360RSW05-WKB-TA05

Cisco 360 CCIE R&S Exercise Workbook Introduction

The Cisco 360 CCIE® R&S Exercise Workbook contains 20 challenging scenarios at the Cisco CCIE level that can be used for rigorous self-paced practice.

Each lab provides an extensive answer key, Mentor Guide support, and verification tables and is designed to maximize learning by providing practical experience. Also, self-paced learning resources such as the Cisco 360 CCIE R&S Reference Library and Cisco 360 CCIE R&S lessons supplement the Exercise Workbook scenarios.

Cisco 360 CCIE R&S Exercise Workbook Lab 5 Troubleshooting Section Answer Key

Table of Contents

Answer Key Structure

Section One

The answer key PDF document is downloadable from the web portal.

Section Two

To obtain a comprehensive view of the configuration for a specific section, access the Mentor Guide engine in the web portal.

Exercise Workbook Lab 5 Troubleshooting Section Answer Key

Note	Regardless of any configuration you perform in this lab, it is very important that you conform to the general guidelines that are provided in the "Restrictions and Goals" section. If you do not conform to the guidelines, you could have a significant deduction of points in your final score.

Grading and Duration

- Troubleshooting lab duration: 2 hours
- Troubleshooting lab maximum score: 24 points

Note	You can assess your progress on the self-paced labs in this workbook by adding up the points that are assigned to sections and tasks. Consider taking the full Assessment Labs to assess your readiness level.

Difficulty Level

- Difficulty: Intermediate

Restrictions and Goals

Note	Read this section carefully.

- To receive any credit for a subsection, you must fully complete the subsection as per the requirements. You will *not* receive partial credit for partially completed subsections.
- IPv4 subnets that are displayed in the scenario diagram must be used. Points will be deducted from multiple sections for failing to assign correct IPv4 addresses.
- Advertise loopback interfaces with their original masks.
- All IP addresses involved in this scenario must be reachable, unless explicitly specified otherwise.
- Unless explicitly specified otherwise, addresses and networks that are advertised in the BGP section need to be reachable by all BGP routers but do not have to be reachable by routers that use only IGP.
- Use conventional routing algorithms only, unless specified otherwise.
- Do not create new interfaces to fulfill IGP requirements, and do not summarize unless explicitly asked to do so.

- Do not modify the hostname, console, or vty configuration unless you are specifically asked to do so.
- Do not modify the initial interface or IP address numbering.

Explanation of Each of the Restrictions and Goals

IPv4 subnets that are displayed in the "Lab IPv4 IGP" diagram must be used.

All IPv4 IP addresses in this lab use a second octet that is equal to the default administrative distance value of the IGP routing protocol that the address is assigned to.

Advertise loopback interfaces with their original masks.

The original mask is the mask configured on the loopback interface. Open Shortest Path First (OSPF), by default, treats loopback interfaces as host routes and advertises them as /32 prefixes. The requirement to advertise loopback interfaces with their original masks precludes using the default OSPF network type for the loopback interfaces. You must provide a solution such as changing the OSPF network type or summarizations.

All IP addresses involved in this scenario must be reachable.

This goal is a key goal to observe. It requires that all your IGPs and routing policy tasks must be configured properly. The key elements of your routing policy include route redistribution and the controlling of routing updates using the **distribute-lists**, **route-maps**, and **distance** commands. A key point to remember about this lab is that the term "redistribution" is not explicitly used. However, you must perform redistribution to ensure that all IP addresses are reachable without the use of static routes or 0.0.0.0/0 routes.

Addresses and networks that are advertised in the BGP section need to be reachable by all BGP routers but do not have to be reachable by IGP-only routers.

This statement relaxes the requirement that all IP addresses must be reachable. The BGP prefixes need only be reachable among the routers specified in the BGP section. They can be used in other unicast tables. However, BGP routers need to have the prefixes in the routing tables as well as be able to forward traffic to the addresses known via BGP.

Use conventional routing algorithms.

This restriction prevents you from solving any problems by configuring policy routing. At the heart of this restriction is the interpretation of "conventional routing algorithms." Although this phrase can be interpreted in several different ways, the following interpretation is applied in this workbook:

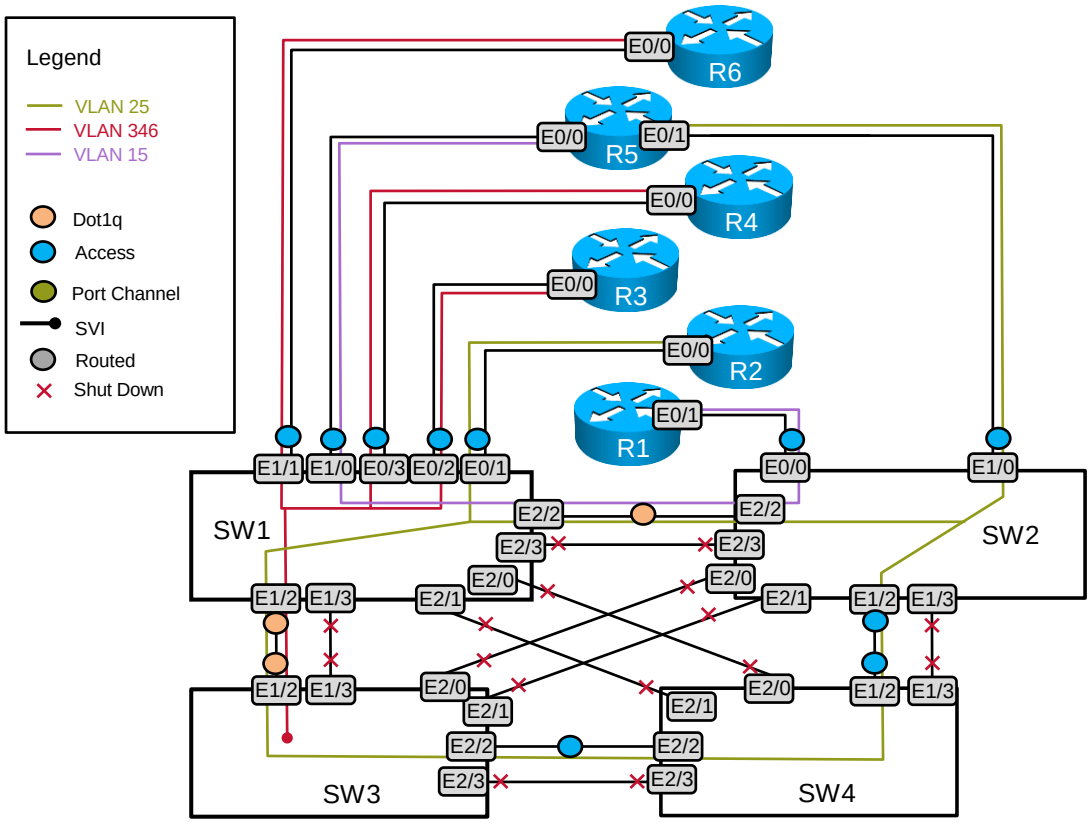
> Conventional routing algorithms are routing algorithms that apply destination-based prefix lookups in a routing table. Conventional routing algorithms do not use any type of information other than the destination address to make a packet-forwarding decision.

Because of this restrictive interpretation, no form of policy routing can be applied. Whenever you see this restriction, you will need to use dynamic routing protocols to fulfill all packet-forwarding requirements.

1. Switched Network Troubleshooting Section

The VLAN propagation diagram is reproduced here for easy reference.

VLAN Propagation Diagram



1.1. Symptom: You cannot ping the 192.168.1.2 address.

Analysis and testing:

According to the requirements of the scenario, VLAN 346 should be provisioned on SW3.

```
SW3#show  vlan brie

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Et0/0, Et0/1, Et0/2, Et0/3
                                                 Et1/0, Et1/1, Et1/3, Et2/0
                                                 Et2/1, Et2/3
25   VLAN0025                          active    Et2/2
```



```
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW3#
```

As you can see, only VLAN 25 is provisioned on SW3.

```
SW3#sh ip inte brief | include Vlan346
Vlan346                192.168.1.2     YES manual down                  down
```

Note that the VLAN 346 interface is in a down/down state. A common reason for this condition is that the VLAN itself is not provisioned on the switch.

Likely cause: VLAN 346 is not configured on SW3.

This issue is clearly indicated by the Cisco IOS **show** commands that are displayed in the following output:

```
SW3#show running-config | inc vlan +[0-9]+
vlan 25
 switchport trunk allowed vlan 25,346
 switchport access vlan 25
SW3#
```

Note that only VLAN 25 is configured on SW3.

Resolution: Configure VLAN 346 on SW3.

This problem can be resolved by entering the command **vlan 346** in global configuration mode on SW3. Next, verify this configuration with the following **show** command:

```
SW3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW3(config)#vlan 346
SW3(config-vlan)#end
SW3#
SW3#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et0/0, Et0/1, Et0/2, Et0/3
                                                Et1/0, Et1/1, Et1/3, Et2/0
                                                Et2/1, Et2/3
25   VLAN0025                         active    Et2/2
346  VLAN0346                         active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW3#
```

You now see that the VLAN 346 interface is an up/up state:

```
SW3#show ip interface brief | include Vlan346
Vlan346                192.168.1.2     YES manual up                    up
```

1.2. Symptom: Additional VLANs are transiting the trunk link between SW1 and SW3.

Analysis and testing:

The scenario states that you must limit the trunk links to permit only the required VLANs. Here you see output from the command **show interfaces trunk** for SW1, SW2, and SW3:

```
SW1#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Et1/2       on               802.1q         trunking      1
Et2/2       on               802.1q         trunking      1

Port        Vlans allowed on trunk
Et1/2       1-4094
Et2/2       15,25

Port        Vlans allowed and active in management domain
Et1/2       1,15,25,346
Et2/2       15,25

Port        Vlans in spanning tree forwarding state and not pruned
Et1/2       1,15,25,346
Et2/2       15,25
SW1#
SW2#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Et2/2       desirable        n-802.1q       trunking      1

Port        Vlans allowed on trunk
Et2/2       15,25

Port        Vlans allowed and active in management domain
Et2/2       15,25

Port        Vlans in spanning tree forwarding state and not pruned
Et2/2       15,25
SW2#
SW3#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Et1/2       desirable        n-802.1q       trunking      1

Port        Vlans allowed on trunk
Et1/2       1-4094

Port        Vlans allowed and active in management domain
Et1/2       1,25,346

Port        Vlans in spanning tree forwarding state and not pruned
Et1/2       1,25,346
SW3#
```

As the "Lab Ethernet Topology" diagram shows, the trunk link between SW1 and SW2 permits only VLANs 15 and 25, as required. The trunk link between SW1 and SW3 should permit only VLANs 25 and 346; however, it permits all VLANs.

Likely cause: No VLAN pruning is being performed on the trunk links between SW1 and SW3.

By default, trunk ports allow all VLANs to be transported across a trunk link. If this behavior needs to be changed, some type of VLAN pruning must be configured.

Two options exist for VLAN pruning:

- Static VLAN pruning with the **switchport trunk allowed vlan** command
- Dynamic VLAN pruning with the VLAN Trunking Protocol (VTP) pruning method

Resolution: Configure VLAN pruning on the trunk links between SW1 and SW3.

Since the switches must be in VTP transparent mode, VTP pruning is not an option. Use the manual VLAN pruning method by configuring the **switchport trunk allowed vlan** command on the E1/2 ports on both SW1 and SW3. Here are the final running configurations of the trunk ports between SW1 and SW3 after the appropriate changes have been added for VLAN pruning:

```
SW1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#int e1/2
SW1(config-if)#switchport trunk allowed vlan 25,346
SW1(config-if)#end
SW1#

SW1#show interfaces e1/2 trunk

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

Port        Vlans allowed on trunk
Et1/2       25,346

Port        Vlans allowed and active in management domain
Et1/2       25,346

Port        Vlans in spanning tree forwarding state and not pruned
Et1/2       25,346
SW1#

SW3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW3(config)#int e1/2
SW3(config-if)# switchport trunk allowed vlan 25,346
SW3(config-if)#end
SW3#
SW3#show interfaces e1/2 trunk

Port        Mode             Encapsulation  Status        Native vlan
Et1/2       desirable        n-802.1q       trunking      1

Port        Vlans allowed on trunk
Et1/2       25,346

Port        Vlans allowed and active in management domain
Et1/2       25,346

Port        Vlans in spanning tree forwarding state and not pruned
Et1/2       25,346
SW3#
```

1.3. Symptom: VLAN 15 resides on SW4.

Analysis and testing:

According to the requirements of the scenario, only the necessary VLANs should be provisioned, one on each switch. However, SW4 appears to have one VLAN that it does not need:

```
SW4#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et0/0, Et0/1, Et0/2, Et0/3
                                                Et1/0, Et1/1, Et1/3, Et2/0
                                                Et2/1, Et2/3
15   VLAN0015                         active
25   VLAN0025                         active    Et1/2, Et2/2
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW4#
```

Resolution: Remove VLAN 15 from SW4.

This change can be accomplished in the following manner:

```
SW4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW4(config)#no vlan 15
SW4(config)#end
SW4#
```

Next, verify this configuration with the following **show** command:

```
SW4#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et0/0, Et0/1, Et0/2, Et0/3
                                                Et1/0, Et1/1, Et1/3, Et2/0
                                                Et2/1, Et2/3
25   VLAN0025                         active    Et1/2, Et2/2
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW4#
```

Note The Mentor Guide engine in the web portal can help you use Cisco IOS Software commands to see a comprehensive view of the configuration for a specific section. With the Mentor Guide engine, you can enter more than 1000 Cisco IOS Software commands as well as a collection of proprietary commands such as **show all**.

Note To learn more about Cisco Catalyst switch troubleshooting methods and techniques, download and watch the VoD sessions from the Cisco 360 “Troubleshooting” lesson module. This lesson module contains more than 8 hours of video content that is dedicated to the subject of troubleshooting.

2. IPv4 OSPF Troubleshooting Section

2.1. Symptom: R6 has no OSPF neighbors.

Analysis and testing:

Router R6 is configured as an OSPF-speaking router in Area 0 sharing VLAN 346 with R3 and R4. However, R6 cannot form any OSPF neighbor relationships with R3 and R4. Begin by performing some basic analysis, issuing the **show ip ospf interface** command on each of these three routers. Doing so will troubleshoot both ends of the OSPF neighbor relationship:

```
R6#show ip ospf int e0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.1.6/29, Area 0, Attached via Network Statement
  Process ID 1, Router ID 192.168.1.8, Network Type BROADCAST, Cost: 10
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           10        no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.1.8, Interface address 192.168.1.6
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    No Hellos (Passive interface)
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
R6#
```

You can clearly see that OSPF is configured on the R6 Ethernet0/0 interface because this OSPF interface output is generated. However, even though OSPF is configured on the interface, this display indicates that no neighbors have been discovered. Now run the same command on routers R3 and R4:

```
R3#show ip ospf int e0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.1.3/29, Area 0, Attached via Network Statement
  Process ID 1, Router ID 10.10.0.3, Network Type BROADCAST, Cost: 10
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           10        no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 10.10.0.3, Interface address 192.168.1.3
  Backup Designated router (ID) 10.10.0.4, Interface address 192.168.1.4
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:03
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 3
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 10.10.0.4  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
R3#
```

This output shows that R3 clearly sees R4 but not R6.

```
R4#show ip ospf int e0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.1.4/29, Area 0, Attached via Network Statement
  Process ID 1, Router ID 10.10.0.4, Network Type BROADCAST, Cost: 10
```

```
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           10        no          no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 10.10.0.3, Interface address 192.168.1.3
  Backup Designated router (ID) 10.10.0.4, Interface address 192.168.1.4
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:05
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 5
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 10.10.0.3  (Designated Router)
  Suppress hello for 0 neighbor(s)
R4#
```

R4 sees R3 but does not see R6. The output makes it clear that all OSPF-enabled interfaces on this segment share the same OSPF network type and OSPF timers.

Likely cause: There is a misconfigured OSPF parameter on router R6.

R3 and R4 can form a successful OSPF neighbor relationship with each other, which indicates that there is a problem on router R6. Review the running configuration of all OSPF commands on router R6. Start with the following **show** command:

```
R6#sh running-config | section router ospf
router ospf 1
 passive-interface default
 network 192.168.1.0 0.0.0.255 area 0
R6#
```

Now the problem is evident. All the OSPF interfaces on R6 are configured to be passive. This configuration will definitely prevent an OSPF interface from forming a neighbor relationship.

Resolution: Remove the R6 OSPF E0/0 interface from passive-interface mode.

The resolution to this issue is remove the R6 Ethernet0/0 interface from the passive state:

```
R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R6(config)#router ospf 1
R6(config-router)#no passive-interface e0/0
R6#
*May 22 19:02:38.441: %OSPF-5-ADJCHG: Process 1, Nbr 10.10.0.3 on Ethernet0/0 from
LOADING to FULL, Loading Done
*May 22 19:02:38.441: %OSPF-5-ADJCHG: Process 1, Nbr 10.10.0.4 on Ethernet0/0 from
LOADING to FULL, Loading Done
R6#
*May 22 19:02:39.798: %SYS-5-CONFIG_I: Configured from console by console
(cierswbv5-te-lab05-sc, SJ)
R6#
```

As you can see, within seconds, R6 forms an OSPF adjacency with routers R3 and R4.

This OSPF neighbor relationship can be verified with the following Cisco IOS **show** command:

```
R6#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
```



```
10.10.0.3         1   FULL/DR         00:00:39    192.168.1.3     Ethernet0/0
10.10.0.4         1   FULL/BDR        00:00:38    192.168.1.4     Ethernet0/0
R6#
```

2.2. Symptom: R1 neighbors are stuck in an OSPF exstart adjacency formation state.

Analysis and testing:

The router R1 E0/0 interface is stuck in the OSPF exstart adjacency formation state:

```
R1#sh ip ospf neighbor detail
 Neighbor 172.16.1.1, interface address 10.10.3.1
    In the area 10 via interface Ethernet0/0
    Neighbor priority is 1, State is EXSTART, 3 state changes
    DR is 10.10.3.1 BDR is 10.10.3.0
    Options is 0x12 in Hello (E-bit, L-bit)
    Options is 0x12 in DBD (E-bit, L-bit)
    LLS Options is 0x1 (LR)
    Dead timer due in 00:00:35
    Neighbor is up for 00:00:47
    Index 0/0, 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
    Number of retransmissions for last database description packet 9
R1#
```

Eventually, the following Cisco IOS console message is displayed if all Cisco IOS console messages are being logged to the console:

```
May 22 19:02:20.022: %OSPF-5-ADJCHG: Process 10, Nbr 172.16.1.1 on Ethernet0/0 from
DOWN to DOWN, Neighbor Down: Ignore timer expired
```

You can further investigate this condition with the following Cisco IOS **debug** utility:

```
R1#debug ip ospf adj

OSPF adjacency events debugging is on

R1#

May 22 19:08:50.022: OSPF-10 ADJ    Et0/0: Rcv DBD from 172.16.1.1 seq 0x858 opt
0x52 flag 0x7 len 32  mtu 1500 state EXSTART
May 22 19:08:50.022: OSPF-10 ADJ   Et0/0: Nbr 172.16.1.1 has larger interface MTU
R1#
```

According to this debug output, a neighbor of R1, in this case R7, has a larger configured maximum transmission unit (MTU) size for this interface. R7 is reporting an MTU size of 1500 bytes. Therefore, R1 must have a smaller configured MTU size.

Likely cause: The R1 MTU size is smaller than the R7 MTU size.

For an OSPF adjacency to fully form, many parameters, including the OSPF MTU size, must match among neighbors. The debug output displayed previously indicates that R1 has a smaller MTU size than its neighbor R7. Check the configuration of R1:

```
R1#sh run interface E0/0
interface Ethernet0/0
ip address 10.10.3.0 255.255.255.254
 ip mtu 1486
```

```
end
```

The MTU size is further confirmed by the following Cisco IOS **show** command:

```
R1#show ip interface E0/0 | i MTU
  MTU is 1486 bytes
```

The source of the problem is that the R1 Ethernet interface has a different MTU size than its neighbor.

Resolution: Set the R1 MTU size to match the MTU size of its neighbor.

The default Ethernet MTU size is 1500 bytes, so the R1 Ethernet0/0 interface can be set to 1500 bytes by removing the currently configured **ip mtu 1486** command:

```
R1#conf t
R1(config)#inte E0/0
R1(config-if)#no ip mtu 1486

Aug 18 21:56:57.148: %OSPF-5-ADJCHG: Process 10, Nbr 172.16.1.1 on Ethernet0/0 from
LOADING to FULL, Loading Done
```

As you can see, within seconds of removing the **ip mtu 1486** command, R1 forms an OSPF adjacency with SW3.

This OSPF neighbor relationship can be verified with the following Cisco IOS **show** command:

```
R1#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
172.16.1.1        1   FULL/DR         00:00:37    10.10.3.1       Ethernet0/0
```

Note To learn more about OSPF troubleshooting methods and techniques, download and watch the VoD sessions from the Cisco 360 “Troubleshooting” lesson module. This lesson module contains more than 8 hours of video content that is dedicated to the subject of troubleshooting.

3. EIGRP Troubleshooting Section

3.1. Symptom: R5 is not forming any EIGRP neighbor relationships.

Analysis and testing:

Routers R1, R2, and R5, which speak EIGRP, are able to ping each other on VLANs 15 and 25 in a topology similar to a hub and spoke, where R5 is the hub and R1 and R2 are the spokes. However, R1 and R2 cannot form an EIGRP neighbor relationship with R5. Begin by using the basic Cisco IOS troubleshooting and verification tools for EIGRP. Start with the following Cisco IOS **show** command on R5:

```
R5#sh ip eigrp interfaces detail
EIGRP-IPv4 Interfaces for AS(10)
                              Xmit  Queue     PeerQ             Mean     Pacing  Time
Multicast    Pending
Interface               Peers   Un/Reliable   Un/Reliable   SRTT    Un/Reliable    Flow
Timer   Routes
Et0/0                     0        0/0        0/0            0        0/0             0
0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
```



```
  Next xmit serial <none>
  Packetized sent/expedited: 0/0
  Hello's sent/expedited: 1346/1
  Un/reliable mcasts: 0/0  Un/reliable ucasts: 0/0
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is not set
Et0/1                     0        0/0       0/0           0       0/0             0
0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 0/0
  Hello's sent/expedited: 1345/1
  Un/reliable mcasts: 0/0  Un/reliable ucasts: 0/0
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is not set
Lo105                     0        0/0       0/0           0       0/0             0
0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 0/0
  Hello's sent/expedited: 0/1
  Un/reliable mcasts: 0/0  Un/reliable ucasts: 0/0
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is not set
R5#
```

This command clearly shows that EIGRP is configured on the correct interfaces on R5. Next, run the same command on routers R1 and R2:

```
R1#sh ip eigrp interfaces detail
EIGRP-IPv4 Interfaces for AS(10)
                              Xmit  Queue     PeerQ             Mean     Pacing  Time
Multicast    Pending
Interface               Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow
Timer   Routes
Et0/1                     0        0/0       0/0           0       0/0             0
0
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 0/0
  Hello's sent/expedited: 1383/1
  Un/reliable mcasts: 0/0  Un/reliable ucasts: 0/0
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is not set
R1#

R2#sh ip eigrp interfaces detail
EIGRP-IPv4 Interfaces for AS(10)
                              Xmit  Queue     PeerQ             Mean     Pacing  Time
Multicast    Pending
Interface               Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow
Timer   Routes
Et0/0                     0        0/0       0/0           0       0/0             0
0
```

```
  Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 0/0
  Hello's sent/expedited: 1384/1
  Un/reliable mcasts: 0/0  Un/reliable ucasts: 0/0
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is not set
R2#
```

Both R1 and R2 also have EIGRP configured on the correct interfaces. It also must be noted that no authentication is configured on any of these EIGRP speakers. Therefore, there are no authentication issues with this particular configuration.

Likely cause: There is a basic parameter mismatch between the spoke routers R1 and R2 and the hub router R5.

Enable the **debug eigrp packet** utility on R5 to get more detailed diagnostic information:

```
R5#
*May 22 19:53:59.597: EIGRP: Received HELLO on Et0/1 - paklen 20 nbr 192.168.1.18
*May 22 19:53:59.597:   AS 10, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
*May 22 19:53:59.598: EIGRP: Ignore multicast Hello Et0/1 192.168.1.18
*May 22 19:53:59.790: EIGRP: Sending HELLO on Et0/0 - paklen 20 nbr 192.168.1.16
*May 22 19:53:59.790:     AS 10, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ
un/rely 0/0
*May 22 19:54:00.022: EIGRP: Received HELLO on Et0/0 - paklen 20 nbr 192.168.1.16
*May 22 19:54:00.022:   AS 10, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
*May 22 19:54:00.022: EIGRP: Ignore multicast Hello Et0/0 192.168.1.16
R5#
```

When you examine this output on R5, you see that R5 is receiving multicast EIGRP packets from both R1 and R2 on its E0/0 and E0/1 interfaces. For some reason, R5 is ignoring these multicast EIGRP hello packets. It appears that R5 is configured to unicast EIGRP packets, and routers R1 and R2 are configured to multicast EIGRP packets. To confirm this issue, enable the **debug eigrp packet** utility on R1 and R2 to get more detailed diagnostic information:

```
R1#
May 22 20:03:59.924: EIGRP: Received HELLO on Et0/1 - paklen 20 nbr 192.168.1.17
May 22 20:03:59.924:   AS 10, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
May 22 20:03:59.924: EIGRP: Ignore unicast Hello from Ethernet0/1 192.168.1.17
R1#
R2#
May 22 20:00:35.990: EIGRP: Received HELLO on Et0/0 - paklen 20 nbr 192.168.1.19
May 22 20:00:35.990:   AS 10, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
May 22 20:00:35.990: EIGRP: Ignore unicast Hello from Ethernet0/0 192.168.1.19
R2#
```

Indeed, R1 and R2 are configured to multicast EIGRP packets and R1 and R2 ignore the unicast packets arriving from R5. Clearly, there is an EIGRP address type mismatch. For EIGRP peers to exchange all EIGRP packets, the peers must either both be unicasting or multicasting.

Resolution: Set all EIGRP-speaking routers to multicast EIGRP packets.

Either all neighbors on the same subnet should use multicast or all should use unicast. The trouble can be fixed by removing the neighbor statements configured under the R5 EIGRP routing process. When these statements are removed, R5 will stop unicasting EIGRP packets and

begin multicasting EIGRP packets. Then all EIGRP speakers should form the expected neighbor relationships:

```
R5#sh run | section eigrp
router eigrp 10
 network 192.168.1.0
 auto-summary
 neighbor 192.168.1.16 Ethernet0/0
 neighbor 192.168.1.18 Ethernet0/1
```

Here are the two neighbor statements that need to be removed:

```
R5#conf t
R5(config)#router eigrp 10
R5(config-router)#no neighbor 192.168.1.16 Ethernet0/0
R5(config-router)#no neighbor 192.168.1.18 Ethernet0/1
R5(config-router)#end
R5#
```

As you can see, as soon as these statements are removed, the EIGRP neighbor relationship is successfully formed:

```
R5#show ip eigrp neighbors
IP-EIGRP neighbors for process 10
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
1   192.168.1.18            E0/1             10 00:23:26    3   200  0  4
0   192.168.1.16            E0/0             12 00:23:33    3   200  0  3
```

Note	To learn more about EIGRP troubleshooting methods and techniques, download and watch the VoD sessions from the Cisco 360 “Troubleshooting” lesson module. This lesson module contains more than 8 hours of video content that is dedicated to the subject of troubleshooting.

4. IPv4 RIP Troubleshooting Section

4.1. Symptom: One RIP route that should be included within a summary is being advertised.

Analysis and testing:

Both R1 and R2 are configured with a RIPv2 summary command:

```
R1#show run int e0/2
Building configuration...

Current configuration : 141 bytes
!
interface Ethernet0/2
 ip address 10.10.10.1 255.255.255.248
 ip pim sparse-mode
 ip summary-address rip 192.168.1.16 255.255.255.248
end

R1#
R2#show run int e0/2
Building configuration...

Current configuration : 141 bytes
!
interface Ethernet0/2
```

```
 ip address 10.10.10.2 255.255.255.248
 ip pim sparse-mode
 ip summary-address rip 192.168.1.16 255.255.255.248
end

R2#
```

However, despite this summary statement, one RIP route is still being leaked to other RIP speaker, R4:

```
R4#sh ip route rip | include 192.168.*/32
R        192.168.1.25/32 [120/3] via 10.10.10.2, 00:00:20, Ethernet0/2
R4#
```

Likely cause: A RIPv2 summary command is configured to be too narrow.

When you examine the RIPv2 summary statements on both R1 and R2, you can see that the prefix and prefix mask length that are specified exclude the 192.168.1.25/32 address.

When the specific routes are listed together, this exclusion can be clearly seen:

```
192.168.1.16/31
192.168.1.17/31
192.168.1.18/31
192.168.1.19/31
192.168.1.25/32
```

When the current summary of 192.168.1.16 mask 255.255.255.248 is applied, this summary applies to only the address block of 192.168.1.16 – 192.168.1.23. The problematic 192.168.1.25/32 address is just outside of this address block.

Resolution: Adjust the RIPv2 summary address commands on R1 and R2 from 255.255.255.248 to 255.255.255.240.

When the RIPv2 summary address is adjusted by one bit from 255.255.255.248 to 255.255.255.240, it will match prefixes in quantities and multiples of 16. This will easily include the 192.168.1.25/32 prefix that is currently not being summarized.

Take a look at the correct interface configuration commands:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface Ethernet0/2
R1(config-if)#ip summary-address rip 192.168.1.16 255.255.255.240
R1(config-if)#no  ip summary-address rip 192.168.1.16 255.255.255.248
R1(config-if)#end
R1#

R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int e0/2
R2(config-if)#ip summary-address rip 192.168.1.16 255.255.255.240
R2(config-if)#no  ip summary-address rip 192.168.1.16 255.255.255.248
R2(config-if)#end
R2#
```

Clear routing table on R3 and R4. Now, only the 192.168.1.16/28 summary appears in the routing tables of routers R3 and R4:

```
R3#sh ip route rip | begin Gateway
```



```
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 8 subnets, 3 masks
R        10.10.0.1/32 [120/1] via 10.10.10.1, 00:00:03, Ethernet0/2
R        10.10.0.2/32 [120/1] via 10.10.10.2, 00:00:03, Ethernet0/2
      192.168.1.0/24 is variably subnetted, 4 subnets, 3 masks
R        192.168.1.16/28 [120/4] via 10.10.10.2, 00:00:03, Ethernet0/2
                         [120/4] via 10.10.10.1, 00:00:03, Ethernet0/2
R3#

R4#sh ip route rip | begin Gateway
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks
R        10.10.0.1/32 [120/1] via 10.10.10.1, 00:00:00, Ethernet0/2
R        10.10.0.2/32 [120/1] via 10.10.10.2, 00:00:14, Ethernet0/2
R        10.10.0.3/32 [120/1] via 10.10.10.3, 00:00:21, Ethernet0/2
R        10.10.2.0/31 [120/1] via 10.10.10.3, 00:00:21, Ethernet0/2
      192.168.1.0/24 is variably subnetted, 5 subnets, 3 masks
R        192.168.1.16/28 [120/4] via 10.10.10.2, 00:00:14, Ethernet0/2
                         [120/4] via 10.10.10.1, 00:00:00, Ethernet0/2
R4#
```

5. BGP Troubleshooting Section

5.1. Symptom: The BGP aggregate is advertising its longer matches.

Analysis and testing:

When you examine the BGP aggregate address generated by router R5, you can see that the longer matches are not being suppressed. As the supplied BGP diagram shows, the immediate BGP neighbors of R5 are R1 and R2. Therefore, begin by examining the BGP tables of R1 and R2:

```
R1#show ip bgp
BGP table version is 695, local router ID is 10.10.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i172.16.0.0/22    192.168.1.6     4294967295    100      0 6 i
*>i                 192.168.1.6     4294967295    100      0 6 i
* i172.16.4.0/24    192.168.1.19             0    100      0 5 i
*>                  192.168.1.17             0             0 5 i
* i172.16.4.0/22    192.168.1.19             0    100      0 5 i
*>                  192.168.1.17             0             0 5 i
* i172.16.5.0/24    192.168.1.19             0    100      0 5 i
*>                  192.168.1.17             0             0 5 i
* i172.16.6.0/24    192.168.1.19             0    100      0 5 i
*>                  192.168.1.17             0             0 5 i
* i172.16.7.0/24    192.168.1.19             0    100      0 5 i
*>                  192.168.1.17             0             0 5 i


R2#sh ip bgp
BGP table version is 45, local router ID is 10.10.0.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
```

```
* i172.16.0.0/22    192.168.1.6     4294967295    100      0 6 i
*>i                 192.168.1.6     4294967295    100      0 6 i
* i172.16.4.0/24    192.168.1.17             0    100      0 5 i
*>                  192.168.1.19             0             0 5 i
* i172.16.4.0/22    192.168.1.17             0    100      0 5 i
*>                  192.168.1.19             0             0 5 i
* i172.16.5.0/24    192.168.1.17             0    100      0 5 i
*>                  192.168.1.19             0             0 5 i
* i172.16.6.0/24    192.168.1.17             0    100      0 5 i
*>                  192.168.1.19             0             0 5 i
* i172.16.7.0/24    192.168.1.17             0    100      0 5 i
*>                  192.168.1.19             0             0 5 i
R2#
```

Likely cause: R5 is not configured with the** summary-only **option within its BGP aggregate command.

Check the configuration of the aggregate statement on R5:

```
R5#sh run | section bgp
router bgp 5
 no synchronization
 bgp log-neighbor-changes
 network 172.16.4.0 mask 255.255.255.0
 network 172.16.5.0 mask 255.255.255.0
 network 172.16.6.0 mask 255.255.255.0
 network 172.16.7.0 mask 255.255.255.0
 aggregate-address 172.16.4.0 255.255.252.0
 neighbor 192.168.1.16 remote-as 1
 neighbor 192.168.1.18 remote-as 1
 maximum-paths ibgp 2
 no auto-summary
```

As you can see, there is no **summary-only** option configured with the BGP aggregate statement.

Resolution: Add the** summary-only **option to the R5 BGP aggregate statement.

When the **summary-only** option is added to the BGP aggregate statement on router R5, R5 will stop advertising the longer matching routes:

```
R5#conf t

Enter configuration commands, one per line.  End with CNTL/Z.

R5(config)#router bgp 5
R5(config-router)#aggregate-address 172.16.4.0 255.255.252.0 summary-only
```

With this change made, re-examine the BGP tables of R1 and R2:

```
R1#sh ip bgp
BGP table version is 131, local router ID is 10.10.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 172.16.0.0/22    192.168.1.6     4294967295    100      0 6 i
 * i                  192.168.1.6     4294967295    100      0 6 i
 * i 172.16.4.0/22    192.168.1.19             0    100      0 5 i
 *>                   192.168.1.17             0             0 5 i
R1#
```



```
R2#show ip bgp
BGP table version is 137, local router ID is 10.10.0.2
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 172.16.0.0/22    192.168.1.6     4294967295    100      0 6 i
 * i                  192.168.1.6     4294967295    100      0 6 i
 *>  172.16.4.0/22    192.168.1.19             0             0 5 i
 * i                  192.168.1.17             0    100      0 5 i
R2#
```

As you can see, the longer matching BGP routes have been removed.

5.2. Symptom: R5 is not selecting the BGP maximum paths for a specific route.

Analysis and testing:

R5 only has one next hop for the 172.16.0.0/22 prefix.

```
R5#sh ip ro bgp
     172.16.0.0/16 is variably subnetted, 6 subnets, 2 masks
B       172.16.4.0/22 [200/0] via 0.0.0.0, 2d01h, Null0
B       172.16.0.0/22 [20/0] via 192.168.1.18, 1d00h
```

When you examine the BGP table and the topology, you see that R5 should have two next hops for the 172.16.0.0/22 prefix:

```
R5#show ip bgp
BGP table version is 150, local router ID is 192.168.1.25
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *   172.16.0.0/22    192.168.1.16                           0 1 6 i
 *>                   192.168.1.18                           0 1 6 i
 s>  172.16.4.0/24    0.0.0.0                  0         32768 i
 *>  172.16.4.0/22    0.0.0.0                            32768 i
 s>  172.16.5.0/24    0.0.0.0                  0         32768 i
 s>  172.16.6.0/24    0.0.0.0                  0         32768 i
 s>  172.16.7.0/24    0.0.0.0                  0         32768 i
R5#
```

Look at the specific details of this prefix:

```
R5#sh ip bgp 172.16.0.0 255.255.252.0
BGP routing table entry for 172.16.0.0/22, version 146
Paths: (2 available, best #2, table default)
Multipath: iBGP
  Advertised to update-groups:
     1
  Refresh Epoch 1
  1 6, (aggregated by 6 192.168.1.8)
    192.168.1.16 from 192.168.1.16 (10.10.0.1)
```

```
      Origin IGP, localpref 100, valid, external, atomic-aggregate
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  1 6, (aggregated by 6 192.168.1.8)
    192.168.1.18 from 192.168.1.18 (10.10.0.2)
      Origin IGP, localpref 100, valid, external, atomic-aggregate, best
      rx pathid: 0, tx pathid: 0x0
R5#
```

This output shows that the multipath option is enabled, but it is enabled for Internal BGP (IBGP) updates. These two BGP updates are External BGP (EBGP) updates.

Likely cause: A BGP multipath is misconfigured on R5.

Next, review the running configuration of the BGP configuration on R5:

```
R5#sh run | section bgp
router bgp 5
 no synchronization
 bgp log-neighbor-changes
 network 172.16.4.0 mask 255.255.255.0
 network 172.16.5.0 mask 255.255.255.0
 network 172.16.6.0 mask 255.255.255.0
 network 172.16.7.0 mask 255.255.255.0
 aggregate-address 172.16.4.0 255.255.252.0 summary-only
 neighbor 192.168.1.16 remote-as 1
 neighbor 192.168.1.18 remote-as 1
 maximum-paths ibgp 2
```

Resolution: Change the** maximum-paths **command on R5.

Change the current **maximum-paths ibgp 2** command to the **maximum-paths 2** command. This change is reflected in the following BGP running configuration section:

```
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#router bgp 5
R5(config-router)#maximum-paths 2
R5(config-router)#no  maximum-paths ibgp 2
R5(config-router)#end
R5#
```

After you make this change, verify the routing table on R5:

```
R5#show ip route bgp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 10 subnets, 3 masks
B        172.16.0.0/22 [20/0] via 192.168.1.18, 00:00:55
                       [20/0] via 192.168.1.16, 00:00:55
B        172.16.4.0/22 [200/0] via 0.0.0.0, 03:17:50, Null0
```



```
R5#
```

For more information on the BGP **maximum-paths** command, go to http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094431.shtml#bgpmpath

5.3. Symptom: R6 is selecting the incorrect BGP next-hop address.

R6 prefers R4 as its next hop, instead of R3 as required:

```
R6#show ip bgp
BGP table version is 2, local router ID is 192.168.1.8
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
 *   172.16.4.0/22    192.168.1.3     4294967295             0 1 5 i
 *>                   192.168.1.4            100             0 1 5 i
R6#
```

Analysis and testing:

R6 has two possible paths for any BGP-learned routes that it receives—through R3 and through R4. The requirements of the lab state that "R6 should prefer the path through R3 to get to the prefix advertised by R5."

This specific issue involves applying the BGP path-selection rules. As a review, here is a general list of the BGP path-selection process:

- Prerequisite: Good next hop, synchronized if necessary
- Highest weight
- Highest local preference
- Locally originated
- Shortest AS path length
- Origin code (i > ?)
- Lowest multi-exit discriminator (MED)
- EBGP over IBGP
- If internal, path with lowest IGP metric to next hop preferred
- If external, multipath considered
- If external, older path preferred
- Lowest router ID or originator ID
- Minimum cluster list length
- Lowest neighbor address

Here are two good links for a more detailed description of the BGP path-selection process and the use of the MED attribute in particular:

- Best path: http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094431.shtml

- MED discussion: http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094934.shtml

Given these rules and applying these rules to the specific topology of this scenario, R6 will use the next hop for the oldest route. Selecting a path based on the age of a candidate route is one of the tie-breaking conditions provided at the end of the BGP selection process.

To get R6 to prefer R3, the initial configuration advertises the route from R4 with a higher MED: 100 from R4 versus default 0 from R3. Note that a lower MED is preferred.

First, review the BGP configuration of R3:

```
R3#sh run | section bgp
 redistribute bgp 1 subnets
router bgp 1
 no synchronization
 bgp log-neighbor-changes
 neighbor 10.10.0.1 remote-as 1
 neighbor 10.10.0.1 update-source Loopback103
 neighbor 10.10.0.2 remote-as 1
 neighbor 10.10.0.2 update-source Loopback103
 neighbor 10.10.0.4 remote-as 1
 neighbor 10.10.0.4 update-source Loopback103
 neighbor 192.168.1.6 remote-as 6
 no auto-summary
 !
```

Notice that the BGP configuration of R3 has absolutely no reference to any MED configuration.

Now, check R4:

```
R4#sh run | begin bgp
router bgp 1
 no synchronization
 bgp log-neighbor-changes
 neighbor 10.10.0.1 remote-as 1
 neighbor 10.10.0.1 update-source Loopback104
 neighbor 10.10.0.2 remote-as 1
 neighbor 10.10.0.2 update-source Loopback104
 neighbor 10.10.0.3 remote-as 1
 neighbor 10.10.0.3 update-source Loopback104
 neighbor 192.168.1.6 remote-as 6
 neighbor 192.168.1.6 route-map SETMED out
 no auto-summary
!
ip prefix-list 172 seq 5 permit 172.16.4.0/22
!
route-map SETMED permit 10
 match ip address prefix-list 172
 set metric 100
```

Notice that router R4 is setting the MED to a value of 100 for all 172.16.4.0/22 prefixes that it advertises to R6.

A problem in this scenario is that R6 is configured with the **bgp bestpath med missing-as-worst** BGP command:

```
R6#sh run | s bgp
router bgp 6
 no synchronization
 bgp log-neighbor-changes
 bgp bestpath med missing-as-worst
 network 172.16.0.0 mask 255.255.255.0
 network 172.16.1.0 mask 255.255.255.0
 network 172.16.2.0 mask 255.255.255.0
```



```
 network 172.16.3.0 mask 255.255.255.0
 aggregate-address 172.16.0.0 255.255.252.0 summary-only
 neighbor 192.168.1.3 remote-as 1
 neighbor 192.168.1.4 remote-as 1
 no auto-summary
```

If you have enabled **bgp bestpath med missing-as-worst**, the paths with the MED of 0 are assigned a MED of 4,294,967,295, as shown in the Cisco IOS **show** output here:

```
R6#show ip bgp
BGP table version is 2, local router ID is 192.168.1.8
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
 *   172.16.4.0/22    192.168.1.3     4294967295             0 1 5 i
 *>                   192.168.1.4            100             0 1 5 i
R6#
```

Note that the MED of 0 is replaced with an MED of 4,294,967,295 on the prefix that is advertised from R3.

Fix this problem by removing the **bgp bestpath med missing-as-worst** BGP configuration command on R6 or setting the MED lower than 100 on R3.

Likely cause: The MED values assigned to routes that are received by R6 are not selecting the desired path.

As you can see, with MED values not set on R3 when they are set on R4, and with R6 configured with the **bgp bestpath med missing-as-worst** command, R6 is not selecting R3 as its best path.

To resolve this issue, two options exist:

- Remove the **bgp bestpath med missing-as-worst** command from R6.
- Manually set the MED on R3 for the 172.16.4.0/22 prefix to a value lower than 100 so that R3 will be selected as a preferred path over R4. Remember that R4 is currently setting a MED of 100 for the 172.16.4.0/22 prefix when it advertises this route to R6.

Of these two options, the first one will be selected for this scenario.

Resolution: Remove the** bgp bestpath med missing-as-worst **command from R6 .

When the **bgp bestpath med missing-as-worst** command is removed from R6, the BGP updates that are received by R6 from R3 are assigned a MED value of 0, which is lower than the MED value of 100 assigned by R4. Therefore, R3 becomes the preferred path for the 172.16.4.0/22 prefix, as specified in the scenario requirements. Now, remove the **bgp bestpath med missing-as-worst** command, and verify the results.

```
R6#conf t
R6(config)#router bgp 6
R6(config-router)#no bgp bestpath med missing-as-worst


R6#show ip bgp
BGP table version is 16, local router ID is 192.168.1.8
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
 s>  172.16.0.0/24    0.0.0.0                  0         32768 i
 *>  172.16.0.0/22    0.0.0.0                            32768 i
 s>  172.16.1.0/24    0.0.0.0                  0         32768 i
 s>  172.16.2.0/24    0.0.0.0                  0         32768 i
 s>  172.16.3.0/24    0.0.0.0                  0         32768 i
 *>  172.16.4.0/22    192.168.1.3                            0 1 5 i
 *                    192.168.1.4            100             0 1 5 i
R6#
```

As you can see, the large MED value of 4,294,967,295 has been removed from all BGP updates learned from R3, and R3 is now the preferred path for the 172.16.4.0/22 prefix as specified by this scenario.

6. IPv4 Redistribution Troubleshooting Section

6.1. Symptom: R3 prefers OSPF routes for the 10.10.0.1/32, 10.10.0.2/32, and 10.10.0.4/32 subnets.

Analysis and testing:

When you review the IP routing table of R3, you can see that R3 prefers OSPF routes for the following three unconnected prefixes:

```
R3#show ip route | inc 10\.10\.0\.[124]
O E2     10.10.0.1/32 [110/20] via 192.168.1.4, 00:02:26, Ethernet0/0
O E2     10.10.0.2/32 [110/20] via 192.168.1.4, 00:02:26, Ethernet0/0
O E2     10.10.0.4/32 [110/20] via 192.168.1.4, 00:02:26, Ethernet0/0
R3#
```

This is the case even though all three of these routes are native RIP routes and R3 maintains a direct connection to RIP:

```
R3#show ip protocols summary
Index Process Name
0     connected
1     static
2     application
3     rip
4     ospf 1
5     bgp 1
*** IP Routing is NSF aware ***

R3#
```

Therefore, R3 should prefer RIP, the native route source, for these routes. Because this topology contains a loop, R3 has multiple sources to learn these routes from.

If R3 is having this problem, you should check to see whether R4 is having the same problem:

```
R4#show ip route | inc 10\.10\.0\.[123]
R        10.10.0.1/32 [120/1] via 10.10.10.1, 00:00:28, Ethernet0/2
R        10.10.0.2/32 [120/1] via 10.10.10.2, 00:00:14, Ethernet0/2
R        10.10.0.3/32 [120/1] via 10.10.10.3, 00:00:24, Ethernet0/2
R4#
```


R4 is not having the same problem. Next, check to make sure that R4 is learning these routes via OSPF:

```
R4#show ip ospf database | b Ex
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.10.0.1       10.10.0.4       446         0x8000000D 0x00085F 0
10.10.0.2       10.10.0.4       446         0x8000000D 0x00FD68 0
10.10.0.3       10.10.0.3       530         0x8000000D 0x00F96C 0
10.10.0.3       10.10.0.4       446         0x8000000D 0x00F371 0
10.10.0.4       10.10.0.4       446         0x8000000D 0x00E97A 0
10.10.2.0       10.10.0.3       530         0x8000000D 0x00FB6C 0
10.10.2.0       10.10.0.4       446         0x8000000D 0x00F571 0
10.10.10.0      10.10.0.3       284         0x8000000D 0x007FE6 0
10.10.10.0      10.10.0.4       446         0x8000000D 0x0079EB 0
192.168.1.0     10.10.0.4       3600        0x80000002 0x0043FC 0
192.168.1.16    10.10.0.4       958         0x8000000D 0x0055BB 0
R4#
```

R4 is learning the routes via OSPF. As you can see, these RIP routes are redistributed into OSPF on R4.

Likely cause: An administrative distance conflict exists in this configuration.

There is a common problem that occurs when a loop exists in a topology and the loop is spread over both OSPF and RIP routing domains. For routers that maintain a connection to both OSPF and RIP, the RIP native routes will be preferred as OSPF external routes because OSPF has a lower administrative distance than all RIP routes.

In this characteristic, OSPF and RIP are unlike EIGRP, which has a low administrative distance of 90 for all its internal networks and a high administrative distance of 170 for all its external networks.

Resolution: Assign a higher administrative distance to all externally learned OSPF routes on R3.

Although this problem is being resolved with the administrative distance command, a solution could also involve filtering the specified prefixes.

In this scenario, the solution involving the application of the administrative distance command is supplied on router R4:

```
R4#sh run | section ^router ospf
router ospf 1
 log-adjacency-changes
 redistribute rip subnets
 network 192.168.1.4 0.0.0.0 area 0
 distance ospf external 121
```

This configuration is why R3 has this poor path selection issue and R4 does not. Therefore, you should apply the same solution to R3. When you do, enable **debug ip routing** and **debug ip ospf lsa-generation** to see the effects of entering this command:

```
R3#debug ip routing
IP routing debugging is on
R3#debug ip ospf lsa-generation
OSPF summary lsa generation debugging is on

R3#conf t
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router ospf 1
R3(config-router)# distance ospf external 121
R3(config-router)#end
R3#
```

We see that Cisco IOS Software has detected a "closer administrative distance," and it begins to flush routes:

```
May 23 00:53:39.924: RT: updating rip 10.10.0.2/32 (0x0):
    via 10.10.10.2 Et0/2

May 23 00:53:39.924: RT: closer admin distance for 10.10.0.2, flushing 1 routes
May 23 00:53:39.924: RT(multicast): delete subnet route to 10.10.0.2/32
```

Now, it adds a new RIP route:

```
Aug 18 18:21:44.371: RT: add 10.10.0.2/32 via 10.10.10.2, rip metric [120/1]
Aug 18 18:21:44.371: RT: NET-RED 10.10.0.2/32
```

Simultaneously, OSPF generates a poisoned external link state advertisement (LSA) for the 10.10.0.2/32 prefix:

```
Aug 18 18:21:46.111: OSPF: Generate external LSA 10.10.0.2, mask 255.255.255.255,
type 5, age 3600, metric 16777215, tag 0, metric-type 2, seq 0x80000002
```

The result of all this is the desired routing table entries:

```
R3#show ip route | inc 10\.10\.0\.[124]
R        10.10.0.1/32 [120/1] via 10.10.10.1, 00:00:17, Ethernet0/2
R        10.10.0.2/32 [120/1] via 10.10.10.2, 00:00:02, Ethernet0/2
R        10.10.0.4/32 [120/1] via 10.10.10.4, 00:00:21, Ethernet0/2
R3#
```

Now that you seem to have addressed all of the IPv4 unicast issues, you will test reachability using this simple Tool Command Language (Tcl) script: Enter the command **tclsh** and paste in this script. When it is complete, you will have a record of successful and unsuccessful pings. Enter the command **tclquit** to exit the command interpreter. Notice that this list excludes the IP addresses in VPN1.

```
tclsh
foreach address {
10.10.0.1
10.10.10.1
192.168.1.16
10.10.0.2
10.10.10.2
192.168.1.18
10.10.0.3
10.10.10.3
192.168.1.3
10.10.0.4
10.10.10.4
192.168.1.4
192.168.1.25
192.168.1.17
192.168.1.19
172.16.5.1
172.16.4.1
172.16.7.1
172.16.6.1
```



```
192.168.1.8
192.168.1.6
172.16.1.1
172.16.0.1
172.16.3.1
172.16.2.1
192.168.1.1
} {ping $address}
```

7. MPLS Troubleshooting Section

7.1. Symptom: R7 has not learned any VPN routes from the remote site.

Analysis and testing:

R7 has formed an OSPF neighbor relationship with R1, but it has not learned any VPN routes from the remote site.

```
R7#sh ip ospf neighbor
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.10.3.0         1   FULL/BDR        00:00:33    10.10.3.0       Ethernet0/1
R7#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

R7#
```

This command displays absolutely nothing.

Based on this information, it will be useful to check whether OSPF is configured for the appropriate VRF instance on R3. Review the R3 OSPF configuration:

```
R3#show running-config | section router ospf
router ospf 20 vrf VPN1
 router-id 10.10.2.0
 domain-id 1.1.1.1
 redistribute bgp 1 subnets
 network 10.10.2.0 0.0.0.0 area 20
router ospf 1
 router-id 10.10.0.3
 redistribute rip subnets
 network 192.168.1.3 0.0.0.0 area 0
 distance ospf external 121
R3#
```

You see that R3 has the OSPF process 20 configured for VPN1 and also possesses a network statement. But the following **show ip ospf interface brief** output does not display any interfaces that are associated with process ID, or PID, 20:

```
R3#show ip ospf interface brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
E0/0         1     0               192.168.1.3/29     1     BDR   2/2
```

Likely cause: Interface E0/1 on R3 is not assigned to VPN1.

This likely cause can be verified with the following **show** command:

```
R3#show ip vrf
  Name                             Default RD          Interfaces
  VPN1                             1:100
```

As you can see, no interfaces are listed under the interfaces column, which clearly indicates a configuration problem.

Resolution: Configure** ip vrf forwarding VPN1 **on R3.

When you configure **ip vrf forwarding VPN1** on the E0/1 interface of R3, R7 should receive OSPF routes.

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#int e0/1
R3(config-if)#ip vrf forwarding VPN1
% Interface Ethernet0/1 IPv4 disabled and address(es) removed due to enabling VRF
VPN1
R3(config-if)# ip address 10.10.2.0 255.255.255.254
% Warning: use /31 mask on non point-to-point interface cautiously
R3(config-if)#end
R3#
```

Now, check the VRF table on R3:

```
R3#sh ip vrf
  Name                             Default RD          Interfaces
  VPN1                             1:100               Et0/1
```

Before entering the **ip vrf forwarding VPN1** command, no interfaces were referenced. Now the E0/1 interface is listed.

Now, review the routing table of R7:

```
R7#sh ip route ospf | begin Gate
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O E2     10.10.2.0/31 [110/1] via 10.10.3.0, 00:01:30, Ethernet0/0
O E2     10.10.2.10/32 [110/11] via 10.10.3.0, 00:01:24, Ethernet0/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O E2     172.16.2.0/24 [110/20] via 10.10.3.0, 00:01:24, Ethernet0/0
R7#
```

R7 is now learning routes via R3.

7.2. Symptom: All OSPF routes on R7 and R8 are external.

Analysis and testing:

The scenario requires that subnets in OSPF Area 20 be seen as internal routes on R7 and that subnets in OSPF Area 10 be seen as internal routes on R8. Here are the current routing tables on these switches:

```
R7#sh ip route ospf | begin Gate
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O E2     10.10.2.0/31 [110/1] via 10.10.3.0, 00:01:30, Ethernet0/0
O E2     10.10.2.10/32 [110/11] via 10.10.3.0, 00:01:24, Ethernet0/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O E2     172.16.2.0/24 [110/20] via 10.10.3.0, 00:01:24, Ethernet0/0
R7#
R8#sh ip route ospf | begin Gate
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O E2     10.10.3.0/31 [110/1] via 10.10.2.0, 00:07:24, Ethernet0/0
O E2     10.10.3.9/32 [110/11] via 10.10.2.0, 00:07:24, Ethernet0/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O E2     172.16.1.0/24 [110/20] via 10.10.2.0, 00:07:24, Ethernet0/0
R8#
```

Likely cause: Each site in VPN1 uses a different OSPF domain ID.

BGP associates an OSPF domain ID with each OSPF route that is learned from a distant site. Here you see the domain ID for prefix 10.10.2.10/32 in the BGP table of R1, for example:

```
R1#show bgp vpnv4 unicast vrf VPN1 10.10.2.10
BGP routing table entry for 1:100:10.10.2.10/32, version 9
Paths: (1 available, best #1, table VPN1)
  Not advertised to any peer
  Refresh Epoch 1
  Local
    10.10.0.3 (metric 1) from 10.10.0.3 (10.10.0.3)
      Origin incomplete, metric 11, localpref 100, valid, internal, best
      Extended Community: RT:1:20 OSPF DOMAIN ID:0x0005:0x010101010200
        OSPF RT:0.0.0.20:2:0 OSPF ROUTER ID:10.10.2.0:0
      mpls labels in/out nolabel/24
      rx pathid: 0, tx pathid: 0x0
R1#
```

The domain ID is represented in hexadecimal; by default it is equal to the OSPF process ID in the home site, 20 in this case. This value is compared to the local OSPF domain ID, which can be seen here for R1:

```
R1# show ip ospf 10 | inc Domain
 Routing Process "ospf 10" with ID 10.10.3.0
   Domain ID type 0x0005, value 0.0.0.10
```

Here the value is expressed in dotted decimal, and equals the OSPF process ID. When the domain IDs at two sites differ, the routes are treated as external. When the domain IDs are equal, OSPF routes learned from the far site are treated as interarea routes.

Resolution: Configure matching domain IDs for the VPN OSPF processes on R1 and R3.

The scenario requires different OSPF process IDs in the two sites. You can override the default behavior and statically configure the domain IDs at each site. We chose the value 1.1.1.1, but any value will work as long as they are the same at each site. After this change you will see the following routing information on R7 and R8:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router ospf 10 vrf VPN1
R1(config-router)# domain-id 1.1.1.1
R1(config-router)#end
R1#

R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router ospf 20 vrf VPN1
R3(config-router)# domain-id 1.1.1.1
R3(config-router)#end
R3#

R7#sh ip route ospf | begin Gate
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O IA     10.10.2.0/31 [110/11] via 10.10.3.0, 00:02:10, Ethernet0/0
O IA     10.10.2.10/32 [110/21] via 10.10.3.0, 00:02:10, Ethernet0/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O E2     172.16.2.0/24 [110/20] via 10.10.3.0, 00:14:45, Ethernet0/0
R7#
R8#sh ip route ospf | begin Gate
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O IA     10.10.3.0/31 [110/11] via 10.10.2.0, 00:01:18, Ethernet0/0
O IA     10.10.3.9/32 [110/21] via 10.10.2.0, 00:01:18, Ethernet0/0
      172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
O E2     172.16.1.0/24 [110/20] via 10.10.2.0, 00:14:43, Ethernet0/0
R8#
```

7.3. Symptom: There is no reachability between sites.

Analysis and testing:

Though each of the switches in VPN1 has full routes to IP addresses in the far site, you cannot ping between them.

You know that VPN traffic between the sites must be switched by MPLS across the link between routers R1 and R3, so you verify the operation of the Label Distribution Protocol (LDP), between these routers:

```
R1#show mpls ldp neighbor
R1#show mpls interfaces
Interface              IP            Tunnel   BGP Static Operational
R1#
```

You find that R1 does not see R3 as an LDP neighbor, and has no interfaces associated with MPLS. Checking the configuration of R1 interface E0/2, you find that the command **mpls ip** has not been configured.

Resolution: Configure** mpls ip **on R1.

When you configure **mpls ip** on the R1 E0/2 you see a neighbor relationship immediately form:

```
R1(config)#int E0/2
R1(config-if)#mpls ip
R1(config-if)#end
R1#
00:05:59.155: %LDP-5-NBRCHG: LDP Neighbor 10.10.0.3:0 (1) is UP
R1#show mpls ldp neighbor
    Peer LDP Ident: 10.10.0.3:0; Local LDP Ident 10.10.0.1:0
        TCP connection: 10.10.0.3.11417 - 10.10.0.1.646
        State: Oper; Msgs sent/rcvd: 14/14; Downstream
        Up time: 00:00:10
        LDP discovery sources:
          Ethernet0/2, Src IP addr: 10.10.10.3
        Addresses bound to peer LDP Ident:
          192.168.1.3     10.10.10.3      10.10.0.3
R1#
```

Now VPN traffic can be encapsulated with a VPN header as it is sent into the core by R1 and R3, and the sites have full reachability.

```
R7#ping 10.10.2.10 source 10.10.3.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.2.10, timeout is 2 seconds:
Packet sent with a source address of 10.10.3.9
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R7#
```

8. Multicast Troubleshooting Section

8.1. Symptom: Traffic from the multicast source R6 cannot be registered.

Analysis and testing:

First, examine the mroute table on R4:

```
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,
       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
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.255.1.1), 00:01:06/00:01:52, RP 0.0.0.0, flags: SP
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null

(*, 239.255.1.7), 00:00:31/00:02:28, RP 0.0.0.0, flags: SP
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null

(*, 224.0.1.40), 00:12:33/00:02:10, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
```

```
    Ethernet0/0, Forward/Sparse, 00:12:34/00:02:09

R4#sh ip pim int

Address          Interface                Ver/   Nbr    Query  DR     DR
                                          Mode   Count  Intvl  Prior
192.168.1.4      Ethernet0/0          v2/S   1      30     1      192.168.1.4
```

R4 is the designated router (DR) on the 192.168.1.0/29 subnet because it has the highest IP address on the link. As the DR, R4 must register the traffic from R6 with the rendezvous points (RPs). But R4 E0/2 is not configured with Protocol Independent Multicast (PIM), so R4 cannot register the multicast traffic from R6. Issuing the **debug ip pim bsr** command on R4 shows that R4 is rejecting bootstrap router (BSR) messages that are received on E0/0, because E0/0 is not the Reverse Path Forwarding (RPF) interface to 10.10.0.1 or 10.10.0.2.

```
R4#
May 23 01:32:50.994: PIM-BSR(0): bootstrap (10.10.0.2) on non-RPF path Ethernet0/0
or from non-RPF neighbor 0.0.0.0 discarded
R4#
```

Likely cause: PIM is not selecting the correct router as the DR.

This problem can be fixed by putting PIM on R4 E0/2. The problem could also be fixed by changing the routing table or by adding a static mroute so that R4 E0/0 becomes the RPF interface to the RP addresses. All these solutions are ruled out.

Resolution: Configure R3 as the DR for the VLAN 346 link.

The preferred solution is to make R3 the DR for the link by raising the DR priority on R3 E0/0. As the DR, R3 can then register the traffic with the RPs:

```
R3(config)#int e0/0
R3(config-if)#ip pim dr-priority 2
R3(config-if)#end

R3#debug ip pim

17:08:40.183: %PIM-5-DRCHG: DR change from neighbor 192.168.1.4 to 192.168.1.3 on
interface Ethernet0/0
```

R3 is now the PIM DR.

8.2. Symptom: The BSR hash calculation is malfunctioning.

Analysis and testing:

The BSR with the larger priority value is preferred. If the priority values are the same, the router with the larger IP address is the BSR. The default value is 0.

When you set the *priority* argument, the BSR with the larger priority value is preferred. If the priority values are the same, the router with the larger IP address is the BSR.

The RP candidate with the highest priority is preferred. When RP candidate priorities are equal, the RP with the highest hash value is chosen for each group, allowing the RPs to load-share the RP burden.

Each router calculates a hash value from the group address, each RP address, and the hash mask length that is advertised by the BSRs. This process allows RP load sharing. With hash mask

length 0, all groups will have the same hash value, and all will map to the same RP. The number of 0 bits in the hash mask, to a power of two, indicates the number of consecutive group addresses that will hash to the same hash value. For each group, the RP with the highest hash value for that group will be chosen.

The BSR values for this lab are configured with a hash mask length of 29, so eight group addresses in a row will map to the same RP. Here are the hash values for the eight groups 239.255.1.0 through 239.255.1.7:

```
R5#sh ip pim rp-hash 239.255.1.0
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:24:47, expires: 00:01:38
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
R5#sh ip pim rp-hash 239.255.1.1
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:24:51, expires: 00:01:34
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
R5#sh ip pim rp-hash 239.255.1.2
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:24:53, expires: 00:01:32
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
R5#sh ip pim rp-hash 239.255.1.3
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:24:55, expires: 00:01:31
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
R5#sh ip pim rp-hash 239.255.1.4
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:24:56, expires: 00:01:29
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
R5#sh ip pim rp-hash 239.255.1.5
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:25:00, expires: 00:02:26
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
R5#sh ip pim rp-hash 239.255.1.6
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:25:03, expires: 00:02:24
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
R5#sh ip pim rp-hash 239.255.1.7
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:25:06, expires: 00:02:21
  PIMv2 Hash Value (mask 255.255.255.248)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720
```

Note that all eight groups have identical hash values from each RP candidate. RP 10.10.0.2 happens to have the highest hash value, so all eight groups will use RP 10.10.0.2, not RP 10.10.0.1:

```
R5#show ip mroute | inc 10.10.0.1
R5#show ip mroute | inc 10.10.0.2
(*, 239.255.1.0), 00:48:22/00:02:34, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.1), 00:51:17/00:02:33, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.2), 00:48:19/00:02:34, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.3), 00:48:17/00:02:30, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.4), 00:48:16/00:02:27, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.5), 00:48:14/00:02:35, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.6), 00:49:15/00:02:31, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.7), 00:46:53/00:02:32, RP 10.10.0.2, flags: SJCL
```

Likely cause: PIM hash length values are not correctly configured.

As mentioned earlier, each router calculates a hash value from the group address, each RP address, and the hash mask length that is advertised by the BSRs. This process allows RP load-sharing. With hash mask length 0, all groups will have the same hash value, and all will map to the same RP. The number of 0 bits in the hash mask, to a power of two, indicates the number of consecutive group addresses that will hash to the same hash value. For each group, the RP with the highest hash value for that group will be chosen.

Resolution: The BSR hash length is configured as 29 but should be 30 on R1 and R2.

The lab requires that four groups use RP 10.10.0.1 and four use RP 10.10.0.2. To do this, you can change the hash mask length to 30 on each BSR.

```
R1#show run | inc bsr
ip pim bsr-candidate Loopback101 29
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip pim bsr-candidate Loopback101 30
R1(config)#end
R1#

R2#show run | inc bsr
ip pim bsr-candidate Loopback102 29
R2#
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip pim bsr-candidate Loopback102 30
R2(config)#end
```

After issuing the **clear ip pim rp-mapping** command on R5 and waiting about a minute, you see the desired result:

```
R5#clear ip pim rp-mapping

R5#show ip mroute | inc 10.10.0.1
(*, 239.255.1.4), 00:56:52/00:02:52, RP 10.10.0.1, flags: SJCL
(*, 239.255.1.5), 00:56:50/00:02:59, RP 10.10.0.1, flags: SJCL
(*, 239.255.1.6), 00:57:51/00:02:52, RP 10.10.0.1, flags: SJCL
(*, 239.255.1.7), 00:55:29/00:02:51, RP 10.10.0.1, flags: SJCL

R5#show ip mroute | inc 10.10.0.2
(*, 239.255.1.0), 00:57:02/00:02:51, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.1), 00:59:57/00:02:50, RP 10.10.0.2, flags: SJCL
```



```
(*, 239.255.1.2), 00:56:59/00:02:50, RP 10.10.0.2, flags: SJCL
(*, 239.255.1.3), 00:56:57/00:02:51, RP 10.10.0.2, flags: SJCL
```

Here are the resulting hash values:

```
R5#sh ip pim rp-hash 239.255.1.0
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:04:36, expires: 00:01:51
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720

R5#sh ip pim rp-hash 239.255.1.1
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:04:50, expires: 00:01:37
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720

R5#sh ip pim rp-hash 239.255.1.2
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:04:53, expires: 00:01:34
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720

R5#sh ip pim rp-hash 239.255.1.3
  RP 10.10.0.2 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:04:56, expires: 00:01:31
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 680079633
    RP 10.10.0.2, via bootstrap, priority 0, hash value 1843141720

R5#sh ip pim rp-hash 239.255.1.4
  RP 10.10.0.1 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:04:58, expires: 00:01:30
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 1116707509
    RP 10.10.0.2, via bootstrap, priority 0, hash value 132285948

R5#sh ip pim rp-hash 239.255.1.5
  RP 10.10.0.1 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:04:59, expires: 00:01:29
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 1116707509
    RP 10.10.0.2, via bootstrap, priority 0, hash value 132285948

R5#sh ip pim rp-hash 239.255.1.6
  RP 10.10.0.1 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:05:01, expires: 00:02:25
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 1116707509
    RP 10.10.0.2, via bootstrap, priority 0, hash value 132285948

R5#sh ip pim rp-hash 239.255.1.7
  RP 10.10.0.1 (?), v2
    Info source: 10.10.0.2 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:05:03, expires: 00:02:23
  PIMv2 Hash Value (mask 255.255.255.252)
    RP 10.10.0.1, via bootstrap, priority 0, hash value 1116707509
```

```
    RP 10.10.0.2, via bootstrap, priority 0, hash value 132285948
```

Note The Mentor Guide engine in the web portal can help you use Cisco IOS Software commands to see a comprehensive view of the configuration for a specific section. With the Mentor Guide engine, you can enter more than 1000 Cisco IOS Software commands as well as a collection of proprietary commands such as **show all**.

9. Gateway Redundancy Troubleshooting Section

9.1. Symptom: Router R4 is not decrementing properly for VRRP.

Analysis and testing:

R4 is configured to be the master VRRP router on VLAN 346. This is verified by the following Cisco IOS **show** command:

```
R4#sh vrrp all
Ethernet0/0 - Group 1
  State is Master
  Virtual IP address is 192.168.1.1
  Virtual MAC address is 0000.5e00.0101
  Advertisement interval is 1.000 sec
  Preemption enabled
  Priority is 120
    Track object 1 state Up decrement 15
  Master Router is 192.168.1.4 (local), priority is 120
  Master Advertisement interval is 1.000 sec
  Master Down interval is 3.531 sec
```

R4 is tracking the status of interface E0/2:

```
R4#show track
Track 1
  Interface Ethernet0/2 line-protocol
  Line protocol is Up
    3 changes, last change 07:48:26
  Tracked by:
    VRRP Ethernet0/0 1
R4#
```

However, even given this configuration state, when the tracked interface of R4 is shut down, R4 retains the VRRP state of master.

```
R4#configure t
R4(config)#inte E0/2
R4(config-if)#shut
R4(config-if)#end
R4#sh vrrp all
Ethernet0/0 - Group 1
  State is Master
  Virtual IP address is 192.168.1.1
  Virtual MAC address is 0000.5e00.0101
  Advertisement interval is 1.000 sec
  Preemption enabled
  Priority is 105 (cfgd 120)
    Track object 1 state Down decrement 15
  Master Router is 192.168.1.4 (local), priority is 105
  Master Advertisement interval is 1.000 sec
  Master Down interval is 3.531 sec
```


Likely cause: A VRRP decrement setting is misconfigured.

Either a VRRP decrement setting on R4 is configured to be too great or the VRRP decrement setting on R3 is configured to be too minimal. You should check the configuration of both routers:

```
R3#sh run inte E0/0
interface Ethernet0/0
 ip address 192.168.1.3 255.255.255.248
…
vrrp 1 ip 192.168.1.1
end

R4#sh run inte E0/0
interface Ethernet0/0
 ip address 192.168.1.4 255.255.255.248
…
vrrp 1 ip 192.168.1.1
 vrrp 1 priority 120
 vrrp 1 track 1 decrement 15
end
```

As you can see, R3 is configured with the default VRRP settings. Next, determine what the default VRRP priority of R3 is:

```
R3#sh vrrp all
Ethernet0/0 - Group 1
  State is Backup
  Virtual IP address is 192.168.1.1
  Virtual MAC address is 0000.5e00.0101
  Advertisement interval is 1.000 sec
  Preemption enabled
  Priority is 100
  Master Router is 192.168.1.4, priority is 105
  Master Advertisement interval is 1.000 sec
  Master Down interval is 3.609 sec (expires in 3.057 sec)
```

You see that the default VRRP priority is 100. This value is not high enough for R3 to assume the role of VRRP master if the tracked interface on R4 goes down. R3 as backup should have a priority above 105.

Resolution: One possible solution is to set the VRRP priority above 105 on R3.

Enter in the following configuration command on router R3:

```
R3#conf t
R3(config)#inte E0/0
R3(config-if)#vrrp 1 priority 110
R3(config-if)#end
```

Now, go to R4 and check to see whether R3 takes over as the VRRP master router when the tracked interface on R4 attains a down state:

```
R4(config-if)#inte E0/2
R4(config-if)#shut
```

```
Aug 18 16:55:27.102: %TRACKING-5-STATE: 1 interface Se0/0/0 line-protocol Up->Down
R4(config-if)#
Aug 18 16:55:29.102: %LINK-5-CHANGED: Interface Serial0/0/0, changed state to
administratively down

Aug 18 16:55:30.102: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0,
changed state to down
Aug 18 16:55:30.494: %VRRP-6-STATECHANGE: E0/0 Grp 1 state Master -> Backup
R4
```

As you can see, R4 is no longer the VRRP master. It is now the VRRP backup. R3 is now the VRRP master:

```
R3#sh vrrp
Ethernet0/0 - Group 1
  State is Master
  Virtual IP address is 192.168.1.1
  Virtual MAC address is 0000.5e00.0101
  Advertisement interval is 1.000 sec
  Preemption enabled
  Priority is 110
  Master Router is 192.168.1.3 (local), priority is 110
  Master Advertisement interval is 1.000 sec
  Master Down interval is 3.570 sec
```

9.2. Symptom: SW3 does not use the VRRP address as its default gateway.

Analysis and testing:

The scenario requires that SW3 use the VRRP address as its default gateway. To verify, run **show ip route** on SW3:

```
SW3#show ip route
Default gateway is 192.168.1.4

Host               Gateway           Last Use    Total Uses  Interface
ICMP redirect cache is empty
```

Resolution: Set the default gateway on SW3 to IP address 192.168.1.1 .

Enter the following configuration command on SW3:

```
SW3(config)#ip default-gateway 192.168.1.1
SW3#

SW3#ping 10.10.10.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 50/249/1048 ms
SW3#
```

This task was not technically challenging; it was designed to test the thoroughness of your verification procedures.

Note	The Mentor Guide engine in the web portal can help you use Cisco IOS Software commands to see a comprehensive view of the configuration for a specific section. With the Mentor Guide engine, you can enter more than 1000 Cisco IOS Software commands as well as a collection of proprietary commands such as **show all**.