SPADVROUTE

Deploying Cisco Service Provider Advanced Network Routing

Version 1.01

Lab Guide

Text Part Number: 97-3152-02

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Table of Contents

SPADVROUTE

Lab Guide

Overview

This guide presents the instructions and other information concerning the lab activities for this course. You can find the solutions in the lab activity Answer Key.

Outline

This guide includes these activities:

- Job Aids
- Lab 2-1: Implement BGP Route Reflectors
- Lab 3-1: Implement BGP Security Options
- Lab 3-2: Improve BGP Scalability
- Lab 4-1: Implement Layer 2 and Layer 3 Multicast
- Lab 5-1: Enable and Optimize PIM-SM
- Lab 5-2: Implement PIM-SM Enhancements
- Lab 5-3: Implement Rendezvous Point Distribution
- Lab 6-1: Implement a DHCPv6 Server with Prefix Delegation
- Lab 6-2: Implement IPv6 Multicasting
- Lab 6-3: Implement Tunnels for IPv6
- Tear-Out Section

Job Aids

These job aids are available to help you complete lab activities.

Pod Access Information

The instructor will provide you with the team and pod numbers as well as other team and pod access information. Write down the information in the table for future reference.

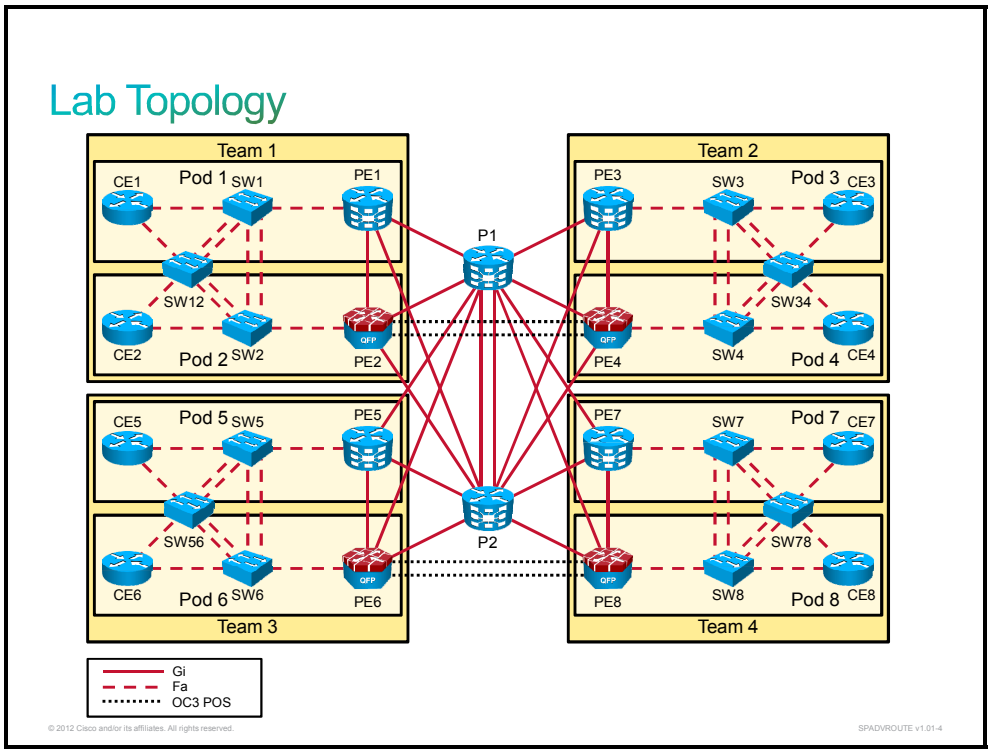
Parameter	Default Value	Value
Team number	z=1–4	
Pod number	x=1, 3, 5, or 7 or y=2, 4, 6, or 8	
Remote lab SSH access IP address	128.107.245.9	
Remote lab SSH access username	instr	
Remote lab SSH access password	testMe	
Pod PE (Cisco IOS XR Software) router username	root	
Pod PE (Cisco IOS XR Software) router password	1ronMan	
Pod CE, SW, and PE privileged-level password	cisco	

Device Information

This lab topology consists of four (4) teams and eight (8) pods. Two students will work in one pod, and two pods will work in one team. Each pod has one switch and two routers. Two pods share one additional switch. All teams share the same core routers (P1 and P2).

The CE routers in both pods are running Cisco IOS Software. The first pod within a team (pod 1, 3, 5, or 7) will work on the PE router that is running Cisco IOS XR Software, and the second pod within the same team (pod 2, 4, 6, or 8) will work on the PE router that is running Cisco IOS XE Software.

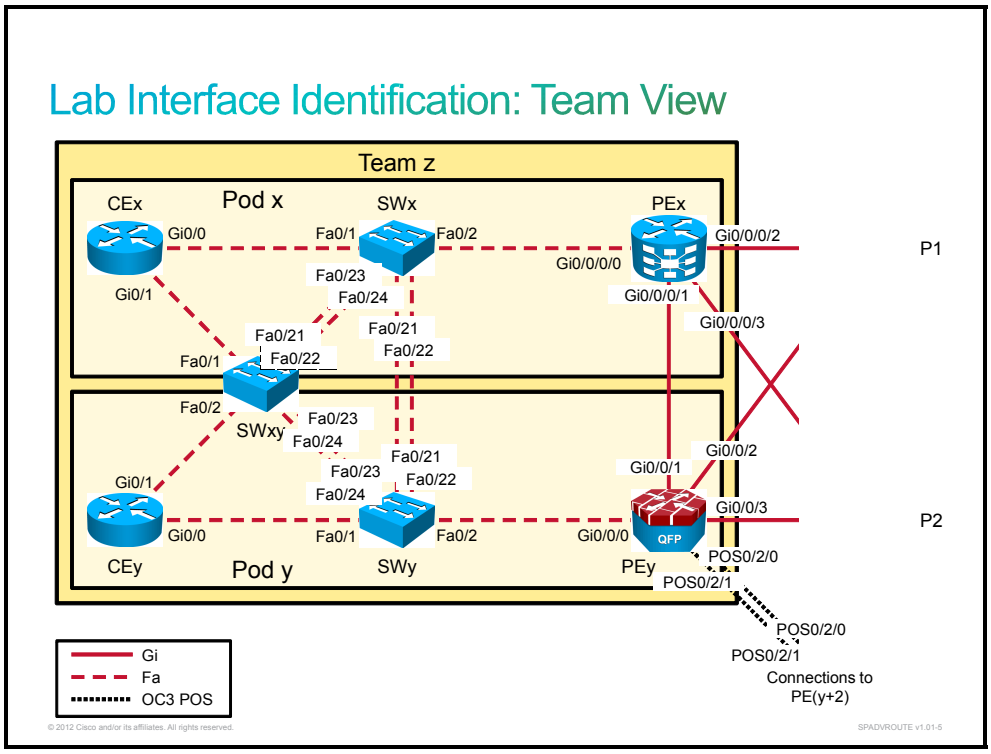
Devices in the lab are connected with Fast Ethernet and Gigabit Ethernet connections, and two teams have a redundant Packet-over-SONET/SDH (POS) connection, as shown in the figure.



Device Roles and Loopback IP Addresses

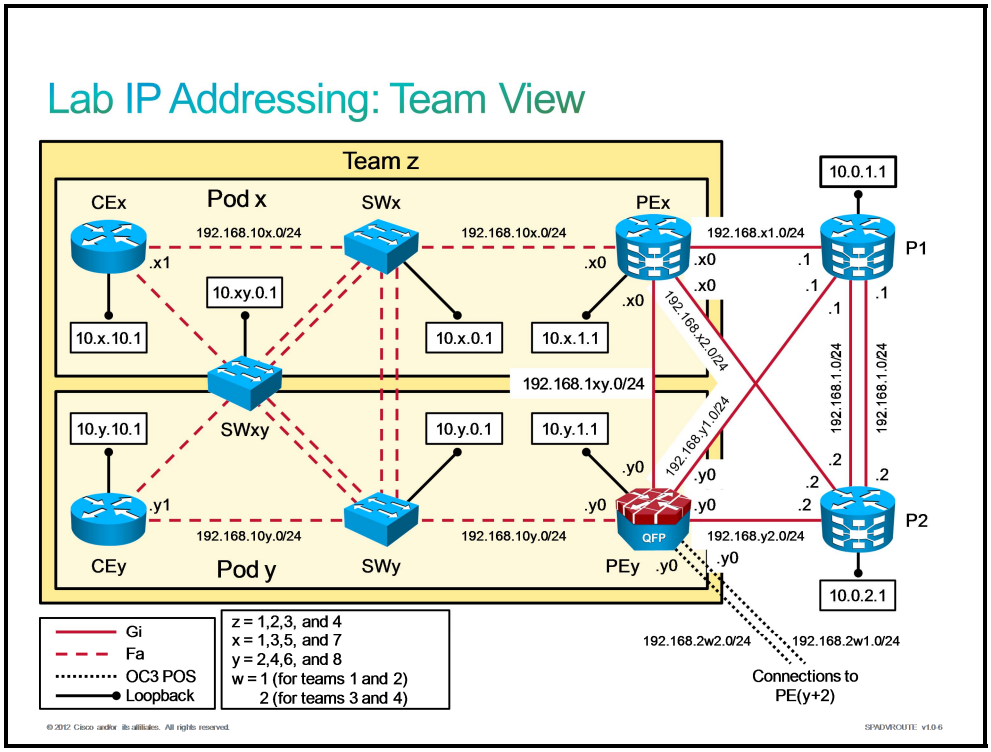
Device Name	Device Role	Lo0 IPv4 Address	Lo0 IPv6 Address
CEx CEy	Cisco 2900 pod router	10.x.10.1/32 10.y.10.1/32	2001:db8:10:x:10::1/128 2001:db8:10:y:10::1/128
PEx PEy	Cisco ASR 9000 or Cisco ASR 1000 pod router	10.x.1.1/32 10.y.1.1/32	2001:db8:10:x:1::1/128 2001:db8:10:y:1::1/128
SWx SWy	Cisco ME340x pod switch	10.x.0.1/32 10.y.0.1/32	2001:db8:10:x:0::1/128 2001:db8:10:y:0::1/128
SWxy	Cisco ME340x pod switch shared inside a team	10.xy.0.1/32	2001:db8:10:xy:0::1/128
P1	Cisco ASR 9000 core router	10.0.1.1/32	2001:db8:10:0:1::1/128
P2	Cisco ASR 9000 core router	10.0.2.1/32	2001:db8:10:0:2::1/128

The figure illustrates the interface identification that is used in this lab setup.

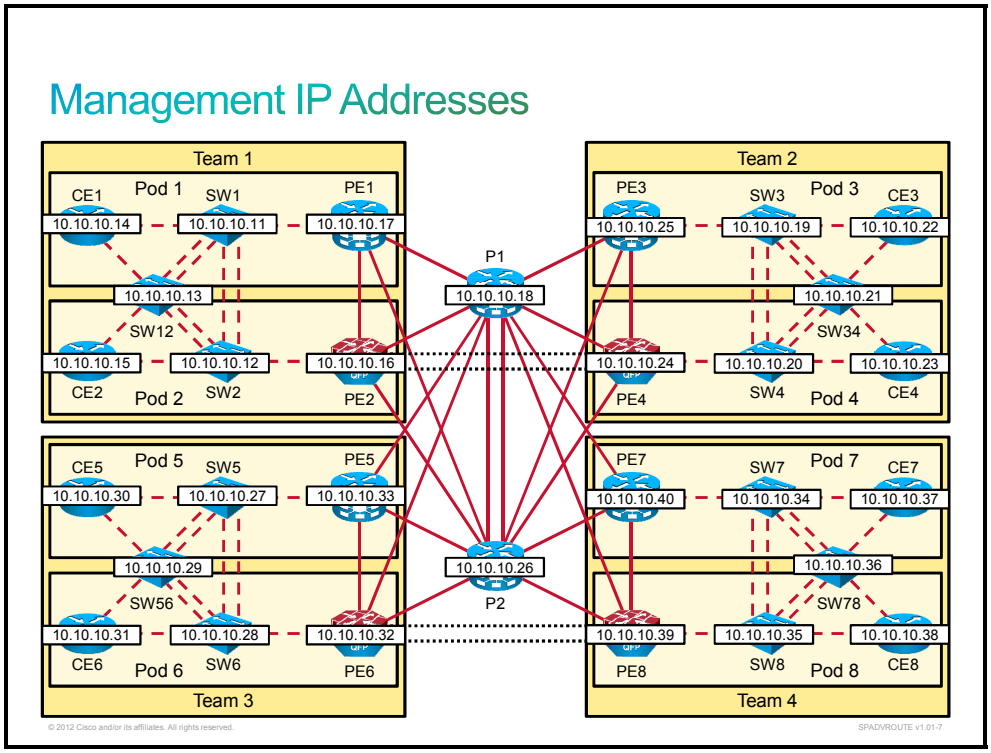


IP Addressing

The figure illustrates the IP addressing scheme that is used in this lab setup.



The figure illustrates the management IP addresses that are used in this lab setup.



Note Replace the x or y with your pod number to get the IP addresses within your pod (so x is for odd number pods 1, 3, 5, and 7; and y is for even number pods 2, 4, 6, and 8). Replace the xy (where x < y) with numbers of the pods within the same team (for example, 12, 34, 56, or 78) to get IP addresses on the link between those pods.

Pod IP Addressing

Device	Interface	IPv4 Address	IPv6 Address
CEx	Gi0/0	192.168.10x.x1/24	2001:db8:192:168:10x::x1/80
CEy	Gi0/0	192.168.10y.y1/24	2001:db8:192:168:10y::y1/80
P1		192.168.x1.1/24	2001:db8:192:168:x1::1/80
		192.168.y1.1/24	2001:db8:192:168:y1::1/80
P2		192.168.x2.2/24	2001:db8:192:168:x2::2/80
		192.168.y2.2/24	2001:db8:192:168:y2::2/80
PE2	POS0/2/0	192.168.211.20/24	2001:db8:192:168:211::20/80
	POS0/2/1	192.168.212.20/24	2001:db8:192:168:212::20/80
PE4	POS0/2/0	192.168.211.40/24	2001:db8:192:168:211::40/80
	POS0/2/1	192.168.212.40/24	2001:db8:192:168:212::40/80
PE6	POS0/2/0	192.168.221.60/24	2001:db8:192:168:221::60/80
	POS0/2/1	192.168.222.60/24	2001:db8:192:168:222::60/80
PE8	POS0/2/0	192.168.221.80/24	2001:db8:192:168:221::80/80
	POS0/2/1	192.168.222.80/24	2001:db8:192:168:222::80/80

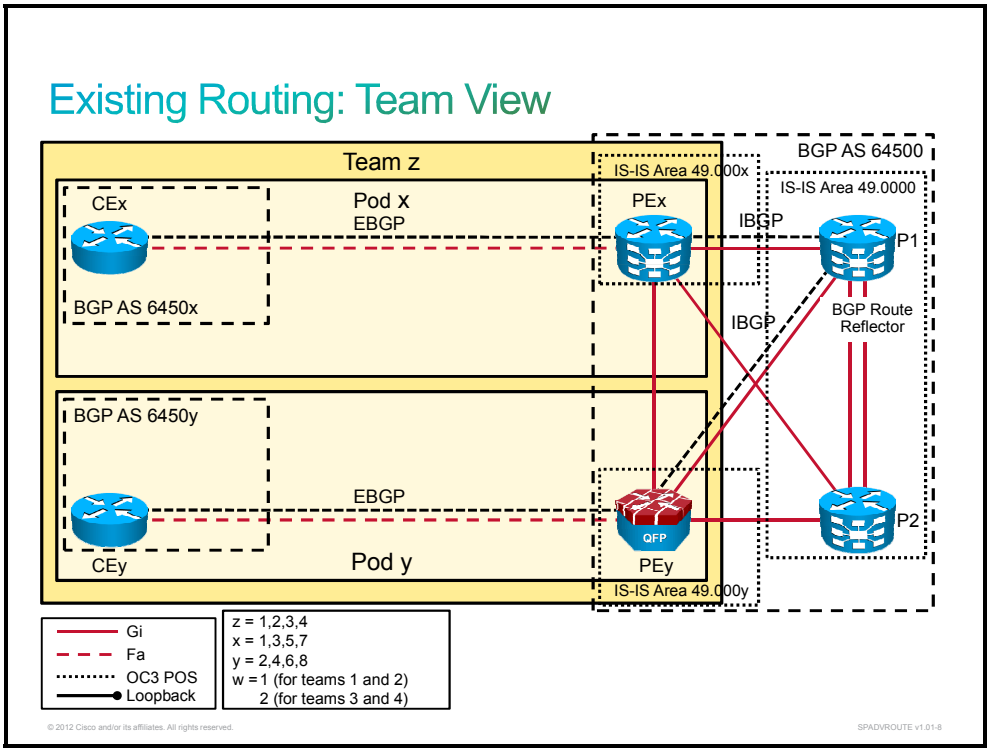
Device	Interface	IPv4 Address	IPv6 Address
PEx	Gi0/0/0/0	192.168.10x.x0/24	2001:db8:192:168:10x::x0/80
	Gi0/0/0/1	192.168.1xy.x0/24	2001:db8:192:168:1xy::x0/80
	Gi0/0/0/2	192.168.x1.x0/24	2001:db8:192:168:x1::x0/80
	Gi0/0/0/3	192.168.x2.x0/24	2001:db8:192:168:x2::x0/80
PEy	Gi0/0/0	192.168.10y.y0/24	2001:db8:192:168:10y::y0/80
	Gi0/0/1	192.168.1xy.y0/24	2001:db8:192:168:1xy::y0/80
	Gi0/0/2	192.168.y1.y0/24	2001:db8:192:168:y1::y0/80
	Gi0/0/3	192.168.y2.y0/24	2001:db8:192:168:y2::y0/80

Core IP Addressing

Device	Device IP Address	Peer	Peer IP Address
P1	192.168.1.1/24 2001:db8:192:168:1::1/80	P2	192.168.1.2/24 2001:db8:192:168:1::2/80
	192.168.2.1/24 2001:db8:192:168:2::1/80		192.168.2.2/24 2001:db8:192:168:2::2/80

Existing IP Routing

The figure illustrates the existing IP routing in the lab setup.



BGP AS Numbering

This subtopic includes a table with AS numbers that are used for BGP routing in the lab setup.

Pod and Backbone AS Numbers

Replace the “x” with your pod number to get the AS numbers.

Router	AS Number
P1	64500
P2	64500
CEx	6450x
PEx	6450x
CEy	6450y
PEy	6450y

NET Addressing

This subtopic gives a table with NET addresses that are used for IS-IS routing in the lab setup.

Pod and Backbone NET Addresses

Replace the “x” with your pod number to get the NET addresses for routers.

Router	NET Address
P1	49.0000.0100.0000.1001.00
P2	49.0000.0100.0000.2001.00
PEx	49.000x.0100.0x00.1001.00
PEy	49.000y.0100.0y00.1001.00

Lab 2-1: Implement BGP Route Reflectors

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will configure a backbone router (P2) to act as a redundant route reflector. You will also configure an additional IBGP session between the PE router in your pod and the redundant route reflector (P2). P1 already has been preconfigured by your instructor as a route reflector for the pod PE routers.

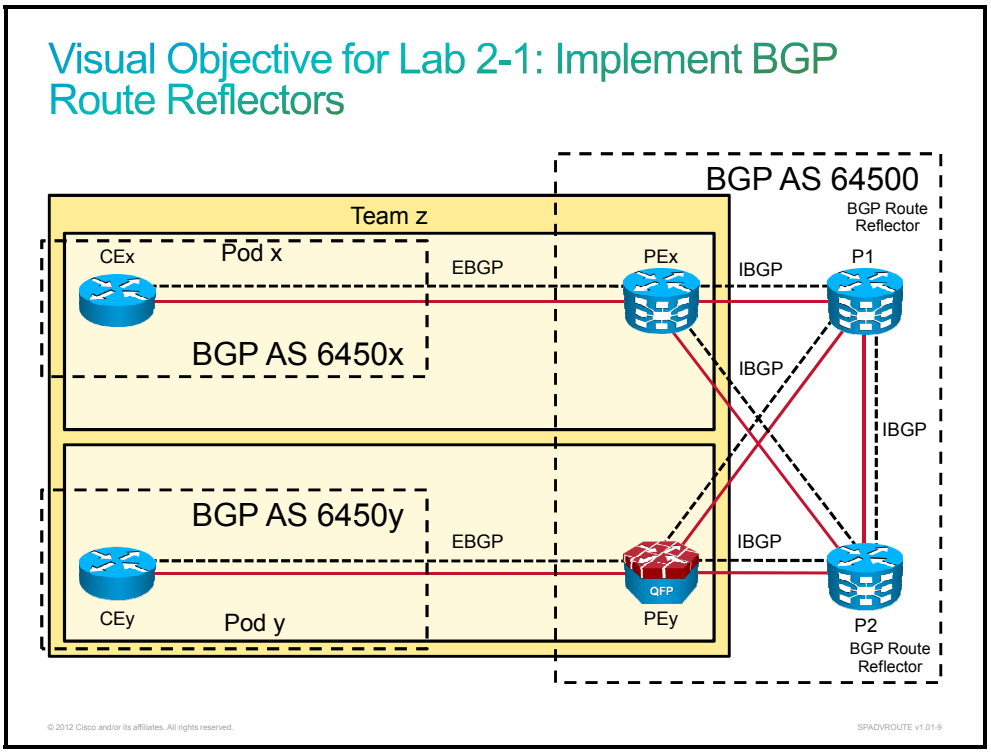
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router that is running Cisco IOS XR Software, and the second pod in the same team will work on the PE router that is running Cisco IOS XE Software. Students in the same team should coordinate their activities.

You will work on different Cisco routers that are running Cisco IOS (c2900), Cisco IOS XE (asr1001), and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Verify existing EBGP and IBGP sessions
- Configure a route reflector and IBGP session between a pod router and backbone router
- Restrict route propagation to a route reflector client

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Commands

Command	Description
[no] shutdown	Enables or disables the interface on the router
configure terminal	Enters configuration mode
interface *interface*	Enters interface configuration mode
ip address *ip_address subnet_mask*	Sets a primary or secondary IPv4 address for an interface and the subnet mask
ip|ipv6 router isis	Enables IS-IS routing to the interface
ipv6 address *ip_address/prefix*	Sets the IPv6 address for an interface
ipv6 enable	Enables IPv6 support on the interface
isis circuit-type level-1 | level-1-2 | level-2-only	Enables the IS-IS routing process to establish a selected IS-IS circuit type on the interface
neighbor *IP_address* **next-hop-self**	Enables the internal BGP router to send BGP routes with the router BGP IP address
neighbor *IP_address* **remote-as** *AS-number*	Configures the BGP neighbor
neighbor *IP_address* **update-source** *interface*	Enables the BGP router to send BGP packets from the source IP address
ping *dest_IP source source_IP*	Verifies connectivity between the source IP and destination IP
router bgp *AS-number*	Creates a BGP process and enters BGP process configuration mode
show ip bgp [*prefix***]**	Displays the BGP routing table
show ip bgp summary	Displays the BGP routing protocol characteristics, including BGP neighbor status
show ip interface brief	Displays the interface status and the IPv4 addresses that are configured
show isis neighbors	Displays the IS-IS neighbor information

Cisco IOS XR Commands

Command	Description
[no] shutdown	Enables or disables the interface on the router
address-family ipv4|ipv6 unicast	Enables IPv4 or IPv6 IS-IS or BGP routing and enters address family configuration mode for IS-IS or BGP (in router IS-IS or BGP configuration mode)
bgp cluster-id *cluster_ID*	Sets the router reflector cluster ID
circuit-type level-1 | level-1-2 | level-2-only	Enables the IS-IS routing process to establish the selected IS-IS circuit type on the interface
commit	Commits changes to the running configuration

Command	Description
configure terminal	Enters configuration mode
interface *interface* **(global)**	Enters interface configuration mode
interface *interface* **(router)**	Defines the interfaces on which the IS-IS protocol runs
ip address *ip_address mask*	Sets the IPv4 address for an interface
ipv6 address *ip_address/prefix*	Sets the IPv6 address for an interface
neighbor *IP-address*	Configures the BGP neighbor and enters BGP neighbor configuration mode
next-hop-self	Enables the internal BGP router to send BGP routes with the router BGP IP address (BGP neighbor address family mode)
pass	Passes the route for further processing (route-policy configuration mode)
ping *dest_IP source source_IP*	Verifies connectivity between the source IP and destination IP (IPv4 and IPv6)
remote-as *AS_number*	Configures the AS number for the BGP neighbor (BGP neighbor mode)
route-policy *route_policy_name*	Creates the route policy and enters route policy configuration mode
route-policy *route_policy_name* **in** | **out**	Applies the route policy to the BGP neighbor
router bgp *AS_number*	Creates a BGP process and enters BGP process configuration mode
router isis *process_ID*	Creates an IS-IS process
route-reflector-client	Configures an IBGP neighbor as the route reflector client
show bgp [*prefix*]	Displays the BGP routing table
show bgp summary	Displays the BGP routing protocol characteristics, including the BGP neighbor status
show ipv4 interface brief	Displays the interface status and the IPv4 addresses that are configured
show isis neighbors	Displays the IS-IS neighbor information
update-source *interface*	Enables the BGP router to send the BGP packets from the source IP address (BGP neighbor configuration mode)

Task 1: Verify the Existing BGP Sessions

In this task, you will verify that BGP is already configured and running in your pod, as preconfigured by your instructor. An IBGP session should be established between the PE router in your pod and the P1 backbone router. An EBGP session should be established between the PE and CE routers.

Activity Procedure

Complete these steps:

Step 1 Use Telnet to connect to the P1 router and examine the BGP configuration. You should see that the PE routers are configured as route reflector clients.

```
RP/0/RSP0/CPU0:P1#show running-config router bgp
router bgp 64500
 address-family ipv4 unicast
  redistribute static route-policy RTBH
 !
 address-family ipv6 unicast
 !
neighbor 10.1.1.1
  remote-as 64500
  update-source Loopback0
  address-family ipv4 unicast
   route-reflector-client
  !
 !
 neighbor 10.2.1.1
  remote-as 64500
  update-source Loopback0
  address-family ipv4 unicast
   route-reflector-client
```

Step 2 On the PE router, examine the EBGP and IBGP sessions configuration.

The PE router (Cisco IOS XR Software) output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show running-config router bgp
router bgp 64500
address-family ipv4 unicast
 !
 neighbor 10.0.1.1
  remote-as 64500
  update-source Loopback0
  address-family ipv4 unicast
   next-hop-self
  !
 !
 neighbor 192.168.101.11
  remote-as 64501
  address-family ipv4 unicast
   route-policy PASS in
   route-policy PASS out
```

The PE router (Cisco IOS Software) output should be similar to the following, taken from Pod 1:

```
PE6# show running-config | section router bgp
router bgp 64500
 bgp log-neighbor-changes
 neighbor 10.0.1.1 remote-as 64500
 neighbor 10.0.1.1 update-source Loopback0
 neighbor 10.0.1.1 next-hop-self
 neighbor 192.168.102.21 remote-as 64502
```

Step 3 Verify that the EBGP session is established between the PE and CE routers in your pod. In the "state/prefix received" column, you should see a number other than zero.

Step 4 Verify that IBGP session is established between the PE router in your pod and the P1 router. In the "state/prefix received" column, you should see a number other than zero.

Step 5 On the PE router in your pod, determine if there are any routes in the BGP table. You should see at least the route from the other pod in the team.

Step 6 On the PE router in your pod, verify the originator and cluster-ID list BGP attributes that have been inserted when the other pod route was reflected by the P1 router. The P1 router has been preconfigured by your instructor as a route reflector for the PEx and PEy routers in your team.

Activity Verification

You have completed this task when you attain these results:

- Verify that the EBGP session is established between the PE and CE routers in your pod. In the "state/prefix received" column, you should see a number other than zero. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          0 64500      30      26       30    0    0 00:06:23           2
192.168.101.11    0 64501   13988   12724       30    0    0    4d21h           1
```

- Verify that the IBGP session is established between the PE router in your pod and the P1 router. In the "state/prefix received" column, you should see a number other than zero. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          0 64500      30      26       30    0    0 00:06:23           2
192.168.101.11    0 64501   13988   12724       30    0    0    4d21h           1
```

- On the PE router in your pod, determine if there are any routes in the BGP table. You should see at least the route from the other pod in the team:

```
RP/0/RSP0/CPU0:PE1# show bgp
Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
*> 10.1.10.0/32       192.168.101.11           0             0 64501 i
*>i10.2.10.1/32       10.2.1.1                 0    100      0 64502 i
<…output omitted…>
```


Step 7 On the PE router, verify the originator and cluster-ID list BGP attributes that have been inserted when the other pod route was reflected by the P1 router. The P1 router has been preconfigured by your instructor as a route reflector for the PEx and PEy routers in your team:

```
RP/0/RSP0/CPU0:PE1# show bgp 10.2.10.1/32
<…output omitted…>
Paths: (1 available, best #1)
  Advertised to peers (in unique update groups):
    192.168.101.11
  Path #1: Received by speaker 0
  Advertised to peers (in unique update groups):
    192.168.101.11
  64502
    10.2.1.1 (metric 2) from 10.0.1.1 (10.2.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best, group-best
      Received Path ID 0, Local Path ID 1, version 30
      Originator: 10.2.1.1, Cluster list: 10.0.1.1
```

Task 2: Configure a Route Reflector and Internal BGP Session

In this task, you will configure the P2 router as a redundant route reflector. You will also configure internal BGP between the PE router in your pod and the P2 router in the backbone. Additionally, you will enable the BGP next-hop-self feature on the PE router.

Activity Procedure

Complete these steps:

Step 1 IS-IS is used as the IGP in AS 64500. On the PE router in your pod, enable the interface that connects to the P2 router and enable IS-IS level-2 routing. IS-IS has been preconfigured by your instructor and is enabled on loopback interfaces and on links between the CE, PE, and P1 routers. Enable the interface and routing for both IPv4 and IPv6.

Step 2 On the PE router in your pod, configure the internal BGP neighbor. The internal BGP neighbor is the P2 router in the backbone with IP address 10.0.2.1. The PE router in your pod should source BGP packets from the Loopback0 IP address.

Step 3 Configure the PE router in your pod to set the Loopback0 interface IP address for all BGP routes that are sent to the P2 IBGP neighbor (BGP next-hop-self).

Step 4 Configure the P2 router as an internal BGP neighbor to the PE router in your pod. Configure the P2 router as a route reflector for the PE router in your pod. Make sure that the cluster ID on the P2 router is set to the same value as on the P1 router. Coordinate this step with other teams if necessary.

Step 5 Verify the IBGP sessions on the PE router in your pod. You should see P2 as the IBGP neighbor.

Step 6 Verify the BGP table on the PE router. You should see the other pod route that is accessible over two paths. Note that the next hop is in both cases the originating router, the PE router in the other pod.

Note Recall that the route reflector does not change the next-hop IP address when a route is reflected.

Step 7 Examine the details about the other pod route. You should see that this route actually originated from P1 and P2, respectively.

Step 8 Answer the following question:

Which BGP path selection criterion is being used to select the path to the other pod route? _____

Activity Verification

You have completed this task when you attain these results:

- On the PE router, you should see that the interface toward the P2 router is configured and running:

```
RP/0/RSP0/CPU0:PE1#show ipv4 interface brief | include Up
Loopback0                      10.1.1.1        Up                  Up
MgmtEth0/RSP0/CPU0/0           10.10.10.33     Up                  Up
GigabitEthernet0/0/0/0         192.168.101.10  Up                  Up
GigabitEthernet0/0/0/1         192.168.112.10  Up                  Up
GigabitEthernet0/0/0/2         192.168.11.10   Up                  Up
GigabitEthernet0/0/0/3         192.168.12.10   Up                  Up
```

- On the PE router, you should see the P2 router as an IS-IS neighbor:

```
RP/0/RSP0/CPU0:PE1#show isis neighbors
IS-IS 1 neighbors:
System Id      Interface        SNPA           State Holdtime Type IETF-NSF
CE1            Gi0/0/0/0        e8b7.482c.a180 Up    7        L1   Capable
P1             Gi0/0/0/2        4055.392e.d822 Up    7        L2   Capable
P2             Gi0/0/0/3        4055.392f.42dc Up    8        L2   Capable
PE2            Gi0/0/0/1        e8b7.48fb.5801 Up    8        L2   Capable

Total neighbor count: 4
```

- You should see the P2 as an IBGP neighbor:

```
RP/0/RSP0/CPU0:PE1# show bgp summary
<…output omitted…>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          0 64500     177     172       32    0    0 02:32:01           2
10.0.2.1          0 64500       7       6       32    0    0 00:02:13           2
192.168.101.11    0 64501   14149   12870       32    0    0    5d00h           1
```

- You should see the other pod route as accessible over two paths:

```
RP/0/RSP0/CPU0:PE1# show bgp
<…output omitted…>
Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
*> 10.1.10.1/32       192.168.101.11           0             0 64501 i
*>i10.2.10.1/32       10.2.1.1                 0    100      0 64502 i
* i                   10.2.1.1                 0    100      0 64502 i
```

- You should see that the route actually originated from P1 and P2 respectively:

```
RP/0/RSP0/CPU0:PE1# show bgp 10.2.10.1/32
<…output omitted…>
Paths: (2 available, best #1)
  Advertised to peers (in unique update groups):
    192.168.101.11
  Path #1: Received by speaker 0
```



```
  Advertised to peers (in unique update groups):
    192.168.101.11
  64502
    10.2.1.1 (metric 2) from 10.0.1.1 (10.2.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best, group-best
      Received Path ID 0, Local Path ID 1, version 30
      Originator: 10.2.1.1, Cluster list: 10.0.1.1
  Path #2: Received by speaker 0
  Not advertised to any peer
  64502
    10.2.1.1 (metric 2) from 10.0.2.1 (10.2.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal
      Received Path ID 0, Local Path ID 0, version 0
      Originator: 10.2.1.1, Cluster list: 10.0.1.1
```

Task 3: (Optional) Restrict Route Propagation to a Client

In this optional task, you will restrict route propagation to route reflector clients on the P2 router. You will allow the P2 router to advertise only other pod routes to route reflector clients.

Activity Procedure

Complete these steps:

Step 1 On the P2 router, create a route policy that will allow only a route originating from the other pod AS (for example, for Pod 1, allow only routes that originate in AS 64502). Use **FILTER_TO_CLIENT_PODX** or **FILTER_TO_CLIENT_PODY** name as the route policy name.

Step 2 On the P2 router, apply the route policy to the IBGP session with the pod PE router in the outbound direction.

Note Verification of this task is possible only when at least one pod from the other team has a working BGP configuration.

Step 3 On the PE router, verify the BGP table. You should see the routes from pods from other teams that are available only over the P1 router (because the P2 router reflects only routes from the other pod).

Activity Verification

You have completed this task when you attain these results:

- You should see the routes from pods from other teams available only over the P1 router (because the P2 router reflects only routes from the other pod). Routers from the other pod should be available over two paths. Output should be similar to the following, taken from PE1 router:

```
RP/0/RSP0/CPU0:PE1# show bgp
<…output omitted…>
Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
*> 10.1.10.1/32       192.168.101.11           0             0 64501 i
*>i10.2.10.1/32       10.2.1.1                 0    100      0 64502 i
* i                   10.2.1.1                 0    100      0 64502 i
```

```
*>i10.5.10.1/32       10.5.1.1                 0    100      0 64505 i
<…output omitted…>

RP/0/RSP0/CPU0:PE1# show bgp 10.5.10.1/32
<…output omitted…>
Paths: (1 available, best #1)
  Advertised to peers (in unique update groups):
    192.168.101.11
  Path #1: Received by speaker 0
  Advertised to peers (in unique update groups):
    192.168.101.11
  64505
    10.5.1.1 (metric 3) from 10.0.1.1 (10.5.1.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best, group-best
      Received Path ID 0, Local Path ID 1, version 33
      Originator: 10.5.1.1, Cluster list: 10.0.1.1
```


Lab 3-1: Implement BGP Security Options

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will first configure BGP neighbor authentication between the CE, PE, and P1 routers. You will also enable a BGP TTL security check between the same routers. You will also configure CoPP on the CE router. Finally, you will implement source-based RTBH filtering by using the P1 router as an RTBH triggering router.

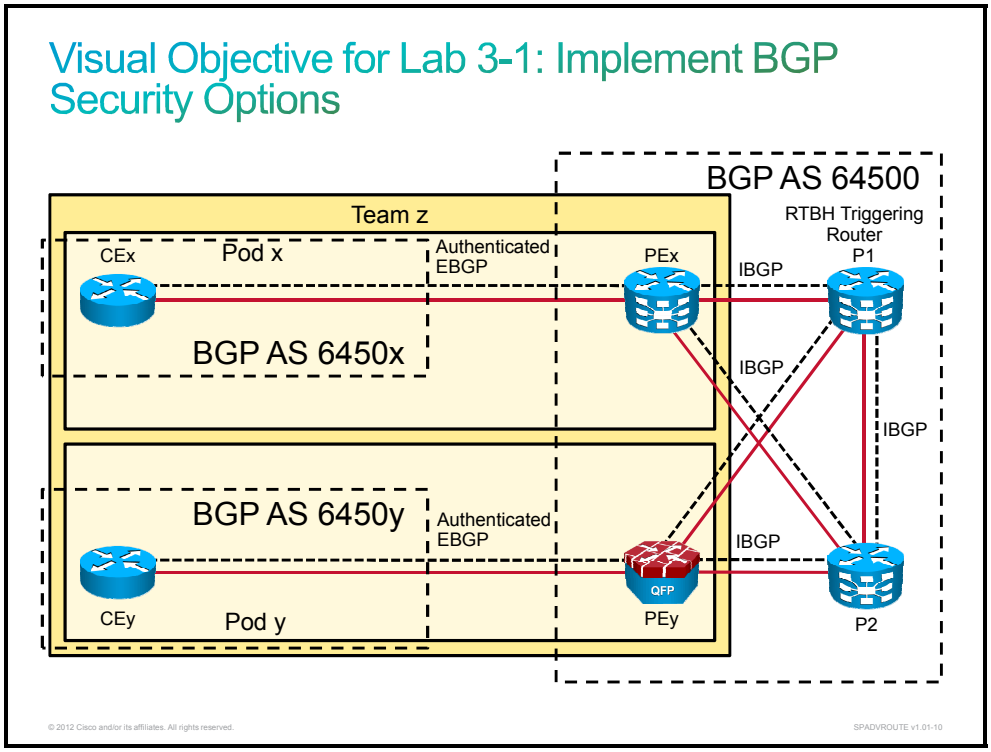
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router that is running Cisco IOS XR Software, and the second pod in the same team will work on the PE router that is running Cisco IOS XE Software. Students in the same team should coordinate their activities.

You will work on different Cisco routers that are running Cisco IOS (c2900), Cisco IOS XE (asr1001), and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Implement BGP neighbor authentication using a password
- Implement BGP neighbor authentication using key chains
- Enable the BGP TTL security check
- Enable CoPP
- Enable RTBH filtering

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Software Commands

Command	Description
class *class_map_name*	Specifies the name of the class whose policy you want to create or change
class-map *class_map_name*	Creates a class map
configure terminal	Enters configuration mode
control-plane	Enters control plane virtual interface configuration mode
interface *interface*	Enters interface configuration mode
ip access-list extended *acl_name*	Creates an extended access control list
ip address *ip_address subnet_mask*	Sets a primary or secondary IPv4 address for an interface and the subnet mask
ip route *network subnet_mask outgoing_interface*	Creates a static route
ip verify unicast source reachable-via rx	Enables strict uRPF on an interface
match access-group name *acl_name*	Specifies ACL matching criteria inside a class map
neighbor ip_address **password** *password*	Enables BGP neighbor authentication
neighbor ip_address **ttl-security hops** *hops_number*	Enables a BGP TTL security check
network *network* **mask** *mask*	Advertises the network through BGP
permit *protocol source [operator] [port] destination [operator] [port]*	Creates a permit ACL entry
ping *dest_ip_source source_interface*	Verifies connectivity between the source IP and destination IP
police rate *pps* **pps conform-action transmit exceed-action drop**	Configures traffic policing
policy-map *policy_map_name*	Creates a policy map
router bgp *AS-number*	Creates a BGP process and enters the process configuration mode
service-policy input *policy_map_name*	Applies a policy map to an interface in the inbound direction
show access-lists	Displays BGP neighbor information
show class-map	Displays BGP neighbor information

show ip bgp [*prefix*]	Displays the BGP routing table
show ip bgp neighbors *ip_address*	Displays BGP neighbor information
show policy-map control-plane	Displays BGP neighbor information

Cisco IOS XR Software Commands

Command	Description
accept-lifetime *start_time end_time*	Specifies key accept validity
address-family ipv4|**ipv6 unicast**	Enters address family configuration mode
commit	Commits changes to the running configuration
configure terminal	Enters configuration mode
cryptographic-algorithm *algorithm*	Specifies the cryptographic algorithm for a key
interface *interface*	Enters interface configuration mode
ipv4 verify unicast source reachable-via rx	Enables strict uRPF on an interface
key chain *keychain_name*	Creates a key chain
key *key_id*	Specifies a key ID
keychain *keychain_name*	Enables BGP neighbor authentication using a key chain
neighbor *IP-address*	Configures the BGP neighbor and enters BGP neighbor configuration mode
network/prefix outgoing_interface **[tag** *tag***]**	Configures the static route
password *password*	Enables BGP neighbor authentication by using a password
ping *dest_IP source source_IP*	Verifies connectivity between the source IP and destination IP (IPv4 and IPv6)
router bgp *AS-number*	Creates a BGP process and enters the BGP process configuration mode
router static	Enters static route configuration mode
send-lifetime *start_time end_time*	Specifies a key accept validity
show bgp [*prefix*]	Displays the BGP routing table
show bgp summary	Displays the BGP routing protocol characteristics, including the BGP neighbor status
show lpts flows	Displays information about Local Packet Transport Services (LPTS) flows
ttl-security	Enables BGP TTL security check

Task 1: Implement BGP Neighbor Authentication Using Passwords

In this task, you will implement BGP neighbor authentication on the EBGP session between the CE and PE routers.

Activity Procedure

Complete these steps:

Step 1 Verify that the EBGP session is established between the PE and CE routers in your pod. In the "state/prefix received" column, you should see a number other than zero.

Step 2 Enable BGP neighbor authentication on the CE router on the EBGP session with the PE router. Use **C!sc()** as a password. Clear the BGP session because it will not be torn down automatically. Observe the CE router console. You should see that the CE router does not receive the MD5 hash from the PE router:

```
CE1#
*Oct  4 13:31:05.686: %TCP-6-BADAUTH: No MD5 digest from 192.168.101.10(28585)
to 192.168.101.11(179)
```

Step 3 Enable BGP neighbor authentication on the PE router.

Step 4 Verify that the EBGP session is established between the PE and CE routers in your pod.

Activity Verification

You have completed this task when you attain these results:

- Verify that an EBGP session is established between the PE and CE routers in your pod. In the "state/prefix received" column, you should see a number other than zero. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor      Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1        0 64500      30      26       30    0    0 00:06:23          2
10.0.2.1        0 64500    8747    8746      116    0    0    6d01h          1
192.168.101.11  0 64501   13988   12724       30    0    0    4d21h          1
```

- Verify that the EBGP session is established after authentication configuration between the PE and CE routers in your pod. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor      Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1        0 64500    8966    8949      118    0    0    4d00h          5
10.0.2.1        0 64500    8776    8775      118    0    0    6d02h          1
192.168.101.11  0 64501   23796   21641      118    0    0 00:03:48          1
```


Task 2: Implement BGP Neighbor Authentication Using Key Chains

In this task, you will implement BGP neighbor authentication on the IBGP session between PE and P1 routers using key chains. BGP neighbor authentication using key chains is also called advanced BGP authentication and is not compatible with classic BGP authentication using passwords. This task applies to pods that are running the Cisco IOS XR Software router as the PE router only.

Activity Procedure

Complete these steps:

Step 1 Verify that the IBGP session is established between the PE and P1 routers in your pod. In the "state/prefix received" column, you should see a number other than zero.

Step 2 Configure a key chain on the PE router. Use **C!sc()** as a key string and **HMAC-MD5** as a hashing algorithm. Specify a valid send and accept lifetime as well; otherwise, the key will not be valid.

Step 3 Apply the key chain on the PE router to the IBGP session with the P1 router. You should see that the PE router received packets with invalid authentication:

```
RP/0/RSP0/CPU0:Oct  4 13:54:57.539 : tcp[395]: %IP-TCP-3-BADAUTH : Invalid EA
digest from 10.0.1.1:16485 to 10.1.1.1:179
```

Step 4 Use Telnet to connect to the P1 router. Configure a key chain on the P1 router. Use a pod-specific name for the key chain, (for example, BGP_PODX or BGP_PODY). Use **C!sc()** as a key string and **HMAC-MD5** as a hashing algorithm. Specify a valid send and accept lifetime as well; otherwise, the key will not be valid.

Step 5 Apply the key chain on the P1 router to the IBGP session with the PE router in your pod. You should not see the message about the received packets with invalid authentication anymore.

Step 6 Verify that the IBGP session is still established between the PE and P1 routers in your pod. In the "state/prefix received" column, you should see a number other than zero.

Activity Verification

You have completed this task when you attain these results:

- Verify that an IBGP session is established between the PE and P1 routers in your pod. In the "state/prefix received" column, you should see a number other than zero. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          0 64500      30       26       30    0    0 00:06:23           2
10.0.2.1          0 64500    8747     8746      116    0    0    6d01h           1
192.168.101.11    0 64501   13988    12724       30    0    0    4d21h           1
```

- Verify that the IBGP session is still established between the PE and P1 routers in your pod. In the " state/prefix received" column, you should see a number other than zero. The he PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          0 64500      30      26       30    0    0 00:09:23          2
10.0.2.1          0 64500    8747    8746      116    0    0    6d01h          1
192.168.101.11    0 64501   13988   12724       30    0    0    4d21h          1
```

Task 3: Enable BGP TTL Security Check

In this task, you will enable a BGP TTL security check on the EBGP session between the CE and PE routers.

Activity Procedure

Complete these steps:

Step 1 Enable TTL security check for the EBGP session on the CE router in your pod. What is the number that you have to specify with the command to enforce that EBGP neighbors are directly connected?

_____

Step 2 Wait 3 minutes. After this, you should see that the EBGP session was torn down by the CE router:

```
CE1#
*Oct  5 09:08:12.329: %BGP-5-ADJCHANGE: neighbor 192.168.101.10 Down BGP
Notification sent
*Oct  5 09:08:12.329: %BGP-3-NOTIFICATION: sent to neighbor 192.168.101.10 4/0
(hold time expired) 0 bytes
*Oct  5 09:08:12.329: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.101.10 IPv4
Unicast topology base removed from session
*Oct  5 09:11:21.805: %BGP-3-BGP_NO_REMOTE_READ: 192.168.101.10 connection
timed out - has not accepted a message from us for 180000ms (hold time), 0
messages pending transmition.
*Oct  5 09:11:21.805: %BGP-3-NOTIFICATION: sent to neighbor 192.168.101.10
active 4/0 (hold time expired) 0 bytes
*Oct  5 09:11:21.805: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.101.10 IPv4
Unicast topology base removed from session  BGP Notification sent
```

Step 3 On the PE router, enable the TTL security check. You should see that the EBGP session went up:

```
CE1#
*Oct  5 09:17:47.757: %BGP-5-ADJCHANGE: neighbor 192.168.101.10 Up
```

Step 4 On the PE router, verify the expected TTL value for BGP packets from the CE neighbor.

Step 5 On the CE router, verify incoming and outgoing TTL settings for the PE neighbor.

Activity Verification

You have completed this task when you attain these results:

- On the CE router, verify incoming and outgoing TTL settings for the neighbor PE. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show ip bgp neighbors 192.168.101.10 | include TTL
Connection is ECN Disabled, Mininum incoming TTL 254, Outgoing TTL 255
```

- On the PE router, verify the expected TTL value for the BGP packets from the CE neighbor. On the PE router, verify the expected TTL value for the BGP packets from the CE neighbor. On the PE1 (Cisco IOS XR Software) router, the output should be similar to the following::

```
RP/0/RSP0/CPU0:PE1#show lpts flows | begin BGP
<…output omitted…>
L3-proto    : IPV4(2)
L4-proto    : TCP(6)
VRF-ID      : default (0x60000000)
Local-IP    : any
Remote-IP   : 192.168.105.51
Local-Port  : 179
Remote-Port : any
Interface   : any (0x0)
Flow-type   : BGP-cfg-peer
Min-TTL     : 255
Slice       : BGP4_FM
Flags       : 0x8 (in Pre-IFIB)
Location    : 0/RSP0/CPU0
Element References
location / count / scope
0/RSP0/CPU0 / 1 / LR
<…output omitted…>
```

- On the PE2 (Cisco IOS XE Software) router, the output should be similar to the following:

```
PE2#show ip bgp neighbors 192.168.102.21 | include TTL
Connection is ECN Disabled, Mininum incoming TTL 254, Outgoing TTL 255
```

- On the CE router, verify incoming and outgoing TTL settings for the neighbor PE. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show ip bgp neighbors 192.168.101.10 | include TTL
Connection is ECN Disabled, Mininum incoming TTL 254, Outgoing TTL 255
```

Task 4: (Optional) Enable CoPP

In this optional task, you will enable CoPP to rate-limit BGP traffic to the CE router.

Activity Procedure

Complete these steps:

Step 1 On the CE router, configure a named ACL that will permit BGP traffic from the PE to the CE router. Verify the configured ACL.

Step 2 On the CE router, create a class map that will refer to the previously configured ACL. Verify the configured class map.

Step 3 On the CE router, create a policy map that will rate-limit BGP traffic from the PE to the CE router to 200 packets per second.

Step 4 On the CE router, apply the configured policy map to the control plane virtual interface using the service policy. Verify the applied policy map.

Activity Verification

You have completed this task when you attain these results:

- Verify the configured ACL. The output should be similar to the following, taken from Pod 1:

```
CE1# show access-lists
Extended IP access list BGP_TRAFFIC
    10 permit tcp host 192.168.101.10 host 192.168.101.11 eq bgp
    20 permit tcp host 192.168.101.10 eq bgp host 192.168.101.11 (9 matches)
```

- Verify the configured class map. The output should be similar to the following, taken from Pod 1:

```
CE1#show class-map
 Class Map match-any class-default (id 0)
   Match any

 Class Map match-all BGP_CLASS (id 1)
   Match access-group name BGP_TRAFFIC
```

- Verify the applied policy map. The output should be similar to the following, taken from Pod 1:

```
CE1# show policy-map control-plane
<…output omitted…>
    Class-map: BGP_CLASS (match-all)
      11 packets, 953 bytes
      5 minute offered rate 0 bps, drop rate 0 bps
      Match: access-group name BGP_TRAFFIC
      police:
          rate 200 pps, burst 48 packets
        conformed 11 packets; actions:
          transmit
        exceeded 0 packets; actions:
          drop
        conformed 0 pps, exceed 0 pps

    Class-map: class-default (match-any)
      273 packets, 24331 bytes
      5 minute offered rate 1000 bps, drop rate 0 bps
      Match: any
```

Task 5: (Optional) Enable RTBH Filtering

In this optional task, you will enable source-based RTBH filtering.

Activity Procedure

Complete these steps:

Step 1 On the CE router, create the Loopback1 interface. Use 10.x.100.1/32 or 10.y.100.1/32 as the IP address on the interface. This interface will be used as a source of traffic that you would like to black-hole.

Step 2 On the CE router, advertise the previously configured /32 network on the loopback interface into BGP.

Step 3 From the CE router, ping the other pod CE router from the Loopback1 interface. You should be successful.

Step 4 On the PE router, create a static route for the 172.16.x.0/24 (or 172.16.y.0/24) network that points to the null0 interface.

Step 5 On the PE router, enable strict uRPF on the CE-facing interface. Use the Job Aids section to determine the interface.

Step 6 Use Telnet to connect to the P1 router. Create a static route for the 172.16.x.0/24 (or 172.16.y.0/24) network that points to the null0 interface.

Step 7 On the P1 router, examine the preconfigured route policy named RTBH:

```
RP/0/RSP0/CPU0:P1#show running-config route-policy RTBH
Wed Oct  5 12:33:27.153 UTC
route-policy RTBH
  if tag eq 1 then
    set next-hop 172.16.1.1
    set local-preference 1000
    set community (no-export)
  elseif tag eq 2 then
    set next-hop 172.16.2.1
    set local-preference 1000
    set community (no-export)
  elseif tag eq 3 then
    set next-hop 172.16.3.1
    set local-preference 1000
    set community (no-export)
  elseif tag eq 4 then
    set next-hop 172.16.4.1
    set local-preference 1000
    set community (no-export)
  elseif tag eq 5 then
    set next-hop 172.16.5.1
    set local-preference 1000
    set community (no-export)
  elseif tag eq 6 then
    set next-hop 172.16.6.1
    set local-preference 1000
    set community (no-export)
  elseif tag eq 7 then
    set next-hop 172.16.7.1
    set local-preference 1000
    set community (no-export)
  elseif tag eq 8 then
    set next-hop 172.16.8.1
    set local-preference 1000
    set community (no-export)
  else
    drop
  endif
end-policy
```

Step 8 On the P1 router, examine the BGP configuration. You should see that static routes are redistributed into BGP using the RTBH route policy as a filter:

```
RP/0/RSP0/CPU0:P1#show running-config router bgp
Wed Oct  5 12:35:49.014 UTC
router bgp 64500
 address-family ipv4 unicast
  redistribute static route-policy RTBH
<…output omitted…>
```

Step 9 Answer the following questions:

Why are the redistributed routes tagged with no-export community?

_____

Why is the local preference of redistributed routes set to 1000?

_____

Step 10 On the P1 router, trigger black-holing of traffic originating from the CE Loopback1 interface. Configure a static route for the Loopback1 interface that is tagged with x (or y) and points to the null0 interface.

Step 11 On the PE router, examine the BGP table. You should see an additional path to the 10.x.100.1/32 (or 10.y.100.1/32) network that points to the 172.16.x.1 (or 172.16.y.1) next hop.

Step 12 On the PE router, examine the detailed Cisco Express Forwarding entry for the 10.x.100.1/32 (or 10.y.100.1/32) network. You should see that the outgoing interface is null0.

Step 13 From the CE router, ping the other pod CE router. Use the Loopback1 interface as the source interface.

Activity Verification

You have completed this task when you attain these results:

- From the CE router, ping the other pod CE router from the Loopback1 interface. You should be successful. The output should be similar to the following, taken from Pod 1:

```
CE1#ping 10.2.10.1 source Loopback1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.6.10.1, timeout is 2 seconds:
Packet sent with a source address of 10.5.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

- On the PE router, examine the BGP table. You should see an additional path to the 10.x.100.1/32 (or 10.y.100.1/32) network that points to the 172.16.x.1 (or 172.16.y.1) next hop. The output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1# show bgp
<…output omitted…>
Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
*> 10.1.10.1/32       192.168.101.11           0             0 64501 i
```



```
*>i10.1.100.1/32      172.16.1.1               0   1000       0 ?
*                     192.168.101.11           0              0 64501 i
<…output omitted…>
```

- On the PE router, examine the detailed Cisco Express Forwarding entry for the 10.x.100.1/32 (or 10.y.100.1/32) network. You should see that the outgoing interface is null0. The output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show cef 10.1.100.1/32 detail
<…output omitted..>
    Hash  OK  Interface                 Address
    0     Y   recursive                 null0
```

- From the CE router, ping the other pod CE router. Use the Loopback1 interface as the source interface. The output should be similar to the following, taken from Pod 1:

```
CE1# ping 10.2.10.1 source Loopback1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.6.10.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.100.1
.....
Success rate is 0 percent (0/5)
```

Lab 3-2: Improve BGP Scalability

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this activity, you will first migrate an existing BGP configuration for EBGP neighbors to template-based configuration. Then you will limit the number of prefixes that can be received from a BGP neighbor. You will also improve BGP convergence by changing the BGP scan and advertisement interval, as well as improve BGP convergence by enabling BFD. Finally, you will implement BGP route dampening. All configurations will be performed on the PE router.

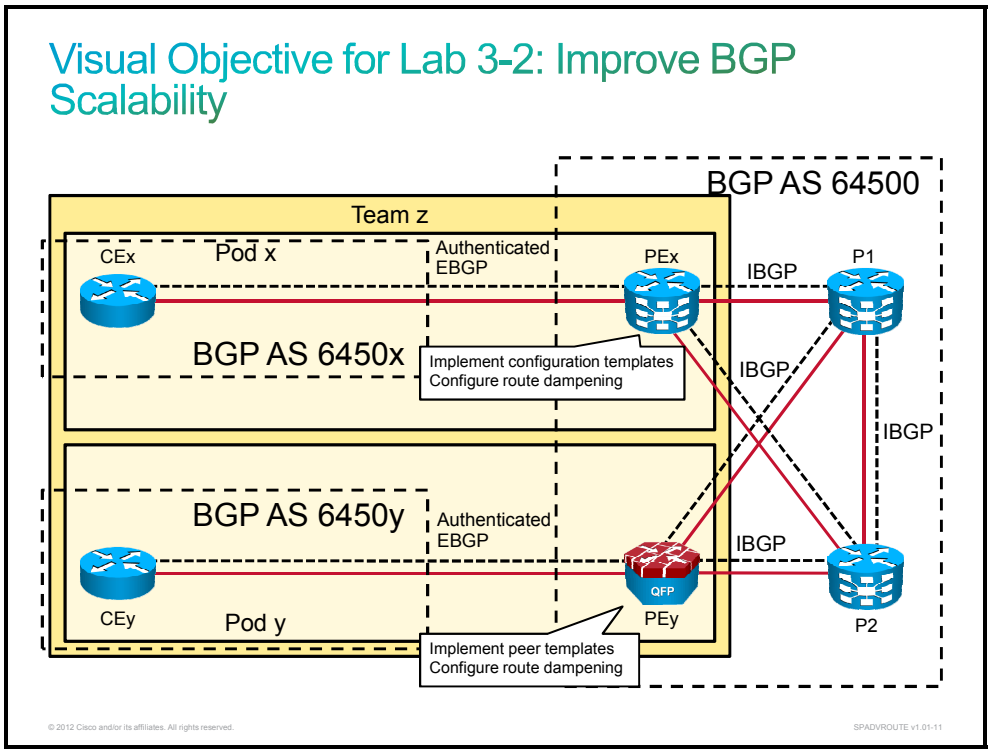
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router that is running Cisco IOS XR Software, and the second pod in the same team will work on the PE router that is running Cisco IOS XE Software. Students in the same team should coordinate their activities.

You will work on different Cisco routers that are running Cisco IOS (c2900), Cisco IOS XE (asr1001), and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Implement BGP configuration and peer templates
- Limit the number of prefixes that are received from a BGP neighbor
- Improve BGP convergence by changing the BGP scan and advertisement interval
- Improve BGP convergence by configuring BFD
- Implement BGP route dampening

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Software Commands

Command	Description
[no] shutdown	Enables and disables an interface
address-family ipv4|**ipv6**	Enters address family configuration mode inside the BGP process
bfd interval *send_timer* **min_rx** *receive_timer* **multiplier** *number*	Enables BFD on an interface
bgp dampening	Enables BGP dampening with default parameters
bgp scan-time *scan_time*	Configures scanning intervals of BGP routers for next-hop validation
configure terminal	Enters configuration mode
debug ip bgp dampening	Enables BGP dampening debugging
interface *interface*	Enters interface configuration mode
maximum-prefix *num_of_prefixes*	Enables the maximum prefix feature inside a peer policy template
neighbor ip_address **advertisement-interval** *advertisement_interval*	Changes the advertisement interval for a neighbor
neighbor *ip_address* **fall-over bfd**	Enables BFD support for BGP
neighbor ip_address **inherit peer-policy** *peer_policy_name*	Sends a peer policy template to a neighbor so that the neighbor can inherit the configuration
neighbor ip_address **inherit peer-session** *peer_session_name*	Sends a peer session template to a neighbor so that the neighbor can inherit the configuration
neighbor ip_address **remote-as** *remote_as*	Adds an entry to the BGP or MP-BGP neighbor table
password *password*	Enables BGP authentication inside a peer session template
router bgp *AS-number*	Creates a BGP process and enters the BGP process configuration mode
show ip bgp [*prefix***]**	Displays the BGP routing table
show ip bgp dampening dampened-paths	Displays BGP dampened routes
show ip bgp neighbors *ip_address* **[configuration]**	Displays BGP neighbor information
show ip bgp summary	Displays the status of all BGP connections
show ip bgp template peer-policy	Displays locally configured peer policy templates

Command	Description
show ip bgp template peer-session	Displays locally configured peer session templates
switchport access vlan *vlan_id*	Changes the VLAN for a switch port
template peer-policy *peer_policy_name*	Creates a peer policy template and enters policy-template configuration mode
template peer-session *peer_session_name*	Creates a peer session template and enters session-template configuration mode
ttl-security hops *hops*	Enables TTL security inside a peer session template
undebug all	Disables all debugging

Cisco IOS XR Software Commands

Command	Description
address-family ipv4|ipv6 unicast	Enters address family configuration mode
af-group *name* **address-family ipv4 unicast**	Creates an address family group for BGP neighbors and enters address family group configuration mode
bfd fast-detect	Enables BFD support for the under router BGP configuration mode
bfd minimum-interval *interval*	Sets the BFD minimum interval value under router BGP configuration mode
bfd multiplier *number*	Sets the BFD multiplier value under router BGP configuration mode
bgp dampening	Enables BGP dampening with default parameters
bgp scan-time *scan_time*	Configures the scanning intervals of BGP routers for next-hop validation
commit	Commits changes to the running configuration
configure terminal	Enters configuration mode
debug bgp dampening	Enables BGP dampening debugging
maximum-prefix *num_of_prefixes*	Enables the maximum prefix feature inside a peer policy template
neighbor *IP_address*	Configures the BGP neighbor and enters BGP neighbor configuration mode
neighbor-group *name*	Creates a neighbor group and enters neighbor group configuration mode
password *password*	Enables BGP neighbor authentication using a password
router bgp *AS-number*	Creates a BGP process and enters the BGP process configuration mode
show bgp [*prefix*]	Displays the BGP routing table
show bgp address-family ipv4|ipv6 unicast dampened-paths	Displays BGP dampened routes

Command	Description
show bgp af-group *af_group_name* **configuration**	Displays effective BGP configuration for address family groups
show bgp neighbor-group *neighbor_group_name* **configuration**	Displays effective BGP configuration for neighbor groups
show bgp neighbors *ip_address* **[configuration]**	Displays BGP neighbor information
show bgp summary	Displays BGP routing protocol characteristics, including BGP neighbor status
ttl-security	Enables BGP TTL security check
undebug all	Disables all debugging
use af-group *af_group_name*	Inherits a configuration from an address family group
use neighbor-group *neighbor_group_name*	Inherits a configuration from a neighbor group

Task 1: Implement the BGP Configuration and Peer Templates

In this task, you will implement the BGP configuration and peer templates on the PE router for the existing EBGP session with the CE router.

Activity Procedure

Complete these steps:

Step 1 Verify that an EBGP session is established between the PE and CE routers in your pod. In the "state/prefix received" column, you should see a number other than zero. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          0 64500      30      26       30    0    0 00:06:23           2
10.0.2.1          0 64500    8747    8746      116    0    0    6d01h           1
192.168.101.11    0 64501   13988   12724       30    0    0    4d21h           1
```

Step 2 On the PE router, verify the BGP configuration for the CE neighbor. The PE router output (Cisco IOS XR Software) should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show running-config router bgp
Mon Oct 17 12:58:29.039 UTC
router bgp 64500
<…output omitted…>
 neighbor 192.168.101.11
  remote-as 64501
  password encrypted 0130471758434F
  ttl-security
  address-family ipv4 unicast
   route-policy PASS in
   route-policy PASS out
```

- The PE router output (Cisco IOS XE Software) should be similar to the following, taken from Pod 2:

```
PE2#show running-config | section router bgp
router bgp 64500
<…output omitted…>
 neighbor 192.168.102.21 remote-as 64502
 neighbor 192.168.102.21 password C!sc()
 neighbor 192.168.102.21 ttl-security hops 1
```

Step 3 On the PE router that is running Cisco IOS XR Software only, do as follows:

- Migrate all session-specific commands for the EBGP neighbor into the neighbor group (except the **remote-as** command). Use **EBGP** as the name of the neighbor group.
- Migrate all address-family specific commands for the EBGP neighbor into the address family group. Use **IPV4** as the name of the address-family group.
- Configure the EBGP neighbor group to inherit the configuration from the *IPV4* address-family group.
- Delete the CE router as an EBGP neighbor. Add the CE router as an EBGP neighbor again and configure the neighbor to inherit the configuration from the neighbor group.

Step 4 On the PE router that is running Cisco IOS XE Software only, do as follows:

- Migrate all session-specific commands for the EBGP neighbor into the peer session template (except the **remote-as** command). Use **EBGP_SESSION** as the name of the peer session template.
- Migrate all address-family specific commands for the EBGP neighbor into the peer policy template. If no address-family specific configuration is present, create an empty peer policy template because it will be used in the next tasks. Use **EBGP_POLICY** as the name of the peer policy template.
- Delete the CE router as an EBGP neighbor. Add the CE router as an EBGP neighbor again and configure the neighbor to inherit the configuration from the peer session and peer policy templates.

Step 5 On the PE router that is running Cisco IOS XR Software only, do as follows:

- Verify the configured address-family group.
- Verify the configured neighbor group. You should see the inherited configuration from the address-family group.
- Verify the CE neighbor configuration. You should see the inherited configuration from the neighbor group.

Step 6 On the PE router that is running Cisco IOS XE Software only, do as follows:

- Verify the configured peer policy template.
- Verify the configured peer session template.

Step 7 Verify that the EBGP session is established between the PE and CE routers in your pod. In the "state/prefix received" column, you should see a number other than zero.

Activity Verification

You have completed this task when you attain these results:

- On the PE router that is running Cisco IOS XR Software, verify the configured address-family group:

```
RP/0/RSP0/CPU0:PE1#show bgp af-group IPV4 configuration
af-group IPV4 address-family IPv4 Unicast
  policy PASS in                          []
  policy PASS out                         []
```

- On the PE router that is running Cisco IOS XR Software, verify the configured neighbor group. You should see the inherited configuration from the address-family group:

```
RP/0/RSP0/CPU0:PE1#show bgp neighbor-group EBGP configuration
neighbor-group EBGP
 password encrypted 143453180F4C63 []
 ttl-security                      []
 address-family IPv4 Unicast       []
  policy PASS in                   [a:IPV4]
  policy PASS out                  [a:IPV4]
```

- On the PE router that is running Cisco IOS XR Software, verify the CE neighbor configuration. You should see the inherited configuration from the neighbor group.

```
RP/0/RSP0/CPU0:PE1#show bgp neighbors 192.168.101.11 configuration
neighbor 192.168.101.11
 remote-as 64501                   []
 password encrypted 143453180F4C63 [n:EBGP]
```



```
 ttl-security                       [n:EBGP]
 address-family IPv4 Unicast        [n:EBGP]
  policy PASS in                    [n:EBGP a:IPV4]
  policy PASS out                   [n:EBGP a:IPV4]
```

- On the PE router that is running Cisco IOS XE Software, verify the configured peer policy template:

```
PE2#show ip bgp template peer-policy
Template:EBGP_POLICY, index:1.
Local policies:0x0, Inherited polices:0x0
Local disable policies:0x0, Inherited disable policies:0x0
Locally configured policies:
Inherited policies:
```

- On the PE router that is running Cisco IOS XE Software, verify the configured peer session template:

```
PE2#show ip bgp template peer-session
Template:EBGP_SESSION, index:1
Local policies:0x810, Inherited polices:0x0
Locally configured session commands:
 password is configured
 ttl-security hops 1
Inherited session commands:
```

- Verify that the EBGP session is established between the PE and CE routers in your pod. In the "state/prefix received" column, you should see a number other than zero. The PE router output should be similar to the following, taken from the Pod 1 PE1 Cisco IOS-XR Software router:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
<…output omitted…>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          1 64500   14718   14703       67    0    0     1w3d           8
10.0.2.1          1 64500   14683   14703       67    0    0     1w3d           0
192.168.101.11    1 64501      28      30       67    0    0 00:21:06           2
```

Task 2: Limit the Number of Prefixes Received from a BGP Neighbor

In this task, you will enable the maximum prefixes feature on the PE router. The configuration will be added to the template that was configured in the previous task.

Activity Procedure

Complete these steps:

Step 1 On the PE router, examine how many routes are received from the CE router.

Step 2 On the PE router, enable the maximum prefix feature for routes that are received from EBGP neighbors. Configure the feature in the previously configured address-family group or in the peer policy template (depending on the software that is running on the PE router). The maximum number of allowed prefixes should be large enough to accommodate all of the routes that are received from the CE router.

Verify the Configuration of the Maximum Prefix for the CE Neighbor. Activity Verification

You have completed this task when you attain these results:

- On the PE router, examine how many routes are received from the CE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.1.1          1 64500   15753   15733       85    0    0 00:00:51          8
10.0.2.1          1 64500   15709   15733       85    0    0 00:00:49          0
192.168.105.51    1 64505    1155    1062       85    0    0 00:01:12          2
```

- Verify the configuration of the maximum prefix for the CE neighbor. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp neighbors 192.168.101.11 | include Maximum
  Maximum prefixes allowed 2
```

Task 3: Improve BGP Convergence by Changing the BGP Scan and Advertisement Interval

In this task, you will improve BGP convergence by changing the BGP scan and advertisement interval on the PE router.

Activity Procedure

Complete these steps:

Step 1 On the PE router, verify the default scan interval. On the PE router, set the scan interval to 30 seconds.

Step 2 Verify that the scan interval is set to 30 seconds.

Step 3 On the CE router, verify the default advertisement interval for the PE neighbor.

Step 4 On the CE router, set the advertisement interval for the PE neighbor to 15 seconds.

Note	If required, disable BGP next-hop tracking on the CE and PE routers using the **no bgp nexthop trigger enable** command in BGP configuration mode.

Step 5 On the CE router, verify the advertisement interval for the PE neighbor.

Step 6 On the CE router, shut down the Loopback0 interface. Shutting down the interface will cease the advertisement of the 10.1.10.1/32 network to the PE router. Immediately bring the interface back up. On the PE router, examine the BGP table. You should see that it takes 15 seconds for the network 10.1.10.1/32 to reappear in the BGP table because the minimum time between successive updates on the CE router is set to 15 seconds.

Activity Verification

You have completed this task when you attain these results:

- On the PE router, verify the default scan interval. The PE router (Cisco IOS XR Software) output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
BGP router identifier 10.1.1.1, local AS number 64500
BGP generic scan interval 60 secs
BGP table state: Active
```



```
Table ID: 0xe0000000   RD version: 29
BGP main routing table version 29
Dampening enabled
BGP scan interval 60 secs
```

The PE router (Cisco IOS XE Software) output should be similar to the following, taken from Pod 2:

```
PE2#show ip bgp summary
<…output omitted…>
BGP activity 68/59 prefixes, 124/114 paths, scan interval 60 secs
```

- Verify the scan interval. It should be set to 30 seconds. The PE router (Cisco IOS XR Software) output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp summary
BGP router identifier 10.1.1.1, local AS number 64500
BGP generic scan interval 30 secs
BGP table state: Active
Table ID: 0xe0000000   RD version: 29
BGP main routing table version 29
Dampening enabled
BGP scan interval 60 secs
```

The PE router (Cisco IOS XE Software) output should be similar to the following, taken from Pod 2:

```
PE2#show ip bgp summary
<…output omitted…>
BGP activity 68/59 prefixes, 124/114 paths, scan interval 30 secs
```

- On the CE router, verify the default advertisement interval for the PE neighbor. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show ip bgp neighbors 192.168.101.10 | include minimum time
  Default minimum time between advertisement runs is 30 seconds
```

- On the CE router, verify the default advertisement interval for the PE neighbor. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show ip bgp neighbors  192.168.101.10 | include Minimum time
  Minimum time between advertisement runs is 15 seconds
```

Task 4: Improve BGP Convergence by Enabling BFD

In this task, you will improve BGP convergence by enabling BFD between PE and CE routers in the pod.

Activity Procedure

Complete these steps:

Step 1 Access the SW switch in your pod. Configure the switch port that is connecting the CE router (FastEthernet0/1) to be in another VLAN (for example, 5). This will effectively disable communication between the CE and PE router without shutting down the interfaces.

Step 2 Observe the logging messages on the CE router. After a while, the BGP session should be torn down. It can take up to 3 minutes for the session to come down:

```
Nov 29 18:40:50.115: %BGP-5-ADJCHANGE: neighbor 192.168.101.10 Down BGP
Notification sent
Nov 29 18:40:50.115: %BGP-3-NOTIFICATION: sent to neighbor 192.168.101.10 4/0
(hold time expired) 0 bytes
CE1#
```

```
Nov 29 18:40:50.115: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.101.10 IPv4
Unicast topology base removed from session  BGP Notification sent
```

Step 3 Return to the SW switch and put the FastEthernet0/1 switch port back into VLAN 1.

Step 4 On the CE router, clear the BGP session to re-establish the BGP session with the PE router.

Step 5 On the CE router, enable BFD for the PE neighbor with the following parameters:

- Send timer: 100 ms
- Receive timer: 100 ms
- Multiplier: 3

Step 6 On the PE router, enable BFD for the CE neighbor with the following parameters:

- Send timer: 100 ms
- Receive timer: 100 ms (which is not needed on Cisco IOS XR Software)
- Multiplier: 3

Step 7 On the CE router, verify the BFD session. You should see that the BFD session with the PE router is established.

Step 8 Return to the SW switch and change the VLAN of the FastEthernet0/1 interface again. Observe the CE console. You should see that the BGP adjacency went down immediately because of the enabled BFD.

```
Nov 29 18:52:15.799: %BGP-5-ADJCHANGE: neighbor 192.168.101.10 Down BFD
adjacency down
Nov 29 18:52:15.799: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.101.10 IPv4
Unicast topology base removed from session  BFD adjacency down
CE1#
Nov 29 18:53:04.767: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.101.10 IPv4
Unicast topology base removed from session  Peer closed the session
```

Step 9 On the SW switch, return the FastEthernet0/1 switch port to the VLAN.

Step 10 On the CE router, clear the BGP session in order to re-establish the BGP session with the PE router.

Activity Verification

You have completed this task when you attain these results:

- On the CE router, verify the BFD session. You should see that the BFD session with the PE router is established. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show bfd neighbors

NeighAddr                              LD/RD    RH/RS     State     Int
192.168.101.10                          1/-2146893823 Up        Up        Gi0/0
```

Task 5: Implement BGP Route Dampening

In this task, you will implement BGP route dampening.

Activity Procedure

Complete these steps:

Step 1 On the PE router, enable BGP route dampening with the default parameters. Enable debugging of BGP dampening events.

Step 2 On the CE router, shut down the Loopback0 interface. Shutting down the interface will cease the advertisement of the 10.1.10.1/32 network to the PE router.

Step 3 On the PE router, verify the BGP table. You should see that the route was not removed from the table.

Step 4 On the CE router, enable the Loopback0 interface to come back up. Disable and enable the interface a few times. Wait for more than 15 seconds between disabling and enabling the interface because the advertisement interval is set to 15 seconds. Disabling and enabling the interface a few times causes the 10.1.10.1/32 network to flap.

Step 5 On the PE router, observe the logging messages about penalizing and eventually suppressing the route. The PE router output should be similar to the following, taken from Pod 1:

```
<…output omitted…>
RP/0/RSP0/CPU0:PE1#RP/0/RSP0/CPU0:Oct 18 08:26:22.659 : bgp[1047]: [rtr]
(ip4u): Charge penalty for 10.1.10.1/32 path 64501 with halflife-time 15 min
reuse/suppress 750/2000 Flapped 2 times in 00:03:55. New penalty is 1837
RP/0/RSP0/CPU0:Oct 18 08:27:23.881 : bgp[1047]: [rtr] (ip4u): Charge penalty
for 10.1.10.1/32 path 64501 with halflife-time 15 min reuse/suppress 750/2000
Flapped 3 times in 00:04:57. New penalty is 2766
RP/0/RSP0/CPU0:PE1#RP/0/RSP0/CPU0:Oct 18 08:27:54.676 : bgp[1047]: [rtr]
(ip4u): Suppress 10.1.10.1/32 path 64501 for 00:27:40 (penalty 2702) halflife-
time 15, reuse/suppress 750/2000
```

- Answer the following question: How many times did you have to flap the route for the PE router to suppress the route? What is the default suppress penalty, half-life time, and reuse penalty?_____

Step 6 On the PE router, verify the dampened routes.

Step 7 On the PE router, examine information about the 10.1.10.1/32 route. You should see BGP dampening information about the route.

Step 8 Disable BGP dampening debugging on the PE router.

Activity Verification

You have completed this task when you attain these results:

- On the PE router, verify the BGP table. You should see that the route was not removed from the table. The route should be put into the history state. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp
   Network            Next Hop            Metric LocPrf Weight Path
*>i10.1.1.1/32        10.1.1.1                 0    100      0 i
*>i10.1.10.1/32       10.1.1.1                 0    100      0 64501 i
*>i10.2.1.1/32        10.2.1.1                 0    100      0 i
*>i10.2.10.1/32       10.2.1.1                 0    100      0 64502 i
 h 10.1.10.1/32       192.168.101.11           0             0 64501 i
```

- On the PE router, observe the logging messages about penalizing and eventually suppressing the route. The PE router output should be similar to the following, taken from Pod 1:

```
<…output omitted…>
RP/0/RSP0/CPU0:PE1#RP/0/RSP0/CPU0:Oct 18 08:26:22.659 : bgp[1047]: [rtr]
(ip4u): Charge penalty for 10.1.10.1/32 path 64501 with halflife-time 15 min
reuse/suppress 750/2000 Flapped 2 times in 00:03:55. New penalty is 1837
```

```
RP/0/RSP0/CPU0:Oct 18 08:27:23.881 : bgp[1047]: [rtr] (ip4u): Charge penalty
for 10.1.10.1/32 path 64501 with halflife-time 15 min reuse/suppress 750/2000
Flapped 3 times in 00:04:57. New penalty is 2766
RP/0/RSP0/CPU0:PE1#RP/0/RSP0/CPU0:Oct 18 08:27:54.676 : bgp[1047]: [rtr]
(ip4u): Suppress 10.1.10.1/32 path 64501 for 00:27:40 (penalty 2702) halflife-
time 15, reuse/suppress 750/2000
```

- On the PE router, verify the dampened routes. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp ipv4 unicast dampened-paths
   Network            From            Reuse    Path
*d 10.1.10.1/32       192.168.101.11  00:25:50 64501 i
```

- On the PE router, examine information about the 10.1.10.1/32 route. You should see BGP dampening information about the route. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show bgp 10.1.10.1/32
Paths: (1 available, no best path)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Not advertised to any peer
  64501, (suppressed due to dampening)
    192.168.101.11 from 192.168.101.11 (10.1.100.1)
      Origin IGP, metric 0, localpref 100, valid, external
      Received Path ID 0, Local Path ID 0, version 0
      Dampinfo: penalty 2659, flapped 4 times in 00:12:46, reuse in 00:27:20
                half life 00:15:00, suppress value 2000, reuse value 750
                Maximum suppress time 01:00:00
```


Lab 4-1: Implement Layer 2 and Layer 3 Multicast

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will implement and verify the operations of IGMP and MLD as well as observe multicast flooding on the LAN when IGMP snooping is implemented.

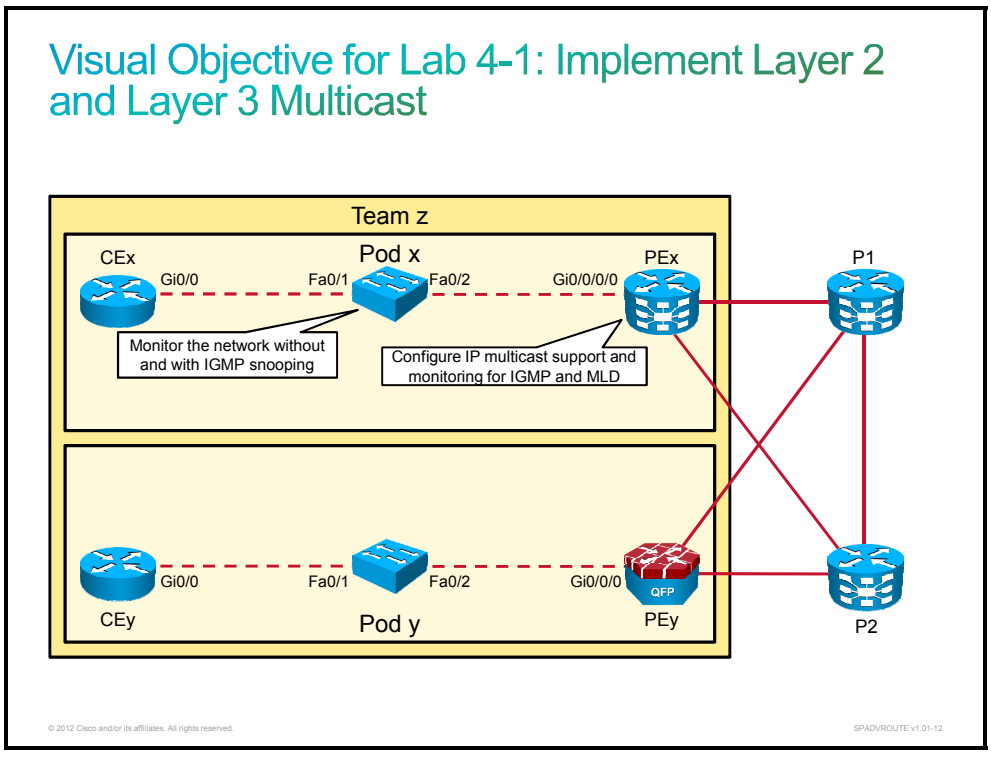
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router that is running Cisco IOS XR Software, and the second pod in the same team will work on the PE router that is running Cisco IOS XE Software. Students in the same team should coordinate their activities.

You will work on different Cisco routers that are running Cisco IOS (c2900), Cisco IOS XE (asr1001) and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Configure IP multicast support and monitoring for IGMP and MLD
- Monitor the network without and with IGMP snooping

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Software Commands

Command	Description
[no] ip igmp snooping	Enables or disables IGMP snooping globally on the switch
configure terminal	Enters configuration mode
interface *interface*	Enters interface configuration mode
ip igmp join-group *group*	Enables an interface to join a multicast group
ip igmp query-interval *interval*	Sets the IGMP query interval on the interface
ip igmp version *version*	Sets the IGMP router version on the interface
ip multicast-routing distributed	Enables IP multicast routing globally on the router
ipv6 mld query-interval *interval*	Sets the MLD query interval on the interface
ipv6 mld router	Enables the MLD router on the interface
ping *dest_ip_source source_interface*	Verifies connectivity between the source IP and destination IP
show ip | **ipv6 igmp** | **mld groups** *interface*	Displays IGMP or MLD group information
show ip | **ipv6 igmp** | **mld interface** *interface*	Displays IGMP or MLD interface information
show ip igmp snooping groups	Displays IGMP snooping information on the switch

Cisco IOS XR Software Commands

Command	Description
address-family ipv4 | **ipv6**	Enters IPv4 or IPv6 address family in multicast routing mode
commit	Commits changes to the running configuration
configure terminal	Enters configuration mode
enable	Enables multicast routing on the interface
interface *interface*	Enters interface configuration mode
multicast-routing	Enables multicast routing and enters multicast routing configuration mode
ping *dest_IP source source_IP*	Verifies connectivity between the source IP and destination IP (IPv4 and IPv6)
query-interval *interval*	Sets the IGMP or MLD query interval
router enable	Enables the MLD router

Command	Description
router mld	Enables the MLD router and enters MLD router configuration mode
show igmp | **mld groups** *interface*	Displays IGMP or MLD group information
show igmp | **mld interface** *interface*	Displays IGMP or MLD interface information
version *version*	Sets the IGMP or MLD router version

Task 1: Enable IGMP and MLD

In this task, you will configure and verify IGMP and MLD support on the pod PE router.

Activity Procedure

Complete these steps:

Step 1 On the pod PE router, enable IPv4 and IPv6 multicast routing on the first Gigabit Ethernet interface.

On the pod PE router, verify the IGMP version and query interval:

```
RP/0/RSP0/CPU0:PE1#show igmp interface GigabitEthernet 0/0/0/0
Wed Nov  2 11:31:03.533 UTC

GigabitEthernet0/0/0/0 is up, line protocol is up
  Internet address is 192.168.101.10/24
  IGMP is enabled on interface
  Current IGMP version is 3
  IGMP query interval is 60 seconds
  IGMP querier timeout is 125 seconds
  IGMP max query response time is 10 seconds
  Last member query response interval is 1 seconds
  IGMP activity: 6 joins, 0 leaves
  IGMP querying router is 192.168.101.10 (this system)
```

On the pod PE router, verify the MLD version and query interval:

```
RP/0/RSP0/CPU0:PE1#show mld interface GigabitEthernet 0/0/0/0
Wed Nov  2 11:31:36.179 UTC

GigabitEthernet0/0/0/0 is up, line protocol is up
  Internet address is fe80::4255:39ff:fe2e:c420
  MLD is enabled on interface
  Current MLD version is 2
  MLD query interval is 125 seconds
  MLD querier timeout is 255 seconds
  MLD max query response time is 10 seconds
  Last member query response interval is 1 seconds
  MLD activity: 7 joins, 0 leaves
  MLD querying router is fe80::4255:39ff:fe2e:c420 (this system)
```

Step 2 On the pod PE router of the first Gigabit Ethernet interface, configure IGMP version 2 and the IGMP query interval at 30 seconds.

```
RP/0/RSP0/CPU0:PE1#show igmp interface GigabitEthernet 0/0/0/0
Wed Nov  2 11:35:48.230 UTC
```

```
GigabitEthernet0/0/0/0 is up, line protocol is up
  Internet address is 192.168.101.10/24
  IGMP is enabled on interface
  Current IGMP version is 2
  IGMP query interval is 30 seconds
  IGMP querier timeout is 65 seconds
  IGMP max query response time is 10 seconds
  Last member query response interval is 1 seconds
  IGMP activity: 6 joins, 0 leaves
  IGMP querying router is 192.168.101.10 (this system)
```

Step 3 On the pod PE router of the first Gigabit Ethernet interface, enable the MLD router and configure the MLD query interval at 60 seconds.

```
RP/0/RSP0/CPU0:PE1#show mld interface GigabitEthernet 0/0/0/0
Wed Nov  2 11:39:10.998 UTC

GigabitEthernet0/0/0/0 is up, line protocol is up
  Internet address is fe80::4255:39ff:fe2e:c420
  MLD is enabled on interface
  Current MLD version is 2
  MLD query interval is 60 seconds
  MLD querier timeout is 125 seconds
  MLD max query response time is 10 seconds
  Last member query response interval is 1 seconds
  MLD activity: 9 joins, 0 leaves
  MLD querying router is fe80::4255:39ff:fe2e:c420 (this system)
```

Step 4 On the pod CE router, configure the first Gigabit Ethernet interface to join multicast group 234.1.1.1.

Activity Verification

You have completed this task when you attain these results:

- On the pod PE router, verify IGMP groups on the first Gigabit Ethernet interface. Verify that multicast group 234.1.1.1 appears in the IGMP groups table.

```
RP/0/RSP0/CPU0:PE1#show igmp groups GigabitEthernet 0/0/0/0
Wed Nov  2 11:41:30.060 UTC
IGMP Connected Group Membership
Group Address   Interface                      Uptime    Expires   Last
Reporter
224.0.0.2       GigabitEthernet0/0/0/0         00:18:27  never
192.168.101.10
224.0.0.5       GigabitEthernet0/0/0/0         00:20:55  never
192.168.101.10
224.0.0.6       GigabitEthernet0/0/0/0         00:20:55  never
192.168.101.10
224.0.0.13      GigabitEthernet0/0/0/0         00:18:27  never
192.168.101.10
224.0.0.22      GigabitEthernet0/0/0/0         00:18:27  never
192.168.101.10
224.0.1.40      GigabitEthernet0/0/0/0         00:18:27  never
192.168.101.10
234.1.1.1       GigabitEthernet0/0/0/0         00:01:09  00:00:48
192.168.102.21
```


- On the pod PE router, verify MLD groups on the first Gigabit Ethernet interface.

```
RP/0/RSP0/CPU0:PE1#show mld groups GigabitEthernet 0/0/0/0
Wed Nov  2 11:43:09.733 UTC
MLD Connected Group Membership

GigabitEthernet0/0/0/0

Group Address : ff02::2
Last Reporter : fe80::eab7:48ff:fe2c:a180
       Uptime : 00:20:07
      Expires : never
Group Address : ff02::5
Last Reporter : fe80::eab7:48ff:fe2c:a180
       Uptime : 00:20:00
      Expires : 00:01:51
Group Address : ff02::6
Last Reporter : fe80::eab7:48ff:fe2c:a180
       Uptime : 00:20:00
      Expires : 00:01:51
Group Address : ff02::d
Last Reporter : fe80::4255:39ff:fe2e:c420
       Uptime : 00:20:07
      Expires : never
Group Address : ff02::16
Last Reporter : fe80::4255:39ff:fe2e:c420
       Uptime : 00:20:07
      Expires : never
Group Address : ff02::1:ff00:11
Last Reporter : fe80::eab7:48ff:fe2c:a180
       Uptime : 00:20:00
      Expires : 00:01:51
Group Address : ff02::1:ff00:21
Last Reporter : fe80::4255:39ff:fe86:f968
       Uptime : 00:11:21
      Expires : 00:01:46
Group Address : ff02::1:ff2c:a180
Last Reporter : fe80::eab7:48ff:fe2c:a180
       Uptime : 00:20:00
      Expires : 00:01:51
Group Address : ff02::1:ff86:f968
Last Reporter : fe80::4255:39ff:fe86:f968
       Uptime : 00:11:21
      Expires : 00:01:46
```

Task 2: Verify IGMP Snooping

In this task, you will verify IGMP snooping on the pod switch. You will disable IGMP snooping and verify the results.

Activity Procedure

Complete these steps:

Step 1 On the pod switch, verify IGMP snooping:

```
SW1#show ip igmp snooping groups
Vlan      Group                   Type        Version     Port List
------------------------------------------------------------------------
1         224.0.1.40              igmp        v2,v3       Fa0/2
1         234.1.1.1               igmp        v2          Fa0/1, Fa0/2,
                                                          Fa0/23
```

Step 2 On the pod switch, disable IGMP snooping.

Verify that IGMP snooping is disabled:

```
SW1#show ip igmp snooping groups
SW1#
```

Step 3 Enable IGMP snooping.

Step 4 On the pod CE router, configure the first Gigabit Ethernet interface to leave multicast group 234.1.1.1.

Activity Verification

You have completed this task when you attain these results:

- On the pod switch, verify IGMP snooping:

```
SW1#show ip igmp snooping groups
Vlan      Group                   Type        Version     Port List
------------------------------------------------------------------------
1         224.0.1.40              igmp        v2          Fa0/2
1         234.1.1.1               igmp        v2          Fa0/1, Fa0/2,
                                                          Fa0/23
```


Lab 5-1: Enable and Optimize PIM-SM

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will enable multicast on the router in your team. The P1 router is preconfigured to act as an RP for your multicast traffic. You will configure receivers for multicast traffic on the CE and PE router. The other pod CE router will act as a multicast source.

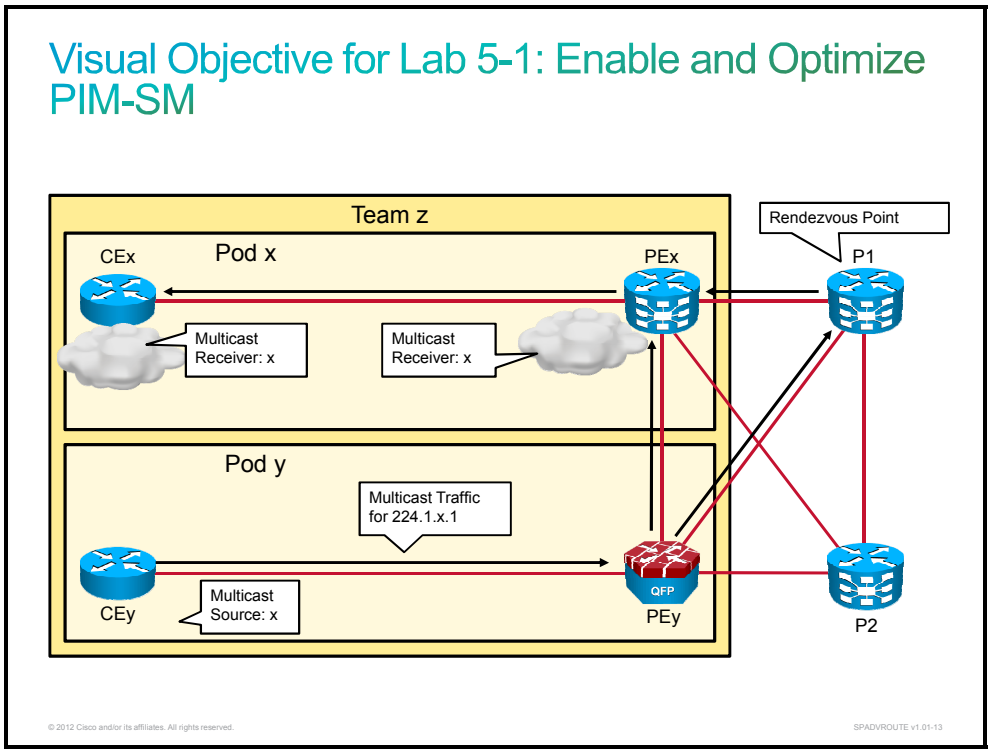
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router that is running Cisco IOS XR Software, and the second pod in the same team will work on the PE router that is running Cisco IOS XE Software. Students in the same team should coordinate their activities.

You will work on different Cisco routers that are running Cisco IOS (c2900), Cisco IOS XE (asr1001), and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Implement multicast routing, PIM-SM, and manual RP configuration
- Observe shared tree formation
- Observe the switchover from the shared tree to the SPT

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Software Commands

Command	Description
configure terminal	Enters configuration mode
interface *interface*	Enters interface configuration mode
ip igmp join-group *group_address*	Configures an interface on the router to join the specified group or channel
ip multicast-routing	Enables IP multicast routing
ip pim rp-address *RP_address*	Statically configures the address of a PIM RP for multicast groups
ip pim sparse-mode	Enables an interface for PIM-SM
ip pim spt-threshold *threshold*	Configures when a PIM leaf router should join the shortest path source tree
ping *dest_IP source source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
show ip mroute	Displays the contents of the multicast routing table
show ip pim interface	Displays information about interfaces that are configured for PIM
show ip pim neighbor	Displays information about PIM neighbors discovered by PIMv1 router query messages or PIMv2 hello messages
show ip route	Displays routing table on a router

Cisco IOS XR Software Commands

Command	Description
show route	Displays routing table on a router
address-family ipv4	Enters IPv4 address family under specific configuration mode
commit	Commits changes to the running configuration
configure	Enters configuration mode
show mrib route	Displays the contents of the multicast routing table
show pim neighbor	Displays information about PIM neighbors discovered by PIMv1 router query messages or PIMv2 hello messages
show pim interface	Displays information about interfaces that are configured for PIM
interface *interface*	Enters interface configuration mode
enable	Enables an interface for multicast routing or PIM (under the appropriate configuration mode)
multicast-routing	Enters multicast routing configuration mode

Command	Description
router pim	Enters PIM configuration mode
ping *dest_IP source source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
rp-address *RP_address*	Statically configures the address of a PIM RP for multicast groups under PIM configuration mode
spt-threshold infinity	Configures that a PIM leaf router should join the shortest path source tree immediately
ip igmp join-group *group_address*	Configures an interface on the router to join the specified group or channel
router igmp	Enters IGMP configuration mode

Task 1: Implement PIM-SM

In this task, you will enable multicast on all routers.

Activity Procedure

Complete these steps:

Step 1 Access the PE router. Make sure that the route for the P1 Loopback0 interface points to GigabitEthernet0/0/0/2 (GigabitEthernet0/0/2 on PEy). Make sure that the route for the other pod CE router Loopback0 interface points to GigabitEthernet0/0/0/1 (GigabitEthernet0/0/1 on PEy).

Step 2 Enable IP multicast routing on PE and CE router. Enable multicast on all interfaces that have IP addresses assigned (Cisco IOS XR).

Step 3 Enable PIM-SM on CE and PE routers on all interfaces that have IP addresses assigned.

Step 4 On CE and PE router, define the SPT threshold as infinity. This should force the routers to always stay on the shared tree.

Step 5 Manually configure the RP address on the CE and PE router. The P1 router with Loopback0 IP address will act as RP for all multicast groups.

Step 6 Both pods from the same team should be finished with the previous steps at this point.

Step 7 Verify PIM state on interfaces on the PE router.

Step 8 Verify PIM neighbors on the PE router.

Activity Verification

You have completed this task when you attain these results:

- Make sure that the route for the P1 Loopback0 interface points to GigabitEthernet0/0/0/2 interface (GigabitEthernet0/0/2 on PEy). Make sure that the route for the other pod CE router Loopback0 interface points to GigabitEthernet0/0/0/1 interface (GigabitEthernet0/0/1 on PEy). The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show route 10.0.1.1
Routing entry for 10.0.1.1/32
  Known via "isis 1", distance 115, metric 20, type level-2
  Installed Nov 17 21:21:00.385 for 15:19:22
```

```
  Routing Descriptor Blocks
    192.168.11.1, from 10.0.1.1, via GigabitEthernet0/0/0/2
      Route metric is 20
  No advertising protos.
RP/0/RSP0/CPU0:PE1#show route 10.2.10.1
Routing entry for 10.2.10.1/32
  Known via "ospf 1", distance 110, metric 12, type inter area
  Installed Nov 17 21:31:46.639 for 15:09:28
  Routing Descriptor Blocks
    192.168.112.20, from 10.2.1.1, via GigabitEthernet0/0/0/1
      Route metric is 12
  No advertising protos.
```

- Verify PIM state on interfaces on the PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show pim interface
PIM interfaces in VRF default
Address               Interface                  PIM  Nbr   Hello  DR    DR
                                                      Count Intvl  Prior

10.1.1.1              Loopback0                  on   1     30     1
this system
192.168.101.10        GigabitEthernet0/0/0/0     on   2     30     1
192.168.101.11
192.168.112.10        GigabitEthernet0/0/0/1     on   2     30     1
192.168.112.20
192.168.11.10         GigabitEthernet0/0/0/2     on   2     30     1
this system
192.168.12.10         GigabitEthernet0/0/0/3     on   2     30     1
this system
```

- Verify PIM neighbors on the PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show pim neighbor
PIM neighbors in VRF default

Neighbor Address      Interface              Uptime    Expires  DR pri   Flags

10.1.1.1*             Loopback0              02:40:55  00:01:15 1 (DR) B P
192.168.101.10*       GigabitEthernet0/0/0/0 1d03h     00:01:28 1      B P
192.168.101.11        GigabitEthernet0/0/0/0 1d03h     00:01:16 1 (DR) P
192.168.112.10*       GigabitEthernet0/0/0/1 22:10:41  00:01:43 1      B P
192.168.112.20        GigabitEthernet0/0/0/1 22:10:41  00:01:19 1 (DR) P
192.168.11.1          GigabitEthernet0/0/0/2 1d03h     00:01:22 1      B
192.168.11.10*        GigabitEthernet0/0/0/2 1d03h     00:01:40 1 (DR) B P
192.168.12.2          GigabitEthernet0/0/0/3 1d03h     00:01:16 1      B
192.168.12.10*        GigabitEthernet0/0/0/3 1d03h     00:01:21 1 (DR) B P
```


Task 2: Shared Tree Formation—Receivers

In this task, you will configure multicast receivers. You will observe multicast routing tables on routers when receivers announce their presence.

Activity Procedure

Complete these steps:

Step 1 Simulate multicast receivers for group 224.1.x.1 (or 224.1.y.1) on the Loopback0 interface on the CE and PE routers.

Note Throughout the lab exercise, use the **ip igmp join-group** command on Cisco IOS and IOS-XE Software under interface configuration mode to simulate multicast receivers. On the Cisco IOS XR Software, use the **join-group** command under interface configuration mode under router igmp configuration mode.

Step 2 Examine the multicast routing table on the PE, CE, and P1 routers.

Note Use Telnet to connect to the P1 router to examine multicast routing table.

Step 3 Answer the following question and complete the table for the (*,G) entry for your pod:

Why are there no incoming interfaces on the P1 router for the (*,G) entry?

_____

Router	Incoming Interface	OIL
CE		
PE		
P1		

Activity Verification

You have completed this task when you attain these results:

- Examine the multicast routing table on the PE, CE, and P1 routers. The routers output should be similar to the following, taken from Pod 1:

```
CE1#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       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

(*, 224.1.1.1), 00:42:14/00:02:05, RP 10.0.1.1, flags: SCL
  Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.101.10
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:42:14/00:02:05

(*, 224.0.1.40), 00:42:14/00:02:58, RP 10.0.1.1, flags: SPCL
  Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.101.10
  Outgoing interface list: Null

RP/0/RSP0/CPU0:PE1#show mrib route
IP Multicast Routing Information Base
Entry flags: L - Domain-Local Source, E - External Source to the Domain,
    C - Directly-Connected Check, S - Signal, IA - Inherit Accept,
    IF - Inherit From, D - Drop, MA - MDT Address, ME - MDT Encap,
    MD - MDT Decap, MT - MDT Threshold Crossed, MH - MDT interface handle
    CD - Conditional Decap, MPLS - MPLS Decap, MF - MPLS Encap, EX - Extranet
    MoFE - MoFRR Enabled, MoFS - MoFRR State
Interface flags: F - Forward, A - Accept, IC - Internal Copy,
    NS - Negate Signal, DP - Don't Preserve, SP - Signal Present,
    II - Internal Interest, ID - Internal Disinterest, LI - Local Interest,
    LD - Local Disinterest, DI - Decapsulation Interface
    EI - Encapsulation Interface, MI - MDT Interface, LVIF - MPLS Encap,
    EX - Extranet, A2 - Secondary Accept

(*,224.0.0.0/4) RPF nbr: 192.168.51.1 Flags: C
  Up: 1d03h

(*,224.0.0.0/24) Flags: D
  Up: 1d04h

(*,224.0.1.39) Flags: S
  Up: 1d04h

(*,224.0.1.40) Flags: S
  Up: 1d04h
  Outgoing Interface List
    GigabitEthernet0/0/0/0 Flags: II LI, Up: 1d04h

(*,224.1.1.1) RPF nbr: 192.168.51.1 Flags: C
  Up: 04:03:32
  Incoming Interface List
    GigabitEthernet0/0/0/2 Flags: A, Up: 04:03:32
  Outgoing Interface List
    Loopback0 Flags: F IC NS II LI, Up: 04:02:01
    GigabitEthernet0/0/0/0 Flags: F NS, Up: 00:01:43

(*,232.0.0.0/8) Flags: D
  Up: 1d04h

RP/0/RSP0/CPU0:P1#show mrib route
```



```
<…output omitted…>
(*,224.1.1.1) RPF nbr: 10.0.1.1 Flags: C
  Up: 00:52:44
  Incoming Interface List
    Decapstunnel0 Flags: A, Up: 00:52:44
  Outgoing Interface List
    GigabitEthernet0/0/0/4 Flags: F NS, Up: 00:52:44

(*,224.1.2.1) RPF nbr: 10.0.1.1 Flags: C
  Up: 00:52:53
  Incoming Interface List
    Decapstunnel0 Flags: A, Up: 00:52:53
  Outgoing Interface List
    GigabitEthernet0/0/0/8 Flags: F NS, Up: 00:52:53
```

Task 3: Shared Tree Formation—Sources

In this task, you will trigger some multicast traffic. Then you will observe the multicast routing tables on routers.

Activity Procedure

Complete these steps:

Step 1 Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging to your multicast group address. Use the GigabitEthernet0/0 interface as a source interface. Send a large number of ICMP packets—100, for example. You should see that both receivers reply to the ping.

Step 2 Examine the multicast routing table on your CE router. Answer the following questions:

Did the entry for your group change? Why or why not?

_____

Are there any (S,G) entries present for your group? Why or why not?

_____

Step 3 Use Telnet to connect the other pod PE router. Examine the multicast routing table. Answer the following questions:

What is the incoming interface for the (S,G) entry for your group?

_____

Which interfaces are present in the OIL for the (S,G) entry for your group?

_____

Why is the OIL of the (*,G) entry for your group empty?

_____

Step 4 Interrupt the multicast ping from the other pod CE router. This is usually done using the Shift + Ctrl + 6 key combination.

Activity Verification

You have completed this task when you attain these results:

- Start the multicast traffic by pinging to your multicast group address from the other pod CE router. The CE router output should be similar to the following, taken from Pod 2:

```
CE2#ping 224.1.1.1 repeat 100 source GigabitEthernet0/0

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.102.21

Reply to request 0 from 10.1.10.1, 1 ms
Reply to request 0 from 10.1.1.1, 1 ms
Reply to request 1 from 10.1.10.1, 1 ms
Reply to request 1 from 10.1.1.1, 1 ms
<…output omitted…>
```

- Examine the multicast routing table on your CE router. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show ip mroute
<…output omitted…>
(*, 224.1.1.1), 01:26:59/00:02:22, RP 10.0.1.1, flags: SCL
  Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.101.10
  Outgoing interface list:
    Loopback0, Forward/Sparse, 01:26:59/00:02:22

(*, 224.0.1.40), 01:26:59/00:02:18, RP 10.0.1.1, flags: SPCL
  Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.101.10
  Outgoing interface list: Null
```

- Examine the multicast routing table on the other pod PE router. The PE router output should be similar to the following, taken from Pod 2:

```
PE2#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       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

(*, 224.1.2.1), 01:30:33/00:02:35, RP 10.0.1.1, flags: SCL
  Incoming interface: GigabitEthernet0/0/2, RPF nbr 192.168.21.1
  Outgoing interface list:
    GigabitEthernet0/0/0, Forward/Sparse, 01:29:36/00:02:30
    Loopback0, Forward/Sparse, 01:30:33/00:02:35

(*, 224.1.1.1), 00:03:45/stopped, RP 10.0.1.1, flags: SP
  Incoming interface: GigabitEthernet0/0/2, RPF nbr 192.168.21.1
```



```
  Outgoing interface list: Null

(192.168.102.21, 224.1.1.1), 00:03:45/00:03:14, flags: T
  Incoming interface: GigabitEthernet0/0/0, RPF nbr 192.168.102.21
  Outgoing interface list:
    GigabitEthernet0/0/1, Forward/Sparse, 00:03:45/00:02:44
<…output omitted…>
```

Task 4: Switching to the SPT

In this task, you will configure the last-hop routers to switch to SPT immediately after the first packet is received over the shared tree.

Activity Procedure

Complete these steps:

Step 1 Configure the CE and PE routers to switch to SPT immediately after the first packet arrives over the shared tree.

Step 2 Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging to your multicast group address. Use the GigabitEthernet0/0 interface as a source interface. Send a large number of ICMP packets—100, for example. You should see that both receivers reply to the ping.

Step 3 Examine the multicast routing table on your CE router. Answer the following questions:

Are there any (S,G) entries present for your group? Why or why not?

_____

Step 4 Examine the multicast routing table on your PE router. Answer the following questions:

Are there any (S,G) entries present for your group? Why or why not?

_____

Which interface is used as the incoming interface for the (S,G) entry for your group?

_____

Step 5 Remove the simulated multicast receivers for group 224.1.x.1 (or 224.1.y.0) from the Loopback0 interface on the CE and PE routers.

Note You should see that traffic now flows between the PE routers directly because the SPT has been built. When the SPT switchover was disabled, traffic went over the RP router.

Activity Verification

You have completed this task when you attain these results:

- Start the multicast traffic by pinging to your multicast group address from the other pod CE router. The CE router output should be similar to the following, taken from Pod 2:

```
CE2#ping 224.1.1.1 repeat 100 source GigabitEthernet0/0

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.102.21
```

```
Reply to request 0 from 10.1.10.1, 1 ms
Reply to request 0 from 10.1.1.1, 1 ms
Reply to request 1 from 10.1.10.1, 1 ms
Reply to request 1 from 10.1.1.1, 1 ms
<…output omitted…>
```

- Examine the multicast routing table on your CE router. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show ip mroute
<…output omitted…>
(*, 224.1.1.1), 02:01:15/stopped, RP 10.0.1.1, flags: SJCL
  Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.101.10
  Outgoing interface list:
    Loopback0, Forward/Sparse, 02:01:15/00:02:59

(192.168.102.21, 224.1.1.1), 00:00:07/00:02:52, flags: LJT
  Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.101.10
  Outgoing interface list:
    Loopback0, Forward/Sparse, 00:00:07/00:02:59
<…output omitted…>
```

- Examine the multicast routing table on your PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show mrib route
<…output omitted…>


(*,224.1.1.1) RPF nbr: 192.168.51.1 Flags: C
  Up: 05:19:22
  Incoming Interface List
    GigabitEthernet0/0/0/2 Flags: A NS, Up: 05:19:22
  Outgoing Interface List
    Loopback0 Flags: F IC NS II LI, Up: 05:17:51
    GigabitEthernet0/0/0/0 Flags: F NS, Up: 01:17:33

(192.168.102.21,224.1.1.1) RPF nbr: 192.168.152.20 Flags:
  Up: 00:00:39
  Incoming Interface List
    GigabitEthernet0/0/0/1 Flags: A, Up: 00:00:39
  Outgoing Interface List
    Loopback0 Flags: F IC NS, Up: 00:00:39
    GigabitEthernet0/0/0/0 Flags: F NS, Up: 00:00:39
<…output omitted…>
```

Lab 5-2: Implement PIM-SM Enhancements

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will first configure and monitor PIM-SSM on the CE and PE routers. Then you will configure and monitor BIDIR-PIM on the CE and PE routers.

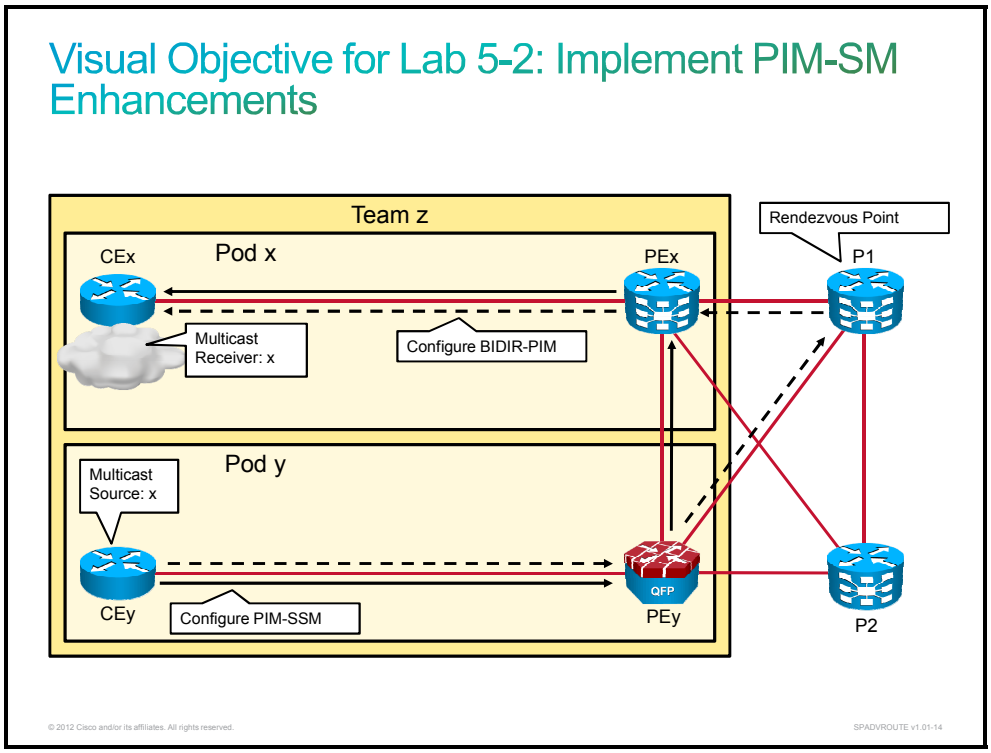
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router that is running Cisco IOS XR Software, and the second pod in the same team will work on the PE router that is running Cisco IOS XE Software. Students in the same team should coordinate their activities.

You will work on different Cisco routers that are running Cisco IOS (c2900), Cisco IOS XE (asr1001), and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Implement and monitor PIM-SSM
- Implement and monitor BIDIR-PIM

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Software Commands

Command	Description
configure terminal	Enters configuration mode
interface *interface*	Enters interface configuration mode
ip access-list standard *acl_name*	Creates a standard ACL and enters access list configuration mode
ip igmp join-group *group_address* [**source** *source*]	Configures an interface on the router to join the specified group or channel
ip pim bidir-enable	Globally enables BIDIR-PIM
ip pim rp-address *RP_address* **[bidir]**	Statically configures the address of a PIM RP for multicast groups and enables BIDIR-PIM
ip pim ssm range *acl_name*	Enables PIM-SSM for specified groups
permit | **deny host** *IP_address*	Creates a standard ACL entry under access list configuration mode
ping *dest_IP source source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
show ip mroute	Displays the contents of the multicast route (mroute) table
show ip pim interface	Displays information about interfaces that are configured for PIM
show ip pim interface df	Displays the IP address of the elected DF for each RP of an interface
show ip pim neighbor	Displays information about PIM neighbors discovered by PIMv1 router query messages or PIMv2 hello messages

Cisco IOS XR Software Commands

Command	Description
address-family ipv4	Enters IPv4 address family under specific configuration mode
commit	Commits changes to the running configuration
configure	Enters configuration mode
interface *interface*	Enters interface configuration mode
ipv4 access-list *acl_name*	Creates an ACL and enters access list configuration mode
join-group *group_address* **[***source***]**	Configures an interface on the router to join the specified group or channel
multicast-routing	Enters multicast routing configuration mode
permit | **deny host** *IP_address*	Creates a standard ACL entry under access list configuration mode

Command	Description
ping *dest_IP source source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
router igmp	Enters IGMP configuration mode
router pim	Enters PIM configuration mode
rp-address *RP_address* **[bidir]**	Statically configures the address of a PIM RP for multicast groups and enables BIDIR-PIM
show mrib route	Displays the contents of the multicast routing (mroute) table
show pim df winner	Displays the IP address of the elected DF for each RP of an interface
show pim interface	Displays information about interfaces that are configured for PIM
show pim neighbor	Displays information about PIM neighbors discovered by PIMv1 router query messages or PIMv2 hello messages
ssm range *acl_name*	Enables PIM-SSM for specified groups under multicast configuration mode

Task 1: Implement PIM-SSM

In this task, you will enable PIM-SSM on the CE and PE routers in your pod. Multicast routing and PIM should be already enabled from the previous lab exercise.

Activity Procedure

Complete these steps:

Step 1 Enable PIM-SSM on the PE and CE router. Configure the SSM address range to include the 224.1.x.1 and 224.1.y.1 multicast groups.

Note Both pods from the same team should be finished with the previous step at this point.

Step 2 Verify the PIM state on interfaces on the PE router.

Step 3 Verify the PIM neighbors on the PE router.

Step 4 Simulate multicast receivers for group 224.1.x.1 (or 224.1.y.0) on the Loopback0 interface on the CE and PE router. The routers should be interested only in traffic coming from the other pod CE router GigabitEthernet0/0 interface.

Note Throughout the lab exercise, use the **ip igmp join-group source** command under interface configuration mode to simulate SSM multicast receivers. On the Cisco IOS XR Software, use **join-group source** command under interface configuration mode under router igmp configuration mode.

Step 5 Examine the multicast routing table on the PE router. Answer the following questions:

What is the incoming interface for the (S,G) entry for your group?

_____

What is the OIL for the (S,G) entry for your group?

_____

Step 6 Use Telnet to connect to the other pod PE router. Examine the multicast routing table and answer the following questions:

What is the incoming interface for the (S,G) entry for your group?

_____

What is the OIL for the (S,G) entry for your group?

_____

Note You should see that routers created SPT across all routers between the source and receivers.

Step 7 Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging to your multicast group address. Use the GigabitEthernet0/0 interface as a source interface. Send a large number of ICMP packets—100, for example. You should see that the CE and PE routers reply to the ping.

Step 8 Remove the simulated multicast receivers for group 224.1.x.1 (or 224.1.y.0) from the Loopback0 interface on the CE and PE routers.

Step 9 Disable PIM-SSM on the PE and CE routers.

Activity Verification

You have completed this task when you attain these results:

- Verify the PIM state on interfaces on the PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show pim interface
PIM interfaces in VRF default
Address              Interface                      PIM  Nbr    Hello  DR    DR
                                                        Count  Intvl  Prior

10.2.1.1             Loopback0                       on   1      30     1
this system
192.168.101.10       GigabitEthernet0/0/0/0          on   2      30     1
192.168.101.11
192.168.112.10       GigabitEthernet0/0/0/1          on   2      30     1
192.168.112.20
192.168.11.10        GigabitEthernet0/0/0/2          on   2      30     1
this system
192.168.12.10        GigabitEthernet0/0/0/3          on   2      30     1
this system
```

- Verify the PIM neighbors on the PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show pim neighbor
PIM neighbors in VRF default

Neighbor Address      Interface              Uptime    Expires  DR pri   Flags

10.1.1.1*             Loopback0              3d22h     00:01:44 1 (DR) B P
192.168.101.10*       GigabitEthernet0/0/0/0 4d23h     00:01:16 1      B P
192.168.101.11        GigabitEthernet0/0/0/0 23:39:30  00:01:33 1 (DR) P
```



```
192.168.112.10*       GigabitEthernet0/0/0/1 4d18h     00:01:42 1      B P
192.168.112.20        GigabitEthernet0/0/0/1 4d18h     00:01:28 1 (DR) P
192.168.11.1          GigabitEthernet0/0/0/2 4d23h     00:01:21 1      B
192.168.11.10*        GigabitEthernet0/0/0/2 4d23h     00:01:39 1 (DR) B P
192.168.12.2          GigabitEthernet0/0/0/3 4d23h     00:01:44 1      B
192.168.12.10*        GigabitEthernet0/0/0/3 4d23h     00:01:20 1 (DR) B P
```

- Examine the multicast routing table on the PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show mrib route
<…output omitted…>
 (192.168.102.21,224.1.1.1) RPF nbr: 192.168.112.20 Flags:
  Up: 18:22:05
  Incoming Interface List
    GigabitEthernet0/0/0/1 Flags: A, Up: 18:22:05
  Outgoing Interface List
    Loopback0 Flags: F IC NS II LI, Up: 18:04:26
    GigabitEthernet0/0/0/0 Flags: F NS, Up: 18:22:05
<…output omitted…>
```

- Examine the multicast routing table on the other pod PE router. The PE router output should be similar to the following, taken from Pod 2:

```
PE2#show ip mroute
<…output omitted…>
 (192.168.102.21, 224.1.1.1), 18:23:36/00:02:54, flags: sT
  Incoming interface: GigabitEthernet0/0/0, RPF nbr 192.168.102.21
  Outgoing interface list:
    GigabitEthernet0/0/1, Forward/Sparse, 18:23:36/00:02:54

(*, 224.0.1.40), 3d20h/00:02:35, RP 10.0.1.1, flags: SCL
  Incoming interface: GigabitEthernet0/0/2, RPF nbr 192.168.21.1
  Outgoing interface list:
    GigabitEthernet0/0/0, Forward/Sparse, 3d20h/00:02:47
```

- Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging to your multicast group address:

```
CE2#ping 224.1.1.1 repeat 100 source GigabitEthernet0/0
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.102.21

Reply to request 0 from 10.1.10.1, 4 ms
Reply to request 0 from 10.1.1.1, 4 ms
Reply to request 1 from 10.1.10.1, 1 ms
Reply to request 1 from 10.1.1.1, 1 ms
<…output omitted…>
```

Task 2: Implement BIDIR-PIM

In this task, you will configure BIDIR-PIM on the routers in your pod.

Activity Procedure

Complete these steps:

Step 1 Enable BIDIR-PIM on the CE and PE routers. Configure BIDIR-PIM to include the 224.1.x.1 and 224.1.y.1 multicast groups.

Note	Both pods from the same team should be finished with the previous step at this point.

Step 2 Use Telnet to connect to the P1 router. Verify the PIM configuration. Enable the P1 router for BIDIR-PIM if it is not already enabled.

Note	Coordinate the previous step with the other pod and other teams.

Step 3 Return to the PE router. Determine which router is elected as DF for each segment.

Step 4 Simulate multicast receivers for group 224.1.x.1 (or 224.1.y.0) on the Loopback0 interface on your pod CE and PE routers.

Step 5 Examine multicast routing table on the PE router. Answer the following questions:

Which significant (*,G) entries are present in the table?

_____

Which interfaces are in the OIL in the (*,G entry) for your group?

_____

Step 6 Use Telnet to connect to the other pod CE router. Examine the multicast routing table.

Which significant (*,G) entries are present in the table?

_____

Step 7 Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging to your multicast group address. Use the GigabitEthernet0/0 interface as a source interface. Send a large number of ICMP packets—100, for example. You should see that the CE and PE routers reply to the ping.

Step 8 Return to the Telnet session to the other pod CE router. Examine the multicast routing table on the PE router again. Answer the following questions:

Have been there any significant changes to the multicast routing table? Why or why not?

_____

Activity Verification

You have completed this task when you attain these results:

- Verify the PIM configuration on the P1 router:

```
RP/0/RSP0/CPU0:P1#show running-config router pim
router pim
 address-family ipv4
  rp-address 10.0.1.1 bidir
```

- On the PE router, verify which router is elected as DF for each segment. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show pim df winner
RP               Interface                DF Winner        Metrics
10.0.1.1         GigabitEthernet0/0/0/3   192.168.12.10    [115/20]
10.0.1.1         GigabitEthernet0/0/0/2   192.168.11.1     [0/0]
10.0.1.1         GigabitEthernet0/0/0/1   192.168.112.20   [115/20]
10.0.1.1         GigabitEthernet0/0/0/0   192.168.101.10   [115/20]
10.0.1.1         Loopback0                10.1.1.1         [115/20]
```


- Examine the multicast routing table on the PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show mrib route
<…output omitted…>
(*,224.1.1.1) RPF nbr: 192.168.11.1 Flags: IA IF
  Up: 00:38:57
  Incoming Interface List
    Loopback0 Flags: F A IC II LI, Up: 00:38:56
    GigabitEthernet0/0/0/0 Flags: F A, Up: 00:38:56
    GigabitEthernet0/0/0/2 Flags: F A, Up: 00:38:57
    GigabitEthernet0/0/0/3 Flags: A, Up: 00:38:56
  Outgoing Interface List
    Loopback0 Flags: F A IC II LI, Up: 00:38:56
    GigabitEthernet0/0/0/0 Flags: F A, Up: 00:38:56
    GigabitEthernet0/0/0/2 Flags: F A, Up: 00:38:57

(*,224.1.2.1) RPF nbr: 192.168.11.1 Flags: IF
  Up: 00:38:57
  Incoming Interface List
    Loopback0 Flags: A, Up: 00:38:56
    GigabitEthernet0/0/0/0 Flags: A, Up: 00:38:56
    GigabitEthernet0/0/0/2 Flags: F A, Up: 00:38:57
    GigabitEthernet0/0/0/3 Flags: A, Up: 00:38:56
  Outgoing Interface List
    GigabitEthernet0/0/0/2 Flags: F A, Up: 00:38:57
```

- Examine the multicast routing table on the other pod CE router. The CE router output should be similar to the following, taken from Pod 2:

```
PE2#show ip mroute
<…output omitted…>
(*,224.1.2.1), 00:43:24/-, RP 10.0.1.1, flags: B
  Bidir-Upstream: GigabitEthernet0/0, RPF nbr: 192.168.102.20
  Incoming interface list:
    Loopback0, Accepting/Sparse
    GigabitEthernet0/0, Accepting/Sparse

(*,224.1.1.1), 00:43:24/-, RP 10.0.1.1, flags: B
  Bidir-Upstream: GigabitEthernet0/0, RPF nbr: 192.168.102.20
  Incoming interface list:
    Loopback0, Accepting/Sparse
    GigabitEthernet0/0, Accepting/Sparse

(*, 224.0.1.40), 00:43:24/00:02:54, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/0, Forward/Sparse, 00:43:24/00:02:54
```

- Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging to your multicast group address.

```
CE2#ping 224.1.1.1 repeat 100 source GigabitEthernet0/0
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.102.21
```

```
Reply to request 0 from 10.1.10.1, 4 ms
Reply to request 0 from 10.1.1.1, 4 ms
Reply to request 0 from 10.1.10.1, 4 ms
Reply to request 0 from 10.1.1.1, 4 ms
```

- Examine the multicast routing table on the other pod CE router again. The CE router output should be similar to the following, taken from Pod 2:

```
PE2#show ip mroute
<…output omitted…>
(*,224.1.2.1), 00:43:24/-, RP 10.0.1.1, flags: B
  Bidir-Upstream: GigabitEthernet0/0, RPF nbr: 192.168.102.20
  Incoming interface list:
    Loopback0, Accepting/Sparse
    GigabitEthernet0/0, Accepting/Sparse

(*,224.1.1.1), 00:43:24/-, RP 10.0.1.1, flags: B
  Bidir-Upstream: GigabitEthernet0/0, RPF nbr: 192.168.102.20
  Incoming interface list:
    Loopback0, Accepting/Sparse
    GigabitEthernet0/0, Accepting/Sparse

(*, 224.0.1.40), 00:43:24/00:02:54, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/0, Forward/Sparse, 00:43:24/00:02:54
```


Lab 5-3: Implement Rendezvous Point Distribution

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this activity, you will use PIM-SM. Avoiding the configuration of static RP information, you will choose the Auto-RP solution as a dynamic mechanism for RP announcement. You will then configure the standard bootstrap mechanism as an alternative to Auto-RP to verify the redundant setup of BSR routers and RPs.

Finally, you will configure two RPs with the same IP address, sharing the same range of groups. This action will create the Anycast RP solution that will require a simple MSDP configuration.

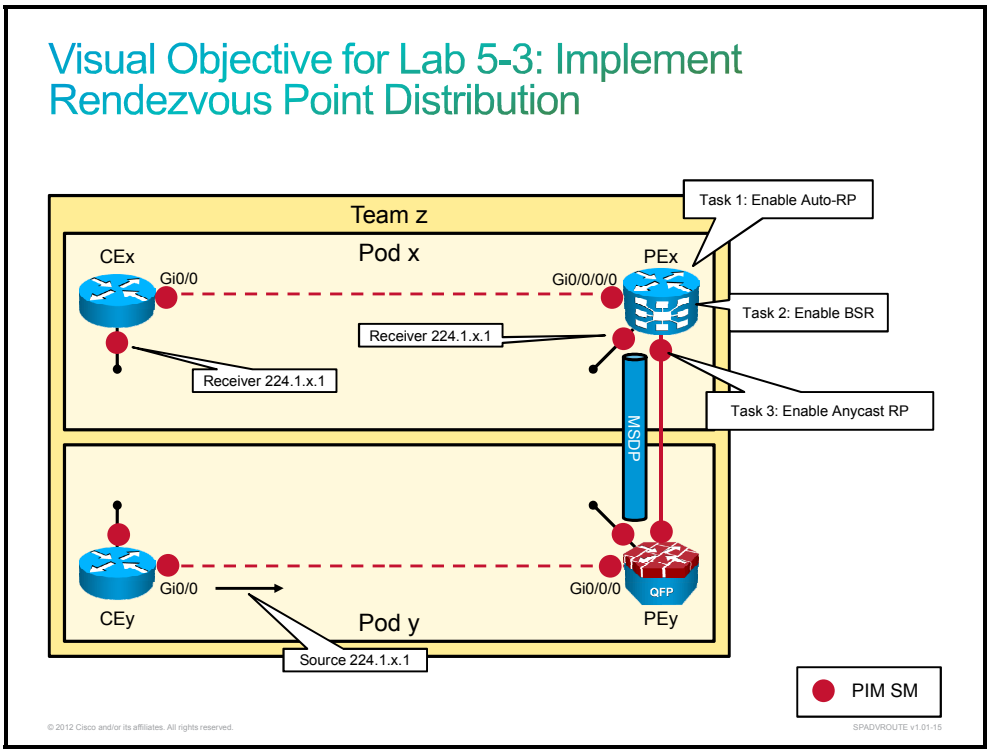
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router that is running Cisco IOS XR Software, and the second pod in the same team will work on the PE router that is running Cisco IOS XE Software. Students in the same team should coordinate their activities.

In the lab activity, you will work on different Cisco routers that are running Cisco IOS (c2900), Cisco IOS XE (asr1001), and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Enable Auto-RP
- Enable BSR
- Enable Anycast RP

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Software Commands

Command	Description
clear ip pim rp-mapping	Clears the group-to-RP mapping table
configure terminal	Enters configuration mode
interface *interface*	Enters interface configuration mode
ip igmp join-group *group_address* [**source** *source*]	Configures an interface on the router to join the specified group or channel
ip msdp peer *IP-address* **connect-source** *interface*	Enables an MSDP peer
ip pim bsr-candidate *interface*	Enables a candidate BSR
ip pim rp-address *RP_address*	Statically configures the address of a PIM RP for multicast groups
ip pim rp-candidate *interface*	Enables a BSR candidate RP
ip pim send-rp-announce *interface* **scope** *scope*	Enables an Auto-RP candidate RP
ip pim send-rp-discovery *interface* **scope** *scope*	Enables an Auto-RP mapping agent
ip pim sparse-mode	Enables PIM-SM on the interface
ip router isis	Enables IS-IS on the interface
isis circuit-type level-1	In interface configuration mode, enables IS-IS Level-1 circuit type
isis circuit-type level-2-only	In interface configuration mode, enables IS-IS Level-2 circuit type
is-type level-1	Enables an IS-IS router to run in the Level-1 mode only
neighbor *peer-address* **shutdown**	Disables BGP neighbor adjacency
net *net-address*	Configures a NET address in IS-IS router configuration mode
ping *dest_IP source source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
router bgp *AS-number*	Enables the BGP routing protocol and enters router BGP routing protocol configuration mode
router isis	Enables the IS-IS routing protocol and enters router IS-IS routing protocol configuration mode
show ip mroute	Displays the contents of the multicast routing (mroute) table
show ip msdp peer	Displays MSDP peer adjacency information
show ip pim rp mapping	Displays group-to-RP mapping table

Cisco IOS XR Software Commands

Command	Description
address-family ipv4	Enters IPv4 address family under specific configuration mode
auto-rp candidate-rp *interface* **scope** *scope*	In router PIM configuration mode, enables an Auto-RP candidate RP
auto-rp mapping-agent *interface* **scope** *scope*	In router PIM configuration mode, enables an Auto-RP mapping agent
bsr candidate-bsr *IP-address*	In router PIM configuration mode, enables a candidate BSR
bsr candidate-rp *IP-address*	In router PIM configuration mode, enables a BSR candidate RP
circuit-type level-1	In IS-IS router configuration mode, enables an IS-IS Level-1 circuit type on the interface
circuit-type level-2-only	In IS-IS router configuration mode, enables an IS-IS Level-2 circuit type on the interface
clear pim bsr | **autorp**	Clears the BSR or Auto-RP group-to-RP mapping table
commit	Commits changes to the running configuration
configure	Enters configuration mode
connect-source *interface*	Enables an MSDP source interface
enable	In router PIM configuration mode, enables PIM-SM on the interface
interface *interface*	Enters interface configuration mode
join-group *group_address* **[source** *source***]**	Configures an interface on the router to join the specified group or channel
multicast-routing	Enters multicast routing configuration mode
net *net-address*	Configures a NET address in IS-IS router configuration mode
peer *IP-address*	Enables an MSDP peer IP address
ping *dest_IP source source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
router bgp *AS-number*	Enables the BGP routing protocol and enters router BGP routing protocol configuration mode
router igmp	Enters IGMP configuration mode
router isis *process-ID*	Enables the IS-IS routing protocol and enters router IS-IS routing protocol configuration mode
router msdp	Enters MSDP configuration mode
router pim	Enters PIM configuration mode
rp-address *RP_address*	Statically configures the address of a PIM RP for multicast groups
show mrib group	Displays the contents of the multicast routing (mroute) table
show mrib route	Displays the contents of the multicast routing (mroute) table
show msdp peer	Displays MSDP peer adjacency information
show pim group-map	Displays group-to-RP mapping table
shutdown	Disables an interface

Task 1: Enable Auto-RP

In this task, you will configure and verify Auto-RP.

Activity Procedure

Complete these steps:

Step 1 On the pod PE router, disable both interfaces toward the core devices (P1 and P2 routers).

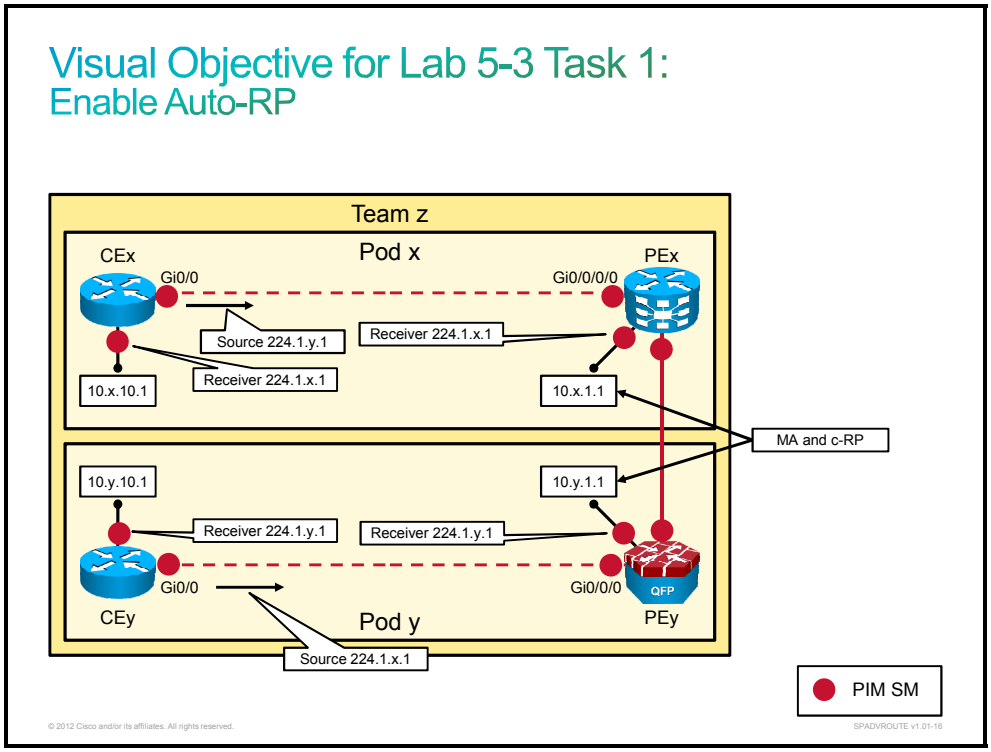
Step 2 On the pod CE and PE routers, make sure that IS-IS is enabled. On the pod CE router, shut down the BGP neighbor. Use the following NET addresses for IS-IS:

Router	NET Address
CEx	49.000x.0100.0x01.0001.00
CEy	49.000y.0100.0y01.0001.00

Step 3 On the pod CE router, make sure that PIM-SM is enabled on the Loopback0 and first Gigabit Ethernet interfaces. On the pod PE router, make sure PIM-SM is enabled on the Loopback0 interface and first and second Gigabit Ethernet interfaces.

Step 4 On the pod PE router, use the Loopback0 interface to configure the Auto-RP mapping agent and RP candidate. The pod PE router running Cisco IOS XE will not forward Auto-RP packets across sparse mode interfaces by default. To get a consistent view across Auto-RP mapping agents on all CE routers in the team, configure the pod PE router running Cisco IOS XE to allow Auto-RP packets to cross sparse mode interfaces.

The figure shows what you will accomplish in this task.



Activity Verification

You have completed this task when you attain these results:

- On the pod PE router, verify that RP-to-group mapping information was obtained from Auto-RP mapping agents. This output is taken from Team 1:

```
RP/0/RSP0/CPU0:PE1#show pim group-map
Thu Nov 24 10:11:36.584 UTC

IP PIM Group Mapping Table
(* indicates group mappings being used)
(+ indicates BSR group mappings active in MRIB)

Group Range          Proto Client   Groups RP address      Info

224.0.1.39/32*       DM    perm     1      0.0.0.0
224.0.1.40/32*       DM    perm     1      0.0.0.0
224.0.0.0/24*        NO    perm     0      0.0.0.0
232.0.0.0/8*         SSM   config   0      0.0.0.0
224.0.0.0/4*         SM    autorp   1      10.2.1.1         RPF:
Gi0/0/0/1,192.168.112.20
224.0.0.0/4          SM    autorp   0      10.1.1.1         RPF: Null,0.0.0.0
224.0.0.0/4          SM    static   0      0.0.0.0          RPF: Null,0.0.0.0
!
PE2#show ip pim rp mapping
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)
This system is a candidate RP (v2)
This system is an RP-mapping agent (Loopback0)

Group(s) 224.0.0.0/4
  RP 10.2.1.1 (?), v2v1
    Info source: 10.2.1.1 (?), elected via Auto-RP
         Uptime: 00:01:46, expires: 00:02:11
  RP 10.1.1.1 (?), v2
    Info source: 10.1.1.1 (?), via Auto-RP
         Uptime: 00:01:47, expires: 00:02:08
```

- On the pod CE router, verify that RP-to-group mapping information was obtained from the Auto-RP mapping agents. All routers in the team should have same group-to-RP mapping information. In the output, the RP with IP address 10.2.1.1 is mapped to the 224.0.0.0/4 multicast groups. This output is taken from Team 1:

```
CE1#show ip pim rp mapping
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 10.2.1.1 (?), v2v1
    Info source: 10.1.1.1 (?), elected via Auto-RP
         Uptime: 00:03:35, expires: 00:02:23
  RP 10.1.1.1 (?), v2
    Info source: 10.2.1.1 (?), via Auto-RP
         Uptime: 00:02:35, expires: 00:00:20
!
CE2#show ip pim rp mapping
```



```
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 10.2.1.1 (?), v2v1
    Info source: 10.1.1.1 (?), elected via Auto-RP
         Uptime: 00:03:39, expires: 00:02:17
  RP 10.1.1.1 (?), v2
    Info source: 10.2.1.1 (?), via Auto-RP
         Uptime: 00:02:39, expires: 00:00:17
```

- Verify that the pod CE and PE router Loopback0 interfaces are joined to multicast group 224.1.x.1 or 224.1.y.1 (where x or y is your pod number).

CE1 (Cisco IOS Software):

```
interface Loopback0
 ip igmp join-group 224.1.1.1
```

PE1 (Cisco IOS XR Software):

```
router igmp
 interface Loopback0
  join-group 224.1.1.1
```

CE2 (Cisco IOS Software):

```
interface Loopback0
 ip igmp join-group 224.1.2.1
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 ip igmp join-group 224.1.2.1
```

- From the neighbor pod CE router, ping to the multicast group configured on your pod CE and PE routers. You should get responses from both pod routers.

```
CE2#ping 224.1.1.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds:

Reply to request 0 from 10.1.1.1, 4 ms
Reply to request 0 from 10.1.10.1, 4 ms
!
CE1#ping 224.1.2.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.1.2.1, timeout is 2 seconds:

Reply to request 0 from 10.2.1.1, 1 ms
Reply to request 0 from 10.2.10.1, 4 ms
```

Task 2: Enable BSR

In this task, you will configure and verify BSR.

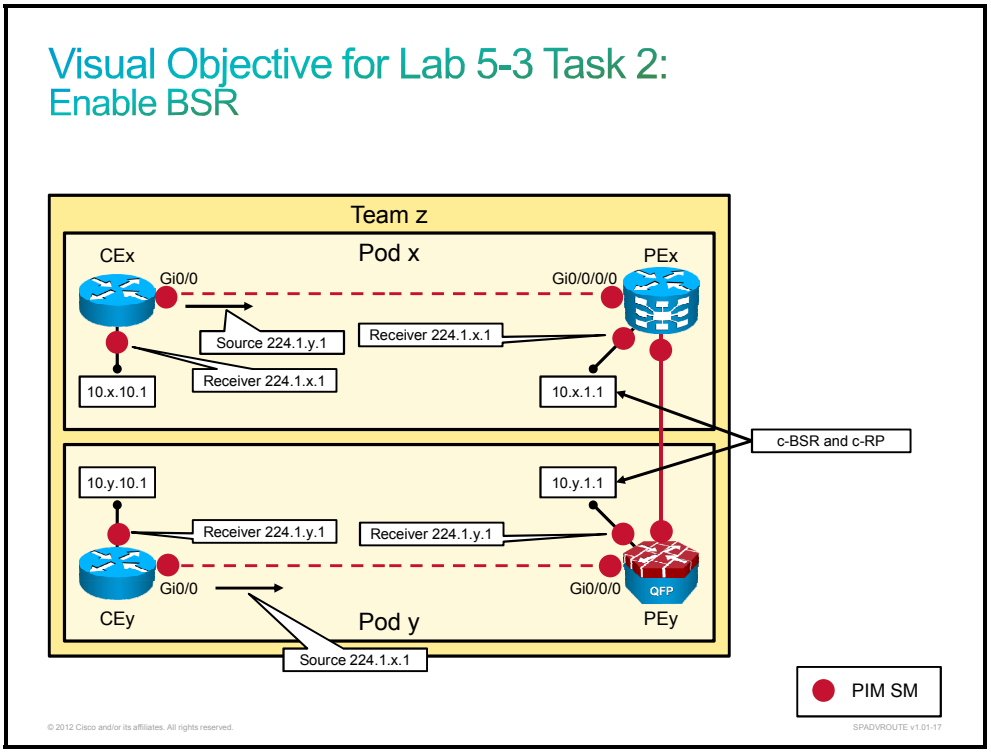
Activity Procedure

Complete these steps:

Step 1 On the pod PE router, remove the Auto-RP mapping agent and Auto-RP RP candidate configuration.

Step 2 On the pod PE router, use the Loopback0 interface to configure a BSR candidate and BSR RP candidate.

The figure shows what you will accomplish in this task.



Activity Verification

You have completed this task when you attain these results:

- Before you continue with the verification, clear group-to-RP mapping table on the pod CE and PE router. The **clear ip pim rp-mapping** command should be entered on the Cisco IOS/IOS XE router and the **clear pim autorp** command should be entered on the Cisco IOS XR router.
- On the pod PE router, verify that group-to-RP mapping information was obtained from the BSR. This output is taken from Team 1:

```
RP/0/RSP0/CPU0:PE1#show pim group-map
Thu Nov 24 10:40:51.748 UTC

IP PIM Group Mapping Table
(* indicates group mappings being used)
(+ indicates BSR group mappings active in MRIB)

Group Range          Proto Client   Groups RP address      Info

224.0.1.39/32*       DM    perm     0      0.0.0.0
224.0.1.40/32*       DM    perm     1      0.0.0.0
224.0.0.0/24*        NO    perm     0      0.0.0.0
232.0.0.0/8*         SSM   config   0      0.0.0.0
224.0.0.0/4*         SM    bsr+     1      10.2.1.1         RPF:
Gi0/0/0/1,192.168.112.20
224.0.0.0/4          SM    bsr      0      10.1.1.1         RPF: Null,0.0.0.0
224.0.0.0/4          SM    static   0      0.0.0.0          RPF: Null,0.0.0.0
!
PE2#show ip pim rp mapping
```

```
PIM Group-to-RP Mappings
This system is a candidate RP (v2)

Group(s) 224.0.0.0/4
  RP 10.2.1.1 (?), v2
    Info source: 10.1.1.1 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:00:09, expires: 00:02:17
  RP 10.1.1.1 (?), v2
    Info source: 10.1.1.1 (?), via bootstrap, priority 192, holdtime 150
         Uptime: 00:00:09, expires: 00:02:16
```

- On the pod CE router, verify that group-to-RP mapping information was obtained from the BSR. All routers in the team should have same group-to-RP mapping information. In the output, the RP with IP address 10.2.1.1 and better priority 0 is mapped to the 224.0.0.0/4 multicast groups. This output is taken from Team 1:

```
CE1#show ip pim rp mapping
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 10.2.1.1 (?), v2
    Info source: 10.1.1.1 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:01:17, expires: 00:02:08
  RP 10.1.1.1 (?), v2
    Info source: 10.1.1.1 (?), via bootstrap, priority 192, holdtime 150
         Uptime: 00:01:17, expires: 00:02:08
!
CE2#show ip pim rp mapping
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 10.2.1.1 (?), v2
    Info source: 10.1.1.1 (?), via bootstrap, priority 0, holdtime 150
         Uptime: 00:01:23, expires: 00:02:03
  RP 10.1.1.1 (?), v2
    Info source: 10.1.1.1 (?), via bootstrap, priority 192, holdtime 150
         Uptime: 00:01:23, expires: 00:02:03
```

- On the pod CE router, verify that the RP with the better (lower number) priority is used in multicast routing:

```
CE1#show ip mroute | include RP
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
(*, 224.1.1.1), 00:41:31/00:02:27, RP 10.2.1.1, flags: SJCL
  Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.101.10
```

- From the neighbor pod CE router, ping to the multicast group configured on your pod CE and PE routers. You should get responses from both pod routers.

```
CE2#ping 224.1.1.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds:
Reply to request 0 from 10.1.1.1, 4 ms
Reply to request 0 from 10.1.10.1, 32 ms
!
CE1#ping 224.1.2.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 224.1.2.1, timeout is 2 seconds:
```

```
Reply to request 0 from 10.2.1.1, 1 ms
Reply to request 0 from 10.2.10.1, 24 ms
```

Task 3: Enable Anycast RP

In this task, you will configure and verify Anycast RP.

Activity Procedure

Complete these steps:

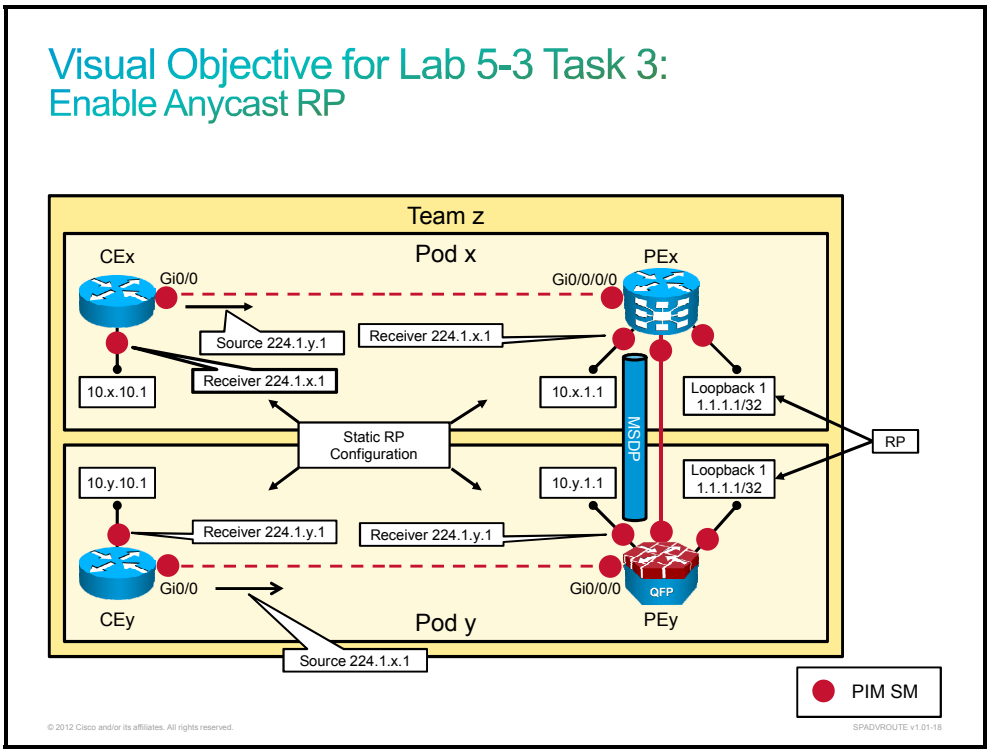
Step 1 On the pod PE router, remove the BSR candidate and BSR RP candidate configuration.

Step 2 On the pod PE router, enable the Loopback1 interface and assign IP address 1.1.1.1/32. Make sure that the Loopback1 subnet is announced to the pod CE router via the IS-IS routing protocol. On the newly configured Loopback1 interface, enable PIM-SM.

Step 3 On the pod CE and PE routers, configure 1.1.1.1 as a static RP.

Step 4 Between the pod PE and neighbor pod PE routers, establish MSDP adjacency. Use the PE router Loopback0 IP address as the source address of MSDP packets and as the originator ID.

The figure shows what you will accomplish in this task.



Activity Verification

You have completed this task when you attain these results:

- Before you continue with the verification, clear the group-to-RP mapping table on the pod CE and PE routers. The **clear ip pim rp-mapping** command should be entered on the Cisco IOS/IOS XE router, and the **clear pim bsr** command should be entered on the Cisco IOS XR router.
- On the pod PE router, verify that an MSDP session was established with the neighbor pod PE router. This output is taken from Team 1:

```
RP/0/RSP0/CPU0:PE1#show msdp peer
Fri Dec  9 09:59:11.908 UTC
MSDP Peer 10.2.1.1 (?), AS 0
Description:
  Connection status:
    State: Up, Resets: 1, Connection Source: 10.1.1.1
    Uptime(Downtime): 00:44:01, SA messages received: 0
    TLV messages sent/received: 89/44
  Output messages discarded: 0
    Connection and counters cleared 00:44:01 ago
  SA Filtering:
    Input (S,G) filter: none
    Input RP filter: none
    Output (S,G) filter: none
    Output RP filter: none
  SA-Requests:
    Input filter: none
    Sending SA-Requests to peer: disabled
  Password: None
  Peer ttl threshold: 0
  Input queue size: 0, Output queue size: 0
  KeepAlive timer period: 30
  Peer Timeout timer period: 75
!
PE2#show ip msdp  peer
MSDP Peer 10.1.1.1 (?), AS ?
  Connection status:
    State: Up, Resets: 0, Connection source: Loopback0 (10.2.1.1)
    Uptime(Downtime): 00:02:20, Messages sent/received: 2/6
    Output messages discarded: 0
    Connection and counters cleared 00:02:35 ago
  SA Filtering:
    Input (S,G) filter: none, route-map: none
    Input RP filter: none, route-map: none
    Output (S,G) filter: none, route-map: none
    Output RP filter: none, route-map: none
  SA-Requests:
    Input filter: none
  Peer ttl threshold: 0
  SAs learned from this peer: 1
  Number of connection transitions to Established state: 1
    Input queue size: 0, Output queue size: 0
  MD5 signature protection on MSDP TCP connection: not enabled
```

```
  Message counters:
    RPF Failure count: 0
    SA Messages in/out: 3/0
    SA Requests in: 0
    SA Responses out: 0
    Data Packets in/out: 1/0
```

- From the neighbor pod CE router, ping to the multicast group that is configured on your pod CE and PE routers.

```
CE2#ping 224.1.1.1 source Loopback0 repeat 100
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.2.10.1

Reply to request 0 from 10.1.10.1, 1 ms
Reply to request 0 from 10.1.10.1, 1 ms
Reply to request 1 from 10.1.10.1, 1 ms
Reply to request 1 from 10.1.10.1, 1 ms
```

- Verify the MSDP SA cache on the PE routers:

```
RP/0/RSP0/CPU0:PE1#show msdp sa-cache
Fri Dec  9 10:09:10.124 UTC

MSDP Flags:
E - set MRIB E flag , L - domain local source is active,
EA - externally active source, PI - PIM is interested in the group,
DE - SAs have been denied.  Timers age/expiration,
Cache Entry:
(10.2.10.1, 224.1.1.1), RP 10.2.1.1, MBGP/AS 0, 00:00:19/00:02:19
  Learned from peer 10.2.1.1, RPF peer 10.2.1.1
  SAs recvd 2, Encapsulated data received: 100
    grp flags: PI,  src flags: E, EA, PI
!
PE2#show ip msdp sa-cache
MSDP Source-Active Cache - 1 entries
(10.1.10.1, 224.1.1.10), RP 10.1.1.1, AS ?,00:07:22/00:02:26, Peer 10.1.1.1
```


Lab 6-1: Implement a DHCPv6 Server with Prefix Delegation

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will configure the PE router act as a prefix delegation DHCPv6 server. You will configure the PE router to delegate a prefix to the CE router. You will then configure the CE router interface with an IPv6 address from the delegated prefix. You will also configure the CE router to act as a DHCPv6 Lite server, with DNS server IP address options obtained from the PE router. This is a valid scenario, where the service provider would assign a prefix to a CE router using prefix delegation, and the CE router would then advertise the prefix to LAN endpoints to enable stateless autoconfiguration.

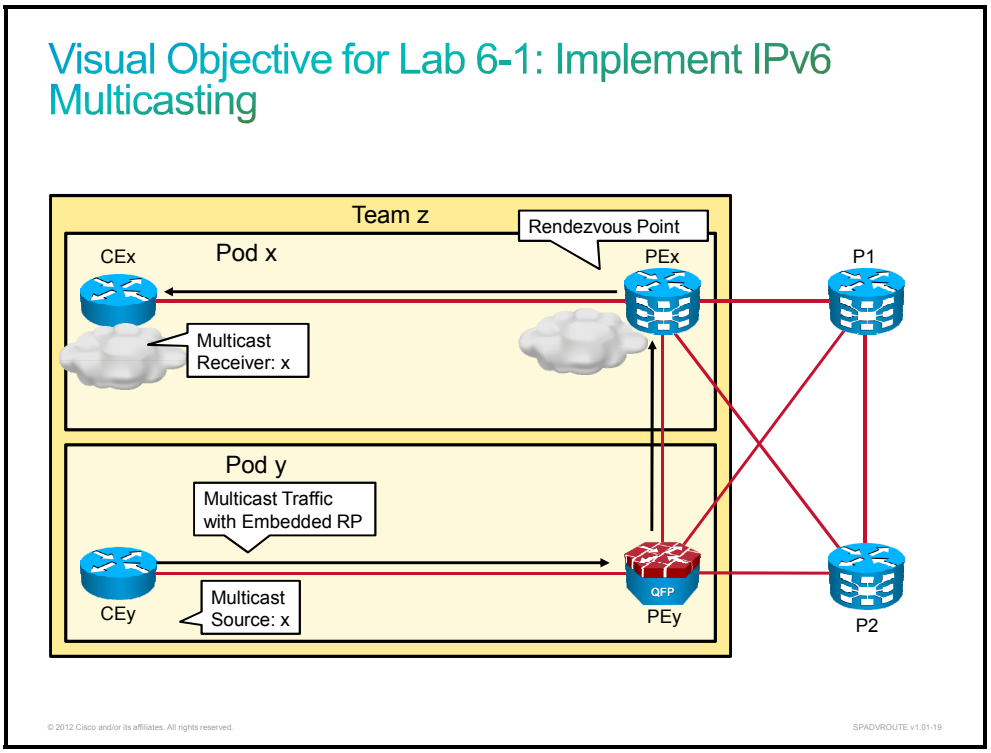
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router running Cisco IOS XR Software, and the second pod in the same team will work on the PE router running Cisco IOS XE Software. Students in the same team should coordinate their activities.

You will work on different Cisco routers running Cisco IOS (c2900) Software, Cisco IOS XE (asr1001) Software, and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet these objectives:

- Implement a prefix delegation DHCPv6 server
- Implement a DHCPv6 client
- Implement a DHCPv6 Lite server

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Commands

Command	Description
configure terminal	Enters configuration mode
dns-server *IPv6_address*	Specifies the DNS IPv6 servers available to a DHCP server for IPv6 client
import dns-server	Imports the DNS name server option into DHCP server for IPv6 client
interface *interface*	Enters interface configuration mode
ipv6 address autoconfig	Enables automatic configuration of IPv6 addresses using stateless autoconfiguration on an interface and enables IPv6 processing on the interface
ipv6 address *prefix_name suffix /prefix-length*	Configures an IPv6 address based on an IPv6 general prefix and enables IPv6 processing on an interface
ipv6 dhcp client pd *prefix_name*	Enables DHCP for IPv6 client process and enables requests for prefix delegation through a specified interface
ipv6 dhcp pool *pool_name*	Configures DHCP for IPv6 server configuration information pool and enters DHCP for IPv6 pool configuration mode
ipv6 dhcp server *pool_name*	Enables DHCP server for IPv6 on an interface
ipv6 local pool *prefix_pool_name prefix/prefix_length assigned_length*	Configures a local IPv6 prefix pool
ipv6 nd other-config-flag	Sets the "other stateful configuration" flag in IPv6 router advertisements
ping *dest_ip_source source_interface*	Verifies connectivity between source IP and destination IP
prefix-delegation pool *prefix_pool_name*	Specifies a named IPv6 local prefix pool from which prefixes are delegated to DHCP server for IPv6 clients
show ipv6 dhcp interface	Displays DHCP for IPv6 interface information
show ipv6 dhcp pool	Displays DHCP for IPv6 configuration pool information
show ipv6 interface brief	Displays IPv6 addresses on interfaces and status of interfaces

Cisco IOS XR Commands

Command	Description
dhcp ipv6	Enables DHCP for IPv6 and enters DHCP IPv6 configuration mode
pool *pool_name*	Creates a DHCP pool and enters DHCP pool configuration mode
commit	Commits changes to the running configuration.
configure terminal	Enters configuration mode
dns-server *ipv6_address*	Specifies DNS server for DHCP use
interface *interface*	Enters interface configuration mode
interface *interface* **server**	Enables DHCP server on an interface and enters DHCP interface configuration mode
pd *prefix/prefix_length*	Specifies IPv6 prefix for delegation using DHCP
pool *pool_name*	Assigns DHCP pool to DHCP enabled interface
ping *dest_IP source source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
show dhcp ipv6 binding	Displays DHCP bindings for IPv6

Task 1: Configure a Prefix Delegation DHCPv6 Server and Client

In this task, you will configure a prefix delegation DHCPv6 server on the PE router.

Activity Procedure

Complete these steps:

Step 1 Ping the directly connected interface of the CE router using IPv6. You should be successful.

Step 2 Access the PE router. Configure the PE router as a prefix delegation DHCPv6 server with the following parameters:

Pod	Delegated Prefix	DNS Server
Pod x	2001:db8:100:X::/64	2001:db8:100::X
Pod y	2001:db8:100:Y::/64	2001:db8:100::X

Enable the DHCPv6 server on the interface that is facing the CE router.

Note Note that configuration of DHCPv6 on Cisco IOS XR Software is significantly different from its configuration on Cisco IOS and IOS XE Software.

Step 3 Access the CE router. Enable a DHCPv6 prefix delegation client on the interface that is facing the PE router. Use **SP_ASSIGNED_PREFIX** as the name of the delegated prefix.

Step 4 Assign the first IP address from the delegated prefix to the GigabitEthernet0/1 interface.

Step 5 Verify the assigned IP address on the CE GigabitEthernet0/1 interface.

Step 6 Ping a nonexistent host name from the CE router. This will trigger DNS name lookup and you will be able to verify whether a DNS server has been assigned via DHCP.

Step 7 Return to the PE router. Verify DHCP bindings.

Activity Verification

You have completed this task when you attain these results:

- Ping the CE router directly connected interface using IPv6. You should be successful. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#ping 2001:DB8:192:168:101::11
Fri Nov 11 08:08:01.817 UTC
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:db8:192:168:101::11, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/9 ms
```

- Verify the assigned IP address on the CE GigabitEthernet0/1 interface. On the CE router, the output should be similar to the following, taken from Pod 1:

```
CE1#show ipv6 interface brief
Embedded-Service-Engine0/0 [administratively down/down]
    unassigned
GigabitEthernet0/0         [up/up]
    FE80::4255:39FF:FE84:4A70
    2001:DB8:192:168:101::11
GigabitEthernet0/1         [up/up]
    FE80::4255:39FF:FE84:4A71
    2001:DB8:100:1::1
<…output omitted…>
```

- Ping a nonexistent host name. On the CE router, the output should be similar to the following, taken from Pod 1:

```
CE1#ping asfdg
Translating "asfdg"...domain server (2001:DB8:100::1)
<…output omitted…>
```

- Verify DHCP bindings on the PE router. On the PE router running Cisco IOS XR Software, the output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show dhcp ipv6 binding
Client: fe80::4255:39ff:fe84:4a70 (GigabitEthernet0/0/0/0)
  DUID: 00030001405539844a70
  IA PD: IA ID 0x00030001, T1 302400, T2 483840
    Prefix: 2001:db8:100:1::/64
            preferred lifetime 604800, valid lifetime 2592000
            expires at Dec 11 2011 07:57 (2591251 seconds)
```

Task 2: Configure DHCPv6 Lite Server

In this task, you will configure the CE router, which also acts as DHCPv6 client, as DHCPv6 Lite server. The CE router will advertise DNS server IP address, which will be obtained from the PE router, to LAN clients. LAN clients will obtain IPv6 addresses using stateless autoconfiguration.

Activity Procedure

Complete these steps:

Step 1 Return to the CE router. Configure a DHCPv6 pool. The DNS server should be imported as received from the PE router.

Step 2 Enable the DHCPv6 server on the GigabitEthernet0/1 interface. Configure the router to instruct DHCP clients not to use DHCP for address assignments. However, clients should obtain other parameters, such as DNS server, using DHCP.

Step 3 Verify configured DHCP pools on the CE router.

Step 4 Verify configured DHCP on the CE router. You should see that one interface acts as the DHCP client and the other acts as the DHCP server.

Note Coordinate your activities with the other pod if you would like to test the DHCP server on the CE router. Complete the following optional steps to verify the DHCP server on the CE router.

Step 5 Remove the previously configured IP address from the other pod CE router GigabitEthernet0/1 interface. Configure the interface to obtain an IP address via stateless autoconfiguration.

Step 6 Verify the IPv6 address on the other pod CE router GigabitEthernet0/1 interface. You should see the IP address that is combined with the prefix advertised by the DHCP server and suffix generated by EUI-64.

Step 7 Ping a nonexistent host name from the other pod CE router. This will trigger DNS name lookup and you will be able to verify whether a DNS server has been assigned via DHCP. You should see that the router has been assigned with two DNS servers. One has been assigned by the PEy router directly, and one has been assigned by the CEx router acting as the DHCPv6 Lite server.

Activity Verification

You have completed this task when you attain these results:

- Verify configured DHCP pools. On the CE router, the output should be similar to the following, taken from Pod 1:

```
CE1#show ipv6 dhcp pool
DHCPv6 pool: POOL
  Imported DNS server: 2001:DB8:100::1
  Active clients: 0
```

- Verify how DHCP is enabled on interfaces. On the CE router, the output should be similar to the following, taken from Pod 1:

```
CE1#show ipv6 dhcp interface
GigabitEthernet0/0 is in client mode
  Prefix State is OPEN
  Renew will be sent in 3d10h
  Address State is IDLE
  List of known servers:
    Reachable via address: FE80::4255:39FF:FE2E:7D80
    DUID: 0003000140553931959E
    Preference: 0
    Configuration parameters:
      IA PD: IA ID 0x00030001, T1 302400, T2 483840
        Prefix: 2001:DB8:100:1::/64
```



```
                preferred lifetime 604800, valid lifetime 2592000
                expires at Dec 11 2011 08:00 AM (2586659 seconds)
      DNS server: 2001:DB8:100::1
      Information refresh time: 0
  Prefix name: SP_ASSIGNED_PREFIX
  Prefix Rapid-Commit: disabled
  Address Rapid-Commit: disabled
GigabitEthernet0/1 is in server mode
  Using pool: POOL
  Preference value: 0
  Hint from client: ignored
  Rapid-Commit: disabled
```

- Verify the IPv6 address on the other pod CE router GigabitEthernet0/1 interface. On the CE router, the output should be similar to the following, taken from Pod 2:

```
CE2#show ipv6 interface brief
Embedded-Service-Engine0/0 [administratively down/down]
    unassigned
GigabitEthernet0/0         [up/up]
    FE80::EAB7:48FF:FE2C:A330
    2001:DB8:192:168:102::21
GigabitEthernet0/1         [up/up]
    FE80::EAB7:48FF:FE2C:A331
    2001:DB8:100:1:EAB7:48FF:FE2C:A331
```

- Ping a nonexistent host name from the other pod CE router. On the CE router, the output should be similar to the following, taken from Pod 2:

```
CE2#ping wf
Translating "wf"...domain server (2001:DB8:100::2) (2001:DB8:100::1)
<…output omitted…>
```

Lab 6-2: Implement IPv6 Multicasting

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will first implement IPv6 multicast using embedded RPs. Your pod CE router will act as the multicast receiver, while the other pod CE router will act as multicast source. Your pod PE router will act as RP.

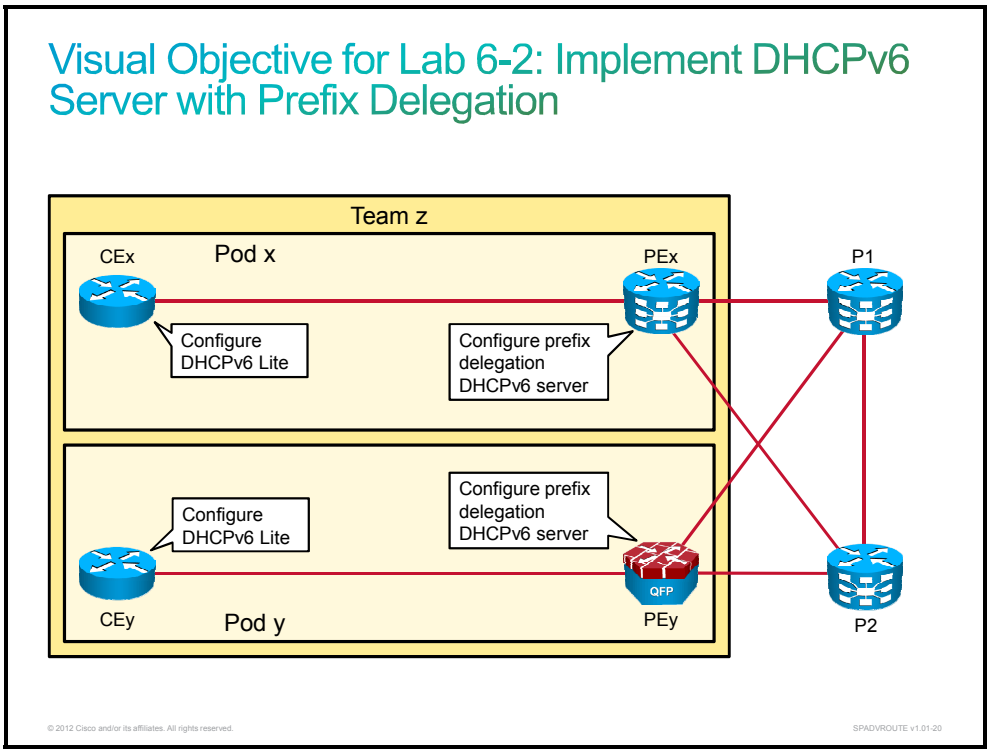
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. The first pod in the team will work on the PE router running Cisco IOS XR Software, and the second pod in the same team will work on the PE router running Cisco IOS XE software. Students in the same team should coordinate their activities.

You will work on different Cisco routers running Cisco IOS (c2900), Cisco IOS XE (asr1001), and Cisco IOS XR (asr9k) Software. After completing this activity, you will be able to meet this objective:

- Implement IPv6 multicast using embedded RPs

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Commands

Command	Description
configure terminal	Enters configuration mode
interface *interface*	Enters interface configuration mode
ipv6 address *ipv6_address/prefix*	Assigns an IPv6 address to an interface
ipv6 igmp join-group *group_address*	Configures an interface on the router to join the specified group or channel
ipv6 multicast-routing	Enables IPv6 multicast routing
ipv6 pim rp-address *RP_address*	Statically configures the address of a PIM RP for multicast groups
ipv6 pim spt-threshold *threshold*	Configures when a PIM leaf router should join the shortest path source tree
ipv6 router isis	Enables IS-IS routing protocol on an interface and advertises a network on the interface
ping *dest_IP* **source** *source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
show ip protocols	Displays configured routing protocols for IPv4
show ipv6 mroute	Displays the contents of the multicast routing (mroute) table
show ipv6 pim interface	Displays information about interfaces that are configured for PIM
show ipv6 pim neighbor	Displays information about PIM neighbors discovered by PIMv1 router query messages or PIMv2 hello messages
show ipv6 protocols	Displays configured routing protocols for IPv6
show ipv6 route	Displays a routing table on a router
show isis neighbors	Displays IS-IS neighbors

Cisco IOS XR Commands

Command	Description
address-family ipv6	Enters IPv6 address family under specific configuration mode Enables an interface for a specified address family under IS-IS configuration mode
commit	Commits changes to the running configuration
configure	Enters configuration mode
embedded-rp *RP_address access_list_name*	Statically configures the address of an embedded RP for multicast groups under PIM configuration mode

Command	Description
enable	Enables an interface for multicast routing or PIM (under the appropriate configuration mode)
interface *interface*	Enters interface configuration mode
ipv6 access-list *acl_name*	Creates an ACL and enters access list configuration mode
multicast-routing	Enters multicast routing configuration mode
permit | **deny** *protocol source_address destination_address*	Creates an entry in an ACL
ping *dest_IP* **source** *source_IP*	Verifies connectivity between source IP and destination IP (IPv4 and IPv6)
router isis	Enters IS-IS configuration mode
router pim	Enters PIM configuration mode
show mrib ipv6 route	Displays the contents of the multicast routing (mroute) table
show pim ipv6 interface	Displays information about interfaces that are configured for PIM
show pim ipv6 neighbor	Displays information about PIM neighbors discovered by PIMv1 router query messages or PIMv2 hello messages
spt-threshold infinity	Configures that a PIM leaf router should join the shortest path source tree immediately

Task 1: Create a New Loopback Interface and Verify Connectivity

In this task, you will configure a new loopback interface on the PE router that will act as the RP for your pod.

Activity Procedure

Complete these steps:

Step 1 On the PE router, create a new loopback interface. Use **x0** (or **y0**) as the interface identifier. This interface will be used as RP for your pod.

Step 2 Assign the following IP address to the loopback interface. Advertise the loopback interface using the IS-IS routing protocol.

Router	IPv6 Address
PEx	2001.db8:x:x::1/128
PEy	2001.db8:y:y::1/128

Step 3 On the CE router, make sure that the GigabitEthernet0/0 and Loopback0 interfaces are enabled for IS-IS and that an adjacency is established with the PE router. Make sure that IS-IS is enabled for IPv4 and IPv6.

Step 4 Verify the connectivity between the CE and PE routers by pinging the created loopback interface from the CE router. Use the Loopback0 interface as a source interface.

Step 5 On the CE router, verify that the loopback interface of the new PE has been learned through IS-IS. If the loopback interface was learned through BGP, shut down the BGP IPv6 neighbor on the CE router.

Activity Verification

You have completed this task when you attain these results:

- On the CE router, make sure that the GigabitEthernet0/0 interface is enabled for IS-IS and that an adjacency is established with the PE router. Make sure that IS-IS is enabled for IPv4 and IPv6. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show isis neighbors

Tag null:
System Id      Type Interface   IP Address      State Holdtime Circuit Id
PE5            L1   Gi0/0       192.168.101.10  UP    22       CE1.02
PE5            L2   Gi0/0       192.168.101.10  UP    22       CE1.02

CE1#show ip protocols
<…output omitted…>
Routing Protocol is "isis"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: isis
  Address Summarization:
    None
  Maximum path: 4
  Routing for Networks:
    GigabitEthernet0/0
    Loopback0
<…output omitted…>

CE1#show ipv6 protocols
<…output ommitted…>
IPv6 Routing Protocol is "isis"
  Interfaces:
    Loopback0
    GigabitEthernet0/0
<…output ommitted…>
```

- Ping the directly connected interface of the CE router using IPv6. You should be successful. The PE router output should be similar to the following, taken from Pod 1:

```
CE1#ping 2001:db8:1:1::1 source 2001:DB8:10:1:10::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:1:1::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:10:1:10::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/4 ms
```

- On the CE router, verify that the PE loopback interface has been learned through IS-IS. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show ipv6 route 2001:DB8:1:1::1
Routing entry for 2001:DB8:1:1::1/128
  Known via "isis", distance 115, metric 20, type level-1
  Route count is 1/1, share count 0
```

```
Routing paths:
  FE80::4255:39FF:FE2E:7D80, GigabitEthernet0/0
    Last updated 19:22:30 ago
```

Task 2: Implement IPv6 Multicast Using Embedded RPs

In this task, you will implement IPv6 multicast on the PE router and CE routers. The pod PE router will act as RP. The multicast group address will carry the RP IP address information.

Activity Procedure

Complete these steps:

Step 1 On the CE router, enable IPv6 multicast routing. Configure the router to always stay on the shared tree.

Step 2 On the PE router, enable IPv6 multicast routing. On the PE router running Cisco IOS XR Software, make sure that you enable PIM on all relevant interfaces, including the newly created loopback interface. Configure the router to always stay on the shared tree.

Step 3 Verify the PIM state for IPv6 on interfaces on the PE router.

Step 4 Verify the PIM neighbors for IPv6 on the PE router.

Step 5 On the PE router, manually configure the RP address. Loopbackx0 (or Loopbacky0) will act as an RP for your pod. On the PE router running Cisco IOS XR Software, you also have to provide multicast groups that will be served by that RP using an ACL.

Note Manual RP configuration on other routers is not needed because the other routers will learn the RP information from the multicast group IP address.

Both pods from the same team should be finished with the previous steps at this point.

Step 6 Calculate the multicast group addresses that can be served by RP assigned to your pod. Use site-local multicast group addresses. Complete the following table:

Router	RP Address	Group Addresses
PEx	2001.db8:x:x::1/128	
PEy	2001.db8:y:y::1/128	

Step 7 Answer the following question:

Step 8 How many multicast groups can an embedded RP serve?

_____

Step 9 Return to the CE router. Enable Loopback0 as a multicast receiver for one of the multicast groups that is served by your RP (for example, FF75:0140:2001:db8:x:x::1).

Note Throughout the lab exercise, use the **ipv6 mld join-group** command on Cisco IOS and IOS-XE Software under interface configuration mode to simulate multicast receivers.

Step 10 Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging your multicast group address. Use the Loopback0 interface as a source interface. Send a large number of ICMP packets—100, for example. You should see that your CE router replies to the ping.

Step 11 Examine the multicast routing table on the PE router.

Activity Verification

You have completed this task when you attain these results:

- Verify the PIM state for IPv6 on the interfaces on the PE router. The PE router output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show pim ipv6 interface
PIM interfaces in VRF default
Interface                  PIM  Nbr   Hello  DR
                                Count Intvl  Prior

Loopback0                   on   1     30     1
    Primary Address : fe80::6cd1:a6ff:fe93:dce6
                 DR : this system

Loopback10                  on   1     30     1
    Primary Address : fe80::6cd1:a6ff:fe93:dce6
                 DR : this system

GigabitEthernet0/0/0/0      on   2     30     1
    Primary Address : fe80::4255:39ff:fe2e:7d80
                 DR : fe80::4255:39ff:fe84:4a70

GigabitEthernet0/0/0/1      on   2     30     1
    Primary Address : fe80::4255:39ff:fe2e:7d81
                 DR : fe80::eab7:48ff:fefb:7101
```

- Verify the PIM neighbors for IPv6 on the PE router. The PE router running Cisco IOS XR Software output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show pim ipv6 neighbor
PIM neighbors in VRF default
Loopback0

Neighbor Address                        Uptime    Expires  DR pri  DR  Flags
fe80::6cd1:a6ff:fe93:dce6*              05:52:40  00:01:37 1      (DR) B P

Loopback50

Neighbor Address                        Uptime    Expires  DR pri  DR  Flags
fe80::6cd1:a6ff:fe93:dce6*              01:21:35  00:01:29 1      (DR) B P

GigabitEthernet0/0/0/0

Neighbor Address                        Uptime    Expires  DR pri  DR  Flags
fe80::4255:39ff:fe2e:7d80*              05:52:40  00:01:16 1           B P
fe80::4255:39ff:fe84:4a70               05:52:39  00:01:25 1      (DR) B

GigabitEthernet0/0/0/1
```

```
Neighbor Address                          Uptime    Expires  DR pri  DR  Flags
fe80::4255:39ff:fe2e:7d81*                05:52:40  00:01:43 1           B P
fe80::eab7:48ff:fefb:7101                 05:52:40  00:01:15 1       (DR) B
```

The PE router running Cisco IOS XE Software output should be similar to the following, taken from Pod 2:

```
PE2#show ipv6 pim neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, G - GenID Capable
Neighbor Address           Interface          Uptime    Expires  Mode DR pri

FE80::EAB7:48FF:FE2C:A330  Gi0/0/0            05:54:07  00:01:21 B G     1
FE80::4255:39FF:FE2E:7D81  Gi0/0/1            05:53:31  00:01:23 B G     1
```

- Use Telnet to connect to the other pod CE router. Start the multicast traffic by pinging your multicast group address. Use the Loopback0 interface as a source interface. Send large numbers of ICMP packets—100, for example. You should see that your CE router replies to the ping. On the CE router, the output should be similar to the following, taken from Pod 2:

```
CE2#ping ff75:0140:2001:db8:1:1::1 repeat 50 source 2001:DB8:10:2:10::1
Output Interface: Loopback0
Type escape sequence to abort.
Sending 50, 100-byte ICMP Echos to FF75:140:2001:DB8:1:1:0:1, timeout is 2
seconds:
Packet sent with a source address of 2001:DB8:10:2:10::1

Request 0 timed out
Request 1 timed out
Reply to request 2 received from 2001:DB8:10:1:10::1, 0 ms
<…output omitted…>
```

- Examine the multicast routing table on the PE router. On the PE router running Cisco IOS XR Software, the output should be similar to the following, taken from Pod 1:

```
RP/0/RSP0/CPU0:PE1#show mrib ipv6 route
<…output omitted…>
(*,ff75:140:2001:db8:1:1::/96)
  RPF nbr: 2001:db8:1:1::1 Flags: L C
  Up: 01:38:46
  Outgoing Interface List
    Decaps6tunnel3 Flags: NS DI, Up: 01:25:13

(*,ff75:140:2001:db8:1:1:0:1)
  RPF nbr: 2001:db8:5:5::1 Flags: C
  Up: 01:38:46
  Incoming Interface List
    Decaps6tunnel3 Flags: A, Up: 01:25:13
  Outgoing Interface List
    GigabitEthernet0/0/0/0 Flags: F NS, Up: 01:38:46

(2001:db8:10:2:10::1,ff75:140:2001:db8:1:1:0:1)
  RPF nbr: fe80::eab7:48ff:fefb:7101 Flags: L
  Up: 00:01:28
  Incoming Interface List
```



```
    GigabitEthernet0/0/0/1 Flags: A, Up: 00:01:28
  Outgoing Interface List
    GigabitEthernet0/0/0/0 Flags: F NS, Up: 00:01:28
```

The PE router running Cisco IOS XE Software output should be similar to the following, taken from Pod 2:

```
PE2#show ipv6 mroute
<…output omitted…>
(*, FF75:140:2001:DB8:2:2:0:1), 00:00:45/00:02:44, RP 2001:DB8:2:2::1, flags:
S
  Incoming interface: Tunnel2
  RPF nbr: 2001:DB8:2:2::1
  Immediate Outgoing interface list:
    GigabitEthernet0/0/0, Forward, 00:00:45/00:02:44

(2001:DB8:10:1:10::1, FF75:140:2001:DB8:2:2:0:1), 00:00:16/00:03:17, flags: ST
  Incoming interface: GigabitEthernet0/0/1
  RPF nbr: FE80::4255:39FF:FE2E:7D81
  Inherited Outgoing interface list:
    GigabitEthernet0/0/0, Forward, 00:00:45/00:02:44
```

Lab 6-3: Implement Tunnels for IPv6

Complete this lab activity to practice what you learned in the related module.

Activity Objective

In this lab activity, you will first establish a static IPv6-in-IPv4 tunnel between two CE routers in different pods in the same team. Then you will deploy 6RD tunnels and configure a 6RD CE router and 6RD Border Relay.

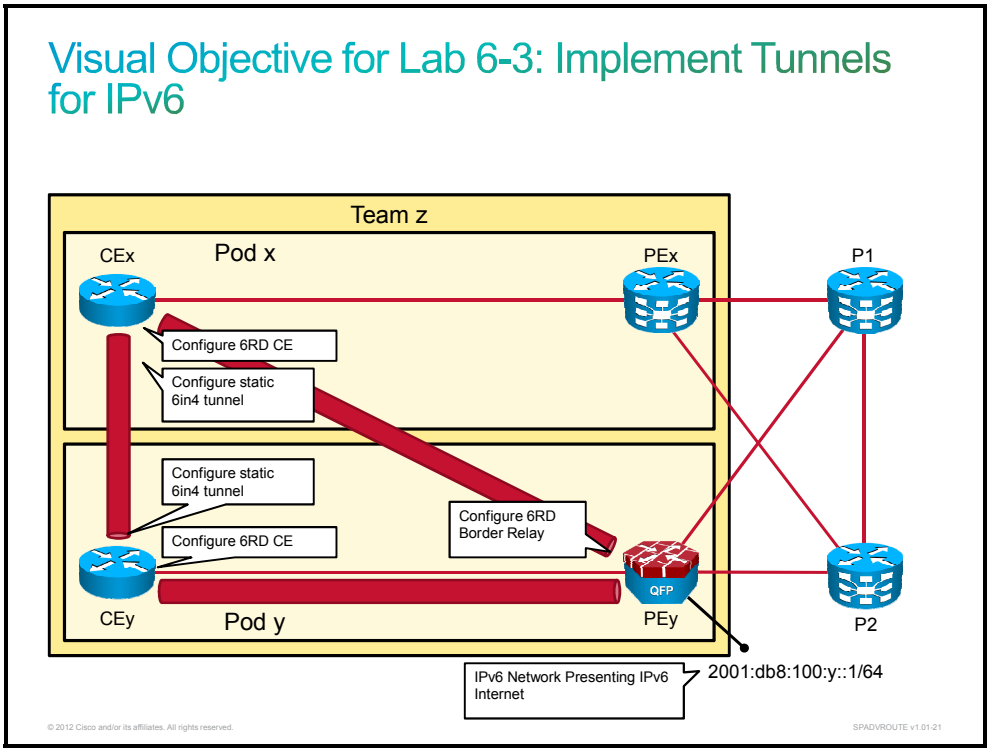
Note	Students from two different pods work as a team. The CE routers in both pods are running Cisco IOS Software. In this activity, students will work on CE routers. Students from both pods will work on the PE router from the second pod only. Students in the same team should coordinate their activities.

You will work on Cisco routers running Cisco IOS (c2900) and Cisco IOS XE (asr1001) Software. After completing this activity, you will be able to meet these objectives:

- Configure a static IPv6-in-IPv4 tunnel
- Configure 6RD

Visual Objective

The figure illustrates what you will accomplish in this activity.



Command List

The table describes the commands that are used in this lab activity.

Cisco IOS/IOS XE Commands

Command	Description
configure terminal	Enters configuration mode
ipv6 address *IPv6_address*	Configures IPv6 address on an interface
interface *interface*	Enters interface configuration mode
ipv6 enable	Enables an interface for IPv6
interface tunnel *id*	Creates a tunnel interface and enters interface configuration mode
tunnel source *interface*	Specifies tunnel source
tunnel destination *ip_address*	Specifies tunnel destination
tunnel mode ipv6ip	Sets tunnel mode to IPv6-in-IPv4
tunnel mode ipv6ip 6rd	Sets tunnel mode to 6RD
tunnel 6rd ipv4 prefix-len *length*	Sets a common prefix length for 6RD
tunnel 6rd prefix *prefix*	Sets a 6RD prefix
tunnel 6rd br	Sets a 6RD Border Relay IP address
ipv6 route *prefix/length outgoing_interface next_hop_IP_***address**	Configures a static IPv6 route
ping *dest_ip_source source_interface*	Verifies connectivity between source IP and destination IP
router bgp *as_number*	Enters BGP configuration mode
network *network* **mask** *mask*	Advertises a network into BGP
show interfaces *interface*	Displays interface information and traffic statistics
show tunnel 6rd	Displays information about 6RD tunnels

Task 1: Configure a Static IPv6-in-IPv4 Tunnel

In this task, you will establish a static IPv6-in-IPv4 tunnel between two CE routers in different pods in the same team.

Activity Procedure

Complete these steps:

Step 1 Access the CE router. Ping the other pod CE router using IPv4. Pings should be sourced from the Loopback0 interface and destined to the Loopback0 interface. You should be successful.

Step 2 Ping the other pod CE router using IPv6. Pings should be sourced from the Loopback0 interface and destined to the Loopback0 interface. You should not be successful.

Step 3 Create a tunnel interface on the CE router. Enable IPv6 on the interface. Use link-local IPv6 addresses for tunnel interface addressing.

Note If you are unable to create the tunnel interface because of PIM registering, disable PIM on CE and PE routers on all relevant interfaces.

Step 4 Specify the Loopback0 interface as the tunnel source. Specify the IP address of the other pod CE router Loopback0 interface as the tunnel destination.

Step 5 Set the tunnel mode to IPv6-in-IPv4.

Step 6 Create a static IPv6 route for other pod CE router Loopback0 interface that will point to the tunnel interface.

Step 7 Ping the other pod CE router using IPv6. Pings should be sourced from the Loopback0 interface and destined to the Loopback0 interface. You should be successful.

Note The other pod should be finished with the configuration of the CE router for the ping to be successful.

Step 8 Verify that pings went over the tunnel interface by examining the tunnel interface traffic statistics. You should see that five packets went into and out of the tunnel interface.

Step 9 Remove the tunnel interface from the CE router. Remove the static route that was created in this task as well.

Activity Verification

You have completed this task when you attain these results:

- Ping the other pod CE router using IPv4. Pings should be sourced from the Loopback0 interface and destined to the Loopback0 interface. You should be successful. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#ping 10.2.10.1 source Loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.10.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.10.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```


- Ping the other pod CE router using IPv6. Pings should be sourced from the Loopback0 interface and destined to the Loopback0 interface. You should not be successful. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#ping 2001:DB8:10:2:10::1 source Loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:10:2:10::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:10:1:10::1
.....
Success rate is 0 percent (0/5)
```

- Ping the other pod CE router using IPv6. Pings should be sourced from the Loopback0 interface and destined to the Loopback0 interface. You should be successful. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#ping 2001:DB8:10:2:10::1 source Loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:10:2:10::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:10:1:10::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/4 ms
```

- Verify that pings went over the tunnel interface by examining the tunnel interface traffic statistics. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show interfaces Tunnel0
Tunnel0 is up, line protocol is up
<…output omitted…>
     5 packets input, 700 bytes, 0 no buffer
      Received 0 broadcasts (0 IP multicasts)
      0 runts, 0 giants, 0 throttles
      0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
      5 packets output, 600 bytes, 0 underruns
      0 output errors, 0 collisions, 0 interface resets
      0 unknown protocol drops
      0 output buffer failures, 0 output buffers swapped out
```

Task 2: Configure Dynamic 6RD Tunnels

In this task, you will deploy 6RD tunnels and configure a 6RD CE router and 6RD Border Relay. PE routers running Cisco IOS XE Software will act as 6RD Border Relay and CE routers will act as 6RD CE routers. You will configure 6RD tunnels between existing loopback interfaces.

Activity Procedure

Complete these steps:

Note	Coordinate this task with the other pod, because only one PE router will act as a 6RD Border Relay. Recall that 6RD is not supported on routers running Cisco IOS XR Software.

Step 1 Access the PEy router (Cisco ASR1001). Advertise the network configured on the Loopback0 interface through BGP.

Step 2 Access the CE router. Ping the PEy Loopback0 interface. Use Loopback0 as source interface. You should be successful.

Step 3 Answer the following questions:

How long is a prefix that is common to CEx, CEy, and PEy loopback interfaces?

_____

How many bits from the IPv4 address will be used to construct 6RD networks?

_____

Step 4 Construct 6RD networks for IPv6 sites behind the CE routers. Use 2001:db8:aa00::/40 as the 6RD prefix and the last three octets of the IPv4 address on the loopback interface. Verify the results with the other pod in the team. Fill in the following table:

CE Router	IPv4 Tunnel Endpoint	6RD Prefix	6RD Network
CEx	10.x.10.1	2001:db8:aa00::/40	
CEy	10.y.10.1	2001:db8:aa00::/40	

Step 5 Construct a 6RD network for the PEy router as well. This network will be used later to configure a default route on the CE routers. Verify the results with the other pod in the team.

PE Router	IPv4 Tunnel Endpoint	6RD Prefix	6RD Network
PEy	10.y.1.1	2001:db8:aa00::/40	

Step 6 Return to the CE router. Configure the tunnel interface with the following parameters:

- Enable the tunnel interface for IPv6
- Tunnel source: Loopback0
- Tunnel mode: 6rd
- Common prefix: 8 bits
- 6RD prefix: 2001:db8:aa00::/40
- 6RD BR: 10.y.1.1

Step 7 Verify information about the 6RD tunnel on the CE router. Compare the displayed 6RD network with the network that you calculated in the previous steps.

Step 8 Assign the first IPv6 address from the 6RD network to a new loopback interface. Use **10** as the interface identifier.

Step 9 Create a static route for the 6RD prefix that will use the tunnel interface as the outgoing interface.

Step 10 Return to the PEy router. Configure the tunnel interface with the following parameters:

- Enable the tunnel interface for IPv6
- Tunnel source: Loopback0
- Tunnel mode: 6rd
- Common prefix: 8 bits
- 6RD prefix: 2001:db8:aa00::/40

Step 11 Verify the information about the 6RD tunnel on the PEy router. Compare the displayed 6RD network with the network that you calculated in the previous steps.

Step 12 Create another loopback interface on the PEy router. Use **10** as the interface identifier and **2001:db8:100:y::1** as the IP address on the interface. This interface will present the IPv6 Internet that is available over the 6RD Border Relay.

Step 13 Return to the CE router. Create a default route that will use the tunnel interface as the outgoing interface and will point to the 6RD network of the PEy router.

Step 14 From the CE router, ping the other CE router Loopback10 interface. Use Loopback10 as a source interface. You should be successful and traffic should go directly to the other CE router.

Note The other pod should be done with the configuration of CE router for the ping to be successful.

Step 15 From the CE router, ping the PEy router Loopback10 interface. Use Loopback10 as a source interface. You should be successful and traffic should go directly to the PEy router.

Step 16 Examine traffic statistics on the tunnel interface. You should see a number of packets going over the interface. The number should correspond to the number of pings sent to the other routers.

Activity Verification

You have completed this task when you attain these results:

- Ping the PEy Loopback0 interface from the CE router. Use Loopback0 as the source interface. You should be successful. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#ping 10.2.1.1 source Loopback0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.1.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.10.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

- Verify information about the 6RD tunnel on the CE router. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#show tunnel 6rd Tunnel0
Interface Tunnel0:
  Tunnel Source: 10.1.10.1
  6RD: Operational, V6 Prefix: 2001:DB8:AA00::/40
       V4 Prefix, Length: 8, Value: 10.0.0.0
       V4 Suffix, Length: 0, Value: 0.0.0.0
       Border Relay address: 10.2.1.1
  General Prefix: 2001:DB8:AA01:A01::/64
```

- Verify information about the 6RD tunnel on the PEy router. The PEy router output should be similar to the following, taken from Pod 2:

```
PE2#show tunnel 6rd
Interface Tunnel0:
  Tunnel Source: 10.2.1.1
  6RD: Operational, V6 Prefix: 2001:DB8:AA00::/40
       V4 Prefix, Length: 8, Value: 10.0.0.0
```

```
      V4 Suffix, Length: 0, Value: 0.0.0.0
  General Prefix: 2001:DB8:AA02:101::/64
```

- From the CE router, ping the other CE router Loopback10 interface. You should be successful and traffic should go directly to the other CE router. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#ping 2001:DB8:AA02:A01::1 source Loopback10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:AA02:A01::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:AA01:A01::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/1/4 ms
```

- From the CE router, ping the PEy router Loopback10 interface. You should be successful and traffic should go directly to the PEy router. The CE router output should be similar to the following, taken from Pod 1:

```
CE1#ping 2001:DB8:100:2::1 source Loopback10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:100:2::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:AA01:A01::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/4 ms
```

- Examine traffic statistics on the tunnel interface. You should see a number of packets going over the interface. The number of packets should correspond to the number of pings sent to the other routers.

```
CE1#show interfaces Tunnel0
Tunnel0 is up, line protocol is up
<…output omitted…>
  5 minute output rate 0 bits/sec, 0 packets/sec
     10 packets input, 1400 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     10 packets output, 1200 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
```


Answer Key

The correct answers and expected solutions for the lab activities that are described in this guide appear here.

Lab 2-1 Answer Key: Implement BGP Route Reflectors

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Verify Existing BGP Sessions

There are no solutions available in this task.

Task 2: Configure a Route Reflector and Internal BGP Session

Step 1 On the PE router, enable the interface and IS-IS routing toward the P1 router:

PE1 (Cisco IOS XR):

```
interface GigabitEthernet0/0/0/3
 ipv4 address 192.168.12.10 255.255.255.0
 ipv6 address 2001:db8:192:168:12::10/80
 no shutdown
!
router isis 1
 interface GigabitEthernet0/0/0/3
  circuit-type level-2-only
  address-family ipv4 unicast
  address-family ipv6 unicast
commit
```

PE2 (Cisco IOS XE):

```
interface GigabitEthernet0/0/3
 ip address 192.168.22.20 255.255.255.0
 ipv6 enable
 ipv6 address 2001:db8:192:168:22::20/80
 ip router isis
 ipv6 router isis
 isis circuit-type level-2-only
```

Step 2 The BGP configuration on the PE routers is as follows:

PE1 (Cisco IOS XR):

```
router bgp 64500
 neighbor 10.0.2.1
  remote-as 64500
  update-source Loopback0
  address-family ipv4 unicast
commit
```

PE2 (Cisco IOS XE):

```
router bgp 64500
 neighbor 10.0.2.1 remote-as 64500
 neighbor 10.0.2.1 update-source Loopback0
```

Step 3 The BGP next-hop-self on the PE router is as follows:

PE1 (Cisco IOS XR):

```
router bgp 64500
 neighbor 10.0.2.1
  address-family ipv4 unicast
   next-hop-self
!
commit
```

PE2 (Cisco IOS XE):

```
router bgp 64500
 neighbor 10.0.2.1 next-hop-self
```

Step 4 The BGP route reflector configuration on the P2 router is as follows:

```
router bgp 64500
 bgp cluster-id 10.0.1.1
 neighbor 10.1.1.1
  remote-as 64500
  update-source Loopback0
  address-family ipv4 unicast
   route-reflector-client
commit
```

Step 5 Which BGP path selection criterion is being used to select the path to the other pod route? The lowest router ID among the sending routers.

Task 3: (Optional) Restrict Route Propagation to a Client

Step 1 Create a route policy on the P2 router:

```
route-policy FILTER_TO_CLIENT_POD5
  if as-path originates-from '64502'  then
    pass
  endif
end-policy
commit
```

Step 2 On the P2 router, apply the route policy to the IBGP session with the pod PE router in the outbound direction:

```
router bgp 64500
 neighbor 10.1.1.1
  address-family ipv4 unicast
   route-policy FILTER_TO_CLIENT_POD5 out
commit
```

Lab 3-1 Answer Key: Implement BGP Security Options

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Implement BGP Neighbor Authentication Using Passwords

Step 6 Enable BGP neighbor authentication on the CE router:

```
router bgp 64501
 neighbor 192.168.101.10 password C!sc()
```

Step 7 Enable BGP neighbor authentication on the PE router:

PE1 (Cisco IOS XR):

```
router bgp 64500
 neighbor 192.168.101.11
  password C!sc()
commit
```

PE2 (Cisco IOS XE):

```
router bgp 64500
 neighbor 192.168.101.11 password C!sc()
```

Task 2: Implement BGP Neighbor Authentication Using Key Chains

Step 2 Configure a key chain on the PE router:

PE1 (Cisco IOS XR):

```
key chain BGP
 key 1
  accept-lifetime 00:00:00 january 01 2011 infinite
  key-string password 143453180F4C63
  send-lifetime 00:00:00 january 01 2011 infinite
  cryptographic-algorithm HMAC-MD5
commit
```

Step 3 Apply the key chain on the PE router to the IBGP session with the P1 router:

PE1 (Cisco IOS XR):

```
router bgp 64500
 neighbor 10.0.1.1
  keychain BGP
commit
```

Step 4 Configure a key chain on the P1 router:

```
key chain BGP_POD1
 key 1
  accept-lifetime 00:00:00 january 01 2011 infinite
  key-string password 143453180F4C63
  send-lifetime 00:00:00 january 01 2011 infinite
  cryptographic-algorithm HMAC-MD5
commit
```

Step 5 Apply the key chain on the P1 router to the IBGP session with the PE router:

```
router bgp 64500
 neighbor 10.1.1.1
  keychain BGP
commit
```

Task 3: Enable BGP TTL Security Check

Step 3 Enable a TTL security check for the EBGP session on the CE router. What is the number that you have to specify with the command to enforce that EBGP neighbors are directly connected?

1, since the neighbors are directly connected

```
router bgp 64501
 neighbor 192.168.101.10 ttl-security hops 1
```

Step 4 On the PE router, enable TTL security check:

PE1 (Cisco IOS XR):

```
router bgp 64500
 neighbor 192.168.101.11
  ttl-security
commit
```

PE2 (Cisco IOS XE):

```
router bgp 64500
 neighbor 192.168.101.11 ttl-security hops 1
```

Task 4: (Optional) Enable CoPP

Step 1 On the CE router, configure a named ACL that will permit BGP traffic from the PE to the CE router:

```
ip access-list extended BGP_TRAFFIC
 permit tcp host 192.168.101.10 host 192.168.101.11 eq bgp
 permit tcp host 192.168.101.10 eq bgp host 192.168.101.11
```

Step 2 On the CE router, create a class map that will refer to the previously configured ACL:

```
class-map match-all BGP_CLASS
 match access-group name BGP_TRAFFIC
```

Step 3 On the CE router, create a policy map that will rate-limit BGP traffic from the PE to the CE router to 200 packets per second:

```
policy-map COPP
 class BGP_CLASS
   police rate 200 pps conform-action transmit exceed-action drop
```

Step 4 On the CE router, apply the configured policy map to the control plane virtual interface using service policy:

```
control-plane
 service-policy input COPP
```

Task 5: (Optional) Enable RTBH Filtering

Step 2 On the CE router, create the Loopback1 interface:

```
interface Loopback1
 ip address 10.1.100.1 255.255.255.255
```

Step 2 On the CE router, advertise the previously configured /32 network on the loopback interface into BGP:

```
router bgp 64501
 network 10.1.100.1 mask 255.255.255.255
```

Step 4 On the PE router, create a static route for 172.16.x.0/24 (or 172.16.y.0/24) network that points to the null0 interface:

PE1 (Cisco IOS XR):

```
router static
 address-family ipv4 unicast
   172.16.1.0/24 Null0
commit
```

PE2 (Cisco IOS XE):

```
ip route 172.16.2.0/24 Null0
```

Step 5 On the PE router, enable strict uRPF on the CE-facing interface:

PE1 (Cisco IOS XR):

```
interface GigabitEthernet0/0/0/0
```



```
 ipv4 verify unicast source reachable-via rx
commit
```

PE2 (Cisco IOS XE):

```
interface GigabitEthernet0/0/0
 ip verify unicast source reachable-via rx
```

Step 3 Use Telnet to connect to the P1 router. Create a static route for 172.16.x.0/24 (or 172.16.y.0/24) network that points to the null0 interface:

```
router static
 address-family ipv4 unicast
  172.16.1.0/24 Null0
commit
```

Step 9 Answer the following questions:

Why are the redistributed routes tagged with no-export community?

The black-holed routes should not be advertised outside the AS; therefore, they are tagged with no-export community.

Why is local preference of redistributed routes set to 1000?

To prefer the existing route for the same network that has been reflected from the R2 route reflector and that points to CE router on the PE router

Step 10 On the P1 router, trigger black-holing of traffic originating from the CE Loopback1 interface:

```
router static
 address-family ipv4 unicast
  10.1.100.1/32 Null0 tag 5
commit
```

Lab 3-2 Answer Key: Improve BGP Scalability

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Implement BGP Configuration and Peer Templates

Step 4 Configure an address family and neighbor group on the PE router:

PE1 (Cisco IOS XR):

```
router bgp 64500
 af-group IPV4 address-family ipv4 unicast
  route-policy PASS in
  route-policy PASS out
 !
 neighbor-group EBGP
  password C!sc()
   ttl-security
   address-family ipv4 unicast
    use af-group IPv4
 !
 no neighbor 192.168.101.11
```

```
 neighbor 192.168.101.11
  remote-as 64501
  use neighbor-group EBGP
!
commit
```

Step 5 Configure peer session and peer policy templates on the PE router:

PE2 (Cisco IOS XE):

```
router bgp 64500
 template peer-policy EBGP_POLICY
 exit-peer-policy
 !
 template peer-session EBGP_SESSION
  password C!sc()
  ttl-security hops 2
 exit-peer-session
 !
 no neighbor 192.168.102.21 remote-as 64502
 neighbor 192.168.102.21 remote-as 64502
 neighbor 192.168.102.21 inherit peer-session EBGP_SESSION
 !
 address-family ipv4
  neighbor 192.168.102.21 inherit peer-policy EBGP_POLICY
 exit-address-family
```

Task 2: Limit Number of Prefixes Received from a BGP Neighbor

Step 6 On the PE router, enable the maximum prefix feature for routes received from EBGP neighbors:

PE1 (Cisco IOS XR):

```
router bgp 64500
 af-group IPV4 address-family ipv4 unicast
  maximum-prefix 2
commit
```

PE2 (Cisco IOS XE):

```
router bgp 64500
 template peer-policy EBGP_POLICY
  maximum-prefix 2
```

Task 3: Improve BGP Convergence by Changing BGP Scan and Advertisement Interval

Step 2 On the PE router, set the scan interval to 30 seconds:

PE1 (Cisco IOS XR):

```
router bgp 64500
 bgp scan-time 30
commit
```

PE2 (Cisco IOS XE):

```
router bgp 64500
 bgp scan-time 30
```

Step 5 On the CE router, set the advertisement interval for the PE neighbor to 15 seconds:

```
router bgp 64501
```



```
 neighbor 192.168.101.10 advertisement-interval 15
```

Task 4: Improve BGP Convergence by Enabling BFD

Step 1 Configure the switch port connecting the CE router (FastEthernet0/1) to be in another VLAN:

```
interface FastEthernet0/1
 switchport access vlan 5
```

Step 3 Return to the SW switch and put the FastEthernet0/1 switch port back into VLAN 1:

```
interface FastEthernet0/1
 switchport access vlan 1
```

Step 5 On the CE router, enable BFD for the PE neighbor:

```
interface GigabitEthernet0/0
 bfd interval 100 min_rx 100 multiplier 3
!
router bgp 64501
 neighbor 192.168.101.10 fall-over bfd
```

Step 6 On the PE router, enable BFD for the CE neighbor with the following parameters:

PE1 (Cisco IOS XR):

```
router bgp 64500
 bfd minimum-interval 100
 bfd multiplier 3
 neighbor 192.168.101.11
   bfd fast-detect
commit
```

PE2 (Cisco IOS XE):

```
interface GigabitEthernet0/0/0
 bfd interval 100 min_rx 100 multiplier 3
!
router bgp 64500
 neighbor 192.168.101.11 fall-over bfd
```

Step 8 Return to the SW switch and change the VLAN of the FastEthernet0/1 interface:

```
interface FastEthernet0/1
 switchport access vlan 5
```

Step 9 On the SW switch, return the FastEthernet0/1 switch port to the VLAN:

```
interface FastEthernet0/1
 switchport access vlan 1
```

Task 5: Implement BGP Route Dampening

Step 2 On the PE router, enable BGP route dampening with the default parameters:

PE1 (Cisco IOS XR):

```
router bgp 64500
 address-family ipv4 unicast
  bgp dampening
commit
end
debug bgp dampening
```

PE2 (Cisco IOS XE):

```
router bgp 64500
```

```
  address-family ipv4
  bgp dampening
end
debug ip bgp dampening
```

Step 6 How many times did you have to flap the route for the PE router to suppress the route? What is the default suppress penalty, half-life time, and reuse penalty?

Three times. 2000. 15 minutes. 750.

Step 9 Disable BGP dampening debugging on the PE router:

PE1 (Cisco IOS XR):

```
undebug all
```

PE2 (Cisco IOS XE):

```
undebug all
```

Lab 4-1 Answer Key: Implement Layer 2 and Layer 3 Multicast

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Enable IGMP and MLD

Step 1 Enable IPv4 and IPv6 multicast routing:

PE1 (Cisco IOS XR):

```
multicast-routing
 address-family ipv4
  interface GigabitEthernet0/0/0/0
   enable
!
 address-family ipv6
  interface GigabitEthernet0/0/0/0
   enable
!
commit
```

PE2 (Cisco IOS XE):

```
ip multicast-routing distributed
```

Step 6 Configure the IGMP version and query interval:

PE1 (Cisco IOS XR):

```
router igmp
 interface GigabitEthernet0/0/0/0
  version 2
  query-interval 30
!
commit
```

PE2 (Cisco IOS XE):

```
interface GigabitEthernet0/0/0
 ip igmp version 2
 ip igmp query-interval 30
```

Step 7 Enable the MLD router and configure the MLD query interval:

PE1 (Cisco IOS XR):


```
router mld
 interface GigabitEthernet0/0/0/0
  query-interval 60
  router enable
!
commit
```

PE2 (Cisco IOS XE):

```
interface GigabitEthernet0/0/0
 ipv6 mld router
 ipv6 mld query-interval 60
```

Step 8

CE1 (Cisco IOS Software):

```
interface GigabitEthernet0/0
 ip igmp join-group 234.1.1.1
```

CE2 (Cisco IOS Software):

```
interface GigabitEthernet0/0
 ip igmp join-group 234.1.1.1
```

Task 2: Verify IGMP Snooping

Step 9 Disable IGMP snooping:

SW1 and SW2 (Cisco IOS Software):

```
no ip igmp snooping
```

Step 10 Enable IGMP snooping:

SW1 and SW2 (Cisco IOS Software):

```
ip igmp snooping
```

Step 11 Configure the CE router to leave the group:

CE1 (Cisco IOS Software):

```
interface GigabitEthernet0/0
 no ip igmp join-group 234.1.1.1
```

CE2 (Cisco IOS Software):

```
interface GigabitEthernet0/0
 no ip igmp join-group 234.1.1.1
```

Lab 5-1 Answer Key: Enable and Optimize PIM-SM

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Implement PIM-SM

Step 1 Enable IP multicast routing on the PE router:

PE1 (Cisco IOS XR):

```
multicast-routing
 address-family ipv4
  interface Loopback0
   enable
  !
  interface GigabitEthernet0/0/0/0
```

```
   enable
  !
  interface GigabitEthernet0/0/0/1
   enable
  !
  interface GigabitEthernet0/0/0/2
   enable
  !
  interface GigabitEthernet0/0/0/3
   enable
commit
```

PE2 (Cisco IOS XE):

```
ip multicast-routing
```

Step 2 Enable IP multicast routing on the CE router:

```
ip multicast-routing
```

Step 3 Enable PIM-SM on the PE router on all interfaces that have an IP address assigned:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  interface Loopback0
   enable
  !
  interface GigabitEthernet0/0/0/0
   enable
  !
  interface GigabitEthernet0/0/0/1
   enable
  !
  interface GigabitEthernet0/0/0/2
   enable
  !
  interface GigabitEthernet0/0/0/3
   enable
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 ip pim sparse-mode
!
interface GigabitEthernet0/0/0
 ip pim sparse-mode
!
interface GigabitEthernet0/0/1
 ip pim sparse-mode
!
interface GigabitEthernet0/0/2
 ip pim sparse-mode
!
interface GigabitEthernet0/0/3
ip pim sparse-mode
```

Step 12 Enable PIM-SM on the CE router on all interfaces that have an IP address assigned:


```
interface Loopback0
ip pim sparse-mode
!
interface GigabitEthernet0/0
ip pim sparse-mode
```

Step 13 On the PE router, define the SPT threshold as infinity:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  spt-threshold infinity
commit
```

PE2 (Cisco IOS XE):

```
ip pim spt-threshold infinity
```

Step 14 On the CE router, define the SPT threshold as infinity:

```
ip pim spt-threshold infinity
```

Step 15 Manually configure the RP address on the CE and PE router:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  rp-address 10.0.1.1
commit
```

PE2 (Cisco IOS XE):

```
ip pim rp-address 10.0.1.1
```

CE1 (Cisco IOS Software):

```
ip pim rp-address 10.0.1.1
```

Task 2: Shared Tree Formation—Receivers

Step 1 Simulate multicast receivers for group 224.1.x.1 (or 224.1.y.0) on the Loopback0 interface on the CE and PE routers:

PE1 (Cisco IOS XR):

```
router igmp
 interface Loopback0
  join-group 224.1.1.1
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 ip igmp join-group 224.1.2.1
```

CE1 (Cisco IOS Software):

```
interface Loopback0
 ip igmp join-group 224.1.1.1
```

Step 3 Answer the following question and complete the table for the (*,G) entry for your pod:

Why are there no incoming interfaces on the P1 router for the (*,G) entry?

P1 acts as the RP. Since there are no sources active, the RP does not receive any traffic for the group. Therefore, there are no incoming interfaces for the group.

Router	Incoming Interface	OIL
CE	GigabitEthernet0/0	Loopback0
PE	GigabitEthernet0/0/0/2	Loopback0 GigabitEthernet0/0/0/0
P1	None	GigabitEthernet0/0/0/8

Task 3: Shared Tree Formation—Sources

Step 2 Examine the multicast routing table on your CE router. Answer the following questions:

Did the entry for your group change? Why or why not?

The entry did not change. The CE router uses the shared tree to receive multicast traffic, and the active source does not influence the multicast routing table on the router.

Are there any (S,G) entries present for your group? Why or why not?

There are no (S,G) entries present. Because the CE router uses the shared tree, the only entry for the group is (*,G).

Step 3 Use Telnet to connect to the other pod PE router. Examine the multicast routing table. Answer the following questions:

What is the incoming interface for the (S,G) entry for your group?

GigabitEthernet0/0/0

Which interfaces are present in the OIL for the (S,G) entry for your group?

GigabitEthernet0/0/1

Why is the OIL of the (*,G) entry for your group empty?

The PE router would use the (*,G) entry to receive multicast traffic from the RP. Because there no receivers present in the path to the RP through the PE router, the OIL is empty.

Task 4: Switching to the SPT

Step 1 Configure the CE and PE routers to switch to the SPT immediately after the first packet arrives over the shared tree:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  no spt-threshold infinity
commit
```

PE2 (Cisco IOS XE):

```
ip pim spt-threshold 0
```

CE1 (Cisco IOS Software):

```
ip pim spt-threshold 0
```

Step 3 Examine the multicast routing table on your CE router. Answer the following questions:

Are there any (S,G) entries present for your group? Why or why not?

Yes. Because the CE router created the SPT, the (S,G) entry exists.

Step 4 Examine the multicast routing table on your PE router. Answer the following questions:

Are there any (S,G) entries present for your group? Why or why not?

Yes. Because the PE router created the SPT, the (S,G) entry exists.

Which interface is used as incoming interface for the (S,G) entry for your group?

GigabitEthernet0/0/0/1.

Step 5 Remove the simulated multicast receivers from the Loopback0 interface on the CE and PE routers:

Step 6 PE1 (Cisco IOS XR):

```
router igmp
 interface Loopback0
  no join-group 224.1.5.1
commit
```

Step 7 PE2 (Cisco IOS XE):

```
interface Loopback0
 no ip igmp join-group 224.1.5.1
```

Step 8 CE1 (Cisco IOS):

```
interface Loopback0
 no ip igmp join-group 224.1.6.1
```

Lab 5-2 Answer Key: Implement PIM-SM Enhancements

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Implement PIM-SSM

Step 1 Enable PIM-SSM on the PE and CE router. Configure the SSM address range to include the 224.1.x.1 and 224.1.y.1 multicast groups:

PE1 (Cisco IOS XR):

```
ipv4 access-list SSM_POD1
 10 permit ipv4 224.1.1.1
 20 permit ipv4 224.1.2.1
!
multicast-routing
 ssm range SSM_POD1
commit
```

PE2 (Cisco IOS XE):

```
ip access-list standard SSM_POD1
 permit 224.1.1.1
 permit 224.1.2.1
!
ip pim ssm range SSM_POD1
```

CE1 (Cisco IOS Software):

```
ip access-list standard SSM_POD1
 permit 224.1.1.1
```

```
 permit 224.1.2.1
!
ip pim ssm range SSM_POD1
```

Step 4 Simulate multicast receivers for group 224.1.x.1 (or 224.1.y.0) on the Loopback0 interface on the CE and PE router:

PE1 (Cisco IOS XR):

```
router igmp
 interface Loopback0
  join-group 224.1.1.1 source 192.168.102.21
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 ip igmp join-group 224.1.2.1 source 192.168.101.11
```

CE1 (Cisco IOS Software):

```
interface Loopback0
 ip igmp join-group 224.1.1.1 source 192.168.102.21
```

Step 5 Examine the multicast routing table on the PE router. Answer the following questions:

What is the incoming interface for the (S,G) entry for your group?

GigabitEthernet0/0/0/1

What is the OIL for the (S,G) entry for your group?

GigabitEthernet0/0/0/1, Loopback0

Step 6 Use Telnet to connect to the other pod PE router. Examine the multicast routing table and answer the following questions:

What is the incoming interface for the (S,G) entry for your group?

GigabitEthernet0/0/0

What is the OIL for the (S,G) entry for your group?

GigabitEthernet0/0/1

Step 8 Remove the simulated multicast receivers for group 224.1.x.1 (or 224.1.y.0) from the Loopback0 interface on the CE and PE router:

PE1 (Cisco IOS XR):

```
router igmp
 interface Loopback0
  no join-group 224.1.1.1
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 no ip igmp join-group 224.1.2.1
```

CE1 (Cisco IOS Software):

```
interface Loopback0
 no ip igmp join-group 224.1.1.1
```

Step 9 Disable PIM-SSM on the PE and CE routers:

PE1 (Cisco IOS XR):

```
multicast-routing
```



```
 no ssm range SSM_POD1
!
no ipv4 access-list SSM_POD1
commit
```

PE2 (Cisco IOS XE):

```
no ip pim ssm range SSM_POD1
!
no ip access-list standard SSM_POD1
```

CE1 (Cisco IOS Software):

```
no ip pim ssm range SSM_POD1
!
no ip access-list standard SSM_POD1
```

Task 2: Implement BIDIR-PIM

Step 1 Enable BIDIR-PIM on the CE and PE routers:

PE1 (Cisco IOS XR):

```
ipv4 access-list BIDIR_MCAST
 permit 224.1.1.1
 permit 224.1.2.1
!
router pim
 address-family ipv4
  no rp-address 10.0.1.1
  rp-address 10.0.1.1 BIDIR_MCAST bidir
commit
```

PE2 (Cisco IOS XE):

```
access-list 10 permit 224.1.1.1
access-list 10 permit 224.1.2.1
!
ip pim bidir-enable
ip pim rp-address 10.0.1.1 10 bidir
```

CE1 (Cisco IOS Software):

```
access-list 10 permit 224.1.1.1
access-list 10 permit 224.1.2.1
!
ip pim bidir-enable
ip pim rp-address 10.0.1.1 10 bidir
```

Step 2 Enable BIDIR-PIM on the P1 router:

```
router pim
 address-family ipv4
  no rp-address 10.0.1.1
  rp-address 10.0.1.1 bidir
commit
```

Step 4 Simulate multicast receivers for group 224.1.x.1 (or 224.1.y.0) on the Loopback0 interface on the CE and PE router:

PE1 (Cisco IOS XR):

```
router igmp
 interface Loopback0
  join-group 224.1.1.1
```

```
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 ip igmp join-group 224.1.2.1
```

CE1 (Cisco IOS Software):

```
interface Loopback0
 ip igmp join-group 224.1.1.1
```

Step 5 Examine the multicast routing table on the PE router. Answer the following questions:

Which significant (*,G) entries are present in the table?

(*,224.1.1.1), (*,224.1.2.1)

Which interfaces are in the OIL in the (*,G) entry for your group?

Loopback0, GigabitEthernet0/0/0/0, GigabitEthernet0/0/0/2

Step 6 Use Telnet to connect to the other pod CE router. Examine the multicast routing table.

Which significant (*,G) entries are present in the table?

(*,224.1.1.1), (*,224.1.2.1)

Step 8 Return to the Telnet session to the other pod CE router. Examine the multicast routing table on the CE router again. Answer the following questions:

Have there been any significant changes to the multicast routing table? Why or why not?

There have not been any significant changes. The router uses the previously created entries to send the multicast traffic to the RP.

Step 9 Remove the simulated multicast receivers for group 224.1.x.1 (or 224.1.y.0) from the Loopback0 interface on the CE and PE router:

PE1 (Cisco IOS XR):

```
router igmp
 interface Loopback0
  no join-group 224.1.1.1
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 no ip igmp join-group 224.1.2.1
```

CE1 (Cisco IOS Software):

```
interface Loopback0
 no ip igmp join-group 224.1.1.1
```

Step 10 Disable BIDIR-PIM on the P1, PE, and CE routers:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  no rp-address 10.0.1.1 BIDIR_MCAST bidir
  rp-address 10.0.1.1
commit
```

PE2 (Cisco IOS XE):


```
no ip pim bidir-enable
no ip pim rp-address 10.0.1.1 10 bidir
ip pim rp-address 10.0.1.1
```

CE1 (Cisco IOS Software):

```
no ip pim bidir-enable
no ip pim rp-address 10.0.1.1 10 bidir
ip pim rp-address 10.0.1.1
```

P1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  no rp-address 10.0.1.1 bidir
  rp-address 10.0.1.1
commit
```

Lab 5-3 Answer Key: Implement Rendezvous Point Distribution

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Enable Auto-RP

Step 1 Disable interfaces:

PE1 (Cisco IOS XR):

```
interface GigabitEthernet0/0/0/2
 shutdown
!
interface GigabitEthernet0/0/0/3
 shutdown
!
commit
```

PE2 (Cisco IOS XE):

```
interface GigabitEthernet0/0/2
 shutdown
!
interface GigabitEthernet0/0/3
 shutdown
```

Step 16 Verify IS-IS and disable the BGP neighbor:

CE1 (Cisco IOS Software):

```
interface Loopback0
 ip router isis
!
interface GigabitEthernet0/0
 ip router isis
!
router isis
 net 49.0001.0100.0101.0001.00
 is-type level-1
!
router bgp 64501
 neighbor 192.168.101.10 shutdown
```

PE1 (Cisco IOS XR):

```
router isis 1
 net 49.0001.0100.0100.1001.00
 interface Loopback0
  address-family ipv4 unicast
 interface GigabitEthernet0/0/0/0
  circuit-type level-1
  address-family ipv4 unicast
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
!
commit
```

CE2 (Cisco IOS Software):

```
interface Loopback0
 ip router isis
!
interface GigabitEthernet0/0
 ip router isis
!
router isis
 net 49.0002.0100.0201.0001.00
 is-type level-1
!
router bgp 64502
 neighbor 192.168.102.20 shutdown
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 ip router isis
!
interface GigabitEthernet0/0/0
 ip router isis
 isis circuit-type level-1
!
interface GigabitEthernet0/0/1
 ip router isis
 isis circuit-type level-2-only
!
router isis
 net 49.0002.0100.0200.1001.00
```

Step 17 Verify PIM-SM:

CE1 (Cisco IOS Software):

```
interface Loopback0
 ip pim sparse-mode
!
interface GigabitEthernet0/0
 ip pim sparse-mode
```

PE1 (Cisco IOS XR):

```
router pim
  interface Loopback0
```



```
   enable
  interface GigabitEthernet0/0/0/0
   enable
  interface GigabitEthernet0/0/0/1
   enable
!
commit
```

CE2 (Cisco IOS Software):

```
interface Loopback0
 ip pim sparse-mode
!
interface GigabitEthernet0/0
 ip pim sparse-mode
```

PE2 (Cisco IOS XE):

```
interface Loopback0
 ip pim sparse-mode
!
interface GigabitEthernet0/0/0
 ip pim sparse-mode
!
interface GigabitEthernet0/0/1
 ip pim sparse-mode
```

Step 18 Enable Auto-RP:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  auto-rp mapping-agent Loopback0 scope 16
  auto-rp candidate-rp Loopback0 scope 16
!
commit
```

PE2 (Cisco IOS XE):

```
ip pim send-rp-announce Loopback0 scope 16
ip pim send-rp-discovery Loopback0 scope 16
ip pim autorp listener
```

Task 2: Enable BSR

Step 1 Disable Auto-RP:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  no auto-rp mapping-agent Loopback0 scope 16
  no auto-rp candidate-rp Loopback0 scope 16
!
commit
```

PE2 (Cisco IOS XE):

```
no ip pim send-rp-announce Loopback0 scope 16
no ip pim send-rp-discovery Loopback0 scope 16
```

Step 2 Enable BSR:

PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  bsr candidate-bsr 10.1.1.1
  bsr candidate-rp 10.1.1.1
!
commit
```

PE2 (Cisco IOS XE):

```
ip pim bsr-candidate Loopback 0
ip pim rp-candidate Loopback 0
```

Task 3: Enable Anycast RP

Step 1 Disable BSR:

PE1 (Cisco IOS XR):

```
router pim
 no bsr candidate-bsr 10.1.1.1
 no bsr candidate-rp 10.1.1.1
!
commit
```

PE2 (Cisco IOS XE):

```
no ip pim bsr-candidate Loopback0
no ip pim rp-candidate Loopback0
```

Step 2 Enable Loopback1 and announce Loopback1 to the IS-IS:

PE1 (Cisco IOS XR):

```
interface Loopback1
 ipv4 address 1.1.1.1 255.255.255.255
!
multicast-routing
 address-family ipv4
  interface Looback1
   enable
!
router isis 1
 interface Loopback1
  address-family ipv4 unicast
!
router pim
 address-family ipv4
  interface Loopback1
   enable
!
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback1
 ip address 1.1.1.1 255.255.255.255
 ip router isis
 ip pim sparse-mode
```

Step 3 Configure static RP:

CE1 (Cisco IOS Software):

```
ip pim rp-address 1.1.1.1
```


PE1 (Cisco IOS XR):

```
router pim
 address-family ipv4
  rp-address 1.1.1.1
!
commit
```

CE2 (Cisco IOS Software):

```
ip pim rp-address 1.1.1.1
```

PE2 (Cisco IOS XE):

```
ip pim rp-address 1.1.1.1
```

Step 4 Configure MSDP session:

PE1 (Cisco IOS XR):

```
router msdp
 originator-id Loopback0
 peer 10.2.1.1
  connect-source Loopback0
!
commit
```

PE2 (Cisco IOS XE):

```
ip msdp peer 10.1.1.1 connect-source Loopback0
ip msdp originator-id Loopback0
```

Lab 6-1 Answer Key: Implement a DHCPv6 Server with Prefix Delegation

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Configure a Prefix Delegation DHCPv6 Server and Client

Step 2 Configure the PE router as a prefix delegation DHCPv6 server:

PE1 (Cisco IOS XR):

```
dhcp ipv6
 pool POOL
  dns-server 2001:db8:100::1
 !
 interface GigabitEthernet0/0/0/0 server
  pd 2001:db8:100:1::/64
  pool POOL
commit
```

PE2 (Cisco IOS XE):

```
ipv6 local pool PREFIX 2001:DB8:100:2::/64 64
!
ipv6 dhcp pool Customers
 prefix-delegation pool PREFIX
 dns-server 2001:DB8:100::2
```

Step 3 Enable a DHCPv6 prefix delegation client on the interface on the CE router:

```
interface GigabitEthernet0/0
```

```
 ipv6 dhcp client pd SP_ASSIGNED_PREFIX
!
interface GigabitEthernet0/1
 ipv6 address SP_ASSIGNED_PREFIX ::1/64
```

Task 2: Configure a DHCPv6 Lite Server

Step 1 Configure a DHCPv6 pool on the CE router. The DNS server should be imported as received from the PE router:

```
ipv6 dhcp pool POOL
 import dns-server
```

Step 2 Enable the DHCPv6 server on the CE router on the GigabitEthernet0/1 interface. Configure the router to instruct DHCP clients not to use DHCP for address assignments. However, clients should obtain other parameters, such as DNS server, using DHCP:

```
interface GigabitEthernet0/1
 ipv6 nd other-config-flag
 ipv6 dhcp server POOL
```

Step 5 Optionally, remove the previously configured IP address from the the other pod CE router GigabitEthernet0/1 interface. Configure the interface to obtain an IP address via stateless autoconfiguration:

```
interface GigabitEthernet0/1
 no ipv6 address SP_ASSIGNED_PREFIX ::1/64
 ipv6 address autoconfig
```

Lab 6-2 Answer Key: Implement IPv6 Multicasting

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Create New Loopback Interface and Verify Connectivity

Step 1 On the PE router, create a new loopback interface. Use x0 (or y0) as the interface identifier:

PE1 (Cisco IOS XR):

```
interface Loopback10
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback20
```

Step 2 Assign the following IP address to the loopback interface. Advertise the loopback interface using the IS-IS routing protocol:

PE1 (Cisco IOS XR):

```
interface Loopback10
 ipv6 address 2001:db8:1:1::1/128
!
router isis 1
 interface Loopback50
  address-family ipv6 unicast
commit
```

PE2 (Cisco IOS XE):

```
interface Loopback20
```



```
 ipv6 address 2001:DB8:2:2::1/128
 ipv6 router isis
```

Task 2: Implement IPv6 Multicast Using Embedded RPs

Step 1 On the CE router, enable IPv6 multicast routing. Configure the router to always stay on the shared tree:

```
ipv6 multicast-routing
ipv6 pim spt-threshold infinity
```

Step 2 On the PE router, enable IPv6 multicast routing:

PE1 (Cisco IOS XR):

```
multicast-routing
 address-family ipv6
  interface all enable
!
router pim
 address-family ipv6
  spt-threshold infinity
  interface Loopback50
   enable
  !
  interface GigabitEthernet0/0/0/0
   enable
  !
  interface GigabitEthernet0/0/0/1
  enable
commit
```

PE2 (Cisco IOS XE):

```
ipv6 multicast-routing
ipv6 pim spt-threshold infinity
```

Step 5 On the PE router, manually configure the RP address:

PE1 (Cisco IOS XR):

```
ipv6 access-list MCAST_POD1
 permit ipv6 any ff75:140:2001:db8:1:1::/96
!
router pim
 address-family ipv6
  embedded-rp 2001:db8:1:1::1 MCAST_POD1
commit
```

PE2 (Cisco IOS XE):

```
ipv6 pim rp-address 2001:DB8:2:2::1
```

Step 6 Calculate the multicast group addresses that can be served by the RP that is assigned to your pod. Use site-local multicast group addresses. Complete the following table:

Router	RP Address	Group Addresses
PEx	2001.db8:x:x::1/128	FF75:0140:2001:db8:y:y::/96
PEy	2001.db8:y:y::1/128	FF75:0140:2001:db8:y:y::1/96

Step 7 Answer the following questions:

How many multicast groups can an embedded RP serve?

$2^{32} = 4294967296$

Step 8 Enable the Loopback0 interface on the CE router as a multicast receiver for one of the multicast groups served by your RP (for example, FF75:0140:2001:db8:x:x::1):

```
interface Loopback0
 ipv6 mld join-group FF75:140:2001:DB8:1:1:0:1
```

Lab 6-3 Answer Key: Implement Tunnels for IPv6

When you complete this lab activity, the device configuration and device outputs will be similar to the results shown here, with differences that are specific to your pod.

Task 1: Configure Static IPv6-in-IPv4 Tunnel

Step 3 Create a tunnel interface on the CE router. Enable IPv6 on the interface. Use link-local IPv6 addresses for tunnel interface addressing.

```
interface Tunnel0
 ipv6 enable
```

Step 4 Specify the Loopback0 interface as the tunnel source. Specify the IP address of the other pod CE router Loopback0 interface as the tunnel destination.

```
interface Tunnel0
 tunnel source Loopback0
 tunnel destination 10.2.10.1
```

Step 5 Set the tunnel mode to IPv6-in-IPv4.

```
interface Tunnel0
 tunnel mode ipv6ip
```

Step 6 Create a static IPv6 route for the other pod CE router Loopback0 interface that will point to the tunnel interface.

```
ipv6 route 2001:DB8:10:6:10::1/128 Tunnel0
```

Step 9 Remove the tunnel interface from the CE router. Remove the static route that was created in this task as well.

```
no interface Tunnel0
no ipv6 route 2001:DB8:10:6:10::1/128 Tunnel0
```

Task 2: Configure Dynamic 6RD Tunnels

Step 1 On the PEy router, advertise the network configured on the Loopback0 interface through BGP:

```
router bgp 64500
 network 10.2.1.1 mask 255.255.255.255
```

Step 3 Answer the following questions:

How long is a prefix that is common to the CEx, CEy, and PEy loopback interfaces?

The prefix is 8 bits long.

How many bits from the IPv4 address will be used to construct 6RD networks?

24 bits of the IPv4 address will be used to construct 6RD networks.

Step 4 Construct 6RD networks for IPv6 sites behind the CE routers:

CE Router	IPv4 Tunnel Endpoint	6RD Prefix	6RD Network
CEx	10.x.10.1	2001:db8:aa00::/40	2001:DB8:AA0x:A01::/64
CEy	10.y.10.1	2001:db8:aa00::/40	2001:DB8:AA0y:A01::/64

Step 5 Construct a 6RD network for the PEy router as well:

PE Router	IPv4 Tunnel Endpoint	6RD Prefix	6RD Network
PEy	10.y.1.1	2001:db8:aa00::/40	2001:DB8:AA0y:101::/64

Step 6 Configure the tunnel interface on the CE router:

```
interface Tunnel0
 ipv6 enable
 tunnel source Loopback0
 tunnel mode ipv6ip 6rd
 tunnel 6rd ipv4 prefix-len 8
 tunnel 6rd prefix 2001:DB8:AA00::/40
 tunnel 6rd br 10.2.1.1
```

Step 8 Assign the first IPv6 address from the 6RD network to a new loopback interface. Use 10 as interface identifier:

```
interface Loopback10
 ipv6 address 2001:DB8:AA01:A01::1/64
 ipv6 enable
```

Step 9 Create a static route for the 6RD prefix that will use the tunnel interface as the outgoing interface:

```
ipv6 route 2001:DB8:AA00::/40 Tunnel0
```

Step 10 Configure the tunnel interface on the PEy router:

```
interface Tunnel0
 ipv6 enable
 tunnel source Loopback0
 tunnel mode ipv6ip 6rd
 tunnel 6rd ipv4 prefix-len 8
 tunnel 6rd prefix 2001:DB8:AA00::/40
```

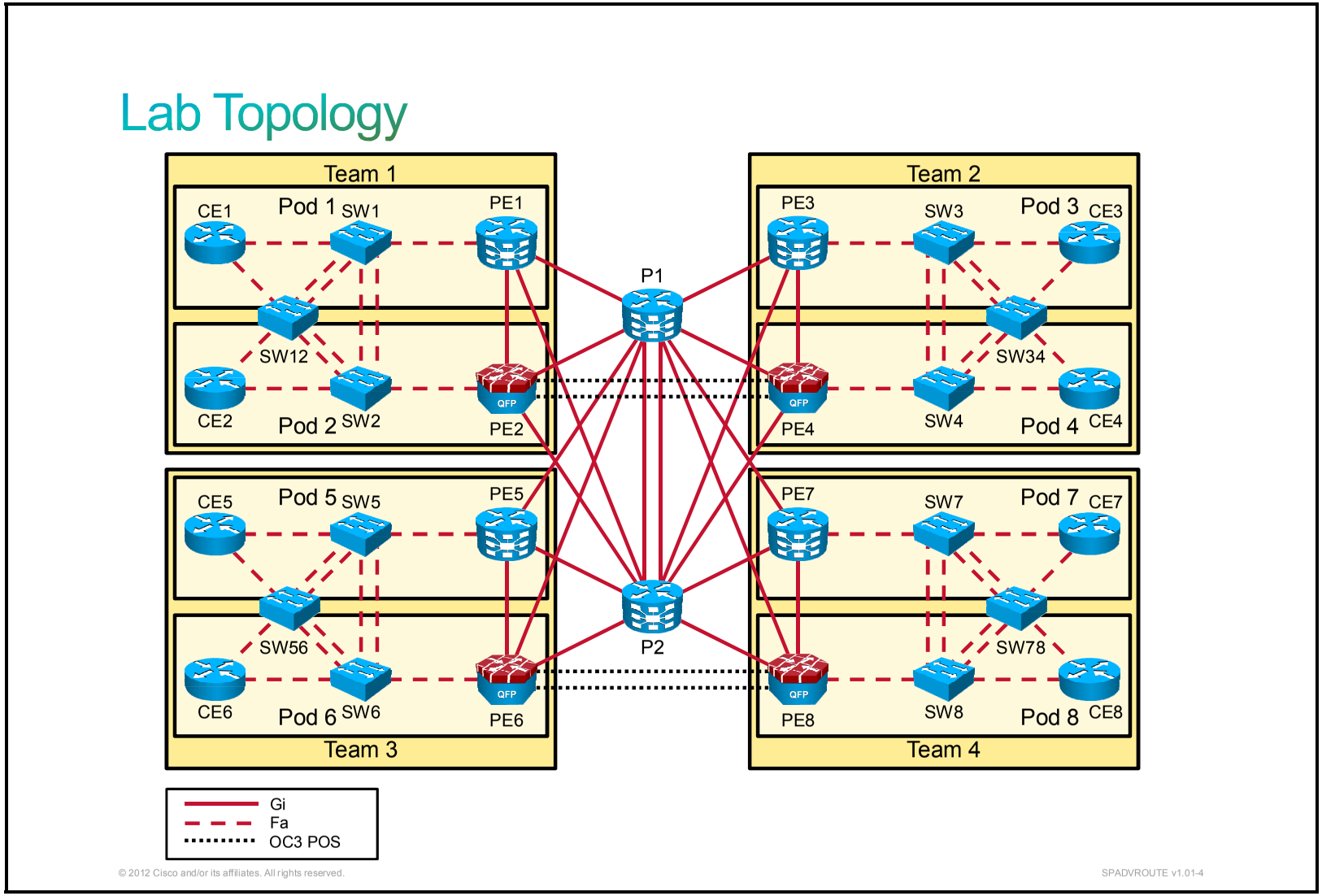
Step 12 Create another loopback interface on the PEy router. Use 10 as interface identifier and 2001:db8:100:y::1 as IP address on the interface:

```
interface Loopback10
 ipv6 address 2001:DB8:100:2::1/64
 ipv6 enable
```

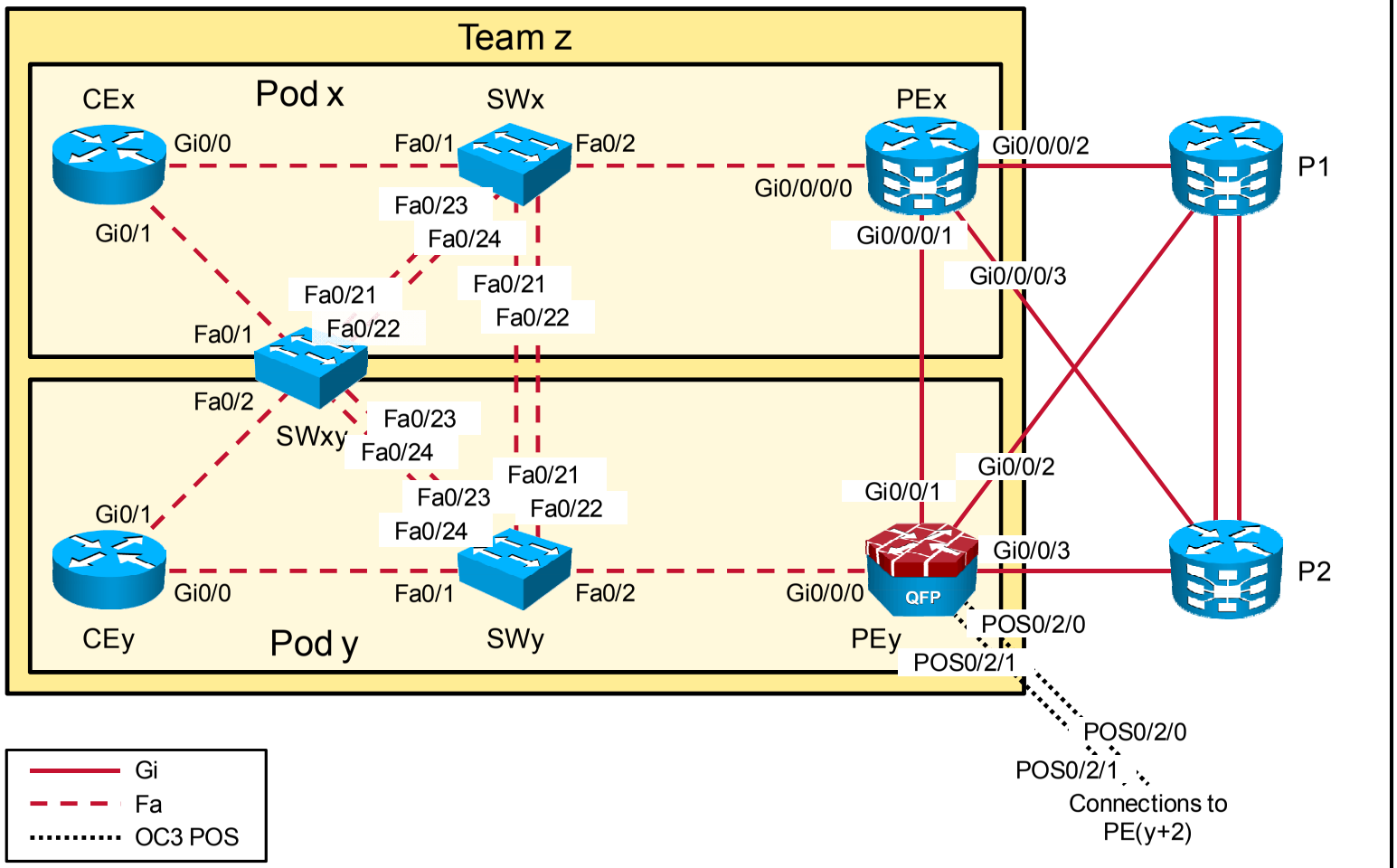
Step 13 On the CE router, create a default route that will use the tunnel interface as the outgoing interface and will point to the 6RD network of the PEy router:

```
ipv6 route ::/0 Tunnel0 2001:DB8:AA02:101::
```

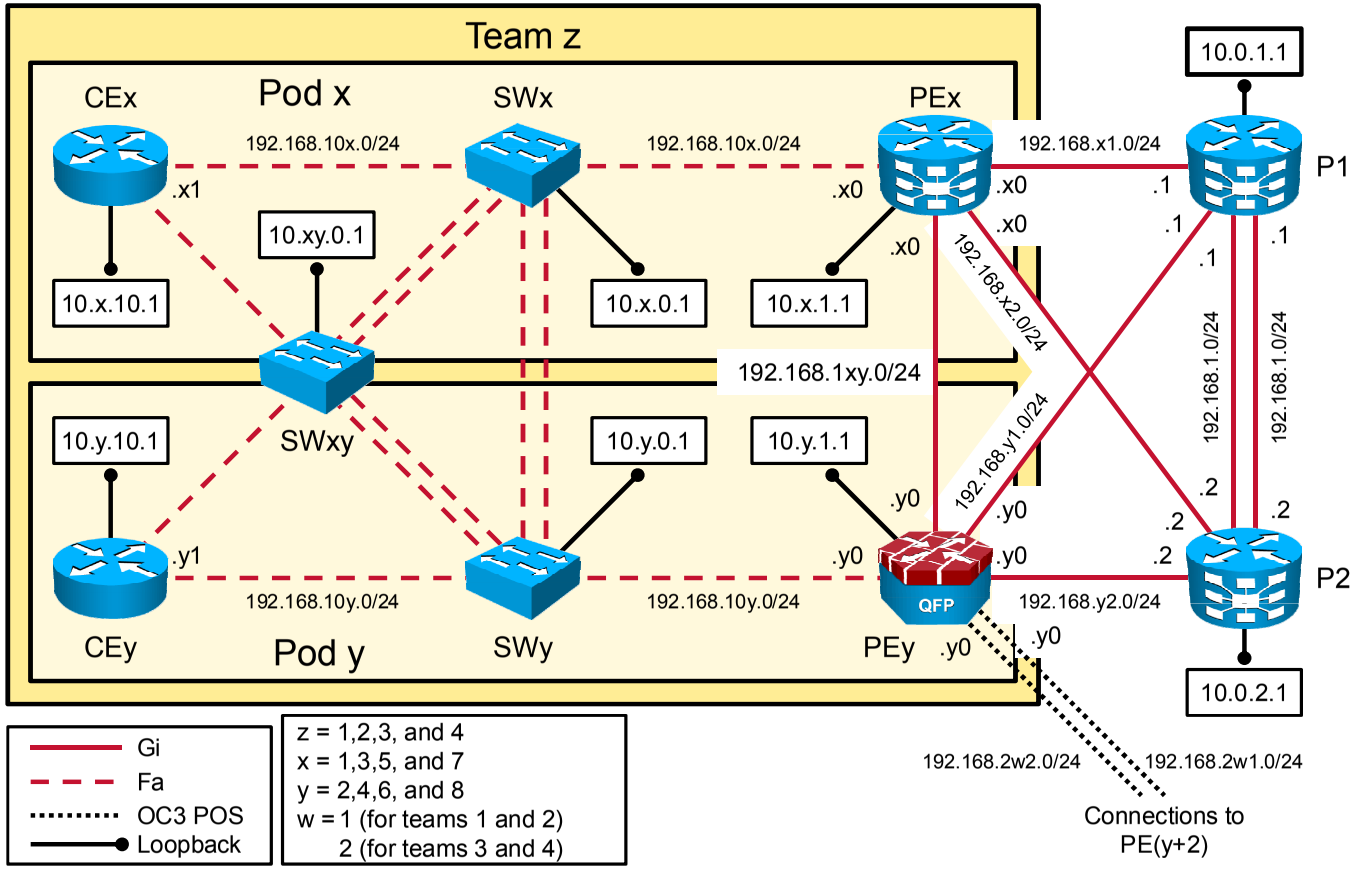
Appendix A: Lab Topology



Lab Interface Identification: Team View

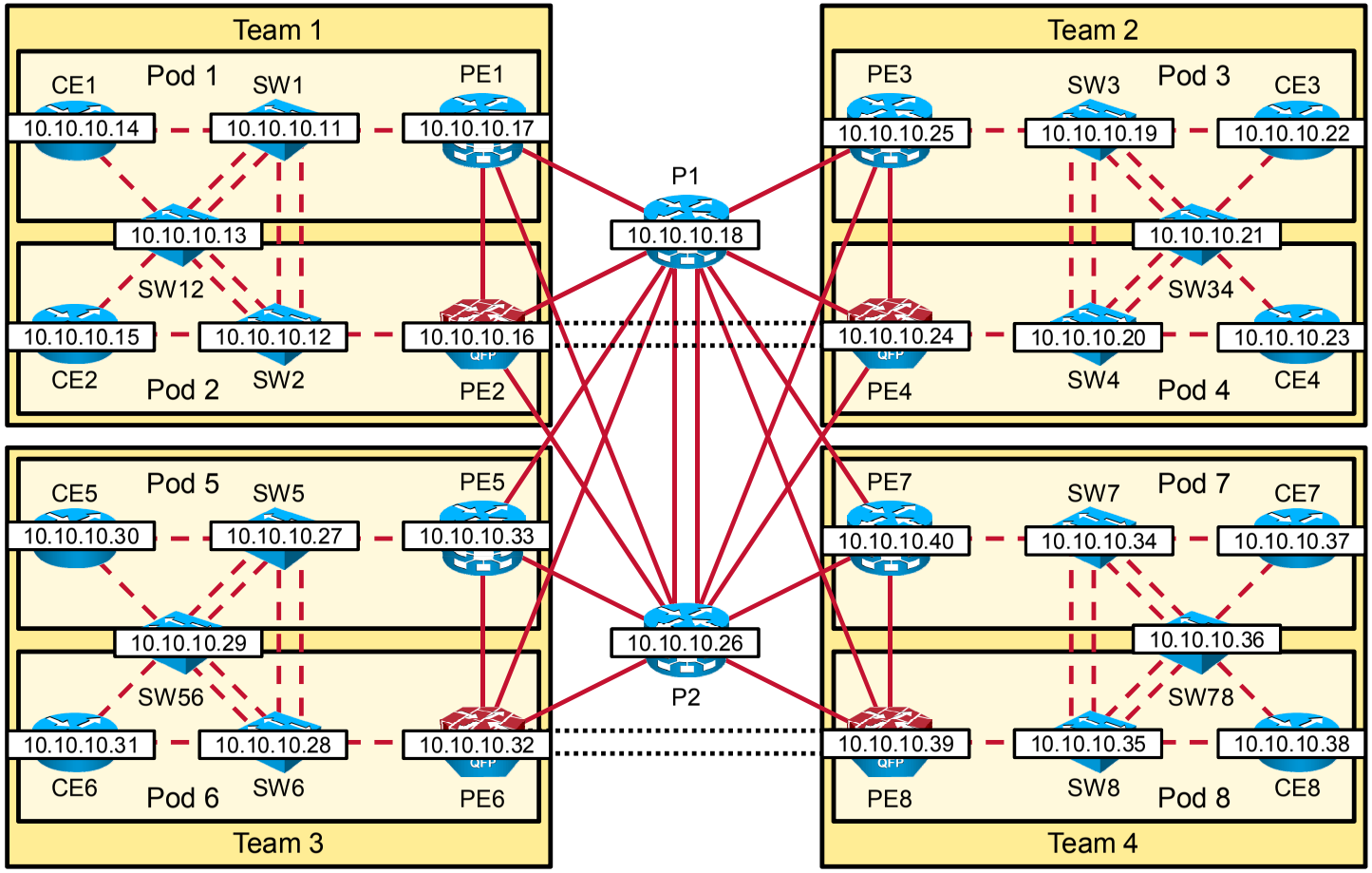


Lab IP Addressing: Team View

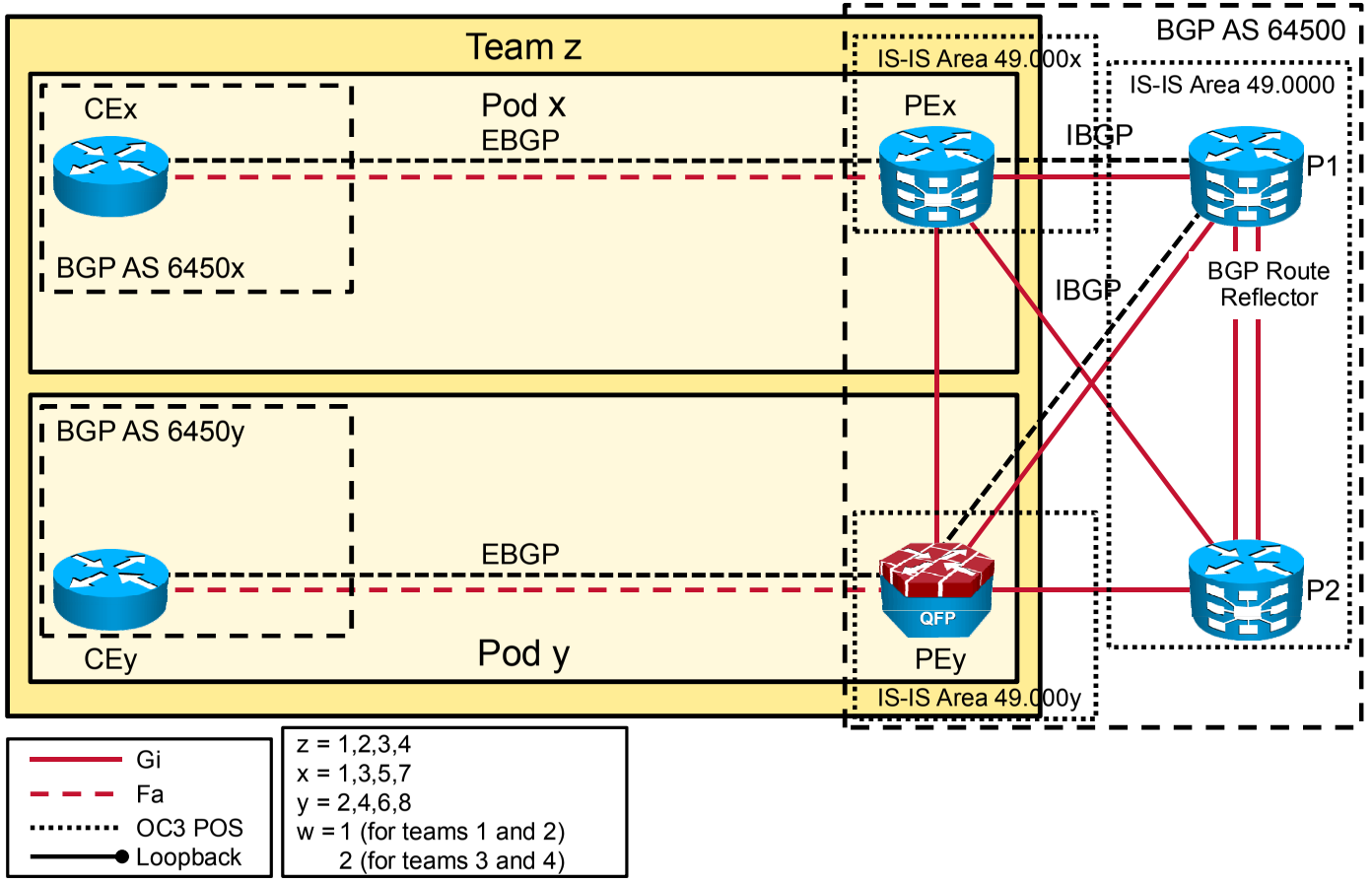


SPADVROUTE v1.01-6

Management IP Addresses



Existing Routing: Team View



SPADVROUTE v1.01-8