 10.255.255.1
      1   FULL/BDR        00:00:34    10              GigabitEthernet1.12
 OSPFv3 1 address-family ipv6
 (router-id 10.255.255.2)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface 10.255.255.6
      1   FULL/DR         00:00:39    10              GigabitEthernet1.26 10.255.255.5
      1   FULL/DR         00:00:33    11              GigabitEthernet1.25 10.255.255.4
```

https://t.me/learningnets

```
R2#show ip route ospfv3

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 23 subnets, 3 masks
O        10.255.1.0/30 [110/2] via 10.255.4.2, 00:06:04, GigabitEthernet1.24
                       [110/2] via 10.255.0.1, 00:06:14, GigabitEthernet1.12
O        10.255.3.0/30 [110/2] via 10.255.5.2, 00:06:04, GigabitEthernet1.25
                       [110/2] via 10.255.0.1, 00:06:04, GigabitEthernet1.12
O        10.255.6.0/30 [110/2] via 10.255.5.2, 00:06:04, GigabitEthernet1.25
                       [110/2] via 10.255.4.2, 00:06:04, GigabitEthernet1.24
O        10.255.9.0/30 [110/2] via 10.255.7.2, 00:06:04, GigabitEthernet1.23
                       [110/2] via 10.255.5.2, 00:06:04, GigabitEthernet1.25
O        10.255.10.0/30 [110/2] via 10.255.8.2, 00:05:54, GigabitEthernet1.26
                        [110/2] via 10.255.5.2, 00:05:54, GigabitEthernet1.25
O        10.255.11.0/30 [110/2] via 10.255.8.2, 00:05:54, GigabitEthernet1.26
                        [110/2] via 10.255.7.2, 00:05:54, GigabitEthernet1.23O 10.255.255.1/32
 [110/1] via 10.255.0.1, 00:06:24, GigabitEthernet1.12O 10.255.255.3/32
 [110/1] via 10.255.7.2, 00:06:14, GigabitEthernet1.23O 10.255.255.4/32
 [110/1] via 10.255.4.2, 00:06:04, GigabitEthernet1.24O 10.255.255.5/32
 [110/1] via 10.255.5.2, 00:00:12, GigabitEthernet1.25O 10.255.255.6/32
 [110/1] via 10.255.8.2, 00:05:54, GigabitEthernet1.26
R2#show ipv6 route ospf
IPv6 Routing Table - default - 35 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       la - LISP alt, lr - LISP site-registrations, ld - LISP dyn-eid
       a - ApplicationO 2001:10:255:255::1/128
 [110/1]
     via FE80::250:56FF:FE8D:5443, GigabitEthernet1.12O 2001:10:255:255::3/128
 [110/1]
     via FE80::250:56FF:FE8D:5D84, GigabitEthernet1.23
```

https://t.me/learningnets

```
O 2001:10:255:255::4/128
 [110/1]
     via FE80::250:56FF:FE8D:4E77, GigabitEthernet1.24O 2001:10:255:255::5/128
 [110/1]
     via FE80::250:56FF:FE8D:44F2, GigabitEthernet1.25O 2001:10:255:255::6/128
 [110/1]
     via FE80::250:56FF:FE8D:7E27, GigabitEthernet1.26
```

Task 2.2 Core Security

```
R1 - R6:
key chain OSPF_KEYS
 key 1
   key-string CISCO
   cryptographic-algorithm hmac-sha-512
!
router ospfv3 1
 address-family ipv4 unicast
  area 0 authentication key-chain OSPF_KEYS
 exit-address-family
 !
 address-family ipv6 unicast
  area 0 authentication key-chain OSPF_KEYS
 exit-address-family
```

Task 2.2 Verification

```
R1#show ospfv3 interface gig1.12
GigabitEthernet1.12 is up, line protocol is up
  Link Local Address FE80::250:56FF:FE8D:5443, Interface ID 10 Internet Address 10.255.0.1/30
 Area 0
, Process ID 1, Instance ID 64, Router ID 10.255.255.1
  Network Type BROADCAST, Cost: 1 Cryptographic authentication enabled
 Sending SA: Key 1, Algorithm HMAC-SHA-512 - key chain OSPF_KEYS
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 10.255.255.2, local address FE80::250:56FF:FE8D:5837
  Backup Designated router (ID) 10.255.255.1, local address FE80::250:56FF:FE8D:5443
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:03
  Graceful restart helper support enabled
  Index 1/2/2, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
```

https://t.me/learningnets

```
  Last flood scan length is 1, maximum is 7
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 10.255.255.2  (Designated Router)
  Suppress hello for 0 neighbor(s)
GigabitEthernet1.12 is up, line protocol is up  Link Local Address FE80::250:56FF:FE8D:5443
, Interface ID 10 Area 0
, Process ID 1, Instance ID 0, Router ID 10.255.255.1
  Network Type BROADCAST, Cost: 1 Cryptographic authentication enabled
 Sending SA: Key 1, Algorithm HMAC-SHA-512 - key chain OSPF_KEYS

  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 10.255.255.2, local address FE80::250:56FF:FE8D:5837
  Backup Designated router (ID) 10.255.255.1, local address FE80::250:56FF:FE8D:5443
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:07
  Graceful restart helper support enabled
  Index 1/2/2, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 3
  Last flood scan time is 1 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 10.255.255.2  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

Task 2.3 Core Resource Optimization

```
R1:
interface Loopback0
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.12
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.14
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.15
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
```

https://t.me/learningnets

```
router ospfv3 1
 prefix-suppression

R2:
interface Loopback0
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.12
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.23
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.24
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.25
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.26
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
router ospfv3 1
 prefix-suppression

R3:
interface Loopback0
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.23
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.35
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.36
 ospfv3 1 ipv6 network point-to-point
```

https://t.me/learningnets

```
 ospfv3 1 ipv4 network point-to-point
!
router ospfv3 1
 prefix-suppression

R4:
interface Loopback0
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.14
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.24
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.45
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
router ospfv3 1
 prefix-suppression

R5:
interface Loopback0
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.15
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.25
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.35
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.45
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
```

https://t.me/learningnets

```
interface GigabitEthernet1.56
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
router ospfv3 1
 prefix-suppression

R6:
interface Loopback0
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.26
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.36
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
interface GigabitEthernet1.56
 ospfv3 1 ipv6 network point-to-point
 ospfv3 1 ipv4 network point-to-point
!
router ospfv3 1
 prefix-suppression
```

Task 2.3 Verification

```
R2#show ip route ospfv3
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 17 subnets, 3 masks O        10.255.255.1/32
 [110/1] via 10.255.0.1, 00:00:11, GigabitEthernet1.12 O        10.255.255.3/32
 [110/1] via 10.255.7.2, 00:09:08, GigabitEthernet1.23 O        10.255.255.4/32
 [110/1] via 10.255.4.2, 00:09:08, GigabitEthernet1.24 O        10.255.255.5/32
```

https://t.me/learningnets

```
R2#show ospfv3 database network



          OSPFv3 1 address-family ipv4 (router-id 10.255.255.2)


          OSPFv3 1 address-family ipv6 (router-id 10.255.255.2)
R2#show ospfv3 neighbor



          OSPFv3 1 address-family ipv4 (router-id 10.255.255.2)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface 10.255.255.6      0 FULL/  -
        00:00:33    10              GigabitEthernet1.26 10.255.255.5      0 FULL/  -
        00:00:37    11              GigabitEthernet1.25 10.255.255.4      0 FULL/  -
        00:00:36    11              GigabitEthernet1.24 10.255.255.3      0 FULL/  -
        00:00:34    10              GigabitEthernet1.23 10.255.255.1      0 FULL/  -
        00:00:34    10              GigabitEthernet1.12

          OSPFv3 1 address-family ipv6 (router-id 10.255.255.2)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface 10.255.255.6      0 FULL/  -
        00:00:39    10              GigabitEthernet1.26 10.255.255.5      0 FULL/  -
        00:00:35    11              GigabitEthernet1.25 10.255.255.4      0 FULL/  -
        00:00:35    11              GigabitEthernet1.24 10.255.255.3      0 FULL/  -
        00:00:35    10              GigabitEthernet1.23 10.255.255.1      0 FULL/  -
        00:00:30    10              GigabitEthernet1.12
```

Task 2.4 Core Convergence Optimization

```
R1:
interface GigabitEthernet1.12
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.14
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.15
 bfd interval 250 min_rx 250 multiplier 3
!
router ospfv3 1
 bfd all-interfaces

R2:
interface GigabitEthernet1.12
```

https://t.me/learningnets

```
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.23
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.24
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.25
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.26
 bfd interval 250 min_rx 250 multiplier 3
!
router ospfv3 1
 bfd all-interfaces
```

```
R3:
interface GigabitEthernet1.23
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.35
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.36
 bfd interval 250 min_rx 250 multiplier 3
!
router ospfv3 1
 bfd all-interfaces
```

```
R4:
interface GigabitEthernet1.14
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.24
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.45
 bfd interval 250 min_rx 250 multiplier 3
!
router ospfv3 1
 bfd all-interfaces
```

```
R5:
interface GigabitEthernet1.15
 bfd interval 250 min_rx 250 multiplier 3
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.25
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.35
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.45
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.56
 bfd interval 250 min_rx 250 multiplier 3
!
router ospfv3 1
 bfd all-interfaces

R6:
interface GigabitEthernet1.26
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.36
 bfd interval 250 min_rx 250 multiplier 3
!
interface GigabitEthernet1.56
 bfd interval 250 min_rx 250 multiplier 3
!
router ospfv3 1
 bfd all-interfaces
```

Task 2.4 Verification

R2#show bfd neighbors

IPv4 Sessions NeighAddr LD/RD RH/RS State Int 10.255.0.1 4101/4099 **Up Up Gi1.12** 10.255.4.2 4099/4098 **Up Up Gi1.24** 10.255.5.2 4098/4100 **Up Up Gi1.25** 10.255.7.2 4100/4099 **Up Up Gi1.23** 10.255.8.2 4097/4099 **Up Up Gi1.26**

IPv6 Sessions NeighAddr LD/RD RH/RS State Int FE80::250:56FF:FE8D:44F2 2/4 **Up Up Gi1.25** FE80::250:56FF:FE8D:4E77 3/2 **Up Up Gi1.24** FE80::250:56FF:FE8D:5443 5/3 **Up Up Gi1.12** FE80::250:56FF:FE8D:5D84 4/3 **Up Up Gi1.23** FE80::250:56FF:FE8D:7E27 1/3 **Up Up Gi1.26**

https://t.me/learningnets

Task 3.1 PE to CE OSPFv2 Routing

```
R1:
interface GigabitEthernet1.17
 ip ospf 1 area 0

R2:
interface GigabitEthernet1.28
 ip ospf 1 area 0

R4:
interface GigabitEthernet1.49
 ip ospf 1 area 0

R5:
interface GigabitEthernet1.135
 ip ospf 1 area 0
!
interface GigabitEthernet1.145
 ip ospf 1 area 0

R7:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.17
 ip ospf 1 area 0
!
interface GigabitEthernet1.78
 ip ospf 1 area 0
!
interface GigabitEthernet1.227
 ip ospf 1 area 0

R8:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.28
 ip ospf 1 area 0
!
interface GigabitEthernet1.78
 ip ospf 1 area 0
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.228
 ip ospf 1 area 0

R9:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.49
 ip ospf 1 area 0
!
interface GigabitEthernet1.109
 ip ospf 1 area 0

R10:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.102
 ip ospf 1 area 1
!
interface GigabitEthernet1.109
 ip ospf 1 area 0

R11:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.102
 ip ospf 1 area 1
!
interface GigabitEthernet1.109
 ip ospf 1 area 0

R12:
interface Loopback0
 ip ospf 1 area 1
!
interface GigabitEthernet1.102
 ip ospf 1 area 1

R13:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.100
```

https://t.me/learningnets

```
 ip ospf 1 area 0
!
interface GigabitEthernet1.135
 ip ospf 1 area 0

R14:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.100
 ip ospf 1 area 0
!
interface GigabitEthernet1.145
 ip ospf 1 area 0

R15:
interface Loopback0
 ip ospf 1 area 0
!
interface GigabitEthernet1.100
 ip ospf 1 area 0

SW2:
ip routing
!
interface Loopback0
 ip ospf 1 area 0
!
interface Vlan227
 ip ospf 1 area 0
!
interface Vlan228
 ip ospf 1 area 0
```

Task 3.1 Verification

```
R12#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
```

https://t.me/learningnets

```
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/32 is subnetted, 4 subnets
O IA
     10.255.255.9
           [110/3] via 172.30.102.11, 00:01:29, GigabitEthernet1.102
           [110/3] via 172.30.102.10, 00:01:26, GigabitEthernet1.102
O IA     10.255.255.10
           [110/2] via 172.30.102.10, 00:01:29, GigabitEthernet1.102
O IA     10.255.255.11
           [110/2] via 172.30.102.11, 00:01:29, GigabitEthernet1.102
      172.30.0.0/16 is variably subnetted, 4 subnets, 2 masks
O IA     172.30.49.0/24
           [110/3] via 172.30.102.11, 00:01:29, GigabitEthernet1.102
           [110/3] via 172.30.102.10, 00:01:26, GigabitEthernet1.102
O IA     172.30.109.0/24
           [110/2] via 172.30.102.11, 00:01:29, GigabitEthernet1.102
           [110/2] via 172.30.102.10, 00:01:29, GigabitEthernet1.102
R9#show ip ospf database

            OSPF Router with ID (10.255.255.9) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.255.255.9    10.255.255.9    166         0x80000004 0x00123A 3
10.255.255.10   10.255.255.10   167         0x80000003 0x00BCA0 2
10.255.255.11   10.255.255.11   167         0x80000003 0x00D088 2
172.30.49.4     172.30.49.4     175         0x80000002 0x001F12 1

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
172.30.49.4     172.30.49.4     175         0x80000001 0x0032F4
172.30.109.11   10.255.255.11   167         0x80000001 0x00A7F9
Summary Net Link States
 (Area 0)
```

https://t.me/learningnets

```
Link ID         ADV Router      Age         Seq#       Checksum 10.255.255.12    10.255.255.10
   152         0x80000001 0x000C05 10.255.255.12    10.255.255.11
   152         0x80000001 0x00060A 172.30.102.0     10.255.255.10
   210         0x80000001 0x00688E 172.30.102.0     10.255.255.11
   202         0x80000001 0x006293
```

Task 3.2 MPLS Label Distribution

```
R1:
interface GigabitEthernet1.12
 mpls ip
!
interface GigabitEthernet1.14
 mpls ip
!
interface GigabitEthernet1.15
 mpls ip

R2:
interface GigabitEthernet1.12
 mpls ip
!
interface GigabitEthernet1.23
 mpls ip
!
interface GigabitEthernet1.24
 mpls ip
!
interface GigabitEthernet1.25
 mpls ip
!
interface GigabitEthernet1.26
 mpls ip

R3:
interface GigabitEthernet1.23
 mpls ip
!
interface GigabitEthernet1.35
 mpls ip
!
interface GigabitEthernet1.36
 mpls ip
```

https://t.me/learningnets

```
R4:
interface GigabitEthernet1.14
 mpls ip
!
interface GigabitEthernet1.24
 mpls ip
!
interface GigabitEthernet1.45
 mpls ip

R5:
interface GigabitEthernet1.15
 mpls ip
!
interface GigabitEthernet1.25
 mpls ip
!
interface GigabitEthernet1.35
 mpls ip
!
interface GigabitEthernet1.45
 mpls ip
!
interface GigabitEthernet1.56
 mpls ip

R6:
interface GigabitEthernet1.26
 mpls ip
!
interface GigabitEthernet1.36
 mpls ip
!
interface GigabitEthernet1.56
 mpls ip
```

Task 3.2 Verification

```
R1#show mpls ldp neighbor
    Peer LDP Ident: 10.255.255.5:0; Local LDP Ident 10.255.255.1:0
TCP connection: 10.255.255.5.18276 - 10.255.255.1.646
State: Oper
; Msgs sent/rcvd: 30/33; Downstream
        Up time: 00:13:19
```

https://t.me/learningnets

```
        LDP discovery sources:
          GigabitEthernet1.15, Src IP addr: 10.255.3.2
        Addresses bound to peer LDP Ident:
          10.255.3.2      10.255.5.2      10.255.9.2      10.255.6.2
          10.255.10.1     192.168.1.5     10.255.255.5
    Peer LDP Ident: 10.255.255.2:0; Local LDP Ident 10.255.255.1:0
TCP connection: 10.255.255.2.20014 - 10.255.255.1.646
 State: Oper
; Msgs sent/rcvd: 31/34; Downstream
        Up time: 00:13:19
        LDP discovery sources:
          GigabitEthernet1.12, Src IP addr: 10.255.0.2
        Addresses bound to peer LDP Ident:
          10.255.0.2      169.254.20.0    10.255.7.1      10.255.4.1
          10.255.5.1      10.255.8.1      192.168.1.2     10.255.255.2
    Peer LDP Ident: 10.255.255.4:0; Local LDP Ident 10.255.255.1:0
TCP connection: 10.255.255.4.41091 - 10.255.255.1.646
 State: Oper
; Msgs sent/rcvd: 31/31; Downstream
        Up time: 00:13:17
        LDP discovery sources:
          GigabitEthernet1.14, Src IP addr: 10.255.1.2
        Addresses bound to peer LDP Ident:
          10.255.1.2      10.255.4.2      10.255.6.1      192.168.1.4
          10.255.255.4
R1#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface               17         17
10.255.255.3/32
  0             Gi1.12     10.255.0.2             18 10.255.255.3/32
  0             Gi1.15     10.255.3.2  18         Pop Label 10.255.255.4/32
  0             Gi1.14     10.255.1.2  19         Pop Label 10.255.255.5/32
  0             Gi1.15     10.255.3.2  20         20 10.255.255.6/32
  0             Gi1.12     10.255.0.2             20 10.255.255.6/32
  0             Gi1.15     10.255.3.2  21         Pop Label 10.255.255.2/32
  0             Gi1.12     10.255.0.2
```

Task 3.3 MPLS VPNv4 Routing

```
R1:
vrf definition SITE_A
 rd 10.255.255.1:1
 route-target export 1.20000:1
 route-target import 1.20000:1
```

https://t.me/learningnets

```
!
router ospf 1 vrf SITE_A
 redistribute bgp 1.20000 subnets
!
router bgp 1.20000
 address-family ipv4 vrf SITE_A
  redistribute ospf 1
 exit-address-family
```

R2:

```
vrf definition SITE_A
 rd 10.255.255.2:1
 route-target export 1.20000:1
 route-target import 1.20000:1
!
router ospf 1 vrf SITE_A
 redistribute bgp 1.20000 subnets
!
router bgp 1.20000
 address-family ipv4 vrf SITE_A
  redistribute ospf 1
 exit-address-family
```

R4:

```
vrf definition SITE_B
 rd 10.255.255.4:1
 route-target export 1.20000:1
 route-target import 1.20000:1
!
router ospf 1 vrf SITE_B
 redistribute bgp 1.20000 subnets
!
router bgp 1.20000
 address-family vpnv4
  neighbor 10.255.255.1 route-reflector-client
  neighbor 10.255.255.2 route-reflector-client
  neighbor 10.255.255.5 route-reflector-client
 exit-address-family
 !
 address-family ipv4 vrf SITE_B
  redistribute ospf 1
  maximum-paths ibgp 2
 exit-address-family
```

R5:

```
vrf definition SITE_C
```

https://t.me/learningnets

```
 rd 10.255.255.5:1
 route-target export 1.20000:1
 route-target import 1.20000:1
!
router ospf 1 vrf SITE_C
 redistribute bgp 1.20000 subnets
!
router bgp 1.20000
 address-family ipv4 vrf SITE_C
  redistribute ospf 1
  maximum-paths ibgp 2
 exit-address-family
```

Task 3.3 Verification

```
R5#show bgp vpnv4 unicast all summary
BGP router identifier 10.255.255.5, local AS number 1.20000
BGP table version is 99, main routing table version 99
44 network entries using 11264 bytes of memory
52 path entries using 6240 bytes of memory
2 multipath network entries and 4 multipath paths
14/14 BGP path/bestpath attribute entries using 3696 bytes of memory
2 BGP rrinfo entries using 80 bytes of memory
5 BGP extended community entries using 300 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 21580 total bytes of memory
BGP activity 44/0 prefixes, 52/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd 10.255.255.4
    4      1.20000      45      23       99    0    0 00:10:39 23
R5#show bgp vpnv4 unicast vrf SITE_C
BGP table version is 99, local router ID is 10.255.255.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 10.255.255.5:1 (default for vrf SITE_C)
 * i 10.255.255.7/32  10.255.255.2             3    100      0 ?
 *>i                  10.255.255.1             2    100      0 ?
 *>i 10.255.255.8/32  10.255.255.2             2    100      0 ?
```

https://t.me/learningnets

```
 * i                  10.255.255.1             3    100      0 ?
 *>i 10.255.255.9/32  10.255.255.4             2    100      0 ?
 *>i 10.255.255.10/32 10.255.255.4             3    100      0 ?
 *>i 10.255.255.11/32 10.255.255.4             3    100      0 ?
 *>i 10.255.255.12/32 10.255.255.4             4    100      0 ?
 *>  10.255.255.13/32 172.29.135.13            2         32768 ?
 *>  10.255.255.14/32 172.29.145.14            2         32768 ?
 *>  10.255.255.15/32 172.29.135.13            3         32768 ?*mi 10.255.255.22/32 10.255.255.2
            3    100      0 ?*>i                  10.255.255.1
            3    100      0 ?
 *>  172.29.100.0/24  172.29.135.13            2         32768 ?
 *>  172.29.135.0/24  0.0.0.0                  0         32768 ?
 *>  172.29.145.0/24  0.0.0.0                  0         32768 ?
 *>i 172.30.49.0/24   10.255.255.4             0    100      0 ?
 *>i 172.30.102.0/24  10.255.255.4             3    100      0 ?
 *>i 172.30.109.0/24  10.255.255.4             2    100      0 ?
 * i 172.31.17.0/24   10.255.255.2             3    100      0 ?
 *>i                  10.255.255.1             0    100      0 ?
 *>i 172.31.28.0/24   10.255.255.2             0    100      0 ?
 * i                  10.255.255.1             3    100      0 ?
 *mi 172.31.78.0/24   10.255.255.2             2    100      0 ?
 *>i                  10.255.255.1             2    100      0 ?
 * i 172.31.227.0/24  10.255.255.2             3    100      0 ?
 *>i                  10.255.255.1             2    100      0 ?
 *>i 172.31.228.0/24  10.255.255.2             2    100      0 ?
 * i                  10.255.255.1             3    100      0 ?
R5#show bgp vpnv4 unicast vrf SITE_C 10.255.255.22/32
BGP routing table entry for 10.255.255.5:1:10.255.255.22/32, version 94
Paths: (2 available, best #2, table SITE_C)
Multipath: iBGP
  Not advertised to any peer
  Refresh Epoch 2  Local, imported path from10.255.255.2:1:10.255.255.22/32
 (global)10.255.255.2
 (metric 1) (via default) from 10.255.255.4 (10.255.255.4)
      Origin incomplete, metric 3, localpref 100, valid, internal,multipath
(oldest)
      Extended Community: OSPF DOMAIN ID:0x0005:0x000000010200 RT:1.20000:1
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:172.31.28.2:0
 Originator: 10.255.255.2, Cluster list: 10.255.255.4
      mpls labels in/out nolabel/23
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 2  Local, imported path from10.255.255.1:1:10.255.255.22/32
 (global)10.255.255.1
 (metric 1) (via default) from 10.255.255.4 (10.255.255.4)
      Origin incomplete, metric 3, localpref 100, valid, internal,multipath
, best
```

https://t.me/learningnets

```
      Extended Community: OSPF DOMAIN ID:0x0005:0x000000010200 RT:1.20000:1
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:172.31.17.1:0
 Originator: 10.255.255.1, Cluster list: 10.255.255.4
      mpls labels in/out nolabel/24
      rx pathid: 0, tx pathid: 0x0
R9#traceroute 10.255.255.22 source lo0
Type escape sequence to abort.
Tracing the route to 10.255.255.22
VRF info: (vrf in name/id, vrf out name/id)
  1 172.30.49.4 4 msec 1 msec 1 msec 2 172.31.28.2
 [MPLS: Label 23 Exp 0] 2 msec 6 msec 4 msec
  3 172.31.28.8 18 msec 10 msec 27 msec
  4 172.31.228.22 8 msec *  3 msec
R9#traceroute 10.255.255.22 source gig1.109
Type escape sequence to abort.
Tracing the route to 10.255.255.22
VRF info: (vrf in name/id, vrf out name/id)
  1 172.30.49.4 4 msec 1 msec 1 msec 2 172.31.17.1
 [MPLS: Label 24 Exp 0] 2 msec 1 msec 4 msec
  3 172.31.17.7 24 msec 9 msec 10 msec
  4 172.31.227.22 10 msec *  4 msec
```

Task 4.1 OSPF Inter-Area Traffic Engineering

```
R10:
router ospf 1
 area 1 stub no-summary
 area 1 default-cost 1000

R11:
router ospf 1
 area 1 stub no-summary

R12:
router ospf 1
 area 1 stub
```

Task 4.1 Verification

```
R12#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
```

https://t.me/learningnets

```
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is 172.30.102.11 to network 0.0.0.0
 O*IA  0.0.0.0/0 [110/2] via 172.30.102.11
, 00:00:04, GigabitEthernet1.102
R12#traceroute 10.255.255.22
Type escape sequence to abort.
Tracing the route to 10.255.255.22
VRF info: (vrf in name/id, vrf out name/id) 1 172.30.102.11
 4 msec 1 msec 1 msec
  2 172.30.109.9 1 msec 1 msec 1 msec
  3 172.30.49.4 2 msec 1 msec 2 msec
  4 172.31.17.1 [MPLS: Label 24 Exp 0] 2 msec 11 msec 21 msec
  5 172.31.17.7 17 msec 9 msec 10 msec
  6 172.31.227.22 10 msec *  4 msec

R11#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R11(config)#int gig1R11(config-if)#shut


R12#%OSPF-5-ADJCHG: Process 1, Nbr 10.255.255.11 on GigabitEthernet1.102 from FULL to DOWN
, Neighbor Down: Dead timer expired
R12#show ip route ospf

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is 172.30.102.10 to network 0.0.0.0
 O*IA  0.0.0.0/0 [110/1001] via 172.30.102.10
, 00:00:02, GigabitEthernet1.102
R12#traceroute 10.255.255.22
```

https://t.me/learningnets

```
Type escape sequence to abort.
Tracing the route to 10.255.255.22
VRF info: (vrf in name/id, vrf out name/id) 1 172.30.102.10
 3 msec 2 msec 1 msec
  2 172.30.109.9 1 msec 1 msec 1 msec
  3 172.30.49.4 2 msec 1 msec 3 msec
  4 172.31.17.1 [MPLS: Label 24 Exp 0] 10 msec 18 msec 20 msec
  5 172.31.17.7 17 msec 10 msec 9 msec
  6 172.31.227.22 10 msec *  5 msec
```

Task 4.2 OSPF Intra-Area Traffic Engineering

```
R13:
interface GigabitEthernet1.100
 ip ospf network point-to-multipoint non-broadcast
!
router ospf 1
 neighbor 172.29.100.14
 neighbor 172.29.100.15 cost 1000

R14:
interface GigabitEthernet1.100
 ip ospf network point-to-multipoint non-broadcast
!
router ospf 1
 neighbor 172.29.100.13
 neighbor 172.29.100.15

R15:
interface GigabitEthernet1.100
 ip ospf network point-to-multipoint non-broadcast
!
router ospf 1
 neighbor 172.29.100.13
 neighbor 172.29.100.14 cost 1000
```

Task 4.2 Verification

```
R15#show ip route 10.255.255.12
Routing entry for 10.255.255.12/32
  Known via "ospf 1", distance 110, metric 6, type inter area
  Last update from 172.29.100.13 on GigabitEthernet1.100, 00:01:35 ago
```

https://t.me/learningnets

```
  Routing Descriptor Blocks:   *172.29.100.13
, from 172.29.145.5, 00:01:35 ago, via GigabitEthernet1.100
      Route metric is 6, traffic share count is 1
R5#show ip route vrf SITE_C 10.255.255.15

Routing Table: SITE_C
Routing entry for 10.255.255.15/32
  Known via "ospf 1", distance 110, metric 3, type intra area
  Redistributing via bgp 1.20000
  Advertised by bgp 1.20000
  Last update from 172.29.145.14 on GigabitEthernet1.145, 00:01:58 ago
  Routing Descriptor Blocks:   *172.29.145.14
, from 10.255.255.15, 00:01:58 ago, via GigabitEthernet1.145
      Route metric is 3, traffic share count is 1
R15#traceroute 10.255.255.12
Type escape sequence to abort.
Tracing the route to 10.255.255.12
VRF info: (vrf in name/id, vrf out name/id)  1 172.29.100.13
 3 msec 2 msec 1 msec  2 172.29.135.5
 1 msec 1 msec 1 msec
  3 172.30.49.4 [MPLS: Label 19 Exp 0] 6 msec 2 msec 2 msec
  4 172.30.49.9 11 msec 10 msec 10 msec
  5 172.30.109.10 10 msec 10 msec 14 msec
  6 172.30.102.12 15 msec *  3 msec
R12#traceroute 10.255.255.15
Type escape sequence to abort.
Tracing the route to 10.255.255.15
VRF info: (vrf in name/id, vrf out name/id)
  1 172.30.102.11 3 msec 2 msec 1 msec
  2 172.30.109.9 1 msec 1 msec 1 msec
  3 172.30.49.4 2 msec 1 msec 1 msec  4 172.29.145.5
 [MPLS: Label 30 Exp 0] 3 msec 10 msec 19 msec  5 172.29.145.14
 17 msec 5 msec 10 msec
  6 172.29.100.15 9 msec *  3 msec
```

Task 5.1 BGP Peering

```
R2, R3, R6:
router bgp 1.20000
 address-family ipv4
  neighbor 10.255.255.4 next-hop-self
 exit-address-family

R4:
```

https://t.me/learningnets

```
router bgp 1.20000
 address-family ipv4
  neighbor 10.255.255.2 route-reflector-client
  neighbor 10.255.255.3 route-reflector-client
  neighbor 10.255.255.6 route-reflector-client
 exit-address-family

R16:
router bgp 1831
 network 10.255.255.16 mask 255.255.255.255
 neighbor 169.254.160.1 remote-as 10000

R17:
router bgp 1831
 network 10.255.255.17 mask 255.255.255.255
 neighbor 169.254.170.1 remote-as 10000

R18:
router bgp 1832
 network 10.255.255.18 mask 255.255.255.255
 neighbor 169.254.180.1 remote-as 10000
 neighbor 169.254.181.1 remote-as 30000

R19:
router bgp 1833
 network 10.255.255.19 mask 255.255.255.255
 neighbor 169.254.190.1 remote-as 30000

R20:
router bgp 1834
 network 10.255.255.20 mask 255.255.255.255
 neighbor 169.254.200.1 remote-as 30000
```

Task 5.1 Verification

```
R2#show bgp ipv4 unicast regexp _183[0-9]$
BGP table version is 102, local router ID is 10.255.255.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
```

https://t.me/learningnets

```
 *>  10.255.255.16/32 169.254.20.1                          0 10000 1831 i
 *>  10.255.255.17/32 169.254.20.1                          0 10000 1831 i
 *>  10.255.255.18/32 169.254.20.1                          0 10000 1832 i
 *   10.255.255.19/32 169.254.20.1                          0 10000 1832 30000 1833 i
 *>i                  10.255.255.6             0    100      0 30000 1833 i
 *   10.255.255.20/32 169.254.20.1                          0 10000 1832 30000 1834 i
 *>i                  10.255.255.6             0    100      0 30000 1834 i
R3#show bgp ipv4 unicast regexp _183[0-9]$
BGP table version is 104, local router ID is 10.255.255.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 10.255.255.16/32 10.255.255.2             0    100      0 10000 1831 i
 *>                  169.254.30.1                           0 10000 1831 i
 * i 10.255.255.17/32 10.255.255.2             0    100      0 10000 1831 i
 *>                  169.254.30.1                           0 10000 1831 i
 *>  10.255.255.18/32 169.254.30.1                          0 10000 1832 i
 * i                  10.255.255.2             0    100      0 10000 1832 i
 *   10.255.255.19/32 169.254.30.1                          0 10000 1832 30000 1833 i
 *>i                  10.255.255.6             0    100      0 30000 1833 i
 *   10.255.255.20/32 169.254.30.1                          0 10000 1832 30000 1834 i
 *>i                  10.255.255.6             0    100      0 30000 1834 i
R4#show bgp ipv4 unicast regexp _183[0-9]$
BGP table version is 259, local router ID is 10.255.255.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path * i 10.255.255.16/32 10.255.255.3
             0    100      0 10000 1831 i *>i 10.255.255.2
             0    100      0 10000 1831 i * i 10.255.255.17/32 10.255.255.3
             0    100      0 10000 1831 i *>i 10.255.255.2
             0    100      0 10000 1831 i * i 10.255.255.18/32 10.255.255.3
             0    100      0 10000 1832 i *>i 10.255.255.2
             0    100      0 10000 1832 i * i 10.255.255.6
             0    100      0 30000 1832 i *>i 10.255.255.19/32 10.255.255.6
             0    100      0 30000 1833 i *>i 10.255.255.20/32 10.255.255.6
             0    100      0 30000 1834 i
R6#show bgp ipv4 unicast regexp _183[0-9]$
```

https://t.me/learningnets

```
BGP table version is 84, local router ID is 10.255.255.6
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 10.255.255.16/32 10.255.255.2             0    100      0 10000 1831 i
 *>i 10.255.255.17/32 10.255.255.2             0    100      0 10000 1831 i
 * i 10.255.255.18/32 10.255.255.2             0    100      0 10000 1832 i
 *>                  169.254.60.1                           0 30000 1832 i
 *>  10.255.255.19/32 169.254.60.1                           0 30000 1833 i
 *>  10.255.255.20/32 169.254.60.1                           0 30000 1834 i
R16#show bgp ipv4 unicast regexp _183[0-9]$
BGP table version is 64, local router ID is 10.255.255.16
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.255.255.18/32 169.254.160.1                          0 10000 1832 i
 *>  10.255.255.19/32 169.254.160.1                          0 10000 1832 30000 1833 i
 *>  10.255.255.20/32 169.254.160.1                          0 10000 1832 30000 1834 i
R17#show bgp ipv4 unicast regexp _183[0-9]$
BGP table version is 64, local router ID is 10.255.255.17
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.255.255.18/32 169.254.170.1                          0 10000 1832 i
 *>  10.255.255.19/32 169.254.170.1                          0 10000 1832 30000 1833 i
 *>  10.255.255.20/32 169.254.170.1                          0 10000 1832 30000 1834 i
R18#show bgp ipv4 unicast regexp _183[0-9]$
BGP table version is 65, local router ID is 10.255.255.18
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

https://t.me/learningnets

```
     Network          Next Hop            Metric LocPrf Weight Path
 *   10.255.255.16/32 169.254.181.1                           0 30000 85536 10000 1831 i
 *>                   169.254.180.1                           0 10000 1831 i
 *   10.255.255.17/32 169.254.181.1                           0 30000 85536 10000 1831 i
 *>                   169.254.180.1                           0 10000 1831 i
 *>  10.255.255.19/32 169.254.181.1                           0 30000 1833 i
 *>  10.255.255.20/32 169.254.181.1                           0 30000 1834 i
R19#show bgp ipv4 unicast regexp _183[0-9]$
BGP table version is 65, local router ID is 10.255.255.19
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.255.255.16/32 169.254.190.1                           0 30000 85536 10000 1831 i
 *>  10.255.255.17/32 169.254.190.1                           0 30000 85536 10000 1831 i
 *>  10.255.255.18/32 169.254.190.1                           0 30000 1832 i
 *>  10.255.255.20/32 169.254.190.1                           0 30000 1834 i
R20#show bgp ipv4 unicast regexp _183[0-9]$

BGP table version is 65, local router ID is 10.255.255.20
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.255.255.16/32 169.254.200.1                           0 30000 85536 10000 1831 i
 *>  10.255.255.17/32 169.254.200.1                           0 30000 85536 10000 1831 i
 *>  10.255.255.18/32 169.254.200.1                           0 30000 1832 i
 *>  10.255.255.19/32 169.254.200.1                           0 30000 1833 i
```

Task 5.2 BGP Transit Filtering

https://t.me/learningnets

```
R18:
router bgp 1832
 neighbor 169.254.180.1 filter-list 1 out
 neighbor 169.254.181.1 filter-list 1 out
!
ip as-path access-list 1 permit ^$
```

Task 5.2 Verification

Before transit filter is applied:

```
R18#show bgp ipv4 unicast neighbors 169.254.180.1 advertised-routes
BGP table version is 65, local router ID is 10.255.255.18
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  4.128.0.0/10     169.254.181.1            0             0 30000 ?
 *>  8.0.0.0/9        169.254.181.1            0             0 30000 ?
 *>  8.96.30.0/24     169.254.180.1            0             0 10000 ?
 *>  10.192.0.0/10    169.254.181.1            0             0 30000 ?
 *>  10.255.255.16/32 169.254.180.1                          0 10000 1831 i
 *>  10.255.255.17/32 169.254.180.1                          0 10000 1831 i
 *>  10.255.255.18/32 0.0.0.0                  0         32768 i
 *>  10.255.255.19/32 169.254.181.1                          0 30000 1833 i
 *>  10.255.255.20/32 169.254.181.1                          0 30000 1834 i
 <snip>
Total number of prefixes 63

R18#show bgp ipv4 unicast neighbors 169.254.181.1 advertised-routes
BGP table version is 65, local router ID is 10.255.255.18
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  4.128.0.0/10     169.254.181.1            0             0 30000 ?
 *>  8.0.0.0/9        169.254.181.1            0             0 30000 ?
```

https://t.me/learningnets

```
 *>  8.96.30.0/24     169.254.180.1            0             0 10000 ?
 *>  10.192.0.0/10    169.254.181.1            0             0 30000 ?
 *>  10.255.255.16/32 169.254.180.1                          0 10000 1831 i
 *>  10.255.255.17/32 169.254.180.1                          0 10000 1831 i
 *>  10.255.255.18/32 0.0.0.0                  0         32768 i
 *>  10.255.255.19/32 169.254.181.1                          0 30000 1833 i
 *>  10.255.255.20/32 169.254.181.1                          0 30000 1834 i
 <snip>
Total number of prefixes 63
```

After transit filter is applied:

```
R18#show bgp ipv4 unicast neighbors 169.254.180.1 advertised-routes
BGP table version is 65, local router ID is 10.255.255.18
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.255.255.18/32 0.0.0.0                  0         32768 i
Total number of prefixes 1
R18#show bgp ipv4 unicast neighbors 169.254.181.1 advertised-routes
BGP table version is 65, local router ID is 10.255.255.18
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.255.255.18/32 0.0.0.0                  0         32768 i
Total number of prefixes 1
```

Task 5.3 BGP Traffic Engineering

```
R18:
router bgp 1832
 neighbor 169.254.180.1 route-map PREFER_ISP_A_FOR_OUTBOUND_TRAFFIC in
 neighbor 169.254.181.1 route-map PREFER_ISP_A_FOR_INBOUND_TRAFFIC out
```

https://t.me/learningnets

```
!
route-map PREFER_ISP_A_FOR_INBOUND_TRAFFIC permit 10
 set as-path prepend 1832 1832 1832 1832 1832
!
route-map PREFER_ISP_A_FOR_OUTBOUND_TRAFFIC permit 10
 set local-preference 200
```

Task 5.3 Verification

Before traffic engineering:

```
R18#traceroute 10.255.255.20 source lo0
Type escape sequence to abort.
Tracing the route to 10.255.255.20
VRF info: (vrf in name/id, vrf out name/id)
  1 169.254.181.1 1 msec 1 msec 1 msec
  2 169.254.200.0 2 msec *  3 msec
R20#traceroute 10.255.255.18 source lo0

Type escape sequence to abort.
Tracing the route to 10.255.255.18
VRF info: (vrf in name/id, vrf out name/id)
  1 169.254.200.1 5 msec 1 msec 1 msec
  2 169.254.181.0 1 msec *  3 msec
```

After traffic engineering:

```
R18#traceroute 10.255.255.20 source lo0
Type escape sequence to abort.
Tracing the route to 10.255.255.20
VRF info: (vrf in name/id, vrf out name/id)
  1 169.254.180.1 4 msec 2 msec 1 msec
  2 169.254.20.0 1 msec 1 msec 1 msec
  3 10.255.8.2 [AS 30000] 1 msec 1 msec 2 msec
  4 169.254.60.1 9 msec 9 msec 10 msec
  5 169.254.200.0 9 msec *  4 msec
R20#traceroute 10.255.255.18 source lo0
Type escape sequence to abort.
Tracing the route to 10.255.255.18
VRF info: (vrf in name/id, vrf out name/id)
  1 169.254.200.1 4 msec 1 msec 6 msec
  2 169.254.60.0 2 msec 1 msec 1 msec
  3 10.255.8.1 [AS 30000] 2 msec 1 msec 2 msec
```

https://t.me/learningnets

```
  4 169.254.20.1 4 msec 9 msec 10 msec
  5 169.254.180.0 9 msec *  4 msec
R18#show bgp ipv4 unicast 10.255.255.20
BGP routing table entry for 10.255.255.20/32, version 106
Paths: (2 available, best #1, table default)
  Not advertised to any peer
  Refresh Epoch 5
  10000 85536 30000 1834
    169.254.180.1 from 169.254.180.1 (10.255.255.15)      Origin IGP, localpref 200, valid, external,
best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 3
  30000 1834
    169.254.181.1 from 169.254.181.1 (10.255.255.12)      Origin IGP, localpref 100
, valid, external
      rx pathid: 0, tx pathid: 0
R12#show bgp vrf ISPB vpnv4 unicast 10.255.255.18/32
BGP routing table entry for 10000:10000:10.255.255.18/32, version 218
Paths: (2 available, best #1, table ISPB)
  Advertised to update-groups:
     1
  Refresh Epoch 1 1.20000 10000 1832
    169.254.60.0 (via vrf ISPB) from 169.254.60.0 (10.255.255.6)
      Origin IGP, localpref 100, valid, external, best
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 3 1832 1832 1832 1832 1832 1832

    169.254.181.0 (via vrf ISPB) from 169.254.181.0 (10.255.255.18)
      Origin IGP, metric 0, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
```

Task 6.1 DMVPN

```
R16:
interface Tunnel1
 ip address 183.100.1.16 255.255.255.0
 ip nhrp authentication DMVPN1
 ip nhrp map multicast dynamic
 ip nhrp network-id 1
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 1

R17:
```

https://t.me/learningnets

```
interface Tunnel2
 ip address 183.100.2.17 255.255.255.0
 ip nhrp authentication DMVPN2
 ip nhrp map multicast dynamic
 ip nhrp network-id 2
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2

R18:
interface Tunnel1
 ip address 183.100.1.18 255.255.255.0
 ip nhrp authentication DMVPN1
 ip nhrp map multicast 10.255.255.16
 ip nhrp map 183.100.1.16 10.255.255.16
 ip nhrp network-id 1
 ip nhrp nhs 183.100.1.16
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 1
!
interface Tunnel2
 ip address 183.100.2.18 255.255.255.0
 ip nhrp authentication DMVPN2
 ip nhrp map multicast 10.255.255.17
 ip nhrp map 183.100.2.17 10.255.255.17
 ip nhrp network-id 2
 ip nhrp nhs 183.100.2.17
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2

R19:
interface Tunnel1
 ip address 183.100.1.19 255.255.255.0
 ip nhrp authentication DMVPN1
 ip nhrp map multicast 10.255.255.16
 ip nhrp map 183.100.1.16 10.255.255.16
 ip nhrp network-id 1
 ip nhrp nhs 183.100.1.16
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 1
!
interface Tunnel2
 ip address 183.100.2.19 255.255.255.0
```

https://t.me/learningnets

```
 ip nhrp authentication DMVPN2
 ip nhrp map multicast 10.255.255.17
 ip nhrp map 183.100.2.17 10.255.255.17
 ip nhrp network-id 2
 ip nhrp nhs 183.100.2.17
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2

R20:
interface Tunnel1
 ip address 183.100.1.20 255.255.255.0
 ip nhrp authentication DMVPN1
 ip nhrp map multicast 10.255.255.16
 ip nhrp map 183.100.1.16 10.255.255.16
 ip nhrp network-id 1
 ip nhrp nhs 183.100.1.16
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 1
!
interface Tunnel2
 ip address 183.100.2.20 255.255.255.0
 ip nhrp authentication DMVPN2
 ip nhrp map multicast 10.255.255.17
 ip nhrp map 183.100.2.17 10.255.255.17
 ip nhrp network-id 2
 ip nhrp nhs 183.100.2.17
 tunnel source Loopback0
 tunnel mode gre multipoint
 tunnel key 2
```

Task 6.1 Verification

```
R16#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================
```

https://t.me/learningnets

```
Interface: Tunnel1, IPv4 NHRP Details
Type:Hub, NHRP Peers:3,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----      1
10.255.255.18      183.100.1.18    UP 00:00:27     D
     1 10.255.255.19      183.100.1.19    UP 00:00:19     D
     1 10.255.255.20      183.100.1.20    UP 00:00:10     D
R17#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
============================================================================

Interface: Tunnel2, IPv4 NHRP Details
Type:Hub, NHRP Peers:3,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----      1
10.255.255.18      183.100.2.18    UP 00:00:27     D
     1 10.255.255.19      183.100.2.19    UP 00:00:19     D
     1 10.255.255.20      183.100.2.20    UP 00:00:10     D
R18#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
============================================================================
 Interface: Tunnel1
, IPv4 NHRP Details
Type:Spoke, NHRP Peers:1,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----      1
10.255.255.16      183.100.1.16    UP 00:00:27     S
 Interface: Tunnel2
, IPv4 NHRP Details
Type:Spoke, NHRP Peers:1,
```

https://t.me/learningnets

```
 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----      1
10.255.255.17      183.100.2.17    UP 00:00:27     S
R18#show ip route eigrp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
D        10.255.255.23/32 [90/79365120] via 183.100.2.17, 00:01:05, Tunnel2
 [90/79365120] via 183.100.1.16, 00:01:05, Tunnel1

      183.16.0.0/24 is subnetted, 1 subnets
D        183.16.23.0 [90/76805120] via 183.100.1.16, 00:01:05, Tunnel1
      183.17.0.0/24 is subnetted, 1 subnets
D        183.17.23.0 [90/76805120] via 183.100.2.17, 00:01:05, Tunnel2
```

Task 6.2 DMVPN over IPsec

```
R16:
crypto isakmp policy 10
 encr 3des
 hash sha
 authentication pre-share
 group 5
!
crypto isakmp key DMVPNPASS address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-128-MD5 esp-aes 128 esp-md5-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-128-MD5
!
interface Tunnel1
 tunnel protection ipsec profile DMVPN_PROFILE
```

https://t.me/learningnets

```
R17:
crypto isakmp policy 10
 encr 3des
 hash sha
 authentication pre-share
 group 5
!
crypto isakmp key DMVPNPASS address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-128-MD5 esp-aes 128 esp-md5-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-128-MD5
!
interface Tunnel2
 tunnel protection ipsec profile DMVPN_PROFILE

R18:
crypto isakmp policy 10
 encr 3des
 hash sha
 authentication pre-share
 group 5
!
crypto isakmp key DMVPNPASS address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-128-MD5 esp-aes 128 esp-md5-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-128-MD5
!
interface Tunnel1
 tunnel protection ipsec profile DMVPN_PROFILE
!
interface Tunnel2
 tunnel protection ipsec profile DMVPN_PROFILE

R19:
crypto isakmp policy 10
 encr 3des
 hash sha
 authentication pre-share
 group 5
```

https://t.me/learningnets

```
!
crypto isakmp key DMVPNPASS address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-128-MD5 esp-aes 128 esp-md5-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-128-MD5
!
interface Tunnel1
 tunnel protection ipsec profile DMVPN_PROFILE
!
interface Tunnel2
 tunnel protection ipsec profile DMVPN_PROFILE

R20:
crypto isakmp policy 10
 encr 3des
 hash sha
 authentication pre-share
 group 5
!
crypto isakmp key DMVPNPASS address 0.0.0.0
!
crypto ipsec transform-set ESP-AES-128-MD5 esp-aes 128 esp-md5-hmac
 mode transport
!
crypto ipsec profile DMVPN_PROFILE
 set transform-set ESP-AES-128-MD5
!
interface Tunnel1
 tunnel protection ipsec profile DMVPN_PROFILE
!
interface Tunnel2
 tunnel protection ipsec profile DMVPN_PROFILE
```

Task 6.2 Verification

```
R18#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status10.255.255.18   10.255.255.16   QM_IDLE
           1004 ACTIVE10.255.255.18   10.255.255.17   QM_IDLE
           1003 ACTIVE
10.255.255.17   10.255.255.18   QM_IDLE           1001 ACTIVE
```

https://t.me/learningnets

```
10.255.255.16   10.255.255.18   QM_IDLE           1002 ACTIVE

IPv6 Crypto ISAKMP SA
R18#show crypto ipsec sa
interface: Tunnel1
    Crypto map tag: Tunnel1-head-0, local addr 10.255.255.18

   protected vrf: (none)   local  ident (addr/mask/prot/port): (10.255.255.18/255.255.255.255/47
/0)   remote ident (addr/mask/prot/port): (10.255.255.16/255.255.255.255/47
/0)
   current_peer 10.255.255.16 port 500
     PERMIT, flags={origin_is_acl,} #pkts encaps: 24, #pkts encrypt: 24, #pkts digest: 24
 #pkts decaps: 24, #pkts decrypt: 24, #pkts verify: 24
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 10.255.255.18, remote crypto endpt.: 10.255.255.16
     plaintext mtu 1458, path mtu 1500, ip mtu 1500, ip mtu idb (none)
     current outbound spi: 0x3E76A0CF(1047961807)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0xF24704FD(4064740605) transform: esp-aes esp-md5-hmac
 , in use settings ={Transport, }
        conn id: 2003, flow_id: CSR:3, sibling_flags FFFFFFFF80004008, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/3557)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)
      spi: 0x1DE17043(501313603)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 2007, flow_id: CSR:7, sibling_flags FFFFFFFF80000008, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4607998/3561)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x494465A3(1229219235)
```

https://t.me/learningnets

```
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 2004, flow_id: CSR:4, sibling_flags FFFFFFFF80004008, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4608000/3557)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)
      spi: 0x3E76A0CF(1047961807)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 2008, flow_id: CSR:8, sibling_flags FFFFFFFF80000008, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4607998/3561)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:

interface: Tunnel2
    Crypto map tag: Tunnel2-head-0, local addr 10.255.255.18

   protected vrf: (none)   local  ident (addr/mask/prot/port): (10.255.255.18/255.255.255.255/47
/0)   remote ident (addr/mask/prot/port): (10.255.255.17/255.255.255.255/47
/0)
   current_peer 10.255.255.17 port 500
     PERMIT, flags={origin_is_acl,} #pkts encaps: 22, #pkts encrypt: 22, #pkts digest: 22
 #pkts decaps: 24, #pkts decrypt: 24, #pkts verify: 24
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 10.255.255.18, remote crypto endpt.: 10.255.255.17
     plaintext mtu 1458, path mtu 1500, ip mtu 1500, ip mtu idb (none)
     current outbound spi: 0x5C59BFA6(1549385638)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0xF6FA1B7D(4143586173) transform: esp-aes esp-md5-hmac
 , in use settings ={Transport, }

        conn id: 2005, flow_id: CSR:5, sibling_flags FFFFFFFF80000008, crypto map: Tunnel2-head-0
        sa timing: remaining key lifetime (k/sec): (4607998/3557)
        IV size: 16 bytes
```

https://t.me/learningnets

```
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

    inbound ah sas:

    inbound pcp sas:

    outbound esp sas:
     spi: 0x5C59BFA6(1549385638)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 2006, flow_id: CSR:6, sibling_flags FFFFFFFF80000008, crypto map: Tunnel2-head-0
        sa timing: remaining key lifetime (k/sec): (4607998/3557)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

    outbound ah sas:

    outbound pcp sas:
```

Task 6.3 Routing over DMVPN

```
R16:
interface Tunnel1
 delay 1000
!
router eigrp SITE_W
 !
 address-family ipv4 unicast autonomous-system 1831
  !
  af-interface Tunnel1
   no next-hop-self
   no split-horizon
  exit-af-interface

R17:
interface Tunnel2
 delay 2000
!
router eigrp SITE_W
 !
 address-family ipv4 unicast autonomous-system 1831
  !
```

https://t.me/learningnets

```
  af-interface Tunnel2
   no next-hop-self
   no split-horizon
  exit-af-interface

R18:
interface Tunnel1
 delay 1000
!
interface Tunnel2
 delay 2000

R19:
interface Tunnel1
 delay 1000
!
interface Tunnel2
 delay 2000

R20:
interface Tunnel1
 delay 1000
!
interface Tunnel2
 delay 2000
```

Task 6.3 Verification

Traffic from R20 to SW3 prefers to use R16.

```
R20#traceroute 10.255.255.23 source gig1.201
Type escape sequence to abort.
Tracing the route to 10.255.255.23
VRF info: (vrf in name/id, vrf out name/id) 1 183.100.1.16
 4 msec 2 msec 7 msec
  2 183.16.23.23 33 msec *  5 msec
```

Traffic between spokes routes over on-demand GRE tunnels.

```
R20#traceroute 183.18.100.100 source gig1.201
Type escape sequence to abort.
Tracing the route to 183.18.100.100
VRF info: (vrf in name/id, vrf out name/id)
  1  *
```

https://t.me/learningnets

```
183.100.1.18
 3 msec 3 msec
  2 183.18.100.100 5 msec *  4 msec

R20#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
10.255.255.17   10.255.255.20   QM_IDLE           1002 ACTIVE
10.255.255.20   10.255.255.18   QM_IDLE           1005 ACTIVE

10.255.255.16   10.255.255.20   QM_IDLE           1001 ACTIVE
10.255.255.20   10.255.255.17   QM_IDLE           1003 ACTIVE
10.255.255.20   10.255.255.16   QM_IDLE           1004 ACTIVE

IPv6 Crypto ISAKMP SA
```

If R16 fails, traffic is rerouted to R17:

```
R16#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R16(config)#int gig1R16(config-if)#shut
R16(config-if)#

R20#%DUAL-5-NBRCHANGE: EIGRP-IPv4 1831:Neighbor 183.100.1.16 (Tunnel1) is down
: holding time expired
R20#traceroute 10.255.255.23 source gig1.201

Type escape sequence to abort.
Tracing the route to 10.255.255.23
VRF info: (vrf in name/id, vrf out name/id)1 183.100.2.17
 7 msec 3 msec 5 msec
  2 183.17.23.23 34 msec *  7 msec
```

Task 7.1 Multicast over GRE

```
R7:
ip multicast-routing distributed
!
interface Tunnel79
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 10.255.255.9
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.227
 ip pim sparse-mode

R8:
ip multicast-routing distributed
!
interface Tunnel89
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 10.255.255.9
!
interface GigabitEthernet1.228
 ip pim sparse-mode

R9:
ip multicast-routing distributed
!
interface Tunnel79
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 10.255.255.7
!
interface Tunnel89
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 10.255.255.8
!
interface Tunnel915
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 10.255.255.15
 ip pim sparse-mode
!
interface GigabitEthernet1.109
 ip pim sparse-mode

R10:
ip multicast-routing distributed
!
interface GigabitEthernet1.102
 ip pim sparse-mode
```

https://t.me/learningnets

```
!
interface GigabitEthernet1.109
 ip pim sparse-mode

R11:
ip multicast-routing distributed
!
interface GigabitEthernet1.102
 ip pim sparse-mode
!
interface GigabitEthernet1.109
 ip pim sparse-mode

R12:
ip multicast-routing distributed
!
interface GigabitEthernet1.102
 ip pim sparse-mode

R13:
ip multicast-routing distributed
!
interface GigabitEthernet1.100
 ip pim sparse-mode

R14:
ip multicast-routing distributed
!
interface GigabitEthernet1.100
 ip pim sparse-mode

R15:
ip multicast-routing distributed
!
interface Tunnel915
 ip unnumbered Loopback0
 ip pim sparse-mode
 tunnel source Loopback0
 tunnel destination 10.255.255.9
 ip pim sparse-mode
!
interface GigabitEthernet1.100
 ip pim sparse-mode
```

https://t.me/learningnets

Task 7.1 Verification

```
R9#show ip pim neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable,
      L - DR Load-balancing Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode 10.255.255.7       Tunnel79
                 00:15:52/00:01:41 v2    1 / S P G 172.30.109.11     GigabitEthernet1.109
     00:08:37/00:01:29 v2    1 / DR S P G 172.30.109.10     GigabitEthernet1.109
     00:08:41/00:01:27 v2    1 / S P G 10.255.255.8      Tunnel89
                 00:13:51/00:01:43 v2    1 / S P G 10.255.255.15     Tunnel915
                00:11:52/00:01:39 v2    1 / S P G
```

Task 7.2 Multicast RP Distribution

```
R7:
ip pim autorp listener
!
ip mroute 0.0.0.0 0.0.0.0 Tunnel79

R8:
ip pim autorp listener
!
ip mroute 0.0.0.0 0.0.0.0 Tunnel89

R9:
interface Loopback0
 ip pim sparse-mode
!
ip pim autorp listener
!
ip pim send-rp-announce Loopback0 scope 255 interval 5
ip pim send-rp-discovery Loopback0 scope 255 interval 5

R10:
ip pim autorp listener

R11:
ip pim autorp listener
```

https://t.me/learningnets

```
R12:
ip pim autorp listener

R13:
ip pim autorp listener
!
ip mroute 0.0.0.0 0.0.0.0 172.29.100.15

R14:
ip pim autorp listener
!
ip mroute 0.0.0.0 0.0.0.0 172.29.100.15

R15:
ip pim autorp listener
!
ip mroute 0.0.0.0 0.0.0.0 Tunnel915
```

Task 7.2 Verification

R9 is the Auto-RP and Mapping Agent.

```
R9#show ip pim rp mapping
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)
This system is an RP-mapping agent (Loopback0)

Group(s) 224.0.0.0/4
  RP 10.255.255.9 (?), v2v1
    Info source: 10.255.255.9 (?), elected via Auto-RP
         Uptime: 00:05:48, expires: 00:00:11
R15#show ip pim rp mapping

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 10.255.255.9 (?), v2v1
    Info source: 10.255.255.9 (?), elected via Auto-RP
         Uptime: 00:03:51, expires: 00:00:14
```

https://t.me/learningnets

Remote sites have manually modified the RPF check for the Source 10.255.255.9, which allows them to accept both the Auto-RP control plane groups.

```
R15#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.39), 00:05:34/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.100, Forward/Sparse, 00:05:34/stopped
    Tunnel915, Forward/Sparse, 00:05:34/stopped
 (10.255.255.9, 224.0.1.39)
, 00:05:34/00:01:14, flags: PT Incoming interface: Tunnel915, RPF nbr 10.255.255.9, Mroute
  Outgoing interface list:
    GigabitEthernet1.100, Prune/Sparse, 00:00:55/00:02:04

(*, 224.0.1.40), 00:05:39/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1.100, Forward/Sparse, 00:05:39/stopped
    Tunnel915, Forward/Sparse, 00:05:39/stopped
 (10.255.255.9, 224.0.1.40)
, 00:05:34/00:02:57, flags: LT Incoming interface: Tunnel915, RPF nbr 10.255.255.9, Mroute

  Outgoing interface list:
    GigabitEthernet1.100, Forward/Sparse, 00:04:31/stopped
```

https://t.me/learningnets

Task 7.3 Multicast Traffic Engineering

```
R11:
interface GigabitEthernet1.102
 ip ospf cost 2

R13:
interface GigabitEthernet1.100
 ip igmp join-group 224.1.2.3

R14:
interface GigabitEthernet1.100
 ip igmp join-group 224.1.2.3

SW2:
ip multicast-routing distributed
!
interface Vlan227
 ip pim sparse-mode
 ip igmp join-group 224.1.2.3
!
interface Vlan228
 ip pim sparse-mode
 ip igmp join-group 224.1.2.3
```

Task 7.3 Verification

```
R12#ping 224.1.2.3
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.1.2.3, timeout is 2 seconds:
Reply to request 0 from 172.29.100.14
, 6 ms Reply to request 0 from 172.31.228.22
, 7 ms Reply to request 0 from 172.29.100.13
, 6 ms
```

Before traffic engineering:

```
R9# show ip mroute 224.1.2.3
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
```

https://t.me/learningnets

```
        T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
        X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
        U - URD, I - Received Source Specific Host Report,
        Z - Multicast Tunnel, z - MDT-data group sender,
        Y - Joined MDT-data group, y - Sending to MDT-data group,
        G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
        N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
        Q - Received BGP S-A Route, q - Sent BGP S-A Route,
        V - RD & Vector, v - Vector, p - PIM Joins on route,
        x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.2.3), 00:08:24/00:02:59, RP 10.255.255.9, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel89, Forward/Sparse, 00:06:27/00:02:55
    Tunnel915, Forward/Sparse, 00:08:24/00:02:59

(172.30.102.12, 224.1.2.3), 00:05:59/00:02:35, flags: T
Incoming interface: GigabitEthernet1.109, RPF nbr 172.30.109.11
  Outgoing interface list:
    Tunnel915, Forward/Sparse, 00:05:59/00:03:26
    Tunnel89, Forward/Sparse, 00:05:59/00:03:23
R9#show ip rpf 172.30.102.12
RPF information for ? (172.30.102.12)
  RPF interface: GigabitEthernet1.109 RPF neighbor: ? (172.30.109.11)
  RPF route/mask: 172.30.102.0/24
  RPF type: unicast (ospf 1)
  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base, originated from ipv4 unicast base
R9#show ip route 172.30.102.12
Routing entry for 172.30.102.0/24
  Known via "ospf 1", distance 110, metric 2, type inter area
  Last update from 172.30.109.11 on GigabitEthernet1.109, 00:01:18 ago
  Routing Descriptor Blocks: 172.30.109.11
, from 10.255.255.11, 00:01:18 ago, via GigabitEthernet1.109
      Route metric is 2, traffic share count is 1   *172.30.109.10
, from 10.255.255.10, 15:10:39 ago, via GigabitEthernet1.109
      Route metric is 2, traffic share count is 1
```

After traffic engineering:

https://t.me/learningnets

```
R9#show ip route 172.30.102.12

Routing entry for 172.30.102.0/24
  Known via "ospf 1", distance 110, metric 2, type inter area
  Last update from 172.30.109.10 on GigabitEthernet1.109, 00:02:19 ago
  Routing Descriptor Blocks:   * 172.30.109.10
, from 10.255.255.10, 15:11:40 ago, via GigabitEthernet1.109
      Route metric is 2, traffic share count is 1
R9#show ip rpf 172.30.102.12

RPF information for ? (172.30.102.12)
  RPF interface: GigabitEthernet1.109 RPF neighbor: ? (172.30.109.10)
  RPF route/mask: 172.30.102.0/24
  RPF type: unicast (ospf 1)
  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base, originated from ipv4 unicast base
R9#show ip mroute 224.1.2.3
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.2.3), 00:10:13/00:03:07, RP 10.255.255.9, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel89, Forward/Sparse, 00:08:16/00:03:04
    Tunnel915, Forward/Sparse, 00:10:13/00:03:07

(172.30.102.12, 224.1.2.3), 00:07:48/00:02:44, flags: T
Incoming interface: GigabitEthernet1.109, RPF nbr 172.30.109.10

  Outgoing interface list:
    Tunnel915, Forward/Sparse, 00:07:48/00:03:07
```

https://t.me/learningnets

```
    Tunnel89, Forward/Sparse, 00:07:48/00:03:04
```

Task 8.1 Denial of Service Tracking

```
R9:
ip access-list extended LOG_HTTP_SYN
 permit tcp 10.0.0.0 0.255.255.255 host 172.30.102.100 eq www
 permit tcp 172.16.0.0 0.15.255.255 host 172.30.102.100 eq www
 permit tcp 192.168.0.0 0.0.255.255 host 172.30.102.100 eq www
 permit tcp any host 172.30.102.100 eq www syn log-input
 permit ip any any
!
interface GigabitEthernet1.49
 ip access-group LOG_HTTP_SYN in
```

Task 8.2 Syslog

```
R9:
logging host 172.30.102.200
!
service sequence-numbers
!
service timestamps log datetime msec
!
ip access-list log-update threshold 10
```

Task 8.3 Traffic Filtering

```
R9:
interface GigabitEthernet1.49
 ip verify unicast source reachable-via rx
```

Task 9.1 NTP

```
R8:
ntp source Loopback0
ntp master 3
ntp peer 10.255.255.9
```

https://t.me/learningnets

```
ntp peer 10.255.255.13

R9:
ntp source Loopback0
ntp master 3
ntp peer 10.255.255.8
ntp peer 10.255.255.13

R13:
ntp source Loopback0
ntp master 3
ntp peer 10.255.255.8
ntp peer 10.255.255.9


SW2:
ntp source Loopback0
ntp server 10.255.255.8

R7:
ntp source Loopback0
ntp server 10.255.255.8


R10:
ntp source Loopback0
ntp server 10.255.255.9

R11:
ntp source Loopback0
ntp server 10.255.255.9

R12:
ntp source Loopback0
ntp server 10.255.255.9


R14:
ntp source Loopback0
ntp server 10.255.255.13

R15:
ntp source Loopback0
```

https://t.me/learningnets

```
ntp server 10.255.255.13
```

Task 9.2 NTP Security

```
R8:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp trusted-key 123
ntp peer 10.255.255.9 key 123
ntp peer 10.255.255.13 key 123
!
access-list 10 permit 10.255.255.9
access-list 10 permit 10.255.255.13
access-list 20 permit 10.255.255.7
access-list 20 permit 10.255.255.22
!
ntp access-group peer 10
ntp access-group serve-only 20


R9:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp trusted-key 123
ntp peer 10.255.255.8 key 123
ntp peer 10.255.255.13 key 123
!
access-list 10 permit 10.255.255.8
access-list 10 permit 10.255.255.13
access-list 20 permit 10.255.255.10
access-list 20 permit 10.255.255.11
access-list 20 permit 10.255.255.12
!
ntp access-group peer 10
ntp access-group serve-only 20


R13:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp trusted-key 123
ntp peer 10.255.255.8 key 123
ntp peer 10.255.255.9 key 123
!
```

https://t.me/learningnets

```
access-list 10 permit 10.255.255.8
access-list 10 permit 10.255.255.9
access-list 20 permit 10.255.255.14
access-list 20 permit 10.255.255.15
!
ntp access-group peer 10
ntp access-group serve-only 20


SW2:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp server 10.255.255.8 key 123
ntp trusted-key 123

R7:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp server 10.255.255.8 key 123
ntp trusted-key 123


R10:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp server 10.255.255.9 key 123
ntp trusted-key 123

R11:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp server 10.255.255.9 key 123
ntp trusted-key 123

R12:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp server 10.255.255.9 key 123
ntp trusted-key 123


R14:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp server 10.255.255.13 key 123
ntp trusted-key 123
```

https://t.me/learningnets

```
R15:
ntp authenticate
ntp authentication-key 123 md5 NTPK3Y
ntp server 10.255.255.13 key 123
ntp trusted-key 123
```

Task 9.3 QoS

```
R9:
policy-map SHAPE_VPN
 class class-default
  shape average 50000000
!
interface GigabitEthernet1.49
 service-policy output SHAPE_VPN
```

Task 9.4 QoS Sub-Rate

https://t.me/learningnets

```
R9:
class-map match-all VOIP
 match dscp ef
class-map match-all VOIP_SIGNALING
 match dscp cs3
class-map match-any TRANSACTIONAL_BULK
 match dscp af11
 match dscp af21
!
policy-map CHILD_QUEUING
 class VOIP
  priority 10000
 class VOIP_SIGNALING
  bandwidth 1000
 class TRANSACTIONAL_BULK
  bandwidth 30000
  random-detect dscp-based
 class class-default
!
policy-map SHAPE_VPN
 class class-default
  service-policy CHILD_QUEUING
```

https://t.me/learningnets