360RSW05-WKB-TA06

Cisco 360 CCIE R&S Exercise Workbook Introduction

The Cisco 360 CCIE® R&S Exercise Workbook contains 20 challenging scenarios at the 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 6 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 6 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 credit for a subsection, you must fully complete the subsection per requirements. You will *not* receive partial credit for partially completed subsections.
- IPv4 subnets displayed in the IPv4 IGP diagram are /24 networks of 192.168.0.0, except for CustomerA VRF, which is 172.16.0.0.
- *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 the instructions explicitly specify otherwise.
- Unless the instructions explicitly specify otherwise, addresses and networks that are advertised in the Border Gateway Protocol (BGP) section need to be reachable by all BGP routers but do not have to be reachable by routers that use only interior gateway protocol (IGP).
- Use conventional routing algorithms only, unless the instructions specify otherwise.

- Do not create new interfaces to fulfill IGP requirements, and do not summarize unless you are 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 scenario IPv4 IGP diagram belong to networks 192.168.0.0, 172.16.0.0, and 10.10.10.0.

All IP addresses in this lab belong to the 192.168.0.0, 172.16.0.0, and 10.10.10.0 address space, except for prefixes that are explicitly specified as being part of a different IP space.

Advertise loopback interfaces with their original masks.

The original mask is the mask configured on the loopback interface. OSPF treats loopback interfaces as host routes by default 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 interface. You need to provide a solution such as changing the OSPF network type or summarizations.

All IP addresses that are involved in this scenario must be reachable.

This is a key goal to observe. It requires that all your IGPs and your routing policy tasks 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 only 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 and to be able to forward traffic to the addresses that are known via BGP.

Use conventional routing algorithms.

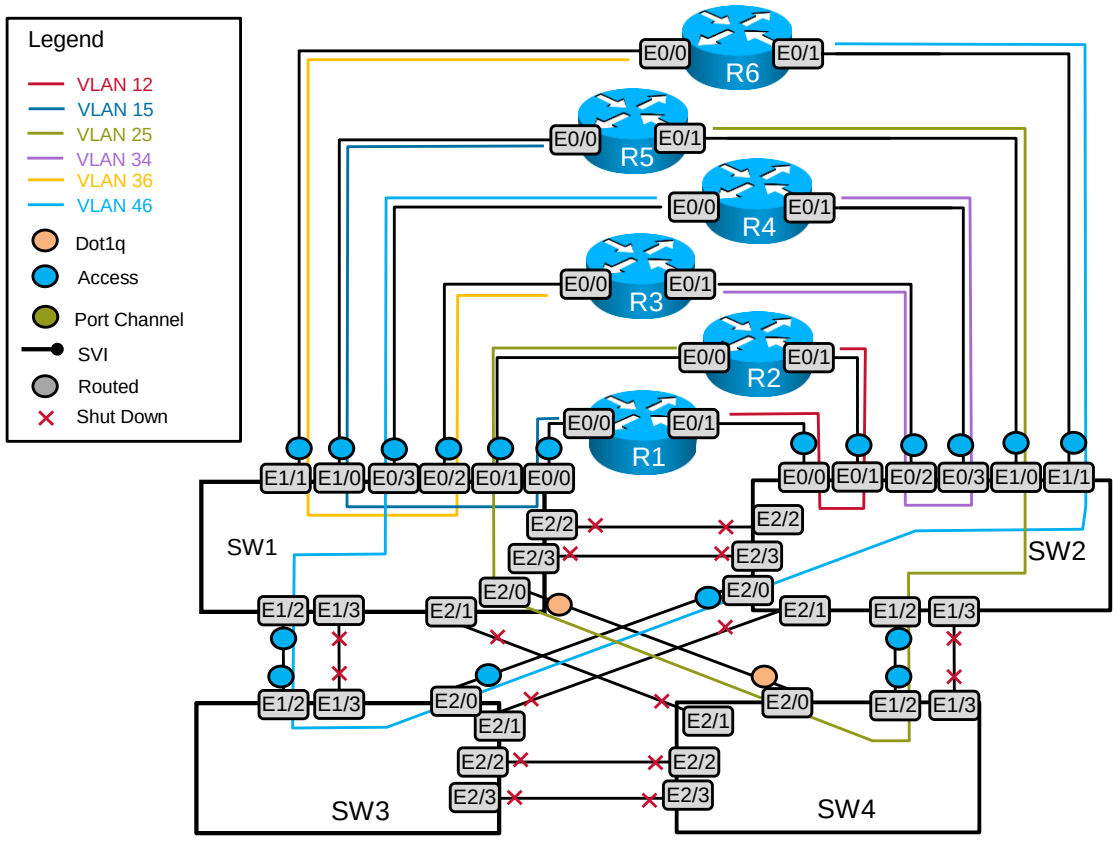
This restriction prevents you from solving any problems by configuring policy routing. At the heart of this restriction is the interpretation of "conventional routing algorithms." Although this phrase can be interpreted in different ways, this interpretation is applied in this workbook:

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

1. Switched Network Troubleshooting Section

The VLAN propagation diagram is reproduced here for easy reference.

VLAN Propagation Diagram



1.1. Symptom: R5 and R6 cannot ping all connected IP addresses.

Analysis and testing:

You pinged the connected networks on R5 and R6 and found that R5 cannot ping the IP address 192.168.10.4 of R2 on VLAN 25. Also, R6 cannot ping the IP address 192.168.10.10 of R4 on VLAN 46.

```
R5#ping 255.255.255.255
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:
.
Reply to request 1 from 192.168.10.2, 1 ms
Reply to request 2 from 192.168.10.2, 1 ms
Reply to request 3 from 192.168.10.2, 1 ms
Reply to request 4 from 192.168.10.2, 1 ms
R5#
```


Note that R5 receives the ping replies from R1 but not from R2.

```
R6#ping 255.255.255.255
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:

Reply to request 0 from 192.168.10.8, 1 ms
Reply to request 1 from 192.168.10.8, 1 ms
Reply to request 2 from 192.168.10.8, 1 ms
Reply to request 3 from 192.168.10.8, 1 ms
Reply to request 4 from 192.168.10.8, 1 ms
R6#
```

Note that R6 receives the ping replies from R3 but not from R4.

You can start your troubleshooting either from R5 or from R6. R5 is used as a starting point in this answer key. Verify a link between R5 and SW2:

```
R5#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW1              Eth 0/0            136             R S   Linux Uni Eth 1/0
SW2              Eth 0/1            138             R S   Linux Uni Eth 1/0
R5#
```

Note that R5 shows an operational connection to SW2.

Move to SW2. You can see periodic messages on SW2 similar to the following output:

```
SW2#
*Jun 15 19:25:16.483: %CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered
on Ethernet2/0 (25), with SW3 Ethernet2/0 (46).
SW2#
*Jun 15 19:25:18.458: %CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered
on Ethernet1/2 (46), with SW4 Ethernet1/2 (25).
```

Note that SW2 notifies you about the VLAN 25 and VLAN 46 mismatch on the Ethernet2/0 and the Ethernet1/2 interfaces.

Verify the VLAN configuration on SW2:

```
SW2#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et1/3, Et2/1, Et2/2, Et2/3
12   VLAN0012                         active    Et0/0, Et0/1
15   VLAN0015                         active
25   VLAN0025                         active    Et1/0, Et2/0
34   VLAN0034                         active    Et0/2, Et0/3
36   VLAN0036                         active
46   VLAN0046                         active    Et1/1, Et1/2
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW2#
```

Note that according to the VLAN distribution diagram, VLAN 25 should be configured on the E1/0 and E1/2 interfaces, instead of the E1/0 and E2/0 interfaces as displayed in the output. VLAN 46

should be configured on the E1/1 and E2/0 interfaces, instead of the E1/1 and E1/2 interfaces as displayed in the output.

Resolution: Fix the VLAN 25 and VLAN 46 port assignments on SW2.

Configure the Ethernet1/2 and Ethernet2/0 with the correct VLAN numbers:

```
SW2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#interface Ethernet1/2
SW2(config-if)#switchport access vlan 25
SW2(config-if)#interface Ethernet2/0
SW2(config-if)# switchport access vlan 46
SW2(config-if)#end
SW2#
```

1.2. Symptom: The lab requirements state that only the necessary VLANs should be created on each switch.

Analysis and testing:

If you take the time to create a diagram like the one above, the VLANs required for each switch become obvious. SW2 is the only switch that needs VLANs 12 and 34, and SW1 is the only switch that needs VLANs 15 and 36. The diagram makes it clear that SW3 requires only one VLAN, which is VLAN 46, and SW4 requires only one VLAN, which is VLAN 25.

If you issue the command **show vlan brief** on each switch, however, you will find that all of the VLANs have been created on each switch:

```
SW1#show vlan brie

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et1/3, Et2/1, Et2/2, Et2/3
12   VLAN0012                         active
15   VLAN0015                         active    Et0/0, Et1/0
25   VLAN0025                         active    Et0/1
34   VLAN0034                         active
36   VLAN0036                         active    Et0/2, Et1/1
46   VLAN0046                         active    Et0/3, Et1/2
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW1#

SW2#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et1/3, Et2/1, Et2/2, Et2/3
12   VLAN0012                         active    Et0/0, Et0/1
15   VLAN0015                         active
25   VLAN0025                         active    Et1/0, Et1/2
34   VLAN0034                         active    Et0/2, Et0/3
36   VLAN0036                         active
46   VLAN0046                         active    Et1/1, Et2/0
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
```



```
1004 fddinet-default                act/unsup
1005 trnet-default                  act/unsup
SW2#

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, Et2/1
                                                 Et2/2, Et2/3
12   VLAN0012                          active
15   VLAN0015                          active
25   VLAN0025                          active
34   VLAN0034                          active
36   VLAN0036                          active
46   VLAN0046                          active    Et1/2, Et2/0
1002 fddi-default                      act/unsup
1003 token-ring-default                act/unsup
1004 fddinet-default                   act/unsup
1005 trnet-default                     act/unsup
SW3#

SW4#show vlan brie

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Et0/0, Et0/1, Et0/2, Et0/3
                                                 Et1/0, Et1/1, Et1/3, Et2/1
                                                 Et2/2, Et2/3
12   VLAN0012                          active
15   VLAN0015                          active
25   VLAN0025                          active    Et1/2
34   VLAN0034                          active
36   VLAN0036                          active
46   VLAN0046                          active
1002 fddi-default                      act/unsup
1003 token-ring-default                act/unsup
1004 fddinet-default                   act/unsup
1005 trnet-default                     act/unsup
SW4#
```

Resolution: Remove unnecessary VLANs from the switches.

Remove VLANs 12 and 34 from SW1, VLANs 15 and 36 from SW2, all but VLAN 46 from SW3, and all but VLAN 25 from SW4. The solution is easily implemented; the challenge for this task is organizing the available information in such a way that the Layer 2 topology is clear.

```
SW1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#no vlan 12,34
SW1(config)#end
SW1#

SW2# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#no vlan 15,36
SW2(config)#end
SW2#

SW3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW3(config)#no vlan 12,15,25,34,36
SW3(config)#end
SW3#

SW4#conf t
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
SW4(config)#no vlan 12,15,34,36,46
SW4(config)#end
SW4
```

1.3. Symptom: Port Eth2/3 is up between SW1 and SW2.

Analysis and testing:

According to the "VLAN Propagation" diagram, SW1 and SW2 should not maintain any direct connections with each other.

```
SW1#sh cdp neighbors | include SW2
SW2              Eth 2/3           170              R S   Linux Uni Eth 2/3
SW1#
```

This port must be shut down.

Likely cause: Neither of the Eth2/3 ports on SW1 and SW2 were administratively shut down.

This issue can be clearly demonstrated in two ways with two instances of the Cisco IOS **show** command:

```
SW1#show interfaces status | include Et2/3
Et2/3                     connected    trunk         auto   auto unknown
SW1#

SW2#show interfaces status | include Et2/3
Et2/3                     connected    trunk         auto   auto unknown
SW2#
```

Resolution: Administratively shut down the Et2/3 ports on SW1 and SW2.

This resolution can be accomplished in the following manner:

```
SW1#conf t

SW1(config)#inte Et2/3
SW1(config-if)#shutdown
SW1(config-if)#end

SW2#configure t
SW2(config)#inte Et2/3
SW2(config-if)#shutdown
SW2(config-if)#end
```

Verify this configuration with the following **show** command:

```
SW1#show interfaces status | include Et2/3
Et2/3                     disabled     1            auto   auto unknown

SW2#show interfaces status | include Et2/3
Et2/3                     disabled     1            auto   auto unknown
```


You can see that these ports are now in a disabled state.

Verify the connectivity on R5 and R6 again:

```
R5#ping 255.255.255.255
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:

Reply to request 0 from 192.168.10.2, 1 ms
Reply to request 1 from 192.168.10.2, 1 ms
Reply to request 1 from 192.168.10.4, 2 ms
Reply to request 2 from 192.168.10.2, 1 ms
Reply to request 2 from 192.168.10.4, 1 ms
Reply to request 3 from 192.168.10.2, 1 ms
Reply to request 3 from 192.168.10.4, 1 ms
Reply to request 4 from 192.168.10.2, 1 ms
Reply to request 4 from 192.168.10.4, 1 ms
R5#
```

Note that R5 can now ping R2 on VLAN 25.

```
R6#ping 255.255.255.255
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:

Reply to request 0 from 192.168.10.8, 1 ms
Reply to request 0 from 192.168.10.10, 1 ms
Reply to request 1 from 192.168.10.8, 1 ms
Reply to request 1 from 192.168.10.10, 1 ms
Reply to request 2 from 192.168.10.8, 1 ms
Reply to request 2 from 192.168.10.10, 1 ms
Reply to request 3 from 192.168.10.8, 1 ms
Reply to request 3 from 192.168.10.10, 1 ms
Reply to request 4 from 192.168.10.8, 1 ms
Reply to request 4 from 192.168.10.10, 1 ms
R6#
```

Note that R5 can now ping R4 on VLAN 46.

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**. 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: OSPF adjacencies are not forming between R6 and each of its neighbors.

Analysis and testing:

Verify the OSPF neighbors on R6:

```
R6#show ip ospf neighbor
```

```
Neighbor ID     Pri   State           Dead Time   Address         Interface
N/A               0   ATTEMPT/DROTHER     -       192.168.10.10   Ethernet0/1
N/A               0   ATTEMPT/DROTHER     -       192.168.10.8    Ethernet0/0
R6#
```

A suggested opening verification command to troubleshoot any OSPF problem is the **show ip ospf interface** command. It verifies that OSPF is configured on a specific interface. If it is not, then OSPF operations on that interface will not work. Therefore, begin your troubleshooting process with this command on R6:

```
R6#show ip ospf interface Et0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.10.9/31, Area 0
  Process ID 1, Router ID 6.6.6.6, Network Type NON_BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 6.6.6.6, Interface address 192.168.10.9
  No backup designated router on this network
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:03
  Supports Link-local Signaling (LLS)
  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#show ip ospf interface Et0/1
Ethernet0/1 is up, line protocol is up
  Internet Address 192.168.10.11/31, Area 0
  Process ID 1, Router ID 6.6.6.6, Network Type NON_BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 6.6.6.6, Interface address 192.168.10.11
  No backup designated router on this network
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
```

You can see that OSPF is enabled on these interfaces, but no neighbors are discovered. After troubleshooting VLANs 36 and 46, you know that the neighbors of R6 can be pinged:

```
R6#ping 192.168.10.8

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms

R6#ping 192.168.10.10

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```


Both of R6's OSPF neighbors can be pinged, but they cannot form a neighbor relationship.

Likely cause: R6 and its OSPF neighbors have an OSPF network type mismatch among them.

Upon closer inspection of the **show ip ospf interface** display on R6, you see that the OSPF network type on its Fast Ethernet interfaces is nonbroadcast:

```
R6#show ip ospf interface Et0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.10.9/31, Area 10
  Process ID 1, Router ID 6.6.6.6, Network Type NON_BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 6.6.6.6, Interface address 192.168.10.9
  No backup designated router on this network
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:03
  Supports Link-local Signaling (LLS)
  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#show ip ospf interface Et0/1
Ethernet0/1 is up, line protocol is up
  Internet Address 192.168.10.11/31, Area 10
  Process ID 1, Router ID 6.6.6.6, Network Type NON_BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 6.6.6.6, Interface address 192.168.10.11
  No backup designated router on this network
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
```

By default, this configuration will fail to match the other neighboring routers attached to the same Fast Ethernet segments that R6 is attached to:

```
R3#show ip ospf interface Et0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.10.8/31, Area 10
  Process ID 1, Router ID 3.3.3.3, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 3.3.3.3, Interface address 192.168.10.8
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:08
  Supports Link-local Signaling (LLS)
  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)
```

```
R4#show ip ospf interface Et0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.10.10/31, Area 10
  Process ID 1, Router ID 4.4.4.4, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 4.4.4.4, Interface address 192.168.10.10
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:06
  Supports Link-local Signaling (LLS)
  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)
```

You can see that the problem is that R6's neighbors are configured with an OSPF network type that is different from R6. Furthermore, the scenario explicitly states that the originally configured OSPF network types cannot be changed.

Resolution: Change the OSPF hello interval on R6.

The broadcast and nonbroadcast OSPF network types are compatible in that they both model the link as a transit link, both use a designated router (DR), and both network types use a network link-state advertisement (LSA). However the two network types use different hello and dead timers. These timers are advertised in hello packets and must match if two routers are to become adjacent. The problem can be resolved by adjusting the OSPF hello timer so that it matches the same setting on both R3 and R4. This will automatically adjust the dead timer.

```
R6#sh run inte Et0/0
Building configuration...
Current configuration : 190 bytes
!
interface Ethernet0/0
 ip address 192.168.10.9 255.255.255.254
 ip ospf network non-broadcast
 ip ospf hello-interval 10
end

R6#sh run inte Et0/1
Building configuration...
Current configuration : 191 bytes
!
interface Ethernet0/1
 ip address 192.168.10.11 255.255.255.254
 ip ospf network non-broadcast
 ip ospf hello-interval 10
 duplex auto
end
```

Once these commands are entered, execute the **show ip ospf interface** command again. The output shows you that R6 now sees its OSPF neighbors:

```
R6#show ip ospf interface Et0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.10.9/31, Area 10
  Process ID 1, Router ID 6.6.6.6, Network Type NON_BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 6.6.6.6, Interface address 192.168.10.9
  Backup Designated router (ID) 3.3.3.3, Interface address 192.168.10.8
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
```



```
    Hello due in 00:00:08
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 2
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 3.3.3.3  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)

R6#show ip ospf interface Et0/1
Ethernet0/1 is up, line protocol is up
  Internet Address 192.168.10.11/31, Area 10
  Process ID 1, Router ID 6.6.6.6, Network Type NON_BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 6.6.6.6, Interface address 192.168.10.11
  Backup Designated router (ID) 4.4.4.4, Interface address 192.168.10.10
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:04
  Supports Link-local Signaling (LLS)
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 4
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 4.4.4.4  (Backup Designated Router)
  Suppress hello for 0 neighbor(s)
```

This output is the information that should display when OSPF has successfully discovered its neighbors. This troubleshooting ticket is successfully resolved.

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. IPv4 EIGRP Troubleshooting Section

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

Analysis and testing:

Verify the Enhanced Interior Gateway Routing Protocol (EIGRP) neighbors on R5:

```
R5#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(10)
R5#
```

Note that R5 is not forming any EIGRP neighbor relationships.

As you know from the “Switched Network Troubleshooting” section, R5 can ping its EIGRP neighbors on both VLANs 15 and 25:

```
R5#ping 192.168.10.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.2, timeout is 2 seconds:
```

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R5#ping 192.168.10.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
R5#
```

The output of the **show ip eigrp interfaces** command indicates that R5 has EIGRP enabled on both the VLAN 15 and VLAN 25 interfaces:

```
R5#show ip eigrp interfaces
IP-EIGRP interfaces for process 10

                        Xmit Queue   Mean   Pacing Time   Multicast    Pending
Interface        Peers  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Et0/0              0        0/0         1       0/1           50           0
Et0/1              0        0/0         1       0/1           50           0
Lo105              0        0/0         0       0/1            0           0
R5#
```

However, R5 does not see R1 or R2 as Enhanced Interior Gateway Routing Protocol (EIGRP) peers. To troubleshoot this problem, you can enable the **debug eigrp packet hello** utility. It is a useful learning and diagnostic tool for EIGRP. However, it creates a lot of output. Therefore, as with all Cisco IOS Software debug utilities, use this tool with extreme caution. To limit the use of this debug tool to a single interface, use the conditional debug utility **debug interface.** This will limit the debug output for **debug eigrp packet hello** to only interface Eth0/0. Begin by applying these debug utilities only on R5:

```
R5#debug interface Et0/0
Condition 1 set

R5#debug eigrp packets hello
EIGRP Packets debugging is on
    (HELLO)
R5#
*Jun  16 17:10:02.494: EIGRP: Sending HELLO on Ethernet0/0
*Jun  16 17:10:02.494:   AS 10, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Jun  16 17:10:04.178: EIGRP: pkt authentication key id = 1, key not defined
```

As the **debug eigrp packet hello** output clearly displays, R5 is receiving debug EIGRP messages with an authentication key of 1, and the debug output is stating that R5 does not have this key.

Now that the **debug eigrp packet hello** command has displayed what you need, disable this debug utility as well as the **debug interface** utility. Please note that when you disable a conditional debug utility like **debug interface**, the **undebug all** command will not suffice. You must explicitly disable a conditional debug utility like **debug interface** with a command such as **debug interface Et0/0.** When you do this and the specific debug tool is the last conditional debug utility applied, the Cisco IOS Software will return with the following prompt:

```
R5#undebug interface Et0/0
This condition is the last interface condition set.
```



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

Proceed with removal? [yes/no]: y
Condition 1 has been removed
```

Answer "yes" to the prompt and the conditional debug utility will be removed. Verify that all debug utilities have been disabled with the **show debug** command.

Likely cause: R5 is configured with an EIGRP authentication key uniquely identified with the number 1.

To verify this as the problem, examine the current EIGRP configuration on R5:

```
R5#sh run | section eigrp
key chain eigrp
 key 11
   key-string san-fran
 ip authentication mode eigrp 10 md5
 ip authentication key-chain eigrp 10 eigrp
 ip authentication mode eigrp 10 md5
 ip authentication key-chain eigrp 10 eigrp
router eigrp 10
 network 192.168.10.0
 auto-summary
```

You can see that there is no EIGRP key ID with the value of 1 assigned to it; there is only a key numbered 11.

Resolution: Configure an EIGRP authentication key with the key ID of 1 to match the key ID of its EIGRP neighbors.

When the EIGRP authentication key IDs match, R5 will form EIGRP neighbor relationships with both R1 and R2:

```
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#key chain eigrp
R5(config-keychain)#key 1
R5(config-keychain-key)# key-string san-fran
R5(config-keychain-key)#no  key 11
R5(config-keychain)#end
R5#

R5#show ip eigrp nei
IP-EIGRP neighbors for process 10
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
1   192.168.10.2            Et0/0             13 02:10:00  816  4896  0  5
0   192.168.10.4            Et0/1             13 02:10:00  815  4890  0  5
```

3.2. Symptom: R1 and R2 are not forming an EIGRP neighbor relationship.

Analysis and testing:

The analysis begins with the following Cisco IOS **show** command:

```
R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(10)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
0   192.168.10.3            Et0/0                    13 00:01:11   15   100  0  7
R1#


R1#sh run inte Et0/1
Building configuration...

interface Ethernet0/1
 ip address 192.168.10.0 255.255.255.254
 ip authentication mode eigrp 10 md5
 ip authentication key-chain eigrp 10 eigrp
 delay 16777210
 ipv6 enable
 ipv6 eigrp 125
end

R1#
```

You can see that EIGRP authentication is configured on the R1 interface attached to VLAN 12, the VLAN it shares with its unreachable EIGRP neighbor R2. Now run the same Cisco IOS **show** command on R2:

```
R2#show run int e0/1
Building configuration...

Current configuration : 139 bytes
!
interface Ethernet0/1
 ip address 192.168.10.1 255.255.255.254
 ip authentication mode eigrp 10 md5
 delay 16777210
 ipv6 eigrp 125
end

R2#
```

As you can see, the following command is missing on R2: **ip authentication key-chain eigrp 10 eigrp**. Without this command, EIGRP authentication will not work.

Resolution: Add the EIGRP** ip authentication key-chain **command under the Et0/1 interface of R2.

This task is done in the following manner:

```
R2#conf t
R2(config)#inte Et0/1
R2(config-if)#ip authentication key-chain eigrp 10 eigrp

*Jun  16  16:56:22.273:  %SYS-5-CONFIG_I:  Configured  from  console  by  console
(cierswbv5-te-lab06-sc, SJ)
R2#
*Jun  16  16:56:23.753:  %DUAL-5-NBRCHANGE:  EIGRP-IPv4  10:  Neighbor  192.168.10.0
(Ethernet0/1) is up: new adjacency
R2#
```


As soon as this command is entered, the Cisco IOS console announces that the EIGRP keychain has changed. If the correct keychain is added, the EIGRP neighbor relationship is rapidly formed, as indicated by the second console message above. This can be further verified by the following Cisco IOS **show** command:

```
R2#show ip eigrp neighbors
IP-EIGRP neighbors for process 10
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
1   192.168.10.0            Et0/1             11 00:00:39 1021  5000  0  9
0   192.168.10.5            Et0/0             14 02:19:11    2   200  0  9
```

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: RIP routes are not shown in the routing table on R1.

Analysis and testing:

Below, you see the output of the Routing Information Protocol (RIP) routing table from R1:

```
R1#show ip route rip
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

R1#
```

You checked R2, R3, and R4 and noticed that these routers learn RIP routes from R1. Here is an example from R2:

```
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

      192.168.10.0/24 is variably subnetted, 9 subnets, 3 masks
R        192.168.10.0/24 [120/3] via 10.10.10.4, 00:00:03, Ethernet0/2
```

```
                        [120/3] via 10.10.10.3, 00:00:28, Ethernet0/2
                        [120/3] via 10.10.10.1, 00:00:08, Ethernet0/2
R2#
```

Note that R1 sends an update for the prefix 192.168.10.0/24. But R1 does not learn any RIP prefixes from R2, R3, or R4.

Run the **debug ip rip** command on R1:

```
R1#debu ip rip
RIP protocol debugging is on
R1#
*Jun  15  20:27:48.577:  RIP:  sending  v2  update  to  224.0.0.9  via  Ethernet0/2
(10.10.10.1)
*Jun 15 20:27:48.577: RIP: build update entries
*Jun 15 20:27:48.577:   192.168.10.0/24 via 0.0.0.0, metric 3, tag 0
R1#
*Jun  15  20:28:17.190:  RIP:  sending  v2  update  to  224.0.0.9  via  Ethernet0/2
(10.10.10.1)
*Jun 15 20:28:17.190: RIP: build update entries
*Jun 15 20:28:17.190:   192.168.10.0/24 via 0.0.0.0, metric 3, tag 0
R1#

R1#u all
All possible debugging has been turned off
R1#
```

Note that R1 is sending updates on the Ethernet0/2 interface, but R1 is not receiving the updates.

Verify the RIP configuration on R1:

```
R1#show ip protocols | sec "rip"
Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 16 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: eigrp 10, rip
  Default version control: send version 2, receive version 2
    Interface              Send  Recv  Triggered RIP  Key-chain
    Ethernet0/2            2     2
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
     10.0.0.0
    10.0.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
  Distance: (default is 120)
R1#
```

Note that the Ethernet0/2 interface is not passive and is configured to send and receive the RIP version 2 updates. What is blocking the incoming RIP updates on the Ethernet0/2 interface?

Likely cause: There is a misconfigured access control list applied to the Ethernet0/2 interface on R1.

Verify the access list configuration on R1:

```
R1#show running-config int e0/2 | inc 111
ip access-group 111 in
R1#

R1#show access-lists
Standard IP access list 1
```



```
    10 permit 192.168.10.0
Extended IP access list 111
    10 deny ip any host 224.0.0.9 (596 matches)
    20 permit ip any any (212 matches)
Extended IP access list 145
    10 permit icmp any any dscp 45
    20 permit ip any any (1249 matches)
R1#
```

Note that access list 111 shows matches for the packets destined to 224.0.0.9, which is the destination multicast IP address of the RIP version 2 updates.

Resolution: Remove the access list reference from the Ethernet0/2 interface on R1.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int e0/2
R1(config-if)#no ip access-group 111 in
R1(config-if)#end
R1#
```

Verify the RIP table on R1 again:

```
R1#show ip route rip
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      192.168.10.0/24 is variably subnetted, 9 subnets, 3 masks
R        192.168.10.0/24 [120/3] via 10.10.10.4, 00:00:08, Ethernet0/2
                         [120/3] via 10.10.10.3, 00:00:12, Ethernet0/2
                         [120/3] via 10.10.10.2, 00:00:18, Ethernet0/2
R1#
```

Note that the RIP prefixes are now listed in the routing table on R1.

Note To obtain a comprehensive view of the configuration tasks in this section, access the Mentor Guide engine. You can enter more than 1000 Cisco IOS Software commands into the engine, as well as a collection of proprietary commands such as **show all**.

---

5. IPv4 Redistribution Troubleshooting Section

5.1. Symptom: A routing loop may form for packets crossing the 10.10.10.0/29 network.

Analysis and testing:

It is likely that one or more IP addresses in the OSPF domain are unreachable from R1, R2, and R6. It is also possible that one or more IP addresses in the EIGRP domain are unreachable from R3, R4, or R6. This answer key uses R6 as an example. When R6 attempts to reach the

192.168.10.101 address assigned to the Loopback 101 interface of R1, it sends the packet out to router R3 and R4:

```
R6#show ip route 192.168.10.101
Routing entry for 192.168.10.0/24
  Known via "ospf 1", distance 110, metric 20, type extern 2, forward metric 10
  Last update from 192.168.10.10 on Ethernet0/1, 00:41:35 ago
  Routing Descriptor Blocks:
    192.168.10.10, from 4.4.4.4, 00:41:35 ago, via Ethernet0/1
      Route metric is 20, traffic share count is 1
  * 192.168.10.8, from 3.3.3.3, 00:41:40 ago, via Ethernet0/0
      Route metric is 20, traffic share count is 1
R6#
```

R3 sends the packet to R4, and R4 sends the packet to R3, creating a routing loop.

```
R3#show ip route 192.168.10.101
Routing entry for 192.168.10.0/24
  Known via "rip", distance 120, metric 3
  Redistributing via ospf 1, rip
  Advertised by ospf 1 subnets
  Last update from 10.10.10.4 on Ethernet0/2, 00:00:04 ago
  Routing Descriptor Blocks:
  *  10.10.10.4, from 10.10.10.4, 00:00:04 ago, via Ethernet0/2
      Route metric is 3, traffic share count is 1
    10.10.10.2, from 10.10.10.2, 00:00:14 ago, via Ethernet0/2
      Route metric is 3, traffic share count is 1
    10.10.10.1, from 10.10.10.1, 00:00:19 ago, via Ethernet0/2
      Route metric is 3, traffic share count is 1
R3#

R4#show ip route 192.168.10.101
Routing entry for 192.168.10.0/24
  Known via "rip", distance 120, metric 3
  Redistributing via ospf 1, rip
  Advertised by ospf 1 subnets
  Last update from 10.10.10.1 on Ethernet0/2, 00:00:07 ago
  Routing Descriptor Blocks:
  * 10.10.10.3, from 10.10.10.3, 00:00:18 ago, via Ethernet0/2
      Route metric is 3, traffic share count is 1
    10.10.10.2, from 10.10.10.2, 00:00:29 ago, via Ethernet0/2
      Route metric is 3, traffic share count is 1
    10.10.10.1, from 10.10.10.1, 00:00:07 ago, via Ethernet0/2
      Route metric is 3, traffic share count is 1
R4#

R6#traceroute 192.168.10.101
Type escape sequence to abort.
Tracing the route to 192.168.10.101
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.10.8 0 msec
    192.168.10.10 0 msec
    192.168.10.8 1 msec
  2 10.10.10.3 0 msec
    10.10.10.4 0 msec
    10.10.10.3 1 msec
  3 10.10.10.3 0 msec
    10.10.10.4 0 msec
    10.10.10.3 1 msec
  4 10.10.10.3 0 msec
    10.10.10.4 1 msec
    10.10.10.3 0 msec
<skipped for brevity>
```


Likely cause: All RIPv2 routers are sourcing a 192.168.10.0/24 summary.

Given the topology of this specific scenario and the IP address assignment scheme, each RIP router will have three routes to a 192.168.10.0/24 summary route.

This was the state of the routes on R1 and R2 at the moment the loop mentioned above occurred:

```
R3#show ip route 192.168.10.101
Routing entry for 192.168.10.0/24
  Known via "rip", distance 120, metric 3
  Redistributing via ospf 1, rip
  Advertised by ospf 1 subnets
  Last update from 10.10.10.4 on Ethernet0/2, 00:00:04 ago
  Routing Descriptor Blocks:
  *  10.10.10.4, from 10.10.10.4, 00:00:04 ago, via Ethernet0/2
       Route metric is 3, traffic share count is 1
     10.10.10.2, from 10.10.10.2, 00:00:14 ago, via Ethernet0/2
       Route metric is 3, traffic share count is 1
     10.10.10.1, from 10.10.10.1, 00:00:19 ago, via Ethernet0/2
       Route metric is 3, traffic share count is 1
R3#

R4#show ip route 192.168.10.101
Routing entry for 192.168.10.0/24
  Known via "rip", distance 120, metric 3
  Redistributing via ospf 1, rip
  Advertised by ospf 1 subnets
  Last update from 10.10.10.1 on Ethernet0/2, 00:00:07 ago
  Routing Descriptor Blocks:
  * 10.10.10.3, from 10.10.10.3, 00:00:18 ago, via Ethernet0/2
       Route metric is 3, traffic share count is 1
     10.10.10.2, from 10.10.10.2, 00:00:29 ago, via Ethernet0/2
       Route metric is 3, traffic share count is 1
     10.10.10.1, from 10.10.10.1, 00:00:07 ago, via Ethernet0/2
       Route metric is 3, traffic share count is 1
R4#
```

The particular pattern you observe will depend on timing.

Resolution: Configure the prefixes learned from the routers on the other end of the 10.10.10.0/29 subnet so that they have a higher administrative distance than the value learned over the Ethernet connections that are outside the RIP domain.

Routes cannot be filtered completely because there is a redundancy requirement in this lab. Instead, the administrative distance can be configured so that the route can still act as a backup if the primary route disappears.

On each RIP router, enter commands similar to those shown here:

```
R1:
router rip
distance 121 10.10.10.2 0.0.0.0 1
!
access-list 1 permit 192.168.10.0

R2:
router rip
distance 121 10.10.10.1 0.0.0.0 1
!
access-list 1 permit 192.168.10.0
```

```
R3:
router rip
distance 121 10.10.10.4 0.0.0.0 1
!
access-list 1 permit 192.168.10.0

R4:
router rip
distance 121 10.10.10.3 0.0.0.0 1
!
access-list 1 permit 192.168.10.0
```

On R1, raise the administrative distance for the summary learned from R2, and on R2, raise the administrative distance for the summary learned from R1. On R3, raise the administrative distance for the summary learned from R4, and on R4, raise the administrative distance for the summary learned from R3.

Each RIP router will then have just two summary routes to prefix 192.168.10.0/24 in its forwarding table; one for each border router in the other routing domain. However, if the primary paths disappear, the newly assigned administrative distance values will install them. In other words, the backup routes are there if they are needed.

After you change the administrative distance, you may need to clear the IP routing tables on R1, R2, R3, and R4 using the **clear ip route *** command.

Now that you seem to have addressed all of the IPv4 unicast issues, you can 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. Note that the IP addresses from the 172.16.0.0 subnet require working BGP for full reachability.

```
tclsh
foreach address {
192.168.10.101
10.10.10.1
192.168.10.2
192.168.10.0
192.168.10.102
10.10.10.2
192.168.10.1
192.168.10.4
192.168.10.103
10.10.10.3
192.168.10.6
192.168.10.8
192.168.10.104
10.10.10.4
192.168.10.7
192.168.10.10
192.168.10.105
192.168.10.3
192.168.10.5
192.168.10.106
192.168.10.11
192.168.10.9
} {ping $address}
```


Note To obtain a comprehensive view of the configuration tasks in this section, access the Mentor Guide engine. You can enter more than 1000 Cisco IOS Software commands into the engine, as well as a collection of proprietary commands such as **show all**.

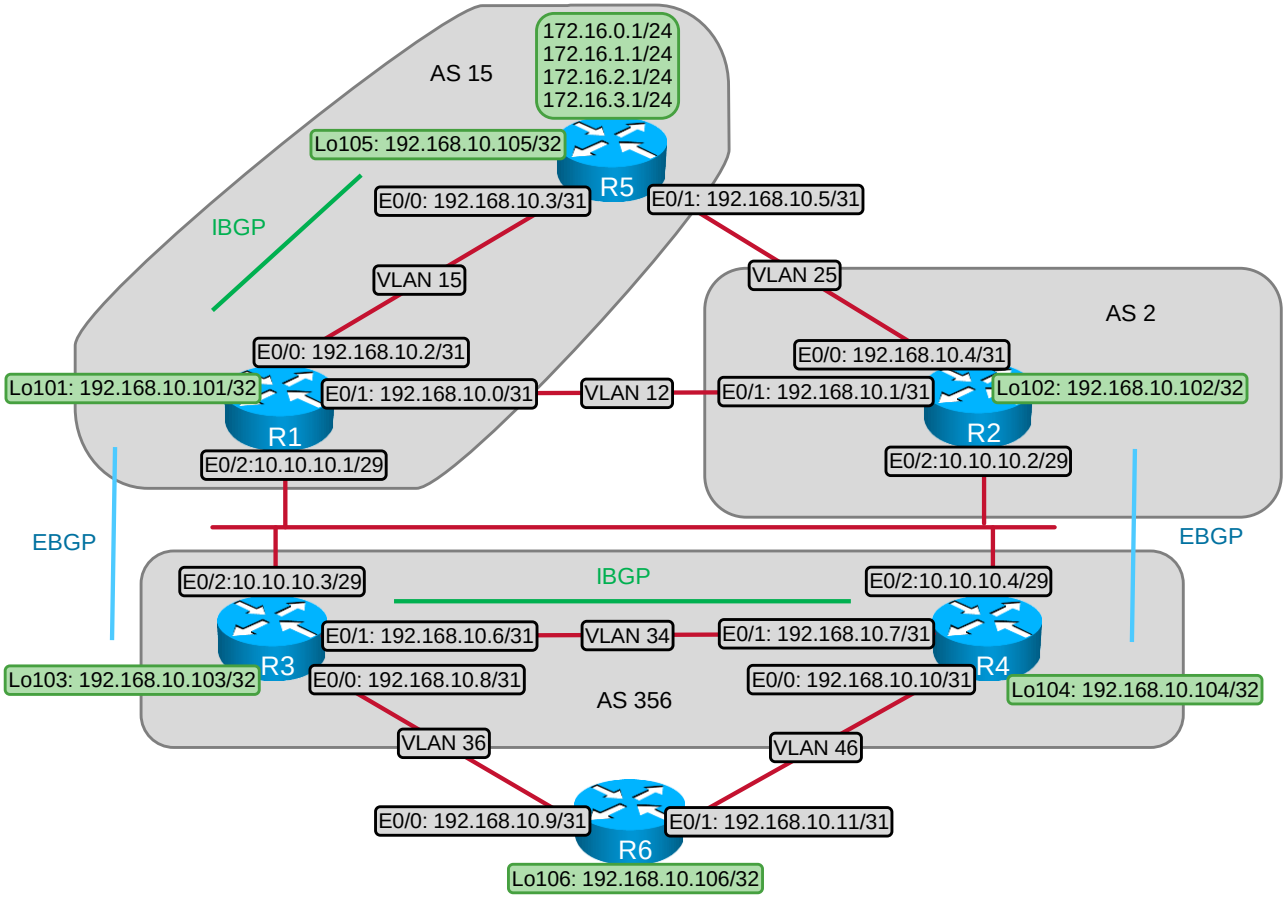
6. BGP Troubleshooting Section

6.1. Symptom: R6 does not have routes from the 172.16.0.0 subnet.

Analysis and testing:

Begin your analysis of BGP by reviewing this BGP topology diagram.

IPv4 BGP Diagram



Notice in the diagram that R6 is not configured as a BGP speaker. Therefore, R6 will receive BGP routes via redistribution of BGP routes into OSPF. When you examine the R6 routing table, you see that there are no BGP routes. The BGP routes all begin with a 172.16.0.0 prefix:

```
R6#sh ip ro
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
```

The gateway of last resort is not set:

```
     192.168.10.0/24 is variably subnetted, 7 subnets, 3 masks
O       192.168.10.103/32 [110/2] via 192.168.10.8, 23:53:25, Ethernet0/0
C       192.168.10.106/32 is directly connected, Loopback106
O       192.168.10.104/32 [110/2] via 192.168.10.10, 23:53:25, Ethernet0/1
O E2    192.168.10.0/24 [110/20] via 192.168.10.10, 23:53:25, Ethernet0/1
                        [110/20] via 192.168.10.8, 23:53:25, Ethernet0/0
O       192.168.10.6/31 [110/2] via 192.168.10.8, 23:53:25, Ethernet0/0
C       192.168.10.10/31 is directly connected, Ethernet0/1
C       192.168.10.8/31 is directly connected, Ethernet0/0
     10.0.0.0/29 is subnetted, 1 subnets
O E2    10.10.10.0 [110/20] via 192.168.10.10, 23:53:27, Ethernet0/1
                   [110/20] via 192.168.10.8, 23:53:27, Ethernet0/0
```

As you can see, none of the 172.16 prefixes are in the R6 routing table.

Likely cause: There is a redistribution configuration error on router R3.

Because R3 is the only External Border Gateway Protocol (EBGP) speaker in AS 356 to AS 15, and it is the originator of all 172.16.*.*/24 BGP learned prefixes, you must review the OSPF redistribution configuration on router R3. First, check the BGP table of R3 to make sure that it has learned the 172.16. BGP routes:

```
R3#sh ip bgp
BGP table version is 5, local router ID is 3.3.3.3
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
*> 172.16.0.0/24    10.10.10.1                             0 15 i
*> 172.16.1.0/24    10.10.10.1                             0 15 i
*> 172.16.2.0/24    10.10.10.1                             0 15 i
*> 172.16.3.0/24    10.10.10.1                             0 15 i
R3#
```

Next, make sure that R3 has these BGP prefixes in its local routing table:

```
R3#sh ip route bgp
     172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
B       172.16.0.0/24 [20/0] via 10.10.10.1, 1d00h
B       172.16.1.0/24 [20/0] via 10.10.10.1, 1d00h
B       172.16.2.0/24 [20/0] via 10.10.10.1, 1d00h
B       172.16.3.0/24 [20/0] via 10.10.10.1, 1d00h
R3#
```

R3 has the BGP routes in its local IP routing table. In particular, check the **redistribute bgp** configuration statement.


```
R3#sh run | section router ospf
router ospf 1
 router-id 3.3.3.3
 log-adjacency-changes
 redistribute rip subnets
 redistribute bgp 356
 network 192.168.10.0 0.0.0.255 area 10
ipv6 router ospf 1
 log-adjacency-changes
```

Resolution: The subnets keyword is missing from the** redistribute bgp **command under the OSPF routing process on router R3.

In order for subnets to be redistributed into OSPF, the **subnets** keyword must be supplied. It is supplied for RIP under the OSPF routing process on R3, but not for BGP. This is underscored by viewing the external routes section of the OSPF database on R3:

```
R3#show ip ospf database | begin Ex
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.10.10.0      4.4.4.4         1392        0x8000002C 0x0085C8 0
192.168.10.0    3.3.3.3         1593        0x8000002C 0x0017DE 0
192.168.10.0    4.4.4.4         1392        0x8000002C 0x00F8F8 0
```

Notice that the RIP external routes are in the database but not the BGP routes. Therefore, add the **subnets** keyword to the OSPF **redistribute bgp** command and then reexamine the OSPF database:

```
R3#conf t

R3(config)#router ospf 1
R3(config-router)#redistribute bgp 356 subnets
R3(config-router)#end

R3#show ip ospf database | begin Ex
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.10.10.0      3.3.3.3         1732        0x8000002C 0x00A3AE 0
10.10.10.0      4.4.4.4         1531        0x8000002C 0x0085C8 0
172.16.0.0      3.3.3.3         472         0x8000002E 0x00ADFC 0
172.16.0.255    3.3.3.3         1237        0x8000002C 0x0002AE 15
172.16.1.0      3.3.3.3         1732        0x8000002C 0x00F6B8 15
172.16.2.0      3.3.3.3         1732        0x8000002C 0x00EBC2 15
172.16.3.0      3.3.3.3         1732        0x8000002C 0x00E0CC 15
192.168.10.0    3.3.3.3         1732        0x8000002C 0x0017DE 0
192.168.10.0    4.4.4.4         1531        0x8000002C 0x00F8F8 0
```

You see the BGP external routes in the OSPF database. Now that they are in the R3 OSPF database, check to see if they are in the R6 OSPF database and IP routing table:

```
R6#show ip ospf database | begin Ex
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.10.10.0      3.3.3.3         1865        0x8000002C 0x00A3AE 0
10.10.10.0      4.4.4.4         1662        0x8000002C 0x0085C8 0
172.16.0.0      3.3.3.3         605         0x8000002E 0x00ADFC 0
```

```
172.16.0.255    3.3.3.3         1370        0x8000002C 0x0002AE 15
172.16.1.0      3.3.3.3         1865        0x8000002C 0x00F6B8 15
172.16.2.0      3.3.3.3         1865        0x8000002C 0x00EBC2 15
172.16.3.0      3.3.3.3         1865        0x8000002C 0x00E0CC 15
192.168.10.0    3.3.3.3         1865        0x8000002C 0x0017DE 0
192.168.10.0    4.4.4.4         1662        0x8000002C 0x00F8F8 0
```

The BGP routes are now in the R6 OSPF database as external routes, as expected.

```
R6#sh ip route | include E2
       E1 - OSPF external type 1, E2 - OSPF external type 2
O E2    192.168.10.0/24 [110/20] via 192.168.10.10, 1d00h, Ethernet0/1
O E2    172.16.0.0/24 [110/1] via 192.168.10.8, 1d00h, Ethernet0/0
O E2    172.16.0.0/16 [110/20] via 192.168.10.8, 1d00h, Ethernet0/0
O E2    172.16.1.0/24 [110/1] via 192.168.10.8, 1d00h, Ethernet0/0
O E2    172.16.2.0/24 [110/1] via 192.168.10.8, 1d00h, Ethernet0/0
O E2    172.16.3.0/24 [110/1] via 192.168.10.8, 1d00h, Ethernet0/0
O E2    10.10.10.0 [110/20] via 192.168.10.10, 1d00h, Ethernet0/1
```

The BGP-originated routes have been installed in the R6 local routing table. Now R6 can forward packets to the routes that originated from BGP.

6.2. Symptom: R2 does not have the BGP routes that originated from AS 15 in its IP routing table.

Analysis and testing:

R2 should learn these routes from the BGP peering with R4, but the output of **show ip bgp** on R4 indicates that no routes have been learned. The output of **show ip bgp summary** indicates, however, that R2 and R4 have formed a BGP peer relationship:

```
R2#show ip bgp summary
BGP router identifier 192.168.10.102, local AS number 2
BGP table version is 1, main routing table version 1

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.10.10.4      4   356      47      47        1    0    0 00:44:22        0
```

Check R4 to see if it has these routes in its BGP table:

```
R4#show ip bgp
BGP table version is 1, local router ID is 192.168.10.104
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
* i172.16.0.0/24    10.10.10.1               0    100      0 15 i
* i172.16.1.0/24    10.10.10.1               0    100      0 15 i
* i172.16.2.0/24    10.10.10.1               0    100      0 15 i
* i172.16.3.0/24    10.10.10.1               0    100      0 15 i
```

R4 has the routes in its table, but they are not preceded by the Best indicator (>), so these prefixes are not eligible for local use or advertisement. To determine the cause, review the details for one of these prefixes:

```
R4#show ip bgp 172.16.1.0
BGP routing table entry for 172.16.1.0/24, version 19
Paths: (1 available, no best path)
```



```
  Not advertised to any peer
  15
    10.10.10.1 from 192.168.10.103 (192.168.10.103)
      Origin IGP, metric 0, localpref 100, valid, internal, not synchronized

R4#sh ip route 172.16.1.0
Routing entry for 172.16.1.0/24
  Known via "ospf 1", distance 122, metric 1
  Tag 15, type extern 2, forward metric 1
  Redistributing via rip
  Advertised by rip metric 3
  Last update from 192.168.10.6 on Ethernet0/1, 00:08:34 ago
  Routing Descriptor Blocks:
  * 192.168.10.6, from 3.3.3.3, 00:08:34 ago, via Ethernet0/1
      Route metric is 1, traffic share count is 1
      Route tag 15
```

You can see that the route is in the R4 BGP table, but it is listed as "not synchronized."

The rule of synchronization states that, for an Internal Border Gateway Protocol (IBGP) learned update to use by BGP, it must be learned by an IGP. This requirement has been fulfilled. as shown in the **show ip route** output for the 172.16.1.0 prefix. The BGP update of 172.16.1.0/24 is learned via the IGP OSPF, as shown by the Cisco IOS **show ip route** output.

Likely cause: The OSPF router ID and BGP router ID are mismatched.

When the rule of synchronization is enabled and the underlying IGP is OSPF, the router ID of the OSPF Autonomous System Boundary Router (ASBR) must match the router ID of the advertising IBGP speaker. As you can see from reexamining the following two Cisco IOS **show** command displays, this is clearly not the case:

```
R4#show ip bgp 172.16.1.0
BGP routing table entry for 172.16.1.0/24, version 19
Paths: (1 available, no best path)
  Not advertised to any peer
  15
    10.10.10.1 from 192.168.10.103 (192.168.10.103)
      Origin IGP, metric 0, localpref 100, valid, internal, not synchronized

R4#sh ip route 172.16.1.0
Routing entry for 172.16.1.0/24
  Known via "ospf 1", distance 122, metric 1
  Tag 15, type extern 2, forward metric 1
  Redistributing via rip
  Advertised by rip metric 3
  Last update from 192.168.10.6 on Ethernet0/1, 00:08:34 ago
  Routing Descriptor Blocks:
  * 192.168.10.6, from 3.3.3.3, 00:08:34 ago, via Ethernet0/1
      Route metric is 1, traffic share count is 1
      Route tag 15
```

Resolution: Statically configure the BGP router ID to match the statically configured OSPF router ID.

OSPF has a router ID configuration command. BGP has the same command:

```
R3#sh run | section router
router ospf 1
 router-id 3.3.3.3
 log-adjacency-changes
 redistribute rip subnets
 redistribute bgp 356 subnets
 network 192.168.10.0 0.0.0.255 area 10
 distance ospf external 122

router bgp 356
 synchronization
 bgp router-id 3.3.3.3
 bgp log-neighbor-changes
 neighbor 10.10.10.1 remote-as 15
 neighbor 192.168.10.104 remote-as 356
 neighbor 192.168.10.104 update-source Loopback103
```

You will see the neighbor relationships reform. R4 will display the BGP learned routes as synchronized, and will advertise them to R2.

```
R4#sh ip bgp 172.16.1.0
BGP routing table entry for 172.16.1.0/24, version 31
Paths: (1 available, best #1, table Default-IP-Routing-Table, RIB-failure(17))
  Advertised to update-groups:
     2
  15
    10.10.10.1 from 192.168.10.103 (3.3.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal, synchronized, best
```

It is also worth noting that these routes appear in the BGP table with an "r" for "RIB failure":

```
R4#show ip bgp
BGP table version is 33, local router ID is 192.168.10.104
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
r>i172.16.0.0/24    10.10.10.1               0    100      0 15 i
r>i172.16.1.0/24    10.10.10.1               0    100      0 15 i
r>i172.16.2.0/24    10.10.10.1               0    100      0 15 i
r>i172.16.3.0/24    10.10.10.1               0    100      0 15 i
```

The "r" code is displayed because the routes are not installed in the main routing table; they are IBGP-learned routes with an administrative distance of 200. Synchronization is enabled and these same routes are redistributed into OSPF, so the OSPF route source will be preferred and the routes will be placed in the local routing table as OSPF routes:

```
R4#show ip route | i E2
       E1 - OSPF external type 1, E2 - OSPF external type 2
O E2    172.16.0.0/24 [122/1] via 192.168.10.11, 1d00h, Ethernet0/0
O E2    172.16.0.0/16 [122/20] via 192.168.10.11, 1d00h, Ethernet0/0
O E2    172.16.1.0/24 [122/1] via 192.168.10.11, 1d00h, Ethernet0/0
O E2    172.16.2.0/24 [122/1] via 192.168.10.11, 1d00h, Ethernet0/0
O E2    172.16.3.0/24 [122/1] via 192.168.10.11, 1d00h, Ethernet0/
```


Therefore, do not be alarmed if you see a BGP update marked with the "RIB failure" code. It usually means that there is another route source for the same route with a lower administrative distance. This situation is clearly displayed with the following Cisco IOS **show** command:

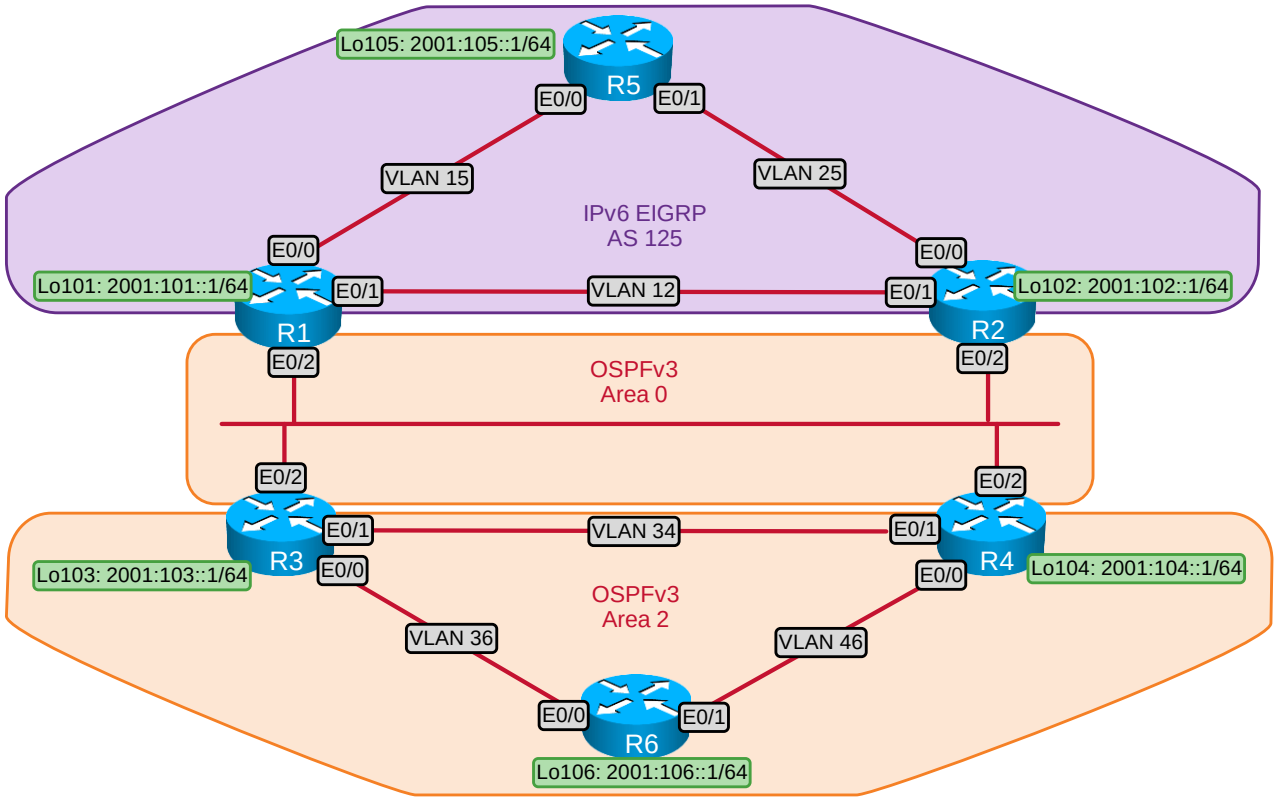
```
R4#show ip bgp rib-failure

Network            Next Hop                 RIB-failure   RIB-NH Matches
172.16.0.0/24      10.10.10.1         Higher admin distance             n/a
172.16.1.0/24      10.10.10.1         Higher admin distance             n/a
172.16.2.0/24      10.10.10.1         Higher admin distance             n/a
172.16.3.0/24      10.10.10.1         Higher admin distance             n/a
```

Note To obtain a comprehensive view of the configuration tasks in this section, access the Mentor Guide engine. You can enter more than 1000 Cisco IOS Software commands into the engine, as well as a collection of proprietary commands such as **show all**.

7. IPv6 Troubleshooting Section

Begin by reviewing the IPv6 diagram for this scenario:

IPv6 BGP Diagram



7.1. Symptom: OSPFv3 neighbor relationships are formed incorrectly in OSPFv3 Area 0.

Analysis and testing:

The OPSFv3 for IPv6 neighbor relationships are formed using the DR/BDR elections, but they should not do so:

```
R1#show ipv6 ospf neighbor

            OSPFv3 Router with ID (192.168.10.101) (Process ID 1)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
192.168.10.102    1   2WAY/DROTHER    00:00:35    5               Ethernet0/2
192.168.10.103    1   FULL/BDR        00:00:34    5               Ethernet0/2
192.168.10.104    1   FULL/DR         00:00:31    5               Ethernet0/2
R1#
```

Examine the **show ipv6 ospf interface** output. Begin with R1. The **show ip ospf interface** command is useful for troubleshooting an OSPFv2 problem for IPv4, and the **show ipv6 ospf interface** command is a useful initial **show** command for troubleshooting an OSPFv3 for IPv6 problem.


```
R1#show ipv6 ospf interface e0/2
Ethernet0/2 is up, line protocol is up
  Link Local Address FE81::1, Interface ID 5
  Area 0, Process ID 1, Instance ID 0, Router ID 192.168.10.101
  Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 192.168.10.104, local address FE80::4
  Backup Designated router (ID) 192.168.10.103, local address FE80::3
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:04
  Graceful restart helper support enabled
  Index 1/1/1, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 3, Adjacent neighbor count is 2
    Adjacent with neighbor 192.168.10.103  (Backup Designated Router)
    Adjacent with neighbor 192.168.10.104  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

Note that the Ethernet0/2 interface is configured with the default OSPFv3 broadcast network type. According to the lab restrictions, you cannot elect DR/BDR on OSPFv3 Area 0. Also, the OSPFv3 packets must be unicast in the OSPFv3 Area 0.

Likely cause: The OSPFv3 default network type is configured on the Ethernet0/2 interface of R1, R2, R3, and R4.

All four routers have the correct IPv6 OSPFv3 neighbor statements, but they have the wrong OPSFv3 network type. They are all configured for the OSPFv3 broadcast network type on the E0/2 interface. This network type does not unicast its hellos, it multicasts them.

Therefore, this must be changed to the OSPFv3 point-to-multipoint nonbroadcast network type.

Resolution: Configure the OSPFv3 point-to-multipoint nonbroadcast network on the E0/2 interfaces.

The following display shows the final configuration of the correct OSPF configurations on the E0/2 interfaces:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int e0/2
R1(config-if)#ipv6 ospf netw point-to-multipoint non-broadcast
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)#ipv6 ospf netw point-to-multipoint non-broadcast
R2(config-if)#end
R2#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#int e0/2
R3(config-if)#ipv6 ospf netw point-to-multipoint non-broadcast
R3(config-if)#end
R3

R4#conf t
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#int e0/2
R4(config-if)#ipv6 ospf netw point-to-multipoint non-broadcast
R4(config-if)#end
R4#
```

When this configuration change has been made, all OSPFv3 routers can form neighbor relationships. Note that it may take a few minutes for the neighbor relationships to form.

```
R1#show ipv6 ospf neighbor

            OSPFv3 Router with ID (192.168.10.101) (Process ID 1)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
N/A               0   DOWN/  -           -        0               Ethernet0/2
192.168.10.103    0   FULL/  -        00:01:51    5               Ethernet0/2
192.168.10.104    0   FULL/  -        00:01:51    5               Ethernet0/2
R1#
```

Note that the full OSPFv3 adjacency is formed with R3 and R4, but not with R2.

Run the **deb ipv6 ospf hello** command on R1.

```
R1#deb ipv6 ospf hello
OSPFv3 hello events debugging is on for process 1, IPv6, Default vrf
R1#
R1#
*Jun 15 21:55:00.967: OSPFv3-1-IPv6 HELLO Et0/2: Send hello to FE80::2 area 0 from
FE81::1 interface ID 5
*Jun 15 21:55:00.967: OSPFv3-1-IPv6 HELLO Et0/2: Send hello to FE80::3 area 0 from
FE81::1 interface ID 5
*Jun 15 21:55:00.967: OSPFv3-1-IPv6 HELLO Et0/2: Send hello to FE80::4 area 0 from
FE81::1 interface ID 5
*Jun 15 21:55:00.967: OSPFv3-1-IPv6 HELLO Et0/2: Rcv hello from 192.168.10.104 area
0 from FE80::4 interface ID 5
*Jun 15 21:55:00.967: OSPFv3-1-IPv6 HELLO Et0/2: Rcv hello from 192.168.10.103 area
0 from FE80::3 interface ID 5
R1#u all
```

Note that R1 is sending hello packets from the wrong IPv6 link-local address, FE81::1. The address should be FE80::1.

Verify the IPv6 link-local address configuration on two problematic routers, R1 and R2:

```
R1#show ipv6 int e0/2 | inc link
  IPv6 is enabled, link-local address is FE81::1
  No Virtual link-local address(es):
R1#
R2#show ipv6 int e0/2 | inc link
  IPv6 is enabled, link-local address is FE81::2
  No Virtual link-local address(es):
R2#
```

Fix the link-local IPv6 configuration on the Ethernet0/2 interfaces of R1 and R2:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int e0/2
```



```
R1(config-if)#ipv6 address fe80::1 link-local
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)#ipv6 address fe80::2 link-local
R2(config-if)#end
R2#
```

When you have addressed all of the OSPFv3 issues, you can test IPv6 unicast connectivity 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.

```
tclsh
foreach address {
2001:101::1
2001:102::1
2001:103::1
2001:104::1
2001:105::1
2001:106::1
} {ping $address}
```

You find that R5 cannot ping the IPv6 networks that are originated in the OSPFv3 domain:

```
R5#tclsh
R5(tcl)#foreach address {
+>(tcl)#2001:101::1
+>(tcl)#2001:102::1
+>(tcl)#2001:103::1
+>(tcl)#2001:104::1
+>(tcl)#2001:105::1
+>(tcl)#2001:106::1
+>(tcl)#} {ping $address}
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:101::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:102::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:103::1, timeout is 2 seconds:

% No valid route for destination
Success rate is 0 percent (0/1)
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:104::1, timeout is 2 seconds:

% No valid route for destination
Success rate is 0 percent (0/1)
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:105::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/5 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:106::1, timeout is 2 seconds:

% No valid route for destination
Success rate is 0 percent (0/1)
```

```
R5(tcl)#exit
R5#
```

Verify the IPv6 route redistribution configuration on R1 and R2:

```
R1#show running-config | sec ipv6 router eigrp
ipv6 router eigrp 125
 redistribute ospf 1 include-connected
R1#
R2#show running-config | sec ipv6 router eigrp
ipv6 router eigrp 125
 redistribute ospf 1 include-connected
R2#
```

Note that the IPv6 EIGRP metrics are not configured. Fix the IPv6 EIGRP configuration on R1 and R2:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ipv6 router eigrp 125
R1(config-rtr)#default-metric 1 1 1 1 1
R1(config-rtr)#end
R1#

R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ipv6 router eigrp 125
R2(config-rtr)#default-metric 1 1 1 1 1
R2(config-rtr)#end
R2#
```

Verify the IPv6 connectivity from R5 again:

```
R5#tclsh
R5(tcl)#foreach address {
+>(tcl)#2001:101::1
+>(tcl)#2001:102::1
+>(tcl)#2001:103::1
+>(tcl)#2001:104::1
+>(tcl)#2001:105::1
+>(tcl)#2001:106::1
+>(tcl)#} {ping $address}
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:101::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:102::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:103::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:104::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
```



```
Sending 5, 100-byte ICMP Echos to 2001:105::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:106::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R5(tcl)#exit
R5#
```

Note To obtain a comprehensive view of the configuration tasks in this section, access the Mentor Guide engine. You can enter more than 1000 Cisco IOS Software commands into the engine, as well as a collection of proprietary commands such as **show all**.

8. IP QoS Troubleshooting Section

8.1. Symptom: Router R1 is not receiving packets with a DSCP value of 45.

Analysis and testing:

According to the scenario requirements, packets originating from the R2 Loopback 102 interface should be marked with IP precedence flash or 3 on the Ethernet0/0 interface on R2 and marked with the DSCP value of 45 on R5.

To test whether router R1 is receiving any packets with a DSCP value of 45, you could create and apply a simple policy or access list, like the following:

```
R1(config)#access-list 145 permit icmp any any dscp 45
R1(config)#access-list 145 permit ip any any
R1(config)#int E0/0
R1(config-if)#ip access-group 145 in
```

Note that this access list is already applied to the E0/0 interface on R1 during the lab initialization.

Ping R1 from R2 according to the lab requirements:

```
R2#ping 192.168.10.101 source Lo102
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.101, timeout is 2 seconds:
Packet sent with a source address of 192.168.10.102
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R2#
```

Observe the ACL counters on R1:

```
R1#show access-list 145
Extended IP access list 145
    10 permit icmp any any dscp 45
    20 permit ip any any (516 matches)
R1#
```

Notice that a number of packets arrived, but none were marked with the DSCP value of 45. Does R2 mark the outgoing packets with the IP precedence flash?

```
R2#show policy-map interface
 Ethernet0/0
```

```
  Service-policy output: PREC

    Class-map: PREC (match-all)
      5 packets, 570 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: access-group 150
      QoS Set
        precedence 3
          Packets marked 5

    Class-map: class-default (match-any)
      962 packets, 98954 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
R2#
```

Note that R2 marks the outgoing ICMP packets with the IP precedence 3.

Likely cause: The QoS policy on R5 is misconfigured.

Check the QoS configuration of R5:

```
R5#show policy-map interface
 Ethernet0/1

  Service-policy input: PREC-DSCP

    Class-map: PREC (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: protocol icmp
      Match: ip precedence 3
      Match:  precedence 3
      Match: discard-class 3
      QoS Set
        dscp 45
          Packets marked 0

    Class-map: class-default (match-any)
      2934 packets, 320265 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
R5#
```

Notice that no packets are marked with the DSCP value of 45. If you look at the matching criteria, you will see that the match rule is too restrictive. It is configured to match the ICMP protocol *and* the IP precedence 3 *and* the discard class 3. You should not match the discard class in this lab.

Resolution: Remove the matching configuration for the discard class.

Fix the QoS configuration on R5:

```
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#class-map match-all PREC
R5(config-cmap)#no  match discard-class 3
R5(config-cmap)#end
R5
```

Ping R1 from R2 again:

```
R2#ping 192.168.10.101 source Lo102
```



```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.101, timeout is 2 seconds:
Packet sent with a source address of 192.168.10.102
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R2#
```

Observe the ACL counters on R1:

```
R1#show access-list 145
Extended IP access list 145
    10 permit icmp any any dscp 45 (5 matches)
    20 permit ip any any (673 matches)
R1#
```

Notice that packets arrived and were marked with the DSCP value of 45.

Note To obtain a comprehensive view of the configuration tasks in this section, access the Mentor Guide engine. You can enter more than 1000 Cisco IOS Software commands into the engine, as well as a collection of proprietary commands such as **show all**.