360RSW05-WKB-CA06

Cisco 360 CCIE R&S Exercise Workbook Introduction

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

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

Cisco 360 CCIE R&S Exercise Workbook Lab 6 Configuration Section Answer Key

Table of Contents

Answer Key Structure

Section One

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

Section Two

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

Exercise Workbook Lab 6 Configuration 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

- Configuration lab duration: 6 hours
- Configuration lab maximum score: 76 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 to Advanced

Restrictions and Goals

Note	Read this section carefully.

- To receive credit for a subsection, you must fully complete the subsection per the requirements. You will *not* receive partial credit for partially completed subsections.
- IPv4 subnets that are displayed in the scenario diagram belong to network 170.18.0.0/16.
- *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 for IPv4 and IPv6 protocols.
- Do not use the **ip default-network** or **ip default-gateway** commands.
- All IP addresses that are involved in the same virtual routing and forwarding instance, or VRF, must be reachable, unless an explicitly stated filtering requirement restricts reachability.
- Do not create new interfaces, and do not summarize prefixes unless you are explicitly asked to do so.

- Do not introduce any new IPv4 or IPv6 addresses unless the instructions specifically require it.
- Use conventional routing algorithms only.
- Do not modify the hostname, console, or vty configuration unless you are specifically asked to do so.
- Do not modify the initial interface or IP address numbering.

Explanation of Each of the Restrictions and Goals

IPv4 subnets that are displayed in the scenario IPv4 IGP diagram belong to network 172.18.0.0/16.

All IP addresses in this lab belong to the 172.18.0.0/16 address space, except for prefixes that are used in the BGP section.

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. OSPF treats loopback interfaces as host routes by default and advertises them as /32 prefixes. The requirement to advertise loopback interfaces with their original masks precludes using the default OSPF network type for the loopback interface. You need to provide a solution such as changing the OSPF network type or summarizations.

Do not use the ip default-network **command.**

This command can be used to solve reachability issues by setting the gateway of last resort. This command generates 0.0.0.0/0 in the Routing Information Protocol (RIP) environment. You cannot use it in this scenario.

All IP addresses that are involved in the same VRF must be reachable, unless an explicitly stated filtering requirement restricts reachability.

This is a key goal to observe. This requires that all of your interior gateway protocols (IGPs) are configured properly. In addition, all of 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-list**, **route-map**, and **distance** commands. A key point to remember throughout this lab is that the term "redistribution" is never explicitly used. However, you must perform redistribution to assure that all IP addresses are reachable without the use of static routes.

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 type of information other than the destination address to make a packet forwarding decision.

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

1. Switch Configuration

General Tasks:

As with any switch configuration, you must address the following basic configuration requirements: setting the VLAN Trunking Protocol (VTP) mode, configuring trunk ports, and statically assigning ports to VLANs. For a good reference on mastering basic Cisco Catalyst

3560 Switch configuration tasks, access the full set of Catalyst video-on-demand (VoD) sessions within the “Link Layer” lesson in the Cisco 360 learning portal. These self-paced sessions provide more than 7 hours of instruction on a range of basic Catalyst switch configuration tasks.

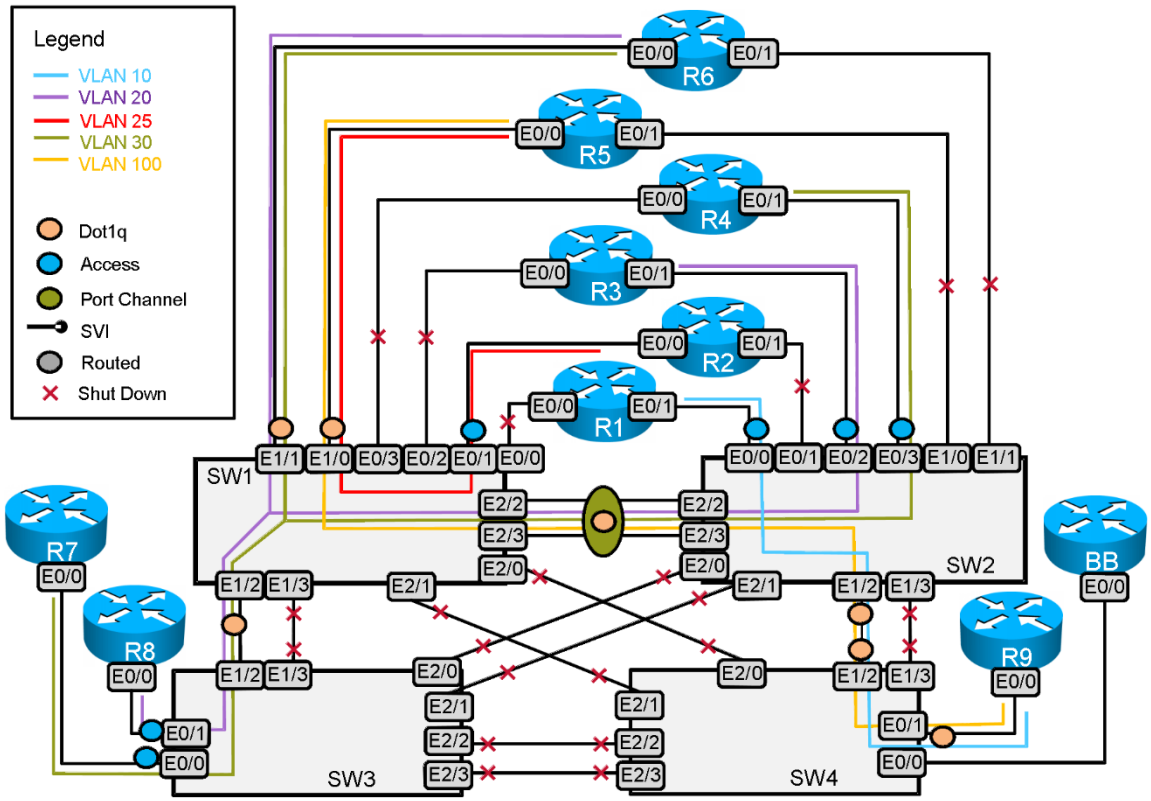
Note that not all Cisco Catalyst 3560 Switch configuration features are supported on the virtual Cisco IOS Software on UNIX.

Configure the VLANs and the VLAN names according to the scenario specifications and assign the ports of the switches to these VLANs. Make sure that the VLAN names are spelled correctly and match the letter case.

Use the “VLAN” and “Switch-to-Router Connections” tables to analyze the VLAN propagation in this lab.

See the following diagram for the VLAN layout.

VLAN Propagation Diagram



Carefully review the entire scenario. Closely examine the supplied diagram and any associated tables. Determine how you need to configure VTP, how to configure ports that are assigned as trunks, and how to configure ports that are assigned as simply static VLAN ports. Use the **switchport mode access** command to statically assign ports to a VLAN.

Some of the steps you must take to perform the configuration that is required by this scenario include the following:

- Configure SW1, SW2, SW3, and SW4 in VTP transparent mode by issuing the **vtp mode transparent** command.

- Create all the necessary VLANs for this exercise—10, 20, 25, 30, and 100—by issuing the **vlan 10,20,25,30,100** command.
- Configure access switch ports as follows:

```
interface EthernetX/X
 switchport mode access
 switchport access vlan N
```

- Configure trunk ports, allowing only the necessary VLANs to be on a trunk; for example, on SW1:

```
interface Ethernet1/1
switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30
 switchport mode trunk
```

- To verify VTP mode, issue the **show vtp status** command. This is an example of VTP verification on SW1:

```
SW1#show vtp status
VTP Version capable             : 1 to 3
VTP version running             : 1
VTP Domain Name                 :
VTP Pruning Mode                : Disabled
VTP Traps Generation            : Disabled
Device ID                       : aabb.cc00.0700
Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Feature VLAN:
--------------
VTP Operating Mode              : Transparent
Maximum VLANs supported locally : 1005
Number of existing VLANs        : 10
Configuration Revision          : 0
MD5 digest                      : 0x73 0x0F 0xAF 0x62 0x50 0xE5 0x05 0x9D
                                  0xCA 0xD1 0x8C 0xFE 0xFB 0xE5 0xD4 0x3A
SW1#
```

- Verify access port assignment using the **show VLAN brief** command. Here is an example on SW2:

```
SW2#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et0/1, Et1/0, Et1/1, Et1/3
                                                Et2/0, Et2/1
10   VLAN0010                         active    Et0/0
20   VLAN0020                         active    Et0/2
25   VLAN0025                         active
30   VLAN0030                         active    Et0/3
100  VLAN0100                         active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
SW2#
```

- Verify trunk status using the **show interface trunk** command. The following output is an example from SW4:

```
SW4#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Et0/1       on               802.1q         trunking      1
Et1/2       on               802.1q         trunking      1
```

```
Port        Vlans allowed on trunk
Et0/1       10,100
Et1/2       10,100

Port        Vlans allowed and active in management domain
Et0/1       10,100
Et1/2       10,100

Port        Vlans in spanning tree forwarding state and not pruned
Et0/1       10,100
Et1/2       10,100
SW4#
```

Issue: Make sure that you see the following output from the **show etherchannel summary** command.

Solution:

This task requires you to bundle parallel links E2/1 and E2/2 into an EtherChannel using the Cisco Port Aggregation Protocol (PAgP).

Configure the EtherChannel on SW1 and SW2:

SW1:

```
interface Ethernet2/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30,100
 switchport mode trunk
 duplex auto
 channel-group 1 mode desirable
!
interface Ethernet2/3
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30,100
 switchport mode trunk
 duplex auto
 channel-group 1 mode desirable
!
interface Port-channel1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30,100
 switchport mode trunk
!
```

SW2:

```
interface Ethernet2/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30,100
 switchport mode trunk
 duplex auto
 channel-group 1 mode desirable
!
interface Ethernet2/3
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30,100
 switchport mode trunk
 duplex auto
 channel-group 1 mode desirable
!
interface Port-channel1
 switchport
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30,100
 switchport mode trunk
```



```
!
```

Note that the channel group is configured in the desirable mode. The desirable mode negotiates the PAgP EtherChannels. Also, according to the requirements of the lab, the EtherChannel bundle of the E2/2 and E2/3 interface between SW1 and SW2 is configured as a dot1q trunk.

Verify the EtherChannel status, protocol, and the trunking status. This is an example on SW1:

```
SW1#show etherchannel summary | begin Number
Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
-
1      Po1(SU)          PAgP      Et2/2(P)    Et2/3(P)

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

Group: 1
----------
Protocol:  PAgP

SW1#
SW1#show interface trunk


Port        Mode             Encapsulation  Status        Native vlan
Et1/0       on               802.1q         trunking      1
Et1/1       on               802.1q         trunking      1
Et1/2       on               802.1q         trunking      1
Po1         on               802.1q         trunking      1

Port        Vlans allowed on trunk
Et1/0       25,100
Et1/1       20,30
Et1/2       20,30
Po1         20,30,100

Port        Vlans allowed and active in management domain
Et1/0       25,100
Et1/1       20,30
Et1/2       20,30
Po1         20,30,100

Port        Vlans in spanning tree forwarding state and not pruned
Et1/0       25,100
Et1/1       20,30
Et1/2       20,30

Port        Vlans in spanning tree forwarding state and not pruned
Po1         20,30,100
SW1#
```

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

2. IPv4 OSPF

Note	This section was partially preconfigured in the initialization file so that you could concentrate on more advanced topics.

Issue: Configure OSPF Area 0 between R1, R2, and R3. Use an OSPF network type that elects a DR but does not require a neighbor statement. Make sure that R3 is the DR and that the other OSPF speakers are DROTHERs on subnet 170.18.123.0/24.

Solution:

The OSPF network type that elects a DR but does require a neighbor statement is the OSPF broadcast network type. The OSPF network type "broadcast" is the default OSPF network type on the Ethernet 170.18.123.0/24 subnet. Set R1 and R2 to an OSPF priority of 0 to prevent them from being elected as either a DR or backup designated router (BDR). Here is an example on R1:

```
interface Ethernet0/3
 ip ospf network broadcast
 ip ospf priority 0
!
```

Issue the **show ip ospf interface e0/3** command to verify the OSPF network type on the link, the neighbors that are discovered, and the neighbors with adjacencies. Here is an example on R3:

```
R3#show ip ospf interface e0/3
Ethernet0/3 is up, line protocol is up
  Internet Address 170.18.123.3/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 170.18.103.1, Network Type BROADCAST, Cost: 10
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           10        no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 170.18.103.1, Interface address 170.18.123.3
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:07
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 3/3, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 3
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 170.18.101.1
    Adjacent with neighbor 170.18.102.1
  Suppress hello for 0 neighbor(s)
R3#
```

Issue the **show ip ospf neighbor e0/3** command to verify the OSPF neighbor state and the OSPF priority. Here is an example on R3:

```
R3#show ip ospf neighbor e0/3

Neighbor ID     Pri   State           Dead Time   Address         Interface
170.18.101.1      0   FULL/DROTHER    00:00:38    170.18.123.1    Ethernet0/3
170.18.102.1      0   FULL/DROTHER    00:00:38    170.18.123.2    Ethernet0/3
R3#
```


Issue: Configure OSPF Area 0 on the subnet between R1, R3, and R4. Use an OSPF network type that elects a DR and requires neighbor statements. Make sure that R4 is the designated router and the other OSPF speakers are DROTHER on the subnet 170.18.134.0/24.

Solution:

The OSPF network type that elects a DR and requires a neighbor statement is the OSPF nonbroadcast network type. Therefore, configure the OSPF network type "nonbroadcast" on the 170.18.134.0/24 subnet. Set R1 and R3 to an OSPF priority of 0 to prevent them from being elected as either a DR or BDR. Here is an example on R1:

```
interface Ethernet0/2
ip ospf network non-broadcast
 ip ospf priority 0
```

Configure neighbor statements in the router OSPF process on R4, referencing R1 and R3 as follows:

R4:

```
router ospf 1
neighbor 170.18.134.1
neighbor 170.18.134.3
```

Issue the **show ip ospf interface e0/2** command to verify the OSPF network type on the link, the neighbors that are discovered, and the neighbors with adjacencies. Here is an example on R4:

```
R4#show ip ospf interface e0/2
Ethernet0/2 is up, line protocol is up
  Internet Address 170.18.134.4/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 170.18.104.1, Network Type NON_BROADCAST, Cost: 10
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           10        no          no            Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 170.18.104.1, Interface address 170.18.134.4
  No backup designated router on this network
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    oob-resync timeout 120
    Hello due in 00:00:11
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 3
  Last flood scan time is 0 msec, maximum is 1 msec
Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 170.18.101.1
    Adjacent with neighbor 170.18.103.1
  Suppress hello for 0 neighbor(s)
R4#
```

Issue the **show ip ospf neighbor e0/2** command to verify the OSPF neighbor state and the OSPF priority. Here is an example on R4:

```
R4#show ip ospf neighbor e0/2

Neighbor ID     Pri   State           Dead Time   Address         Interface
170.18.101.1      0   FULL/DROTHER    00:00:46    170.18.134.1    Ethernet0/2
170.18.103.1      0   FULL/DROTHER    00:00:52    170.18.134.3    Ethernet0/2
R4#
```

Issue: For the subnet that is shared by R1, R3, and R4, make sure that the loss of a neighbor relationship is detected twice as fast as the default.

Solution:

To fulfill this requirement, configure one of the two following OSPF interface configuration commands: **ip ospf hello-interval 15** or **ip ospf dead-interval 60.** The values are set to 15 for the **new-hello** interval and 60 for the **new-dead** interval, since the default **hello** and **dead** intervals for an OSPF nonbroadcast network type are 30 and 120, respectively.

R4 (example):

```
R4#show run int e0/2
Building configuration...

Current configuration : 126 bytes
!
interface Ethernet0/2
 ip address 170.18.134.4 255.255.255.0
 ip ospf network non-broadcast
 ip ospf hello-interval 15
end

R4#
```

Verify the updated OSPF timers on R4:

```
R4#show ip ospf interface e0/2 | inc Time
  Timer intervals configured, Hello 15, Dead 60, Wait 60, Retransmit 5
R4#
```

Issue: Place VLAN10 and loopback 120 into OSPF Area 1. Make sure that OSPF sends the minimum information to R9.

Solution:

Since dynamic default routes are not ruled out by the scenario ground rules, the most straight-forward way to accomplish this task would be to make Area 1 totally stubby. Issue the command **area 1 stub** on R9 and the command **area 1 stub no-summary** on R1. Note that all routers in a stub area must have the **stub** keyword, but only the Area Border Router (ABR) must have the **no-summary** keyword. Making the area stubby keeps out external prefixes (no Type 5 link-state advertisements [LSAs]). Adding the **no-summary** keyword keeps the ABR from sending Type 3 and Type 4 LSAs into the area.

Here are configuration examples on R1 and R9:

```
R1#show running-config | sec router ospf
router ospf 1
 area 1 stub no-summary
 network 170.18.10.0 0.0.0.255 area 1
 network 170.18.101.1 0.0.0.0 area 0
 network 170.18.123.0 0.0.0.255 area 0
 network 170.18.134.0 0.0.0.255 area 0
R1#

R9#show running-config | sec router ospf
router ospf 1
 area 1 stub
 network 170.18.0.0 0.0.255.255 area 1
R9#
```

Verify the OSPF routing table on R9:


```
R9#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 170.18.10.1 to network 0.0.0.0

O*IA  0.0.0.0/0 [110/11] via 170.18.10.1, 02:15:07, Ethernet0/0.10
R9#
```

Note that R9 receives only the default route 0.0.0.0/0 from R1 via OSPF.

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

3. IPv4 RIP

Note	This section was partially preconfigured in the initialization file so that you could concentrate on more advanced topics.

Issue Configure RIPv2 updates to exchange only over the VLAN25 connection between R2 and R5. Send the minimum required information from R2 to R5.

Solution:

This lab requires that the RIPv2 updates should be exchanged only over the VLAN25 connection. You can control the RIPv2 update transmission using the **passive-interface** command.

The minimum required information would be a 170.18.0.0/16 summary, since all addresses that must be reached are from this subnet.

Here is a RIPv2 configuration example on R2:

```
R2#sho running-config | sec router rip
router rip
 version 2
 passive-interface default
 no passive-interface Ethernet0/0
 network 170.18.0.0
R2#sho running-config int e0/0
Building configuration...

Current configuration : 114 bytes
!
interface Ethernet0/0
 ip address 170.18.25.2 255.255.255.0
 ip summary-address rip 170.18.0.0 255.255.0.0
```

```
end

R2#
```

Issue: Allow only the following networks to be advertised to R2 using the minimum number of access list statements:

192.80.2.0/24 -192.88.3.0/24
192.80.3.0/24 -170.18.105.0/24
192.88.2.0/24

Solution:

This configuration requirement can be fulfilled with the following access list:

```
access-list 17 permit 192.80.2.0 0.8.1.0
access-list 17 permit 170.18.105.0
```

This list allows the fourth bit in the second octet and the first bit in the third octet to vary, resulting in the required match. Since the task says "advertised to," the access list should be applied to an outbound distribute list under the RIP routing process of R5. The outbound distribute list should explicitly reference the Et0/0.25 interface of R5.

Here is a RIPv2 configuration example on R5:

```
R5#sh running-config | sec router rip
router rip
 version 2
 passive-interface default
 no passive-interface Ethernet0/0.25
 network 170.18.0.0
 network 192.80.1.0
 network 192.80.2.0
 network 192.80.3.0
 network 192.80.4.0
 network 192.88.1.0
 network 192.88.2.0
 network 192.88.3.0
 network 192.88.4.0
 distribute-list 17 out Ethernet0/0.25
 no auto-summary
R5#
R5#show access-lists
Standard IP access list 17
    10 permit 170.18.105.0 (663 matches)
    20 permit 192.80.2.0, wildcard bits 0.8.1.0 (2652 matches)
R5#

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

Gateway of last resort is not set

      170.18.0.0/16 is variably subnetted, 9 subnets, 3 masks
R        170.18.0.0/16 [120/1] via 170.18.25.2, 00:00:12, Ethernet0/0.25
R5#
```


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

4. IPv4 EIGRP

Note	This section was preconfigured in the initialization file so that you could concentrate on more advanced topics.

Issue: Configure EIGRP AS 100 between R4, R3, R6, and R7.

Solution:

R3:

```
router eigrp 100
 network 170.18.255.0 0.0.0.255
!
```

R4:

```
router eigrp 100
 network 170.18.64.0 0.0.0.255
!
```

R6:

```
router eigrp 100
 network 170.18.64.0 0.0.0.255
 network 170.18.106.0 0.0.0.255
 network 170.18.255.0 0.0.0.255
!
```

R7:

```
router eigrp 100
 network 170.18.64.0 0.0.0.255
 network 170.18.110.0 0.0.0.255
!
```

Verify the EIGRP neighbor relationships. Here is an example on R6:

```
R6#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
2   170.18.64.10            Et0/0.30                 12 02:47:18    5   100  0  11
1   170.18.64.4             Et0/0.30                 13 02:50:42    1   100  0  11
0   170.18.255.3            Et0/0.20                 10 02:50:43    2   100  0  14
R6#
```

Note that R6 forms EIGRP neighbor relationships with R3, R4, and R7.

Verify the EIGRP routing table on R3:

```
R3#show ip route eigrp
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
```

```
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
        a - application route
        + - replicated route, % - next hop override

Gateway of last resort is not set

      170.18.0.0/16 is variably subnetted, 19 subnets, 2 masks
D        170.18.64.0/24 [90/307200] via 170.18.255.6, 02:58:13, Ethernet0/1
D        170.18.106.0/24 [90/409600] via 170.18.255.6, 02:58:13, Ethernet0/1
D        170.18.110.0/24 [90/435200] via 170.18.255.6, 02:54:47, Ethernet0/1
R3#
```

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

5. Routing Stability

Note	This section was preconfigured in the initialization file so that you could concentrate on more advanced topics.

Issue: Imagine that the E0/1 interface on R4 is flapping, causing instability throughout the network. Implement a feature on this interface that will isolate failures so that disturbances are not propagated.

Solution:

You should implement the damping feature on this interface. You can find it documented at this link:
http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/s_ipevdp.html

Issue: Configure the maximum configurable value for the half-life period and set the suppress threshold to 200 seconds, the reuse threshold value to 10 percent of the Cisco IOS default value, and the maximum suppress time to a value twice the value of the reuse threshold..

Solution:

To determine the correct dampening parameters settings, refer to the documentation at this link:
http://www.cisco.com/en/US/docs/ios/iproute_pi/command/reference/iri_pi1.html#wp1011620

Implement the event dampening on the Ethernet0/1 interface on R4:

```
R4#show run int e0/1 | inc damp
 dampening 30 100 200 200
R4#

R4#show ip int brief | inc E.*0/1
Ethernet0/1                170.18.64.4     YES manual up                    up
R4#

R4#show interface dampening
Ethernet0/1
  Flaps Penalty    Supp ReuseTm   HalfL  ReuseV   SuppV  MaxSTm    MaxP Restart
      0       0   FALSE       0      30     100     200     200   10158       0
R4#
```

Note that the interface dampening suppression state is currently FALSE.

You can test the interface dampening feature by performing a shut/no shut operation on the E0/1 interface of R4 several times. To get a response, we lowered the thresholds and increased the half-life. Note in the output shown below that when the interface is dampened, the connected interface is not in the routing table. It returns when the penalty expires.

```
R4#show ip int brief | inc E.*0/1
Ethernet0/1                170.18.64.4     YES manual up                    up
R4#

R4#show interface dampening
Ethernet0/1
  Flaps Penalty    Supp ReuseTm   HalfL  ReuseV   SuppV  MaxSTm    MaxP Restart
      6    1339    TRUE     111      30     100     200     200   10158        0
R4#

R4#show ip route con
...
      170.18.0.0/16 is variably subnetted, 16 subnets, 2 masks
C        170.18.104.0/24 is directly connected, Loopback104
L        170.18.104.1/32 is directly connected, Loopback104
C        170.18.134.0/24 is directly connected, Ethernet0/2
L        170.18.134.4/32 is directly connected, Ethernet0/2
R4#
```

Note that in this example after a few flaps the Ethernet0/1 interface was suppressed and the connected entry for the Ethernet0/1 interface was removed from the routing table.

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

6. IPv4 Route Redistribution

Before examining the specific issues that are related to configuring each of the IGPs that are involved in this scenario, it is a good idea to first survey the entire topology and determine how all of the different IGPs will interoperate. Performing such a survey will force you to consider the issues that are related to route redistribution.

When evaluating a single internetwork topology that contains multiple routing protocols, a good starting point of analysis is to determine whether there is more than one direct or indirect connecting point between two routing protocols. If there is only one connecting point between two routing protocols, providing connectivity between them is relatively simple. If there are two or more connecting points, then providing connectivity between the two routing protocols can be complex. When two or more connecting points exist, you can use them to provide redundancy, as well as load balancing and optimum path selection. However, when two or more connecting points exist, you must also assure, at the very least, that no routing loops exist and, whenever possible, that no suboptimal paths are selected.

In this scenario, the core protocol is OSPF. RIP is an edge protocol; it will not provide transit services. RIP routes are redistributed into OSPF, but OSPF routes are not redistributed into RIP. Instead a summary is provided to the RIP domain to enable R5 to reach other routers within the pod.

EIGRP is another protocol; it will not provide transit services. However, there are two redistribution points, R3 and R4, between EIGRP and OSPF, and all OSPF routes are

redistributed into EIGRP. Mutual redistribution at two or more points should always raise a red flag for route feedback. However, this scenario requires filtering in the EIGRP-to-OSPF direction. Upon close examination, the filtering requirement limits route propagation to EIGRP internal routes only.

R3 and R4

```
router ospf 1
 redistribute eigrp 100 subnets route-map EIGRP-->OSPF
!
route-map EIGRP-->OSPF permit 10
 match route-type internal
!
```

The following table provides a useful summary of the prefixes that were imported into a given routing protocol. Whenever a permit column for a given routing protocol is completely empty, it reflects that no prefixes were redistributed into the routing protocol. In these cases, the routing protocol is involved in one-way redistribution.

IPv4 IGP Redistribution

Redist Point	Into RIP	Into OSPF	Into EIGRP 1
R2	---	All RIP routes	
R3		EIGRP internal routes	All OSPF routes

Below is a Tool Command Language (Tcl) script that you can use to test universal reachability. To use the script, enter the command **tclsh** in privileged mode, and paste in this script. To stop failing pings, hold down **Ctrl-Shift** and press the **6** key twice. When you are finished, enter **tclq** to exit Tcl mode. This list excludes the 140.10.0.0/16 and 1.1.1.0/24 addresses that are part of the BGP task:

```
tclsh
foreach addr {
170.18.134.1
170.18.10.1
170.18.123.1
170.18.101.1
170.18.25.2
170.18.123.2
170.18.102.1
170.18.134.3
170.18.255.3
170.18.123.3
170.18.103.1
170.18.134.4
170.18.104.1
170.18.64.1
170.18.64.4
192.80.3.1
192.80.2.1
192.88.3.1
192.88.2.1
170.18.25.5
170.18.105.1
170.18.255.6
170.18.255.10
170.18.106.1
170.18.64.6
170.18.255.1
170.18.110.1
```



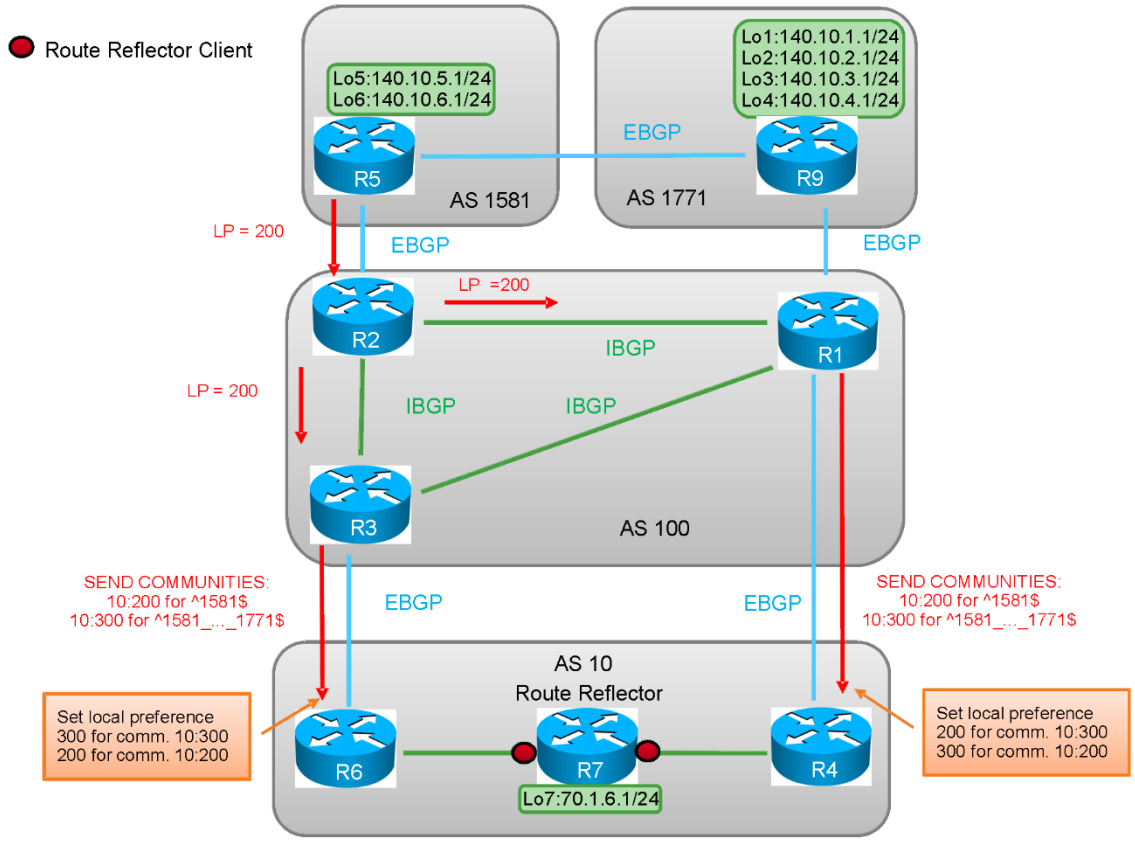
```
170.18.64.10
170.18.10.20
170.18.120.1
} {ping $addr}
```

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

7. Border Gateway Protocol

The figure shows the required BGP peering. Most candidates find the creation of such a diagram to be a good first step.

BGP Diagram



Issue: Do not peer R4 and R6 to meet the requirements of this section.

Solution:

To avoid configuring a full mesh in AS 10, you can configure a route reflector or a confederation. If you are not directed to create a confederation, configure a route reflector instead, since it is simpler to do. Configure R7 to be a route reflector by configuring its neighbors as follows:

R7:

```
router bgp 10
 no synchronization
 bgp log-neighbor-changes
 network 70.1.6.0 mask 255.255.255.0
 neighbor 170.18.64.4 remote-as 10
 neighbor 170.18.64.4 route-reflector-client
 neighbor 170.18.64.6 remote-as 10
 neighbor 170.18.64.6 route-reflector-client
 no auto-summary
!
```

Issue: Prefer R2 as an exit point to networks 140.10.*.0/24 that were advertised from AS 1581 and AS 1771.

Solution:

As one of the configuration options to make R2 the preferred exit point for 140.10.*.0/24 networks, assign a higher local preference for these prefixes on R5 and retain the default local preference setting on all other routers. The default local preference is 100; a higher local preference is preferred. By setting all R5 learned 140.10.*.0/24 prefixes to a higher local preference on R2, R5 will act as the exit point for these prefixes in AS 100 via R2.

R2:

```
router bgp 100
 neighbor 170.18.25.5 route-map Set-Local-Pref in
!
ip as-path access-list 10 permit _(1581|1771)$
!
ip prefix-list 140.10.*.0/24 seq 5 permit 140.10.0.0/16 ge 24 le 24
!
route-map Set-Local-Pref permit 10
 match ip address prefix-list 140.10.*.0/24
 match as-path 10
 set local-preference 200
!
route-map Set-Local-Pref permit 20
!
```

Note that the configured route map matches the prefix list for the 140.10.*.0/24 prefixes and the as-path access list for the BGP AS 1581 and AS 1771. Read the next issue solution explanation for more details on the as-path access list configuration.

Issue the **show ip bgp regexp** command on R1, and verify that R2 is a preferred exit point:

```
R1# show ip bgp regexp _(1581|1771)$
BGP table version is 21, local router ID is 170.18.101.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 * i 1.1.1.0/24       170.18.25.5              0    100      0 1581 i
```



```
 *>                    170.18.10.20             0             0 1771 i
 *>i 140.10.1.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.2.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.3.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.4.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.5.0/24     170.18.25.5              0    200      0 1581 i
 *                     170.18.10.20                           0 1771 1581 i
 *>i 140.10.6.0/24     170.18.25.5              0    200      0 1581 i
 *                     170.18.10.20                           0 1771 1581 i
R1#
```

Note that R5 is used as a next hop for the 140.10.*.*/24 prefixes on R1. The next IP address 170.18.25.5 is reachable via R2 via OSPF. Therefore, all the traffic destined to the 140.10.*.*/24 prefixes will be forwarded to R2 within the BGP AS 100:

```
R1#show ip route 170.18.25.5
Routing entry for 170.18.25.0/24
  Known via "ospf 1", distance 110, metric 20, type extern 2, forward metric 10
  Last update from 170.18.123.2 on Ethernet0/3, 21:43:04 ago
  Routing Descriptor Blocks:
  * 170.18.123.2, from 170.18.102.1, 21:43:04 ago, via Ethernet0/3
      Route metric is 20, traffic share count is 1
R1
```

Issue: R1 should accept from R9 only prefixes originating in AS 1581 or AS 1771.

Solution:

This task calls for an **as-path** filter. It should permit only paths that end in either 1581 or 1771. All prefixes that have paths that end in any other AS number should be denied. The following as-path access list meets this requirement:

```
router bgp 100
 bgp log-neighbor-changes
 redistribute connected
 neighbor 170.18.10.20 remote-as 1771
 neighbor 170.18.10.20 filter-list 10 in
!
ip forward-protocol nd
!
ip as-path access-list 10 permit _(1581|1771)$
!
```

The underscore indicates that any symbols or the beginning of the string may precede the target AS number, and the dollar sign indicates the end of the path. The pipe symbol indicates an “or.” To test the filter, run the **show ip bgp regexp _(1581|1771)$** command on R1. You should see matches only from the prefixes that have paths that end in 1581 or 1771.

```
R1#show ip bgp regexp _(1581|1771)$
BGP table version is 21, local router ID is 170.18.101.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
```

```
 * i 1.1.1.0/24        170.18.25.5              0    100      0 1581 i
 *>                    170.18.10.20             0             0 1771 i
 *>i 140.10.1.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.2.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.3.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.4.0/24     170.18.25.5              0    200      0 1581 1771 i
 *                     170.18.10.20             0             0 1771 i
 *>i 140.10.5.0/24     170.18.25.5              0    200      0 1581 i
 *                     170.18.10.20                           0 1771 1581 i
 *>i 140.10.6.0/24     170.18.25.5              0    200      0 1581 i
 *                     170.18.10.20                           0 1771 1581 i
R1#
```

Issue: In AS 10, set the local preference for all prefixes originating from AS 1771 and traversing AS 1581 to prefer R6 as a next hop. In addition, set the local preference for all prefixes originating from AS 1581 to prefer R4 as a next hop. Use values 200 and 300 to accomplish this task. Do not use the AS path or the IP address prefix as match criteria to set the local preference in AS 10.

Solution:

If you cannot set the local preference in AS 10 by matching on AS path strings or prefixes, you can use either the AS path or address prefix as a matching criteria in AS 100 and set a unique community string to the desired prefixes in AS 100. The restriction stated above, "Do not use the AS path or the IP address prefix as match criteria to set the local preference in AS 10," applies only to AS 10 and *not* to AS 100. Once the community is set in AS 100, it will be propagated to AS 10. Once it reaches AS 10, you can set the local preferences to the specified values that are based on community strings. Remember that for all BGP speakers that need to advertise a prefix with a community attached to it, you must enter the BGP neighbor command **neighbor X.X.X.X send-community**. Refer to the diagram for the community and local preference values sent between the autonomous systems.

R1:

```
router bgp 100
 neighbor 170.18.134.4 route-map Set-Community
out
!
ip bgp-community new-format
ip as-path access-list 2 permit ^1581$
ip as-path access 3 permit _1581_([0-9]+_)*1771$
!
route-map Set-Community permit 10
 match as-path 2
 set community 10:200
!
route-map Set-Community permit 20
 match as-path 3
 set community 10:300
!
route-map Set-Community permit 30
!
```

R3:

```
router bgp 100
 neighbor 170.18.255.6 route-map Set-Community
out
!
ip bgp-community new-format
ip as-path access-list 2 permit ^1581$
ip as-path access 3 permit _1581_([0-9]+_)*1771$
!
route-map Set-Community permit 10
 match as-path 2
 set community 10:200
!
route-map Set-Community permit 20
 match as-path 3
 set community 10:300
!
route-map Set-Community permit 30
!
```

R4:

```
router bgp 10
 neighbor 170.18.134.1 route-map Set-Local-Pref
in
!
ip bgp-community new-format
ip community-list 2 permit 10:200
ip community-list 3 permit 10:300
!
route-map Set-Local-Pref permit 10
 match community 2
 set local-preference 300
!
route-map Set-Local-Pref permit 20
 match community 3
 set local-preference 200
!
route-map Set-Local-Pref permit 30
!
```

R6:

```
router bgp 10
 neighbor 170.18.255.3 route-map Set-Local-Pref
in
!
ip bgp-community new-format
ip community-list 2 permit 10:200
ip community-list 3 permit 10:300
!
route-map Set-Local-Pref permit 10
 match community 2
 set local-preference 200
!
route-map Set-Local-Pref permit 20
 match community 3
 set local-preference 300
!
route-map Set-Local-Pref permit 30
!
```

R5 is AS 1581; R9 is AS 1771. Both autonomous systems originate prefixes into the BGP network: four prefixes from AS 1771 and two prefixes from AS 1581.

When these prefixes enter AS 100, R2 is selected as a preferred exit point in AS 100. Setting a local preference of 200 to all prefixes that are received on R2 from R5 accomplishes this goal. R2 will advertise the new local preference value to the other AS 100 peers.

In AS 10, the outbound traffic to the networks that originated from AS 1581 should be forwarded via R6, and the outbound traffic to the networks that originated in AS 1771 and traverse AS 1581 should prefer R4 as an exit point. This configuration requires us to set the local preference to 300 for prefixes that originated in AS 1771 and are passing thru AS 1581, and to local preference 200 for all other prefixes. On R4, a local preference of 300 should be assigned for prefixes that originated in AS 1581, and a local preference of 200 should be assigned to others.

The restriction of not using AS path or IP address filters in AS 10 does not prohibit using them in AS 100. In AS 100. These prefixes will be classified using **as-path access-list** commands and will be assigned communities 10:200 for prefixes that originated in AS 1581 and 10:300 for prefixes that originated in AS 1771 and traverse AS 1581. AS 10 sets local preferences according to these community numbers.

On R7, you can verify the BGP prefixes that are originated in BGP AS 1771 and traversed via BGP AS 1581.

```
R7#show ip bgp reg _1581_([0-9]+_)*1771$
BGP table version is 23, local router ID is 170.18.110.1
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 140.10.1.0/24    170.18.255.3             0    300      0 100 1581 1771 i
 *>i 140.10.2.0/24    170.18.255.3             0    300      0 100 1581 1771 i
 *>i 140.10.3.0/24    170.18.255.3             0    300      0 100 1581 1771 i
 *>i 140.10.4.0/24    170.18.255.3             0    300      0 100 1581 1771 i
R7#
R7#show ip route 170.18.255.3
Routing entry for 170.18.255.0/24
  Known via "eigrp 100", distance 90, metric 307200, type internal
  Redistributing via eigrp 100
  Last update from 170.18.64.6 on Ethernet0/0, 18:24:41 ago
  Routing Descriptor Blocks:
  * 170.18.64.6, from 170.18.64.6, 18:24:41 ago, via Ethernet0/0
      Route metric is 307200, traffic share count is 1
      Total delay is 2000 microseconds, minimum bandwidth is 10000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1
R7#
```

Note that the regular expression _1581_([0-9]+_)*1771$ will match the general case when transit BGP AS 1581 is not necessarily a neighboring AS to BGP AS 1771. As you can see from the command outputs, R6 is used as the next hop for the prefixes that are originated in BGP AS 1771 and traversed via BGP AS 1581.

On R7, you can verify the BGP prefixes that are originated in BGP AS 1581.

```
R7#show ip bgp reg _1581$
BGP table version is 13, local router ID is 170.18.110.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 140.10.5.0/24    170.18.64.4              0    300      0 100 1581 i
 *>i 140.10.6.0/24    170.18.64.4              0    300      0 100 1581 i
R7#
R7#show ip route 170.18.64.4
Routing entry for 170.18.64.0/24
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Redistributing via eigrp 100
  Routing Descriptor Blocks:
  * directly connected, via Ethernet0/0
      Route metric is 0, traffic share count is 1
R7
```

As you can see from the command outputs, R4 is used as the next hop for the prefixes that are originated in BGP AS 1581.

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

8. MPLS Layer 3 VPNs

Issue: Configure VPN46 on R4 and R6 using the route distinguisher value 10:46. Include only interfaces loopback 114 on R4 and loopback 116 on R6 in the VPN. Use the export route-target value 10:114 on R4 and the export route-target value 10:116 on R6.

Solution:

Create VRF VPN46 using the route distinguisher value 10:46 on both R4 and R6. The challenge here is to remember to use the import route-target 10:116 on R4 and the import route-target 10:114 on R6. Add the required loopback into the VRF using the **ip vrf forwarding** command, and re-enter the IP address. Here you see the relevant configuration for R4:

```
ip vrf VPN46
 rd 10:46
 route-target export 10:114
 route-target import 10:116

interface Loopback114
 ip vrf forwarding VPN46
 ip address 10.114.0.1 255.255.255.0
```

Issue: Configure BGP to support site-to-site reachability.

Solution:

BGP AS 10 was configured on R4 and R6 as part of an earlier task, but they peer through R7 using route reflection. This task requires a direct peering that is used only for VPNv4 addresses. The trick is to deactivate this peering for the IPv4 address family. As usual, the peering must be explicitly activated for the VPNv4 address family. To enable connectivity between the loopback addresses in the VPN, redistribute connected subnets into BGP under the IPv4 address family for the VRF. Finally, use the command **mpls ip** on the interfaces connecting R4 and R6 to enable the Label Distribution Protocol (LDP). This is needed to add VPN labels to the VPN traffic. Here is the relevant BGP configuration on R4.

```
router bgp 10
 bgp log-neighbor-changes
 neighbor 170.18.64.10 remote-as 10
 neighbor 170.18.106.1 remote-as 10
 neighbor 170.18.106.1 update-source Loopback104
 neighbor 170.18.134.1 remote-as 100
 !
 address-family ipv4
  neighbor 170.18.64.10 activate
  neighbor 170.18.64.10 next-hop-self
  no neighbor 170.18.106.1 activate
  neighbor 170.18.134.1 activate
  neighbor 170.18.134.1 route-map Set-Local-Pref in
!
 address-family vpnv4
  neighbor 170.18.106.1 activate
  neighbor 170.18.106.1 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf VPN46
  redistribute connected
  no synchronization
 exit-address-family
```

These following two commands demonstrate that the peering with 170.18.106.1 is exclusively for the VPNv4 address family. By default, all configured peerings are used for the IPv4 address family, and you would see it listed under **show ip bgp summary**.

```
R4#show ip bgp summary | begin Neighbor
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
170.18.64.10    4           10      19      20       17    0    0 00:10:48        5
170.18.134.1    4          100      22      16       17    0    0 00:11:55       11
R4#

R4#show bgp vpnv4 unicast all summary | begin Neighbor
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
170.18.106.1    4           10      18      18        4    0    0 00:12:45        1
R4#
```

Verify the Layer 3 VPN BGP table on R4:

```
R4#show bgp vpnv4 unicast all
BGP table version is 4, local router ID is 170.18.104.1
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
Route Distinguisher: 10:46 (default for vrf VPN46)
 *>  10.114.0.0/24    0.0.0.0                  0         32768 ?
 *>i 10.116.0.0/24    170.18.106.1             0    100      0 ?
R4#
```

Note that R4 shows a locally originated prefix 10.114.0.0/24 and a prefix 10.116.0.0/24 learned from R6 via BGP.

Verify reachability within the VPN using the command **ping vrf**, as you see here:

```
R4#ping vrf VPN46 10.116.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.116.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

R6#ping vrf VPN46 10.114.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.114.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

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

9. Router Maintenance

Issue: Configure R8 to act as a host with an IP address of 170.18.255.1/24. R8 should dynamically prefer R3 as a gateway and use R6 when R3 is not available. Do not use HSRP, VRRP, or GLBP. Do not use any static configuration.

Solution:

If R8 will act as a host device, you can disable IP routing in global configuration mode and rely on a gateway discovery protocol, such as ICMP Router Discovery Protocol (IRDP). Configure IRDP on the VLAN 20-assigned interfaces of routers R3 and R6, and configure the **ip gdp irdp** command in global configuration mode on R8.

To make R3 the more preferred gateway on VLAN 20 for R8, set R3 with a higher IRDP preference than R6.

Note Choosing the best gateway based on IRDP preference is a function of the client.

RFC 1256 specifies the following:

> The host is "expected to choose from those router addresses that have the highest preference level"

Cisco IOS Software documentation for the **ip irdp** command agrees:

> A higher value increases the preference level of the router. You can modify a particular router so that it will be the preferred router to which other routers will home.

However, some Cisco IOS Software releases actually prefer a lower preference value.

It is recommended that in a situation in which the client is under control during configuration, the configuration should include the preference (higher or lower)of the client, and then configuration should be performed in accordance with the testing results. If the client is not available for testing, then perform configuration assuming that client prefers the higher value.

By default, IRDP is a protocol in which advertisements rarely occur. The hold-time value default is 30 minutes.

```
R6#show ip irdp Ethernet0/0.20
Advertisements will occur between every 450 and 600 seconds.
Advertisements are sent with broadcasts.
Advertisements are valid for 1800 seconds.
```

Configure R3 and R6 as IRDP information sources on VLAN 20, and adjust IRDP (or ICMP) timers:

R3:

```
interface Ethernet0/1
 ip address 170.18.255.3 255.255.255.0
 ip irdp
 ip irdp maxadvertinterval 30
 ip irdp minadvertinterval 10
 ip irdp holdtime 90
 ip irdp preference 200
!
```

R6:

```
interface Ethernet0/0.20
 encapsulation dot1Q 20
 ip address 170.18.255.6 255.255.255.0
 ip irdp
 ip irdp maxadvertinterval 30
```

```
 ip irdp minadvertinterval 10
 ip irdp holdtime 90
 ip irdp preference 100
!
```

Configure R8 as an IRDP client: Issue the **ip gdp irdp** global configuration command.

R8:

```
ip gdp irdp
```

Issue the **show ip route** command on R8:

```
R8#show ip route
Gateway         Using  Interval  Priority   Interface
170.18.255.3    IRDP       35       200     Ethernet0/0
170.18.255.6    IRDP       39       100     Ethernet0/0

Default gateway is 170.18.255.3

Host               Gateway           Last Use    Total Uses  Interface
ICMP redirect cache is empty
R8#
```

Note that R3 is preferred as the default gateway.

Issue: All packets that originate from R8, except for the packets that are destined to R3, should have the source IP address changed at the first-hop router. The source IP address must be in the 170.18.255.0/24 range.

Solution:

This is a Network Address Translation (NAT) problem. From a basic NAT configuration perspective, the R3 VLAN20 interface will be the NAT inside the interface, and the R3 Ethernet0/2 interface will be the NAT outside the interface.

Implement the same configuration on R6 to ensure that this requirement is satisfied in situations in which R8 loses its primary gateway and falls back to the secondary gateway (R6), and to support asymmetric traffic flows (such as communication between R8 and R2: direct path R8-R3-R2, and return path R2-R6-R8). Note that since this is a static address translation scenario, there is no need to use any special NAT features.

The traffic that is received by R6 and destined to R3 may not fit in NAT schema. For example, it may be returned asymmetrically, which will make reachability impossible. To avoid this issue, exclude such traffic from NAT on R6.

R3:

```
interface Ethernet0/3
 ip address 170.18.123.3 255.255.255.0
 ip nat outside
!
interface Ethernet0/2
 ip address 170.18.134.3 255.255.255.0
 ip nat outside
!
interface Ethernet0/1
 ip address 170.18.255.3 255.255.255.0
 ip nat inside
!
ip nat inside source static 170.18.255.1 170.18.255.10
!
```

R6:

```
interface Ethernet0/0.20
 encapsulation dot1Q 20
```

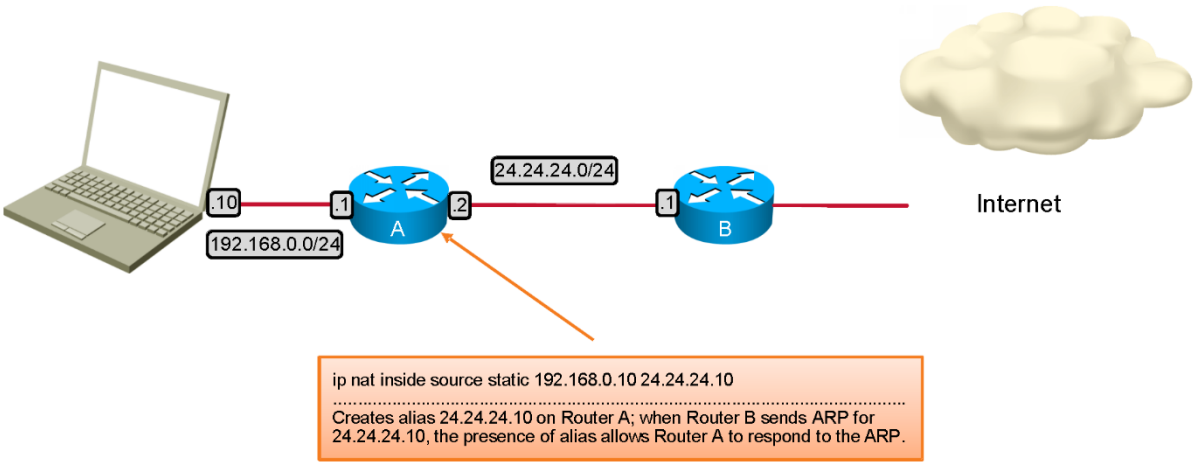


```
 ip address 170.18.255.6 255.255.255.0
 ip nat inside
!
interface Ethernet0/0.30
 encapsulation dot1Q 30
 ip address 170.18.64.6 255.255.255.0
 ip nat outside
!
ip nat inside source static 170.18.255.1 170.18.255.10 route-map NAT no-alias
!
ip access-list extended NAT
 deny   ip any host 170.18.123.3
 deny   ip any host 170.18.134.3
 deny   ip any host 170.18.103.1
 permit ip any any
!
route-map NAT permit 10
 match ip address NAT
!
```

Note that R6 is configured with the **no-alias** keyword.

Cisco IOS Software includes an auto-aliasing feature. This feature creates an alias on the router to enable it to reply to ARP requests. To understand how the feature works, examine a simple static NAT on the Internet border:

Static NAT Example



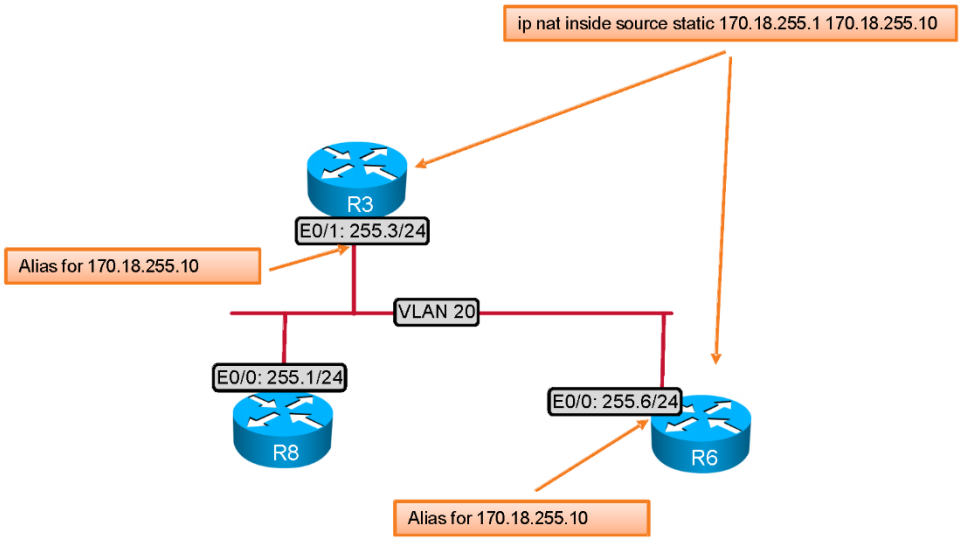
In this example, Router A provides static NAT for a PC on a LAN segment that is addressed using the RFC 1918 address space, and an unused IP address from the 24.24.24.0/24 segment is used for NAT. Router B and all Internet hosts will see packets from the PC that are sourced with IP address 24.24.24.10. When Router B receives a packet for the PC, it sends an ARP request for 24.24.24.10, since this address is on Router B's connected subnet. To establish communication, Router A must reply to the ARP packet with its MAC address. This is exactly what alias enables—it allows Router A to reply to the ARP request.

As shown in the above example, the aliases are needed when an IP space with NAT is assigned from unused IP space on an interface. Aliases are not necessary if the IP space with NAT was assigned from an IP space not on any interface, or, for example, from an IP space on a loopback interface. In this case, IP routing would have provided a means for Router B to deliver the packet.

Cisco IOS Software creates aliases for NAT pools that are used as an inside global IP address or an outside local IP address when the pool comprises addresses on an attached subnet, as well as for inside global or outside local addresses in static entries. Furthermore, when the Cisco IOS Software creates an alias, it does not differentiate on the attached subnet being on the local or global side. (An alias is created in both cases.)

Here is how the aliases will be created in this scenario:

Static NAT in This Scenario



Note that because 170.18.255.10 is being used for the inside global IP address, there is no need for an alias because global routers are not connected to this segment. However, the presence of duplicate aliases on a single segment does have the undesirable effect of looking like a duplicate

IP address. As a workaround, you can switch off auto-aliasing on one or both routers using the **no-alias** keyword of the **ip nat** command.

To verify your NAT configuration, try to connect via Telnet from R8 to 172.18.102.1 on R2, for example:

```
R8#telnet 170.18.102.1
Trying 170.18.102.1 ... Open

------------------------------------------------
Cisco 360 R&S Exercise Workbook
Product, POD location:  cierswbv5-ce-lab06-sc, SJ
Device:                 R2
------------------------------------------------

R2#who
    Line       User       Host(s)              Idle       Location
   0 con 0                 idle                 01:40:30   cierswbv5-ce-lab06-
sc, SJ
*  2 vty 0                 idle                 00:00:00 170.18.255.10

  Interface    User               Mode         Idle     Peer Address

R2#exit
 [Connection to 170.18.102.1 closed by foreign host]
R8#
```

Note that R8 is represented with the IP address 170.18.255.10 from the NAT pool.

Issue the **show ip nat translations** command on R3:

```
R3#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
tcp 170.18.255.10:58680 170.18.255.1:58680 170.18.102.1:23   170.18.102.1:23
--- 170.18.255.10      170.18.255.1       ---                ---
R3#
```

Issue: Configure R7 so that it can be managed by a network management service that uses UDP port 161. Set read-only access using the string of **RS-CCIE**. Set read-write access using the string **CCIE**.

Solution:

Configure Simple Network Management Protocol, or SNMP, on R7 using the read-only and read-write community strings that are specified. SNMP uses UDP port 161.

```
snmp-server community RS-CCIE RO
snmp-server community CCIE RW
```

Issue: Configure HSRP between R4 and R6. Make R4 the preferred gateway. Switch to R6 if the Ethernet0/2 interface on R4 goes down. The virtual gateway IP address is 170.18.64.1.

Solution:

To fulfill this requirement, configure HSRP between R4 and R6 using the virtual gateway IP address of 170.18.64.1. Set R4 with a higher standby-group priority than R6. To enable switching the preferred gateway from R4 to R6, when the Ethernet0/2 interface on R4 becomes inactive, configure the “standby track” option on R4 so that R4 decreases its standby priority, making R6 preferred. Here are the commands to configure HSRP on R4 and R6:

R4:

```
track 1 interface Ethernet0/2 line-protocol
```

```
!
interface Ethernet0/1
 ip address 170.18.64.4 255.255.255.0
 no ip redirects
 standby 0 ip 170.18.64.1
 standby 0 priority 110
 standby 0 preempt
 standby 0 track 1 decrement 20
!
```

R6:

```
interface Ethernet0/0.30
 encapsulation dot1Q 30
 ip address 170.18.64.6 255.255.255.0
 no ip redirects
 standby 0 ip 170.18.64.1
 standby 0 preempt
!
```

Issue the **show standby** command on R4 and R6:

R4:

```
R4#show standby
Ethernet0/1 - Group 0
  State is Active
    5 state changes, last state change 16:21:46
  Virtual IP address is 170.18.64.1
  Active virtual MAC address is 0000.0c07.ac00
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.736 secs
  Preemption enabled
  Active router is local
  Standby router is 170.18.64.6, priority 100 (expires in 10.336 sec)
  Priority 110 (configured 110)
    Track object 1 state Up decrement 20
  Group name is "hsrp-Et0/1-0" (default)
R4#
```

R6:

```
R6#show standby
Ethernet0/0.30 - Group 0
  State is Standby
    7 state changes, last state change 02:57:49
  Virtual IP address is 170.18.64.1
  Active virtual MAC address is 0000.0c07.ac00
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.920 secs
  Preemption enabled
  Active router is 170.18.64.4, priority 110 (expires in 7.656 sec)
  Standby router is local
  Priority 100 (default 100)
  IP redundancy name is "hsrp-Et0/0.30-0" (default)
```

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

10. IPv6 Routing

Issue: Establish R1-to-R3 IPv6 connectivity and assign IPv6 addresses.

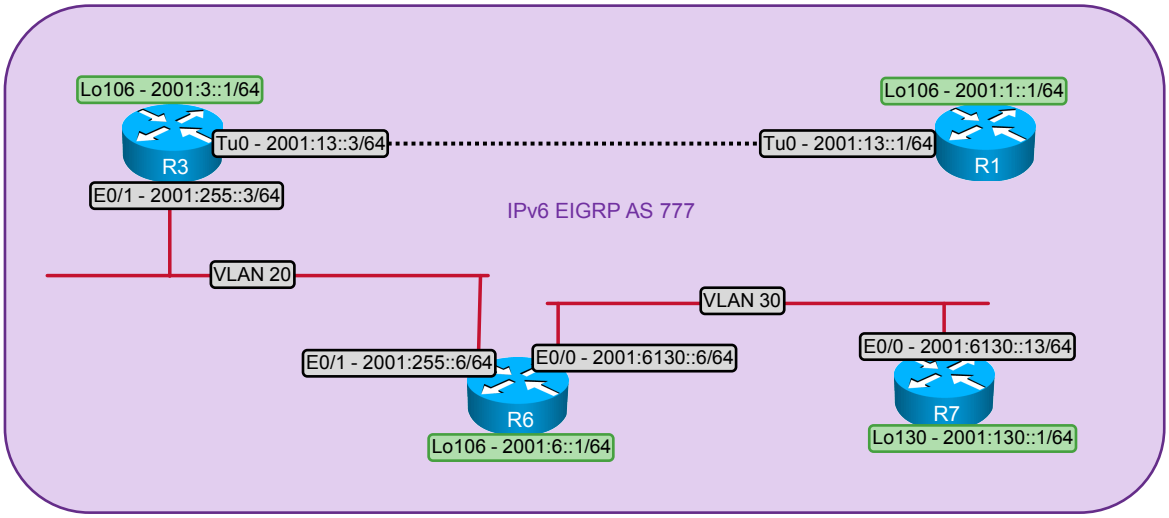
Solution:

The IPv6 connection between R1 and R3 is an IPv6 IP tunnel. Since there are only two devices to be connected over the IPv4 area, a manual tunnel is the simplest approach. Here is the configuration on R1:

```
interface Tunnel10
 no ip address
 ipv6 address 2001:13::1/64

 tunnel source Loopback101
 tunnel mode ipv6ip
 tunnel destination 170.18.103.1
```

IPv6 Diagram



Issue: Enable IPv6 EIGRP AS 777.

Solution:

To accomplish this task, enter the command **ipv6 eigrp 777** on each IPv6 interface and enter the **ipv6 router eigrp 777** global configuration command on each IPv6 router. Do not forget to enable the IPv6 unicast routing on the IPv6 routers.

Here is an example configuration on R3:

```
ipv6 unicast-routing
!
interface Loopback103
ipv6 address 2001:3::1/64
 ipv6 eigrp 777
!
interface Tunnel10
 no ip address
 ipv6 address 2001:13::3/64
 ipv6 eigrp 777
 tunnel source Loopback103
 tunnel mode ipv6ip
 tunnel destination 170.18.101.1
!
interface Ethernet0/1
  ipv6 address 2001:255::3/64
 ipv6 eigrp 777
!
ipv6 router eigrp 777
```

Next, you see the resulting IPv6 EIGRP routes on R3. Note that IPv6 EIGRP uses an administrative distance of 90 for the internal prefixes and the composite EIGRP metric, just like IPv4 EIGRP. Notice, however, that the next hop is a link-local IPv6 address.

```
R3#show ipv6 route eigrp
IPv6 Routing Table - default - 11 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, HA - Home Agent, MR - Mobile Router, R - RIP
       H - NHRP, I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       IS - ISIS summary, D - EIGRP, EX - EIGRP external, NM - NEMO
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2, ls - LISP site
       ld - LISP dyn-EID, a - Application
D   2001:1::/64 [90/27008000]
     via FE80::AA12:6501, Tunnel10
D   2001:6::/64 [90/409600]
     via FE80::A8BB:CCFF:FE00:600, Ethernet0/1
D   2001:130::/64 [90/435200]
     via FE80::A8BB:CCFF:FE00:600, Ethernet0/1
D   2001:6130::/64 [90/307200]
     via FE80::A8BB:CCFF:FE00:600, Ethernet0/1
R3#
```

Issue: Enable IPv6 multicast from R1 to R6 for the group FF08:106:1. Statically configure the RP as 2001:3::1.

Solution:

The command **ipv6 multicast-routing** automatically enables PIM on active IPv6 interfaces. After entering just this command on R3 and R6, you would see that R3 and R6 are IPv6 PIM neighbors; no explicit interface configuration is necessary.

```
R3#show ipv6 pim neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, G - GenID Capable
Neighbor Address             Interface          Uptime    Expires  Mode DR pri

FE80::A8BB:CCFF:FE00:600     Ethernet0/1        19:50:39  00:01:16 B G  DR 1

R3#
```


IPv6 PIM supports both static and BSR methods for RP configuration. Here, enter the command **ipv6 pim rp-address 2001:3::1** on R3 and R6. IPv6 uses virtual tunnel interfaces to support the PIM registration process, as you see here:

```
R6#show ipv6 pim tunnel
Tunnel0*
  Type  : PIM Encap
  RP    : 2001:3::1
  Source: 2001:6::1
Tunnel1*
  Type  : PIM Encap
  RP    : Embedded RP Tunnel
  Source: 2001:6::1

R6#
```

The embedded RP tunnel feature is on by default.

Finally, simulate a client on loopback 106 with the command **ipv6 mld join-group FF08:106::1**. Multicast Listener Discovery (MLD) is similar in function and operation to IGMP.

```
R6#show ipv6 mld groups
MLD Connected Group Membership
Group Address                           Interface           Uptime    Expires
FF08:106::1                             Loopback106         22:20:19  never

R6#
```

Here is the result when you test a ping from R1:

```
R1#ping ff08:106::1
Output Interface: tunnel10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FF08:106::1, timeout is 2 seconds:
Packet sent with a source address of 2001:13::1

Reply to request 0 received from 2001:6::1, 54 ms
Reply to request 1 received from 2001:6::1, 1 ms
Reply to request 2 received from 2001:6::1, 1 ms
Reply to request 3 received from 2001:6::1, 1 ms
Reply to request 4 received from 2001:6::1, 1 ms
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/11/54 ms
5 multicast replies and 0 errors.
R1#

R3#show ipv6 mroute
Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group,
       C - Connected, L - Local, I - Received Source Specific Host Report,
       P - Pruned, R - RP-bit set, F - Register flag, T - SPT-bit set,
       J - Join SPT, Y - Joined MDT-data group,
       y - Sending to MDT-data group
       g - BGP signal originated, G - BGP Signal received
       q - BGP Src-Active originated, Q - BGP Src-Active received
Timers: Uptime/Expires
Interface state: Interface, State

(*, FF08:106::1), 19:47:12/00:02:54, RP 2001:3::1, flags: S
  Incoming interface: Tunnel4
  RPF nbr: 2001:3::1
  Immediate Outgoing interface list:
    Ethernet0/1, Forward, 19:47:12/00:02:54

(2001:13::1, FF08:106::1), 00:00:30/00:02:59, flags: SFT
  Incoming interface: Tunnel10
```

```
  RPF nbr: 2001:13::1
  Immediate Outgoing interface list:
    Ethernet0/1, Forward, 00:00:29/00:02:59
```

R3

Similar to IPv4 multicast, you see the (*,G) state for the shared tree to the RP and the (S,G) state for the source tree.

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

11. Quality of Service

Issue: Restrict Test TCP, or TTCP, traffic from R9 that is destined to 170.18.11.1 port 5001 to 1000000 b/s on R1. Allow burst traffic up to 512,000 bytes. Traffic that exceeds the above-specified condition shall be dropped.

Solution:

Configure traffic policing on the appropriate inbound R1 Ethernet subinterface. This configuration can be performed using either the **rate-limit** command or the Modular QoS CLI (MQC). If you use the MQC, you need to perform five steps:

1) Create an access list to match the appropriate TCP traffic.
2) Associate the access list with a class map.
3) Associate the class map to a policy map.
4) Enter a **police** command under the policy map.
5) Apply the policy map to the R1 appropriate inbound Ethernet subinterface.

If you use the **rate-limit** command, you need to perform two steps:

1) Create an access list to match the appropriate TCP traffic.

2) Apply the access list to an inbound **rate-limit** command on the appropriate Ethernet interface on R1.

Since the MQC configuration is ruled out by the lab restrictions, the **rate-limit** command is used in this lab.

On R1, create an access list to match the traffic pattern:

```
access-list 107 permit tcp host 170.18.10.20 host 170.18.11.1 eq 5001
```

Under interface Ethernet0/1, configure the committed access rate (CAR) by issuing the command:

```
rate-limit input access-group 107 1000000 512000 512000 conform-action transmit exceed-
action drop
```

The **rate-limit** command has the syntax **rate-limit** *rate bc bc+be*.

Issue the **show interface Ethernet0/1 rate-limit** command:

```
R1#show interface Ethernet0/1 rate-limit
Ethernet0/1
  Input
    matches: access-group 107
      params:  1000000 bps, 512000 limit, 512000 extended limit
```



```
conformed 0 packets, 0 bytes; action: transmit
exceeded 0 packets, 0 bytes; action: drop
last packet: 171952988ms ago, current burst: 0 bytes
last cleared 1d23h ago, conformed 0 bps, exceeded 0 bps
```

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

12. Multicast

Issue: Enable multicast routing between routers R1, R2, and R3. Enable a multicast routing protocol that will use any unicast routing protocol for source address determination and that is also based on a shared tree.

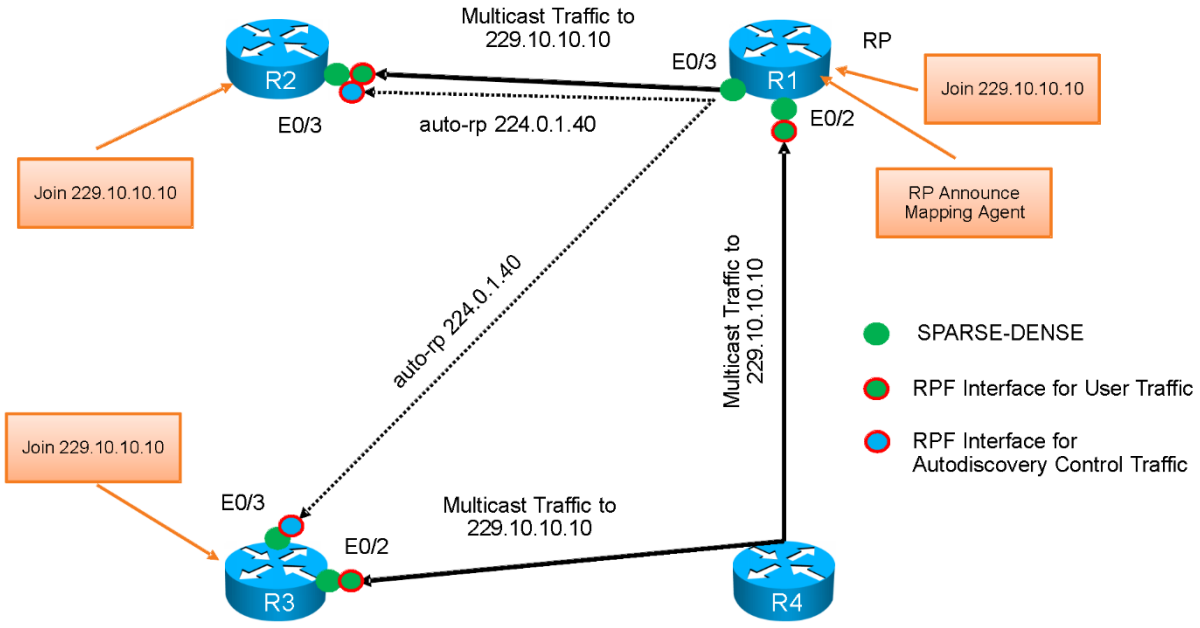
Solution:

PIM is a protocol that uses any unicast routing protocol for source address determination. Now you must determine which version of PIM: sparse mode, dense mode or sparse-dense mode. The answer lies in the phrase "based on a shared tree." PIM sparse mode is based on a shared tree. At the root of the shared tree is the RP of a given multicast-group. Therefore, you need to configure PIM sparse-dense mode to fulfill the configuration requirement. Use the **ip pim sparse-dense-mode** command on each required interface. Remember to enable multicast routing by issuing the command **ip multicast-routing** in global configuration mode before configuring PIM or Internet Group Management Protocol (IGMP).

You can use the command **show ip pim neighbors** to verify your PIM configuration. See the diagram that follows. Here is an example on R1:

```
R1#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
170.18.134.3      Ethernet0/2              06:02:14/00:01:32 v2    1 / DR S P G
170.18.123.3      Ethernet0/3              06:02:14/00:01:19 v2    1 / DR S P G
170.18.123.2      Ethernet0/3              06:02:56/00:01:34 v2    1 / S P G
R1#
```

Multicast Diagram



Issue: Configure all of the above-listed routers to join multicast group 229.10.10.10. Associate this multicast group with a loopback interface on each router.

Solution:

Configure the **ip igmp join-group x.x.x.x** command under a loopback interface on the listed routers. In addition, remember to configure **ip pim sparse-dense mode** on the same loopbacks to make them respond to pings.

Issue: Use the 224.0.1.39 PIM dense group for this configuration. Make R1 the root of the shared tree using the loopback 101 interface. Accomplish this task by configuring only R1.

Solution:

Because you have been instructed to use 224.0.1.39, you must configure the Auto-Rendezvous Point (Auto-RP). The Auto-RP uses the reserved multicast groups 224.0.1.39 and 224.0.1.40. When you configure the Auto-RP, you must configure not only a candidate RP, but also a mapping agent. Both the RP announcer and the mapping agent can be configured on the same router or different routers. Use the following commands on R1:

```
ip pim send-rp-announce Loopback101 scope 2
ip pim send-rp-discovery Loopback101 scope 2
```

When configuring Auto-RP, you must either specify sparse mode using the **ip pim sparse-mode** command and configure the Auto-RP listener feature using the **ip pim autorp listener** command, or specify sparse-dense mode using the **ip pim sparse-dense mode** command. Either

method can be used in this scenario. Configurations that are provided in the Mentor Guide engine use the second method of configuration.

Refer to this link for configuration information:

http://www.cisco.com/en/US/docs/ios-xml/ios/ipmulti/config_library/15-mt/imc-15-mt-library.html#GUID-FADD0C42-C50B-4F36-B520-ACD2B85E69C0

To verify that all routers have learned the RP address, use the command **show ip pim rp** [**mapping**]:

```
R1#sh ip pim rp mapping
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)
This system is an RP-mapping agent (Loopback101)

Group(s) 224.0.0.0/4
  RP 170.18.101.1 (?), v2v1
    Info source: 170.18.101.1 (?), elected via Auto-RP
         Uptime: 00:40:12, expires: 00:02:43
R1#

R2#show ip pim rp mapping
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 170.18.101.1 (?), v2v1
    Info source: 170.18.101.1 (?), elected via Auto-RP
         Uptime: 00:39:33, expires: 00:02:09
R2#

R3# show ip pim rp mapping
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 170.18.101.1 (?), v2v1
    Info source: 170.18.101.1 (?), elected via Auto-RP
         Uptime: 00:37:54, expires: 00:02:50
R3#
```

Issue: Ping the multicast group 229.10.10.10 from R4 to all other multicast routers.

Solution:

The requirements of this scenario do not define whether R1, R3, or both need to have PIM enabled on subnet 170.18.134.0/24. At least one of these routers must have PIM enabled on the interface to forward multicast packets that are generated by R4.

- If PIM is enabled on both R1 and R3, then R1 will receive packets directly from R4 and these same packets will be forwarded via R3.

Here are multicast configuration examples on R1, R2, and R3:

R1:

```
ip multicast-routing
!
interface Loopback101
 ip address 170.18.101.1 255.255.255.0
 ip pim sparse-dense-mode
 ip igmp join-group 229.10.10.10
!
interface Ethernet0/3
 ip address 170.18.123.1 255.255.255.0
```

```
 ip pim sparse-dense-mode

!
interface Ethernet0/2
 ip address 170.18.134.1 255.255.255.0
 ip pim sparse-dense-mode
!
ip pim send-rp-announce Loopback101 scope 2
ip pim send-rp-discovery Loopback101 scope 2
```

R2:

```
ip multicast-routing
!
interface Loopback102
 ip address 170.18.102.1 255.255.255.0
 ip pim sparse-dense-mode
 ip igmp join-group 229.10.10.10
!
interface Ethernet0/3
 ip address 170.18.123.2 255.255.255.0
 ip pim sparse-dense-mode
!
```

R3:

```
ip multicast-routing
!
interface Loopback103
 ip address 170.18.103.1 255.255.255.0
 ip pim sparse-dense-mode
 ip igmp join-group 229.10.10.10
!
interface Ethernet0/3
 ip address 170.18.123.3 255.255.255.0
 ip pim sparse-dense-mode
!
interface Ethernet0/2
 ip address 170.18.134.3 255.255.255.0
ip pim sparse-dense-mode
!
!
```

Here is the verification of multicast routes (with ping active on R4):

```
R4#ping 229.10.10.10 rep 10000
Type escape sequence to abort.
Sending 10000, 100-byte ICMP Echos to 229.10.10.10, timeout is 2 seconds:

Reply to request 0 from 170.18.101.1, 1 ms
Reply to request 0 from 170.18.102.1, 1 ms
Reply to request 0 from 170.18.103.1, 1 ms
Reply to request 1 from 170.18.103.1, 1 ms
Reply to request 1 from 170.18.102.1, 1 ms
Reply to request 1 from 170.18.101.1, 1 ms
Reply to request 2 from 170.18.103.1, 1 ms


R1#show ip mroute | begin Interface
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 229.10.10.10), 00:53:37/00:02:37, RP 170.18.101.1, flags: SJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/2, Forward/Sparse-Dense, 00:26:33/00:02:37
    Ethernet0/3, Forward/Sparse-Dense, 00:46:41/00:02:31
    Loopback101, Forward/Sparse-Dense, 00:53:37/00:02:29

(170.18.134.4, 229.10.10.10), 00:43:27/00:02:29, flags: LT
```



```
  Incoming interface: Ethernet0/2, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback101, Forward/Sparse-Dense, 00:43:27/00:02:29

(*, 224.0.1.39), 00:53:37/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback101, Forward/Sparse-Dense, 00:48:39/stopped
    Ethernet0/2, Forward/Sparse-Dense, 00:48:39/stopped
    Ethernet0/3, Forward/Sparse-Dense, 00:53:37/stopped

(170.18.101.1, 224.0.1.39), 00:48:54/00:02:04, flags: LT
  Incoming interface: Loopback101, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/2, Forward/Sparse-Dense, 00:48:39/stopped
    Ethernet0/3, Forward/Sparse-Dense, 00:48:39/stopped, A

(*, 224.0.1.40), 00:53:37/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/2, Forward/Sparse-Dense, 00:26:33/stopped
    Loopback101, Forward/Sparse-Dense, 00:53:37/stopped
    Ethernet0/3, Forward/Sparse-Dense, 00:53:08/stopped

(170.18.101.1, 224.0.1.40), 00:48:38/00:02:59, flags: LT
  Incoming interface: Loopback101, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/2, Forward/Sparse-Dense, 00:26:33/stopped
    Ethernet0/3, Forward/Sparse-Dense, 00:48:38/stopped, A

R1#


R2#show ip mroute | begin Interface
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 229.10.10.10), 00:30:06/stopped, RP 170.18.101.1, flags: SJCL
  Incoming interface: Ethernet0/3, RPF nbr 170.18.123.1
  Outgoing interface list:
    Loopback102, Forward/Sparse-Dense, 00:30:06/00:02:14

(170.18.134.4, 229.10.10.10), 00:30:05/00:02:51, flags: LJT
  Incoming interface: Ethernet0/3, RPF nbr 170.18.123.3*
  Outgoing interface list:
    Loopback102, Forward/Sparse-Dense, 00:30:05/00:02:14

(*, 224.0.1.39), 00:29:19/stopped, RP 0.0.0.0, flags: DC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/3, Forward/Sparse-Dense, 00:29:19/stopped

(170.18.101.1, 224.0.1.39), 00:28:39/00:02:08, flags: PT
  Incoming interface: Ethernet0/3, RPF nbr 170.18.123.1*
  Outgoing interface list: Null

(*, 224.0.1.40), 00:30:06/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/3, Forward/Sparse-Dense, 00:30:06/stopped
    Loopback102, Forward/Sparse-Dense, 00:30:06/stopped

(170.18.101.1, 224.0.1.40), 00:29:35/00:02:14, flags: LT
  Incoming interface: Ethernet0/3, RPF nbr 170.18.123.1*
```

```
  Outgoing interface list:
    Loopback102, Forward/Sparse-Dense, 00:29:35/stopped

R2#

R3#show ip mroute | begin Interface
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 229.10.10.10), 00:30:39/stopped, RP 170.18.101.1, flags: SJCLF
  Incoming interface: Ethernet0/2, RPF nbr 170.18.134.1
  Outgoing interface list:
    Loopback103, Forward/Sparse-Dense, 00:30:39/00:01:59

(170.18.134.4, 229.10.10.10), 00:30:38/00:01:51, flags: LFT
  Incoming interface: Ethernet0/2, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback103, Forward/Sparse-Dense, 00:30:38/00:01:59
    Ethernet0/3, Forward/Sparse-Dense, 00:30:28/00:02:32, A

(*, 224.0.1.39), 00:30:35/stopped, RP 0.0.0.0, flags: DC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/2, Forward/Sparse-Dense, 00:27:40/stopped
    Ethernet0/3, Forward/Sparse-Dense, 00:30:35/stopped

(170.18.101.1, 224.0.1.39), 00:30:02/00:02:54, flags: PTX
  Incoming interface: Ethernet0/2, RPF nbr 170.18.134.1
  Outgoing interface list:
    Ethernet0/3, Prune/Sparse-Dense, 00:27:02/00:02:30

(*, 224.0.1.40), 00:30:39/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback103, Forward/Sparse-Dense, 00:30:39/stopped
    Ethernet0/2, Forward/Sparse-Dense, 00:27:40/stopped
    Ethernet0/3, Forward/Sparse-Dense, 00:30:39/stopped

(170.18.101.1, 224.0.1.40), 00:29:58/00:02:51, flags: LT
  Incoming interface: Ethernet0/2, RPF nbr 170.18.134.1
  Outgoing interface list:
    Ethernet0/3, Prune/Sparse-Dense, 00:26:59/00:02:33
    Loopback103, Forward/Sparse-Dense, 00:29:58/stopped

R3#
```

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

