360RSW05-WKB-TA10

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 10 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 10 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 that are displayed in the scenario diagram belong to network 172.16.0.0/16.
- *Points will be deducted from multiple sections for failing to assign correct IPv4 addresses.*
- Do not use any static routes.
- Advertise loopback interfaces with their original masks.
- Network 0.0.0.0/0 should not appear in any routing table (**show ip route**).
- Do not use the **ip default-gateway** or **ip default-network** commands.
- Do not introduce any new IP addresses.
- Unless explicitly specified 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 interior gateway protocol (IGP)-only routers.

- Use conventional routing algorithms only, unless the instructions specify 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 displayed in the scenario "IPv4 IGP" diagram belong to network 172.16.0.0/16.

All IP addresses in this exam belong to the 172.16.0.0/16 address space with the exception of prefixes that are explicitly specified as being part of a different IP space.

Do not use any static routes.

Static routes can be used to solve a range of reachability problems. However, you cannot use them in this lab. You must rely on skillful configuration of all your unicast routing protocols.

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 need to provide a solution such as changing the OSPF network type or summarizations. Remember that this rule applies to both IPv4 and IPv6 networks.

Network 0.0.0.0/0 should not appear in any routing table (show ip route**).**

A 0.0.0.0/0 entry can be used to solve a range of reachability problems. In particular, a 0.0.0.0/0 entry can be used to set up the gateway of last resort. However, in this exercise, you cannot use any 0.0.0.0/0 entries. An alternative to using the 0.0.0.0/0 route to solve the reachability problem is route summarization.

Do not use the ip default-gateway **or** ip default-network **commands.**

These commands can be used to solve reachability issues by setting the gateway of last resort. They generate a 0.0.0.0/0 route in the Routing Information Protocol (RIP) environment. You cannot use them in this scenario.

All IP addresses involved in this scenario must be reachable.

This is a key goal to observe. It requires that all your IGPs and all your 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 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 must 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.

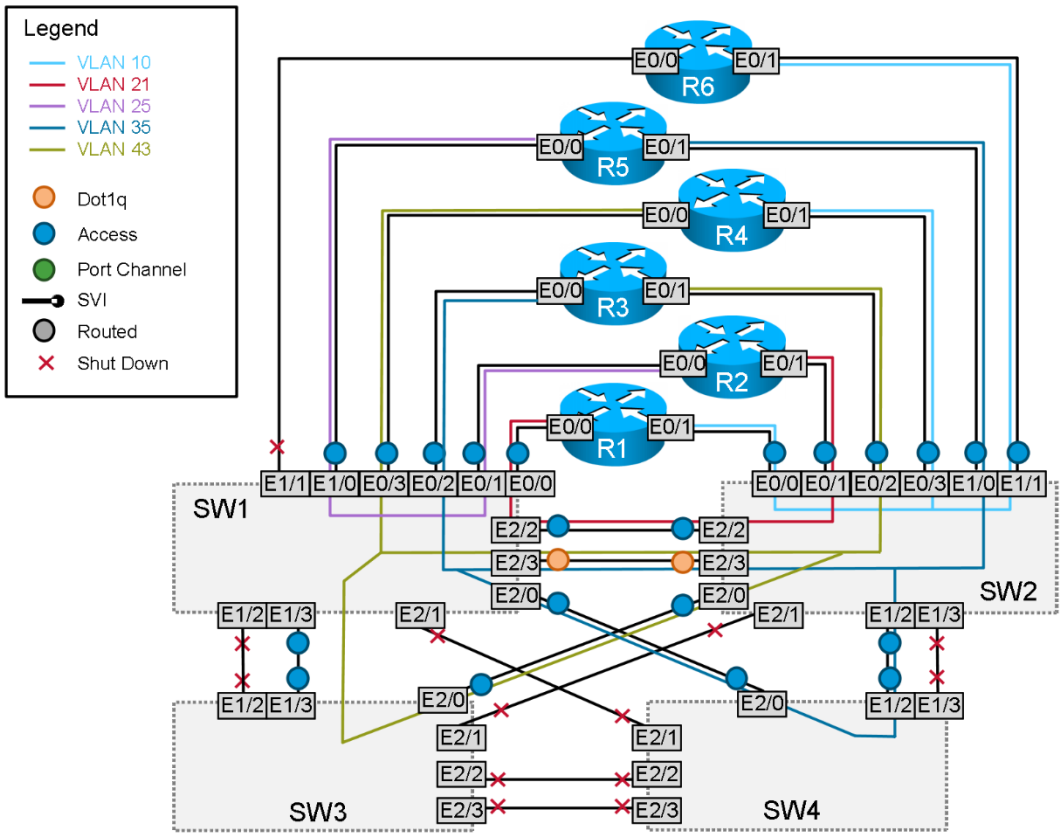
This restriction prevents you from solving any problems by configuring policy routing. At the heart of this restriction is the interpretation of "conventional routing algorithms." Although this phrase can be interpreted in different ways, this interpretation is applied in this workbook:

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

Begin with a review of a Layer 2 diagram for this specific lab:

VLAN Propagation



Use this diagram and the Layer 3 diagram from the scenario as guides to test Ethernet same-subnet reachability and correspondence with scenario requirements.

1.1. Symptom: There is a connectivity issue between R3 and R4 on VLAN 43.

Analysis and Testing:

Using ping between R3 and R4 is not successful:

```
R3#ping 172.16.43.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.43.4, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R3#
```

There are two possible forwarding paths between R3 and R4 on VLAN 43:

1) Via the trunk link between the E2/3 interfaces of SW1 and SW2.
2) Via SW3 if forwarding via the trunk link between the E2/3 interfaces of SW1 and SW2 is not available.

The unsuccessful ping tells you that neither path works.

The following console messages periodically appear on SW1 and SW2:

```
SW1#
*Aug 27 17:07:29.096: %CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered
on Ethernet1/3 (43), with SW3 Ethernet1/3 (1).
SW1#

SW2#
*Aug 27 17:09:02.479: %CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered
on Ethernet2/0 (43), with SW3 Ethernet2/0 (1).
SW2#
```

This type of error message indicates some form of mismatched configuration between the two ends of a link.

Likely Cause: There is a configuration mismatch between the E1/3 and E2/0 interfaces of SW3, the E1/3 interface of SW1, and the E2/0 interface of SW2.

The VLAN mismatch message shows that native VLAN 1 is configured on the E1/3 and E2/0 interfaces of SW3.

Begin by verifying the port configurations on SW3:

```
SW3#sh run inte E1/3
Building configuration...

Current configuration : 103 bytes
!
interface Ethernet1/3
 switchport trunk encapsulation dot1q
 switchport mode trunk
 duplex auto
end

SW3#

SW3#sh run inte E2/0
Building configuration...

Current configuration : 103 bytes
!
interface Ethernet2/0
 switchport trunk encapsulation dot1q
 switchport mode trunk
 duplex auto
end

SW3#
```

Note that the E1/3 and E2/0 ports on SW3 are configured for trunking, while the remote ports are configured for access ports:

```
SW1#show run int e1/3
Building configuration...

Current configuration : 93 bytes
!
interface Ethernet1/3
 switchport access vlan 43
 switchport mode access
 duplex auto
end

SW1#
SW2#show run int e2/0
```

```
Building configuration...

Current configuration : 93 bytes
!
interface Ethernet2/0
 switchport access vlan 43
 switchport mode access
 duplex auto
end

SW2#
```

Resolution: Configure SW3 for access ports as well.

Since the requirements of this lab specify that these ports should be configured as access ports, you must change the configuration to meet the requirements of the lab. All configuration changes will be on SW3:

```
SW3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW3(config)#int e1/3
SW3(config-if)#switchport mode access
SW3(config-if)#switchport access vlan 43
SW3(config-if)#int e2/0
SW3(config-if)#switchport access vlan 43
SW3(config-if)#switchport mode access
SW3(config-if)#end
SW3#
```

You can now see that the native VLAN mismatch error message is no longer being generated.

Try to ping R4 from R3 again:

```
R3#ping 172.16.43.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.43.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R3#
```

Note that forwarding is now established via SW3. But the forwarding path via SW3 is only a backup path. Why couldn't R3 and R4 communicate via the trunk between SW1 and SW2?

1.2. The trunk between SW1 and SW2 does not forward the VLAN 43 packets.

Analysis and Testing:

Begin by analyzing the trunk interfaces on SW1 and SW2:

```
SW1#show int trunk

Port        Mode             Encapsulation  Status        Native vlan
Et2/3       on               802.1q         trunking      1

Port        Vlans allowed on trunk
Et2/3       35

Port        Vlans allowed and active in management domain
Et2/3       35

Port        Vlans in spanning tree forwarding state and not pruned
Et2/3       35
SW1#
SW2#show int trunk
```



```
Port          Mode             Encapsulation  Status        Native vlan
Et2/3         on               802.1q         trunking      1

Port          Vlans allowed on trunk
Et2/3         35

Port          Vlans allowed and active in management domain
Et2/3         35

Port          Vlans in spanning tree forwarding state and not pruned
Et2/3         35
SW2#
```

Note that the trunk is operational.

Likely Cause: VLAN 43 is not allowed on the trunk between SW1 and SW2.

Begin by reviewing the interface configurations for both SW1 and SW2:

```
SW1#show run int e2/3
Building configuration...

Current configuration : 137 bytes
!
interface Ethernet2/3
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 35
 switchport mode trunk
 duplex auto
end

SW1#


SW2#show run int e2/3
Building configuration...

Current configuration : 137 bytes
!
interface Ethernet2/3
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 35
 switchport mode trunk
 duplex auto
end

SW2#
```

Note that only VLAN 35 is allowed on the trunk.

Resolution: Allow VLAN 43 on the trunk between SW1 and SW2.

```
SW1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW1(config)#int e2/3
SW1(config-if)# switchport trunk allowed vlan add 43
SW1(config-if)#end
SW1#
```

```
SW2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)#int e2/3
SW2(config-if)# switchport trunk allowed vlan add 43
SW2(config-if)#end
SW2#
```

Verify the trunk configuration on either SW1 or SW2 again:

```
SW1#show int trunk

Port        Mode             Encapsulation  Status        Native vlan
Et2/3       on               802.1q         trunking      1

Port        Vlans allowed on trunk
Et2/3       35,43

Port        Vlans allowed and active in management domain
Et2/3       35,43

Port        Vlans in spanning tree forwarding state and not pruned
Et2/3       35.43
SW1#
```

Verify the VLAN 43 forwarding path. Here is an example from SW1:

```
SW1#show spanning-tree vlan 43

VLAN0043
  Spanning tree enabled protocol ieee
  Root ID    Priority    43
             Address     aabb.cc00.0700
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    43     (priority 0 sys-id-ext 43)
             Address     aabb.cc00.0700
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  15  sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et0/3               Desg FWD 100       128.4    Shr
Et1/3               Desg FWD 100       128.8    Shr
Et2/3               Desg FWD 100       128.12   Shr


SW1#
```

Note that the E2/3 interface is forwarding VLAN 43 data.

Here is an example from SW3:

```
SW3#show spanning-tree vlan 43

VLAN0043
  Spanning tree enabled protocol ieee
  Root ID    Priority    43
             Address     aabb.cc00.0700
             Cost        100
             Port        8 (Ethernet1/3)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32811  (priority 32768 sys-id-ext 43)
             Address     aabb.cc00.0900
```



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

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Et1/3               Root FWD 100       128.8    Shr
Et2/0               Altn BLK 100       128.9    Shr


SW3#
```

Note that the forwarding path via SW3 is currently blocked.

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: R1 and R2 are not OSPF neighbors.

Analysis and Testing:

Verify the OSPF neighbor relationship between R1 and R2:

```
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
172.16.106.1      0   FULL/  -        00:01:51    172.16.10.6     Ethernet0/1
R1#
```

Note that R1 and R2 do not form the OSPF neighbor relationship.

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 OSPF is not configured on an interface, then all other OSPF operations on that interface will not work. Therefore, begin your troubleshooting process with this command:

```
R1#show ip ospf interface e0/0
%OSPF: OSPF not enabled on Ethernet0/0
```

This is a fundamental OSPF troubleshooting issue.

Likely Cause: OSPF has not been properly configured for the address associated with the E0/0 interface on R1.

Cisco IOS Software provides two options for enabling OSPF on a given interface:

1. By configuring the **network** command under the OSPF router configuration mode.
2. By configuring **ip ospf** *PID* **area** *area-number* on the interface.

Start by reviewing the OSPF router configuration section of router R1 to see if you can detect any issues:

```
R1#show running-config | section ^router ospf
router ospf 1
 router-id 172.16.101.1
 log-adjacency-changes
 area 10 virtual-link 172.16.104.1
 redistribute connected subnets
 network 172.16.10.0 0.0.0.255 area 10
 network 172.16.21.0 0.0.0.0 area 0
```

You see that there is an explicit network statement for the IP address configured on the E0/0 interface of R1, but OSPF is still not enabled on this interface. However, when you look closely at the OSPF **network** configuration command, you see the error. The prefix that is specified in the statement is 172.16.21.0, but the wildcard mask is configured as 0.0.0.0, which means that the prefix must be an exact match, or complete 32-bit address.

Resolution: Correct the misconfigured OSPF network ***command under the OSPF router configuration on R1.***

This misconfiguration can be corrected in one of the following two ways:

```
R1(config)#router ospf 1
R1(config-router)#no network 172.16.21.0 0.0.0.0 area 0
R1(config-router)#network 172.16.21.1 0.0.0.0 area 0
```

or

```
R1(config)#router ospf 1
R1(config-router)#network 172.16.21.0 0.0.0.255 area 0
```

As soon as either of these commands is entered, the **show ip ospf interface** command displays as it should when the basic OSPF enabling commands are correctly entered:

```
R1#show ip ospf interface e0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 172.16.21.1/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 172.16.101.1, Network Type BROADCAST, Cost: 2
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           2         no          no            Base
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 172.16.102.1, Interface address 172.16.21.2
  Backup Designated router (ID) 172.16.102.1, Interface address 172.16.21.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:06
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 8
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 172.16.102.1  (Designated Router)
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
    Youngest key id is 1
R1#
```

This information should be seen when OSPF is properly enabled on an interface. This troubleshooting ticket is successfully resolved.

2.2. Symptom: R2 does not possess any OSPF routes.

Analysis and Testing:

Even though R2 is an OSPF neighbor of R1, R2 possesses no OSPF routes:

```
R2#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 14 subnets, 2 masks
D EX     172.16.10.0/24 [170/2867200] via 172.16.25.5, 00:14:15, Ethernet0/0
C        172.16.21.0/24 is directly connected, Ethernet0/1
L        172.16.21.2/32 is directly connected, Ethernet0/1
C        172.16.23.0/24 is directly connected, Serial1/0
L        172.16.23.2/32 is directly connected, Serial1/0
C        172.16.25.0/24 is directly connected, Ethernet0/0
L        172.16.25.2/32 is directly connected, Ethernet0/0
D        172.16.35.0/24 [90/537600] via 172.16.25.5, 01:01:58, Ethernet0/0
D EX     172.16.43.0/24 [170/2867200] via 172.16.25.5, 00:14:15, Ethernet0/0
C        172.16.102.0/24 is directly connected, Loopback102
L        172.16.102.1/32 is directly connected, Loopback102
D EX     172.16.103.0/24 [170/2867200] via 172.16.25.5, 00:14:15, Ethernet0/0
D EX     172.16.104.0/24 [170/2867200] via 172.16.25.5, 00:11:12, Ethernet0/0
D        172.16.105.0/24 [90/409600] via 172.16.25.5, 00:14:14, Ethernet0/0
R2#
```

As can be seen, R2 only possesses Enhanced Interior Gateway Routing Protocol (EIGRP) and connected routes. R2 should also possess a set of OSPF routes such as 172.16.10.0/24 and 172.16.106.0/24. Clearly, these routes are not in the R2 routing table.

Also, you may see the following logging message on R2:

```
R2#
*Aug 27 16:30:29.985: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.101.1 on Ethernet0/1
from LOADING to FULL, Loading Done
R2#
*Aug 27 16:30:33.278: %OSPF-4-NET_TYPE_MISMATCH: Received Hello from 172.16.101.1
on Ethernet0/1 indicating a  potential
            network type mismatch
R2#
```

To maintain a consistent troubleshooting approach, begin by entering the **show ip ospf interface brief** command on both R1 and R2:

```
R1#show ip ospf interface e0/0
Ethernet0/0 is up, line protocol is up
  Internet Address 172.16.21.1/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 172.16.101.1, Network Type BROADCAST, Cost: 2
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           2         no          no            Base
  Transmit Delay is 1 sec, State DROTHER, Priority 1
  Designated Router (ID) 172.16.102.1, Interface address 172.16.21.2
  Backup Designated router (ID) 172.16.102.1, Interface address 172.16.21.2
  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 8
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 172.16.102.1  (Designated Router)
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
    Youngest key id is 1
R1#

R2#show ip ospf interface e0/1
Ethernet0/1 is up, line protocol is up
  Internet Address 172.16.21.2/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 172.16.102.1, Network Type POINT_TO_POINT, Cost: 10
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           10        no          no            Base
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 9
  Last flood scan time is 0 msec, maximum is 1 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 172.16.101.1
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
    Youngest key id is 1
R2#
```

R1 and R2 see each other as fully adjacent OSPF routers. All seems to be fully operational; however, there is one significant anomaly when comparing these two displays. R1 lists the state of R2 as "Designated Router" and R2 lists the state of R1 as adjacent with no reference to a designated router (DR), backup designated router (BDR), or a router that is neither a DR nor a BDR (DROTHER). This presents a problem. If one neighbor has a designation of BDR, all other neighbors on the same segment should have a designation of either DR or DROTHER.

Likely Cause: There is a mismatch of OSPF network types between R1 and R2 on VLAN 21.

To verify this, reenter the **show ip ospf interface** command on both R1 and R2:

```
R1#show ip ospf interface e0/0 | inc Network Type|Timer
  Process ID 1, Router ID 172.16.101.1, Network Type BROADCAST, Cost: 2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
R1#

R2#show ip ospf interface e0/1 | inc Network Type|Timer
  Process ID 1, Router ID 172.16.102.1, Network Type POINT_TO_POINT, Cost: 10
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
R2#
```

As can be seen, the two routers do, in fact, possess different network types. This is the source of the problem. You might ask why R1 and R2 were able to form OSPF neighbor relationships when their OSPF network types did not match. The answer to this question lies in the fact that both the OSPF broadcast and point-to-point network types possess the same hello and dead timers. Parameters such as hello and dead timers need to match to form an OSPF neighbor relationship and not OSPF network types.

Resolution: Set the OSPF network types to match between R1 and R2.

While either the OSPF broadcast network type or the OSPF point-to-point network type could be used in this scenario, the OSPF point-to-point network type will be used. A restriction of using the OSPF point-to-point network type is that only two OSPF peers can reside on the link where this OSPF network type is used. This criterion is fulfilled in this scenario. Since R2 is already configured with the OSPF point-to-point network type, and the restrictions in the lab state that the initially configured OSPF network types on router R2 are not to be changed, only R1 needs to be configured with the following command:

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int e0/0
R1(config-if)#ip ospf network point-to-point
R1(config-if)#end
R1#
```

Once this command is entered, R2 now has OSPF routes:

```
R2#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 18 subnets, 2 masks
O IA     172.16.10.1/32 [110/10] via 172.16.21.1, 00:00:31, Ethernet0/1
O IA     172.16.10.6/32 [110/20] via 172.16.21.1, 00:00:31, Ethernet0/1
O E2     172.16.101.0/24 [171/20] via 172.16.21.1, 00:00:31, Ethernet0/1
O IA     172.16.106.1/32 [110/21] via 172.16.21.1, 00:00:31, Ethernet0/1
…
R2#
```

To learn more about troubleshooting OSPF routing issues related to OSPF network type mismatches, please download the Troubleshooting OSPF Routing videos on demand (VoDs) from the Cisco 360 Troubleshooting Lesson Module.

2.3. Symptom: Subnet 172.16.104.0/24 is not reachable.

Analysis and Testing:

OSPF virtual links are configured between R1 and R4 via R6 with Message Digest 5 (MD5) authentication, but the virtual link on R4 is not operational. Currently, the **show ip ospf virtual-links** display lists the following on router R1:

```
R4#show ip ospf virtual-links
Virtual Link OSPF_VL0 to router 172.16.106.1 is up
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 10, via interface Ethernet0/1
 Topology-MTID    Cost    Disabled     Shutdown      Topology Name
        0           10        no          no            Base
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:01
R4#
```

Even though the **show ip ospf virtual-links** display lists the virtual link as up, the virtual link is not actually up. When a virtual link is fully operational, it should display "Adjacency State Full."

Likely Cause: The MD5 authentication string is misconfigured.

Many times, when any form of authentication is configured with virtual links, the following general rule regarding OSPF virtual links is overlooked: Virtual links extend OSPF Area 0. Therefore, when area-based OSPF MD5 authentication is configured for Area 0, which it is in this scenario, care must be taken to make sure that the **area 0 authentication message-digest** command is entered on the end of the virtual link that does not maintain a direct connection to Area 0. You can check to see if this command is entered on R4:

```
R4#show run | section ^router ospf
router ospf 1
 router-id 172.16.104.1
 area 10 virtual-link 172.16.106.1 message-digest-key 1 md5 CISCO
 redistribute rip subnets route-map test
 network 172.16.10.0 0.0.0.255 area 10
 network 172.16.104.0 0.0.0.255 area 104
 distribute-list 10 in
R4#
```

You can see that it has not been configured. This is the source of the problem causing the virtual link not to activate.

Resolution: Configure the area 0 authentication message-digest ***command on R4.***

```
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#router ospf 1
R4(config-router)#area 0 authentication message-digest
R4(config-router)#end
R4#
*Aug  27  17:01:33.692:  %SYS-5-CONFIG_I:  Configured  from  console  by  console
(cierswbv5-te-lab10-sc, SJ)
R4#
*Aug 27 17:01:37.222: %OSPF-5-ADJCHG: Process 1, Nbr 172.16.106.1 on OSPF_VL0 from
LOADING to FULL, Loading Done
R4#
```

As can be seen, the OSPF virtual link becomes active in a few seconds. This is further verified by the following command:

```
R4#show ip ospf virtual-links
Virtual Link OSPF_VL0 to router 172.16.106.1 is up
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 10, via interface Ethernet0/1
 Topology-MTID    Cost    Disabled     Shutdown      Topology Name
        0          10        no          no            Base
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:06
    Adjacency State FULL (Hello suppressed)
    Index 1/2, 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
  Message digest authentication enabled
    Youngest key id is 1
R4#
```

This is the desired output for verifying the status of an OSPF virtual link. In conclusion, when configuring authentication over an OSPF virtual link, remember that a virtual link extends Area 0. Therefore, when configuring area-based authentication, and in particular area-based

authentication for Area 0, remember to configure the OSPF router configuration command **area 0 authentication** on the far end of the virtual link.

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: EIGRP routes are not being exchanged between R3 and R5.

Analysis and Testing:

You verified the EIGRP table on R3 and noticed that R3 does not learn any prefixes from R5:

```
R3#sh ip route eigrp | begin 172.16
      172.16.0.0/16 is variably subnetted, 19 subnets, 2 masks
D EX     172.16.10.1/32 [170/26137600] via 172.16.23.2, 00:22:44, Serial1/0
D EX     172.16.10.4/32 [170/26137600] via 172.16.23.2, 00:22:44, Serial1/0
D EX     172.16.10.6/32 [170/26137600] via 172.16.23.2, 00:22:44, Serial1/0
D EX     172.16.21.0/24 [170/26137600] via 172.16.23.2, 00:22:44, Serial1/0
D        172.16.25.0/24 [90/26137600] via 172.16.23.2, 00:22:44, Serial1/0
D EX     172.16.101.0/24 [170/26137600] via 172.16.23.2, 00:22:44, Serial1/0
D        172.16.102.0/24 [90/26240000] via 172.16.23.2, 00:22:44, Serial1/0
D EX     172.16.104.0/24 [170/26137600] via 172.16.23.2, 00:22:44, Serial1/0
D        172.16.105.0/24 [90/26265600] via 172.16.23.2, 00:22:44, Serial1/0
D EX     172.16.106.1/32 [170/26137600] via 172.16.23.2, 00:22:44, Serial1/0
R3#
```

Note that R3 learns EIGRP prefixes only via the Serial1/0 interface.

R3 and R5 can ping each other on VLAN 35:

```
R3#ping 172.16.35.5

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.35.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

And R3 and R5 have EIGRP enabled on their respective VLAN 35 interfaces:

```
R3#show ip eigrp interfaces e0/0
EIGRP-IPv4 Interfaces for AS(100)
                              Xmit  Queue     PeerQ             Mean     Pacing  Time
Multicast    Pending
Interface               Peers  Un/Reliable  Un/Reliable  SRTT    Un/Reliable    Flow
Timer   Routes
Et0/0                    0        0/0       0/0           0       0/2           50
0
R3#

R5#show ip eigrp interfaces e0/1
EIGRP-IPv4 Interfaces for AS(100)
                              Xmit  Queue     PeerQ             Mean     Pacing  Time
Multicast    Pending
Interface               Peers  Un/Reliable  Un/Reliable  SRTT    Un/Reliable    Flow
Timer   Routes
Et0/1                    0        0/0       0/0           0       0/2           50
0
R5#
```

However, R3 and R5 do not see each other as EIGRP peers. To troubleshoot this problem, you can enable the **debug eigrp packet** utility. The **debug eigrp packet** utility is a useful learning and diagnostic tool for EIGRP. However, it creates a lot of output. Therefore, and 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** to only interface E0/0. As an example, **debug interface** will suppress all **debug eigrp packet** traffic generated on the E0/0 interface of R3. Begin by applying these debug utilities only on R3:

```
R3#debug interface e0/0
Condition 1 set
R3#debug eigrp packet
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, UNKNOWN, PROBE, ACK, STUB, SIAQUERY,
SIAREPLY)
EIGRP Packet debugging is on
R3#
*Aug 27 16:14:57.705: EIGRP: Sending HELLO on Se1/0 - paklen 20
*Aug 27 16:14:57.705:     AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ
un/rely 0/0
*Aug 27 16:14:58.016: EIGRP: Received HELLO on Se1/0 - paklen 20 nbr 172.16.23.2
*Aug 27 16:14:58.016:     AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ
un/rely 0/0 peerQ un/rely 0/0
R3#
*Aug 27 16:14:59.732: EIGRP: Received HELLO on Et0/0 - paklen 20 nbr 172.16.35.5
*Aug 27 16:14:59.732:   AS 100, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0
*Aug 27 16:14:59.732: EIGRP: Ignore unicast Hello from Ethernet0/0 172.16.35.5
R3#
R3#u all
All possible debugging has been turned off
R3#
```

As the **debug eigrp packet** output clearly displays, R5 is sending EIGRP packets to R3 using a unicast address. And as the **debug eigrp packet** output shows, R3 is ignoring these packets.

Now that **debug eigrp packet** 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 e0/0.** When you do this and the specific debug tool is the very last conditional debug utility applied, Cisco IOS Software will return with the following prompt:

```
R3#undebug interface e0/0
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
```

Simply 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 unicasting EIGRP packets to R3 and R3 is multicasting EIGRP packets to R5.

This situation is clearly exhibited in the **debug eigrp packet** display above. This can be further verified by examining the router EIGRP configuration on both R3 and R5:

```
R3#show running-config | section eigrp
router eigrp 100
 network 172.16.23.0 0.0.0.255
 network 172.16.35.0 0.0.0.255
 redistribute rip metric 1000 100 255 1 1500
 offset-list 1 in 506880 Ethernet0/0
```



```
 auto-summary
 redistribute eigrp 100 metric 2
R3#

R5

R5#show running-config | section eigrp
router eigrp 100
 network 172.16.0.0
 auto-summary
 neighbor 172.16.35.3 Ethernet0/1
```

Now that you know why R3 and R5 are not forming EIGRP neighbor relationships, you can resolve this issue.

Resolution: Configure both R3 and R5 to consistently exchange either unicast EIGRP packets or multicast EIGRP packets.

Because there are only two EIGRP speakers on VLAN 35, R3 will be configured to unicast EIGRP packets to R5:

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router eigrp 100
R3(config-router)#neighbor 172.16.35.5 e0/0
R3(config-router)#end
R3#
*April 13 23:17:44.434: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 172.16.35.5
(Ethernet0/0) is up: new adjacency
```

As can be seen by the console message, as soon as the **neighbor 172.16.35.5 e0/0** command was entered on R3, the EIGRP adjacency to R5 was formed. You can further verify that an EIGRP neighbor relationship has formed between R3 and R5:

```
R3#
*Aug 27 16:19:07.542: %SYS-5-CONFIG_I: Configured from console by console
(cierswbv5-te-lab10-sc, SJ)
R3#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   172.16.35.5             Et0/0                    13 00:00:14    8   100  0  379
0   172.16.23.2             Se1/0                    12 22:08:05   41  1458  0  313
R3#
```

As can be seen, the desired EIGRP neighbor relationship has been successfully formed.

3.2. Symptom: The 172.16.23.0/24 EIGRP internal route is not load balanced on R5.

Analysis and Testing:

From a topology perspective, R5 possesses two identical paths to the 172.16.23.0/24 subnet. Due to this fact, R5 should maintain two equal-cost load-balanced paths to the 172.16.23.0/24 subnet. However, it does not. Begin by reviewing the R5 routing table below:

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

```
Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 18 subnets, 2 masks
D EX     172.16.10.0/24 [170/2611200] via 172.16.35.3, 00:10:37, Ethernet0/1
D EX     172.16.10.1/32 [170/2841600] via 172.16.25.2, 00:00:47, Ethernet0/0
D EX     172.16.10.4/32 [170/2841600] via 172.16.25.2, 00:00:47, Ethernet0/0
D EX     172.16.10.6/32 [170/2841600] via 172.16.25.2, 00:00:47, Ethernet0/0
D EX     172.16.21.0/24 [170/2841600] via 172.16.25.2, 00:00:47, Ethernet0/0
D        172.16.23.0/24 [90/26137600] via 172.16.35.3, 00:00:47, Ethernet0/1
C        172.16.25.0/24 is directly connected, Ethernet0/0
L        172.16.25.5/32 is directly connected, Ethernet0/0
C        172.16.35.0/24 is directly connected, Ethernet0/1
L        172.16.35.5/32 is directly connected, Ethernet0/1
D EX     172.16.43.0/24 [170/2611200] via 172.16.35.3, 20:28:15, Ethernet0/1
D EX     172.16.101.0/24 [170/2841600] via 172.16.25.2, 00:00:47, Ethernet0/0
D        172.16.102.0/24 [90/640000] via 172.16.25.2, 00:00:47, Ethernet0/0
D EX     172.16.103.0/24 [170/2611200] via 172.16.35.3, 20:28:15, Ethernet0/1
D EX     172.16.104.0/24 [170/2841600] via 172.16.25.2, 00:00:47, Ethernet0/0
C        172.16.105.0/24 is directly connected, Loopback105
L        172.16.105.1/32 is directly connected, Loopback105
D EX     172.16.106.1/32 [170/2841600] via 172.16.25.2, 00:00:47, Ethernet0/0
R5#
```

Note that only one entry is in the R5 routing table for the 172.16.23.0/24 subnet. R3 is preferred as the next hop. You can now investigate the R5 EIGRP topology table:

```
R5#show ip eigrp topology 172.16.23.0 255.255.255.0
EIGRP-IPv4 Topology Entry for AS(100)/ID(172.16.105.1) for 172.16.23.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 26137600
  Descriptor Blocks:
  172.16.35.3 (Ethernet0/1), from 172.16.35.3, Send flag is 0x0
      Composite metric is (26137600/26112000), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 21000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 172.16.103.1
  172.16.25.2 (Ethernet0/0), from 172.16.25.2, Send flag is 0x0
      Composite metric is (26368000/26112000), route is Internal
      Vector metric:
        Minimum bandwidth is 100 Kbit
        Total delay is 30000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 172.16.102.1
R5#
```

This **show** command provides some useful information. First, you can see that the locally calculated composite metric on R5 is greater for the route learned from R2 than for the route learned from R3. However, both R2 and R3 calculated the metric for the 172.16.23.0/24 prefix before it was advertised to R5. Because only the locally calculated composite metric is different between the two routes learned by R5, the difference in the calculation must be due to something configured on R5 itself. Exactly what was configured differently on R5 is reflected by the two total delay values displayed above. The total delay to the route learned from R3 is 20,100 microseconds. The total delay to the route learned from R2 is 30,000 microseconds. This accounts for the difference in the composite metric between the two paths for this single 172.16.23.0/24 prefix.

Likely Cause: The interface delay parameter has been adjusted on one of the EIGRP-enabled interfaces on R5.

As mentioned above, it appears that all aspects of the metric calculation for the 172.16.23.0/24 prefix are identical, with the exception of the local calculation performed on R5 itself. Also as mentioned above, it appears that the delay value has been adjusted on R5. Examine the R5 interface configuration to see if this is the case:

```
R5#show running-config | section interface Ethernet0\/[01]
interface Ethernet0/0
 ip address 172.16.25.5 255.255.255.0
 delay 1000
interface Ethernet0/1
 ip address 172.16.35.5 255.255.255.0
R5#
```

As you can see, the E0/0 interface—the interface connected to the subnet shared with EIGRP neighbor R2—has a statically configured delay value of 10,000 microseconds. This is greater than the default delay value provisioned by Cisco IOS Software for Ethernet. The default delay value is 1000 microseconds. Recall that the EIGRP delay value is entered in tens of microseconds, so a configured value of 1000 results in a delay of 10,000 microseconds.

```
R5#show interfaces Ethernet 0/0 | i BW
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 10000 usec,
R5#
R5#show interfaces Ethernet 0/1 | i BW
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 1000 usec,
R5
```

Therefore, it is this statically configured interface delay value that is preventing equal-cost load balancing to occur on R5 for the 172.16.23.0/24 subnet.

Resolution: Set the interface delay value to 1000 on the E0/1 interface of R5.

In order to resolve this issue and to observe the restriction in the scenario of not **removing** any EIGRP-related commands from R5, you will increase the locally assigned delay parameter on the E0/1 interface of R5 to 1000. For learning purposes, before you do this, enable **debug ip routing** on R5 to see how the routing table entries have been changed due to the configuration change. Before you change the delay setting on E0/1, review the R5 routing table to see how many entries exist for the 172.16.23.0/24 prefix:

```
R5#show ip route eigrp | begin 172.16.23.0
D         172.16.23.0/24 [90/26137600] via 172.16.35.3, 00:14:01, Ethernet0/1
D EX      172.16.43.0/24 [170/2611200] via 172.16.35.3, 20:41:29, Ethernet0/1
D EX      172.16.101.0/24 [170/2841600] via 172.16.25.2, 00:14:01, Ethernet0/0
D         172.16.102.0/24 [90/640000] via 172.16.25.2, 00:14:01, Ethernet0/0
D EX      172.16.103.0/24 [170/2611200] via 172.16.35.3, 20:41:29, Ethernet0/1
D EX      172.16.104.0/24 [170/2841600] via 172.16.25.2, 00:14:01, Ethernet0/0
D EX      172.16.106.1/32 [170/2841600] via 172.16.25.2, 00:14:01, Ethernet0/0
R5#
```

As expected, there is only one prefix with a next-hop point of R3. Now you can enable **debug ip routing** and change the delay parameter to 1000 on E0/1 on R5 so that it matches the currently set delay parameter on interface E0/0:

```
R5#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R5(config)#int e0/1
R5(config-if)# delay 1000
R5(config-if)#
*Aug 27 15:31:21.074: RT: updating eigrp 172.16.10.0/24 (0x0):
```

```
    via 172.16.35.3 Et0/1

*Aug 27 15:31:21.074: RT: eigrp's 172.16.10.0/24 (via 172.16.35.3) metric changed
from distance/metric [170/2611200] to [170/2841600]
*Aug 27 15:31:21.074: RT: updating eigrp 172.16.103.0/24 (0x0):
    via 172.16.35.3 Et0/1

*Aug 27 15:31:21.074: RT: eigrp's 172.16.103.0/24 (via 172.16.35.3) metric changed
from distance/metric [170/2611200] to [170/2841600]
*Aug 27 15:31:21.074: RT: updating eigrp 172.16.43.0/24 (0x0):
    via 172.16.35.3 Et0/1

*Aug 27 15:31:21.074: RT: eigrp's 172.16.43.0/24 (via 172.16.35.3) metric changed
from distance/metric [170/2611200] to [170/2841600]
*Aug 27 15:31:21.074: RT: updating eigrp 172.16.23.0/24 (0x0):
    via 172.16.35.3 Et0/1

*Aug 27 15:31:21.074: RT: eigrp's 172.16.23.0/24 (via 172.16.35.3) metric changed
from distance/metric [90/26137600] to [90/26368000]
*Aug 27 15:31:21.074: RT: updating eigrp 172.16.23.0/24 (0x0):
    via 172.16.25.2 Et0/0

*Aug 27 15:31:21.074: RT: add 172.16.23.0/24 via 172.16.25.2, eigrp metric
[90/26368000]
R5(config-if)#end
R5#
```

Note that when the **delay** command is entered, the Cisco IOS Software route target (RT) process recalculates the EIGRP route for the 172.16.23.0/24 prefix. The result is that R5 maintains two load-balanced paths to the 172.16.23.0/24 prefix:

```
R5#show ip route eigrp | begin 172.16.23.0
D        172.16.23.0/24 [90/26368000] via 172.16.35.3, 00:03:07, Ethernet0/1
                        [90/26368000] via 172.16.25.2, 00:03:07, Ethernet0/0
D EX     172.16.43.0/24 [170/2841600] via 172.16.35.3, 00:03:07, Ethernet0/1
D EX     172.16.101.0/24 [170/2841600] via 172.16.25.2, 00:18:32, Ethernet0/0
D        172.16.102.0/24 [90/640000] via 172.16.25.2, 00:18:32, Ethernet0/0
D EX     172.16.103.0/24 [170/2841600] via 172.16.35.3, 00:03:07, Ethernet0/1
D EX     172.16.104.0/24 [170/2841600] via 172.16.25.2, 00:18:32, Ethernet0/0
D EX     172.16.106.1/32 [170/2841600] via 172.16.25.2, 00:18:32, Ethernet0/0
R5#
```

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: R4 is not receiving the 172.16.103.0/24 route.

Analysis and Testing:

R4 can learn the 172.16.103.0/24 route via either redistribution to OSPF or RIP. Since OSPF has a lower administrative distance than RIP, it should be preferred. However, for the 172.16.103.0/24 prefix, R4 is configured with the following access list, which is applied using a distribute list under the OSPF routing process on R4:

```
R4#sh run | b ^ro
router ospf 1
 . . . .
 network 172.16.10.0 0.0.0.255 area 10
```



```
 distribute-list 10 in
!
access-list 10 deny   172.16.103.0
access-list 10 permit any
```

Therefore, R4 will only learn the 172.16.103.0/24 route via RIP from R3. However, R4 is not learning this route.

```
R4#show ip route 172.16.103.0
% Subnet not in table
```

Begin troubleshooting this problem by using the **show protocols rip | section "rip"** command on both R3 and R4:

```
R3#show ip protocols | section "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 100, rip
  Neighbor(s):
     172.16.43.4
    172.16.43.4
  Default version control: send version 2, receive version 2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    172.16.0.0
  Passive Interface(s):
    Ethernet0/0
    Ethernet0/1
    Ethernet0/2
    Ethernet0/3
    Serial1/0
    Serial1/1
    Serial1/2
    Serial1/3
  Passive Interface(s):
    Loopback103
    RG-AR-IF-INPUT1
    VoIP-Null0
  Routing Information Sources:
    Gateway         Distance      Last Update
    172.16.43.4          171      00:16:33
  Distance: (default is 120)
    Address         Wild mask       Distance  List
    0.0.0.0         255.255.255.255      171  1
R3#

R4#show ip protocols | section "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 9 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: ospf 1 (internal, external 1 & 2, nssa-external 1 & 2)

  Redistributing: rip
  Neighbor(s):
    172.16.34.3
  Default version control: send version 1, receive version 1
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    172.16.0.0
  Passive Interface(s):
```

```
    Ethernet0/0
    Ethernet0/1
    Ethernet0/2
    Ethernet0/3
    Serial1/0
    Serial1/1
    Serial1/2
  Passive Interface(s):
    Serial1/3
    Loopback104
    RG-AR-IF-INPUT1
    Tunnel0
    VoIP-Null0
  Routing Information Sources:
    Gateway         Distance      Last Update
    172.16.43.3           120      00:09:30
  Distance: (default is 120)
R4#
```

Likely Cause: There is a RIP version mismatch.

From the **show ip protocols** display above, it is evident that there is a RIP version mismatch. R3 is configured for RIP version 2 and R4 is configured for RIP version 1.

Resolution: Configure R4 for RIP version 2.

```
R4(config)#router rip
R4(config-router)#version 2
```

Once both routers are configured with the same version of RIP, you can check the routing table of R4 to see if it has received the 172.16.103.0/24 prefix:

```
R4#show ip route 172.16.103.0
Routing entry for 172.16.103.0/24
  Known via "rip", distance 120, metric 1
  Redistributing via ospf 1, rip
  Advertised by ospf 1 subnets route-map test
  Last update from 172.16.43.3 on Ethernet0/0, 00:00:04 ago
  Routing Descriptor Blocks:
  * 172.16.43.3, from 172.16.43.3, 00:00:04 ago, via Ethernet0/0
      Route metric is 1, traffic share count is 1
R4#
```

R4 is now learning the 172.16.103.0/24 prefix via RIPv2.

4.2. Symptom: R3 is not receiving RIP updates.

Analysis and Testing:

While R4 is now receiving RIP routes, R3 is still not receiving RIP routes.

```
R3#show ip route rip
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

R3#
```

Once again, begin your troubleshooting of RIP with the **show ip protocols** command on R4:


```
R4#show ip protocols | section "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 10 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: ospf 1 (internal, external 1 & 2, nssa-external 1 & 2)

  Redistributing: rip
  Neighbor(s):
    172.16.34.3
  Default version control: send version 2, receive version 2
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    172.16.0.0
  Passive Interface(s):
    Ethernet0/0
    Ethernet0/1
    Ethernet0/2
    Ethernet0/3
    Serial1/0
    Serial1/1
    Serial1/2
  Passive Interface(s):
    Serial1/3
    Loopback104
    RG-AR-IF-INPUT1
    Tunnel0
    VoIP-Null0
  Routing Information Sources:
    Gateway         Distance      Last Update
    172.16.43.3          120      00:00:09
  Distance: (default is 120)
R4#
```

You can see that R4 is configured with a passive interface for E0/0, the interface used to advertise RIP routes to R3. R4 is also configured with a neighbor statement so that it will unicast routes to R3.

Likely Cause: The unicast address is incorrectly specified in the RIP neighbor statement.

Upon close inspection of the neighbor field in the **show ip protocols** display above, you will notice that the address is 172.16.34.3 and not the correct address of 172.16.43.3. With this unicast address, the RIP updates from R4 will never be advertised to R3.

Resolution: Reconfigure the RIP neighbor statement on R4 with the correct IP address for R3.

```
R4(config)#router rip
R4(config-router)#neighbor 172.16.43.3
```

Once this configuration change is made, you can verify that R3 is receiving RIP routes:

```
R3#sh ip route rip
     172.16.0.0/16 is variably subnetted, 14 subnets, 2 masks
R       172.16.21.0/24 [120/2] via 172.16.43.4, 00:00:06, Ethernet0/1
R       172.16.10.6/32 [120/2] via 172.16.43.4, 00:00:06, Ethernet0/1
R       172.16.10.0/24 [120/1] via 172.16.43.4, 00:00:06, Ethernet0/1
R       172.16.10.1/32 [120/2] via 172.16.43.4, 00:00:06, Ethernet0/1
R       172.16.104.0/24 [120/1] via 172.16.43.4, 00:00:06, Ethernet0/1
R       172.16.106.1/32 [120/2] via 172.16.43.4, 00:00:06, Ethernet0/1
R       172.16.101.0/24 [120/2] via 172.16.43.4, 00:00:06, Ethernet0/1
```

It is now receiving RIP routes and this problem is resolved.

In closing, it must be noted that RIP does not have nearly as many Cisco IOS Software **show** commands as EIGRP or OSPF. The most informative troubleshooting and verification **show** command for RIP is the **show ip protocols** command.

5. IPv4 Redistribution Troubleshooting Section

5.1. Symptom: No external routes are injected into OSPF from other dynamic routing protocols.

Analysis and Testing:

Verify the OSPF routes, for example on R1:

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

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 10 subnets, 2 masks
O        172.16.10.4/32 [110/20] via 172.16.10.6, 02:32:03, Ethernet0/1
O        172.16.10.6/32 [110/10] via 172.16.10.6, 12:05:12, Ethernet0/1
O IA     172.16.104.0/24 [110/21] via 172.16.10.6, 02:14:56, Ethernet0/1
O        172.16.106.1/32 [110/11] via 172.16.10.6, 12:05:12, Ethernet0/1
R1#
```

Note that R1 does not show any external OSPF routes.

Only two OSPF Autonomous System Boundary Routers (ASBRs) interfacing with another dynamic routing protocol exist in this lab's topology—routers R2 and R4. Begin by examining the basic syntax of the redistribution commands on these routers:

```
R2#show running-config | begin ^router
router eigrp 100
 network 172.16.23.0 0.0.0.255
 network 172.16.25.0 0.0.0.255
 network 172.16.102.0 0.0.0.255
 auto-summary
!
router ospf 1
 router-id 172.16.102.1
 . . . .
 redistribute eigrp 100 subnets route-map test
 network 172.16.21.0 0.0.0.255 area 0
!
route-map test permit 10
 set metric 4294967294

R4#show run | begin ^router
router ospf 1
 router-id 172.16.104.1
 . . . .
 redistribute rip subnets route-map test
 network 172.16.10.0 0.0.0.255 area 10
 network 172.16.104.0 0.0.0.255 area 104
 . . . .
```



```
 !
router rip
 version 2
 redistribute ospf 1 metric 2
 passive-interface Ethernet0/0
 network 172.16.0.0
 neighbor 172.16.43.3
 no auto-summary
!
route-map test permit 10
 set metric 4294967294
```

Both R2 and R4 have very simple OSPF redistribution configurations. They only characteristic that is out of the ordinary is the route map that is associated with a very large metric. The requirements of this lab stipulate that the metric to be used needs to be the maximum usable metric. The metric used in the route map is 4,294,967,294. This is exactly one less than the maximum value for a metric in a route map. The maximum value is not used because that normally indicates a metric of "infinity" and the route is poisoned.

Still, it is likely that this large metric is preventing the external routes learned from the dynamic routing protocols from being injected into OSPF. You can use some OSPF **show** commands to further investigate this issue. Start with the following **show** command:

```
R2#show ip ospf database | b External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.101.0    172.16.101.1    80          0x80000218 0x0074CD 0

R4#show ip ospf database | begin External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.101.0    172.16.101.1    127         0x80000218 0x0074CD 0
```

On both R2 and R4, only one external route has been injected into OSPF. This is the 172.16.101.0/24 prefix, which originated from R1 and was redistributed into OSPF via **redistribute connected** on R1. This is irrelevant to the current analysis. As a troubleshooting step, remove the metric value under the route map on one router. Do this on router R2. After doing this, once again examine the number of external OSPF routes in the OSPF database.

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#route-map test
R2(config-route-map)#no set metric
R2(config-route-map)#do show ip ospf database | b External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.10.0     172.16.102.1    11          0x80000001 0x008E27 0
172.16.23.0     172.16.102.1    11          0x80000001 0x00FEA9 0
172.16.25.0     172.16.102.1    11          0x80000001 0x00E8BD 0
172.16.35.0     172.16.102.1    11          0x80000001 0x007A22 0
172.16.43.0     172.16.102.1    11          0x80000001 0x002272 0
172.16.101.0    172.16.101.1    723         0x80000025 0x0060D6 0
172.16.102.0    172.16.102.1    11          0x80000001 0x0096C2 0
172.16.103.0    172.16.102.1    11          0x80000001 0x008BCC 0
172.16.105.0    172.16.102.1    11          0x80000001 0x0075E0 0
R2(config-route-map)#
```

As you can see, when the metric statement is removed, the EIGRP routes are injected into OSPF on R2. Therefore, it can be deduced that the metric setting has something to do with preventing

the EIGRP routes from being injected into OSPF. Reenter this command in the route map on R2 and see what happens:

```
R2(config-route-map)#set metric 4294967294
R2(config-route-map)#do show ip ospf database | b External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.10.0     172.16.102.1    3600        0x80000002 0x00C305 0
172.16.23.0     172.16.102.1    3600        0x80000002 0x003487 0
172.16.25.0     172.16.102.1    3600        0x80000002 0x001E9B 0
172.16.35.0     172.16.102.1    3600        0x80000002 0x00AFFF 0
172.16.43.0     172.16.102.1    3600        0x80000002 0x005750 0
172.16.101.0    172.16.101.1    805         0x80000025 0x0060D6 0
172.16.102.0    172.16.102.1    3600        0x80000002 0x00CBA0 0
172.16.103.0    172.16.102.1    3600        0x80000002 0x00C0AA 0
172.16.105.0    172.16.102.1    3600        0x80000002 0x00AABE 0
R2(config-route-map)#
R2(config-route-map)#do show ip ospf database | b External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.101.0    172.16.101.1    828         0x80000025 0x0060D6 0
R2(config-route-map)#
```

The result is interesting. As soon as the metric is reentered, all OSPF external routes learned from EIGRP are immediately poisoned with an OSPF maximum age timer value of 3600 and are then shortly purged from the OSPF database. Therefore, it can be concluded that the metric value is related to the problem of not being able to redistribute prefixes from other dynamic routing protocols into OSPF.

Likely Cause: The metric value specified in the route map is too high for OSPF.

The central question is: What is the maximum metric value used by OSPF? Actually, this question must be qualified further. The central question is: What is the maximum metric value of *external* OSPF routes? This distinction must be made because OSPF has a different metric value for external routes that it does for intra-area and interarea routes. You can use the Cisco IOS Software context-sensitive help to find an answer to this question:

```
R2(config-route-map)# router ospf 1
R2(config-router)#redistribute eigrp 100 metric ?
  <0-16777214>  OSPF default metric
```

The Cisco IOS Software help facility tells us that the maximum OSPF external route metric is 16,777,214. This is a value that is far smaller than the very large value specified in the route map, which is 4,294,967,294. This is why the OSPF routes are being immediately poisoned by the route map. Because the route map value is greater than the maximum value of an OSPF external route, the OSPF process is assigning the metric value of "infinity" and is poisoning the routes. To prove this, you can change the route map metric to the value provided by the Cisco IOS Software—16,777,214:

```
R2(config-router)#route-map test
R2(config-route-map)#set metric 16777214
R2(config-route-map)#do show ip ospf database | b External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.10.0     172.16.102.1    7           0x80000001 0x00BB0F 0
172.16.23.0     172.16.102.1    7           0x80000001 0x002C91 0
172.16.25.0     172.16.102.1    7           0x80000001 0x0016A5 0
172.16.35.0     172.16.102.1    7           0x80000001 0x00A70A 0
172.16.43.0     172.16.102.1    7           0x80000001 0x004F5A 0
172.16.101.0    172.16.101.1    1268        0x80000025 0x0060D6 0
172.16.102.0    172.16.102.1    7           0x80000001 0x00C3AA 0
172.16.103.0    172.16.102.1    7           0x80000001 0x00B8B4 0
```



```
172.16.105.0    172.16.102.1    7           0x80000001 0x00A2C8 0
R2(config-route-map)#end
R2#
```

The OSPF external routes have now been reinstalled in the OSPF database.

Resolution: Set the OSPF metric in the route maps configured on R2 and R4 to 16,777,214.

While the configuration change was made on R2, you can make the same change on R4 and verify that two sources of OSPF externals exist in the OSPF database:

```
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#route-map test
R4(config-route-map)#set metric 16777214
R4(config-route-map)#do show ip ospf database | begin External
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
172.16.10.0     172.16.102.1    125         0x80000001 0x00BB0F 0
172.16.23.0     172.16.102.1    125         0x80000001 0x002C91 0
172.16.23.0     172.16.104.1    10          0x80000001 0x001E9D 0
172.16.25.0     172.16.102.1    125         0x80000001 0x0016A5 0
172.16.25.0     172.16.104.1    10          0x80000001 0x0008B1 0
172.16.35.0     172.16.102.1    125         0x80000001 0x00A70A 0
172.16.35.0     172.16.104.1    10          0x80000001 0x009916 0
172.16.43.0     172.16.102.1    125         0x80000001 0x004F5A 0
172.16.43.0     172.16.104.1    10          0x80000001 0x004166 0
172.16.101.0    172.16.101.1    1383        0x80000025 0x0060D6 0
172.16.101.0    172.16.104.1    10          0x80000001 0x00C0AC 0
172.16.102.0    172.16.102.1    125         0x80000001 0x00C3AA 0
172.16.102.0    172.16.104.1    10          0x80000001 0x00B5B6 0
172.16.103.0    172.16.102.1    125         0x80000001 0x00B8B4 0
172.16.103.0    172.16.104.1    10          0x80000001 0x00AAC0 0
172.16.105.0    172.16.102.1    125         0x80000001 0x00A2C8 0
172.16.105.0    172.16.104.1    10          0x80000001 0x0094D4 0
R4(config-route-map)#end
R4#
```

As can be seen, there are now two route sources for the routes injected into OSPF from the dynamic routing protocols of RIP and EIGRP. As expected, the two route sources are R2 (172.16.102.1) and R4 (172.16.104.1). The redistribution troubleshooting problem has been resolved.

As a closing note, OSPF external routes use a 24-bit metric and the route map allows for the manipulation of a metric that can go as far as 32 bits in size. This is why there is such a disparity between the two metric settings: (1) in a route map and (2) in the actual OSPF **redistribute** command.

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.

```
tclsh
foreach address {
172.16.21.1
172.16.10.1
172.16.101.1
172.16.21.2
172.16.23.2
172.16.25.2
172.16.102.1
172.16.23.3
```

```
172.16.43.3
172.16.35.5
172.16.103.1
172.16.10.4
172.16.43.4
172.16.104.1
172.16.25.5
172.16.35.5
172.16.105.1
172.16.10.6
172.16.106.1
} {ping $address}
```

6. Security Troubleshooting Section

6.1. Symptom: The Telnet connection to R6 is not working.

Analysis and Testing:

You tested the Telnet connection to R6 and found that it was not successful. Here is an example from R1:

```
R1#telnet 172.16.10.6
Trying 172.16.10.6 ...
% Destination unreachable; gateway or host down

R1#
```

Verify the access list on R6:

```
R6#show access-lists
Extended IP access list 101
    10 permit pim any any (33 matches)
    20 permit ospf any any (24 matches)
    30 permit icmp any any echo
    40 permit icmp any any echo-reply
    50 deny ip any any log (56 matches)
R6#
```

Note that the Telnet protocol is not explicitly permitted, therefore it is implicitly denied.

Due to the last line in the access list, the log option is generating console messages that help determine what other control plane traffic must be permitted by the access list. But you do not see any logging messages on the console of R6.

Enable logging to console on R6 so you can examine the logging output, which should be similar to the following:

```
R6#clear access-list counters
R6#
*Aug 27 12:48:44.241: %SEC-6-IPACCESSLOGP: list 101 denied tcp 172.16.102.1(179) ->
172.16.104.1(54448), 1 packet
*Aug 27 12:48:44.241: %SEC-6-IPACCESSLOGP: list 101 denied tcp 172.16.102.1(179) ->
172.16.104.1(14797), 1 packet
*Aug 27 12:48:44.241: %SEC-6-IPACCESSLOGP: list 101 denied tcp 172.16.102.1(179) ->
172.16.104.1(38173), 1 packet
*Aug 27 12:48:44.241: %SEC-6-IPACCESSLOGSP: list 101 denied igmp 172.16.10.1 ->
224.0.0.1 (17), 3 packets
*Aug 27 12:48:44.241: %SEC-6-IPACCESSLOGSP: list 101 denied igmp 172.16.10.4 ->
224.0.1.40 (22), 3 packets
*Aug 27 12:48:44.241: %SEC-6-IPACCESSLOGP: list 101 denied tcp 172.16.102.1(179) ->
172.16.104.1(22495), 2 packets
R6#
```

Note that BGP and IGMP traffic is denied on R6.

It must be noted that due to the OSPF and PIM filtering between the Ethernet0/1 interfaces of R1 and R4, the R6 Ethernet interface is the hub on a hub-and-spoke OSPF and PIM topologies on VLAN 10. Therefore, not only must PIM and OSPF packets be permitted by the access list, BGP must be permitted as well. As can be seen, BGP is missing from the access list.

BGP and Internet Group Management Protocol (IGMP) need to be added to the access list.

Likely Cause: The access list is not configured to permit Telnet, BGP, or IGMP traffic.

The log information generated by the last line of the access list—the "permit ip any any" log message—has helped uncover any hidden problems with this access list. Now you can see that the access list not only did not permit Telnet, but also did not permit BGP and IGMP. Make changes to the access list to support these three protocols.

Resolution: Modify the inbound access list to include BGP and IGMP.

The following is a display showing the final configuration access list:

```
R6#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R6(config)#no logging console
R6(config)#no access-list 101
R6(config)#access-list 101 permit pim any any
R6(config)#access-list 101 permit ospf any any
R6(config)#access-list 101 permit icmp any any echo
R6(config)#access-list 101 permit icmp any any echo-reply
R6(config)#access-list 101 permit tcp any any eq telnet
R6(config)#access-list 101 permit igmp any any
R6(config)#access-list 101 permit tcp any eq bgp any
R6(config)#access-list 101 permit tcp any any eq bgp
R6(config)#access-list 101 deny   ip any any log
R6(config)#end
R6#
```

Verify the Telnet traffic from R1 to R6 again:

```
R1#telnet 172.16.10.6
Trying 172.16.10.6 ... Open

------------------------------------------------
Cisco 360 R&S Exercise Workbook
Product, POD location:  cierswbv5-te-lab10-sc, SJ
Device:                 R6
------------------------------------------------

R6#exit

[Connection to 172.16.10.6 closed by foreign host]
R1#
```

Note that the Telnet connection is successful. Also, R6 no longer generates the logging messages for denied BGP and IGMP traffic.

Once again, you can use the last line of this access list to determine which access list entries must be added in the future for any additional protocols or traffic types that may need to be forwarded through the access list.

7. BGP Troubleshooting Section

7.1. Symptom: A BGP neighbor relationship between R2 and R4 has not been established.

Analysis and Testing:

You found that R2 and R4 are not forming any BGP peer relationships:

```
R2#show ip bgp summary
BGP router identifier 172.16.102.1, local AS number 200
BGP table version is 2, main routing table version 2
1 network entries using 140 bytes of memory
1 path entries using 76 bytes of memory
1/1 BGP path/bestpath attribute entries using 140 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 356 total bytes of memory
BGP activity 4/3 prefixes, 5/4 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
172.16.104.1    4          400       0       0        1    0    0 00:30:08 Idle
R2#
```

After examining the lab diagram, you can see that the External Border Gateway Protocol (EBGP) neighbor relationship is being formed between two EBGP speakers that do not share a common subnet. Furthermore, you must perform careful hop-by-hop analysis of the entire path between the two EBGP speakers. In this lab, you must not overlook the fact that VLAN 10 imitates a hub-and-spoke OSPF topology due to the blocked OSPF traffic between the E0/1 interfaces of R1 and R4 on VLAN 10. Therefore, you must not forget that R6, the hub router on VLAN 10, is part of the path that both R2 and R4 will use to form their EBGP neighbor relationship.

Likely Cause: The TTL for the EBGP TTL session between R2 and R4 must be increased.

This BGP EBGP peer formation issue most likely has to do with the following default characteristic of BGP: by default, EBGP packets are transmitted with a Time to Live (TTL) of 1. Because R2 and R4 do not share a common subnet, their EBGP packets will have their TTL expire and the neighbor relationship will not be formed. As a remedy, both R2 and R4 are already configured during the lab initialization with EBGP multihop in the following manner:

```
R2#sh running-config | section bgp
router bgp 200
 no synchronization
 bgp log-neighbor-changes
 network 172.16.102.0 mask 255.255.255.255
 neighbor 172.16.104.1 remote-as 400
 neighbor 172.16.104.1 ebgp-multihop 2
 neighbor 172.16.104.1 update-source Loopback102
 no auto-summary

R4#show running-config | section bgp
router bgp 400
 no synchronization
 bgp log-neighbor-changes
 network 172.16.104.0 mask 255.255.255.0
 neighbor 172.16.102.1 remote-as 200
 neighbor 172.16.102.1 ebgp-multihop 2
 neighbor 172.16.102.1 update-source Loopback104
no auto-summary
```

Even with this configuration, the EBGP neighbor relationship between R2 and R4 is not being formed.

Once again, because of the hub-and-spoke topology of VLAN 10, there is an extra hop in the path of this EBGP neighbor relationship.

Run the **debug ip icmp** command on R2 and examine the output. Do not forget to enable logging:

```
R2#debug ip icmp
ICMP packet debugging is on
R2#
*Aug 27 13:16:20.905: ICMP: time exceeded rcvd from 172.16.10.6
R2#
*Aug 27 13:16:22.909: ICMP: time exceeded rcvd from 172.16.10.6
R2#un all
R2#
```

When the utility **debug ip error** is enabled on R6, the hub router in the hub-and-spoke topology on VLAN 10, the following debug messages are generated:

```
R6#debug ip error
IP packet errors debugging is on
R6#
*Aug   27   13:15:41.876:   IP:   s=172.16.102.1   (Ethernet0/1),   d=172.16.104.1
(Ethernet0/1), len 44, dispose ip.hopcount
R6#
*Aug   27   13:15:43.876:   IP:   s=172.16.102.1   (Ethernet0/1),   d=172.16.104.1
(Ethernet0/1), len 44, dispose ip.hopcount
R6#
```

Note that R6 is discarding the BGP packets and sending the ICMP time exceeded messages.

Both the R2 and R4 EBGP multihop setting must be increased to 3 to account for the extra hop in the path between R2 and R4.

Resolution: Increase the EBGP multihop setting on R2 and R4 to 3.

-
```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#router bgp 200
R2(config-router)#neighbor 172.16.104.1 ebgp-multihop 3
R2(config-router)#end

R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#router bgp 400
R4(config-router)# neighbor 172.16.102.1 ebgp-multihop 3
R4(config-router)#end
R4#
```

As soon as the **ebgp-multihop** command is adjusted, the EBGP neighbor relationship is formed. It will generate the following console message:

```
R4#
*Aug 27 13:24:26.952: %BGP-5-ADJCHANGE: neighbor 172.16.102.1 Up
R4#
```

7.2. Symptom: The BGP route of 172.16.102.0/24 is not being originated on R2.

Analysis and Testing:

Examine the BGP table on R2:

```
R2#sho ip bgp
BGP table version is 4, local router ID is 172.16.102.1
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
 *>  172.16.104.0/24  172.16.104.1             0             0 400 i
R2#
```

Note that R2 does not originate the Loopback102 network in BGP.

Verify the Loopback102 interface on R2:

```
R2#show ip interface Lo102
Loopback102 is up, line protocol is up
  Internet address is 172.16.102.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1514 bytes
<skipped>
R2#
```

After reviewing the following configuration on R2, you notice that the 172.16.102.0/24 prefix is not being originated into BGP:

```
R2#show run | section bgp
router bgp 200
 no synchronization
 bgp log-neighbor-changes
 network 172.16.102.1 mask 255.255.255.255
 neighbor 172.16.104.1 remote-as 400
 neighbor 172.16.104.1 ebgp-multihop 3
 neighbor 172.16.104.1 update-source Loopback102
 no auto-summary
```

Likely Cause: The BGP network statement is misconfigured.

BGP requires that prefixes specified in a network configuration statement exactly match a mask in the interface and routing table configuration:

```
R2#show ip interface Lo102 | inc 102
Loopback102 is up, line protocol is up
  Internet address is 172.16.102.1/24
R2#
R2#show ip route connected | inc 172.16.102.0
C        172.16.102.0/24 is directly connected, Loopback102
R2#
```

Note that the 172.16.102.1 address is listed as a 172.16.102.0/24 prefix in the R2 routing table. Therefore, the BGP **network** statement must be reconfigured to reflect this.

Resolution: Configure the BGP** network **statement to precisely reflect the 172.16.102.0/24 prefix.

```
R2(config)#router bgp 200
R2(config-router)#network 172.16.102.0 mask 255.255.255.0
```

Verify your configuration change:

```
R2#sho ip bgp
BGP table version is 11, local router ID is 172.16.102.1
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
 *>  172.16.102.0/24  0.0.0.0                  0         32768 i
 *>  172.16.104.0/24  172.16.104.1             0             0 400 i
R2#
```


7.3. The BGP neighbor relationship is flapping.

Analysis and Testing:

Once the BGP peer relationships are up and are exchanging routes for the respective update source address of each BGP speaker, the BGP peer relationship goes down. After it goes down, it comes back up and repeats the cycles as in the following example.

```
R2#
*Aug 27 13:38:28.895: %BGP-5-NBR_RESET: Neighbor 172.16.104.1 reset (Peer closed
the session)
*Aug 27 13:38:28.895: %BGP-3-NOTIFICATION: received from neighbor 172.16.104.1 4/0
(hold time expired) 0 bytes
R2#
*Aug 27 13:38:28.895: %BGP-5-ADJCHANGE: neighbor 172.16.104.1 Down Peer closed the
session
*Aug 27 13:38:28.895: %BGP_SESSION-5-ADJCHANGE: neighbor 172.16.104.1 IPv4 Unicast
topology base removed from session  Peer closed the session
*Aug 27 13:38:29.516: %BGP-5-ADJCHANGE: neighbor 172.16.104.1 Up
R2#
*Aug 27 13:41:29.678: %BGP-3-NOTIFICATION: received from neighbor 172.16.104.1 4/0
(hold time expired) 0 bytes
R2#
*Aug 27 13:41:29.678: %BGP-5-NBR_RESET: Neighbor 172.16.104.1 reset (BGP
Notification received)
*Aug 27 13:41:29.678: %BGP-5-ADJCHANGE: neighbor 172.16.104.1 Down BGP Notification
received
*Aug 27 13:41:29.678: %BGP_SESSION-5-ADJCHANGE: neighbor 172.16.104.1 IPv4 Unicast
topology base removed from session  BGP Notification received
*Aug 27 13:41:30.361: %BGP-5-ADJCHANGE: neighbor 172.16.104.1 Up
```

A symptom of the problem can be best understood by examining the following routing table entry on router R2:

```
R2#sh 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, 19 subnets, 2 masks
B        172.16.104.0/24 [20/0] via 172.16.104.1, 00:01:23
R2#
```

Notice how R2 has installed in its routing table a route for the R4 BGP update source address, using BGP itself as a routing protocol.

Likely Cause: There is a recursive routing issue involving the BGP update source addresses.

Once again, examine the routing table entry above very closely. Notice that it states that in order to forward the packets to the destination address of 172.167.104.0/24, you must use the next hop of 172.16.104.1. This is clearly circular reasoning. Furthermore, notice that the BGP entry doesn't reference an exit interface. When a viable BGP routing entry is inserted in a routing table, the next-hop address contains a second recursive route that references how to reach the next hop using a specific exit interface. In the entry above, the forwarding instructions are circular. The next-hop address implicitly references the 172.16.102.0/24 prefix. As a result, the

BGP neighbor relationship continually collapses and then re-forms. This cycle repeats itself over and over.

Resolution: Configure the network backdoor ***command on both routers R2 and R4.***

```
R2

R2(config)#router bgp 200
R2(config-router)#network 172.16.104.0 mask 255.255.255.0 backdoor

R4

R4(config)#router bgp 400
R4(config-router)#network 172.16.102.0 mask 255.255.255.0 backdoor
```

The effect of the **network backdoor** command is to set the administrative distance of a BGP learned route, namely an EBGP-learned route, to 200. This way, any routes learned from an IGP will be more preferred. Remember that EBGP has a default administrative distance of 20. This can cause problems when an EBGP update source is advertised via EBGP. It will cause this recursive routing issue. With the **network backdoor** command, the route to the EBGP neighbor is now in the local routing table as an OSPF external route:

```
R2#sh ip ro 172.16.104.0
Routing entry for 172.16.104.0/24
  Known via "ospf 1", distance 110, metric 31, type inter area
  Redistributing via eigrp 100
  Advertised by eigrp 100 metric 1000 100 255 1 1500
  Last update from 172.16.21.1 on Ethernet0/1, 00:00:45 ago
  Routing Descriptor Blocks:
  * 172.16.21.1, from 172.16.104.1, 00:00:45 ago, via Ethernet0/1
      Route metric is 31, traffic share count is 1
R2#
R2#show ip route | inc 104
O IA     172.16.104.0/24 [110/31] via 172.16.21.1, 00:02:05, Ethernet0/1
R2#
```

8. QoS Troubleshooting Section

8.1. Symptom: All MQC priority traffic is dropped.

Analysis and Testing:

Whenever possible, thoroughly test your QoS configurations. To test the QoS policy operations, use the extended ping facility to send 1000-byte packets marked EF (DSCP 46, TOS 184). Here you see the results:

```
R2#ping
Protocol [ip]:
Target IP address: 172.16.23.3
Repeat count [5]:
Datagram size [100]: 1000
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
Type of service [0]: 184
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 1000-byte ICMP Echos to 172.16.23.3, timeout is 2 seconds:
```



```
.....
Success rate is 0 percent (0/5)
R2#
```

Note that all packets are dropped.

A good starting point for investigating this problem is to use the most information-rich and versatile MQC show command, the **show policy-map interface** command:

```
R2#show policy-map interface
 Serial1/0

  Service-policy output: ccie

    Class-map: ef (match-all)
      5 packets, 5020 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp ef (46)
      police:
          cir 100000 bps, bc 1000 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 5 packets, 5020 bytes; actions:
          drop
        conformed 0000 bps, exceeded 0000 bps

    Class-map: af41 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp af41 (34)
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 32% (494 kbps)

    Class-map: af11 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp af11 (10)
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 16% (247 kbps)

    Class-map: class-default (match-any)
      48 packets, 2922 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any

      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 48/2922
R2#
```

Note that the five packets are 5020 bytes. Each packet is 1004 bytes. The extra 4 bytes represents the HDLC header.

One parameter of interest in the **show policy-map interface** display above is the bc parameter for the MQC policy configuration.

Likely Cause: The MQC policy burst parameter is set with too low of a value.

When implementing the MQC policy command, a general rule to remember is: do not make the MQC policy burst value less than the number of bytes in the packet being processed by the MQC policy mechanism. If this is done, the MQC policy mechanism will drop all packets.

Verify the initialized MQC configuration on R2:

```
class-map match-all af41
 match dscp af41
class-map match-all ef
 match dscp ef
class-map match-all af11
 match dscp af11
!
policy-map ccie
 class ef
  police cir 100000 bc 1000 conform-action transmit
 class af41
  bandwidth percent 32
 class af11
  bandwidth percent 16
!
```

Resolution: Configure an increase of the MQC policy bc parameter to 1004.

Each frame processed by this policy includes the 1000-byte IP packet plus the 4-byte HDLC header.

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#policy-map ccie
R2(config-pmap)# class ef
R2(config-pmap-c)#  police cir 100000 bc 1004 conform-action transmit
R2(config-pmap-c-police)#end
R2#
```

Once this command is entered, its setting can be verified with the following Cisco IOS Software **show** command:

```
R2#show policy-map interface
 Serial1/0

  Service-policy output: ccie

    Class-map: ef (match-all)
      5 packets, 5020 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp ef (46)
      police:
          cir 100000 bps, bc 1004 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 5 packets, 5020 bytes; actions:
          drop
        conformed 0000 bps, exceeded 0000 bps

    Class-map: af41 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp af41 (34)
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 32% (494 kbps)
```



```
    Class-map: af11 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp af11 (10)
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 16% (247 kbps)

    Class-map: class-default (match-any)
      251 packets, 15949 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any

      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 251/15949
R2#
```

Test the MQC policy configuration again:

```
R2#ping
Protocol [ip]:
Target IP address: 172.16.23.3
Repeat count [5]:
Datagram size [100]: 1000
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
Type of service [0]: 184
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 1000-byte ICMP Echos to 172.16.23.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 90/91/94 ms
R2#show policy-map interface
 Serial1/0

  Service-policy output: ccie

    Class-map: ef (match-all)
      10 packets, 10040 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp ef (46)
      police:
          cir 100000 bps, bc 1004 bytes
        conformed 5 packets, 5020 bytes; actions:
          transmit
        exceeded 5 packets, 5020 bytes; actions:
          drop
        conformed 0000 bps, exceeded 0000 bps

    Class-map: af41 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp af41 (34)
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 32% (494 kbps)
```

```
    Class-map: af11 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp af11 (10)
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 0/0
      bandwidth 16% (247 kbps)

    Class-map: class-default (match-any)
      285 packets, 18175 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any

      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 290/23195
R2#
```

9. IP Multicast Troubleshooting Section

9.1. Symptom: R4 stops responding to multicast pings.

Analysis and Testing:

Pings to multicast address 239.255.1.1 from R2 result in replies similar to the following:

```
R2#ping 239.255.1.1 source 172.16.21.2 rep 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 239.255.1.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.21.2

Reply to request 0 from 172.16.101.1, 1 ms
Reply to request 0 from 172.16.106.1, 1 ms
Reply to request 1 from 172.16.101.1, 1 ms
Reply to request 1 from 172.16.106.1, 1 ms
Reply to request 2 from 172.16.101.1, 1 ms
Reply to request 2 from 172.16.106.1, 1 ms
Reply to request 3 from 172.16.101.1, 5 ms
Reply to request 3 from 172.16.106.1, 5 ms
Reply to request 4 from 172.16.101.1, 1 ms
Reply to request 4 from 172.16.106.1, 1 ms
R2#
```

Note that R4 is not responding. Here is the configuration for the multicast group 239.255.1.1 on R6, the RP for this group:

```
R6#show ip pim rp mapping 239.255.1.1
PIM Group-to-RP Mappings

Group(s): 224.0.0.0/4, Static
    RP: 172.16.106.1 (?)
R6#
R6#sh ip mroute 239.255.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
```



```
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector
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), 00:07:31/00:03:23, RP 172.16.106.1, flags: SJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse, 00:07:24/00:03:23
    Loopback106, Forward/Sparse, 00:07:31/00:02:22

(172.16.21.2, 239.255.1.1), 00:00:06/00:02:57, flags: LT
  Incoming interface: Ethernet0/1, RPF nbr 172.16.10.1
  Outgoing interface list:
    Loopback106, Forward/Sparse, 00:00:06/00:02:53
```

Note that R6 is configured as an RP and has a (*,G) entry in its mroute table. Loopback106 and Ethernet 0/1 are listed as the outgoing interfaces.

```
R6#

R6#show ip pim neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode
172.16.10.1       Ethernet0/1              02:56:43/00:01:41 v2    1 / S P G
172.16.10.4       Ethernet0/1              00:16:15/00:01:42 v2    1 / S P G
R6#
```

Note that R6 forms PIM neighbor relationships with R1 and R4.

Verify which member of the multicast group 239.255.1.1 joins the RP on R6. Here is selected output of **debug ip pim 239.255.1.1** from R6:

```
R6#deb ip pim 239.255.1.1
PIM debugging is on
R6#
*Aug  27  02:48:20.644:  PIM(0):  Received  v2  Join/Prune  on  Ethernet0/1  from
172.16.10.1, to us
*Aug 27 02:48:20.644: PIM(0): Join-list: (*, 239.255.1.1), RPT-bit set, WC-bit set,
S-bit set
*Aug 27 02:48:20.644: PIM(0): Update Ethernet0/1/172.16.10.1 to (*, 239.255.1.1),
Forward state, by PIM *G Join
R6#
*Aug 27 02:48:49.737: PIM(0): Building Periodic (*,G) Join / (S,G,RP-bit) Prune
message for 239.255.1.1
R6#
*Aug  27  02:49:19.344:  PIM(0):  Received  v2  Join/Prune  on  Ethernet0/1  from
172.16.10.1, to us
*Aug 27 02:49:19.344: PIM(0): Join-list: (*, 239.255.1.1), RPT-bit set, WC-bit set,
S-bit set
*Aug 27 02:49:19.344: PIM(0): Update Ethernet0/1/172.16.10.1 to (*, 239.255.1.1),
Forward state, by PIM *G Join
R6#
*Aug 27 02:49:20.737: PIM(0): Send RP-reachability for 239.255.1.1 on Ethernet0/1
R6#
*Aug 27 02:49:49.237: PIM(0): Building Periodic (*,G) Join / (S,G,RP-bit) Prune
message for 239.255.1.1
R6#
```

```
*Aug  27  02:50:19.144:  PIM(0):  Received  v2  Join/Prune  on  Ethernet0/1  from
172.16.10.1, to us
*Aug 27 02:50:19.144: PIM(0): Join-list: (*, 239.255.1.1), RPT-bit set, WC-bit set,
S-bit set
*Aug 27 02:50:19.144: PIM(0): Update Ethernet0/1/172.16.10.1 to (*, 239.255.1.1),
Forward state, by PIM *G Join
R6#

R6#u all
All possible debugging has been turned off
R6#
```

Note that R6 receives PIM Join/Prune messages only from R1 172.16.10.1. R6 does not see any PIM Join/Prune messages from R4.

Likely Cause: R4 does not generate PIM Join/Prune messages.

Run the **debug ip pim 239.255.1.1** on R4 and examine the output:

```
R4# debug ip pim 239.255.1.1
PIM debugging is on
R4#
*Aug  27  02:59:50.738:  PIM(0):  Received  RP-Reachable  on  Ethernet0/1  from
172.16.106.1
*Aug  27  02:59:50.738:  PIM(0):  Received  RP-Reachable  on  Ethernet0/1  from
172.16.106.1
*Aug 27 02:59:50.738:      for group 239.255.1.1
*Aug 27 02:59:50.738: PIM(0): Group 239.255.1.1 not found
R4#
*Aug  27  03:00:10.240:  PIM(0):  Received  v2  Join/Prune  on  Ethernet0/1  from
172.16.10.1, not to us
R4#u all
All possible debugging has been turned off
R4#
```

Note that R4 cannot find group 239.255.1.1.

According to the lab requirements, the Loopback104 interface on R4 should be joined to 239.255.1.1:

```
R4#show ip igmp interface lo104
Loopback104 is up, line protocol is up
  Internet address is 172.16.104.1/24
  IGMP is enabled on interface
  Current IGMP host version is 2
  Current IGMP router version is 2
  IGMP query interval is 60 seconds
  IGMP configured query interval is 60 seconds
  IGMP querier timeout is 120 seconds
  IGMP configured querier timeout is 120 seconds
  IGMP max query response time is 10 seconds
  Last member query count is 2
  Last member query response interval is 1000 ms
  Inbound IGMP access group is not set
  IGMP activity: 2 joins, 1 leaves
  Multicast routing is disabled on interface
  Multicast TTL threshold is 0
  Multicast groups joined by this system (number of users):
      239.255.1.1(1)
R4#
```

Note that the Loopback104 interface joined the group 239.255.1.1.

Verify the mroute table on R4:

```
R4#show ip mroute 239.255.1.1
Group 239.255.1.1 not found
R4#
```


Note that the group 239.255.1.1 is not in the mroute table. Is PIM enabled on the Loopback0 interface?

```
R4#show ip pim interface

Address          Interface                Ver/   Nbr    Query  DR     DR
                                          Mode   Count  Intvl  Prior
172.16.10.4      Ethernet0/1              v2/S   2      30     1      172.16.10.6
R4#
```

Note that PIM is not enabled on the Loopback104 interface on R4.

Resolution: Configure PIM on the Loopback104 interface of R4 and verify the multicast connectivity again.

To resolve this problem, configure the command **ip pim sparse mode** on the Loopback104 interface on R4.

```
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#int lo104
R4(config-if)#ip pim sparse-mode
R4(config-if)#end
R4#
R4#
R4#
R4#show ip mroute 239.255.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector
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), 00:00:11/00:02:49, RP 172.16.106.1, flags: SJCL
  Incoming interface: Ethernet0/1, RPF nbr 172.16.10.6
  Outgoing interface list:
    Loopback104, Forward/Sparse, 00:00:10/00:02:49

R4#
```

When you repeat the ping from R2, you should get continued responses from all three routers: R1, R6 and R4.

```
R2#ping 239.255.1.1 source 172.16.21.2 rep 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 239.255.1.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.21.2

Reply to request 0 from 172.16.101.1, 9 ms
Reply to request 0 from 172.16.104.1, 29 ms
Reply to request 0 from 172.16.106.1, 9 ms
```

```
Reply to request 1 from 172.16.101.1, 1 ms
Reply to request 1 from 172.16.104.1, 1 ms
Reply to request 1 from 172.16.106.1, 1 ms
Reply to request 1 from 172.16.106.1, 1 ms
Reply to request 2 from 172.16.101.1, 1 ms
Reply to request 2 from 172.16.104.1, 1 ms
Reply to request 2 from 172.16.106.1, 1 ms
Reply to request 3 from 172.16.101.1, 1 ms
Reply to request 3 from 172.16.104.1, 1 ms
Reply to request 3 from 172.16.106.1, 1 ms
Reply to request 4 from 172.16.101.1, 1 ms
Reply to request 4 from 172.16.106.1, 1 ms
Reply to request 4 from 172.16.104.1, 1 ms
R2#
```

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**.

