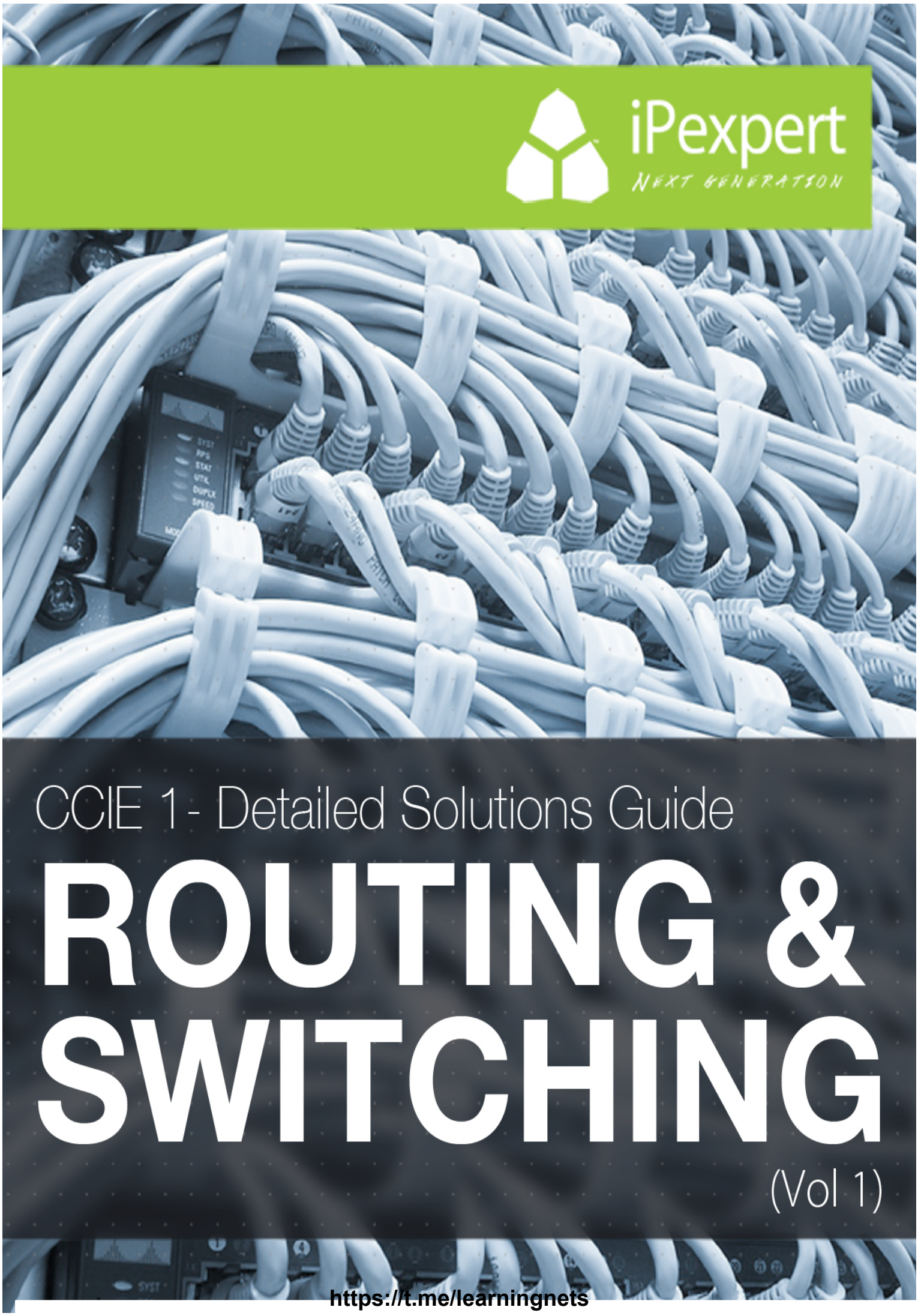



CCIE 1 - Detailed Solutions Guide

# ROUTING & SWITCHING

(Vol 1)

# Table of Contents



# Lab 1: Configure and troubleshoot switch port modes

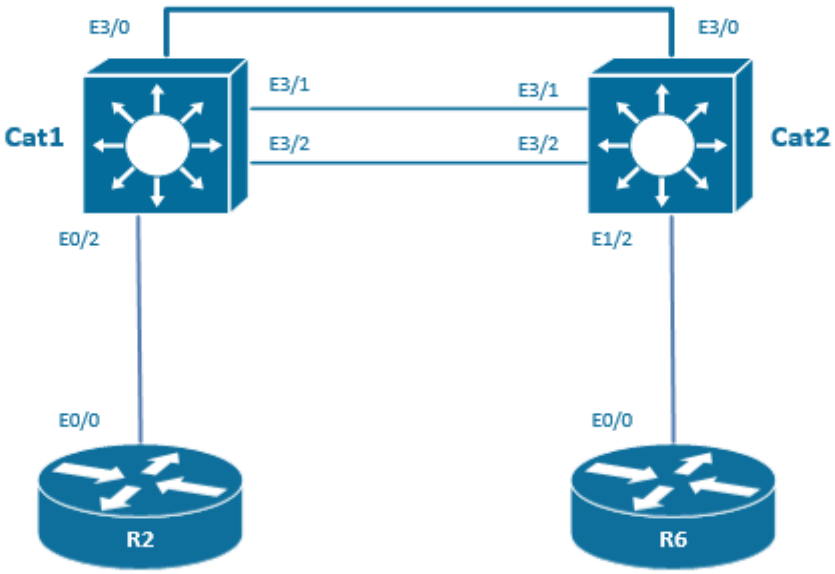
## Technologies covered

- CDP
- Access ports
- VLAN database
- VLAN
- Trunking
- dot1Q
- Native VLAN
- Manual pruning
- Layer 3 native interfaces
- SVIs
- Router-on-a-stick

## Overview

You have been tasked to configure the layer 2 part of the network and to enable the routing between 2 VLANs in a router-on-a-stick topology.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

### Task 1.1 Disable CDP on R2.

We can disable globally CDP on a device.

On R2, configure the following:

```
no cdp run
```

### Task 1.2 Disable CDP on the connection between R6 and Cat2.

On Cat2, we can see R6 in the list of the neighbors detected by CDP.

```
Cat2#sh 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
SW4              Eth 5/1           152              R S   Linux Uni Eth 5/1
SW4              Eth 5/0           152              R S   Linux Uni Eth 5/0
BB2              Eth 5/2           163              R B   Linux Uni Eth 0/0
BB3              Eth 5/3           153              R B   Linux Uni Eth 0/0
SW3              Eth 4/1           152              R S   Linux Uni Eth 4/1
SW3              Eth 4/0           152              R S   Linux Uni Eth 4/0
R2               Eth 0/2           110              R B   Linux Uni Eth 0/1
R6               Eth 1/2           157              R B   Linux Uni Eth 0/0
Cat1             Eth 3/2           172              R S   Linux Uni Eth 3/2
Cat1             Eth 3/1           172              R S   Linux Uni Eth 3/1
Cat1             Eth 3/0           172              R S   Linux Uni Eth 3/0
R8               Eth 2/0           149              R B   Linux Uni Eth 0/1
```

We have to disable CDP on the connection between Cat2 and R6.

On Cat2, configure the following:

```
int e1/2
no cdp enable
```

On R6, configure the following:

```
int e0/0
no cdp enable
```

### Task 1.3 Between Cat1 and Cat2, CDP should only be running on the E3/1 and E3/2 interfaces. The updates should be sent every 20 seconds, and the neighbor should be declared lost after 5 missing updates.

There are 3 connections between Cat1 and Cat2 that is to say E3/0, E3/1, and E3/2. We have to disable CDP on the E3/0 interface.

On Cat1, configure the following:

```
int e3/0
no cdp enable
```

On Cat2, configure the following:

```
int e3/0
no cdp enable
```

The CDP updates should be sent every 20 seconds, and the neighbor should be declared lost after 5 missing updates. Default value can be seen using the following show command:

```
Cat2#sh cdp
Global CDP information:
        Sending CDP packets every 60 seconds
        Sending a holdtime value of 180 seconds
        Sending CDPv2 advertisements is enabled
```

On Cat1 and Cat2, configure the following:

```
cdp timer 20
cdp holdtime 120
```

We can check that the configuration has taken effect:

```
Cat2#sh cdp
Global CDP information:
        Sending CDP packets every 20 seconds
        Sending a holdtime value of 120 seconds
        Sending CDPv2 advertisements is enabled
```

**Task 1.4** Between Cat1 and Cat2, the broadcasted CDP packets should not report mismatched native VLAN IDs.

Reporting mismatched native VLAN ID with a syslog message is one of the very nice features that are supported by CDP version 2. Between Cat1 and Cat2, we have to send only CDP version 1 updates. Modifying the CDP version is not supported on an interface level, but only on a global level.

On Cat1 and Cat2, use the following:

```
no cdp advertise-v2
```

**Task 1.5** Configure VLAN 101, 102, and 103 in the VLAN local database of Cat1 and Cat2 with the respective name of VLAN101, VLAN102, and VLAN103. The configuration of the VLANs should appear in the running-configuration and no VLAN distribution protocol should be running.

On Cat1 and Cat2, configure the following:

```
vlan 101
name VLAN101
vlan 102
name VLAN102
vlan 103
name VLAN103
```

The VLANs that were just created are appearing when typing the show vlan command. However, the configuration of the VLANs is not appearing in the running-configuration file. This is due to the fact that the default VTP mode is set to server.

```
Cat2#sh vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1
VTP Domain Name                 :
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : aabb.cc00.6600
Configuration last modified by 172.16.102.102 at 10-4-14 10:02:18
Local updater ID is 172.16.102.102 on interface Lo0 (first layer3 interface found)
```

```
Feature VLAN:
--------------
VTP Operating Mode                : Server
Maximum VLANs supported locally   : 1005
Number of existing VLANs          : 8
Configuration Revision            : 6
MD5 digest                        : 0x6B 0x99 0x57 0x33 0x79 0x8A 0xD8 0xFB
                                    0xF4 0xC3 0xEB 0x40 0xEE 0xD7 0xF5 0x6C
```

We have to modify the VTP mode to transparent in order to see the VLAN configuration into the running configuration file.

On Cat1 and Cat2, configure the following:

```
vtp mode transparent
```

**Task 1.6** Configure interface E3/0 in access mode VLAN 101 on Cat1 and Cat2.

On Cat1 and Cat2, configure the following:

```
int e3/0
switchport
switchport mode access
switchport access vlan 101
```

**Task 1.7** Configure the following IP addresses under the following interfaces:

| | |
|---|---|
| Cat1 E0/2 | 10.1.0.1/24 |
| R2 E0/0 | 10.1.0.2/24 |

Make sure that the ping is working.

On Cat1, configure the following:

```
int E0/2
no switchport
ip address 10.1.0.1 255.255.255.0
```

On R2, configure the following:

```
int E0/0
ip address 10.1.0.2 255.255.255.0
```

I can check the ping from R2 to Cat1 is working:

```
R2#ping 10.1.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.0.1, timeout is 2 seconds:
.!!!!
Success
rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms

The first ping is lost because it corresponds to the time for the ARP transactions to take
place.
```

**Task 1.8** Configure an ISL trunk allowing VLAN 102 on the E3/1. Leave it to DTP to negotiate, or not, a trunk.

```
Cat1#sh int trunk

Port        Mode             Encapsulation  Status        Native vlan
Et3/1       desirable        n-isl          trunking      1
Et3/2       desirable        n-isl          trunking      1
Et4/0       desirable        n-isl          trunking      1
```

```
Et4/1       desirable        n-isl          trunking      1
Port        Vlans allowed on trunk
Et3/1       1-4094
Et3/2       1-4094
Et4/0       1-4094
Et4/1       1-4094

Port        Vlans allowed and active in management domain
Et3/1       1,101-103
Et3/2       1,101-103
Et4/0       1,101-103
Et4/1       1,101-103

Port        Vlans in spanning tree forwarding state and not pruned
Et3/1       1,101-103
Et3/2       1,101-103
Et4/0       1,101-103

Port        Vlans in spanning tree forwarding state and not pruned
Et4/1       1,101-103
```

By default, an ISL trunk is negotiated as soon as the port E3/1 come up, so without configuring anything, we have a working trunk for VLANs 101,102, and 103. We are going to limit this trunk to transmit only on the VLAN 102.

On Cat1 and Cat2, configure the following:

```
int e3/1
switchport trunk allowed vlan 102
```

We can check that only VLAN 102 is trunked on port E3/1.

```
Cat1#sh int e3/1 trunk

Port        Mode             Encapsulation  Status        Native vlan
Et3/1       desirable        n-isl          trunking      1

Port        Vlans allowed on trunk
Et3/1       102

Port        Vlans allowed and active in management domain
Et3/1       102

Port        Vlans in spanning tree forwarding state and not pruned
Et3/1       102
```

**Task 1.9** Configure a dot1q trunk allowing VLAN 103 on the E3/2. Disable DTP on this connection. VLAN 103 should be sent untagged.

We have been asked to disable DTP on this connection. By hard-coding the mode to trunk and the encapsulation to dot1q, we are actually disabling DTP at the same time.

On Cat1 and Cat2, configure the following:

```
int e3/2
switchport trunk encapsulation dot1q
switchport mode trunk
```

We are going to limit this trunk to transmit only on the VLAN 103 and to configure the VLAN 103 as the native VLAN on the trunk. The native VLAN is sent untagged.

On Cat1 and Cat2, configure the following:

```
int e3/2
switchport trunk allowed vlan 103
switchport trunk native vlan 103
```

We can check our configuration with the following command:

```
Cat2#sh int e3/2 trunk
Port        Mode             Encapsulation  Status        Native vlan
Et3/2       on               802.1q         trunking      103

Port        Vlans allowed on trunk
Et3/2       103

Port        Vlans allowed and active in management domain
Et3/2       103

Port        Vlans in spanning tree forwarding state and not pruned
Et3/2       103
```

**Task 1.10** Configure only the following SVIs:

| | |
|---|---|
| Cat1 Vlan 103 | 10.103.0.1/24 |
| Cat2 Vlan 101 | 10.101.0.2/24 |

On Cat1, configure the following:

```
int vlan 103
ip address 10.103.0.1 255.255.255.0
no shut
```

On Cat2, configure the following:

```
int vlan 101
ip address 10.101.0.2 255.255.255.0
no shut
```

**Task 1.11** Configure the following sub-interfaces on the E0/0 of the R6:

| | |
|---|---|
| E0/0.101 | 10.101.0.6/24 |
| E0/0.103 | 10.103.0.6/24 |

We are going to configure a router on a stick topology. R6 is going to do the on-a-stick inter-Vlan routing.

On R6, configure the following:

```
interface Ethernet0/0.101
 encapsulation dot1Q 101
 ip address 10.101.0.6 255.255.255.0

interface Ethernet0/0.103
 encapsulation dot1Q 103
 ip address 10.103.0.6 255.255.255.0
```

On Cat2, configure the following:

```
interface Ethernet1/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 101,103
 switchport mode trunk
```

**Task 1.12** Ensure that you can ping from the interface Vlan 103 on Cat1 to the interface Vlan 101 on Cat2 by using R6 as the inter-VLAN routing point. Do not use the “ip route” command.

In order to route from VLAN 103 to VLAN 101 over the router on a stick R6, we have to give each VLAN a default gateway. As we are not allowed to use the “ip route” command, we can use the “default-gateway” command instead.

On Cat1, configure the following:

```
no ip routing
ip default-gateway 10.103.0.6
```

On Cat2, configure the following:

```
no ip routing
ip default-gateway 10.101.0.6
```

Interesting enough, please note that you have to disable IP routing on the switch. Otherwise, the ip default-gateway will not be taken into account.

The ping from Cat2 to Cat1 routed from R6 is not up and running:

```
Cat1#ping 10.101.0.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.101.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

Cat1#traceroute 10.101.0.2
Type escape sequence to abort.
Tracing the route to 10.101.0.2
VRF info: (vrf in name/id, vrf out name/id)
  1 10.103.0.6 0 msec 0 msec 1 msec
  2 10.101.0.2 1 msec *  0 msec
```

**You have completed Lab 1**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 2: Configure and troubleshoot VTP

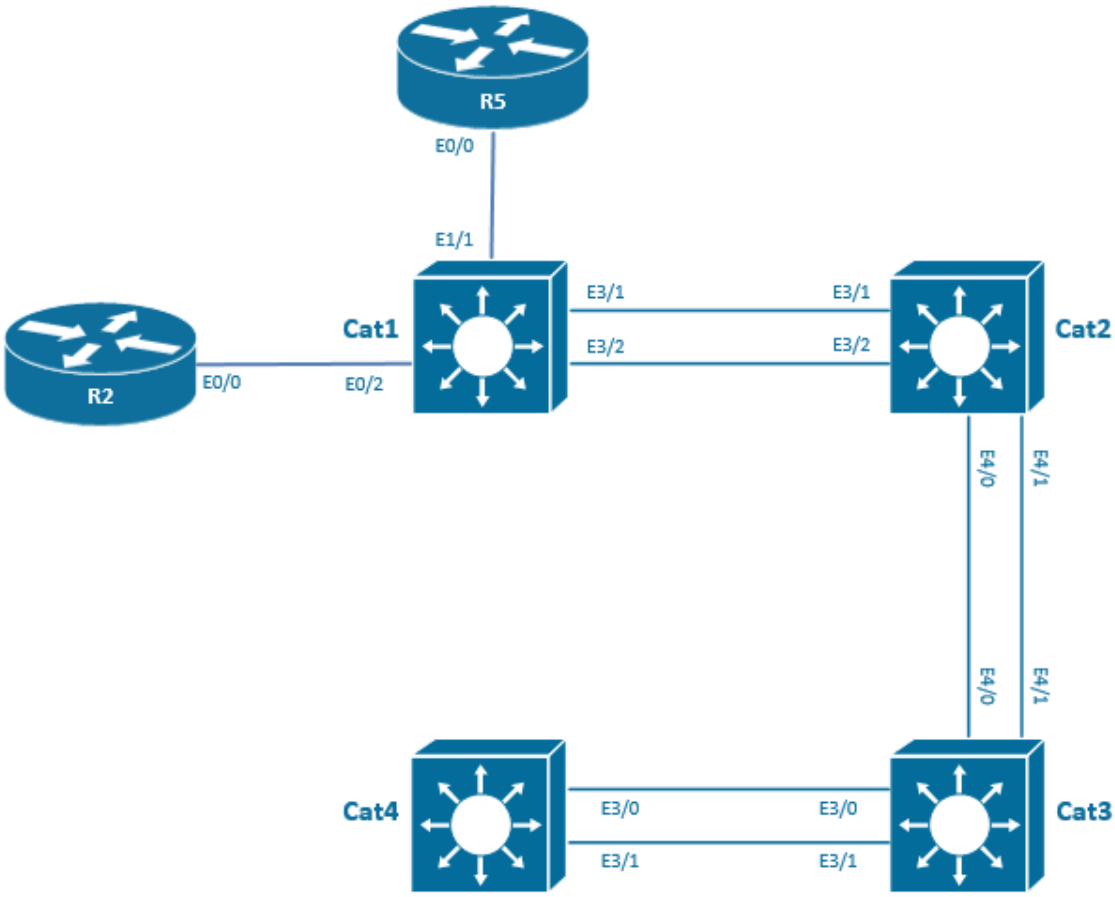
## Technologies covered

- VTPv1
- VTPv2
- VTPv3
- VTP pruning

## Overview

You have been tasked to automatically distribute the VLANs in the network using VTP. You have to propagate normal VLANs as well as extended VLANs. Your VTP set-up should be secured and high available.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

### Task 2.1 Configure a dot1q trunk allowing all VLANS on all the connections between Cat1 and Cat2, between Cat2 and Cat3, and between Cat3 and Cat4.

On Cat1, configure the following:

```
int e3/1
switchport trunk encapsulation dot1q
switchport mode trunk

int e3/2
switchport trunk encapsulation dot1q
switchport mode trunk
```

On Cat2, configure the following:

```
int e3/1
switchport trunk encapsulation dot1q
switchport mode trunk
int e3/2
switchport trunk encapsulation dot1q
switchport mode trunk

int e4/0
switchport trunk encapsulation dot1q
switchport mode trunk

int e4/1
switchport trunk encapsulation dot1q
switchport mode trunk
```

On Cat3, configure the following:

```
int e4/0
switchport trunk encapsulation dot1q
switchport mode trunk

int e4/1
switchport trunk encapsulation dot1q
switchport mode trunk

int e3/0
switchport trunk encapsulation dot1q
switchport mode trunk

int e3/1
switchport trunk encapsulation dot1q
switchport mode trunk
```

On Cat4, configure the following:

```
int e3/0
switchport trunk encapsulation dot1q
switchport mode trunk

int e3/1
switchport trunk encapsulation dot1q
switchport mode trunk
```

### Task 2.2 Configure Cat4 as the server of the VTP domain iPexpert.

On Cat4, configure the following:

```
vtp mode server
vtp domain iPexpert
```

**Task 2.3** Configure Cat3 not to update its VLAN database. VTP packets should be silently forwarded by Cat3.

On Cat3, configure the following:

```
vtp mode transparent
```

It is important to configure the VTP in mode transparent and not in mode off. The mode off would not forward the VTP packets, and the server Cat4 would not be able to reach the clients Cat1 and Cat 2.

**Task 2.4** Configure Cat1 and Cat2 as client of Cat4.

On Cat1 and Cat2, configure the following:

```
vtp mode client
vtp domain iPexpert
```

**Task 2.5** Add VLAN 150 and 151 on Cat4, and check that those VLANs are now present on Cat1 and Cat2, but not on Cat3.

On Cat4, the VTP server, configure the following:

```
vlan 150
vlan 151
```

Let's check if those 2 VLANs have been propagated to the VTP clients Cat1 and Cat2. Make sure that all the trunks on the path from Cat4 to Cat1 are up and running and trunking properly.

```
Cat1#sh vlan
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et0/0, Et0/1, Et0/2, Et0/3
                                                Et1/0, Et1/1, Et1/2, Et1/3
                                                Et2/0, Et2/1, Et2/2, Et2/3
                                                Et3/0, Et3/3, Et4/0, Et4/1
                                                Et4/2, Et4/3, Et5/0, Et5/1
                                                Et5/2, Et5/3, Et6/0, Et6/1
                                                Et6/2, Et6/3, Et7/0, Et7/1
                                                Et7/2, Et7/3
150  VLAN0150                         active
151  VLAN0151                         active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
150  enet  100150     1500  -      -      -        -    -        0      0
151  enet  100151     1500  -      -      -        -    -        0      0

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1002 fddi  101002     1500  -      -      -        -    -        0      0
1003 tr    101003     1500  -      -      -        -    srb      0      0
1004 fdnet 101004     1500  -      -      -        ieee -        0      0
1005 trnet 101005     1500  -      -      -        ibm  -        0      0

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

**Task 2.6** Add VLAN 1500 on Cat4, and make sure that it is propagated to Cat1 and Cat2, but not to Cat3.

VTP version 1 and 2 support only the propagation of the VLANs ranging from 1-1001. In order to forward the VLAN with the VLAN ID 1500, we have to upgrade the VTP version to version 3.

On Cat1, Cat2, Cat3 and Cat4, configure the following:

```
vtp version 3
spanning-tree extend system-id
```

The default operational state of a switch configured with VTP v3 is to be in secondary server mode. Cat4 has to be converted into a primary VTP version 3 server. This is done by typing on the command line (not in configuration mode) the following:

```
vtp primary vlan
```

Let's observe what is happening when this command is entered:

```
Cat4#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: aabb.cc00.6800 has become the primary server for the
VLAN VTP feature

Cat4#sh vtp status
VTP Version capable             : 1 to 3
VTP version running             : 3
VTP Domain Name                 : iPexpert
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : aabb.cc00.6800

Feature VLAN:
--------------
VTP Operating Mode                : Primary Server
Number of existing VLANs          : 7
Number of existing extended VLANs : 0
Maximum VLANs supported locally   : 4096
Configuration Revision            : 1
Primary ID                        : aabb.cc00.6800
Primary Description               : Cat4
MD5 digest                        : 0x7B 0x3D 0xBC 0x71 0xB5 0x80 0xA9 0xDF
                                    0x47 0xA4 0x1D 0x7E 0x50 0xF8 0x5C 0xEB


Feature MST:
--------------
VTP Operating Mode                : Transparent


Feature UNKNOWN:
--------------
VTP Operating Mode                : Transparent
```

Now that the network has been upgraded to VTP version 3, we can configure VLAN ID 1500 on Cat4 and this VLAN will be propagated to Cat1 and Cat2.

On Cat4, configure the following:

```
vlan 1500
```

On Cat1, we can check that this VLAN has been propagated to the Version 3 clients:

```
Cat1#sh vlan


VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et0/0, Et0/1, Et0/2, Et0/3
                                                Et1/0, Et1/1, Et1/2, Et1/3
                                                Et2/0, Et2/1, Et2/2, Et2/3
                                                Et3/0, Et3/3, Et4/0, Et4/1
                                                Et4/2, Et4/3, Et5/0, Et5/1
                                                Et5/2, Et5/3, Et6/0, Et6/1
                                                Et6/2, Et6/3, Et7/0, Et7/1
                                                Et7/2, Et7/3
150  VLAN0150                         active
151  VLAN0151                         active
1002 fddi-default                     act/unsup
1003 trcrf-default                    act/unsup
1004 fddinet-default                  act/unsup
1005 trbrf-default                    act/unsup
1500 VLAN1500                         active


VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
150  enet  100150     1500  -      -      -        -    -        0      0

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
151  enet  100151     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0
1003 trcrf 101003     4472  1005   3276   -        -    srb      0      0
1004 fdnet 101004     1500  -      -      -        ieee -        0      0
1005 trbrf 101005     4472  -      -      15       ibm  -        0      0
1500 enet  101500     1500  -      -      -        -    -        0      0


VLAN AREHops STEHops Backup CRF
---- ------- ------- ----------
1003 7       7       off

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

**Task 2.7** Configure the VTP domain with a password of “090909”. This password should be stored in the NVRAM database.

On Cat1, Cat2, Cat3 and Cat4, configure the following:

```
vtp password 090909 hidden
```

By using the hidden keyword, the secret key generated from the password string is saved in the nvam:vlan.dat file.

Once we have configured the VTP password, we have to re-enable Cat4 as the primary VTP server and type the password.

```
Cat4#vtp primary vlan
This system is becoming primary server for feature vlan
Enter VTP Password:
No conflicting VTP3 devices found.
Do you want to continue? [confirm]
%SW_VLAN-4-VTP_PRIMARY_SERVER_CHG: aabb.cc00.6800 has become the primary server for the
VLAN VTP feature
```

**Task 2.8** Ensure that the next VLAN created will not be propagated to switches where this VLAN is not allowed on any trunks.

When an allowed-list is configured on a trunk, only VTP information regarding VLANs allowed on the trunk should be transmitted. This feature limiting flooding of VTP traffic is called VTP pruning.
VTP pruning in VTP version 3 should be enabled on all switches in the domain except the ones in VTP transparent mode.

On Cat1, Cat2 and Cat4, configure the following:

```
vtp pruning
```

**Task 2.9** Ensure that Cat2 will take over the server role in the case of a failure of Cat4.

On Cat2, configure the following:

```
vtp mode server
```

The Cat2 will be acting as a VTP version 3 secondary server, the primary server being Cat4.

**Task 2.10** Configure R2 in VLAN 150 and R5 in VLAN 1500 as client ports. As Cat1 is not having any client's port in VLAN 151, make sure that broadcast packets in VLAN 151 will never be transmitted to Cat1.

On Cat1, configure the following:

```
interface Ethernet0/2
 switchport access vlan 150
 switchport mode access

interface Ethernet1/1
 switchport access vlan 1500
 switchport mode access

interface Ethernet3/1
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 150,1500
 switchport mode trunk
 duplex auto

interface Ethernet3/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 150,1500
 switchport mode trunk
 duplex auto


On Cat2, configure the following:
interface Ethernet3/1
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 150,1500
 switchport mode trunk
 duplex auto

interface Ethernet3/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 150,1500
 switchport mode trunk
 duplex auto
```

On Cat4, configure the following:

```
vlan 151
```

Even if the VLAN 151 has been propagated with VTP to Cat1, the broadcast on VLAN 151 will net be sent to Cat1 thanks to the allowed list configured on the trunks.

```
Cat1#sh vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et0/0, Et0/1, Et0/3, Et1/0
                                                Et1/2, Et1/3, Et2/0, Et2/1
                                                Et2/2, Et2/3, Et3/0, Et3/3
                                                Et4/0, Et4/1, Et4/2, Et4/3
                                                Et5/0, Et5/1, Et5/2, Et5/3
                                                Et6/0, Et6/1, Et6/2, Et6/3
                                                Et7/0, Et7/1, Et7/2, Et7/3
150  VLAN0150                         active    Et0/2
151  VLAN0151                         active
1002 fddi-default                     act/unsup
1003 trcrf-default                    act/unsup
1004 fddinet-default                  act/unsup
1005 trbrf-default                    act/unsup
1500 VLAN1500                         active    Et1/1

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
150  enet  100150     1500  -      -      -        -    -        0      0
151  enet  100151     1500  -      -      -        -    -        0      0

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1002 fddi  101002     1500  -      -      -        -    -        0      0
1003 trcrf 101003     4472  1005   3276   -        -    srb      0      0
1004 fdnet 101004     1500  -      -      -        ieee -        0      0
1005 trbrf 101005     4472  -      -      15       ibm  -        0      0
1500 enet  101500     1500  -      -      -        -    -        0      0


VLAN AREHops STEHops Backup CRF
---- ------- ------- ----------
1003 7       7       off

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

**You have completed Lab 2**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 3: Configure and troubleshoot Portchannels

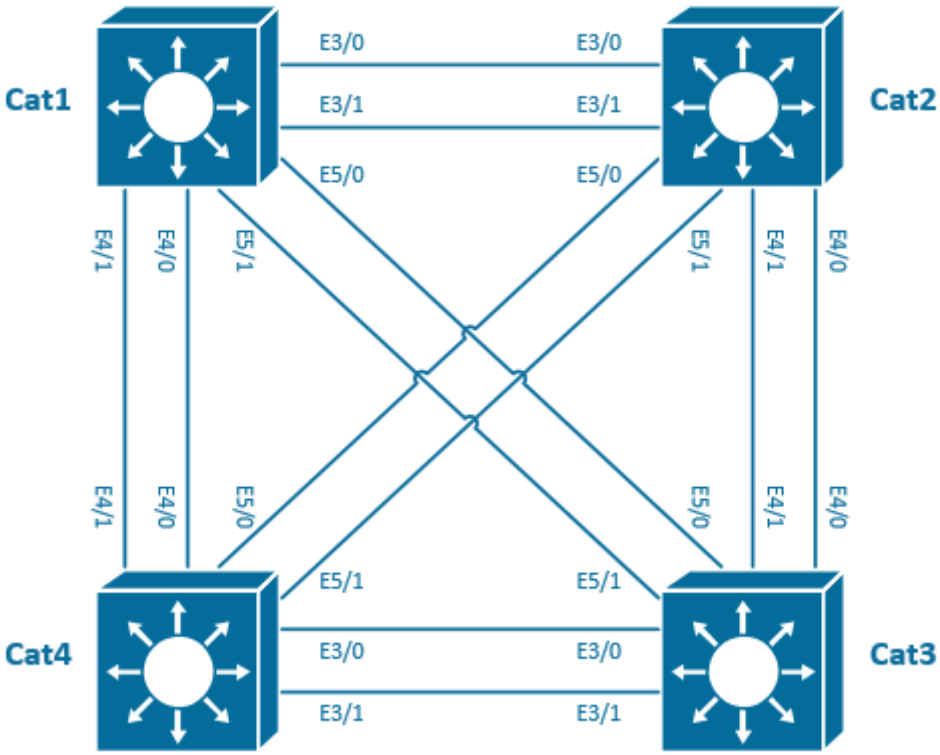
## Technologies covered

- LACP etherchannel
- PagP etherchannel
- Manual etherchannel
- L2 etherchannel
- L3 etherchannel
- Load-balancing
- Etherchannel misconfiguration guard

## Overview

You have been tasked to configure seamless redundancy in the network by bundling several physical connections into a logical connection called port-channel. In addition, you should traffic-engineer the way that traffic is distributed on the different members of those port-channels.

The topology used in the lab will be the following:




**Estimated time to complete: 2-3 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

### Task 3.1

Between Cat2 and Cat3, configure a static port-channel Po23 trunking in a dot1q encapsulation the VLAN 101.

On Cat2 and Cat3, configure the following:

```
vtp mode transparent
vlan 101

int range e4/0-1
channel-group 23 mode on
```

Once this is configured, an interface Port-channel 23 is being created.

On Cat2 and Cat3, configure the following:

```
int po23
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 101
switchport mode trunk
int range e4/0-1
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 101
switchport mode trunk
```

We can check that the Po23 is up and running:

```
SW3#sh 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
------+-------------+-----------+-----------------------------------------------
23     Po23(SU)         -        Et4/0(P)    Et4/1(P)
```

### Task 3.2

Between Cat3 and Cat4, configure a PagP port-channel Po34 trunking in an ISL packet the VLAN 101. The Cat3 should not start the negotiation. Configure PagP in a way that the port-channel is protected against unidirectional failure.

We have been told to configure the PagP Cisco proprietary protocol to aggregate the physical connections into a logical one. We have to make sure that the bundling is not starting before bi-

directional traffic has been detected. This is the case in PagP non-silent mode. Silent mode is the default.

The Cat3 should not start the negotiation so Cat3's side will be configured in auto mode. The Cat4 side has to initiate the PagP negotiation and will be configured in desirable mode.

On Cat3, configure the following:

```
int range e3/0-1
channel-group 34 mode auto non-silent
```

On Cat4, configure the following:

```
vtp mode transparent
vlan 101

int range e3/0-1
channel-group 34 mode desirable non-silent
```

On Cat3 and Cat4, configure the following:

```
int po34
switchport
switchport trunk encapsulation isl
switchport trunk allowed vlan 101
switchport mode trunk

int range e3/0-1
switchport
switchport trunk encapsulation isl
switchport trunk allowed vlan 101
switchport mode trunk
```

We can check that the Po34 is up and running:

```
Cat4#sh 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
------+-------------+-----------+-----------------------------------------------
34     Po34(SU)         PAgP      Et3/0(P)    Et3/1(P)
```

We can check that the Po34 is trunking in an ISL mode:

```
Cat4#sh int po34 trunk

Port        Mode             Encapsulation  Status        Native vlan
Po34        desirable        n-isl          trunking      1

Port        Vlans allowed on trunk
Po34        1-4094

Port        Vlans allowed and active in management domain
Po34        1,101,150-151,1500

Port        Vlans in spanning tree forwarding state and not pruned
Po34        1,101,150-151,1500
```

**Task 3.3** Between Cat2 and Cat4, configure a LACP port-channel Po24 trunking in the port in VLAN 102. The Cat2 should never start the negotiation.

We have to bundle the circuits between Cat2 and Cat4 in a LACP port-channel. The cat2 should never start the negotiation and will therefore be configured as the passive side. As no trunking encapsulation is specified, we are going to use the dot1q encapsulation.

On Cat2, configure the following:

```
vlan 102

int range e5/0-1
channel-group 24 mode passive
```

On Cat4, configure the following:

```
vlan 102

int range e5/0-1
channel-group 24 mode active
```

On Cat2 and Cat4, configure the following:

```
int po24
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 102
switchport mode trunk
int range e5/0-1
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 102
switchport mode trunk
```

We can check that the Po24 is up and running:

```
Cat4#sh 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: 2
Number of aggregators:           2

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
24     Po24(SU)         LACP      Et5/0(P)    Et5/1(P)
   Po34(SU)         PAgP      Et3/0(P)    Et3/1(P)
```

**Task 3.4** Ensure that Cat4 is leading the LACP negotiation.

In order to ensure that Cat4 is leading the LACP negotiation on all aspects, we can modify the LACP system priority on Cat4. The lower the value, the higher the priority. Please note that this LACP system priority is globally configured and will therefore apply to all the port-channels configured on Cat4.

On Cat4, configure the following:

```
lacp system-priority 0
```

**Task 3.5** Ensure that E5/0 will be used as LACP failover if 9 members are present in the Port-channel.

The LACP port-priority configured on an interface will be compared in the case that a Port-channel is containing more than 8 members. The 8 interfaces with the lowest priority will be used in the bundle and the others will be placed into failover mode. The default priority is 32768. E5/0 is supposed to go into LACP failover when more than 8 members are present.

On Cat2 and Cat4, configure the following:

```
int E5/0
lacp port-priority 65535
```

**Task 3.6** Between Cat1 and Cat2, configure a static port-channel Po12 with the following IP address:

| | |
|---|---|
| Cat1 Po12 | 10.12.0.1/24 |
| Cat2 Po12 | 10.12.0.2/24 |

We are asked to configure a static L3 port-channel.

On Cat1 and Cat2, configure the following:

```
int range e3/0-1
no switchport
channel-group 12 mode on
```

On Cat1, configure the following:

```
int po12
no switchport
ip address 10.12.0.1 255.255.255.0
```

On Cat2, configure the following:

```
int po12
no switchport
ip address 10.12.0.2 255.255.255.0
```

**Task 3.7** Between Cat1 and Cat3, configure a PagP port-channel Po13 with the following IP address:

| | |
|---|---|
| Cat1 Po13 | 10.13.0.1/24 |
| Cat3 Po13 | 10.13.0.3/24 |

We are asked to configure a PagP L3 port-channel.

On Cat1 and Cat3, configure the following:

```
int range e5/0-1
no switchport
channel-group 13 mode desirable
```

On Cat1, configure the following:

```
int po13
no switchport
ip address 10.13.0.1 255.255.255.0
```

On Cat3, configure the following:

```
int po13
no switchport
ip address 10.13.0.3 255.255.255.0
```

**Task 3.8** Between Cat1 and Cat4, configure a LACP port-channel Po14 with the following IP address:

| | |
|---|---|
| Cat1 Po14 | 10.14.0.1/24 |
| Cat4 Po14 | 10.14.0.4/24 |

We are asked to configure a LACP L3 port-channel.

On Cat1 and Cat4, configure the following:
```
int range e4/0-1
no switchport
channel-group 14 mode active
```

On Cat1, configure the following:
```
int po14
no switchport
ip address 10.14.0.1 255.255.255.0
```

On Cat4, configure the following:
```
int po14
no switchport
ip address 10.14.0.4 255.255.255.0
```

**Task 3.9** On the Port-channel between the Cat1 and the Cat2, all the TCP flows from a source MAC address to the same destination MAC address should be using the same member in all the port-channels just configured.

This question is pointing towards modifying the etherchannel load-balancing mechanism. Please note that this command is configured in global mode and is therefore affecting all the port-channels configured on the device.

The default mechanism for IP traffic is source-destination IP load-balancing.
```
Cat2#sh etherchannel load-balance
EtherChannel Load-Balancing Configuration:
        src-dst-ip

EtherChannel Load-Balancing Addresses Used Per-Protocol:
Non-IP: Source XOR Destination MAC address
  IPv4: Source XOR Destination IP address
  IPv6: Source XOR Destination IP address
```

On Cat 1 and Cat2, configure the following:
```
port-channel load-balance src-dst-mac
```

**Task 3.10** On the Port-channel between the Cat3 and the Cat4, make sure that all the flows coming from a MAC address are using the same PagP member when the packet returns to this MAC address.

The question is referring to the PagP learn method. Please note that this mechanism is working only when the load-distribution method is set to source-based distribution, so that any given source MAC address is always sent on the same physical port.

On Cat3 and Cat4, configure the following:
```
port-channel load-balance src-mac

int range e3/0-1
pagp learn-method physical-port
```

**Task 3.11** Configure the four switches with a mechanism to disable the port-channel in the case of a mis-configuration that is leading to the port-channel receiving Spanning-Tree BPDUs on two different members.

On Cat1, Cat2, Cat3 and Cat4, configure the following:

```
spanning-tree etherchannel guard misconfig
```

**You have completed Lab 3**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

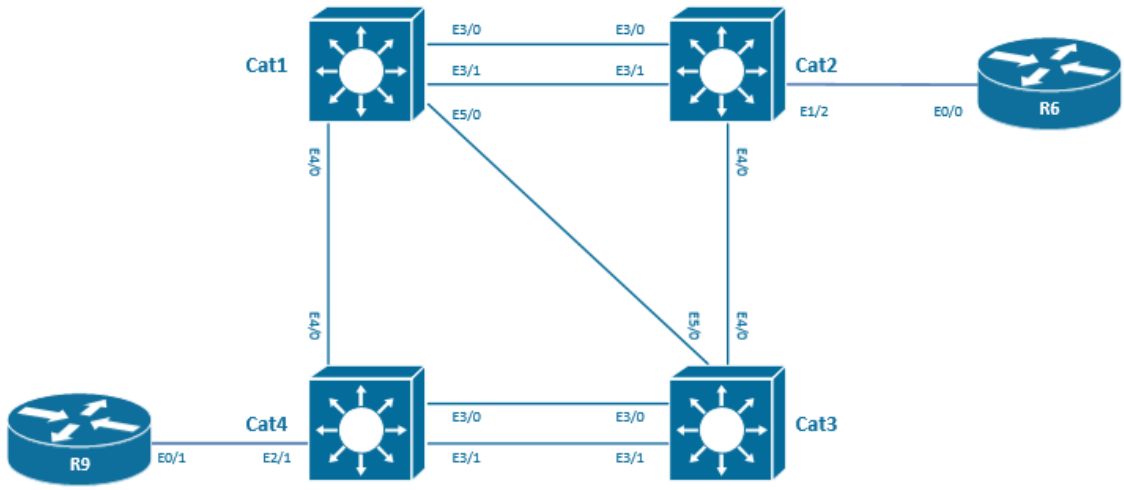
# Lab 4: Configure and troubleshoot Spanning-tree Protocol

## Technologies covered

- PVST+
- Switch priority
- Port priority
- Path cost
- STP timers
- Port fast
- BPDUguard, BPDUfilter
- Loopguard
- Rootguard
- Backbonefast
- Loopfast
- UDLD

## Overview

You have been tasked to guarantee in a redundant L2 network a loop-free topology by configuring the Spanning Tree protocol. Traffic engineering and optimization is also required. The 2 routers R6 and R9 wil be considered as hosts that should not make part of the spanning-tree topology.




**Estimated time to complete: 3-4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

### Task 4.1 Configure the 4 Catalysts to run PVST+ (and not rapid PVST+)

On SW1, SW2, SW3 and SW4, configure the following:

```
spanning-tree mode pvst
```

### Task 4.2 Configure all the inter-switches connection as trunk dot1q trunking all the VLANs.

On SW1, configure the following:

```
int E3/0
switchport trunk encapsulation dot1q
switchport mode trunk

int E3/1
switchport trunk encapsulation dot1q
switchport mode trunk

int E5/0
switchport trunk encapsulation dot1q
switchport mode trunk

int E4/0
switchport trunk encapsulation dot1q
switchport mode trunk
```

On SW2, configure the following:

```
int E3/0
switchport trunk encapsulation dot1q
switchport mode trunk

int E3/1
switchport trunk encapsulation dot1q
switchport mode trunk

int E4/0
switchport trunk encapsulation dot1q
switchport mode trunk
```

On SW3, configure the following:

```
int E4/0
switchport trunk encapsulation dot1q
switchport mode trunk

int E5/0
switchport trunk encapsulation dot1q
switchport mode trunk

int E3/0
switchport trunk encapsulation dot1q
switchport mode trunk

int E3/1
switchport trunk encapsulation dot1q
switchport mode trunk
```

On SW4, configure the following:

```
int E3/0
switchport trunk encapsulation dot1q
switchport mode trunk

int E3/1
switchport trunk encapsulation dot1q
```

```
switchport mode trunk

int E4/0
switchport trunk encapsulation dot1q
switchport mode trunk
```

**Task 4.3** Configure Cat1 as VTP server for domain iPexpert and configure VLAN 21 and 22.

On SW1, configure the following:

```
vtp mode server
vtp domain iPexpert
```

On SW2, SW3 and SW4, configure the following :

```
vtp mode client
vtp domain iPexpert
```

**Task 4.4** Configure the primary root bridge on Cat2 for VLAN 21. Configure the secondary root bridge on Cat4 for VLAN 21. Do not use a command containing “priority” in order to achieve this. Optimize the timers to the number of switches.

On SW2, configure the following:

```
spanning-tree vlan 21 root primary
```

On SW4, configure the following:

```
spanning-tree vlan 21 root secondary
```

**Task 4.5** Configure the primary root bridge on Cat3 for VLAN 22. Configure the secondary root bridge on Cat1 for VLAN 22. Do not use a command containing “root” in order to achieve this.

On SW3, configure the following:

```
spanning-tree vlan 22 priority 4096
```

On SW1, configure the following:

```
spanning-tree vlan 22 priority 8192
```

**Task 4.6** In VLAN 22, make sure that Cat2 and Cat4 will never become root of the network.

On SW2, configure the following:

```
spanning-tree vlan 22 priority 0
```

On SW4, configure the following:

```
spanning-tree vlan 22 priority 0
```

**Task 4.7** On VLAN 22, change the hello timer to 5s, the max aging time to 20s and the forward delay to 15s.

On SW1, SW2, SW3 and SW4, configure the following:

```
spanning-tree vlan 22 hello-time 5
spanning-tree vlan 22 forward-time 15
spanning-tree vlan 22 max-age 20
```

**Task 4.8** All the connections being up and running, on VLAN 21, the traffic from R6 to R9 should be forwarded using the following path: Cat2-Cat1-Cat3-Cat4.

On SW3 and SW4, configure the following:

```
int e4/0
spanning-tree vlan 21 cost 200000000
```

**Task 4.9** All the connections being up and running, on VLAN 22, the traffic from R6 to R9 should be forwarded using the following path: Cat2-Cat3-Cat4.

On SW1, configure the following:

```
int e3/0
spanning-tree vlan 22 cost 200000000
int e3/1
spanning-tree vlan 22 cost 200000000
```

**Task 4.10** All the connections being up and running, on VLAN 21, the traffic from Cat1 to Cat2 and from Cat3 and Cat4 should flow over the E3/0 connections.

On SW2 and SW3, configure the following:

```
int e3/0
spanning-tree vlan 21 port-priority 0
```

**Task 4.11** All the connections being up and running, on VLAN 22, the traffic from Cat3 and Cat4 should flow over the E3/0 connection.

On SW3, configure the following:

```
int e3/0
spanning-tree vlan 22 port-priority 0
```

**Task 4.12** Reduce the convergence time associated with indirect failures in the network.

On SW1, SW2, SW3 and SW4, configure the following:

```
spanning-tree backbonefast
```

**Task 4.13** Enable the Uplinkfast feature on the switches where it cannot create loops. When a failure occurs on a switch with Uplinkfast feature on, a maximum of 100 dummy multicast packets have to generated every second in order to update the rest of the network bridging tables.

On SW2 and SW4, configure the following:

```
spanning-tree uplinkfast max-update-rate 100
```

**Task 4.14** Configure R6 as a client in VLAN 21 in access mode.

On SW2, configure the following:

```
int e1/2
switchport mode access
switchport access vlan 21
```

**Task 4.15** Configure R9 as a client with a trunk connection allowing VLAN 22. VLAN 22 should be the native of the dot1q trunk.

On SW4, configure the following:

```
int e2/1
switchport trunk encapsulation dot1q
switchport mode trunk
switchport trunk allowed vlan 22
switchport trunk native vlan 22
```

**Task 4.16** Allow the port connected to the routers to transition immediately from blocked to fowarding.

On SW2, configure the following:

```
int e1/2
spanning-tree portfast
```

On SW4, configure the following:

```
int e2/1
spanning-tree portfast trunk
```

**Task 4.17** R6 could be sending BPDUs and we would like the port to be put in error-disabled in the case that it happens. Configure the port to re-enable itself automatically after 1 minute.

On SW2, configure the following:

```
int e1/2
spanning-tree bpduguard enable

errdisable recovery cause bpduguard
errdisable recovery interval 60
```

**Task 4.18** VLAN R9 is sending BPDUs, but we would like to ignore them and to silently drop them.

On SW4, configure the following:

```
int e2/1
spanning-tree bpdufilter enable
```

**Task 4.19** The link between Cat1 and Cat3 should be protected from a loop caused by a unidirectionnal link. Do not use UDLD.

On SW1 and on SW3, configure the following:

```
spanning-tree loopguard default
int e5/0
spanning-tree link-type point-to-point
```

**Task 4.20** The link between Cat1 and Cat4 should be removed from the network topology if an unidirectionnal link is detected. The port on Cat1 should be put in err-disabled when an unidirectional event happens but not the port on Cat4. Configure the port to re-enable itself automatically after 5 minutes.

On SW1, configure the following:

```
int E4/0
udld port aggressive

errdisable recovery cause udld
errdisable recovery interval 300
```

On SW4, configure the following:

```
int E4/0
udld port

errdisable recovery cause udld
errdisable recovery interval 300
```

**You have completed Lab 4**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 5: Configure and troubleshoot Multi-Instance Spanning-tree Protocol (MST)

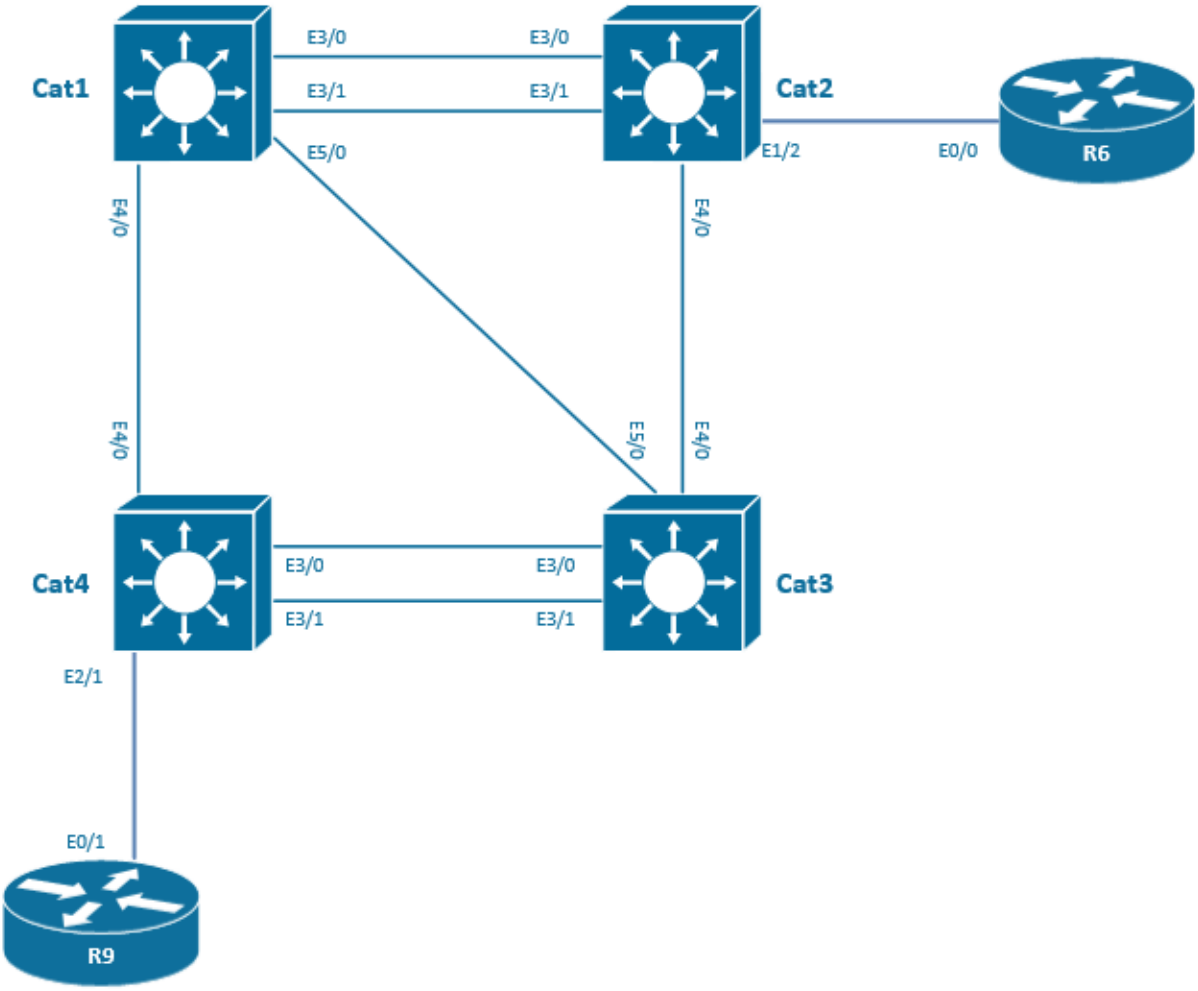
## Technologies covered

- MST
- MST region
- CST
- RPVST+

## Overview

The switches will run very CPU intensive processes. You have been tasked to optimize the spanning-tree protocol in order to create fewer burdens on the CPU of the switches. Running one SPT process for a group of VLANs is made possible with MST.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 5.1** Configure the Cat1, Cat 2, and Cat 3 Switches to run the MST protocol with the name iPexpertRegion. Configure VLAN 100, 110, and VLAN 200, 210 on the Cat1, Cat 2, and Cat3 Switches.

On Cat1, Cat2 and Cat3, configure the following:

```
vtp mode transparent
vlan 100
vlan 110
vlan 200
vlan 210

spanning-tree mode mst
spanning-tree mst configuration
name iPexpertRegion
revision 1
```

**Task 5.2** Instance 10 with the name iPexpert10 will encompass the VLAN range 100-150.

On Cat1, Cat2 and Cat3, configure the following:

```
spanning-tree mst configuration
instance 10 vlan 100-150
```

**Task 5.3** Instance 20 with the name iPexpert20 will encompass the VLANs 200, 210, 220, 230, 240, and 250.

On Cat1, Cat2 and Cat3, configure the following:

```
spanning-tree mst configuration
instance 20 vlan 200,210,220,230,240,250
```

At this stage, we can check that MST is running on the VLANs.

```
Cat1#sh spanning-tree mst configuration
Name      [iPexpertRegion]
Revision  1     Instances configured 3

Instance  Vlans mapped
--------  ------------------------------------------------------------------------
0         1-99,151-199,201-209,211-219,221-229,231-239,241-249,251-4094
10        100-150
20        200,210,220,230,240,250
--------------------------------------------------------------------------------
```

Please note that as soon as MST is enabled on the switches, all the VLANs are by default part of the MST tree instance 0.

For example, VLAN 100 is running MST instance 10. This tree is called MST10 in the output below.

```
Cat1#sh spanning-tree vlan 100

MST10
  Spanning tree enabled protocol mstp
  Root ID    Priority    32778
             Address     aabb.cc00.6500
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32778  (priority 32768 sys-id-ext 10)
             Address     aabb.cc00.6500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Desg FWD 2000000   128.13   Shr
Et3/1               Desg FWD 2000000   128.14   Shr
Et4/0               Desg FWD 2000000   128.17   Shr Bound(PVST)
Et5/0               Desg FWD 2000000   128.21   Shr
```

**Task 5.4** Configure all the inter-switches connection as trunk dot1q trunking all the VLANs.

On Cat1, configure the following:

```
int range e3/0-1
switchport
switchport trunk encapsulation dot1q
switchport mode trunk

int e5/0
switchport
switchport trunk encapsulation dot1q
switchport mode trunk

int e4/0
switchport
switchport trunk encapsulation dot1q
switchport mode trunk
```

On Cat2, configure the following:

```
int range e3/0-1
switchport
switchport trunk encapsulation dot1q
switchport mode trunk

int e4/0
switchport
switchport trunk encapsulation dot1q
switchport mode trunk
```

On Cat3, configure the following:

```
int range e3/0-1
switchport
switchport trunk encapsulation dot1q
switchport mode trunk

int e5/0
switchport
switchport trunk encapsulation dot1q
switchport mode trunk

int e4/0
switchport
```

```
switchport trunk encapsulation dot1q
switchport mode trunk
```

On Cat4, configure the following:

```
int range e3/0-1
switchport
switchport trunk encapsulation dot1q
switchport mode trunk
int e4/0
switchport
switchport trunk encapsulation dot1q
switchport mode trunk
```

**Task 5.5** For instance 10, configure Cat2 to be the root primary and Cat3 to be the root secondary. Do not use the priority command.

We have to configure the root primary and secondary of the MST instance 10 tree. As we are not allowed to use the priority command, we are going to use the macros “spanning-tree mst 10 root primary” and “spanning-tree mst 10 root secondary”

On Cat3, configure the following:

```
spanning-tree mst 10 root secondary
```

On Cat2, configure the following:

```
spanning-tree mst 10 root primary
```

Please note that it is important to run first the secondary macro and then the primary macro. If you run the primary macro before the secondary macro, the root priority may be changed to a value that will not permit a secondary priority value to be inserted between the primary priority value and the default value. This is due to the fact that a bridge priority has to be a multiple of 4096.

Those macros have merely checked the priority configured on all the switches in the MST 10 tree and have generated the lines of configuration “spanning-tree mst 10 priority 24576” on Cat2 and “spanning-tree mst 10 priority 28672” on Cat3.

**Task 5.6** For instance 20, configure Cat3 to always be the root primary and Cat2 to be the root secondary.

In this question, we are not going to use the macros and we are going to configure directly the root priority. We are going to use a priority of 0 on Cat3. It is our interpretation of the “always” word in the question.

On Cat3, configure the following:

```
spanning-tree mst 20 priority 0
```

On Cat2, configure the following:

```
spanning-tree mst 20 priority 4096
```

By using VLAN 100 which is part of MST10 and by using VLAN 200 which is part of MST20, let’s verify that Cat2 is the root for MST10 and that Cat3 is the root for MST20.

```
Cat2#sh spanning-tree vlan 100

MST10
  Spanning tree enabled protocol mstp
  Root ID    Priority    24586
```

```
              Address     aabb.cc00.6600
              This bridge is the root
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

    Bridge ID  Priority    24586  (priority 24576 sys-id-ext 10)
              Address     aabb.cc00.6600
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Desg FWD 2000000   128.13   Shr
Et3/1               Desg FWD 2000000   128.14   Shr
Et4/0               Desg FWD 2000000   128.17   Shr

Cat3#sh spanning-tree vlan 200

MST20
  Spanning tree enabled protocol mstp
  Root ID    Priority    20
             Address     aabb.cc00.6700
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    20     (priority 0 sys-id-ext 20)
             Address     aabb.cc00.6700
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Desg FWD 2000000   128.13   Shr Bound(PVST)
Et3/1               Desg FWD 2000000   128.14   Shr Bound(PVST)
Et4/0               Desg FWD 2000000   128.17   Shr
Et5/0               Desg FWD 2000000   128.21   Shr
```

**Task 5.7** Between Cat1 and Cat2, make sure that the blocked path is on the E3/0 for instance 10.

Let's have a look at the current state of the spanning-tree topology MST10.

```
Cat1#sh spanning-tree vlan 100

MST10
  Spanning tree enabled protocol mstp
  Root ID    Priority    24586
             Address     aabb.cc00.6600
             Cost        2000000
             Port        13 (Ethernet3/0)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32778  (priority 32768 sys-id-ext 10)
             Address     aabb.cc00.6500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Root FWD 2000000   128.13   Shr
Et3/1               Altn BLK 2000000   128.14   Shr
Et4/0               Desg FWD 2000000   128.17   Shr Bound(PVST)
Et5/0               Altn BLK 2000000   128.21   Shr
```

At the moment, E3/1 is the interface blocked by spanning-tree and E3/0 is the interface forwarding between Cat1 and Cat2. According to the question, it has to be the other way around.

We have to use the port-priority in order to influence the decision on which interfaces will be blocked. The default priority is 128. The lower the number, the higher the priority. Please note that port-priority has to be a multiple of 16.

On Cat2, configure the following:

```
int e3/1
spanning-tree mst 10 port-priority 0
```

Please note that this port-priority has to configure on the side of the root Cat2 in order to have the port on the other side of the connection on Cat1 to be blocked.

```
Cat1#sh spanning-tree vlan 100
MST10
  Spanning tree enabled protocol mstp
  Root ID    Priority    24586
             Address     aabb.cc00.6600
             Cost        2000000
             Port        14 (Ethernet3/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32778  (priority 32768 sys-id-ext 10)
             Address     aabb.cc00.6500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Altn BLK 2000000   128.13   Shr
Et3/1               Root FWD 2000000   128.14   Shr
Et4/0               Desg FWD 2000000   128.17   Shr Bound(PVST)
Et5/0               Altn BLK 2000000   128.21   Shr
```

We have now the E3/0 in a blocking state and the E3/1 in a forwarding state. Mission completed!

### **Task 5.8** Configure VLAN 100, 110, 200, and 210 on Cat4.

On Cat4, configure the following:

```
vtp mode transparent
vlan 100
vlan 110
vlan 200
vlan 210
```

### **Task 5.9** Configure the MST region iPexpertRegion to always be the root of the CST.

The CST (Common Spanning-tree) is running by default between MST region and non-MST region to ensure inter-operability between the different Spanning-tree modes and to guarantee a loop-free topology at any times. The whole MST region is seen by R4 as a big switch. Let's have a look at the current status of the network.

```
Cat4#sh spanning-tree vlan 100

VLAN0100
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
             Address     aabb.cc00.6500
             Cost        100
             Port        17 (Ethernet4/0)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32868  (priority 32768 sys-id-ext 100)
             Address     aabb.cc00.6800
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

```
              Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Altn BLK 100       128.13   Shr
Et3/1               Altn BLK 100       128.14   Shr
Et4/0               Root FWD 100       128.17   Shr
```

In order to make sure that the MST region iPexpertRegion is the root of the CST, we have to configure the Cat4 with the highest root priority possible, making it the least preferred. We have to understand that the priority of the root of the big switches MST1 and MST2 will be compared with the priority of Cat4 to elect the root of the CST.

On Cat4, configure the following:

```
spanning-tree vlan 100,110,200,210 priority 61440
```

**Task 5.10** Ensure that the port E4/0 on the Cat4 is in BLK state.

Cat4 is connected with 3 connections to the big MST10 switch. Therefore, in order to avoid loops, only one connection is in forwarding mode, Eth4/0 in the current topology.

We have to modify spanning-tree cost to have a topology change and to have E4/0 transition to a blocking state. The current cost on all connection is 100. We are going to increase the cost on the interface 4/0 to 2000.

On Cat4, configure the following:

```
int e4/0
spanning-tree vlan 100,110,200,210 cost 2000
```

The E4/0 is now in a blocking state.

```
Cat4#sh spanning-tree vlan 100

VLAN0100
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
             Address     aabb.cc00.6500
             Cost        100
             Port        13 (Ethernet3/0)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    61540  (priority 61440 sys-id-ext 100)
             Address     aabb.cc00.6800
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  300 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Root FWD 100       128.13   Shr
Et3/1               Altn BLK 100       128.14   Shr
Et4/0               Altn BLK 2000      128.17   Shr
```

**Task 5.11** Ensure that the port E3/0 on the Cat4 is in BLK state.

E3/0 and E3/1 on Cat4 is connected to E3/0 and E3/1 on Cat3. The current situation is that E3/1 is the port that is in a spanning-tree blocking state. The question asks us to have E3/0 in a blocking state.

In a “normal” spanning-tree traffic engineering scenario, we would have to use the port-priority in order to influence the decision on which interfaces will be blocked. But we are here in a CST topology

traffic-engineering where the spanning-tree cost has to be used even if the 2 connections E3/0 and E3/1 are between the same physical switches.

On Cat4, configure the following:

```
int e3/0
spanning-tree vlan 100,101,200,201 cost 2000
```

We have now reached the topology that is targeted by this scenario.

```
Cat4#sh spanning-tree vlan 100

VLAN0100
  Spanning tree enabled protocol ieee
  Root ID     Priority    32768
              Address     aabb.cc00.6500
              Cost        100
              Port        14 (Ethernet3/1)
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    61540  (priority 61440 sys-id-ext 100)
              Address     aabb.cc00.6800
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et3/0               Altn BLK 2000      128.13   Shr
Et3/1               Root FWD 100       128.14   Shr
Et4/0               Altn BLK 2000      128.17   Shr
```

**Task 5.12** Make sure that the spanning-tree reconfiguration on Cat4 occurs in less than one second with 802.1w.

We have first to enable the rapid spanning-tree protocol on Cat4 and then to enable the spanning-tree point-to-point port type on the connection between Cat4 and the MST domain.

On Cat4, configure the following:

```
spanning-tree mode rapid-pvst
int e3/0
spanning-tree link-type point-to-point
int e3/1
spanning-tree link-type point-to-point
int e4/0
spanning-tree link-type point-to-point
```

On Cat3, configure the following:

```
int e3/0
spanning-tree link-type point-to-point
int e3/1
spanning-tree link-type point-to-point
```

On Cat1, configure the following:

```
int e4/0
spanning-tree link-type point-to-point
```

**You have completed Lab 5**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 6: Miscellaneous Layer 2 Topics

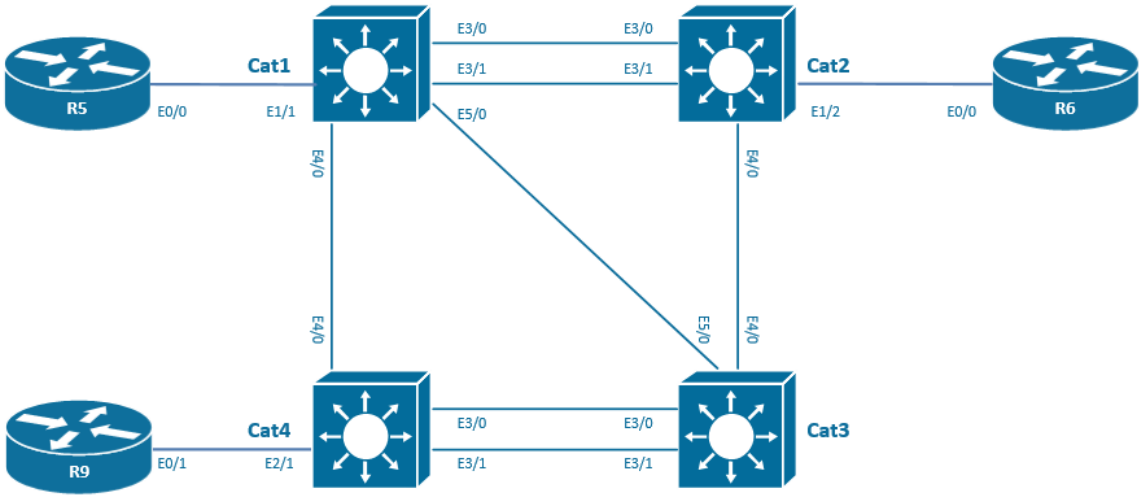
## Technologies covered

- Managing MAC address table
- Protected ports
- Stormcontrol
- SPAN
- RSPAN
- ERSPAN
- Voice VLANs
- Smartports Macros
- Private VLAN

## Overview

There are some application problems in the network. You have been tasked to troubleshoot and understand the performance issues by sniffing the problematic traffic and setting up a SPAN and RSPAN session. As Cisco IP phones will be hooked up to the network, you will be asking to configure those ports and guarantee voice quality.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 6.1** On Cat1, the dynamic MAC-address table entries should be removed from the table when they are not re-learned after 10 seconds.

We have to configure the dynamic learnt MAC address to time out after 10 seconds. The default is 5 minutes. Please note that this timeout value is very low and is not recommended in a production network, because it will generate a lot of unnecessary flooding.

On Cat1, configure the following:

```
mac address-table aging-time 10
```

**Task 6.2** On Cat1, for troubleshooting reasons, enable the MAC address change notification feature. Configure the switch to send SNMP traps to server 10.1.99.99 with a community iPexpert1 as soon as a MAC address is removed or added on interface E1/1. Keep up to 500 entries in the MAC notification table.

On Cat1, configure the following:

```
mac address-table notification < not supported on the switches used in the iPexpert pods
mac address-table notification history-size 500 < not supported on the switches used in the
iPexpert pods

snmp-server host 10.1.99.99 traps iPexpert1
snmp-server enable traps mac-notification

int E1/1
snmp trap mac-notification added < not supported on the switches used in the iPexpert pods
```

**Task 6.3** On Cat1 configure interface E1/1 as an access port in VLAN 120.

On Cat1, configure the following:

```
vtp mode transparent
vlan 120

int E1/1
switchport mode access
switchport access vlan 120
```

**Task 6.4** On Cat1, disable MAC address learning in VLAN 120 and add a static entry that indicates the MAC address of the interface E0/0 of R5 is located in VLAN 120 behind interface E1/1.

On Cat1, configure the following:

```
no mac-address-table learning vlan 120 < not supported on the switches used in the iPexpert
pods
mac-address-table static aabb.cc00.0500 vlan 120 interface E1/1
```

We can check that the MAC address-table entry for the MAC address of R5 is now static.

```
SW1#sh mac address-table vlan 120
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
 120    aabb.cc00.0500    STATIC      Et1/1
Total Mac Addresses for this criterion: 1
```

**Task 6.5** On Cat1, enable unicast MAC address filtering in VLAN 120 and configure the switch to drop packets that have a source or destination address of cafe.2222.2222.

On Cat 1, configure the following:

```
mac address-table static cafe.2222.2222 vlan 120 drop < not supported on the switches used
in the iPexpert pods
```

**Task 6.6** On Cat1, ensure that a server connected to the interface E5/1 and a server connected to the interface E1/2 cannot send traffic to each other at layer 2. Do not use port-security.

We are going to use the protected port feature. It is a more limited version of port-security. The key rule is that, within the same VLAN, traffic between two protected ports is blocked but traffic between a protected and unprotected port is allowed.

On Cat1, configure the following:

```
int e5/1
switchport protected < not supported on the switches used in the iPexpert pods
int e1/2
switchport protected < not supported on the switches used in the iPexpert pods
```

**Task 6.7** On Cat4, prevent traffic on the LAN from being disrupted from a broadcast and unicast storm on the interface E2/1. A storm is considered a storm when more than 50% of the bandwidth is used by broadcast packets and when more than 80% of the bandwidth is used by unicast packets.

On Cat4, configure the following:

```
int e2/1
storm-control unicast level 80 < not supported on the switches used in the iPexpert pods
storm-control broadcast level 50 < not supported on the switches used in the iPexpert pods
```

**Task 6.8** Multicast packets should always be dropped on the interface E2/1. Use storm-control.

If you are configuring the storm-control to 0% of the bandwidth, you are actually suppressing all the traffic.

```
int e2/1
storm-control multicast level 0
```

**Task 6.9** Configure a dot1q trunk between Cat2 and R6. This trunk should be allowed on VLAN 121 and VLAN 122.

On R6, configure the following:

```
interface Ethernet0/0.121
 encapsulation dot1Q 121

interface Ethernet0/0.122
 encapsulation dot1Q 122
```

On Cat2, configure the following:

```
vtp mode transparent
vlan 121
vlan 122
```

```
interface Ethernet1/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 121,122
 switchport mode trunk
```

We can check that E1/2 is trunking VLAN 121 and 122.

```
SW2#sh int e1/2 trunk

Port        Mode             Encapsulation  Status        Native vlan
Et1/2       on               802.1q         trunking      1

Port        Vlans allowed on trunk
Et1/2       121-122
Port        Vlans allowed and active in management domain
Et1/2       121-122

Port        Vlans in spanning tree forwarding state and not pruned
Et1/2       121-122
```

**Task 6.10** A laptop called Laptop1 with a Wireshark sniffer is connected on Cat2 on the port E1/3. Configure this port with a dot1q trunk encapsulation allowing all the VLANs.

On Cat2, configure the following:

```
interface Ethernet1/3
 switchport trunk encapsulation dot1q
 switchport mode trunk
```

**Task 6.11** Configure the Cat2 switch to mirror all the traffic transiting in VLAN 121 on Cat2 on E1/2 to the port where the sniffer Laptop1 is connected. Use session number 60.

On Cat2, configure the following:

```
monitor session 60 source vlan 121 < not supported on the switches used in the iPexpert
pods
monitor session 60 destination interface E1/2 < not supported on the switches used in the
iPexpert pods
```

**Task 6.12** The port where the Sniffer is connected should accept incoming traffic with a dot1q encapsulation. Default ingress VLAN is VLAN 121.

On Cat2, configure the following:

```
monitor session 60 destination interface E1/2 ingress dot1q vlan 121 < not supported on the
switches used in the iPexpert pods
```

Please note that a destination port in a SPAN never accepts any incoming traffic. However, this ingress command added to the monitor session command is used when you need to receive some traffic incoming on the SPAN destination port.

**Task 6.13** Configure a LACP port-channel between Cat1 and Cat2. Bundle interface E3/0 with E3/1 on both sides. This port-channel is a dot1q trunk allowing VLAN 121, VLAN 122, and VLAN 500.

On Cat1 and Cat2, configure the following:

```
vtp mode transparent
vlan 121
```

```
vlan 122
vlan 500

int range e3/0-1
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 121,122,500
switchport mode trunk
channel-group 1 mode active

int po1
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 121,122,500
switchport mode trunk
```

**Task 6.14** A laptop called Laptop2 with a Wireshark sniffer is connected on Cat1 on the port E0/3. Configure this port with an access port in VLAN 1.

On Cat1, configure the following:

```
int E0/3
switchport mode access
switchport access vlan 1
```

**Task 6.15** Configure the mirroring of the sent traffic transiting in VLAN 122 on Cat2 on E1/2 to the port where the sniffer Laptop2 is connected. Use session number 61 and VLAN 500 as RSPAN VLAN.

The vlan 500 will be used as our VLAN dedicated to carry the rspan traffic between the source and the destination Laptop2. Remote span is a vlan type.

On Cat2 and Cat3, configure the following:

```
vlan 500
remote span < not supported on the switches used in the iPexpert pods
```

On Cat2, configure the following:

```
monitor session 61 source interface ethernet1/2 tx
monitor session 61 filter vlan 122
monitor session 61 destination remote vlan 500
```

On Cat1, configure the following:

```
monitor session 61 source remote vlan 500
monitor session 61 destination interface Ethernet0/3
```

Please note that the session ID used on Cat1 and Cat2 for Rspan has to match.

**Task 6.16** On Cat3, there will be a Cisco IP phone connected to the port E1/0. Enable QOS on the Cat3 and configure the port E1/0 to trust COS.

On Cat 3, configure the following:

```
mls qos < not supported on the switches used in the iPexpert pods
int E1/0
mls qos trust cos
```

**Task 6.17** Configure a VLAN of 33 reserved for voice traffic on Cat3. The voice traffic on E1/0 should use this voice VLAN.

On Cat 3, configure the following:

```
int E1/0
switchport voice vlan 33
```

The IP packets with IP precedence 5 will be switched to vlan 33. The IP packets with other IP precedence will end up in the “normal” access VLAN configured on the port.

**Task 6.18** The incoming Data frames coming from a computer connected on the Cisco IP phone should be tagged by the switch with a COS of 2.

There is a port on the IP phone, where a computer can be connected in order to simplify the cabling and reduce the costs. We have to mark this traffic with an IP precedence of 2.

On Cat3, configure the following:

```
int E1/0
switchport priority extend cos 2
```

**Task 6.19** On Cat3, configure a macro called “Bounce-int” to bounce (shut followed by a no shut) an interface. Use a variable called $int. Test and run the macro for int E1/0.

On Cat3, configure the following:

```
macro name Bounce-int
int $int
shut
no shut
end
@
```

To test and run the macro, we can issue the following command in configuration mode on Cat3:

```
macro global apply Bounce-int $int E1/0
```

**Task 6.20** On Cat3, there will be an additional Cisco IP phone connected to the E1/1. Use the preconfigured macro called “cisco-phone” to configure the port. Voice VLAN has to be VLAN 2 and Data VLAN has to be VLAN 1.

There is a preconfigured macro called cisco-phone that will configure the voice and data VLAN on a port as well as all the QOS features.
On Cat3, configure the following:

```
int e1/1
macro apply cisco-phone $AVID 1 $VVID 3
```

**Task 6.21** On Cat1 and on Cat4, configure VLAN 120 as the primary VLAN, VLAN 130 as the isolated VLAN, and VLAN 140 as the community VLAN. Configure E4/1 Cat4 as the PVLAN promiscuous port. Configure interface E4/0 and int E5/0 Cat1 as the PVLAN host port for VLAN 130, interface E5/1, and interface E3/0 Cat1 as the PVLAN host port for VLAN 140. The connection between Cat1 and Cat4 has to be configured as a trunk port that will support the setup.

We are first going to configure the Vlans and the trunk between Cat1 and Cat4:

On Cat1, configure the following:

```
vtp mode transparent

vlan 120
private-vlan primary
private-vlan association 130,140
vlan 130
private-vlan isolated
vlan 140
private-vlan community
int e4/0
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 120,130,140
switchport mode trunk
```

Then, we have to configure the ports as promiscuous or host ports.
On Cat1, configure the following:

```
interface E4/1
switchport mode private-vlan promiscuous
switchport private-vlan mapping 120 130,140
```

On Cat4, configure the following:

```
interface E4/0
switchport mode private-vlan host
switchport private-vlan host-association 120 130

interface E5/0
switchport mode private-vlan host
switchport private-vlan host-association 120 130

interface E5/1
switchport mode private-vlan host
switchport private-vlan host-association 120 140

interface E3/0
switchport mode private-vlan host
switchport private-vlan host-association 120 140
```

**You have completed Lab 6**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 7: HDLC and PPP/PPPoE

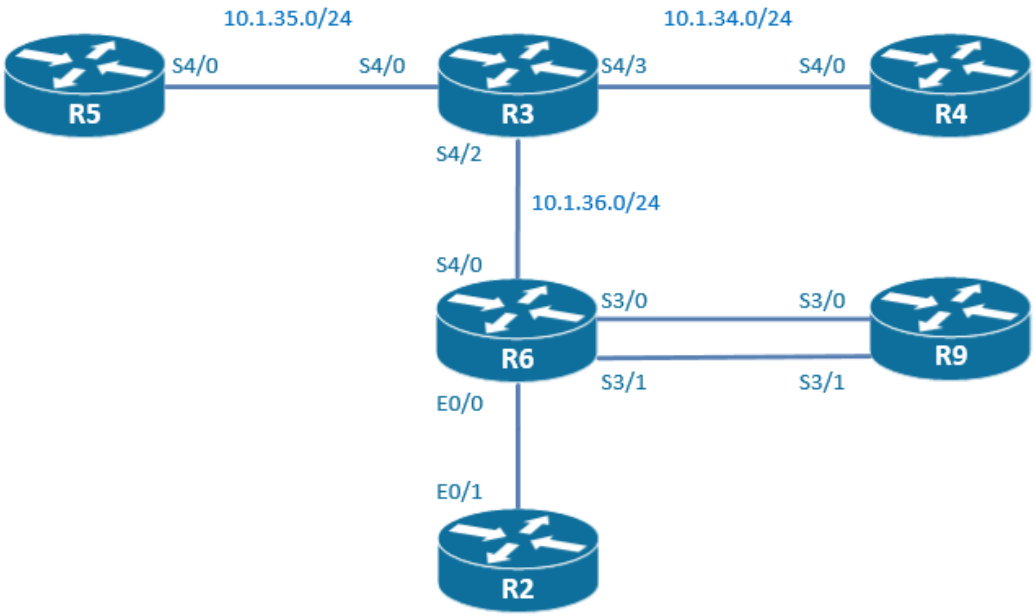
## Technologies covered

- HDLC
- PPP PAP, CHAP
- PPPoE
- MLPPP
- PPP inter-leaving
- RTP reserve
- Virtual-assembly

## Overview

You have been tasked to configure the serial connections of your network with the HDLC and PPP encapsulation. PPP connection may have to be authenticated or aggregated in a bundle.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 7.1** The link between R3 and R4 should be using the HDLC encapsulation. Check that you can ping from R3 to R4.

The link has been pre-configured with PPP and we have to configure the HDLC encapsulation instead.

On R3, configure the following:

```
int s4/3
encapsulation hdlc
```

On R4, configure the following:

```
int s4/0
encapsulation hdlc
```

The ping from R3 to R4 is working:

```
R3#ping 10.1.34.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.34.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 7.2** The link between R3 and R5 should be using the PPP encapsulation. Turn on the CHAP authentication with the password of “Password35”. Check that you can ping from R3 to R5.

On R3, configure the following:

```
username R5CHAP1 password Password35

int s4/0
encapsulation ppp
ppp authentication chap
ppp chap hostname R3CHAP1
```

On R5, configure the following:

```
username R3CHAP1 password Password35

int s4/0
encapsulation ppp
ppp authentication chap
ppp chap hostname R5CHAP1
```

Let’s check that we can still ping from R5 to R3.

```
R5#ping 10.1.35.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.35.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 7.3** The link between R3 and R6 should be using the PPP encapsulation. Turn on the PAP authentication with the password of “Password361”. If the PAP authentication is unsuccessful, CHAP authentication has to kick in with a password of “Password362”. Check that you can ping from R3 to R6.

On R3, configure the following:

```
username R6USERPAP password Password361
username R6CHAP2 password Password362

int s4/2
encapsulation ppp
```

```
ppp authentication pap chap
ppp pap sent-username R3USERPAP password Password361
ppp chap hostname R3CHAP2
```

On R6, configure the following:

```
username R3USERPAP password Password361
username R3CHAP2 password Password362

int s4/0
encapsulation ppp
ppp authentication pap chap
ppp pap sent-username R6USERPAP password Password361
ppp chap hostname R6CHAP2
```

Let's check that we can still ping from R6 to R3.

```
R6#ping 10.1.36.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.36.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

In order to test that CHAP is working when PAP is not working, we can use the command ppp pap refuse.

**Task 7.4** Configure PPPoE between the R6 and the R2 routers. R6 is the server side and R2 is the client side. On the server side, a BBA is called "iPexpertgroup". The IP pool is called" iPexpertpool" and the range is from 10.1.26.10 to 10.1.26.20. The virtual-template number should use id 23 and the IP address configured on the virtual template is 10.1.26.6 255.255.255.0.

On the server side R6, configure the following:

```
bba-group pppoe iPexpertgroup
virtual-template 23

interface E0/0
no ip address
pppoe enable group iPexpertgroup

interface virtual-template 23
ip address 10.1.26.6 255.255.255.0
peer default ip address pool iPexpertpool

ip local pool iPexpertpool 10.1.26.10 10.1.26.20
```

**Task 7.5** Limit the number of sessions established (per client MAC address) to 3.

This task should be configured in the broadband aggregation (BBA) group.

On the server side R6, configure the following:

```
bba-group pppoe iPexpertgroup
sessions per-mac limit 3
```

**Task 7.6** On the client side, use the id 26 for both the dialer interface and the dialer-pool-number interface. Check that you can ping from R6 to R2.

On the client side R2, configure the following:

```
interface E0/1
no ip address
```

```
pppoe enable
pppoe-client dial-pool-number 26

interface dialer 26
ip address negotiated
encapsulation ppp
dialer pool 26
```

The client side has been assigned the IP address 10.1.26.10 by the server.

```
R2#    sh int dialer 26
Dialer26 is up, line protocol is up (spoofing)
  Hardware is Unknown
  Internet address is 10.1.26.10/32
  MTU 1500 bytes, BW 56 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Closed, loopback not set
  Keepalive set (10 sec)
  DTR is pulsed for 1 seconds on reset
  Interface is bound to Vi2
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 00:08:17
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     5 packets input, 68 bytes
     5 packets output, 62 bytes
Bound to:
Virtual-Access2 is up, line protocol is up
  Hardware is Virtual Access interface
  MTU 1500 bytes, BW 56 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Stopped: CDPCP
  Open: IPCP
  PPPoE vaccess, cloned from Dialer26
  Vaccess status 0x44, loopback not set
  Keepalive set (10 sec)
  Interface is bound to Di26 (Encapsulation PPP)
  Last input 00:00:00, output never, output hang never
  Last clearing of "show interface" counters 00:00:43
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     11 packets input, 148 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
     12 packets output, 142 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

I can ping from R6 to R2 using PPPoE.

```
R6#ping 10.1.26.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.26.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 7.7** Make sure that unnecessary fragmentation is avoided.

The PPP header adds 8 bytes of overhead to each frame. As the default Ethernet MTU is 1500 bytes, it is recommended to lower our MTU on the dialer interface to 1492 to avoid unnecessary fragmentation.

On the client side R2, configure the following:

```
interface dialer 26
mtu 1492
```

**Task 7.8** The client R2 should authenticate when connecting on the server. Create a local account username called R2 with the password of “Password26”.

On the R6, configure the following:

```
username R2 password Password26

interface virtual-template 23
ppp authentication chap callin
```

On R2, configure the following:

```
interface dialer 26
ppp chap password Password26
```

Let’s check that the ping between R6 and R2 using PPPoE is working.

```
R6#ping 10.1.26.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.26.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 7.9** Bundle with PPP multilink the two serial connections between R6 and R9. Use a group ID of 69.

On R6 and R9, configure the following:

```
int multilink 69

interface serial 3/0
encapsulation ppp
ppp multilink group 69
interface serial 3/1
encapsulation ppp
ppp multilink group 69
```

**Task 7.10** Configure the IP address of 10.1.69.6/24 on the R6 PPP multilink69. Configure the IP address of 10.1.69.9/24 on the R9 PPP multilink69. Check that you can ping from R6 to R9.

On R6, configure the following:

```
interface multilink 69
ip address 10.1.69.6 255.255.255.0
no shut
```

On R9, configure the following:

```
interface multilink 69
ip address 10.1.69.9 255.255.255.0
no shut
```

I can ping over the multilink circuit:

```
R6#ping  10.1.69.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.69.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 7.11** Ensure that it is checked on the PPP multilink interfaces that all the fragments of an IP datagram are received on the virtual interfaces before forwarding them.

On R6 and R9, configure the following:

```
int multilink69
ip virtual-reassembly
```

**Task 7.12** There will be voice traffic running over the multilink PPP connection. Ensure that a small voice packet is delayed a maximum of 20 ms because of the transmission of a big data packet.

On R6 and R9, configure the following:

```
int multilink69
ppp multilink fragment delay 20
ppp multilink interleave
```

**Task 7.13** Reserve 1 Mbps in a special queue for real-time packet flows designated to the UDP port starting 32768 and ending 32867.

On R6 and R9, configure the following:

```
class-map match-all RTP
 match ip rtp 32768 100

policy-map RTP
 class RTP
  priority

int multilink69
ppp multilink multiclass
service-policy output RTP
```

**You have completed Lab 7**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 8: Configure and troubleshoot Basic IP routing

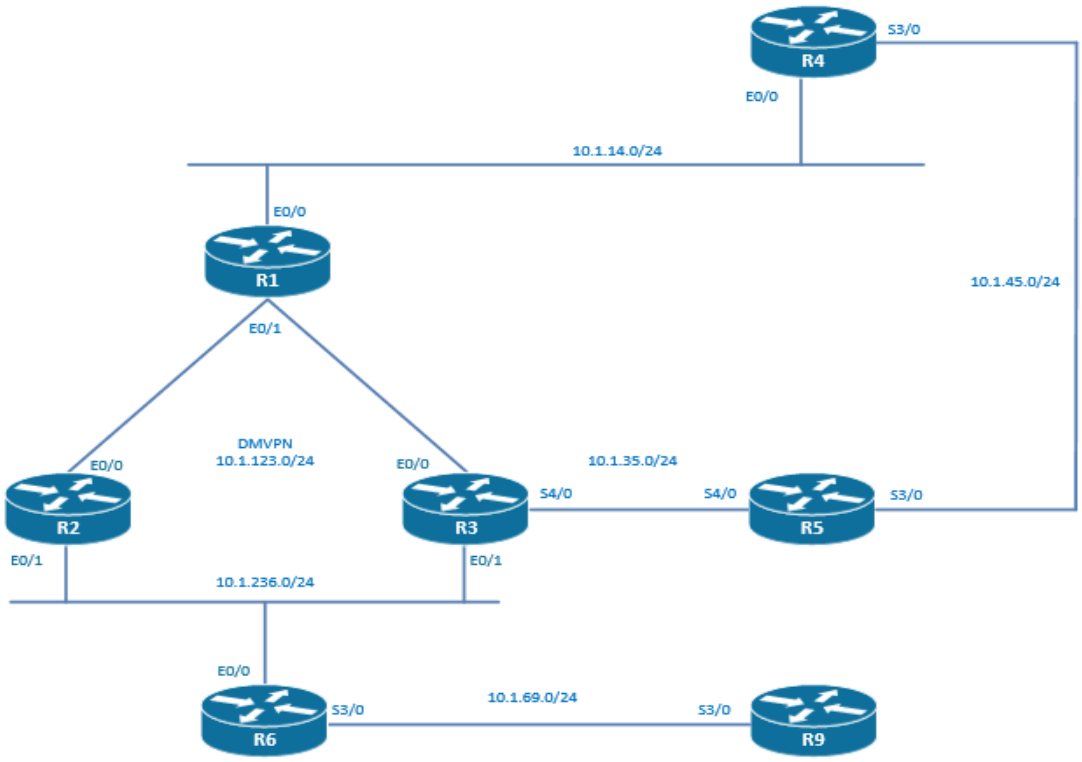
## Technologies covered

- Static route
- Traffic engineering
- Floating static route
- Object tracking
- PBR
- GRE

## Overview

You have been tasked to configure the routing in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 8.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. Configure DMVPN phase 2 as the underlying technology. Multicast support has to be configured.

On R1, configure the following:

```
interface Tunnel23
 ip address 11.1.1.1 255.255.255.0
 no ip redirects
 ip nhrp map multicast dynamic
 ip nhrp network-id 11
 tunnel source 10.1.123.1
 tunnel mode gre multipoint
```

On R2, configure the following:

```
interface Tunnel23
 ip address 11.1.1.2 255.255.255.0
 no ip redirects
 ip nhrp map 11.1.1.1 10.1.123.1
 ip nhrp map multicast 10.1.123.1
 ip nhrp network-id 11
 ip nhrp nhs 11.1.1.1
 tunnel source 10.1.123.2
 tunnel mode gre multipoint
```

On R3, configure the following:

```
interface Tunnel23
 ip address 11.1.1.3 255.255.255.0
 no ip redirects
 ip nhrp map multicast 10.1.123.1
 ip nhrp map 11.1.1.1 10.1.123.1
 ip nhrp network-id 11
 ip nhrp nhs 11.1.1.1
 tunnel source 10.1.123.3
 tunnel mode gre multipoint
```

From R3, I can ping the tunnel interfaces of R1 and R2 and the traceroute from R3 to R2 is not transiting via the hub (as it should be in phase 2).

```
R3#ping 11.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R3#ping 11.1.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/12 ms

R3#traceroute 11.1.1.2
Type escape sequence to abort.
Tracing the route to 11.1.1.2
VRF info: (vrf in name/id, vrf out name/id)
  1 11.1.1.2 0 msec *  1 msec
```

**Task 8.2** On R1, configure a static route to the loopback0 of R3 using the tunnel interface on R3 as the next-hop. Check that you can ping the loopback0 of R3 with a ping sourcing on the tunnel interface of R1.

On R1, configure the following:

```
ip route 10.1.3.3 255.255.255.255 11.1.1.3
```

I can ping the loopback0 R3 from a ping sourcing from the tu23 of R1:

```
R1#ping 10.1.3.3 source 11.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 11.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 8.3** On R2 tunnel interface, disable proxy-arp.

On R2, configure the following:

```
int tu23
no ip proxy-arp
```

**Task 8.4** On R1, configure a static route to the loopback0 of R2 using the tunnel interface on R2 as the egress interface.

On R2, configure the following:

```
ip route 10.1.2.2 255.255.255.255 11.1.1.2
```

**Task 8.5** On R1, ensure that you can ping the loopback0 of R2 with a ping sourcing on the tunnel interface of R1. Create a static arp entry to achieve this task.

I can ping the loopback0 of R2 with a ping sourcing on the tunnel interface of R1. Please note that we are not seeing any ARP entry for the 11.1.1.2 next-hop, which is expected as the tunnel interface does not have any MAC address. As a matter of fact, the IP payload is encapsulated into a GRE packet which is encapsulated into the IP packet using the physical addresses. ARP is mapping those physical IP interfaces to MAC addresses.

```
R1#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.1.14.1              -   aabb.cc00.0100  ARPA   Ethernet0/0
Internet  10.1.123.1             -   aabb.cc00.0110  ARPA   Ethernet0/1
Internet  10.1.123.2            47   aabb.cc00.0200  ARPA   Ethernet0/1
Internet  10.1.123.3            51   aabb.cc00.0300  ARPA   Ethernet0/1
```

We have to create a static ARP entry to achieve this task. It is not necessary to crest this entry for the ping to work because we have a dynamic entry for 10.1.123.2, but we are going to configure a static entry for the ARP entry that is used by the ping.

On R1, configure the following:

```
arp 10.1.123.2 aabb.cc00.0200 arpa
```

**Task 8.6** On R6, configure a static route to network 10.1.0.0/16 pointing to E0/0. Check that you can ping the loopback0 of R2 and R3.

On R6, configure the following:

```
ip route 10.1.0.0 255.255.0.0 E0/0
```

We can ping the loopback0 of R2 and R3.

```
R6#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R6#ping 10.1.3.3
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

Let's analyze what is happening now. When pinging 10.1.2.2, the router, R6 sends an ARP request on the 10.1.236.0/24 network, to get 10.1.2.2's MAC address. It is due to the fact that R6 thinks that the 10.1.2.2 host is directly connected through the 10.1.0.0/16 LAN as we can see in the routing table of R6.

```
R6#sh ip route
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, 3 masks
S        10.1.0.0/16 is directly connected, Ethernet0/0
C        10.1.6.0/24 is directly connected, Loopback0
L        10.1.6.6/32 is directly connected, Loopback0
C        10.1.36.0/24 is directly connected, Serial4/0
L        10.1.36.4/32 is directly connected, Serial4/0
C        10.1.236.0/24 is directly connected, Ethernet0/0
L        10.1.236.6/32 is directly connected, Ethernet0/0
```

R2 sees the ARP request on the E0/1 interface. As Proxy-ARP is enabled by default and because R2 knows how to route to the 10.1.2.2 host, it will respond to the ARP request with the MAC address of the E0/1 interfaces and the IP connectivity is established. Same happens for R3.

**Task 8.7** Disable proxy-arp on E0/1 of R2 and R3. Ensure that you can ping the loopback0 of R2 and R3 with a ping sourcing from the E0/0 ip address of R6.

On R2 and R3, configure the following:

```
int E0/1
no ip proxy-arp
```

Now that we have disable ip proxy-arp, let's see if the pings from R6 to R2 and R3 are still working. Pings are still successful. This is not what we expect, isn't it? It is due to the fact that the ARP entries have not timed out yet. So the pings are still using information that was collected when proxy-arp was running on R2 and R3. As ARP entry is not timing out before four hours!

```
R6#ping 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/5 ms
R6#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R6#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.1.2.2              22   aabb.cc00.0210  ARPA   Ethernet0/0
Internet  10.1.3.3              17   aabb.cc00.0310  ARPA   Ethernet0/0
Internet  10.1.236.2            23   aabb.cc00.0210  ARPA   Ethernet0/0
Internet  10.1.236.6             -   aabb.cc00.0600  ARPA   Ethernet0/0
```

I am using the clear arp command to try to get rid of those 2 entries but for some reason, it doesn't work. I'm reloading the router R6 (after having saved the configuration). That is the ultimate way to clear the ARP cache!

After the reload, the ARP cache has been finally emptied and the pings from R6 to R2 and R3 are not working because ip proxy-arp has been disabled.

```
R6#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.1.236.6              -   aabb.cc00.0600  ARPA   Ethernet0/0

R6#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R6#ping 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

We have to ensure that those pings are working! What we have to do now is to manually configure what proxy-arp is automatically doing.

On R6, configure the following:

```
arp 10.1.2.2 aabb.cc00.0210 arpa
arp 10.1.3.3 aabb.cc00.0310 arpa
```

The ping are again successful.

```
R6#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R6#ping 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 8.8** Configure a GRE tunnel interface Tunnel0 between the loopback0 of R6 and the loopback0 of R3. Use ip address 36.0.0.3/24 on R3 and 36.0.0.6/24 on R6. Configure default routes on R6 and R3 with an AD of 250.

On R6, configure the following:

```
int tu0
ip address 36.0.0.6 255.255.255.0
tunnel source lo0
tunnel destination 10.1.3.3

ip route 0.0.0.0 0.0.0.0 10.1.236.3 250
```

On R3, configure the following:

```
int tu0
ip address 32.0.0.3 255.255.255.0
tunnel source lo0
tunnel destination 10.1.6.6

ip route 0.0.0.0 0.0.0.0 10.1.236.6 250
```

On R3, I can ping the other side of the tunnel interface.

```
R3#ping 36.0.0.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 36.0.0.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 8.9** On R6, configure a static route to the loopback network of the router R3 using the Tu0 as egress with an AD of 5. The tunnel0 interface should go down because of a recursion issue. Leave this tunnel0 down as it is.

On R6, configure the following:

```
ip route 10.1.3.3 255.255.255.255 36.0.0.3 5
R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R6(config)#ip route 10.1.3.3 255.255.255.255 36.0.0.3 5
%ADJ-5-PARENT: Midchain parent maintenance for IP midchain out of Tunnel0 - looped chain
attempting to stack
%SYS-5-CONFIG_I: Configured from console by console
R6#
%TUN-5-RECURDOWN: Tunnel0 temporarily disabled due to recursive routing
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to down
```

The tunnel is going down because the tunnel0 destination, that is to say 10.1.3.3 is routed through the tunnel. It is a recursion issue.

**Task 8.10** Configure static routing so that you can ping the loopback0 of R1 with a ping sourcing from the loopback0 ip address of R6. The ping should follow the R6-R3-R1 route and use the DMVPN tunnel.

On R1, configure the following:

```
ip route 10.1.6.6 255.255.255.255 11.1.1.3
```

On R3, configure the following:

```
ip route 10.1.1.1 255.255.255.255 11.1.1.1
```

This is not enough! On R6, we have a default route towards 10.1.236.3, but this default route will not enter into action because there is a more specific route, that is to say the route to 10.1.0.0/16 added in an earlier question. This route will be used to route to 10.1.1.1. As we have seen before, R6 will ARP for 10.1.1.1. R3 will not reply to this ARP request because Proxy-ARP has been disabled.

On R6, configure the following:

```
arp 10.1.1.1 aabb.cc00.0310 ARPA
```

**Task 8.11** Configure a GRE tunnel interface Tunnel16 between the loopback0 of R6 and the loopback0 of R1. Use ip address 16.0.0.1/24 on R1 and 16.0.0.6/24 on R6.

On R6, configure the following:

```
int tu16
ip address 16.0.0.6 255.255.255.0
tunnel source lo0
tunnel destination 10.1.1.1
```

On R1, configure the following:

```
int tu16
ip address 16.0.0.1 255.255.255.0
tunnel source lo0
tunnel destination 10.1.6.6
```

The tunnel 16 is up and running.

```
R1#ping 16.0.0.6 source 16.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 16.0.0.6, timeout is 2 seconds:
Packet sent with a source address of 16.0.0.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 8.12** On R3, configure a floating static route that will be used in the case that the tunnel interface to R1 goes down. This floating route should not point to R1, but to R5 as a next-hop. At this point, you are not asked to configure all the static routing that will make the backup path operational.

On R3, we have already a default route with an AD of 250. We are going to add one new floating static route towards R5 with an AD of 254 in order not to impact the routing of what has be configured so far.

On R3, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.35.5 254
```

**Task 8.13** On R4, configure a default-route using the next-hop of R1. On R1, configure a static route to the network 10.1.4.0/24 pointing to the next-hop on R4.

On R4, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.14.1
```

On R1, configure the following:

```
ip route 10.1.4.0 255.255.255.0 10.1.14.4
```

From R1, I can ping the loopback0 of R4.

```
R1#ping 10.1.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 8.14** On R4, configure a default-route using the next-hop of R5 with an AD of 5.

On R4, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.45.5 5
```

**Task 8.15** The default-route using the next-hop of R5 should be used when the loopback0 of R1 has become unreachable. Use object tracking and IP SLA.

On R4, we have those 2 default routes.

```
ip route 0.0.0.0 0.0.0.0 10.1.14.1
ip route 0.0.0.0 0.0.0.0 10.1.45.5 5
```

We are going to track the loopback0 of R1 and the default route with a next-hop of 10.1.41.4 would be conditional to the reachability of the loopback0 of R1.

```
ip sla 1
icmp-echo 10.1.1.1
ip sla schedule 1 life forever start-time now

track 1 ip sla 1 reachability

ip route 0.0.0.0 0.0.0.0 10.1.14.1 track 1
```

**Task 8.16** On R5, configure default routing using policy-based routing. This default routing should be pointing to a next-hop of R3 IP address using PBR. When CDP detects that R5 to R3 connectivity is down, the traffic should be routed over R4. Do not use local policy-based routing.

There are two ways to configure Policy based routing with next hop reachability verification, either via CDP or via enhanced object tracking.

```
ip access-list extended DEFAULT_TO_R3
 permit ip any any

route-map POLICY permit 10
 match ip address DEFAULT_TO_R3
 set ip next-hop 10.1.35.3
 set ip next-hop verify-availability
 set ip default next-hop 10.1.45.4

int S3/0
ip address 10.1.45.5 255.255.255.0
ip policy route-map POLICY

int S4/0
ip address 10.1.35.5 255.255.255.0
ip policy route-map POLICY
```

We are using PBR with next hop reachability via CDP. When CDP detects that the set ip next-hop IP address is unreachable, the set ip default next-hop will kick in.

**Task 8.17** On R9, use local-policy based routing to route to the loopback interface of R6.

On R9, configure the following:

```
ip access-list extended Lo0_R6
permit ip host 10.1.9.9 host 10.1.6.6

route-map To_Lo0_R6 permit 10
match ip address Lo0_6
set ip next-hop 10.1.69.6

ip local policy route-map To_Lo0_R6
```

**Task 8.18** On R6, use local-policy based routing to route to the loopback interface of R9. You should be able to ping the loopback0 of R6 with a ping sourcing from the loopback0 of R9.

On R6, configure the following:

```
ip access-list extended Lo0_R9
permit ip host 10.1.6.6 host 10.1.9.9

route-map To_Lo0_R9 permit 10
```

```
match ip address Lo0_R9
set ip next-hop 10.1.69.9

ip local policy route-map To_Lo0_R9
```

We can check that I can ping the loopback0 of R6 with a ping sourcing from the loopback0 of R9 even if there is no routing entry in the routing table. Policy-based routing is taken into account before the routing table.

```
R9#ping 10.1.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms

R9#sh ip route
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, 4 subnets, 2 masks
C        10.1.9.0/24 is directly connected, Loopback0
L        10.1.9.9/32 is directly connected, Loopback0
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.9/32 is directly connected, Serial3/0
```

**You have completed Lab 8**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 9: Configure and troubleshoot Routing Information Protocol (Part 1)

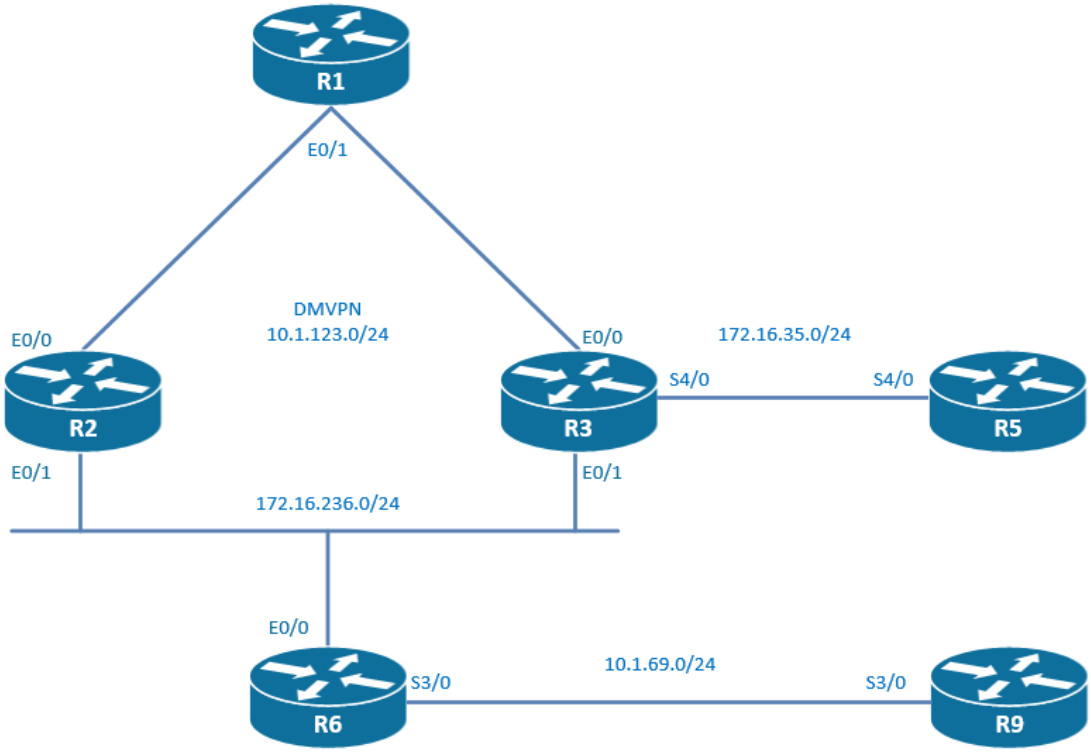
## Technologies covered

- RIP version 2
- Split-horizon
- Auto-summarization
- Send and receive version
- Manual summarization
- Convergence timers
- Offset-list
- Distribute-list
- Per neighbor AD filtering

## Overview

You have been tasked to configure routing in your network using the RIP version 2 protocol.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 9.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Configure RIP version 2 in this DMVPN network.

The DMVPN network phase 2 is already pre-configured. Let's configure RIP version 2 over this DMVPN network. RIPv2 is carrying a subnet mask field but the classless behavior will only take place once the no auto-summary is configured. Even if the network statement is configured as classfull network under the router RIP but VLSM is supported when version 2 and no auto-summary is supported.

On R1, R2 and R3:

```
router rip
 version 2
 network 11.0.0.0
 no auto-summary
```

The RIP protocol is a distance-vector protocol so there is no neighborship relation created. A good command to check the configuration of RIP is the show ip protocols command.

```
R1#sh ip protocols
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 14 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    Tunnel23              2     2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    11.0.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    11.1.1.3             120      00:00:05
    11.1.1.2             120      00:00:09
  Distance: (default is 120)
```

**Task 9.2** Advertise the loopbacks 10 of R1, R2, and R3 in the RIP process.

On R1:

```
router rip
network 1.0.0.0
```

On R2:

```
router rip
network 2.0.0.0
```

On R3:

```
router rip
network 3.0.0.0
```

**Task 9.3** Ensure full reachability in this hub and spoke technology. On R2, check that you can ping the loopback10 of R3 sourcing from the loopback10 of R2.

Let's try to ping the spokes R2 and R3 from the hub R1:

```
R1#ping 3.3.3.3 source 1.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, 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 = 1/1/1 ms
R1#ping 2.2.2.2  source 1.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, 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 = 1/1/1 ms
```

The pings are working and this is validated by the routing table. 2.2.2.2 and 3.3.3.3 are in the routing table.

```
R1#sh 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

      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/1] via 11.1.1.2, 00:00:13, Tunnel23
      3.0.0.0/32 is subnetted, 1 subnets
R        3.3.3.3 [120/1] via 11.1.1.3, 00:00:01, Tunnel23
```

On R2, let's try to ping the  other spoke R3 and the hub R1:

```
R2#ping 1.1.1.1 source 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 2.2.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R2#ping 3.3.3.3 source 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
Packet sent with a source address of 2.2.2.2
.....
Success rate is 0 percent (0/5)
```

The ping from spoke to spoke is not working. Let's check the routing table.

```
R2#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:04, Tunnel23
```

OK, the ping is not working from spoke to spoke because there is no route to the other spoke 3.3.3.3 Let's check if we have a similar situation on R3. We will ping the other spoke R2 and the hub R1:

```
R3#ping 1.1.1.1 source 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 3.3.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
R3#ping 2.2.2.2 source 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 3.3.3.3
.....
Success rate is 0 percent (0/5)


R3#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:07, Tunnel23
```

We have here a general problem; the IP connectivity from hub to spoke is working, but the IP connectivity from spoke to spoke is not working.

Remember that we are using a topology of DMVPN phase 2 so the traffic from spoke to spoke is not transiting via the hub. But the multicast traffic is always transiting via the hub because of the command "ip nhrp map multicast 10.1.123.1" configured on the tunnel interfaces of the spokes, and this is exactly where the problem lays. The RIP protocol is sending multicast updates that are forwarded though the Hub. RIP has a default mechanism not to forward an update from an interface from which the update is received. This is called split-horizon. This makes sense right? But not in a point-to-multipoint topology! In order to enable IP connectivity between the spokes, we have to disable the split-horizon RIP mechanism on the hub router. The feature is on by default.

On R1:

```
int tu23
no ip split-horizon
```

After doing this change on R1, the missing routes are appearing in the routing table of the spokes and the pings from spoke to spoke are working.

```
R2#sh 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
```

```
      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:11, Tunnel23
      3.0.0.0/32 is subnetted, 1 subnets
R        3.3.3.3 [120/2] via 11.1.1.3, 00:00:11, Tunnel23

R2#ping 3.3.3.3 source 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
Packet sent with a source address of 2.2.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R3#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:13, Tunnel23
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/2] via 11.1.1.2, 00:00:13, Tunnel23

R3#ping 2.2.2.2 source 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 3.3.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

Let's check that the DMVPN version 2 setup is working and that traffic going from spoke to spoke is only not transit via the hub.

```
R2#traceroute 3.3.3.3 source 2.2.2.2
Type escape sequence to abort.
Tracing the route to 3.3.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.123.3 0 msec *  1 msec
```

**Task 9.4** Configure RIP version 2 between R5 and R3. Advertise the loopbacks of R5 in the RIP process.

On R5:

```
router rip
 version 2
 network 172.16.0.0
 network 200.0.0.0
 network 201.0.0.0
 network 5.0.0.0
 no auto-summary
```

On R3:

```
router rip
network 172.16.0.0
```

Let's check if we can ping the loopback10 of R2 from R5. Yes, RIP is doing its job.

```
R5#ping 2.2.2.2 source 200.0.0.1
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 200.0.0.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 6/8/9 ms
```

**Task 9.5** Ensure that there is a single 10.0.0.0/8 entry in the routing table of R5. Use manual summarization.

At this moment, the 10.0.0.0 network is not advertised in the RIP protocols, so we have no network to summarize. Let's advertise the 10.0.0.0 network into RIP on R1, R2, and R3.

On R1, R2 and R3:

```
router rip
network 10.0.0.0
```

I can now see the 10.x.x.x networks in the routing table of R5:

```
R5#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/2] via 172.16.35.3, 00:00:02, Serial4/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/2] via 172.16.35.3, 00:00:02, Serial4/0
      3.0.0.0/32 is subnetted, 1 subnets
R        3.3.3.3 [120/1] via 172.16.35.3, 00:00:02, Serial4/0
      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
R        10.1.1.0/24 [120/2] via 172.16.35.3, 00:00:02, Serial4/0
R        10.1.2.0/24 [120/2] via 172.16.35.3, 00:00:02, Serial4/0
R        10.1.3.3/32 [120/1] via 172.16.35.3, 00:00:02, Serial4/0
R        10.1.123.0/24 [120/1] via 172.16.35.3, 00:00:02, Serial4/0
      11.0.0.0/24 is subnetted, 1 subnets
R        11.1.1.0 [120/1] via 172.16.35.3, 00:00:02, Serial4/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R        172.16.236.0/24 [120/1] via 172.16.35.3, 00:00:02, Serial4/0
```

The question states that R5 should have in the routing table a summary route representing all the 10.x.x.x networks. The RIP protocol has to aggregate on R3 all the 10.x.x.x networks when advertising routes to R5. This is called manual summarization. Manual summarization is taking place on an interface level.

The following has to be configured on R3 on the interface pointing to R5.

```
int s4/0
ip summary-address rip 10.0.0.0 255.0.0.0
```

On R5, we can now see that all the /24 and /32 routes are not refreshed anymore. The route age for those routes is increasing. At the time of the capture below, the routes weren't refreshed for 49 seconds. Those routes are staying in the routing table until the age is reaching 3 minutes. We are using the default RIP timers. With RIP, you have to be patient. This protocol is extremely slow!

```
R5#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/2] via 172.16.35.3, 00:00:19, Serial4/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/2] via 172.16.35.3, 00:00:19, Serial4/0
      3.0.0.0/32 is subnetted, 1 subnets
R        3.3.3.3 [120/1] via 172.16.35.3, 00:00:19, Serial4/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 3 masks
R        10.0.0.0/8 [120/1] via 172.16.35.3, 00:00:19, Serial4/0
R        10.1.1.0/24 [120/2] via 172.16.35.3, 00:00:49, Serial4/0
R        10.1.2.0/24 [120/2] via 172.16.35.3, 00:00:49, Serial4/0
R        10.1.3.3/32 [120/1] via 172.16.35.3, 00:00:49, Serial4/0
R        10.1.123.0/24 [120/1] via 172.16.35.3, 00:00:49, Serial4/0
      11.0.0.0/24 is subnetted, 1 subnets
R        11.1.1.0 [120/1] via 172.16.35.3, 00:00:19, Serial4/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R        172.16.236.0/24 [120/1] via 172.16.35.3, 00:00:19, Serial4/0
```

When the 3 minutes aging is reached, the routes are timed out. Let's check the routing table now.

```
R5#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/2] via 172.16.35.3, 00:00:25, Serial4/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/2] via 172.16.35.3, 00:00:25, Serial4/0
      3.0.0.0/32 is subnetted, 1 subnets
R        3.3.3.3 [120/1] via 172.16.35.3, 00:00:25, Serial4/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 3 masks
R        10.0.0.0/8 [120/1] via 172.16.35.3, 00:00:25, Serial4/0
R        10.1.1.0/24 is possibly down,
          routing via 172.16.35.3, Serial
R        10.1.2.0/24 is possibly down,
          routing via 172.16.35.3, Serial
R        10.1.3.3/32 is possibly down,
          routing via 172.16.35.3, Serial
R        10.1.123.0/24 is possibly down,
          routing via 172.16.35.3, Seri
      11.0.0.0/24 is subnetted, 1 subnets
R        11.1.1.0 [120/1] via 172.16.35.3, 00:00:25, Serial4/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R        172.16.236.0/24 [120/1] via 172.16.35.3, 00:00:25, Serial4/0
```

Oh my god, the routes are still there! 10.1.123.0/24 possible down…. Why on earth would RIP not update its routing table? This is because of the flush timers. 3 more minutes to wait! Remember that we are running the default timers.

Let's check again the routing table of R5.

```
R5#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/2] via 172.16.35.3, 00:00:20, Serial4/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/2] via 172.16.35.3, 00:00:20, Serial4/0
      3.0.0.0/32 is subnetted, 1 subnets
R        3.3.3.3 [120/1] via 172.16.35.3, 00:00:20, Serial4/0
R     10.0.0.0/8 [120/1] via 172.16.35.3, 00:00:20, Serial4/0
      11.0.0.0/24 is subnetted, 1 subnets
R        11.1.1.0 [120/1] via 172.16.35.3, 00:00:20, Serial4/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R        172.16.236.0/24 [120/1] via 172.16.35.3, 00:00:20, Serial4/0
```

This time we have the summary route only in the routing table as it was required in this task. RIP is definitely not the number 1 protocol regarding convergence!

**Task 9.6** Ensure that the network 200.0.0.0/24 is advertised to the router R3. Do not use manual summarization.

On R3, the routing table is currently containing each of the /30 networks of the 200.x.x.x networks which are loopbacks located on the R5 router. The task instructs us that R3 should see a single entry /24 for all those 3 networks. Manual summarization is not allowed. The only way to summarize is therefore to use automatic summarization.

```
R3#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:20, Tunnel23
                 [120/1] via 10.1.123.1, 00:00:24, Ethernet0/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/1] via 10.1.123.2, 00:00:17, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:20, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:24, Ethernet0/0
R        10.1.2.0/24 [120/1] via 10.1.123.2, 00:00:17, Ethernet0/0
      200.0.0.0/30 is subnetted, 3 subnets
R        200.0.0.0 [120/1] via 172.16.35.5, 00:00:02, Serial4/0
R        200.0.0.4 [120/1] via 172.16.35.5, 00:00:02, Serial4/0
R        200.0.0.8 [120/1] via 172.16.35.5, 00:00:02, Serial4/0
```

Let's configure automatic summarization on R5:

```
router rip
auto-summary
```

Let's check the routing table of R3. It looks promising. The 200.x.x.x /30 networks are not refreshed anymore and are slowly timing out.

```
R3#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:23, Tunnel23
                 [120/1] via 10.1.123.1, 00:00:00, Ethernet0/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/1] via 10.1.123.2, 00:00:24, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:23, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:00, Ethernet0/0
R        10.1.2.0/24 [120/1] via 10.1.123.2, 00:00:24, Ethernet0/0
      200.0.0.0/24 is variably subnetted, 4 subnets, 2 masks
R        200.0.0.0/24 [120/1] via 172.16.35.5, 00:00:13, Serial4/0
R        200.0.0.0/30 [120/1] via 172.16.35.5, 00:01:06, Serial4/0
R        200.0.0.4/30 [120/1] via 172.16.35.5, 00:01:06, Serial4/0
R        200.0.0.8/30 [120/1] via 172.16.35.5, 00:01:06, Serial4/0
```

Possibly down status now. A little bit more of patience and the magic will happen!

```
R3#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:09, Tunnel23
                 [120/1] via 10.1.123.1, 00:00:11, Ethernet0/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/1] via 10.1.123.2, 00:00:08, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:09, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:11, Ethernet0/0
R        10.1.2.0/24 [120/1] via 10.1.123.2, 00:00:08, Ethernet0/0
      200.0.0.0/24 is variably subnetted, 4 subnets, 2 masks
R        200.0.0.0/24 [120/1] via 172.16.35.5, 00:00:22, Serial4/0
R        200.0.0.0/30 is possibly down,
          routing via 172.16.35.5, Seria
R        200.0.0.4/30 is possibly down,
          routing via 172.16.35.5, Seria
R        200.0.0.8/30 is possibly down,
          routing via 172.16.35.5, Seria
```

That's the result that we expected. There is now only the 200.0.0.0/24 entry in the routing table.

```
R3#sh 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

      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 11.1.1.1, 00:00:20, Tunnel23
                 [120/1] via 10.1.123.1, 00:00:25, Ethernet0/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/1] via 10.1.123.2, 00:00:16, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:20, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:25, Ethernet0/0
R        10.1.2.0/24 [120/1] via 10.1.123.2, 00:00:16, Ethernet0/0
R     200.0.0.0/24 [120/1] via 172.16.35.5, 00:00:06, Serial4/0
```

### Task 9.7 Enable RIP on the 172.16.236.0/24 network.

On R2, R3:

```
router rip
network 172.16.0.0
```

RIP is still not running on R6. As nothing is specified, I assume that version 2 and auto-summary is the part of a "default "RIP deployment.

```
router rip
network 172.16.0.0
version 2
auto-summary
```

### Task 9.8 Advertise the loopbacks 0 and 1 of R6 in the RIP process. R6 is running version 1.

Contrary to what I assumed in the previous task, it is now explicitly asked to run version 1 on R6. Let's configure version 1 and route the loopbacks of R6 using the RIP process.

On R6:

```
router rip
no version 2
network 21.0.0.0
network 22.0.0.0
network 23.0.0.0
network 24.0.0.0
```

The difference between version 1 and version 2 is that version 1 is not supporting VSLM. Only glassful networks will be advertised as the subnet mask is not carried into the version 1 packet.

Let's look for the networks 21.x.x.x, 22.x.x.x, 23.x.x.x and 24.x.x.x in the routing table. We expect to see there this 4 networks with a /8 subnet mask.

```
R1#sh 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

      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/1] via 11.1.1.2, 00:00:06, Tunnel23
                 [120/1] via 10.1.123.2, 00:00:13, Ethernet0/1
      3.0.0.0/32 is subnetted, 1 subnets
R        3.3.3.3 [120/1] via 11.1.1.3, 00:00:12, Tunnel23
                 [120/1] via 10.1.123.3, 00:00:14, Ethernet0/1
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.2.0/24 [120/1] via 11.1.1.2, 00:00:06, Tunnel23
                     [120/1] via 10.1.123.2, 00:00:13, Ethernet0/1
R        10.1.3.3/32 [120/1] via 11.1.1.3, 00:00:12, Tunnel23
                     [120/1] via 10.1.123.3, 00:00:14, Ethernet0/1
      172.16.0.0/24 is subnetted, 2 subnets
R        172.16.35.0 [120/1] via 11.1.1.3, 00:00:12, Tunnel23
                     [120/1] via 10.1.123.3, 00:00:14, Ethernet0/1
R        172.16.236.0 [120/1] via 11.1.1.3, 00:00:12, Tunnel23
                      [120/1] via 11.1.1.2, 00:00:06, Tunnel23
                      [120/1] via 10.1.123.3, 00:00:14, Ethernet0/1
                      [120/1] via 10.1.123.2, 00:00:13, Ethernet0/1
      200.0.0.0/24 is subnetted, 1 subnets
R        200.0.0.0 [120/2] via 11.1.1.3, 00:00:12, Tunnel23
                   [120/2] via 10.1.123.3, 00:00:14, Ethernet0/1
```

The networks are not in the R1 routing table. The configuration on R6 appears to be correct. Let's run debug on R2 and R3 and monitor if R6 is sending updates.

```
R3#debug ip rip events
RIP event debugging is on
R3#
RIP: received v2 update from 172.16.236.2 on Ethernet0/1
RIP: Update contains 6 routes
R3#
RIP: ignored v1 packet from 172.16.236.6 (illegal version)
```

We can see from the debugs that the version 1 packets are getting ignored by R3. This is going to be fixed in the next task.

**Task 9.9** Make sure that the interfaces part of network 172.16.236.0/24 can send and receive either version 1 or version 2 packets.

On R2 and R3:

```
int e0/1
ip rip send version 1
ip rip receive version 1
```

Let's now check the routing table of R1:

```
R1#sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:23, Ethernet0/1
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
R        2.2.2.2/32 [120/1] via 11.1.1.2, 00:00:13, Tunnel23
                    [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                   [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R        3.3.3.3/32 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.2.0/24 [120/1] via 11.1.1.2, 00:00:13, Tunnel23
                     [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
R        10.1.3.3/32 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
R     21.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R     22.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R     23.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R     24.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
      172.16.0.0/24 is subnetted, 2 subnets
R        172.16.35.0 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
R        172.16.236.0 [120/1] via 11.1.1.2, 00:00:13, Tunnel23
                      [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
R     200.0.0.0/24 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:23, Ethernet0/1
R     201.0.0.0/24 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
```

We can now see that the loopbacks of R6 are present in the routing table with the expected /8 mask. This is due to the fact that R6 is running on RIP version 1 and RIP version 1 only advertises classfull networks.

**Task 9.10** Configure RIP MD5 authentication on the 11.1.1.0/24 network. Use a key chain of “iPexpertchain”, a key number 1, and a key-string of ”iPpassword”.

Authentication is only supported on RIP version 2. We are running version 2 on all the routers involved with the 11.1.1.0/24 network, that is to say R1, R2, and R3.

For MD5 authentication, the key number and the key string have to match because both are used to generate the hash. For clear-text authentication, only the key string has to match.

Let’s configure MD5 authentication on the tunnel interfaces of R1, R2, and R3:

```
key chain iPexpertchain
 key 1
   key-string iPpassword

interface Tu23
 ip rip authentication mode md5
 ip rip authentication key-chain iPexpertchain

R1#sh ip protocols
*** IP Routing is NSF aware ***

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 25 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    Ethernet0/1           2     2
    Loopback0             2     2
    Loopback10            2     2
    Tunnel23              2     2                   iPexpertchain
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    1.0.0.0
    10.0.0.0
    11.0.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    11.1.1.3             120      00:00:22
    11.1.1.2             120      00:00:10
    10.1.123.2           120      00:00:15
    10.1.123.3           120      00:00:23
  Distance: (default is 120)
```

We can see with the “sh ip protocols” command that the encryption is in place on the tunnel interface. Moreover, the router R1 is still receiving the RIP updates from R2 and R3 after the encryption is enforced.

**Task 9.11** On R2, the network 200.0.0.0/8 received on Ethernet0/0 should be rejected, and the network 201.0.0.0/8 received on Ethernet0/1 should be rejected. Do not use distribute-list or administrative distance poisoning.

Let’s check the current situation before applying the filtering:

```
R2#sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/2] via 172.16.236.3, 00:00:24, Ethernet0/1
R        1.1.1.1/32 [120/1] via 11.1.1.1, 00:00:03, Tunnel23
                    [120/1] via 10.1.123.1, 00:00:09, Ethernet0/0
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/2] via 10.1.123.3, 00:00:06, Ethernet0/0
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/1] via 172.16.236.3, 00:00:24, Ethernet0/1
R        3.3.3.3/32 [120/1] via 10.1.123.3, 00:00:06, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:03, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:09, Ethernet0/0
R        10.1.3.3/32 [120/1] via 10.1.123.3, 00:00:06, Ethernet0/0
R     21.0.0.0/8 [120/1] via 172.16.236.6, 00:00:21, Ethernet0/1
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R        172.16.35.0/24 [120/1] via 172.16.236.3, 00:00:24, Ethernet0/1
                        [120/1] via 10.1.123.3, 00:00:06, Ethernet0/0
```

```
      200.0.0.0/24 is subnetted, 1 subnets
R        200.0.0.0 [120/2] via 172.16.236.3, 00:00:24, Ethernet0/1
                   [120/2] via 10.1.123.3, 00:00:06, Ethernet0/0
R     201.0.0.0/24 [120/2] via 172.16.236.3, 00:00:24, Ethernet0/1
                   [120/2] via 10.1.123.3, 00:00:06, Ethernet0/0
```

On R2, the loopbacks of R5, 200.0.0.0/24 and 201.0.0.0/24, are reachable and load-balanced between Ethernet0/0 and Ethernet 0/1 because this is the shortest route to R5 according to the hop count metric used by RIP (2 hops away).

As we are told not to use distribute-list or administrative distance poisoning, the only option that we have is to use offset-list. If the hop count reaches 16, the network is considered unreachable and the update is rejected.

We are adding an “artificial” hop-count of 15 to the existing real hop-count on the updates arriving on interface e0/0 and this hop-count will only apply to network 200.0.0.0. This way, R2 will consider 200.0.0.0 to be unreachable via interface e0/0.

We are adding an “artificial” hop-count of 15 to the existing real hop-count on the updates arriving on interface e0/1 and this hop-count will only apply to network 201.0.0.0. This way, R2 will consider 201.0.0.0 to be unreachable via interface e0/1.

The following configuration has to be applied on R2:

```
access-list 1 permit 200.0.0.0
access-list 2 permit 201.0.0.0
!
router rip
 offset-list 1 in 15 e0/0
 offset-list 2 in 15 e0/1
```

Let’s see the effect that those commands have in the R2 routing table:

```
R2# sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/2] via 172.16.236.3, 00:00:13, Ethernet0/1
R        1.1.1.1/32 [120/1] via 11.1.1.1, 00:00:22, Tunnel23
                    [120/1] via 10.1.123.1, 00:00:03, Ethernet0/0
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/2] via 10.1.123.3, 00:00:17, Ethernet0/0
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/1] via 172.16.236.3, 00:00:13, Ethernet0/1
R        3.3.3.3/32 [120/1] via 10.1.123.3, 00:00:17, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:22, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:03, Ethernet0/0
R        10.1.3.3/32 [120/1] via 10.1.123.3, 00:00:17, Ethernet0/0
R     21.0.0.0/8 [120/1] via 172.16.236.6, 00:00:26, Ethernet0/1
R     22.0.0.0/8 [120/1] via 172.16.236.6, 00:00:26, Ethernet0/1
R     23.0.0.0/8 [120/1] via 172.16.236.6, 00:00:26, Ethernet0/1
R     24.0.0.0/8 [120/1] via 172.16.236.6, 00:00:26, Ethernet0/1
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
```

```
R        172.16.35.0/24 [120/1] via 172.16.236.3, 00:00:13, Ethernet0/1
                        [120/1] via 10.1.123.3, 00:00:17, Ethernet0/0
R     200.0.0.0/24 [120/2] via 172.16.236.3, 00:00:13, Ethernet0/1
R     201.0.0.0/24 [120/2] via 10.1.123.3, 00:00:17, Ethernet0/0
```

The network 200.0.0.0/24 is reachable only via the interface Ethernet 0/1 and the network 201.0.0.0/24 is reachable only via the interface Ethernet 0/0. No more load-balancing. Mission accomplished.

**Task 9.12** On R1, all the traffic should be sent to R2 and R3 should never be considered as a next hop. Do not use offset-list or administrative distance poisoning. Configure 2 Prefix-lists.

Let's check the current routing table of R1.

To reach the loopback of R3, namely the 3.0.0.0/8, the traffic is routed directly to R3 through either Tunnel 23 or Ethernet0/1.

To reach the loopback of R6, for example the 21.0.0.0/8, the traffic is load-balanced between R2 and R3.

```
R1#sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/3] via 11.1.1.3, 00:00:11, Tunnel23
                   [120/3] via 11.1.1.2, 00:00:09, Tunnel23
                   [120/3] via 10.1.123.3, 00:00:27, Ethernet0/1
                   [120/3] via 10.1.123.2, 00:00:23, Ethernet0/1
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/2] via 11.1.1.3, 00:00:11, Tunnel23
                   [120/2] via 10.1.123.3, 00:00:27, Ethernet0/1
R        2.2.2.2/32 [120/1] via 11.1.1.2, 00:00:09, Tunnel23
                    [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/2] via 11.1.1.2, 00:00:09, Tunnel23
                   [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R        3.3.3.3/32 [120/1] via 11.1.1.3, 00:00:11, Tunnel23
                    [120/1] via 10.1.123.3, 00:00:27, Ethernet0/1
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.2.0/24 [120/1] via 11.1.1.2, 00:00:09, Tunnel23
                     [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
R        10.1.3.3/32 [120/1] via 11.1.1.3, 00:00:11, Tunnel23
                     [120/1] via 10.1.123.3, 00:00:27, Ethernet0/1
R     21.0.0.0/8 [120/2] via 11.1.1.3, 00:00:11, Tunnel23
                 [120/2] via 11.1.1.2, 00:00:09, Tunnel23
                 [120/2] via 10.1.123.3, 00:00:27, Ethernet0/1
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R     22.0.0.0/8 [120/2] via 11.1.1.3, 00:00:11, Tunnel23
                 [120/2] via 11.1.1.2, 00:00:09, Tunnel23
                 [120/2] via 10.1.123.3, 00:00:27, Ethernet0/1
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
      172.16.0.0/24 is subnetted, 2 subnets
```

```
R        172.16.35.0 [120/1] via 11.1.1.3, 00:00:11, Tunnel23
                     [120/1] via 10.1.123.3, 00:00:27, Ethernet0/1
R        172.16.236.0 [120/1] via 11.1.1.3, 00:00:11, Tunnel23
                      [120/1] via 11.1.1.2, 00:00:09, Tunnel23
                      [120/1] via 10.1.123.3, 00:00:27, Ethernet0/1
                      [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
R     200.0.0.0/24 [120/2] via 11.1.1.3, 00:00:11, Tunnel23
                   [120/2] via 10.1.123.3, 00:00:27, Ethernet0/1
R     201.0.0.0/24 [120/2] via 11.1.1.3, 00:00:11, Tunnel23
                   [120/2] via 10.1.123.3, 00:00:27, Ethernet0/1
```

We are instructed not to use any offset-list or AD manipulation. The only way to achieve the desired filtering is therefore to use distribute-list. We create a first prefix-list called FILTER that specifies which networks are going to be processed by the second filter to filter the next-hop. We create a second prefix-list called NOT-R3 that is specifying which next-hop will be denied for the networks specified in the first filter.

On R1, let's configure the following:

```
ip prefix-list FILTER seq 10 permit 0.0.0.0/0 le 32
ip prefix-list NOT-R3 seq 5 deny 11.1.1.3/32
ip prefix-list NOT-R3 seq 6 deny 10.1.123.3/32
ip prefix-list NOT-R3 seq 10 permit 0.0.0.0/0 le 32
router rip
distribute-list prefix FILTER gateway NOT-R3 in
```

Let's have a look at the routing table of R1 once the distribute-list is applied:

```
R1#sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/3] via 11.1.1.2, 00:00:11, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:21, Ethernet0/1
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/3] via 11.1.1.2, 00:00:11, Tunnel23
R        2.2.2.2/32 [120/1] via 11.1.1.2, 00:00:11, Tunnel23
                    [120/1] via 10.1.123.2, 00:00:21, Ethernet0/1
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
                   [120/2] via 10.1.123.2, 00:00:21, Ethernet0/1
R        3.3.3.3/32 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.2.0/24 [120/1] via 11.1.1.2, 00:00:11, Tunnel23
                     [120/1] via 10.1.123.2, 00:00:21, Ethernet0/1
R        10.1.3.3/32 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
R     21.0.0.0/8 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:21, Ethernet0/1
R     22.0.0.0/8 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:21, Ethernet0/1
R     23.0.0.0/8 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:21, Ethernet0/1
R     24.0.0.0/8 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:21, Ethernet0/1
      172.16.0.0/24 is subnetted, 2 subnets
R        172.16.35.0 [120/2] via 11.1.1.2, 00:00:11, Tunnel23
```

```
R        172.16.236.0 [120/1] via 11.1.1.2, 00:00:11, Tunnel23
                      [120/1] via 10.1.123.2, 00:00:21, Ethernet0/1
R     200.0.0.0/24 [120/3] via 11.1.1.2, 00:00:11, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:21, Ethernet0/1
R     201.0.0.0/24 [120/3] via 11.1.1.2, 00:00:11, Tunnel23
```

To reach the loopback of R3, namely the 3.0.0.0/8, the traffic is not routed directly to R3. 3.0.0.0/8 is now reachable via R2 even if R3 is directly connected.

To reach the loopbacks of R6, for example the 21.0.0.0/8, the traffic is not load-balanced anymore between R2 and R3. 21.0.0.0/8 is reachable only via R2.

**Task 9.13** On R1, the network 23.0.0.0/8 should be routed via the tu23 and the network 24.0.0.0/8 should be routed via the E0/1. Use administrative distance poisoning.

Let's have a look at the routing table on R1 before applying the filtering.

```
R1#sh 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
      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:23, Ethernet0/1
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
R        2.2.2.2/32 [120/1] via 11.1.1.2, 00:00:13, Tunnel23
                    [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                   [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R        3.3.3.3/32 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.2.0/24 [120/1] via 11.1.1.2, 00:00:13, Tunnel23
                     [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
R        10.1.3.3/32 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
R     21.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R     22.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R     23.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
R     24.0.0.0/8 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:23, Ethernet0/1
      172.16.0.0/24 is subnetted, 2 subnets
R        172.16.35.0 [120/2] via 11.1.1.2, 00:00:13, Tunnel23
R        172.16.236.0 [120/1] via 11.1.1.2, 00:00:13, Tunnel23
                      [120/1] via 10.1.123.2, 00:00:23, Ethernet0/1
R     200.0.0.0/24 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:23, Ethernet0/1
R     201.0.0.0/24 [120/3] via 11.1.1.2, 00:00:13, Tunnel23
```

To reach the network 23.0.0.0/8, the traffic is load-balanced between Tunnel23 and Ethernet0/1.
To reach the network 24.0.0.0/8, the traffic is load-balanced between Tunnel23 and Ethernet0/1.

We have to use AD poisoning in order to engineer the traffic as requested. The administrative distance of a RIP update is 120. We can increase the AD of a RIP update and make it less preferred. It is important to remember that changing ADs is locally significant and that it will only impact the routing decision on the router where ADs are modified and not be propagated. This is true for all the other routing protocols.

We create an access-list to specify for which network we are going to change the AD.
In the distance command, we are going to specify an AD of 255 for the network specified in the access-list previously created. We are also going to specify directly in the distance command the IP address from which the update is originated.

Configure on R1 the following:

```
access-list 1 permit 23.0.0.0
access-list 2 permit 24.0.0.0
router rip
distance 255 10.1.123.2 0.0.0.0 1
distance 255 11.1.1.2 0.0.0.0 2
```

Let's have a look at the routing table of R1 after applying the filters.

```
R1#sh 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
      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/3] via 11.1.1.2, 00:00:18, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:09, Ethernet0/1
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/3] via 11.1.1.2, 00:00:18, Tunnel23
R        2.2.2.2/32 [120/1] via 11.1.1.2, 00:00:18, Tunnel23
                    [120/1] via 10.1.123.2, 00:00:09, Ethernet0/1
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/2] via 11.1.1.2, 00:00:18, Tunnel23
                   [120/2] via 10.1.123.2, 00:00:09, Ethernet0/1
R        3.3.3.3/32 [120/2] via 11.1.1.2, 00:00:18, Tunnel23
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R        10.1.2.0/24 [120/1] via 11.1.1.2, 00:00:18, Tunnel23
                     [120/1] via 10.1.123.2, 00:00:09, Ethernet0/1
R        10.1.3.3/32 [120/2] via 11.1.1.2, 00:00:18, Tunnel23
R     21.0.0.0/8 [120/2] via 11.1.1.2, 00:00:18, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:09, Ethernet0/1
R     22.0.0.0/8 [120/2] via 11.1.1.2, 00:00:18, Tunnel23
                 [120/2] via 10.1.123.2, 00:00:09, Ethernet0/1
R     23.0.0.0/8 [120/2] via 11.1.1.2, 00:00:18, Tunnel23
R     24.0.0.0/8 [120/2] via 10.1.123.2, 00:00:09, Ethernet0/1
      172.16.0.0/24 is subnetted, 2 subnets
R        172.16.35.0 [120/2] via 11.1.1.2, 00:00:18, Tunnel23
R        172.16.236.0 [120/1] via 11.1.1.2, 00:00:18, Tunnel23
                      [120/1] via 10.1.123.2, 00:00:09, Ethernet0/1
R     200.0.0.0/24 [120/3] via 11.1.1.2, 00:00:18, Tunnel23
                   [120/3] via 10.1.123.2, 00:00:09, Ethernet0/1
R     201.0.0.0/24 [120/3] via 11.1.1.2, 00:00:18, Tunnel23
```

To reach the network 23.0.0.0/8, the traffic is not load-balanced between Tunnel23 and Ethernet0/1. It is using only Tunnel23.

To reach the network 24.0.0.0/8, the traffic is not load-balanced between Tunnel23 and Ethernet0/1. It is using only Ethernet0/1.

The AD poisoning is working as expected.

**Task 9.14** Configure RIP filtering so that R3 does not learn 5.0.0.0/24. Do not use any access-list, distribute-list, and do not change AD values. R5 should learn all RIP subnets.

Let's check the routing table before modification:

```
R3#sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/2] via 172.16.236.2, 00:00:04, Ethernet0/1
R        1.1.1.1/32 [120/1] via 11.1.1.1, 00:00:13, Tunnel23
                    [120/1] via 10.1.123.1, 00:00:03, Ethernet0/0
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/1] via 172.16.236.2, 00:00:04, Ethernet0/1
R        2.2.2.2/32 [120/1] via 10.1.123.2, 00:00:05, Ethernet0/0
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/2] via 10.1.123.2, 00:00:05, Ethernet0/0
R     5.0.0.0/8 [120/1] via 172.16.35.5, 00:00:06, Serial4/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:13, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:03, Ethernet0/0
R        10.1.2.0/24 [120/1] via 10.1.123.2, 00:00:05, Ethernet0/0
R     21.0.0.0/8 [120/1] via 172.16.236.6, 00:00:23, Ethernet0/1
R     22.0.0.0/8 [120/1] via 172.16.236.6, 00:00:23, Ethernet0/1
R     23.0.0.0/8 [120/1] via 172.16.236.6, 00:00:23, Ethernet0/1
R     24.0.0.0/8 [120/1] via 172.16.236.6, 00:00:23, Ethernet0/1
R     200.0.0.0/24 [120/1] via 172.16.35.5, 00:00:06, Serial4/0
R     201.0.0.0/24 [120/1] via 172.16.35.5, 00:00:06, Serial4/0
```

As we are not allowed to use any access-list, distribute-list and do not change AD values, we have to be creative. Let's put the interface serial4/0 on the R5 router in passive mode.

```
router rip
passive-interface s4/0
```

By applying this configuration, the network 5.0.0.0/8 update should disappear from the routing table of R3. Let's check it. (Remember to be patient; RIP with default timers is slow!)

```
R3#sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/2] via 172.16.236.2, 00:00:14, Ethernet0/1
R        1.1.1.1/32 [120/1] via 11.1.1.1, 00:00:26, Tunnel23
                    [120/1] via 10.1.123.1, 00:00:19, Ethernet0/0
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/1] via 172.16.236.2, 00:00:14, Ethernet0/1
R        2.2.2.2/32 [120/1] via 10.1.123.2, 00:00:13, Ethernet0/0
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/2] via 10.1.123.2, 00:00:13, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R        10.1.1.0/24 [120/1] via 11.1.1.1, 00:00:26, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:19, Ethernet0/0
R        10.1.2.0/24 [120/1] via 10.1.123.2, 00:00:13, Ethernet0/0
R     21.0.0.0/8 [120/1] via 172.16.236.6, 00:00:18, Ethernet0/1
R     22.0.0.0/8 [120/1] via 172.16.236.6, 00:00:18, Ethernet0/1
R     23.0.0.0/8 [120/1] via 172.16.236.6, 00:00:18, Ethernet0/1
R     24.0.0.0/8 [120/1] via 172.16.236.6, 00:00:18, Ethernet0/1
```

We managed to remove the 5.0.0.0/8 routing entries from the routing table of R3. The task stated that R5 should still have all the RIP routing entries. Let's confirm this is also the case.

```
R5#sh 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

      1.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        1.0.0.0/8 [120/3] via 172.16.35.3, 00:00:18, Serial4/0
R        1.1.1.1/32 [120/2] via 172.16.35.3, 00:00:18, Serial4/0
      2.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        2.0.0.0/8 [120/2] via 172.16.35.3, 00:00:18, Serial4/0
R        2.2.2.2/32 [120/2] via 172.16.35.3, 00:00:18, Serial4/0
      3.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R        3.0.0.0/8 [120/3] via 172.16.35.3, 00:00:18, Serial4/0
R        3.3.3.3/32 [120/1] via 172.16.35.3, 00:00:18, Serial4/0
R     10.0.0.0/8 [120/1] via 172.16.35.3, 00:00:18, Serial4/0
      11.0.0.0/24 is subnetted, 1 subnets
R        11.1.1.0 [120/1] via 172.16.35.3, 00:00:18, Serial4/0
R     21.0.0.0/8 [120/2] via 172.16.35.3, 00:00:18, Serial4/0
R     22.0.0.0/8 [120/2] via 172.16.35.3, 00:00:18, Serial4/0
R     23.0.0.0/8 [120/2] via 172.16.35.3, 00:00:18, Serial4/0
R     24.0.0.0/8 [120/2] via 172.16.35.3, 00:00:18, Serial4/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R        172.16.236.0/24 [120/1] via 172.16.35.3, 00:00:18, Serial4/0
```

R5 is still receiving the RIP routes even if the interface s4/0 is in passive-mode. We have completed successfully this task.

### Task 9.15 Configure the RIP timers on R1, R2, and R3 to 20 second updates, 40 second invalid, 10 second hold, and 80 second flush.

Let's modify the default timers on R1, R2 and R3:

```
router rip
timers basic 20 40 10 80
```

Let's check on R3 that those timers have been modified:

```
R3#sh ip protocols
*** IP Routing is NSF aware ***

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 20 seconds, next due in 17 seconds
  Invalid after 40 seconds, hold down 10, flushed after 80
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    Ethernet0/0           2     2
    Ethernet0/1           1     1
    Serial4/0             2     2
    Loopback0             2     2
    Loopback10            2     2
    Tunnel23              2     2                   iPexpertchain
  Automatic network summarization is not in effect
  Address Summarization:
    10.0.0.0/8 for Serial4/0
  Maximum path: 4
  Routing for Networks:
    3.0.0.0
    10.0.0.0
    11.0.0.0
    172.16.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    11.1.1.1             120      00:00:15
    172.16.236.2         120      00:00:13
    Gateway         Distance      Last Update
    172.16.236.6         120      00:00:03
    10.1.123.1           120      00:00:11
    10.1.123.2           120      00:00:16
    172.16.35.5          120      00:22:43
  Distance: (default is 120)
```

**Task 9.16** On R3, configure Serial4/0 to send updates every 6 seconds towards R5.

Let's modify the timer on the S4/0 interface of R3:

```
interface Serial4/0
ip rip advertise 6
```

The command configured on the interface overwrites the command configured under the rip process command.

**You have completed Lab 9**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 10: Configure and troubleshoot Routing Information Protocol (Part 2)

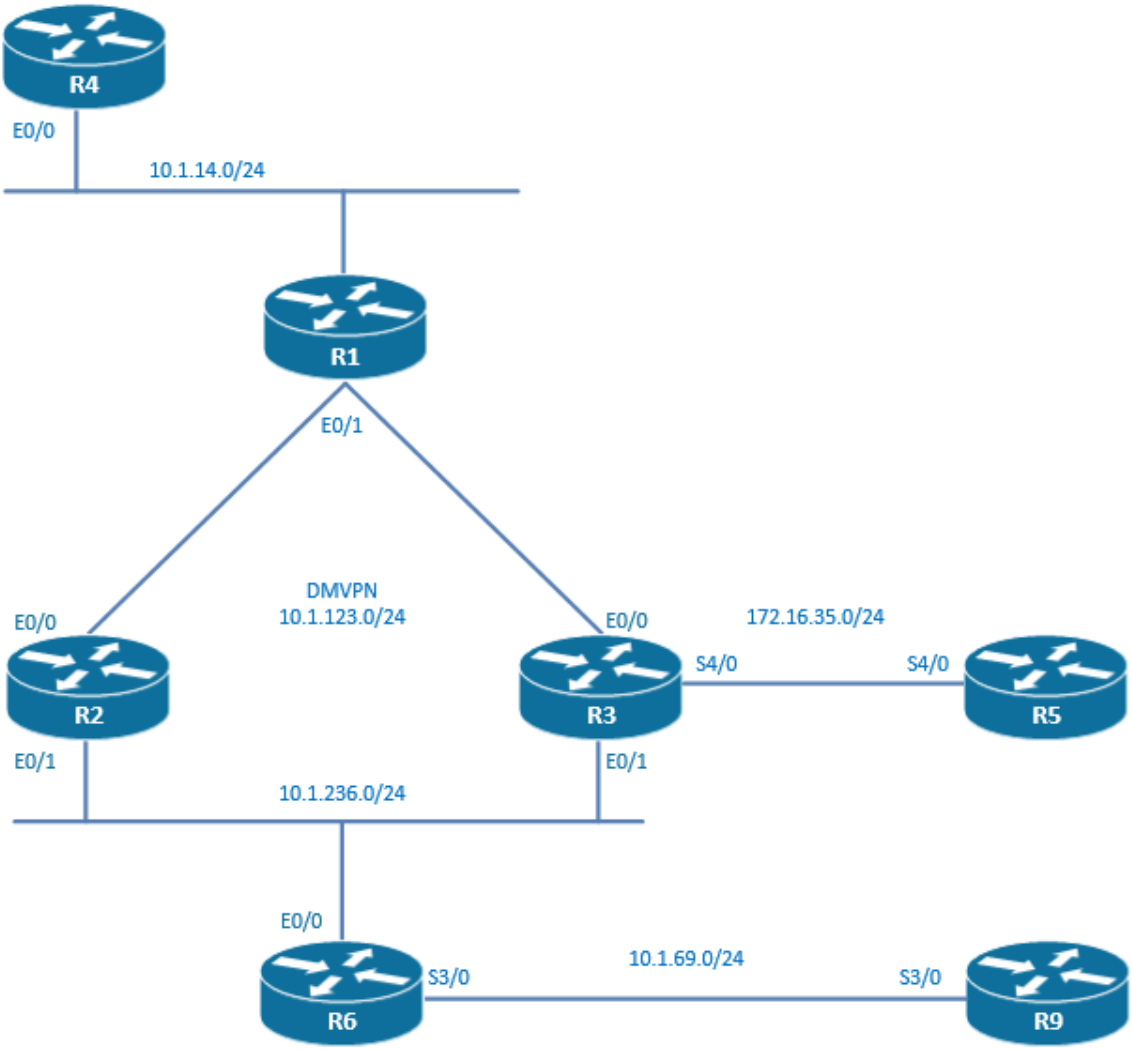
## Technologies covered

- RIP default route
- RIP update
- Unicast update
- Broadcast update
- Triggered update
- Source validation

## Overview

You have been tasked to configure routing in your network using the RIP version 2 protocol.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 10.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Configure RIP version 2 in this DMVPN network.

The DMVPN network phase 2 is already pre-configured. Let's configure RIP version 2 over this DMVPN network. RIPv2 is carrying a subnet mask field, but the classless behavior will only take place once the no auto-summary is configured.

On R1, R2, and R3, configure the following:

```
router rip
 version 2
 network 11.0.0.0
 no auto-summary
```

**Task 10.2** The RIP updates have to be sent as unicast packets on the DMVPN tunnels.

The neighbor command turns on unicast for RIP updates. However, a router will still send multicast packets on group 224.0.0.9. In order to turn off multicast RIP updates completely, the passive-interface command is required.

On R1, configure the following:

```
router rip
neighbor 11.1.1.2
neighbor 11.1.1.3
passive-interface Tunnel23
```

On R2, configure the following:

```
router rip
neighbor 11.1.1.1
neighbor 11.1.1.3
passive-interface Tunnel23
```

On R3, configure the following:

```
router rip
neighbor 11.1.1.1
neighbor 11.1.1.2
passive-interface Tunnel23
```

**Task 10.3** Advertise the loopbacks 0 of R1, R2, and R3 in the RIP process.

On R1, configure the following:

```
router rip
network 10.0.0.0
passive-interface e0/1
```

On R2 and R3, configure the following:

```
router rip
network 10.0.0.0
passive-interface e0/0
```

We are configuring passive-interface on the ethernet interfaces between R1, R2, and R3 because we would like RIP to run over the tunnel interfaces only.

At this point in the lab, I can ping from the loopback0 from R3 to the loopback0 of R1, but I cannot ping from the loopback0 from R3 to the loopback0 of R2.

```
R3#ping 10.1.1.1 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

R3#ping 10.2.2.2 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
.....
Success rate is 0 percent (0/5)
```

**Task 10.4** Ensure full reachability in this hub and spoke technology. On R2, check that you can ping the loopback of R3 sourcing from the loopback of R2.

We already have full reachability in this hub and speak topology.

```
R2#ping 10.1.3.3 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

The command “no ip split-horizon” is not necessary on the DMVPN tunnels because the RIP updates are unicast packets so they are going directly from spoke to spoke because we are in DMVPN phase 2.

**Task 10.5** Configure RIP version 2 between R1 and R4. Advertise the loopback of R4 into the RIP process.

On R4, configure the following:

```
router rip
 version 2
 network 10.0.0.0
 no auto-summary
```

I can ping from R4 to R2, which means RIP has taken care of the establishment of the IP connectivity.

```
R4#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 10.6** R1 should advertise a default route to all its RIP neighbors with the exception of R4.

On R1, configure the following:

```
route-map TU23 permit 10
set interface tu23
router rip
default-information originate route-map TU23
```

We can see that a default route has been advertised by RIP to R2.

```
R2#sh ip route
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 11.1.1.1 to network 0.0.0.0

R*    0.0.0.0/0 [120/13] via 11.1.1.1, 00:00:03, Tunnel23
      10.0.0.0/8 is variably subnetted, 11 subnets, 2 masks
R        10.1.1.1/32 [120/1] via 11.1.1.1, 00:00:09, Tunnel23
                     [120/1] via 10.1.123.1, 00:00:20, Ethernet0/0
C        10.1.2.2/32 is directly connected, Loopback0
R        10.1.3.3/32 [120/1] via 10.1.236.3, 00:00:13, Ethernet0/1
                     [120/1] via 10.1.123.3, 00:00:00, Ethernet0/0
R        10.1.4.4/32 [120/2] via 11.1.1.1, 00:00:09, Tunnel23
                     [120/2] via 10.1.123.1, 00:00:20, Ethernet0/0
R        10.1.14.0/24 [120/1] via 11.1.1.1, 00:00:09, Tunnel23
                      [120/1] via 10.1.123.1, 00:00:20, Ethernet0/0
C        10.1.25.0/24 is directly connected, Serial5/0
L        10.1.25.1/32 is directly connected, Serial5/0
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.2/32 is directly connected, Ethernet0/0
C        10.1.236.0/24 is directly connected, Ethernet0/1
L        10.1.236.2/32 is directly connected, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.2/32 is directly connected, Tunnel23
```

**Task 10.7** If the E0/0 interface is going down, R1 will stop advertising this default route.

On R1, configure the following:

```
ip access-list standard E_0_0
 permit 10.1.14.0 0.0.0.255

route-map CHECK_LINK permit 10
 match ip address E_0_0
 set interface tu23
```

Let's check that this conditional advertisement is working and shut down the e0/0 of R1.
On R2, I can observe that RIP is poisoning the default route with a metric of 16, which makes it unreachable.

```
RIP: build flash update entries
        0.0.0.0/0 via 0.0.0.0, metric 16, tag 0
```

**Task 10.8** Configure RIP version 2 on the LAN connecting R2, R3, and R6. Advertise the loopback of R6 into the RIP process.

On R6, configure the following:

```
router rip
 version 2
 network 10.0.0.0
 no auto-summary
```

I can ping from R6 to R4, which means RIP has taken care of the establishment of the IP connectivity.

```
R6#ping 10.1.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 10.9** The RIP updates should be broadcasted on the LAN 10.1.236.0/24.

On R2 and R3, configure the following

```
interface Ethernet0/1
  ip rip v2-broadcast
```

On R6, configure the following

```
interface Ethernet0/0
  ip rip v2-broadcast
```

**Task 10.10** Configure RIP version 2 on the serial connection between R3 and R5. Advertise the loopback 0 of R5 into the RIP process.

On R3, configure the following:

```
router rip
network 172.16.0.0
```

On R5, configure the following:

```
router rip
 version 2
 network 172.16.0.0
 network 10.0.0.0
 no auto-summary
```

I can ping from R4 to R5, which means RIP has taken care of the establishment of the IP connectivity.

```
R4#ping 10.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.5.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 10.11** The RIP updates between R3 and R5 should stay silent. Updates should be sent only when there is a change in the topology.

RIP updates will be exchanged only when there is a topology change once the RIP triggered command is configured on an interface.

On R3, configure the following:

```
interface Serial4/0
  ip rip triggered
```

On R5, configure the following:

```
interface Serial4/0
  ip rip triggered

R5#sh ip protocols
```

```
*** IP Routing is NSF aware ***

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 20 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    Ethernet0/0           2     2
    Serial4/0             2     2          Yes
    Serial5/0             2     2
    Loopback0             2     2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    10.0.0.0
    172.16.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.1.25.1            120      00:00:03
    172.16.35.3          120      00:00:01
  Distance: (default is 120)
```

On R3, as shown in the output below, the networks advertised from R5 have not been refreshed since 9 minutes and 18 seconds but are still in the routing table. By default, an entry is refreshed every 30 seconds and is invalid after 3 minutes.

```
R3#sh ip route
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, 18 subnets, 2 masks
R        10.1.1.1/32 [120/2] via 10.1.236.2, 00:00:05, Ethernet0/1
R        10.1.2.2/32 [120/1] via 10.1.236.2, 00:00:05, Ethernet0/1
C        10.1.3.3/32 is directly connected, Loopback0
R        10.1.4.4/32 [120/3] via 10.1.236.2, 00:00:05, Ethernet0/1
R        10.1.5.5/32 [120/1] via 172.16.35.5, 00:09:18, Serial4/0
R        10.1.6.0/24 [120/1] via 10.1.236.6, 00:00:12, Ethernet0/1
R        10.1.14.0/24 [120/2] via 10.1.236.2, 00:00:05, Ethernet0/1
R        10.1.25.0/24 [120/1] via 172.16.35.5, 00:09:18, Serial4/0
                      [120/1] via 10.1.236.2, 00:00:05, Ethernet0/1
R        10.1.35.0/24 [120/1] via 172.16.35.5, 00:09:18, Serial4/0
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
C        10.1.236.0/24 is directly connected, Ethernet0/1
L        10.1.236.3/32 is directly connected, Ethernet0/1
R        10.11.6.0/24 [120/1] via 10.1.236.6, 00:00:12, Ethernet0/1
R        10.22.6.0/24 [120/1] via 10.1.236.6, 00:00:12, Ethernet0/1
R        10.33.6.0/24 [120/1] via 10.1.236.6, 00:00:12, Ethernet0/1
R        10.44.6.0/24 [120/1] via 10.1.236.6, 00:00:12, Ethernet0/1
R        10.55.6.0/24 [120/1] via 10.1.236.6, 00:00:12, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.2/32 is directly connected, Tunnel23
      172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C        172.16.35.0/24 is directly connected, Serial4/0
L        172.16.35.3/32 is directly connected, Serial4/0
```

**Task 10.12** Configure RIP version 2 on the serial connection between R6 and R9. Advertise the loopback of R9 into the RIP process.

On R9, configure the following:

```
router rip
 version 2
 network 10.0.0.0
 no auto-summary
```

I can ping from R1 to R9, which means RIP has taken care of the establishment of the IP connectivity.

```
R1#ping 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms
```

**Task 10.13** Configure PPP encapsulation on the serial connection between R6 and R9. Use IPCP for address allocation with PPP. R6 is the server side (IP address 10.1.69.6/24) and R9 is client side (IP address 10.1.69.9/32 assigned by server). Ensure that R6 is getting the RIP updates from R9 and that you can ping the loopback of R9 sourcing from the loopback of R6.

Let's configure the PPP encapsulation.

On R6 and R9, configure the following:

```
int s3/0
encapsulation ppp
```

Let's configure IPCP.

On R6, on the server side, configure the following:

```
interface Serial4/0
  peer default ip address 10.1.69.9 < not supported on the current iPexpert POD
```

On R9, on the server side, configure the following:

```
interface Serial3/0
  ip address negotiated
```

In the routing table of R9, two host routes for the host 10.1.69.9 and 10.1.69.6 will appear in the routing table but not the network 10.1.69.0/24. RIP updates will be ignored because the two ends of the connection don't appear to be on the same network. This can be fixed by disabling the validate-update-source check.

On R9, configure the following:

```
router rip
no validate-update-source
```

**Task 10.14** R5 should advertise a default-route to R3. This default-route should only be advertised if the network 10.1.2.2/32 is present in the routing table.

We are going to track the network 10.1.2.0/24 using IP SLA.

```
ip sla monitor 1
 type echo protocol ipIcmpEcho 10.1.2.2

ip sla monitor schedule 1 life forever start-time now
```

In order to create a bond between the route tracked in the routing table and the route-map used for conditional advertisement of the default route, we have to create a fake route that is tracked by the IP SLA and that will be used in the route-map.

```
track 10 rtr 1
ip route 2.2.2.2 255.255.255.255 Null0 track 10

ip access-list standard FAKE
  permit 2.2.2.2

route-map DEFAULT_ROUTE permit 10
  match ip address FAKE
  set interface Serial4/0

router rip
  default-information originate route-map DEFAULT_ROUTE
```

**You have completed Lab 10**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 11: Configure and troubleshoot EIGRP (Part 1)

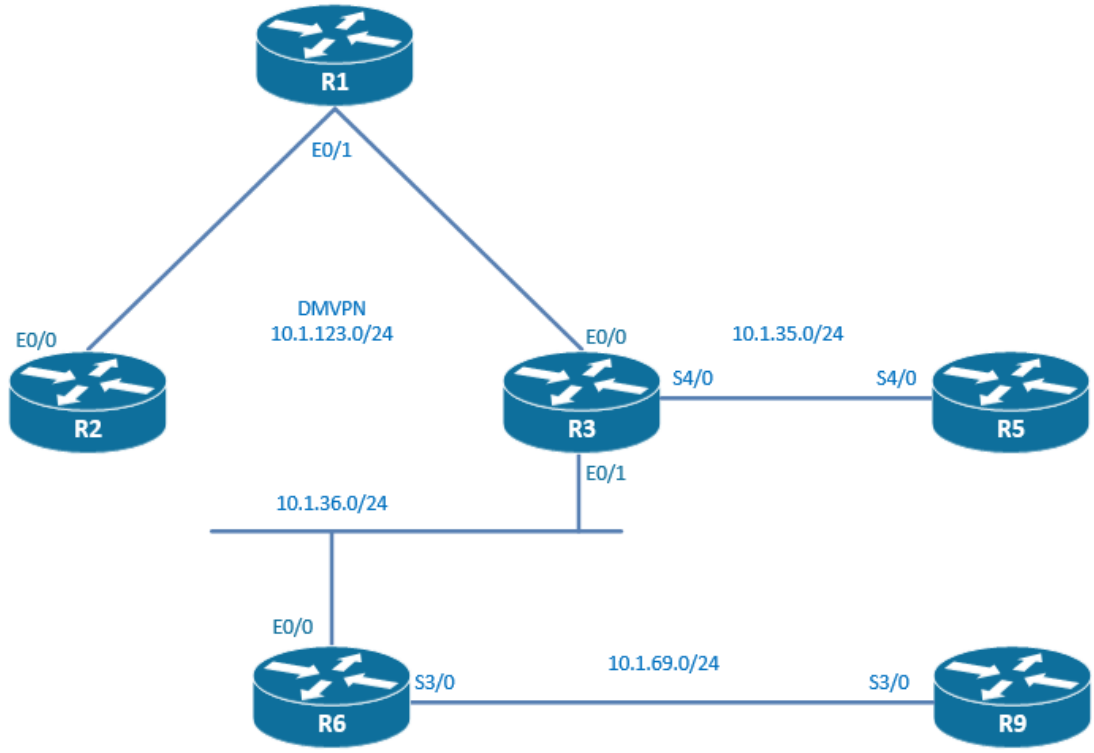
## Technologies covered

- EIGRP AS mode
- EIGRP named mode
- Stub
- Summarization
- Authentication
- Key chain rotation
- Prefix number limiting

## Overview

You have been tasked to configure the routing reachability in your network using the EIGRP protocol.

The topology used in the lab will be the following:




**Estimated time to complete: 2-3 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 11.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Setup EIGRP routing in autonomous configuration mode with AS11 in this DMVPN network.

Let's configure EIGRP on the mGRE tunnel network. On R1, R2, and R3, configure the following:

```
router eigrp 11
 network 11.1.1.0 0.0.0.255
```

We can check that the EIGRP neighbor ships between R1 and R2 and between R1 and R3 are up and running:

```
R1#sh ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(11)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   11.1.1.3                Tu23                     13 00:00:46    4  1470  0  1
0   11.1.1.2                Tu23                     12 00:00:51    1  1470  0  1
```

This is kind of expected as the DMVPN tunnels are configured for multicast support and EIGRP hellos and updates are sent on the multicast group address 224.0.0.10.

**Task 11.2** Advertise the loopbacks of R2 and R3 in the EIGRP process. Only the 12.1.x.x/24 networks should be redistributed from connected into the routing protocol.

To inject into EIGRP the loopbacks of R2 and R3 except the 12.1.x.x/24 networks, we are going to use network statements.

On R2, configure the following:

```
router eigrp 11
network 10.1.2.0 0.0.0.255
passive-interface loopback0
```

On R3, configure the following:

```
router eigrp 11
network 10.1.3.0 0.0.0.255
network 3.3.3.3 0.0.0.0
passive-interface loopback0
passive-interface loopback10
```

It is best practice to configure the loopbacks as EIGRP passive interfaces because no EIGRP hellos have to be sent on the loopbacks.

To inject into EIGRP the 12.1.x.x/24 networks, we are going to use the redistribute connected statement. We have to use a route-map to make sure that only the 12.1.x.x/24 networks are redistributed.

On R2, we have to configure the following:

```
route-map CONNECTED permit 10
 match interface Loopback1 Loopback2 Loopback3 Loopback4
!
router eigrp 11
 redistribute connected route-map CONNECTED
```

### Task 11.3 Redistribute only the loopback0 of R1 in the EIGRP process.

We are going to redistribute the loopback0 into EIGRP by using a redistribute connected statement. We are asked to redistribute only the loopback0 and not all connected networks of R1. Therefore, we have to filter the connected networks that we are injecting into the routing protocol by using a route-map.

On R1, configure the following:

```
route-map CONNECTED permit 10
 match interface Loopback0
!
router eigrp 11
 redistribute connected route-map CONNECTED
```

### Task 11.4 Make sure that there is full connectivity between loopbacks with the DMVPN network.

From the hub R1, I can ping the spokes R2 and R3:

```
R1#ping 10.1.2.2 source 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R1#ping 10.1.3.3 source 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/5 ms
```

But there is no IP connectivity from spoke to spoke:

```
R2#ping 10.1.3.3 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
.....
Success rate is 0 percent (0/5)
```

We can see in the routing table of R2 that there is no route to 10.1.3.0/24 network.

```
R2#sh ip route
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, 5 subnets, 2 masks
D EX     10.1.1.0/24 [170/27008000] via 11.1.1.1, 00:17:12, Tunnel23
C        10.1.2.0/24 is directly connected, Loopback0
L        10.1.2.2/32 is directly connected, Loopback0
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.2/32 is directly connected, Ethernet0/0
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.2/32 is directly connected, Tunnel23
```

This is due to the EIGRP loop prevention mechanism called split-horizon. This mechanism is preventing an update to be sent out of the interface where this update was received. In a mGRE topology, this mechanism is preventing the spokes to receive the routes from the other spokes. We therefore have to disable it on R1 tunnel interface.

On R1, configure the following:

```
int tu23
no ip split-horizon eigrp 11
```

Let's have now a look at the routing table of R2:

```
R2#sh ip route
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

      3.0.0.0/32 is subnetted, 1 subnets
D        3.3.3.3 [90/28288000] via 11.1.1.1, 00:01:15, Tunnel23
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
D EX     10.1.1.0/24 [170/27008000] via 11.1.1.1, 00:26:33, Tunnel23
C        10.1.2.0/24 is directly connected, Loopback0
L        10.1.2.2/32 is directly connected, Loopback0
D        10.1.3.0/24 [90/28288000] via 11.1.1.1, 00:01:15, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.2/32 is directly connected, Ethernet0/0
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.2/32 is directly connected, Tunnel23
```

This is looking a lot better. There is a route now to the 10.1.3.0/24 network. Let's check also that the ping from spoke to spoke is working.

```
R2#ping 10.1.3.3 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 11.5** Make sure that the traffic from the spoke to spoke is not transiting by the hub.

We can see that the trace route from R2 to R3 is transiting through the hub R1.

```
R2#traceroute 10.1.3.3  source 10.1.2.2
Type escape sequence to abort.
Tracing the route to 10.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 11.1.1.1 1 msec 1 msec 0 msec
  2 11.1.1.3 1 msec *  2 msec
```

However, the underlying DMVPN network is configured in phase 2 that means that the traffic from spoke to spoke is going directly from spoke to spoke. In EIGRP the next hop advertised is the router itself, but in DMVPN you want to make sure the spokes know about each other. In order to allow this to happen, you need to tell EIGRP not to change in updates the next-hop to itself when transiting

through the hub. This is accomplished by using the command no IP next-hop-self on the tunnel interface on the hub.

On R1, configure the following:

```
interface tu23
no ip next-hop-self eigrp 11
```

Let's check now the output of the trace route:

```
R2#traceroute 10.1.3.3  source 10.1.2.2
Type escape sequence to abort.
Tracing the route to 10.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 11.1.1.3 0 msec *  1 msec
```

It is now working as desired. The traffic from spoke to spoke is bypassing the hub.

**Task 11.6** R2 should advertise the 12.1.0.0/16 network out to R1 with a metric using the following parameters:

| | |
|---|---|
| bandwidth | 100 000 kilobits per s |
| delay | 5 tens of microsecond |
| reliability | 255 |
| load | 20 |
| mtu | 1500 bytes |

Let's check the current routing table of R1:

```
R1#sh ip route
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

      3.0.0.0/32 is subnetted, 1 subnets
D        3.3.3.3 [90/27008000] via 11.1.1.3, 00:11:45, Tunnel23
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
C        10.1.1.0/24 is directly connected, Loopback0
L        10.1.1.1/32 is directly connected, Loopback0
D        10.1.2.0/24 [90/27008000] via 11.1.1.2, 00:11:45, Tunnel23
D        10.1.3.0/24 [90/27008000] via 11.1.1.3, 00:11:45, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
      12.0.0.0/24 is subnetted, 4 subnets
D EX     12.1.1.0 [170/27008000] via 11.1.1.2, 00:00:08, Tunnel23
D EX     12.1.2.0 [170/27008000] via 11.1.1.2, 00:00:08, Tunnel23
D EX     12.1.3.0 [170/27008000] via 11.1.1.2, 00:00:08, Tunnel23
D EX     12.1.4.0 [170/27008000] via 11.1.1.2, 00:00:08, Tunnel23
```

There are four 12.1.x.x/24 entries in the routing table of R1. R1 should only receive a single 12.1.0.0/16 entry representing all the 12.1.x.x/24 entries. This is going to be achieved using manual summarization.

On R2, let's configure the following:

```
int tunnel23
ip summary-address eigrp 11 12.1.0.0 255.255.0.0
```

On R1, let's observe the effect of this summarization in the routing table:

```
R1#sh ip route
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

      3.0.0.0/32 is subnetted, 1 subnets
D        3.3.3.3 [90/27008000] via 11.1.1.3, 00:19:13, Tunnel23
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
C        10.1.1.0/24 is directly connected, Loopback0
L        10.1.1.1/32 is directly connected, Loopback0
D        10.1.2.0/24 [90/27008000] via 11.1.1.2, 00:01:17, Tunnel23
D        10.1.3.0/24 [90/27008000] via 11.1.1.3, 00:19:13, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
      12.0.0.0/16 is subnetted, 1 subnets
D        12.1.0.0 [90/27008000] via 11.1.1.2, 00:01:17, Tunnel23
```

The 12.1.x.x/24 has all be suppressed and replaced by the 12.1.0.0/16 summary.
In the question, we are also asked to specify the metric of the summary route. We have to remember that the metric used by EIGRP manual summary route is the minimum metric of the specific routes. The specific routes have been redistributed into the routing protocol. When redistribution is taking place, the metric of the redistributed routes can be specified.

On R2, configure the following using the K parameters specified in the question.

```
router eigrp 11
 redistribute connected route-map CONNECTED metric 100000 5 255 20 1500
```

**Task 11.7** R2 is not transiting any traffic, so R2 should not receive any EIGRP query packets anymore. Configuration for this task should be performed on R2, and loopbacks of R2 should stay reachable.

In order not to receive any EIGRP query packets, you can configure an EIGRP router as a stub. This is used to save resources in a hub and spoke topology. The hub doesn't have to receive the EIGRP queries because the only networks that it has are the ones that are advertised in the stub commands. In our case, we have to configure that the only routes that have to be reached on the stub R2 are the loopbacks. In a previous question, we manually summarized some of the loopbacks. This summary should also be part of the only network that the stub is advertising.

On R2, configure the following:

```
router eigrp 1
eigrp stub connected summary
```

Let's check on R3 that the loopbacks of R2 are still present in the routing table.

```
R3#sh ip route
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

      3.0.0.0/32 is subnetted, 1 subnets
C        3.3.3.3 is directly connected, Loopback10
      10.0.0.0/8 is variably subnetted, 10 subnets, 2 masks
D EX     10.1.1.0/24 [170/27008000] via 11.1.1.1, 1d03h, Tunnel23
D        10.1.2.0/24 [90/28288000] via 11.1.1.2, 00:15:24, Tunnel23
C        10.1.3.0/24 is directly connected, Loopback0
L        10.1.3.3/32 is directly connected, Loopback0
C        10.1.35.0/24 is directly connected, Serial4/0
L        10.1.35.3/32 is directly connected, Serial4/0
C        10.1.36.0/24 is directly connected, Ethernet0/1
L        10.1.36.3/32 is directly connected, Ethernet0/1
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.3/32 is directly connected, Tunnel23
      12.0.0.0/16 is subnetted, 1 subnets
D        12.1.0.0 [90/28161280] via 11.1.1.2, 00:15:24, Tunnel23
```

**Task 11.8** On R6 and R9, setup EIGRP routing in named configuration mode using AS11 and the name of "iPexpert". Advertise the loopbacks of R6 and R9 in the EIGRP process. On R9, ensure that you can ping the loopback1 of R2 from the loopback0 of R9.

Let's configure EIGRP in named mode configuration mode. This is just another way to configure the same EIGRP protocol. EIGRP in AS mode and EIGRP in named mode are therefore inter-compatible. R3 will be running EIGRP in AS mode and R6 will be running EIGRP in named and a working adjacency will be formed.

On R3, configure the following:

```
router eigrp 11
network 10.1.36.0 0.0.0.255
```

On R6, configure the following:

```
router eigrp iPexpert
address-family ipv4 unicast autonomous-system 11
  network 10.1.6.0 0.0.0.255
  network 10.1.36.0 0.0.0.255
  network 10.1.69.0 0.0.0.255
  network 10.11.6.0 0.0.0.255
  network 10.22.6.0 0.0.0.255
  network 10.33.6.0 0.0.0.255
  network 10.44.6.0 0.0.0.255
  network 10.55.6.0 0.0.0.255
 exit-address-family
```

On R9, configure the following:

```
router eigrp iPexpert
```

```
 address-family ipv4 unicast autonomous-system 11
  network 10.1.9.0 0.0.0.255
  network 10.1.69.0 0.0.0.255
 exit-address-family
```

Let's check if the end-to-end IP connectivity is there.

```
R9#ping 12.1.1.2 source 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 12.1.1.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.9.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/10/11 ms
```

I can ping the loopback1 of R2 from R9. All is working fine!

**Task 11.9** Configure the only possible EIGRP authentication mode between R6 and R3. Use a key chain called "keyiPexpert1" with 2 keys. Key 1 with a key-string of "Password1" is used since 03:00:00 Jan 1 2014 until 03:00:00 Jan 1 2015, but can already be used one month before and is still valid one month after. Key 2 with a key-string of "Password2" will be used from 03:00:00 Jan 1 2015 onwards, but can be used since 03:00:00 Dec 15 2014.

Let's begin to configure the keychain management. We can specify which key is going to be used at which period of time by using the send-lifetime parameter. When the key is changing from key 1 to key2, we can configure an overlap period when key1 and key2 will both be valid by using the accept-lifetime parameter.

On R3 and R6, let's configure the following key-chain.

```
key chain keyiPexpert1
  key 1
    key-string Password1
    accept-lifetime 03:00:00 Dec 01 2013 03:00:00 Feb 01 2015
    send-lifetime 03:00:00 Jan 01 2014 03:00:00 Jan 01 2015
  key 2
    key-string Password2
    accept-lifetime 03:00:00 Dec 15 2015 infinite
    send-lifetime 03:00:00 Jan 01 2015 infinite
```

We can check that the key is correctly working:

```
R3#sh key chain
Key-chain iPexpertchain:
    key 1 -- text "iPpassword"
        accept lifetime (always valid) - (always valid) [valid now]
        send lifetime (always valid) - (always valid) [valid now]
Key-chain keyiPexpert1:
    key 1 -- text "Password1"
        accept lifetime (03:00:00 UTC Dec 1 2013) - (03:00:00 UTC Feb 1 2015) [valid now]
        send lifetime (03:00:00 UTC Jan 1 2014) - (03:00:00 UTC Jan 1 2015) [valid now]
    key 2 -- text "Password2"
        accept lifetime (03:00:00 UTC Dec 15 2015) - (infinite)
        send lifetime (03:00:00 UTC Jan 1 2015) - (infinite)
```

We are told to configure only possible EIGRP authentication mode between R6 and R3. EIGRP is running in AS mode on R3. The 2 possible authentication modes in AS mode are clear text and MD5. EIGRP is running in named mode on R3. The 2 possible authentication modes in named mode are MD5 and SHA-256. So the only inter-operability authentication mode between R3 and R6 is MD5.

Let's configure EIGRP authentication on R6:

```
router eigrp iPexpert
address-family ipv4 unicast autonomous-system 11
  af-interface e0/0
   authentication mode md5
   authentication key-chain keyiPexpert1
  exit-af-interface
```

Let's configure EIGRP authentication on R3:

```
interface E0/1
 ip authentication mode eigrp 11 md5
 ip authentication key-chain eigrp 11 keyiPexpert1
```

**Task 11.10** Configure EIGRP HMAC-SHA-256 authentication between R6 and R9. Use a key-string of "Password3".

The difference between MD5 and SHA-256 authentication is that MD5 authentication is configured using a key chain whereas SHA-256 is configured using a key that is configured inline.

Let's configure EIGRP SHA-256 authentication between R6 and R9.

On R6 and R9, the following configuration has to be applied:

```
router eigrp iPexpert
address-family ipv4 unicast autonomous-system 11
  af-interface s3/0
   authentication mode hmac-sha-256 Password3
  exit-af-interface
```

**Task 11.11** On R6, generate a syslog message when the maximum prefix limit of 10 has been accepted from the neighbor R9. Do not take any other action when this max limit of 10 is exceeded.

We have to limit the number of prefixes that R9 is able to advertise to R6. We don't want the adjacency to be torn down when the max-limit is exceeded, only a syslog message has to be sent.

```
On R6, configure the following:
router eigrp iPexpert
address-family ipv4 unicast autonomous-system 11
neighbor 10.1.69.9 maximum-prefix 10 warning-only
```

**Task 11.12** On R6, tear down the EIGRP neighborship relations when more than 20 prefixes are received by the EIGRP process, and generate a syslog message when more than 10 prefixes have been accepted.

Contrary to the previous question, we are now asked to place a max prefix-limit on the number of prefixes that the EIGRP process can handle, which is the sum of all the prefixes received by all the EIGRP neighbors.

On R6, configure the following:

```
router eigrp iPexpert
address-family ipv4 unicast autonomous-system 11
maximum-prefix 20 10
```

As soon as I configured it, I got the following syslog that popped up:

```
%DUAL-4-PFXLIMITTHR: EIGRP-IPv4 11: Neighbor threshold prefix level(1) reached.
```

This is due to the fact that the EIGRP process has already more than 10 EIGRP prefixes received from its neighbors. However, the peering's are not torn down as the 20 prefixes threshold is not yet reached.

**You have completed Lab 11**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 12: Configure and troubleshoot EIGRP (Part 2)

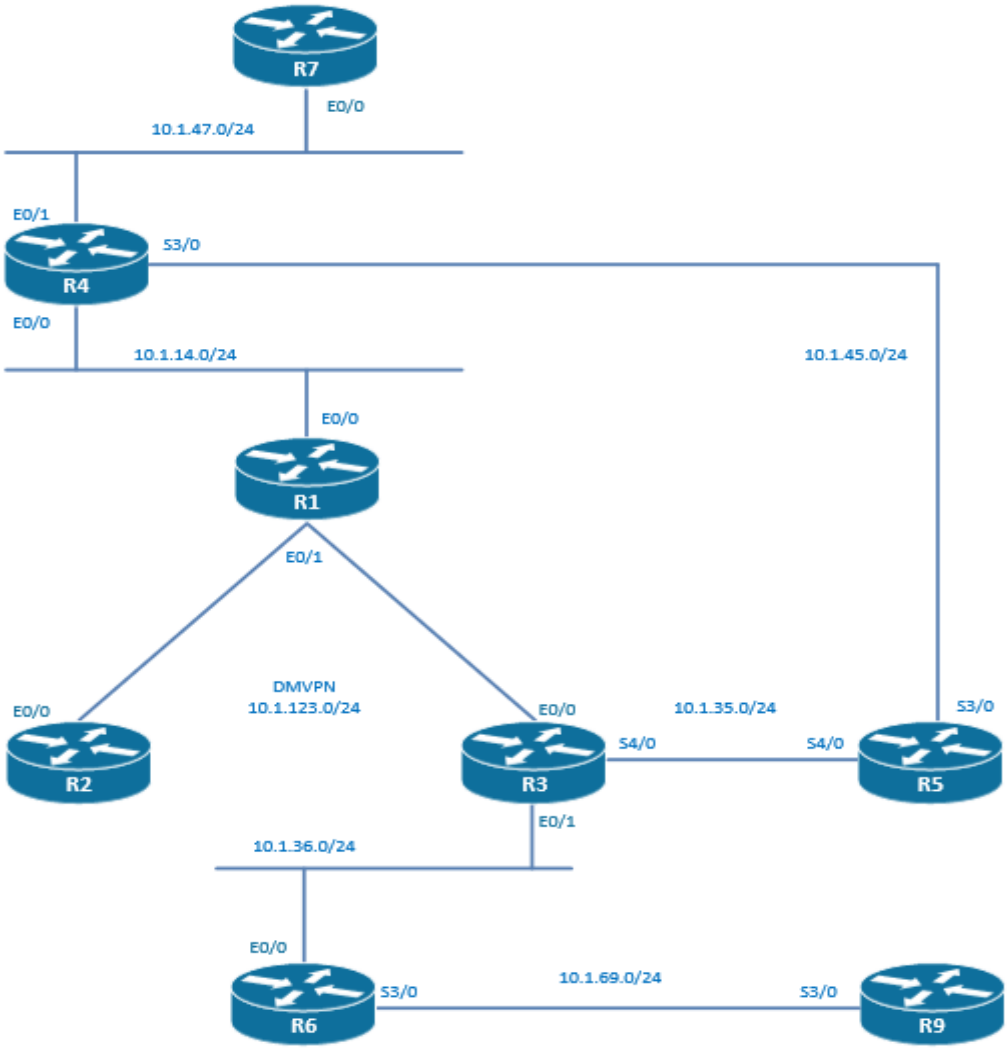
## Technologies covered

- Summarization with default routing
- Summarization with leak-map
- Summarization with floating default routing
- EIGRP metric weights
- TE
- Unequal cost load balancing
- EIGRP timers

## Overview

You have been tasked to configure the routing reachability in your network using the EIGRP protocol.

The topology used in the lab will be the following:




**Estimated time to complete: 2-3 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 12.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Setup EIGRP routing in autonomous configuration mode with AS4 in this DMVPN network.

On R1, R2, and R3, configure the following:

```
router eigrp 4
 network 11.1.1.0 0.0.0.255
```

**Task 12.2** Advertise the loopback0 of R1, R2, and R3 in the EIGRP process using network statements.

On R1, configure the following:

```
router eigrp 4
 network 10.1.1.1 0.0.0.0
```

On R2, configure the following:

```
router eigrp 4
 network 10.1.2.2 0.0.0.0
```

On R3, configure the following:

```
router eigrp 4
 network 10.1.3.3 0.0.0.0
```

I can ping from the spoke R3 to the hub R1, but I cannot ping from the spoke R3 to the spoke R2.

```
R3#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R3#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

We have to disable slip-horizon on the hub because EIGRP is not advertising back a network out of the interface where this network was learned. As we are running DMVPN phase 2, we have to also make sure that traffic from spoke to spoke will be forwarded directly through a dynamic tunnel. In order to have this working with EIGRP, we have to disable the next-hop-self behavior of EIGRP because we want to decouple the path that the updates are taking with the path that the data plane is taking.

On R1, configure the following:

```
int tu23
no ip split-horizon eigrp 4
no ip next-hop-self eigrp 4
```

The ping from spoke to spoke is now working.

```
R3#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 12.3** Setup EIGRP routing between R3 and R5. Advertise the loopback0 into the EIGRP process.

On R3, configure the following:

```
router eigrp 4
 network 10.1.35.0 0.0.0.255
```

On R5, configure the following:

```
router eigrp 4
 network 10.1.35.0 0.0.0.255
 network 10.1.5.5 0.0.0.0
```

On R5, I can ping the loopback0 of R1.

```
R5#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/10 ms
```

**Task 12.4** On R3, configure summarization in a way that R5 only receives a default-route from R3. Leak also the loopback 10.1.4.4.

On R3, configure the following:

```
access-list 1 permit 10.1.4.4 0.0.0.0

route-map DEFAULT_LEAK permit 10
  match ip address 1

 interface s4/0
ip summary-address eigrp 4 0.0.0.0 0.0.0.0 leak-map DEFAULT_LEAK

On R5, there is only a default route that has been received from R3. I can still ping R1
from R5.
R5#sh ip route
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 10.1.35.3 to network 0.0.0.0

D*    0.0.0.0/0 [90/2297856] via 10.1.35.3, 00:00:10, Serial4/0
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
C        10.1.5.5/32 is directly connected, Loopback0
C        10.1.25.0/24 is directly connected, Serial5/0
L        10.1.25.2/32 is directly connected, Serial5/0
C        10.1.35.0/24 is directly connected, Serial4/0
L        10.1.35.5/32 is directly connected, Serial4/0

R5#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/8/9 ms
```

**Task 12.5** Setup EIGRP routing between R3 and R6, and between the R6 and R9. Advertise the loopback0 of R6 and R9 into the EIGRP process. On R3, check that you can ping the loopback of R9 using the loopback of R3 as a source.

On R3, configure the following:

```
router eigrp 4
 network 10.1.36.0 0.0.0.255
```

On R6, configure the following:

```
router eigrp 4
 network 10.1.36.0 0.0.0.255
 network 10.1.69.0 0.0.0.255
 network 10.1.6.6 0.0.0.0
```

On R9, configure the following:

```
router eigrp 4
 network 10.1.69.0 0.0.0.255
 network 10.1.9.9 0.0.0.0
```

At this point, I can ping from R9 to R1:

```
R9#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms
```

**Task 12.6** On R3, configure summarization in a way that R6 only receives a default-route from R3.

On R3, configure the following:

```
interface e0/1
ip summary-address eigrp 4 0.0.0.0 0.0.0.0
```

At this point, I can ping from R9 to R1:

```
R9#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms
```

**Task 12.7** On R6, configure summarization in a way that R9 only receives a default-route from R6.

On R6, configure the following:

```
interface Serial3/0
  ip summary-address eigrp 4 0.0.0.0 0.0.0.0
```

At this point, I cannot ping anymore from R9 to R1:

```
R9#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
```

**Task 12.8** On R3, check that you can ping the loopback of R9 using the loopback of R3 as a source. Use a floating route summarization.

I cannot ping the loopback of R9 using the loopback of R3 as a source.

```
R3#ping 10.1.9.9 source 10.1.3.3
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
.....
Success rate is 0 percent (0/5)
```

This is due to the fact that we created a routing black hole on R6.

```
R6# sh ip route
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

D*    0.0.0.0/0 is a summary, 00:08:46, Null0
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
C        10.1.6.6/32 is directly connected, Loopback0
D        10.1.9.9/32 [90/2297856] via 10.1.69.9, 00:18:06, Serial3/0
C        10.1.36.0/24 is directly connected, Ethernet0/0
L        10.1.36.6/32 is directly connected, Ethernet0/0
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.11.6.6/32 is directly connected, Loopback1
```

This is due to the fact that a default route to null0 with an AD of 5 is created on the router where a summary-address 0.0.0.0 0.0.0.0 is originated. The default-route advertised by R3 has an AD of 90, so the default-route with an AD of 5 is preferred and in use in the routing-table. In order to fix this routing problem, we have to give to the summary route generated on R6 an AD superior to 90.

On R6, configure the following:

```
router eigrp 4
summary-metric 0.0.0.0 0.0.0.0 distance 250
```

The default route used in the routing table of R6 is not anymore the route to null0.

```
R6#sh ip route
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 10.1.36.3 to network 0.0.0.0

D*    0.0.0.0/0 [90/409600] via 10.1.36.3, 00:00:06, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
C        10.1.6.6/32 is directly connected, Loopback0
D        10.1.9.9/32 [90/2297856] via 10.1.69.9, 00:29:03, Serial3/0
C        10.1.36.0/24 is directly connected, Ethernet0/0
L        10.1.36.6/32 is directly connected, Ethernet0/0
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.11.6.6/32 is directly connected, Loopback1
```

The IP connectivity is again established.

```
R3#ping 10.1.9.9 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/8/9 ms
```

**Task 12.9** Setup EIGRP routing between R1 and R4, and between the R4 and R7. Advertise the loopback0 of R4 and R7 into the EIGRP process. On R1, check that you can ping the loopback of R7 using the loopback of R1 as a source.

On R1, configure the following:

```
router eigrp 4
 network 10.1.14.0 0.0.0.255
```

On R4, configure the following:

```
router eigrp 4
 network 10.1.14.0 0.0.0.255
 network 10.1.47.0 0.0.0.255
 network 10.1.4.4 0.0.0.0
```

On R7, configure the following:

```
router eigrp 4
 network 10.1.47.0 0.0.0.255
 network 10.1.7.7 0.0.0.0
```

At this point, I can ping from R7 to R9:

```
R7#ping 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/10/13 ms
```

**Task 12.10** On R4, configure summarization in a way that R7 receives from R4 a default-route and the loopback0 networks of R1, R2, and R3.

On R4, configure the following:

```
access-list 1 permit 10.1.1.1 0.0.0.0
access-list 1 permit 10.1.2.2 0.0.0.0
access-list 1 permit 10.1.3.3 0.0.0.0

route-map DEFAULT_LEAK permit 10
  match ip address 1

 interface e0/1
ip summary-address eigrp 4 0.0.0.0 0.0.0.0 leak-map DEFAULT_LEAK
```

Let's check the routing table of R7. In the routing table, we can see that the default route and the 3 leaked networks are advertised by EIGRP.

```
R7#sh ip route
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 10.1.47.4 to network 0.0.0.0

D*    0.0.0.0/0 [90/409600] via 10.1.47.4, 00:03:41, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
D        10.1.1.1/32 [90/435200] via 10.1.47.4, 00:20:30, Ethernet0/0
C        10.1.1.7/32 is directly connected, Loopback0
D        10.1.2.2/32 [90/27059200] via 10.1.47.4, 00:20:30, Ethernet0/0
D        10.1.3.3/32 [90/27059200] via 10.1.47.4, 00:20:30, Ethernet0/0
C        10.1.47.0/24 is directly connected, Ethernet0/0
L        10.1.47.7/32 is directly connected, Ethernet0/0
```

**Task 12.11** Setup EIGRP routing between R4 and R5.

On R4 and R5, configure the following:

```
router eigrp 4
 network 10.1.45.0 0.0.0.255
```

**Task 12.12** In the whole EIGRP domain, configure the metric calculation to use K1=0, K2=0, K3=1, K4=0, and K5=0.

The formula to calculate EIGRP metric is the following:
EIGRP Metric = 256*(((k1*Bandwidth) + (k2*Bandwidth)/(256-Load) + k3*Delay))*(k5/(Reliability + k4)))

The default K values are k1=1,k2=0,k3=1,k4=0,k5=0. When k5 is equal to 0 then [k5/( k4 + reliability)] is defined to be 1. That means that the default formula is the following:
EIGRP Metric = 256*(Bandwidth + Delay)

Weird enough, for the final metric computation, EIGRP is not using the delay or bandwidth as they are observed in the output of the router, but use the interface values by inverting the bandwidth and scaling the delay with following calculations:

Bandwidth = $10^7$/ actual interface bandwidth
Delay = actual interface delay /10

The default formula is therefore the following:
EIGRP Metric = 256*(($10^7$/actual interface bandwidth) + (actual interface delay /10))

The following k values should be configured, k1=0, k2=0, k3=1, k4=0, and k5=0.

On R1, R2, R3, R4, R5, R6, R7, and R9, configure the following:

```
router eigrp 4
 metric weight 0 0 0 1 0 0
```

When using those values, the EIGRP metric calculation will be well simplified:

EIGRP Metric = 256*(actual interface delay /10)

```
R4#          sh ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "eigrp 4"
  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(4)
```

```
    Metric weight K1=0, K2=0, K3=1, K4=0, K5=0
    NSF-aware route hold timer is 240
    Router-ID: 10.1.4.4
    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
  Address Summarization:
    0.0.0.0/0 for Et0/1
      Summarizing 14 components with metric 2560
  Maximum path: 4
  Routing for Networks:
    10.1.4.4/32
    10.1.14.0/24
    10.1.45.0/24
    10.1.47.0/24
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.1.14.1             90      00:05:08
    10.1.45.5             90      00:05:03
    10.1.47.7             90      00:36:15
  Distance: internal 90 external 170
```

**Task 12.13** Configure a delay of 512 on the link between R4 and R5, a delay of 256 on the link between R4 and R1, a delay of 256 on the link between R1 and R3, and a delay of 128 on the link between R3 and R5.

According to the formula explained in the earlier question, in order to get a delay of 256, I have to configure an actual interface delay of 10. In order to get a delay of 512, I have to configure an actual interface delay of 20.

On R4 and R5, configure the following:

```
int s3/0
delay 20
```

On R4 and R1, configure the following:

```
int e0/0
delay 10
```

On R1 and R3, configure the following:

```
int tu23
delay 10
```

On R3 and R5, configure the following:

```
int s4/0
delay 5
```

**Task 12.14** Configure bidirectional un-equal cost load-balancing between R4 and R5. Use off-set list when it is necessary.

Let's have a look at the network 10.1.5.5/32 in the EIGRP database of R4.

```
R4#show ip eigrp topology 10.1.5.5 255.255.255.255
EIGRP-IPv4 Topology Entry for AS(4)/ID(10.1.4.4) for 10.1.5.5/32
  State is Passive, Query origin flag is 1, 2 Successor(s), FD is 133120
  Descriptor Blocks:
  10.1.45.5 (Serial3/0), from 10.1.45.5, Send flag is 0x0
      Composite metric is (133120/128000), route is Internal
```

```
        Vector metric:
          Minimum bandwidth is 1544 Kbit
          Total delay is 5200 microseconds
          Reliability is 255/255
          Load is 1/255
          Minimum MTU is 1500
          Hop count is 1
          Originating router is 10.1.5.5
    10.1.14.1 (Ethernet0/0), from 10.1.14.1, Send flag is 0x0
        Composite metric is (134400/131840), route is Internal
        Vector metric:
          Minimum bandwidth is 100 Kbit
          Total delay is 5250 microseconds
          Reliability is 255/255
          Load is 1/255
          Minimum MTU is 1476
          Hop count is 3
          Originating router is 10.1.5.5
```

In order to implement unequal cost LB between the path R4-R5 and the path R4-R1-R3-R5, the feasibility condition should be met. FD=133120 and RD=131840. FD>RD so the route via the next hop 10.1.14.1 is a physical successor and can be used in the unequal load balancing.

At this point, we have no load-balancing from R4 to the 10.1.5.5.

```
R4#sh ip route
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

D*    0.0.0.0/0 is a summary, 00:01:37, Null0
      10.0.0.0/8 is variably subnetted, 16 subnets, 2 masks
D        10.1.1.1/32 [90/130560] via 10.1.14.1, 00:26:36, Ethernet0/0
D        10.1.2.2/32 [90/133120] via 10.1.14.1, 00:26:08, Ethernet0/0
D        10.1.3.3/32 [90/133120] via 10.1.14.1, 00:01:37, Ethernet0/0
C        10.1.4.4/32 is directly connected, Loopback0
D        10.1.5.5/32 [90/133120] via 10.1.45.5, 00:01:48, Serial3/0
D        10.1.6.6/32 [90/158720] via 10.1.14.1, 00:26:09, Ethernet0/0
D        10.1.9.9/32 [90/670720] via 10.1.14.1, 00:26:09, Ethernet0/0
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.4/32 is directly connected, Ethernet0/0
D        10.1.35.0/24 [90/6400] via 10.1.45.5, 00:01:37, Serial3/0
                      [90/6400] via 10.1.14.1, 00:01:37, Ethernet0/0
D        10.1.36.0/24 [90/30720] via 10.1.14.1, 00:26:09, Ethernet0/0
C        10.1.45.0/24 is directly connected, Serial3/0
L        10.1.45.4/32 is directly connected, Serial3/0
C        10.1.47.0/24 is directly connected, Ethernet0/1
L        10.1.47.4/32 is directly connected, Ethernet0/1
D        10.1.69.0/24 [90/542720] via 10.1.14.1, 00:26:09, Ethernet0/0
      11.0.0.0/24 is subnetted, 1 subnets
D        11.1.1.0 [90/5120] via 10.1.14.1, 00:26:36, Ethernet0/0
```

We would like to have the feasible successor route to be taken into account for load-balancing. The metric of the successor is 133120 and the metric of the feasible successor is 134400. By configuring variance 2, all the feasible successors with a metric within 133120 and 2*133120= 266240 are included in the paths used for load-balancing. Up to 6 paths are used for unequal load-balancing.

On R4, configure the following:

```
router eigrp 1
variance 2
```

We can see that the unequal load-balancing is implemented on R4.

```
R4#sh ip route
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

D*    0.0.0.0/0 is a summary, 00:00:06, Null0
      10.0.0.0/8 is variably subnetted, 16 subnets, 2 masks
D        10.1.1.1/32 [90/130560] via 10.1.14.1, 00:00:06, Ethernet0/0
D        10.1.2.2/32 [90/133120] via 10.1.14.1, 00:00:06, Ethernet0/0
D        10.1.3.3/32 [90/134400] via 10.1.45.5, 00:00:06, Serial3/0
                     [90/133120] via 10.1.14.1, 00:00:06, Ethernet0/0
C        10.1.4.4/32 is directly connected, Loopback0
D        10.1.5.5/32 [90/133120] via 10.1.45.5, 00:00:06, Serial3/0
                     [90/134400] via 10.1.14.1, 00:00:06, Ethernet0/0
D        10.1.6.6/32 [90/158720] via 10.1.14.1, 00:00:06, Ethernet0/0
D        10.1.9.9/32 [90/670720] via 10.1.14.1, 00:00:06, Ethernet0/0
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.4/32 is directly connected, Ethernet0/0
D        10.1.35.0/24 [90/6400] via 10.1.45.5, 00:00:06, Serial3/0
                      [90/6400] via 10.1.14.1, 00:00:06, Ethernet0/0
D        10.1.36.0/24 [90/30720] via 10.1.14.1, 00:00:06, Ethernet0/0
C        10.1.45.0/24 is directly connected, Serial3/0
L        10.1.45.4/32 is directly connected, Serial3/0
C        10.1.47.0/24 is directly connected, Ethernet0/1
L        10.1.47.4/32 is directly connected, Ethernet0/1
D        10.1.69.0/24 [90/542720] via 10.1.14.1, 00:00:06, Ethernet0/0
      11.0.0.0/24 is subnetted, 1 subnets
D        11.1.1.0 [90/5120] via 10.1.14.1, 00:00:06, Ethernet0/0
```

We also have to guarantee the load-balancing in the other direction, which is to say from R4 to R5.

```
R5#sh ip eigrp topology 10.1.4.4 255.255.255.255
EIGRP-IPv4 Topology Entry for AS(4)/ID(10.1.5.5) for 10.1.4.4/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 133120
  Descriptor Blocks:
  10.1.45.4 (Serial3/0), from 10.1.45.4, Send flag is 0x0
      Composite metric is (133120/128000), route is Internal
      Vector metric:
        Minimum bandwidth is 1544 Kbit
        Total delay is 5200 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 10.1.4.4
  10.1.35.3 (Serial4/0), from 10.1.35.3, Send flag is 0x0
      Composite metric is (134400/133120), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 5250 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1476
```

```
        Hop count is 3
        Originating router is 10.1.4.4
```

10.1.35.3 will not be a feasible successor because it doesn't meet the feasibility condition as FD=133120= RD. We apply an offset-list on R5 in order to increase the FD and make sure that FD>RD where RD is the reported distance as seen from the perspective of R3.

On R4, configure the following:

```
router eigrp 4
offset-list 0 out 10 serial 3/0
```

Once this is configured, the FD is incremented by 10 and the feasibility condition of the alternative route is met. We can then configure the variance and there will be load-balancing from R5 to R4.

```
R5#sh ip eigrp topology 10.1.4.4 255.255.255.255
EIGRP-IPv4 Topology Entry for AS(4)/ID(10.1.5.5) for 10.1.4.4/32
  State is Passive, Query origin flag is 1, 2 Successor(s), FD is 133120
  Descriptor Blocks:
  10.1.45.4 (Serial3/0), from 10.1.45.4, Send flag is 0x0
      Composite metric is (133130/128010), route is Internal
      Vector metric:
        Minimum bandwidth is 1544 Kbit
        Total delay is 5200 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 10.1.4.4
  10.1.35.3 (Serial4/0), from 10.1.35.3, Send flag is 0x0
      Composite metric is (134400/133120), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 5250 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1476
        Hop count is 3
        Originating router is 10.1.4.4
```

On R5, configure the following:

```
router eigrp 1
variance 2

R5#sh ip route
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 10.1.35.3 to network 0.0.0.0

D*    0.0.0.0/0 [90/2560] via 10.1.35.3, 00:04:26, Serial4/0
      10.0.0.0/8 is variably subnetted, 17 subnets, 2 masks
D        10.1.1.1/32 [90/135690] via 10.1.45.4, 00:03:57, Serial3/0
D        10.1.2.2/32 [90/138250] via 10.1.45.4, 00:03:57, Serial3/0
D        10.1.3.3/32 [90/138250] via 10.1.45.4, 00:03:57, Serial3/0
D        10.1.4.4/32 [90/133130] via 10.1.45.4, 00:03:57, Serial3/0
                     [90/134400] via 10.1.35.3, 00:03:57, Serial4/0
C        10.1.5.5/32 is directly connected, Loopback0
D        10.1.6.6/32 [90/163850] via 10.1.45.4, 00:03:57, Serial3/0
D        10.1.9.9/32 [90/675850] via 10.1.45.4, 00:03:57, Serial3/0
```

```
D        10.1.14.0/24 [90/7690] via 10.1.45.4, 00:03:57, Serial3/0
C        10.1.25.0/24 is directly connected, Serial5/0
L        10.1.25.2/32 is directly connected, Serial5/0
C        10.1.35.0/24 is directly connected, Serial4/0
L        10.1.35.5/32 is directly connected, Serial4/0
D        10.1.36.0/24 [90/35850] via 10.1.45.4, 00:03:57, Serial3/0
C        10.1.45.0/24 is directly connected, Serial3/0
L        10.1.45.5/32 is directly connected, Serial3/0
D        10.1.47.0/24 [90/30730] via 10.1.45.4, 00:03:57, Serial3/0
D        10.1.69.0/24 [90/547850] via 10.1.45.4, 00:03:57, Serial3/0
      11.0.0.0/24 is subnetted, 1 subnets
D        11.1.1.0 [90/10250] via 10.1.45.4, 00:03:57, Serial3/0
```

**Task 12.15** Configure R6 to send EIGRP hello packets every 1 s to R9.

On R6, configure the following:

```
interface Serial3/0
ip hello-interval eigrp 4 2
```

Please note that in EIGRP the hello and hold interval do not have to match between neighbors.

**Task 12.16** In the EIGRP domain, ensure that a router that has not replied to an EIGRP Query packets for 2 minutes is declared Stuck in Active.

On R1, R2, R3, R4, R5, R6, R7, and R9, configure the following:

```
router eigrp 4
timers active-time 2
```

**Task 12.17** On R9, configure a NSF during 5 minutes when the R6 NSF-capable router is undertaking a switchover.

On R9, configure the following:

```
router eigrp 4
  timers nsf route-hold 300
```

EIGRP NSF awareness is enabled by default. NFS route-hold timer ranges from 20-300 seconds. The default is 240 seconds.

**You have completed Lab 12**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 13: Configure and troubleshoot EIGRP (Part 3)

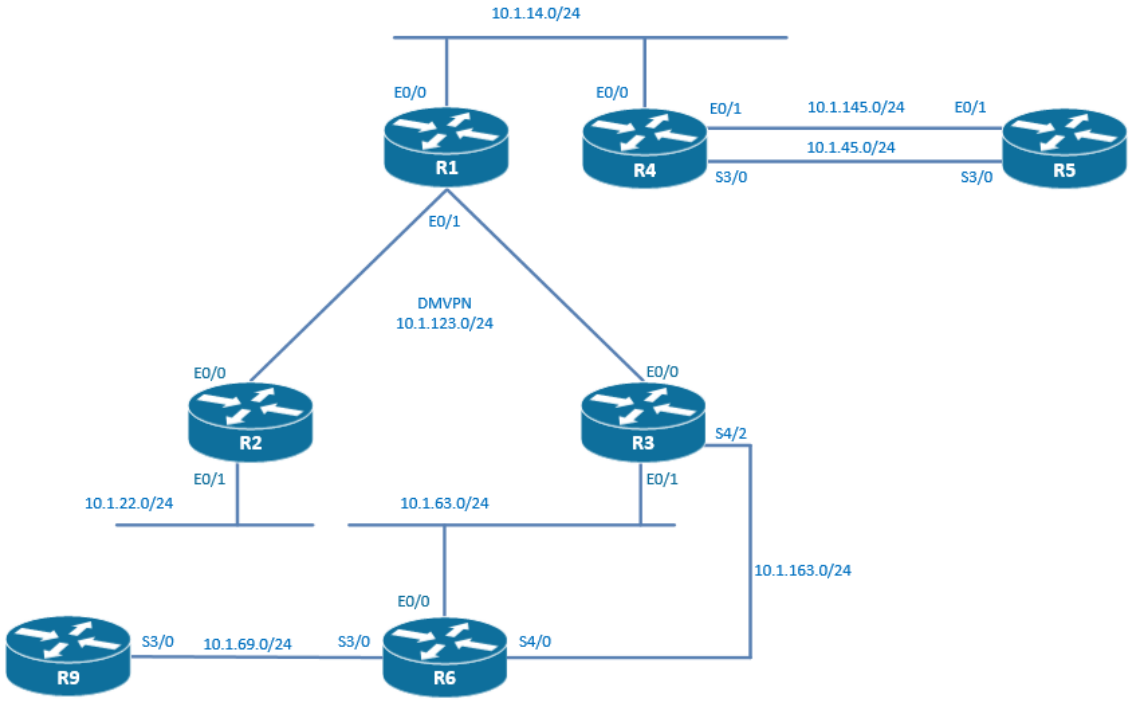
## Technologies covered

- Stub routing with leak-map
- Filtering with passive interfaces
- Filtering with distribute-list
- Filtering with offset-list
- Filtering with AD
- Filtering with route-maps
- Bandwidth pacing
- Neighbor logging
- Router-id
- Maximum hops

## Overview

You have been tasked to configure the routing reachability in your network using the EIGRP protocol.

The topology used in the lab will be the following:




**Estimated time to complete: 2-3 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 13.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Setup EIGRP routing in autonomous configuration mode with AS33 in this DMVPN network.

On R1, R2, and R3, configure the following:

```
router eigrp 33
network 11.1.1.0 0.0.0.255
```

In order for EIGRP to send updates from spoke to spoke, we have to disable split-horizon on the interface tunnel interface on R1.

As we are running DMVPN phase 2, we disable IP next-hop-self on the tu23 of R1. This way, traffic will be routed from spoke to spoke without transiting via the hub R1.

On R1, configure the following:

```
int tu23
no ip split-horizon eigrp 33
no ip next-hop-self eigrp 33
```

**Task 13.2** Advertise the loopbacks of R1, R2, and R3 in the EIGRP process. Use network statements.

On R1, configure the following:

```
router eigrp 33
 network 10.1.1.1 0.0.0.0
```

On R2, configure the following:

```
router eigrp 33
 network 10.1.2.2 0.0.0.0
```

On R3, configure the following:

```
router eigrp 33
 network 10.1.3.3 0.0.0.0
```

**Task 13.3** Configure EIGRP on the LAN between R3 and R6.

On R3 and R6, configure the following:

```
router eigrp 33
network 10.1.63.0 0.0.0.255
```

**Task 13.4** On R6, redistribute all the preconfigured loopbacks in the EIGRP process. Use network statements.

On R6, configure the following:

```
router eigrp 33
redistribute connected
```

**Task 13.5** Configure R2 and R3 as stub routers that advertised connected and summary routes.

On R2 and R3, configure the following:

```
router eigrp 33
eigrp stub connected summary
```

**Task 13.6** R3 should still advertise towards R1 the network 10.11.6.0/24, 10.22.6.0/24, and 10.33.6.0/24.

On R3, configure the following:

```
ip prefix-list Loopbacks_leaking seq 5 permit 10.11.6.0/24
ip prefix-list Loopbacks_leaking seq 10 permit 10.22.6.0/24
ip prefix-list Loopbacks_leaking seq 15 permit 10.33.6.0/24

route-map LEAK permit 10
  match ip address Loopbacks_leaking

router eigrp 33
eigrp stub connected summary leak-map LEAK
```

I have applied the leak-map but it doesn't seem to work. All the networks are leaked and not only the ones that are specified in the prefix-list Loopbacks_leaking. If you have an explanation, please post it in our iPexpert forum.

```
R1#sh ip route
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, 14 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
D        10.1.3.3/32 [90/27008000] via 11.1.1.3, 00:00:49, Tunnel23
D EX     10.1.6.6/32 [170/27033600] via 11.1.1.3, 00:00:49, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
D        10.1.63.0/24 [90/26905600] via 11.1.1.3, 00:00:49, Tunnel23
D EX     10.1.69.0/24 [170/27417600] via 11.1.1.3, 00:00:49, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
D EX     10.11.6.0/24 [170/27033600] via 11.1.1.3, 00:00:49, Tunnel23
D EX     10.22.6.0/24 [170/27033600] via 11.1.1.3, 00:00:49, Tunnel23
D EX     10.33.6.0/24 [170/27033600] via 11.1.1.3, 00:00:49, Tunnel23
D EX     10.44.6.0/24 [170/27033600] via 11.1.1.3, 00:00:49, Tunnel23
D EX     10.55.6.0/24 [170/27033600] via 11.1.1.3, 00:00:49, Tunnel23
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
```

**Task 13.7** Configure EIGRP on the serial connection between R3 and R6.

On R3, configure the following:

```
router eigrp 33
network 10.1.163.0 0.0.0.255
```

On R6, configure the following:

```
router eigrp 33
network 10.1.163.0 0.0.0.255
```

**Task 13.8** Configure EIGRP on the LAN between R1 and R4. Advertise the loopbacks of R4 in the EIGRP process. Use a network statement.

On R1, configure the following:

```
router eigrp 33
network 10.1.14.0 0.0.0.255
```

On R4, configure the following:

```
router eigrp 33
network 10.1.14.0 0.0.0.255
network 10.1.4.4 0.0.0.0
network 10.11.4.0 0.0.0.255
```

**Task 13.9** Configure a distribute-list with prefix-list to prevent R1 from advertising the network 10.1.4.4/32

Let's have a look at the routing table of R3:

```
R3#sh ip route
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, 19 subnets, 2 masks
D        10.1.1.1/32 [90/27008000] via 11.1.1.1, 01:09:41, Tunnel23
D        10.1.2.2/32 [90/28288000] via 11.1.1.2, 01:07:22, Tunnel23
C        10.1.3.3/32 is directly connected, Loopback0
D        10.1.4.4/32 [90/27033600] via 11.1.1.1, 00:04:41, Tunnel23
D EX     10.1.6.6/32 [170/409600] via 10.1.63.6, 01:09:41, Ethernet0/1
D        10.1.14.0/24 [90/26905600] via 11.1.1.1, 00:22:05, Tunnel23
C        10.1.63.0/24 is directly connected, Ethernet0/1
L        10.1.63.3/32 is directly connected, Ethernet0/1
D EX     10.1.69.0/24 [170/2195456] via 10.1.63.6, 01:09:41, Ethernet0/1
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
C        10.1.163.0/24 is directly connected, Serial4/2
L        10.1.163.3/32 is directly connected, Serial4/2
D        10.11.4.0/24 [90/27033600] via 11.1.1.1, 00:04:41, Tunnel23
D EX     10.11.6.0/24 [170/409600] via 10.1.63.6, 01:09:41, Ethernet0/1
D EX     10.22.6.0/24 [170/409600] via 10.1.63.6, 01:09:41, Ethernet0/1
D EX     10.33.6.0/24 [170/409600] via 10.1.63.6, 01:09:41, Ethernet0/1
D EX     10.44.6.0/24 [170/409600] via 10.1.63.6, 01:09:41, Ethernet0/1
D EX     10.55.6.0/24 [170/409600] via 10.1.63.6, 01:09:41, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.3/32 is directly connected, Tunnel23
```

On R1, configure the following:

```
ip prefix-list EIGRP_OUT seq 5 deny 10.1.4.4/32
ip prefix-list EIGRP_OUT seq 10 permit 0.0.0.0/0 le 32


router eigrp 33
distribute-list prefix EIGRP_OUT out
```

After applying the distribute-list, we can check that the 10.1.4.4/32 network has disappeared from the routing table of R3.

**Task 13.10** Configure a distribute-list with prefix-list to prevent R1 from learning the network 10.33.6.0/24

Let's have a look at the routing table of R1:

```
R1#sh ip route
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, 18 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
D        10.1.2.2/32 [90/27008000] via 11.1.1.2, 01:17:03, Tunnel23
D        10.1.3.3/32 [90/27008000] via 11.1.1.3, 00:04:07, Tunnel23
D        10.1.4.4/32 [90/409600] via 10.1.14.4, 00:14:22, Ethernet0/0
D EX     10.1.6.6/32 [170/27033600] via 11.1.1.3, 00:04:07, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
D        10.1.63.0/24 [90/26905600] via 11.1.1.3, 00:04:07, Tunnel23
D EX     10.1.69.0/24 [170/27417600] via 11.1.1.3, 00:04:07, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
D        10.1.163.0/24 [90/27392000] via 11.1.1.3, 00:04:07, Tunnel23
D        10.11.4.0/24 [90/409600] via 10.1.14.4, 00:14:22, Ethernet0/0
D EX     10.11.6.0/24 [170/27033600] via 11.1.1.3, 00:04:07, Tunnel23
D EX     10.22.6.0/24 [170/27033600] via 11.1.1.3, 00:04:07, Tunnel23
D EX     10.33.6.0/24 [170/27033600] via 11.1.1.3, 00:04:07, Tunnel23
D EX     10.44.6.0/24 [170/27033600] via 11.1.1.3, 00:04:07, Tunnel23
D EX     10.55.6.0/24 [170/27033600] via 11.1.1.3, 00:04:07, Tunnel23
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
```

On R1, configure the following:

```
ip prefix-list EIGRP_IN seq 5 deny 10.33.6.0/24
ip prefix-list EIGRP_IN seq 10 permit 0.0.0.0/0 le 32


router eigrp 33
distribute-list prefix EIGRP_IN in
```

After applying the distribute-list, we can check that the 10.33.6.0/24 network has disappeared from the routing table of R1.

**Task 13.11** Configure EIGRP on the connections between R4 and R5. Advertise the loopbacks of R5 in the EIGRP process. Use network statements. Make sure that the traffic is load-balanced on the 2 connections.

On R4, configure the following:

```
router eigrp 33
   network 10.1.45.0 0.0.0.255
```

```
network 10.1.145.0 0.0.0.255
variance 100
```

On R5, configure the following:

```
router eigrp 33
network 10.1.45.0 0.0.0.255
network 10.1.145.0 0.0.0.255
network 10.1.5.5 0.0.0.0
network 10.11.5.0 0.0.0.255
variance 100
```

On R4, we can check that we are load-balancing traffic between the 2 connections between R4 and R5.

```
R4#sh ip route
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, 22 subnets, 2 masks
D        10.1.1.1/32 [90/409600] via 10.1.14.1, 00:01:02, Ethernet0/0
D        10.1.2.2/32 [90/27033600] via 10.1.14.1, 00:01:02, Ethernet0/0
D        10.1.3.3/32 [90/27033600] via 10.1.14.1, 00:01:02, Ethernet0/0
C        10.1.4.4/32 is directly connected, Loopback1
D        10.1.5.5/32 [90/409600] via 10.1.145.5, 00:01:16, Ethernet0/1
                     [90/2297856] via 10.1.45.5, 00:01:16, Serial3/0
D EX     10.1.6.6/32 [170/27059200] via 10.1.14.1, 00:01:02, Ethernet0/0
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.4/32 is directly connected, Ethernet0/0
C        10.1.45.0/24 is directly connected, Serial3/0
L        10.1.45.4/32 is directly connected, Serial3/0
D        10.1.63.0/24 [90/26931200] via 10.1.14.1, 00:01:02, Ethernet0/0
D EX     10.1.69.0/24 [170/27443200] via 10.1.14.1, 00:01:02, Ethernet0/0
C        10.1.145.0/24 is directly connected, Ethernet0/1
L        10.1.145.4/32 is directly connected, Ethernet0/1
D        10.1.163.0/24 [90/27417600] via 10.1.14.1, 00:01:02, Ethernet0/0
C        10.11.4.0/24 is directly connected, Loopback11
L        10.11.4.4/32 is directly connected, Loopback11
D        10.11.5.0/24 [90/409600] via 10.1.145.5, 00:01:16, Ethernet0/1
                      [90/2297856] via 10.1.45.5, 00:01:16, Serial3/0
D EX     10.11.6.0/24 [170/27059200] via 10.1.14.1, 00:01:02, Ethernet0/0
D EX     10.22.6.0/24 [170/27059200] via 10.1.14.1, 00:01:02, Ethernet0/0
D EX     10.44.6.0/24 [170/27059200] via 10.1.14.1, 00:01:02, Ethernet0/0
D EX     10.55.6.0/24 [170/27059200] via 10.1.14.1, 00:01:02, Ethernet0/0
      11.0.0.0/24 is subnetted, 1 subnets
D        11.1.1.0 [90/26905600] via 10.1.14.1, 00:01:02, Ethernet0/0
      172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C        172.16.4.0/24 is directly connected, Loopback0
L        172.16.4.4/32 is directly connected, Loopback0
```

**Task 13.12** On R4, create a filter based on ACL. R4 should use the Ethernet connection to reach 10.1.5.5/32. Use a standard access-list to achieve this.

On R4, configure the following:

```
access-list 1 deny 10.1.5.5 0.0.0.0
access-list 1 permit any
```

```
router eigrp 33
distribute-list 1 in Serial3/0
```

On R4, we can check that R4 is using E0/1 to reach 10.1.5.5/32.

```
R4#sh ip route 10.1.5.5
Routing entry for 10.1.5.5/32
  Known via "eigrp 33", distance 90, metric 409600, type internal
  Redistributing via eigrp 33
  Last update from 10.1.145.5 on Ethernet0/1, 00:00:24 ago
  Routing Descriptor Blocks:
  * 10.1.145.5, from 10.1.145.5, 00:00:24 ago, via Ethernet0/1
      Route metric is 409600, traffic share count is 1
      Total delay is 6000 microseconds, minimum bandwidth is 10000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

**Task 13.13** On R4, create filters based on ACL. R4 should use the serial connection to reach 10.11.5.0/24. Use an extended access-list to achieve this.

On R4, configure the following:

```
access-list 100 deny ip host 10.1.145.5 host 10.11.5.0
access-list 100 permit ip any any

router eigrp 33
distribute-list 100 in E0/1
```

On R4, we can check that R4 is using serial3/0 to reach 10.11.5.0/24.

```
R4#sh ip route 10.11.5.0
Routing entry for 10.11.5.0/24
  Known via "eigrp 33", distance 90, metric 2297856, type internal
  Redistributing via eigrp 33
  Last update from 10.1.45.5 on Serial3/0, 00:01:04 ago
  Routing Descriptor Blocks:
  * 10.1.45.5, from 10.1.45.5, 00:01:04 ago, via Serial3/0
      Route metric is 2297856, traffic share count is 1
      Total delay is 25000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
```

**Task 13.14** Configure EIGRP on the serial connection between R6 and R9. Advertise the loopbacks of R9 in the EIGRP process except loopback 3. Use network statements. Between R3 and R6, make sure that the traffic is load-balanced between the serial interface and the ethernet interface.

On R3, configure the following:

```
router eigrp 33
variance 100
```

On R6, configure the following:

```
router eigrp 33
network 10.1.69.0 0.0.0.255
variance 100
```

On R9, configure the following:

```
router eigrp 33
network 10.1.69.0 0.0.0.255
network 10.1.9.9 0.0.0.255
network 10.11.9.9 0.0.0.255
```

**Task 13.15** On R3, create a filter based on offset-list. R3 should use the serial 4/2 connection to reach 10.1.9.9/32.

Let's check the routes to 10.1.9.9/32 on R3.

```
 R3# sh ip route 10.1.9.9
Routing entry for 10.1.9.9/32
  Known via "eigrp 33", distance 90, metric 2323456, type internal
  Redistributing via eigrp 33
  Last update from 10.1.163.6 on Serial4/2, 00:00:12 ago
  Routing Descriptor Blocks:
    10.1.163.6, from 10.1.163.6, 00:00:12 ago, via Serial4/2
      Route metric is 2809856, traffic share count is 33
      Total delay is 45000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
  * 10.1.63.6, from 10.1.63.6, 00:00:12 ago, via Ethernet0/1
      Route metric is 2323456, traffic share count is 40
      Total delay is 26000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
```

On R3, configure the following:

```
access-list 1 permit 10.1.9.9

router eigrp 33
offset-list 1 in 1000000000 E0/1
```

After applying the offset-list, R3 is only using S4/2 to reach 10.1.9.9/32.

```
R3# sh ip route 10.1.9.9
Routing entry for 10.1.9.9/32
  Known via "eigrp 33", distance 90, metric 2809856, type internal
  Redistributing via eigrp 33
  Last update from 10.1.163.6 on Serial4/2, 00:00:41 ago
  Routing Descriptor Blocks:
  * 10.1.163.6, from 10.1.163.6, 00:00:41 ago, via Serial4/2
      Route metric is 2809856, traffic share count is 1
      Total delay is 45000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
```

**Task 13.16** On R6, create a filter based on offset-list. R3 should use the E0/1 connection to reach 10.11.9.0/24.

On R6, configure the following:

```
access-list 1 permit 10.11.9.0

router eigrp 33
offset-list 1 out 1000000000 s4/0
```

Let's check the route to 10.11.9.0 on R3:

```
R3# sh ip route 10.11.9.0
Routing entry for 10.11.9.0/24
  Known via "eigrp 33", distance 90, metric 2323456, type internal
  Redistributing via eigrp 33
  Last update from 10.1.63.6 on Ethernet0/1, 00:01:37 ago
  Routing Descriptor Blocks:
  * 10.1.63.6, from 10.1.63.6, 00:01:37 ago, via Ethernet0/1
      Route metric is 2323456, traffic share count is 1
      Total delay is 26000 microseconds, minimum bandwidth is 1544 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
```

**Task 13.17** Configure R1 not to install the route 10.11.6.0/24 when received from R3. Manipulate AD.

On R1, configure the following:

```
ip access-list standard R6_loopbacks
  permit 10.11.6.0

router eigrp 33
distance 255 11.1.1.3 0.0.0.0 R6_loopbacks
```

After applying this command, I had to reload the router R1 for the change to take effect. Resetting the EIGRP neighborships was not enough.

**Task 13.18** Configure R1 not to install the route 10.22.6.0/24 when received from R3. Manipulate AD.

On R1, configure the following:

```
ip access-list standard R6_loopbacks
  permit 10.22.6.0
```

**Task 13.19** On R9, there is a preconfigured static route to 172.16.1.0/24. Redistribute this static route into EIGRP and tag this route with a tag of 666.

On R9, configure the following:

```
route-map STATIC_TO_EIGRP permit 10
 set metric 1000 100 255 1 1500
 set tag 666

router eigrp 33
redistribute static route-map STATIC_TO_EIGRP
```

**Task 13.20** Filter on R6 this route out based on the tag 666.

On R3, the static route redistributed in the previous question is present in the routing table:

```
R3#sh ip route
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, 16 subnets, 2 masks
C        10.1.3.3/32 is directly connected, Loopback0
D EX     10.1.6.6/32 [170/2297856] via 10.1.163.6, 00:13:42, Serial4/2
                     [170/409600] via 10.1.63.6, 00:13:42, Ethernet0/1
D        10.1.9.9/32 [90/2809856] via 10.1.163.6, 00:13:42, Serial4/2
C        10.1.63.0/24 is directly connected, Ethernet0/1
L        10.1.63.3/32 is directly connected, Ethernet0/1
D        10.1.69.0/24 [90/2681856] via 10.1.163.6, 00:13:42, Serial4/2
                      [90/2195456] via 10.1.63.6, 00:13:42, Ethernet0/1
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
C        10.1.163.0/24 is directly connected, Serial4/2
L        10.1.163.3/32 is directly connected, Serial4/2
```

```
D EX     10.11.6.0/24 [170/2297856] via 10.1.163.6, 00:13:42, Serial4/2
                      [170/409600] via 10.1.63.6, 00:13:42, Ethernet0/1
D        10.11.9.0/24 [90/2323456] via 10.1.63.6, 00:13:42, Ethernet0/1
D EX     10.22.6.0/24 [170/2297856] via 10.1.163.6, 00:13:42, Serial4/2
                      [170/409600] via 10.1.63.6, 00:13:42, Ethernet0/1
D EX     10.33.6.0/24 [170/2297856] via 10.1.163.6, 00:31:18, Serial4/2
                      [170/409600] via 10.1.63.6, 00:31:18, Ethernet0/1
D EX     10.44.6.0/24 [170/2297856] via 10.1.163.6, 00:13:42, Serial4/2
                      [170/409600] via 10.1.63.6, 00:13:42, Ethernet0/1
D EX     10.55.6.0/24 [170/2297856] via 10.1.163.6, 00:13:42, Serial4/2
                      [170/409600] via 10.1.63.6, 00:13:42, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.3/32 is directly connected, Tunnel23
      172.16.0.0/24 is subnetted, 1 subnets
D EX     172.16.1.0 [170/3609600] via 10.1.163.6, 00:05:29, Serial4/2
                    [170/3123200] via 10.1.63.6, 00:05:29, Ethernet0/1
```

On R6, configure the following:

```
route-map FILTER_EIGRP deny 10
  match tag 666
 route-map FILTER_EIGRP permit 20

router eigrp 33
  distribute-list route-map FILTER_EIGRP in
```

After applying the filter based on tag 666 on R6, the route 172.16.1.0 has disappeared from the routing table on R3.

**Task 13.21** On the serial connection between R4 and R5, make sure that EIGRP control traffic cannot exceed 25% of the bandwidth.

On R4 and R5, configure the following:

```
int s3/0
ip bandwidth-percent eigrp 33 25
```

**Task 13.22** The R4 and R5 routers should log EIGRP neighbor relationship changes.

On R4 and R5, configure the following:

```
router eigrp 33
eigrp log-neighbor-changes
```

**Task 13.23** On R9, configure an EIGRP router-id as 9.9.9.9 and redistribute the loopback3 into EIGRP.

On R9, configure the following:

```
route-map LOOPBACK3 permit 10
  match interface Loopback3

router eigrp 33
  redistribute connected route-map LOOPBACK3
  eigrp router-id 10.1.9.9
```

**Task 13.24** On R6, configure the EIGRP process to reject the 10.22.9.0/24 network. You are only allowed to change the EIGRP router-id.

By configuring the same router-id on R6 and R9, you will make sure that EIGRP is confused and will not accept updates with the same router-id.

On R6, configure the following:

```
router eigrp 33
eigrp router-id 10.1.9.9
```

**Task 13.25** On the R6 and R9, configure the EIGRP process to reject EIGRP packets that have transited over more than 10 hops.

On R6 and R9, configure the following:

```
router eigrp 33
metric maximum-hops 10
```

**You have completed Lab 13**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 14: Configure and troubleshoot OSPF (Part 1)

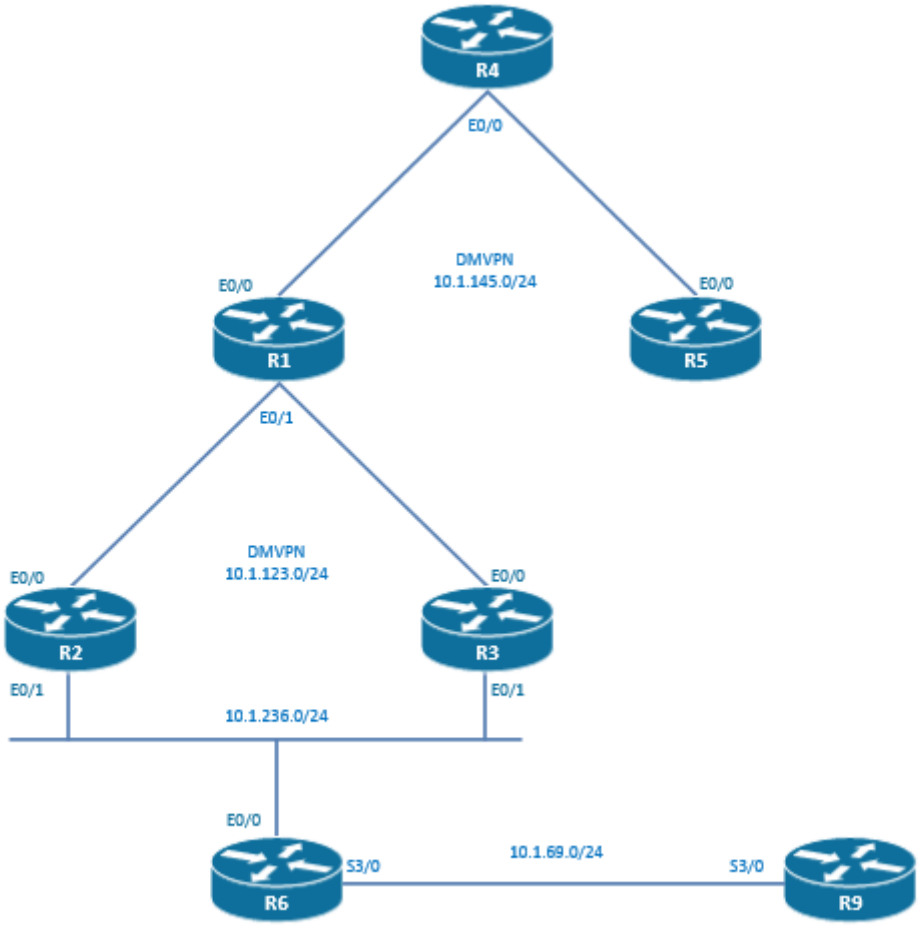
## Technologies covered

- DR/BDR
- OSPF network types
- OSPF path selection
- OSPF per neighbor cost
- OSPF auto-cost reference bandwidth
- OSPF version 3 address-family support

## Overview

You have been tasked to configure the routing in a network using OSPF.

The topology used in the lab will be the following:




**Estimated time to complete: 3-4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 14.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Configure OSPF process 1 area 0 in this network. The election of a DR should not take place. On routers R2 and R3, you are not allowed to change the default network type and not allowed to modify the timers.

Before configuring the routing protocol, let's analyze the DMVPN configuration of the tu23 on R1, R2, and R3. R1 is the hub router and R2 and R3 are the spoke routers. We are running DMVPN phase 2 with dynamic mappings. Multicast support has been enabled with the "ip nhrp map multicast dynamic" on the hub router and "ip nhrp map multicast 10.1.123.1" on the spoke routers.

The default OSPF network type of a tunnel interface is point-to-point. As this tunnel is a mGRE tunnel, the OSPF network type has to be changed to either broadcast type or to the point-to-multipoint type. It is stated in the question that the election of a DR should not take place. The only option that we have is to enable point-to-multipoint on the hub routers. We could also change the OSPF network type to point-to-multipoint on the spokes and we will have an up and running OSPF configuration. In the question, it is stated that on routers R2 and R3, we are not allowed to change the default network type which is point-to-point. The point-to-point network type and the point-to-multipoint network type are not having the same OSPF timers. The OSPF timers are 10/40 seconds for the hello/dead timers for the point-to-point network type. The OSPF timers are 30/120 seconds for the hello/dead timers for the point-to-multipoint network type. The OSPF timers have to match in order for OSPF to bring an adjacency up. So we could either adjust the spokes to the timers of the hub, or adjust the hub to the timers of the spokes. As we are instructed not to modify the timers on the spokes, the solution is to decrease the timers on the hub to match those on the spoke.

On R1, configure the following:

```
interface Tunnel23
ip ospf network point-to-multipoint
ip ospf dead-interval 40
ip ospf hello-interval 10

router ospf 1
network 11.1.1.0 0.0.0.255 area 0
```

On R2, configure the following:

```
router ospf 1
 network 11.1.1.0 0.0.0.255 area 0
```

On R3, configure the following:

```
router ospf 1
 network 11.1.1.0 0.0.0.255 area 0
```

Let's check that the OSPF adjacencies are up and running and that no DR has been elected.

```
R1#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.2.2          0   FULL/  -        00:00:32    11.1.1.2        Tunnel23
10.1.3.3          0   FULL/  -        00:00:32    11.1.1.3        Tunnel23
```

**Task 14.2** R4, R5, and R1 are also in a hub and spoke topology where R4 is the hub and R1 and R5 are the spokes. DMVPN is the underlying used technology. Configure OSPF process 1 area 0 in this network. The election of a DR should take place in this network. The DR should always be on the hub router. Multicast is not enabled on the DMVPN tunnels.

As described in the question, multicast support is not enabled on the interface tu15 interfaces of R1, R4, and R5. No "ip nhrp map multicast" command is configured neither on the hub router R4, nor on the spoke routers R1 and R5.

As also described in the question, the election of a DR has to take place. A DR is elected when the OSPF network type is NBMA or broadcast. As multicast is not enabled, configuring the broadcast OSPF network type is not going to be possible. The hub's and the spoke's tunnel15 interfaces will then be configured as non-broadcast network type interfaces. The OSPF hellos and updates packets have to be sent as unicast packets because multicast is not supported. This will be accomplished by using the neighbor statements on the hub R4.

The DR should always be located on the hub router. A priority of 0 is configured on the spokes. This priority makes them ineligible as DR.

The following have to be configured on R4:

```
interface Tunnel15
ip ospf network non-broadcast

router ospf 1
 network 44.1.1.0 0.0.0.255 area 0
 neighbor 44.1.1.1
 neighbor 44.1.1.5
```

The following have to be configured on R1:

```
interface Tunnel15
ip ospf network non-broadcast
ip ospf priority 0

router ospf 1
network 44.1.1.0 0.0.0.255 area 0
```

The following have to be configured on R5:

```
interface Tunnel15
ip ospf network non-broadcast
ip ospf priority 0

router ospf 1
network 44.1.1.0 0.0.0.255 area 0
```

We can check that the OSPF neighborships are up and running and that R4 is the DR:

```
R4#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.1.1          0   FULL/DROTHER    00:01:46    44.1.1.1        Tunnel15
10.1.5.5          0   FULL/DROTHER    00:01:40    44.1.1.5        Tunnel15
```

**Task 14.3** On R1, R2, R3, R4, and R5, configure the loopbacks 0 as the OSPF router-ids and advertise the loopback0 of the routers into OSPF in the following areas:

| | |
|---|---|
| R1 | Area 1 |
| R2 | Area 2 |
| R3 | Area 3 |
| R4 | Area 4 |
| R5 | Area 5 |

Check that you have full reachability between the loopbacks, especially on R2, check that you can ping the loopback of R5 sourcing from the loopback of R2.

On R1, the following has to be configured:
```
router ospf 1
router-id 10.1.1.1
network 10.1.1.0 255.255.255.0 area 1
```

On R2, the following has to be configured:
```
router ospf 1
router-id 10.1.2.2
network 10.1.2.0 255.255.255.0 area 2
```

On R3, the following has to be configured:
```
router ospf 1
router-id 10.1.3.3
network 10.1.3.0 255.255.255.0 area 3
```

On R4, the following has to be configured:
```
router ospf 1
router-id 10.1.4.4
network 10.1.4.0 255.255.255.0 area 4
```

On R5, the following has to be configured:
```
router ospf 1
router-id 10.1.5.5
network 10.1.5.0 255.255.255.0 area 5
```

In order to enforce the use of the new OSPF router-id, we have to perform the following on R1, R2, R3, R4, and R5:
```
R5#clear ip ospf 1 process
Reset OSPF process 1? [no]: yes
```

Let's check that IP connectivity from R2 to R5 is established.
```
R2#ping 10.1.5.5 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

### **Task 14.4** Configure the network 10.1.236.0/24 into area 236 on R2, R3, and R6.

On R2, R3, and R6, configure the following:
```
router ospf 1
network 10.1.236.0 255.255.255.0 area 236
```

OSPF is up and running on the network 10.1.236.0/24:
```
R6#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.2.2          1   FULL/DROTHER    00:00:33    10.1.236.2      Ethernet0/0
10.1.3.3          1   FULL/BDR        00:00:38    10.1.236.3      Ethernet0/0
```

**Task 14.5** The R2 should always be elected as the DR, and R3 should always be elected as the BDR.

By default, all OSPF routers are assigned a DR priority of 1. Ties among routers with equal DR priorities are broken by router ID, with the highest RID being preferred. That's why, in the current situation, the router R6 with router ID 10.1.6.6 is the DR because it has the highest router-id between R2, R3, and R6.

```
R6#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.2.2          1   FULL/DROTHER    00:00:33    10.1.236.2      Ethernet0/0
10.1.3.3          1   FULL/BDR        00:00:38    10.1.236.3      Ethernet0/0
```

Let's configure R2 with the highest priority possible which is 255. R3 will be configured with a priority of 254 in order to make it the BDR. R6 will be left to the default priority.

On R2, configure the following:

```
int e0/1
ip ospf priority 255
```

On R3, configure the following:

```
int e0/1
ip ospf priority 254

Let's check the OSPF adjacencies once the priority changes are done:

R6#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.2.2        255   FULL/DROTHER    00:00:39    10.1.236.2      Ethernet0/0
10.1.3.3        254   FULL/BDR        00:00:33    10.1.236.3      Ethernet0/0
```

The DR is still the router R6 because changing the priority is not preemptive. In order to have it enforced, the OSPF process on the current DR has to be cleared.

```
R6#clear ip ospf 1 process
Reset OSPF process 1? [no]: yes

%OSPF-5-ADJCHG: Process 1, Nbr 10.1.2.2 on Ethernet0/0 from FULL to DOWN, Neighbor Down:
Interface down or detached
%OSPF-5-ADJCHG: Process 1, Nbr 10.1.3.3 on Ethernet0/0 from FULL to DOWN, Neighbor Down:
Interface down or detached

%OSPF-5-ADJCHG: Process 1, Nbr 10.1.2.2 on Ethernet0/0 from LOADING to FULL, Loading Done
%OSPF-5-ADJCHG: Process 1, Nbr 10.1.3.3 on Ethernet0/0 from LOADING to FULL, Loading Done

R6#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.2.2        255   FULL/BDR        00:00:39    10.1.236.2      Ethernet0/0
10.1.3.3        254   FULL/DR         00:00:39    10.1.236.3      Ethernet0/0
```

Now that the OSPF process has been cleared on R6, we can notice that R3 has taken over the DR role. This is due to the fact that R3 was the BDR when R6 OSPF process was cleared.

We have now to clear the OSPF process on R3:

```
R3#clear ip ospf process
Reset ALL OSPF processes? [no]: yes

%OSPF-5-ADJCHG: Process 1, Nbr 10.1.1.1 on Tunnel23 from FULL to DOWN, Neighbor Down:
Interface down or detached
%OSPF-5-ADJCHG: Process 1, Nbr 10.1.2.2 on Ethernet0/1 from FULL to DOWN, Neighbor Down:
Interface down or detached
```

```
%OSPF-5-ADJCHG: Process 1, Nbr 10.1.6.6 on Ethernet0/1 from FULL to DOWN, Neighbor Down:
Interface down or detached
%OSPF-5-ADJCHG: Process 1, Nbr 10.1.1.1 on Tunnel23 from LOADING to FULL, Loading Done
%OSPF-5-ADJCHG: Process 1, Nbr 10.1.2.2 on Ethernet0/1 from LOADING to FULL, Loading Done
%OSPF-5-ADJCHG: Process 1, Nbr 10.1.6.6 on Ethernet0/1 from LOADING to FULL, Loading Done
```

We have now the desired situation with R2 in the role of the DR and R3 in the role of the BDR.

```
R6#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.2.2        255   FULL/DR         00:00:33    10.1.236.2      Ethernet0/0
10.1.3.3        254   EXSTART/BDR     00:00:32    10.1.236.3      Ethernet0/0
```

**Task 14.6** Advertise only the loopback 0 of R6 into OSPF area 236. Do not use a network statement. On R5, check that you can ping the loopback of R6 sourcing from the loopback of R5.

On R6, configure the following:

```
route-map CONNECTED permit 10
 match interface Loopback0

router ospf 1
 redistribute connected subnets route-map CONNECTED
```

Let's verify that the loopback of R6 can be pinged from R5:

```
R5#ping 10.1.6.6 source 10.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.6.6, timeout is 2 seconds:
Packet sent with a source address of 10.1.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 14.7** We are going to have links faster than 100M in the network. In the whole OSPF network, a gigaethernet link should have a cost of 1 and a fast ethernet link should have a 10.

The default reference bandwidth for OSPF is 10^8 bps or 100Mbps. The problem with this default reference bandwidth is that a link with 100 Mbps, 1 Gbps and 10 Gbps has the same cost of 1.
In order to have a gigaethernet link with a cost of 1 and a fasethernet link with a cost of 10, we have to configure a reference bandwidth of 10^9 bps or 1000 Mbps.

On R1, R2, R3, R4, R5, R6, and R9, configure the following:

```
router ospf 1
auto-cost reference-bandwidth 1000
```

**Task 14.8** Manipulate the OSPF cost so that R1 prefers R2 over R3 to reach the loopback of R6. Do not configure anything under the interfaces.

Let's have a look at the routing table of R1. The traffic from R1 to the loopback0 of R6 is load-balanced between R2 and R3. This route has a metric of 20.

```
R1#sh ip route
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, 12 subnets, 2 masks
C        10.1.1.0/24 is directly connected, Loopback0
L        10.1.1.1/32 is directly connected, Loopback0
O IA     10.1.2.2/32 [110/10001] via 11.1.1.2, 00:08:38, Tunnel23
O IA     10.1.3.3/32 [110/10001] via 11.1.1.3, 00:08:38, Tunnel23
O IA     10.1.4.4/32 [110/10001] via 44.1.1.4, 00:08:38, Tunnel15
O IA     10.1.5.5/32 [110/10001] via 44.1.1.5, 00:08:38, Tunnel15
O E2     10.1.6.6/32 [110/20] via 11.1.1.3, 00:19:43, Tunnel23
                     [110/20] via 11.1.1.2, 00:08:03, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
C        10.1.145.0/24 is directly connected, Ethernet0/0
L        10.1.145.1/32 is directly connected, Ethernet0/0
O IA     10.1.236.0/24 [110/10100] via 11.1.1.3, 00:08:13, Tunnel23
                       [110/10100] via 11.1.1.2, 00:08:13, Tunnel23
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
      44.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        44.1.1.0/24 is directly connected, Tunnel15
L        44.1.1.1/32 is directly connected, Tunnel15
```

We are going to use the neighbor cost command to influence the OSPF Path. The 'neighbor address cost' command can only be used with OSPF point-to-multipoint mode. As the metric via R3 is 20, we are going to assign a metric of 10 to the path via R2.

On R1, configure the following:

```
router ospf 1
neighbor 11.1.1.2 cost 10
```

The traffic is now transiting via R2 and is not load.balanced anymore, as we can see in the routing table of R1.

```
R1#sh ip route
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, 12 subnets, 2 masks
C        10.1.1.0/24 is directly connected, Loopback0
L        10.1.1.1/32 is directly connected, Loopback0
O IA     10.1.2.2/32 [110/11] via 11.1.1.2, 00:00:34, Tunnel23
O IA     10.1.3.3/32 [110/22] via 11.1.1.3, 00:02:22, Tunnel23
O IA     10.1.4.4/32 [110/10001] via 44.1.1.4, 00:25:33, Tunnel15
O IA     10.1.5.5/32 [110/10001] via 44.1.1.5, 00:25:33, Tunnel15
O E2     10.1.6.6/32 [110/20] via 11.1.1.2, 00:00:34, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
C        10.1.145.0/24 is directly connected, Ethernet0/0
L        10.1.145.1/32 is directly connected, Ethernet0/0
O IA     10.1.236.0/24 [110/110] via 11.1.1.2, 00:00:34, Tunnel23
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
```

```
      44.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        44.1.1.0/24 is directly connected, Tunnel15
L        44.1.1.1/32 is directly connected, Tunnel15
```

**Task 14.9** Configure OSPF version 3 area 0 for IPv4 between R6 and R9.

Use the following global unicast addresses:

| | |
|---|---|
| R6 s3/0 | 2001: :6/64 |
| R9 s3/0 | 2001: :9/64 |

We are going to use OSPFv3 in order to route IPv4 prefixes. Enable OSPFv3 for IPv4 on R6 and R9:

```
ipv6 unicast-routing
router ospfv3 1
address-family ipv4 unicast
```

In order to do that, we are going to configure the IPv6 addresses and enable OSPFv3 for IPv4 on the link between R6 and R9.

This configuration has to be applied on R6:

```
interface Serial3/0
ipv6 address 2001::6/64
 ospfv3 1 ipv4 area 0
```

This configuration has to be applied on R9:

```
interface Serial3/0
ipv6 address 2001::9/64
 ospfv3 1 ipv4 area 0
```

**Task 14.10** Create the following IPv4 address loopback1:

| | |
|---|---|
| R6 | 20.1.6.6/32 |
| R9 | 20.1.9.9/32 |

This configuration has to be applied on R6:

```
interface Loopback1
 ip address 20.1.6.6 255.255.255.255
```

This configuration has to be applied on R9:

```
interface Loopback1
 ip address 20.1.9.9 255.255.255.255
```

**Task 14.11** Advertise the IPv4 address loopback1 of R6 and R9 into area 0 of the OSPF version 3 processes.

If necessary, use the IPv6 following address for the loopback0:

| | |
|---|---|
| R6 | 2001:bd8: :6/64 |
| R9 | 2001:bd8: :9/64 |

When trying to enable OSPFv3 on the loopback1, the following is happening:

```
R6(config)#int lo1
R6(config-if)#ospfv3 1 ipv4 area 0
% OSPFv3: IPV6 is not enabled on this interface
```

We can see that, even if we are only routing IPv4 prefixes, it is necessary to configure unicast IPv6 address on an interface on which we want to enable OSPFv3.

This configuration has to be applied on R6:

```
interface Loopback1
 ipv6 address 2001:BD8::6/64
 ospfv3 1 ipv4 area 0
```

This configuration has to be applied on R9:

```
interface Loopback1
 ipv6 address 2001:BD8::9/64
 ospfv3 1 ipv4 area 0
```

**Task 14.12** On R6, make sure that you can ping the loopback of R9 sourcing from the loopback of R6.

Routing IPv4 with OSPFv3 is working!

```
R6#ping 20.1.9.9 source 20.1.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 20.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 20.1.6.6
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

Please note that OSPFv3 and OSPFv2 are not backward-compatible so we can only ping the loopback1 of R9 from the loopback1 of R6.

**You have completed Lab 14**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 15: Configure and troubleshoot OSPF (Part 2)

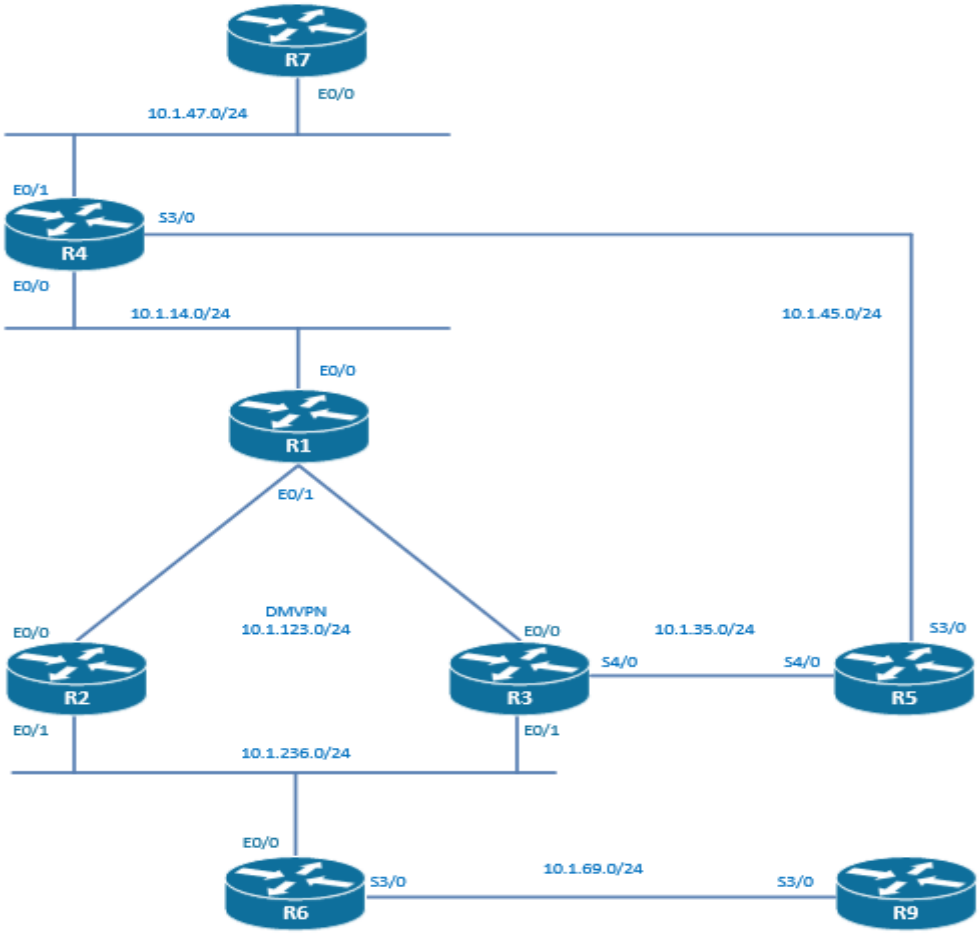
## Technologies covered

- Discontiguous area
- Virtual-links
- GRE tunnels
- Non-backbone transit area
- OSPF authentication
- Flood reduction
- Demand circuit
- Summarization
- Discard-route
- Flood reduction

## Overview

You have been tasked to configure OSPF as the routing protocol of your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 15.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Configure OSPF process 1 area 0 in this network. The election of a DR should take place in this network. The DR should always be on the hub router.

On R1, configure the following:

```
int tu23
ip ospf network broadcast
ip ospf priority 10

router ospf 1
network 11.1.1.0 255.255.255.0 area 0
```

On R2, configure the following:

```
int tu23
ip ospf network broadcast
ip ospf priority 0

router ospf 1
network 11.1.1.0 255.255.255.0 area 0
```

On R3, configure the following:

```
int tu23
ip ospf network broadcast
ip ospf priority 0

router ospf 1
network 11.1.1.0 255.255.255.0 area 0
```

**Task 15.2** The loopback0 networks of R1, R2, and R3 should present in the OSPF database of R1 as LSAs type 1.

On R1, configure the following:

```
router ospf 1
network 10.1.1.1 255.255.255.255 area 0
```

On R2, configure the following:

```
router ospf 1
network 10.1.2.2 255.255.255.255 area 0
```

On R3, configure the following:

```
router ospf 1
network 10.1.3.3 255.255.255.255 area 0


R1#sh ip ospf database
```

```
            OSPF Router with ID (10.1.1.1) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.1.1        10.1.1.1        80          0x80000003 0x00965B 2
10.1.2.2        10.1.2.2        75          0x80000003 0x00A149 2
10.1.3.3        10.1.3.3        64          0x80000003 0x00AC37 2

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
11.1.1.1        10.1.1.1        80          0x80000001 0x00A944
```

**Task 15.3** Configure the network 10.1.236.0/24 into area 236 on R2, R3, and R6. Redistribute only the loopback0 of R6 into the area 236.

On R2, configure the following:

```
router ospf 1
network 10.1.236.0 255.255.255.0 area 236
```

On R3, configure the following:

```
router ospf 1
network 10.1.236.0 255.255.255.0 area 236
```

On R6, configure the following:

```
route-map Loopback0
match interface loopback0

router ospf 1
network 10.1.236.0 255.255.255.0 area 236
redistribute connected route-map Loopback0 subnets

R1#ping 10.1.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 15.4** Configure the network 10.1.69.0/24 into area 69 on R6 and R9. Add the loopback0 of R9 into the area 69 process as a network statement.

On R6, configure the following:

```
router ospf 1
network 10.1.69.0 255.255.255.0 area 69
```

On R9, configure the following:

```
router ospf 1
network 10.1.69.0 255.255.255.0 area 69
network 10.1.9.9 255.255.255.255 area 69
```

**Task 15.5** Configure area 236 as a stub area.

On R2, configure the following:

```
router ospf 1
area 236 stub
```

On R3, configure the following:

```
router ospf 1
area 236 stub
```

On R6, configure the following:

```
router ospf 1
area 236 stub
```

**Task 15.6** Ensure that there is IP connectivity between the loopback0 of R9 and the loopback0 of R1. Do not use a virtual-link, as the transit area is a stub area. The path through R3 should be used. Use an IP address of 36.0.0.3/24 and 36.0.0.6/24 when necessary.

On R3, configure the following:

```
int tunnel 1
ip add 36.0.0.3 255.255.255.0
tunnel source 10.1.236.3
tunnel destination 10.1.236.6

router ospf 1
network 36.0.0.0 255.255.255.0 area 0
```

On R6, configure the following:

```
int tunnel 1
ip add 36.0.0.6 255.255.255.0
tunnel source 10.1.236.6
tunnel destination 10.1.236.3

router ospf 1
network 36.0.0.0 255.255.255.0 area 0

R9#ping 10.1.1.1 source 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.9.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 15.7** Configure the network 10.1.14.0/24 into area 14 on R1 and R4. Add the loopback0 of R4 into the area 14 process as a network statement.

On R1, configure the following:

```
router ospf 1
network 10.1.14.0 255.255.255.0 area 14
```

On R4, configure the following:

```
router ospf 1
network 10.1.14.0 255.255.255.0 area 14
network 10.1.4.4 255.255.255.255 area 14
```

**Task 15.8** Configure the network 10.1.47.0/24 into area 47 on R4 and R7. Add the loopback0 of R7 into the area 47 process as a network statement.

On R4, configure the following:

```
router ospf 1
network 10.1.47.0 255.255.255.0 area 47
```

On R7, configure the following:

```
router ospf 1
network 10.1.47.0 255.255.255.0 area 47
network 10.1.7.7 255.255.255.255 area 47
```

**Task 15.9** Ensure that there is IP connectivity between the loopback0 of R7 and the loopback0 of R2.

On R1, configure the following:

```
router ospf 1
router-id 10.1.1.1
area 14 virtual-link 10.1.4.4
```

On R4, configure the following:

```
router ospf 1
router-id 10.1.4.4
area 14 virtual-link 10.1.1.1

R1#sh ip ospf virtual-links
Virtual Link OSPF_VL0 to router 10.1.4.4 is up
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 14, via interface Ethernet0/0
 Topology-MTID    Cost    Disabled     Shutdown      Topology Name
        0           10        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 00:00:03
    Adjacency State FULL (Hello suppressed)
    Index 3/4, 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

R7#ping 10.1.2.2 source 10.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.7.7
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 15.10** Configure the network 10.1.35.0/24 to be part of area 0.

On R3, configure the following:

```
router ospf 1
network 10.1.35.0 255.255.255.0 area 0
```

On R5, configure the following:

```
router ospf 1
network 10.1.35.0 255.255.255.0 area 0
```

**Task 15.11** Configure the network 10.1.45.0/24 and the network 10.1.5.5/32 to be part of area 45.

On R4, configure the following:

```
router ospf 1
network 10.1.45.0 255.255.255.0 area 45
```

On R5, configure the following:

```
router ospf 1
network 10.1.45.0 255.255.255.0 area 45
network 10.1.5.5 255.255.255.255 area 45
```

**Task 15.12** Configure an OSPF cost of 60000 on the interfaces belonging to the network 10.1.14.0/24.

On R1 and R4, configure the following:

```
int e0/0
ip ospf cost 60000
```

**Task 15.13** On R7, when performing a trace route from the loopback of R7 to the loopback of R3, we can observe that the trace route is following the path R7, R4, R5, and R3. The routing is using a non-backbone area, that is to say area 45, as a transit. Without modifying any OSPF costs, ensure that the trace route is using the R7, R4, R1, and R3 path.

The total OSPF cost of the path R7, R4, R5, and R3 is the following:

```
R7#sh ip route 10.1.4.4
Routing entry for 10.1.4.4/32
  Known via "ospf 1", distance 110, metric 11, type inter area
  Last update from 10.1.47.4 on Ethernet0/0, 01:12:35 ago
  Routing Descriptor Blocks:
  * 10.1.47.4, from 10.1.4.4, 01:12:35 ago, via Ethernet0/0
      Route metric is 11, traffic share count is 1

R4#sh ip route 10.1.5.5
Routing entry for 10.1.5.5/32
  Known via "ospf 1", distance 110, metric 65, type intra area
  Last update from 10.1.45.5 on Serial3/0, 00:00:37 ago
  Routing Descriptor Blocks:
  * 10.1.45.5, from 10.1.5.5, 00:00:37 ago, via Serial3/0
      Route metric is 65, traffic share count is 1

R5#sh ip route 10.1.3.3
Routing entry for 10.1.3.3/32
  Known via "ospf 1", distance 110, metric 65, type intra area
  Last update from 10.1.35.3 on Serial4/0, 00:41:09 ago
  Routing Descriptor Blocks:
  * 10.1.35.3, from 10.1.3.3, 00:41:09 ago, via Serial4/0
      Route metric is 65, traffic share count is 1
```

The total cost is 11+65+65= 141

The total OSPF cost of the path R7, R4, R1, and R3 is the following:

```
R7#sh ip route 10.1.4.4
Routing entry for 10.1.4.4/32
  Known via "ospf 1", distance 110, metric 11, type inter area
  Last update from 10.1.47.4 on Ethernet0/0, 01:12:35 ago
  Routing Descriptor Blocks:
  * 10.1.47.4, from 10.1.4.4, 01:12:35 ago, via Ethernet0/0
      Route metric is 11, traffic share count is 1

R4#sh ip route 10.1.1.1
Routing entry for 10.1.1.1/32
  Known via "ospf 1", distance 110, metric 60001, type intra area
  Last update from 10.1.14.1 on Ethernet0/0, 01:15:28 ago
  Routing Descriptor Blocks:
  * 10.1.14.1, from 10.1.1.1, 01:15:28 ago, via Ethernet0/0
      Route metric is 60001, traffic share count is 1

R1#sh ip route 10.1.3.3
Routing entry for 10.1.3.3/32
  Known via "ospf 1", distance 110, metric 1001, type intra area
  Last update from 11.1.1.3 on Tunnel23, 01:28:19 ago
  Routing Descriptor Blocks:
  * 11.1.1.3, from 10.1.3.3, 01:28:19 ago, via Tunnel23
      Route metric is 1001, traffic share count is 1
```

The total cost is 11+60001+1001= 61013

Intra-area routes should be preferred over inter-area routes by the OSPF process. However, capability transit is enabled by default on Cisco IOS. The destination 10.1.3.3 has a lower OSPF cost through a transit non-backbone area 45 and OSPF process will prefer this route even if there is a virtual-link in area 0 between R4 and R1. In order to change this behavior, we have to disable capability transit.

On R1 and R4, configure the following:

```
router ospf 1
no capability transit
```

The traceroute will now follow the path through the virtual link:

```
R7#traceroute 10.1.3.3 source 10.1.7.7
Type escape sequence to abort.
Tracing the route to 10.1.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.47.4 1 msec 1 msec 0 msec
  2 10.1.14.1 1 msec 0 msec 1 msec
  3 11.1.1.3 1 msec *  2 msec
```

**Task 15.14** OSPF should not exchange periodic hellos and periodic refreshes of LSAs over the point-to-point connection between R6 and R9. Configuration can only be applied on R9.

On R9, configure the following:

```
int s3/0
ip ospf demand-circuit
```

Please note that this command has only to be configured on one side of the connection.
The connection will go down and the new parameters will be negotiated.

**Task 15.15** Configure plain-text authentication on the connection between R6 and R9. The key value should be set to “iPexpert”. Make sure that this authentication is enforced even if this is an on-demand circuit.

On R6 and R9, configure the following:

```
router ospf 1
  area 69 authentication

interface s3/0
  ip ospf authentication-key iPexpert
```

Remember that we have suppressed the OSPF hellos by configuring the ip ospf demand-circuit. This means that, once the neighborship is up, no hellos are exchanged. As a result, any changes that you make to the OSPF authentication do not take effect until you clear the OSPF process with the clear ip ospf process command.

**Task 15.16** Configure MD5 authentication only on the connection between R5 and R3. The key value should be set to 2 and the password to” iPexpert2015”. On R5, configure authentication under the routing process.

On R5, configure the following:

```
router ospf 1
  area 0 authentication message-digest
```

```
interface Serial3/0
  ip ospf authentication null

interface Serial4/0
 ip ospf message-digest-key 2 md5 iPexpert2015
```

On R3, configure the following:

```
interface Serial4/0
  ip ospf authentication message-digest
  ip ospf message-digest-key 2 md5 iPexpert2015
```

**Task 15.17** Protect the connection between R5 and R4 with the Null authentication.

We had to configure on R5 the authentication under the routing process, but we don't want to authenticate the connection between R5 and R4. We have therefor to use the ip ospf authentication null command in interface configuration mode.

```
interface Serial3/0
  ip ospf authentication null
```

**Task 15.18** OSPF process is reflooding by default every LSAs every 30 minutes. This should not be necessary for LSAs sent out of the two serial interfaces on R5.

The LSAs that are sent out of the 2 serial interfaces of R5 will have the DonotAge Flag set and therefore will never age out.

On R5, configure the following:

```
int s3/0
ip ospf flood-reduction
int s4/0
ip ospf flood-reduction
```

**Task 15.19** Configure the following loopback on R9:

| | |
|---|---|
| Loopback 8 | 10.8.9.9/16 |
| Loopback 9 | 10.9.9.9/16 |
| Loopback 10 | 10.10.9.9/16 |

On R9, configure the following:

```
int lo8
ip address 10.8.9.9 255.255.0.0
int lo9
ip address 10.9.9.9 255.255.0.0
int lo10
ip address 10.10.9.9 255.255.0.0
```

**Task 15.20** Those 3 loopbacks should be seen in the area 0 routing table as a single summary network. Use internal summary.

Internal summary can only take place on the ABR. In our example, the ABR is the router R6. We have first to advertise the 3 loopbacks into OSPF using network statements.

On R6, configure the following:

```
router ospf 1
area 69 range 10.8.0.0 255.252.0.0
```

On R9, configure the following:

```
router ospf 1
network 10.8.9.9 255.255.0.0 area 69
network 10.9.9.9 255.255.0.0 area 69
network 10.10.9.9 255.255.0.0 area 69
```

**Task 15.21** On R6, ensure that the summary route created in Task 15.20 is not present in the routing table pointing to Null0.

Let's have a look at the routing table of R6:

```
R6#sh ip route
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        10.1.1.1/32 [110/2001] via 36.0.0.3, 00:00:07, Tunnel1
O        10.1.2.2/32 [110/2001] via 36.0.0.3, 00:00:07, Tunnel1
O        10.1.3.3/32 [110/1001] via 36.0.0.3, 00:00:07, Tunnel1
O IA     10.1.4.4/32 [110/62001] via 36.0.0.3, 00:00:07, Tunnel1
O IA     10.1.5.5/32 [110/1065] via 36.0.0.3, 00:00:07, Tunnel1
C        10.1.6.0/24 is directly connected, Loopback0
L        10.1.6.6/32 is directly connected, Loopback0
O IA     10.1.7.7/32 [110/62011] via 36.0.0.3, 00:00:07, Tunnel1
O        10.1.9.9/32 [110/65] via 10.1.69.9, 00:00:07, Serial3/0
O IA     10.1.14.0/24 [110/62000] via 36.0.0.3, 00:00:07, Tunnel1
O        10.1.35.0/24 [110/1064] via 36.0.0.3, 00:00:07, Tunnel1
O IA     10.1.45.0/24 [110/1128] via 36.0.0.3, 00:00:07, Tunnel1
O IA     10.1.47.0/24 [110/62010] via 36.0.0.3, 00:00:07, Tunnel1
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.1.236.0/24 is directly connected, Ethernet0/0
L        10.1.236.6/32 is directly connected, Ethernet0/0
O        10.8.0.0/14 is a summary, 00:00:07, Null0
O        10.8.9.9/32 [110/65] via 10.1.69.9, 00:00:07, Serial3/0
O        10.9.9.9/32 [110/65] via 10.1.69.9, 00:00:07, Serial3/0
O        10.10.9.9/32 [110/65] via 10.1.69.9, 00:00:07, Serial3/0
C        10.11.6.0/24 is directly connected, Loopback1
L        10.11.6.6/32 is directly connected, Loopback1
      11.0.0.0/24 is subnetted, 1 subnets
O        11.1.1.0 [110/2000] via 36.0.0.3, 00:00:07, Tunnel1
      36.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        36.0.0.0/24 is directly connected, Tunnel1
L        36.0.0.6/32 is directly connected, Tunnel1
```

By default, on the ABR router where the summary is configured, there is a route to the summary address pointing to null0. We have been asked to get rid of this route.

On R6, configure the following:

```
router ospf 1
no discard-route internal
```

The route to null0 is not present anymore:

```
R6#sh ip route
```

```
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, 22 subnets, 2 masks
O        10.1.1.1/32 [110/2001] via 36.0.0.3, 00:01:51, Tunnel1
O        10.1.2.2/32 [110/2001] via 36.0.0.3, 00:01:51, Tunnel1
O        10.1.3.3/32 [110/1001] via 36.0.0.3, 00:01:51, Tunnel1
O IA     10.1.4.4/32 [110/62001] via 36.0.0.3, 00:01:51, Tunnel1
O IA     10.1.5.5/32 [110/1065] via 36.0.0.3, 00:01:51, Tunnel1
C        10.1.6.0/24 is directly connected, Loopback0
L        10.1.6.6/32 is directly connected, Loopback0
O IA     10.1.7.7/32 [110/62011] via 36.0.0.3, 00:01:51, Tunnel1
O        10.1.9.9/32 [110/65] via 10.1.69.9, 00:01:51, Serial3/0
O IA     10.1.14.0/24 [110/62000] via 36.0.0.3, 00:01:51, Tunnel1
O        10.1.35.0/24 [110/1064] via 36.0.0.3, 00:01:51, Tunnel1
O IA     10.1.45.0/24 [110/1128] via 36.0.0.3, 00:01:51, Tunnel1
O IA     10.1.47.0/24 [110/62010] via 36.0.0.3, 00:01:51, Tunnel1
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.1.236.0/24 is directly connected, Ethernet0/0
L        10.1.236.6/32 is directly connected, Ethernet0/0
O        10.8.9.9/32 [110/65] via 10.1.69.9, 00:01:51, Serial3/0
O        10.9.9.9/32 [110/65] via 10.1.69.9, 00:01:51, Serial3/0
O        10.10.9.9/32 [110/65] via 10.1.69.9, 00:01:51, Serial3/0
C        10.11.6.0/24 is directly connected, Loopback1
L        10.11.6.6/32 is directly connected, Loopback1
      11.0.0.0/24 is subnetted, 1 subnets
O        11.1.1.0 [110/2000] via 36.0.0.3, 00:01:51, Tunnel1
      36.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        36.0.0.0/24 is directly connected, Tunnel1
L        36.0.0.6/32 is directly connected, Tunnel1
```

**Task 15.22** On R9, redistribute the pre-configured routes into OSPF and make sure that they appear as one routing entry in the routing table in all other OSPF routers.

On R9, configure the following:

```
router ospf 1
redistribute static subnets
summary-address  172.16.128.0 255.255.252.0
```

Let's check the routing table of R6. We can see that only the summary route has been advertised.

```
R6#sh ip route
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, 22 subnets, 2 masks
O        10.1.1.1/32 [110/2001] via 36.0.0.3, 00:08:27, Tunnel1
O        10.1.2.2/32 [110/2001] via 36.0.0.3, 00:08:27, Tunnel1
O        10.1.3.3/32 [110/1001] via 36.0.0.3, 00:08:27, Tunnel1
O IA     10.1.4.4/32 [110/62001] via 36.0.0.3, 00:08:27, Tunnel1
O IA     10.1.5.5/32 [110/1065] via 36.0.0.3, 00:08:27, Tunnel1
C        10.1.6.0/24 is directly connected, Loopback0
L        10.1.6.6/32 is directly connected, Loopback0
O IA     10.1.7.7/32 [110/62011] via 36.0.0.3, 00:08:27, Tunnel1
O        10.1.9.9/32 [110/65] via 10.1.69.9, 00:08:27, Serial3/0
O IA     10.1.14.0/24 [110/62000] via 36.0.0.3, 00:08:27, Tunnel1
O        10.1.35.0/24 [110/1064] via 36.0.0.3, 00:08:27, Tunnel1
O IA     10.1.45.0/24 [110/1128] via 36.0.0.3, 00:08:27, Tunnel1
O IA     10.1.47.0/24 [110/62010] via 36.0.0.3, 00:08:27, Tunnel1
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.1.236.0/24 is directly connected, Ethernet0/0
L        10.1.236.6/32 is directly connected, Ethernet0/0
O        10.8.9.9/32 [110/65] via 10.1.69.9, 00:08:27, Serial3/0
O        10.9.9.9/32 [110/65] via 10.1.69.9, 00:08:27, Serial3/0
O        10.10.9.9/32 [110/65] via 10.1.69.9, 00:08:27, Serial3/0
C        10.11.6.0/24 is directly connected, Loopback1
L        10.11.6.6/32 is directly connected, Loopback1
      11.0.0.0/24 is subnetted, 1 subnets
O        11.1.1.0 [110/2000] via 36.0.0.3, 00:08:27, Tunnel1
      36.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        36.0.0.0/24 is directly connected, Tunnel1
L        36.0.0.6/32 is directly connected, Tunnel1
      172.16.0.0/22 is subnetted, 1 subnets
O E2     172.16.128.0 [110/20] via 10.1.69.9, 00:01:23, Serial3/0
```

**Task 15.23** Configure area 45 in a way that LSAs never age out in this area.

On R4, configure the following:

```
int s3/0
ip ospf flood-reduction
```

**You have completed Lab 15**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 16: Configure and troubleshoot OSPF (Part 3)

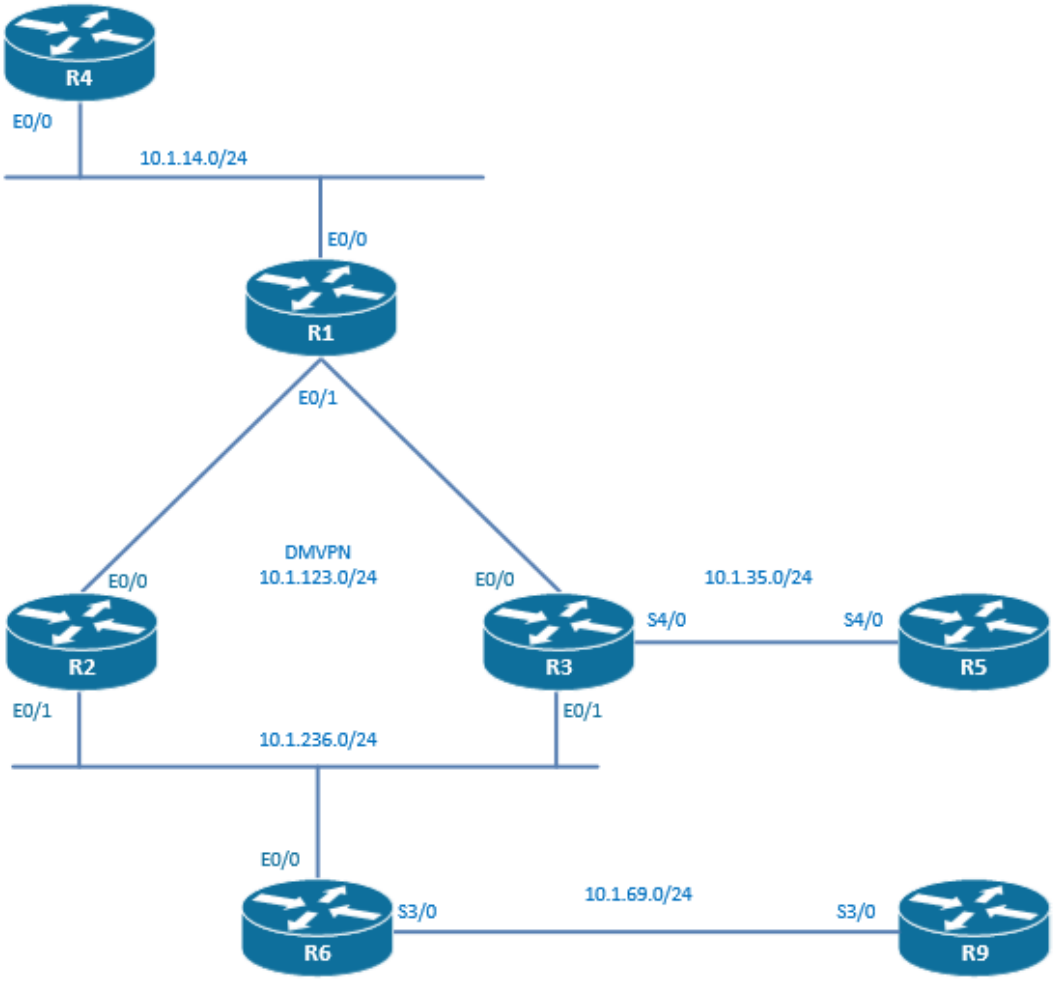
## Technologies covered

- Stub area
- Totally not so stubby area
- NSSA
- NSSA type 5 to type 7 translation
- LSA filtering
- FA Suppression
- Reliable conditional default routing

## Overview

You have been tasked to configure OSPF as the routing protocol of your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 16.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Get OSPF routing up and routing with process 1 area 0 in this DMVPN network. The election of a DR should not take place in this network. Do not modify any OSPF timers.

The DMVPN phase 2 networks with multicast support have already been pre-configured. No DR should be elected means no NBMA or broadcast network types should be configured. We are going to configure therefore network type point-to-multipoint on the spokes and on the hub.

On R1, R2, and R3, configure the following:

```
router ospf 1
network 11.1.1.0 0.0.0.255 area 0

int tu23
ip ospf network point-to-multipoint
```

OSPF neighborships have been established and no DR has been elected.

```
R1#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.3.3          0   FULL/  -        00:01:54    11.1.1.3        Tunnel23
10.1.2.2          0   FULL/  -        00:01:52    11.1.1.2        Tunnel23
```

**Task 16.2** Add the loopback0 of R1, R2, and R3 into the area 0 process as network statements.

On R1, configure the following:

```
router ospf 1
network 10.1.1.1 0.0.0.0 area 0
```

On R2, configure the following:

```
router ospf 1
network 10.1.2.2 0.0.0.0 area 0
```

On R3, configure the following:

```
router ospf 1
network 10.1.3.3 0.0.0.0 area 0
```

I can ping from spoke R3 to spoke R2.

```
R3#ping 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 16.3** On R2, R3, and R6, configure the network 10.1.236.0/24 as part of OSPF area 236. Add the loopback0 of R6 into the area 236 process as a network statement.

On R2 and on R3, configure the following:

```
router ospf 1
network 10.1.236.0 0.0.0.255 area 236
```

On R6, configure the following:

```
router ospf 1
network 10.1.236.0 0.0.0.255 area 236
network 10.1.6.6 0.0.0.0 area 236
```

On R6, I can ping the loopback0 of R1 across the area 236.

```
R6#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 16.4** In R6 routing-table, the only IA OSPF-learned route should be a default route with the ABRs as the next-hop.

Let's check the current state of the OSPF database:

```
R6#sh ip ospf database

            OSPF Router with ID (10.11.6.6) (Process ID 1)

                Router Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.2.2        10.1.2.2        646         0x80000002 0x00D73A 1
10.1.3.3        10.1.3.3        646         0x80000002 0x00C349 1
10.11.6.6       10.11.6.6       645         0x80000002 0x003A88 2

                Net Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.236.6      10.11.6.6       645         0x80000001 0x007F56

                Summary Net Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.1.1        10.1.2.2        703         0x80000001 0x00A78C
10.1.1.1        10.1.3.3        693         0x80000002 0x009898
10.1.2.2        10.1.2.2        703         0x80000001 0x005EBF
10.1.2.2        10.1.3.3        693         0x80000002 0x00B68C
10.1.3.3        10.1.2.2        693         0x80000001 0x00B093
10.1.3.3        10.1.3.3        693         0x80000002 0x003ADE
11.1.1.1        10.1.2.2        703         0x80000001 0x0090A3
11.1.1.1        10.1.3.3        693         0x80000002 0x0081AF
11.1.1.2        10.1.2.2        703         0x80000001 0x0052CC
11.1.1.2        10.1.3.3        693         0x80000002 0x00AA99
11.1.1.3        10.1.2.2        693         0x80000001 0x00AF96
11.1.1.3        10.1.3.3        693         0x80000002 0x0039E1
```

There is currently LSA 1, LSA 2, and LSA 3 in the OSPF database.

We have to configure the routers so that area 236 does not receive the LSAs 3. Actually, by configuring area 236 as a totally stubby area, the future LSAs 4 and LSAs 5 will also be filtered out.

On R2 and R3, configure the following:

```
router ospf 1
  area 236 stub no-summary
```

On R6, configure the following:

```
router ospf 1
  area 236 stub
```

The area 236 is now a totally stub area. The only LSA 3 in the ospf database of R6 is a default route.

```
R6#sh ip ospf database

            OSPF Router with ID (10.11.6.6) (Process ID 1)

                Router Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.2.2        10.1.2.2        18          0x80000005 0x007092 2
10.1.3.3        10.1.3.3        16          0x80000004 0x007885 2
10.11.6.6       10.11.6.6       14          0x80000004 0x00546E 2

                Net Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.236.3      10.1.3.3        35          0x80000001 0x00917E
10.1.236.6      10.11.6.6       14          0x80000003 0x00993C
                Summary Net Link States (Area 236)
Link ID         ADV Router      Age         Seq#       Checksum
0.0.0.0         10.1.2.2        39          0x80000001 0x0035F9
0.0.0.0         10.1.3.3        36          0x80000001 0x002805

R6# sh ip route
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 10.1.236.3 to network 0.0.0.0

O*IA  0.0.0.0/0 [110/11] via 10.1.236.3, 00:02:02, Ethernet0/0
                [110/11] via 10.1.236.2, 00:02:02, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
C        10.1.6.0/24 is directly connected, Loopback0
L        10.1.6.6/32 is directly connected, Loopback0
C        10.1.236.0/24 is directly connected, Ethernet0/0
L        10.1.236.6/32 is directly connected, Ethernet0/0
C        10.11.6.0/24 is directly connected, Loopback1
L        10.11.6.6/32 is directly connected, Loopback1
```

**Task 16.5** On R6 and R9, configure static routing to ensure the reachability of the loopback0, loopback1, and loopback2 network of R9.

On R6, configure the following:

```
ip route 10.1.9.9 255.255.255.255 10.1.69.9
ip route 10.11.9.9 255.255.255.255 10.1.69.9
ip route 10.22.9.9 255.255.255.255 10.1.69.9
```

On R9, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.69.6
```

**Task 16.6** On R6, redistribute the static routes configured in Task 16.5 (except loopback2) into OSPF. In the routing-table of R1, 10.1.9.0/24 should show as E1 and 10.11.9.0/24 should show as E2. On R1, ensure that you can ping the loopback0 and loopback1 of R9 from the loopback0 of R1 as a source.

We redistribute into OSPF the static routes configured earlier. We apply a route-map that is assigning a different metric-type to each network. A route-map has an explicit deny for networks that has not been matched in any route-map rules, so the network 10.22.9.9/32 will not be redistributed.

```
On R6, configure the following:
access-list 1 permit host 10.1.9.9
access-list 2 permit host 10.11.9.9

route-map LOOPBACK_R9 permit 10
 match ip address 1
 set metric-type type-1

route-map LOOPBACK_R9 permit 20
 match ip address 2
 set metric-type type-2

router ospf 1
 redistribute static subnets route-map LOOPBACK_R9
```

When configuring the redistribution, we are getting the following message.

```
%OSPF-4-ASBR_WITHOUT_VALID_AREA: Router is currently an ASBR while having only one area
which is a stub area
```

This is actually due to the fact that the area 236 is a totally not stubby area and LSA 5 is not supported within this type of area. In order to have redistribution support in a totally stub area; we have to configure this area as a totally not so stubby area.

On R2, R3, and R6, configure the following:

```
router ospf 1
no area 236 stub
area 236 nssa no-summary
```

On R6, configure the following:

```
router ospf 1
area 236 nssa
```

Please note that you have to unconfigure area 236 stub first before being able to configure area 236 nssa. You get the following message otherwise and the configuration is not modified.

```
% OSPF: Area is configured as stub area already
```

The redistributed routes are present in our stub area as LSA 7.

```
R6#sh ip ospf database

            OSPF Router with ID (10.11.6.6) (Process ID 1)

                Router Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.2.2        10.1.2.2        1835        0x80000014 0x005F98 1
10.1.3.3        10.1.3.3        1790        0x80000012 0x004FA5 1
10.11.6.6       10.11.6.6       1585        0x80000011 0x00C7E3 2

                Net Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.236.6      10.11.6.6       85          0x80000011 0x0005BA

                Summary Net Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum
0.0.0.0         10.1.2.2        259         0x80000001 0x00BC6A
0.0.0.0         10.1.3.3        255         0x80000001 0x00AF75

                Type-7 AS External Link States (Area 236)

Link ID         ADV Router      Age         Seq#       Checksum Tag
```

```
10.1.9.9        10.11.6.6       85          0x8000000C 0x00892D 0
10.11.9.9       10.11.6.6       85          0x8000000C 0x009497 0
```

The redistributed routes are present in area 0 as LSA 5. The ABR is converting the LSAs 7 into LSAs 5.

```
R1#sh ip ospf database

            OSPF Router with ID (10.1.1.1) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.1.1        10.1.1.1        1717        0x80000005 0x009B1D 4
10.1.2.2        10.1.2.2        85          0x80000007 0x00E3E2 3
10.1.3.3        10.1.3.3        35          0x80000007 0x0011AD 3
10.11.6.6       10.11.6.6       2970        0x80000003 0x00F5CE 2

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.6.6        10.1.2.2        70          0x80000001 0x006E9D
10.1.6.6        10.1.3.3        25          0x80000001 0x0061A8
10.1.236.0      10.1.2.2        959         0x80000002 0x00B278
10.1.236.0      10.1.3.3        930         0x80000002 0x00A583

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.1.9.9        10.1.3.3        24          0x80000001 0x00AB30 0
10.11.9.9       10.1.3.3        24          0x80000001 0x00B69A 0
```

From R1, the loopback0 and loopback1 of R9 are pingable.

```
R1#ping 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms

R1#ping 10.11.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.11.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/8/10 ms
```

Please note that as we migrate area 236 from a totally stubby area to a totally not so stubby area, the redistributed static route will appear on R2 as N1, and N2 type instead of E1 and E2 type.

```
R2#sh ip route
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, 12 subnets, 2 masks
O        10.1.1.1/32 [110/1001] via 11.1.1.1, 00:50:21, Tunnel23
C        10.1.2.2/32 is directly connected, Loopback0
O        10.1.3.3/32 [110/2001] via 11.1.1.1, 00:50:21, Tunnel23
O        10.1.6.6/32 [110/11] via 10.1.236.6, 00:04:06, Ethernet0/1
O N1     10.1.9.9/32 [110/31] via 10.1.236.6, 00:04:06, Ethernet0/1
C        10.1.25.0/24 is directly connected, Serial5/0
L        10.1.25.1/32 is directly connected, Serial5/0
```

```
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.2/32 is directly connected, Ethernet0/0
C        10.1.236.0/24 is directly connected, Ethernet0/1
L        10.1.236.2/32 is directly connected, Ethernet0/1
O N2     10.11.9.9/32 [110/20] via 10.1.236.6, 00:04:06, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
O        11.1.1.1/32 [110/1000] via 11.1.1.1, 00:50:21, Tunnel23
L        11.1.1.2/32 is directly connected, Tunnel23
O        11.1.1.3/32 [110/2000] via 11.1.1.1, 00:50:21, Tunnel23
```

**Task 16.7** Area 236 is a totally Not-so-stub area having two ABRs to area 0. By manipulating OSPF cost, ensure that the default route in R6 routing table is using R3 as a next hop. The cost of the default route to R2 should be modified and this cost should be the default cost + 1.

On R6, the default route is load-balancing the traffic between R2 and R3.

```
R6#sh ip route
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 10.1.236.3 to network 0.0.0.0

O*IA  0.0.0.0/0 [110/11] via 10.1.236.3, 00:02:15, Ethernet0/0
                [110/11] via 10.1.236.2, 00:02:15, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 9 subnets, 2 masks
C        10.1.6.6/32 is directly connected, Loopback0
S        10.1.9.9/32 [1/0] via 10.1.69.9
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.1.236.0/24 is directly connected, Ethernet0/0
L        10.1.236.6/32 is directly connected, Ethernet0/0
C        10.11.6.6/32 is directly connected, Loopback1
S        10.11.9.9/32 [1/0] via 10.1.69.9
S        10.22.9.9/32 [1/0] via 10.1.69.9
```

The metric of the default route on R6 is 11. The OSPF cost of the interface e0/0 of R6 is 10. The default cost of the default route advertised on the ABRs into the NSSA area is 1, hence the total of 11.

The cost of the default route advertised into a NSSA area from an ABR is controlled with the area number default-cost cost router configuration command. As we have to use the default cost + 1 on R2, we are going to configure a default-cost of 2.

On R2, configure the following:

```
router ospf 1
area 236 default-cost 2
```

The result is as expected. The default-route prefers R3 over R2.

```
R6#sh ip route
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 10.1.236.3 to network 0.0.0.0

O*IA  0.0.0.0/0 [110/11] via 10.1.236.3, 00:24:44, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 9 subnets, 2 masks
C        10.1.6.6/32 is directly connected, Loopback0
S        10.1.9.9/32 [1/0] via 10.1.69.9
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.1.236.0/24 is directly connected, Ethernet0/0
L        10.1.236.6/32 is directly connected, Ethernet0/0
C        10.11.6.6/32 is directly connected, Loopback1
S        10.11.9.9/32 [1/0] via 10.1.69.9
S        10.22.9.9/32 [1/0] via 10.1.69.9
```

On R6, there are still the two LSAs 3 for the network 0.0.0.0 in the OSPF database, one with a cost of 1, the other with a cost of 2. The one with a cost of 1 is appearing in the routing table.

```
R6#sh ip ospf database summary 0.0.0.0

            OSPF Router with ID (10.11.6.6) (Process ID 1)

                Summary Net Link States (Area 236)

  LS age: 164
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 0.0.0.0 (summary Network Number)
  Advertising Router: 10.1.2.2
  LS Seq Number: 80000006
  Checksum: 0xBC64
  Length: 28
  Network Mask: /0
        MTID: 0         Metric: 2

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 356
  Options: (No TOS-capability, DC, Upward)
  LS Type: Summary Links(Network)
  Link State ID: 0.0.0.0 (summary Network Number)
  Advertising Router: 10.1.3.3
  LS Seq Number: 80000003
  Checksum: 0xAB77
  Length: 28
  Network Mask: /0
        MTID: 0         Metric: 1
```

**Task 16.8** On R1 and R4, configure the network 10.1.14.0/24 as part of OSPF area 14. Add the loopback0 of R4 into the area 14 process as a network statement.

On R1, configure the following:

```
router ospf 1
network 10.1.14.0 0.0.0.255 area 14
```

On R4, configure the following:

```
router ospf 1
network 10.1.14.0 0.0.0.255 area 14
network 10.1.4.4 0.0.0.0 area 14
```

**Task 16.9** Configure Area 14 in a way that it does not receive any LSA 5 updates. Ensure full reachability and test that you can ping from R4 the loopback 0 of R9 from the loopback 0 of R4 as a source.

The LSAs 5 should be filtered on the ABR. A stub area has to be configured.

On R1 and R4, configure the following:

```
router ospf 1
area 14 stub

The network 10.1.9.9 is not in the routing table because it is a redistibuted route and LSA
5 is filtered on the ABR R1. I can ping from R4 the loopback 0 of R9 from the loopback 0 of
R4 as a source using the default route.
R4#sh ip route
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 10.1.14.1 to network 0.0.0.0

O*IA  0.0.0.0/0 [110/11] via 10.1.14.1, 00:01:04, Ethernet0/1
      10.0.0.0/8 is variably subnetted, 8 subnets, 2 masks
O IA     10.1.1.1/32 [110/11] via 10.1.14.1, 00:01:04, Ethernet0/1
O IA     10.1.2.2/32 [110/1011] via 10.1.14.1, 00:01:04, Ethernet0/1
O IA     10.1.3.3/32 [110/1011] via 10.1.14.1, 00:01:04, Ethernet0/1
C        10.1.4.4/32 is directly connected, Loopback0
O IA     10.1.6.6/32 [110/1021] via 10.1.14.1, 00:01:04, Ethernet0/1
C        10.1.14.0/24 is directly connected, Ethernet0/1
L        10.1.14.4/32 is directly connected, Ethernet0/1
O IA     10.1.236.0/24 [110/1020] via 10.1.14.1, 00:01:04, Ethernet0/1
      11.0.0.0/32 is subnetted, 3 subnets
O IA     11.1.1.1 [110/10] via 10.1.14.1, 00:01:04, Ethernet0/1
O IA     11.1.1.2 [110/1010] via 10.1.14.1, 00:01:04, Ethernet0/1
O IA     11.1.1.3 [110/1010] via 10.1.14.1, 00:01:04, Ethernet0/1

R4#ping 10.1.9.9 source 10.1.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.4.4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms
```

**Task 16.10** Area 35 is a totally NSSA area. On R3 and R5, configure the network 10.1.35.0/24 as part of OSPF area 35. Inject the loopback0 of R5 into the area 35 process as a network statement.

On R3, configure the following:

```
router ospf 1
network 10.1.35.0 0.0.0.255 area 35
area 35 nssa no-summary
```

On R5, configure the following:

```
router ospf 1
network 10.1.35.0 0.0.0.255 area 35
network 10.1.5.5 0.0.0.0 area 35
area 35 nssa
```

**Task 16.11** Redistribute the loopback1, loopback2, loopback3, and loopback4 of R5 into the area 35 each as a N2 route and each with a metric of 55. Make sure that on R5, you can ping to the loopback 0 of R9 with the ping sourcing from the loopback 4 of R5.

On R5, configure the following:

```
route-map LOOPBACKS permit 10
 match interface Loopback1 Loopback2 Loopback3 Loopback4
 set metric 55
 set metric-type type-2

router ospf 1
 redistribute connected subnets route-map LOOPBACKS
```

On R3, the loopbacks of R5 are present in the routing table as N2 and with a metric of 55, because a route with metric-type of type 2 keeps the cost assigned on the ASBR. The cost of this route is not incremented with the cost between the destination and the ASBR when transiting through the OSPF network.

```
R3#sh ip route
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, 19 subnets, 2 masks
O        10.1.1.1/32 [110/1001] via 11.1.1.1, 00:13:47, Tunnel23
O        10.1.2.2/32 [110/2001] via 11.1.1.1, 00:13:47, Tunnel23
C        10.1.3.3/32 is directly connected, Loopback0
O IA     10.1.4.4/32 [110/1011] via 11.1.1.1, 00:13:47, Tunnel23
O        10.1.5.5/32 [110/65] via 10.1.35.5, 00:13:22, Serial4/0
O        10.1.6.6/32 [110/11] via 10.1.236.6, 00:13:47, Ethernet0/1
O N1     10.1.9.9/32 [110/31] via 10.1.236.6, 00:13:47, Ethernet0/1
O IA     10.1.14.0/24 [110/1010] via 11.1.1.1, 00:13:47, Tunnel23
C        10.1.35.0/24 is directly connected, Serial4/0
L        10.1.35.3/32 is directly connected, Serial4/0
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
C        10.1.236.0/24 is directly connected, Ethernet0/1
L        10.1.236.3/32 is directly connected, Ethernet0/1
O N2     10.11.5.0/24 [110/55] via 10.1.35.5, 00:02:10, Serial4/0
O N2     10.11.9.9/32 [110/20] via 10.1.236.6, 00:13:47, Ethernet0/1
O N2     10.21.5.0/24 [110/55] via 10.1.35.5, 00:02:09, Serial4/0
O N2     10.31.5.0/24 [110/55] via 10.1.35.5, 00:02:09, Serial4/0
O N2     10.41.5.0/24 [110/55] via 10.1.35.5, 00:02:09, Serial4/0
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
O        11.1.1.1/32 [110/1000] via 11.1.1.1, 00:13:47, Tunnel23
O        11.1.1.2/32 [110/2000] via 11.1.1.1, 00:13:47, Tunnel23
L        11.1.1.3/32 is directly connected, Tunnel23
```

I can ping to the loopback 0 of R9 with the ping sourcing from the loopback 4 of R5.

```
R5#ping 10.1.9.9 source 10.41.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.41.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/16/17 ms
```

**Task 16.12** Block the LSA 7 to LSA 5 translation for the network 10.11.5.0/24 using a summary-address command.

The type-7 LSAs are translated into type-5 LSA on the ABRs. Those type-5 ABRs are originated on the ABRs so the ABR is the ASBR for those translated type-7 LSAs.

On R3, configure the following:

```
router ospf 1
summary-address 10.11.5.5 255.255.255.0 not-advertise
```

In the routing table of R1, the network 10.11.5.0 has indeed disappeared.

```
R1#sh ip route
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

S*    0.0.0.0/0 is directly connected, Null0
      10.0.0.0/8 is variably subnetted, 17 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
O        10.1.2.2/32 [110/1001] via 11.1.1.2, 00:56:05, Tunnel23
O        10.1.3.3/32 [110/1001] via 11.1.1.3, 00:56:05, Tunnel23
O        10.1.4.4/32 [110/11] via 10.1.14.4, 00:55:45, Ethernet0/0
O IA     10.1.5.5/32 [110/1065] via 11.1.1.3, 00:42:11, Tunnel23
O IA     10.1.6.6/32 [110/1011] via 11.1.1.3, 00:56:05, Tunnel23
                     [110/1011] via 11.1.1.2, 00:56:05, Tunnel23
O E1     10.1.9.9/32 [110/1031] via 11.1.1.3, 00:56:05, Tunnel23
                     [110/1031] via 11.1.1.2, 00:56:05, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
O IA     10.1.35.0/24 [110/1064] via 11.1.1.3, 00:49:00, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
O IA     10.1.236.0/24 [110/1010] via 11.1.1.3, 00:56:05, Tunnel23
                       [110/1010] via 11.1.1.2, 00:56:05, Tunnel23
O E2     10.11.9.9/32 [110/20] via 11.1.1.3, 00:56:05, Tunnel23
                      [110/20] via 11.1.1.2, 00:56:05, Tunnel23
O E2     10.21.5.0/24 [110/55] via 11.1.1.3, 00:30:59, Tunnel23
O E2     10.31.5.0/24 [110/55] via 11.1.1.3, 00:30:59, Tunnel23
O E2     10.41.5.0/24 [110/55] via 11.1.1.3, 00:30:59, Tunnel23
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
O        11.1.1.2/32 [110/1000] via 11.1.1.2, 00:56:05, Tunnel23
O        11.1.1.3/32 [110/1000] via 11.1.1.3, 00:56:05, Tunnel23
```

**Task 16.13** Filter the forwarding address for the type-5 LSAs originated at R5 using the area 35 range no-advertise in command on the ABR.

Let's take the example of the 10.41.5.0/24 network. The following explanation also applies to the other loopbacks redistributed on R5.

This network 10.41.5.0/24 is present in the OSPF database as a type-7 LSA in area 35 and a type-5 LSA. When the translation between LSA 7 to LSA 5 takes place, the forward address is retained. This forward address has to be reachable for the LSA to be used in the OSPF route computation.

```
R3# sh ip ospf database nssa-external

            OSPF Router with ID (10.1.3.3) (Process ID 1)

                Type-7 AS External Link States (Area 35)

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 135
  Options: (No TOS-capability, Type 7/5 translation, DC, Upward)
  LS Type: AS External Link
  Link State ID: 10.41.5.0 (External Network Number )
  Advertising Router: 10.41.5.5
  LS Seq Number: 80000005
  Checksum: 0x11E7
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 55
        Forward Address: 10.1.5.5
        External Route Tag: 0

R3# sh ip ospf database external

            OSPF Router with ID (10.1.3.3) (Process ID 1)

                Type-5 AS External Link States

  LS age: 164
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 10.41.5.0 (External Network Number )
  Advertising Router: 10.1.3.3
  LS Seq Number: 80000003
  Checksum: 0x1409
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 55
        Forward Address: 10.1.5.5
        External Route Tag: 0

At this moment, I can ping 10.41.5.5 from R1.

R1#   ping 10.41.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.41.5.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

Let’s filter the forward address 10.1.5.5 on R3.

On R3, configure the following:

```
router ospf 1
area 35 range 10.1.5.5 255.255.255.255 not-advertise
```

After applying this command, I cannot ping anymore 10.41.5.5 from R1. This is due to the fact that the forward address of the LSA 5 has become unreachable for the rest of the network outside area 35. Please note that the LSAs is contained in the OSPF database on R1 but is not installing the route because the 10.1.5.5 forward address is unreachable.

```
R1#sh ip ospf database external

            OSPF Router with ID (10.1.1.1) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 180
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 10.41.5.0 (External Network Number )
  Advertising Router: 10.1.3.3
  LS Seq Number: 80000006
  Checksum: 0xE0C
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 55
        Forward Address: 10.1.5.5
        External Route Tag: 0
```

**Task 16.14** Instruct R3 to become the forwarding address itself and check that the IP address reachability is restored, that is to say check that you can ping to the loopback0 of R9 with the ping using as a source the loopback4 of R5.

On R3, configure the following:

```
router ospf 1
area 35 nssa translate type7 suppress-fa
```

The IP connectivity has been restored and the ping is working again.

```
R1#   ping 10.41.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.41.5.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

As we can see in the output below, the Forward address has been suppressed by the ABR R3. That's the reason why IP connectivity is again re-established.

```
R1#sh ip ospf database external

            OSPF Router with ID (10.1.1.1) (Process ID 1)

                Type-5 AS External Link States

  Routing Bit Set on this LSA in topology Base with MTID 0
  LS age: 1701
  Options: (No TOS-capability, DC, Upward)
  LS Type: AS External Link
  Link State ID: 10.41.5.0 (External Network Number )
  Advertising Router: 10.1.3.3
  LS Seq Number: 80000007
  Checksum: 0xA24
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        MTID: 0
        Metric: 55
        Forward Address: 0.0.0.0
        External Route Tag: 0
```

**Task 16.15** On R1, there is a default route pre-configured. This default route should be redistributed into OSPF only if the network 10.21.5.0/24 is present in the routing table of R1. Use IP SLA to track, in a reliable way, this network.

On R1, configure the following:

```
ip sla 1
icmp-echo 10.21.5.5 source-ip 10.1.1.1

ip sla schedule 1 life forever start-time now

track 100 ip sla 1

ip route 10.0.0.1 255.255.255.255 Null0 track 100

ip access-list standard FAKE
  permit 10.0.0.1

route-map CONDITION permit 10
 match ip address FAKE
router ospf 1
  default-information originate always route-map CONDITION
```

We have to use a bogus network of 10.0.0.1/32 in order to create a bind between the ip sla and the advertisement of the default route.

The 10.21.5.0/24 network is reachable so the default route originated on R1 is advertised and present in the routing table of R3.

```
R3>sh ip route
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 11.1.1.1 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 11.1.1.1, 00:07:59, Tunnel23
      10.0.0.0/8 is variably subnetted, 19 subnets, 2 masks
O        10.1.1.1/32 [110/1001] via 11.1.1.1, 11:19:06, Tunnel23
O        10.1.2.2/32 [110/2001] via 11.1.1.1, 11:19:06, Tunnel23
C        10.1.3.3/32 is directly connected, Loopback0
O IA     10.1.4.4/32 [110/1011] via 11.1.1.1, 11:19:06, Tunnel23
O        10.1.5.5/32 [110/65] via 10.1.35.5, 11:19:06, Serial4/0
O        10.1.6.6/32 [110/11] via 10.1.236.6, 11:19:06, Ethernet0/1
O N1     10.1.9.9/32 [110/31] via 10.1.236.6, 11:19:06, Ethernet0/1
O IA     10.1.14.0/24 [110/1010] via 11.1.1.1, 11:19:06, Tunnel23
C        10.1.35.0/24 is directly connected, Serial4/0
L        10.1.35.3/32 is directly connected, Serial4/0
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
C        10.1.236.0/24 is directly connected, Ethernet0/1
L        10.1.236.3/32 is directly connected, Ethernet0/1
O N2     10.11.5.0/24 [110/55] via 10.1.35.5, 11:19:06, Serial4/0
O N2     10.11.9.9/32 [110/20] via 10.1.236.6, 11:19:06, Ethernet0/1
O N2     10.21.5.0/24 [110/55] via 10.1.35.5, 11:19:06, Serial4/0
O N2     10.31.5.0/24 [110/55] via 10.1.35.5, 11:19:06, Serial4/0
O N2     10.41.5.0/24 [110/55] via 10.1.35.5, 11:19:06, Serial4/0
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
O        11.1.1.1/32 [110/1000] via 11.1.1.1, 11:19:06, Tunnel23
```

```
O        11.1.1.2/32 [110/2000] via 11.1.1.1, 11:19:06, Tunnel23
L        11.1.1.3/32 is directly connected, Tunnel23
```

On R5, I'm administratively shutting down the loopback2.

```
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#int lo2
R5(config-if)#shut
R5(config-if)#
%LINK-5-CHANGED: Interface Loopback2, changed state to administratively down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback2, changed state to down
```

This is triggering an ip sla state change on the R1 and the withdrawal of the default route on R3.

```
R1#
%TRACK-6-STATE: 100 ip sla 1 state Up -> Down
R1#
```

**You have completed Lab 16**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com

# Lab 17: Configure and troubleshoot OSPF (Part 4)

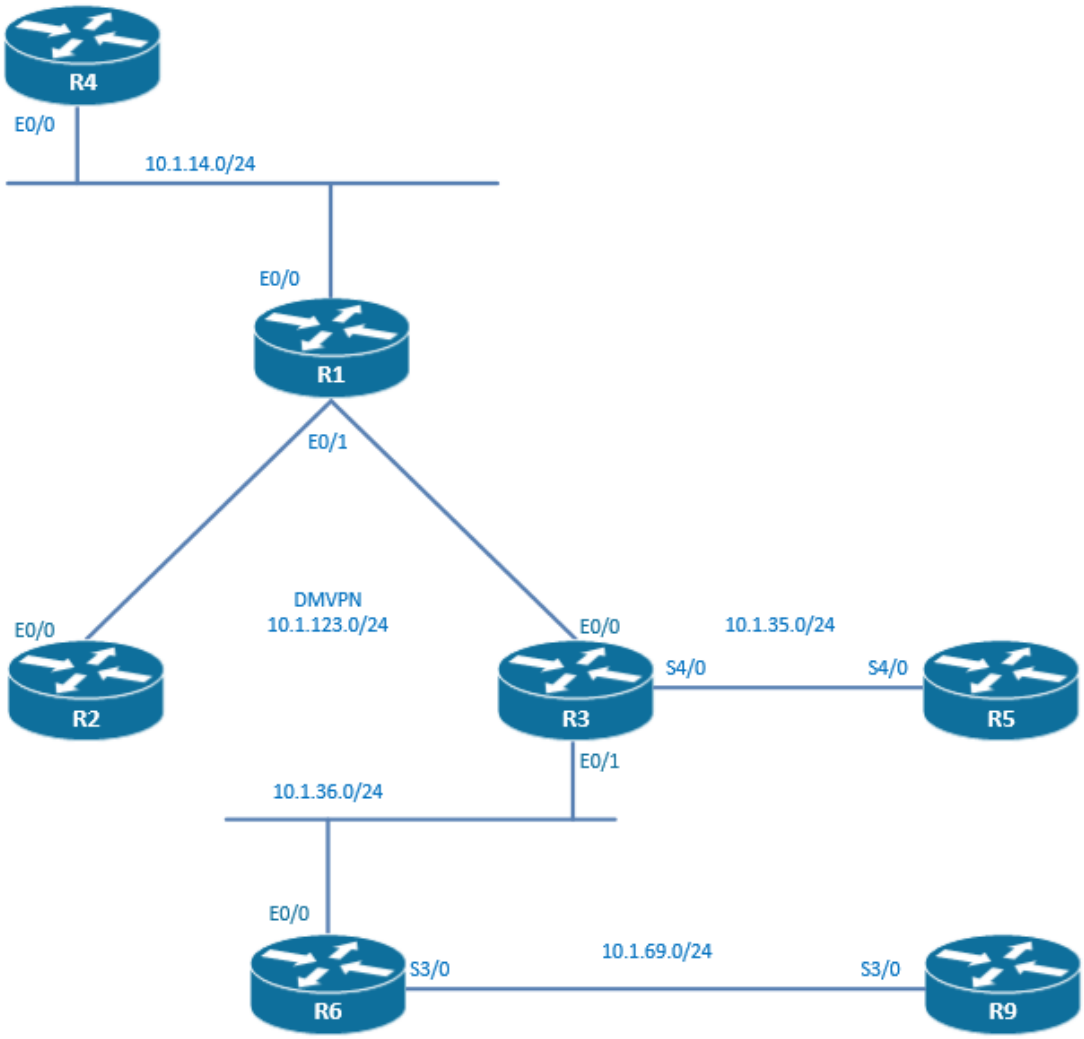
## Technologies covered

- Filtering with distribute-lists
- Filtering with discard-route
- Filtering with administrative distance
- Filtering with route-maps
- NSSA ABR external prefix filtering
- Database filtering
- Stub router advertisement
- OSPF timers optimization
- Resource limiting

## Overview

You have been tasked to configure OSPF as the routing protocol of your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 17.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN is the underlying used technology. Configure OSPF process 1 area 0 in this network. Use point-to-multipoint network type on the hub and the 2 spokes.

The DMVPN phase 2 network with multicast support has already been pre-configured. No DR should be elected means no NBMA or broadcast network types should be configured. We are going to configure therefore network type point-to-multipoint on the spokes and on the hub.

On R1, R2, and R3, configure the following:

```
router ospf 1
network 11.1.1.0 0.0.0.255 area 0

int tu23
ip ospf network point-to-multipoint
```

OSPF neighborships have been established and no DR has been elected.

```
R1#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.1.3.3          0   FULL/  -        00:01:54    11.1.1.3        Tunnel23
10.1.2.2          0   FULL/  -        00:01:52    11.1.1.2        Tunnel23
```

**Task 17.2** Configure the network 10.1.36.0/24 into area 0 on R3 and R6.

On R3 and R6, configure the following:

```
router ospf 1
network 10.1.36.0 0.0.0.255 area 0
```

**Task 17.3** The loopback 0 networks of R1, R2, R3, and R6 should present in the OSPF database of R1 as LSAs type 1.

On R1, configure the following:

```
router ospf 1
network 10.1.1.1 0.0.0.0 area 0
```

On R2, configure the following:

```
router ospf 1
network 10.1.2.2 0.0.0.0 area 0
```

On R3, configure the following:

```
router ospf 1
network 10.1.3.3 0.0.0.0 area 0
```

On R6, configure the following:

```
router ospf 1
network 10.1.6.6 0.0.0.0 area 0
```

The loopback 0 networks of R1, R2, R3 and R6 are present in the OSPF database of R1 as LSAs type 1.

```
R1#sh ip ospf database

            OSPF Router with ID (10.1.1.1) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.1.1        10.1.1.1        263         0x80000003 0x009F1B 4
10.1.2.2        10.1.2.2        256         0x80000002 0x00E4E9 3
10.1.3.3        10.1.3.3        118         0x80000005 0x00CB7A 4
10.1.6.6        10.1.6.6        45          0x80000006 0x005A44 2

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.36.3       10.1.3.3        5           0x80000002 0x007E4F
```

**Task 17.4** On R1, prevent the flooding of link-state advertisements to R2 by using the "database-filter all out" command applied to a neighbor. Make sure that R2 is still having full reachability.

On R1, configure the following:

```
router ospf 1
neighbor 10.1.2.2 database-filter all out
```

By using this command, we can specify neighbors to which we will not send any LSAs. This is useful to limit flooding of LSAs to spokes which are not a transit to any other network, like R2 in our topology. To ensure full reachability, we have to configure a default route on the router R2.

On R2, configure the following:

```
ip route 0.0.0.0 0.0.0.0 tu23
```

**Task 17.5** Configure the network 10.1.69.0/24 into area 69 on R6 and R9. Use network statement to advertise loopback0. Distribute loopback1, loopback2, loopback3, and loopback4 of R9 into the area 69 process as E2 type.

On R6, configure the following:

```
router ospf 1
network 10.1.69.0 0.0.0.255 area 69
```

On R9, configure the following:

```
route-map LOOPBACKS permit 10
 match interface lo1 lo2 lo3 lo4
 set metric-type type-2

router ospf 1
network 10.1.69.0 0.0.0.255 area 69
network 10.1.9.9 0.0.0.0 area 69
redistribute connected subnets route-map LOOPBACKS
```

Let's check the routing table on R1. The loopbacks of R9 are in the routing table with the E2 tag.

```
R1#sh ip route
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, 15 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
O        10.1.2.2/32 [110/1001] via 11.1.1.2, 00:15:18, Tunnel23
O        10.1.3.3/32 [110/1001] via 11.1.1.3, 00:00:11, Tunnel23
O        10.1.6.6/32 [110/1011] via 11.1.1.3, 00:00:11, Tunnel23
O IA     10.1.9.9/32 [110/20] via 11.1.1.3, 00:00:11, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
O        10.1.36.0/24 [110/1010] via 11.1.1.3, 00:00:11, Tunnel23
O IA     10.1.69.0/24 [110/1074] via 11.1.1.3, 00:00:11, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
O E2     10.11.9.9/32 [110/20] via 11.1.1.3, 00:00:11, Tunnel23
O E2     10.21.9.9/32 [110/20] via 11.1.1.3, 00:00:11, Tunnel23
O E2     10.31.9.9/32 [110/20] via 11.1.1.3, 00:00:11, Tunnel23
O E2     10.41.9.9/32 [110/20] via 11.1.1.3, 00:00:11, Tunnel23
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
O        11.1.1.2/32 [110/1000] via 11.1.1.2, 00:15:18, Tunnel23
 11.1.1.3/32 [110/1000] via 11.1.1.3, 00:00:11, Tunnel23
```

R2 is able to ping the loopbacks of R9 thanks to the default route. As a matter of fact, the LSAs have not been flooded to R2.

```
R2#ping 10.11.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.11.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/8/10 ms

R2#ping 10.21.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.21.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms

R2#sh ip route
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

S*    0.0.0.0/0 is directly connected, Tunnel23
      10.0.0.0/8 is variably subnetted, 9 subnets, 2 masks
O        10.1.1.1/32 [110/1001] via 11.1.1.1, 00:20:04, Tunnel23
C        10.1.2.2/32 is directly connected, Loopback0
O        10.1.3.3/32 [110/2001] via 11.1.1.1, 00:20:04, Tunnel23
O        10.1.6.6/32 [110/2011] via 11.1.1.1, 00:20:04, Tunnel23
C        10.1.25.0/24 is directly connected, Serial5/0
```

```
L        10.1.25.1/32 is directly connected, Serial5/0
O        10.1.36.0/24 [110/2010] via 11.1.1.1, 00:20:04, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.2/32 is directly connected, Ethernet0/0
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
O        11.1.1.1/32 [110/1000] via 11.1.1.1, 00:20:04, Tunnel23
L        11.1.1.2/32 is directly connected, Tunnel23
O        11.1.1.3/32 [110/2000] via 11.1.1.1, 00:20:04, Tunnel23
```

**Task 17.6** Configure the following router-ids and make sure that they are in use by the process.

| | |
|---|---|
| R1 | 1.1.1.1 |
| R2 | 2.2.2.2 |
| R3 | 3.3.3.3 |
| R6 | 6.6.6.6 |
| R9 | 9.9.9.9 |

On R1, configure the following:

```
router ospf 1
router-id 1.1.1.1
```

On R2, configure the following:

```
router ospf 1
router-id 2.2.2.2
```

On R3, configure the following:

```
router ospf 1
router-id 3.3.3.3
```

On R6, configure the following:

```
router ospf 1
router-id 6.6.6.6
```

On R9, configure the following:

```
router ospf 1
router-id 9.9.9.9
```

Don't forget to run the clear ip ospf process on each router in order to have the router-ids taken into account.

**Task 17.7** Ensure that the loopback0 network of R1 is not included by the OSPF process in the routing table of R9. Use prefix-list and distribute-list.

As we have to use a prefix-list and a distribute-list, the only way to solve this is to configure a route-map. There is a quicker way (not asked here) to solve this by configuring an access-list and using the access-list in the distribute-list directly.

On R9, configure the following:

```
ip prefix-list 10_1_1_1 seq 5 permit 10.1.1.1/32

route-map FILTER deny 10
  match ip address prefix-list 10_1_1_1
route-map FILTER permit 2

router ospf 1
distribute-list route-map FILTER in
```

The network 10.1.1.1/32 is in the LSA database, but the distribute-list is preventing the network to be listed in the routing table.

```
R9#sh ip route
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, 11 subnets, 2 masks
O IA     10.1.2.2/32 [110/2075] via 10.1.69.6, 00:00:09, Serial3/0
O IA     10.1.3.3/32 [110/75] via 10.1.69.6, 00:00:09, Serial3/0
O IA     10.1.6.6/32 [110/65] via 10.1.69.6, 00:00:09, Serial3/0
C        10.1.9.9/32 is directly connected, Loopback0
O IA     10.1.36.0/24 [110/74] via 10.1.69.6, 00:00:09, Serial3/0
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.9/32 is directly connected, Serial3/0
C        10.11.9.9/32 is directly connected, Loopback1
C        10.21.9.9/32 is directly connected, Loopback2
C        10.31.9.9/32 is directly connected, Loopback3
C        10.41.9.9/32 is directly connected, Loopback4
      11.0.0.0/32 is subnetted, 3 subnets
O IA     11.1.1.1 [110/1074] via 10.1.69.6, 00:00:09, Serial3/0
O IA     11.1.1.2 [110/2074] via 10.1.69.6, 00:00:09, Serial3/0
O IA     11.1.1.3 [110/74] via 10.1.69.6, 00:00:09, Serial3/0

R9#sh ip ospf database

            OSPF Router with ID (9.9.9.9) (Process ID 1)

                Router Link States (Area 69)

Link ID         ADV Router      Age         Seq#       Checksum Link count
6.6.6.6         6.6.6.6         395         0x80000030 0x002950 2
9.9.9.9         9.9.9.9         447         0x80000030 0x000B5E 2

                Summary Net Link States (Area 69)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.1.1        6.6.6.6         395         0x8000002C 0x007D78
10.1.2.2        6.6.6.6         395         0x8000002C 0x009B6C
10.1.3.3        6.6.6.6         395         0x8000002C 0x001FBE
10.1.6.6        6.6.6.6         395         0x8000002C 0x007B66
10.1.36.0       6.6.6.6         395         0x8000002C 0x00C6F9
11.1.1.1        6.6.6.6         395         0x8000002C 0x00668F
11.1.1.2        6.6.6.6         395         0x8000002C 0x008F79
11.1.1.3        6.6.6.6         395         0x8000002C 0x001EC1

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.1.9.9        9.9.9.9         447         0x8000002C 0x0036FD 0
10.11.9.9       9.9.9.9         447         0x8000002C 0x00BD6C 0
10.21.9.9       9.9.9.9         447         0x8000002C 0x0045DA 0
10.31.9.9       9.9.9.9         447         0x8000002C 0x00CC49 0
10.41.9.9       9.9.9.9         447         0x8000002C 0x0054B7 0
```

**Task 17.8** On R9, the network 10.21.9.9/32 should be filtered out and not be propagated. Use distribute-list and access-list.

On R9, configure the following:

```
access-list 1 deny 10.21.9.9
access-list 1 permit any

router ospf 1
distribute-list 1 out connected
```

The filtering is working. We can see that the route 10.21.9.9/32 is not anymore in the routing table of R1.

**Task 17.9** On R3, configure a default route pointing to R5. On R5, configure a default route pointing to R3. Confirm that you can ping from R3 the loopback 0 of R5 10.1.5.5 from the loopback 0 of R3.

On R3, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.35.5
```

On R5, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.35.3
```

I can ping the loopback 0 of R5 10.1.5.5 from the loopback 0 of R3.

```
R3#ping 10.1.5.5 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 17.10** Redistribute this default route into OSPF area 0.

On R3, configure the following:

```
router ospf 1
default-information originate
```

There is now a default route in the routing table of R1:

```
R1#sh ip route
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 11.1.1.3 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 11.1.1.3, 00:01:59, Tunnel23
      10.0.0.0/8 is variably subnetted, 14 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
O        10.1.2.2/32 [110/1001] via 11.1.1.2, 1d05h, Tunnel23
O        10.1.3.3/32 [110/1001] via 11.1.1.3, 1d05h, Tunnel23
O        10.1.6.6/32 [110/1011] via 11.1.1.3, 1d05h, Tunnel23
O IA     10.1.9.9/32 [110/20] via 11.1.1.3, 1d05h, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
O        10.1.36.0/24 [110/1010] via 11.1.1.3, 1d05h, Tunnel23
```

```
O IA     10.1.69.0/24 [110/1074] via 11.1.1.3, 1d05h, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
O E2     10.11.9.9/32 [110/20] via 11.1.1.3, 1d05h, Tunnel23
O E2     10.31.9.9/32 [110/20] via 11.1.1.3, 1d05h, Tunnel23
O E2     10.41.9.9/32 [110/20] via 11.1.1.3, 1d05h, Tunnel23
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
O        11.1.1.2/32 [110/1000] via 11.1.1.2, 1d05h, Tunnel23
O        11.1.1.3/32 [110/1000] via 11.1.1.3, 1d05h, Tunnel23
```

**Task 17.11** On the ABR R6, configure the area 0 to advertise a summary network of 10.1.0.0/16 within the area 69.

On R6, configure the following:

```
router ospf 1
area 0 range 10.1.0.0 255.255.0.0
```

On R9, we can see that a summary route has been advertised from R6, replacing all the specific routes that are originated outside of area 69 within the network 10.1.0.0/16.

```
R9#sh ip route
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 10.1.69.6 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 10.1.69.6, 00:18:10, Serial3/0
      10.0.0.0/8 is variably subnetted, 8 subnets, 3 masks
O IA     10.1.0.0/16 [110/65] via 10.1.69.6, 00:00:09, Serial3/0
C        10.1.9.9/32 is directly connected, Loopback0
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.9/32 is directly connected, Serial3/0
C        10.11.9.9/32 is directly connected, Loopback1
C        10.21.9.9/32 is directly connected, Loopback2
C        10.31.9.9/32 is directly connected, Loopback3
C        10.41.9.9/32 is directly connected, Loopback4
      11.0.0.0/32 is subnetted, 3 subnets
O IA     11.1.1.1 [110/1074] via 10.1.69.6, 04:56:31, Serial3/0
O IA     11.1.1.2 [110/2074] via 10.1.69.6, 04:56:31, Serial3/0
O IA     11.1.1.3 [110/74] via 10.1.69.6, 04:56:31, Serial3/0
```

**Task 17.12** Try to ping the loopback0 of R5 from the loopback0 of R9. Because of the presence of a 10.1.0.0/16 route on the ABR, the default route is not being used and the ping is failing. Ensure that this 10.1.0.0/16 is suppressed.

I cannot ping the loopback0 of R5 from the loopback0 of R9.

```
R9#ping 10.1.5.5 source 10.1.9.9
```

Type escape sequence to abort.

```
Sending 5, 100-byte ICMP Echos to 10.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 10.1.9.9
U.U.U
Success rate is 0 percent (0/5)
```

This is due to the presence of the route to Null0 in the routing table of R6. The default route is not used anymore as a more specific route is preferred.

```
R6#sh ip route
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 10.1.36.3 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 10.1.36.3, 00:11:09, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 14 subnets, 3 masks
O        10.1.0.0/16 is a summary, 00:11:09, Null0
O        10.1.1.1/32 [110/1011] via 10.1.36.3, 00:11:09, Ethernet0/0
O        10.1.2.2/32 [110/2011] via 10.1.36.3, 00:11:09, Ethernet0/0
O        10.1.3.3/32 [110/11] via 10.1.36.3, 00:11:09, Ethernet0/0
C        10.1.6.6/32 is directly connected, Loopback0
O IA     10.1.9.9/32 [110/20] via 10.1.69.9, 00:11:09, Serial3/0
C        10.1.36.0/24 is directly connected, Ethernet0/0
L        10.1.36.6/32 is directly connected, Ethernet0/0
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
C        10.11.6.6/32 is directly connected, Loopback1
O E2     10.11.9.9/32 [110/20] via 10.1.69.9, 00:11:09, Serial3/0
O E2     10.31.9.9/32 [110/20] via 10.1.69.9, 00:11:09, Serial3/0
O E2     10.41.9.9/32 [110/20] via 10.1.69.9, 00:11:09, Serial3/0
      11.0.0.0/32 is subnetted, 3 subnets
O        11.1.1.1 [110/1010] via 10.1.36.3, 00:11:09, Ethernet0/0
O        11.1.1.2 [110/2010] via 10.1.36.3, 00:11:09, Ethernet0/0
O        11.1.1.3 [110/10] via 10.1.36.3, 00:11:09, Ethernet0/0
```

On R6, configure the following:

```
router ospf 1
no discard-route internal
```

The route to Null0 has been removed from the routing table of R6 and the ping from R9 to R5 is working again.

```
R6#sh ip route
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 10.1.36.3 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 10.1.36.3, 00:00:25, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 13 subnets, 2 masks
O        10.1.1.1/32 [110/1011] via 10.1.36.3, 00:00:25, Ethernet0/0
O        10.1.2.2/32 [110/2011] via 10.1.36.3, 00:00:25, Ethernet0/0
O        10.1.3.3/32 [110/11] via 10.1.36.3, 00:00:25, Ethernet0/0
C        10.1.6.6/32 is directly connected, Loopback0
O IA     10.1.9.9/32 [110/20] via 10.1.69.9, 00:00:25, Serial3/0
C        10.1.36.0/24 is directly connected, Ethernet0/0
L        10.1.36.6/32 is directly connected, Ethernet0/0
C        10.1.69.0/24 is directly connected, Serial3/0
L        10.1.69.6/32 is directly connected, Serial3/0
```

```
C        10.11.6.6/32 is directly connected, Loopback1
O E2     10.11.9.9/32 [110/20] via 10.1.69.9, 00:00:25, Serial3/0
O E2     10.31.9.9/32 [110/20] via 10.1.69.9, 00:00:25, Serial3/0
O E2     10.41.9.9/32 [110/20] via 10.1.69.9, 00:00:25, Serial3/0
      11.0.0.0/32 is subnetted, 3 subnets
O        11.1.1.1 [110/1010] via 10.1.36.3, 00:00:25, Ethernet0/0
O        11.1.1.2 [110/2010] via 10.1.36.3, 00:00:25, Ethernet0/0
O        11.1.1.3 [110/10] via 10.1.36.3, 00:00:25, Ethernet0/0

R9#ping 10.1.5.5 source 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 10.1.9.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 13/16/21 ms
```

**Task 17.13** On R1, the network 10.41.9.9/32 should be present in the OSPF database but not in the routing table. Manipulate the administrative distance to achieve this.

To filter this route, we can either issue a specific distance command using 9.9.9.9 as the IP source or we can simply use the 0.0.0.0 255.255.255.255 to imply all IP sources of the route. Let's do the first more specific one.

The LSA is originated on the ASBR which is the router R9 with a router-id of 9.9.9.9.

On R1, configure the following:

```
ip access-list standard FILTER
permit 10.41.9.9 0.0.0.0

router ospf 1
  distance 255 9.9.9.9 0.0.0.0 FILTER
```

On R1, the network 10.41.9.9 has disappeared from the routing table on R1 but is still present in the OSPF database.

```
R1#sh ip route
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 11.1.1.3 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 11.1.1.3, 00:10:17, Tunnel23
      10.0.0.0/8 is variably subnetted, 13 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
O        10.1.2.2/32 [110/1001] via 11.1.1.2, 00:10:17, Tunnel23
O        10.1.3.3/32 [110/1001] via 11.1.1.3, 00:10:17, Tunnel23
O        10.1.6.6/32 [110/1011] via 11.1.1.3, 00:10:17, Tunnel23
O IA     10.1.9.9/32 [110/1075] via 11.1.1.3, 00:10:17, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
O        10.1.36.0/24 [110/1010] via 11.1.1.3, 00:10:17, Tunnel23
O IA     10.1.69.0/24 [110/1074] via 11.1.1.3, 00:10:17, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
O E2     10.11.9.9/32 [110/20] via 11.1.1.3, 00:10:17, Tunnel23
O E2     10.31.9.9/32 [110/20] via 11.1.1.3, 00:10:17, Tunnel23
```

```
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
O        11.1.1.2/32 [110/1000] via 11.1.1.2, 00:10:17, Tunnel23
O        11.1.1.3/32 [110/1000] via 11.1.1.3, 00:10:17, Tunnel23

R1#sh ip ospf database

            OSPF Router with ID (1.1.1.1) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         1052        0x80000059 0x0083FE 4
2.2.2.2         2.2.2.2         669         0x80000050 0x00395E 3
3.3.3.3         3.3.3.3         1102        0x80000055 0x00CD36 4
6.6.6.6         6.6.6.6         1469        0x80000055 0x003FD2 2

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.36.6       6.6.6.6         1469        0x80000049 0x00651B

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.9.9        6.6.6.6         735         0x80000001 0x0015B1
10.1.69.0       6.6.6.6         1724        0x80000014 0x00A8D8

                Summary ASB Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
9.9.9.9         6.6.6.6         1469        0x80000049 0x00195E

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         3.3.3.3         1102        0x80000015 0x00B8D9 1
10.11.9.9       9.9.9.9         1500        0x80000049 0x008389 0
10.31.9.9       9.9.9.9         1500        0x80000049 0x009266 0
10.41.9.9       9.9.9.9         1500        0x80000049 0x001AD4 0
```

**Task 17.14** Configure R6 so that R1 doesn't receive the 10.1.9.9/32 prefix. Use prefix-list and area filter-list.

On R6, configure the following:

```
ip prefix-list LOOPBACK0 seq 5 deny 10.1.9.9/32
ip prefix-list LOOPBACK0 seq 15 permit 0.0.0.0/0 le 32

router ospf 1
  area 69 filter-list prefix LOOPBACK0 out
```

Please note that area filter-list only works for LSA type 3 on the ABR routers. On R1, the network 10.1.9.9 has disappeared from the routing table as well as from the OSPF database.

```
R1# sh ip route
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 11.1.1.3 to network 0.0.0.0
```

```
O*E2  0.0.0.0/0 [110/1] via 11.1.1.3, 00:13:44, Tunnel23
      10.0.0.0/8 is variably subnetted, 12 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
O        10.1.2.2/32 [110/1001] via 11.1.1.2, 00:13:44, Tunnel23
O        10.1.3.3/32 [110/1001] via 11.1.1.3, 00:13:44, Tunnel23
O        10.1.6.6/32 [110/1011] via 11.1.1.3, 00:13:44, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
O        10.1.36.0/24 [110/1010] via 11.1.1.3, 00:13:44, Tunnel23
O IA     10.1.69.0/24 [110/1074] via 11.1.1.3, 00:13:44, Tunnel23
C        10.1.123.0/24 is directly connected, Ethernet0/1
L        10.1.123.1/32 is directly connected, Ethernet0/1
O E2     10.11.9.9/32 [110/20] via 11.1.1.3, 00:13:44, Tunnel23
O E2     10.31.9.9/32 [110/20] via 11.1.1.3, 00:13:44, Tunnel23
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
O        11.1.1.2/32 [110/1000] via 11.1.1.2, 00:13:44, Tunnel23
O        11.1.1.3/32 [110/1000] via 11.1.1.3, 00:13:44, Tunnel23

R1#sh ip ospf database

            OSPF Router with ID (1.1.1.1) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         1261        0x80000059 0x0083FE 4
2.2.2.2         2.2.2.2         878         0x80000050 0x00395E 3
3.3.3.3         3.3.3.3         1311        0x80000055 0x00CD36 4
6.6.6.6         6.6.6.6         1678        0x80000055 0x003FD2 2

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.36.6       6.6.6.6         1678        0x80000049 0x00651B

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.69.0       6.6.6.6         1933        0x80000014 0x00A8D8

                Summary ASB Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
9.9.9.9         6.6.6.6         1678        0x80000049 0x00195E

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         3.3.3.3         1311        0x80000015 0x00B8D9 1
10.11.9.9       9.9.9.9         1709        0x80000049 0x008389 0
10.31.9.9       9.9.9.9         1709        0x80000049 0x009266 0
10.41.9.9       9.9.9.9         1709        0x80000049 0x001AD4 0
```

**Task 17.15** Configure a NSSA area 14 between R1 and R4. On R4, redistribute all connected interfaces into OSPF.

On R1 and R4, configure the following:

```
router ospf 1
network 10.1.14.0 0.0.0.255 area 14
area 14 nssa
```

On R4, configure the following:

```
router ospf 1
redistribute connected subnets
```

On R3, the connected networks of R4 are present on the routing table.

```
R3#sh ip route
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 10.1.35.5 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 10.1.35.5
      10.0.0.0/8 is variably subnetted, 18 subnets, 2 masks
O        10.1.1.1/32 [110/1001] via 11.1.1.1, 1d18h, Tunnel23
O        10.1.2.2/32 [110/2001] via 11.1.1.1, 1d18h, Tunnel23
C        10.1.3.3/32 is directly connected, Loopback0
O E2     10.1.4.4/32 [110/20] via 11.1.1.1, 00:00:41, Tunnel23
O        10.1.6.6/32 [110/11] via 10.1.36.6, 1d18h, Ethernet0/1
O IA     10.1.14.0/24 [110/1010] via 11.1.1.1, 01:30:06, Tunnel23
C        10.1.35.0/24 is directly connected, Serial4/0
L        10.1.35.3/32 is directly connected, Serial4/0
C        10.1.36.0/24 is directly connected, Ethernet0/1
L        10.1.36.3/32 is directly connected, Ethernet0/1
O IA     10.1.69.0/24 [110/74] via 10.1.36.6, 12:50:05, Ethernet0/1
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
O E2     10.11.4.4/32 [110/20] via 11.1.1.1, 00:00:41, Tunnel23
O E2     10.11.9.9/32 [110/20] via 10.1.36.6, 1d18h, Ethernet0/1
O E2     10.22.4.4/32 [110/20] via 11.1.1.1, 00:00:41, Tunnel23
O E2     10.31.9.9/32 [110/20] via 10.1.36.6, 1d18h, Ethernet0/1
O E2     10.41.9.9/32 [110/20] via 10.1.36.6, 1d18h, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
O        11.1.1.1/32 [110/1000] via 11.1.1.1, 1d18h, Tunnel23
O        11.1.1.2/32 [110/2000] via 11.1.1.1, 1d18h, Tunnel23
L        11.1.1.3/32 is directly connected, Tunnel23
```

**Task 17.16** On R1, filter the network 10.11.4.4/32 and 10.22.4.4/32 out and let the other networks coming from area 14 advertise to the area 0. Use summary-address command.

On R1, configure the following:

```
router ospf 1
summary-address 10.11.4.4 255.255.255.255 not-advertise
summary-address 10.22.4.4 255.255.255.255 not-advertise
```

Please note that the summary-address command is only working on an ASBR or on an ABR from a NSSA area because the ABR of the NSSA area translates LSA7 into LSA5 and manipulation of the networks is therefore possible.

In the routing table of R3, we can see that the only external route originated on R4 that is left is the network 10.1.4.4/32.

```
R3#sh ip route
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 10.1.35.5 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 10.1.35.5
      10.0.0.0/8 is variably subnetted, 16 subnets, 2 masks
O        10.1.1.1/32 [110/1001] via 11.1.1.1, 1d18h, Tunnel23
O        10.1.2.2/32 [110/2001] via 11.1.1.1, 1d18h, Tunnel23
C        10.1.3.3/32 is directly connected, Loopback0
O E2     10.1.4.4/32 [110/20] via 11.1.1.1, 00:19:34, Tunnel23
O        10.1.6.6/32 [110/11] via 10.1.36.6, 1d18h, Ethernet0/1
O IA     10.1.14.0/24 [110/1010] via 11.1.1.1, 01:48:59, Tunnel23
C        10.1.35.0/24 is directly connected, Serial4/0
L        10.1.35.3/32 is directly connected, Serial4/0
C        10.1.36.0/24 is directly connected, Ethernet0/1
L        10.1.36.3/32 is directly connected, Ethernet0/1
O IA     10.1.69.0/24 [110/74] via 10.1.36.6, 13:08:58, Ethernet0/1
C        10.1.123.0/24 is directly connected, Ethernet0/0
L        10.1.123.3/32 is directly connected, Ethernet0/0
O E2     10.11.9.9/32 [110/20] via 10.1.36.6, 1d18h, Ethernet0/1
O E2     10.31.9.9/32 [110/20] via 10.1.36.6, 1d18h, Ethernet0/1
O E2     10.41.9.9/32 [110/20] via 10.1.36.6, 1d18h, Ethernet0/1
      11.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
O        11.1.1.1/32 [110/1000] via 11.1.1.1, 1d18h, Tunnel23
O        11.1.1.2/32 [110/2000] via 11.1.1.1, 1d18h, Tunnel23
L        11.1.1.3/32 is directly connected, Tunnel23
```

**Task 17.17** Configure on all the routers the feature that will remove the transit networks from the OSPF database. Check that IP reachability is still working between the OSPF advertised prefixes once this feature is enabled.

On R1, R2, R3, R4, R5, R6, and R9, configure the following:

```
router ospf 1
prefix-suppression
```

I can check in the routing table of R1 that all the transit networks are not advertised anymore. The only transit networks present in the routing table are the directly connected transit networks.

```
R1#sh ip route
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 11.1.1.3 to network 0.0.0.0

O*E2  0.0.0.0/0 [110/1] via 11.1.1.3, 05:41:12, Tunnel23
      10.0.0.0/8 is variably subnetted, 13 subnets, 2 masks
C        10.1.1.1/32 is directly connected, Loopback0
O        10.1.2.2/32 [110/1001] via 11.1.1.2, 05:41:12, Tunnel23
O        10.1.3.3/32 [110/1001] via 11.1.1.3, 05:41:12, Tunnel23
O N2     10.1.4.4/32 [110/20] via 10.1.14.4, 04:11:37, Ethernet0/0
O        10.1.6.6/32 [110/1011] via 11.1.1.3, 05:41:12, Tunnel23
C        10.1.14.0/24 is directly connected, Ethernet0/0
L        10.1.14.1/32 is directly connected, Ethernet0/0
C        10.1.123.0/24 is directly connected, Ethernet0/1
```

```
L        10.1.123.1/32 is directly connected, Ethernet0/1
O N2     10.11.4.4/32 [110/20] via 10.1.14.4, 04:11:37, Ethernet0/0
O E2     10.11.9.9/32 [110/20] via 11.1.1.3, 05:41:12, Tunnel23
O N2     10.22.4.4/32 [110/20] via 10.1.14.4, 04:11:37, Ethernet0/0
O E2     10.31.9.9/32 [110/20] via 11.1.1.3, 05:41:12, Tunnel23
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.1.1.0/24 is directly connected, Tunnel23
L        11.1.1.1/32 is directly connected, Tunnel23
```

For example, I can check the IP connectivity by pinging from R2 the loopback0 of R6.

```
R2#ping 10.1.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 17.18** On R9, configure the minimum interval for accepting the same LSA to 80 ms.

On R9, configure the following:

```
router ospf 1
timers lsa arrival 80
```

The same LSA is considered to have the same LSA ID number, LSA Type and Advertising Router ID.

**Task 17.19** On R9, set the following rate-limit values for LSA advertisement:

| | |
|---|---|
| Start-interval | 10 ms |
| Hold-interval | 100 ms |
| Max-interval | 5000 ms |

On R9, configure the following:

```
router ospf 1
 timers throttle lsa all 10 100 5000
```

**Task 17.20** On R9, configure OSPF throttling timers:

| | |
|---|---|
| Spf-start | 10 ms |
| Spf-hold | 4800 ms |
| Spf-max-wait | 90000 ms |

The SPF algorithm should wait at least 10 milliseconds before it runs again upon receiving a new LSA. The minimum time before two consecutive SPF calculations should be 4, 8 seconds and the maximum should be 1 minute and 30 seconds.

On R9, configure the following:

```
router ospf 1
 timers throttle spf 10 4800 90000
```

**Task 17.21** On R9, configure OSPF Update flood packet-pacing to 5 ms.

On R9, configure the following:

```
router ospf 1
timers pacing flood 5
```

**Task 17.22** On R9, in order to improve convergence, enable incremental SPF.

On R9, configure the following:

```
router ospf 1
 ispf
```

**Task 17.23** R9 should fire up a syslog message when more than 3 prefixes are redistributed. First warning should be sent when 80% of the threshold is reached.

On R9, configure the following:

```
router ospf 1
redistribute maximum-prefix 3 80 warning-only
```

**Task 17.24** On R9, limit to 1000 the number of nonself-generated LSAs the OSPF routing process can keep in the OSPF database.

On R9, configure the following:

```
router ospf 1
max-lsa 1000
```

**You have completed Lab 17**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 18: Configure and troubleshoot BGP (Part 1)

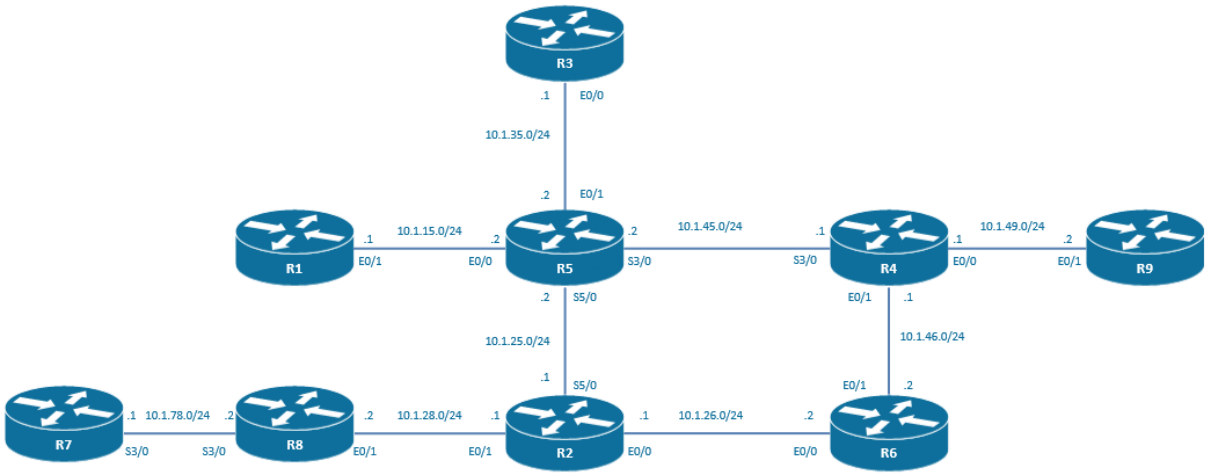
## Technologies covered

- EBGP peering
- EBGP multihop
- EBGP Disable-connected-check
- Update source
- iBGP peering
- Route Reflector

## Overview

You have been tasked to configure the routing in your network using OSPF, EIGRP, RIP, static route, iBGP and eBGP.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 18.1** Routing between R1 and R5 should be configured with RIP version 2. Loopback0 reachability has to be achieved thanks to this protocol.

On R1, configure the following:

```
router rip
 version 2
 network 10.0.0.0
 no auto-summary
```

On R5, configure the following:

```
router rip
 version 2
 network 10.0.0.0
 no auto-summary
```

We have now established IP connectivity using RIPv2 between the loopback0 of R5 and the loopback0 of R1:

```
R5#ping 10.1.1.1 source 10.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 18.2** Configure an eBGP peering between R1 in AS 1 and R5 in AS 65001. This peering should be established between the loopback0 of each router.

We have to create an eBGP peering using the loopback0 IP address as the source of the peering. The loopback0 of R1 has to be reachable from R5 and the loopback0 of R5 has to be reachable from R1. The IGP RIP version 2 has been configured in the previous question for this purpose.
By default, an eBGP peering is expected to be establish over a point-to-point connection using the physical IP address as the source of the peering. In order to modify this default behavior, we configure the neighbor ebgp-multihop x parameter where x is the maximum number of hops that the eBGP peering is allowed to cross to form an eBGP peering. Loopbacks are seen as internal hops so to go from the loopback0 of R5 to the loopback0 of R1; we have to cross 2 hops.

On R1, configure the following:

```
router bgp 1
neighbor 10.1.5.5 remote-as 65001
neighbor 10.1.5.5 ebgp-multihop 2
neighbor 10.1.5.5 update-source Loopback0
```

On R5, configure the following:

```
router bgp 65001
neighbor 10.1.1.1 remote-as 1
neighbor 10.1.1.1 ebgp-multihop 2
neighbor 10.1.1.1 update-source Loopback0
```

Let's check if the peering is up and running:

```
R1#sh ip bgp summary
BGP router identifier 10.1.1.1, local AS number 1
BGP table version is 1, main routing table version 1

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.5.5        4        65001      19      19        1    0    0 00:14:56        0
```

The peering between R1 and R5 is up and running. At the moment, the number of prefixes exchanged on this peering is 0.

**Task 18.3** On the peering between R1 and R5, do not use the ebgp multihop command.

This question is confusing at first glance. How can I enable an eBGP peering originated at the loopback interfaces without using the ebgp multihop command? Remember eBGP has a default rule to only establish peering over connected networks. There is a way to tell the eBGP protocol not to check at all for this rule. This is the neighbor disable-connected-check command. Please not that this command is only working if the 2 neighbors are directly connected, that is to say it is not possible to have a routed hop in between.

Let's remove the ebgp-multihop command on the peering between R1 and R5:

On R1, configure the following:

```
router bgp 1
no  neighbor 10.1.5.5 ebgp-multihop 2

On R5, configure the following:
router bgp 65001
no  neighbor 10.1.1.1 ebgp-multihop 2
```

And clear the bgp process in order to have those changes taken effect.

```
R1#clear ip bgp *
R1#
%BGP-5-ADJCHANGE: neighbor 10.1.5.5 Down User reset
%BGP_SESSION-5-ADJCHANGE: neighbor 10.1.5.5 IPv4 Unicast topology base removed from session
User reset
```

The peering is going down and is not coming up anymore.

Let's disable the connected checking's for eBGP peering's on the peering between R1 and R5:

On R1, configure the following:

```
router bgp 1
neighbor 10.1.5.5 disable-connected-check
```

On R5, configure the following:

```
router bgp 65001
neighbor 10.1.1.1 disable-connected-check
```

As soon as this bypass checking is disabled, is the peering coming up again.

```
R5#sh ip bgp summary
BGP router identifier 10.11.5.5, local AS number 65001
BGP table version is 1, main routing table version 1

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.1.1        4            1       5       5        1    0    0 00:01:18        0
```

**Task 18.4** Advertise the loopback0 of R1 in BGP using a network statement.

On R1, configure the following:

```
router bgp 1
network 10.1.1.1
```

Let's check if I can see the network 10.1.1.1 in the bgp database of R5:

```
R5#sh ip bgp
BGP table version is 4, local router ID is 10.11.5.5
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
 *>  10.1.1.1/32      10.1.1.1                 0             0 1 i
```

**Task 18.5** Routing between R3 and R5 should be configured with static routes. Loopback0 reachability has to be achieved thanks to this protocol.

On R3, configure the following:

```
ip route 10.1.5.5 255.255.255.255 10.1.35.2
```

On R5, configure the following:

```
ip route 10.1.3.3 255.255.255.255 10.1.35.1
```

Let's check that I can ping to the loopback of R3 from the loopback of R5:

```
R5#ping 10.1.3.3 source 10.1.5.5
Type escapes sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.3.3, timeout is 2 seconds:
Packet sent with a source address of 10.1.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 18.6** Configure an eBGP peering between R3 in AS 3 and R5 in AS 65001. This peering should be established between the loopback0 of each router. On the peering between R3 and R5, use the ebgp multihop command.

On R3, configure the following:

```
router bgp 3
  neighbor 10.1.5.5 remote-as 65001
  neighbor 10.1.5.5 ebgp-multihop 2
  neighbor 10.1.5.5 update-source Loopback0
```

On R5, configure the following:

```
router bgp 65001
  neighbor 10.1.3.3 remote-as 3
  neighbor 10.1.3.3 ebgp-multihop 2
  neighbor 10.1.3.3 update-source Loopback0
```

Let's check if the peering is up and running:

```
R3#sh ip bgp sum
BGP router identifier 10.1.3.3, local AS number 3
BGP table version is 2, main routing table version 2
1 network entries using 140 bytes of memory
1 path entries using 80 bytes of memory
1/1 BGP path/bestpath attribute entries using 144 bytes of memory
1 BGP AS-PATH 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 388 total bytes of memory
BGP activity 1/0 prefixes, 1/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.5.5        4        65001       8       4        2    0    0 00:00:44        1
```

The peering between R3 and R5 is up and running. At the moment, the number of prefixes exchanged on this peering is 1, which is the loopback0 of R1 advertised in BGP earlier.

**Task 18.7** Advertise the loopback0 of R3 in BGP using a network statement.

On R3, configure the following:

```
router bgp 3
network 10.1.3.3 mask 255.255.255.255
```

Let's check that this network appears now in the BGP table of the BGP routers:

```
R1#sh ip bgp
BGP table version is 3, local router ID is 10.1.1.1
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
 *>  10.1.1.1/32      0.0.0.0                  0         32768 i
 *>  10.1.3.3/32      10.1.5.5                                 0 65001 3 i
```

**Task 18.8** Routing between R2 and R7 should be configured with EIGRP. Loopback0 reachability has to be achieved thanks to this protocol.

The router R2 and R7 are not directly connected. R2 is connected to R8 and R8 is connected to R7. We have to enable IP connectivity between the loopback0 of R2 and R7 using EIGRP.

On R2, configure the following:

```
router eigrp 1
 network 10.1.2.2 0.0.0.0
 network 10.1.28.0 0.0.0.255
```

On R8, configure the following:

```
router eigrp 1
 network 10.1.28.0 0.0.0.255
 network 10.1.78.0 0.0.0.255
```

On R7, configure the following:

```
router eigrp 1
 network 10.1.7.7 0.0.0.0
 network 10.1.78.0 0.0.0.255
```

Let's check that IP connectivity between the loopback of R7 and R2 is established:

```
R2#ping 10.1.7.7 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/9 ms
```

**Task 18.9** Configure an eBGP peering between R2 in AS 65001 and R7 in AS 7. This peering should be established between the loopback0 of each router. Use the minimum number of hops necessary in the ebg-multihop command.

We have one hop router between R2 and R7. The peering is between the loopback of R2 and the loopback of R7. The peering with loopbacks is adding one additional hop so the neighbor ebgp-multihop command has to specify a minimum a 2 hops.

On R2, configure the following:

```
router bgp 65001
  neighbor 10.1.7.7 remote-as 7
  neighbor 10.1.7.7 ebgp-multihop 2
  neighbor 10.1.7.7 update-source Loopback0
```

On R7, configure the following:

```
router bgp 7
  neighbor 10.1.2.2 remote-as 65001
```

```
 neighbor 10.1.2.2 ebgp-multihop 2
 neighbor 10.1.2.2 update-source Loopback0
```

Let's check if the peering is up and running:

```
R7#sh ip bgp sum
BGP router identifier 10.1.7.7, local AS number 7
BGP table version is 1, main routing table version 1

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.2.2        4        65001       9       9        1    0    0 00:05:47        0
```

**Task 18.10** Advertise the loopback1 of R7 in BGP using a network statement. Check that you can ping from R1 to the loopback1 of R7. Use of static routes on R8 is required.

On R7, configure the following:

```
router bgp 7
network 11.1.7.7 mask 255.255.255.255
```

Let's check that this network appears now in the BGP table of R1:

```
R2#sh ip bgp
BGP table version is 10, local router ID is 10.11.2.2
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
 *>  11.1.7.7/32      10.1.7.7                 0             0 7 i
```

However, it is not possible to ping from R2 to 11.1.7.7:

```
R2#ping 11.1.7.7 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
.....
Success rate is 0 percent (0/5)
```

This is due to the fact that R8 has no routing information for the 11.1.7.7 network. It is a routing black hole for network 11.1.1.7! We have to add a static route on R8.

On R8, configure the following:

```
ip route 11.1.7.7 255.255.255.255 10.1.78.1
```

I can now ping from the loopback0 of R2 to the loopback1 of R7:

```
R2#ping 11.1.7.7 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 18.11** Configure OSPF area 0 on the R2 to R5 connection. Advertise the loopback0 of R2, R5, and into OSPF.

On R2, configure the following:

```
router ospf 1
 network 10.1.2.2 0.0.0.0 area 0
 network 10.1.25.0 0.0.0.255 area 0
```

On R5, configure the following:

```
router ospf 1
 network 10.1.5.5 0.0.0.0 area 0
 network 10.1.25.0 0.0.0.255 area 0
```

Let's check that you can ping the loopback0 of R2 from the loopback0 of R5.

```
R5#ping 10.1.2.2 source 10.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 18.12** Configure iBGP peering between R2 and R5. This peering should be established between the loopback0 of each router. Make sure that the ping from R3 to R7 is up and running. Do not use the redistribute command into BGP at this point of the lab.

On R2, configure the following:

```
router bgp 65001
  neighbor 10.1.5.5 remote-as 65001
  neighbor 10.1.5.5 update-source Loopback0
```

On R5, configure the following:

```
router bgp 65001
  neighbor 10.1.2.2 remote-as 65001
  neighbor 10.1.2.2 update-source Loopback0
```

Let's try if I can ping from the loopback0 of R3 to the loopback1 of R7.

```
R1#ping 11.1.7.7 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
.....
Success rate is 0 percent (0/5)
```

It is not working. Let's have a look at the routing table and at the BGP table of R2:

```
R5#sh ip route 11.1.7.7
% Network not in table

R5#sh ip bgp 11.1.7.7
BGP routing table entry for 11.1.7.7/32, version 553
Paths: (1 available, no best path)
  Not advertised to any peer
  Refresh Epoch 1
  7
    10.1.7.7 (inaccessible) from 10.1.2.2 (10.11.2.2)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
```

The network 11.1.7.7 is in the BGP table and not in the routing table. This is due to the fact that an iBGP peer is not modifying the next-hop of a route to himself when advertising the route. The eBGP next-hop of the route is kept unchanged once the route enters an AS. If the next-hop of route is not reachable by the router that receives the update, the update is accepted and is not placed into the routing table. One of the way to fix this issue is to configure an iBGP to change the next-hop to itself in the same way that eBGP is doing it. This is achieved by configuring the next-hop-sef feature.

On R2, configure the following:

```
router bgp 65001
  neighbor 10.1.5.5 next-hop-self
```

On R5, configure the following:

```
router bgp 65001
  neighbor 10.1.2.2 next-hop-self
```

Now there is a routing entry for 11.1.7.7 in the routing table of R5.

```
R5#sh ip route 11.1.7.7
Routing entry for 11.1.7.7/32
  Known via "bgp 65001", distance 200, metric 0
  Tag 7, type internal
  Last update from 10.1.2.2 00:01:17 ago
  Routing Descriptor Blocks:
  * 10.1.2.2, from 10.1.2.2, 00:01:17 ago
      Route metric is 0, traffic share count is 1
      AS Hops 1
      Route tag 7
      MPLS label: none
```

From this moment on, the ping from loopback0 of R3 to the loopback1 of R7 should be working:

```
R1#ping 11.1.7.7 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
.....
Success rate is 0 percent (0/5)
```

The ping is still not working because the router R8 knows only the routes of R7 and R2 thanks to the EIGRP protocol.
One way to fix this issue is to configure a default route towards R2 on R8.

On R8, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.28.1
```

The ping from the loopback0 of R3 to the loopback1 of R7 is now working.

```
R3#ping 11.1.7.7 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/19 ms
```

**Task 18.13** Enable synchronization on R5. Using a route-map and a prefix-list, redistribute BGP into OSPF on R2. The full IP reachability should be established between the loopback0 of R1, R3, and R7.

Let's enable BGP *synchronization* on R5.

```
router bgp 65001
synchronization
```

The ping from R3 to the loopback1 of R7 has stopped working:

```
R3#ping 11.1.7.7 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
.....
Success rate is 0 percent (0/5)

Let's check the BGP table of R5:

R5#sh ip bgp 11.1.7.7
BGP routing table entry for 11.1.7.7/32, version 682
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
```

```
      7
    Refresh Epoch 1
    7
      10.1.2.2 (metric 65) from 10.1.2.2 (10.11.2.2)
        Origin IGP, metric 0, localpref 100, valid, internal, not synchronized, best
        rx pathid: 0, tx pathid: 0x0
```

BGP synchronization is enabled on R5 and the 11.1.7.7 network is reported as not synchronized. This is due to the fact that the IGP protocol which is the OSPF protocol is not routing the 11.1.7.7 network. What we take note of is exactly the synchronization rule. BGP is removing the 11.1.7.7 network from the routing table because this network is not already known from an IGP routing protocol.

```
R5#sh ip route 11.1.7.7
% Network not in table
```

We have to restore IP connectivity and we will therefore make sure that the 11.1.7.7 network is routed into the OSPF process 1.

On R2, configure the following redistribution:

```
router ospf 1
 redistribute bgp 65001 subnets
```

We can now notice that the network 11.1.7.7 is now reported as synchronized in the BGP database.

```
R5#sh ip bgp 11.1.7.7
BGP routing table entry for 11.1.7.7/32, version 19
Paths: (1 available, best #1, table default, RIB-failure(17))
Flag: 0x820
  Not advertised to any peer
  Refresh Epoch 1
  7
    10.1.2.2 (metric 65) from 10.1.2.2 (10.11.2.2)
      Origin IGP, metric 0, localpref 100, valid, internal, synchronized, best
      rx pathid: 0, tx pathid: 0x0
```

Let's verify that the ping from the loopback0 from R3 to the loopback1 of R7 is again working:

```
R3#ping 11.1.7.7 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/18 ms
```

**Task 18.14** Configure OSPF area 0 on the R5 to R4 connection. Advertise the loopback0 of R4 into OSPF.

On R5, configure the following:

```
router ospf 1
 network 10.1.45.0 0.0.0.255 area 0
```

On R4, configure the following:

```
router ospf 1
 network 10.1.4.4 0.0.0.0 area 0
 network 10.1.45.0 0.0.0.255 area 0
```

Let's check that you can ping the loopback0 of R4 from the loopback0 of R5.

```
R5#ping 10.1.4.4 source 10.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 10.1.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 18.15** Routing between R4 and R9 should be configured with static routes. Loopback0 reachability has to be achieved thanks to this protocol.

On R4, configure the following:
```
ip route 10.1.9.9 255.255.255.255 10.1.49.2
```

On R9, configure the following:
```
ip route 10.1.4.4 255.255.255.255 10.1.49.1
```

let's check that I can ping to the loopback of R4 from the loopback of R9:
```
R9#ping 10.1.4.4 source 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 10.1.9.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
```

**Task 18.16** Configure an eBGP peering between R4 in AS 65001 and R9 in AS 9. This peering should be established between the loopback0 of each router.

On R4, configure the following:
```
router bgp 65001
     neighbor 10.1.9.9 remote-as 9
     neighbor 10.1.9.9 ebgp-multihop 2
     neighbor 10.1.9.9 update-source Loopback0
```

On R9, configure the following:
```
router bgp 9
  neighbor 10.1.4.4 remote-as 65001
  neighbor 10.1.4.4 ebgp-multihop 2
  neighbor 10.1.4.4 update-source Loopback0
```

Let's check that this peering is up and running:
```
R9#sh ip bgp sum
BGP router identifier 10.1.9.9, local AS number 9
BGP table version is 1, main routing table version 1

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.4.4        4        65001       2       2        1    0    0 00:00:12        0
```

**Task 18.17** Advertise the loopback0 of R9 in BGP using a network statement.

On R9, configure the following:
```
router bgp 9
network 10.1.9.9 mask 255.255.255.255
```

Let's check that this network appears now in the BGP table of R4:
```
R4#sh ip bgp
BGP table version is 2, local router ID is 10.11.4.4
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
 r>  10.1.9.9/32      10.1.9.9                 0             0 9 i
```

**Task 18.18** Configure iBGP peering between R4 and R2. This peering should be established between the loopback0 of each router. Configure R2 as a route-reflector for R4 and R5.

On R2, configure the following:

```
router bgp 65001
  neighbor 10.1.4.4 remote-as 65001
  neighbor 10.1.4.4 update-source Loopback0
  neighbor 10.1.4.4 route-reflector-client
  neighbor 10.1.4.4 next-hop-self
  neighbor 10.1.5.5 route-reflector-client
```

On R4, configure the following:

```
router bgp 65001
  neighbor 10.1.2.2 remote-as 65001
  neighbor 10.1.2.2 update-source Loopback0
  neighbor 10.1.2.2 next-hop-self
```

**Task 18.19** On R7, make sure that you can ping from the loopback1 of R7 to the loopback0 of R1, R3, and R9.

Let's try to ping from the loopback1 of R7 to the loopback0 of R9.

```
R7#ping 10.1.9.9 source 11.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 11.1.7.7
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms
```

The ping is working . We did a great job!!!

**Task 18.20** Configure OSPF area 0 on the R2 to R6 connection and on the R4 to R6 connection. Advertise the loopback0 of R6 into OSPF.

On R2, configure the following:

```
router ospf 1
network 10.1.26.0 0.0.0.255 area 0
```

On R6, configure the following:

```
router ospf 1
 network 10.1.6.6 0.0.0.0 area 0
 network 10.1.26.0 0.0.0.255 area 0
 network 10.1.46.0 0.0.0.255 area 0
```

On R4, configure the following:

```
router ospf 1
network 10.1.46.0 0.0.0.255 area 0
```

**Task 18.21** For redundancy, configure R2 and R6 as part of a RR cluster with cluster-id 1.

On R5, configure the following:

```
router bgp 65001
  neighbor 10.1.6.6 remote-as 65001
  neighbor 10.1.6.6 update-source Loopback0
```

On R2, configure the following:

```
router bgp 65001
```

```
  neighbor 10.1.6.6 remote-as 65001
  neighbor 10.1.6.6 update-source Loopback0
```

On R4, configure the following:

```
router bgp 65001
bgp cluster-id 1
  neighbor 10.1.6.6 remote-as 65001
  neighbor 10.1.6.6 update-source Loopback0
```

On R6, configure the following:

```
router bgp 65001
bgp cluster-id 1
  neighbor 10.1.5.5 remote-as 65001
  neighbor 10.1.5.5 update-source Loopback0
  neighbor 10.1.5.5 route-reflector-client
  neighbor 10.1.4.4 remote-as 65001
  neighbor 10.1.4.4 update-source Loopback0
  neighbor 10.1.4.4 route-reflector-client
  neighbor 10.1.2.2 remote-as 65001
  neighbor 10.1.2.2 update-source Loopback0
  neighbor 10.1.2.2 route-reflector-client
```

**You have completed Lab 18**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 19: Configure and troubleshoot BGP (part 2)

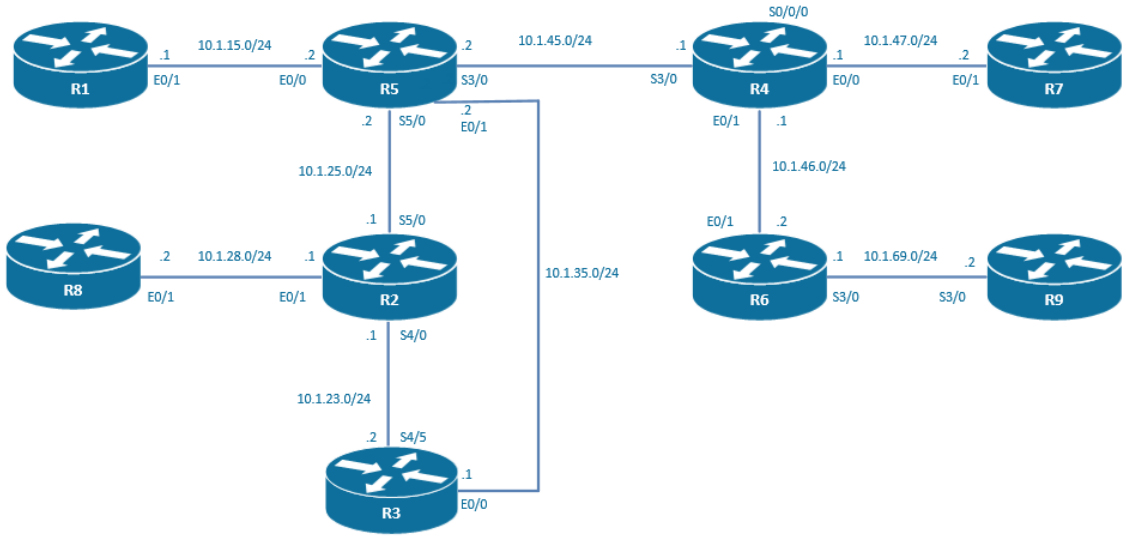
## Technologies covered

- Next-hop-self
- BGP next-hop with route-map
- BGP Confederation
- GRE tunnels

## Overview

You have been tasked to configure the routing in your network using OSPF, EIGRP, RIP, static route, iBGP and eBGP.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 19.1** Routing between R4 and R7 should be configured with EIGRP. Loopback0 reachability has to be achieved thanks to this protocol.

On R4, configure the following:

```
router eigrp 1
network 10.1.4.4 0.0.0.0
network 10.1.47.0 0.0.0.255
```

On R7, configure the following:

```
router eigrp 1
network 10.1.7.7 0.0.0.0
network 10.1.47.0 0.0.0.255
```

On R7, I can ping the loopback0 of R4 with a ping sourcing from the loopback0 of R7.

```
R7#ping 10.1.4.4 source 10.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 10.1.7.7
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

**Task 19.2** Configure an eBGP peering between R4 in AS 65019 and R7 in AS 7. This peering should be established between the loopback0 of each router.

On R4, configure the following:

```
router bgp 65019
neighbor 10.1.7.7 remote-as 7
neighbor 10.1.7.7 ebgp-multihop 2
neighbor 10.1.7.7 update-source lo0
```

On R7, configure the following:

```
router bgp 7
neighbor 10.1.4.4 remote-as 65019
neighbor 10.1.4.4 ebgp-multihop 2
neighbor 10.1.4.4 update-source lo0
```

**Task 19.3** Advertise the loopback0 of R7 in BGP using a network statement.

On R7, configure the following:

```
router bgp 7
network 10.1.7.7 mask 255.255.255.255
```

Once this has been configured, I cannot ping anymore from the loopback0 of R7 to the loopback0 of R4.

```
R7#ping 10.1.4.4 source 10.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 10.1.7.7
.....
Success rate is 0 percent (0/5)
```

We notice that the FD has been set to infinity. This is due to the fact that the 10.1.7.7 network is now known via eBGP and eBGP has an administrative distance of 20 whereas EIGRP has an administrative distance of 90.

```
R4#sh ip eigrp 1 topology
EIGRP-IPv4 Topology Table for AS(1)/ID(10.11.4.4)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.1.4.4/32, 1 successors, FD is 128256
        via Connected, Loopback0
P 10.1.7.7/32, 0 successors, FD is Infinity
        via 10.1.47.2 (409600/128256), Ethernet0/0
P 10.1.47.0/24, 1 successors, FD is 281600
        via Connected, Ethernet0/0
```

On R4, configure the following:

```
access-list 1 permit 10.1.7.7
```

```
router bgp 65019
distance 200 10.1.7.7 0.0.0.0 1
```

Now is the EIGRP route again preferred and the FD is not anymore set to Infinity.

```
R4#sh ip eigrp topology
EIGRP-IPv4 Topology Table for AS(1)/ID(10.11.4.4)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.1.4.4/32, 1 successors, FD is 128256
        via Connected, Loopback0
P 10.1.7.7/32, 1 successors, FD is 409600
        via 10.1.47.2 (409600/128256), Ethernet0/0
P 10.1.47.0/24, 1 successors, FD is 281600
        via Connected, Ethernet0/0
```

**Task 19.4** Routing between R6 and R9 should be configured with static routes. Loopback0 reachability has to be achieved thanks to this protocol.

On R6, configure the following:

```
ip route 10.1.9.9 255.255.255.255 10.1.69.9
```

On R9, configure the following:

```
ip route 10.1.6.6 255.255.255.255 10.1.69.6
```

On R6, I can ping the loopback0 of R9 with a ping sourcing from the loopback0 of R6.

```
R6#ping 10.1.9.9 source 10.1.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.6.6
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 19.5** Configure an eBGP peering between R6 in AS 65019 and R9 in AS 9. This peering should be established between the loopback0 of each router.

On R6, configure the following:

```
router bgp 65019
neighbor 10.1.9.9 remote-as 9
neighbor 10.1.9.9 ebgp-multihop 2
neighbor 10.1.9.9 update-source lo0
```

On R9, configure the following:

```
router bgp 9
neighbor 10.1.6.6 remote-as 65019
neighbor 10.1.6.6 ebgp-multihop 2
neighbor 10.1.6.6 update-source lo0
```

**Task 19.6** Advertise the loopback0 of R9 in BGP using a network statement.

On R9, configure the following:

```
router bgp 9
network 10.1.9.9 mask 255.255.255.255
```

**Task 19.7** Configure OSPF area 0 only between R4 and R6. Advertise the loopback0 of R4 and R6 into OSPF using a network statement. Do not advertise anything else into OSPF.

On R4, configure the following:

```
router ospf 1
network 10.1.46.0 0.0.0.255 area 0
network 10.1.4.4 0.0.0.0 area 0
```

On R6, configure the following:

```
router ospf 1
network 10.1.46.0 0.0.0.255 area 0
network 10.1.6.6 0.0.0.0 area 0
```

### Task 19.8 Configure iBGP between R4 and R6.

On R4, configure the following:

```
router bgp 65019
neighbor 10.1.6.6 remote-as 65019
neighbor 10.1.6.6 update-source lo0
```

On R6, configure the following:

```
router bgp 65019
neighbor 10.1.4.4 remote-as 65019
neighbor 10.1.4.4 update-source lo0
```

### Task 19.9 Use next-hop-self to enable the IP connectivity between the loopback0 of R7 and the loopback0 of R9.

On R7, I cannot see the 10.1.9.9 network in the BGP table.

```
R7#sh ip bgp
BGP table version is 2, local router ID is 10.1.7.7
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
 *>  10.1.7.7/32      0.0.0.0                  0         32768 i

The 10.1.9.9 network has not been sent from R4 to R7 because the next-hop in the update for
10.1.9.9 on R4 is not reachable.

R4#sh ip bgp 10.1.9.9
BGP routing table entry for 10.1.9.9/32, version 0
Paths: (1 available, no best path)
  Not advertised to any peer
  Refresh Epoch 1
  9
    10.1.9.9 (inaccessible) from 10.1.6.6 (10.11.6.6)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
```

On R4, configure the following:

```
router bgp 65019
neighbor 10.1.6.6 next-hop-self
```

On R6, configure the following:

```
router bgp 65019
neighbor 10.1.4.4 next-hop-self
```

Once the next-hop-self-command has been configured, the next hop on the eBGP updates is superseded by the one advertising the update. On R4, the update for 10.1.9.9 has now a reachable next-hop and will be sent to R7.

```
R4#sh ip bgp 10.1.9.9
BGP routing table entry for 10.1.9.9/32, version 83
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     21
  Refresh Epoch 1
  9
    10.1.6.6 (metric 11) from 10.1.6.6 (10.11.6.6)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      rx pathid: 0, tx pathid: 0x0
```

I have now IP connectivity between the loopback0 of R7 and the loopback0 of R9.

```
R7#ping 10.1.9.9 source 10.1.7.7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.7.7
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/10/13 ms
```

**Task 19.10** Routing between R5 and R1 should be configured with RIP. Loopback0 reachability has to be achieved thanks to this protocol.

On R5, configure the following:

```
router rip
network 10.1.15.0
network 10.1.5.5
version 2
no auto-summary
```

On R1, configure the following:

```
router rip
network 10.1.15.0
network 10.1.1.1
version 2
no auto-summary
```

I can now ping from loopback0 to loopback0 between R5 and R1.

```
R1#ping 10.1.5.5 source 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.5.5, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 19.11** Configure an eBGP peering between R1 in AS 1 and R5 in AS 65019. This peering should be established between the loopback0 of each router.

On R1, configure the following:

```
router bgp 1
neighbor 10.1.5.5 remote-as 65019
neighbor 10.1.5.5 ebgp-multihop 2
neighbor 10.1.5.5 update-source lo0
```

On R5, configure the following:

```
router bgp 65019
neighbor 10.1.1.1 remote-as 1
neighbor 10.1.1.1 ebgp-multihop 2
neighbor 10.1.1.1 update-source lo0
```

**Task 19.12** Advertise the loopback0 of R1 in BGP using a network statement.

On R1, configure the following:

```
router bgp 1
network 10.1.1.1 mask 255.255.255.255
```

IP connectivity between loopback0 of R1 and R5 has been broken because the eBGP route is preferred over the RIP route because of the AD.

On R5, configure the following:

```
access-list 1 permit 10.1.1.1

router bgp 65019
distance 200 10.1.1.1 0.0.0.0 1
```

**Task 19.13** Routing between R8 and R2 should be configured with EIGRP. Loopback0 reachability has to be achieved thanks to this protocol.

On R8, configure the following:

```
router eigrp 1
network 10.1.28.0 255.255.255.0
network 10.1.8.8 255.255.255.255

On R2, configure the following:
router eigrp 1
network 10.1.28.0 255.255.255.0
network 10.1.2.2 255.255.255.255
```

**Task 19.14** Configure an eBGP peering between R2 in AS 65019 and R8 in AS 8. This peering should be established between the loopback0 of each router.

On R2, configure the following:

```
router bgp 65019
neighbor 10.1.8.8 remote-as 8
neighbor 10.1.8.8 ebgp-multihop 2
neighbor 10.1.8.8 update-source lo0
```

On R8, configure the following:

```
router bgp 8
neighbor 10.1.2.2 remote-as 65019
neighbor 10.1.2.2 ebgp-multihop 2
neighbor 10.1.2.2 update-source lo0
```

**Task 19.15** Advertise the loopback0 of R8 in BGP using a network statement.

On R8, configure the following:

```
router bgp 8
network 10.1.8.8 mask 255.255.255.255
```

On R2, configure the following:

```
access-list 1 permit 10.1.8.8

router bgp 65019
distance 200 10.1.8.8 0.0.0.0 1
```

**Task 19.16** Configure OSPF area 0 only between R5 and R2. Advertise the loopback0 of R5 and R2 into OSPF using a network statement. Do not advertise anything else into OSPF.

On R5, configure the following:

```
router ospf 1
network 10.1.25.0 0.0.0.255 area 0
network 10.1.5.5 0.0.0.0 area 0
```

On R2, configure the following:

```
router ospf 1
network 10.1.25.0 0.0.0.255 area 0
network 10.1.2.2 0.0.0.0 area 0
```

**Task 19.17** Configure an OSPF cost of 10 on this link.

On R2 and R5, configure the following:

```
int s5/0
ip ospf cost 10
```

**Task 19.18** Configure iBGP between R5 and R2.

On R5, configure the following:

```
router bgp 65019
neighbor 10.1.2.2 remote-as 65019
neighbor 10.1.2.2 update-source lo0
```

On R2, configure the following:

```
router bgp 65019
neighbor 10.1.5.5 remote-as 65019
neighbor 10.1.5.5 update-source lo0
```

**Task 19.19** Use a route-map to enable the IP connectivity between the loopback0 of R1 and the loopback0 of R8.

We are not allowed to use the next-hop-self-command.

On R2, configure the following:

```
route-map SET_NEXT_HOP permit 10
  set ip next-hop 10.1.2.2

 router bgp 65019
  neighbor 10.1.5.5 route-map SET_NEXT_HOP out
```

On R5, configure the following:

```
route-map SET_NEXT_HOP permit 10
  set ip next-hop 10.1.5.5

 router bgp 65019
  neighbor 10.1.2.2 route-map SET_NEXT_HOP out
```

I have IP connectivity between the loopback0 of R1 and R8.

```
R8#ping 10.1.1.1 source 10.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.8.8
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/12 ms
```

**Task 19.20** Configure OSPF area 0 between R5 and R4.

On R5, configure the following:

```
router ospf 1
network 10.1.45.0 0.0.0.255 area 0
```

On R2, configure the following:

```
router ospf 1
network 10.1.45.0 0.0.0.255 area 0
```

**Task 19.21** R2 and R5 are part of confederation with ID 25, R6, and R4 are part of confederation with ID 46.

On R2, configure the following:

```
no router bgp 65019
router bgp 25
 bgp log-neighbor-changes
 neighbor 10.1.5.5 remote-as 25
 neighbor 10.1.5.5 update-source Loopback0
 neighbor 10.1.5.5 route-map SET_NEXT_HOP out
 neighbor 10.1.8.8 remote-as 8
 neighbor 10.1.8.8 ebgp-multihop 2
 neighbor 10.1.8.8 update-source Loopback0
 distance 200 10.1.8.8 0.0.0.0 1
 bgp confederation identifier 65019
 bgp confederation peers 46
 neighbor 10.1.5.5 remote-as 25
 neighbor 10.1.5.5 update-source loopback0
```

On R5, configure the following:

```
access-list 2 permit 10.1.4.4

no router bgp 65019
router bgp 25
 bgp log-neighbor-changes
 neighbor 10.1.1.1 remote-as 1
 neighbor 10.1.1.1 ebgp-multihop 2
 neighbor 10.1.1.1 update-source Loopback0
 neighbor 10.1.2.2 remote-as 25
 neighbor 10.1.2.2 update-source Loopback0
 neighbor 10.1.2.2 route-map SET_NEXT_HOP out
 distance 200 10.1.1.1 0.0.0.0 1
 bgp confederation identifier 65019
 bgp confederation peers 46
 neighbor 10.1.2.2 remote-as 25
 neighbor 10.1.2.2 update-source loopback0
 neighbor 10.1.4.4 remote-as 46
 neighbor 10.1.4.4 ebgp-multihop
 neighbor 10.1.4.4 update-source loopback0
 distance 200 10.1.4.4 0.0.0.0 2
```

On R6, configure the following:

```
no router bgp 65019
router bgp 46
 bgp log-neighbor-changes
 neighbor 10.1.4.4 remote-as 46
 neighbor 10.1.4.4 update-source Loopback0
 neighbor 10.1.4.4 next-hop-self
 neighbor 10.1.9.9 remote-as 9
 neighbor 10.1.9.9 ebgp-multihop 2
 neighbor 10.1.9.9 update-source Loopback0
 bgp confederation identifier 65019
 bgp confederation peers 25
 neighbor 10.1.4.4 remote-as 46
 neighbor 10.1.4.4 update-source loopback0
```

On R4, configure the following

```
access-list 2 permit 10.1.5.5
```

```
no router bgp 65019
router bgp 46
 bgp log-neighbor-changes
 neighbor 10.1.6.6 remote-as 46
 neighbor 10.1.6.6 update-source Loopback0
 neighbor 10.1.6.6 next-hop-self
 neighbor 10.1.7.7 remote-as 7
 neighbor 10.1.7.7 ebgp-multihop 2
 neighbor 10.1.7.7 update-source Loopback0
 distance 200 10.1.7.7 0.0.0.0 1
 bgp confederation identifier 65019
 bgp confederation peers 25
 neighbor 10.1.6.6 remote-as 46
 neighbor 10.1.6.6 update-source loopback0
 neighbor 10.1.5.5 remote-as 25
 neighbor 10.1.5.5 ebgp-multihop
 neighbor 10.1.5.5 update-source loopback0
 distance 200 10.1.5.5 0.0.0.0 2
```

**Task 19.22** Configure the confederation ID 25 and 46 to be part of AS 65019. Ensure full reachability between R1, R7, R8, and R9. As an example, you should be able to ping from R8 to the loopback0 of R7 with the ping sourced from the loopback0 of R8. Use of 2 static routes is allowed.

At this moment, I cannot see the network 10.1.7.7 from R8 and I cannot see the network 10.1.1.1 from R9.

```
R8#sh ip bgp
BGP table version is 10, local router ID is 10.1.8.8
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
 *>  10.1.1.1/32      10.1.2.2                               0 65019 1 i
 *>  10.1.8.8/32      0.0.0.0                  0         32768 i
 *>  10.1.9.9/32      10.1.2.2                               0 65019 9 i


R9#sh ip bgp
BGP table version is 6, local router ID is 10.1.9.9
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
 *>  10.1.7.7/32      10.1.6.6                               0 65019 7 i
 *>  10.1.8.8/32      10.1.6.6                               0 65019 8 i
 *>  10.1.9.9/32      0.0.0.0                  0         32768 i
```

Let’s investigate why. On R4, the network 10.1.1.1 doesn’t have the > best arrow in the BGP database, so it won’t be advertised to R5.

```
R4#sh ip bgp
BGP table version is 6, local router ID is 10.11.4.4
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
```

```
 *   10.1.1.1/32      10.1.1.1                 0    100      0 (25) 1 i
 r>  10.1.7.7/32      10.1.7.7                 0             0 7 i
 *>  10.1.8.8/32      10.1.2.2                 0    100      0 (25) 8 i
 *>i 10.1.9.9/32      10.1.6.6                 0    100      0 9 i
```

This is due to the fact that the 10.1.1.1 update has a next-hop of 10.1.1.1 which is unreachable from the R4 router.

```
R4#sh ip bgp 10.1.1.1
BGP routing table entry for 10.1.1.1/32, version 0
Paths: (1 available, no best path)
Flag: 0x820
  Not advertised to any peer
  Refresh Epoch 1
  (25) 1
    10.1.1.1 (inaccessible) from 10.1.5.5 (10.11.5.5)
      Origin IGP, metric 0, localpref 100, valid, confed-external
      rx pathid: 0, tx pathid: 0
R4#sh ip bgp 10.1.8.8
BGP routing table entry for 10.1.8.8/32, version 3
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
     9          11
  Refresh Epoch 1
  (25) 8
    10.1.2.2 (metric 75) from 10.1.5.5 (10.11.5.5)
      Origin IGP, metric 0, localpref 100, valid, confed-external, best
      rx pathid: 0, tx pathid: 0x0

R4#sh ip route 10.1.1.1
% Subnet not in table
```

On R4, configure the following:

```
ip route 10.1.1.1 255.255.255.255 10.1.45.2
```

This problem happens also on R5. The 10.1.7.7 update has a next-hop of 10.1.7.7 which is unreachable from the R5 router.

On R5, configure the following:

```
ip route 10.1.7.7 255.255.255.255 10.1.45.1
```

I am able to ping from R8 to the loopback0 of R7 with the ping sourced from the loopback0 of R8.

```
R8#ping 10.1.7.7 source 10.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.8.8
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/20 ms
```

**Task 19.23** Configure OSPF area 0 on the connection between R5 and R3 with an OSPF cost of 1.

On R5, configure the following:

```
router ospf 1
network 10.1.35.0 0.0.0.255 area 0

int e0/1
ip ospf cost 1
```

On R3, configure the following:

```
router ospf 1
network 10.1.35.0 0.0.0.255 area 0

int e0/0
ip ospf cost 1
```

**Task 19.24** Configure OSPF area 0 on the connection between R2 and R3 with an OSPF cost of 1.

On R2, configure the following:

```
router ospf 1
network 10.1.23.0 0.0.0.255 area 0

int s4/0
ip ospf cost 1
```

On R3, configure the following:

```
router ospf 1
network 10.1.23.0 0.0.0.255 area 0

int s4/1
ip ospf cost 1
```

**Task 19.25** Restore the IP connectivity between R8 and R1, R8 and R7, and R8 and R9. You are not allowed to redistribute BGP routes into OSPF. Use the network 10.1.145.0/24 for the tunnel interfaces. Check that you are again able to ping from R8 to the loopback0 of R1 with the ping sourced from the loopback0 of R8.

The router R3 is a non-BGP speaker router. The ping from R8 to the loopback0 of R1 with the ping sourced from the loopback0 of R8 is not working because the IGP will attract all the traffic towards R3 that will drop the traffic as he doesn't now the BGP routes, i.e. the source and the destination of the ping.

We neutralize R3 by tunneling through it.

On R5, configure the following:

```
int tu0
tunnel source 10.1.35.2
tunnel destination 10.1.23.1
ip address 10.1.145.1 255.255.255.0
ip ospf cost 1

router ospf 1
network 10.1.145.1 255.255.255.0 area 0

ip route 10.1.23.1 255.255.255.255 10.1.35.1
```

On R2, configure the following:

```
int tu0
tunnel source 10.1.23.1
tunnel destination 10.1.35.2
ip address 10.1.145.2 255.255.255.0
ip ospf cost 1

router ospf 1
network 10.1.145.1 255.255.255.0 area 0

ip route 10.1.35.2 255.255.255.255 10.1.23.2
```

We can check that the ping is working again.

```
R8#ping 10.1.7.7 source 10.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.8.8
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 15/17/21 ms
```

**You have completed Lab 19**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com

# Lab 20: Configure and troubleshoot BGP (part 3)

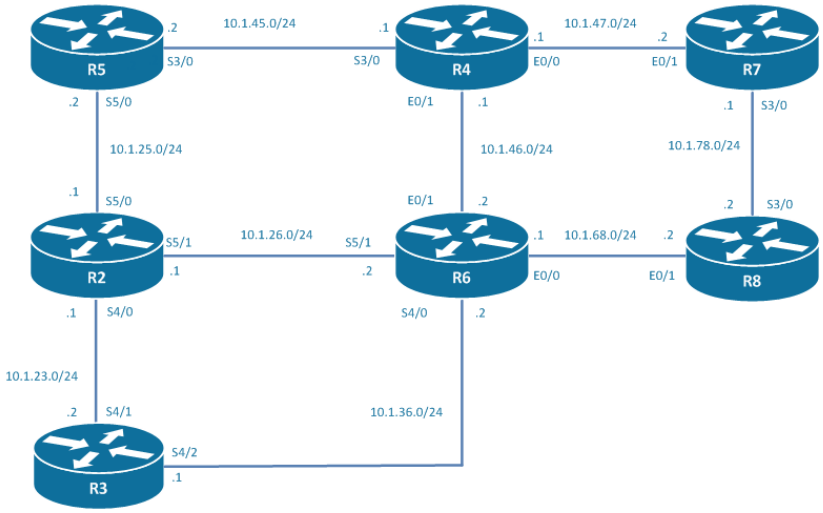
## Technologies covered

- Weight
- Local Preference
- As-path prepending
- Origin
- MED
- Always compare MED
- AS-path ignore
- Maximum AS Limit

## Overview

You have been tasked to configure the routing in your network using OSPF, EIGRP, RIP, static route, iBGP and eBGP.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 20.1** Configure an iBGP peering between R4 and R7 in AS 65001. Make sure that the 10.1.46.0/24 network and that the network 10.1.78.0/24 is carried in the BGP updates with an origin of i.

On R4, configure the following:

```
router bgp 65001
neighbor  10.1.47.2 remote-as 65001
network 10.1.46.0 mask 255.255.255.0
```

On R7, configure the following:

```
router bgp 65001
neighbor  10.1.47.1 remote-as 65001
network 10.1.78.0 mask 255.255.255.0
```

**Task 20.2** Configure an eBGP peering between R4 in AS 65001 and R6 in AS 65002.

On R4, configure the following:

```
router bgp 65001
neighbor  10.1.46.2 remote-as 65002
```

On R6, configure the following:

```
router bgp 65002
neighbor  10.1.46.1 remote-as 65001
```

**Task 20.3** Configure an eBGP peering between R6 in AS 65002 and R8 in AS 8.

On R6, configure the following:

```
router bgp 65002
neighbor  10.1.68.2 remote-as 8
```

On R8,configure the following:

```
router bgp 8
neighbor  10.1.68.1 remote-as 65002
```

**Task 20.4** Configure an eBGP peering between R8 in AS 8 and R7 in AS65001.

On R8, configure the following:

```
router bgp 8
neighbor  10.1.78.1 remote-as 65001
```

On R7, configure the following:

```
router bgp 65001
neighbor  10.1.78.2 remote-as 8
```

**Task 20.5** The loopback0 of R4 should be present in the BGP database with an origin attribute of incomplete. The loopback0 of R7 should be present in the BGP database with an origin attribute of internal.

On R4, configure the following:

```
route-map Loopback0_R4 permit 10
match interface loopback0

router bgp 65001
redistribute connected route-map  Loopback0_R4
```

On R7, configure the following:

```
router bgp 65001
network  10.1.7.7 mask 255.255.255.255
```

Let's check in the BGP database of R4 that the network has the origin attribute of incomplete.

```
R6#sh ip bgp
BGP table version is 4, local router ID is 10.22.6.6
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
 *   10.1.4.4/32      10.1.68.2                              0 8 65001 ?
 *>                   10.1.46.1                   0          0 65001 ?
 r   10.1.46.0/24     10.1.68.2                              0 8 65001 i
 r>                   10.1.46.1                   0          0 65001 i
 *   10.1.78.0/24     10.1.68.2                              0 8 65001 i
 *>                   10.1.46.1                              0 65001 i
```

**Task 20.6** On R8, manipulate the weight attribute so that the route to 10.1.4.4/32 is pointing towards R6. Use a prefix-list called WEIGHT_PL and a route-map called WEIGHT_RM.

Let's have a look at the BGP database on R8.

```
R8#sh ip bgp
BGP table version is 5, local router ID is 10.1.8.8
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
 *>  10.1.4.4/32      10.1.78.1                              0 65001 ?
 *                    10.1.68.1                              0 65002 65001 ?
 *>  10.1.46.0/24     10.1.78.1                              0 65001 i
 *                    10.1.68.1                              0 65002 65001 i
 r>  10.1.78.0/24     10.1.78.1                   0          0 65001 i
 r                    10.1.68.1                              0 65002 65001 i
```

On R8, configure the following:

```
ip prefix-list WEIGHT_PL seq 5 permit 10.1.4.4/32

route-map WEIGHT_RM permit 10
  match ip address prefix-list WEIGHT_PL
  set weight 50
route-map WEIGHT_RM permit 20

 router bgp 8
 neighbor 10.1.68.1 route-map WEIGHT_RM in
```

After applying the route-map, the BGP database looks the following:

```
R8#sh ip bgp
BGP table version is 4, local router ID is 10.1.8.8
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
 *   10.1.4.4/32      10.1.78.1                              0 65001 ?
 *>                   10.1.68.1                             50 65002 65001 ?
```

```
 *>  10.1.46.0/24     10.1.78.1                              0 65001 i
 *                    10.1.68.1                              0 65002 65001 i
 r>  10.1.78.0/24     10.1.78.1                0             0 65001 i
 r                    10.1.68.1                              0 65002 65001 i
```

**Task 20.7** The loopback0 of R6 should be present in the BGP database with an origin attribute of incomplete.

On R6, configure the following:

```
route-map Loopback0_R6 permit 10
match interface loopback0

router bgp 65002
redistribute connected route-map  Loopback0_R6
```

**Task 20.8** In order to reach the 10.1.6.6/32 loopback, the routers in AS 65001 should route the traffic over R8 through AS 8. Change the configuration in R7 and use a route-map called LOCALPRF_RM.

On R4, the only route to 10.1.6.6 is the direct route to R6.

```
R4#sh ip bgp
BGP table version is 5, local router ID is 10.11.4.4
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
 *>  10.1.4.4/32      0.0.0.0                  0         32768 ?
 *>  10.1.6.6/32      10.1.46.2                0             0 65002 ?
 *>  10.1.46.0/24     0.0.0.0                  0         32768 i
 *>i 10.1.78.0/24     10.1.47.2                0    100      0 i
```

On R7, configure the following:

```
access-list 1 permit 10.1.6.6

route-map LOCALPRF_RM permit 10
  match ip address 1
  set local-preference 200

route-map LOCALPRF_RM permit 20

router bgp 65001
neighbor 10.1.78.2 route-map LOCALPRF_RM in
```

The 10.1.6.6 network is advertised to the AS 65001 from the router R6 with a local-preference of 100 (the default local-preference) and from the router R8 with local-preference of 200.

**Task 20.9** The loopback0 of R8 should be present in the BGP database with an origin attribute of IGP.

On R8, configure the following:

```
router bgp 8
network 10.1.8.8 mask 255.255.255.255
```

**Task 20.10** Configure R8 so that the traffic originated on R6 is going through AS 65001 to reach the network 10.1.8.8/32. On R6, 10.1.8.8 should have the following AS-pat attribute 8 8 8 8 i. Use a prefix-list called PREPEND_PL and a route-map called PREPEND_RM.

Let's have a look at the BGP database on R6.

```
R6#sh ip bgp
BGP table version is 9, local router ID is 10.22.6.6
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
 *>  10.1.4.4/32      10.1.46.1                0             0 65001 ?
 *>  10.1.6.6/32      0.0.0.0                  0         32768 ?
 *   10.1.8.8/32      10.1.46.1                              0 65001 8 i
 *>                   10.1.68.2                0             0 8 i
 r   10.1.46.0/24     10.1.68.2                              0 8 65001 i
 r>                   10.1.46.1                0             0 65001 i
 *   10.1.78.0/24     10.1.68.2                              0 8 65001 i
 *>                   10.1.46.1                              0 65001 i
```

On R8, configure the following:

```
ip prefix-list Lo0_R8 seq 5 permit 10.1.8.8/32

route-map PREPEND_RM permit 10
  match ip address prefix-list Lo0_R8
  set as-path prepend 8 8 8

 route-map PREPEND_RM permit 20

router bgp 8
neighbor 10.1.68.1 route-map PREPEND_RM out
```

After prepending 3 AS in the direct route to 10.1.8.8, R6 will prefer to route through AS 65001 to reach 10.1.8.8.

```
R6#sh ip bgp
BGP table version is 16, local router ID is 10.22.6.6
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
 *>  10.1.4.4/32      10.1.46.1                0             0 65001 ?
 *>  10.1.6.6/32      0.0.0.0                  0         32768 ?
 *>  10.1.8.8/32      10.1.46.1                              0 65001 8 i
 *                   10.1.68.2                0             0 8 8 8 8 i
 r   10.1.46.0/24     10.1.68.2                              0 8 65001 i
 r>                   10.1.46.1                0             0 65001 i
 *   10.1.78.0/24     10.1.68.2                              0 8 65001 i
 *>                   10.1.46.1                              0 65001 i
```

**Task 20.11** Configure OSPF area 0 between R6 and R2.

On R6 and R2, configure the following:

```
router ospf 1
network 10.1.26.0 0.0.0.255 area 0
```

**Task 20.12** Configure iBGP connection between R6 and R2.

On R6, configure the following:

```
router bgp 65002
neighbor 10.1.26.1 remote-as 65002
```

On R2, configure the following:

```
router bgp 65002
neighbor 10.1.26.2 remote-as 65002
```

**Task 20.13** The loopback0 of R2 should be present in the BGP database with an origin attribute of incomplete.

On R2, configure the following

```
route-map Loopback0_R2 permit 10
match interface loopback0

router bgp 65002
redistribute connected route-map  Loopback0_R2
```

**Task 20.14** Configure an eBGP connection between R6 and R3 in AS 3. Redistribute the EBGP next-hop in OSPF area 0.

On R3, configure the following:

```
router bgp 3
neighbor  10.1.36.2 remote-as 65002
```

On R6, configure the following:

```
router bgp 65002
neighbor  10.1.36.1 remote-as 3

router ospf 1
network 10.1.36.0 0.0.0.255 area 0
```

**Task 20.15** Configure an eBGP connection between R2 and R3 in AS 3. Redistribute the EBGP next-hop in OSPF area 0.

On R3, configure the following:

```
router bgp 3
neighbor  10.1.23.1 remote-as 65002
```

On R2, configure the following:

```
router bgp 65002
neighbor  10.1.23.2 remote-as 3

router ospf 1
network 10.1.23.0 0.0.0.255 area 0
```

**Task 20.16** Advertise the loopback0 and the loopback1 of R3 using network statements.

On R3, configure the following:

```
router bgp 3
network 10.1.3.3 mask 255.255.255.255
network 10.11.3.0 mask 255.255.255.0
```

**Task 20.17** Ensure that the traffic is routed via R2 to reach network 10.1.3.3. Configure R3 and use the prefix-list called MED_PL 2 and a route-map called MED_RM2. Use a MED value of 200.

R6 is following the direct path towards R3 to reach 10.1.3.3.

```
R6#sh ip bgp
BGP table version is 19, local router ID is 10.22.6.6
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 10.1.2.2/32      10.1.26.1                0    100      0 ?
 * i 10.1.3.3/32      10.1.23.2                0    100      0 3 i
 *>                   10.1.36.1                0             0 3 i
 *>  10.1.4.4/32      10.1.46.1                0             0 65001 ?
 *>  10.1.6.6/32      0.0.0.0                  0         32768 ?
 *>  10.1.8.8/32      10.1.46.1                              0 65001 8 i
 *                    10.1.68.2                0             0 8 8 8 8 i
 r   10.1.46.0/24     10.1.68.2                              0 8 65001 i
 r>                   10.1.46.1                0             0 65001 i
 *   10.1.78.0/24     10.1.68.2                              0 8 65001 i
 *>                   10.1.46.1                              0 65001 i
 * i 10.11.3.0/24     10.1.23.2                0    100      0 3 i
 *>                   10.1.36.1                0             0 3 i
```

On R3, configure the following:

```
ip prefix-list MED_PL2 seq 5 permit 10.1.3.3/32

route-map MED_RM2 permit 10
  match ip address prefix-list MED_PL2
  set metric 300

route-map MED_RM2 permit 20

router bgp 3
neighbor 10.1.36.2 route-map MED_RM2 out
```

On R6, the traffic to 10.1.3.3 is now routed via R2.

```
R6#sh ip bgp
BGP table version is 26, local router ID is 10.22.6.6
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 10.1.2.2/32      10.1.26.1                0    100      0 ?
 *>i 10.1.3.3/32      10.1.23.2                0    100      0 3 i
 *                    10.1.36.1              300             0 3 i
 *>  10.1.4.4/32      10.1.46.1                0             0 65001 ?
 *>  10.1.6.6/32      0.0.0.0                  0         32768 ?
 *>  10.1.8.8/32      10.1.46.1                              0 65001 8 i
 *                    10.1.68.2                0             0 8 8 8 8 i
 r   10.1.46.0/24     10.1.68.2                              0 8 65001 i
 r>                   10.1.46.1                0             0 65001 i
 *   10.1.78.0/24     10.1.68.2                              0 8 65001 i
 *>                   10.1.46.1                              0 65001 i
 * i 10.11.3.0/24     10.1.23.2                0    100      0 3 i
 *>                   10.1.36.1                0             0 3 i
```

**Task 20.18** Ensure that the traffic is routed via R6 to reach network 10.11.3.3. Configure R3 and use the prefix-list called MED_PL 6 and a route-map called MED_RM6. Use a MED value of 300.

Let's check the way the traffic is routed from R2 to reach 10.11.3.3. The BGP update used for the routing is the eBGP update advertised from R3

```
R2#sh ip bgp
BGP table version is 11, local router ID is 10.11.2.2
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
 *>  10.1.2.2/32      0.0.0.0                  0         32768 ?
 *>  10.1.3.3/32      10.1.23.2                0             0 3 i
 * i 10.1.4.4/32      10.1.46.1                0    100      0 65001 ?
 *>i 10.1.6.6/32      10.1.26.2                0    100      0 ?
 * i 10.1.8.8/32      10.1.46.1                0    100      0 65001 8 i
 * i 10.1.46.0/24     10.1.46.1                0    100      0 65001 i
 * i 10.1.78.0/24     10.1.46.1                0    100      0 65001 i
 * i 10.11.3.0/24     10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
```

On R3, configure the following:

```
ip prefix-list MED_PL6 seq 5 permit 10.11.3.0/24

route-map MED_RM6 permit 10
  match ip address prefix-list MED_PL6
  set metric 300

 route-map MED_RM6 permit 20

  router bgp 3
  neighbor 10.1.23.1 route-map MED_RM6  out
```

R3 sends now a BGP update for the network 10.11.3.0 with a MED attribute to the router R2 via eBGP. This changes the routing on R2 and the path to 10.11.3.0 is now towards R6.

```
R2#sh ip bgp
BGP table version is 17, local router ID is 10.11.2.2
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
 *>  10.1.2.2/32      0.0.0.0                  0         32768 ?
 *>  10.1.3.3/32      10.1.23.2                0             0 3 i
 * i 10.1.4.4/32      10.1.46.1                0    100      0 65001 ?
 *>i 10.1.6.6/32      10.1.26.2                0    100      0 ?
 * i 10.1.8.8/32      10.1.46.1                0    100      0 65001 8 i
 * i 10.1.46.0/24     10.1.46.1                0    100      0 65001 i
 * i 10.1.78.0/24     10.1.46.1                0    100      0 65001 i
 *>i 10.11.3.0/24     10.1.36.1                0    100      0 3 i
 *                   10.1.23.2              300             0 3 i
```

**Task 20.19** In R2 and R6, advertise the network 10.1.26.0/24 with a network statement.

On R2 and R6, configure the following:

```
router bgp 65002
network 10.1.26.0 mask 255.255.255.0
```

**Task 20.20** On R3, modify the origin of the route 10.1.26.0/24 and ensure that this route is reached primarily through R6. Use a prefix-list called ORIGIN_PL and a route-map called ORIGIN_RM.

At this moment, R3 is routing towards R2 to reach 10.1.26.0/24.

```
R3#sh ip route 10.1.26.0
Routing entry for 10.1.26.0/24
  Known via "bgp 3", distance 20, metric 0
  Tag 65002, type external
  Last update from 10.1.23.1 00:01:22 ago
  Routing Descriptor Blocks:
  * 10.1.23.1, from 10.1.23.1, 00:01:22 ago
      Route metric is 0, traffic share count is 1
      AS Hops 1
      Route tag 65002
      MPLS label: none
```

For the update 10.1.26.0, all the attributes are the same for the update coming from R2 and the update coming from R6. The tie-breaker is going to be the router-id of the BGP neighbor. Between R2 and R6, R2 has the lowest BGP router-ID.

On R3, configure the following:

```
ip prefix-list ORIGIN_PL  seq 5 permit 10.1.26.0/24

route-map ORIGIN_RM permit 10
  match ip address prefix-list ORIGIN_PL
  set origin incomplete
 route-map ORIGIN_RM permit 20

router bgp 3
neighbor 10.1.23.1 route-map ORIGIN_RM in
```

As the internal origin is preferred over the incomplete origin, the routing from R3 to 10.1.26.0 is now going over R6.

```
R3#sh ip route 10.1.26.0
Routing entry for 10.1.26.0/24
  Known via "bgp 3", distance 20, metric 0
  Tag 65002, type external
  Last update from 10.1.36.2 00:01:14 ago
  Routing Descriptor Blocks:
  * 10.1.36.2, from 10.1.36.2, 00:01:14 ago
      Route metric is 0, traffic share count is 1
      AS Hops 1
      Route tag 65002
      MPLS label: none
```

**Task 20.21** On R6, advertise the network 10.22.6.0/24 using a network statement.

On R6, configure the following:

```
router bgp 65002
network 10.22.6.0 mask 255.255.255.0
```

**Task 20.22** This network should be advertised to the router R4 using the MED 500 and prepending one more AS in the AS-path. Use a prefix-list called ALWAYSCOMPMED_PL and a route-map called ALWAYSCOMPMED_RM.

On R6, configure the following:

```
ip prefix-list ALWAYSCOMPMED_PL seq 5 permit 10.22.6.0/24

route-map ALWAYSCOMPMED_RM permit 10
```

```
 match ip address prefix-list MED_PL6
 set metric 500
 set as-path prepend 65002

route-map ALWAYSCOMPMED_RM permit 20

 router bgp 65002
 neighbor 10.1.46.1 route-map ALWAYSCOMPMED_RM out
```

Even if the update sent from R6 to R4 for the route 10.22.6.0/24 has a metric of 500 and the AS-path attribute is the same, the preferred path is the path with the next-hop of 10.1.46.2 because MED is only compared for updates originated in the same AS.

```
R4#sh ip bgp
BGP table version is 83, local router ID is 10.11.4.4
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 10.1.2.2/32      10.1.78.2                0    100      0 8 65002 ?
 *>                   10.1.46.2              500             0 65002 65002 ?
 * i 10.1.3.3/32      10.1.78.2                0    100      0 8 65002 3 i
 *>                   10.1.46.2              500             0 65002 65002 3 i
 *>  10.1.4.4/32      0.0.0.0                  0         32768 ?
 *>i 10.1.6.6/32      10.1.78.2                0    200      0 8 65002 ?
 *                    10.1.46.2              500             0 65002 65002 ?
 *>i 10.1.8.8/32      10.1.78.2                0    100      0 8 i
 * i 10.1.26.0/24     10.1.78.2                0    100      0 8 65002 i
 *>                   10.1.46.2              500             0 65002 65002 i
 *>  10.1.46.0/24     0.0.0.0                  0         32768 i
 *>i 10.1.78.0/24     10.1.47.2                0    100      0 i
 * i 10.11.3.0/24     10.1.78.2                0    100      0 8 65002 3 i
 *>                   10.1.46.2              500             0 65002 65002 3 i
     Network          Next Hop            Metric LocPrf Weight Path
 * i 10.22.6.0/24     10.1.78.2                0    100      0 8 65002 i
 *>                   10.1.46.2              500             0 65002 65002 i
```

**Task 20.23** Configure R4 and ensure that R4 always prefers the route with the lowest MED, that is to say the route to R6 is pointing to R7 on R4.

On R4, configure the following:

```
router bgp 65001
bgp always-compare-med
```

Once it is configured, MED will always be compared and the preferred route to 10.22.6.0 is via R7 and R8.

```
R4#sh ip bgp
BGP table version is 16, local router ID is 10.11.4.4
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 10.1.2.2/32      10.1.78.2                0    100      0 8 65002 ?
 *                    10.1.46.2              500             0 65002 65002 ?
 *>i 10.1.3.3/32      10.1.78.2                0    100      0 8 65002 3 i
 *                    10.1.46.2              500             0 65002 65002 3 i
 *>i 10.1.6.6/32      10.1.78.2                0    200      0 8 65002 ?
 *                    10.1.46.2              500             0 65002 65002 ?
 *>i 10.1.8.8/32      10.1.78.2                0    100      0 8 i
```

```
 *                   10.1.46.2              500             0 65002 65002 8 8 8 8 i
 *>i 10.1.26.0/24     10.1.78.2                0    100      0 8 65002 i
 *                   10.1.46.2              500             0 65002 65002 i
 *>i 10.1.78.0/24     10.1.47.2                0    100      0 i
 *>i 10.11.3.0/24     10.1.78.2                0    100      0 8 65002 3 i
 *                   10.1.46.2              500             0 65002 65002 3 i
     Network          Next Hop            Metric LocPrf Weight Path
 *>i 10.22.6.0/24     10.1.78.2                0    100      0 8 65002 i
 *                   10.1.46.2              500             0 65002 65002 i
```

**Task 20.24** Configure an eBGP connection between R2 and R5 in AS 5 and between R4 and R5 in AS 5. Advertise the loopback of R5 into BGP with an origin of "?".

On R4, configure the following:

```
router bgp 65001
neighbor 10.1.45.2 remote-as 5
```

On R5, configure the following:

```
route-map Loopback0_R5 permit 10
match interface loopback0

router bgp 5
neighbor 10.1.45.1 remote-as 65001
neighbor 10.1.25.1 remote-as 65002
redistribute connected route-map  Loopback0_R5
```

On R2,configure the following:

```
router bgp 65002
neighbor 10.1.25.2 remote-as 5

router ospf 1
network 10.1.25.0 255.255.255.0 area 0
```

On R6, configure the following:

```
router ospf 1
network 10.1.46.0 255.255.255.0 area 0
```

**Task 20.25** On R4, prepend the AS 65001 4 times when advertising the network 10.1.7.7/32 to R5. The route from R5 to the loopback0 should now be transiting through AS 65002.

On R4, configure the following:

```
ip prefix-list Lo0_R7 seq 5 permit 10.1.7.7/32

route-map PREPEND_RM2 permit 10
  match ip address prefix-list Lo0_R7
  set as-path prepend 65002 65002 65002 65002

 route-map PREPEND_RM2 permit 20

router bgp 65001
neighbor 10.1.45.2 route-map PREPEND_RM2 out
```

Once the route-map is applied, the route from R5 to 10.1.7.7/32 is going via AS65002.

```
R5#sh ip bgp
BGP table version is 38, local router ID is 10.11.5.5
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
 *>  10.1.2.2/32      10.1.25.1                0             0 65002 ?
 *>  10.1.3.3/32      10.1.25.1                              0 65002 3 i
 *   10.1.4.4/32      10.1.25.1                              0 65002 65001 ?
 *>                   10.1.45.1                0             0 65001 ?
 *>  10.1.5.5/32      0.0.0.0                  0         32768 ?
 *   10.1.6.6/32      10.1.45.1                              0 65001 8 65002 ?
 *>                   10.1.25.1                              0 65002 ?
 *>  10.1.7.7/32      10.1.25.1                              0 65002 65001 i
 *                    10.1.45.1                              0 65001 65002 65002 65002
65002 i
 *   10.1.8.8/32      10.1.25.1                              0 65002 65001 8 i
 *>                   10.1.45.1                              0 65001 8 i
 *>  10.1.26.0/24     10.1.25.1                0             0 65002 i
 *   10.1.46.0/24     10.1.25.1                              0 65002 65001 i
     Network          Next Hop            Metric LocPrf Weight Path
 *>                   10.1.45.1                0             0 65001 i
 *   10.1.78.0/24     10.1.25.1                              0 65002 65001 i
 *>                   10.1.45.1                              0 65001 i
 *>  10.11.3.0/24     10.1.25.1                              0 65002 3 i
 *>  10.22.6.0/24     10.1.25.1                              0 65002 i
```

**Task 20.26** On R5, the AS-path attribute should be ignored and the route to the 10.1.7.7/32 network has to point towards R4 and not transit through AS 65002 anymore. Use MED to achieve this.

On R5, configure the following:

```
router bgp 5
bgp bestpath as-path ignore
```

On R2, configure the following:

```
ip prefix-list MED_PL seq 5 permit 10.1.7.7/24

route-map MED_RM permit 10
  match ip address prefix-list MED_PL
  set metric 500

 route-map MED_RM permit 20

router bgp 65002
neighbor 10.1.25.2 route-map MED_RM out
```

In the BGP selection process, the AS-path attribute is checked first before the MED attribute. As we tell the BGP process to skip the AS-path attribute comparison, it is the MED attribute that will be used to select the best route. The best route is now the route with the longest AS path attribute.

```
R5#sh ip bgp
BGP table version is 41, local router ID is 10.11.5.5
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
 *>  10.1.2.2/32      10.1.25.1                0             0 65002 ?
 *   10.1.3.3/32      10.1.25.1                              0 65002 3 i
 *>                   10.1.45.1                              0 65001 8 65002 3 i
 *   10.1.4.4/32      10.1.25.1                              0 65002 65001 ?
 *>                   10.1.45.1                0             0 65001 ?
 *>  10.1.5.5/32      0.0.0.0                  0         32768 ?
 *   10.1.6.6/32      10.1.25.1                              0 65002 ?
 *>                   10.1.45.1                              0 65001 8 65002 ?
 *   10.1.7.7/32      10.1.25.1                500           0 65002 65001 i
```

```
 *>                  10.1.45.1                              0 65001 65002 65002 65002
65002 i
 *   10.1.8.8/32     10.1.25.1                              0 65002 65001 8 i
 *>                  10.1.45.1                              0 65001 8 i
 *   10.1.26.0/24    10.1.25.1                0             0 65002 i
     Network         Next Hop            Metric LocPrf Weight Path
 *>                  10.1.45.1                              0 65001 8 65002 i
 *   10.1.46.0/24    10.1.25.1                              0 65002 65001 i
 *>                  10.1.45.1                0             0 65001 i
 *   10.1.78.0/24    10.1.25.1                              0 65002 65001 i
 *>                  10.1.45.1                              0 65001 i
 *   10.11.3.0/24    10.1.25.1                              0 65002 3 i
 *>                  10.1.45.1                              0 65001 8 65002 3 i
 *   10.22.6.0/24    10.1.25.1                              0 65002 i
 *>                  10.1.45.1                              0 65001 8 65002 i
```

**Task 20.27** On the peering between R2 and R5, shut down the peering if more than 50 BGP updates are advertised from R5 to R2. A syslog message should be sent when more than 40 BGP updates are advertised from R5 to R2.

On R2, configure the following:

```
router bgp 65002
neighbor 10.1.25.2 maximum-prefix 50 80
```

**You have completed Lab 20**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 21: Configure and troubleshoot BGP (part 4)

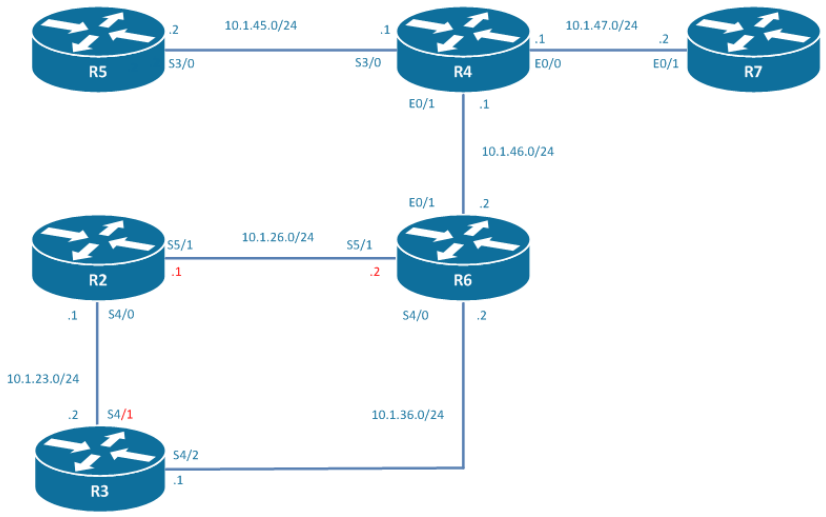
## Technologies covered

- Aggregation
- Summary-only
- Supress-map
- Unsupress-map
- AS-set
- Attribute-map
- Advertise-map
- Community no-export
- Community local-AS
- Community no-advertise

## Overview

You have been tasked to configure the routing in your network using OSPF, EIGRP, RIP, static route, iBGP and eBGP.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 21.1** Configure an iBGP peering between R2 and R6 in AS 65001.

On R2, configure the following:
```
router bgp 65001
neighbor 10.1.26.2 remote-as 65001
```

On R6, configure the following:
```
router bgp 65001
neighbor 10.1.26.1 remote-as 65001
```

**Task 21.2** Configure an eBGP peering between R3 in AS 3 and R6.

On R3, configure the following:
```
router bgp 3
neighbor 10.1.36.2 remote-as 65001
```

On R6, configure the following:
```
router bgp 65001
neighbor 10.1.36.1 remote-as 3
```

**Task 21.3** Configure an eBGP peering between R3 in AS 3 and R2.

On R3, configure the following:
```
router bgp 3
neighbor 10.1.23.1 remote-as 65001
```

On R2, configure the following:
```
router bgp 65001
neighbor 10.1.23.2 remote-as 3
```

**Task 21.4** R3 has to advertise a summary route representing the loopback1, loopback2, loopback3 and the loopback4 addresses of R3. The aggregate address command cannot be used.

As we cannot use the aggregate address command, we create a static route to null0.

On R3, configure the following:
```
ip route 33.3.0.0 255.255.252.0 Null0

router bgp 3
  network 33.3.0.0 mask 255.255.252.0
```

We can check on R2 that we have received the summary route 33.3.0.0/22.
```
R2#sh ip bgp
BGP table version is 2, local router ID is 10.11.2.2
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 33.3.0.0/22      10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
```

**Task 21.5** R3 has to advertise a summary route representing the loopback11, loopback12, loopback13 and the loopback14 addresses of R3. More specific networks should also be advertised. Use redistribution and a prefix-list with one single line.

In order to aggregate the networks, we should first redistribute the loopbacks11, 12, 13 and 14 into BGP.

On R3, configure the following:

```
ip prefix-list NET_133 seq 5 permit 133.0.0.0/14 le 16

route-map LOOPBACKS_133 permit 10
  match ip address prefix-list NET_133

 router bgp 3
 redistribute connected route-map LOOPBACKS_133
```

Now, we are able to create a summary route 133.0.0.0/14.

On R3, configure the following:

```
router bgp 3
  aggregate-address 133.0.0.0 255.252.0.0
```

We can check on R2 that we have received the summary route 133.0.0.0/14 and all the specific networks:

```
R2#sh ip bgp
BGP table version is 7, local router ID is 10.11.2.2
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 33.3.0.0/22      10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.0.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.0.0.0/14     10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.1.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.2.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.3.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
```

**Task 21.6** R3 has to advertise a summary route representing the loopback21, loopback22, loopback23 and the loopback24 addresses of R3. Specific subnets should not be advertised. Use network statements.

On R3, configure the following:

```
router bgp 3
network 143.3.0.0 mask 255.255.224.0
network 143.3.32.0 mask 255.255.224.0
network 143.3.64.0 mask 255.255.224.0
network 143.3.0.0 mask 255.255.0.0

aggregate-address 143.3.0.0 255.255.128.0 summary-only
```

On R2, only the summary route 143.3.0.0/16 has been advertised:

```
R2#sh ip bgp
BGP table version is 46, local router ID is 10.11.2.2
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 33.3.0.0/22      10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.0.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.0.0.0/14     10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.1.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.2.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.3.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 143.3.0.0        10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
```

**Task 21.7** In the addition to the summary route, loopback 21 network should be the only specific network advertised towards R2. Use an unsupress-map.

On R3, configure the following:

```
ip prefix-list UNSUPPRESS_PL_TOR2 seq 5 permit 143.3.0.0/19

 route-map UNSUPPRESS_RM_TOR2 permit 10
  match ip address prefix-list UNSUPPRESS_PL_TOR2

router bgp 3
  neighbor 10.1.23.1 unsuppress-map UNSUPPRESS_RM_TOR2
```

The expected result can be observed in the BGP database of R2.

```
R2#sh ip bgp
BGP table version is 61, local router ID is 10.11.2.2
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 33.3.0.0/22      10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.0.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.0.0.0/14     10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.1.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.2.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 * i 133.3.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 *>  143.3.0.0/19     10.1.23.2                0             0 3 i
 * i 143.3.0.0        10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
```

**Task 21.8** In the addition to the summary route, loopback 22 network should be the only specific network advertised towards R6. Use an unsupress-map.

On R3, configure the following:

```
ip prefix-list UNSUPPRESS_PL_TOR6 seq 5 permit 143.3.32.0/19

 route-map UNSUPPRESS_RM_TOR6 permit 10
  match ip address prefix-list UNSUPPRESS_PL_TOR6

router bgp 3
  neighbor 10.1.36.2 unsuppress-map UNSUPPRESS_RM_TOR6
```

On R6, you can check the expected result.

```
R6#sh ip bgp
BGP table version is 75, local router ID is 10.22.6.6
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 33.3.0.0/22      10.1.23.2                0    100      0 3 i
 *>                   10.1.36.1                0             0 3 i
 * i 133.0.0.0        10.1.23.2                0    100      0 3 ?
 *>                   10.1.36.1                0             0 3 ?
 * i 133.0.0.0/14     10.1.23.2                0    100      0 3 i
 *>                   10.1.36.1                0             0 3 i
 * i 133.1.0.0        10.1.23.2                0    100      0 3 ?
 *>                   10.1.36.1                0             0 3 ?
 * i 133.2.0.0        10.1.23.2                0    100      0 3 ?
 *>                   10.1.36.1                0             0 3 ?
 * i 133.3.0.0        10.1.23.2                0    100      0 3 ?
 *>                   10.1.36.1                0             0 3 ?
 * i 143.3.0.0/19     10.1.23.2                0    100      0 3 i
 * i 143.3.0.0        10.1.23.2                0    100      0 3 i
     Network          Next Hop            Metric LocPrf Weight Path
 *>                   10.1.36.1                0             0 3 i
 *>  143.3.32.0/19    10.1.36.1                0             0 3 i
```

**Task 21.9** In the addition to the summary route, loopback14 network should be the only specific network advertised towards R2. Use a suppress-map.

In order to use a suppress-map, you have to configure the aggregate-address first and then you can apply a suppress-map on it.

On R3, configure the following:

```
ip prefix-list SUPPRESS_PL seq 5 permit 133.0.0.0/16
ip prefix-list SUPPRESS_PL seq 10 permit 133.1.0.0/16
ip prefix-list SUPPRESS_PL seq 15 permit 133.2.0.0/16

route-map SUPPRESS_RM permit 10
  match ip address prefix-list SUPPRESS_PL

router bgp 3
network 133.0.0.0 mask 255.255.255.0
network 133.1.0.0 mask 255.255.255.0
network 133.2.0.0 mask 255.255.255.0
network 133.3.0.0 mask 255.255.255.0

aggregate-address 133.0.0.0 255.252.0.0 summary-only suppress-map SUPPRESS_RM
```

After resetting the BGP neigborship, we can see the expected result on the router R2.

```
R2#sh ip bgp
BGP table version is 6, local router ID is 10.11.2.2
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 33.3.0.0/22      10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.0.0.0/14     10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 133.3.0.0        10.1.36.1                0    100      0 3 ?
 *>                   10.1.23.2                0             0 3 ?
 *>  143.3.0.0/19     10.1.23.2                0             0 3 i
 * i 143.3.0.0        10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2                0             0 3 i
 * i 143.3.32.0/19    10.1.36.1                0    100      0 3 i
```

**Task 21.10** Configure an eBGP peering between R4 in AS 4 and R6 in AS 65001.

On R4, configure the following:

```
router bgp 4
neighbor 10.1.46.2 remote-as 65001
```

On R6, configure the following:

```
router bgp 65001
neighbor 10.1.46.1 remote-as 4
```

**Task 21.11** Configure an eBGP peering between R4 in AS 4 and R7 in AS 7. Advertise the network 200.1.1.0/24 into BGP using a network statement.

On R4, configure the following:

```
router bgp 4
neighbor 10.1.47.2 remote-as 7
```

On R7, configure the following:

```
router bgp 7
neighbor 10.1.47.1 remote-as 4
network 200.1.1.0 mask 255.255.255.0
```

**Task 21.12** Configure an eBGP peering between R4 in AS 4 and R5 in AS 5. Advertise the network 200.2.1.0/24 into BGP using a network statement.

On R4, configure the following:

```
router bgp 4
neighbor 10.1.45.2 remote-as 5
```

On R5, configure the following:

```
router bgp 5
neighbor 10.1.45.1 remote-as 4
network 200.2.1.0 mask 255.255.255.0
```

**Task 21.13** On R4, configure the aggregate 200.0.0.0/14. The more specific networks should not be advertised to R6. This aggregate should have in its AS-path attribute all the ASs that were contained in the AS-path attribute of the more specific networks.

On R4, I have 2 specific entries part of the range 200.x.x.x that has to be summarized in an aggregate route.

```
R4#sh ip bgp
BGP table version is 10, local router ID is 10.22.4.4
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
 *>  33.3.0.0/22      10.1.46.2                              0 65001 3 i
 *>  133.0.0.0/14     10.1.46.2                              0 65001 3 i
 *>  133.3.0.0        10.1.46.2                              0 65001 3 ?
 *>  143.3.0.0        10.1.46.2                              0 65001 3 i
 *>  143.3.32.0/19    10.1.46.2                              0 65001 3 i
 *>  200.1.1.0        10.1.47.2                   0          0 7 i
 *>  200.2.1.0        10.1.45.2                   0          0 5 i
```

On R4, configure the following:

```
router bgp 4
  aggregate-address 200.0.0.0 255.252.0.0 as-set summary-only
```

On R6, we can check that only the summary route 200.0.0.0/14 is advertised from R4.

```
R6#sh ip bgp
BGP table version is 151, local router ID is 10.22.6.6
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 33.3.0.0/22      10.1.23.2                0    100      0 3 i
 *>                   10.1.36.1                0             0 3 i
 *>  133.0.0.0/14     10.1.36.1                0             0 3 i
 * i                  10.1.23.2                0    100      0 3 i
 * i 133.3.0.0        10.1.23.2                0    100      0 3 ?
 *>                   10.1.36.1                0             0 3 ?
 * i 143.3.0.0/19     10.1.23.2                0    100      0 3 i
 *>  143.3.0.0        10.1.36.1                0             0 3 i
 * i                  10.1.23.2                0    100      0 3 i
 *>  143.3.32.0/19    10.1.36.1                0             0 3 i
 *>  200.0.0.0/14     10.1.46.1                0             0 4 {7,5} i
```

**Task 21.14** On R3, advertise the network 153.153.153.0/24 and 153.153.154.0/24 into BGP using network statements.

On R3, configure the following:

```
router bgp 3
network 153.153.153.0 mask 255.255.255.0
network 153.153.154.0 mask 255.255.255.0
```

**Task 21.15** On the peering's with R2 and R6, the network 153.153.153.0/24 has to be sent with the No-Export community. Use a route-map called NOEXPORT_RM.

On R3, configure the following:

```
ip prefix-list 153_153_153_0 seq 5 permit 153.153.153.0/24

 route-map NOEXPORT_RM permit 10
  match ip address prefix-list 153_153_153_0
  set community no-export
```

```
route-map NOEXPORT_RM permit 20

router bgp 3
 neighbor 10.1.23.1 route-map NOEXPORT_RM out
 neighbor 10.1.23.1 send-community
 neighbor 10.1.36.2 route-map NOEXPORT_RM out
 neighbor 10.1.36.2 send-community
```

On R2, we can check that the BGP advertisement from R3 has the community set to no-export. The BGP advertisement from R6 over the iBGP is set to inaccessible which is also right as the next-hop is unreachable.

```
R2#sh ip bgp 153.153.153.0
BGP routing table entry for 153.153.153.0/24, version 17
Paths: (2 available, best #2, table default, not advertised to EBGP peer)
  Advertised to update-groups:
     12
  Refresh Epoch 1
  3
    10.1.36.1 (inaccessible) from 10.1.26.2 (10.22.6.6)
      Origin IGP, metric 0, localpref 100, valid, internal
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  3
    10.1.23.2 from 10.1.23.2 (172.16.3.3)
      Origin IGP, metric 0, localpref 100, valid, external, best
      Community: no-export
      rx pathid: 0, tx pathid: 0x0
```

**Task 21.16** On R6, configure an aggregate for the network 153.153.152.0/22 with the summary-only and with the AS-SET option on.

On R6, configure the following:

```
router bgp 65001
  aggregate-address 153.153.152.0 255.255.252.0 as-set summary-only
```

**Task 21.17** Ensure that this aggregate is advertised to R4. Use a route-map called ATTRIBUTEMAP_RM.

Let's check the routing table of the BGP database on R4.

```
R4#sh ip bgp
BGP table version is 41, local router ID is 10.22.4.4
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
 *>  33.3.0.0/22      10.1.46.2                              0 65001 3 i
 *>  133.0.0.0/14     10.1.46.2                              0 65001 3 i
 *>  133.3.0.0        10.1.46.2                              0 65001 3 ?
 *>  143.3.0.0        10.1.46.2                              0 65001 3 i
 *>  143.3.32.0/19    10.1.46.2                              0 65001 3 i
 *>  200.0.0.0/14     0.0.0.0                          100  32768 {7,5} i
 s>  200.1.1.0        10.1.47.2                0             0 7 i
 s>  200.2.1.0        10.1.45.2                0             0 5 i
```

Because of the community NO-EXPORT set on the advertisement 153.153.153.0/24, the aggregate 153.153.152.0/22 has taken over this community attribute and the router R6 is not advertisement the aggregate 153.153.152.0/22 to the router R4 over the eBGP peering.

```
R6#sh ip bgp  153.153.152.0
```

```
BGP routing table entry for 153.153.152.0/22, version 181
Paths: (1 available, best #1, table default, not advertised to EBGP peer)
  Advertised to update-groups:
     10
  Refresh Epoch 1
  3, (aggregated by 65001 10.22.6.6)
    0.0.0.0 from 0.0.0.0 (10.22.6.6)
      Origin IGP, localpref 100, weight 32768, valid, aggregated, local, best
      Community: no-export
      rx pathid: 0, tx pathid: 0x0
```

On R6, configure the following:

```
route-map ATTRIBUTEMAP_RM permit 10
  set community none

router bgp 65001
aggregate-address 153.153.152.0 255.255.252.0 as-set summary-only attribute-map
ATTRIBUTEMAP_RM
```

After applying the attribute-map, the community NO-EXPORT is not attached anymore to the aggregate and the aggregate will be advertise to R4.

```
R4#sh ip bgp
BGP table version is 56, local router ID is 10.22.4.4
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
 *>  33.3.0.0/22      10.1.46.2                              0 65001 3 i
 *>  133.0.0.0/14     10.1.46.2                              0 65001 3 i
 *>  133.3.0.0        10.1.46.2                              0 65001 3 ?
 *>  143.3.0.0        10.1.46.2                              0 65001 3 i
 *>  143.3.32.0/19    10.1.46.2                              0 65001 3 i
 *>  153.153.152.0/22 10.1.46.2                0            0 65001 3 i
 *>  200.0.0.0/14     0.0.0.0                        100  32768 {7,5} i
 s>  200.1.1.0        10.1.47.2                 0            0 7 i
 s>  200.2.1.0        10.1.45.2                 0            0 5 i
```

**Task 21.18** On R5, advertise the network 200.200.0.0/16 and 200.201.0.0/16 into BGP using network statements.

On R5, configure the following:

```
router bgp 5
network 200.200.0.0 mask 255.255.0.0
network 200.201.0.0 mask 255.255.0.0
```

**Task 21.19** When advertising out the network 200.200.0.0/16 to R4, configure the community of no-advertise.

On R5, configure the following:

```
ip prefix-list LOOPBACKS seq 5 permit 200.200.0.0/16

route-map SET_COMMUNITY permit 10
  match ip address prefix-list LOOPBACKS
  set community no-advertise

 route-map SET_COMMUNITY permit 20

 router bgp 5
 neighbor 10.1.45.1 route-map SET_COMMUNITY out
```

```
 neighbor 10.1.45.1 send-community
```

After applying the route-map, the network 200.200.0.0/16 is carrying the communty no-export on R4.

```
R4#sh ip bgp 200.200.0.0
BGP routing table entry for 200.200.0.0/16, version 22
Paths: (1 available, best #1, table default, not advertised to any peer)
Flag: 0x880
  Advertised to update-groups: (Pending Update Generation)
     1
  Refresh Epoch 1
  5
    10.1.45.2 from 10.1.45.2 (200.201.1.5)
      Origin IGP, metric 0, localpref 100, valid, external, best
      Community: no-advertise
      rx pathid: 0, tx pathid: 0x0
```

**Task 21.20** On R4, configure an aggregate for the network 200.0.0.0/8 with the summary-only and with the AS-SET option on.

On R4, configure the following:

```
router bgp 4
  aggregate-address 200.0.0.0 255.0.0.0 as-set summary-only
```

**Task 21.21** Ensure that the network 200.0.0.0/8 will be advertised to R7 and R6. You are not allowed to use an attribute-map to remove the community. Use a route-map called ADVERTISEMAP_RM.

On R4, configure the following:

```
ip access-list standard TO_AGGREGATE
  deny   200.200.0.0 0.0.255.255
  permit 200.201.0.0 0.0.255.255

route-map ADVERTISEMAP_RM permit 10
  match ip address TO_AGGREGATE

router bgp 4
  aggregate-address 200.0.0.0 255.0.0.0 as-set summary-only advertise-map ADVERTISEMAP_RM
```

We can check that the aggregate 200.0.0.0/8 has been advertised to R6.

```
R6#sh ip bgp
BGP table version is 57, local router ID is 10.22.6.6
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
 *>  33.3.0.0/22      10.1.36.1                0             0 3 i
 * i                  10.1.23.2                0    100      0 3 i
 *>  133.0.0.0/14     10.1.36.1                0             0 3 i
 * i                  10.1.23.2                0    100      0 3 i
 *>  133.3.0.0        10.1.36.1                0             0 3 ?
 * i                  10.1.23.2                0    100      0 3 ?
 * i 143.3.0.0/19     10.1.23.2                0    100      0 3 i
 *>  143.3.0.0        10.1.36.1                0             0 3 i
 * i                  10.1.23.2                0    100      0 3 i
 *>  143.3.32.0/19    10.1.36.1                0             0 3 i
 *>  153.153.152.0/22 0.0.0.0                       100  32768 3 i
 s>  153.153.153.0/24 10.1.36.1                0             0 3 i
 s i                  10.1.23.2                0    100      0 3 i
```

```
 s>  153.153.154.0/24 10.1.36.1               0             0 3 i
     Network          Next Hop            Metric LocPrf Weight Path
 s i                  10.1.23.2                0    100      0 3 i
 *>  200.0.0.0/14     10.1.46.1                0             0 4 {7,5} i
 *>  200.0.0.0/8      10.1.46.1                0             0 4 5 i
```

**Task 21.22** On R4, advertise the network 10.22.4.0/24 into BGP using a network statement.

On R4, configure the following:

```
router bgp 4
network 10.22.4.0 mask 255.255.255.0
```

**Task 21.23** Ensure that the network 10.22.4.0 will be advertised to R6 with a community that will prevent it to be advertised to other eBGP peers.

On R4, configure the following:

```
ip prefix-list 10_22_4_0 seq 5 permit 10.22.4.0/24

route-map SET_COMMUNITY permit 10
 match ip address prefix-list 10_22_4_0
 set community no-export

route-map SET_COMMUNITY permit 20

router bgp 4
 neighbor 10.1.46.2 route-map SET_COMMUNITY out
 neighbor 10.1.46.2 send-community
```

The BGP update 10.22.4.0/24 has the community NO-EXPORT set and will therefore not be advertised to other eBGP peers.

```
R6#sh ip bgp 10.22.4.0
BGP routing table entry for 10.22.4.0/24, version 68
Paths: (1 available, best #1, table default, not advertised to EBGP peer)
  Advertised to update-groups:
     12
  Refresh Epoch 1
  4
    10.1.46.1 from 10.1.46.1 (10.22.4.4)
      Origin IGP, metric 0, localpref 100, valid, external, best
      Community: no-export
      rx pathid: 0, tx pathid: 0x0
```

**You have completed Lab 21**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 22: Configure and troubleshoot BGP (part 5)

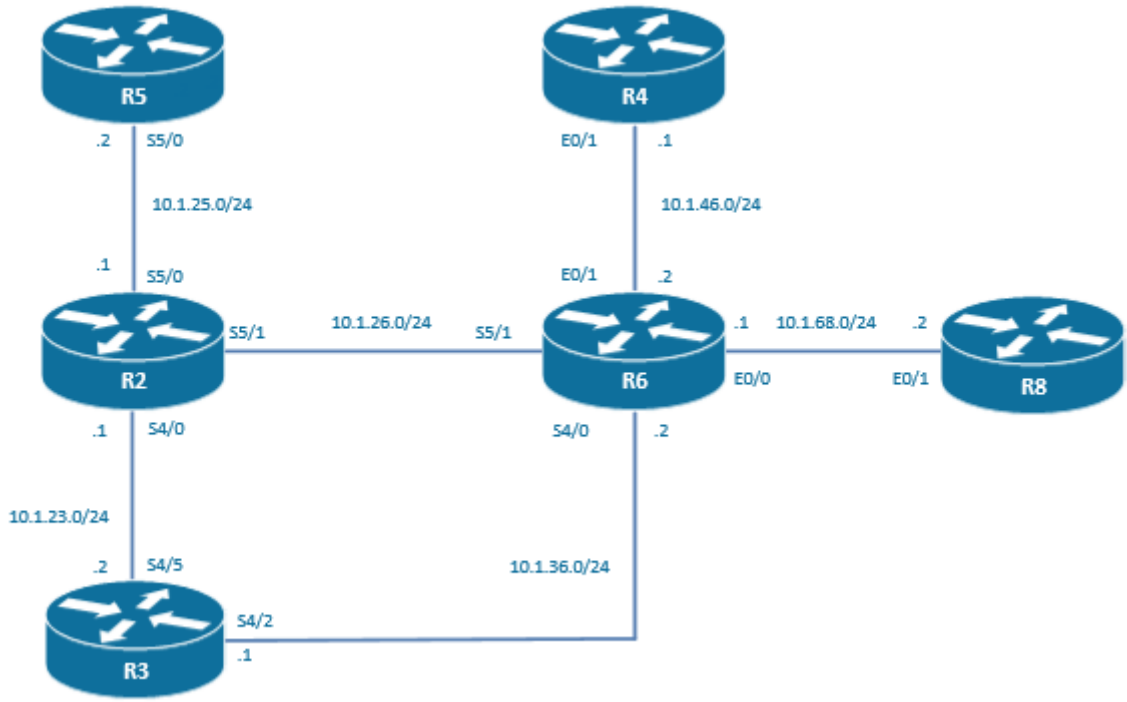
## Technologies covered

- Local AS
- Replace AS
- Dual AS
- Remove Private AS
- Dampening
- ORF
- BGP allowas in

## Overview

You have been tasked to configure the routing in your network using OSPF, EIGRP, RIP, static route, iBGP and eBGP.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

### Task 22.1 Configure an iBGP peering in AS 65001 between R2 and R6.

On R2, configure the following:

```
router bgp 65001
neighbor 10.1.26.2 remote-as 65001
```

On R6, configure the following:

```
router bgp 65001
neighbor 10.1.26.1 remote-as 65001
```

### Task 22.2 Configure an eBGP peering between R6 and R3 in AS 3.

On R6, configure the following:

```
router bgp 65001
neighbor 10.1.36.1 remote-as 3
```

On R3, configure the following:

```
router bgp 3
neighbor 10.1.36.2 remote-as 65001
```

### Task 22.3 Configure an eBGP peering between R2 and R3 in AS 3.

On R2, configure the following:

```
router bgp 65001
neighbor 10.1.23.2 remote-as 3
```

On R3, configure the following:

```
router bgp 3
neighbor 10.1.23.1 remote-as 65001
```

### Task 22.4 On R3, redistribute the network 153.153.153.0/24 and the network 153.153.154.0/24 using network statements.

On R3, configure the following:

```
router bgp 3
network 153.153.153.0 mask 255.255.255.0
network 153.153.154.0 mask 255.255.255.0
```

### Task 22.5 On R3, on the peering between R3 and R2, filter out 153.153.153.0/24. Use prefix-list.

Let's have a look at the BGP table on R2. I have greyed the update that has to be filtered out.

```
R2#sh ip bgp
BGP table version is 3, local router ID is 10.11.2.2
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 153.153.153.0/24 10.1.36.1                0    100      0 3 i
 *>                   10.1.23.2               0             0 3 i
 * i 153.153.154.0/24 10.1.36.1                0    100      0 3 i
```

```
 *>                  10.1.23.2               0             0 3 i
```

On R3, configure the following:

```
ip prefix-list DENY_153 seq 10 deny 153.153.153.0/24
ip prefix-list DENY_153 seq 15 permit 0.0.0.0/0 le 32

router bgp 3
  neighbor 10.1.23.1 prefix-list DENY_153 out
```

After that the peering neighborship has been cleared, the BGP database of R2 looks like that:

```
R2#sh ip bgp
BGP table version is 6, local router ID is 10.11.2.2
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 153.153.153.0/24 10.1.36.1                0    100      0 3 i
 * i 153.153.154.0/24 10.1.36.1                0    100      0 3 i
 *>                  10.1.23.2               0             0 3 i
```

**Task 22.6** On R3, on the peering between R3 and R6, filter out 153.153.154.0/24. Use access-list.

Let's have a look at the BGP table on R6. I have greyed the update that has to be filtered out.

```
R6#sh ip bgp
BGP table version is 7, local router ID is 10.22.6.6
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
 *>  153.153.153.0/24 10.1.36.1               0             0 3 i
 * i 153.153.154.0/24 10.1.23.2               0    100      0 3 i
 *>                  10.1.36.1               0             0 3 i
```

On R3, configure the following:

```
access-list 1 permit 153.153.154.0 0.0.0.255

route-map FILTER_154 deny 10
  match ip address 1

route-map FILTER_154 permit 20

router bgp 3
  neighbor 10.1.36.2 route-map FILTER_154 out
```

After that the peering neighborship has been cleared, the BGP database of R6 looks like that:

```
R6#sh ip bgp
BGP table version is 10, local router ID is 10.22.6.6
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
```

```
 *>  153.153.153.0/24 10.1.36.1               0             0 3 i
 * i 153.153.154.0/24 10.1.23.2               0    100      0 3 i
```

**Task 22.7** R3 should appear to R2 and R6 as if it is using AS 65003 but R3 should still be in AS 3.

On R2, configure the following:

```
router bgp 65001
no neighbor 10.1.23.2 remote-as 3
neighbor 10.1.23.2 remote-as 65003
```

On R6, configure the following:

```
router bgp 65001
no neighbor 10.1.36.1 remote-as 3
neighbor 10.1.36.1 remote-as 65003
```

On R3, configure the following:

```
router bgp 3
neighbor 10.1.23.1 local-as 65003
neighbor 10.1.36.2 local-as 65003
```

After the changes of neighbor AS on R2 and R6, the peering's on R3 are again established.

```
R3#sh ip bgp sum
BGP router identifier 153.153.154.3, local AS number 3
BGP table version is 3, main routing table version 3
2 network entries using 280 bytes of memory
2 path entries using 160 bytes of memory
1/1 BGP path/bestpath attribute 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 584 total bytes of memory
BGP activity 8/6 prefixes, 8/6 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.23.1       4        65001       4       5        3    0    0 00:00:16        0
10.1.36.2       4        65001       6       7        3    0    0 00:02:06        0
```

**Task 22.8** Regarding the routes advertised from R3 to R2, the AS 65003 should not appear in the AS-path.

On R2, in the BGP table, we can see that the 2 Autonomous systems 3 and 65003 are advertised in the BGP updates.

```
R2#sh ip bgp
BGP table version is 12, local router ID is 10.11.2.2
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 153.153.153.0/24 10.1.36.1                0    100      0 65003 3 i
 *>  153.153.154.0/24 10.1.23.2                0             0 65003 3 i
```

On R3, configure the following:

```
router bgp 3
neighbor 10.1.23.1 local-as 65003 no-prepend
```

**Task 22.9** Configure an eBGP peering between R2 and R5 in AS 5.

On R2, configure the following:

```
router bgp 65001
neighbor 10.1.25.2 remote-as 5
```

On R5, configure the following:

```
router bgp 5
neighbor 10.1.25.1 remote-as 65001
```

**Task 22.10** R5 should appear to R2 and R6 as if it is using AS 65005 but R5 should still be in AS 5.

On R2, configure the following:

```
router bgp 65001
no neighbor 10.1.25.2 remote-as 5
neighbor 10.1.25.2 remote-as 65005
```

On R5, configure the following:

```
router bgp 5
neighbor 10.1.25.1 local-as 65005
```

**Task 22.11** Advertise the loopback0 of R5 into BGP. Regarding the routes advertised from R5 to R2, the AS 65005 should not appear in the AS-path.

On R5, configure the following:

```
router bgp 5
network 10.1.5.5 mask 255.255.255.255
```

In the routing table of BGP of R2, the loopback od R5 is advertised with the AS path 65005 5 i

```
R2#sh ip bgp
BGP table version is 9, local router ID is 10.11.2.2
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
 *>  10.1.5.5/32      10.1.25.2                0             0 65005 5 i
 * i 153.153.153.0/24 10.1.36.1                0    100      0 65003 3 i
 *>  153.153.154.0/24 10.1.23.2                0             0 65003 3 i
```

On R5, configure the following:

```
router bgp 5
neighbor 10.1.25.1 local-as 65005 no-prepend
```

**Task 22.12** On R5, the route 153.153.154.0/24 should not contain the AS 3 as well as the AS 65003 in the AS-path.

The BGP database on R5 looks like the following:

```
R5#sh ip bgp
BGP table version is 5, local router ID is 10.11.5.5
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
 *>  10.1.5.5/32      0.0.0.0                  0         32768 i
 *>  153.153.154.0/24 10.1.25.1                              0 65001 65003 3 i
```

On R3, configure the following:

```
router bgp 3
neighbor 10.1.23.1 local-as 65003 no-prepend replace-as
```

After applying this change on R3, The BGP database on R5 looks like expected, the AS3 have been removed from the AS-path.

```
R5#sh ip bgp
BGP table version is 5, local router ID is 10.11.5.5
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
 *>  10.1.5.5/32      0.0.0.0                  0         32768 i
 *>  153.153.154.0/24 10.1.25.1                              0 65001 65003 i
```

**Task 22.13** Configure an eBGP peering between R6 and R4 in AS 4.

On R6, configure the following:

```
router bgp 65001
neighbor 10.1.46.1 remote-as 4
```

On R4, configure the following:

```
router bgp 4
neighbor 10.1.46.2 remote-as 65001
```

**Task 22.14** On R6, in all advertisements sent towards R4, the private AS numbers have to be stripped off from the AS-path before being sent.

On R4, the BGP database looks like that:

```
R4#sh ip bgp
BGP table version is 1, local router ID is 10.11.4.4
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
 *   153.153.153.0/24 10.1.46.2                               0 65001 65003 3 i
```

On R6, configure the following:

```
router bgp 65001
neighbor 10.1.46.1 remove-private-as
```

**Task 22.15** On R3, configure the 153.153.153.0/24 network to use the following dampening parameters:

- Max-Suppress=60 minutes
- Suppress=2000 points
- Reuse=800 points
- Half-Time=15 minutes

On R3, configure the following:

```
ip prefix-list NET_153 seq 5 permit 153.153.153.0/24

route-map DAMPENING permit 10
  match ip address prefix-list NET_153
  set dampening 15 800 2000 60

 router bgp 3
  bgp dampening route-map DAMPENING
```

**Task 22.16** On R3, configure the 153.153.154.0/24 network to use the following dampening parameters:

- Max-Suppress=50 minutes
- Suppress=2500 points
- Reuse=600 points
- Half-Time=10 minutes

On R3, configure the following:

```
ip prefix-list NET_154 seq 5 permit 153.153.154.0/24

route-map DAMPENING permit 20
  match ip address prefix-list NET_154
  set dampening 10 600 2500 50
```

**Task 22.17** Between R6 and R4, configure the BGP peering to use fast session deactivation.

On R6, configure the following:

```
router bgp 65001
neighbor 10.1.46.1 fall-over
```

On R4, configure the following:

```
router bgp 4
neighbor 10.1.46.2 fall-over
```

**Task 22.18** On R4, advertise the loopbacks in BGP using network statements.

On R4, configure the following:

```
router bgp 4
network 10.1.4.4 mask 255.255.255.255
network 10.11.4.0 mask 255.255.255.0
```

**Task 22.19** On R6, filter in the network 10.11.4.0/24 on the peering towards R4.

10.1.4.4/32 and 10.11.4.0/24 has been advertised to R6 via BGP.

```
R6#sh ip bgp
BGP table version is 4, local router ID is 10.22.6.6
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
 *>  10.1.4.4/32      10.1.46.1                0             0 4 i
```

```
 * i 10.1.5.5/32      10.1.25.2                0    100      0 65005 5 i
 *>  10.11.4.0/24     10.1.46.1                0             0 4 i
 *>  153.153.153.0/24 10.1.36.1                0             0 65003 3 i
 * i 153.153.154.0/24 10.1.23.2                0    100      0 65003 i
```

On R6, configure the following:

```
ip prefix-list NET_11 seq 5 deny 10.11.4.0/24
ip prefix-list NET_11 seq 15 permit 0.0.0.0/0 le 32

router bgp 65001
 neighbor 10.1.46.1 prefix-list NET_11 in
```

After applying the filter, the 10.11.4.0/24 network is not in the BGP database of R6 anymore.

```
R6#sh ip bgp
BGP table version is 3, local router ID is 10.22.6.6
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
 *>  10.1.4.4/32      10.1.46.1                0             0 4 i
 * i 10.1.5.5/32      10.1.25.2                0    100      0 65005 5 i
 *>  153.153.153.0/24 10.1.36.1                0             0 65003 3 i
 * i 153.153.154.0/24 10.1.23.2                0    100      0 65003 i
```

**Task 22.20** Make sure that the two routers exchange information via the ORF capability and that R4 will be filtering the network 10.11.4.0/24 and not sending updates for networks that are filtered when arriving on R6.

On R6, configure the following:

```
router bgp 65001
 neighbor 10.1.46.1 capability orf prefix-list receive
```

On R4, configure the following:

```
router bgp 4
 neighbor 10.1.46.2 capability orf prefix-list send
```

**Task 22.21** Configure an eBGP peering between R6 and R8 in AS 4.

On R6, configure the following:

```
router bgp 65001
neighbor 10.1.68.2 remote-as 4
```

On R8, configure the following:

```
router bgp 4
neighbor 10.1.68.1 remote-as 65001
```

**Task 22.22** On R8, advertise the loopback0 into BGP using a network statement.

On R8, configure the following:

```
router bgp 4
network 10.1.8.8 mask 255.255.255.255
```

**Task 22.23** Make sure that you can ping from the loopback0 of R8 which is originated in AS 4 to the loopback0 of R4 which is always originated in AS 4. Use the allowas-in command.

Because R4 and R8 are all part of AS 4, the updates originated from AS4 will not be advertised with eBGP back to AS4. As a matter of fact, the loopback0 of R8 will not be advertised to R4 and the loopback0 of R4 will not be advertised to R8.

On R8, configure the following:

```
router bgp 4
neighbor 10.1.68.1 allowas-in 1
```

On R4, configure the following:

```
router bgp 4
neighbor 10.1.46.2 allowas-in 1
```

By configuring the allowas-in 1, we are tweaking BGP speakers to accept updates that have already once their own AS number in the AS path.

I am now able to ping from the loopback0 of R8 to the loopback0 of R4.

```
R8#ping 10.1.4.4 source 10.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 10.1.8.8
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**You have completed Lab 22**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 23: Configure and troubleshoot Multiprotocol Label Switching (Part 1)

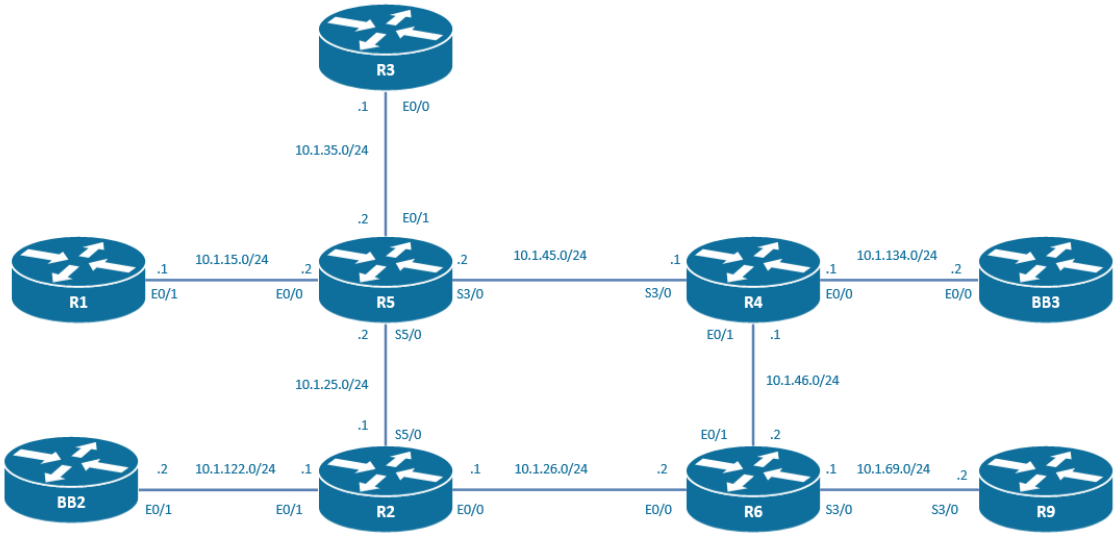
## Technologies covered

- IPv4 VPN address-family
- LSP
- LDP
- L3VPN
- CE
- PE
- P
- Export map

## Overview

You have been tasked to configure a MPLS L3 VPN service on an existing MPLS backbone. The CEs are managed by the Service Provider and the loopbacks of the CEs should be leaked from the VRF of the customer into the management VRF of the Service provider.

The topology used in the lab will be the following:




**Estimated time to complete: 3-4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 23.1** The network is pre-configured with OSPF and LDP and the PEs are the R5, R4, R6, and R2 routers. In order to optimize the building of the MPLS forwarding-table, make sure that only LSPs for the loopback interfaces will be built.

By default, LDP will create a Label Switched path for all the not local entries in the routing table. Let's have a look at the current LDP forwarding table of R6:

```
R6#sh mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  10.1.2.2/32      1008          Et0/0      10.1.26.1
17         Pop Label  10.1.4.4/32      1008          Et0/1      10.1.46.1
18         18         10.1.5.5/32      0             Et0/0      10.1.26.1
           16         10.1.5.5/32      0             Et0/1      10.1.46.1
19         Pop Label  10.1.25.0/24     0             Et0/0      10.1.26.1
20         Pop Label  10.1.45.0/24     0             Et0/1      10.1.46.1
```

Creating label paths for networks that are not loopbacks like 10.1.25.0/24 and 10.1.45.0/24 networks is not necessary. The M-BGP peering's will use the loopback0 as source of the peering's. Those peering's will be configured in a next question.

Therefore, we can configure LDP only to build LSPs for the loopback of the PEs:

On all the PEs, that is to say R2, R4, R5, and R6, configure the following:

```
ip access-list standard LOOPBACKS
permit 10.1.5.5
permit 10.1.2.2
permit 10.1.6.6
permit 10.1.4.4
no mpls ldp advertise-labels
mpls ldp advertise-labels for LOOPBACKS
```

Let's check the MPLS forwarding table on R6 after those changes:

```
R6#sh mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  10.1.2.2/32      1424          Et0/0      10.1.26.1
17         Pop Label  10.1.4.4/32      1248          Et0/1      10.1.46.1
18         18         10.1.5.5/32      0             Et0/0      10.1.26.1
           16         10.1.5.5/32      0             Et0/1      10.1.46.1
19         No Label   10.1.25.0/24     0             Et0/0      10.1.26.1
20         No Label   10.1.45.0/24     0             Et0/1      10.1.46.1
```

Instead of a Pop Outgoing Label imposed for the destination of 10.1.25.0/24 and 10.1.45.0/24, there is now a No Label for those networks meaning no LSPs will be built for those destinations.

**Task 23.2** Configure the following L3 MPLS VPN routing tables on the R5 and on the R6:

| AS | VPN name | rd | rt export | rt import |
|---|---|---|---|---|
| 1 | Customer_A | 1 | 10 | 10 |
| 1 | Customer_B | 2 | 20 | 20 |

On R5 and R6, configure the following:

```
ip vrf Customer_A
rd 1:1
route-target export 1:10
route-target import 1:10
```

```
ip vrf Customer_B
rd 1:2
route-target export 1:20
route-target import 1:20
```

**Task 23.3** Configure the following loopbacks for the VPN Customer_A and Customer_B.

| | | | |
|---|---|---|---|
| R5 | Loopback10 | 10.10.5.5/32 | Customer_A |
| R5 | Loopback20 | 10.20.5.5/32 | Customer_B |
| R6 | Loopback10 | 10.10.6.6/32 | Customer_A |
| R6 | Loopback20 | 10.20.6.6/32 | Customer_B |

On R5, configure the following:

```
interface Loopback10
ip vrf forwarding Customer_A
ip address 10.10.5.5 255.255.255.0

interface Loopback20
ip vrf forwarding Customer_B
ip address 10.20.5.5 255.255.255.0
```

On R6, configure the following:

```
interface Loopback10
ip vrf forwarding Customer_A
ip address 10.10.6.6 255.255.255.0

interface Loopback20
ip vrf forwarding Customer_B
ip address 10.20.6.6 255.255.255.0
```

**Task 23.4** Configure the BGP routing sessions that will permit to exchange the VPNv4 information between the PEs. Use BGP AS 1.

We have to configure VPNv4 support between R5 and R6. Therefore, we have to configure first an iBGP peering between R5 and R6. In a previous question, we have limited the LSP creation to the loopback0 of PEs so remember to use the loopback0 as source of the iBGP peering's.
In order to have BGP to carry the VPNv4 extensions, we have to create a VPNv4 address-family and to make sure that the extended community attribute is transmitted on the iBGP peering's. The extended community attribute is used to carry the RT information.

On R5, configure the following:

```
router bgp 1
bgp log-neighbor-changes
neighbor 10.1.6.6 remote-as 1
neighbor 10.1.6.6 update-source Loopback0



address-family vpnv4
neighbor 10.1.6.6 activate
neighbor 10.1.6.6 send-community both
exit-address-family
```

On R6, configure the following:

```
router bgp 1
bgp log-neighbor-changes
```

```
neighbor 10.1.5.5 remote-as 1
neighbor 10.1.5.5 update-source Loopback0

address-family vpnv4
neighbor 10.1.5.5 activate
neighbor 10.1.5.5 send-community both
exit-address-family
```

**Task 23.5** Redistribute the loopbacks created in Task 23.3 in their respective VPNs and check that you can ping from loopback to loopback within the same VPN.

On R5, configure the following:

```
router bgp 1
address-family ipv4 vrf Customer_A
redistribute connected
exit-address-family
address-family ipv4 vrf Customer_B
redistribute connected
exit-address-family
```

On R6, configure the following:

```
router bgp 1
address-family ipv4 vrf Customer_A
redistribute connected
exit-address-family
address-family ipv4 vrf Customer_B
redistribute connected
exit-address-family
```

On R5, let's try to ping the loopback20 of R6 from the loopback20 of R5. Loopbacks 20 are part of VRF Customer_B.

```
R5#ping vrf Customer_B 10.20.6.6 source 10.20.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.20.6.6, timeout is 2 seconds:
Packet sent with a source address of 10.20.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 23.6** Make sure that the loopbacks redistributed at PE router R5 has a known origin.

By default, redistributed routes have an origin of incomplete. Incomplete origin is indicated as? in the AS-path.

```
R5#sh ip bgp vpnv4 all
BGP table version is 7, local router ID is 10.1.5.5
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: 1:1 (default for vrf Customer_A)
 *>  10.10.5.0/24     0.0.0.0                  0         32768 ?
 *>i 10.10.6.0/24     10.1.6.6                 0    100      0 ?
Route Distinguisher: 1:2 (default for vrf Customer_B)
 *>  10.20.5.0/24     0.0.0.0                  0         32768 ?
 *>i 10.20.6.0/24     10.1.6.6                 0    100      0 ?
```

Let's configure a route-map on R5 and change the origin of the redistributed routes to iGP (indicated with an "i" in the AS-path)

On R5, configure the following:

```
route-map ORIGIN permit 10
 set origin igp

router bgp 1

 address-family ipv4 vrf Customer_A
  redistribute connected route-map ORIGIN
 exit-address-family
 !
 address-family ipv4 vrf Customer_B
  redistribute connected route-map ORIGIN
 exit-address-family
```

Let's check the BGP database once the route-map has been applied:

```
R5#sh ip bgp vpnv4 all
BGP table version is 9, local router ID is 10.1.5.5
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: 1:1 (default for vrf Customer_A)
 *>  10.10.5.0/24     0.0.0.0                  0         32768 i
 *>i 10.10.6.0/24     10.1.6.6                 0    100      0 ?
Route Distinguisher: 1:2 (default for vrf Customer_B)
 *>  10.20.5.0/24     0.0.0.0                  0         32768 i
 *>i 10.20.6.0/24     10.1.6.6                 0    100      0 ?
```

### Task 23.7 Customer_ A and Customer _B companies are merging.

We want to enable route exchange between Customer_A VRF and Customer_B VRF. This is easily accomplished by configuring an additional import RT to each customer VRF:

On R5 and R6, configure the following:

```
ip vrf Customer_A
route-target import 1:20
ip vrf Customer_B
route-target import 1:10
```

We have merged the BGP database and the routing table of the VRF Customer_A and Customer_B, in fact making Customer_A and Customer_B one single VPN.

```
R5#sh ip bgp vpnv4 all
BGP table version is 13, local router ID is 10.1.5.5
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: 1:1 (default for vrf Customer_A)
 *>  10.10.5.0/24     0.0.0.0                  0         32768 i
 *>i 10.10.6.0/24     10.1.6.6                 0    100      0 ?
 *>  10.20.5.0/24     0.0.0.0                  0         32768 i
 *>i 10.20.6.0/24     10.1.6.6                 0    100      0 ?
Route Distinguisher: 1:2 (default for vrf Customer_B)
 *>  10.10.5.0/24     0.0.0.0                  0         32768 i
```

```
 *>i 10.10.6.0/24     10.1.6.6                 0    100      0 ?
 *>  10.20.5.0/24     0.0.0.0                  0         32768 i
 *>i 10.20.6.0/24     10.1.6.6                 0    100      0 ?

R5#sh ip route vrf Customer_A

Routing Table: Customer_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
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
C        10.10.5.0/24 is directly connected, Loopback10
L        10.10.5.5/32 is directly connected, Loopback10
B        10.10.6.0/24 [200/0] via 10.1.6.6, 01:27:31
B        10.20.5.0/24 is directly connected (Customer_B), 00:01:24, Loopback20
L        10.20.5.5/32 is directly connected, Loopback20
B        10.20.6.0/24 [200/0] via 10.1.6.6, 00:01:24
```

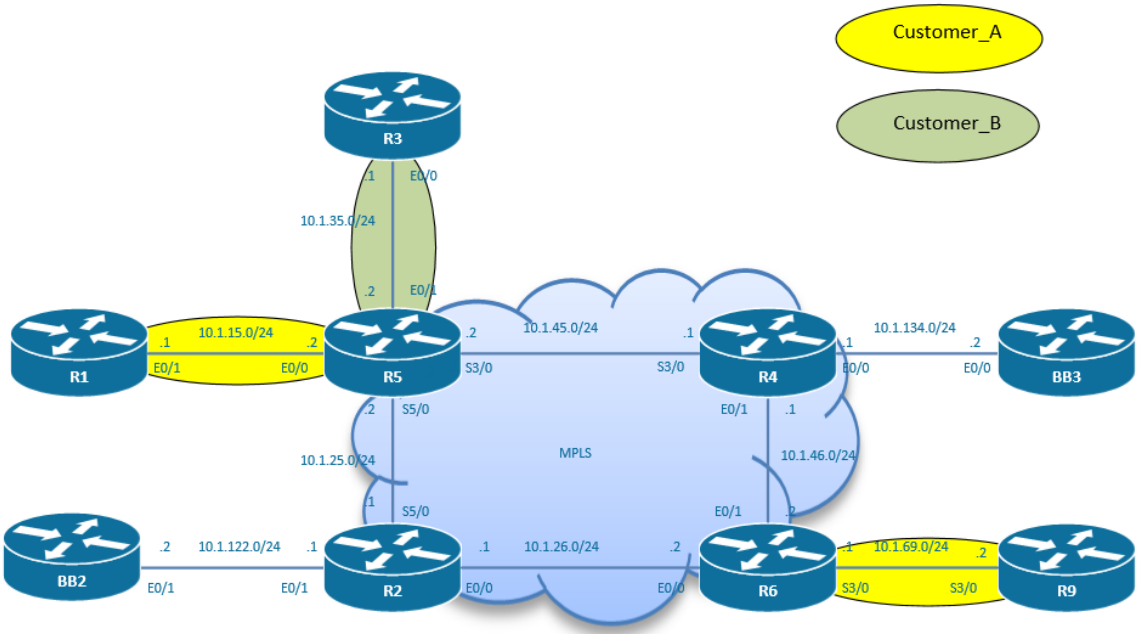
**Task 23.8** The engineer was too quick and the merge between Customer_ A and Customer_B is not going ahead.

We have to un-merge the companies.

On R5 and R6, configure the following:

```
ip vrf Customer_A
no route-target import 1:20
ip vrf Customer_B
no route-target import 1:10
```

**Task 23.9** Configure R1 and R9 to be part of VRF Customer_A and R3 to be part of VRF Customer_B.



Configure the following loopbacks:

| | |
|---|---|
| R1 loopback0 | 10.1.1.1/32 |
| R9 loopback0 | 10.1.9.9/32 |
| R3 loopback0 | 10.1.3.3/32 |

On the CEs, that is to say R1, R3, and R9, we have to configure the loopbacks and a default route pointing to the attached PE:

On R1, configure the following:
```
interface Loopback0
ip address 10.1.1.1 255.255.255.255

ip route 0.0.0.0 0.0.0.0 10.1.15.2
```

On R3, configure the following:
```
interface Loopback0
ip address 10.1.3.3 255.255.255.255

ip route 0.0.0.0 0.0.0.0 10.1.35.2
```

On R9, configure the following:
```
interface Loopback0
ip address 10.1.9.9 255.255.255.255

ip route 0.0.0.0 0.0.0.0 10.1.69.1
```

On the PEs, we have to put the interfaces in their corresponding VRFs and we have to configure a route towards the loopback of the CEs:

On R5, configure the following regarding the connection between R5 and R1:
```
interface Ethernet0/0
ip vrf forwarding Customer_A
ip address 10.1.15.2 255.255.255.0
```

On R5, configure the following regarding the connection between R5 and R3:
```
interface Ethernet0/1
ip vrf forwarding Customer_B
ip address 10.1.35.2 255.255.255.0
```
On R6, configure the following regarding the connection between R6 and R9:
```
interface Serial3/0
ip vrf forwarding Customer_A
ip address 10.1.69.1 255.255.255.0
```

**Task 23.10** Route the loopback0 interfaces of the CEs statically and make sure that those loopbacks are routed in their respective VRF. Verify that R1 loopback0 can ping R9 loopback0.

On R5, configure the following regarding the connection between R5 and R1:
```
router bgp 1
address-family ipv4 vrf Customer_A
redistribute static

ip route vrf Customer_A 10.1.1.1 255.255.255.255 10.1.15.1
```

On R5, configure the following regarding the connection between R5 and R3:
```
router bgp 1
address-family ipv4 vrf Customer_B
redistribute static
```

```
ip route vrf Customer_B 10.1.3.3 255.255.255.255 10.1.35.1
```

On R6, configure the following regarding the connection between R6 and R9:

```
router bgp 1
address-family ipv4 vrf Customer_A
redistribute static

ip route vrf Customer_A 10.1.9.9 255.255.255.255 10.1.69.2
```

The ping from the loopbak0 from R1 to the loopback0 of R9 is working. When performing a traceroute, we can see that the path is Label switched. The VPN Customer_A is up and running!

```
R1#ping 10.1.9.9 source 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 15/17/18 ms

R1#traceroute 10.1.9.9 source 10.1.1.1
Type escape sequence to abort.
Tracing the route to 10.1.9.9
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.15.2 0 msec 1 msec 0 msec
  2 10.1.25.1 [MPLS: Labels 23/17 Exp 0] 16 msec 19 msec 18 msec
  3 10.1.69.1 [MPLS: Label 17 Exp 0] 9 msec 9 msec 9 msec
  4 10.1.69.2 18 msec *  18 msec
```

**Task 23.11** The service provider is offering a service where the CEs are managed. Customer_A has chosen a managed service for its CEs. The management CE of the Service provider is the router called BB2. Create the management VRF on the router R2.

| AS | VPN name | rd | rt export | rt import |
|---|---|---|---|---|
| 1 | SP_Management | 100 | 1000 | 1000,1001 |

The VRF configuration on the R2 is the following:

```
ip vrf SP_Management
 rd 1:100
 route-target export 1:1000
 route-target import 1:1000
 route-target import 1:1001
```

**Task 23.12** The management network is using the network 192.168.1.128/25. Create on BB2 a loopback 100 with the following IP address: 192.168.1.129/25 and route it statically into the SP_Management VPN.

On BB2, configure the following:

```
interface loopback100
ip address 192.168.1.129 255.255.255.128

ip route 0.0.0.0 0.0.0.0 10.1.122.1
```

On R2, configure the following:

```
interface Ethernet0/1
ip vrf forwarding SP_Management
 ip address 10.1.122.1 255.255.255.0

router bgp 1
address-family ipv4 vrf SP_Management
redistribute static

ip route vrf SP_Management 192.168.1.128 255.255.255.128 10.1.122.2
```

**Task 23.13** Configure the multi-protocol BGP environment to enable the exchange of the RT information. As we are using iBGP, we create a full-mesh peering topology between R2, R5, and R6.

R2 has to support the MPLS VPN service. Therefore, R2 has to be included into the iBGP exchanges of the RT extended community attributes that already take place between the PEs R5 and R6. As we are using iBGP, a full-mesh of peering's has to be created.

On R2, configure the following:

```
router bgp 1
bgp log-neighbor-changes
neighbor 10.1.5.5 remote-as 1
neighbor 10.1.5.5 update-source Loopback0
neighbor 10.1.6.6 remote-as 1
neighbor 10.1.6.6 update-source Loopback0

address-family vpnv4
neighbor 10.1.5.5 activate
neighbor 10.1.5.5 send-community both
neighbor 10.1.6.6 activate
neighbor 10.1.6.6 send-community both
```

On R5, configure the following:

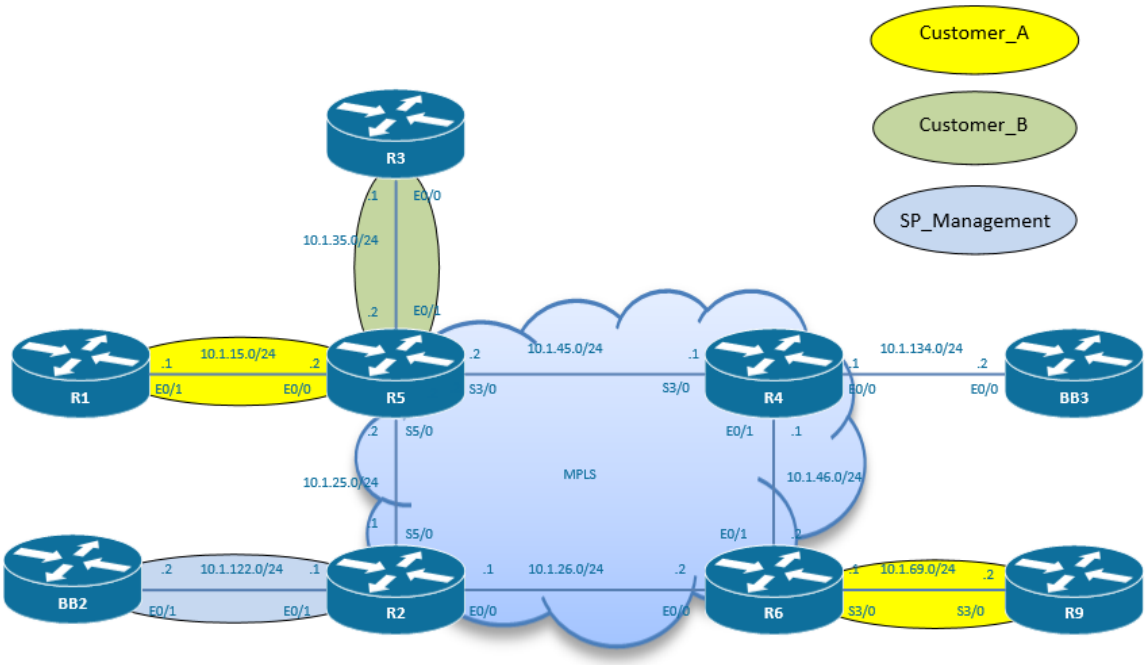
```
router bgp 1
bgp log-neighbor-changes
neighbor 10.1.2.2 remote-as 1
neighbor 10.1.2.2 update-source Loopback0
address-family vpnv4
neighbor 10.1.2.2 activate
neighbor 10.1.2.2 send-community both
```

On R6, configure the following:

```
router bgp 1
bgp log-neighbor-changes
neighbor 10.1.2.2 remote-as 1
neighbor 10.1.2.2 update-source Loopback0

address-family vpnv4
neighbor 10.1.2.2 activate
neighbor 10.1.2.2 send-community both
```




**Task 23.14** The R1 CE and the R9 CE from Customer A has to be reachable from the service provider management network. Use an export map called CE_Loopback_Export on R5 and on R6, and make sure that the management network can only see the loopback of R1 and R9.

The loopbacks of the CEs used for the management are part of the customer VRF routing table and each customer VRF has its own routing table. We have to bear in mind that isolation from one customer VPN to another customer VPN has to be preserved at any time. How can the service provider access in a simple and secure way CE loopback addresses that are part of different VRFs? Let's solve it.

To enable the connectivity between the CE loopbacks and the network management LAN, we are first going to import in the VRF Customer_A all the routes with the route-target 1000 that are present in the management VRF SP_Management.

The configuration on the R5 is the following:

```
ip vrf Customer_A
route-target import 1:1000
```

The configuration on the R6 is the following:

```
ip vrf Customer_A
route-target import 1:1000
```

The network management 192.168.1.128/25 is now present in the BGP database and the routing table of VRF Customer_A.

```
R5#sh ip route vrf Customer_A

Routing Table: Customer_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
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 8 subnets, 2 masks
S        10.1.1.1/32 [1/0] via 10.1.15.1
B        10.1.9.9/32 [200/0] via 10.1.6.6, 00:16:39
C        10.1.15.0/24 is directly connected, Ethernet0/0
L        10.1.15.2/32 is directly connected, Ethernet0/0
B        10.1.69.0/24 [200/0] via 10.1.6.6, 00:20:58
C        10.10.5.0/24 is directly connected, Loopback10
L        10.10.5.5/32 is directly connected, Loopback10
B        10.10.6.0/24 [200/0] via 10.1.6.6, 00:24:16
      192.168.1.0/25 is subnetted, 1 subnets
B        192.168.1.128 [200/0] via 10.1.2.2, 00:00:33

R5#sh ip bgp vpnv4 vrf Customer_A
BGP table version is 17, local router ID is 10.1.5.5
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: 1:1 (default for vrf Customer_A)
 *>  10.1.1.1/32      10.1.15.1                0         32768 ?
 *>i 10.1.9.9/32      10.1.6.6                 0    100      0 ?
 *>  10.1.15.0/24     0.0.0.0                  0         32768 i
 *>i 10.1.69.0/24     10.1.6.6                 0    100      0 ?
 *>  10.10.5.0/24     0.0.0.0                  0         32768 i
 *>i 10.10.6.0/24     10.1.6.6                 0    100      0 ?
 *>i 192.168.1.128/25 10.1.2.2                 0    100      0 ?
```

Now we have to ensure that there is a route back from the management network to the CE loopbacks. We are going to use the already existing route-target of 1:1001 which is going to be used for importing only the leaked routes in VRF SP_Management. The loopback0 of the CEs will be exported and tagged with the BGP attribute of 1:1001 in addition to the BGP attribute of the route-target of the Customer VRF. The CE loopback of a customer VRF will therefore be present in the BGP database of this customer VRF and of the management network VRF.

The following configuration is applied on R5:

```
ip prefix-list CE_Loopback seq 5 permit 10.1.1.1/32

route-map CE_Loopback_Export permit 10
 match ip address prefix-list CE_Loopback
 set extcommunity rt  1:1001 additive

ip vrf Customer_A
 export map CE_Loopback_Export
```

The following configuration is applied on R6:

```
ip prefix-list CE_Loopback seq 5 permit 10.1.9.9/32
route-map CE_Loopback_Export permit 10
 match ip address prefix-list CE_Loopback
 set extcommunity rt  1:1001 additive

ip vrf Customer_A
 export map CE_Loopback_Export
```

We can now ping from the BB2 to the loopback0 of the CEs:

```
BB2#ping  10.1.1.1 source 192.168.1.129
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.129
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms

BB2#ping  10.1.9.9 source 192.168.1.129
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.129
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms
```

Only the loopback of the CEs is routable. Please note that only the management network 192.168.1.129 is routable so the pings have to be sourced from this network. This looks fantastic, we did a great job.

**You have completed Lab 23**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 24: Configure and troubleshoot Multiprotocol Label Switching (Part 2)

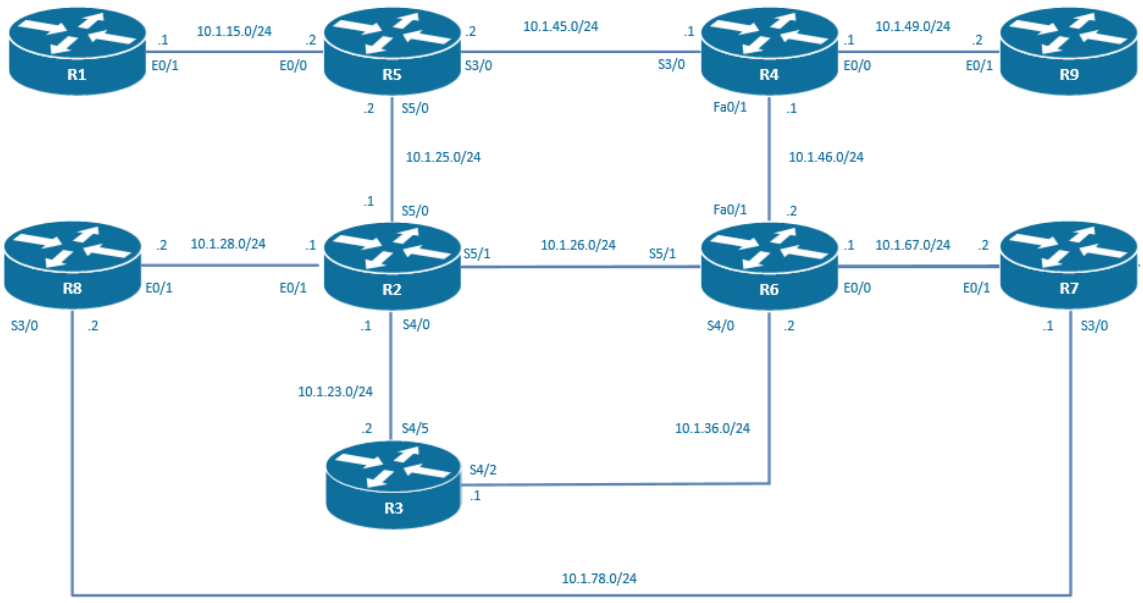
## Technologies covered

- PE-CE static routing
- PE-CE RIP routing
- PE-CE OSPF routing
- OSPF Domain-ID
- OSPF sham-link
- PE-CE EIGRP routing
- EIGRP SoO

## Overview

You have been tasked to configure a MPLS L3 VPN service on an existing MPLS backbone. You will have to configure the routing between the CEs and the PEs for two customer L3 VPNs.

The topology used in the lab will be the following:




**Estimated time to complete: 3-4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 24.1** Configure R5, R4, R6, and R2 as PE routers. The MPLS cloud is using BGP AS 1. Establish MP-BGP sessions between the PEs. Use the loopbacks 0 for the source of the peering's. Use R4 as a route-reflector for all the PEs.

The BGP route-reflector clients are only peering with the route-reflector R4 which will reflect all the routes that it receives via iBGP from the PEs to the other client PEs. The route reflector is breaking the by default iBGP rule of not advertising routes learnt from an iBGP peer to the other iBGP peers.

On the route-reflector clients R2, R5, and R6, configure the following:

```
router bgp 1
 bgp log-neighbor-changes
 neighbor 10.1.4.4 remote-as 1
 neighbor 10.1.4.4 update-source Loopback0

 address-family vpnv4
  neighbor 10.1.4.4 activate
  neighbor 10.1.4.4 send-community both
 exit-address-family
```

On the route-reflector R4, configure the following:

```
router bgp 1
 bgp log-neighbor-changes
 neighbor 10.1.2.2 remote-as 1
 neighbor 10.1.2.2 update-source Loopback0
 neighbor 10.1.5.5 remote-as 1
 neighbor 10.1.5.5 update-source Loopback0
 neighbor 10.1.6.6 remote-as 1
 neighbor 10.1.6.6 update-source Loopback0

 address-family vpnv4
  neighbor 10.1.2.2 activate
  neighbor 10.1.2.2 send-community both
  neighbor 10.1.2.2 route-reflector-client
  neighbor 10.1.5.5 activate
  neighbor 10.1.5.5 send-community both
  neighbor 10.1.5.5 route-reflector-client
  neighbor 10.1.6.6 activate
  neighbor 10.1.6.6 send-community both
  neighbor 10.1.6.6 route-reflector-client
```

**Task 24.2** Create the following L3 VPNs on all PEs.

| AS | VPN name | rd | rt export | rt import |
|---|---|---|---|---|
| 1 | Customer_A | 10 | 10 | 10 |
| 1 | Customer_B | 20 | 20 | 20 |

On all the PEs, configure the following:

```
ip vrf Customer_A
 rd 1:10
 route-target export 1:10
 route-target import 1:10

ip vrf Customer_B
 rd 1:20
 route-target export 1:20
 route-target import 1:20
```

**Task 24.3** Configure the following loopbacks for the VPN Customer_A and Customer_B. Make sure that the loopbacks are routed in the VPN MPLS cloud using network statements.

| | | | |
|---|---|---|---|
| R5 | Loopback15 | 10.10.5.5/32 | Customer_A |
| R5 | Loopback25 | 10.20.5.5/32 | Customer_B |
| R6 | Loopback16 | 10.10.6.6/32 | Customer_A |
| R6 | Loopback26 | 10.20.6.6/32 | Customer_B |
| R2 | Loopback12 | 10.10.2.2/32 | Customer_A |
| R2 | Loopback12 | 10.20.2.2/32 | Customer_B |
| R4 | Loopback14 | 10.10.4.4/32 | Customer_A |
| R4 | Loopback14 | 10.20.4.4/32 | Customer_B |

On R5, configure the following:

```
interface Loopback15
ip vrf forwarding Customer_A
ip address 10.10.5.5 255.255.255.255

interface Loopback25
ip vrf forwarding Customer_B
ip address 10.20.5.5 255.255.255.255
```

On R6, configure the following:

```
interface Loopback16
ip vrf forwarding Customer_A
ip address 10.10.6.6 255.255.255.255
interface Loopback20
ip vrf forwarding Customer_B
ip address 10.20.6.6 255.255.255.255
```

On R4, configure the following:

```
interface Loopback14
ip vrf forwarding Customer_A
ip address 10.10.4.4 255.255.255.255

interface Loopback24
ip vrf forwarding Customer_B
ip address 10.20.4.4 255.255.255.255
```

On R2, configure the following:

```
interface Loopback12
ip vrf forwarding Customer_A
ip address 10.10.2.2 255.255.255.255

interface Loopback22
ip vrf forwarding Customer_B
ip address 10.20.2.2 255.255.255.255
```

We have to redistribute those loopbacks using network statements.
On R5, configure the following:

```
router bgp 1

address-family ipv4 vrf Customer_A
network 10.10.5.5 mask 255.255.255.255

address-family ipv4 vrf Customer_B
network 10.20.5.5 mask 255.255.255.255
```

On R6, configure the following:

```
router bgp 1

address-family ipv4 vrf Customer_A
network 10.10.6.6 mask 255.255.255.255

address-family ipv4 vrf Customer_B
network 10.20.6.6 mask 255.255.255.255
```

On R4, configure the following:

```
router bgp 1

address-family ipv4 vrf Customer_A
network 10.10.4.4 mask 255.255.255.255

address-family ipv4 vrf Customer_B
network 10.20.4.4 mask 255.255.255.255
```

On R2, configure the following:

```
router bgp 1

address-family ipv4 vrf Customer_A
network 10.10.2.2 mask 255.255.255.255

address-family ipv4 vrf Customer_B
network 10.20.2.2 mask 255.255.255.255
```

**Task 24.4** Make sure that you have full reachability between Lo15, Lo16, Lo12, and Lo14 in VPN Customer_A.

Let's verify if the route reflector function configured earlier is working. There should be in the R6 routing tables a route to every loopbacks 1x just advertised into BGP on the PEs.

```
R6#sh ip route vrf Customer_A

Routing Table: Customer_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
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/32 is subnetted, 4 subnets
B        10.10.2.2 [200/0] via 10.1.2.2, 00:01:08
B        10.10.4.4 [200/0] via 10.1.4.4, 00:03:13
B        10.10.5.5 [200/0] via 10.1.5.5, 00:01:08
C        10.10.6.6 is directly connected, Loopback16
```

**Task 24.5** Make sure that you have full reachability between Lo25, Lo26, Lo22, and Lo24 in VPN Customer_B.

Let's verify if the route reflector function configured earlier is working. There should be in the R6 routing tables a route to every loopbacks 2x just advertised into BGP on the PEs.

```
R6#sh ip route vrf Customer_B

Routing Table: Customer_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

      10.0.0.0/32 is subnetted, 4 subnets
B        10.20.2.2 [200/0] via 10.1.2.2, 00:03:40
B        10.20.4.4 [200/0] via 10.1.4.4, 00:05:45
B        10.20.5.5 [200/0] via 10.1.5.5, 00:03:40
C        10.20.6.6 is directly connected, Loopback20
```

**Task 24.6** R1 is a CE in VRF Customer_A. The loopback of the router R1 should be routed statistically within the VPN Customer_A.

Let's configure static PE-CE routing between the PE R5 and the CE R1 in VRF Customer_A.

On R5, configure the following:

```
interface Ethernet0/0
ip vrf forwarding Customer_A
ip address 10.1.15.2 255.255.255.0

ip route vrf Customer_A 10.1.1.0 255.255.255.0 10.1.15.1
router bgp 1
address-family ipv4 vrf  Customer_A
redistribute static
```

On R1, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.15.2
```

Let's check that the network 10.1.1.0/24 is present in the routing table of the customer VRF Customer_A on the PEs:

```
R6#sh ip route vrf Customer_A

Routing Table: Customer_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
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
B        10.1.1.0/24 [200/0] via 10.1.5.5, 00:12:20
B        10.10.2.2/32 [200/0] via 10.1.2.2, 11:59:46
B        10.10.4.4/32 [200/0] via 10.1.4.4, 12:01:51
B        10.10.5.5/32 [200/0] via 10.1.5.5, 11:59:46
C        10.10.6.6/32 is directly connected, Loopback16
```

**Task 24.7** R9 is a CE in VRF Customer_B. The loopback of the router R9 should be routed using RIP version 2 within the VPN Customer_B. Do not redistribute BGP into RIP.

Let's configure RIP version 2 PE-CE routing between the PE R4 and the CE R9 in VRF Customer_B:

On R4, configure the following:

```
interface Ethernet0/0
 ip vrf forwarding Customer_B
 ip address 10.1.49.1 255.255.255.0
```

```
router rip
 version 2
 !
 address-family ipv4 vrf Customer_B
  network 10.0.0.0
  no auto-summary
 exit-address-family

router bgp 1
address-family ipv4 vrf Customer_B
redistribute rip
```

On R9, configure the following:

```
router rip
version 2
no auto-summary
network 10.1.9.0
network 10.1.49.0
ip route 0.0.0.0 0.0.0.0 10.1.49.1
```

Let's check if I can ping from the loopback22 of R2 to the loopback0 of R9 within the VPN Customer_B:

```
R2#ping vrf Customer_B 10.1.9.9 source  10.20.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.20.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 13/16/18 ms
```

**Task 24.8** R7 is a CE connected to PE R6 in VRF Customer_A. The loopback of the router R7 should be routed using OSPF process ID 7 in area 0 within the VPN Customer_A. Ensure that you have IP reachability between lo0 of R1 and lo0 of R7.

Let's configure OSPF PE-CE routing between the PE R6 and the CE R7 in VRF Customer_A:

On R6, configure the following:

```
interface Ethernet0/0
ip vrf forwarding Customer_A
ip address 10.1.67.1 255.255.255.0

router ospf 7 vrf Customer_A
network 10.1.67.1 255.255.255.0 area 0

router bgp 1
address-family ipv4 vrf Customer_A
redistribute ospf 7

router ospf 7 vrf Customer_A
redistribute bgp 1 subnets
```

On R7, configure the following:

```
router ospf 7
network 10.1.7.7 255.255.255.0 area 0
network 10.1.67.2 255.255.255.0 area 0
```

So far, all is running very well. I can ping from the loopback0 of R1 to the loopback0 of R7 through the MPLS VPN Customer_A infrastructure!

```
R1#ping 10.1.7.7 source 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1
```

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 6/8/10 ms

R1#traceroute 10.1.7.7 source 10.1.1.1
Type escape sequence to abort.
Tracing the route to 10.1.7.7
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.15.2 0 msec 1 msec 0 msec
  2 10.1.45.1 [MPLS: Labels 18/23 Exp 0] 9 msec 9 msec 9 msec
  3 10.1.67.1 [MPLS: Label 23 Exp 0] 9 msec 8 msec 9 msec
  4 10.1.67.2 27 msec *  10 msec
```

**Task 24.9** R8 is a CE connected to PE R2 in VRF Customer_A. The loopback of the router R8 should be routed using OSPF process ID 8 in area 0 within the VPN Customer_A. Ensure that you have IP reachability between lo0 of R7, R8, and R1.

Let's configure OSPF PE-CE routing between the PE R2 and the CE R8 in VRF Customer_A:

On R2, configure the following:

```
interface Ethernet0/1
ip vrf forwarding Customer_A
ip address 10.1.28.1 255.255.255.0

router ospf 8 vrf Customer_A
network 10.1.28.0 255.255.255.0 area 0

router bgp 1
address-family ipv4 vrf Customer_A
redistribute ospf 8

router ospf 8 vrf Customer_A
redistribute bgp 1 subnets
```

On R7, configure the following:

```
router ospf 8
network 10.1.8.8 255.255.255.0 area 0
network 10.1.28.0 255.255.255.0 area 0
```

I have full IP reachability between the loopbacks of the CEs part of VPN Customer_A.

```
R8#ping 10.1.7.7 source 10.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.7.7, timeout is 2 seconds:
Packet sent with a source address of 10.1.8.8
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms

R8#ping 10.1.1.1 source 10.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.8.8
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 24.10** On R8, the network 10.1.7.0/24 should be present in the OSPF database as a LSA type 3. If necessary, use a domainID of 78.

R7 and R8 are both CEs that are running OSPF as the CE-PE protocol. The OSPF process used on R7 is ID 7 and the OSPF process used on R8 is ID 8. As the OSPF process ID is different, the loopback0 of R7

will appear as an external OSPF route E2 in the routing table of R8 and as a LSA type-5 in the OSPF database.

```
R8#sh ip route
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, 11 subnets, 2 masks
O E2     10.1.1.0/24 [110/1] via 10.1.28.1, 00:09:27, Ethernet0/1
O E2     10.1.7.7/32 [110/11] via 10.1.28.1, 00:09:27, Ethernet0/1
C        10.1.8.0/24 is directly connected, Loopback0
L        10.1.8.8/32 is directly connected, Loopback0
C        10.1.28.0/24 is directly connected, Ethernet0/1
L        10.1.28.2/32 is directly connected, Ethernet0/1
O E2     10.1.67.0/24 [110/1] via 10.1.28.1, 00:09:27, Ethernet0/1
O E2     10.10.2.2/32 [110/1] via 10.1.28.1, 00:09:27, Ethernet0/1
O E2     10.10.4.4/32 [110/1] via 10.1.28.1, 00:09:27, Ethernet0/1
O E2     10.10.5.5/32 [110/1] via 10.1.28.1, 00:09:27, Ethernet0/1
O E2     10.10.6.6/32 [110/1] via 10.1.28.1, 00:09:27, Ethernet0/1
R8#sh ip ospf database

            OSPF Router with ID (10.1.8.8) (Process ID 8)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.8.8        10.1.8.8        589         0x80000002 0x0088EC 2
10.10.2.2       10.10.2.2       590         0x80000002 0x00F6AE 1

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.28.1       10.10.2.2       590         0x80000001 0x001DA9

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.1.1.0        10.10.2.2       697         0x80000001 0x0091AB 3489660929
10.1.7.7        10.10.2.2       697         0x80000001 0x006DB8 3489660929
10.1.67.0       10.10.2.2       697         0x80000001 0x00B842 3489660929
10.10.2.2       10.10.2.2       697         0x80000001 0x008D24 3489660929
10.10.4.4       10.10.2.2       697         0x80000001 0x00DB51 3489660929
10.10.5.5       10.10.2.2       697         0x80000001 0x00C664 3489660929
10.10.6.6       10.10.2.2       697         0x80000001 0x00B177 3489660929
```

Let’s configure the same OSPF process-ID 78 on both PE-CE connections. Please note that we are only going to modify the OSPF process-ID on the PE side. This is enough because OSPF process-ID is locally significant and BGP will notice that the OSPF process is identical on both sides of the MPLS network entry.

On R2, configure the following:

```
no router ospf 8 vrf Customer_A
router ospf 78 vrf Customer_A
 redistribute bgp 1 subnets
 network 10.1.28.0 0.0.0.255 area 0
!
router bgp 1
```

```
address-family ipv4 vrf Customer_A
no redistribute ospf 8
redistribute ospf 78
```

On R6, configure the following:

```
no router ospf 7 vrf Customer_A
router ospf 78 vrf Customer_A
 redistribute bgp 1 subnets
 network 10.1.67.0 0.0.0.255 area 0
!
router bgp 1
address-family ipv4 vrf Customer_A
no redistribute ospf 7
redistribute ospf 78
```

As the OSPF process ID is identical, the loopback0 of R7 will appear as an OSPF inter-area route in the routing table of R8 and as a LSA type-3 in the OSPF database. Please note that this is seen as an IA route even if we have area 0 on each side of the PE-CE connection.

```
R8#sh ip route
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, 11 subnets, 2 masks
O E2     10.1.1.0/24 [110/1] via 10.1.28.1, 00:07:13, Ethernet0/1
O IA     10.1.7.7/32 [110/21] via 10.1.28.1, 00:06:13, Ethernet0/1
C        10.1.8.0/24 is directly connected, Loopback0
L        10.1.8.8/32 is directly connected, Loopback0
C        10.1.28.0/24 is directly connected, Ethernet0/1
L        10.1.28.2/32 is directly connected, Ethernet0/1
O IA     10.1.67.0/24 [110/11] via 10.1.28.1, 00:06:23, Ethernet0/1
O E2     10.10.2.2/32 [110/1] via 10.1.28.1, 00:07:13, Ethernet0/1
O E2     10.10.4.4/32 [110/1] via 10.1.28.1, 00:07:13, Ethernet0/1
O E2     10.10.5.5/32 [110/1] via 10.1.28.1, 00:07:13, Ethernet0/1
O E2     10.10.6.6/32 [110/1] via 10.1.28.1, 00:07:13, Ethernet0/1
R8#sh ip ospf database

            OSPF Router with ID (10.1.8.8) (Process ID 8)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.8.8        10.1.8.8        447         0x80000003 0x009CD6 2
10.10.2.2       10.10.2.2       443         0x80000003 0x00FEA4 1

                Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.28.2       10.1.8.8        447         0x80000001 0x0001C1

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
10.1.7.7        10.10.2.2       379         0x80000001 0x0089F6
10.1.67.0       10.10.2.2       389         0x80000001 0x00D480

                Type-5 AS External Link States
```

```
Link ID         ADV Router      Age         Seq#       Checksum Tag
10.1.1.0        10.10.2.2       442         0x80000002 0x008FAC 3489660929
10.10.2.2       10.10.2.2       442         0x80000002 0x008B25 3489660929
10.10.4.4       10.10.2.2       442         0x80000002 0x00D952 3489660929
10.10.5.5       10.10.2.2       442         0x80000002 0x00C465 3489660929
10.10.6.6       10.10.2.2       442         0x80000002 0x00AF78 3489660929
```

This "magic" is happening because BGP is carrying the OSPF process-id information as an extended community attribute.

```
R2#sh ip bgp vpnv4 vrf Customer_A 10.1.7.7
BGP routing table entry for 1:10:10.1.7.7/32, version 72
Paths: (1 available, best #1, table Customer_A)
  Not advertised to any peer
  Refresh Epoch 1
  Local
    10.1.6.6 (metric 65) from 10.1.4.4 (10.1.4.4)
      Origin incomplete, metric 11, localpref 100, valid, internal, best
      Extended Community: RT:1:10 OSPF DOMAIN ID:0x0005:0x0000004E0200
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:10.10.6.6:0
      Originator: 10.1.6.6, Cluster list: 10.1.4.4
      mpls labels in/out nolabel/26
      rx pathid: 0, tx pathid: 0x0
```

**Task 24.11** Configure the connection between R7 and R8 in OSPF area 0 with an IP ospf cost of 4000.

On R7, configure the following:

```
router ospf 7
network 10.1.78.1 255.255.255.0 area 0

int s3/0
ip ospf cost 4000
```

On R8, configure the following:

```
router ospf 8
network 10.1.78.2 255.255.255.0 area 0

int s3/0
ip ospf cost 4000
```

**Task 24.12** Make sure that the path over the MPLS backbone is the preferred path for traffic going from R7 to R8. Use the loopback22 with IP address 2.2.2.2/32 on R2. Use the loopback66 with IP address 6.6.6.6/32 on R6.

Let's have a look at the routing table of R7:

```
R7#sh ip route
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, 13 subnets, 2 masks
O E2     10.1.1.0/24 [110/1] via 10.1.67.1, 00:16:59, Ethernet0/1
C        10.1.7.0/24 is directly connected, Loopback0
L        10.1.7.7/32 is directly connected, Loopback0
```

```
O        10.1.8.8/32 [110/4001] via 10.1.78.2, 00:03:10, Serial3/0
O        10.1.28.0/24 [110/4010] via 10.1.78.2, 00:03:10, Serial3/0
C        10.1.67.0/24 is directly connected, Ethernet0/1
L        10.1.67.2/32 is directly connected, Ethernet0/1
C        10.1.78.0/24 is directly connected, Serial3/0
L        10.1.78.1/32 is directly connected, Serial3/0
O E2     10.10.2.2/32 [110/1] via 10.1.67.1, 00:16:59, Ethernet0/1
O E2     10.10.4.4/32 [110/1] via 10.1.67.1, 00:16:59, Ethernet0/1
O E2     10.10.5.5/32 [110/1] via 10.1.67.1, 00:16:59, Ethernet0/1
O E2     10.10.6.6/32 [110/1] via 10.1.67.1, 00:16:59, Ethernet0/1
```

On R7, we notice that even if the direct connection from R7 to R8 is cost out with a high cost of 4000, the preferred path towards 10.1.8.8 is the direct path and not the path using the MPLS backbone. This is due to the fact that the LSA advertising 10.1.8.8 over the MPLS backbone is a LSA type 3 and that LSA advertising 10.1.8.8 over the direct connection is a LSA type 1.

```
R7#sh ip ospf database

            OSPF Router with ID (10.1.7.7) (Process ID 7)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.7.7        10.1.7.7        433         0x8000000B 0x00478F 4
10.1.8.8        10.1.8.8        451         0x80000005 0x00D253 4
10.10.2.2       10.10.2.2       1324        0x80000003 0x00FEA4 1
10.10.6.6       10.10.6.6       1264        0x80000006 0x00CD74 1

                Net Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum
10.1.28.2       10.1.8.8        1328        0x80000001 0x0001C1
10.1.67.2       10.1.7.7        1268        0x80000001 0x00B8DE

                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.1.1.0        10.10.2.2       1323        0x80000002 0x008FAC 3489660929
10.1.1.0        10.10.6.6       1262        0x80000003 0x0059D9 3489660929
10.10.2.2       10.10.2.2       1323        0x80000002 0x008B25 3489660929
10.10.2.2       10.10.6.6       1262        0x80000003 0x00CD59 3489660929
10.10.4.4       10.10.2.2       1323        0x80000002 0x00D952 3489660929
10.10.4.4       10.10.6.6       1262        0x80000003 0x00A37F 3489660929
10.10.5.5       10.10.2.2       1323        0x80000002 0x00C465 3489660929
10.10.5.5       10.10.6.6       1262        0x80000003 0x008E92 3489660929
10.10.6.6       10.10.2.2       1323        0x80000002 0x00AF78 3489660929
10.10.6.6       10.10.6.6       1262        0x80000004 0x00FE9F 3489660929
```

We have to configure a sham-link in order to make the path over the MPLS network look like a direct connection. We have to configure a separate /32 address on the PEs for use for the sham-links. The /32 address must meet the following strict criteria, that is to say belong to a VRF, not be advertised by OSPF, be advertised by BGP.

On R2, configure the following:

```
int lo22
ip vrf forwarding Customer_A
ip address 2.2.2.2 255.255.255.255

router bgp 1
address-family ipv4 vrf Customer_A
network 2.2.2.2 mask 255.255.255.255
```

On R6, configure the following:

```
int lo66
ip vrf forwarding Customer_A
ip address 6.6.6.6 255.255.255.255
```

```
router bgp 1
address-family ipv4 vrf Customer_A
network 6.6.6.6 mask 255.255.255.255
```

Let's build the sham-link:

On R2, configure the following:

```
router ospf 78 vrf Customer_A
area 0 sham-link 2.2.2.2 6.6.6.6 cost 1
```

On R6, configure the following:

```
router ospf 78 vrf Customer_A
area 0 sham-link 6.6.6.6 2.2.2.2 cost 1
```

Let's check the routing table of R7:

```
R7#sh ip route
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
      2.0.0.0/32 is subnetted, 1 subnets
O E2     2.2.2.2 [110/1] via 10.1.67.1, 00:02:56, Ethernet0/1
      6.0.0.0/32 is subnetted, 1 subnets
O E2     6.6.6.6 [110/1] via 10.1.67.1, 00:07:11, Ethernet0/1
      10.0.0.0/8 is variably subnetted, 13 subnets, 2 masks
O E2     10.1.1.0/24 [110/1] via 10.1.67.1, 00:43:08, Ethernet0/1
C        10.1.7.0/24 is directly connected, Loopback0
L        10.1.7.7/32 is directly connected, Loopback0
O        10.1.8.8/32 [110/22] via 10.1.67.1, 00:00:10, Ethernet0/1
O        10.1.28.0/24 [110/21] via 10.1.67.1, 00:00:10, Ethernet0/1
C        10.1.67.0/24 is directly connected, Ethernet0/1
L        10.1.67.2/32 is directly connected, Ethernet0/1
C        10.1.78.0/24 is directly connected, Serial3/0
L        10.1.78.1/32 is directly connected, Serial3/0
O E2     10.10.2.2/32 [110/1] via 10.1.67.1, 00:43:08, Ethernet0/1
O E2     10.10.4.4/32 [110/1] via 10.1.67.1, 00:43:08, Ethernet0/1
O E2     10.10.5.5/32 [110/1] via 10.1.67.1, 00:43:08, Ethernet0/1
O E2     10.10.6.6/32 [110/1] via 10.1.67.1, 00:43:08, Ethernet0/1
```

We have accomplished what we were asked to do! The path through the MPLS core is now the preferred path from R7 to reach R8.

**Task 24.13** R3 is a CE connected to PE R2 in VRF Customer_B. The loopback of the router R3 should be routed using EIGRP ID 1 with AS 200 within the VPN Customer_B. Use metric 1 1 1 1 1 when redistributing BGP into EIGRP on the PE. Ensure that you have IP reachability between lo0 of R9 and lo0 of R3.

Let's configure EIGRP PE-CE routing between the PE R2 and the CE R3 in VRF Customer_B:

On R2, configure the following:

```
int s4/0
ip vrf forwarding Customer_B
```

```
ip address 10.1.23.1 255.255.255.0

router eigrp 1
address-family ipv4 vrf Customer_B
autonomous-system 200
no auto-summary
network 10.1.23.0 0.0.0.255
redistribute bgp 1 metric 1 1 1 1 1

router  bgp 1
address-family ipv4 vrf Customer_B
redistribute eigrp 200
```

On R3, configure the following:

```
router eigrp 200
no auto-summary
network  10.1.23.2 0.0.0.255
network 10.1.3.3 0.0.0.255
```

I have full IP reachability between the loopbacks of the CEs part of VPN Customer_B.

```
R3#ping 10.1.9.9 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/18 ms
```

**Task 24.14** R3 is a CE connected to PE R6 in VRF Customer_B. Routing between R3 and R6 is using EIGRP ID 1 with AS 200. Use metric 1 1 1 1 1 when redistributing BGP into EIGRP on the PE.

Let's configure EIGRP PE-CE routing between the PE R6 and the CE R3 in VRF Customer_B:

On R6, configure the following:

```
int s4/0
ip vrf forwarding Customer_B
ip address 10.1.36.2 255.255.255.0

router eigrp 1
address-family ipv4 vrf Customer_B
autonomous-system 200
no auto-summary
network 10.1.36.0 0.0.0.255
redistribute bgp 1 metric 1 1 1 1 1

router  bgp 1
address-family ipv4 vrf Customer_B
redistribute eigrp 200
```

On R3, configure the following:

```
router eigrp 200
network  10.1.36.0 0.0.0.255
```

I have full IP reachability between the loopbacks of the CEs part of VPN Customer_B.

```
R3#ping 10.1.9.9 source 10.1.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/18 ms
```

**Task 24.15** By using the extended community 1:11 and 1:12, ensure that it is not allowed that an EIGRP route that has been distributed into BGP on R2 cannot be learnt via R6 when BGP is redistributed into EIGRP on R6, and vice-versa.

We have to configure the SoO extended community in order to tag the prefixes and prevent the temporary routing loop that can occur with a distance vector protocol like EIGRP.

On R2, configure the following:

```
route-map SOO permit 10
 set extcommunity soo 1:11
int s4/0
ip vrf sitemap SOO
```

On R6, configure the following:

```
route-map SOO permit 10
 set extcommunity soo 1:12
int s4/0
ip vrf sitemap SOO
```

We can now check that te soo community is attached to the prefixes advertised from EIGRP into BGP.

```
R2#sh bgp vpnv4 unicast vrf Customer_B 10.1.3.3
BGP routing table entry for 1:20:10.1.3.0/24, version 96
Paths: (2 available, best #2, table Customer_B)
  Advertised to update-groups:
     2
  Refresh Epoch 1
  Local
    10.1.6.6 (metric 65) from 10.1.4.4 (10.1.4.4)
      Origin incomplete, metric 2297856, localpref 100, valid, internal
      Extended Community: SoO:1:12 RT:1:20
        Cost:pre-bestpath:128:2297856 (default-2145185791) 0x8800:32768:0
        0x8801:200:640000 0x8802:65281:1657856 0x8803:65281:1500
        0x8806:0:167838467
      Originator: 10.1.6.6, Cluster list: 10.1.4.4
      mpls labels in/out 27/29
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  Local
    10.1.23.2 from 0.0.0.0 (10.1.2.2)
      Origin incomplete, metric 2297856, localpref 100, weight 32768, valid, sourced, best
      Extended Community: SoO:1:11 RT:1:20
        Cost:pre-bestpath:128:2297856 (default-2145185791) 0x8800:32768:0
        0x8801:200:640000 0x8802:65281:1657856 0x8803:65281:1500
        0x8806:0:167838467
      mpls labels in/out 27/nolabel
      rx pathid: 0, tx pathid: 0x0
```

**You have completed Lab 24**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 25: Configure and troubleshoot Ipsec Virtual Private Networks

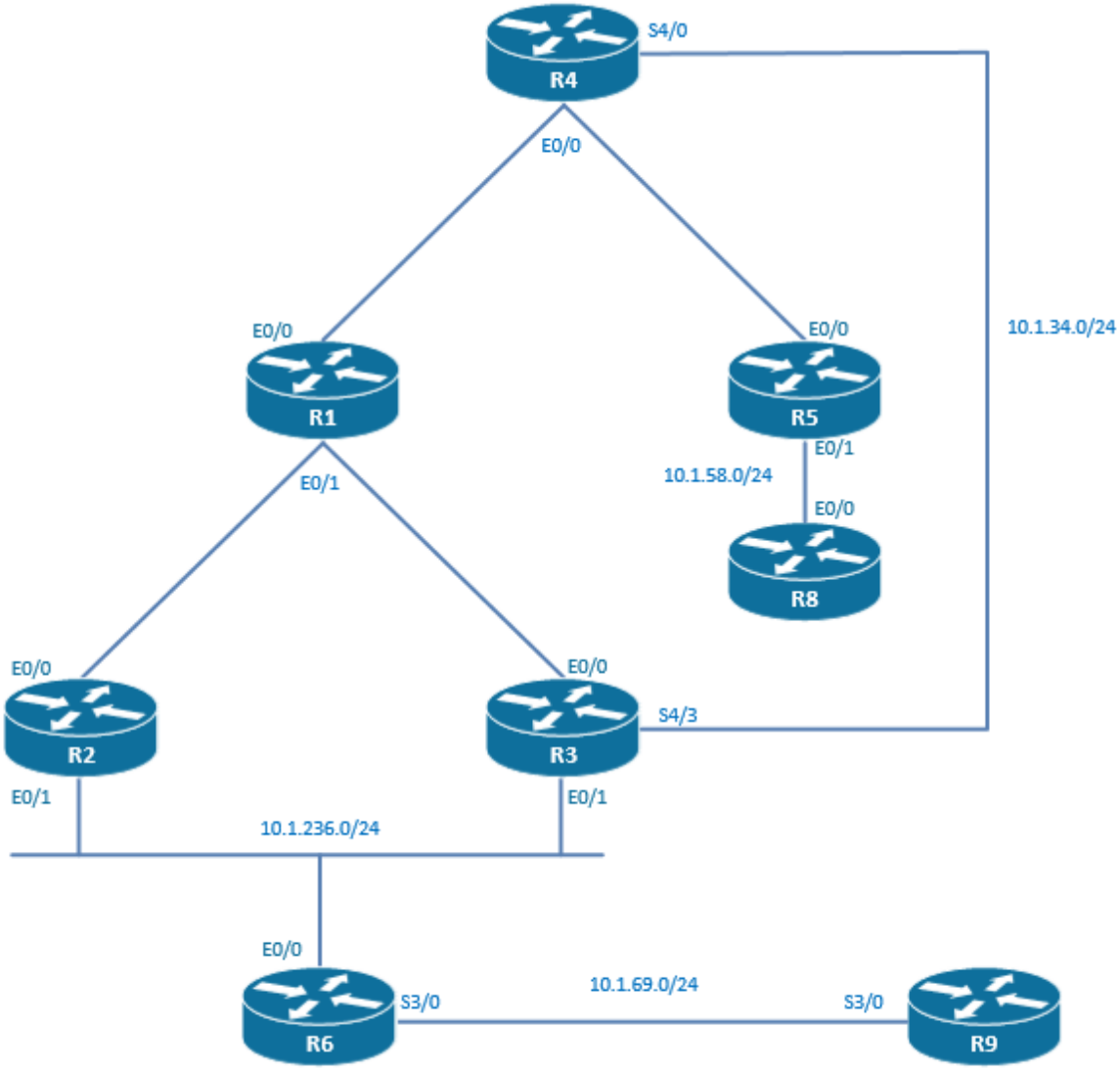
## Technologies covered

- GRE tunnels
- IPsec tunnels
- GRE over IPsec
- IPsec VTIs

## Overview

You have been tasked to configure an IPsec encryption on different connections of your network.

The topology used in the lab will be the following:




**Estimated time to complete: 3-4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 25.1** Configure a LAN-to-LAN IPsec tunnel on the serial connection between R4 and R3. Use a hash of MD5 and pre-shared key of "iPexpert" during the phase 1 negotiation.

We have first to configure the Internet Key Exchange policy.

On R4 and R3, configure the following:

```
crypto isakmp policy 10
 hash md5
 authentication pre-share
```

We have to configure the shared secret.
On R4 and R3, configure the following:

```
crypto isakmp key iPexpert address 0.0.0.0
```

**Task 25.2** Between R4 and R3, use esp-des encryption and an esp-md5-hmac authentication during the phase 2 negotiation.

The encryption and the authentication during the phase 2 negotiation are defined in the transform-set.

On R4 and R3, configure the IPSEC transform-set.

```
crypto ipsec transform-set R4-R3-transform esp-des esp-md5-hmac
```

**Task 25.3** Traffic going from loopback0 of R4 to loopback0 of R5 should be encrypted in both directions. You are not allowed to use a dynamic routing protocol or a default route.

We have to define the crypto-map that is going to be applied on the interface and the access-list which is defining the traffic to be protected by IPSec.

On R4, configure the following:

```
ip access-list extended acl_loopback
 permit ip 10.1.4.4 0.0.0.0 10.1.3.3 0.0.0.0

ip route 10.1.3.3 255.255.255.255 10.1.34.3

crypto map E2E_VPN 1 ipsec-isakmp
set peer 10.1.34.3
set transform-set R4-R3-transform
match address acl_loopback

int s4/0
crypto map E2E_VPN
```

On R3, configure the following:

```
ip access-list extended acl_loopback
 permit ip 10.1.3.3 0.0.0.0 10.1.4.4 0.0.0.0
```

```
ip route 10.1.4.4 255.255.255.255 10.1.34.4

crypto map E2E_VPN 1 ipsec-isakmp
set peer 10.1.34.4
set transform-set R4-R3-transform
match address acl_loopback

int s4/3
crypto map E2E_VPN
```

At this moment of the printing of the DSG, the ping from the loopback0 of R4 to the loopback0 of R3 is not working. This traffic should be encrypted in the IPSec site-to-site VPN tunnel. Troubleshooting still has to take place.

```
R3#ping 10.1.4.4 source 10.1.3.3
Type escapes sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.4.4, timeout is 2 seconds:
Packet sent with a source address of 10.1.3.3
.....
Success rate is 0 percent (0/5)
```

**Task 25.4** Configure a GRE tunnel on the serial connection between R2 and R9. The tunnel1 interface has an IP address of 192.168.29.2/24 on R2 and an IP address of 192.168.29.9/24 on R9. Use the E0/1 of R2 and S3/0 of R9 as source/destination of the tunnel. You are not allowed to configure anything on the R6 router.

We are going to configure a GRE tunnel between R2 and R9. We are going to enable static routing on

R2 and R9, and therefore not configure any routing on R6.

On R2, configure the following:

```
interface Tunnel1
 ip address 192.168.29.2 255.255.255.0
 tunnel source 10.1.236.2
 tunnel destination 10.1.69.9

ip route 10.1.69.0 255.255.255.0 10.1.236.6
```

On R9, configure the following:

```
interface Tunnel1
 ip address 192.168.29.9 255.255.255.0
 tunnel source 10.1.69.9
 tunnel destination 10.1.236.2

ip route 10.1.236.0 255.255.255.0 10.1.69.6
```

We can now check that we can ping from R2 the other end of the tunnel on R9:

```
R2#ping 192.168.29.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.29.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

**Task 25.5** You are not allowed to use a dynamic routing protocol or a default route. Traffic going from loopback0 of R2 to loopback0 of R9 should transit through this GRE tunnel.

By configuring static routing, we are going to forward the traffic from the loopback0 of R2 to the loopback0 of R9 through the GRE tunnel1.

On R2, configure the following:

```
ip route 10.1.9.9 255.255.255.255  192.168.29.6
```

On R9, configure the following:

```
ip route 10.1.2.2 255.255.255.255 192.168.29.2

R2#ping 10.1.9.9 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/7/9 ms

R2#traceroute 10.1.9.9 source 10.1.2.2
Type escape sequence to abort.
Tracing the route to 10.1.9.9
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.29.9 9 msec *  9 msec
```

**Task 25.6** There is a Web server which is connected to a client and the traffic is running over Tunnel 1. The client cannot communicate with the server. The web server is sending IP packets with a size of 1500 bytes and the DF-bit set. Configure the tunnel to restore connectivity between the server and the client. You are not allowed to clear the DF-bit or to intervene in the TCP negotiation.

The GRE encapsulation is adding an additional IP header to the already existing IP header. The size of this additional IP header is 24 bytes.

That's why a ping with a 1500 bytes size and the DF-bit set will not be able to be transmitted over the tunnel.

```
R2#ping 10.1.9.9 source 10.1.2.2 size 1500 df-bit
Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
Packet sent with the DF bit set
.....
Success rate is 0 percent (0/5)
```

On the contrary, a ping with a 1476 bytes size and lower and the DF-bit set will be able to be transferred over the tunnel.

```
R2#ping 10.1.9.9 source 10.1.2.2 size 1476 df-bit
Type escape sequence to abort.
Sending 5, 1476-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
Packet sent with the DF bit set
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/18 ms
```

One of the option would be to increase the MTU to 1524 bytes on the physical circuits that the tunnel1 is crossing and then to increase the MTU to 1500 bytes on the tunnel1.

The tunnel 1 is running over an Ethernet connection between R2 and R6 and over a serial interface between R6 and R9.

```
R2(config)#int s3/0
R2(config-if)#mt
R2(config-if)#mtu ?
  <64-4072>  MTU size in bytes
```

```
R2(config-if)#int e0/1
R2(config-if)#mtu ?
% Unrecognized command
```

The MTU on the serial connection could have been adjusted to 1524, but the MTU on the Ethernet connection is not adjustable and is fixed to 1500 bytes. So this is not an option.

One of the solutions would be to clear the DF-bit using an access-list and a route-map. The other solution would be to use the ip tcp adjust-mss command to make sure the end host sends always TCP packets smaller than 1476 bytes. Those 2 solutions are explicitly excluded in the question.

The only solution left is to increase the IP mtu to 1500 bytes on the tunnel interfaces. This is going to do the job because the DF-bit is not copied from the initial IP packet to the IP header used for GRE so the ping will simply be fragmented when entering the tunnel and will be re-assembled at the tunnel endpoint. The webserver and the client will not notice the fragmentation that took place in between.

Let's implement this solution.

On R2 and R9, configure the following:

```
int tu1
ip mtu 1500
```

The ping with the DF-bit set and the IP MTU of 1500 bytes is now working! It is the tunnel endpoints fragmenting it.

```
R2#
%SYS-5-CONFIG_I: Configured from console by console
R2#ping 10.1.9.9 source 10.1.2.2 size 1500 df-bit
Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
Packet sent with the DF bit set
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/18 ms
```

**Task 25.7** Encrypt the GRE traffic tunnel between R2 and R9. Use a GRE over IPsec tunneling. Use a hash of MD5 and pre-shared key of "iPexpert" during the phase 1 negotiation.

The first step will be to define which traffic has to be encrypted by using an access-list.

On R2, configure the following:

```
access-list 101 permit gre host 10.1.236.2 host 10.1.69.9
```

On R9, configure the following:

```
access-list 101 permit gre host 10.1.236.2 host 10.1.69.9
```

Then we have to configure the isakmp policy and the pre-shared keys.

On R2 and R9, configure the following:

```
crypto isakmp policy 1
  authentication pre-share

crypto isakmp key iPexpert address 0.0.0.0
```

**Task 25.8** Between R2 and R9, use esp-3des encryption and an esp-md5-hmac authentication during the phase 2 negotiation. Make sure that the IP connectivity between the loopback0 of R2 and the loopback0 of R9 is still up and running.

The following transform-set has to be configured on R2 and R9.

```
crypto ipsec transform-set setR2R9 esp-3des esp-md5-hmac
```

We have now to configure the crypto map and to apply it to the tunnel interfaces.

On R2, configure the following:

```
crypto map cryptotu1 1 ipsec-isakmp
set peer 10.1.69.9
set transform-set setR2R9
match address 101
int tu1
crypto map cryptotu1
```

On R9, configure the following:

```
crypto map cryptotu1 1 ipsec-isakmp
set peer 10.1.236.2
set transform-set setR2R9
match address 101

int tu1
crypto map cryptotu1
```

When applying the crypto map to the tunnel interface, I'm getting the following warning:

```
R2(config-crypto-map)#int tu1
R2(config-if)#crypto map cryptotu1
% NOTE: crypto map is configured on tunnel interface.
        Currently only GDOI crypto map is supported on tunnel interface.
```

That means that I have to use IPsec profiles with tunnel protection instead.
Let's remove the crypto map from the tunnel interfaces and configure the VTI profile:

On R2 and R9, configure the following:

```
int tu1
no crypto map cryptotu1

crypto ipsec profile cryptotu1
set transform-set setR2R9

interface Tunnel1
tunnel mode ipsec ipv4
tunnel protection ipsec profile cryptotu1
```

This is the right way to encrypt the traffic going through a tunnel interface! It is now working!

```
R2#ping 10.1.9.9 source 10.1.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 11/12/13 ms
```

**Task 25.9** Configure IPsec encryption on the ethernet connection between R5 and R8. Use an encryption of AES, a DH group number 2 ans pre-shared key of "iPexpert" during the phase 1 negotiation.

On R5 and R8, configure the following:

```
crypto isakmp policy 1
  group 2
  encryption aes 128
  authentication pre-share

crypto isakmp key iPexpert address 0.0.0.0
```

**Task 25.10** Between R5 and R8, use esp-3des encryption and an esp-sha-hmac authentication during the phase 2 negotiation.

On R5 and R8, configure the following:

```
crypto ipsec transform-set Transform esp-3des esp-sha-hmac
```

**Task 25.11** Create a VTI on both ends. IP address on R5 is 192.168.58.5/24 and IP address on R8 is 192.168.58.8/24.

In order to create a static IPSec VTI tunnel, you have to first configure a tunnel interface.

On R5, configure the following:

```
interface Tunnel58
 ip address 192.168.58.5 255.255.255.0
 tunnel source 10.1.58.5
 tunnel destination 10.1.58.8
```

On R8, configure the following:

```
interface Tunnel58
 ip address 192.168.58.8 255.255.255.0
 tunnel source 10.1.58.8
 tunnel destination 10.1.58.5
```

Apply encryption on the tunnel 58 on R5 and R8:

```
crypto ipsec profile cryptotu58
set transform-set Transform

interface Tunnel58
tunnel mode ipsec ipv4
tunnel protection ipsec profile cryptotu58
```

**Task 25.12** Traffic going from loopback0 of R5 to loopback0 from R8 should be encrypted in both directions. You are not allowed to use a dynamic routing protocol or a default route.

All traffic going through the tunnel will be encrypted so we have just to add static routes route the traffic from loopback from loopback.

On R5, configure the following:

```
ip route 10.1.8.8 255.255.255.255 192.168.58.8
```

On R8, configure the following:

```
ip route 10.1.5.5 255.255.255.255 192.168.58.5
```

The pings from the loopback of R5 to the loopback of R8 are working and are encrypted.

```
R5#ping  10.1.8.8 source 10.1.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.8.8, timeout is 2 seconds:
Packet sent with a source address of 10.1.5.5
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/5/6 ms
```

**You have completed Lab 25**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 26: Configure and troubleshoot IPsec Virtual Private Networks (Part 2)

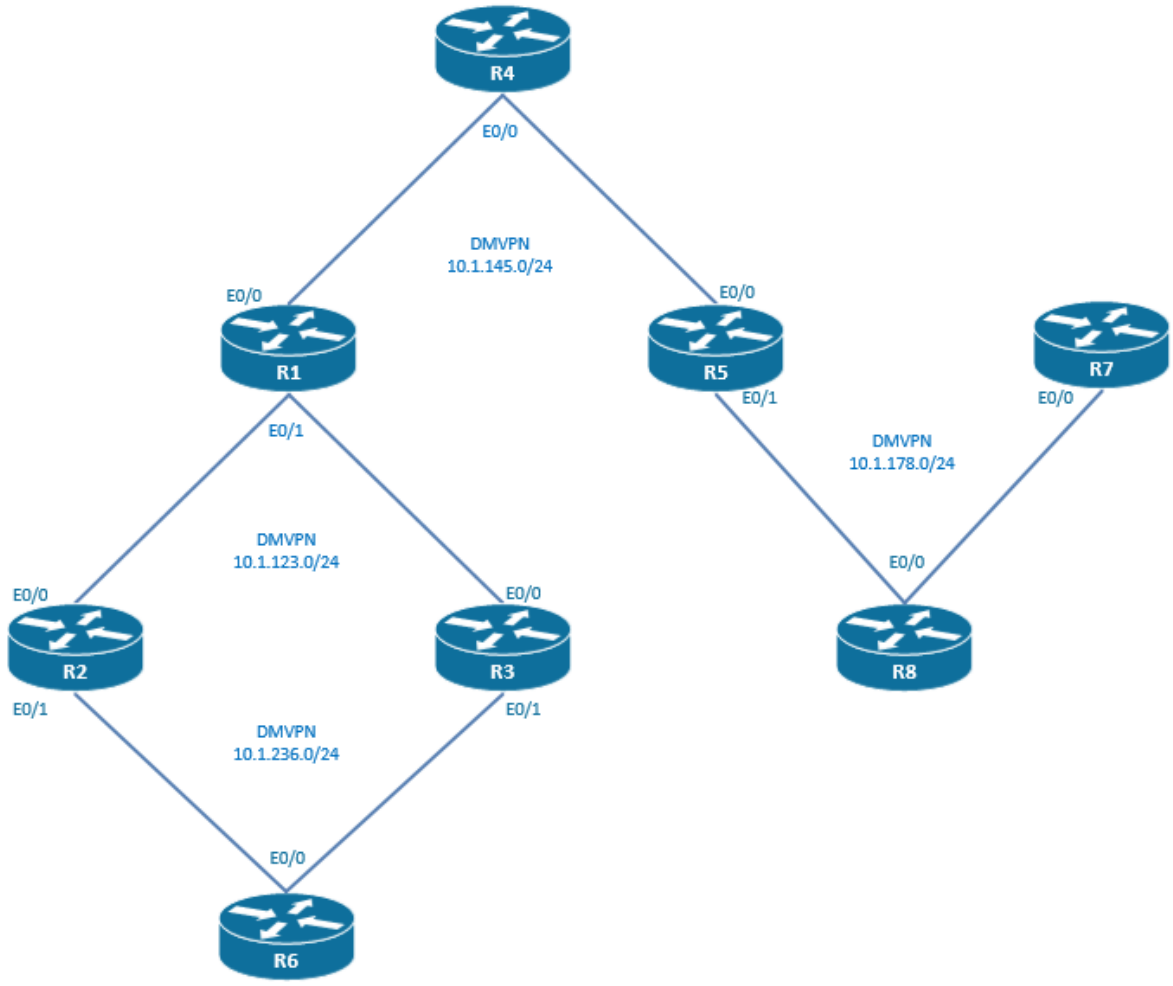
## Technologies covered

- DMVPN phase 1 EIGRP
- DMVPN phase 1 OSPF
- DMVPN phase 2 EIGRP
- DMVPN phase 2 OSPF
- DMVPN phase 1 with IPSec
- DMVPN phase 2 with IPSec

## Overview

You have been tasked to configure an IPsec encryption on different connections of your network.

The topology used in the lab will be the following:




**Estimated time to complete: 3-4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 26.1** Configure EIGRP AS 1 on the network between R2, R3, and R6. EIGRP should enable the IP connectivity between the loopback0 of R2, R3, and R6.

On R2, configure the following:

```
router eigrp 1
 network 10.1.2.2 0.0.0.0
 network 10.1.236.0 0.0.0.255
```

On R3, configure the following:

```
router eigrp 1
 network 10.1.3.3 0.0.0.0
 network 10.1.236.0 0.0.0.255
```

On R6, configure the following:

```
router eigrp 1
 network 10.1.6.6 0.0.0.0
 network 10.1.236.0 0.0.0.255
```

**Task 26.2** Configure DMVPN phase 1 between R2, R3, and R6. The tunnels number 11 is sourced from the loopback0. The Hub has to act as a NHS. The network-ID of the NHRP network is 11. Use a tunnel key of 11. Use the following IP addresses:

| | | |
|---|---|---|
| R2 | 11.0.0.2/24 | Spoke |
| R3 | 11.0.0.3/24 | Spoke |
| R6 | 11.0.0.6/24 | Hub |

We have to configure a DMVPN phase 1 network with dynamic mapping. The tunnel source has to be the loopback0 that we have routed with EIGRP in the previous question.

On the spoke R2, configure the following:

```
interface Tunnel11
 ip address 11.0.0.2 255.255.255.0
 ip nhrp map 11.0.0.6 10.1.6.6
 ip nhrp network-id 11
 ip nhrp nhs 11.0.0.6
 tunnel source Loopback0
 tunnel destination 10.1.6.6
```

On the spoke R3, configure the following:

```
interface Tunnel11
ip address 11.0.0.3 255.255.255.0
 ip nhrp map 11.0.0.6 10.1.6.6
 ip nhrp network-id 11
 ip nhrp nhs 11.0.0.6
 tunnel source Loopback0
 tunnel destination 10.1.6.6
```

On the hub R6, configure the following:

```
interface Tunnel11
 ip address 11.0.0.6 255.255.255.0
```

```
ip nhrp network-id 11
tunnel source Loopback0
tunnel mode gre multipoint
```

We can check that the DVMP phase 1 network is working as expected. From the hub R6, I can ping the tunnel interface on the spokes.

```
R6#ping 11.0.0.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R6#ping 11.0.0.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.0.0.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

From the spoke R2, I can ping the other spoke R3. The ping is transiting through the hub R6.

**Task 26.3** A new registration request should be sent every 10 seconds. A registration request sent by the spokes to the NHS should be kept for 60 seconds if no new update for this entry is received.

The modifications of those timers have to be completed on the Next-hop Client (NHC) which is the spokes. They are sending their registrations to the NHS and they are defining how often and for long are those registrations valid.

On R2 and R3, configure the following:

```
int tu11
ip nhrp registration timeout 10
ip nhrp holdtime 60
```

**Task 26.4** Configure the following loopbacks:

| | | |
|---|---|---|
| R2 | Loopback11 | 10.11.2.2/32 |
| R3 | Loopback11 | 10.11.3.3/32 |
| R6 | Loopback11 | 10.11.6.6/32 |

On R2, configure the following:

```
int lo11
ip address 10.11.2.2 255.255.255.255
```

On R3, configure the following:

```
int lo11
ip address 10.11.3.3 255.255.255.255
```

On R6, configure the following:

```
int lo11
ip address 10.11.6.6 255.255.255.255
```

**Task 26.5** Configure EIGRP AS 11 on the DMVPN tunnels, configure the spokes as EIGRP stub and advertise the loopback 11 of each router with a network statement. Make sure that there is IP reachability between the loopback11 of R2, R3, and R6.

On R2, configure the following:

```
router eigrp 11
 network 10.11.2.2 0.0.0.0
 network 11.0.0.0 0.0.0.255
 eigrp stub connected
```

On R3, configure the following:

```
router eigrp 11
 network 10.11.3.3 0.0.0.0
 network 11.0.0.0 0.0.0.255
 eigrp stub connected
```

On R6, configure the following:

```
router eigrp 11
 network 10.11.6.6 0.0.0.0
 network 11.0.0.0 0.0.0.255
```

Let's see if the EIGRP neighborship relations for AS 11 are up and running:

```
R6#sh ip eigrp 11 neighbors
EIGRP-IPv4 Neighbors for AS(11)
R6#
```

The EIGRP neighborships have not been established. This is due to the fact that multicast support has not been enabled on the DMVPN network. When configuring multicast support, don't forget that the tunnels 11 are sourced from loopback0.

On the spoke R2, configure the following:

```
int tu11
ip nhrp map multicast 10.1.6.6
```

On the spoke R3, configure the following:

```
int tu11
ip nhrp map multicast 10.1.6.6
```

On the hub R6, configure the following:

```
int  tu11
ip nhrp map multicast dynamic
```

Once the multicast support is configured, we can observe that the EIGRP neighborships are established:

```
R6#sh ip eigrp 11 neighbors
EIGRP-IPv4 Neighbors for AS(11)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   11.0.0.2                Tu11                     11 00:00:23 1177  5000  0  5
0   11.0.0.3                Tu11                     10 00:01:56  802  4812  0  4
```

In order to ensure the full IP reachability when EIGRP is working over a DMVPN phase 1 network, the following commands have to be configured on the hub router.

On R6, configure the following:

```
interface Tunnel11
no ip split-horizon eigrp 11
```

Let's check if we have full reachability between the loopback 11 of R2, R3, and R6.

```
R2#ping 10.11.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.11.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R2#ping 10.11.3.3
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.11.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

We can also check with a traceroute that to go from R2 to R3, we are transiting via R6.
The network is up and running, and we can move onto the next task.

**Task 26.6** Secure the traffic with IPSec on the DMVPN tunnels. Use a hash of MD5, a DH group number 2 and a wild-card pre-shared key of " iPexpert" during the phase 1 negotiation. Use esp-des encryption and an esp-md5-hmac authentication during the phase 2 negotiation.

We have to protect the traffic on the DMVPN network. We have to use the VTI feature. That is the only encryption method supported on a tunnel interface in the IOS used in the CCIE R&S v5 lab.
On R2, R3 and R6, configure the following:

```
crypto isakmp policy 11
  group 2
  hash md5
  authentication pre-share

crypto isakmp key iPexpert address 0.0.0.0

crypto ipsec transform-set Transform11 esp-des esp-md5-hmac
mode transport

crypto ipsec profile cryptotu11
set transform-set Transform11

interface Tunnel11
tunnel protection ipsec profile cryptotu11
```

**Task 26.7** Configure OSPF process 2 area 0 on the network between R1, R2, and R3. OSPF should enable the IP connectivity between the loopback0 of R1, R2, and R3.

On R1, configure the following:

```
router ospf 2
 network 10.1.1.1 0.0.0.0 area 0
 network 10.1.123.0 0.0.0.255 area 0
```

On R2, configure the following:

```
router ospf 2
 network 10.1.2.2 0.0.0.0 area 0
 network 10.1.123.0 0.0.0.255 area 0
```

On R3, configure the following:

```
router ospf 2
 network 10.1.3.3 0.0.0.0 area 0
 network 10.1.123.0 0.0.0.255 area 0
```

**Task 26.8** Configure DMVPN phase 1 between R1, R2, and R3. The tunnels number 22 are sourced from the loopback0. The network-ID of the NHRP network is 22. Use dynamic mapping. Use a tunnel key of 22. Use the following IP addresses:

| | | |
|---|---|---|
| R1 | 22.0.0.1/24 | Hub |
| R2 | 22.0.0.2/24 | Spoke |
| R3 | 22.0.0.3/24 | Spoke |

We have to configure DMVPN phase 1 network without dynamic mapping. The tunnel has to be sourced from loopback0.

On the spoke R2, configure the following:

```
interface Tunnel22
 ip address 22.0.0.2 255.255.255.0
 ip nhrp map 22.0.0.1 10.1.1.1
 ip nhrp network-id 22
 tunnel source Loopback0
 tunnel destination 10.1.1.1
```

On the spoke R3, configure the following:

```
interface Tunnel22
 ip address 22.0.0.3 255.255.255.0
 ip nhrp map 22.0.0.1 10.1.1.1
 ip nhrp network-id 22
 tunnel source Loopback0
 tunnel destination 10.1.1.1

 On the hub R1, configure the following:
interface Tunnel22
 ip address 22.0.0.1 255.255.255.0
 ip nhrp map 22.0.0.2 10.1.2.2
 ip nhrp map 22.0.0.3 10.1.3.3
 ip nhrp network-id 22
 tunnel source Loopback0
 tunnel mode gre multipoint
```

Our DMVPN phase 1 network is up and running:

```
R1#ping 22.0.0.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 22.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/5 ms

R1#ping 22.0.0.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 22.0.0.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

We can see the traffic is routed from R2 to R3 via R1 as it should be in the phase 1.

```
R2#traceroute 22.0.0.3
Type escape sequence to abort.
Tracing the route to 22.0.0.3
VRF info: (vrf in name/id, vrf out name/id)
  1 22.0.0.1 5 msec 0 msec 1 msec
  2 22.0.0.3 1 msec
```

**Task 26.9** Authenticate the NHRP network with an ID of 22 with the key “iPexpert”.

On R2, R3, and R1, configure the following:

```
int tu22
ip nhrp authentication iPexpert
```

**Task 26.10** Configure the following loopbacks:

| | | |
|---|---|---|
| R1 | Loopback22 | 10.22.1.1/32 |
| R2 | Loopback22 | 10.22.2.2/32 |
| R3 | Loopback22 | 10.22.3.3/32 |

On R1, configure the following:
```
int lo22
ip address 10.22.1.1 255.255.255.255
```

On R2, configure the following:
```
int lo22
ip address 10.22.2.2 255.255.255.255
```

On R3, configure the following:
```
int lo22
ip address 10.22.3.3 255.255.255.255
```

**Task 26.11** Configure OSPF process 22 area 0 on the DMVPN tunnels and advertise the loopback 22 of each router with a network statement. There should not be any DR elected. Make sure that there is IP reachability between the loopback22 of R2, R3, and R6.

On R1, configure the following:
```
router ospf 22
network 10.22.1.1 0.0.0.0 area 0
network 22.0.0.0 0.0.0.255 area 0
```

On R2, configure the following:
```
router ospf 22
network 10.22.2.2 0.0.0.0 area 0
network 22.0.0.0 0.0.0.255 area 0
```

On R3, configure the following:
```
router ospf 22
network 10.22.3.3 0.0.0.0 area 0
network 22.0.0.0 0.0.0.255 area 0
```

The OSPF neighborships have not been established. This is due to the fact that the default OSPF network type is point-to-point on a tunnel interface on the spokes and on the hub R1. The easiest way to fix it is to configure the type point-to-multipoint on the tunnel interfaces on the spokes and on the hub.

On R1, R2, and R3, configure the following:
```
int tu22
ip ospf network point-to-multipoint
```

**Task 26.12** Secure the traffic with IPSec on the DMVPN tunnels. Use an encryption of AES and a wild-card pre-shared key of “iPexpert” during the phase 1 negotiation. Use esp-aes encryption and an esp-sha-hmac authentication during the phase 2 negotiation.

On R1, R2, and R3, configure the following:
```
crypto isakmp policy 22
  encryption aes 128
  authentication pre-share

crypto isakmp key iPexpert address 0.0.0.0 0.0.0.0

crypto ipsec transform-set Transform22 esp-des esp-md5-hmac
mode transport

crypto ipsec profile cryptotu22
set transform-set Transform22
```

```
interface Tunnel22
tunnel protection ipsec profile cryptotu22
```

**Task 26.13** On the LAN between R1, R4, and R5, setup EIGRP routing in named configuration mode using AS3 and the name of iPexpert. EIGRP should enable the IP connectivity between the loopback0 of R1, R4, and R5.

On R1, configure the following:

```
router eigrp iPexpert
address-family ipv4 autonomous-system 3
network 10.1.1.1 0.0.0.0
network 10.1.145.0 0.0.0.255
```

On R4, configure the following:

```
router eigrp iPexpert
address-family ipv4 autonomous-system 3
network 10.1.4.4 0.0.0.0
network 10.1.145.0 0.0.0.255
```

On R5, configure the following:

```
router eigrp iPexpert
address-family ipv4 autonomous-system 3
network 10.1.5.5 0.0.0.0
network 10.1.145.0 0.0.0.255
```

**Task 26.14** Configure DMVPN phase 2 between R1, R4, and R5. The tunnels numbers 33 are sourced from the loopback0. The network-ID of the NHRP network is 33. Do not use dynamic mapping. Use a tunnel key of 33. Use the following IP addresses:

| | | |
|---|---|---|
| R1 | 33.0.0.1/24 | Spoke |
| R4 | 33.0.0.4/24 | Hub |
| R5 | 33.0.0.5/24 | Spoke |

On the spoke R1, configure the following:

```
interface Tunnel33
 ip address 33.0.0.1 255.255.255.0
 ip nhrp map 33.0.0.4 10.1.4.4
 ip nhrp network-id 33
 ip nhrp nhs 33.0.0.4
 tunnel source Loopback0
 tunnel mode gre multipoint
```

On the spoke R5, configure the following:

```
interface Tunnel33
 ip address 33.0.0.5 255.255.255.0
 ip nhrp map 33.0.0.4 10.1.4.4
 ip nhrp network-id 33
 ip nhrp nhs 33.0.0.4
 tunnel source Loopback0
 tunnel mode gre multipoint
```

On the hub R4, configure the following:

```
interface Tunnel33
 ip address 33.0.0.4 255.255.255.0
 ip nhrp network-id 33
 tunnel source Loopback0
 tunnel mode gre multipoint

 Our DMVPN phase 2 network is up and running:
```

```
R1#ping 33.0.0.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 33.0.0.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R1#ping 33.0.0.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 33.0.0.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
```

We can see the traffic is routed from R1 to R5 is not going through the hub R4 (as it should be in the phase 2)

```
R1#traceroute 33.0.0.5
Type escape sequence to abort.
Tracing the route to 33.0.0.5
VRF info: (vrf in name/id, vrf out name/id)
  1 33.0.0.5 5 msec *  5 msec
```

**Task 26.15** Configure the following loopbacks:

| | | |
|---|---|---|
| R1 | Loopback33 | 10.33.1.1/32 |
| R4 | Loopback33 | 10.33.4.4/32 |
| R5 | Loopback33 | 10.33.5.5/32 |

On R1, configure the following:

```
int lo33
ip address 10.33.1.1 255.255.255.255
```

On R4, configure the following:

```
int lo33
ip address 10.33.4.4 255.255.255.255
```

On R5, configure the following:

```
int lo33
ip address 10.33.5.5 255.255.255.255
```

**Task 26.16** Configure EIGRP process 33 on the DMVPN tunnels and advertise the loopback 33 of each router with a network statement. Make sure that a ping from the loopback 33 of R1 to the loopback 33 of R5 is always going through the hub.

On R1, configure the following:

```
router eigrp 33
 network 10.33.1.1 0.0.0.0
 network 33.0.0.0 0.0.0.255
```

On R4, configure the following:

```
router eigrp 33
 network 10.33.4.4 0.0.0.0
 network 33.0.0.0 0.0.0.255
```

On R5, configure the following:

```
router eigrp 33
 network 10.33.5.5 0.0.0.0
 network 33.0.0.0 0.0.0.255
```

The EIGRP neighborships have not been established. This is due to the fact that multicast support has not been enabled on the DMVPN network. When configuring multicast support, don't forget that the tunnels 11 are sourced from loopback0.

On the spoke R1, configure the following:

```
int tu33
ip nhrp map multicast 10.1.4.4
```

On the spoke R5, configure the following:

```
int tu33
ip nhrp map multicast 10.1.4.4
```

On the hub R4, configure the following:

```
int  tu33
ip nhrp map multicast dynamic
```

In order to ensure the full IP reachability when EIGRP is working over a DMVPN phase 2 network, the following commands have to be configured on the hub router.

On R4, configure the following:

```
interface Tunnel33
  no ip split-horizon eigrp 33
```

I can now ping from the loopback33 of R1 to the loopback33 of R5:

```
R1#ping 10.33.5.5 source 10.33.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.33.5.5, timeout is 2 seconds:
Packet sent with a source address of 10.33.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

We can notice that the traceroute from R1 to R5 is always transiting via the hub R4, even if we have a DMVPN phase 2 infrastructures underneath. The command "no ip next-hop-self eigrp 33" does not have to be configured on the interface tunnel33 of the hub R4.

```
R1#traceroute 10.33.5.5 source 10.33.1.1
Type escape sequence to abort.
Tracing the route to 10.33.5.5
VRF info: (vrf in name/id, vrf out name/id)
  1 33.0.0.4 1 msec 0 msec 0 msec
  2 33.0.0.5 1 msec *  5 msec
```

**Task 26.17** Secure the traffic with IPSec on the DMVPN tunnels. Use an encryption of 3-DES and a wild-card pre-shared key of "iPexpert" during the phase 1 negotiation. Use esp-des encryption and an esp-md5-hmac authentication during the phase 2 negotiation.

On R1, R4, and R5, configure the following:

```
crypto isakmp policy 33
  encryption 3des
  authentication pre-share

crypto isakmp key iPexpert address 0.0.0.0 0.0.0.0

crypto ipsec transform-set Transform33 esp-des esp-md5-hmac
mode transport
crypto ipsec profile cryptotu33
set transform-set Transform33

interface Tunnel33
tunnel protection ipsec profile cryptotu33
```

**Task 26.18** On the LAN between R5, R7, and R8, setup OSPF process 4 area 0. OSPF should enable the IP connectivity between the loopback0 of R5, R7, and R8.

On R5, configure the following:

```
router ospf 4
network 10.1.5.5 0.0.0.0 area 0
network 10.1.178.0 0.0.0.255 area 0
```

On R7, configure the following:

```
router ospf 4
network 10.1.7.7 0.0.0.0 area 0
network 10.1.178.0 0.0.0.255 area 0
```

On R8, configure the following:

```
router ospf 4
network 10.1.8.8 0.0.0.0 area 0
network 10.1.178.0 0.0.0.255 area 0
```

**Task 26.19** Configure DMVPN phase 2 between R5, R7, and R8. The tunnels numbers 44 are sourced from the loopback0. The network-ID of the NHRP network is 44. No NHRP configuration should be done on the hub. Use a tunnel key of 44. Use the following IP addresses:

| | | |
|---|---|---|
| R5 | 44.0.0.5/24 | Spoke |
| R7 | 44.0.0.7/24 | Spoke |
| R8 | 44.0.0.8/24 | Hub |

On the spoke R5, configure the following:

```
interface Tunnel44
 ip address 44.0.0.5 255.255.255.0
 ip nhrp map 44.0.0.8 10.1.8.8
 ip nhrp network-id 44
 ip nhrp nhs 44.0.0.8
 tunnel source Loopback0
 tunnel mode gre multipoint
```

On the spoke R7, configure the following:

```
interface Tunnel44
 ip address 44.0.0.7 255.255.255.0
 ip nhrp map 44.0.0.8 10.1.8.8
 ip nhrp network-id 44
 ip nhrp nhs 44.0.0.8
 tunnel source Loopback0
 tunnel mode gre multipoint
```

On the hub R8, configure the following:

```
interface Tunnel44
 ip address 44.0.0.8 255.255.255.0
 ip nhrp network-id 44
 tunnel source Loopback0
 tunnel mode gre multipoint
```

**Task 26.20** Configure the following loopbacks:

| | | |
|---|---|---|
| R5 | Loopback44 | 10.44.5.5/32 |
| R7 | Loopback44 | 10.44.7.7/32 |
| R8 | Loopback44 | 10.44.8.8/32 |

On R5, configure the following:
```
int lo44
ip address 10.44.5.5 255.255.255.255
```

On R7, configure the following:
```
int lo44
ip address 10.44.7.7 255.255.255.255
```

On R8, configure the following:
```
int lo44
ip address 10.44.8.8 255.255.255.255
```

**Task 26.21** Configure OSPF process 44 area 0 on the DMVPN tunnels and advertise the loopback 44 of each router with a network statement. The election of a DR should take place in this network. The DR should always be on the hub router. Multicast should be enabled on the DMVPN tunnels. Do not use OSPF type broadcast. Make sure that a ping from the loopback 44 of R7 to the loopback 44 of R5 is going directly from R7 to R5.

Let's first enable multicast on the DMVPN infrastructure:

On the spoke R5, configure the following:
```
int tu44
ip nhrp map multicast 10.1.8.8
```

On the spoke R7, configure the following:
```
int tu44
ip nhrp map multicast 10.1.8.8
```

On the hub R8, configure the following:
```
int  tu44
ip nhrp map multicast dynamic
```

Let's configure OSPF on the DMVPN network. The election of a DR should take place, that means OSPF type NBMA or OSPF type broadcast can be used. It is explicitly specified that OSPF network type broadcast cannot be used. Therefore, that leaves us with the NFMA network type.

On the hub R8, configure the following:
```
router ospf 44
network 10.44.8.8 0.0.0.0 area 0
network 44.0.0.0 0.0.0.255 area 0
neighbor 44.0.0.7
neighbor 44.0.0.5

int tu44
ip ospf network non-broadcast
```

On the spoke R5, configure the following:
```
router ospf 44
network 10.44.5.5 0.0.0.0 area 0
network 44.0.0.0 0.0.0.255 area 0
int tu44
ip ospf network non-broadcast
ip ospf priority 0
```

On the spoke R7, configure the following:
```
router ospf 44
network 10.44.7.7 0.0.0.0 area 0
network 44.0.0.0 0.0.0.255 area 0
```

```
int tu44
ip ospf network non-broadcast
ip ospf priority 0
```

On R8, OSPF neighborships has been formed over the Tunnel Tu44 interface.

```
R8#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.44.5.5         0   FULL/DROTHER    00:01:50    44.0.0.5        Tunnel44
10.44.7.7         0   FULL/DROTHER    00:01:45    44.0.0.7        Tunnel44
10.1.7.7          1   FULL/DROTHER    00:00:36    10.1.178.7      Ethernet0/0
10.33.5.5         1   FULL/DR         00:00:34    10.1.178.5      Ethernet0/0
```

We can see that R8 has been elected the DR of the network 44.0.0.0/24.

```
R5#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.44.8.8         1   FULL/DR         00:01:56    44.0.0.8        Tunnel44
10.1.7.7          1   FULL/DROTHER    00:00:34    10.1.178.7      Ethernet0/1
10.1.8.8          1   FULL/BDR        00:00:36    10.1.178.8      Ethernet0/1
```

**Task 26.22** Secure the traffic with IPSec on the DMVPN tunnels. Use an encryption of AES, a DH group number 1 and a wild-card pre-shared key of “iPexpert” during the phase 1 negotiation. Use esp-aes encryption and an esp-sha-hmac authentication during the phase 2 negotiation.

On R5, R7, and R8, configure the following:

```
crypto isakmp policy 44
  encryption aes 128
  group 1
  authentication pre-share

crypto isakmp key iPexpert address 0.0.0.0 0.0.0.0

crypto ipsec transform-set Transform44 esp-aes 128 esp-sha-hmac
mode transport

crypto ipsec profile cryptotu44
set transform-set Transform44

interface Tunnel44
tunnel protection ipsec profile cryptotu44
```

**You have completed Lab 26**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 27: Configure and troubleshoot Protocol Independent Multicast Operations (Part 1)

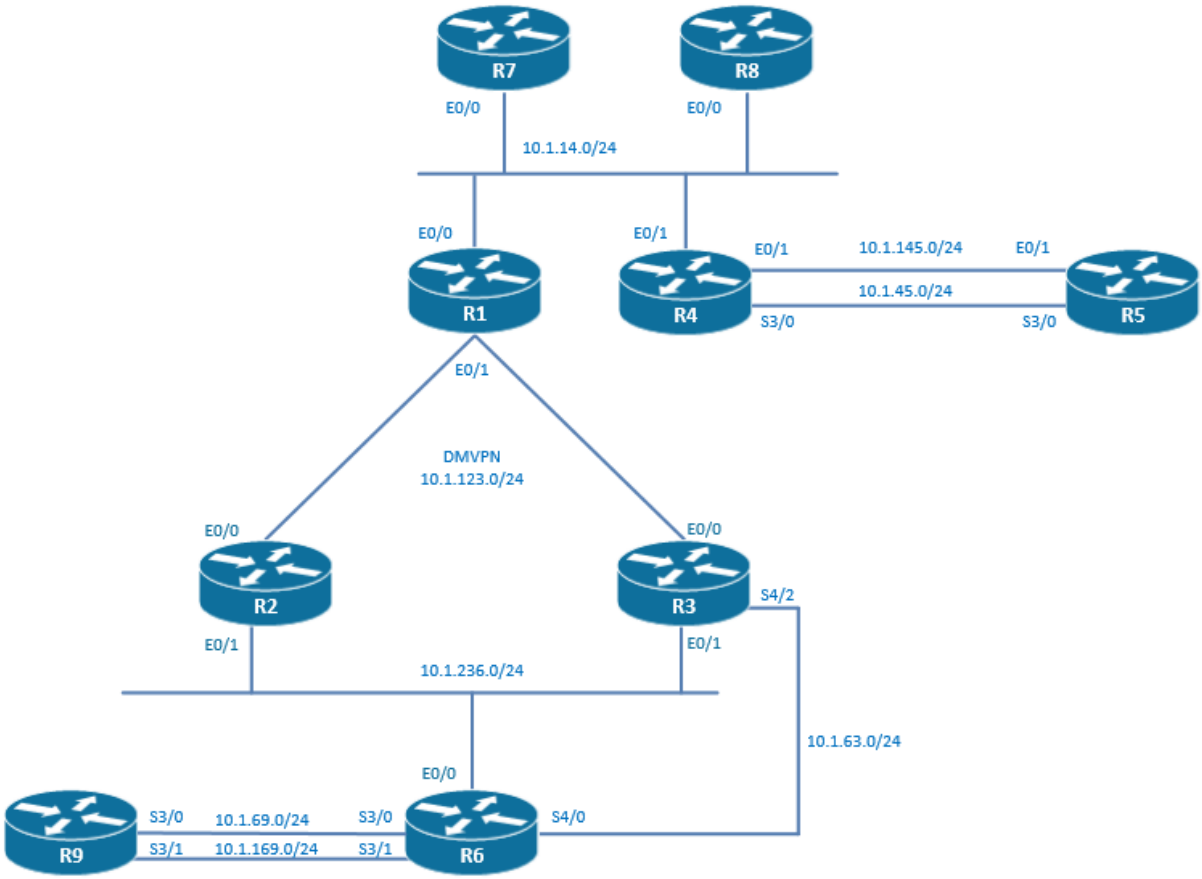
## Technologies covered

- PIM dense mode
- PIM sparse-dense mode
- PIM sparse mode
- RPF failure
- Accept RP
- Accept Register
- DR election
- NMBA mode

## Overview

You have been tasked to configure the multicast routing reachability in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 3 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 27.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN phase 2 without IPSec is the underlying used technology. Setup OSPF in area 0 in this DMVPN network. Configure the OSPF network type as NBMA.

Let's first enable multicast on the DMVPN infrastructure:

On the spoke R2, configure the following:

```
int tu23
ip nhrp map multicast 10.1.123.1
```

On the spoke R3, configure the following:

```
int tu23
ip nhrp map multicast 10.1.123.1
```

On the hub R1, configure the following:

```
int  tu23
ip nhrp map multicast dynamic
```

Let's configure OSPF on the DMVPN network. The election of a DR should take place, that means OSPF type NBMA or OSPF type broadcast can be used. It is explicitly specified that OSPF network type broadcast cannot be used. Therefore, that leaves us with the NFMA network type.

On the hub R1, configure the following:

```
router ospf 1
network 11.1.1.0 0.0.0.255 area 0
neighbor 11.1.1.2
neighbor 11.1.1.3

int tu23
ip ospf network non-broadcast
```

On the spoke R2, configure the following:

```
router ospf 1
network 11.1.1.0 0.0.0.255 area 0

int tu23
ip ospf network non-broadcast
ip ospf priority 0
```

On the spoke R3, configure the following:

```
router ospf 1
network 11.1.1.0 0.0.0.255 area 0

int tu23
ip ospf network non-broadcast
ip ospf priority 0
```

**Task 27.2** Advertise the loopbacks of R1, R2, and R3 in the OSPF process. Use network statements. Make sure that you can ping from the loopback0 of R2 to the loopback0 of R3.

```
On the hub R1, configure the following:
router ospf 1
network 10.1.1.1 0.0.0.0 area 0
```

On the spoke R2, configure the following:

```
router ospf 1
network 10.1.2.2 0.0.0.0 area 0
```

On the spoke R3, configure the following:

```
router ospf 1
network 10.1.3.3 0.0.0.0 area 0
```

**Task 27.3** Configure OSPF in area 55 on all the connections between R1, R4, and R5. R1 is the ABR. Cost out the network 10.1.45.0/24 with an OSPF cost of 2000.

On R1, configure the following:

```
router ospf 1
network 10.1.14.0 0.0.0.255 area 55
```

On R4, configure the following:

```
router ospf 1
network 10.1.14.0 0.0.0.255 area 55
network 10.1.145.0 0.0.0.255 area 55
network 10.1.45.0 0.0.0.255 area 55

int s3/0
ip ospf cost 2000
```

On R5, configure the following:

```
router ospf 1
network 10.1.145.0 0.0.0.255 area 55
network 10.1.45.0 0.0.0.255 area 55

int s3/0
ip ospf cost 2000
```

**Task 27.4** Advertise the loopbacks of R4 and R5 in the OSPF process. Use network statements.

On R4, configure the following:

```
router ospf 1
network 10.1.1.4 0.0.0.0 area 55
```

On R5, configure the following:

```
router ospf 1
network 10.1.1.5 0.0.0.0 area 55
```

**Task 27.5** Configure OSPF in area 99 on all the connections between R2, R3, R6, and R9. R3 is the ABR. Cost out the network 10.1.236.0/24 with an OSPF cost of 2000.

On R2, configure the following:

```
router ospf 1
network 10.1.236.0 0.0.0.255 area 99

int e0/1
ip ospf cost 2000
```

On R3, configure the following:

```
router ospf 1
network 10.1.236.0 0.0.0.255 area 99
network 10.1.63.0 0.0.0.255 area 99
int e0/1
ip ospf priority 255
ip ospf cost 2000
```

On R6, configure the following:

```
router ospf 1
network 10.1.236.0 0.0.0.255 area 99
network 10.1.63.0 0.0.0.255 area 99
network 10.1.69.0 0.0.0.255 area 99
network 10.1.169.0 0.0.0.255 area 99

int e0/0
ip ospf cost 2000
```

On R9, configure the following:

```
router ospf 1
network 10.1.69.0 0.0.0.255 area 99
network 10.1.169.0 0.0.0.255 area 99
```

**Task 27.6** Advertise the loopbacks of R6 and R9 in the OSPF process. Use network statements.

On R6, configure the following:

```
router ospf 1
network 10.1.1.6 0.0.0.0 area 99
```

On R9, configure the following:

```
router ospf 1
network 10.1.1.9 0.0.0.0 area 99
```

**Task 27.7** There is a multicast server connected on R5 that is sending a stream with the IP address 225.5.5.5. The listeners for this group are located on R1 and R4 only. Configure the network to route this multicast stream from the source to the listeners without the use of any RP. Do not enable multicast on the 10.1.145.0/24 network.

On R1, configure the following:

```
int e0/0
ip pim dense-mode
```

On R4, configure the following:

```
int e0/1
ip pim dense-mode

int s3/0
ip pim dense-mode
```

On R5, configure the following:

```
int s3/0
ip pim dense-mode
```

**Task 27.8** Configure R1 E0/0 to join 225.5.5.5 and make sure that you can ping this multicast group from R5. If necessary, the use of mroute is allowed.

On R1, configure the following:

```
int e0/0
ip igmp join-group 225.5.5.5

R5#ping  225.5.5.5
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 225.5.5.5, timeout is 2 seconds:

Reply to request 0 from 10.1.14.1, 2 ms
Reply to request 0 from 10.1.14.1, 7 ms
Reply to request 0 from 10.1.14.1, 2 ms
```

Even if multicast is not enabled on the interface E0/1, the multicast traffic is going from the source to the receiver. This is due to the fact that the source of the traffic will be originated on the PIM enabled interface s3/0 in R5 and therefore the IGP is not used to route from R5 to R4. The RFP problems are avoided for this stream and no static ip mroutes are necessary.

```
R5#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
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), 00:01:23/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial3/0, Forward/Dense, 00:01:23/stopped

(10.1.145.5, 225.5.5.5), 00:01:23/00:01:42, flags:
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial3/0, Forward/Dense, 00:01:23/stopped

(*, 224.0.1.40), 02:10:26/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial3/0, Forward/Dense, 02:09:45/stopped

R4#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
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), 00:30:47/stopped, RP 0.0.0.0, flags: DC
```

```
    Incoming interface: Null, RPF nbr 0.0.0.0
    Outgoing interface list:
      Serial3/0, Forward/Dense, 00:30:17/stopped
      Ethernet0/1, Forward/Dense, 00:30:47/stopped

  (10.1.45.5, 225.5.5.5), 00:00:05/00:02:54, flags: T
    Incoming interface: Serial3/0, RPF nbr 10.1.45.5
    Outgoing interface list:
      Ethernet0/1, Forward/Dense, 00:00:05/stopped

  (10.1.145.5, 225.5.5.5), 00:00:05/00:02:54, flags: T
    Incoming interface: Ethernet0/1, RPF nbr 0.0.0.0
    Outgoing interface list:
      Serial3/0, Forward/Dense, 00:00:05/stopped

  (*, 224.0.1.40), 00:30:47/00:02:16, RP 0.0.0.0, flags: DCL
    Incoming interface: Null, RPF nbr 0.0.0.0
    Outgoing interface list:
      Serial3/0, Forward/Dense, 00:30:17/stopped
      Ethernet0/1, Forward/Dense, 00:30:47/stopped
```

**Task 27.9** There is a multicast server connected on R9 that is sending a stream with the IP address 229.9.9.9. The listeners for this group are located on R5 on network 10.1.45.0/24. Configure the network to route this multicast stream from the source to the listeners with the use of a static RP. Do not enable multicast on the 10.1.63.0/24 network.

We are going to configure the RP to be the loopback0 of the router R1.

On R9, configure the following:

```
ip multicast-routing
ip pim rp-address 10.1.1.1

int s3/0
ip pim sparse-mode
int s3/1
ip pim sparse-mode
```

On R6, configure the following:

```
ip multicast-routing
ip pim rp-address 10.1.1.1

int s3/0
ip pim sparse-mode
int s3/1
ip pim sparse-mode
int e0/0
ip pim sparse-mode
```

On R3, configure the following:

```
ip multicast-routing
ip pim rp-address 10.1.1.1

int e0/1
ip pim sparse-mode
int tu23
ip pim sparse-mode
```

On R2, configure the following:

```
ip multicast-routing
ip pim rp-address 10.1.1.1

int e0/1
```

```
ip pim sparse-mode
int tu23
ip pim sparse-mode
```

We have to configure sparse-dense mode on the connections that we already configured in dense mode, because we would like that any group that cannot be registered with the RP becomes dense mode.

On R1, configure the following:

```
ip pim rp-address 10.1.1.1

int lo0
ip pim sparse-mode
int tu23
ip pim sparse-mode
int e0/0
ip pim sparse-dense-mode
```

On R4, configure the following:

```
ip pim rp-address 10.1.1.1

int e0/0
ip pim sparse-dense-mode
int s3/0
ip pim sparse-dense-mode
```

On R5, configure the following:

```
ip pim rp-address 10.1.1.1
int s3/0
ip pim sparse-dense-mode
```

**Task 27.10** Make sure that R1 is the RP only for the group 229.9.9.9. Use the loopback0 interface for the RP IP address.

On R1, configure the following:

```
ip access-list standard feed_229_999
 permit 229.9.9.9

ip pim rp-address 10.1.1.1 feed_229_999
```

**Task 27.11** Configure R3 to send the PIM join message to the RP on behalf of the 10.1.236.0/24 network.

One router on the LAN will be sending the PIM register message on behalf of the whole LAN. This is the multicast DR. All routers have by default the same priority 1 and the highest priority is becoming the DR. Please note that this election is preemptive and is therefore happening in the instant that the change is configured.

On R3, configure the following:

```
int e0/1
ip pim dr-priority 20
```

**Task 27.12** Configure R5 E0/1 to join 229.9.9.9 and make sure that you can ping this multicast group from R9. The use of mroute is allowed.

On R5, we configure a listener to the group 229.9.9.9:

```
interface Serial3/0
ip igmp join-group 229.9.9.9
```

We can notice that the listener has not been registered on the RP:

```
R1#sh ip pim rp mapping
PIM Group-to-RP Mappings

Group(s): 224.0.0.0/4, Static
    RP: 10.1.1.1 (?)

On R5, we can see that we have an RPF failure.
R5#sh ip mroute count
Use "show ip mfib count" to get better response time for a large number of mroutes.

IP Multicast Statistics
2 routes using 1184 bytes of memory
2 groups, 0.00 average sources per group
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 229.9.9.9, Source count: 0, Packets forwarded: 0, Packets received: 7
  RP-tree: Forwarding: 0/0/0/0, Other: 7/7/0
```

This is due to the fact that the route to the RP is over the Ethernet connection between R5 and R4 because it is following the routes computed by the IGP protocol, whereas the multicast traffic is over the serial connection between R5 and R4 because it is following the routing computed by the PIM protocol.

```
R5#sh ip route 10.1.1.1
Routing entry for 10.1.1.1/32
  Known via "ospf 1", distance 110, metric 21, type inter area
  Last update from 10.1.145.4 on Ethernet0/1, 00:05:31 ago
  Routing Descriptor Blocks:
  * 10.1.145.4, from 10.1.1.1, 00:05:31 ago, via Ethernet0/1
      Route metric is 21, traffic share count is 1

R5#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
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 229.9.9.9), 03:38:29/stopped, RP 10.1.1.1, flags: SJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial3/0, Forward/Sparse-Dense, 00:06:12/00:00:33
```

In order to fix it, we have to add an ip mroute that is going to be used for the RPF check in place of the OSPF routes.

```
ip mroute 10.1.1.1 255.255.255.255 10.1.45.4
```

We can check now that the (*,229.9.9.9) entry is present in the ip mroute of the RP, meaning that the group has been registered with the RP.

```
R1#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
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 229.9.9.9), 00:00:47/00:02:42, RP 10.1.1.1, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/0, Forward/Sparse-Dense, 00:00:47/00:02:42
```

When send the stream from R9, there is another RPF failure situation that occurs on R3.

```
R3#sh ip mroute count
Use "show ip mfib count" to get better response time for a large number of mroutes.

IP Multicast Statistics
5 routes using 1828 bytes of memory
3 groups, 0.66 average sources per group
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 229.9.9.9, Source count: 2, Packets forwarded: 0, Packets received: 1012
  RP-tree: Forwarding: 0/0/0/0, Other: 8/8/0
  Source: 10.1.169.9/32, Forwarding: 0/0/0/0, Other: 502/502/0
  Source: 10.1.69.9/32, Forwarding: 0/0/0/0, Other: 502/502/0
```

The multicast traffic is arriving from R6 using the LAN because it is the path enabled with the PIM protocol whereas the preferred path from R3 to R6 is to use the serial connection as instructed by the IGP.

In order to solve the RFP failure, we have to configure the following on R3:

```
ip mroute 10.1.169.0 255.255.255.0 10.1.236.6
ip mroute 10.1.169.0 255.255.255.0 10.1.236.6
```

The feed generated on R9 is able to reach the multicast listener on R5:

```
R9#ping 229.9.9.9
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 229.9.9.9, timeout is 2 seconds:

Reply to request 0 from 10.1.45.5, 15 ms
Reply to request 0 from 10.1.45.5, 480 ms
Reply to request 0 from 10.1.45.5, 480 ms
Reply to request 0 from 10.1.45.5, 480 ms
Reply to request 0 from 10.1.45.5, 476 ms
Reply to request 0 from 10.1.45.5, 476 ms
Reply to request 0 from 10.1.45.5, 471 ms
```

**Task 27.13** There is a multicast server connected on R3 that is sending a stream with the IP address 233.3.3.3. The listeners for this group are located on R2. Shut down the interface e0/1 on R2. Configure the network to route this multicast stream from the source to the listeners with the use of a static RP.

On R2, configure the following:

```
int tu23
ip igmp join-group 233.3.3.3

int e0/1
shut
```

**Task 27.14** Make sure that R1 is allowed to be the RP for the group 233.3.3.3. Use the loopback0 interface for the RP IP address.

On R1, configure the following:

```
ip access-list standard feed_229_999
 permit 233.3.3.3
```

**Task 27.15** Ensure that R2 and R3 send registers (*,G) entries for the group 233.3.3.3 only to the router R1.

On R2 and R3, configure the following:

```
access-list 10 permit 233.3.3.3
ip pim accept-rp 10.1.1.1 10
```

By configuring this on R2 and R3, we are making sure that R2 and R3 will only register the multicast group 233.3.3.3 to the RP R1.

**Task 27.16** Make sure that you can ping multicast group 233.3.3.3 from R3.

Let's try to ping 233.3.3.3 from R3.

```
R3#ping 233.3.3.3
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 233.3.3.3, timeout is 2 seconds:.
```

It is not working because the multicast traffic that is arriving on one interface cannot by default be forwarded again on the same interface. We have to tell PIM that this behavior has to be allowed on the hub router.

On R1, configure the following:

```
int tu23
ip pim nbma-mode
```

I can now ping the receiver on R2 by sending the stream from R3, enabling spoke-to-spoke multicast communication.

```
R3#ping 233.3.3.3
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 233.3.3.3, timeout is 2 seconds:

Reply to request 0 from 11.1.1.2, 38 ms
Reply to request 0 from 11.1.1.2, 38 ms
```

**Task 27.17** There is a plan to add a new multicast datastream. The multicast group will be 227.7.7.7 and the source is going to be the server 10.1.63.200. Configure the router R3 so that when he becomes the RP for this multicast group, the only allowed source is the IP address 10.1.63.200. All other servers trying to register this group should be denied

On R3, configure the following:

```
ip access-list extended ALLOWED_SOURCE
 permit ip host 10.1.63.200 227.7.7.7 0.0.0.0

ip pim accept-register list ALLOWED_SOURCE
```

**You have completed Lab 27**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 28: Configure and troubleshoot Protocol Independent Multicast Operations (Part 2)

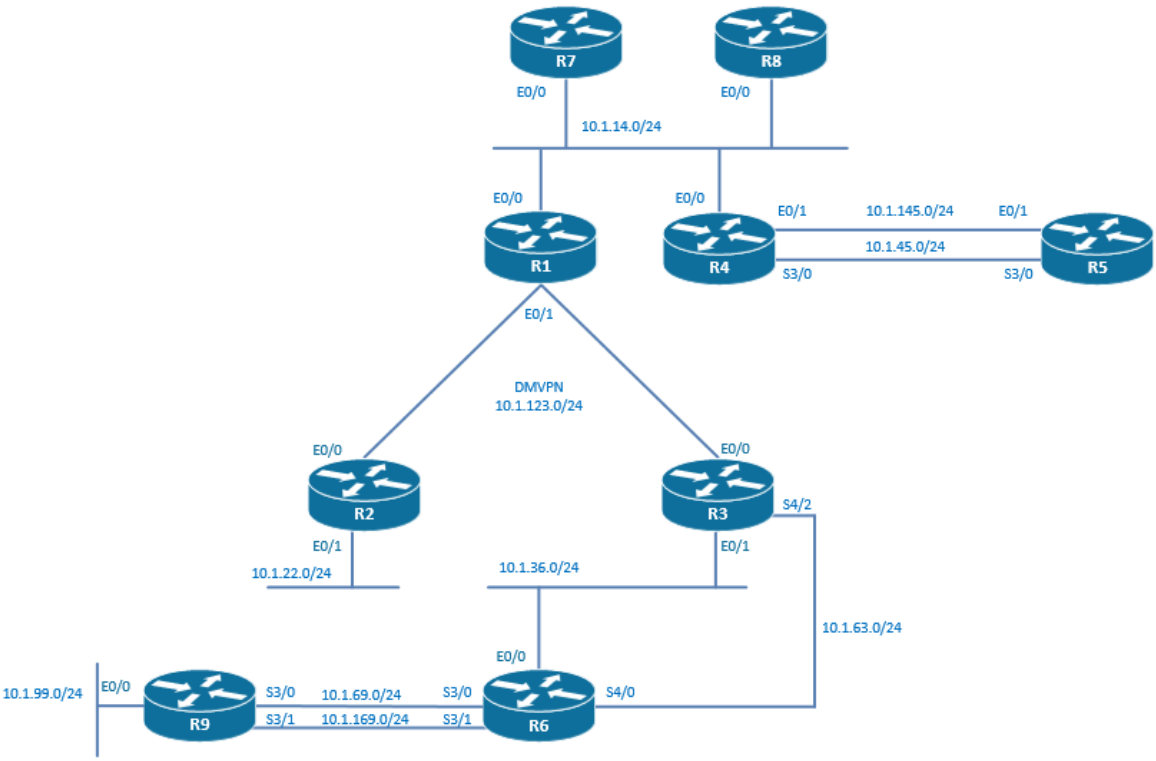
## Technologies covered

- Auto-RP
- Auto-RP filtering
- Auto-RP listener
- Multiple RP candidates
- Multicast boundary
- BSR
- BSR Propagation filtering

## Overview

You have been tasked to configure the multicast routing reachability in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 3 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 28.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN phase 1 without IPSec is the underlying used technology. Setup EIGRP AS 10 in this DMVPN network.

On R1, R2 and R3, configure the following:

```
router eigrp 10
 network 11.1.1.0 0.0.0.255
```

**Task 28.2** Advertise the loopbacks of R1, R2, and R3 in the EIGRP process. Use network statements. Make sure that you can ping from the loopback0 of R2 to the loopback0 of R3.

On R1, configure the following:

```
router eigrp 10
network 10.1.1.1 0.0.0.0
```

On R2, configure the following:

```
router eigrp 10
network 10.1.2.2 0.0.0.0
```

On R3, configure the following:

```
router eigrp 10
network 10.1.3.3 0.0.0.0
```

On R1, configure the following:

```
int tu23
no ip split-horizon eigrp 10
```

**Task 28.3** Extend the EIGRP routing domain to include the network 10.1.14.0/24, the network 10.1.145.0/24, and the network 10.1.45.0/24. Advertise the loopbacks of R7, R8, R4, and R5 in the EIGRP process using network statements.

On R1, R4, R7 and R8, configure the following:

On R1, configure the following:

```
router eigrp 10
 network 10.1.14.0 0.0.0.255
```

On R7, configure the following:

```
router eigrp 10
 network 10.1.7.7 0.0.0.0
 network 10.1.14.0 0.0.0.255
```

On R8, configure the following:

```
router eigrp 10
 network 10.1.8.8 0.0.0.0
 network 10.1.14.0 0.0.0.255
```

On R4, configure the following:

```
router eigrp 10
 network 10.1.4.4 0.0.0.0
 network 10.1.14.0 0.0.0.255
```

```
 network 10.1.45.0 0.0.0.255
 network 10.1.145.0 0.0.0.255
```

On R5, configure the following:

```
router eigrp 10
 network 10.1.5.5 0.0.0.0
 network 10.1.45.0 0.0.0.255
 network 10.1.145.0 0.0.0.255
```

**Task 28.4** Extend the EIGRP routing domain to include the network 10.1.36.0/24, the network 10.1.22.0/24, the network 10.1.63.0/24, the network 10.1.169.0/24, and the network 10.1.69.0/24. Advertise the loopbacks of R6 and R9 in the EIGRP process using network statements.

On R2, configure the following:

```
router eigrp 10
 network 10.1.22.0 0.0.0.255
```

On R3, configure the following:

```
router eigrp 10
 network 10.1.36.0 0.0.0.255
 network 10.1.63.0 0.0.0.255
```

On R6, configure the following:

```
router eigrp 10
 network 10.1.6.6 0.0.0.0
 network 10.1.36.0 0.0.0.255
 network 10.1.63.0 0.0.0.255
 network 10.1.69.0 0.0.0.255
 network 10.1.169.0 0.0.0.255
```

On R9, configure the following:

```
router eigrp 10
 network 10.1.9.9 0.0.0.0
 network 10.1.69.0 0.0.0.255
 network 10.1.169.0 0.0.0.255
```

**Task 28.5** Configure PIM on the 10.1.14.0/24, the network 10.1.145.0/24, and the network 10.1.45.0/24. Auto-RP will be used on those networks. You are not allowed to use “ip pim auto-rp listener” command.

On R1, R4, R7 and R8, configure the following:

```
ip multicast-routing

int e0/0
ip pim sparse-dense-mode
```

On R4, configure the following:

```
ip multicast-routing

int e0/1
ip pim sparse-dense-mode
int s3/0
ip pim sparse-dense-mode
```

On R5, configure the following :

```
ip multicast-routing

int e0/1
```

```
ip pim sparse-dense-mode
int s3/0
ip pim sparse-dense-mode
```

**Task 28.6** Enable R1, R7, and R8 as auto-RP candidates for the following multicast groups: 228.1.1.228, 228.2.2.228, and 228.3.3.228. Their loopback0 should be used in the advertisements.

On R1, configure the following:

```
int lo0
ip pim sparse-dense-mode

access-list 1 permit 228.1.1.228 0.0.0.0
access-list 1 permit 228.2.2.228 0.0.0.0
access-list 1 permit 228.3.3.228 0.0.0.0

ip pim send-rp-announce Loopback0 scope 16 group-list 1
```

On R7, configure the following:

```
int lo0
ip pim sparse-dense-mode

access-list 1 permit 228.1.1.228 0.0.0.0
access-list 1 permit 228.2.2.228 0.0.0.0
access-list 1 permit 228.3.3.228 0.0.0.0

ip pim send-rp-announce Loopback0 scope 16 group-list 1
```

On R8, configure the following:

```
int lo0
ip pim sparse-dense-mode

access-list 1 permit 228.1.1.228 0.0.0.0
access-list 1 permit 228.2.2.228 0.0.0.0
access-list 1 permit 228.3.3.228 0.0.0.0

ip pim send-rp-announce Loopback0 scope 16 group-list 1
```

**Task 28.7** Auto-RP advertisement should be sent every 5 seconds on R1, R7, and R8.

On R1, R7 and R8, configure the following:

```
ip pim send-rp-announce Loopback0 scope 16 group-list 1 interval 5
```

**Task 28.8** R4 should be configured as the mapping agent. The loopback0 has to be used in the advertisements.

On R4, configure the following:

```
int lo0
ip pim sparse-dense-mode
ip pim send-rp-discovery loopback0 scope 16
```

**Task 28.9** Configure the interface E0/1 on R5 to join the group 228.1.1.228 and check that you can ping this multicast group from R7, and that R8 has been chosen to be the PIM DR.

On the R5, configure the following:

```
int e0/1
ip igmp join-group 228.1.1.228
```

We can ping from R7 to the multicast group 228.1.1.228.

```
R7#ping 228.1.1.228
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 228.1.1.228, timeout is 2 seconds:

Reply to request 0 from 10.1.145.5, 5 ms
Reply to request 0 from 10.1.145.5, 20 ms
```

R8 has been chosen to be the PIM DR.

Since all routers have the same priority of 1, the highest IP address has been elected to be PIM DR on this segment.

```
R4#sh 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
10.1.14.8         Ethernet0/0              01:14:51/00:01:42 v2    1 / DR S P G
10.1.14.7         Ethernet0/0              01:14:57/00:01:43 v2    1 / S P G
10.1.14.1         Ethernet0/0              01:15:00/00:01:40 v2    1 / S P G
10.1.145.5        Ethernet0/1              01:13:59/00:01:37 v2    1 / DR S P G
10.1.45.5         Serial3/0                01:13:59/00:01:28 v2    1 / S P G

R8#sh ip pim rp mapp
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)

Group(s) 228.1.1.228/32
  RP 10.1.8.8 (?), v2v1
    Info source: 10.1.4.4 (?), elected via Auto-RP
         Uptime: 00:05:04, expires: 00:02:53
Group(s) 228.2.2.228/32
  RP 10.1.8.8 (?), v2v1
    Info source: 10.1.4.4 (?), elected via Auto-RP
         Uptime: 00:05:04, expires: 00:02:50
Group(s) 228.3.3.228/32
  RP 10.1.8.8 (?), v2v1
    Info source: 10.1.4.4 (?), elected via Auto-RP
         Uptime: 00:05:04, expires: 00:02:53
```

**Task 28.10** Create a “rp-announce-filter” that makes sure that R7 will never become a RP.

On R1, R4, R7 and R8, configure the following:

```
ip pim rp-announce-filter rp-list R7_RP group-list ALL_MULTICAST_GROUPS

ip access-list standard R7_RP
  permit 10.1.7.7

ip access-list standard ALL_MULTICAST_GROUPS
  deny   224.0.0.0 15.255.255.255
```

**Task 28.11** Create 2 “rp-announce-filters” that make sure that R8 will only become the RP for multicast group 228.1.1.228, and that R1 will only become the RP for multicast groups 228.2.2.228 and 228.3.3.228.

On R1, R4, R7 and R8, configure the following:

```
ip pim rp-announce-filter rp-list R1_RP group-list Group1
ip pim rp-announce-filter rp-list R8_RP group-list Group2
```

```
ip access-list standard R8_RP
  permit 10.1.8.8

ip access-list standard R1_RP
  permit 10.1.1.1

ip access-list standard Group1
 permit 228.1.1.228

ip access-list standard Group2
  permit 228.2.2.228
  permit 228.3.3.228
```

**Task 28.12** Configure R1 so that it never does send and receive on interface E0/1 multicast traffic from group 228.1.1.228, 228.2.2.228, and 228.3.3.228. Make sure that the auto-RP advertisements regarding those groups are also filtered.

At this point, PIM is not enabled on the interface ethernet0/1. In the case that it is configured in the future, the filtering is already in place.

On R1, configure the following:

```
access-list 28 deny   228.1.1.228
access-list 28 deny   228.2.2.228
access-list 28 deny   228.3.3.228
access-list 28 permit any

interface Ethernet0/1
  ip multicast boundary 28 filter-autorp
```

**Task 28.13** Configure the interface E0/1 on R5 to join the group 228.2.2.228, and check that you can ping this multicast group from R7, and that R1 has been chosen to be the RP for 228.1.1.228.

On R5, configure the following:

```
int E0/1
ip igmp join-group 228.2.2.228
```

From R7, I can ping the multicast group 228.2.2.228.

```
R7#ping 228.2.2.228
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 228.2.2.228, timeout is 2 seconds:

Reply to request 0 from 10.1.145.5, 46 ms
Reply to request 0 from 10.1.145.5, 46 ms
```

R1 has been chosen to be the RP for 228.1.1.228:

```
R7#sh ip pim rp mapping
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)

Group(s) 228.1.1.228/32
  RP 10.1.1.1 (?), v2v1
    Info source: 10.1.4.4 (?), elected via Auto-RP
         Uptime: 00:03:23, expires: 00:02:31
Group(s) 228.2.2.228/32
  RP 10.1.8.8 (?), v2v1
    Info source: 10.1.4.4 (?), elected via Auto-RP
         Uptime: 00:03:23, expires: 00:02:34
Group(s) 228.3.3.228/32
  RP 10.1.8.8 (?), v2v1
```

```
        Info source: 10.1.4.4 (?), elected via Auto-RP
             Uptime: 00:03:23, expires: 00:02:33
```

**Task 28.14** Ensure that the routers R1, R4, R5, R7, and R8 don't fall back to PIM dense mode for unknown multicast addresses.

On R1, R4, R5, R7 and R8, configure the following:

```
no ip pim dm-fallback
```

**Task 28.15** The 2 connections between R9 and R6 have to be configured with PIM sparse-mode (no PIM sparse-dense mode). R9 has to be configured as an auto-RP candidate for all multicast groups, and R6 has to be configured as the mapping agent.

On R9, configure the following:

```
ip multicast-routing

interface Loopback0
ip pim sparse-mode

int s3/0
ip pim sparse-mode

int s3/1
ip pim sparse-mode

ip pim autorp listener
ip pim send-rp-announce loopback0 scope 16
```

On R6, configure the following interface Loopback0:

```
ip multicast-routing

int lo0
ip pim sparse-mode

int s3/0
ip pim sparse-mode

int s3/1
ip pim sparse-mode

ip pim autorp listener
ip pim send-rp-discovery loopback0 scope 16
```

**Task 28.16** R9 should not become the RP for routers that are more than 1 hop away.

On R9, configure the following:

```
ip pim send-rp-announce Loopback0 scope 1
```

**Task 28.17** Configure the interface S3/0 on R9 to join the group 229.229.229.229, and check that you can ping this multicast group from R6, and that R9 has been chosen to be the RP.

On R9, configure the following:

```
int s3/0
ip igmp join-group 229.229.229.229
```

R9 has be chosen to be the RP for all multicast groups.

```
R9#sh ip pim rp mapping
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)

Group(s) 224.0.0.0/4
  RP 10.1.9.9 (?), v2v1
    Info source: 10.1.6.6 (?), elected via Auto-RP
         Uptime: 00:00:54, expires: 00:02:03
```

I can ping 229.229.229.229 from R6.

```
R6#ping 229.229.229.229
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 229.229.229.229, timeout is 2 seconds:

Reply to request 0 from 10.1.69.9, 31 ms
Reply to request 0 from 10.1.69.9, 31 ms
```

**Task 28.18** Enable PIM sparse mode on all interfaces on the network 11.1.1.0/24.

On R1, R2 and R3, configure the following:

```
ip multicast-routing

interface Tunnel23
ip pim sparse-mode
```

**Task 28.19** Configure R2 as the BSR. Use the interface that is always up on a router.

On R2, configure the following:

```
interface Loopback0
ip pim sparse-mode

ip pim bsr-candidate Loopback0 0
```

**Task 28.20** Configure R1 as the primary RP and configure R3 as a backup RP. One of the two should be configured with the default priority. Use the interfaces that are always up on a router.

On R1, configure the following:

```
interface Loopback0
ip pim sparse-mode

ip pim rp-candidate Loopback0
```

On R3, configure the following:

```
interface Loopback0
ip pim sparse-mode

ip pim rp-candidate Loopback0 priority 30
```

**Task 28.21** Enable PIM sparse mode on the network 10.1.36.0/24 and 10.1.63.0/24. Ensure that R6 doesn't receive information about RPs elected by PIM bootstrap router process.

On R3, configure the following:

```
int s4/2
ip pim bsr-border
ip pim sparse-mode

int e0/1
```

```
ip pim bsr-border
ip pim sparse-mode
```

On R6, configure the following:

```
int s4/0
ip pim sparse-mode

int e0/0
ip pim sparse-mode
```

**Task 28.22** Ensure that R7, R8, and R4 don't receive information about RPs elected by PIM bootstrap router process.

On R1, configure the following:

```
int e0/0
ip pim bsr-border
```

**Task 28.23** Configure the interface E0/1 on R2 to join the group 225.225.225.225 and check that you can ping this multicast group from R1.

On R2, configure the following:

```
int e0/1
ip pim sparse-mode
ip igmp join-group 225.225.225.225
```

I can ping the multicast group 225.225.225.225 from R1.

```
R1#ping 225.225.225.225
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 225.225.225.225, timeout is 2 seconds:

Reply to request 0 from 10.1.22.2, 6 ms
Reply to request 0 from 10.1.22.2, 6 ms
```

On R1, you will be seeing this error message popping up:

```
%PIM-6-INVALID_RP_JOIN: Received (*, 225.225.225.225) Join from 11.1.1.2 for invalid RP
10.1.1.1
```

This is due to the fact that the router R1 is running auto-RP and bootstrap router at the same time. The auto-RP feature has a filter configured that authorizes only the RP 10.1.1.1 to be RP for the group 228.1.1.228. This error message is coming from the auto-RP process. The bootstrap process is not affected and is using 10.1.1.1 as an RP.

**You have completed Lab 28**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 29: Configure and troubleshoot Protocol Independent Multicast Operations (Part 3)

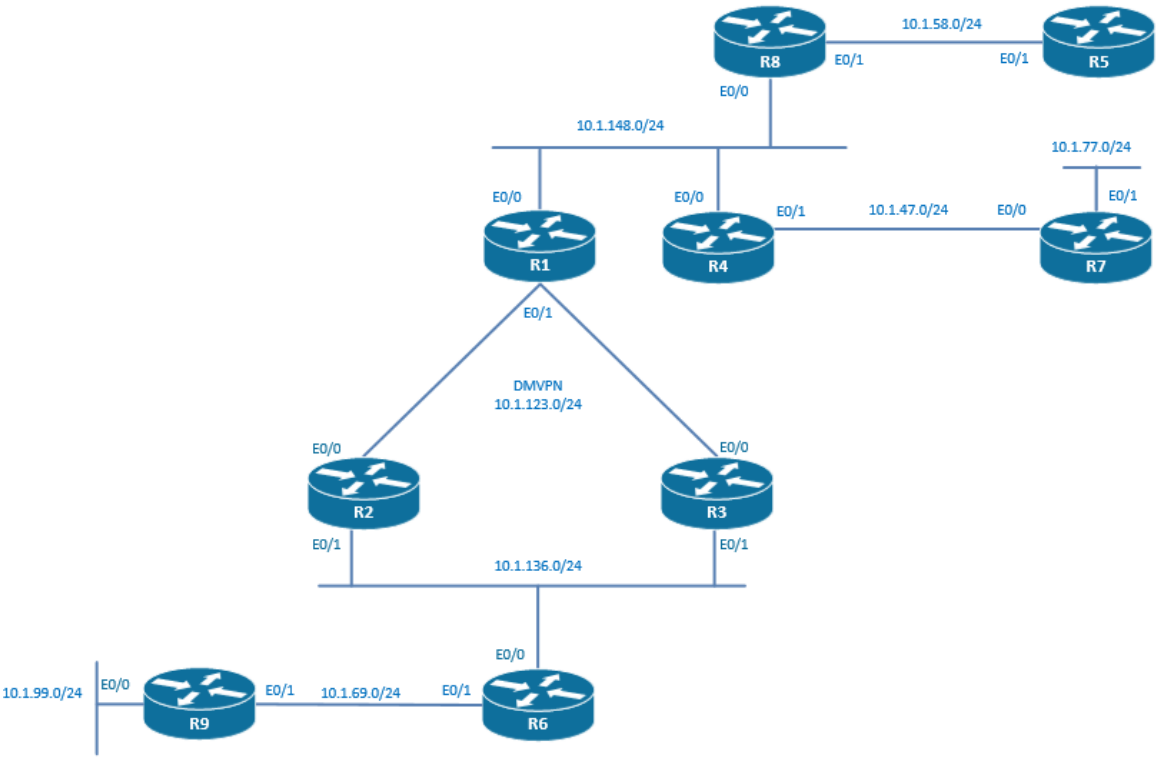
## Technologies covered

- Multicast stub routing
- IP IGMP helper-address
- SSM
- IGMP filtering
- IGMP timers
- Multicast helper map
- PIM bidirectional
- Multicast rate limiting

## Overview

You have been tasked to configure the multicast routing reachability in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 29.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. DMVPN phase 1 without IPSec is the underlying used technology. Setup OSPF area 0 in this DMVPN network. Use the point-to-multipoint OSPF type on the 2 two spokes.

On R1, configure the following:

```
interface Tunnel23
ip ospf network point-to-multipoint

router ospf 1
network 11.1.1.0 0.0.0.255 area 0
```

On R2, configure the following:

```
interface Tunnel23
ip ospf network point-to-multipoint

router ospf 1
network 11.1.1.0 0.0.0.255 area 0
```

On R3, configure the following:

```
interface Tunnel23
ip ospf network point-to-multipoint

router ospf 1
network 11.1.1.0 0.0.0.255 area 0
```

**Task 29.2** Advertise the loopbacks of R1, R2, and R3 in the OSPF process. Use network statements. Make sure that you can ping from the loopback0 of R2 to the loopback0 of R3.

On R1, configure the following:

```
router ospf 1
router-id 10.1.1.1
network 10.1.1.1 0.0.0.0 area 0
```

On R2, configure the following:

```
router ospf 1
router-id 10.1.2.2
network 10.1.2.2 0.0.0.0 area 0
```

On R3, configure the following:

```
router ospf 1
router-id 10.1.3.3
network 10.1.3.3 0.0.0.0 area 0
```

**Task 29.3** Introduce R4, R5, R7, R8, R6, and R9 into the OSPF area 0. Advertise the loopbacks of R4, R5, R7, R8, R6, and R9 in the OSPF process. Use network statements. Make sure that you can ping from the loopback0 of R7 to the loopback0 of R9.

On R1, configure the following:

```
router ospf 1
network 10.1.148.0 0.0.0.255 area 0
```

On R2, configure the following:
```
router ospf 1
network 10.1.136.0 0.0.0.255 area 0
```

On R3, configure the following:
```
router ospf 1
network 10.1.136.0 0.0.0.255 area 0
```

On R4, configure the following:
```
router ospf 1
router-id 10.1.4.4
network 10.1.4.4 0.0.0.0 area 0
network 10.1.148.0 0.0.0.255 area 0
network 10.1.47.0 0.0.0.255 area 0
```

On R5, configure the following:
```
router ospf 1
router-id 10.1.5.5
network 10.1.5.5 0.0.0.0 area 0
network 10.1.58.0 0.0.0.255 area 0
```

On R7, configure the following:
```
router ospf 1
router-id 10.1.7.7
network 10.1.7.7 0.0.0.0 area 0
network 10.1.47.0 0.0.0.255 area 0
```

On R8, configure the following:
```
router ospf 1
router-id 10.1.8.8
network 10.1.8.8 0.0.0.0 area 0
network 10.1.148.0 0.0.0.255 area 0
network 10.1.58.0 0.0.0.255 area 0
```

On R6, configure the following:
```
router ospf 1
router-id 10.1.6.6
network 10.1.6.6 0.0.0.0 area 0
network 10.1.136.0 0.0.0.255 area 0
network 10.1.69.0 0.0.0.255 area 0
```

On R9, configure the following:
```
router ospf 1
router-id 10.1.9.9
network 10.1.9.9 0.0.0.0 area 0
network 10.1.69.0 0.0.0.255 area 0
```

**Task 29.4** Advertise the networks 10.1.77.0/24 and 10.1.99.0/24 in the OSPF process. Use network statements. Make sure that no OSPF neighborships will never be formed on those networks.

On R7, configure the following:
```
router ospf 1
network 10.1.77.0 0.0.0.255 area 0
passive-interface e0/1
```

On R9, configure the following:
```
router ospf 1
network 10.1.99.0 0.0.0.255 area 0
passive-interface e0/0
```

**Task 29.5** Configure PIM sparse mode on the networks 10.1.69.0/24, 10.1.136.0/24, 11.1.1.0/24, and 10.1.148.0/24.

On R8, configure the following:

```
ip multicast-routing
int e0/0
ip pim sparse-mode
```

On R1, configure the following:

```
ip multicast-routing
int tu23
ip pim sparse-mode

int e0/0
ip pim sparse-mode
```

On R4, configure the following:

```
ip multicast-routing

int e0/0
ip pim sparse-mode
```

On R2, configure the following:

```
ip multicast-routing

int tu23
ip pim sparse-mode

int e0/1
ip pim sparse-mode
```

On R3, configure the following:

```
ip multicast-routing

int tu23
ip pim sparse-mode

int e0/1
ip pim sparse-mode
```

On R6, configure the following:

```
ip multicast-routing

int e0/0
ip pim sparse-mode

int e0/1
ip pim sparse-mode
```

On R9, configure the following:

```
ip multicast-routing

int e0/1
ip pim sparse-mode
```

**Task 29.6** Configure IP PIM dense mode on the network 10.1.47.0/24. No PIM adjacency should be formed over this connection. Use the command “ip pim neighbor-filter” on R4.

On R4, configure the following:

```
ip multicast-routing

int e0/1
```

```
ip pim dense-mode
ip pim neighbor-filter 1

access-list 1 deny   10.1.47.7
access-list 1 permit any
```

On R7, configure the following:

```
ip multicast-routing

int e0/0
ip pim dense-mode
```

**Task 29.7** The source of the multicast stream 224.2.2.2 is located on the VLAN 77. The receiver of this multicast stream is on the VLAN 10.1.148.0/24. Enable multicast connectivity between this source and this receiver. You are not allowed to remove the filter configured in the previous question, and consequently not allowed to build a PIM adjacency over the connection between R4 and R7. On R1, R4, R7, and R8, configure statically the loopback0 of R1 as the RP for all multicast groups.

On R1, configure the following:

```
ip pim rp-address 10.1.1.1

int lo0
ip pim sparse-mode
```

On R4, R7 and R8, configure the following:

```
ip pim rp-address 10.1.1.1
```

On R7, configure the following:

```
int e0/1
ip pim dense-mode
ip igmp helper-address 10.1.47.1
```

**Task 29.8** Configure the interface E0/1 on R7 to join the group 224.2.2.2 and check that you can ping this multicast group from R8.

On R7, configure the following:

```
int e0/1
   ip igmp join-group 224.2.2.2
```

On R1, I can ping the multicast address 224.2.2.2.

```
R1#ping 224.2.2.2
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.2.2.2, timeout is 2 seconds:

Reply to request 0 from 10.1.77.7, 30 ms
Reply to request 0 from 10.1.77.7, 31 ms
Reply to request 0 from 10.1.77.7, 31 ms
Reply to request 0 from 10.1.77.7, 30 ms
Reply to request 0 from 10.1.77.7, 30 ms
```

**Task 29.9** Make sure that interface E0/1 on R5 can receive traffic multicast for the group 224.3.3.3 only if it is sourced from the loopback0 of R1. Do not enable PIM on this interface.

On R8, configure the following:

```
int e0/1
ip pim sparse-mode
ip igmp version 3
```

On R1 and R8, configure the following:

```
ip pim ssm default
```

On R5, configure the following:

```
int e0/1
ip igmp join-group 224.3.3.3 source 10.1.1.1
ip igmp version 3
```

**Task 29.10** Verify that you can ping this multicast group 224.3.3.3 from R1 only when the ping is sourced from the loopback0 of R1.

On R8, we can see that the IGMPv3 group has been registered:

```
R8#show ip igmp groups 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:      Ethernet0/1
Group:          224.3.3.3
Flags:
Uptime:         00:17:27
Group mode:     INCLUDE
Last reporter:  10.1.58.5
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
  10.1.1.1         00:17:27  00:02:41  stopped   Yes  R

Interface:      Ethernet0/0
Group:          224.0.1.40
Flags:          L U
Uptime:         01:15:45
Group mode:     EXCLUDE (Expires: 00:02:42)
Last reporter:  10.1.148.8
Source list is empty
```

On R1, the ping to the destination 224.3.3.3 is working with the source of 10.1.1.1.

```
R1#ping 224.3.3.3 source 10.1.1.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.3.3.3, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1

Reply to request 0 from 10.1.58.5, 31 ms
Reply to request 0 from 10.1.58.5, 31 ms
```

**Task 29.11** R9 has to be protected from an IGMP DOS attack. On the interface E0/0 of R9, allow the maximum number of IGMP states to be 25.

On R9, configure the following:

```
int e0/0
ip igmp limit 25
```

**Task 29.12** R6 should only accept on the interface E0/1 multicast clients that want to join a group in the range 225.0.0.0/8.

On R6, configure the following:

```
access-list 1 permit 225.0.0.0 0.255.255.255

interface E0/1
ip igmp access-group 1
```

**Task 29.13** Configure interface E0/1 of R9 to join multicast groups 225.2.2.2 and 226.2.2.2. Check on R6 that the filtering configured in the previous question is working.

On R9, configure the following:

```
interface E0/1
ip igmp join-group 225.2.2.2
ip igmp join-group 226.2.2.2
```

The filtering configured in the previous question is working because only IGMP reports for 225.2.2.2 has been accepted on R6.

```
R6#sh ip igmp membership
Flags: A  - aggregate, T - tracked
       L  - Local, S - static, V - virtual, R - Reported through v3
       I - v3lite, U - Urd, M - SSM (S,G) channel
       1,2,3 - The version of IGMP, the group is in
Channel/Group-Flags:
       / - Filtering entry (Exclude mode (S,G), Include mode (G))
Reporter:
       <mac-or-ip-address> - last reporter if group is not explicitly tracked
       <n>/<m>      - <n> reporter in include mode, <m> reporter in exclude

 Channel/Group                 Reporter        Uptime   Exp.  Flags  Interface
 *,225.2.2.2                   10.1.69.9       00:00:15 02:44 2A     Et0/1
 *,224.0.1.40                  10.1.136.6      02:26:34 02:45 2LA    Et0/0
```

**Task 29.14** On the network 10.1.99.0, there is only one client receiving several multicast streams. As soon as this client is sending an IGMP leave group message, the router should immediately stop forwarding this multicast stream on the LAN and not try to send a group-specific query for this multicast group.

On R9, configure the following:

```
ip access-list standard ALL_MULTICAST_GROUPS
  permit 224.0.0.0 15.255.255.255

interface Ethernet0/0
  ip igmp immediate-leave group-list ALL_MULTICAST_GROUPS
```

**Task 29.15** On the VLAN 136, configure IGMP to send membership queries every 30 seconds. The backup querier should become the querier for this LAN if it hasn't seen a query packet within 1 minute.

On R2, configure the following:

```
int e0/1
ip igmp query-interval 30
```

On R3, configure the following:

```
int e0/1
ip igmp query-interval 30
ip igmp querier-timeout 60
```

On R6, configure the following:

```
int e0/0
ip igmp query-interval 30
```

The querier for multicast is elected based on the lowest IP address on the link so the querier is the router R2 and the backup querier will be the router R3.

**Task 29.16** On R9, IGMP protocol should communicate to the multicast clients that they should report their group's membership in a maximum of 30 seconds after receiving a query.

On R9, configure the following:

```
int e0/0
ip igmp query-interval 30

int e0/1
ip igmp query-interval 30
```

**Task 29.17** There is a server that is connected to the network 10.1.136.0/24. This server is sending broadcast UDP traffic to port 2500 to a client connected to the network 10.1.148.0/24. This broadcast traffic should be transported by the multicast group 227.7.7.7 when crossing the connection between R2 and R1, and the connection between R3 and R1.

On R2 and on R3, configure the following:

```
ip forward-protocol udp 2500

ip access-list extended UDP2500
  permit udp any any eq 2500

interface Ethernet0/1
  ip multicast helper-map broadcast 227.7.7.7 UDP2500
```

**Task 29.18** The multicast traffic should be converted back to a broadcast when reaching the network 10.1.148.0/24.

On R1, configure the following:

```
ip access-list extended UDP2500
permit udp any any eq 2500

ip forward-protocol udp 2500

interface Ethernet0/0
ip broadcast-address 10.1.148.255
ip directed-broadcast

interface tu23
ip multicast helper-map 227.7.7.7 10.1.148.255 UDP2500
```

**Task 29.19** Configure bidirectional PIM for a multicast stream of 224.22.22.22 on the network 11.1.1.0/24 and 10.1.148.0/24. The loopback0 of the R1 has to be configured as the RP and the mapping agent in this PIM bidirectional setup.

On R1, configure the following:

```
ip pim bidir-enable

ip access-list standard NET_22
  permit 224.22.22.22
```

```
ip pim rp-candidate Loopback0 group-list NET_22 bidir
ip pim bsr-candidate Loopback0 0
```

On R2, R3, R4 and R8, configure the following:

```
ip pim bidir-enable
```

**Task 29.20** Configure R6 to limit to total bandwidth for multicast traffic to 20 M on all its interfaces in the egress direction.

The interface command “ip multicast rate-limit out” is no longer supported. Therefore we have to use the modular QOS configuration.

On R6, configure the following:

```
access-list 66 permit 224.0.0.0 15.255.255.255

class-map MULTICAST
 match access-group 66

policy-map MULTICAST
 class MULTICAST
    police 20m 1000 conform-action transmit exceed-action drop

interface e0/0
service-policy output MULTICAST
interface e0/1
service-policy output MULTICAST
```

**Task 29.21** Configure R1 to limit to 5M the bandwidth that the multicast stream with a destination of 224.22.22.22 can use out of the tunnel interface.

On R1, configure the following:

```
ip access-list 33 permit 224.22.22.22 0.0.0.0

class-map MULTICAST
 match access-group 33

policy-map MULTICAST
 class MULTICAST
    police 5m 1000 conform-action transmit exceed-action drop

interface e0/0
```

**You have completed Lab 29**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 30: Configure and troubleshoot Protocol Independent Multicast Operations (Part 4)

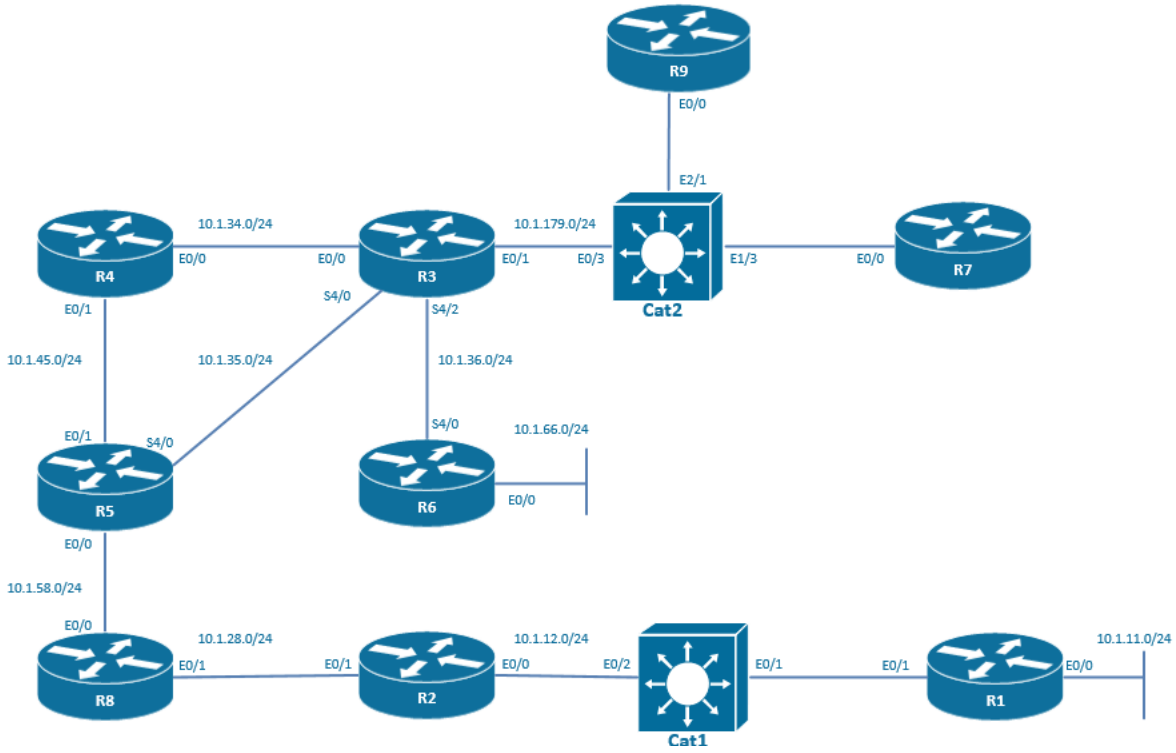
## Technologies covered

- RPF failure
- Multicast BGP extension
- BSR propagation filtering
- MSDP
- Catalyst IGMP snooping

## Overview

You have been tasked to configure the multicast routing reachability in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 3 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 30.1** Configure OSPF area 0 routing on the ethernet connections between R5 and R4, R4 and R3, and on the serial connection between R3 and R6. Advertise the loopbacks of R5, R4, R3, and R6 in the OSPF process. Use network statements.

On R5, configure the following:

```
router ospf 1
 network 10.1.5.5 0.0.0.0 area 0
 network 10.1.45.0 0.0.0.255 area 0
```

On R4, configure the following:

```
router ospf 1
 network 10.1.4.4 0.0.0.0 area 0
 network 10.1.34.0 0.0.0.255 area 0
 network 10.1.45.0 0.0.0.255 area 0
```

On R3, configure the following:

```
router ospf 1
 network 10.1.3.3 0.0.0.0 area 0
 network 10.1.34.0 0.0.0.255 area 0
 network 10.1.36.0 0.0.0.255 area 0
```

On R6, configure the following:

```
router ospf 1
 network 10.1.6.6 0.0.0.0 area 0
 network 10.1.36.0 0.0.0.255 area 0
```

**Task 30.2** Configure PIM sparse-mode on the ethernet connections between R5 and R4, R4 and R3, and R3 and R6.

On R5, configure the following:

```
ip multicast-routing
int E0/1
ip pim sparse-mode
```

On R4, configure the following:

```
ip multicast-routing

int E0/1
ip pim sparse-mode

int E0/0
ip pim sparse-mode
```

On R3, configure the following:

```
ip multicast-routing

int E0/0
ip pim sparse-mode

int S4/2
ip pim sparse-mode
```

On R6, configure the following:

```
ip multicast-routing
int S4/0
ip pim sparse-mode
```

**Task 30.3** R3 should be configured as the BSR and the RP for the all multicast groups. Use the PIM bootstrap router solution to advertise the RP. Use the loopback 0 of R3 as the RP IP address.

On R3, configure the following:

```
int lo0
ip pim sparse-mode

ip pim bsr-candidate Loopback0 0
ip pim rp-candidate Loopback0
```

Let's check that R5 has got R3 configured as a Rendezvous Point.

```
R5#sh ip pim rp mapping
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 10.1.3.3 (?), v2
    Info source: 10.1.3.3 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:02:30, expires: 00:02:00
```

**Task 30.4** On R5, configure on the interface E0/0 an IGMP join for the group 225.7.7.7. Verify that you can ping from R6 to the multicast group 225.7.7.7.

On R5, configure the following:

```
int e0/0
ip igmp join-group 225.7.7.7
```

When simulating a multicast flow from R6, this flow is reaching the destination on R5.

```
R6#ping 225.7.7.7
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 225.7.7.7, timeout is 2 seconds:

Reply to request 0 from 10.1.58.5, 11 ms
Reply to request 0 from 10.1.58.5, 35 ms
Reply to request 0 from 10.1.58.5, 35 ms
Reply to request 0 from 10.1.58.5, 35 ms
Reply to request 0 from 10.1.58.5, 35 ms
Reply to request 0 from 10.1.58.5, 30 ms
Reply to request 0 from 10.1.58.5, 30 ms
Reply to request 0 from 10.1.58.5, 30 ms
Reply to request 0 from 10.1.58.5, 29 ms

Multicast routing is working fine. We can check the ip mroute table.

R6#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
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 225.7.7.7), 00:02:11/stopped, RP 10.1.3.3, flags: SPF
```

```
    Incoming interface: Serial4/0, RPF nbr 10.1.36.3
    Outgoing interface list: Null

  (10.1.36.6, 225.7.7.7), 00:02:11/00:00:47, flags: PFT
    Incoming interface: Serial4/0, RPF nbr 0.0.0.0, Registering
    Outgoing interface list: Null

  (*, 224.0.1.40), 00:18:26/stopped, RP 0.0.0.0, flags: DCL
    Incoming interface: Null, RPF nbr 0.0.0.0
    Outgoing interface list:
      Serial4/0, Forward/Sparse, 00:18:26/stopped
```

**Task 30.5** Configure OSPF area 0 routing on the serial connection between R5 and R3. Do not enable PIM on this link.

On R5 and R3, configure the following:

```
router ospf 1
network 10.1.35.0 0.0.0.255 area 0
```

**Task 30.6** Manipulate this OSPF cost to ensure that the direct link between R5 and R3 is the preferred path for OSPF.

Let's see the current path used to go from R5 to R3 and from R5 to R3.

```
R5#sh ip route 10.1.3.3
Routing entry for 10.1.3.3/32
  Known via "ospf 1", distance 110, metric 21, type intra area
  Last update from 10.1.45.4 on Ethernet0/1, 00:38:51 ago
  Routing Descriptor Blocks:
  * 10.1.45.4, from 10.1.3.3, 00:38:51 ago, via Ethernet0/1
      Route metric is 21, traffic share count is 1

R3#sh ip route 10.1.5.5
Routing entry for 10.1.5.5/32
  Known via "ospf 1", distance 110, metric 21, type intra area
  Last update from 10.1.34.4 on Ethernet0/0, 00:39:26 ago
  Routing Descriptor Blocks:
  * 10.1.34.4, from 10.1.5.5, 00:39:26 ago, via Ethernet0/0
      Route metric is 21, traffic share count is 1
```

To go from R5 to R3 or to go from R3 to R5, the traffic is routed via R4. The total metric of the lowest cost route is 21 in both directions. If I'm configuring an OSPF metric of 1 on the direct link between R5 and R3, this route should become the preferred one.

On R3 and R5, configure the following:

```
int S4/0
ip ospf cost 1
```

Re-routing has occurred and the route from R5 to R3 is now using the direct circuit in both directions.

```
R5#sh ip route 10.1.3.3
Routing entry for 10.1.3.3/32
  Known via "ospf 1", distance 110, metric 2, type intra area
  Last update from 10.1.35.3 on Serial4/0, 00:00:24 ago
  Routing Descriptor Blocks:
  * 10.1.35.3, from 10.1.3.3, 00:00:24 ago, via Serial4/0
      Route metric is 2, traffic share count is 1

R3#sh ip route 10.1.5.5
Routing entry for 10.1.5.5/32
  Known via "ospf 1", distance 110, metric 2, type intra area
  Last update from 10.1.35.5 on Serial4/0, 00:00:20 ago
  Routing Descriptor Blocks:
  * 10.1.35.5, from 10.1.5.5, 00:00:20 ago, via Serial4/0
```

```
        Route metric is 2, traffic share count is 1
```

**Task 30.7** Verify that you cannot ping from R6 to the multicast group 225.7.7.7 because of a RPF failure. To solve the RPF failure, you are not allowed to configure ip mroutes.

We cannot ping anymore from R6 to the multicast group 225.7.7.7

```
R6#ping 225.7.7.7
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 225.7.7.7, timeout is 2 seconds:
```

The feed does not reach the subnet 10.1.58.0/24. This is happening because the RPF check is failed. When R5 is communicating with the router R3 which signaled as the RP, it is using the direct path which is not PIM enabled.

This failure situation could be fixed by configuring a static “ip mroute” on R5 towards the RP with R4 as the next-hop and on R3 a static “ip mroute” towards the network 10.1.58.0 255.255.255.0 with R4 as the next-hop. However, this solution is not valid because it is forbidden by the question.

In order to fix the RPF check failure, we are going to enable PIM on the connection between R3 and R5.

On R5 and R3, configure the following:

```
int s4/0
ip pim sparse-mode

The solution is working. The feed 225.7.7.7 is again reaching the receiver 10.1.58.5.

R6#ping 225.7.7.7 repeat 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 225.7.7.7, timeout is 2 seconds:

Reply to request 0 from 10.1.58.5, 51 ms
Reply to request 0 from 10.1.58.5, 51 ms
Reply to request 1 from 10.1.58.5, 14 ms
Reply to request 1 from 10.1.58.5, 18 ms
Reply to request 2 from 10.1.58.5, 18 ms
Reply to request 2 from 10.1.58.5, 18 ms
Reply to request 3 from 10.1.58.5, 13 ms
Reply to request 3 from 10.1.58.5, 13 ms
Reply to request 4 from 10.1.58.5, 14 ms
Reply to request 4 from 10.1.58.5, 14 ms
Reply to request 5 from 10.1.58.5, 14 ms
Reply to request 5 from 10.1.58.5, 14 ms
```

**Task 30.8** We are going to use multicast BGP. Remove OSPF from all the routers where it is running and shut down the direct connection between R5 and R3.

On R5, R4, R3, and R6, configure the following:

```
no router ospf 1
```

On R5 and R3, configure the following:

```
int S4/0
shut
```

**Task 30.9** Configure an iBGP peering between R5 and R4 in AS20. Use the Physical IP addresses for the peering's.

On R4, configure the following:
```
router bgp 20
 neighbor 10.1.45.5 remote-as 20
```

On R5, configure the following:
```
router bgp 20
 neighbor 10.1.45.4 remote-as 20
```

**Task 30.10** Configure an iBGP peering between R3 and R6 in AS10. Use the Physical IP addresses for the peering's.

On R3, configure the following:
```
router bgp 10
 neighbor 10.1.36.6 remote-as 10
```

On R6, configure the following:
```
router bgp 10
 neighbor 10.1.36.3 remote-as 10
```

**Task 30.11** Configure an eBGP peering between R4 and R3. Use the Physical IP addresses for the peering's.

On R4, configure the following:
```
router bgp 20
neighbor 10.1.34.3 remote-as 10
```

On R3, configure the following:
```
router bgp 10
neighbor 10.1.34.4 remote-as 20
```

**Task 30.12** Configure on each BGP router an "address-family ipv4 multicast". Advertise all the circuits where there is a PIM neighborship into BGP with network statements.

On R3, configure the following:
```
router bgp 10

 address-family ipv4 multicast
  neighbor 10.1.36.6 activate
  neighbor 10.1.36.6 next-hop-self
  neighbor 10.1.34.4 activate
  network 10.1.36.0 mask 255.255.255.0
  network 10.1.34.0 mask 255.255.255.0
```

On R6, configure the following:
```
router bgp 10
 address-family ipv4 multicast
  neighbor 10.1.36.3 activate
  neighbor 10.1.36.3 next-hop-self
  network 10.1.36.0 mask 255.255.255.0
```

On R4, configure the following:
```
router bgp 20

address-family ipv4 multicast
```

```
 neighbor 10.1.34.3 activate
 neighbor 10.1.45.5 activate
 neighbor 10.1.45.5 next-hop-self
 network 10.1.45.0 mask 255.255.255.0
 network 10.1.34.0 mask 255.255.255.0
```

On the router R5, configure the following:

```
router bgp 20
address-family ipv4 multicast
  neighbor 10.1.45.4 activate
  neighbor 10.1.45.4 next-hop-self
  network 10.1.45.0 mask 255.255.255.0
```

**Task 30.13** Advertise the RP IP address into the address-family used for multicast.

On R3, configure the following:

```
router bgp 10
address-family ipv4 multicast
  network 10.1.3.3 mask 255.255.255.255
```

**Task 30.14** Verify that the feed from R6 to the multicast group 225.7.7.7 is again reaching R5 after the migration from OSPF to BGP.

Let's try to ping the multicast group 225.7.7.7 from R6. We have to take into account that only multicast routing has been configured with BGP multicast and that the ping has no unicast routes to send back the echo reply. The ping is not working but it doesn't mean that the multicast stream is not routed to the receiver. A multicast is unidirectional going from the source to the receiver.

```
R6#ping 225.7.7.7 repeat 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 225.7.7.7, timeout is 2 seconds:
.....................
```

Let's enable debug ip icmp on R5.

```
ICMP: echo reply sent, src 10.1.58.5, dst 10.1.36.6, topology BASE, dscp 0 topoid 0
ICMP: echo reply sent, src 10.1.58.5, dst 10.1.36.6, topology BASE, dscp 0 topoid 0
```

We can see that the multicast ping is reaching R5 that is replying with a unicast echo rpy which is not routed by the network because we got rid of the IGP OSPF.

```
The solution is working. The feed 225.7.7.7 is again reaching the receiver 10.1.58.5.
```

**Task 30.15** On Cat1, configure the E0/1 and the E0/2 interfaces into VLAN 12.

On Cat1, configure the following:

```
vtp mode transparent
vlan 12

interface Ethernet0/1
 switchport access vlan 12
 switchport mode access
 duplex auto

interface Ethernet0/2
 switchport access vlan 12
 switchport mode access
 duplex auto
```

**Task 30.16** Configure OSPF area 0 routing on the connection between R5 and R8, on the connection between R8 and R2, and on the connection between R1 and R2.

On R5, configure the following:

```
router ospf 1
 network 10.1.5.5 0.0.0.0 area 0
 network 10.1.58.0 0.0.0.255 area 0
```

On R8, configure the following:

```
router ospf 1
 network 10.1.8.8 0.0.0.0 area 0
 network 10.1.58.0 0.0.0.255 area 0
 network 10.1.28.0 0.0.0.255 area 0
```

On R2, configure the following:

```
router ospf 1
 network 10.1.2.2 0.0.0.0 area 0
 network 10.1.28.0 0.0.0.255 area 0
 network 10.1.12.0 0.0.0.255 area 0
```

On R1, configure the following:

```
router ospf 1
 network 10.1.1.1 0.0.0.0 area 0
 network 10.1.12.0 0.0.0.255 area 0
```

**Task 30.17** Configure PIM in sparse mode on the connection between R5 and R8, on the connection between R8 and R2, and on the connection between R1 and R2.

On R5, configure the following:

```
ip multicast-routing

int E0/0
ip pim sparse-mode
```

On R8, configure the following:

```
ip multicast-routing

int E0/1
ip pim sparse-mode

int E0/0
ip pim sparse-mode
```

On R2, configure the following:

```
ip multicast-routing

int E0/0
ip pim sparse-mode

int E0/1
ip pim sparse-mode
```

On R1, configure the following:

```
ip multicast-routing

int E0/1
ip pim sparse-mode
```

**Task 30.18** R2 should be configured as the BSR and the RP for the all multicast groups. Use the PIM bootstrap router solution to advertise the RP. Use the loopback 0 of R2 as the RP IP address.

On R2, configure the following:

```
int lo0
ip pim sparse-mode

ip pim bsr-candidate Loopback0 0
ip pim rp-candidate Loopback0
```

**Task 30.19** Separate the two BSR domains and make sure that the propagation of the BSR packets is filtered on the connection between R5 and R8.

On R5 and R8, configure the following:

```
int e0/0
ip pim bsr-border
```

**Task 30.20** On R6, configure on the interface E0/0 an IGMP join for the group 228.7.7.7. Make sure that when you ping from R4 to the group 228.7.7.7, the router R6 is replying.

On R6, configure the following:

```
int E0/0
ip igmp join-group 228.7.7.7
```

We are in the same situation as earlier, that is to say the router R6 is going to reply to the ping but the response will not be able to reach R4 because of the lack of IGP. The only way to see that the ping has reached the receiver is to debug ICMP.

**Task 30.21** On R1, configure on the interface E0/0 an IGMP join for the group 228.7.7.7. Make sure that when you ping from R4 to the group 228.7.7.7, the router R6 and R1 are replying. Use MSDP. Enable OSPF process 2 on the R3, R4, and R5 path.

On R1, configure the following:

```
int E0/0
ip igmp join-group 228.7.7.7
```

The 2 receivers for the feed 228.7.7.7 are located in two different BSR domains, each with a RP. In order to have those 2 domains exchange information about the receivers, we have to configure a MSDP peering between the RPs of the different domains.

On R2, configure the following:

```
ip msdp peer 10.1.3.3 connect-source loopback 0
```

On R3, configure the following:

```
ip msdp peer 10.1.2.2 connect-source loopback 0
```

In order for MSDP peering to come up, we have to enable unicast routing between R3 and R2 using the R3, R4, R5, R8 and R2 path.

On R3, configure the following:

```
router ospf 2
 network 10.1.3.3 0.0.0.0 area 0
 network 10.1.34.0 0.0.0.255 area 0
```

On R4, configure the following:
```
router ospf 2
 network 10.1.34.0 0.0.0.255 area 0
 network 10.1.45.0 0.0.0.255 area 0
```

On R5, configure the following:
```
router ospf 2
network 10.1.45.0 0.0.0.255 area 0
redistribute ospf 1 subnets

router ospf 1
redistribute ospf 2 subnets
```

The MSDP peering between R2 and R3 is coming up.
```
%MSDP-5-PEER_UPDOWN: Session to peer 10.1.2.2 going up
```

**Task 30.22** As soon as there is one receiver for a multicast group on VLAN 12 connected to Cat1, this multicast group stream should be replicated on all the ports in VLAN 12 even if the servers connected to those ports are not multicast listeners.

On Cat1, configure the following:
```
no ip igmp snooping vlan 12 < currently not supported on iPexpert POD
```

**Task 30.23** On Cat2, configure the E0/3, E1/3, and the E2/1 interfaces into VLAN 99.

On Cat2, configure the following:
```
vtp mode transparent
vlan 99

int E0/3
switchport
switchport mode access
switchport access vlan 99

int E1/3
switchport
switchport mode access
switchport access vlan 99

int E2/1
switchport
switchport mode access
switchport access vlan 99
```

**Task 30.24** Configure R3 as the PIM DR for the network 10.1.179.0/24.

We have first to enable PIM on the network 10.1.179.0/24.
On R3, configure the following:
```
int e0/1
ip pim sparse-mode
```

On R7 and R9, configure the following:
```
ip multicast-routing
int e0/0
ip pim sparse-mode
```

By default the default priority is 1 and the highest IP address determines who will become the DR. The DR is responsible for registering the source at the rendezvous point. The router with the highest priority becomes the DR.

On R3, configure the following:

```
int e0/1
ip pim dr-priority 100
```

**Task 30.25** Configure IGMP on Cat2 to prevent R7 to join group 229.7.7.7.

On Cat2, configure the following:

```
ip igmp profile 4 < currently not supported on iPexpert POD
permit
range 229.7.7.7

interface E1/3
ip igmp filter 4
```

**Task 30.26** On R7, configure on the interface E0/0 an IGMP join for the group 229.7.7.7. On R9, configure on the interface E0/0 an IGMP join for the group 229.7.7.7. On Cat2, verify that the IGMP filtering configured in the previous question is working.

On R7 and R9, configure the following:

```
Int e0/0
ip igmp join-group 229.7.7.7
```

**You have completed Lab 30**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com

# Lab 31: Configure and troubleshoot IP version 6 (Part 1)

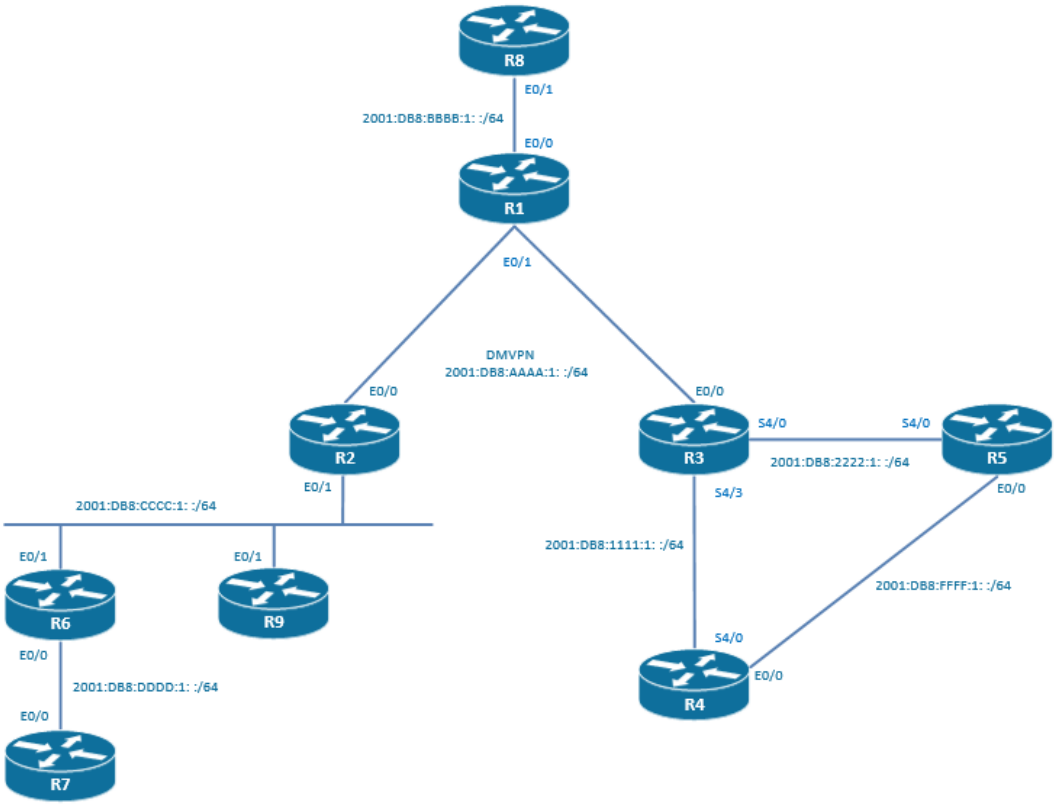
## Technologies covered

- IPv6 addressing
- DMVPN for IPv6
- RIPng
- RIPng prefix filtering
- RIPng summarization
- RIPng offset-list
- RIPng default route

## Overview

You have been tasked to configure the IPv6 routing in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 31.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. Configure the DMVPN phase 3 tunnel infrastructure for IPv6. Do not implement encryption. Use the following addresses:

| | | |
|---|---|---|
| R1 | E0/1 | 10.1.123.1/24 |
| R2 | E0/0 | 10.1.123.2/24 |
| R3 | E0/0 | 10.1.123.3/24 |

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R1 interface Tunnel23 | FE80: :1 | 2001:DB8:AAAA:1: :1/64 |
| R2 interface Tunnel23 | FE80: :2 | 2001:DB8:AAAA:1: :2/64 |
| R3 interface Tunnel23 | FE80: :3 | 2001:DB8:AAAA:1: :3/64 |

On R1, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:AAAA:1::1/64
 ipv6 nhrp map multicast dynamic
 ipv6 nhrp network-id 23
 tunnel source 10.1.123.1
 tunnel mode gre multipoint

interface Ethernet0/1
 ip address 10.1.123.1 255.255.255.0
```

On R2, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:AAAA:1::2/64
 ipv6 nhrp map 2001:DB8:AAAA:1::1/64 10.1.123.1
 ipv6 nhrp map multicast 10.1.123.1
 ipv6 nhrp network-id 23
 ipv6 nhrp nhs 2001:DB8:AAAA:1::1
 ipv6 nhrp shortcut
 tunnel source 10.1.123.2
 tunnel mode gre multipoint

interface Ethernet0/0
 ip address 10.1.123.2 255.255.255.0
```

On R3, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 ipv6 address FE80::3 link-local
 ipv6 address 2001:DB8:AAAA:1::3/64
 ipv6 nhrp map 2001:DB8:AAAA:1::1/64 10.1.123.1
 ipv6 nhrp map multicast 10.1.123.1
 ipv6 nhrp network-id 23
 ipv6 nhrp nhs 2001:DB8:AAAA:1::1
 ipv6 nhrp shortcut
 tunnel source 10.1.123.3
```

```
 tunnel mode gre multipoint

interface Ethernet0/0
 ip address 10.1.123.3 255.255.255.0
```

**Task 31.2** Configure the following IPv6 address:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R1 interface E0/0 | EUI-64 format | 2001:DB8:BBBB:1: :/64 EUI-64 format |
| R2 interface E0/1 | EUI-64 format | 2001:DB8:CCCC:1: :/64 EUI-64 format |

On R1, configure the following:

```
interface Ethernet0/0
no shut
ipv6 address 2001:DB8:BBBB:1::/64 eui-64
```

On R2, configure the following:

```
interface Ethernet0/1
no shut
ipv6 address 2001:DB8:CCCC:1::/64 eui-64
```

On R1, on my Pod, the MAC address of the E0/0 is aabb.cc00.0100 and the IPv6 is obtained by using this MAC address.

MAC address is AABBCC000100, which is 48 bits long.
The FFFE is inserted between the OUI part of the MAC address and the NIC part of the MAC address to create a 64 bits long EUI address AABBCC FFEF 000100.
Next the second byte from the left that is to say AA should have its second bit from the right inserted.

In binary, AA is written 10101010. When I flip the second bit from the right, I will get 10101000, that is to say A8 in hexadecimal.

The IPv6 EUI-64 bit address will therefore be 2001:DB8:BBBB:1:A8BB:CCFF:EF00:0100.

On R1, you can verify it with the following command:

```
R1#sh ipv6 interface e0/0
Ethernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:100
  No Virtual link-local address(es):
  Global unicast address(es):
    2001:DB8:BBBB:1:A8BB:CCFF:FE00:100, subnet is 2001:DB8:BBBB:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::FB
    FF02::1:FF00:100
  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 NS retransmit interval is 1000 milliseconds
```

On R2, on my Pod, the MAC address of the E0/1 is aabb.cc00.0210 and the IPv6 is obtained by using this MAC address.

MAC address is AABBCC000210, which is 48 bits long.
The FFFE is inserting between the OUI part of the MAC address and the NIC part of the MAC address to create a 64 bits long EUI address AABBCC FFEF 000210.
Next the second byte from the left that is to say AA should have its second bit from the right inserted.

In binary, AA is written 10101010. When I flip the second bit from the right, I will get 10101000, that is to say A8 in hexadecimal.

The IPv6 EUI-64 bit address will therefore be 2001:DB8:BBBB:1:A8BB:CCFF:EF00:0210.

On R2, you can verify it with the following command:

```
R2#sh ipv6 interface e0/1
Ethernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:210
  No Virtual link-local address(es):
  Global unicast address(es):
    2001:DB8:CCCC:1:A8BB:CCFF:FE00:210, subnet is 2001:DB8:CCCC:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::FB
    FF02::1:FF00:210
  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 NS retransmit interval is 1000 milliseconds
```

**Task 31.3** Use the RIPng with the identifier of “iPexpert” to enable IP routing between the interface E0/0 of R1 and the interface E0/1 of R2.

On R1, configure the following:

```
ipv6 unicast-routing

ipv6 router rip iPexpert

interface E0/0
ipv6 rip iPexpert enable

int tu23
ipv6 rip iPexpert enable
```

On R2, configure the following:

```
ipv6 unicast-routing

ipv6 router rip iPexpert

interface E0/1
ipv6 rip iPexpert enable

int tu23
ipv6 rip iPexpert enable
```

On R1, I have received to RIP route to the destination 2001:DB8:CCCC:1::/64

```
R1#sh ipv6 route
IPv6 Routing Table - default - 6 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
```

```
         IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
         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
C   2001:DB8:AAAA:1::/64 [0/0]
     via Tunnel23, directly connected
L   2001:DB8:AAAA:1::1/128 [0/0]
     via Tunnel23, receive
C   2001:DB8:BBBB:1::/64 [0/0]
     via Ethernet0/0, directly connected
L   2001:DB8:BBBB:1:A8BB:CCFF:FE00:100/128 [0/0]
     via Ethernet0/0, receive
R   2001:DB8:CCCC:1::/64 [120/2]
     via FE80::A8BB:CCFF:FE00:210, Ethernet0/0
L   FF00::/8 [0/0]
     via Null0, receive
```

I can ping from the interface E0/0 of R1 to the interface E0/1 of R2:

```
R1#ping 2001:DB8:CCCC:1:A8BB:CCFF:FE00:210 source 2001:DB8:BBBB:1:A8BB:CCFF:FE00:100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:CCCC:1:A8BB:CCFF:FE00:210, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:BBBB:1:A8BB:CCFF:FE00:100
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 31.4** On R1, create an IP host mapping called R2LAN for the IPv6 global address of the E0/1 of R2. Check that you can ping R2LAN from R1.

On R1, configure the following:

```
ipv6 host R2LAN 2001:DB8:CCCC:1:A8BB:CCFF:FE00:210
```

Please note that the IPv6 used in this mapping is different if your MAC address is different.

```
R1#ping R2LAN
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:CCCC:1:A8BB:CCFF:FE00:210, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 31.5** On R2, create an IP host mapping called R1LAN for the IPv6 global address of the E0/0 of R1. Check that you can ping R1LAN from R2.

On R2, configure the following:

```
ipv6 host R1LAN 2001:DB8:BBBB:1:A8BB:CCFF:FE00:100
```

Please note that the IPv6 used in this mapping is different if your MAC address is different.

```
R2#ping R1LAN
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:BBBB:1:A8BB:CCFF:FE00:100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

**Task 31.6** Configure the following interfaces to automatically assigned IPv6 addresses to their interfaces:

| | |
|---|---|
| R6 | E0/1 |
| R8 | E0/1 |
| R9 | E0/1 |

On R6, R8 and R9, configure the following:

```
ipv6 unicast-routing

interface E0/1
ipv6 address autoconfig
```

When implementing SLAAC, the IPv6 client is listening for the local RAs and is taking the prefixes that are advertised to form a unique address that can be used on the network.

R6 int E0/1 has been configured with the IPv6 addresses 2001:DB8:CCCC:1:A8BB:CCFF:FE00:610.

```
R6#sh ipv6 int e0/1
Ethernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:610
  No Virtual link-local address(es):
  Stateless address autoconfig enabled
  Global unicast address(es):
    2001:DB8:CCCC:1:A8BB:CCFF:FE00:610, subnet is 2001:DB8:CCCC:1::/64 [EUI/CAL/PRE]
      valid lifetime 2591851 preferred lifetime 604651
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::FB
    FF02::1:FF00:610
  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
  Hosts use stateless autoconfig for addresses.
```

R8 int E0/1 has been configured with the IPv6 addresses 2001:DB8:BBBB:1:A8BB:CCFF:FE00:810.

```
R8#sh ipv6 int e0/1
Ethernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:810
  No Virtual link-local address(es):
  Stateless address autoconfig enabled
  Global unicast address(es):
    2001:DB8:BBBB:1:A8BB:CCFF:FE00:810, subnet is 2001:DB8:BBBB:1::/64 [EUI/CAL/PRE]
      valid lifetime 2591950 preferred lifetime 604750
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::FB
    FF02::1:FF00:810
  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
  Hosts use stateless autoconfig for addresses.
```

R9 int E0/1 has been configured with the IPv6 addresses 2001:DB8:BBBB:1:A8BB:CCFF:FE00:910 and 2001:DB8:CCCC:1:A8BB:CCFF:FE00:910.

```
R9#sh ipv6 int e0/1
Ethernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:910
  No Virtual link-local address(es):
  Stateless address autoconfig enabled
  Global unicast address(es):
    2001:DB8:CCCC:1:A8BB:CCFF:FE00:910, subnet is 2001:DB8:CCCC:1::/64 [EUI/CAL/PRE]
      valid lifetime 2591871 preferred lifetime 604671
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::FB
    FF02::1:FF00:910
  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
  Hosts use stateless autoconfig for addresses.
```

**Task 31.7** Enable RIPng with the identifier of “iPexpert” on R6, R8, and R9. Check that R8 can reach the IPv6 global address that has been previously assigned to the E0/1 of R6 and to the E0/1 of R9.

On R6, R8, and R9, configure the following:

```
ipv6 router rip iPexpert

interface E0/1
ipv6 rip iPexpert enable
```

On R8, RIP has built the following routing table. We can see that 2001:DB8:BBBB:1::/64 is present as NDp prefix, meaning a prefix discovered using RFC 2461 Neighbor Discovery dor IP Version 6.

```
R8#sh ipv6 route
IPv6 Routing Table - default - 5 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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:DB8:AAAA:1::/64 [120/2]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
NDp 2001:DB8:BBBB:1::/64 [2/0]
     via Ethernet0/1, directly connected
L   2001:DB8:BBBB:1:A8BB:CCFF:FE00:810/128 [0/0]
     via Ethernet0/1, receive
R   2001:DB8:CCCC:1::/64 [120/3]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
L   FF00::/8 [0/0]
     via Null0, receive
```

From R8, I can ping the interface on R6 with the IP address 2001:DB8:CCCC:1:A8BB:CCFF:FE00:610 and the interface on R9 with the IP address 2001:DB8:CCCC:1:A8BB:CCFF:FE00:910.

```
R8#ping 2001:DB8:CCCC:1:A8BB:CCFF:FE00:610
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:CCCC:1:A8BB:CCFF:FE00:610, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/5/21 ms

R8#ping 2001:DB8:CCCC:1:A8BB:CCFF:FE00:910
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:CCCC:1:A8BB:CCFF:FE00:910, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/19 ms
```

**Task 31.8** Configure the following IPv6 address on the connection between R6 and R7:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R6 interface E0/0 | FE80: :1 | 2001:DB8:DDDD:1: :6/64 |
| R7 interface E0/0 | FE80: :2 | 2001:DB8:DDDD:1: :7/64 |

On R6, configure the following:

```
interface Ethernet0/0
ipv6 address FE80::1 link-local
ipv6 address 2001:DB8:DDDD:1::6/64
```

On R7, configure the following:

```
interface Ethernet0/0
ipv6 address FE80::2 link-local
ipv6 address 2001:DB8:DDDD:1::7/64
```

**Task 31.9** On R7, configure the following IPv6 loopback addresses:

| | Global Unicast |
|---|---|
| R7 interface Loopback4 | 2001:DB8:EEEE:4: :7/64 |
| R7 interface Loopback5 | 2001:DB8:EEEE:5: :7/64 |
| R7 interface Loopback6 | 2001:DB8:EEEE:6: :7/64 |
| R7 interface Loopback7 | 2001:DB8:EEEE:7: :7/64 |

On the R7, configure the following:

```
interface Loopback4
ipv6 address 2001:DB8:EEEE:4::7/64

interface Loopback5
ipv6 address 2001:DB8:EEEE:5::7/64

interface Loopback6
ipv6 address 2001:DB8:EEEE:6::7/64

interface Loopback7
ipv6 address 2001:DB8:EEEE:7::7/64
```

**Task 31.10** Enable RIPng with the identifier of “iPexpert” on the connection between R6 and R7, and on the 4 loopbacks on R7.

On R6, configure the following:

```
interface Ethernet0/0
ipv6 rip iPexpert enable
```

On R7, configure the following:

```
ipv6 unicast-routing

interface Loopback4
ipv6 rip iPexpert enable
interface Loopback5
ipv6 rip iPexpert enable
interface Loopback6
ipv6 rip iPexpert enable
interface Loopback7
ipv6 rip iPexpert enable
interface Ethernet0/0
ipv6 rip iPexpert enable
Let’s have a look a the routing table of R8. Thanks to RIP, the loopbacks of R7 are
reachable.
R8#sh ipv6 route
IPv6 Routing Table - default - 10 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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:DB8:AAAA:1::/64 [120/2]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
NDp 2001:DB8:BBBB:1::/64 [2/0]
     via Ethernet0/1, directly connected
L   2001:DB8:BBBB:1:A8BB:CCFF:FE00:810/128 [0/0]
     via Ethernet0/1, receive
R   2001:DB8:CCCC:1::/64 [120/3]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
R   2001:DB8:DDDD:1::/64 [120/4]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
R   2001:DB8:EEEE:4::/64 [120/5]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
R   2001:DB8:EEEE:5::/64 [120/5]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
R   2001:DB8:EEEE:6::/64 [120/5]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
R   2001:DB8:EEEE:7::/64 [120/5]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
L   FF00::/8 [0/0]
     via Null0, receive

R8#ping 2001:DB8:EEEE:4::7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:EEEE:4::7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
```

**Task 31.11** Ensure that R6 receives from R7 a summary route encompassing all the loopbacks.

On R7, configure the following:

```
int e0/0
ipv6 rip CCIE summary-address 2001:DB8:EEEE:4::/62
```

After a while, the specific routes will disappear from the routing table of R8 and only the 2001:DB8:EEEE:4::/62 summary routes will stay in the routing table.

```
R8#sh ipv6 route
IPv6 Routing Table - default - 7 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
```

```
        B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
        H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
        IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
        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:DB8:AAAA:1::/64 [120/2]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
NDp 2001:DB8:BBBB:1::/64 [2/0]
     via Ethernet0/1, directly connected
L   2001:DB8:BBBB:1:A8BB:CCFF:FE00:810/128 [0/0]
     via Ethernet0/1, receive
R   2001:DB8:CCCC:1::/64 [120/3]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
R   2001:DB8:DDDD:1::/64 [120/4]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
R   2001:DB8:EEEE:4::/62 [120/5]
     via FE80::A8BB:CCFF:FE00:100, Ethernet0/1
L   FF00::/8 [0/0]
     via Null0, receive
```

From R8, I can still ping all the loopbacks of the router R7 by using the summary route.

```
R8#ping 2001:DB8:EEEE:7::7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:EEEE:7::7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 31.12** Enable RIPng on the tunnel interface of the router R3. Ensure that R3 is able to ping the IPv6 address of the loopback4 of R7.

We have to enable RIP on the tunnel interface of the router R3. R3 is a spoke in the DMVPN topology. R7 is located behind the router R2 which is another spoke. We have to make sure that spoke to spoke connectivity is enabled by disabling split horizon on the hub, that is to say router R1.

On R3, configure the following:

```
ipv6 unicast-routing

ipv6 router rip iPexpert

int tu23
ipv6 rip iPexpert enable
```

We are not able to ping the loopback4 of R7 because the split-horizon mechanism is blocking the updates received from R2 to be advertised to R3.

```
R3#ping 2001:DB8:EEEE:4::7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:EEEE:4::7, timeout is 2 seconds:

% No valid route for destination
Success rate is 0 percent (0/1)
```

On R1, configure the following:

```
ipv6 router rip iPexpert
no split-horizon
```

Once split-horizon is disabled on R1, I can ping from R3 the loopback4 of R7.

```
R3#ping 2001:DB8:EEEE:4::7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:EEEE:4::7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
```

**Task 31.13** Configure the following IPv6 address on the connection between R3 and R4:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R3 interface S4/3 | FE80: :1 | 2001:DB8:1111:1: :3/64 |
| R4 interface S4/0 | FE80: :2 | 2001:DB8:1111:1: :4/64 |

On R3, configure the following:

```
interface Serial4/3
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:1111:1::3/64
```

On R4, configure the following:

```
interface Serial4/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:1111:1::4/64
```

**Task 31.14** Configure the following IPv6 address on the connection between R3 and R5:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R3 interface S4/0 | FE80: :1 | 2001:DB8:2222:1: :3/64 |
| R5 interface S4/0 | FE80: :2 | 2001:DB8:2222:1: :5/64 |

On R3, configure the following:

```
interface Serial4/0
ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:2222:1::3/64
```

On R5, configure the following:

```
interface Serial4/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:2222:1::5/64
```

**Task 31.15** Configure the following IPv6 address on the connection between R4 and R5:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R4 interface E0/0 | FE80: :1 | 2001:DB8:FFFF:1: :4/64 |
| R5 interface E0/0 | FE80: :2 | 2001:DB8:FFFF:1: :5/64 |

On R4, configure the following:

```
interface Ethernet0/0
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:FFFF:1::4/64
```

On R5, configure the following:

```
interface Ethernet0/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:FFFF:1::5/64
```

**Task 31.16** Enable RIPng with the identifier of 345 on the connection between R3 and R4, on the connection between R3 and R5, and on the connection between R4 and R5.

On R3, configure the following:

```
ipv6 unicast-routing
```

```
ipv6 router rip 345

interface S4/0
ipv6 rip 345 enable

interface S4/3
ipv6 rip 345 enable
```

On R4, configure the following:

```
ipv6 unicast-routing

ipv6 router rip 345

interface S4/0
ipv6 rip 345 enable

interface E0/0
ipv6 rip 345 enable

On R5, configure the following:
ipv6 unicast-routing

ipv6 router rip 345

interface S4/0
ipv6 rip 345 enable

interface E0/0
ipv6 rip 345 enable
```

**Task 31.17** Enable full IPv6 connectivity between the 2 RIPng domains, iPexpert and 345.

In order to enable full reachability between the 2 RIPng domains, we have to configure mutual redistribution.

On R3, configure the following:

```
ipv6 router rip iPexpert
 redistribute connected
 redistribute rip 345 metric 1
ipv6 router rip 345
 redistribute connected
 redistribute rip iPexpert metric 1
```

Unlike IPv4 redistribution, IPv6 requires redistribute connected command.
End

Let's try to ping from R5 to the loopback 7 of R7:

```
R5#ping 2001:DB8:EEEE:7::7
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:EEEE:7::7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/16/41 ms
```

**Task 31.18** Ensure that R4 and R5 have a default route pointing towards R3. You have to configure R3 only to complete this task and you are not allowed to configure static routes.

On R3, configure the following:

```
int s4/0
ipv6 rip 345 default-information originate
int s4/3
ipv6 rip 345 default-information originate
```

On R4 and R5 routing table, there is a default route pointing to R3:

```
R4#sh ipv6 route
IPv6 Routing Table - default - 12 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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   ::/0 [120/2]
     via FE80::1, Serial4/0
C   2001:DB8:1111:1::/64 [0/0]
     via Serial4/0, directly connected
L   2001:DB8:1111:1::4/128 [0/0]
     via Serial4/0, receive
R   2001:DB8:2222:1::/64 [120/2]
     via FE80::1, Serial4/0
     via FE80::2, Ethernet0/0
R   2001:DB8:AAAA:1::/64 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:BBBB:1::/64 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:CCCC:1::/64 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:DDDD:1::/64 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:EEEE:4::/62 [120/2]
     via FE80::1, Serial4/0
C   2001:DB8:FFFF:1::/64 [0/0]
     via Ethernet0/0, directly connected
L   2001:DB8:FFFF:1::4/128 [0/0]
     via Ethernet0/0, receive
L   FF00::/8 [0/0]
     via Null0, receive

R5#sh ipv6 route
IPv6 Routing Table - default - 12 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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   ::/0 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:1111:1::/64 [120/2]
     via FE80::1, Serial4/0
     via FE80::1, Ethernet0/0
C   2001:DB8:2222:1::/64 [0/0]
     via Serial4/0, directly connected
L   2001:DB8:2222:1::5/128 [0/0]
     via Serial4/0, receive
R   2001:DB8:AAAA:1::/64 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:BBBB:1::/64 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:CCCC:1::/64 [120/2]
     via FE80::1, Serial4/0
```

```
R   2001:DB8:DDDD:1::/64 [120/2]
     via FE80::1, Serial4/0
R   2001:DB8:EEEE:4::/62 [120/2]
     via FE80::1, Serial4/0
C   2001:DB8:FFFF:1::/64 [0/0]
     via Ethernet0/0, directly connected
L   2001:DB8:FFFF:1::5/128 [0/0]
     via Ethernet0/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

**Task 31.19** The default route and the summarized route for the loopbacks of R7 should be the 2 only RIP process iPexpert entries in the IPv6 routing table of R4 and R5. Configure R3 to achieve this task. Use an IPv6 prefix-list called “SUMMARYR7”.

On R3, configure the following:

```
ipv6 prefix-list SUMMARYR7 seq 5 permit ::/0
ipv6 prefix-list SUMMARYR7 seq 10 permit 2001:DB8:EEEE:4::/62

ipv6 router rip 345
 distribute-list prefix-list SUMMARYR7 out
```

In order to speed up the timeout of the filtered routes, you can issue the command “clear ipv6 rip” on R4 and R5.

Let’s have a look at the routing table from R4:

```
R4#sh ipv6 route
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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   ::/0 [120/2]
     via FE80::1, Serial4/0
C   2001:DB8:1111:1::/64 [0/0]
     via Serial4/0, directly connected
L   2001:DB8:1111:1::4/128 [0/0]
     via Serial4/0, receive
R   2001:DB8:2222:1::/64 [120/2]
     via FE80::2, Ethernet0/0
R   2001:DB8:EEEE:4::/62 [120/2]
     via FE80::1, Serial4/0
C   2001:DB8:FFFF:1::/64 [0/0]
     via Ethernet0/0, directly connected
L   2001:DB8:FFFF:1::4/128 [0/0]
     via Ethernet0/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

The only RIP routes learned from process iPexpert are the default route and the network of the loopbacks of R7. The network 2001:DB8:2222:1::/64 is also learned by RIP but by process 345.

**Task 31.20** The clients on the VLAN 2001:DB8:FFFF:1: :/64 should always be routed over the connection R5-R3. The connection R4-R3 should only be used in case the connection R5-R3 is going down.

We can see that from R3, there are 2 next-hops to the network 2001:DB8:FFFF:1: :/64 and therefore load-balancing is taking place.

```
R3#sh ipv6 route 2001:DB8:FFFF:1::/64
Routing entry for 2001:DB8:FFFF:1::/64
  Known via "rip 345", distance 120, metric 2
  Redistributing via rip iPexpert
  Backup from "rip iPexpert [120]"
  Route count is 2/2, share count 0
  Routing paths:
    FE80::2, Serial4/3
      Last updated 00:31:38 ago
    FE80::2, Serial4/0
      Last updated 00:31:26 ago
```

On R3, configure the following:

```
int s4/3
ipv6 rip 345 metric-offset 3
```

Don't forget to perform the "clear ipv6 rip". Let's verify that the routing table is now pointing to R5 only as a next-hop.

```
R3#sh ipv6 route 2001:DB8:FFFF:1::/64
Routing entry for 2001:DB8:FFFF:1::/64
  Known via "rip 345", distance 120, metric 2
  Redistributing via rip iPexpert
  Backup from "rip iPexpert [120]"
  Route count is 1/1, share count 0
  Routing paths:
    FE80::2, Serial4/0
      Last updated 00:00:40 ago
```

**You have completed Lab 31**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 32: Configure and troubleshoot IP version 6 (Part 2)

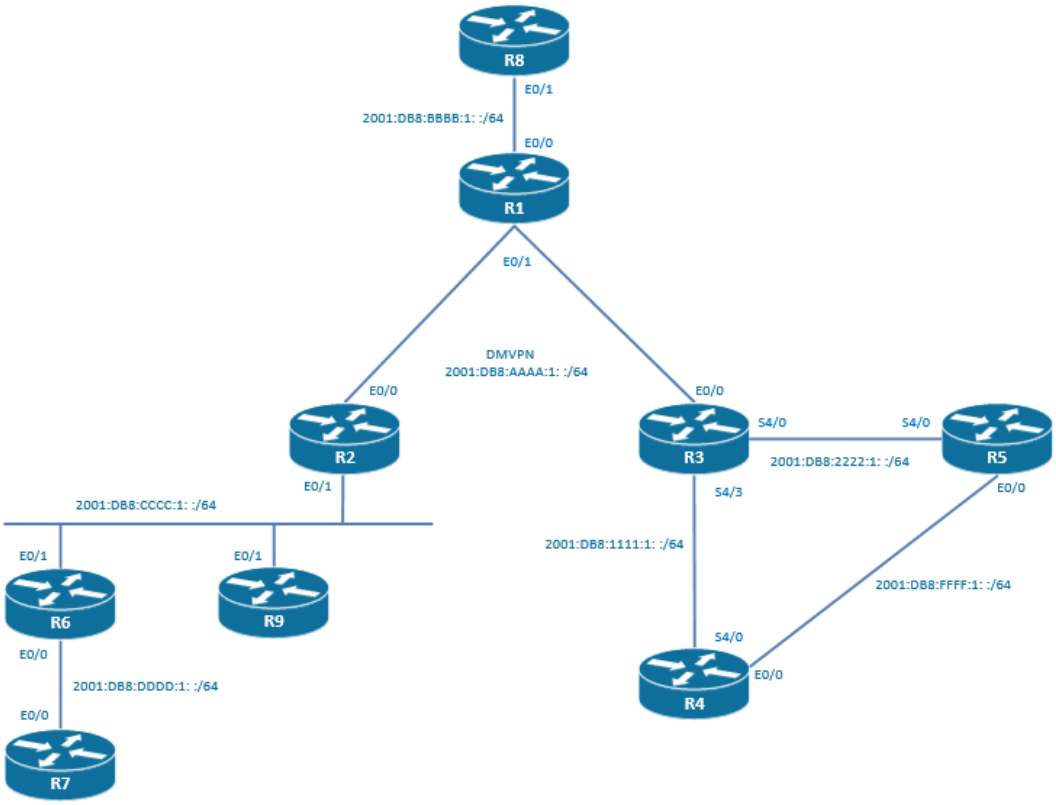
## Technologies covered

- EIGRPv6
- EIGRPv6 summarization
- EIGRPv6 default route
- EIGRPv6 authentication
- EIGRPv6 unequal load balancing

## Overview

You have been tasked to configure the IPv6 routing in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 32.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. Configure the DMVPN phase 3 tunnel infrastructure for IPv6. Do not implement encryption. Use the following addresses:

| | | |
|---|---|---|
| R1 | E0/1 | 10.1.123.1/24 |
| R2 | E0/0 | 10.1.123.2/24 |
| R3 | E0/0 | 10.1.123.3/24 |

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R1 interface Tunnel23 | FE80: :1 | 2001:DB8:AAAA:1: :1/64 |
| R2 interface Tunnel23 | FE80: :2 | 2001:DB8:AAAA:1: :2/64 |
| R3 interface Tunnel23 | FE80: :3 | 2001:DB8:AAAA:1: :3/64 |

On R1, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:AAAA:1::1/64
 ipv6 nhrp map multicast dynamic
 ipv6 nhrp network-id 123
 tunnel source 10.1.123.1
 tunnel mode gre multipoint

interface Ethernet0/1
 ip address 10.1.123.1 255.255.255.0
```

On R2, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:AAAA:1::2/64
 ipv6 nhrp map 2001:DB8:AAAA:1::1/64 10.1.123.1
 ipv6 nhrp map multicast 10.1.123.1
 ipv6 nhrp network-id 123
 ipv6 nhrp nhs 2001:DB8:AAAA:1::1
 ipv6 nhrp shortcut
 tunnel source 10.1.123.2
 tunnel mode gre multipoint

interface Ethernet0/0
 ip address 10.1.123.2 255.255.255.0
```

On R3, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 ipv6 address FE80::3 link-local
 ipv6 address 2001:DB8:AAAA:1::3/64
 ipv6 nhrp map 2001:DB8:AAAA:1::1/64 10.1.123.1
 ipv6 nhrp map multicast 10.1.123.1
 ipv6 nhrp network-id 123
 ipv6 nhrp nhs 2001:DB8:AAAA:1::1
 ipv6 nhrp shortcut
 tunnel source 10.1.123.3
 tunnel mode gre multipoint

interface Ethernet0/0
 ip address 10.1.123.3 255.255.255.0
```

**Task 32.2** Configure an IPv6 NHRP authentication of iPexpert and a NHRP network-id of 123.

On R1, R2, and R3, configure the following:

```
int tunnel 23
ip nhrp authentication iPexpert
```

**Task 32.3** Configure the following loopback IPv6 addresses:

| | Global Unicast |
|---|---|
| R1 interface lo0 | 2001:DB8:A:A: :1/128 |
| R2 interface lo0 | 2001:DB8:A:A: :2/128 |
| R3 interface lo0 | 2001:DB8:A:A: :3/128 |

On R1, configure the following:

```
interface loopback0
ipv6 address 2001:DB8:A:A::1/128
```

On R2, configure the following:

```
interface loopback0
ipv6 address 2001:DB8:A:A::2/128
```

On R3, configure the following:

```
interface loopback0
ipv6 address 2001:DB8:A:A::3/128
```

**Task 32.4** Enable EIGRPv6 with an AS of 123 on the DMVPN network between R1, R2, and R3.

On the spokes R2 and R3, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 123
 no shutdown

interface tu23
 ipv6 eigrp 123
```

On the hub R1, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 123
 no shutdown

interface tu23
 ipv6 eigrp 123
 no ipv6 split-horizon eigrp 123
```

**Task 32.5** Make sure that there is IPv6 connectivity between the loopbacks of R1, R2, and R3.

On R1, R2 and R3, configure the following:

```
interface loopback0
 ipv6 eigrp 123
```

I have full reachability between the loopbacks 0 of R1, R2, and R3:

```
R3#ping 2001:DB8:A:A::2 source 2001:DB8:A:A::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::2, timeout is 2 seconds:
```

```
Packet sent with a source address of 2001:DB8:A:A::3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms

R3#ping 2001:DB8:A:A::1 source 2001:DB8:A:A::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:A:A::3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/5 ms
```

**Task 32.6** Configure EIGRPv6 with an AS of 123 on the LAN 2001:DB8:CCCC:1: :/64. Check that you can ping the loopback0 of R3 from R6 and R9.

On R2, configure the following:

```
interface e0/1
 ipv6 eigrp 123
```

On R6 and R9, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 123
 no shutdown

interface e0/1
 ipv6 eigrp 123
```

From R6, I can ping the loopback0 of R3:

```
R6#ping 2001:DB8:A:A::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/11 ms
```

From R9, I can ping the loopback0 of R3:

```
R9#ping 2001:DB8:A:A::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/17 ms
```

**Task 32.7** In the routing table of R6 and R9, there should be no specific entries for the loopbacks of R1, R2, and R3. There should only be a routing entry to reach the summary route 2001:DB8:A:A: :/126. Check that you can ping the loopback0 of R3 from R6 and R9.

On R2, configure the following:

```
int e0/1
ipv6 summary-address eigrp 123 2001:DB8:A:A::/126
```

On R6, I can check the routing table and notice that there is a summary route to reach the loopback0 of R1, R2, and R3.

```
R6#sh ipv6 route
IPv6 Routing Table - default - 7 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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:DB8:A:A::/126 [90/409600]
     via FE80::A8BB:CCFF:FE00:210, Ethernet0/1
D   2001:DB8:AAAA:1::/64 [90/26905600]
     via FE80::A8BB:CCFF:FE00:210, Ethernet0/1
C   2001:DB8:CCCC:1::/64 [0/0]
     via Ethernet0/1, directly connected
L   2001:DB8:CCCC:1::6/128 [0/0]
     via Ethernet0/1, receive
C   2001:DB8:DDDD:1::/64 [0/0]
     via Ethernet0/0, directly connected
L   2001:DB8:DDDD:1::6/128 [0/0]
     via Ethernet0/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

Let's check that I can still ping from R6 the loopback0 of R3 using this summary route.

```
R6#ping 2001:DB8:A:A::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

Let's check that I can still ping from R9 the loopback0 of R3 using this summary route.

```
R9#ping 2001:DB8:A:A::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 32.8** On R2, create an IPv6 static default route pointing to Null0 and make sure that R2 will be the default router for all packets with an unknown IPv6 addresses in the EIGRP domain AS 123.

On R2, configure the following:

```
ipv6 route ::/0 null0

ipv6 router eigrp 123
redistribute static
```

This default route is now present in the routing table of all the routers configured as part of EIGRP domain iPexpert. For example, we can find it in the R3 routing table:

```
R3#sh ipv6 route
IPv6 Routing Table - default - 12 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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  ::/0 [170/28160000]
     via FE80::1, Tunnel23
D   2001:DB8:A:A::1/128 [90/27008000]
     via FE80::1, Tunnel23
D   2001:DB8:A:A::2/128 [90/28288000]
     via FE80::1, Tunnel23
LC  2001:DB8:A:A::3/128 [0/0]
     via Loopback0, receive
C   2001:DB8:1111:1::/64 [0/0]
     via Serial4/3, directly connected
L   2001:DB8:1111:1::3/128 [0/0]
     via Serial4/3, receive
```

```
C   2001:DB8:2222:1::/64 [0/0]
     via Serial4/0, directly connected
L   2001:DB8:2222:1::3/128 [0/0]
     via Serial4/0, receive
C   2001:DB8:AAAA:1::/64 [0/0]
     via Tunnel23, directly connected
L   2001:DB8:AAAA:1::3/128 [0/0]
     via Tunnel23, receive
D   2001:DB8:CCCC:1::/64 [90/28185600]
     via FE80::1, Tunnel23
L   FF00::/8 [0/0]
     via Null0, receive
```

**Task 32.9** Configure EIGRPv6 with an AS of 123 on the LAN 2001:DB8:BBBB:1: :/64.

On R1, configure the following:

```
interface e0/0
 ipv6 eigrp 123
```

On R8, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 123
 no shutdown

interface e0/1
 ipv6 eigrp 123
```

**Task 32.10** The router R1 should not advertise any specific networks to R8. Only a default route should be advertised. Use the “ipv6 summary-address eigrp” on R1 to resolve this task. Check that you can ping the loopback0 of R3 and the loopback0 of R2 from R8.

On R1, configure the following:

```
int e0/0
ipv6 summary-address eigrp 1 ::/0
```

**Task 32.11** Configure EIGRPv6 authentication between R1 and R8. Use a key chain called “iPexpertchain”, a key number of 2, and a key-string of “iPexpert”.

On R1, configure the following:

```
key chain iPexpertchain
 key 2
   key-string iPexpert

interface Ethernet0/0
 ipv6 authentication mode eigrp 123 md5
 ipv6 authentication key-chain eigrp 123 iPexpertchain
```

On R8, configure the following:

```
key chain iPexpertchain
 key 2
   key-string iPexpert

interface Ethernet0/1
 ipv6 authentication mode eigrp 123 md5
 ipv6 authentication key-chain eigrp 123 iPexpertchain
```

**Task 32.12** Configure the following IPv6 address on the connection between R3 and R4:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R3 interface S4/3 | FE80: :1 | 2001:DB8:1111:1: :3/64 |
| R4 interface S4/0 | FE80: :2 | 2001:DB8:1111:1: :4/64 |

On R3, configure the following:

```
interface Serial4/3
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:1111:1::3/64
```

On R4, configure the following:

```
interface Serial4/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:1111:1::4/64
```

**Task 32.13** Configure the following IPv6 address on the connection between R3 and R5:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R3 interface S4/0 | FE80: :1 | 2001:DB8:2222:1: :3/64 |
| R5 interface S4/0 | FE80: :2 | 2001:DB8:2222:1: :5/64 |

On R3, configure the following:

```
interface Serial4/0
ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:2222:1::3/64
```

On R5, configure the following:

```
interface Serial4/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:2222:1::5/64
```

**Task 32.14** Configure the following IPv6 address on the connection between R4 and R5:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R4 interface E0/0 | FE80: :1 | 2001:DB8:FFFF:1: :4/64 |
| R5 interface E0/0 | FE80: :2 | 2001:DB8:FFFF:1: :5/64 |

On R4, configure the following:

```
interface Ethernet0/0
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:FFFF:1::4/64
```

On R5, configure the following:

```
interface Ethernet0/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:FFFF:1::5/64
```

**Task 32.15** Configure EIGRPv6 with an AS of 345 on the connection between R3 and R4, between R3 and R5, and between R4 and R5.

On R3, configure the following:

```
ipv6 router eigrp 345
 no shutdown
```

```
interface S4/0
ipv6 eigrp 345

interface S4/3
ipv6 eigrp 345
```

On R4, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 345
 no shutdown

interface S4/0
ipv6 eigrp 345

interface E0/0
ipv6 eigrp 345
```

On R5, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 345
 no shutdown

interface S4/0
ipv6 eigrp 345

interface E0/0
ipv6 eigrp 345
```

**Task 32.16** Configure the following loopback IPv6 addresses:

| | Global Unicast |
|---|---|
| R4 interface lo0 | 2001:DB8:A:A: :4/128 |
| R5 interface lo0 | 2001:DB8:A:A: :5/128 |

On R4, configure the following:

```
interface lo0
  ipv6 address 2001:DB8:A:A::4/128
```

On R5, configure the following:

```
interface lo0
  ipv6 address 2001:DB8:A:A::5/128
```

**Task 32.17** Make sure that there is IPv6 connectivity between the loopbacks of R2 and R4.

On R4, configure the following:

```
interface lo0
ipv6 eigrp 345
```

On R5, configure the following:

```
interface lo0
ipv6 eigrp 345
```

On R3, configure the following:

```
ipv6 router eigrp 345
 redistribute eigrp 123 metric 1 1 1 1 1 include-connected

ipv6 router eigrp 123
redistribute eigrp 345 metric 1 1 1 1 1 include-connected
```

Once the redistribution between the 2 EIGRP domains has taken place, I can ping the loopback0 of R2 from the loopback0 of R4.

```
R4#ping 2001:DB8:A:A::2 source 2001:DB8:A:A::4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::2, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:A:A::4
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms
```

**Task 32.18** In the routing table of R3, the routing entry towards the loopback of R5 should contain 2 next-hops, one next-hop being R4 and the other being R5 directly. The cost of the direct path should not be made equal to the cost of the indirect path (via R3). Use the variance command.

Let's analyze the current situation on R3. The route to the loopback0 of R5 is routed directly to R5, which is the shortest path. This makes perfect sense.

```
R3#sh ipv6 route 2001:DB8:A:A::5
Routing entry for 2001:DB8:A:A::5/128
  Known via "eigrp 345", distance 90, metric 2297856, type internal
  Redistributing via eigrp 123
  Route count is 1/1, share count 0
  Routing paths:
    FE80::2, Serial4/0
      Last updated 00:34:37 ago
```

We would like to enable unequal EIGRPv6 load balancing and use the path via R4 at the same time that the direct path to R5.

Let's have a look in the EIGRPv6 topology if this alternative path is a feasible successor:

```
R3#sh ipv6 eigrp topology 2001:DB8:A:A::5/128
EIGRP-IPv6 Topology Entry for AS(123)/ID(172.16.3.3) for 2001:DB8:A:A::5/128
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2560000256
  Descriptor Blocks:
  FE80::2, from Redistributed, Send flag is 0x0
      Composite metric is (2560000256/0), route is External
      Vector metric:
        Minimum bandwidth is 1 Kbit
        Total delay is 10 microseconds
        Reliability is 1/255
        Load is 1/255
        Minimum MTU is 1
        Hop count is 0
      External data:
        Originating router is 172.16.3.3 (this system)
        AS number of route is 345
        External protocol is EIGRP, external metric is 2297856
        Administrator tag is 0 (0x00000000)
EIGRP-IPv6 Topology Entry for AS(345)/ID(172.16.3.3) for 2001:DB8:A:A::5/128
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2297856
  Descriptor Blocks:
  FE80::2 (Serial4/3), from FE80::2, Send flag is 0x0
      Composite metric is (2323456/409600), route is Internal
      Vector metric:
        Minimum bandwidth is 1544 Kbit
        Total delay is 26000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 2
        Originating router is 172.16.5.5
  FE80::2 (Serial4/0), from FE80::2, Send flag is 0x0
```

```
Composite metric is (2297856/128256), route is Internal
Vector metric:
  Minimum bandwidth is 1544 Kbit
  Total delay is 25000 microseconds
  Reliability is 255/255
  Load is 1/255
  Minimum MTU is 1500
  Hop count is 1
  Originating router is 172.16.5.5
```

The metric of the direct route is 2297856.

The metric of the route via R4 is 2323456.

When configuring a variance of 2, all the feasible successor with a metric between 2297856 and 2*2297856=4595712 will be used in the routing table. This will be the case for the feasible successor with a metric of 2323456, that is to say the route via R4.

On R3, configure the following:

```
ipv6 router eigrp 345
 variance 2
```

The routing table of R3 is now containing 2 next-hops for the destination 2001:DB8:A:A::5/128.

```
R3#sh ipv6 route 2001:DB8:A:A::5
Routing entry for 2001:DB8:A:A::5/128
  Known via "eigrp 345", distance 90, metric 2297856, type internal
  Redistributing via eigrp 123
  Route count is 2/2, share count 0
  Routing paths:
    FE80::2, Serial4/0
      Last updated 00:58:54 ago
    FE80::2, Serial4/3
     Last updated 00:07:38 ago
```

**You have completed Lab 32**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 33: Configure and troubleshoot IP version 6 (Part 3)

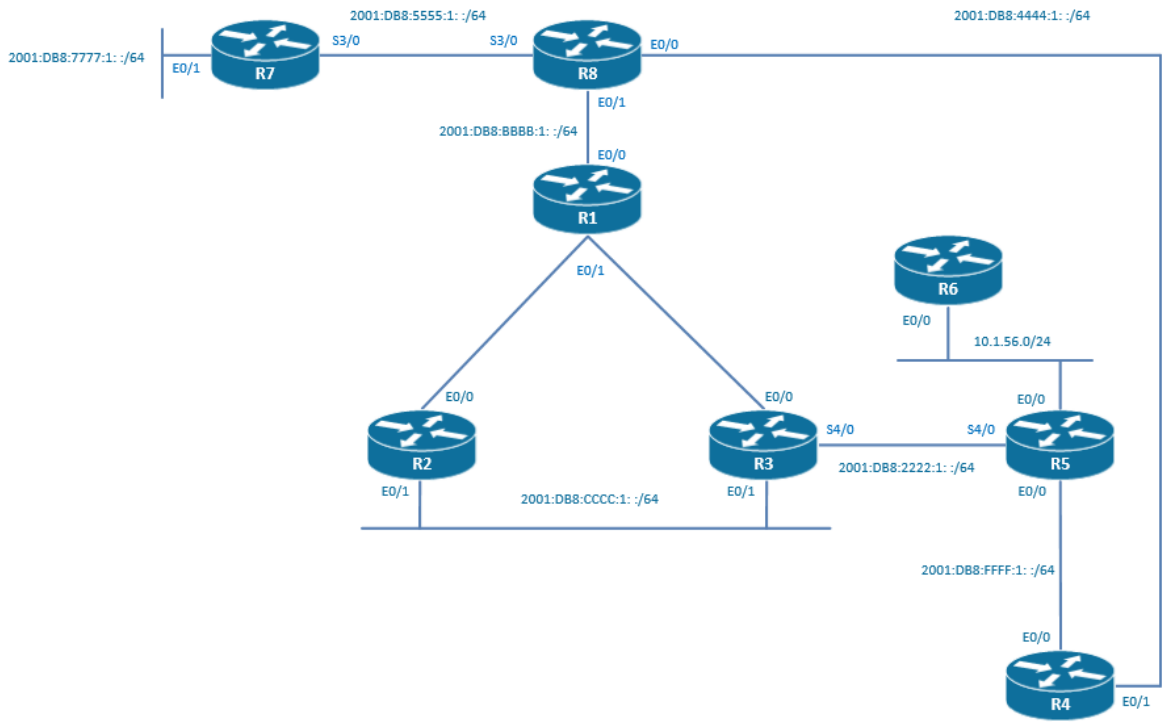
## Technologies covered

- OSPPFv3
- OSPFv3 traffic engineering
- OSFPv3 virtual link
- OSPFv3 summarization
- IPv6 NAT-PT
- Protocol redistribution

## Overview

You have been tasked to configure the IPv6 routing in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 4 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 34.1** R1, R2, and R3 are in a hub and spoke topology where R1 is the hub and R2 and R3 are the spokes. Configure the DMVPN phase 1 tunnel infrastructure for IPv6. Do not implement encryption. Use the following addresses:

| | | |
|---|---|---|
| R1 | E0/1 | 10.1.123.1/24 |
| R2 | E0/0 | 10.1.123.2/24 |
| R3 | E0/0 | 10.1.123.3/24 |
| R1 | lo10 | 1.1.1.1/32 |
| R2 | lo10 | 2.2.2.2/32 |
| R3 | lo10 | 3.3.3.3/32 |
| R4 | lo10 | 4.4.4.4/32 |
| R5 | lo10 | 5.5.5.5/32 |

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R1 interface Tunnel23 | FE80: :1 | 2001:DB8:AAAA:1: :1/64 |
| R2 interface Tunnel23 | FE80: :2 | 2001:DB8:AAAA:1: :2/64 |
| R3 interface Tunnel23 | FE80: :3 | 2001:DB8:AAAA:1: :3/64 |

On R1, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:AAAA:1::1/64
 ipv6 nhrp map multicast dynamic
 ipv6 nhrp network-id 123
 ipv6 nhrp redirect
 ipv6 nhrp shortcut
 tunnel source 10.1.123.1
 tunnel mode gre multipoint

interface Ethernet0/1
 ip address 10.1.123.1 255.255.255.0
 no shut

interface lo10
ip address 1.1.1.1 255.255.255.255
```

On R2, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 no ipv6 redirects
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:AAAA:1::2/64
 ipv6 nhrp map 2001:DB8:AAAA:1::1/64 10.1.123.1
 ipv6 nhrp map multicast 10.1.123.1
 ipv6 nhrp network-id 123
 ipv6 nhrp nhs 2001:DB8:AAAA:1::1
 ipv6 nhrp shortcut
 tunnel source 10.1.123.2
 tunnel mode gre multipoint

interface Ethernet0/0
 ip address 10.1.123.2 255.255.255.0
 no shut

interface lo10
ip address 2.2.2.2 255.255.255.255
```

On R3, configure the following:

```
interface Tunnel23
 no ip address
 no ip redirects
 no ipv6 redirects
 ipv6 address FE80::3 link-local
 ipv6 address 2001:DB8:AAAA:1::3/64
 ipv6 nhrp map 2001:DB8:AAAA:1::1/64 10.1.123.1
 ipv6 nhrp map multicast 10.1.123.1
 ipv6 nhrp network-id 123
 ipv6 nhrp nhs 2001:DB8:AAAA:1::1
 ipv6 nhrp shortcut
 tunnel source 10.1.123.3
 tunnel mode gre multipoint

interface Ethernet0/0
 ip address 10.1.123.3 255.255.255.0
 no shut

interface lo10
ip address 3.3.3.3 255.255.255.255
```

**Task 34.2** Configure an IPv6 NHRP authentication of “iPexpert” and a NHRP network-id of 123.

On R1, R2 and R3, configure the following:

```
int tunnel 23
ipv6 nhrp authentication iPexpert
```

**Task 33.3** Configure the following loopback IPv6 addresses:

| | Global Unicast |
|---|---|
| R1 interface lo0 | 2001:DB8:A:A: :1/128 |
| R2 interface lo0 | 2001:DB8:A:A: :2/128 |
| R3 interface lo0 | 2001:DB8:A:A: :3/128 |

On R1, configure the following:

```
interface loopback0
ipv6 address 2001:DB8:A:A::1/128
```

On R2, configure the following:

```
interface loopback0
ipv6 address 2001:DB8:A:A::2/128
```

On R3, configure the following:

```
interface loopback0
ipv6 address 2001:DB8:A:A::3/128
```

**Task 33.4** Enable OSPFv3 process 99 in area 0 on the DMVPN network between R1, R2, and R3. DR election should not be taking place. On R1, R2, and R3 use the loopback10 IPv4 IP address as the OSPF router-ID.

On R1, configure the following:

```
ipv6 unicast-routing

ipv6 router ospf 99
 router-id 1.1.1.1

interface tu23
 ipv6 ospf 99 area 0
 ipv6 ospf network point-to-multipoint
```

On R2, configure the following:

```
ipv6 unicast-routing

ipv6 router ospf 99
 router-id 2.2.2.2

interface tu23
ipv6 ospf 99 area 0
ipv6 ospf network point-to-multipoint
```

On R3, configure the following:

```
ipv6 unicast-routing

ipv6 router ospf 99
 router-id 3.3.3.3

interface tu23
ipv6 ospf 99 area 0
ipv6 ospf network point-to-multipoint
```

**Task 33.5** Make sure that there is IPv6 connectivity between the loopbacks of R1,R2, and R3.

On R1, configure the following:

```
interface loopback0
ipv6 ospf 99 area 0
```

On R2, configure the following:

```
interface loopback0
ipv6 ospf 99 area 0
```

On R3, configure the following:

```
interface loopback0
ipv6 ospf 99 area 0
```

On R3, I can ping the loopback0 of R1 and R2.

```
R3#ping 2001:DB8:A:A::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms

R3#ping 2001:DB8:A:A::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

**Task 33.6** Configure the following IPv6 address on the connection between R3 and R2:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R2 interface E0/1 | FE80: :1 | 2001:DB8:CCCC:1: :2/64 |
| R3 interface E0/1 | FE80: :2 | 2001:DB8:CCCC:1: :3/64 |

On R2, configure the following:

```
interface E0/1
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:CCCC:1::2/64
no shut
```

On R3, configure the following:

```
interface E0/1
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:CCCC:1::3/64
no shut
```

**Task 33.7** Enable OSPFv3 process 99 in area 0 on the network 2001:DB8:CCCC:1: :/64.

On R2 and R3, configure the following:

```
interface E0/1
ipv6 ospf 99 area 0
```

**Task 33.8** R1 should always route via R2 to reach network 2001:DB8:CCCC:1: :/64. Only in case of a failure of the connectivity between R1 and R2, should the path via R3 be chosen. You have to configure R1 to achieve this task.

Let's have a look at the routing table on R1:

```
R1#sh ipv6 route
IPv6 Routing Table - default - 9 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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
LC  2001:DB8:A:A::1/128 [0/0]
     via Loopback0, receive
O   2001:DB8:A:A::2/128 [110/1000]
     via FE80::2, Tunnel23
O   2001:DB8:A:A::3/128 [110/1000]
     via FE80::3, Tunnel23
C   2001:DB8:AAAA:1::/64 [0/0]
     via Tunnel23, directly connected
L   2001:DB8:AAAA:1::1/128 [0/0]
     via Tunnel23, receive
O   2001:DB8:AAAA:1::2/128 [110/1000]
     via FE80::2, Tunnel23
O   2001:DB8:AAAA:1::3/128 [110/1000]
     via FE80::3, Tunnel23
O   2001:DB8:CCCC:1::/64 [110/1010]
     via FE80::3, Tunnel23
     via FE80::2, Tunnel23
L   FF00::/8 [0/0]
     via Null0, receive
```

The metric of those two routes is 1010.

```
R1#sh ipv6 route 2001:DB8:CCCC:1::/64
Routing entry for 2001:DB8:CCCC:1::/64
  Known via "ospf 99", distance 110, metric 1010, type intra area
  Route count is 2/2, share count 0
  Routing paths:
    FE80::3, Tunnel23
      Last updated 00:01:46 ago
    FE80::2, Tunnel23
      Last updated 00:00:12 ago
```

On R1, configure the following:

```
interface tu23
  ipv6 ospf neighbor FE80::3 cost 2000
```

After that a cost of 2000 has been configured to the neighbor R3, the path to R2 is the preferred one.

```
R1#sh ipv6  route
IPv6 Routing Table - default - 9 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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
LC  2001:DB8:A:A::1/128 [0/0]
     via Loopback0, receive
O   2001:DB8:A:A::2/128 [110/1000]
     via FE80::2, Tunnel23
O   2001:DB8:A:A::3/128 [110/1010]
     via FE80::2, Tunnel23
C   2001:DB8:AAAA:1::/64 [0/0]
     via Tunnel23, directly connected
L   2001:DB8:AAAA:1::1/128 [0/0]
     via Tunnel23, receive
O   2001:DB8:AAAA:1::2/128 [110/1000]
     via FE80::2, Tunnel23
O   2001:DB8:AAAA:1::3/128 [110/1010]
     via FE80::2, Tunnel23
O   2001:DB8:CCCC:1::/64 [110/1010]
     via FE80::2, Tunnel23
L   FF00::/8 [0/0]
     via Null0, receive
```

**Task 33.9** Configure the following IPv6 address on the connection between R3 and R5:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R3 interface S4/0 | FE80: :1 | 2001:DB8:2222:1: :3/64 |
| R5 interface S4/0 | FE80: :2 | 2001:DB8:2222:1: :5/64 |

On R3, configure the following:

```
interface S4/0
ipv6 address 2001:DB8:2222:1::3/64
no shut
```

On R5, configure the following:

```
interface S4/0
ipv6 address 2001:DB8:2222:1::5/64
no shut
```

**Task 33.10** Configure the following IPv6 address on the connection between R5 and R4:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R5 interface E0/1 | FE80: :1 | 2001:DB8:FFFF:1: :5/64 |
| R4 interface E0/0 | FE80: :2 | 2001:DB8:FFFF:1: :4/64 |

On R5, configure the following:

```
interface E0/1
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:FFFF:1::5/64
no shut
```

On R4, configure the following:

```
interface E0/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:FFFF:1::4/64
no shut
```

**Task 33.11** Configure the following loopback IPv6 addresses:

| | Global Unicast |
|---|---|
| R5 interface lo0 | 2001:DB8:A:A: :5/128 |
| R4 interface lo0 | 2001:DB8:A:A: :4/128 |

On R5, configure the following:

```
Interface lo0
ipv6 address 2001:DB8:A:A::5/64
```

On R4, configure the following:

```
interface lo0
ipv6 address 2001:DB8:A:A::4/64
```

**Task 33.12** Enable OSPFv3 process 99 in area 55 on the network 2001:DB8:2222:1: :/64.

On R3, configure the following:

```
interface S4/0
ipv6 ospf 99 area 55
```

On R5, configure the following:

```
ipv6 unicast-routing

int lo10
ip address 5.5.5.5 255.255.255.255

ipv6 router ospf 99
 router-id 5.5.5.5

interface s4/0
ipv6 ospf 99 area 55
```

**Task 33.13** Enable OSPFv3 process 99 in area 44 on the network 2001:DB8:FFFF:1: :/64.

On R5, configure the following:

```
interface E0/1
ipv6 ospf 99 area 44
```

On R4, configure the following:

```
ipv6 unicast-routing

int lo10
ip address 4.4.4.4 255.255.255.255

ipv6 router ospf 99
 router-id 4.4.4.4

interface e0/0
ipv6 ospf 99 area 44
```

**Task 33.14** Make sure that there is IPv6 connectivity between the loopbacks of R1, R2, R3, R4, and R5.

On R3, configure the following:

```
ipv6 router ospf 99
  area 55 virtual-link 5.5.5.5
```

On R5, configure the following:

```
ipv6 router ospf 99
  area 55 virtual-link 3.3.3.3
```

**Task 33.15** Configure the following IPv6 address on the connection between R1 and R8:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R1 interface E0/0 | FE80: :1 | 2001:DB8:BBBB:1: :1/64 |
| R8 interface E0/1 | FE80: :2 | 2001:DB8:BBBB:1: :8/64 |

On R1, configure the following:

```
interface e0/0
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:BBBB:1::1/64
no shut
```

On R8, configure the following:

```
interface e0/1
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:BBBB:1::8/64
no shut
```

**Task 33.16** Enable OSPFv3 area 88 on the connection between R1 and R8.

On R1, configure the following:

```
interface e0/0
ipv6 ospf 99 area 88
```

On R8, configure the following:

```
ipv6 unicast-routing

ipv6 router ospf 99

interface e0/1
ipv6 ospf 99 area 88
```

**Task 33.17** On R8, configure the following loopback IPv6 addresses:

| | Global Unicast |
|---|---|
| R8 interface lo8 | 2001:DB8:F:F:8000: :8 /80 |
| R8 interface lo9 | 2001:DB8:F:F:9000: :8/80 |
| R8 interface lo10 | 2001:DB8:F:F:A000: :8/80 |
| R8 interface lo11 | 2001:DB8:F:F:B000: :8/80 |

On R8, configure the following:

```
interface loopback8
ipv6 address 2001:DB8:F:F:8000::8/128

interface loopback9
ipv6 address 2001:DB8:F:F:9000::8/128
```

```
interface loopback10
ipv6 address 2001:DB8:F:F:A000::8/128

interface loopback11
ipv6 address 2001:DB8:F:F:B000::8/128
```

**Task 33.18** On R8, enable OSPFv3 on the loopback8, loopback9, loopback10, and loopback11, and on R1 advertise a single summary network encompassing all the 4 loopbacks.

On R8, configure the following:

```
interface loopback8
ipv6 ospf network point-to-point
ipv6 ospf 99 area 88

interface loopback9
ipv6 ospf network point-to-point
ipv6 ospf 99 area 88

interface loopback10
ipv6 ospf network point-to-point
ipv6 ospf 99 area 88

interface loopback11
ipv6 ospf network point-to-point
ipv6 ospf 99 area 88
```

On R1, configure the following:

```
ipv6 router ospf 99
 area 88 range 2001:DB8:F:F:8000::/66
```

On R3, I can see that only the summary route 2001:DB8:F:F:8000::/66 has been advertised.

```
R3#sh ipv6 route
IPv6 Routing Table - default - 16 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       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:DB8:A:A::1/128 [110/1000]
     via FE80::1, Tunnel23
O   2001:DB8:A:A::2/128 [110/10]
     via FE80::1, Ethernet0/1
LC  2001:DB8:A:A::3/128 [0/0]
     via Loopback0, receive
OI  2001:DB8:F:F:8000::/66 [110/1010]
     via FE80::1, Tunnel23
C   2001:DB8:2222:1::/64 [0/0]
     via Serial4/0, directly connected
L   2001:DB8:2222:1::3/128 [0/0]
     via Serial4/0, receive
O   2001:DB8:2222:1::5/128 [110/64]
     via FE80::A8BB:CCFF:FE00:500, Serial4/0
C   2001:DB8:AAAA:1::/64 [0/0]
     via Tunnel23, directly connected
O   2001:DB8:AAAA:1::1/128 [110/1000]
     via FE80::1, Tunnel23
O   2001:DB8:AAAA:1::2/128 [110/10]
     via FE80::1, Ethernet0/1
L   2001:DB8:AAAA:1::3/128 [0/0]
```

```
     via Tunnel23, receive
OI  2001:DB8:BBBB:1::/64 [110/1010]
     via FE80::1, Tunnel23
C   2001:DB8:CCCC:1::/64 [0/0]
     via Ethernet0/1, directly connected
L   2001:DB8:CCCC:1::3/128 [0/0]
     via Ethernet0/1, receive
OI  2001:DB8:FFFF:1::/64 [110/74]
     via FE80::A8BB:CCFF:FE00:500, Serial4/0
L   FF00::/8 [0/0]
     via Null0, receive
```

**Task 33.19** Configure the following IPv6 address:

| | Link Local Unicast | Global Unicast |
|---|---|---|
| R8 interface E0/0 | FE80: :1 | 2001:DB8:4444:1: :8/64 |
| R8 interface S3/0 | FE80: :1 | 2001:DB8:5555:1: :8/64 |
| R7 interface S3/0 | FE80: :2 | 2001:DB8:5555:1: :7/64 |
| R7 interface E0/1 | FE80: :1 | 2001:DB8:7777:1: :7/64 |
| R4 interface E0/1 | FE80: :2 | 2001:DB8:4444:1: :4/64 |

On R8, configure the following:

```
interface e0/0
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:4444:1::8/64
no shut

interface s3/0
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:5555:1::8/64
no shut
```

On R7, configure the following:

```
interface s3/0
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:5555:1::7/64
no shut

interface e0/1
 ipv6 address FE80::1 link-local
 ipv6 address 2001:DB8:7777:1::7/64
no shut
```

On R4, configure the following:

```
interface e0/1
 ipv6 address FE80::2 link-local
 ipv6 address 2001:DB8:4444:1::4/64
no shut
```

**Task 33.20** On R4 and on R8, configure RIPng with an ID of 48 on the connection between R4 and R8.

On R4, configure the following:

```
ipv6 unicast-routing

interface e0/1
ipv6 rip 48 enable

ipv6 router rip 48
```

On R8, configure the following:

```
ipv6 unicast-routing
```

```
interface e0/0
ipv6 rip 48 enable

ipv6 router rip 48
```

**Task 33.21** On R8 and on R7, configure EIGRPv6 in AS 78 on the connection between R8 and R7.

On R8, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 78
 no shutdown

interface Serial3/0
 ipv6 eigrp 78
```

On R7, configure the following:

```
ipv6 unicast-routing

ipv6 router eigrp 78
 no shutdown

interface Serial3/0
 ipv6 eigrp 78
```

**Task 33.22** EIGRPv6 in AS 78 should also running on the interface E0/1 of R7.

On R7, configure the following:

```
interface E0/1
 ipv6 eigrp 78
```

**Task 33.23** Ensure IPv6 connectivity between the RIPng routing domain, the OSPFv3 routing domain, and the EIGRPv6 routing domain. In particular, you should be able to IPv6 ping the lo0 of R2 from the router R7, you should be able to ping the IP address 2001:DB8:4444:1: :8/64 from the router R3, and you should be able to ping the IP address 2001:DB8:4444:1: :8/64 from the router R7.

On R8, configure the following:

```
ipv6 router eigrp 78
redistribute ospf 99 metric 1 1 1 1 1 include-connected
redistribute rip 48 metric 1 1 1 1 1 include-connected

ipv6 router ospf 99
redistribute eigrp 78 metric 100 include-connected
redistribute rip 48 metric 100 include-connected

ipv6 router rip 48
redistribute ospf 99 metric 3 include-connected
redistribute eigrp 78 metric 3 include-connected
```

I can ping from a network advertised in EIGRP to a network advertised in OSPF.

```
R7#ping 2001:DB8:A:A::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:A:A::2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/9 ms
```

I can ping from a network advertised in OSPF to a network advertised in RIP.

```
R3#ping 2001:DB8:4444:1::8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:4444:1::8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

I can ping from a network advertised in EIGRP to a network advertised in RIP.

```
R7#ping 2001:DB8:4444:1::8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:4444:1::8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 7/8/9 ms
```

**Task 33.24** The IPv4 protocol is running on the LAN between R5 and R6. Configure the following IP addresses:

| | |
|---|---|
| R5 E0/0 | 10.1.56.5/24 |
| R6 E0/0 | 10.1.56.6/24 |

On R5, configure the following:

```
int e0/0
ip address 10.1.56.5 255.255.255.0
no shut
```

On R6, configure the following:

```
int e0/0
ip address 10.1.56.6 255.255.255.0
no shut
```

**Task 33.25** R3 should be able to ping 10.1.56.6 by using the IPv6 address 2001:DB8:6666:1: :6. You are allowed to configure a static route on R3. The rest of the configuration should be performed on R5.

On R3, configure the following:

```
ipv6 route 2001:DB8:6666:1::/64 2001:DB8:2222:1::5
```

On R5, configure the following:

```
interface Serial4/0
ipv6 nat

interface Ethernet0/0
ipv6 nat

ipv6 nat v6v4 source 2001:DB8:2222:1::3 10.1.56.103
ipv6 nat v4v6 source 10.1.56.6 2001:DB8:6666:1::6

ipv6 nat prefix 2001:DB8:6666:1::/96
```

I can ping from R3 to the IPv6 address 2001:DB8:6666:1: :6:

```
R3#ping 2001:DB8:6666:1::6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:6666:1::6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 9/9/10 ms
```

**Task 33.26** Make sure that you can ping IPv6 2001:DB8:6666:1: :6 from all the loopbacks 0 in the routing domain.

On R3, configure the following:

```
ipv6 router ospf 99
redistribute static
```

On R5, configure the following:

```
ipv6 nat v6v4 source 2001:DB8:A:A::1 10.1.56.11
ipv6 nat v6v4 source 2001:DB8:A:A::2 10.1.56.12
ipv6 nat v6v4 source 2001:DB8:A:A::3 10.1.56.13
ipv6 nat v6v4 source 2001:DB8:A:A::4 10.1.56.14
ipv6 nat v6v4 source 2001:DB8:A:A::5 10.1.56.15
```

**You have completed Lab 33**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 34: Configure and Troubleshoot Quality of Service Mechanisms (Part 2)

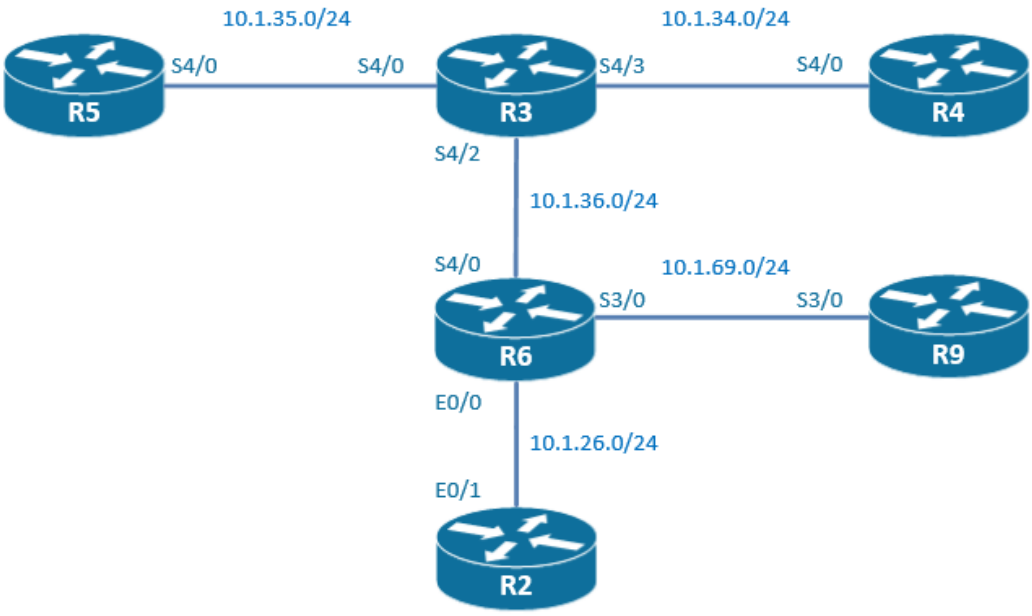
## Technologies covered

- Classification and marking
- Bandwidth percent
- LLQ
- WRED
- Dynamic flows
- ECNs

## Overview

Voice over IP will be deployed in your network and you have been tasked to configure QOS in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 34.1** R2 is a customer managed CE and R6 is the entry point to the service provider. The traffic received on the E0/0 is untrusted and should be re-marked when entering the service provider network. A class called VOICE should be created for traffic with destination ports in the RTP range 32512 32768, a class called SQL should be created for traffic with destination ports in the TCP range 1433 1434 and a class called OFFICE_BOSS should be created for traffic originated from the LAN 10.1.222.0/24.

On R6, configure the following:

```
ip access-list extended SQL_ACL
permit tcp any any range 1433 1434

ip access-list extended OFFICE_BOSS_ACL
permit tcp 10.1.222.0 0.0.0.255 any

class-map VOICE
match ip rtp 32512 256
class-map SQL
match access-group name SQL_ACL
class-map OFFICE_BOSS
match access-group name OFFICE_BOSS_ACL
```

On R6, configure a policy-map called TRAFFIC_COLOURING. This policy-map should mark the VOICE traffic with the DSCP EF, the SQL traffic with the DSCP AF31, and the OFFICE_BOSS with the DSCP AF21. The remaining unclassified traffic should have the DSCP field reset to 0.

On R6, configure the following:

```
policy-map TRAFFIC_COLOURING
class VOICE
set ip dscp ef
class SQL
set ip dscp af31
class OFFICE_BOSS
set ip dscp af21
class class-default
set ip dscp 0

int E0/0
service-policy input TRAFFIC_COLOURING
```

**Task 34.2** On the WAN link between R3 and R6, a QOS policy will be enforced. The Voice traffic should be prioritized before any other traffic in case of congestion. 10% of the bandwidth is allocated to VOICE traffic.

On R6, configure the following:

```
policy-map WAN
class VOICE
priority percent 10

interface Serial4/0
service-policy output WAN
```

On R3, configure the following:

```
class-map VOICE
```

```
match ip rtp 32512 256

policy-map VOICE_TRAFFIC
class VOICE
priority percent 10

interface Serial4/2
service-policy output VOICE_TRAFFIC
```

**Task 34.3** In case of congestion, the SQL traffic should have 30% of the bandwidth reserved and the OFFICE_BOSS traffic should have 20% of the bandwidth reserved.

On R6, configure the following:

```
policy-map WAN
class SQL
bandwidth percent 30
class OFFICE_BOSS
bandwidth percent 20
```

On R3, configure the following:

```
ip access-list extended SQL_ACL
permit tcp any any range 1433 1434

ip access-list extended OFFICE_BOSS_ACL
permit tcp 10.1.222.0 0.0.0.255 any

class-map SQL
match access-group name SQL_ACL
class-map OFFICE_BOSS
match access-group name OFFICE_BOSS_ACL

policy-map WAN
class SQL
bandwidth percent 30
class OFFICE_BOSS
bandwidth percent 20
```

**Task 34.4** In order to slow-down TCP traffic in case of congestion, some packets in the default queue should be randomly dropped before the queue is getting full and tail-dropping.

On R6, configure the following:

```
policy-map WAN
class class-default
random-detect
```

On R3, configure the following:

```
policy-map WAN
class class-default
random-detect
```

**Task 34.5** On the interface S3/0 of R6, enable WRED to begin to randomly drop packets with the IP precedence of 3 when the queue contains 20 packets and to tail-drop when the number of packets in the queue reaches more than 30 packets. 1 out of 5 packets should be randomly dropped.

On R6, configure the following:

```
policy-map WAN1
class class-default
random-detect
random-detect precedence 3 20 30 5

int s3/0
service-policy output WAN1
```

**Task 34.6** On the interface S3/0 of R6, configure the minimum possible queue size.

Queue-limit cannot be less than the configured random-detect max-threshold.

On R6, configure the following:

```
policy-map WAN1
class class-default
queue-limit 30 packets
```

**Task 34.7** On the interface S4/0 of R4, configure a hold queue of 200 packets.

On R4, configure the following:

```
int s4/0
hold-queue 200 in
hold-queue 200 out
```

**Task 34.8** On the interface S4/0 of R3, ensure that packets with a DSCP of AF21 begin to be randomly dropped when the queue contains 100 packets and to tail-drop when the number of packets in the queue reach more than 200 packets. 1 out of 10 packets should be randomly dropped.

On R3, configure the following:

```
policy-map WAN3
class class-default
random-detect dscp-based
random-detect dscp af21 100 200 10

int s4/0
service-policy output WAN3
```

You will get the following error message:

```
%QOS-6-WRED_QLIMIT_OUT_OF_SYNC: On interface Serial4/0 user-defined wred max thr
eshold higher than default queue-limit
```

This means that the queue-limit has also to be adjusted to a number bigger than 200.

On R3, configure the following:

```
policy-map WAN3
class class-default
queue-limit 300 packets
```

**Task 34.9** The TCP hosts that are transiting on the connection between R3 and R4 are supporting ECN. Enable WRED to take into account the DSCP field. Instead of randomly beginning to drop packets, WRED should be configured to mark the packet that was supposed to be dropped. The goal of this marking is to trigger the receiver to suggest the source to decrease the TCP windows size.

On R3, configure the following:

```
policy-map ECN
class class-default
random-detect dscp-based
random-detect ecn
int s4/3
service-policy output ECN
```

On R4, configure the following:

```
policy-map ECN
class class-default
random-detect dscp-based
random-detect ecn

int s4/0
service-policy output ECN
```

**You have completed Lab 34**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 35: Configure and Troubleshoot Quality of Service Mechanisms (Part 3)

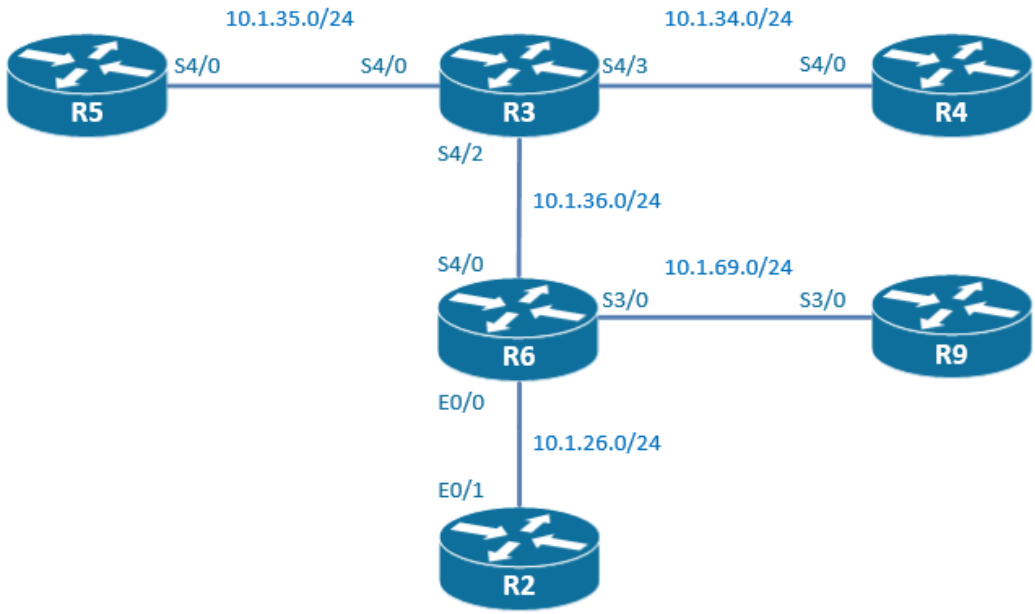
## Technologies covered

- Traffic shaping
- Policing
- Hierarchical policers
- Percent-based policers
- Header compression
- NBAR

## Overview

You have been tasked to configure QOS in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 35.1** On the WAN link between R3 and R6, enforce a QOS policy using a policy-map called Serial_Policy1. This QOS policy has 3 classes of services. Under congestion, a class called BRONZE matching DSCP AF21 has 256 kbits/s reserved, a class called SILVER matching DSCP AF31 has 256 kbits/s reserved, and a class called GOLD matching DSCP EF has 512 kbits/s reserved.

On R3, configure the following:

```
class-map match-any BRONZE
match ip dscp af21

class-map match-any SILVER
match ip dscp af31

class-map match-any GOLD
match ip dscp ef

policy-map Serial_Policy1
class GOLD
bandwidth 512
class SILVER
bandwidth 256
class BRONZE
bandwidth 256

interface s4/2
service-policy output Serial_Policy1
```

On R6, configure the following:

```
class-map match-any BRONZE
match ip dscp af21

class-map match-any SILVER
match ip dscp af31

class-map match-any GOLD
match ip dscp ef

policy-map Serial_Policy1
class GOLD
bandwidth 512
class SILVER
bandwidth 256
class BRONZE
bandwidth 256

interface s4/0
service-policy output Serial_Policy1
```

**Task 35.2** Class SILVER has to be shaped to 512 kbits/s with a normal burst size of 2048 bits.

On R3, configure the following:

```
policy-map Serial_Policy1
class SILVER
shape average 512000 2048
```

On R6, configure the following:

```
policy-map Serial_Policy1
class SILVER
shape average 512000 2048
```

**Task 35.3** Class BRONZE can obtain throughput up to a peak of 512 kbps if enough bandwidth is available.

On R3, configure the following:

```
policy-map Serial_Policy1
class BRONZE
shape peak 256000
```

On R6, configure the following:

```
policy-map Serial_Policy1
class BRONZE
shape peak 256000
```

With the shape peak CIR, the system will let you pass PIR = CIR(1 + Be/Bc). If you don't specify Be explicitly, the system uses Be = Bc, so in effect, the system will pass PIR = CIR*2.

**Task 35.4** On the interface S3/0 of R6, configure traffic-shaping. Limit the egress traffic to 512 kbps. When a BECN is received on this interface, the traffic should be shaped to a minimum of 32 kbps. Make sure that R9 reflects back to R6 the FECNs that he received.

The fecn-adapt feature is only supported on frame-relay interfaces.

On R6, configure the following:

```
policy-map VLAN101
class class-default
shape average 512000
shape adaptive 64000

int s3/0
encapsulation frame-relay
int s3/0.101 point-to-point
service-policy output VLAN101
```

On R9, configure the following:

```
policy-map VLAN101
class class-default
shape average 512000
shape fecn-adapt

int s3/0
encapsulation frame-relay
int s3/0.101 point-to-point
service-policy output VLAN101
```

**Task 35.5** On the interface E0/1.101 of R2, configure traffic-shaping. Limit the egress TCP traffic for destination port 80 to 1 kbps and the egress TCP traffic for destination port 443 to 300 kbps. Traffic not matching any access-list should be shaped to 100 kbps.

On R2, configure the following:

```
access-list 101 permit tcp any any eq 80
access-list 102 permit tcp any any eq 443

access-list 103 deny tcp any any eq 80
access-list 103 deny tcp any any eq 443
access-list 103 permit ip any any

class-map 101
```

```
match access-group 101
class-map 102
match access-group 102
class-map 103
match access-group 103

policy-map LAN
class 101
shape average 1000
class 102
shape average 300000
class 103
shape average 100000

interface e0/1.101
service-policy output LAN
```

**Task 35.6** On R3 and R6, in the policy-map called Serial_policy1, add the following classes: the class called CUSTOMER1 is matching IP DSCP CS4 and the class called CUSTOMER2 is matching IP traffic with a destination TCP port of 69.

On R3 and R6, configure the following:

```
class-map match-any CUSTOMER1
match ip dscp CS4

access-list 166 permit tcp any any eq 69

class-map match-any CUSTOMER2
match access-group 166
```

**Task 35.7** On R3 and R6, in the class called CUSTOMER1, police the traffic to a CIR of 128 kbps with a Bc of 1500 bytes and a PIR of 256 kbps with a Be of 4500 bytes. Packets that conform are sent, packets that exceed are re-marked with a COS of 0 and transmitted, and packets that violate are dropped.

This is a configuration of a two-rate policer. On R3 and R6, configure the following:

```
policy-map Serial_Policy1
class CUSTOMER1
police cir 128000 bc 1500 pir 256000 be 4500 conform-action transmit exceed-action set-
prec-transmit 0 violate-action drop
```

**Task 35.8** On R3 and R6, in the class called CUSTOMER2, police the traffic to a CIR of 64 kbps with a Bc of 1500 bytes and a PIR of 128 kbps with a Be of 3000 bytes. Packets marked with a DSCP of AF32 and AF33 that conform are sent, packets with a DSCP of AF32 and AF33 that exceed are re-marked with a DSCP of AF11 and transmitted, and packets that violate are dropped. Packets that belong to neither AF32 nor AF33 are re-marked with a DSCP of AF12. Create a class-map called AF3233.

We have to configure a hierarchical policer.

On R3 and R6, configure the following:

```
class-map match-any AF3233
   match ip dscp af32
   match ip dscp af33
policy-map CHILD_DSCP
```

```
class AF3233
police cir 128000 bc 1500 pir 256000 be 4500 conform-action transmit exceed-action set-
dscp-transmit af11 violate-action drop
class class-default
police cir 128000 bc 1500 pir 256000 be 4500 conform-action set-dscp-transmit af12

policy-map Serial_Policy1
class CUSTOMER2
service-policy CHILD_DSCP
```

**Task 35.9** On the WAN link between R3 and R4, enforce a QOS policy using a policy-map called Serial_Policy_Parent. This QOS policy has only the class default. This policy-map is used to police the traffic to 100 kbps.

On R3, configure the following:

```
policy-map Serial_Policy_Parent
class class-default
police rate 100000

Interface s4/3
Service-policy output Serial_Policy_Parent
```

On R4, configure the following:

```
policy-map Serial_Policy_Parent
class class-default
police rate 100000

Interface s4/0
Service-policy output Serial_Policy_Parent
```

**Task 35.10** Create a policy-map called Serial_Policy_Child and enforce this QOS policy on the traffic that has already been policed in the previous question. The service-policy Serial_Policy_Child has two classes called CLASS1 and CLASS2. CLASS1 is matching UDP traffic and CLASS2 is matching TCP traffic. CLASS1 should be policed to 20 kbps and CLASS 2 should be policed to 50 kbps.

On R3 and R4, configure the following:

```
access-list 171 permit udp any any
access-list 172 permit tcp any any

class-map CLASS1
match access-group 171
class-map CLASS2
match access-group 172

policy-map Serial_Policy_Child
class CLASS1
police rate 20000
class CLASS2
police rate 50000

policy-map Serial_Policy_Parent
class class-default
service-policy Serial_Policy_Child
```

**Task 35.11** On the WAN link between R3 and R5, enforce a QOS policy using a policy-map called Serial_Policy_Percentage. This QOS policy has only the class default. This policy-map is used to police the traffic to a CIR of 60 % of the available bandwidth and to a PIR of 90% of the available bandwidth.

On R3 and R5, configure the following:

```
policy-map Serial_Policy_Percentage
class class-default
police cir percent 60 pir percent 90

int s4/0
service-policy output Serial_Policy_Percentage
```

**Task 35.12** On the WAN link between R3 and R5, configure PPP encapsulation and enable RTP enhanced header compression.

On R3 and R5, configure the following:

```
int s4/0
encapsulation ppp
ip rtp header-compression ietf-format
ip header-compression recoverable-loss dynamic
```

**Task 35.13** Consider that the connection between R3 and R4 is a satellite link. Enable RTP header compression on this connection.

On R3, configure the following:

```
int s4/3
ip rtp header-compression ietf-format periodic-refresh
```

On R4, configure the following:

```
int s4/0
ip rtp header-compression ietf-format periodic-refresh
```

**Task 35.14** On the link between R6 and R2, enforce a QOS policy using a policy-map called Serial_Policy_NBAR on R6. This QOS policy has 2 classes called LOTUS and URL. LOTUS class is matching Lotus notes traffic and is shaped to 512 kbps. URL class is matching HTTP traffic that contains a URL of /iPexpert is policed to 512 kbps.

On R6, configure the following:

```
class-map LOTUS
match protocol lotus-notes
class-map URL
match protocol http url /iPexpert

policy-map Serial_Policy_NBAR
class LOTUS
shape average 512000
class URL
police rate 512000
int e0/0.101
service-policy output Serial_Policy_NBAR
```

On R2, configure the following:

```
class-map LOTUS
match protocol lotus-notes
class-map URL
match protocol http url /iPexpert

policy-map LAN
class LOTUS
shape average 512000
class URL
police rate 512000
```

**You have completed Lab 35**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 36: Security Part I

## Overview

Please look at the provided diagrams and read through the whole lab before you start. Read the directions very carefully to make sure you are doing what is being asked of you. This concept is very important when you take the CCIE lab administered by Cisco.

It is recommended to create your own diagram at the beginning of each lab so any potential information you find useful during your preparations can be reflected on this drawing, making it much easier when you step into the real lab.

Multiple topology drawings are available for this chapter.

## General Rules

You will need to pre-configure the network with the base configuration files.

---

***NOTE: Static/default routes are NOT allowed unless otherwise stated in the task.***

***NOTE: You can use "cisco" for any password if other password was not explicitly mentioned in the question.***

---

**Estimated Time to Complete: 3-4 hours**

## Pre-Lab setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

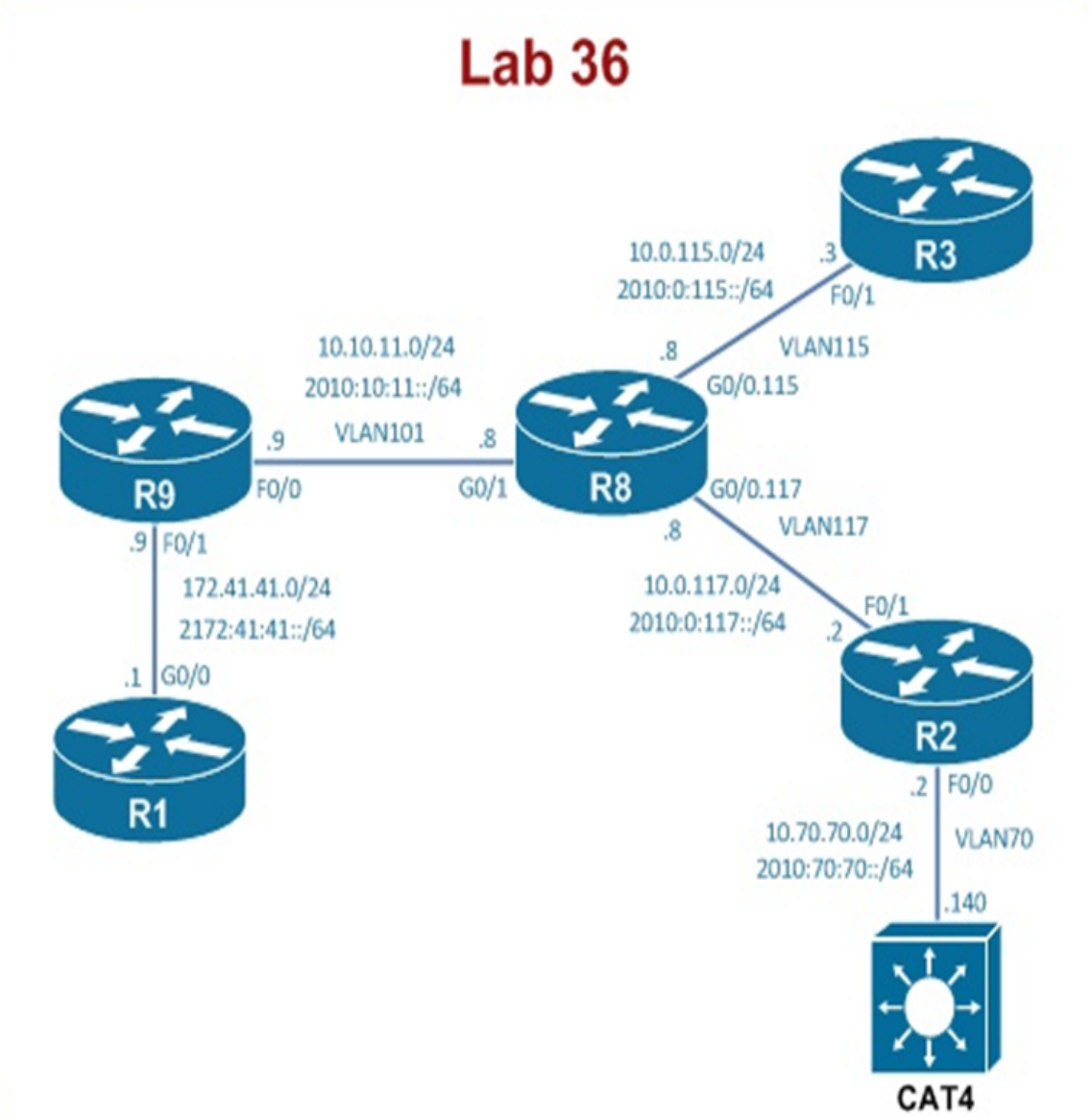
## Prerequisites

Load the initial configuration files before starting to work on the tasks.

| Device | Port | VLAN | IP Address |
|---|---|---|---|
| R1 | G0/0 | 41 | 172.41.41.1/24<br>2172:41:41::1/64 |
| | Loop0 | | 1.1.1.1/24<br>1::1/64 |
| R3 | F0/1 | 115 | 10.0.115.3/24<br>2010:0:115::3/64 |
| | Loop0 | | 3.3.3.3/24<br>3::3/64 |
| R2 | F0/0 | 70 | 10.70.70.2/24<br>2010:70:70::2/64 |
| | F0/1 | 117 | 10.0.117.2/24<br>2010:0:117::2/64 |
| | Loop0 | | 2.2.2.2/24<br>2::2/64 |
| R9 | F0/1 | 41 | 172.41.41.9/24<br>2172:41:41::9/64 |
| | F0/0 | 101 | 10.10.11.9/24<br>2010:10:11::9/64 |
| | Loop0 | | 9.9.9.9/24<br>9::9/64 |
| R8 | G0/1 | 101 | 10.10.11.8/24<br>2010:10:11::8/64 |
| | G0/0.115 | 115 | 10.0.115.8/24<br>2010:0:115::8/64 |
| | G0/0.117 | 117 | 10.0.117.8/24<br>2010:0:117::8/64 |
| | Loop0 | | 8.8.8.8/24<br>8::8/64 |
| CAT4 | SVI70 | 70 | 10.70.70.140/24 |



### Task 36.1 AAA

- Configure R1 for AAA.
- Users who telnet to this device should be authenticated by the default method list using a line password ("iPexpert"). Console line should not be affected.
- PPP authentication requests should be authenticated using RADIUS server (10.10.11.90).
- Protect RADIUS communication using key "iPexpert". RADIUS traffic should be sent using new port numbers.
- Network access should be authorized – if RADIUS is down authorization should succeed for authenticated users.
- Enable accounting for network traffic – records should be kept for when a session initiates and when it terminates.

## Detailed Solution

**R1**

```
aaa new-model

aaa authentication login default line
aaa authentication login NO none

aaa authentication ppp default radius
aaa authorization network default group radius if-authenticated
aaa accounting network default start-stop group radius
line vty 0 4
 password iPexpert
line con 0
 login authentication NO
radius-server host 10.10.11.90 auth-port 1812 acct-port 1813 key iPexpert
```

Note the difference between a "default" and custom method list – "default" list does not even have to be applied to a line/interface to take effect. To activate a custom list you must always apply it.

It is possible to configure fallback methods for Authentication and Authorization. This can be configured by adding a second/third/etc. method right after a primary method was defined via "aaa authentication/authorization". In our case "if-authenticated" was added meaning that if RADIUS server is not available (e.g. is down), users will be automatically authorized to the network if they successfully authenticated.

Even that there is no RADIUS server on the exam you may still be asked to prepare your AAA Client (NAD) for RADIUS/TACACS+ communication. New port numbers assigned by IANA for RADIUS are 1812 and 1813 (vs "old" 1645/1646).

## Verification

```
R9#telnet 1.1.1.1
Trying 1.1.1.1 ... Open


User Access Verification

Password:

R1>
```

Now go to R1 (console) and say "exit" :

```
R1# exit

R1 con0 is now available
```

```
Press RETURN to get started.

R1>enable
R1#
R1#sh radius server-group all
Server group radius
    Sharecount = 1  sg_unconfigured = FALSE
    Type = standard  Memlocks = 1
    Server(10.10.11.90:1812,1813) Transactions:
    Authen: 0   Author: 0       Acct: 0
    Server_auto_test_enabled: FALSE
```

## Task 36.2 Local Authentication & Authorization

- Enable SSH on R3. Use domain-name “ipexpert.com”.
- Create two local user accounts – “admin” and “secops”.
- When “admin” connects to R3 remotely via SSH it should be automatically placed at level 15 after successful authentication.
- When someone authenticates as “secops” he/she should be placed at level 8.
- Anyone who knows enable password (“cisco”) should be able to access Privilege Level.
- Make sure that enable password is MD5-encrypted.
- Don’t use AAA to accomplish this task.

## Detailed Solution

**R3**

```
ip domain-name ipexpert.com
cry key gen rsa mod 1024

username admin privilege 15 password cisco
username secops privilege 8 password cisco
enable secret cisco

line vty 0 4
 login local
```

To enable SSH access on a router/switch you must configure a domain-name and generate an RSA Key Pair. Whenever modulus 768 or higher is used SSHv2 is also supported (otherwise only SSHv1 is available).

Local database is a simplest way of assigning users to Privilege Levels & CLI Views.

Using “enable secret” instead of “enable password” ensures that password is MD5 encrypted and cannot be easily reversed like if “service password-encryption” was used.

## Verification

```
R9#ssh -l admin 3.3.3.3
Password:

R3#sh privilege
Current privilege level is 15
R3#
R3#exit

[Connection to 3.3.3.3 closed by foreign host]
R9#
R9#ssh -l secops 3.3.3.3
Password:
```

```
R3#show privilege
Current privilege level is 8
R3#
R3#enable
Password:
R3#show privilege
Current privilege level is 15
R3#who
    Line       User       Host(s)              Idle       Location
   0 con 0                 idle                 00:02:47
*514 vty 0     secops     idle                 00:00:00 10.10.11.9

  Interface    User               Mode         Idle     Peer Address

R3#
R3#sh run | in enable
enable secret 5 $1$HuUP$nc3fHDVs4J7Uo.rs9XMaa0
```

### Task 36.3 AAA Exec Authorization

- Remove local authentication on R3. Enable AAA.
- Users “admin” and “secops” should be still assigned to privilege levels 15 and 8, respectively, after successful authentication.
- User “secops” should be able to access the following commands :
  - show running-config
  - configure terminal
  - ip routing
  - ip route
- User “admin” should have access to all commands.
- When “secops” issues the “enable” command he should be automatically given Privilege Level access without prompting for password.
- Don’t use any default method lists in this task.

## Detailed Solution

**R3**

```
line vty 0 4
 no login local

aaa new-model
aaa authentication login VTYAUTHC local
aaa authorization exec VTYAUTHZ local
aaa authentication enable default none

line vty 0 4
 authorization exec VTYAUTHZ
 login authentication VTYAUTHC

privilege exec level 8 show running-config
privilege exec level 8 configure terminal
privilege configure level 8 ip route
privilege configure level 8 ip routing
```

Since default lists are not allowed in this task we had to use custom ones which means we have to apply them so they take effect.

The “privilege” command can be used for basic local command authorization when you want to limit the available commands for a user.

## Verification

```
R9#ssh -l admin 3.3.3.3
Password:

R3#show privilege
Current privilege level is 15
R3#
R3#show running-config
Building configuration...


Current configuration : 2131 bytes
!
! Last configuration change at 16:47:02 UTC Tue Jul 15 2014
!
version 15.1
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R3
!
boot-start-marker
boot-end-marker
!
!
enable secret 5 $1$HuUP$nc3fHDVs4J7Uo.rs9XMaa0
!
aaa new-model
!
!
          ---- output omitted ----
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router bgp 10
R3(config-router)#exit
R3(config)#router ospf 1
R3(config-router)#exit

R9#ssh -l secops 3.3.3.3
Password:

R3#show privilege
Current privilege level is 8
R3#show startup-config
           ^
% Invalid input detected at '^' marker.

R3#show running-config
Building configuration...

Current configuration : 130 bytes
!
! Last configuration change at 16:52:03 UTC Tue Jul 15 2014 by admin
!
boot-start-marker
boot-end-marker
!
!
!
!
!
!
```

```
!
!
!
end
R3#conf terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#ip routing
R3(config)#ip route 1.2.3.4 255.255.255.255 null0
R3(config)#no ip route 1.2.3.4 255.255.255.255 null0

R3(config)#router bgp 10
              ^
% Invalid input detected at '^' marker.

R3(config)#router ospf 1
             ^
% Invalid input detected at '^' marker.

R3(config)#end

R3#enable
R3#show privilege
Current privilege level is 15
R3#who
    Line       User       Host(s)              Idle       Location
   0 con 0                 idle                 00:00:17
*514 vty 0     secops     idle                 00:00:00 10.10.11.9
```

## Task 36.4 AAA with CLI Views

- Configure R2 for CLI Views using AAA.
- Create a local user account “administrator” who should be given access to all commands.
- Create a local user account “netops” who should be able to do the following :
  - Access all “show” commands except for any “show crypto” command.
  - Issue “ping” and “telnet”.
  - Configure any dynamic routing protocol.
- Create a local user account “secops” who should be able to do the following :
  - Access all “show crypto” commands.
  - Configure any “crypto” command in the global config mode.
- Create another user account - “ops”. This person should be always able to do what “netops” and “secops” can do.
- Use “iPexpert” as a password for all views.

## Detailed Solution

**R2**

```
enable secret cisco
```

`enable view` (privileged mode command)

```
parser view noc
 secret iPexpert
 commands exec include telnet
 commands exec include ping
 commands exec include configure terminal
 commands exec include all show
 commands configure include all router

parser view sec
 secret iPexpert
 commands exec include configure terminal
 commands exec include-exclusive all show crypto
```

```
 commands configure include all crypto

parser view ops superview
 secret iPexpert
 view noc
 view sec

aaa new-model
aaa authentication login VTYAUTHC local
aaa authorization exec VTYAUTHZ local

line vty 0 4
 authorization exec VTYAUTHZ
 login authentication VTYAUTHC

username administrator view root password cisco
username netops view noc password cisco
username secops view sec password cisco
username ops view ops password cisco
```

Role Based CLI Access (RBCA) is a tool you can use locally to mimic real command authorization that can be only performed with TACACS+. RBCA associates users with Views – a View is simply a set of commands you want to make accessible to a particular user. Remember that EXEC authorization must be enabled so that configured View can enforced (you assign a View to the user via the "view" command).

To configure this feature you need to first define "enable" password which is needed to switch from the Privileged Mode to the View Context via the "parser view" command (specifically to a special type of View - Root View).

We have two special types of Views: Root (exists by default) and Superview. Root View is like Privileged Mode (gives access to all commands) but in addition allows you to create regular Views (that's why before you create a View you have to first leave the Privileged Mode and enter the Root View). A Superview is a combination of 2 or more regular Views – any commands available in those regular Views will be automatically available in the Superview.

Finally notice that any command that was added to a View exclusively ("include-exclusive") will not be available in any other regular View.

## Verification

Telnet to R2 and test this configuration using different user accounts:

```
R9#telnet 2.2.2.2
Trying 2.2.2.2 ... Open


User Access Verification

Username: administrator
Password:

R2>show privilege
Currently in View Context with view 'root'
R2>show cry sess
R2>
R2>conf t
R2(config)>router ospf 1
R2(config-router)>exit
R2>sh run
Building configuration...
```

```
Current configuration : 2535 bytes
!
! Last configuration change at 22:08:05 UTC Tue Jul 15 2014 by administrator
!
version 15.1
parser view noc
 secret 5 $1$hL/L$37tHSt66abyg1S6FNvSQF/
 commands configure include all router
 commands exec include telnet
 commands exec include ping
 commands exec include configure terminal
 commands exec include configure
 commands exec include all show
!
parser view sec
 secret 5 $1$Kt7H$G40kmQoh5MSP4ju6iXZUO0
 commands configure include all crypto
 commands exec include configure terminal
 commands exec include configure
 commands exec include-exclusive all show crypto
 commands exec include show
!
     ---- Output omitted ----
R9#telnet 2.2.2.2
Trying 2.2.2.2 ... Open

User Access Verification

Username: netops
Password:

R2>show privi
Currently in View Context with view 'noc'
R2>show parser view
Current view is 'noc'

R2>sh run
Building configuration...

Current configuration : 133 bytes
!
! Last configuration change at 22:08:05 UTC Tue Jul 15 2014 by administrator
!
!
!
!
!
!
router ospf 1
 router-id 2.2.2.2
!
!
end

R2>show ip route
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

Gateway of last resort is 10.0.117.8 to network 0.0.0.0
```

```
O*E2  0.0.0.0/0 [110/1] via 10.0.117.8, 06:14:26, FastEthernet0/1
      5.0.0.0/32 is subnetted, 1 subnets
O        3.3.3.3 [110/3] via 10.0.117.8, 06:14:36, FastEthernet0/1
      7.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        7.7.7.0/24 is directly connected, Loopback0
L        2.2.2.2/32 is directly connected, Loopback0
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
O        10.0.115.0/24 [110/2] via 10.0.117.8, 06:14:36, FastEthernet0/1
C        10.0.117.0/24 is directly connected, FastEthernet0/1
L        10.0.117.2/32 is directly connected, FastEthernet0/1
O        10.10.11.0/24 [110/2] via 10.0.117.8, 06:14:26, FastEthernet0/1
C        10.70.70.0/24 is directly connected, FastEthernet0/0
L        10.70.70.2/32 is directly connected, FastEthernet0/0
      11.0.0.0/32 is subnetted, 1 subnets
O        11.11.11.11 [110/2] via 10.0.117.8, 06:14:36, FastEthernet0/1

R2>show crypto sess
          ^
% Invalid input detected at '^' marker.

R2>show crypto ?
% Unrecognized command

R2>?
Exec commands:
  <1-99>      Session number to resume
  configure   Enter configuration mode
  credential  load the credential info from file system
  enable      Turn on privileged commands
  exit        Exit from the EXEC
  ping        Send echo messages
  show        Show running system information
  telnet      Open a telnet connection

R2>conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)>router ospf 1
R2(config-router)>exit
R2(config)>router bgp 10
R2(config-router)>exit
R2(config)>?
Configure commands:
  do      To run exec commands in config mode
  exit    Exit from configure mode
  router  Enable a routing process

R2(config)>router ?
  bgp       Border Gateway Protocol (BGP)
  eigrp     Enhanced Interior Gateway Routing Protocol (EIGRP)
  isis      ISO IS-IS
  iso-igrp  IGRP for OSI networks
  mobile    Mobile routes
  odr       On Demand stub Routes
  ospf      Open Shortest Path First (OSPF)
  rip       Routing Information Protocol (RIP)

R9#telnet 2.2.2.2
Trying 2.2.2.2 ... Open


User Access Verification

Username: secops
Password:

R2>show parser view
Current view is 'sec'
```

```
R2>show privilege
        ^
% Invalid input detected at '^' marker.

R2>show ip route
       ^
% Invalid input detected at '^' marker.

R2>show ?
  crypto  Encryption module
  flash:  display information about flash: file system
  parser  Display parser information

R2>show cry sess
R2>?
Exec commands:
  <1-99>      Session number to resume
  configure   Enter configuration mode
  credential  load the credential info from file system
  enable      Turn on privileged commands
  exit        Exit from the EXEC
  show        Show running system information


R2>configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)>?
Configure commands:
  crypto  Encryption module
  do      To run exec commands in config mode
  exit    Exit from configure mode

R2(config)>crypto ?
  call          Configure Crypto Call Admission Control
  ctcp          Configure cTCP encapsulation
  dynamic-map   Specify a dynamic crypto map template
  engine        Enter a crypto engine configurable menu
  gdoi          Configure GDOI policy
  identity      Enter a crypto identity list
  ipsec         Configure IPSEC policy
  isakmp        Configure ISAKMP policy
  key           Long term key operations
  keyring       Key ring commands
  logging       logging messages
  map           Enter a crypto map
  mib           Configure Crypto-related MIB Parameters
  pki           Public Key components
  provisioning  Secure Device Provisioning
  wui           Crypto HTTP configuration interfaces
  xauth         X-Auth parameters


R9#telnet 2.2.2.2
Trying 2.2.2.2 ... Open


User Access Verification

Username: ops
Password:

R2>?
Exec commands:
  <1-99>      Session number to resume
  configure   Enter configuration mode
  credential  load the credential info from file system
  enable      Turn on privileged commands
```

```
  exit        Exit from the EXEC
  ping        Send echo messages
  show        Show running system information
  telnet      Open a telnet connection

R2>show crypto sess
R2>
R2>show crypto ?
  call              Show crypto call admission info
  ctcp              cTCP connections
  datapath          Data Path
  debug-condition   Debug Condition filters
  dynamic-map       Crypto map templates
  eli               Encryption Layer Interface
  engine            Show crypto engine info
  gdoi              Show crypto gdoi
  ha                Crypto High Availability information
  identity          Show crypto identity list
  ipsec             Show IPSEC policy
  isakmp            Show ISAKMP
  key               Show long term public keys
  map               Crypto maps
  mib               Show Crypto-related MIB Parameters
  optional          Optional Encryption Status
  pki               Show PKI
  route             Show crypto VPN routes
  ruleset           Show crypto rules on outgoing packets
  session           Show crypto sessions (tunnels)
  sockets           Secure Socket Information
  tech-support      Displays relevant crypto information

R2>conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)>?
Configure commands:
  crypto  Encryption module
  do      To run exec commands in config mode
  exit    Exit from configure mode
  router  Enable a routing process

R2(config)>crypto ?
  call           Configure Crypto Call Admission Control
  ctcp           Configure cTCP encapsulation
  dynamic-map    Specify a dynamic crypto map template
  engine         Enter a crypto engine configurable menu
  gdoi           Configure GDOI policy
  identity       Enter a crypto identity list
  ipsec          Configure IPSEC policy
  isakmp         Configure ISAKMP policy
  key            Long term key operations
  keyring        Key ring commands
  logging        logging messages
  map            Enter a crypto map
  mib            Configure Crypto-related MIB Parameters
  pki            Public Key components
  provisioning   Secure Device Provisioning
  wui            Crypto HTTP configuration interfaces
  xauth          X-Auth parameters

R2(config)>router ?
  bgp       Border Gateway Protocol (BGP)
  eigrp     Enhanced Interior Gateway Routing Protocol (EIGRP)
  isis      ISO IS-IS
  iso-igrp  IGRP for OSI networks
  mobile    Mobile routes
  odr       On Demand stub Routes
  ospf      Open Shortest Path First (OSPF)
  rip       Routing Information Protocol (RIP)
```

### Task 36.5 Traffic Filtering – Standard ACLs

- R8 is configured with the following loopback networks :
  - 111.111.111.2/32
  - 111.111.111.4/32
  - 111.111.111.6/32
- R1 should be configured to drop & log packets sourced from those addresses using a Standard ACL. This ACL should have as few entries as possible with a minimum overlap.
- All routers should be able to reach R3 only from interfaces configured with odd IPv4 addresses.
- Traffic sourced from other IPv4 addresses should be dropped.
- Implement this using a Standard ACL with a single "deny" entry.

## Detailed Solution

```
R1
access-list 11 deny   111.111.111.0 0.0.0.6 log
access-list 11 permit any

int g0/0
 ip access-group 11 in

R3
access-list 11 deny   0.0.0.0 255.255.255.254
access-list 11 permit any

int f0/1
 ip access-group 11 in
```

Standard ACLs can only filter traffic based on source addresses.

The only difficulty in this task was to figure out the correct wildcard mask to meet the task requirements. Probably the easiest way is to write down an address in binary (or just a particular octet that changes) and see what corresponding bits change vs. what stay the same:

2 = 00000010
4 = 00000100
6 = 00000110

Here we see that bits number 2 and 3 change – there are two bits total which means that we have four possible values to look for – 00, 01, 10, and 11. Three of them cover our addresses but 00 does not. So 00 is our minimum overlap that we will have if we want to cover those addresses in a single ACL line.

OK so how we build our wildcard? Remember that '0s' in wildcard means that respective bits should stay the same but '1s' means that we don't care. So we end up with 00000110 which are 6 in decimal.

For the second part of the task we want to deny traffic coming from even addresses. Even addresses are those that end with "0" in binary (1st bit); odd addresses always have this bit set to "1". That's why in our wildcard we need to say that we only care about bit number 1 (which is set to 0 in the address part) - so the wildcard is 11111110 in binary.

## Verification

```
R8#ping 1.1.1.1 so l12
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 111.111.111.2
U.U.U
Success rate is 0 percent (0/5)

R8#ping 1.1.1.1 so l14
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 111.111.111.4
U.U.U
Success rate is 0 percent (0/5)

R8#ping 1.1.1.1 so l16
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 111.111.111.6
U.U.U
Success rate is 0 percent (0/5)

R1#sh access-list 11
Standard IP access list 11
    10 deny   111.111.111.0, wildcard bits 0.0.0.6 log (15 matches)
    20 permit any (8 matches)
```

Make sure routing is not affected :

```
R3#clear ip ospf pro
Reset ALL OSPF processes? [no]: yes
R3#
R3#
*Jul 16 14:31:16.874: %OSPF-5-ADJCHG: Process 1, Nbr 11.11.11.11 on FastEthernet0/1 from
FULL to DOWN, Neighbor Down: Interface down or detached
*Jul 16 14:31:16.886: %OSPF-5-ADJCHG: Process 1, Nbr 11.11.11.11 on FastEthernet0/1 from
LOADING to FULL, Loading Done
R3#
R3#sh access-l
Standard IP access list 11
    10 deny   0.0.0.0, wildcard bits 255.255.255.254
    20 permit any (16 matches)

R8#ping 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

R8#ping 3.3.3.3 so l12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
Packet sent with a source address of 111.111.111.2
U.U.U
Success rate is 0 percent (0/5)

R2#ping 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R3#sh access-list 11
Standard IP access list 11
    10 deny   0.0.0.0, wildcard bits 255.255.255.254 (8 matches)
20 permit any (49 matches)
```

## **Task 36.6** Traffic Filtering – Extended ACLs

- Configure an IPv4 ACL on R9's F0/0 inbound. Allow the following traffic :
  - OSPFv2 – be very specific here.
  - R1 acts as a Telnet, Web and SQLNET (TCP 1521) server – permit this traffic only to its loopback0 in a single ACL line.
  - UDP-based traceroute (IOS) to any destination – use a single ACL line.
  - All TCP segments destined to R1's Loopback 44 but only with SYN and ACK bits set and FIN bit being not set.
  - All IP packets with any source and destination with a TTL 0-253 and 255 (in a single ACL line).
  - Routers R2, R9, and R8 should be able to ping all interfaces of R1 (regardless of the TTL in the packets). R1 should be able to ping all routers except R3 as well.
- Configure an IPv6 ACL on R9's F0/0 inbound in the following way :
  - Allow Telnet to R1's Loopback 0.
  - Deny all IPv6 packets with a missing or unknown L4 information.
  - Deny all IPv6 packets with Routing Extension Header.
  - Make sure OSPFv3 adjacencies are not affected, same as all ICMPv6 packets.
- Deny and log all other IPv4 & IPv6 traffic. Make sure you see a log message for every packet dropped by this entry.

## Detailed Solution

**R9**

```
ip access-list log-update threshold 1

ip access-list extended OUT_IN
 permit ospf host 10.10.11.8 host 224.0.0.5
 permit ospf host 10.10.11.8 host 224.0.0.6
 permit ospf host 10.10.11.8 host 10.10.11.9
 permit tcp any host 1.1.1.1 eq telnet www 1521
 permit udp any any range 33434 33464
 permit tcp any host 44.44.44.44 match-all +ack -fin +syn
 deny   tcp any host 44.44.44.44
 permit icmp any any echo
 permit icmp any any echo-reply
 permit ip any host 144.144.144.144 ttl neq 254
 deny   ip any any log

ipv6 access-list OUT6_IN
 permit tcp any host 1::1 eq 23
 permit 89 any any
 permit icmp any any
 deny ipv6 any any undetermined-transport
 deny ipv6 any any routing
 deny ipv6 any any log

int F0/0
 ip access-group OUT_IN in
 ipv6 traffic-filter OUT6_IN in
```

Extended ACLs can filter traffic based on many criteria – source & destination addresses/port numbers, TCP flags, ToS values, IP Options, TTL values, and other fields where some of them depend on the protocol specified in the ACL line.

Same as Standard ACLs Extended access-lists are processed top-down until a match is found. If no explicit line was matched an implicit "deny ip any" will drop the packet.

ICMP entries were put above the TTL entry for Loopback144 so pings are always allowed regardless of the TTL.

IPv6 ACLs (no standard here, only extended) work in the same way as in IPv4. Some matching criteria are obviously different as the protocol is. Note a small difference in the syntax – to apply an IPv6 ACL you should use the "ipv6 traffic-filter" command.

## Verification

First, test OSPF adjacency:

```
R9#clear ip ospf process
Reset ALL OSPF processes? [no]: yes

R9#sh access-list
Extended IP access list OUT_IN
    10 permit ospf host 10.10.11.8 host 224.0.0.5 (5 matches)
    20 permit ospf host 10.10.11.8 host 224.0.0.6
    30 permit ospf host 10.10.11.8 host 10.10.11.9 (8 matches)
    40 permit tcp any host 1.1.1.1 eq telnet www 1521
    50 permit udp any any range 33434 33464
    60 permit tcp any host 44.44.44.44 match-all +ack -fin +syn
    70 deny tcp any host 44.44.44.44
    75 permit icmp any any echo
    76 permit icmp any any echo-reply
    80 permit ip any host 144.144.144.144 ttl neq 254
    100 deny ip any any log
```

Let's now have a look at Telnet, WWW and SQLNET :

```
R8#telnet 1.1.1.1
Trying 1.1.1.1 ... Open
User Access Verification

Password:

R1>

R8#telnet 1.1.1.1 80
Trying 1.1.1.1, 80 ... Open
get /
HTTP/1.1 400 Bad Request
Date: Wed, 16 Jul 2014 16:01:42 GMT
Server: cisco-IOS
Connection: close
Accept-Ranges: none

400 Bad Request
[Connection to 1.1.1.1 closed by foreign host]

R8#telnet 1.1.1.1 1521
Trying 1.1.1.1, 1521 ...
% Connection refused by remote host

R9#sh access-list OUT_IN
Extended IP access list OUT_IN
    10 permit ospf host 10.10.11.8 host 224.0.0.5 (67 matches)
    20 permit ospf host 10.10.11.8 host 224.0.0.6
    30 permit ospf host 10.10.11.8 host 10.10.11.9 (8 matches)
    40 permit tcp any host 1.1.1.1 eq telnet www 1521 (40 matches)
    50 permit udp any any range 33434 33464
    60 permit tcp any host 44.44.44.44 match-all +ack -fin +syn
    70 deny tcp any host 44.44.44.44
    75 permit icmp any any echo
    76 permit icmp any any echo-reply
    80 permit ip any host 144.144.144.144 ttl neq 254
    100 deny ip any any log
```

Moving on we can look at traceroute and Pings :

```
R8#traceroute 1.1.1.1
Type escape sequence to abort.
Tracing the route to 1.1.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.11.9 4 msec 0 msec 0 msec
  2 172.41.41.1 0 msec *  0 msec

R8#traceroute 172.41.41.1
Type escape sequence to abort.
Tracing the route to 172.41.41.1
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.11.9 0 msec 0 msec 0 msec
  2 172.41.41.1 0 msec *  0 msec

R2#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R2#ping 144.144.144.144

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 144.144.144.144, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R8#ping 144.144.144.144
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 144.144.144.144, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R9#ping 144.144.144.144
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 144.144.144.144, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

R1#ping 9.9.9.9

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 9.9.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R1#ping 10.10.11.8

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.11.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R1#ping 3.3.3.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

R1#ping 2.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R9#sh access-list OUT_IN
Extended IP access list OUT_IN
    10 permit ospf host 10.10.11.8 host 224.0.0.5 (113 matches)
    20 permit ospf host 10.10.11.8 host 224.0.0.6
    30 permit ospf host 10.10.11.8 host 10.10.11.9 (8 matches)
    40 permit tcp any host 1.1.1.1 eq telnet www 1521 (61 matches)
    50 permit udp any any range 33434 33464 (12 matches)
    60 permit tcp any host 44.44.44.44 match-all +ack -fin +syn
    70 deny tcp any host 44.44.44.44
    75 permit icmp any any echo (15 matches)
    76 permit icmp any any echo-reply (15 matches)
    80 permit ip any host 144.144.144.144 ttl neq 254
    100 deny ip any any log
```

Finally we will now test loopback 44 and 144 access. First Telnet packet from R8 only has a SYN flag set – that's why we match "deny tcp any host 44.44.44.44". Then when you telnet from R1 to 11 it sends a SYN and gets SYN+ACK back from R8 (which is allowed). The Telnet session does not fully establish since remaining segments from R8 don't have a SYN flag turned and match our deny.

```
R8#telnet 44.44.44.44
Trying 44.44.44.44 ...
% Destination unreachable; gateway or host down

R1#telnet 11.11.11.11 /source l44
Trying 11.11.11.11 ... Open
```

Note the difference – packets from R2 are coming with TTL of 253 but R8 sends them with TTL 254 – thus R8 telnet matches our explicit deny:

```
R2#telnet 144.144.144.144
Trying 144.144.144.144 ... Open

User Access Verification
Password:

R1>exit

R8#telnet 144.144.144.144
Trying 144.144.144.144 ...
% Destination unreachable; gateway or host down

R9#
*Jul 16 15:49:54.757: %SEC-6-IPACCESSLOGP: list OUT_IN denied tcp 10.10.11.8(22531) ->
144.144.144.144(23), 1 packet

R9#sh access-list OUT_IN
Extended IP access list OUT_IN
    10 permit ospf host 10.10.11.8 host 224.0.0.5 (180 matches)
    20 permit ospf host 10.10.11.8 host 224.0.0.6
    30 permit ospf host 10.10.11.8 host 10.10.11.9 (8 matches)
    40 permit tcp any host 1.1.1.1 eq telnet www 1521 (61 matches)
    50 permit udp any any range 33434 33464 (12 matches)
    60 permit tcp any host 44.44.44.44 match-all +ack -fin +syn (1 match)
    70 deny tcp any host 44.44.44.44 (18 matches)
    75 permit icmp any any echo (15 matches)
    76 permit icmp any any echo-reply (15 matches)
    80 permit ip any host 144.144.144.144 ttl neq 254 (29 matches)
    100 deny ip any any log (6 matches)
```

Now IPv6 :

```
R9#clear ipv6 ospf proc
Reset ALL OSPF processes? [no]: yes

R9#sh ipv ospf neig
```

```
            OSPFv3 Router with ID (9.9.9.9) (Process ID 1)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
1.1.1.1           1   FULL/DR         00:00:39    4               GigabitEthernet0/1
11.11.11.11       1   FULL/DR         00:00:33    4               GigabitEthernet0/0

R8#telnet 1::1
Trying 1::1 ... Open

User Access Verification
Password:

R1>who
    Line       User       Host(s)              Idle       Location
   0 con 0                 idle                 00:13:05
*514 vty 0                 idle                 00:00:00 2010:10:11::8

  Interface    User               Mode         Idle     Peer Address

R8#telnet 2172:41:41::1
Trying 2172:41:41::1 ...
% Destination unreachable; gateway or host down

R8#ping 1::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R8#traceroute ipv6
Target IPv6 address: 1::1
Source address:
Insert source routing header? [no]: yes
Nexthop address: 1::1
Nexthop address: 3::3
Nexthop address: 1::1
Nexthop address:
Numeric display? [no]: yes
Timeout in seconds [3]: 2
Probe count [3]: 2
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Priority [0]:
Port Number [0]:
Type escape sequence to abort.
Tracing the route to 1::1

  1 2010:10:11::9 !A  !A



R9#sh ipv access-l
IPv6 access list OUT6_IN
    permit tcp any host 1::1 eq telnet (34 matches) sequence 10
    permit 89 any any (107 matches) sequence 15
    permit icmp any any (15 matches) sequence 16
    deny ipv6 any any undetermined-transport (17 matches) sequence 20
    deny ipv6 any any routing (2 matches) sequence 30
    deny ipv6 any any log (37 matches) sequence 40
```

## **Task 36.7** Traffic Filtering – Time Ranges & Object-Groups

- All web traffic destined to R8's Loopback 12, 14, and 16 interfaces should be denied during business hours Mon-Fri 9am-5pm. This Includes encrypted traffic.

- November 11, 2014 has been declared a no-work day. Ensure that no traffic is allowed to the above mentioned loopbacks for the entire day.
- Permit and log all IPv4 DNS traffic (TCP and UDP) to R8's Loopback0 and 12. Include source MAC address in the logs. Use a single ACL entry to configure this.
- All other traffic should not be affected.

## Detailed Solution

### R8

```
time-range OFF
 absolute start 00:00 11 November 2014 end 23:59 11 November 2014
time-range WORK
 periodic weekdays 9:00 to 16:59

object-group service DNS
 tcp eq domain
 udp eq domain
object-group network DNSSERVERS
 host 11.11.11.11
 host 111.111.111.2

ip access-list extended BLOCK
 deny   ip any host 111.111.111.2 time-range OFF
 deny   ip any host 111.111.111.4 time-range OFF
 deny   ip any host 111.111.111.6 time-range OFF
 deny   tcp any host 111.111.111.2 eq www 443 time-range WORK
 deny   tcp any host 111.111.111.4 eq www 443 time-range WORK
 deny   tcp any host 111.111.111.6 eq www 443 time-range WORK
 permit object-group DNS any object-group DNSSERVERS log-input
 permit ip any any

interface GigabitEthernet0/1
 ip access-group BLOCK in
interface GigabitEthernet0/0.115
 ip access-group BLOCK in
interface GigabitEthernet0/0.117
 ip access-group BLOCK in
```

Time-ranges give you the ability to control traffic based on time. You can specify an absolute or recurring time like shown in this task.

Object-groups are useful to group multiple elements of a certain time (addresses and services on IOS). Objects can be then used in the ACL to simplify their structure and overall management (whenever you update an Object it automatically updates an ACL).

Watch the order in the ACL – since all traffic is to be denied on November 11, it must come before a deny for Web traffic, which is only for 9am-5pm (meaning outside those hours you would normally match "permit ip any any" and the traffic would be allowed).

****** NOTE: Object-Groups don't work well/not work at all with time-ranges (don't mix the two features in a single ACL line) ******

## Verification

Being outside of a given time-range inactivates appropriate ACEs:

```
R8#sh clock
*08:30:27.237 UTC Thu Jul 17 2014

R8#sh time-range
time-range entry: OFF (inactive)
```

```
   absolute start 00:00 11 November 2014 end 23:59 11 November 2014
   used in: IP ACL entry
time-range entry: WORK (inactive)
   periodic weekdays 9:00 to 16:59
   used in: IP ACL entry

R2#telnet 111.111.111.4 80
Trying 111.111.111.4, 80 ...
% Connection refused by remote host

R8#clock set 09:15:00 17 jul 2014
R8#sh time-range
time-range entry: OFF (inactive)
   absolute start 00:00 11 November 2014 end 23:59 11 November 2014
   used in: IP ACL entry
time-range entry: WORK (active)
   periodic weekdays 9:00 to 16:59
   used in: IP ACL entry

R2#telnet 111.111.111.2 80
Trying 111.111.111.2, 80 ...
% Destination unreachable; gateway or host down

R8#sh access-l
Extended IP access list BLOCK
    10 deny ip any host 111.111.111.2 time-range OFF (inactive)
    11 deny ip any host 111.111.111.4 time-range OFF (inactive)
    12 deny ip any host 111.111.111.6 time-range OFF (inactive)
    20 deny tcp any host 111.111.111.2 eq www 443 time-range WORK (active) (2 matches)
    21 deny tcp any host 111.111.111.4 eq www 443 time-range WORK (active)
    22 deny tcp any host 111.111.111.6 eq www 443 time-range WORK (active)
    30 permit object-group DNS any object-group DNSSERVERS log-input
    40 permit ip any any (566 matches)

R8#clock set 14:00:00 11 nov 2014

R8#sh time-range
time-range entry: OFF (active)
   absolute start 00:00 11 November 2014 end 23:59 11 November 2014
   used in: IP ACL entry
   used in: IP ACL entry
   used in: IP ACL entry
time-range entry: WORK (active)
   periodic weekdays 9:00 to 16:59
   used in: IP ACL entry
   used in: IP ACL entry
   used in: IP ACL entry

R2#telnet 111.111.111.2 80
Trying 111.111.111.2, 80 ...
% Destination unreachable; gateway or host down

R2#telnet 111.111.111.2 25
Trying 111.111.111.2, 25 ...
% Destination unreachable; gateway or host down

R2#telnet 111.111.111.2 1234
Trying 111.111.111.2, 1234 ...
% Destination unreachable; gateway or host down

R8#sh access-l
Extended IP access list BLOCK
    10 deny ip any host 111.111.111.2 time-range OFF (active) (3 matches)
    11 deny ip any host 111.111.111.4 time-range OFF (active)
    12 deny ip any host 111.111.111.6 time-range OFF (active)
    20 deny tcp any host 111.111.111.2 eq www 443 time-range WORK (active) (2 matches)
    21 deny tcp any host 111.111.111.4 eq www 443 time-range WORK (active)
```

```
    22 deny tcp any host 111.111.111.6 eq www 443 time-range WORK (active)
    30 permit object-group DNS any object-group DNSSERVERS log-input
    40 permit ip any any (623 matches)

R8#clock set 10:00:00 15 nov 2014

R2#telnet 11.11.11.11 53
Trying 11.11.11.11, 53 ...
% Connection refused by remote host

R8#
Nov 15 10:00:09.279: %SEC-6-IPACCESSLOGP: list BLOCK permitted tcp 10.0.117.2(24380)
(GigabitEthernet0/2 001b.d517.ba89) -> 11.11.11.11(53), 1 packet

R8#sh access-l
Extended IP access list BLOCK
    10 deny ip any host 111.111.111.2 time-range OFF (inactive) (3 matches)
    11 deny ip any host 111.111.111.4 time-range OFF (inactive)
    12 deny ip any host 111.111.111.6 time-range OFF (inactive)
    20 deny tcp any host 111.111.111.2 eq www 443 time-range WORK (inactive) (2 matches)
    21 deny tcp any host 111.111.111.4 eq www 443 time-range WORK (inactive)
    22 deny tcp any host 111.111.111.6 eq www 443 time-range WORK (inactive)
    30 permit object-group DNS any object-group DNSSERVERS log-input (1 match)
    40 permit ip any any (683 matches)
```

## Task 36.8 Traffic Filtering – IP Fragments

- Modify an ACL from the previous task to block all IPv4 fragments regardless of the time/date.
- Block all IPv4 and IPv6 fragments coming to F0/1 on R2 – don't use an access-list to accomplish that.

## Detailed Solution

**R8**

```
ip access-list resequence BLOCK 15 10

ip access-list ext BLOCK
 5 deny ip any any fragments
```

**R2**

```
int f0/1
 ip virtual-reassembly in drop-fragments
 ipv6 virtual-reassembly in drop-fragments
```

Re-sequencing was not needed unless you did not have a place for a new entry in the beginning of the ACL. Here is more, just to show you the feature.

VFR (Virtual Fragmentation & Reassembly) can be also used to drop IP fragments as shown in this task.

## Verification

```
R2#ping 1.1.1.1 size 1500

Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms

R2#ping 1.1.1.1 size 1501
```

```
Type escape sequence to abort.
Sending 5, 1501-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

R8#sh access-list BLOCK
Extended IP access list BLOCK
    5 deny ip any any fragments (5 matches)
    15 deny ip any host 111.111.111.2 time-range OFF (inactive) (5 matches)
    25 deny ip any host 111.111.111.4 time-range OFF (inactive)
    35 deny ip any host 111.111.111.6 time-range OFF (inactive)
    45 deny tcp any host 111.111.111.2 eq www 443 time-range WORK (inactive) (2 matches)
    55 deny tcp any host 111.111.111.4 eq www 443 time-range WORK (inactive)
    65 deny tcp any host 111.111.111.6 eq www 443 time-range WORK (inactive)
    75 permit object-group DNS any object-group DNSSERVERS log-input (1 match)
    85 permit ip any any (4696 matches)
R8#

R8#ping 2.2.2.2 size 1500
Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R8#ping 2.2.2.2 size 1501
Type escape sequence to abort.
Sending 5, 1501-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

R8#ping 2::2 size 1500
Type escape sequence to abort.
Sending 5, 1500-byte ICMP Echos to 2::2, timeout is 2 seconds:
!!!!!

R8#ping 2::2 size 1501
Type escape sequence to abort.
Sending 5, 1501-byte ICMP Echos to 2::2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

R2#sh ip virtual-reassembly f0/1
FastEthernet0/1:
   Virtual Fragment Reassembly (VFR) is ENABLED [in]
   Concurrent reassemblies (max-reassemblies): 16
   Fragments per reassembly (max-fragments): 32
   Reassembly timeout (timeout): 3 seconds
   Drop fragments: ON

   Current reassembly count:0
   Current fragment count:0
   Total reassembly count:0
   Total reassembly timeout count:0

R2#sh ipv6 virtual-reassembly f0/1
%Interface FastEthernet0/1 [in]
   IPv6 configured concurrent reassemblies (max-reassemblies): 64
   IPv6 configured fragments per reassembly (max-fragments): 16
   IPv6 configured reassembly timeout (timeout): 3 seconds
   IPv6 configured drop fragments: ON

   IPv6 current reassembly count:0
   IPv6 current fragment count:0
   IPv6 total reassembly count:0
```

## Task 36.9 Traffic Filtering – Reflexive Access-Lists

- Users at VLAN 70 should be allowed through R2 to any destination when using WWW, Telnet, and SSH.
- Return traffic should be allowed dynamically. Dynamic entries should timeout after a minute.
- Only allow OSPF, ICMP, and Telnet inbound on F0/1.
- Use Reflexive Access-Lists.

### Detailed Solution

```
R2
ip access-list extended OUT_IN
 permit ospf any any
 permit icmp any any
 permit tcp any any eq telnet
 evaluate MIRROR

ip access-list extended RACL
 permit tcp 10.70.70.0 0.0.0.255 any eq www reflect MIRROR timeout 60
 permit tcp 10.70.70.0 0.0.0.255 any eq 22 reflect MIRROR timeout 60
 permit tcp 10.70.70.0 0.0.0.255 any eq telnet reflect MIRROR timeout 60
 permit ip any any

interface FastEthernet0/1
 ip access-group OUT_IN in
 ip access-group RACL out
```

Reflexive access lists allow IP packets to be filtered based on upper-layer session information. You can use reflexive access lists to permit IP traffic for sessions originating from within your network but to deny IP traffic for sessions originating from outside your network.

This feature can only be used with extended named IP access lists. You cannot define reflexive access lists with numbered or standard named IP access lists or with other protocol access lists.

Reflexive access lists contain only temporary entries; these entries are automatically created when a new IP session begins (for example, with an outbound packet), and the entries are removed when the session ends – we can control the “lifetime” of those dynamic sessions via “timeout”.

Remember that this feature does not work with some applications that use port numbers that change during a session (like FTP).

### Verification

```
CAT4#telnet 1.1.1.1 80
Trying 1.1.1.1, 80 ... Open
get /
HTTP/1.1 400 Bad Request
Date: Thu, 17 Jul 2014 13:08:40 GMT
Server: cisco-IOS
Connection: close
Accept-Ranges: none

400 Bad Request
[Connection to 1.1.1.1 closed by foreign host]

CAT4#telnet 1.1.1.1 22
Trying 1.1.1.1, 22 ...
% Destination unreachable; gateway or host down

CAT4#telnet 1.1.1.1 23
```

```
Trying 1.1.1.1 ... Open

User Access Verification

Password:

R1>

R2#sh access-l
Reflexive IP access list MIRROR
     permit tcp host 1.1.1.1 eq telnet host 10.70.70.140 eq 43485 (49 matches) (time left
57)
     permit tcp host 1.1.1.1 eq 22 host 10.70.70.140 eq 26477 (1 match) (time left 50)
Extended IP access list OUT_IN
    5 permit ospf any any (44 matches)
    10 permit icmp any any (1 match)
    20 permit tcp any any eq telnet
    30 evaluate MIRROR
Extended IP access list RACL
    60 permit tcp 10.70.70.0 0.0.0.255 any eq www reflect MIRROR (11 matches)
    70 permit tcp 10.70.70.0 0.0.0.255 any eq 22 reflect MIRROR (3 matches)
    80 permit tcp 10.70.70.0 0.0.0.255 any eq telnet reflect MIRROR (48 matches)
    90 permit ip any any
```

### Task 36.10 Dynamic (Lock & Key) Access-Lists

- You decided that traffic originating in VLAN 70 should be allowed through R2 only for authenticated users.
- Users will be authenticating using Telnet to 2.2.2.2 over port 3023.
- Sessions should not be idle for more than 2 minutes.
- Sessions longer than 30 minutes require re-authentication.
- A valid local user account for this task is “intuser” with password “cisco”.
- AAA should be already enabled on this device (from one of the previous tasks).

## Detailed Solution

**R2**

```
aaa authentication login DYNAUTHC local
aaa authorization exec DYNAUTHZ local

username intuser password 0 cisco
username intuser autocommand access-enable host timeout 2

ip access-list extended INSIDE_IN
 permit tcp any host 2.2.2.2 eq 3023
 dynamic DACL timeout 30 permit ip any any
 deny   ip any any

interface FastEthernet0/0
 ip access-group INSIDE_IN in

line vty 5
 authorization exec DYNAUTHZ
 login authentication DYNAUTHC
 rotary 23
```

With Dynamic (lock-and-key) ACLs, you can specify which users are permitted access to which source and destination hosts. These users must pass a user authentication process before they are permitted access to their designated hosts.

If you do not define an idle timeout with the "autocommand access-enable" command, you must define an absolute timeout for a dynamic ACE. You must define either an idle timeout or an absolute timeout - otherwise, the temporary access list entry will remain configured indefinitely on the interface (even after the user has terminated the session) until the entry is removed manually by an administrator (note that you can configure both idle and absolute timeouts if you wish.)

Another place where you could configure the "autocommand" is a VTY Line. Then everyone who successfully authenticates on that line gets access defined by the dynamic ACL entries.

## Verification

```
CAT4#telnet 1.1.1.1
Trying 1.1.1.1 ...
% Destination unreachable; gateway or host down

CAT4#telnet 2.2.2.2 3023
Trying 2.2.2.2, 3023 ... Open

User Access Verification

Username: intuser
Password:

[Connection to 2.2.2.2 closed by foreign host]

CAT4#telnet 1.1.1.1
Trying 1.1.1.1 ... Open

User Access Verification

Password:

R1>

R2#sh access-list INSIDE_IN
Extended IP access list INSIDE_IN
    10 permit tcp any host 2.2.2.2 eq 3023 (25 matches)
    20 Dynamic DACL permit ip any any
       permit ip host 10.70.70.140 any (20 matches) (time left 114)
    30 deny ip any any (14 matches)
```

### Task 36.11 Policy-Based Routing

- Telnet traffic sourced from R2's loopback0 destined to 3.3.3.3 should be blackholed on R8.
- Use PBR to accomplish that.

## Detailed Solution

**R8**

```
access-list 130 permit tcp host 2.2.2.2 host 3.3.3.3 eq telnet

route-map PBR permit 10
 match ip address 130
 set interface Null0

interface G0/0.117
 ip policy route-map PBR
```

Policy-Based routing can be also used as a security tool. Setting the outgoing interface to "Null0" effectively blackholes entry-matched traffic.

## Verification

```
R8#sh ip policy
Interface      Route map
Gi0/0.117          PBR

R2#telnet 3.3.3.3 /so loop0
Trying 3.3.3.3 ...
% Destination unreachable; gateway or host down

R8#deb ip policy
Policy routing debugging is on
Nov 15 16:33:43.962: IP: s=2.2.2.2 (GigabitEthernet0/0.117), d=3.3.3.3, len 44, FIB policy
match
Nov 15 16:33:43.962: IP: s=2.2.2.2 (GigabitEthernet0/0.117), d=3.3.3.3, len 44, PBR Counted
Nov 15 16:33:43.962: IP: s=2.2.2.2 (GigabitEthernet0/0.117), d=3.3.3.3, len 44, policy
match
Nov 15 16:33:43.962: IP: route map PBR, item 10, permit
Nov 15 16:33:43.962: IP: s=2.2.2.2 (GigabitEthernet0/0.117), d=3.3.3.3 (Null0), len 44,
policy routed
Nov 15 16:33:43.962: IP: GigabitEthernet0/0.117 to Null0 3.3.3.3

R8#sh route-map
route-map PBR, permit, sequence 10
  Match clauses:
    ip address (access-lists): 130
  Set clauses:
    interface Null0
  Policy routing matches: 2 packets, 120 bytes
```

### Task 36.12 Unicast Reverse Path Forwarding (URPF)

- Enable Loose Mode uRPF on R8.
- Packets received with unknown sources should be dropped.
- Don’t use a default route when uRPF decisions are made.
- An exception to this policy is packets coming from 192.168.1.0/24 – they should be allowed and logged.

## Detailed Solution

**R8**

```
access-list 1 permit 192.168.1.0 0.0.0.255 log

int g0/1
 ip verify unicast source reachable-via any 1
int g0/0.115
 ip verify unicast source reachable-via any 1
int g0/0.117
 ip verify unicast source reachable-via any 1
```

URPF works by enabling a router to verify the reachability of the source address in packets being forwarded. This capability can limit the appearance of spoofed addresses on a network. If the source IP address is not valid, the packet is discarded. There two modes of uRPF – Strict and Loose.

In strict mode (“reachable-via rx”), the packet must be received on the interface that the router would use to forward the return packet (to the source). This mode does not work in asymmetric environments.

In loose mode (“reachable-via any”), the source address must just appear in the routing table. As long as an interface used to reach the source is something else than Null0, the packet will be allowed. Note that a default route is considered as a valid route unless you enable it (“allow-default”).

It is also possible to make exceptions to the URPF process – by applying an ACL. Permit ACL entries define sources that the feature should ignore meaning that packets coming from those addresses will be always allowed.

## Verification

Create two loopbacks on R3 to test this configuration: Loopback 30 – 10.30.30.5/32 and Loopback 192 – 192.168.1.5/32. Then enable debug ICMP on R8 and observe if Echoes are processed or not (if not they are dropped by uRPF):

```
R8#debug ip icmp

R3#ping 10.10.11.8 so loop30

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.11.8, timeout is 2 seconds:
Packet sent with a source address of 10.30.30.5
.....
Success rate is 0 percent (0/5)

R8#sh ip int g0/0.115 | se verify
  IP verify source reachable-via ANY, ACL 1
   5 verification drops
   0 suppressed verification drops
   0 verification drop-rate
```

When you ping off Loopback192 the packets will be allowed (ACL exception). Of course we are not getting the replies since R9 does not know about 192.168.1.0/24:

```
R3#ping 10.10.11.8 so loop192

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.11.8, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.5
.....
Success rate is 0 percent (0/5)

R8#
Nov 15 16:53:39.078: ICMP: echo reply sent, src 10.10.11.8, dst 192.168.1.5, topology BASE,
dscp 0 topoid 0
Nov 15 16:53:39.078: ICMP: dst (10.10.11.8) host unreachable rcv from 10.10.11.9

R8#
Nov 15 16:58:59.030: %SEC-6-IPACCESSLOGNP: list 1 permitted 0 192.168.1.5 -> 10.10.11.8, 5
packets

R8#sh ip int g0/0.115 | se verify
  IP verify source reachable-via ANY, ACL 1
   5 verification drops
   5 suppressed verification drops
   0 verification drop-rate
```

### You have completed Lab 36

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 37: Security Part II

## Overview

Please look at the provided diagrams and read through the whole lab before you start. Read the directions very carefully to make sure you are doing what is being asked of you. This concept is very important when you take the CCIE lab administered by Cisco.

It is recommended to create your own diagram at the beginning of each lab so any potential information you find useful during your preparations can be reflected on this drawing, making it much easier when you step into the real lab.

Multiple topology drawings are available for this chapter.

## General Rules

You will need to pre-configure the network with the base configuration files.

---

***NOTE: Static/default routes are NOT allowed unless otherwise stated in the task***

***NOTE: You can use "cisco" for any password if other password was not explicitly mentioned in the question***

---

**Estimated Time to Complete: 2-3 hours**

## Pre-Lab Setup

Please login to your Security vRack at ProctorLabs.com and load the initial Configuration,
Verify basic L2/L3 connectivity. Use IP Addressing Table, Lab Diagram, and the Physical Topology.

This lab is intended to be used with online rack access provided by our partner Proctor Labs (www.proctorlabs.com). Connect to the terminal server and complete the configuration tasks as detailed below.

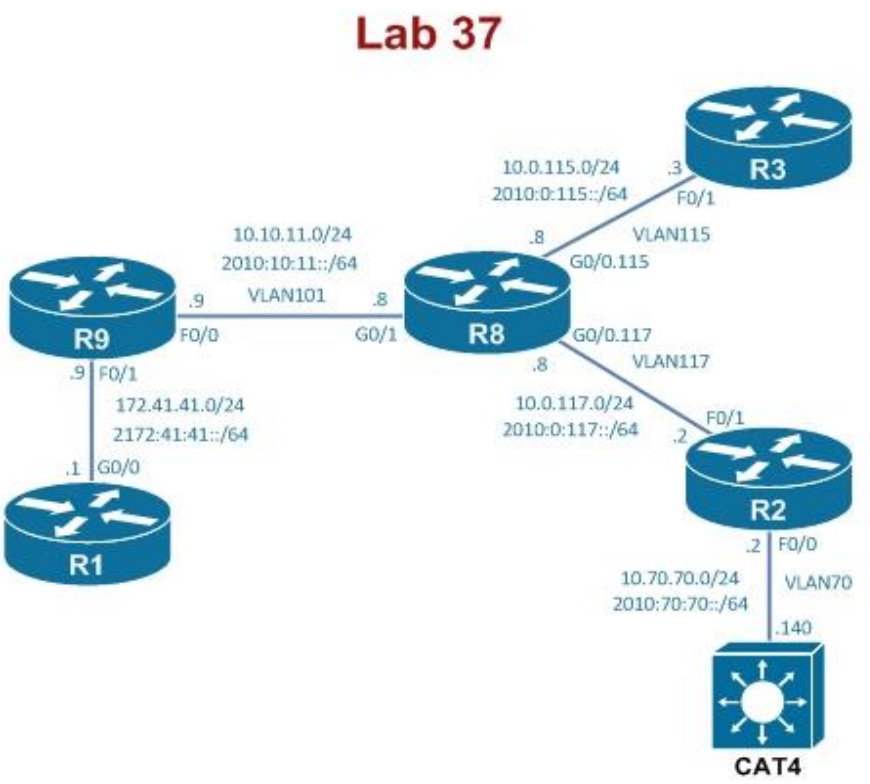
## Prerequisites

Load the initial configuration files before starting to work on the tasks.

| Device | Port | VLAN | IP Address |
|---|---|---|---|
| R1 | G0/0 | 41 | 172.41.41.1/24<br>2172:41:41::1/64 |
| | Loop0 | | 1.1.1.1/24<br>1::1/64 |
| R3 | F0/1 | 115 | 10.0.115.3/24<br>2010:0:115::3/64 |
| | Loop0 | | 3.3.3.3/24<br>3::3/64 |
| R2 | F0/0 | 70 | 10.70.70.2/24<br>2010:70:70::2/64 |
| | F0/1 | 117 | 10.0.117.2/24<br>2010:0:117::2/64 |
| | Loop0 | | 2.2.2.2/24<br>2::2/64 |
| R9 | F0/1 | 41 | 172.41.41.9/24<br>2172:41:41::9/64 |
| | F0/0 | 101 | 10.10.11.9/24<br>2010:10:11::9/64 |
| | Loop0 | | 9.9.9.9/24<br>9::9/64 |
| R8 | G0/1 | 101 | 10.10.11.8/24<br>2010:10:8::8/64 |
| | G0/0.115 | 115 | 10.0.115.8/24<br>2010:0:115::8/64 |
| | G0/0.117 | 117 | 10.0.117.8/24<br>2010:0:117::8/64 |
| | Loop0 | | 8.8.8.8/24<br>8::8/64 |
| CAT4 | SVI70 | 70 | 10.70.70.140/24 |




## Task 37.1 NBAR

- Using NBAR create and apply a policy outbound on R2's F0/1 to drop the Slammer worm traffic.
- The Slammer worm propagates over UDP port 1434 and its packets are exactly 404B long.
- In the same policy all HTTP packets with string "attack" in the URL should be dropped but only when traffic is going to a WWW server 8.8.8.8 (R8).
- The string should be case insensitive.

## Detailed Solution

**R2**

```
ip nbar custom SLAM1434 udp 1434

class-map match-all SLAMMER
 match protocol SLAM1434
 match packet length min 404 max 404

class-map match-all BAD_HTTP
 match protocol http url "*[Aa][Tt][Tt][Aa][Cc][Kk]*"
 match protocol http host "8.8.8.8"

policy-map NBARPOL
 class SLAMMER
  drop
 class BAD_HTTP
  drop

int F0/1
 service-policy output NBARPOL
```

Network-Based Application Recognition (NBAR) is a classification tool available under MQC class-maps via "match protocol" command. NBAR engine can recognize a wide variety of applications, including Web-based applications and client/server applications that dynamically assign TCP or UDP port numbers.

NBAR supports a wide range of network protocols, including some of these stateful protocols that were difficult to classify before NBAR like FTP, TFTP, SUN RPC, SQLNet, RealAudio and many more. Also remember that whenever HTTP is matched you have the ability to classify traffic based on application-layer information – URL, Host header or MIME types.

## Verification

Only when you try to download "attack" from address 8.8.8.8 a policy is matched and drop action is taken:

```
CAT4#copy http://8.8.8.8/test null0
Destination filename [null0]?
Accessing http://8.8.8.8/test...
%Error opening http://8.8.8.8/sd (No such file or directory)

CAT4#copy http://10.0.117.8/aTTacK null0
Destination filename [null0]?
Accessing http://10.0.117.8/aTTacK...
%Error opening http://10.0.117.8/aTTacK (No such file or directory)

CAT4#copy http://8.8.8.8/aTTacK null0
Destination filename [null0]?
Accessing http://8.8.8.8/aTTacK...
%Error opening http://8.8.8.8/aTTacK (I/O error)

R2#sh policy-map interface f0/1
 FastEthernet0/1

  Service-policy output: NBARPOL

    Class-map: SLAMMER (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: protocol SLAM1434
      Match: packet length min 404 max 404
      drop

    Class-map: BAD_HTTP (match-all)
      10 packets, 1556 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: protocol http url "*[Aa][Tt][Tt][Aa][Cc][Kk]*"
      Match: protocol http host "8.8.8.8"
      drop

    Class-map: class-default (match-any)
      124 packets, 10865 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: any

R2#sh ip nbar port-map SLAM1434
port-map SLAM1434                udp 1434
```

### Task 37.2 NBAR Next-Gen (NBAR2)

- Configure R8 to drop all terminal-related traffic except PCANYWHERE.
- Also implement a policy for peer-to-peer traffic :
  - All clear-text packets should be rate-limited to 200kbps.

    - All encrypted traffic should be dropped.
- Enable classification of IPv6 traffic that is carried over Teredo tunnels.
- Use a technology that examines IPv4 and IPv6 packets.
- Apply the policy outbound on G0/1.

## Detailed Solution

**R8**

```
ip cef
ipv6 cef

class-map match-all PCANYWHERE
 match proto attribute sub-category terminal pcanywhere

class-map match-all TERMINAL_APPS
 match protocol attribute sub-category terminal

class-map match-all ENC_P2P
 match protocol attribute p2p-technology p2p-tech-yes
 match protocol attribute encrypted encrypted-yes

class-map match-all CLEAR_P2P
 match protocol attribute p2p-technology p2p-tech-yes
 match protocol attribute encrypted encrypted-no

policy-map NBAR2POL
 class PCANYWHERE
 class TERMINAL_APPS
  drop
 class CLEAR_P2P
  police 200000
 class ENC_P2P
  drop

int g0/1
 service-policy output NBAR2POL

ip nbar classification tunneled-traffic teredo
```

Next Generation NBAR (NBAR2) is basically re-architectured old-style Network Based Application Recognition with improved classification engine, increased accuracy and way more signatures available. This feature is only available on ISR G2 platforms.

The main advantage of NBAR2 is advanced classification including the ability to match protocols running on top of IPv6. Packets are identified using a new SCE engine which allows classification of not only IPv4 and IPv6 packets but also the transition techniques. So things like ISATAP, Teredo, and this traffic can be now matched using this feature.

NBAR version 2 groups applications based on various attributes (`match protocol attribute`), and an attribute can be one of the following:

1. “Application-group” - a grouping of applications that are part of the same suite or brand. An example would be the “`Yahoo-Messenger-group`” keyword that actually matches Yahoo Messenger, Yahoo VoIP and VoIP over SIP traffic.
2. “Category” - a group of applications which support similar functionality from an end-user standpoint. For example `` `email' ``, `` `gaming' `` or `'file-sharing'`.
3. “Sub-category” - similar to category, but the classification of applications was done more from the networking standpoint. Examples : `'routing-protocol'`, `` `network-management' `` or `` `terminal' ``.

4. "Peer-to-Peer" - to match applications based on whether they were classified as P2P, not Peer-to-Peer and unassigned.
5. "Tunnel" is to match traffic that was classified as tunneled, not tunneled or unassigned.
6. "Encrypted" is to configure matching criteria based on encryption (yes, no, unassigned).

To figure out what protocols were classified as characterized by a particular attribute use the "`show ip nbar attribute`" command. To see all attributes of a particular protocol use "`show ip nbar protocol-attribute`".

Also don't forget that NBAR and NBAR2 require CEF to be turned on (`ip cef, ipv6 cef`).

## Verification

We're going to test with Telnet which is part of the Terminal applications :

```
R8#sh ip nbar protocol-attribute telnet
            Protocol Name :  telnet
                 category :  net-admin
             sub-category :  terminal
        application-group :  other
           p2p-technology :  p2p-tech-no
                   tunnel :  tunnel-no
                encrypted :  encrypted-no
```

We need to prepare R1 for incoming telnet connections (either configure a password or disable it like shown here) :

```
R1(config)#line vty 0 4
R1(config-line)#no login
```

Note that even we connected the remaining Telnet packets are dropped by R8 :

```
R2#telnet 1.1.1.1
Trying 1.1.1.1 ... Open

R1>
R2#telnet 1::1
Trying 1::1 ... Open

R1>

R8#sh policy-map int g0/0
 GigabitEthernet0/0

  Service-policy output: NBAR2POL

    Class-map: PCANYWHERE (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps
      Match: protocol pcanywhere

    Class-map: TERMINAL_APPS (match-all)
      45 packets, 2994 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: protocol attribute sub-category terminal
      drop

    Class-map: CLEAR_P2P (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: protocol attribute p2p-technology p2p-tech-yes
      Match: protocol attribute encrypted encrypted-no
      police:
          cir 200000 bps, bc 6250 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
```

```
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0000 bps, exceeded 0000 bps

    Class-map: ENC_P2P (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: protocol attribute p2p-technology p2p-tech-yes
      Match: protocol attribute encrypted encrypted-yes
      drop

    Class-map: class-default (match-any)
      228 packets, 22948 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

### Task 37.3 NBAR Protocol Discovery

- Enable NBAR Protocol Discovery on R9's F0/0.
- Make sure statistics are obtained for IPv4 and IPv6 traffic.

## Detailed Solution

**R9**

```
interface FastEthernet0/0
 ip nbar protocol-discovery
```

NBAR includes a feature called Protocol Discovery. Protocol Discovery provides an easy way of discovering the application protocols that are operating on an interface so that appropriate QoS features can be applied. With Protocol Discovery, you can discover any protocol traffic that is supported by NBAR/NBAR2 and obtain statistics (number of input/output packets and bit rates) that are associated with that protocol. The statistics can then be used when you later define classes and traffic policies.

Pay attention to the command syntax – whenever you include "ipv4" or "ipv6" keyword you only enable the feature for a particular protocol's traffic (in our case since we want to collect statistics for both protocols we skip these options).

## Verification

```
R8#copy http://1.1.1.1/file.txt null0
Destination filename [null0]?
Accessing http://1.1.1.1/file.txt...
%Error opening http://1.1.1.1/file.txt (No such file or directory)

R8#ping 1::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms


R9#sh ip nbar protocol-discovery

 FastEthernet0/0

 Last clearing of "show ip nbar protocol-discovery" counters 00:05:28


                            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)
----------------------- ------------------------ ------------------------
ospf                    37                       113
                        4022                     11094
                        0                        0
                        0                        0
http                    4                        4
                        359                      353
                        0                        0
                        0                        0
ipv6-icmp               9                        9
                        472                      222
                        0                        0
                        0                        0
unknown                 14                       10
                        840                      689
                        0                        0
                        0                        0
Total                   59                       129
                        5693                     12358
                        0                        0
                        0                        0
```

## Task 37.4 TCP Intercept

- There are multiple servers in VLAN 70 hosting various TCP-based applications.
- Several DoS attacks took place recently targeted at those devices.
- Configure R2 to intercept TCP connection requests to this segment.
- If the total number of half-open connections reaches 400, R2 should start randomly dropping them.
- This should cease if the number of half-open sessions falls below 200.
- Make sure router stops managing the sessions after 40 minutes of inactivity.

## Detailed Solution

**R2**

```
access-list 150 permit tcp any 10.70.70.0 0.0.0.255

ip tcp intercept list 150
ip tcp intercept connection-timeout 2400
ip tcp intercept max-incomplete low 200 high 400
ip tcp intercept drop-mode random
ip tcp intercept mode intercept
```

The TCP Intercept feature implements software to protect TCP servers from TCP SYN-flooding attacks, which are a type of denial-of-service attack.

A SYN-flooding attack occurs when a hacker floods a server with a barrage of requests for connection. Because these messages have unreachable return addresses, the connections cannot be established. The resulting volume of unresolved open connections eventually overwhelms the server and can cause it to deny service to valid requests, thereby preventing legitimate users from connecting to a web site, accessing e-mail, using FTP service, and so on.

The TCP intercept can operate in either active intercept mode or passive watch mode. The default is intercept mode. In intercept mode, the software actively intercepts each incoming connection request (SYN) and responds on behalf of the server with an SYN-ACK, then waits for an ACK from the

client. When that ACK is received, the original SYN is sent to the server and the software performs a three-way handshake with the server. When this is complete, the two half-connections are joined. Be aware that this mode can eventually move the attack burden from the victim(s) to the router and is generally not recommended for production networks.

## Verification

Enable and observe the debug – since packets are sent from an existing and reachable address the router will treat the session as fully established after the handshake is finished:

```
R2#debug ip tcp intercept

R1#telnet 10.70.70.140
Trying 10.70.70.140 ... Open

Password required, but none set

R2#
*Jul 24 18:09:15.058: INTERCEPT: new connection (172.41.41.1:52027 SYN -> 10.70.70.140:23)
*Jul 24 18:09:15.062: INTERCEPT(*): (172.41.41.1:52027 <- ACK+SYN 10.70.70.140:23)
*Jul 24 18:09:15.062: INTERCEPT: 1st half of connection is established (172.41.41.1:52027
ACK -> 10.70.70.140:23)
*Jul 24 18:09:15.066: INTERCEPT(*): (172.41.41.1:52027 SYN -> 10.70.70.140:23)
*Jul 24 18:09:15.066: INTERCEPT: 2nd half of connection established  (172.41.41.1:52027 <-
ACK+SYN 10.70.70.140:23)
*Jul 24 18:09:15.066: INTERCEPT(*): (172.41.41.1:52027 ACK -> 10.70.70.140:23)
*Jul 24 18:09:15.070: INTERCEPT(*): (172.41.41.1:52027 <- WINDOW 10.70.70.140:23)

R2#sh tcp intercept connections
Incomplete:
Client                  Server                  State    Create   Timeout  Mode

Established:
Client                  Server                  State    Create   Timeout  Mode
172.41.41.1:52027       10.70.70.140:23         ESTAB    00:00:02 00:00:04 I

R2#sh tcp intercept statistics
Intercepting new connections using access-list 150
0 incomplete, 1 established connections (total 1)
1 connection requests per minute
```

### Task 37.5 TCP Intercept Passive Mode

- There are some other TCP servers that were recently attacked with large amount of spoofed SYN requests (3.3.3.0/24 segment).
- R3 should be configured to send a reset to the server under attack but it should not participate in the handshake.
- The reset segment should be sent if a session does not establish within 20 seconds.
- If a number of connection attempts within the last minute exceed 100, or when a total number of half-open sessions exceed 300, the sessions should be reset faster - after 10 seconds.
- If a FIN exchange or RST packet was seen for a session it should be dropped after 7 seconds.

## Detailed Solution

**R3**

```
access-list 150 permit tcp any 3.3.3.0 0.0.0.255

ip tcp intercept list 150
ip tcp intercept watch-timeout 20
ip tcp intercept max-incomplete low 300 high 300
```

```
ip tcp intercept one-minute low 100 high 100
ip tcp intercept mode watch
ip tcp intercept finrst-timeout 7
```

In watch mode, connection requests are allowed to pass through the router to the server but are watched until they become established. If they fail to become established within 30 seconds (configurable with the "ip tcp intercept watch-timeout" command), the software sends a Reset to the server to clear up its state.

If either one of the high limits is exceeded the software automatically reduces the watch timeout by half.

## Verification

You will not be able to test this task since TCP Intercept is designed to track transit packets.

### Task 37.6 Packet Logging

- Configure R1 to send all logged messages to a Syslog server located at 10.70.70.100.
- Use facility type local1.
- Use detailed time stamps for log and debugs including local time zone, and the time of day.
- Logs should be also sent to a buffer – allocate 16384B of memory for this purpose.
- Log messages should be sent with source of 1.1.1.1 and they should be rate-limited to 200 per second except for Sev 1 messages

## Detailed Solution

**R1**

```
logging host 10.70.70.100
logging facility local1
service timestamps log datetime local show-timezone
service timestamps debug datetime local show-timezone
logging buffered 16384
logging source-int loop0
logging trap debugging
logging rate-limit 200 except alerts
logging on
```

Logging is used to record and store device-generated messages. Once the logging process receives a system even or a message it will then send it to a certain destination[s], depending on how we configured logging (console, Syslog server, internal buffer, VTY lines, SNMP Server).

Which messages are sent depends on Logging Severity setting (also known as logging levels). When you set the level to 7 (lowest), the router also automatically includes messages associated with levels above (6, 5, 4 down to 0). So to send all logs you want to enable logging at the debugging level.

There are a couple of options to can be configured for logging and the easiest way to figure out the necessary syntax (or just see what you can do) is to use the "logging ?"

## Verification

```
R1#show logging
Syslog logging: enabled (0 messages dropped, 4 messages rate-limited, 0 flushes, 0
overruns, xml disabled, filtering disabled)

No Active Message Discriminator.
```

```
No Inactive Message Discriminator.

    Console logging: level debugging, 41 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 2 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 debugging, 44 message lines logged
        Logging to 10.70.70.100  (udp port 514, audit disabled,
              link up),
              2 message lines logged,
              0 message lines rate-limited,
              0 message lines dropped-by-MD,
              xml disabled, sequence number disabled
              filtering disabled

Log Buffer (16384 bytes):

*Jul 24 18:32:20 UTC: %SYS-5-CONFIG_I: Configured from console by console
*Jul 24 18:32:21 UTC: %SYS-6-LOGGINGHOST_STARTSTOP: Logging to host 10.70.70.100 port 514
started - CLI initiated
```

### Task 37.7 VLAN Filtering

- Configure a VACL on CAT4 to deny the following traffic within VLAN 117 :
  - TCP packets destined to 2.2.2.2 over port 3023.
  - All DNS traffic.
  - Non-IP frames destined to 00:04:cc:1e:12:34.
- Log dropped IP packets – set the log table size to 300 flows.
- Ensure that a log message is seen for every dropped packet.

## Detailed Solution

**CAT4**

```
access-list 140 permit tcp any host 2.2.2.2 eq 3023
access-list 140 permit tcp any any eq domain
access-list 140 permit udp any any eq domain

mac access-list extended MACL
 permit any host 0004.cc1e.1234

vlan access-map VACL 10
 match ip address 140
 action drop log
vlan access-map VACL 30
 match mac address MACL
 action drop
vlan access-map VACL 100
 action forward

vlan filter VACL vlan-list 117

vlan access-log maxflow 300
vlan access-log threshold 1
```

VLAN Access-Lists (VACLs) provide the ability to control IPv4 (release 15.0(1)SY1 and later releases also support IPv6) and non-IP traffic at the switch level on a per-VLAN basis. Packets can either enter a VLAN through a Layer 2 port or through a Layer 3 port after being routed – and VACLs can be used to selectively permit or deny traffic within a VLAN.

Each VLAN access map can consist of one or more map entries (sequenced) where each entry/sequence has a match clause and an action clause. The match clause specifies IP (IP traffic) or MAC ACLs (non-IP traffic) for traffic filtering and the action clause specifies the action to be taken when a match occurs. When a flow matches a permit ACL entry, the associated action is taken and the flow is not checked against the remaining sequences. When a flow matches a deny ACL entry, it will be checked against the next ACL in the same sequence or the next entry will be evaluated.

Be aware that if at least one ACL was configured (as a matching criteria) for a certain packet type, if a packet does not match any ACL entry in a map for that traffic type, the packet is denied. For example, if you are matching packets with IP ACLs and no specific match was found, IP packets will be denied – but not non-IP packets. Non-IP packets would be only affected if MAC ACL was used (at least once) in that VACL.

## Verification

Prepare R2 for incoming telnet connections over port 3023 to test our first requirement (remove this configuration after you finished testing):

```
R2(config)#line vty 0
R2(config-line)#no login
R2(config-line)#rotary 23

R8#telnet 10.0.117.2 3023
Trying 10.0.117.2, 3023 ... Open

R2>

R8#telnet 2.2.2.2 3023
Trying 2.2.2.2, 3023 ...
% Connection timed out; remote host not responding

R8#telnet 2.2.2.2 3023
Trying 2.2.2.2, 3023 ...
% Connection timed out; remote host not responding

CAT4#
*Mar  1 20:39:36.055: %VLMAPLOG-6-L4: vlan 117 (port Fa0/8) denied tcp 10.0.117.8(32951) ->
2.2.2.2(3023), 1 packet
*Mar  1 20:39:38.060: %VLMAPLOG-6-L4: vlan 117 (port Fa/0/8) denied tcp 10.0.117.8(32951) -
> 2.2.2.2(3023), 1 packet

CAT4#sh vlan access-log statistics
VACL Logging Statistics:
        total packets          :4
        logged                 :4
        dropped                :0
        buffered               :0
Dropped Packets Statistics:
        no packet buffer       :0
        hash queue full        :0
        flow table full        :0
Misc Information:
        free packet buffers    :8192
        log messages sent      :4
        flow table size        :2

CAT4#sh vlan access-log config
```

```
VACL Logging Configuration:
        max log table size     :300
        log threshold          :1
        rate limiter           :NOT ENABLED

CAT4#sh vlan access-log flow tcp any
 any
Matched flows:
  id  prot    src_ip        dst_ip        sport dport vlan    port      count       total
lastlog
-------------------------------------------------------------------------------------------
----------
   1   6     10.0.117.8         2.2.2.2 59889  3023  117   Fa0/8      0            2
20:40:14.475
   2   6     10.0.117.8         2.2.2.2 32951  3023  117   Fa0/8      0            2
20:39:38.060
```

Now quickly test DNS :

```
R8(config)#ip name-server 2.2.2.2
R8(config)#ip domain-lookup
R8#ping cisco.com
Translating "cisco.com"...domain server (2.2.2.2)

CAT4#
*Mar  1 20:45:30.373: %VLMAPLOG-6-L4: vlan 117 (port Fa0/8) denied udp 10.0.117.8(62079) ->
2.2.2.2(53), 1 packet
*Mar  1 20:45:33.376: %VLMAPLOG-6-L4: vlan 117 (port Fa0/8) denied udp 10.0.117.8(62079) ->
2.2.2.2(53), 1 packet
*Mar  1 20:45:36.379: %VLMAPLOG-6-L4: vlan 117 (port Fa0/8) denied udp 10.0.117.8(62079) ->
2.2.2.2(53), 1 packet
*Mar  1 20:45:39.374: %VLMAPLOG-6-L4: vlan 117 (port Fa0/8) denied udp 10.0.117.8(62079) ->
2.2.2.2(53), 1 packet
```

Now the second part. First thing let's see if we have non-IP connectivity between R2 and R8 :

```
R2#clear arp
R2#ping 10.0.117.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.117.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R2#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.0.117.2              -   001b.d517.ba89  ARPA   FastEthernet0/1
Internet  10.0.117.8              0   c84c.751f.ddc2  ARPA   FastEthernet0/1
Internet  10.70.70.2              -   001b.d517.ba88  ARPA   FastEthernet0/0
Internet  10.70.70.140            0   0007.7dbc.c6c1  ARPA   FastEthernet0/0
```

We will now change MAC on R2 to be 0004.cc1e.1234. This means that when R2 sends ARP request for R8's IP, R8 will reply with an ARP frame destined to this address:

```
R2(config)#int f0/1
R2(config-if)#mac-address 0004.cc1e.1234

R2# clear arp
R2# clear arp
R2#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.0.117.2              -   0004.cc1e.1234  ARPA   FastEthernet0/1
Internet  10.70.70.2              -   001b.d517.ba88  ARPA   FastEthernet0/0
Internet  10.70.70.140            5   0007.7dbc.c6c1  ARPA   FastEthernet0/0

R2#ping 10.0.117.8 rep 2

Type escape sequence to abort.
Sending 2, 100-byte ICMP Echos to 10.0.117.8, timeout is 2 seconds:
..
```

```
Success rate is 0 percent (0/2)
```

Note that even R8 is sending the replies back R2 does not get anything :

```
R8#debug arp
*Jul 25 10:30:44.882: IP ARP: rcvd req src 10.0.117.2 0004.cc1e.1234, dst 10.0.117.8
GigabitEthernet0/0.117
*Jul 25 10:30:44.882: IP ARP: sent rep src 10.0.117.8 c84c.751f.ddc2,
                 dst 10.0.117.2 0004.cc1e.1234 GigabitEthernet0/0.117

R2#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.0.117.2              -   0004.cc1e.1234  ARPA   FastEthernet0/1
Internet  10.0.117.8              0   Incomplete      ARPA
Internet  10.70.70.2              -   001b.d517.ba88  ARPA   FastEthernet0/0
Internet  10.70.70.140            5   0007.7dbc.c6c1  ARPA   FastEthernet0/0

CAT4#sh access-lists hardware counters  | in Drop:
     All  Drop:                       frame count: 2
     All  IPv6 Drop:                  frame count: 0
     All  Drop:                       frame count: 0
     All  IPv6 Drop:                  frame count: 0
```

## Task 37.8 Port Security

- Enable Port Security on CAT2.
- Make sure that port connected to R1 will accept frames with R1's MAC, but don't configure address statically.
- On the same interface also allow frames coming from 0000.2222.3333.
- If a violation occurs frames should be dropped, and a Syslog and SNMP traps should be generated. The switch should try to automatically recover from a violation every 50 seconds.
- Anytime the switch reboots it should not affect the Port Security table.

## Detailed Solution

### CAT2

```
int f0/1
 switchport port-security
 switchport port-security maximum 2
 switchport port-security violation restrict
 switchport port-security mac-address sticky
 switchport port-security mac-address 0000.2222.3333 vlan access

errdisable recovery cause psecure-violation
errdisable recovery interval 50
```

Port Security is a L2 security feature designed to enable protection against CAM Flooding and MAC spoofing attacks. With Port Security we have the ability to specify MAC addresses allowed through the port, and we can also limit the total number of addresses to be accepted (e.g. like you are connecting via a Hub).

MAC Addresses can be learned in three different ways: dynamically, statically or via a Sticky option. This feature enables the switch to learn addresses dynamically, but once they are learned they will be automatically added to the running-config. This means that when you save your configuration they will not disappear even after a reboot. Regular dynamic addresses disappear from the Port Security table when the link goes down, not to mention a reload.

A security violation occurs under two situations:

1. When a total number of MAC addresses seen on a port exceeded the configured threshold.

2. When an already secured MAC (learned dynamically or configured statically) was found on another interface within the same VLAN.

The default action for a violation is to shutdown (err-disable) the interface, increase the violation counter and generate a Syslog and SNMP Trap. Protect will silently drop the frames where Restrict will drop them but in addition also increase the violation counter, and generate a Sylog and SNMP Trap.

With the default violation action (shutdown) to re-enable the port you can either do this manually (shut, then no shut) or you can use the Error-Disable mechanism for “psecure-violation”.

Be aware that Port Security can be also configured on interfaces that carry traffic from multiple VLANs (trunks or ports connected to IP Phones) – you have the ability to specify a VLAN number along with a MAC address.

## Verification

```
CAT2#sh port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
                (Count)       (Count)          (Count)
---------------------------------------------------------------------------
      Fa0/1              2            2                  0         Restrict
---------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 1
Max Addresses limit in System (excluding one mac per port) : 6144


CAT2#sh port-security interface f0/1
Port Security              : Enabled
Port Status                : Secure-up
Violation Mode             : Restrict
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 2
Total MAC Addresses        : 2
Configured MAC Addresses   : 1
Sticky MAC Addresses       : 1
Last Source Address:Vlan   : 001b.d50f.f371:41
Security Violation Count   : 0

CAT2#sh port-security interface f0/1 address
          Secure Mac Address Table
-----------------------------------------------------------------------------
Vlan    Mac Address       Type                        Ports   Remaining Age
                                                                 (mins)
----    -----------       ----                        -----   -------------
  41    0000.2222.3333    SecureConfigured            Fa0/1         -
  41    001b.d50f.f371    SecureSticky                Fa0/1         -
-----------------------------------------------------------------------------
Total Addresses: 2
```

Now change MAC to be something else from what we have in the table. This should trigger a violation which means no connectivity through the port:

```
R1(config)#int G0/0
R1(config-if)#mac-address 0000.9999.9999

R1#ping 172.41.41.9

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.41.41.9, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

```
CAT2#
*Mar  1 01:14:15.063: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred,
caused by MAC address 0000.9999.9999 on port FastEthernet0/4.

CAT2#sh port-security int f0/1

Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
               (Count)       (Count)          (Count)
-----------------------------------------------------------------------------
      Fa0/1             2           2                 34         Restrict
-----------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 1
Max Addresses limit in System (excluding one mac per port) : 6144
```

Once you change back to its original value or other address that we statically allowed we should regain connectivity :

```
R1(config)#int G0/0
R1(config-if)#mac-address 0000.2222.3333

R1#ping 172.41.41.9

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.41.41.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/24 ms

CAT2#sh errdisable recovery | in psecure
psecure-violation           Enabled
```

**You have completed Lab 37**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 38: Security Part III

## Overview

Please look at the provided diagrams, and read through the whole lab before you start. Read the directions very carefully to make sure you are doing what is being asked of you. This concept is very important when you take the CCIE lab administered by Cisco.

It is recommended to create your own diagram at the beginning of each lab so any potential information you find useful during your preparations can be reflected on this drawing, making it much easier when you step into the real lab.

Multiple topology drawings are available for this chapter.

## General Rules

You will need to pre-configure the network with the base configuration files.

---

***NOTE: Static/default routes are NOT allowed unless otherwise stated in the task***

***NOTE: You can use "cisco" for any password if other password was not explicitly mentioned in the question***

---

**Estimated Time to Complete: 2-3 hours**

## Pre-Lab Setup

Please login to your Security vRack at ProctorLabs.com and load the initial Configuration.
Verify basic L2/L3 connectivity. Use IP Addressing Table, Lab Diagram, and the Physical Topology.

This lab is intended to be used with online rack access provided by our partner Proctor Labs (www.proctorlabs.com). Connect to the terminal server and complete the configuration tasks as detailed below.

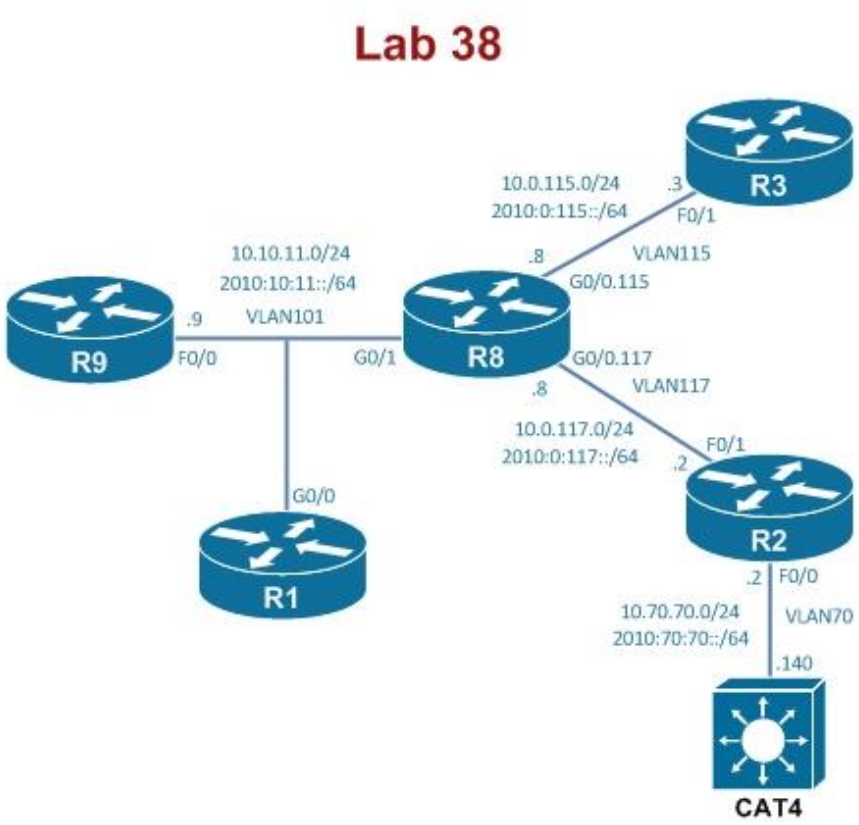
## Prerequisites

Load the initial configuration files before starting to work on the tasks.

| Device | Port | VLAN | IP Address |
|---|---|---|---|
| R1 | G0/0 | 101 | 10.10.11.1/24<br>2010:10:11::1/64 |
| | Loop0 | | 1.1.1.1/24<br>1::1/64 |
| R3 | F0/1 | 115 | 10.0.115.3/24<br>2010:0:115::3/64 |
| | Loop0 | | 3.3.3.3/24<br>3::3/64 |
| R2 | F0/0 | 70 | 10.70.70.2/24<br>2010:70:70::2/64 |
| | F0/1 | 117 | 10.0.117.2/24<br>2010:0:117::2/64 |
| | Loop0 | | 2.2.2.2/24<br>2::2/64 |
| R9 | F0/0 | 101 | 10.10.11.9/24<br>2010:10:11::9/64 |
| | Loop0 | | 9.9.9.9/24<br>9::9/64 |
| R8 | G0/1 | 101 | 10.10.11.8/24<br>2010:10:11::8/64 |
| | G0/0.115 | 115 | 10.0.115.8/24<br>2010:0:115::8/64 |
| | G0/0.117 | 117 | 10.0.117.8/24<br>2010:0:117::8/64 |
| | Loop0 | | 8.8.8.8/24<br>8::8/64 |
| CAT4 | SVI70 | 70 | 10.70.70.140/24 |




## Task 38.1 DHCP Snooping

- Secure DHCP communication in VLAN 101 using DHCP Snooping.
- Configure R9 to act as a DHCP Server in this VLAN.
- Make sure R1 and R8 obtain their address dynamically.
- Rate-limit client DHCP traffic to 15pps.
- Ensure that snooping bindings don't disappear after a reload. The lease times should be accurate - configure & use R9 as a NTP server.

## Detailed Solution

**R9**

```
ip dhcp excluded-address 10.10.11.9

ip dhcp pool VLAN101
 network 10.10.11.0 255.255.255.0

ntp master
```

**R1**

```
int g0/0
 ip add dhcp
```

**R8**

```
int g0/1
 ip add dhcp
```

**CAT2**

```
interface Vlan101
 ip address 10.10.11.120 255.255.255.0

ntp server 10.10.11.9
```

```
ip dhcp snooping vlan 101
ip dhcp snooping

no ip dhcp snooping information option

ip dhcp snooping database flash:sn.db

interface FastEthernet0/1
 ip dhcp snooping limit rate 15

interface FastEthernet0/8
 ip dhcp snooping limit rate 15

int f0/9
 ip dhcp snooping trust
```

DHCP snooping is a L2 security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table.

When you enable the feature on a VLAN (or VLANs), by default all interfaces that belong to that VLAN (VLANs) are considered to be untrusted ports. This means that only client-related DHCP messages will be accepted on those. That's why at least one interface must be always designated as trusted (where a DHCP Server connects) – otherwise clients would never be able to obtain IP addresses. Note that if a server was connected to another switch, you would have to trust a trunk.

Snooping binding table stores information about the assigned addresses and client identifiers so that subsequent DHCP packets can be verified. This information is also used by other features (which we will examine in the next two tasks) to secure ARP and data communication.

When you enable DHCP Snooping it forces a switch to modify all DHCP Discovery messages by default – the switch will insert a so-called DHCP Option 82 with GIADDR field set to 0.0.0.0. This option is normally used by DHCP Relay Agents that set GIADDR to their own address, but since the switch inserts 0.0.0.0 it makes an IOS DHCP Server to drop the packet instead of processing it. That's why we had to disable Option 82 insertion ("no ip dhcp snooping information option").

To keep the bindings when the switch reloads, you must use the DHCP snooping database. Enabling NTP ensures that the lease time in the database is accurate.

## Verification

```
R9#sh ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/              Lease expiration        Type
                    Hardware address/
                    User name
10.10.11.1          0063.6973.636f.2d63.    Jul 27 2014 12:03 PM    Automatic
                    3834.632e.3735.3166.
                    2e64.6463.302d.4769.
                    302f.30
10.10.11.2          0063.6973.636f.2d30.    Jul 27 2014 12:04 PM    Automatic
                    3031.622e.6435.3066.
                    2e66.3337.312d.4661.
                    302f.31

CAT2#sh ip dhcp snooping
Switch DHCP snooping is enabled
DHCP snooping is configured on following VLANs:
101
DHCP snooping is operational on following VLANs:
101
```

```
DHCP snooping is configured on the following L3 Interfaces:

Insertion of option 82 is disabled
   circuit-id default format: vlan-mod-port
   remote-id: 001b.d4c1.5400 (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)
-----------------------    -------    ------------    ----------------
FastEthernet0/1            no         no              15
  Custom circuit-ids:
FastEthernet0/9            yes        yes             unlimited
  Custom circuit-ids:
FastEthernet0/8            no         no              15

CAT2#sh ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  --------------------
00:1B:D5:0F:F3:71   10.10.11.2       86025       dhcp-snooping   101   FastEthernet0/1
C8:4C:75:1F:DD:C0   10.10.11.1       85997       dhcp-snooping   101   FastEthernet0/8
Total number of bindings: 2

CAT2#sh ip dhcp snooping database
Agent URL : flash:sn.db
Write delay Timer : 300 seconds
Abort Timer : 300 seconds

Agent Running : No
Delay Timer Expiry : 270 (00:04:30)
Abort Timer Expiry : Not Running

Last Succeded Time : 13:25:52 UTC Sat Jul 26 2014
Last Failed Time : 00:41:23 UTC Mon Mar 1 1993
Last Failed Reason : Error reading the remote database.

Total Attempts       :        2   Startup Failures :        0
Successful Transfers :        1   Failed Transfers :        1
Successful Reads     :        0   Failed Reads     :        1
Successful Writes    :        1   Failed Writes    :        0
Media Failures       :        1

CAT2#sh ntp status
Clock is synchronized, stratum 9, reference is 10.10.11.9
nominal freq is 119.2092 Hz, actual freq is 119.2092 Hz, precision is 2**18
reference time is D77E2A8F.F0D7C9FB (13:24:31.940 UTC Sat Jul 26 2014)
clock offset is 3.5137 msec, root delay is 1.91 msec
root dispersion is 6.39 msec, peer dispersion is 2.61 msec
```

## Task 38.2 Dynamic ARP Inspection

- Prevent ARP Man-in-the-Middle attacks in VLAN 101.
- Routers R1, R9, and R8 should be able to successfully communicate.
- ARP packets generated by those devices should be logged.
- Enable source and destination MAC address validation.
- Rate-limit ARP traffic on port connected to R1 to 10 pps. Set the burst interval to 3 seconds.
- Disable ARP Inspection on trunks to other switches.
- You are not allowed to modify the DHCP Snooping database in this task.

## Detailed Solution

**CAT2**

```
arp access-list ARP_R9
 permit ip host 10.10.11.9 mac host 30e4.dbce.8490

ip arp inspection vlan 101 logging acl-match matchlog
ip arp inspection vlan 101 logging dhcp-bindings all
ip arp inspection validate src-mac dst-mac
ip arp inspection filter ARP_R9 vlan  101
ip arp inspection vlan 101

interface FastEthernet0/1
 ip arp inspection limit rate 10 burst interval 3

int range f0/19 - 24
 ip arp inspection trust
```

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a DHCP Snooping table (optionally also based on the content of an ARP ACL which in our case was created for R9 that acts as a DHCP server here). If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces (all by default), the switch forwards the packet only if it is valid – if it matches one of our dynamic/static entries.

We were explicitly asked to log ARP packets generated by R1, R9, and R8. This means that you cannot use "ip arp inspection trust" on the port connected to R9 (which uses a static address) because you would not be able to see the logs for R9. That's why an ARP ACL was used instead, and logging was enabled for both – DHCP Snooping table & ARP access-list entries.

In our lab the trunks were made trusted just to show you the command. This normally would not be required since there is just one switch where Dynamic ARP Inspection was enabled, and it has the correct bindings in its DHCP snooping table. This command would be useful in certain scenarios where a DHCP Server is on a different switch than the clients.

## Verification

Clear ARP few times on R1 and R9 then go to CAT2 and observe the logs:

```
CAT2#
Jul 26 13:36:48.092: %SW_DAI-6-DHCP_SNOOPING_PERMIT: 2 ARPs (Res) on Fa0/8, vlan
101.([c84c.751f.ddc0/10.10.11.1/30e4.dbce.8490/10.10.11.9/13:36:47 UTC Sat Jul 26 2014])
Jul 26 13:36:48.092: %SW_DAI-6-DHCP_SNOOPING_PERMIT: 2 ARPs (Res) on Fa0/1, vlan
101.([001b.d50f.f371/10.10.11.2/30e4.dbce.8490/10.10.11.9/13:36:47 UTC Sat Jul 26 2014])
Jul 26 13:36:49.098: %SW_DAI-6-DHCP_SNOOPING_PERMIT: 2 ARPs (Res) on Fa0/8, vlan
101.([c84c.751f.ddc0/10.10.11.1/30e4.dbce.8490/10.10.11.9/13:36:48 UTC Sat Jul 26 2014])
Jul 26 13:36:49.098: %SW_DAI-6-DHCP_SNOOPING_PERMIT: 2 ARPs (Res) on Fa0/1, vlan
101.([001b.d50f.f371/10.10.11.2/30e4.dbce.8490/10.10.11.9/13:36:48 UTC Sat Jul 26 2014])

CAT2#
Jul 26 13:37:15.272: %SW_DAI-6-DHCP_SNOOPING_PERMIT: 2 ARPs (Res) on Fa0/1, vlan
101.([001b.d50f.f371/10.10.11.2/ffff.ffff.ffff/10.10.11.2/13:37:15 UTC Sat Jul 26 2014])
Jul 26 13:37:15.272: %SW_DAI-6-DHCP_SNOOPING_PERMIT: 2 ARPs (Req) on Fa0/1, vlan
101.([001b.d50f.f371/10.10.11.2/c84c.751f.ddc0/10.10.11.1/13:37:15 UTC Sat Jul 26 2014])

CAT2#sh ip arp inspection

Source Mac Validation      : Enabled
```

```
Destination Mac Validation : Enabled
IP Address Validation      : Disabled

 Vlan     Configuration    Operation   ACL Match          Static ACL
 ----     -------------    ---------   ---------          ----------
  101     Enabled          Active      ARP_R9            No

 Vlan     ACL Logging      DHCP Logging      Probe Logging
 ----     -----------      ------------      -------------
  101     Acl-Match        All               Off

 Vlan      Forwarded        Dropped     DHCP Drops      ACL Drops
 ----      ---------        -------     ----------      ---------
  101             46              0              0              0

 Vlan   DHCP Permits    ACL Permits  Probe Permits   Source MAC Failures
 ----   ------------    -----------  -------------   -------------------
  101             24             22              0                     0

 Vlan   Dest MAC Failures   IP Validation Failures   Invalid Protocol Data
 ----   -----------------   ----------------------   ---------------------

 Vlan   Dest MAC Failures   IP Validation Failures   Invalid Protocol Data
 ----   -----------------   ----------------------   ---------------------
  101                   0                        0                      0

CAT2#sh ip arp inspection int f0/1

 Interface        Trust State     Rate (pps)    Burst Interval
 ---------------  -----------     ----------    --------------
 Fa0/1            Untrusted               10                 3
```

Finally change an IP address on R1 to something static (re-enable DHCP after testing) and look at the logs on the switch:

```
Jul 26 13:56:10.180: %SW_DAI-4-DHCP_SNOOPING_DENY: 2 Invalid ARPs (Res) on Fa0/1, vlan
101.([001b.d50f.f371/10.10.11.23/ffff.ffff.ffff/10.10.11.23/13:56:10 UTC Sat Jul 26 2014])

Jul 26 13:56:10.180: %SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Res) on Fa0/1, vlan
101.([001b.d50f.f371/10.10.11.23/c84c.751f.ddc0/10.10.11.1/13:56:10 UTC Sat Jul 26 2014])
```

### Task 38.3 IP Source Guard

- Configure Cat2 to prevent against IPv4 spoofing attacks in VLAN101.
- Not only IP addresses should be validated but also MACs.
- Enable IP Source Guard on F0/1 and F0/8.
- Also configure a static source binding for R9.

## Detailed Solution

**CAT2**

```
interface f0/1
 switchport port-security
 ip verify source port-security

interface f0/8
 switchport port-security
 ip verify source port-security

ip source binding 30E4.DBCE.8490 vlan 101 10.10.11.9 interface Fa0/9
```

IP Source Guard is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured bindings. You can use this feature to prevent traffic attacks if a host tries to use the IP address of its neighbor.

It is also possible to extend that basic L3 functionality with L2 information – so the switch not only looks at the IP but also MAC address. A pre-requisite is to enable Port Security on the interface.

Note that "ip source binding" is a Privileged EXEC Mode command. An entry was asked to be configured for R9, but it would not normally take effect since F0/10 is a DHCP-trusted port (Source Guard would be inactive even if you enabled it on that port).

## Verification

```
CAT2#sh ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Fa0/1      ip-mac       active       10.10.11.3       00:1B:D5:0F:F3:71  101
Fa0/8     ip-mac       active       10.10.11.1       C8:4C:75:1F:DD:C0  101
```

Now change MAC address on R1's G0/0 and try to ping R9 :

```
R1(config-if)#do ping 10.10.11.9

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.11.9, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

CAT2#deb ip verify source packet
Ip source guard debug packet debugging is on
CAT2#
Jul 28 16:33:53.695: DHCP_SECURITY_SW: validate port security packet, recv port:
FastEthernet0/1, recv vlan: 101, mac: 0000.1234.1234, invalid flag: 1.
Jul 28 16:33:53.695: DHCP_SECURITY_SW: validate port security packet, recv port:
FastEthernet0/1, recv vlan: 101, mac: 0000.1234.1234, invalid flag: 1.
```

### Task 38.4 Catalyst Ingress Access-lists

- Configure Port ACLs on CAT4.
- ICMP Echos and Telnet packets received on Fa0/8 should be dropped and logged.
- On the same interface block AppleTalk and ARP frames coming from 0000.cc1e.cc1e.
- Other traffic should not be affected.

## Detailed Solution

**CAT4**

```
ip access-list extended L2ACL
 deny   icmp any any echo log
 deny   tcp any any eq telnet log
 permit ip any any

mac access-list extended MACL
 deny   host 0000.cc1e.cc1e any appletalk
 deny   host 0000.cc1e.cc1e any 0x806 0x0
 permit any any

interface FastEthernet0/8
 ip access-group L2ACL in
 mac access-group MACL in
```

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces and can be applied only on interfaces in the inbound direction. They are processed the same way as regular ACLs on a router – top down until a match is found. If there is no match an implicit "deny" applies.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

## Verification

Pings from R2 to R8 should work, but not in the opposite direction:

```
R2#ping 10.0.117.8

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.117.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R8#ping 10.0.117.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.117.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

R8#telnet 2.2.2.2
Trying 2.2.2.2 ...
% Connection timed out; remote host not responding

CAT4#
*Mar  3 05:35:47.063: %SEC-6-IPACCESSLOGDP: list L2ACL denied icmp 10.0.117.8 -> 10.0.117.2
(8/0), 1 packet
*Mar  3 05:36:02.892: %SEC-6-IPACCESSLOGP: list L2ACL denied tcp 10.0.117.8(38612) ->
2.2.2.2(23), 1 packet
```

Now change MAC on R8's G0/0.117 to be 0000.cc1e.cc1e. Clear ARP and try to connect to R2 :

```
R8#debug arp
*Jul 28 17:27:48.938: IP ARP: sent req src 10.0.117.8 0000.cc1e.cc1e,
                 dst 10.0.117.2 0000.0000.0000 GigabitEthernet0/2
R8#
*Jul 28 17:27:53.814: IP ARP: sent req src 10.0.117.8 0000.cc1e.cc1e,
                 dst 10.0.117.2 0000.0000.0000 GigabitEthernet0/2

R8(config-if)#do sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.0.115.3              2   001b.d50f.f2f9  ARPA   GigabitEthernet0/0.115
Internet  10.0.115.8              -   c84c.751f.ddc1  ARPA   GigabitEthernet0/0.115
Internet  10.0.117.2              0   Incomplete      ARPA
Internet  10.0.117.8              -   0000.cc1e.cc1e  ARPA   GigabitEthernet0/0.117
Internet  10.10.11.1              -   c84c.751f.ddc0  ARPA   GigabitEthernet0/1
Internet  10.10.11.3              2   001b.d50f.f371  ARPA   GigabitEthernet0/1
Internet  10.10.11.9              2   30e4.dbce.8490  ARPA   GigabitEthernet0/1

CAT4#sh access-l
Extended IP access list L2ACL
    10 deny icmp any any echo log (5 matches)
    20 deny tcp any any eq telnet log (2 matches)
    30 permit ip any any (71 matches)
Extended MAC access list MACL
    deny   host 0000.cc1e.cc1e any appletalk
    deny   host 0000.cc1e.cc1e any 0x806 0x0
    permit any any
```

### Task 38.5 Controlling Terminal Line Access

- Secure VTY lines on R9 and R1.
- Management traffic should be allowed from the following subnets :
  - 10.0.115.0/24

    - 10.0.117.0/24
    - 2010:0:117::/64
- R1 should only accept Telnet.
- R9 should only allow SSH access (user: cisco, pw: cisco).
- You will have to disable IP Source Guard to test this configuration.

## Detailed Solution

**CAT2**

```
int range f0/1, f0/8, f0/9
 no ip verify source
```

**R1**

```
ip access-list standard VTY4
 permit 10.0.115.0 0.0.0.255
 permit 10.0.117.0 0.0.0.255

ipv6 access-list VTY6
 permit tcp 2010:0:117::/64 any eq telnet

line vty 0 4
 access-class VTY4 in
 ipv6 access-class VTY6 in
 password cisco
 login
 transport input telnet
```

**R9**

```
ip domain-name iPexpert.com
crypto key gen rsa mod 1024
username cisco password cisco

ip access-list standard VTY4
 permit 10.0.115.0 0.0.0.255
 permit 10.0.117.0 0.0.0.255

ipv6 access-list VTY6
 permit tcp 2010:0:117::/64 any eq ssh

line vty 0 4
 access-class VTY4 in
 ipv6 access-class VTY6 in
 login local
 transport input ssh
```

You can control who can access the virtual terminal lines (VTYs) to a router/switch by applying an access list inbound (“access-class .. in” or “ipv6 access-class .. in”). You can also control the destinations that the VTYs from a router can reach by applying an access list to outbound VTYs (“access-class .. out” or “ipv6 access-class .. out”).

## Verification

```
R3#telnet 10.10.11.1
Trying 10.10.11.1 ... Open

User Access Verification

Password:
R1>exi

[Connection to 10.10.11.1 closed by foreign host]

R3#telnet 10.10.11.1 /source-interface loop0
Trying 10.10.11.1 ...
% Connection refused by remote host
```

```
R3#telnet 1::1
Trying 1::1 ...
% Connection refused by remote host

R2#telnet 1::1
Trying 1::1 ... Open

User Access Verification

Password:
R1>

R2#telnet 1.1.1.1
Trying 1.1.1.1 ... Open


User Access Verification

Password:
R1>

R1#sh access-l
Standard IP access list VTY4
    10 permit 10.0.115.0, wildcard bits 0.0.0.255 (2 matches)
    20 permit 10.0.117.0, wildcard bits 0.0.0.255 (2 matches)
IPv6 access list VTY6
    permit tcp 2010:0:117::/64 any eq telnet (2 matches) sequence 10

R3#ssh -l cisco 10.10.11.9

Password:

R9>

R2#ssh -l cisco 10.10.11.9

Password:
R9>

R8#ssh -l cisco 10.10.11.9
% Connection refused by remote host
R3#ssh -l cisco 9::9
% Connection refused by remote host

R2#ssh -l cisco 9::9

Password:
R9>

R2#telnet 9::9
Trying 9::9 ...
% Connection refused by remote host

R9(config)#do sh access-l
Standard IP access list VTY4
    10 permit 10.0.115.0, wildcard bits 0.0.0.255 (2 matches)
    20 permit 10.0.117.0, wildcard bits 0.0.0.255 (4 matches)
IPv6 access list VTY6
 ….permit tcp 2010:0:117::/64 any (2 matches) sequence 10
```

## **Task 38.6** Control Plane Policing

- R8 should be configured to protect its CPU using CoPP.
- Rate-limit all ICMP packets to 15 per second.
- Rate-limit all ICMPv6 packets to 70000bps.

- All HTTP packets originating from 3.3.3.3 should be dropped.
- Outbound telnet packets destined to 1.1.1.1 should be dropped and logged. Log messages should be generated every 2 seconds and they should include TTL and length of dropped packets.
- OSPFv2 and OSPFv3 packets should not be affected by this configuration.

## Detailed Solution

**R8**

```
ip access-list extended ICMP
 permit icmp any any

ipv6 access-list ICMP6
 permit icmp any any

ip access-list extended HTTP_FROM_R3
 permit tcp host 3.3.3.3 any eq www

ip access-list extended TELNET_TO_R1
 permit tcp any host 1.1.1.1 eq telnet

ip access-list extended OSPF
 permit ospf any any

ipv6 access-list OSPFv3
 permit 89 any any

class-map match-all ICMP_CLASS
 match access-group name ICMP

class-map match-all OSPF_CLASS
 match access-group name OSPF

class-map match-all OSPFv3_CLASS
 match access-group name OSPFv3

class-map match-all ICMP6_CLASS
 match access-group name ICMP6

class-map match-all TELNET_TO_R1_CLASS
 match access-group name TELNET_TO_R1

class-map match-all HTTP_FROM_R3_CLASS
 match access-group name HTTP_FROM_R3

policy-map COPP_OUT_POL
 class TELNET_TO_R1_CLASS
  drop
  log interval 2000 ttl total-length

policy-map COPP_IN_POL
 class OSPF_CLASS
 class OSPFv3_CLASS
 class ICMP_CLASS
  police rate 15 pps burst 5 packets
 class ICMP6_CLASS
  police 70000
 class HTTP_FROM_R3_CLASS
  drop

control-plane
 service-policy input COPP_IN_POL
 service-policy output COPP_OUT_POL
```

Control Plane Policing is kind of a virtual interface that sits before CPU. It provides filtering and rate-limiting capabilities. What is important for us, are two main characteristics of this feature.

1. It works on Aggregate Traffic from all interfaces – it affects ALL packets destined to the Route Processor.
2. It can be configured for ingress and egress flows, meaning from the router as well. Note that for the egress direction it does not provide any performance benefits, it is just to limit the traffic leaving the router.

To configure CoPP we use well-known MQC syntax. Traffic of interest must be first identified in a class-map and then the policy can be specified inside a policy-map. The main tool used for classification is an Access-List; however NBAR can be used to match ARP and PPPoE packets. The last option is to use DSCP or IP Precedence values.

It is also possible to log traffic matching our classes by using the "log" option within the policy. You can specify the Interval at which messages are generated and optionally you can include TTL and Total Length of the packets.

Even that feature is supported for IPv6 it may still not work as one would expect – its behavior depends on the IOS code used on the device.

## Verification

```
R1#ping 8.8.8.8 rep 20

Type escape sequence to abort.
Sending 20, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!!.!!!!!.!!!!!.!!
Success rate is 85 percent (17/20), round-trip min/avg/max = 1/1/4 ms

R1#ping ipv6 8::8 repeat 300

Type escape sequence to abort.
Sending 300, 100-byte ICMP Echos to 8::8, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!.!!!!!!!!!
Success rate is 97 percent (292/300), round-trip min/avg/max = 0/0/4 ms

R3#telnet 8.8.8.8 80 /source-interface l0
Trying 8.8.8.8, 80 ...
% Connection timed out; remote host not responding

R8#telnet 1.1.1.1
Trying 1.1.1.1 ...
% Connection timed out; remote host not responding

R8#
*Jul 29 10:08:44.822: %CP-6-TCP: DROP Egress Control-plane Policing ttl=255 length=44
10.10.11.1(57106) -> 1.1.1.1(23)
*Jul 29 10:08:46.822: %CP-6-TCP: DROP Egress Control-plane Policing ttl=255 length=44
10.10.11.1(57106) -> 1.1.1.1(23)

R8#telnet 1.1.1.1 /source-interface l0
Trying 1.1.1.1 ...
*Jul 29 10:09:23.802: %CP-6-TCP: DROP Egress Control-plane Policing ttl=255 length=44
8.8.8.8(26245) -> 1.1.1.1(23)
*Jul 29 10:09:25.802: %CP-6-TCP: DROP Egress Control-plane Policing
ttl=255 length=44 8.8.8.8(26245) -> 1.1.1.1(23)
```

```
% Connection timed out; remote host not responding

R8#telnet 10.10.11.1
Trying 10.10.11.1 ...
% Connection refused by remote host

R8#sh control-plane aggregate features
Control plane aggregate path features :

------------------------------------------------------
Control-plane Policing activated Jul 29 2014 09:4

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

R8#sh control-plane aggregate counters
Control plane aggregate path counters :

Feature                  Packets Processed/Dropped/Errors

------------------------------------------------------
Control-plane Policing        905/13/0

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

R8#sh policy-map control-plane
 Control Plane

  Service-policy input: COPP_IN_POL

    Class-map: OSPF_CLASS (match-all)
      168 packets, 16112 bytes
      5 minute offered rate 0000 bps
      Match: access-group name OSPF

    Class-map: OSPFv3_CLASS (match-all)
      168 packets, 16296 bytes
      5 minute offered rate 0000 bps
      Match: access-group name OSPFv3

    Class-map: ICMP_CLASS (match-all)
      20 packets, 2280 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: access-group name ICMP
      police:
          rate 15 pps, burst 5 packets
        conformed 17 packets, 17 bytes; actions:
          transmit
        exceeded 3 packets, 3 bytes; actions:
          drop
        conformed 0 pps, exceeded 0 pps

    Class-map: ICMP6_CLASS (match-all)
      209 packets, 23754 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: access-group name ICMP6
      police:
          cir 70000 bps, bc 2187 bytes
        conformed 201 packets, 22842 bytes; actions:
          transmit
        exceeded 8 packets, 912 bytes; actions:
          drop
        conformed 0000 bps, exceeded 0000 bps

    Class-map: HTTP_FROM_R3_CLASS (match-all)
      2 packets, 120 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: access-group name HTTP_FROM_R3
```

```
        drop

      Class-map: class-default (match-any)
        285 packets, 143296 bytes
        5 minute offered rate 0000 bps, drop rate 0000 bps
        Match: any

    Service-policy output: COPP_OUT_POL

      Class-map: TELNET_TO_R1_CLASS (match-all)
        2 packets, 120 bytes
        5 minute offered rate 0000 bps, drop rate 0000 bps
        Match: access-group name TELNET_TO_R1
        drop
         log

      Class-map: class-default (match-any)
        1871 packets, 780560 bytes
        5 minute offered rate 1000 bps, drop rate 0000 bps
        Match: any
```

## Task 38.7 Control Plane Protection

- Enable Control Plane Protection on R9.
- Packets destined to non-listening ports should be silently dropped.
- Telnet connections over port 3020 should be unaffected.
- Input queue of R9 should not be overwhelmed by any single protocol traffic.
- No more than 100 BGP and 4 SSH packets should be queued.
- No more than 30 packets for all other TCP/UDP protocols enabled on the router should be seen in the queue.
- All IPv4 transit traffic punted to the CPU should be policed to 512kbps.

## Detailed Solution

**R9**

```
ip cef

ip access-list extended ALLIP
 permit ip any any

class-map match-all ALLIP
 match access-group name ALLIP

class-map type port-filter match-all PF_CLASS
 match closed-ports
 match not port tcp 3020

class-map type queue-threshold match-all SSH_CLASS
 match protocol ssh

class-map type queue-threshold match-all BGP_CLASS
 match protocol bgp

class-map type queue-threshold match-all OTHER_CLASS
 match host-protocols

policy-map type queue-threshold QT_POL
 class BGP_CLASS
   queue-limit 100
 class SSH_CLASS
   queue-limit 4
 class OTHER_CLASS
   queue-limit 30
```

```
policy-map type port-filter PF_POL
 class PF_CLASS
   drop

policy-map TRANSIT_POL
 class ALLIP
  police 512000

control-plane host
 service-policy type port-filter input PF_POL
 service-policy type queue-threshold input QT_POL

control-plane transit
 service-policy input TRANSIT_POL
```

Control Plane Protection allows more granular control (than CoPP) over what packets going to the CPU will be examined by the router to make a forwarding decision. This feature includes traffic classifier that breaks aggregate interface of Control Plane Policing into three categories, known as subinterfaces:

- CEF Exception
- Host
- And Transit

CEF Exception receives all traffic that is either redirected as a result of a configured input feature in the CEF packet forwarding path for process switching or directly enqueued in the control plane input queue by the interface driver (that is, ARP, external BGP (eBGP), OSPF, LDP, Layer2 Keepalives, and all non-IP host traffic).

Transit subinterface receives all control-plane IP traffic that is software switched by the route processor. This means packets that are not directly destined to the router itself but rather traffic traversing through the router.

Host subinterface receives all control-plane IP traffic that is directly destined for one of the router interfaces. Examples include management traffic or routing protocols such as SSH, SNMP, internal BGP (iBGP), and EIGRP. All host traffic terminates on and is processed by the router. This CPPr component also allows toconfigure two additional features which are addition to the CoPP technology – Port Filtering and Queue Thresholding.

Port-filtering enhances control plane protection by providing for early dropping of packets directed toward closed or nonlistened IOS TCP/UDP ports on the router. The Port Filter maintains a global database of all open TCP and UDP ports on the router, including random ephemeral ports created by applications.

Queue Thresholding provides a mechanism for limiting the number of unprocessed packets a protocol can have at process-level. The intent of this feature is to prevent the input queue from being overwhelmed by any single protocol traffic. Same as Port Filtering, Queue Thresholding can be only applied to the Host subinterface.

Configuration is similar to CoPP. Speaking of differences MQC components can be now of a particular type (`port-filter, queue-threshold and logging`). If you don't specify a type it means that you are configuring a drop/rate-limit policy.

Control Plane Protection feature set depends on Cisco Express Forwarding (CEF) for IP packet redirection – CEF must be always enabled when working with this feature.

## Verification

```
R9#sh control-plane features
Total 3 features configured

Control plane host path features :
-------------------------------------------------------
TCP/UDP Portfilter activated Jul 29 2014 12:3
Protocol Queue Thresholding activated Jul 29 2014 12:3
-------------------------------------------------------

Control plane transit path features :
-------------------------------------------------------
Control-plane Policing activated Jul 29 2014 12:3
-------------------------------------------------------
```

Ports 1234 and 3020 are closed :

```
R9#show control-plane host open-ports
Active internet connections (servers and established)
Prot               Local Address             Foreign Address                   Service
State
 tcp                        *:22                        *:0               SSH-Server
LISTEN
 tcp                        *:23                        *:0                   Telnet
LISTEN
 udp                        *:67                        *:0            DHCPD Receive
LISTEN
 udp                       *:123                        *:0                     NTP
LISTEN

R9#show control-plane counters
Feature Path             Packets processed/dropped/errors
Aggregate                  127677/0/0
Host                         1681/0/0
Transit                         0/0/0
Cef-exception              125996/0/0

R1#telnet 10.10.11.9 1234
Trying 10.10.11.9, 1234 ...
% Connection timed out; remote host not responding

R1#telnet 10.10.11.9 3020
Trying 10.10.11.9, 3020 ...
% Connection refused by remote host
```

Only packets sent to 1234 were silently dropped; 3020 was a no-match for our PF_CLASS :

```
R9#show control-plane counters
Feature Path             Packets processed/dropped/errors
Aggregate                  127727/0/0
Host                         1683/2/0
Transit                         0/0/0
Cef-exception              126044/0/0

R9#show policy-map type port-filter control-plane host
 Control Plane Host

  Service-policy port-filter input: PF_POL

    Class-map: PF_CLASS (match-all)
      2 packets, 120 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
```

```
        Match:  closed-ports
        Match: not  port tcp 3020
        drop

      Class-map: class-default (match-any)
        1 packets, 60 bytes
        5 minute offered rate 0000 bps, drop rate 0000 bps
        Match: any
```

Now test Queue Threshold for SSH :

```
R3#ssh -l cisco 10.10.11.9

Password:
R9>
R9#sh policy-map type queue-threshold control-plane host
 Control Plane Host

  Service-policy queue-threshold input: QT_POL

    Class-map: BGP_CLASS (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  protocol bgp
      queue-limit 100
      queue-count 0     packets allowed/dropped 0/0

    Class-map: SSH_CLASS (match-all)
      32 packets, 2926 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  protocol ssh
      queue-limit 4
      queue-count 1     packets allowed/dropped 32/0

    Class-map: OTHER_CLASS (match-all)
      1 packets, 90 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  host-protocols
      queue-limit 30
      queue-count 0     packets allowed/dropped 1/0

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

We will not be able to test the Transit policy since R9 is a stub router :

```
R9#sh policy-map control-plane transit
 Control Plane Transit

  Service-policy input: TRANSIT_POL

    Class-map: ALLIP (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: access-group name ALLIP
      police:
          cir 512000 bps, bc 16000 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0000 bps, exceeded 0000 bps

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

### Task 38.8 Control Plane Protection – Logging

- All malformed & allowed packets received on Host subinterface should be logged.
- Rate-limit those log messages to one every 5 seconds.
- Log all dropped Transit packets that entered R9 through interface F0/0.
- Allowed and over the Input Queue limit SSH traffic should be logged as well.

## Detailed Solution

**R9**

```
policy-map type queue-threshold QT_POL
 class SSH_CLASS
    log

class-map type logging match-any HOST_LOG_CLASS
 match packets permitted
 match packets error

class-map type logging match-all TRANSIT_LOG_CLASS
 match packets dropped
 match input-interface FastEthernet0/0

policy-map type logging HOST_LOG_POL
 class HOST_LOG_CLASS
    log interval 5000

policy-map type logging TRANSIT_LOG_POL
 class TRANSIT_LOG_CLASS
    log

control-plane host
 service-policy type logging input HOST_LOG_POL

control-plane transit
 service-policy type logging input TRANSIT_LOG_POL
```

Control Plane Logging feature can be either configured globally for the entire subinterface (`type logging class/policy –map`) or within a class-map (class-specific logging). Global Logging allows you to generate a log message for packets allowed, dropped and/or malformed. Class-specific logging logs all packets matching a class.

## Verification

```
R9#sh control-plane features
Total 5 features configured

Control plane host path features :

------------------------------------------------------
Control-plane Logging activated Jul 29 2014 13:0
TCP/UDP Portfilter activated Jul 29 2014 12:3
Protocol Queue Thresholding activated Jul 29 2014 12:3
------------------------------------------------------

Control plane transit path features :

------------------------------------------------------
Control-plane Logging activated Jul 29 2014 13:0
Control-plane Policing activated Jul 29 2014 12:3
------------------------------------------------------
```

Now enable HTTP server on R9 and connect from R1. Since our Host interface policy only drops packets sent to closed ports (or over the queue limit), few HTTP packets should be permitted and this way also logged :

```
R1#telnet 10.10.11.9 80
Trying 10.10.11.9, 80 ... Open
get /
HTTP/1.1 400 Bad Request
Date: Tue, 29 Jul 2014 13:15:17 GMT
Server: cisco-IOS
Connection: close
Accept-Ranges: none

400 Bad Request

[Connection to 10.10.11.9 closed by foreign host]

R9#
Jul 29 13:15:16.831: %CP-6-TCP: PERMIT  10.10.11.1(36483) -> 10.10.11.9(80)

R9#sh policy-map type logging control-plane all
 Control Plane Host

  Service-policy logging input: HOST_LOG_POL

    Class-map: HOST_LOG_CLASS (match-any)
      40 packets, 2522 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  packets permitted
        0 packets, 0 bytes
        5 minute rate 0 bps
      Match:  packets error
        0 packets, 0 bytes
        5 minute rate 0 bps
      log interval 5000

    Class-map: class-default (match-any)
      19 packets, 1232 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
 Control Plane Transit

  Service-policy logging input: TRANSIT_LOG_POL

    Class-map: TRANSIT_LOG_CLASS (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  packets dropped
      Match: input-interface FastEthernet0/0
      log

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

To generate logs for allowed and over the queue limit SSH packets connect from R3 and issue “show ip int br” few times:

```
R3#ssh -l cisco 10.10.11.9

Password:

R9>sh ip int br
Interface                  IP-Address      OK? Method Status                Protocol
Embedded-Service-Engine0/0 unassigned      YES unset  administratively down down
FastEthernet0/0         10.10.11.9     YES manual up                    up
Loopback0                  9.9.9.9  YES manual up                    up
```

```
R9#
Jul 29 13:00:23.892: %CP-6-TCP: PERMIT  10.0.115.3(54982) -> 10.10.11.9(22)
Jul 29 13:00:24.088: %CP-6-TCP: PERMIT  10.0.115.3(54982) -> 10.10.11.9(22)
Jul 29 13:00:24.892: %CP-6-TCP: PERMIT  10.0.115.3(54982) -> 10.10.11.9(22)
Jul 29 13:00:24.896: %CP-6-TCP: DROP Protocol Queue Thresholding  10.0.115.3(54982) ->
10.10.11.9(22)
Jul 29 13:00:24.896: %CP-6-TCP: DROP Protocol Queue Thresholding  10.0.115.3(54982) ->
10.10.11.9(22)
Jul 29 13:00:24.896: %CP-6-TCP: DROP Protocol Queue Thresholding  1
0.0.115.5(54982) -> 10.10.11.9(22)
```

### Task 38.9 Flexible Packet Matching

- Use Flexible Packet Matching to drop & log malicious traffic going through R2.
- The offending packets are sourced in VLAN 101 and they contain string “xExe” within the first 200B from the beginning of TCP Payload.
- Those packets are destined to TCP port 8013.
- Other traffic flowing over the same port number should not be affected.

## Detailed Solution

**R2**

```
load protocol system:/fpm/phdf/ip.phdf
load protocol system:/fpm/phdf/tcp.phdf

class-map type stack match-all STACK
 match field IP protocol eq 0x6 next TCP
 match field TCP dest-port eq 0x1F4D next TCP

class-map type access-control match-all NOGO
 match field IP source-addr range 10.10.11.1 10.10.11.254
 match start TCP payload-start offset 0 size 200 regex ".*xExe.*"

policy-map type access-control NOGO
 class NOGO
   drop
   log

policy-map type access-control FPM_POL
 class STACK
  service-policy NOGO

interface FastEthernet0/1
 service-policy type access-control input FPM_POL
```

FPM or Flexible Packet Matching can be thought of as a next-generation access-list providing more thorough and customized packet filters. The main advantage of this feature is that it allows us to match an arbitrary string of bits within either the packet header or its payload.

Things to be aware of about FPM:

- It is completely stateless; it does not keep track of dynamic ports.
- It cannot match across packets - it treats each packet independently from each other.
- It cannot classify packets with IP Options.
- It is not supported on tunnel and MPLS interfaces.

One important component of FPM is Protocol Header Definition Files. These are just predefined files containing structure of protocol headers and their fields.

Whenever you configure FPM you will have to create two class-maps and two policy-maps. First, build a protocol stack (class-map type stack) which defines a general structure of packets that should be inspected (in our case TCP with destination 8013 which in hex is 0x1F4D). Then your specific matches should be part of "class-map type access-control" (here the source of VLAN 101 and string "xExe" in the payload). Now the policies, both "type access-control". First policy is where you configure actions for your specific class and then the second one operates on the stack class. Note that first policy is nested in the second via "service-policy".

As a general rule you should always try to use hex numbers wherever possible. Decimal matches are supported but they may not work 100% of the time.

## Verification

Reconfigure CAT4 to test:

```
CAT4(config)#ip http server
CAT4(config)#ip http port 8013

R1#copy http://10.70.70.140:8013/OKfile.exe null0
Destination filename [null0]?
Accessing http://10.70.70.140:8013/OKfile.exe...
%Error opening http://10.70.70.140:8013/OKfile.exe (No such file or directory)

R1#copy http://10.70.70.140:8013/xexe null0
Destination filename [null0]?
Accessing http://10.70.70.140:8013/xexe...
%Error opening http://10.70.70.140:8013/xexe (No such file or directory)

R1#copy http://10.70.70.140:8013/xExe null0
Destination filename [null0]?
Accessing http://10.70.70.140:8013/xExe...
%Error opening http://10.70.70.140:8013/xExe (I/O error)

R2#
*Jul 29 17:34:09.239: %SEC-6-IPACCESSLOGP: list NOGO denied tcp 10.10.11.1(33101)
(FastEthernet0/1 ) -> 10.70.70.140(8013), 7 packets

R2#show policy-map type access-control interface F0/1
 FastEthernet0/1

  Service-policy access-control input: FPM_POL

    Class-map: STACK (match-all)
      55 packets, 4506 bytes
      5 minute offered rate 0 bps
      Match: field IP protocol eq 0x6 next TCP
      Match: field TCP dest-port eq 0x1F4D next TCP

      Service-policy access-control : NOGO

        Class-map: NOGO (match-all)
          7 packets, 1260 bytes
          5 minute offered rate 0 bps
          Match: field IP source-addr range 10.10.11.1 10.10.11.254
          Match: start TCP payload-start offset 0 size 200 regex ".*xExe.*"
            drop
            log

        Class-map: class-default (match-any)
          48 packets, 3246 bytes
          5 minute offered rate 0 bps, drop rate 0 bps
          Match: any

    Class-map: class-default (match-any)
```

```
        26 packets, 2452 bytes
        5 minute offered rate 0 bps, drop rate 0 bps
        Match: any
```

Packets from R3 were allowed since they are not coming from VLAN 101:

```
R3#copy http://10.70.70.140:8013/xExe null0
Destination filename [null0]?
Accessing http://10.70.70.140:8013/xExe...
%Error opening http://10.70.70.140:8013/xExe (No such file or directory)

CAT4(config)#no ip http server
CAT4(config)#no ip http port 8013
```

**You have completed Lab 38**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 39: Configure and Troubleshoot IP/IOS Services (Part 1)

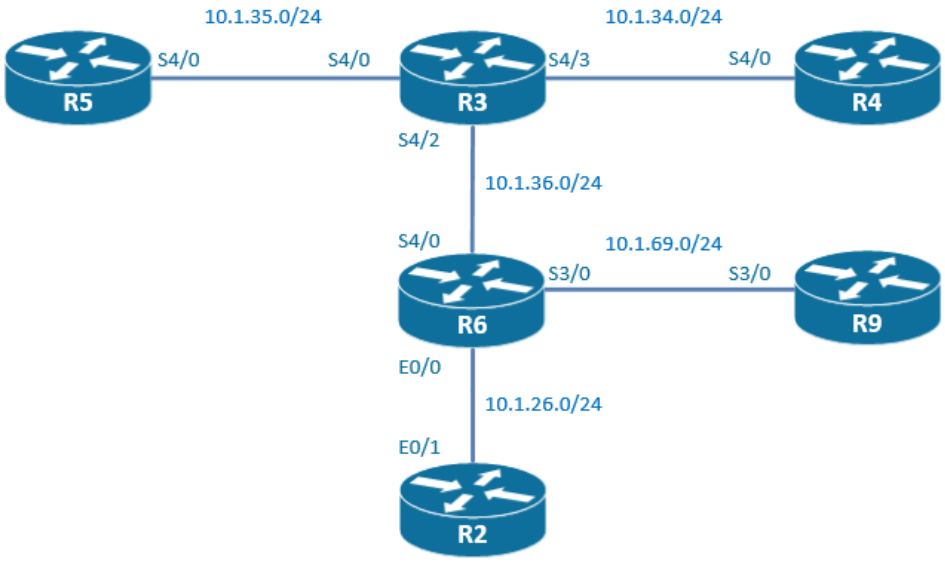
## Technologies covered

- Syslog logging
- Logging timestamps
- Logging to flash
- Configuration change notification
- Configuration archive and rollback
- Conditional debugging

## Overview

You have been tasked to configure management services in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 39.1** Configure R2 to log system messages to a syslog server with the IP address 10.2.2.2. Send only emergencies, alerts, and critical messages.

On R2, configure the following:

```
logging 10.2.2.2 emergencies
logging traps 2
Here is the correspondance between numbers and names.
R2(config)#logging trap ?
  <0-7>          Logging severity level
  alerts         Immediate action needed           (severity=1)
  critical       Critical conditions               (severity=2)
  debugging      Debugging messages                (severity=7)
  emergencies    System is unusable                (severity=0)
  errors         Error conditions                  (severity=3)
  informational  Informational messages            (severity=6)
  notifications  Normal but significant conditions (severity=5)
  warnings       Warning conditions                (severity=4)
  <cr>
```

**Task 39.2** Configure R2 to log all messages with a severity from 1 to 7 in an internal buffer. The size of this buffer should be 20000.

On R2, configure the following:

```
logging buffered 20000 debugging
```

**Task 39.3** Make sure that any type of log messages has the exact date and time stamp (and not the uptime).

On R2, configure the following:

```
service timestamps log datetime year
```

**Task 39.4** If two system messages arrive with the same timestamps, make sure (with sequence numbers) that you can still know which one was generated first.

On R2, configure the following:

```
service sequence-numbers
```

**Task 39.5** Configure R2 to log only emergencies, alerts, critical, and errors messages to the console.

On R2, configure the following:

```
logging console 3
```

**Task 39.6** Ensure that the router does keep a history file of 10 logged messages prepared to be sent as SNMP traps.

On R2, configure the following:

```
logging history size 10
```

**Task 39.7** Limit the rate of logging messages to 70 per second for all logging messages, except for those with a severity level between 5 and 7.

On R2, configure the following:

```
logging rate-limit all 70 except 5
```

**Task 39.8** On R6, write the syslog messages into a file on the flash memory in a directory called "syslog". Once the size of the sum of all the logging files is reaching 64000 bytes, the oldest file is deleted. Each file should have a maximum size of 10000 bytes.

On R6, configure the following:

```
logging persistent url disk0:/syslog size 64000 filesize 10000
```

**Task 39.9** Log every configuration command entered on R9. Log the last 500 configuration command messages locally. Make sure that the passwords and SNMP community strings are replaced by ****asterisk****. Log, also the configuration command messages on a syslog server.

On R9, configure the following:

```
archive
 log config
  logging size 500
  notify syslog contenttype plaintext
  hidekeys
```

**Task 39.10** On R3, enable the archive feature to store the configuration files on the flash. The maximum number of archive saved should be 10.

On R3, configure the following:

```
archive
path disk0:
maximum 10
```

**Task 39.11** Save the configuration on R3. Change the hostname of R3 to R3-TEST, don't confirm and make sure that the configuration is automatically rolled back to the hostname R3 after 1 minutes.

On R3, we save the configuration:

```
R3#wr
Building configuration...
[OK]
```

I configure the hostname from R3 to R3-TEST.

```
R3# configure terminal revert timer 1
R3(config)#hostname R3-TEST
R3-TEST(config)#
```

After 1 minutes, the configuration is automatically reverted to the last saved configuration. In order to avoid the rollback, the "configure confirm" has to be entered before the 1 minutes timer.

**You have completed Lab 39**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 40: Configure and Troubleshoot IP/IOS Services (Part 2)

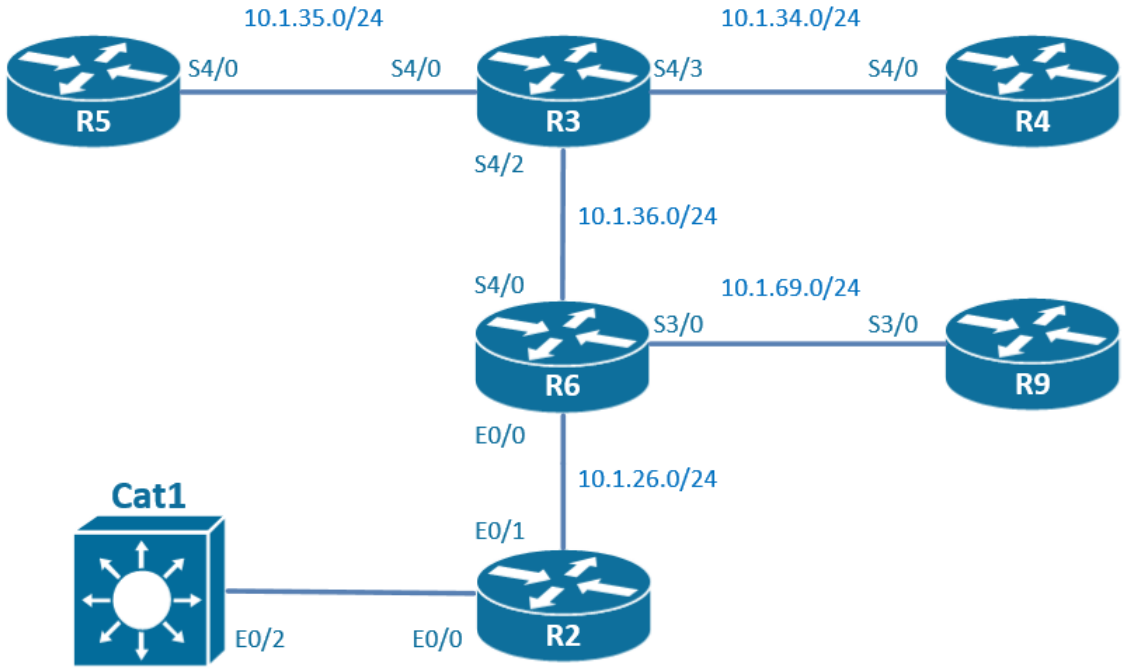
## Technologies covered

- SNMP v2
- SNMP v3
- NTP

## Overview

You have been tasked to configure management services in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 40.1** On R2, permit any SNMP server to poll the router with read-only permission using the community string iPexpert.

On R2, configure the following:

```
snmp-server community iPexpert RO
```

**Task 40.2** R2 should send IPSEC traps to the server 10.4.4.4 using SNMPv2c. The community iPexpert is included in the traps.

On R2, configure the following:

```
snmp-server enable traps ipsec cryptomap add
snmp-server enable traps isakmp  policy add
snmp-server host 10.4.4.4 traps version2c ipsec
snmp-server host 10.4.4.4 iPexpert
```

**Task 40.3** On R6, permit only hosts 10.4.4.4 and 10.4.4.3 to poll the router with read-only permission using the community string iPexpert. Use access-list number 6.

On R6, configure the following:

```
access-list 6 permit 10.4.4.4
access-list 6 permit 10.4.4.3
snmp-server community iPexpert RO 6
```

**Task 40.4** R2 should send all syslog messages as SNMP ACKed traps to the server 10.4.4.4 using SNMPv2c. ACKed trap means that an ACK packet should be sent by the server back to R2 to confirm that he received the trap. The community iPexpert is included in the traps.

We can configure the router to forward syslog messages to your network management server as SNMP traps instead of syslog packets. We have to forward SNMP informs and not SNMP traps because the SNMP messages should be acknowledged by the NMS.

On R2, configure the following:

```
logging history informational
snmp-server enable traps
snmp-server host 10.4.4.4 informs version 2c iPexpert syslog
```

**Task 40.5** R3 is going to be polled by a NMS with an IP address of 10.5.5.5. This polling should be configured according to the AuthPriv security model. Create two views, a RO view called ROVIEW and a RW view called RWVIEW. Make the MIB-2 objects accessible for both views.

By referring to the AuthPriv security model, the question is asking to configure SNMPv3. SNMPv3 offers 3 different security levels (noAuthNoPriv, AuthNoPriv, AuthPriv).

On R3, configure the following:

```
snmp-server view ROVIEW mib-2 include
snmp-server view RWVIEW mib-2 include
```

**Task 40.6** On R3, define a RO group called ROGROUP. Associate to this group the following user:

- username: Username1
- password: Password1
- encryption password: iPexpert
- Use the SHA authentication method and the 3-DES encryption method.

On R3, configure the following:

```
snmp-server group ROGROUP v3 auth read ROVIEW
snmp-server user Username1 ROGROUP v3 auth sha Password1 priv 3des iPexpert
```

**Task 40.7** On R3, define a RW group called RWGROUP. Associate to this group the following user:

- username: Username2
- password: Password2
- encryption password: iPexpert
- Use the MD5 authentication method and the AES-256 encryption method.

On R3, configure the following:

```
snmp-server group RWGROUP v3 auth read ROVIEW write RWVIEW
snmp-server user Username2 RWGROUP v3 auth md5 Password2 priv aes 256 iPexpert
```

**Task 40.8** On R3, enable traps and informs to be sent to 10.5.5.5 using payload encryption. The user Username1 generates the traps and informs.

On R3, configure the following:

```
snmp-server host 10.5.5.5 version 3 priv Username1
snmp-server host 10.5.5.5 informs version 3 priv Username1
```

**Task 40.9** Configure Cat1 to send an SNMP version 2C trap with a community of "iPexpert" to the NMS 10.5.5.5 whenever the switch learns or time-outs a MAC address.

Please be aware that the MAC notification table will store events that are generated for dynamic addresses and not for internal addresses, multicast addresses or other static addresses.

On Cat1, configure the following:

```
snmp-server host 10.5.5.5 traps iPexpert
snmp-server enable traps mac-notification

interface range e0/0-3
snmp trap mac-notification added

interface range e1/0-3
snmp trap mac-notification added

interface range e2/0-3
snmp trap mac-notification added

interface range e3/0-3
snmp trap mac-notification added

interface range e4/0-3
snmp trap mac-notification added

interface range e5/0-3
snmp trap mac-notification added

interface range e6/0-3
snmp trap mac-notification added

interface range e7/0-3
snmp trap mac-notification added
```

**Task 40.10** On Cat1, enable the MAC address notification feature. Store the MAC address notification traps and send them to the NMS every 30 seconds. Keep a historical table of the 10 last MAC address notification messages locally on the switches.

On the Cat1, configure the following:

```
mac address-table notification
mac address-table notification interval 30
mac address-table notification history-size 10
```

**Task 40.11** Configure R5 as a stratum 5 NTP master.

On R5, configure the following:

```
ntp master 5
```

We can see that R5 is synchronized with its internal clock 127.127.1.1.

```
R5#sh 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 700 (1/100 of seconds), resolution is 4000
reference time is D7EE2FC0.4ED91760 (13:40:32.308 CET Sun Oct 19 2014)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 7939.05 msec, peer dispersion is 7937.98 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 16, last update was 7 sec ago.show ntp association detail
```

**Task 40.12** NTP server on R5 should source from interface S4/0.

On R5, configure the following:

```
ntp source s4/0
```

**Task 40.13** Configure R3 as client from NTP server R5. Configure NTP authentication between R3 and R5 with a key number of 1 and a password of “iPexpert”.

On R3, configure the following:

```
ntp server 10.1.35.5
```

Let’s see if the NTP client R3 has synchronized with the NTP server R5.

```
R3#sh ntp status
Clock is synchronized, stratum 6, reference is 10.1.35.5
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 8300 (1/100 of seconds), resolution is 4000
reference time is D7EE36D7.245A1D10 (14:10:47.142 CET Sun Oct 19 2014)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 7941.96 msec, peer dispersion is 189.49 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 64, last update was 18 sec ago.
```

It has synchronized and the stratum of the NTP client is 6 because it synchronized with a server which has a stratum of 5.

Let’s configure NTP authentication:

```
On the client side R3, configure the following:
ntp authentication-key 1 md5 iPexpert
ntp authenticate
ntp trusted-key 1
ntp server 10.1.35.5 key 1
```

On the server side R5, configure the following:

```
ntp authentication-key 1 md5 iPexpert
```

**Task 40.14** On R5, make sure that the only NTP client that can synchronized with R5 is the client with the IP address 10.1.35.3. Use an access-list called NTPCLIENT.

On R5, configure the following:

```
ip access-list standard NTPCLIENT
permit 10.1.35.3
ntp access-group serve-only NTPCLIENT
```

**Task 40.15** Make sure that only 10.1.35.5 can be the NTP server for R3. Configure on R3 an access-list called NTPSERVER.

On R3, configure the following:

```
ip access-list standard NTPSERVER
permit 10.1.35.5
ntp access-group peer NTPSERVER
```

**You have completed Lab 40**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 41: Configure and Troubleshoot IP/IOS Services (Part 3)

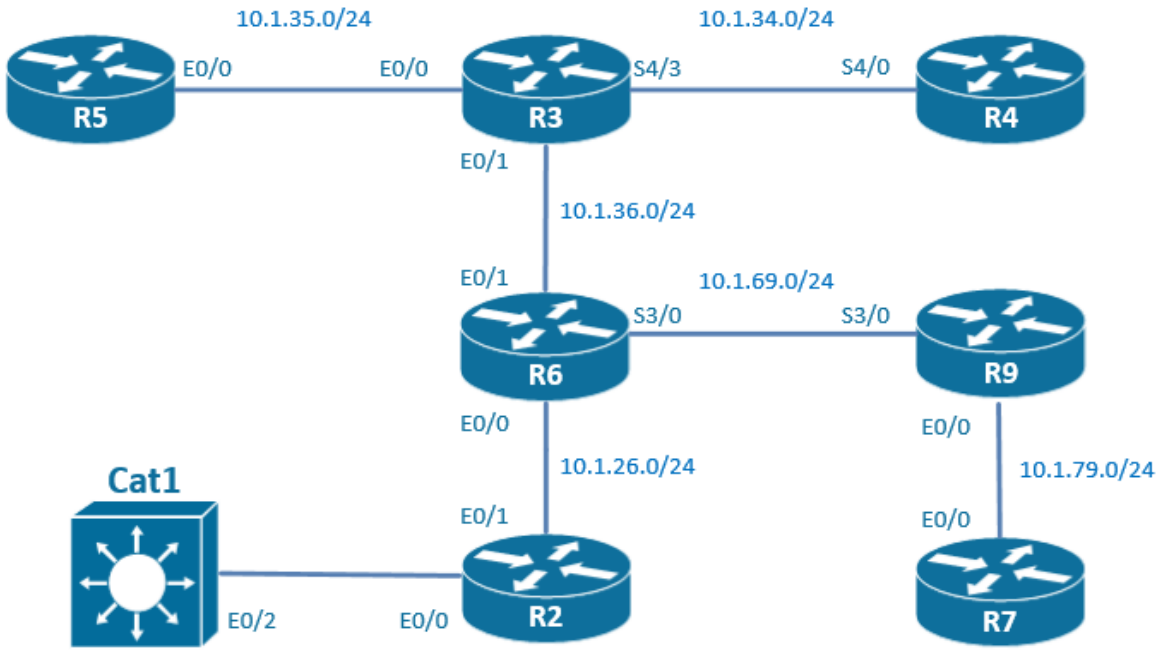
## Technologies covered

- EEM
- Proxy ARP
- Local Proxy ARP
- DHCP

## Overview

You have been tasked to configure management services in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 41.1** On R2, when the interface E0/1 is going down and the router is generating a syslog message regarding this event, create an EEM applet that will perform a show int E0/1 and send to the email noc@ipexpert.com the output of the command in the body of the mail. The mail server is 10.3.3.3, the originator of the mail is R2@ipexpert.com, the subject of the mail is ALERT_R2_E0_1_DOWN.

On R2, configure the following:

```
event manager environment mail_smtp  10.3.3.3
event manager environment mail_recipient noc@ipexpert.com
event manager environment mail_originator R2@ipexpert.com

event manager applet E0_1_DOWN
event syslog pattern "%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/1, changed
state to down"
action 1.0 cli command "show interface Ethernet0/1"
action 2.0 mail server "$mail_smtp" to "$mail_recipient" from "$mail_originator" subject "
ALERT_R2_E0_1_DOWN " body "$_cli_result"
```

Remember: the action cli command returns the output generated by the IOS CLI command in the $_cli_result variable.

```
R2(config)#int e0/1
R2(config-if)#shut
R2(config-if)#
%LINK-5-CHANGED: Interface Ethernet0/1, changed state to administratively down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/1, changed state to down
R2(config-if)#
%HA_EM-3-FMPD_SMTP: Error occurred when sending mail to SMTP server: 10.3.3.3 : error in
connecting to SMTP server
%HA_EM-3-FMPD_ERROR: Error executing applet E0_1_DOWN statement 2.0
```

**Task 41.2** On R2, when someone is trying to reload the router, the reload command should have no effect. It should trigger an EEM applet to check who is currently logged in and store the output of this command in the system flash in a file called reload_user. The EEM applet should also send the following syslog message: "Someone tried to reload the router R2".

On R2, configure the following:

```
event manager applet NO_RELOAD
 event cli pattern "reload" sync no skip yes
 action 1.0 info type routername
 action 2.0 cli command "enable"
 action 3.0 cli command "sh users | append system:reload_user"
 action 4.0 syslog msg "someone tried to reload the router $_info_routername"
```

You can use the action info type routername EEM applet command which sets the $_info_routername variable and use that variable in the action mail command.

**Task 41.3** On R6, when E0/1 is up, S3/0 has to be administratively shut down. When E0/1 is in a down state, S3/0 has to be enabled. Use 2 EEM applets to achieve this.

On R6, we are going to track the line protocol of the E0/1 interface with a track command.

On R6, configure the following:

```
track 1 interface Ethernet 0/1 line-protocol
```

```
event manager applet E01-up
event track 1 state up
action 1.0 cli command "enable"
action 1.1 cli command "config t"
action 1.2 cli command "int s3/0"
action 1.3 cli command "sh"
action 1.4 cli command "end"

event manager applet E01-down
event track 1 state down
action 1.0 cli command "enable"
action 1.1 cli command "config t"
action 1.2 cli command "int s3/0"
action 1.3 cli command "no sh"
action 1.4 cli command "end"
```

On R6, interface e0/1 is administratively shut down and s3/0 is in an up/up state. Let's unshut the interface e0/1. We can see that as soon as e0/1 is coming up, interface Serial3/0 is being administratively shut down by the EEM applet.

```
R6(config-if)#int e0/1
R6(config-if)#
R6(config-if)#no shut
R6(config-if)#
%TRACK-6-STATE: 1 interface Et0/1 line-protocol Down -> Up
R6(config-if)#
%SYS-5-CONFIG_I: Configured from console by  on vty0 (EEM:E01-up)
R6(config-if)#
%LINK-3-UPDOWN: Interface Ethernet0/1, changed state to up
R6(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to administratively down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to down
```

Now we are in a situation where interface e0/1 is in an up/up state and where s3/0 is in an administratively shut down state. Let's shut the interface e0/1. We can see that as soon as e0/1 is going down, interface Serial3/0 is administratively unshut by the EEM applet and is coming online..

```
R6(config-if)#int e0/1
R6(config-if)#shut
R6(config-if)#
%TRACK-6-STATE: 1 interface Et0/1 line-protocol Up -> Down
R6(config-if)#
%SYS-5-CONFIG_I: Configured from console by  on vty0 (EEM:E01-down)
R6(config-if)#
%LINK-5-CHANGED: Interface Ethernet0/1, changed state to administratively down
R6(config-if)#
%LINK-3-UPDOWN: Interface Serial3/0, changed state to up
R6(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
```

**Task 41.4** On R6, configure an EEM applet that is saving the configuration to NVRAM every hour. Each time the script is run; generate a syslog message stating "Configuration saved by EEM applet".

This applet will run every hour and a syslog will be fired up after the configuration has been saved by the EEM applet.

```
event manager applet SAVE_CONFIG_HOURLY
event timer watchdog time 360
action 1.0 cli command "enable"
action 1.1 cli command "wr mem"
action 1.2 syslog msg  "Configuration saved by EEM applet"
```

**Task 41.5** Configure the IP address 10.1.36.6 with a mask 255.255.0.0 on the interface E0/1 of R6. Do not modify this mask on the other side of the connection between R6 and R3. In the routing table of R6, there are only the connected networks. However, R6 is able to ping 10.1.35.5 with the ping sourced from IP address 10.1.36.6. On the interface of R3, disable the mechanism that makes this IP connectivity possible.

Let's try to ping from R6 to the destination 10.1.35.5. This ping is working.

```
R6#ping 10.1.35.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.35.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

We can verify that there is neither a route to the destination 10.1.35.0/24 network, nor a default route. How can the ping be successful? This is due to the fact proxy-arp is enabled by default on all the Ethernet interfaces. Because of the mask /16 on the interface E0/1 of R6, when pinging 10.1.35.5, the router R6 thinks that this destination is directly connected and ARP for the destination. The router R3 will receive a copy of this ARP request and will reply on behalf of 10.1.35.3 because it knows this network.

```
R6#sh ip route
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, 2 subnets, 2 masks
C        10.1.0.0/16 is directly connected, Ethernet0/1
L        10.1.36.6/32 is directly connected, Ethernet0/1
      172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C        172.16.6.0/24 is directly connected, Loopback0
L        172.16.6.6/32 is directly connected, Loopback0
```

We have been instructed to disable the proxy-arp mechanism on the interface of R3.

On R3, configure the following:

```
int e0/1
no ip proxy-arp
```

We can check that once proxy-arp is not enabled, the pings are failing.

**Task 41.6** On R2, make sure that the interface E0/1 is replying to all the ARP requests sent on the network 10.1.26.0/24.

When enabling local proxy ARP on the interface E0/1, the interface will reply for all ARP requests even if R2 has no routing towards the destination. Local proxy ARP requires that proxy ARP is active. ICMP redirects are disabled on an interface which is configured with local proxy ARP.

On R2, configure the following:

```
int e0/1
ip local-proxy-arp
```

**Task 41.7** Configure R3 as a DHCP server for the network 10.1.35.0/24 and 10.1.36.0/24. Default gateways are 10.1.35.1 and 10.1.36.1 respectively. The DNS server IP address is 10.2.2.2.

On R3, configure the following:

```
ip dhcp pool NET-35
network 10.1.35.0 255.255.255.0
default-router 10.1.35.1
dns-server 10.2.2.2

ip dhcp pool NET-36
network 10.1.36.0 255.255.255.0
default-router 10.1.36.1
dns-server 10.2.2.2
```

**Task 41.8** The IP address range 10.1.35.1-10.1.35.11 should be excluded from the IP addresses allocated to the clients by the server.

In order to avoid duplicate IP addresses, it makes sense that we have to exclude the range of addresses containing the IP addresses of the interfaces of the routers.

On R3, configure the following:

```
ip dhcp pool NET-35
ip dhcp excluded-address 10.1.35.1 10.1.35.11
```

**Task 41.9** The IP address range 10.1.36.1-10.1.36.11 should be excluded from the IP addresses allocated to the clients by the server.

In order to avoid duplicate IP addresses, it makes sense that we have to exclude the range of addresses containing the IP addresses of the interfaces of the routers.

On R3, configure the following:

```
ip dhcp pool NET-36
ip dhcp excluded-address 10.1.36.1 10.1.36.11
```

**Task 41.10** R3 will also be DHCP servers for the network 10.1.26.0/24. Default gateway is 10.1.26.1. The DNS server IP address is 10.2.2.2. Use static routing in order to enable routing between R2 and R3.

Let's first enable routing between R2 and R3.

On R2, configure the following:

```
ip route 10.1.36.0 255.255.255.0 10.1.26.6
```

On R3, configure the following:

```
ip route 10.1.26.0 255.255.255.0 10.1.36.6
```

Please note that you will not be able to unshut the ethernet 0/0 interface of R6 because of the /16 mask used on the interface ethernet 0/1. Leave it unshut.

Let's configure R3 as the DHCP server for the 10.1.26.0/24 network:

On R3, configure the following:

```
ip dhcp pool NET-26
network 10.1.26.0 255.255.255.0
default-router 10.1.26.1
dns-server 10.2.2.2
```

The DHCP clients on the network 10.1.26.0/24 will be sending broadcast DHCP packets. Those packets will have to be sent as unicast to the DHCP server.
On R6, configure the following:

```
int e0/0
ip helper-address 10.1.36.3
```

**Task 41.11** The IP address 10.1.35.100 should always be assigned to the server with the mac address aaaa.bbbb.cccc.

In order to create a static DHCP entry, we have to configure a new pool.
On R3, configure the following:

```
ip dhcp pool Static_35
host 10.1.35.100
hardware-address aaaa.bbbb.cccc
```

**Task 41.12** Configure R9 as a DHCP server for the network 10.1.79.0/24. Default gateway is 10.1.79.1. The DNS server IP address is 10.2.2.2. Exclude 10.1.79.1-11 from the DHCP range.

On R9, configure the following:

```
ip dhcp pool NET-79
network 10.1.79.0 255.255.255.0
default-router 10.1.79.1
dns-server 10.2.2.2
ip dhcp excluded-address 10.1.79.1 10.1.79.11
```

**Task 41.13** The interface E0/0 of R7 should retrieve an IP address from the DHCP pool configured earlier.

On R7, configure the following:

```
int e0/0
ip address dhcp
```

All the connections are on the same VLAN on the switches and we have to guarantee that the R7 is getting an IP address in the range 10.1.79.0. We are going to configure a VLAN 79 on the switch.

On Cat2, configure the following:

```
vlan 79

int e1/3
switchport mode access
switchport access vlan 79

int e2/1
switchport mode access
switchport access vlan 79
```

DHCP server on R9 has assigned an IP address to the interface e0/0 of R7.

```
R7#sh ip int brie
Interface                  IP-Address      OK? Method Status                Protocol
Ethernet0/0                10.1.79.12      YES DHCP   up                    up
```

**Task 41.14** On R9, configure AAA and Radius for DHCP accounting. The RADIUS server has IP address 10.2.2.2.

On R9, configure the following:

```
aaa new-model
aaa group server radius RADiPexpert
server 10.2.2.2 auth-port 1645 acct-port 1646

aaa accounting network RADIUS-GROUP start-stop group RADiPexpert
aaa session-id common

ip radius source-interface Ethernet0/0

radius-server host 10.2.2.2 auth-port 1645 acct-port 1646
radius-server attribute 31 send nas-port-detail mac-only
radius-server retransmit 3

ip dhcp pool NET-79
accounting RADIUS-GROUP
```

**You have completed Lab 41**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 42: Configure and Troubleshoot IP/IOS Services (Part 4)

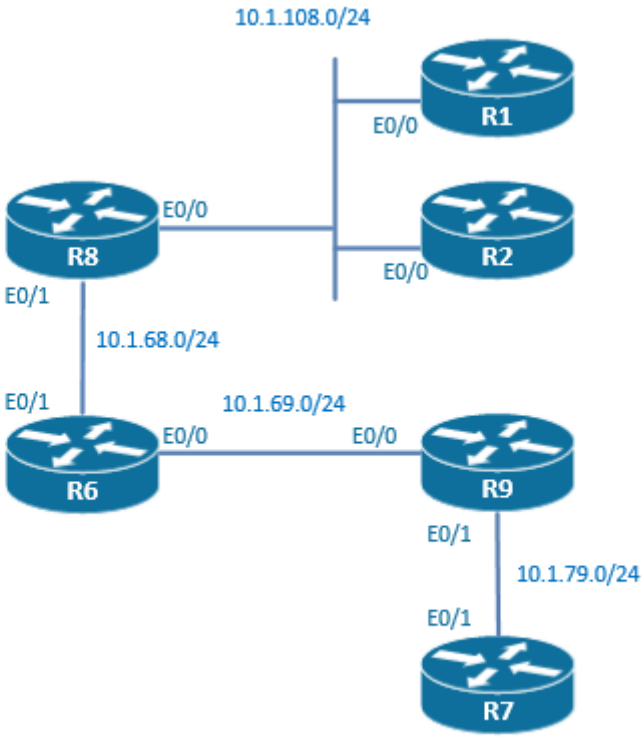
## Technologies covered

- IP SLA
- HSRP
- VRRP
- GLBP

## Overview

You have been tasked to configure management services in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 42.1** On the connection between R7 and R9, configure IP SLA on R7 to measure the UDP jitter. UDP packets should be sent to 10.1.79.9 port 3200 every 10 seconds with a DSCP marking of EF. This measurement should run indefinitely.

On R7, configure the following:

```
ip sla 1
udp-jitter 10.1.79.9 3200
tos 184
frequency 10
ip sla schedule 1 start-time now life forever
```

On R9, configure the following:

```
ip sla responder
```

**Task 42.2** When the connection between R7 and R9 is lost, R7 will send a trap and trigger a ping 10.1.79.9 every 5 seconds during 60 seconds. If the connectivity is not re-established after 60 seconds, a second trap will again be sent. Enable R7 to send CISCO-SYSLOG-MIB traps to the SNMP server 10.1.222.200 with the community iPexpert.

On R7, configure the following:

```
ip sla 44
icmp-echo 10.1.79.9
ip sla schedule 44 life forever start-time now
ip sla reaction-configuration 44 react timeout action-type trapandTrigger

ip sla 45
icmp-echo 10.1.79.9
frequency 5
ip sla schedule 45 life 60 start-time pending
ip sla reaction-configuration 45 45 react timeout action-type traponly

ip sla reaction-trigger 44 45

snmp-server community iPexpert ro
snmp-server enable traps syslog
snmp-server host 10.1.222.200 iPexpert
snmp-server host 10.1.222.200 iPexpert syslog
```

**Task 42.3** Between R6 and R9, configure on R6 an IP SLA job that will generate an ICMP echo with a packet size of 1000 bytes every 10 seconds. Those packets have to be sent to 10.1.69.9.

On R6, configure the following:

```
ip sla 2
icmp-echo 10.1.69.9
request-data-size 1000
frequency 10
ip sla schedule 2 start now
```

**Task 42.4** The IP SLA control messages between R6 and R9 have to be authenticated using key-chain called “iPexpert”. This key-chain should use key number 3 and a key string of “iPexpert”.

On R6, R9 and R7, configure the following:

```
key chain iPexpert
key 3
key-string iPexpert

ip sla key-chain iPexpert
```

**Task 42.5** Between R6 and R8, configure on R6 a TCP operation to 10.1.68.8 on port 443 that doesn't require R8 to be configured as a responder.

On R6, configure the following:

```
ip sla 443
tcp-connect 10.1.68.8 443 control disable
ip sla schedule 443 start-time now
```

**Task 42.6** Between R8 and R2, configure on R8 a TCP operation to 10.1.108.2 on port 80 that requires R2 to be configured as a responder.

On R2, configure the following:

```
ip sla responder
```

On R8, configure the following:

```
ip sla 80
tcp-connect 10.1.108.2 80
ip sla schedule 80 start-time now
```

**Task 42.7** Configure R8 to perform every 30 seconds a DNS lookup on the DNS server 10.1.222.222 for the website www.ipexpert.com.

On R8, configure the following:

```
ip sla 33
dns www.ipexpert.com name-server 10.1.222.222
frequency 30
ip sla schedule 33 start-time now
```

**Task 42.8** Configure GLBP between R8, R2, and R1 on the network 10.1.108.0/24. Virtual IP address is 10.1.108.133. 10% of the traffic should use R2 as a gateway and 10% of the traffic should use R1 as a gateway.

On R2 and R1, configure the following:

```
int e0/0
glbp 1 ip 10.1.108.133
glbp 1 load-balancing weighted
glbp 1 weighting 10
```

On R8, configure the following:

```
int e0/0
glbp 1 ip 10.1.108.133
glbp 1 load-balancing weighted
glbp 1 weighting 80
```

**Task 42.9** Authenticate the GLBP routers with a MD5 hashed password of " iPexpert133".

On R8, R2, and R1, configure the following:

```
int e0/0
glbp 1 authentication md5 key-string iPexpert133
```

**Task 42.10** Configure VRRP between R2 and R1 on the network 10.1.108.0/24. Virtual IP address is 10.1.108.144. When R2 is up and running, it should always be the master.

On R1, configure the following:

```
int e0/0
vrrp 20 ip 10.1.108.144
```

On R2, configure the following:

```
int e0/0
vrrp 20 ip 10.1.108.144
vrrp 20 preempt
vrrp 20 priority 200
```

**Task 42.11** Authenticate the VRRP routers with a password of “iPexpert”.

On R1 and R2, configure the following:

```
int e0/0
vrrp 20 authentication iPexpert
```

**Task 42.12** Configure HSRP between R8 and R2 on the network 10.1.108.0/24. Virtual IP address is 10.1.108.155. As long as R8 is up and running, it should stay the master and when an outage occurs, it should recover this role 1 minutes after coming back online.

On R8, configure the following:

```
interface e0/0
standby 1 ip 10.1.108.155
standby 1 priority 105
standby 1 preempt delay min 60
```

On R2, configure the following:

```
interface e0/0
standby 1 ip 10.1.108.155
```

**Task 42.13** When the ICMP echo from R8 to R6 is failing, the priority should be decreased the minimun in such a way that R2 is taking over the primary role.

On R8, configure the following:

```
ip sla 11
icmp-echo 10.1.68.6
track 10 ip sla 11 reachability
ip sla schedule 11 life forever start-time now
ip sla enable reaction-alerts

interface e0/0
standby 1 track 10 decrement 10
```

On R2, configure the following:

```
interface e0/0
standby 1 preempt
```

**Task 42.14** Authenticate the HSRP routers with a clear text password of “iPexpert”.

On R8 and R2, configure the following:

```
int e0/0
standby 1 authentication iPexpert
```

**You have completed Lab 42**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 43: Configure and Troubleshoot IP/IOS Services (Part 5)

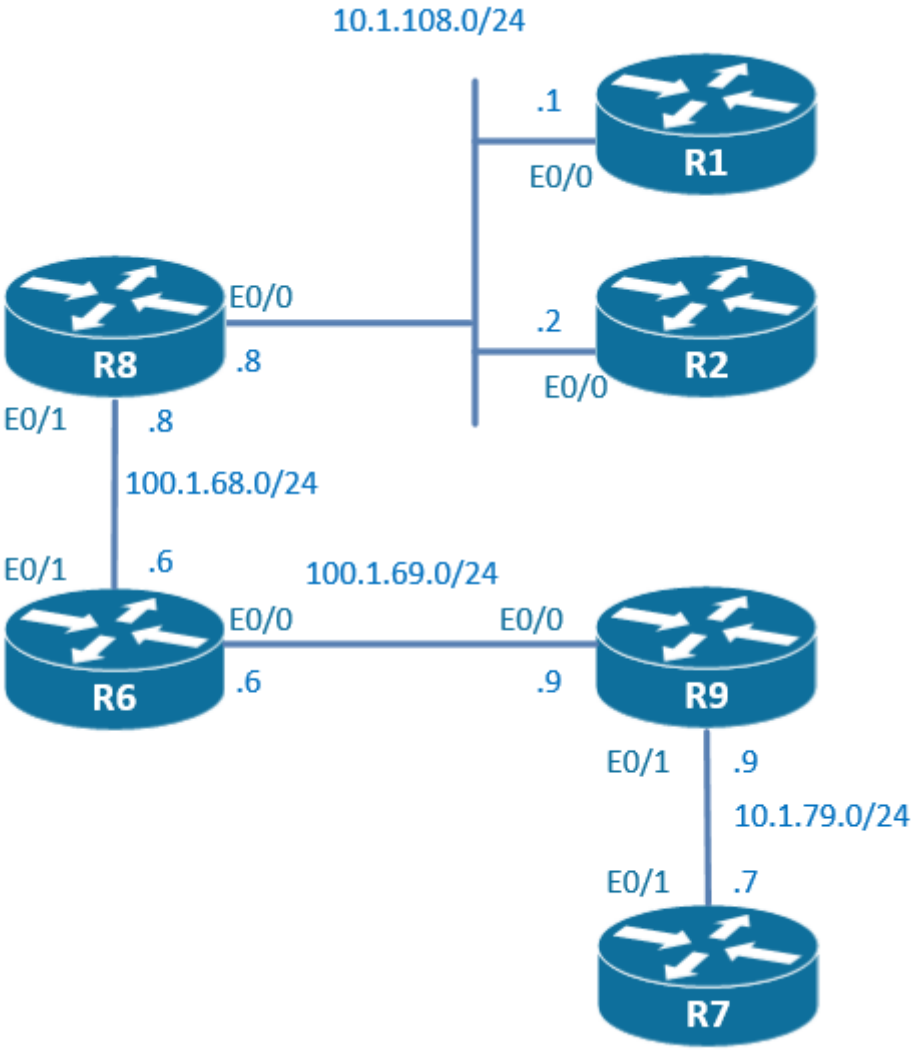
## Technologies covered

- NAT Overload
- NAT Route-maps
- Static NAT
- Static PAT
- NAT no alias
- NAT no payload
- Policy NAT

## Overview

You have been tasked to configure management services in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 2 hours**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 43.1** On R7, configure a default route towards R9. 10.1.79.0/24 is the inside network, 100.1.69.0/24 is the outside network. Make sure that the ping from R7 to 100.1.69.6 is successful using a static NAT between 100.1.69.20 and 10.1.79.7.

On R7, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.79.9
```

On R9, configure the following:

```
int E0/1
ip nat inside
int E0/0
ip nat outside

ip nat inside source static 10.1.79.7 100.1.69.20
```

On R7, I can ping the IP address 100.1.69.6 thanks to the NAT translation that takes place on R9.

```
R7#ping 100.1.69.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.69.6, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/2 ms

R9#sh ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
icmp 100.1.69.20:1     10.1.79.7:1        100.1.69.6:1       100.1.69.6:1
--- 100.1.69.20        10.1.79.7          ---                ---
```

**Task 43.2** We don't want R9 to respond to the ARP request for 100.1.69.20. Clear the ARP cache and verify that the ping from R7 to 100.1.69.6 is unsuccessful.

On R9, configure the following:

```
ip nat inside source static 10.1.79.7 100.1.69.20 no-alias
```

After clearing the ARP cache on R6, the ping is not working anymore because R9 is not answering for ARP requests for 100.1.69.20.

```
R6#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  100.1.68.6              -   aabb.cc00.0610  ARPA   Ethernet0/1
Internet  100.1.69.6              -   aabb.cc00.0600  ARPA   Ethernet0/0
Internet  100.1.69.9              2   aabb.cc00.0900  ARPA   Ethernet0/0
Internet  100.1.69.20            20   aabb.cc00.0900  ARPA   Ethernet0/0

R6#clear ip arp 100.1.69.20

R6#sh arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  100.1.68.6              -   aabb.cc00.0610  ARPA   Ethernet0/1
Internet  100.1.69.6              -   aabb.cc00.0600  ARPA   Ethernet0/0
Internet  100.1.69.9              6   aabb.cc00.0900  ARPA   Ethernet0/0
```

```
R7#ping 100.1.69.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.69.6, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

**Task 43.3** Ensure that the ping from R7 to 100.1.69.6 is again successfull by configuring a static ARP entry on the router R6.

On R6, configure the following:

```
arp 100.1.69.20 aabb.cc00.0900 arpa
```

The ping from R7 to 100.1.69.6 is working again.

```
R7#ping 100.1.69.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.69.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/5 ms
```

**Task 43.4** Ensure that the payload will not be modified by the static NAT entry configured on R9.

On R9, configure the following:

```
ip nat inside source static 10.1.79.7 100.1.69.20 no-alias no-payload
```

**Task 43.5** On R9, configure a loopback0 with an IP address of 10.1.9.9/24. 10.1.9.0/24 is the inside network, 100.1.69.0/24 is the outside network.

On R9, configure the following:

```
int lo0
ip address 10.1.9.9 255.255.255.0
ip nat inside
```

**Task 43.6** On R9, configure a dynamic NAT that maps the internal range 10.1.9.0/24 to the public addresse range 100.1.69.241-100.1.69.255. When no more address is available in the public range, a new connection will use a mapping of an already mapped public IP address with a different TCP port number.

On R9, configure the following:

```
access-list 90 permit 10.1.9.0 0.0.0.255
 ip nat pool nat-pool 100.1.69.241 100.1.69.255 netmask 255.255.255.0
 ip nat inside source list 90 pool nat-pool overload
```

**Task 43.7** On R9, configure a loopback1 with an IP address of 11.1.9.9/24. 11.1.9.0/24 is the inside network, 100.1.69.0/24 is the outside network.

On R9, configure the following:

```
int lo1
ip address 11.1.9.9 255.255.255.0
ip nat inside
```

**Task 43.8** On R9, configure a dynamic PAT that maps the internal range 11.1.9.0/24 to the interface E0/0.

On R9, configure the following:

```
access-list 91 permit 11.1.9.0 0.0.0.255
 ip nat inside source list 91 interface E0/0
```

**Task 43.9** On R9, enable the TCP small server service on TCP port 13 called “datetime”.

On R9, configure the following:

```
service tcp-small-servers
```

**Task 43.10** On R8, configure a default route towards R6. 100.1.68.0/24 is the inside network, 100.1.69.0/24 is the outside network. Make sure that the ping from R8 to 100.1.69.9 is successful and that the telnet 100.1.69.9 on port 4000 will return the daytime information.

On R8, configure the following:

```
ip route 0.0.0.0 0.0.0.0 100.1.68.6
```

On R6, configure the following:

```
int e0/1
ip nat inside

int e0/0
ip nat outside

ip nat inside source static 100.1.68.8 100.1.69.30
```

On R8, the telnet to the port 13 reponds the daytime.

```
R8#telnet 100.1.69.9 13
Trying 100.1.69.9, 13 ... Open
Wednesday, January 14, 2015 07:47:16-CET

[Connection to 100.1.69.9 closed by foreign host]
```

**Task 43.11** 10.1.108.0/24 is the inside network, 100.1.68.0/24 is the outside network. Make sure that the ping from R1 to 100.1.68.6 is successful without configuring a default route pointing to R8 on R1. You have to use the ip nat outside command on R8.

On R1, configure the following:

```
ip route 0.0.0.0 0.0.0.0 e0/0
```

On R8, configure the following:

```
int e0/1
ip nat outside

int e0/0
ip nat inside

ip nat inside source static 10.1.108.1 100.1.68.50
ip nat outside source static 100.1.68.6 10.1.108.50
```

**Task 43.12** Ensure that you can telnet from R1 to 10.1.68.6 by using the add-route keyword in a command.

On R8, configure the following:

```
ip nat outside source static 100.1.68.6 10.1.108.50 add-route
```

**Task 43.13** On R2, configure a default route towards R8. Traffic coming from R2 should be statically NATed to the IP address 100.1.68.20. Use a route-map to achieve this task. Verify that you can ping from R2 to 100.1.68.6.

On R2, configure the following:

```
ip route 0.0.0.0 0.0.0.0 10.1.108.8

ip access-list extended fromR2
permit ip 10.1.108.0 255.255.255.0 any

route-map fromR2 permit 10
match ip address fromR2

ip nat pool POOL 10.1.68.20 10.1.68.20 netmask 255.255.255.0
ip nat inside source route-map fromR2 pool POOL
```

**You have completed Lab 43**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 44: Configure and Troubleshoot IP/IOS Services (Part 6)

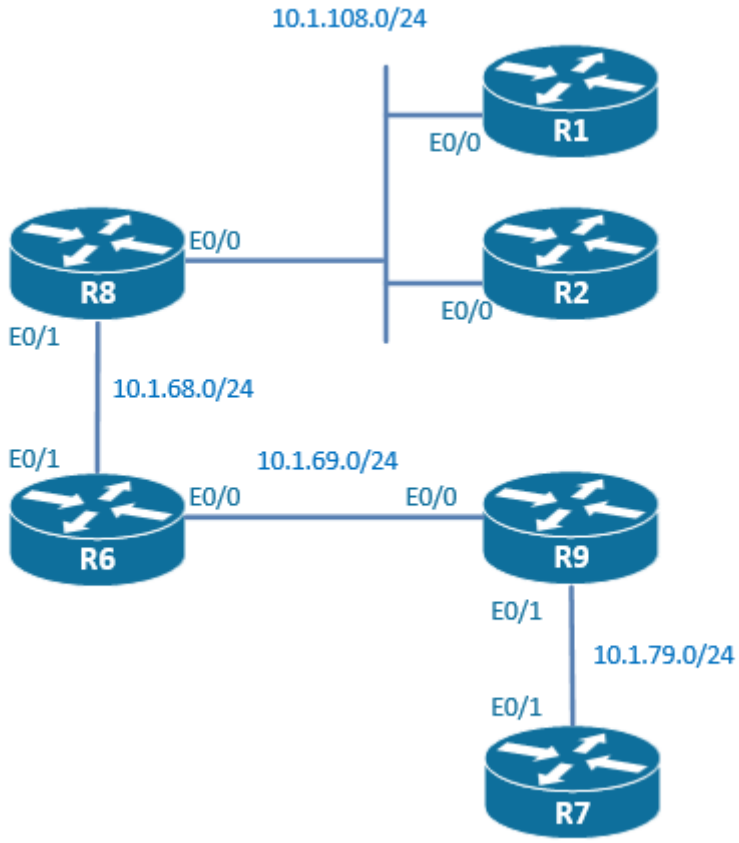
## Technologies covered

- IP precedence accounting
- IP output packet accounting
- IP access violation accounting
- MAC address accounting
- TCP optimization

## Overview

You have been tasked to configure management services in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 1 hour**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 44.1** On R7, perform on the E0/1 accounting based on IP precedence on received packets.

On R7, configure the following:

```
interface E0/1
ip accounting precedence input
```

**Task 44.2** Configure the following loopbacks:

| | |
|---|---|
| R8 loopback0 | 10.1.8.8/32 |
| R9 loopback0 | 10.1.9.9/32 |

On R8, configure the following:

```
int lo0
ip address 10.1.8.8 255.255.255.255
```

On R9, configure the following:

```
int lo0
ip address 10.1.9.9 255.255.255.255
```

**Task 44.3** Enable OSPF area 0 on the path between R8 and R9, and advertise the loopback0 of R8 and R9 using network statements.

On R8, configure the following:

```
router ospf 1
network 10.1.68.0 255.255.255.0 area 0
network 10.1.8.8 255.255.255.255 area 0
```

On R6, configure the following:

```
router ospf 1
network 10.1.68.0 255.255.255.0 area 0
network 10.1.69.0 255.255.255.0 area 0
```

On R9, configure the following:

```
router ospf 1
network 10.1.69.0 255.255.255.0 area 0
network 10.1.9.9 255.255.255.255 area 0
```

**Task 44.4** On R6, create an access-list to block traffic going from the loopback0 of R8 to the loopback0 of R9.

On R6, configure the following:

```
access-list 101 deny ip host 10.1.8.8 host 10.1.9.9
access-list 101 permit ip any any

int e0/0
ip access-group 101 out
```

**Task 44.5** Apply this access-list on the interface E0/0 and ensure that IP accounting displays the number of packets blocked by this access-list.

On R6, configure the following:

```
int e0/0
ip accounting output-packets
ip accounting access-violations
```

On R8, I perform a ping from 10.1.9.9 with a source of 10.1.8.8.

```
R8#ping 10.1.9.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R8#ping 10.1.9.9 source 10.1.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.9.9, timeout is 2 seconds:
Packet sent with a source address of 10.1.8.8
U.U.U
Success rate is 0 percent (0/5)
```

I can see that the packets denied by the ACL are logged by the IP accounting mechanism.

```
R6#sh ip accounting access-violations
   Source           Destination              Packets               Bytes   ACL
 10.1.8.8         10.1.9.9                       3                 300   101

Accounting data age is 8
```

**Task 44.6** On the interface E0/1 of R6, collect statistics about traffic per MAC address in the egress and ingress direction.

On R6, configure the following:

```
int e0/1
ip accounting mac-address input
ip accounting mac-address output
```

**Task 44.7** On R8, activate high performance TCP options as described in RFC 1323.

On R8, configure the following:

```
ip tcp timestamp
ip tcp window-size 75000
```

**Task 44.8** On R2, configure the outgoing TCP queue to contain a maximum of 10 packets.

On R2, configure the following:

```
ip tcp queuemax 10
```

**Task 44.9** On R2, activate the TCP connection to discover the minimum MTU size along the path of the TCP connection and therefore avoid fragmentation.

On R2, configure the following:

```
ip tcp path-mtu-discovery
```

**Task 44.10** R8 should wait for a maximum of 10 seconds to receive a TCP SYN.

On R8, configure the following:

```
ip tcp synwait-time 10
```

**Task 44.11** Make sure that R8 will not be affected by the “TCP silly window syndrome”.

On R8, configure the following:

```
ip tcp chunk-size 64000
```

**You have completed Lab 44**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.

# Lab 45: Configure and Troubleshoot IP/IOS Services (Part 7)

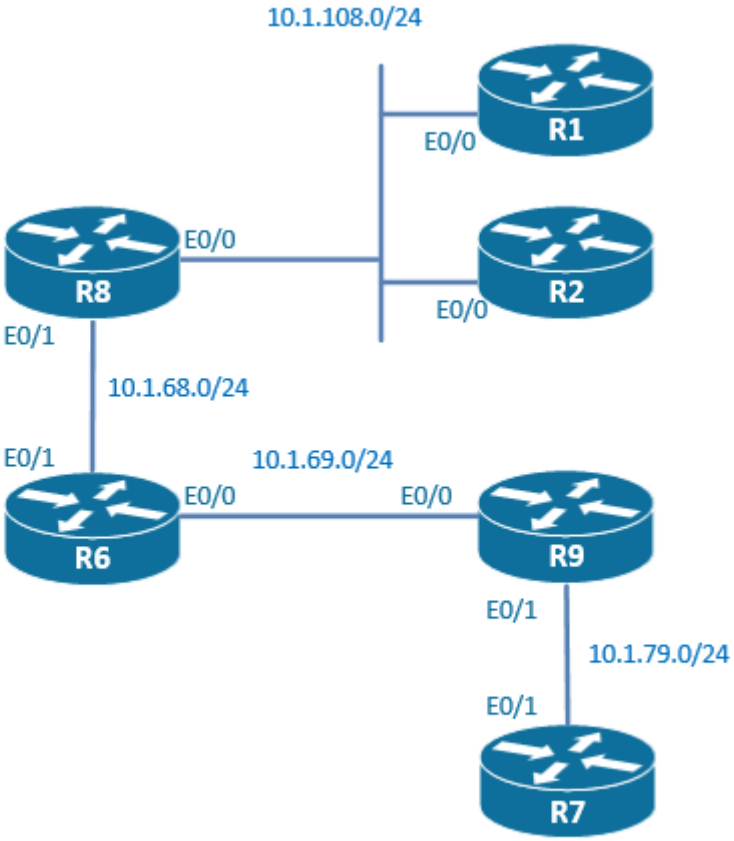
## Technologies covered

- Netflow ingress amd egress
- Netflow top talkers
- Netflow aggregation cache
- Netflow random sampling
- Netflow input filters

## Overview

You have been tasked to configure management services in your network.

The topology used in the lab will be the following:




**Estimated time to complete: 1 hour**

## Pre-Lab Setup

Logically connect and configure your network as displayed in the drawing below. You may also refer to the Diagram located within your configuration files for topology information.

This lab is intended to be used with online rack access provided by www.proctorlabs.com. Connect to the terminal server for the online rack, and complete the configuration tasks as detailed below.

## Prerequisites

Load the initial configuration files before starting to work on the tasks.

**Task 45.1** Setup R8 to collect Netflow version 9 statistics on E0/0 and E0/1, and to send them to server 10.1.33.33 on port 2333 in version 5 format. If R8 uses BGP, the peer AS should be included in exports. Make sure that the flows information is not duplicated.

On R8, configure the following:

```
int E0/0
ip flow ingress
int E0/1
ip flow ingress
ip flow-export version 5
ip flow-export destination 10.1.33.33 2333
ip flow-export ver 9 peer-as
```

**Task 45.2** Configure R8 to export flow records every 2 minutes.

On R8, configure the following:

```
ip flow-cache timeout active 2
```

**Task 45.3** On R8, ensure that a flow in the cache that was not refreshed during 10 seconds expires.

On R8, configure the following:

```
ip flow-cache timeout inactive 10
```

**Task 45.4** Setup R9 to collect Netflow version 9 statistics on E0/0 and E0/1, and to send them to server 10.1.33.33 on port 2333 in version 5 format. Only 1 out of 50 packets should be captured by Netflow. Ensure that it is not 1 every 50 packets which is captured but randomly 1 out of 50 packets. Make sure that the flows information is not duplicated.

On R9, configure the following:

```
ip flow-export version 5
ip flow-export destination 10.1.33.33 2333

flow-sampler-map 1OUTOF50
mode random one-out-of 50

int E0/0
flow-sampler 1OUTOF50

int E0/1
flow-sampler 1OUTOF50
```

**Task 45.5** On R6, configure Netflow on interface E0/1 and interface E0/0 to only capture traffic between 10.1.8.8 and 10.1.9.9. Only 1 out of 2 packets from this flow should be captured. Use a class-map called “NETFLOWCLASS” and a policy-map called “NETFLOWPOLICY”.

On R6, configure the following:

```
flow-sampler-map FILTERED_NETFLOW
```

```
mode random one-out-of 2

ip access-list extended ACL
permit ip host 10.1.8.8 host 10.1.9.9

class-map match-all NETFLOWCLASS
match access-group name ACL

policy-map NETFLOWPOLICY
class NETFLOWCLASS
netflow-sampler FILTERED_NETFLOW

int E0/0
service-policy input NETFLOWPOLICY
int E0/1
service-policy input NETFLOWPOLICY
```

**Task 45.6** On R1, configure Netflow version 9 on interface E0/0 to capture Netflow statistics in egress and ingress directions. The Netflow template should be sent every minute in version 9 to server 10.1.44.44.

On R1, configure the following:

```
int E0/0
ip flow ingress
ip flow egress

ip flow-export destination 10.1.44.44 4444
ip flow-export version 9
ip flow-export template timeout-rate 60
```

**Task 45.7** On R1, on the Netflow running on the E0/0, aggregate flow based of destination prefix present in the routing table. Never aggregate with a mask number lower than /24.

On R1, configure the following:

```
ip flow-aggregation cache destination-prefix
export destination 10.1.44.44 4444
mask destination minimum 24
export version 9
enabled
```

**Task 45.8** On R2, setup netflow to display in the command line the 20 top speakers going through interface E0/0. Sort the top speaker by bytes.

On R2, configure the following:

```
ip flow-top-talkers
top 20
sort-by bytes
```

**Task 45.9** On R2 interface E0/0, configure Netflow to capture the statistics for IPv6 packets.

On R2, configure the following:

```
int E0/0
ipv6 flow ingress
ipv6 flow egress

ipv6 flow-export version 9
```

**Task 45.10** On R7, configure Flexible Netflow to collect the source and destination IP address, the flow direction, the next-hop IP address using a flow record called “IPEXPERTRECORD”.

On R7, configure the following:

```
flow record IPEXPERTRECORD
match ipv4 source address
match ipv4 destination address
collect flow direction
collect routing next-hop address ipv4
```

**Task 45.11** On R7, configure Flexible Netflow to export statistics to the server 10.1.55.55 on port 3444 every 30 seconds using a flow exporter called “IPEXPERTEXPORTER”.

On R7, configure the following:

```
flow exporter IPEXPERTEXPORTER
destination 10.1.55.55
transport udp 3444
template data timeout 30
```

**Task 45.12** Apply a flow monitor called “IPEXPERTMONITOR” in the ingress and egress direction on interface E0/1.

On R7, configure the following:

```
flow monitor IPEXPERTMONITOR
record IPEXPERTRECORD
exporter IPEXPERTEXPORTER

interface E0/1
ip flow monitor IPEXPERTMONITOR input
ip flow monitor IPEXPERTMONITOR output
```

**You have completed Lab 45**

For verification of your work, please refer to this Workbook's accompanying Detailed Solutions Guide. If you need assistance with any of this book's content, please visit our Member Community at http://community.ipexpert.com.