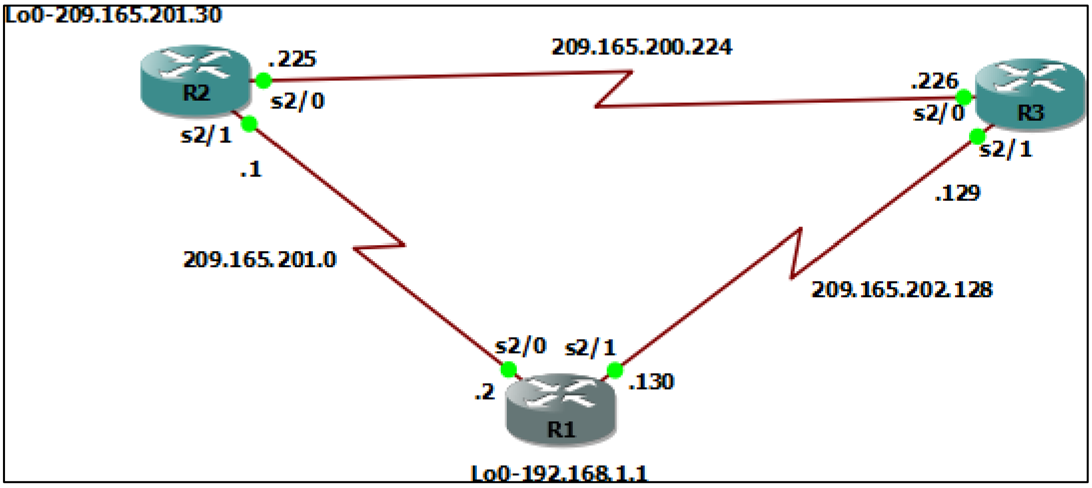
Practical -1

Title: - Configure IP SLA Tracking and Path Control

Topology: -



Topology for SLA Tracking and Path Control

Step 1: Prepare the routers and configure the router hostname and interface addresses.

a. Cable the network as shown in the topology diagram. Erase the startup configuration and reload each router to clear the previous configurations. Using the addressing scheme in the diagram, create the loopback interfaces and apply IP addresses to them as well as the serial interfaces on R1, ISP1, and ISP2.

Router R1

```
conf t
hostname R1
interface Loopback 0
description R1 LAN
ip address 192.168.1.1 255.255.255.0
interface Serial3/0
description R1 --> ISP1
ip address 209.165.201.2 255.255.255.252
clock rate 128000
```

```
bandwidth 128
no shutdown
interface Serial3/1
description R1 --> ISP2
ip address 209.165.202.130 255.255.255.252
bandwidth 128
no shutdown
```

Router ISP1 (R2)

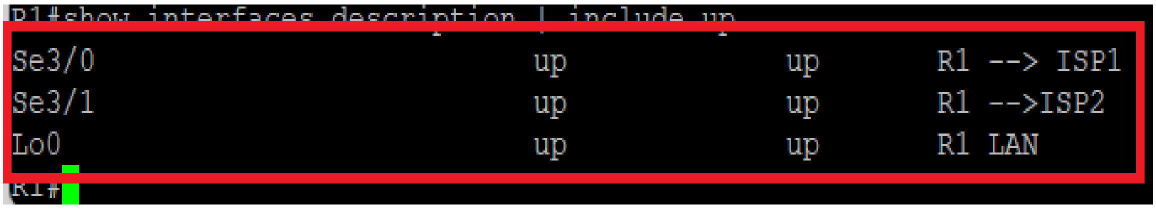
```
conf t
hostname ISP1
interface Loopback0
description Simulated Internet Web Server
ip address 209.165.200.254 255.255.255.255
interface Loopback1
description ISP1 DNS Server
ip address 209.165.201.30 255.255.255.255
interface Serial3/1
description ISP1 --> R1
ip address 209.165.201.1 255.255.255.252
bandwidth 128
no shutdown
interface Serial3/0
description ISP1 --> ISP2
ip address 209.165.200.225 255.255.255.252
clock rate 128000
bandwidth 128
no shutdown
```

Router ISP2 (R3)

```
conf t
hostname ISP2
interface Loopback0
description Simulated Internet Web Server
ip address 209.165.200.254 255.255.255.255
interface Loopback1
description ISP2 DNS Server
ip address 209.165.202.158 255.255.255.255
interface Serial3/1
description ISP2 --> R1
ip address 209.165.202.129 255.255.255.252
clock rate 128000
bandwidth 128
no shutdown
interface Serial3/0
description ISP2 --> ISP1
ip address 209.165.200.226 255.255.255.252
bandwidth 128
no shutdown
```

b. Verify the configuration by using the show interfaces description command. The output from router R1 is shown here as an example.

```
R1# show interfaces description | include up
```



```
R1#show interfaces description | include up
Se3/0                          up             up        R1 --> ISP1
Se3/1                          up             up        R1 -->ISP2
Lo0                            up             up        R1 LAN
R1#
```

Step 2: Configure static routing.

a. Implement the routing policies on the respective routers. You can copy and paste the following configurations.

Router R1

```
conf t
R1(config)# ip route 0.0.0.0 0.0.0.0 209.165.201.1
```

Router ISP1 (R2)

```
ISP1(config)# router eigrp 1
ISP1(config-router)# network 209.165.200.224 0.0.0.3
ISP1(config-router)# network 209.165.201.0 0.0.0.31
ISP1(config-router)# no auto-summary
ISP1(config-router)# exit
ISP1(config-router)# ip route 192.168.1.0 255.255.255.0
                      209.165.201.2
```

Router ISP2 (R3)

```
ISP2(config)# router eigrp 1
ISP2(config-router)# network 209.165.200.224 0.0.0.3
ISP2(config-router)# network 209.165.202.128 0.0.0.31
ISP2(config-router)# no auto-summary
ISP2(config-router)# exit
ISP2(config)# ip route 192.168.1.0 255.255.255.0
209.165.202.130
```

b. The Cisco IOS IP SLA feature enables an administrator to monitor network performance between Cisco devices (switches or routers) or from a Cisco device to a remote IP device.
Router 1

```
R1# tclsh
     foreach address {
     209.165.200.254
     209.165.201.30
     209.165.202.158
     } {
     ping $address source 192.168.1.1
     }
```

All pings should be successful. Troubleshoot if necessary.

c. Trace the path taken to the web server, ISP1 DNS server, and ISP2 DNS server. You can copy the following Tcl script and paste it into R1.

```
R1# tclsh

foreach address {
209.165.200.254
209.165.201.30
209.165.202.158
} {
trace $address source 192.168.1.1
}
```

Step 3: Configure IP SLA probes.

a. Create an ICMP echo probe on R1 to the primary DNS server on ISP1 using the ip sla command.

Router 1

```
R1(config)# ip sla 11
R1(config-ip-sla)# icmp-echo 209.165.201.30
R1(config-ip-sla-echo)# frequency 10
R1(config-ip-sla-echo)# exit
R1(config)# ip sla schedule 11 life forever start-time now
```

b. Verify the IP SLAs configuration of operation 11 using the show ip sla configuration 11 command.

Router 1

```
R1#show ip sla configuration 11
```

```
IP SLAs Infrastructure Engine-III
Entry number: 11
Owner:
Tag:
Operation timeout (milliseconds): 5000
Type of operation to perform: icmp-echo
Target address/Source address: 209.165.201.30/0.0.0.0
Type Of Service parameter: 0x0
Request size (ARR data portion): 28
Verify data: No
Vrf Name:
Schedule:
   Operation frequency (seconds): 10  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
   Number of statistic hours kept: 2
 --More--
```

c. Issue the show ip sla statistics command to display the number of successes, failures, and results of the latest operations.

Router 1

```
R1#show ip sla statistics
```

```
R1# show ip sla statistics
IPSLAs Latest Operation Statistics

IPSLA operation id: 11
        Latest RTT: 20 milliseconds
Latest operation start time: 19:02:10 UTC Wed Mar 8 2023
Latest operation return code: OK
Number of successes: 66
Number of failures: 0
Operation time to live: Forever
```

d. Although not actually required because IP SLA session 11 alone could provide the desired fault tolerance, create a second probe, 22, to test connectivity to the second DNS server located on router ISP2.

Router 1

```
R1(config)# ip sla 22
R1(config-ip-sla)# icmp-echo 209.165.202.158
R1(config-ip-sla-echo)# frequency 10
R1(config-ip-sla-echo)# exit
R1(config)# ip sla schedule 22 life forever start-time now
R1(config)# end
```

e. Verify the new probe using the show ip sla configuration and show ip sla statistics commands.

Router 1

R1#show ip sla configuration 22

```
IP SLAs Infrastructure Engine-III
Entry number: 22
Owner:
Tag:
Operation timeout (milliseconds): 5000
Type of operation to perform: icmp-echo
Target address/Source address: 209.165.202.158/0.0.0.0
Type Of Service parameter: 0x0
Request size (ARR data portion): 28
Verify data: No
Vrf Name:
Schedule:
   Operation frequency (seconds): 10  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
   Number of statistic hours kept: 2
```

R1#show ip sla statistics 22

```
R1# show ip sla statistics 22
IPSLAs Latest Operation Statistics

IPSLA operation id: 22
        Latest RTT: 60 milliseconds
Latest operation start time: 19:14:57 UTC Wed Mar 8 2023
Latest operation return code: OK
Number of successes: 56
Number of failures: 0
Operation time to live: Forever
```

Step 4: Configure tracking options.

a. On R1, remove the current default route and replace it with a floating static route having an administrative distance of 5.

Router 1

```
R1(config)# no ip route 0.0.0.0 0.0.0.0 209.165.201.1
R1(config)# ip route 0.0.0.0 0.0.0.0 209.165.201.1 5
R1(config)# exit
```

b. Verify the routing table

Router 1

R1#show ip route | begin Gateway

```
R1#show ip route | begin Gateway
Gateway of last resort is 209.165.201.0 to network 0.0.0.0

S*    0.0.0.0/0 [5/0] via 209.165.201.0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Loopback0
L        192.168.1.1/32 is directly connected, Loopback0
      209.165.201.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.201.0/30 is directly connected, Serial3/0
L        209.165.201.2/32 is directly connected, Serial3/0
      209.165.202.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.202.128/30 is directly connected, Serial3/1
L        209.165.202.130/32 is directly connected, Serial3/1
```

c. From global configuration mode on R1, use the track 1 ip sla 11 reachability command to enter the config-track sub configuration mode.

Router 1

```
conf t
R1(config)# track 1 ip sla 11 reachability
```

d. Specify the level of sensitivity to changes of tracked objects to 10 seconds of down delay and 1 second of up delay using the delay down 10 up 1 command. The delay helps to alleviate the effect of flapping objects—objects that are going down and up rapidly. In this situation, if the DNS server fails momentarily and comes back up within 10 seconds, there is no impact.

Router 1

```
R1(config-track)# delay down 10 up 1
R1(config-track)# exit
```

e. To view routing table changes as they happen, first enable the debug ip routing command.

Router 1

```
R1#debug ip routing
```

f. Configure the floating static route that will be implemented when tracking object 1 is active. Use the ip route 0.0.0.0 0.0.0.0 209.165.201.1 2 track 1 command to create a floating static default route via 209.165.201.1 (ISP1). Notice that this command references the tracking object number 1, which in turn references IP SLA operation number 11.

Router 1

```
conf t
R1(config)# ip route 0.0.0.0 0.0.0.0 209.165.201.1 2 track 1
```

```
R1(config)#ip route 0.0.0.0 0.0.0.0 209.165.201.1 2 track 1
R1(config)#
*Mar  8 19:27:32.111: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.201.0   1048578

*Mar  8 19:27:32.119: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.201.1   1048578

*Mar  8 19:27:32.123: RT: closer admin distance for 0.0.0.0, flushing 1 routes
*Mar  8 19:27:32.123: RT: add 0.0.0.0/0 via 209.165.201.1, static metric [2/0]
*Mar  8 19:27:32.127: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.201.0   1048578

*Mar  8 19:27:32.131: RT: rib update return code: 17
```

g. Repeat the steps for operation 22, track number 2, and assign the static route an admin distance higher than track 1 and lower than 5. On R1, copy the following configuration, which sets an admin distance of 3.

Router 1

```
conf t
R1(config)# track 2 ip sla 22 reachability
R1(config-track)# delay down 10 up 1
R1(config-track)# exit
R1(config)# ip route 0.0.0.0 0.0.0.0 209.165.202.129 3 track 2
```

h. Verify the routing table again.

Router 1

```
R1#show ip route | begin Gateway
```

```
R1#show ip route | begin Gateway
Gateway of last resort is 209.165.201.1 to network 0.0.0.0

S*    0.0.0.0/0 [2/0] via 209.165.201.1
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Loopback0
L        192.168.1.1/32 is directly connected, Loopback0
      209.165.201.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.201.0/30 is directly connected, Serial3/0
L        209.165.201.2/32 is directly connected, Serial3/0
      209.165.202.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.202.128/30 is directly connected, Serial3/1
L        209.165.202.130/32 is directly connected, Serial3/1
```

Step 5: Verify IP SLA operation.

a. On ISP1 disable the loopback interface 1.

ISP1 (Router 2)

```
conf t
ISP1(config-if)# int lo1
ISP1(config-if)# shutdown
```

```
ISP1(config)#int lo1
ISP1(config-if)#int lo1
ISP1(config-if)#shutdown
ISP1(config-if)#
*Mar  8 19:36:50.059: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1,
 changed state to down
*Mar  8 19:36:50.059: %LINK-5-CHANGED: Interface Loopback1, changed state to adm
inistratively down
```

b. On R1, observe the debug output being generated. Recall that R1 will wait up to 10 seconds before initiating action therefore several seconds will elapse before the output is generated.

Router 1

`R1#     (observe the output)`

```
R1#
*Mar  8 19:37:13.583: %TRACKING-5-STATE: 1 ip sla 11 reachability Up->Down
*Mar  8 19:37:13.583: RT: del 0.0.0.0 via 209.165.201.1, static metric [2/0]
*Mar  8 19:37:13.583: RT: delete network route to 0.0.0.0/0
*Mar  8 19:37:13.583: RT: default path has been cleared
*Mar  8 19:37:13.583: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.202.129   1048578

*Mar  8 19:37:13.583: RT: add 0.0.0.0/0 via 209.165.202.129, static metric [3/0]
*Mar  8 19:37:13.583: RT: default path is now 0.0.0.0 via 209.165.202.129
*Mar  8 19:37:13.583: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.201.0   1048578

*Mar  8 19:37:13.583: RT: rib update return code: 17
*Mar  8 19:37:13.591: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.202.129   1048578

R1#
*Mar  8 19:37:13.595: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.201.0   1048578

*Mar  8 19:37:13.599: RT: rib update return code: 17
```

c. On R1, verify the routing table.

Router 1

`R1#show ip route | begin Gateway`

```
R1#show ip route | begin Gateway
Gateway of last resort is 209.165.202.129 to network 0.0.0.0

S*    0.0.0.0/0 [3/0] via 209.165.202.129
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Loopback0
L        192.168.1.1/32 is directly connected, Loopback0
      209.165.201.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.201.0/30 is directly connected, Serial3/0
L        209.165.201.2/32 is directly connected, Serial3/0
      209.165.202.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.202.128/30 is directly connected, Serial3/1
L        209.165.202.130/32 is directly connected, Serial3/1
```

d. Verify the IP SLA statistics.

Router 1

`R1#show ip sla statistics`

```
R1#show ip sla statistics
IPSLAs Latest Operation Statistics

IPSLA operation id: 11
        Latest RTT: NoConnection/Busy/Timeout
Latest operation start time: 19:45:30 UTC Wed Mar 8 2023
Latest operation return code: Timeout
Number of successes: 271
Number of failures: 55
Operation time to live: Forever


IPSLA operation id: 22
        Latest RTT: 24 milliseconds
Latest operation start time: 19:45:37 UTC Wed Mar 8 2023
Latest operation return code: OK
Number of successes: 238
Number of failures: 2
Operation time to live: Forever
```

e. On R1, initiate a trace to the web server from the internal LAN IP address.

Router 1

`R1#Trace 209.165.200.254 source 192.168.1.1`

```
R1#trace 209.165.200.254 source 192.168.1.1
Type escape sequence to abort.
Tracing the route to 209.165.200.254
VRF info: (vrf in name/id, vrf out name/id)
  1 209.165.202.129 20 msec 28 msec 28 msec
```

f. On ISP1, re-enable the DNS address by issuing the no shutdown command on the loopback 1 interface to examine the routing behaviour when connectivity to the ISP1 DNS is restored.

ISP1 (Router 2)

`ISP1#no shutdown`

```
ISP1(config-if)#no shut
ISP1(config-if)#
*Mar  8 19:49:31.571: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1,
changed state to up
ISP1(config-if)#
*Mar  8 19:49:31.575: %LINK-3-UPDOWN: Interface Loopback1, changed state to up
ISP1(config-if)#
```

Notice the output of the debug ip routing command on R1.

Router 1

`R1#    (observe the output)`

```
R1#
*Mar  8 19:49:54.663: %TRACKING-5-STATE: 1 ip sla 11 reachability Down->Up
R1#
*Mar  8 19:49:54.667: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.202.129   1048578

*Mar  8 19:49:54.667: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.201.0   1048578

*Mar  8 19:49:54.667: RT: rib update return code: 17
*Mar  8 19:49:54.667: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.201.1   1048578

*Mar  8 19:49:54.667: RT: closer admin distance for 0.0.0.0, flushing 1 routes
*Mar  8 19:49:54.667: RT: add 0.0.0.0/0 via 209.165.201.1, static metric [2/0]
*Mar  8 19:49:54.667: RT: updating static 0.0.0.0/0 (0x0):
    via 209.165.202.129   1048578

*Mar  8 19:49:54.667: RT: rib update return code: 17
R1#
```

g. Again, examine the IP SLA statistics.

Router 1

`R1#Show ip sla statistics`

```
R1#show ip sla statistics
IPSLAs Latest Operation Statistics

IPSLA operation id: 11
        Latest RTT: 20 milliseconds
Latest operation start time: 19:55:40 UTC Wed Mar 8 2023
Latest operation return code: OK
Number of successes: 27
Number of failures: 0
Operation time to live: Forever



IPSLA operation id: 22
        Latest RTT: 64 milliseconds
Latest operation start time: 19:55:37 UTC Wed Mar 8 2023
Latest operation return code: OK
Number of successes: 298
Number of failures: 2
Operation time to live: Forever
```

h. Verify the routing table.

Router 1

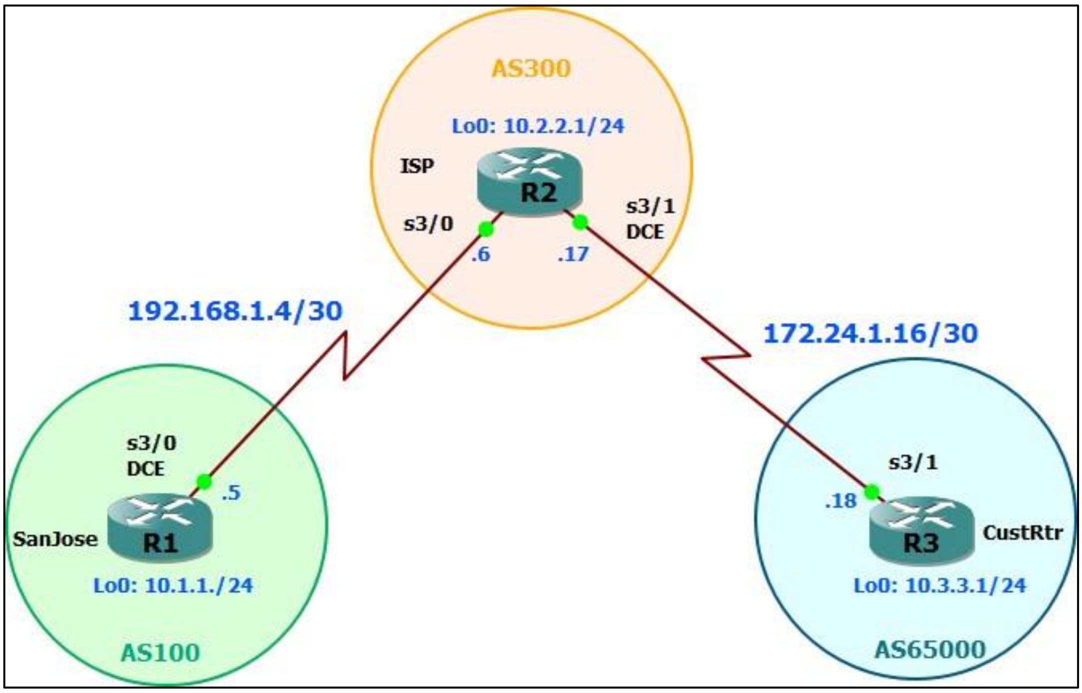
`R1#Show ip route | begin Gateway`

```
R1# show ip route | begin Gateway
Gateway of last resort is 209.165.201.1 to network 0.0.0.0

S*    0.0.0.0/0 [2/0] via 209.165.201.1
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Loopback0
L        192.168.1.1/32 is directly connected, Loopback0
      209.165.201.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.201.0/30 is directly connected, Serial3/0
L        209.165.201.2/32 is directly connected, Serial3/0
      209.165.202.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.202.128/30 is directly connected, Serial3/1
L        209.165.202.130/32 is directly connected, Serial3/1
```

Practical -2

Title: - Using the AS_PATH Attribute

Topology: -



Topology for AS_PATH Attribute

Step 1: Prepare the routers for the lab.

Cable the network as shown in the topology diagram. Erase the startup configuration and reload each router to clear previous configurations.

Step 2: Configure the hostname and interface addresses.

a. Configure the Routers

Router 1(Hostname- SanJose)

```
Conf t
hostname SanJose
interface Loopback0
ip address 10.1.1.1 255.255.255.0
interface s3/0
ip address 192.168.1.5 255.255.255.252
```

```
clock rate 128000
no shutdown
```

Router 2(Hostname- ISP)

```
Conf t
hostname ISP
interface Loopback0
ip address 10.2.2.1 255.255.255.0
interface s3/0
ip address 192.168.1.6 255.255.255.252
no shutdown
interface s3/1
ip address 172.24.1.17 255.255.255.252
clock rate 128000
no shutdown
```

Router 3(Hostname- CustRtr)

```
Conf t
hostname CustRtr
interface Loopback0
ip address 10.3.3.1 255.255.255.0
interface s3/0
ip address 172.24.1.18 255.255.255.252
no shutdown
```

b. Use **ping** to test the connectivity between the directly connected routers.

Note: SanJose will not be able to reach either ISP's loopback (10.2.2.1) or CustRtr's loopback (10.3.3.1), nor will it be able to reach either end of the link joining ISP to CustRtr (172.24.1.17 and 172.24.1.18).

Step 3: Configure BGP.

a. Configure BGP for normal operation. Enter the appropriate BGP commands on each router so that they identify their BGP neighbors and advertise their loopback networks.

Router 1

```
SanJose(config)#router bgp 100
SanJose(config-router)#neighbor 192.168.1.6 remote-as 300
network 10.1.1.0 mask 255.255.255.0
```

Router 2

```
ISP(config)#router bgp 300
ISP(config-router)#neighbor 192.168.1.5 remote-as 100
ISP(config-router)#neighbor 172.24.1.18 remote-as 65000
ISP(config-router)#network 10.2.2.0 mask 255.255.255.0
```

Router 3

```
CustRtr(config)#router bgp 65000
CustRtr(config-router)#neighbor 172.24.1.17 remote-as 300
CustRtr(config-router)#network 10.3.3.0 mask 255.255.255.0
```

b. Verify that these routers have established the appropriate neighbor relationships by issuing the **show ip bgp neighbors** command on each router.

Router 2

```
R1#show ip bgp neighbors
```

```
BGP neighbor is 172.24.1.18,  remote AS 65000, external link
  BGP version 4, remote router ID 10.3.3.1
  BGP state = Established, up for 00:05:51
  Last read 00:00:15, last write 00:00:34, hold time is 180, keepalive interv
l is 60 seconds
```

```
BGP neighbor is 192.168.1.5,  remote AS 100, external link
  BGP version 4, remote router ID 10.1.1.1
  BGP state = Established, up for 00:08:44
  Last read 00:00:21, last write 00:00:22, hold time is 180, keepalive interval is 60 seconds
```

Step 4: Remove the private AS.

a. Display the SanJose routing table using the **show ip route** command. SanJose should have a route to both 10.2.2.0 and 10.3.3.0. Troubleshoot if necessary.

Router 1

```
SanJose#Show ip route
```

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

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        10.1.1.0/24 is directly connected, Loopback0
L        10.1.1.1/32 is directly connected, Loopback0
B        10.2.2.0/24 [20/0] via 192.168.1.6, 00:14:11
B        10.3.3.0/24 [20/0] via 192.168.1.6, 00:14:11
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.4/30 is directly connected, Serial3/0
L        192.168.1.5/32 is directly connected, Serial3/0
```

b. Ping the 10.3.3.1 address from SanJose.

(It fails.)

```
SanJose#ping 10.3.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.3.3.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

c. Ping again, this time as an extended ping, sourcing from the Loopback0 interface address.

```
SanJose#ping
Protocol [ip]:
Target IP address: 10.3.3.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.1.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.3.3.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 52/61/68 ms
```

Note: You can bypass extended ping mode and specify a source address using one of these commands:

```
SanJose# ping 10.3.3.1 source 10.1.1.1
```

d. Check the BGP table from SanJose by using the **show ip bgp** command. Note the AS path for the 10.3.3.0 network. The AS 65000 should be listed in the path to 10.3.3.0.

Router 1

```
SanJose# show ip bgp
```

```
BGP table version is 4, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,

              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.1.1.0/24      0.0.0.0                  0         32768 i
 *>  10.2.2.0/24      192.168.1.6              0             0 300 i
 *>  10.3.3.0/24      192.168.1.6                            0 300 65000 i
```

e. Configure ISP to strip the private AS numbers from BGP routes exchanged with SanJose using the following commands.

Router 2

```
Conf t
ISP(config)#router bgp 300
ISP(config-router)#neighbor 192.168.1.5 remove-private-as
```

f. After issuing these commands, use the **clear ip bgp *** command on ISP to reestablish the BGP relationship between the three routers. Wait several seconds and then return to SanJose to check its routing table.
   Note: The clear ip bgp * soft command can also be used to force each router to resend its BGP table. Does SanJose still have a route to 10.3.3.0?

Router 2

```
ISP#clear ip bgp *
```

SanJose should be able to ping 10.3.3.1 using its loopback 0 interface as the source of the ping.

Router 1

```
SanJose# ping 10.3.3.1 source lo0
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.3.3.1, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/64/68 ms
```

g. Now check the BGP table on SanJose. The AS_ PATH to the 10.3.3.0 network should be AS 300. It no longer has the private AS in the path.

Router 1

```
SanJose# show ip bgp
```

```
BGP table version is 9, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,

              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.1.1.0/24      0.0.0.0                  0         32768 i
 *>  10.2.2.0/24      192.168.1.6              0             0 300 i
 *>  10.3.3.0/24      192.168.1.6                            0 300 i
```

Step 5: Use the AS_PATH attribute to filter routes.

a. Configure a special kind of access list to match BGP routes with an AS_PATH attribute that both begins and ends with the number 100. Enter the following commands on ISP.

Router 2

```
Conf t
ISP(config)#ip as-path access-list 1 deny ^100$
ISP(config)#ip as-path access-list 1 permit .*
```

b. Apply the configured access list using the neighbor command with the filter-list option.

```
ISP(config)#router bgp 300
ISP(config-router)#neighbor 172.24.1.18 filter-list 1 out
```

c. Use the **clear ip bgp *** command to reset the routing information. Wait several seconds and then check the routing table for ISP. The route to 10.1.1.0 should be in the routing table.
Note: To force the local router to resend its BGP table, a less disruptive option is to use the clear ip bgp * out or clear ip bgp * soft command (the second command performs both outgoing and incoming route resync).

```
ISP2# clear ip bgp *
ISP2# show ip route
```

```
      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
B        10.1.1.0/24 [20/0] via 192.168.1.5, 00:00:24
C        10.2.2.0/24 is directly connected, Loopback0
L        10.2.2.1/32 is directly connected, Loopback0
B        10.3.3.0/24 [20/0] via 172.24.1.18, 00:00:24
      172.24.0.0/16 is variably subnetted, 2 subnets, 2 masks
C        172.24.1.16/30 is directly connected, Serial3/1
L        172.24.1.17/32 is directly connected, Serial3/1
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.4/30 is directly connected, Serial3/0
L        192.168.1.6/32 is directly connected, Serial3/0
```

d. Check the routing table for CustRtr. It should not have a route to 10.1.1.0 in its routing table.

Router 3

```
CustRtr#show ip route
```

```
     10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
B       10.2.2.0/24 [20/0] via 172.24.1.17, 00:03:36
C       10.3.3.0/24 is directly connected, Loopback0
L       10.3.3.1/32 is directly connected, Loopback0
     172.24.0.0/16 is variably subnetted, 2 subnets, 2 masks
C       172.24.1.16/30 is directly connected, Serial3/1
L       172.24.1.18/32 is directly connected, Serial3/1
```

e. Return to ISP and verify that the filter is working as intended. Issue the **show ip bgp regexp ^100$** command.

Router 2

```
ISP# show ip bgp regexp ^100$
```

```
BGP table version is 4, local router ID is 10.2.2.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.1.1.0/24      192.168.1.5              0             0 100 i
ISP#
```

f. Run the following Tcl script on all routers to verify whether there is connectivity. All pings from ISP should be successful. SanJose should not be able to ping the CustRtr loopback 10.3.3.1 or the WAN link 172.24.1.16/30. CustRtr should not be able to ping the SanJose loopback 10.1.1.1 or the WAN link 192.168.1.4/30.

Router 2

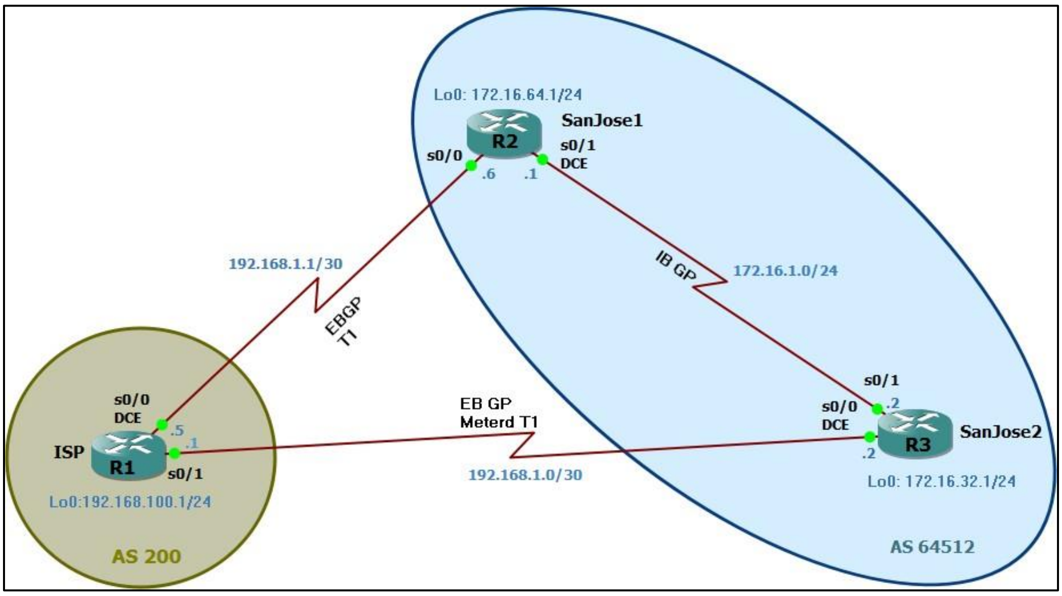
```
ISP# tclsh
foreach address {
10.1.1.1
10.2.2.1
10.3.3.1
192.168.1.5
192.168.1.6
172.24.1.17
172.24.1.18
} {
ping $address }
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/30/36 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.3.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/28/36 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/29/32 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/61/72 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.24.1.17, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/61/72 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.24.1.18, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/31/36 ms
```

Practical -3

Title: - Configuring IBGP and EBGP Sessions, Local Preference, and MED

Topology: -



Topology for IBGP&EBGP sessions, Local Preference &MED

Step 0: Suggested starting configurations.

a. Apply the following configuration to each router along with the appropriate **hostname**. The **exec-timeout 0 0** command should only be used in a lab environment.

```
Router# conf t
Router(config)#no ip domain-lookup
Router(config)#line con 0
Router(config-line)# logging synchronous
Router(config-line)# exec-timeout 0 0
```

Step 1: Configure interface addresses.

Using the addressing scheme in the diagram, create the loopback interfaces and apply IPv4 addresses to these and the serial interfaces on ISP (R1), SanJose1 (R2), and SanJose2 (R3).

Router R1 (hostname ISP)

```
ISP#conf t
ISP(config)# hostname ISP
```

```
ISP(config-if)#interface Loopback0
ISP(config-if)# ip address 192.168.100.1 255.255.255.0
ISP(config-if)# exit
ISP(config)#interface Serial0/0
ISP(config-if)# ip address 192.168.1.5 255.255.255.252
ISP(config-if)# clock rate 128000
ISP(config-if)# no shutdown
ISP(config-if)# exit
ISP(config)#interface Serial0/1
ISP(config-if)# ip address 192.168.1.1 255.255.255.252
ISP(config-if)# no shutdown
ISP(config-if)# end
```

Router R2 (hostname SanJose1)

```
SanJose1#conf t
SanJose1(config)#hostname SanJose1
SanJose1(config)#interface Loopback0
SanJose1(config-if)#ip address 172.16.64.1 255.255.255.0
SanJose1(config-if)#exit
SanJose1(config)#interface Serial0/0
SanJose1(config-if)#ip address 192.168.1.6 255.255.255.252
SanJose1(config-if)#no shutdown
SanJose1(config-if)#exit
SanJose1(config)#interface Serial0/1
SanJose1(config-if)#ip address 172.16.1.1 255.255.255.0
SanJose1(config-if)#clock rate 128000
SanJose1(config-if)#no shutdown
SanJose1(config-if)#end
```

Router R3 (hostname SanJose2)

```
SanJose2# conf t
SanJose2(config)#hostname SanJose2
SanJose2(config)#interface Loopback0
SanJose2(config-if)#ip address 172.16.32.1 255.255.255.0
SanJose2(config-if)#exit
SanJose2(config)#interface Serial0/0
SanJose2(config-if)#ip address 192.168.1.2 255.255.255.252
SanJose2(config-if)#clock rate 128000
SanJose2(config-if)#no shutdown
SanJose2(config-if)#exit
SanJose2(config)#interface Serial0/1
SanJose2(config-if)#ip address 172.16.1.2 255.255.255.0
SanJose2(config-if)#no shutdown
SanJose2(config-if)#end
```

b. Use **ping** to test the connectivity between the directly connected routers. Both SanJose routers should be able to ping each other and their local ISP serial link IP address. The ISP router cannot reach the segment between SanJose1 and SanJose2.

Step 2: Configure EIGRP.

Configure EIGRP between the SanJose1 and SanJose2 routers. (Note: If using an IOS prior to 15.0, use the no auto-summary router configuration command to disable automatic summarization. This command is the default beginning with IOS 15.)

SanJose 1

```
SanJose1#conf t
SanJose1(config)#router eigrp 1
SanJose1(config-router)#network 172.16.0.0
```

SanJose 2

```
SanJose2#conf t
SanJose2(config)#router eigrp 1
SanJose2(config-router)#network 172.16.0.0
```

Step 3: Configure IBGP and verify BGP neighbors.

a. Configure IBGP between the SanJose1 and SanJose2 routers. On the SanJose1 router, enter the following configuration.

SanJose1

```
SanJose1#conf t
SanJose1(config)#router bgp 64512
SanJose1(config-router)#neighbor 172.16.32.1 remote-as 64512
SanJose1(config-router)#neighbor 172.16.32.1 update-source lo0
```

b. Complete the IBGP configuration on SanJose2 using the following commands.

SanJose2

```
SanJose2#conf t
SanJose2(config)#router bgp 64512
SanJose2(config-router)#neighbor 172.16.64.1 remote-as 64512
SanJose2(config-router)#neighbor 172.16.64.1 update-source lo0
```

c. Verify that SanJose1 and SanJose2 become BGP neighbors by issuing the **show ip bgp neighbors** command on SanJose1. View the following partial output. If the BGP state is not established, troubleshoot the

SanJose2

```
SanJose2#show ip bgp neighbors
```

```
BGP neighbor is 172.16.64.1,  remote AS 64512, internal l
ink
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Idle
  Neighbor sessions:
    0 active, is not multisession capable (disabled)
    Stateful switchover support enabled: NO
  Default minimum time between advertisement runs is 0 se
conds
```

Step 4: Configure EBGP and verify BGP neighbors.

a. Configure ISP to run EBGP with SanJose1 and SanJose2. Enter the following commands on ISP.

ISP

```
ISP# conf t
ISP(config)#router bgp 200
ISP(config-router)#neighbor 192.168.1.6 remote-as 64512
ISP(config-router)#neighbor 192.168.1.2 remote-as 64512
ISP(config-router)#network 192.168.100.0
```

a. Configure a discard static route for the 172.16.0.0/16 network. Any packets that do not have a more specific match (longer match) for a 172.16.0.0 subnet will be dropped instead of sent to the ISP. Later in this lab we will configure a default route to the ISP.

SanJose1

```
SanJose1# conf t
SanJose1(config)#ip route 172.16.0.0 255.255.0.0 null0
```

b. Configure SanJose1 as an EBGP peer to ISP.

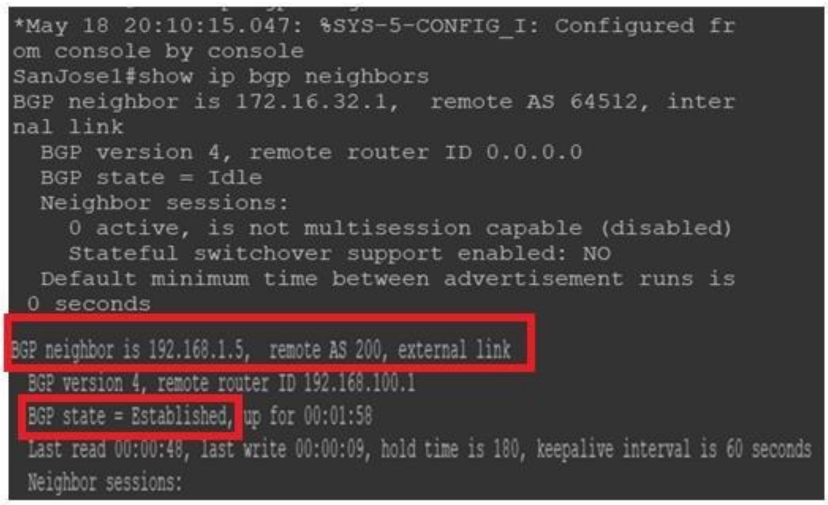
SanJose1

```
SanJose1# conf t
SanJose1(config)#router bgp 64512
SanJose1(config-router)#neighbor 192.168.1.5 remote-as 200
SanJose1(config-router)#network 172.16.0.0
```

c. Use the **show ip bgp neighbors** command to verify that SanJose1 and ISP have reached the established state. Troubleshoot if necessary.

SanJose1

```
SanJose1#show ip bgp neighbors
```



```
*May 18 20:10:15.047: %SYS-5-CONFIG_I: Configured fr
om console by console
SanJose1#show ip bgp neighbors
BGP neighbor is 172.16.32.1,  remote AS 64512, inter
nal link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Idle
  Neighbor sessions:
    0 active, is not multisession capable (disabled)
    Stateful switchover support enabled: NO
  Default minimum time between advertisement runs is
 0 seconds
BGP neighbor is 192.168.1.5,  remote AS 200, external link
  BGP version 4, remote router ID 192.168.100.1
  BGP state = Established, up for 00:01:58
  Last read 00:00:48, last write 00:00:09, hold time is 180, keepalive interval is 60 seconds
  Neighbor sessions:
```

d. Configure a discard static route for 172.16.0.0/16 on SanJose2 and as an EBGP peer to ISP.

SanJose2

```
SanJose2# conf t
SanJose2(config)#ip route 172.16.0.0 255.255.0.0 null0
SanJose2(config)#router bgp 64512
SanJose2(config-router)#neighbor 192.168.1.1 remote-as 200
SanJose2(config-router)#network 172.16.0.0
```

Step 5: View BGP summary output.

SanJose2

```
SanJose2#show ip bgp summary
BGP router identifier 172.16.32.1, local AS number 64512
BGP table version is 3, main routing table version 3
2 network entries using 288 bytes of memory
2 path entries using 160 bytes of memory
2/2 BGP path/bestpath attribute entries using 272 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 744 total bytes of memory
BGP activity 2/0 prefixes, 2/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.64.1     4        64512       0       0        1    0    0 never    Idle
192.168.1.1     4          200       9       8        3    0    0 00:02:46        1
SanJose2#
```

Step 6: Verify which path the traffic takes.

e. Clear the IP BGP conversation with the **clear ip bgp *** command on ISP. Wait for the conversations to reestablish with each SanJose router.

ISP

```
ISP# clear ip bgp *
```

```
*May 18 20:26:37.679: %BGP-5-ADJCHANGE: neighbor 192.168.1.2 Down User reset
*May 18 20:26:37.683: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.1.2 IPv4 Unicast topology base removed from session  User reset
*May 18 20:26:37.687: %BGP-5-ADJCHANGE: neighbor 192.168.1.6 Down User reset
*May 18 20:26:37.691: %BGP_SESSION-5-ADJCHANGE: neighbor 192.168.1.6 IPv4 Unicast topology base removed from session  User reset
*May 18 20:26:38.159: %BGP-5-ADJCHANGE: neighbor 192.168.1.6 Up
*May 18 20:26:38.163: %BGP-5-ADJCHANGE: neighbor 192.168.1.2 Up
ISP#
```

f. Test whether ISP can ping the loopback 0 address of 172.16.64.1 on SanJose1 and the serial link between SanJose1 and SanJose2, 172.16.1.1.

ISP

```
ISP# ping 172.16.64.1
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.64.1, timeout is 2 seconds:
UUUUU
Success rate is 0 percent (0/5)
ISP#
```

```
ISP# ping 172.16.1.1
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
ISP#
```

g. Now ping from ISP to the loopback 0 address of 172.16.32.1 on SanJose2 and the serial link between SanJose1 and SanJose2, 172.16.1.2.

ISP

```
ISP# ping 172.16.32.1
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.32.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/27/32 ms
ISP#
```

```
ISP# ping 172.16.1.2
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/26/32 ms
ISP#
```

h. Issue the **show ip bgp** command on ISP to verify BGP routes and metrics.

ISP `ISP# Show ip bgp`

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/26/32 ms
ISP#Show ip bgp
BGP table version is 3, local router ID is 192.168.100.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  172.16.0.0       192.168.1.2              0             0 64512 i
 *                    192.168.1.6              0             0 64512 i
 *>  192.168.100.0    0.0.0.0                  0         32768 i
ISP#
```

i. At this point, the ISP router should be able to get to each network connected to SanJose1 and SanJose2 from the loopback address 192.168.100.1. Use the extended **ping** command and specify the source address of ISP Lo0 to test.

ISP

`ISP# ping 172.16.1.1 source 192.168.100.1`

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/44/52 ms
ISP#
```

`ISP# ping 172.16.32.1 source 192.168.100.1`

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.32.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/16 ms
ISP#
```

`ISP# ping 172.16.1.2 source 192.168.100.1`

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.2, timeout is 2 seconds:
Packet sent with a source address of 192.168.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/12 ms
ISP#
```

`ISP# ping 172.16.64.1 source 192.168.100.1`

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.64.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/16 ms
ISP#
```

Step 7: Configure the BGP next-hop-self feature.

a. Issue the following commands on the ISP router.

ISP

```
ISP# conf t
ISP(config)# router bgp 200
ISP(config-router)#network 192.168.1.0 mask 255.255.255.252
ISP(config-router)#network 192.168.1.4 mask 255.255.255.252
```

b. Issue the **show ip bgp** command to verify that the ISP is correctly injecting its own WAN links into BGP.

ISP

```
ISP# show ip bgp
```

```
BGP table version is 5, local router ID is 192.168.100.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  172.16.0.0       192.168.1.2              0             0 64512 i
 *                    192.168.1.6              0             0 64512 i
 *>  192.168.1.0/30   0.0.0.0                  0         32768 i
 *>  192.168.1.4/30   0.0.0.0                  0         32768 i
 *>  192.168.100.0    0.0.0.0                  0         32768 i
ISP#
```

c. Verify on SanJose1 and SanJose2 that the opposite WAN link is included in the routing table. The output from SanJose2 is as follows

SanJose2

```
SanJose2# show ip route
```

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

Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
S        172.16.0.0/16 is directly connected, Null0
C        172.16.1.0/24 is directly connected, Serial5/1
L        172.16.1.2/32 is directly connected, Serial5/1
C        172.16.32.0/24 is directly connected, Loopback0
L        172.16.32.1/32 is directly connected, Loopback0
      192.168.1.0/24 is variably subnetted, 3 subnets, 2 masks
C        192.168.1.0/30 is directly connected, Serial5/2
L        192.168.1.2/32 is directly connected, Serial5/2
B        192.168.1.4/30 [20/0] via 192.168.1.1, 00:02:35
B     192.168.100.0/24 [20/0] via 192.168.1.1, 01:13:23
SanJose2#
```

d. To better understand the **next-hop-self** command we will remove ISP advertising its two WAN links and shutdown the WAN link between ISP and SanJose2. The only possible path from SanJose2 to ISP's 192.168.100.0/24 is through SanJose1.

ISP

```
ISP# conf t
ISP(config)#router bgp 200
ISP(config-router)#no network 192.168.1.0 mask 255.255.255.252
ISP(config-router)#no network 192.168.1.4 mask 255.255.255.252
ISP(config-router)#exit
ISP(config)#interface serial0/1
ISP(config-if)#shutdown
```

e. Display SanJose2's BGP table using the **show ip bgp** command and the IPv4 routing table with **show ip route**.

SanJose2

```
SanJose2# show ip bgp
```

```
   Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.0.0       0.0.0.0                  0         32768 i
* i                 172.16.64.1              0    100      0 i
*> 192.168.1.4/30   192.168.1.1              0             0 200 i
* i                 192.168.1.5              0    100      0 200 i
* i192.168.100.0    192.168.1.5              0    100      0 200 i
*>                  192.168.1.1              0             0 200 i
SanJose2#
```

```
SanJose2# show ip route
```

```
SanJose2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 00:15:45, Serial0/1
```

f. Issue the **next-hop-self** command on SanJose1 and SanJose2 to advertise themselves as the next hop to their IBGP peer.

SanJose1

```
SanJose1#conf t
SanJose1(config)#router bgp 64512
SanJose1(config-router)#neighbor 172.16.32.1 next-hop-self
```

SanJose2

```
SanJose2# conf t
SanJose2(config)#router bgp 64512
SanJose2(config-router)#neighbor 172.16.64.1 next-hop-self
```

g. Reset BGP operation on either router with the **clear ip bgp *** command.

SanJose1

```
SanJose1#clear ip bgp *
```

SanJose2

```
SanJose2#clear ip bgp *
```

h. After the routers have returned to established BGP speakers, issue the **show ip bgp** command on SanJose2 and notice that the next hop is now SanJose1 instead of ISP.

SanJose2

```
SanJose2#show ip bgp
```

```
SanJose2#show ip bgp
BGP table version is 1, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i192.168.100.0    172.16.64.1              0    100      0 200 i
SanJose2#
```

i. The **show ip route** command on SanJose2 now displays the 192.168.100.0/24 network because SanJose1 is the next hop, 172.16.64.1, which is reachable from SanJose2.

SanJose2

```
SanJose2#show ip route
```

```
SanJose2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 00:21:27, Serial0/1
B    192.168.100.0/24 [200/0] via 172.16.64.1, 00:00:48
```

j. Before configuring the next BGP attribute, restore the WAN link between ISP and SanJose3. This will change the BGP table and routing table on both routers. For example, SanJose2's routing table shows 192.168.100.0/24 will now have a better path through ISP.

ISP

```
ISP#conf t
ISP(config)#interface serial0/1
ISP(config-if)#no shutdown
```

SanJose2

```
SanJose2# show ip route
```

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

Gateway of last resort is not set

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 00:23:34, Serial0/1
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial0/0
B    192.168.100.0/24 [200/0] via 172.16.64.1, 00:02:54
SanJose2#
```

Step 8: Set BGP local preference.

a. Because the local preference value is shared between IBGP neighbors, configure a simple route map that references the local preference value on SanJose1 and SanJose2. This policy adjusts outbound traffic to prefer the link off the SanJose1 router instead of the metered T1 off SanJose2.

SanJose1

```
SanJose1# conf t
SanJose1(config)#route-map PRIMARY_T1_IN permit 10
SanJose1(config-route-map)#set local-preference 150
SanJose1(config-route-map)#exit
SanJose1(config)#router bgp 64512
SanJose1(config-router)#neighbor 192.168.1.5 route-map PRIMARY_T1_IN
in
```

SanJose2

```
SanJose2# conf t
SanJose2(config)#route-map SECONDARY_T1_IN permit 10
SanJose2(config-route-map)#set local-preference 125
SanJose2(config-route-map)#exit
SanJose2(config)#router bgp 64512
SanJose2(config-router)#neighbor 192.168.1.1 route-map
SECONDARY_T1_IN in
```

b. Use the **clear ip bgp * soft** command after configuring this new policy. When the conversations have been reestablished, issue the **show ip bgp** command on SanJose1 and SanJose2.

SanJose1

```
SanJose1# clear ip bgp * soft
```

SanJose2

```
SanJose2# clear ip bgp * soft
```

SanJose1

```
SanJose1# show ip bgp
```

```
SanJose1#show ip bgp
BGP table version is 7, local router ID is 172.16.64.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i172.16.0.0       172.16.32.1              0    100      0 i
*>                  0.0.0.0                  0         32768 i
*> 192.168.100.0    192.168.1.5              0    150      0 200 i
```

SanJose2

`SanJose2# show ip bgp`

```
SanJose2#show ip bgp
BGP table version is 5, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.0.0       0.0.0.0                  0         32768 i
* i                 172.16.64.1              0    100      0 i
*  192.168.100.0    192.168.1.1              0    125      0 200 i
*>i                 172.16.64.1              0    150      0 200 i
```

Step 9: Set BGP MED.

a. In the previous step we saw that SanJose1 and SanJose2 will route traffic for 192.168.100.0/24 using the link between SanJose1 and ISP. Examine what the return path ISP takes to reach AS 64512. Notice that the return path is different from the original path. This is known as asymmetric routing and is not necessarily an unwanted trait.

ISP

`ISP#show ip bgp`

```
ISP#show ip bgp
BGP table version is 12, local router ID is 192.168.100.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  172.16.0.0       192.168.1.2              0             0 64512 i
*>                  192.168.1.6              0             0 64512 i
*> 192.168.100.0    0.0.0.0                  0         32768 i
ISP#
```

```
ISP# show ip route
```

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

Gateway of last resort is not set

B    172.16.0.0/16 [20/0] via 192.168.1.6, 00:23:53
     192.168.1.0/30 is subnetted, 2 subnets
C       192.168.1.0 is directly connected, Serial0/1
C       192.168.1.4 is directly connected, Serial0/0
C    192.168.100.0/24 is directly connected, Loopback0
ISP#
```

b. Use an extended **ping** command to verify this situation. Specify the **record** option and compare your output to the following. Notice the return path using the exit interface 192.168.1.1 to SanJose2.

SanJose2

```
SanJose2# Ping
```

```
Target IP address: 192.168.100.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 172.16.32.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: record
Number of hops [ 9 ]:
Loose, Strict, Record, Timestamp, Verbose[RV]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.32.1
Packet has IP options:  Total option bytes= 39, padded length=40
```

```
Record route: <*>
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
```

```
Record route:
  (172.16.1.2)
  (192.168.1.6)
  (192.168.100.1)
  (192.168.1.5)
  (172.16.1.1)
  (172.16.32.1) <*>
  (0.0.0.0)
  (0.0.0.0)
  (0.0.0.0)
End of list
```

```
Reply to request 1 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list
```

```
Reply to request 2 (16 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list
```

```
Reply to request 3 (24 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list
```

```
Reply to request 4 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/8/24 ms
```

c. Create a new policy to force the ISP router to return all traffic via SanJose1. Create a second route map utilizing the MED (metric) that is shared between EBGP neighbors.

SanJose1

```
SanJose1# conf t
SanJose1(config)#route-map PRIMARY_T1_MED_OUT permit 10
SanJose1(config-route-map)#set Metric 50
SanJose1(config-route-map)#exit
SanJose1(config)#router bgp 64512
SanJose1(config-router)#neighbor 192.168.1.5 route-map PRIMARY_T1_MED_OUT
out
```

SanJose2

```
SanJose2# conf t
SanJose2(config)#route-map SECONDARY_T1_MED_OUT permit 10
SanJose2(config-route-map)#set Metric 75
SanJose2(config-route-map)#exit
SanJose2(config)#router bgp 64512
SanJose2(config-router)#neighbor 192.168.1.1 route-map
                        SECONDARY_T1_MED_OUT out
```

d. Use the **clear ip bgp * soft** command after issuing this new policy. Issuing the **show ip bgp** command as follows on SanJose1 or SanJose2 does not indicate anything about this newly defined policy.

SanJose1

```
SanJose1# clear ip bgp * soft
```

SanJose2

```
SanJose2# clear ip bgp * soft
```

SanJose1

```
SanJose1# show ip bgp
```

```
SanJose1#show ip bgp
BGP table version is 7, local router ID is 172.16.64.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i172.16.0.0       172.16.32.1              0    100      0 i
*>                  0.0.0.0                  0         32768 i
*> 192.168.100.0    192.168.1.5              0    150      0 200 i
```

SanJose2

```
SanJose2# show ip bgp
```

```
SanJose2#show ip bgp
BGP table version is 5, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.0.0       0.0.0.0                  0         32768 i
* i                 172.16.64.1              0    100      0 i
*  192.168.100.0    192.168.1.1              0    125      0 200 i
*>i                 172.16.64.1              0    150      0 200 i
```

e. Reissue an extended **ping** command with the **record** command. Notice the change in return path using the exit interface 192.168.1.5 to SanJose1.

SanJose2

`SanJose2# ping`

```
SanJose2#ping
Protocol [ip]:
Target IP address: 192.168.100.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 172.16.32.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: record
Number of hops [ 9 ]:
Loose, Strict, Record, Timestamp, Verbose[RV]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.32.1
 Packet has IP options:  Total option bytes= 39, padded length=40
  Record route: <*>
    (0.0.0.0)
    (0.0.0.0)
    (0.0.0.0)
    (0.0.0.0)
    (0.0.0.0)
    (0.0.0.0)
    (0.0.0.0)
    (0.0.0.0)
    (0.0.0.0)

 Reply to request 0 (12 ms).  Received packet has options
  Total option bytes= 40, padded length=40
```

```
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list

Reply to request 1 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
```

```
Reply to request 1 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list
```

```
Reply to request 2 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list
```

```
Reply to request 3 (24 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list
```

```
Reply to request 4 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.2)
   (192.168.1.6)
   (192.168.100.1)
   (192.168.1.5)
   (172.16.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/24 ms
```

ISP

```
ISP# show ip bgp
```

```
ISP#show ip bgp
BGP table version is 14, local router ID is 192.168.100.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  172.16.0.0       192.168.1.2             75             0 64512 i
*>                  192.168.1.6             50             0 64512 i
*> 192.168.100.0    0.0.0.0                  0         32768 i
```

Step 10: Establish a default route.

a. Configure ISP to inject a default route to both SanJose1 and SanJose2 using BGP using the **default-originate** command. This command does not require the presence of 0.0.0.0 in the ISP router. Configure the 10.0.0.0/8 network which will not be advertised using BGP. This network will be used to test the default route on SanJose1 and SanJose2.

ISP

```
ISP# conf t
ISP(config)#router bgp 200
ISP(config-router)#neighbor 192.168.1.6 default-originate
ISP(config-router)#neighbor 192.168.1.2 default-originate
ISP(config-router)#exit
ISP(config)#interface loopback 10
ISP(config-if)#ip address 10.0.0.1 255.255.255.0
```

b. Verify that both routers have received the default route by examining the routing tables on SanJose1 and SanJose2. Notice that both routers prefer the route between SanJose1 and ISP.

SanJose1

```
SanJose1#show ip route
```

```
SanJose1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.1.5 to network 0.0.0.0

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
D       172.16.32.0/24 [90/2297856] via 172.16.1.2, 01:07:30, Serial0/1
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
C       172.16.64.0/24 is directly connected, Loopback0
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.4 is directly connected, Serial0/0
B    192.168.100.0/24 [20/0] via 192.168.1.5, 00:49:25
B*   0.0.0.0/0 [20/0] via 192.168.1.5, 00:00:14
```

SanJose2

```
SanJose2#show ip route
```

```
SanJose2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 172.16.64.1 to network 0.0.0.0

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 01:09:26, Serial0/1
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial0/0
B    192.168.100.0/24 [200/0] via 172.16.64.1, 00:36:07
B*   0.0.0.0/0 [200/0] via 172.16.64.1, 00:02:09
```

c. The preferred default route is by way of SanJose1 because of the higher local preference attribute configured on SanJose1 earlier.

SanJose2

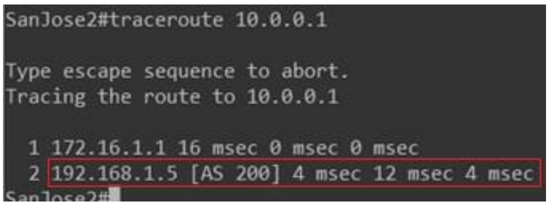
```
SanJose2#show ip bgp
```

```
SanJose2#show ip bgp
BGP table version is 7, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i0.0.0.0          172.16.64.1              0    150      0 200 i
*                   192.168.1.1              0    125      0 200 i
*> 172.16.0.0       0.0.0.0                  0         32768 i
* i                 172.16.64.1              0    100      0 i
*  192.168.100.0    192.168.1.1              0    125      0 200 i
*>i                 172.16.64.1              0    150      0 200 i
```

d. Using the traceroute command verify that packets to 10.0.0.1 is using the default route through SanJose1.

SanJose2

```
SanJose2#traceroute 10.0.0.1
```



```
SanJose2#traceroute 10.0.0.1

Type escape sequence to abort.
Tracing the route to 10.0.0.1

  1 172.16.1.1 16 msec 0 msec 0 msec
  2 192.168.1.5 [AS 200] 4 msec 12 msec 4 msec
SanJose2#
```

e. Next, test how BGP adapts to using a different default route when the path between SanJose1 and ISP goes down.

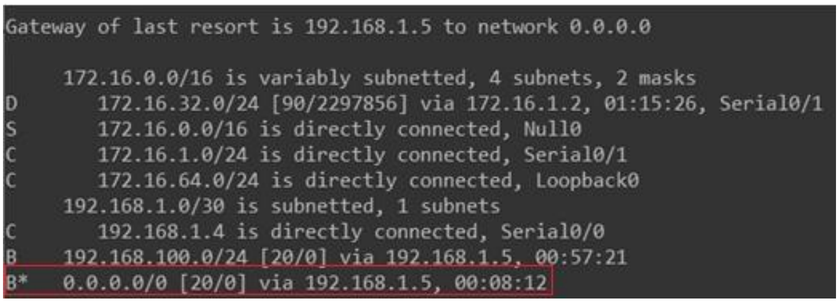
ISP

```
ISP(config)#interface serial0/0
ISP(config-if)#shutdown
```

f. Verify that both routers are modified their routing tables with the default route using the path between SanJose2 and ISP.

SanJose1

```
SanJose1#show ip route
```



```
Gateway of last resort is 192.168.1.5 to network 0.0.0.0

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
D       172.16.32.0/24 [90/2297856] via 172.16.1.2, 01:15:26, Serial0/1
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
C       172.16.64.0/24 is directly connected, Loopback0
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.4 is directly connected, Serial0/0
B    192.168.100.0/24 [20/0] via 192.168.1.5, 00:57:21
B*   0.0.0.0/0 [20/0] via 192.168.1.5, 00:08:12
```

SanJose2

```
SanJose2#show ip route
```

```
SanJose2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.1.1 to network 0.0.0.0

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 01:21:57, Serial0/1
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial0/0
B    192.168.100.0/24 [20/0] via 192.168.1.1, 00:06:19
B*   0.0.0.0/0 [20/0] via 192.168.1.1, 00:06:19
```

g. Verify the new path using the traceroute command to 10.0.0.1 from SanJose1. Notice the default route is now through SanJose2.

SanJose1

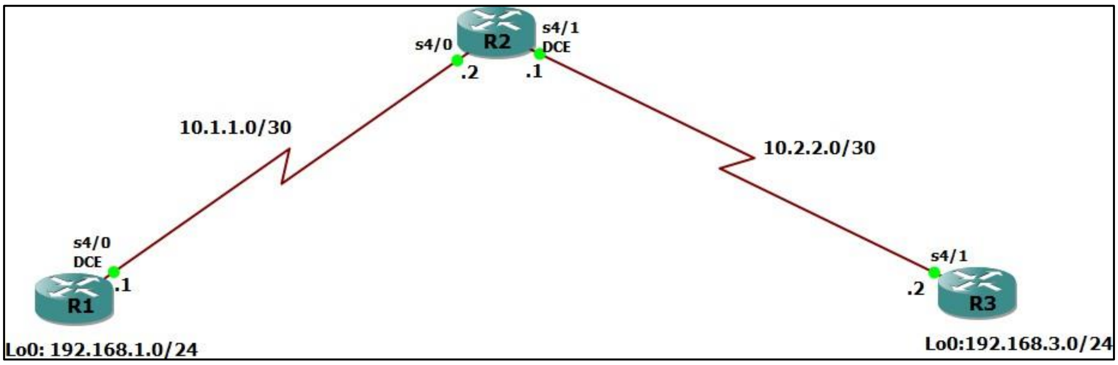
```
SanJose1#trace 10.0.0.1
```

```
SanJose1#trace 10.0.0.1

Type escape sequence to abort.
Tracing the route to 10.0.0.1

  1 172.16.1.2 0 msec 0 msec 0 msec
  2 192.168.1.1 [AS 200] 16 msec 4 msec 12 msec
```

Practical -4

Title: - Secure the Management Plane.

Topology: -



Topology for securing management plane

Step 1: Configure loopbacks and assign addresses.

a. Cable the network as shown in the topology diagram. Erase the startup configuration and reload each router to clear previous configurations. Using the addressing scheme in the diagram, apply the IP addresses to the interfaces on the R1, R2, and R3 routers.

R1

```
conf t
hostname R1
interface Loopback 0
description R1 LAN
ip address 192.168.1.1 255.255.255.0
exit
interface Serial4/0
description R1 --> R2
ip address 10.1.1.1 255.255.255.252
clock rate 128000
no shutdown
exit
end
```

R2

```
conf t
hostname R2
interface Serial4/0
description R2 --> R1
ip address 10.1.1.2 255.255.255.252
no shutdown
exit
interface Serial4/1
description R2 --> R3
ip address 10.2.2.1 255.255.255.252
clock rate 128000
no shutdown
exit
end
```

R3

```
conf t
hostname R3
interface Loopback0
description R3 LAN
ip address 192.168.3.1 255.255.255.0
exit
interface Serial4/1
description R3 --> R2
ip address 10.2.2.2 255.255.255.252
no shutdown
exit
end
```

Step 2: Configure static routes.

a. On R1, configure a default static route to ISP.

R1

```
conf t
R1(config)#ip route 0.0.0.0 0.0.0.0 10.1.1.2
```

b. On R3, configure a default static route to ISP.

R3

```
conf t
R3(config)#ip route 0.0.0.0 0.0.0.0 10.2.2.1
```

c. On R2, configure two static routes.

R2

```
conf t
R2(config)#ip route 192.168.1.0 255.255.255.0 10.1.1.1
R2(config)#ip route 192.168.3.0 255.255.255.0 10.2.2.2
```

d. From the R1 router, run the following Tcl script to verify connectivity.

foreach address {

192.168.1.1
10.1.1.1
10.1.1.2
10.2.2.1
10.2.2.2
192.168.3.1
} { ping $address }

R1

```
R1#tclsh
foreach address {
192.168.1.1
10.1.1.1
10.1.1.2
10.2.2.1
10.2.2.2
192.168.3.1
} { ping $address }
```

```
R1#tclsh
R1(tcl)#foreach address {
+>(tcl)#192.168.1.1
+>(tcl)#10.1.1.1
+>(tcl)#10.1.1.2
+>(tcl)#10.2.2.1
+>(tcl)#10.2.2.2
+>(tcl)#192.168.3.1
+>(tcl)#} { ping $address }
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/8 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/63/68 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/28/36 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/33/52 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/61/68 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 52/60/64 ms
```

Step 3: Secure management access.

a. On R1, use the **security passwords** command to set a minimum password length of 10 characters.

R1

```
conf t
R1(config)#security passwords min-length 10
```

b. Configure the enable secret encrypted password on both routers.

R1

```
conf t
R1(config)#enable secret class12345
```

c. Configure a console password and enable login for routers. For additional security, the **exec-timeout** command causes the line to log out after 5 minutes of inactivity. The **logging synchronous** command prevents console messages from interrupting command entry.

R1

```
R1#Conf  t
R1(config)#line console 0
R1(config-line)#password ciscoconpass
R1(config-line)#exec-timeout 5 0
R1(config-line)#login
R1(config-line)#logging synchronous
R1(config-line)#exit
```

d. Configure the password on the vty lines for router R1.

R1

```
R1#conf t
R1(config)#line vty 0 4
R1(config-line)#password ciscovtypass
R1(config-line)#exec-timeout 5 0
R1(config-line)#login
R1(config-line)#exit
```

e. The aux port is a legacy port used to manage a router remotely using a modem and is hardly ever used. Therefore, disable the aux port.

R1

```
R1# conf t
R1(config-line)#line aux 0
R1(config-line)#no exec
R1(config-line)#end
```

f. Enter privileged EXEC mode and issue the **show run** command. Can you read the enable secret password? Why or why not?

g. Use the **service password-encryption** command to encrypt the line console and vty passwords.

R1

```
R1#conf t
R1(config)#service password-encryption
```

h. Issue the **show run** command. Can you read the console, aux, and vty passwords? Why or why not?

i. Configure a warning to unauthorized users with a message-of-the-day (MOTD) banner using the **banner motd** command. When a user connects to one of the routers, the MOTD banner appears before the login prompt. In this example, the dollar sign ($) is used to start and end the message.

R1

```
R1#conf t
R1(config)#banner motd $Unauthorized access strictly prohibited!$
R1(config)#exit
```

j. Issue the show run command. What does the $ convert to in the output?

k. Exit privileged EXEC mode using the **disable** or **exit** command and press **Enter** to get started. Does the MOTD banner look like what you created with the **banner motd** command? If the MOTD banner is not as you wanted it, recreate it using the **banner motd** command.

l. Repeat the configuration portion of **steps 3a through 3k on router R3.**

Step 4: Configure enhanced username password security.

a. To create local database entry encrypted to level 4 (SHA256), use the **username** *name* **secret** *password* global configuration command. In global configuration mode, enter the following command:

R1

```
R1#conf t
R1(config)#username JR-ADMIN secret class12345
R1(config)#username ADMIN secret class54321
```

b. Set the console line to use the locally defined login accounts.

R1

```
R1#conf t
```

```
R1(config)#line console 0
R1(config-line)#login local
R1(config-line)#exit
```

c. Set the vty lines to use the locally defined login accounts.

R1

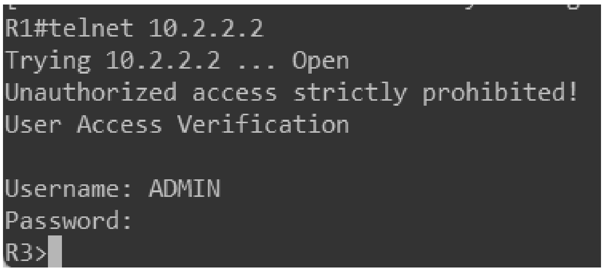
```
R1# conf t
R1(config)#line vty 0 4
R1(config-line)#login local
R1(config-line)#end
```

d. Repeat the steps 4a to 4c on R3.

e. To verify the configuration, telnet to R3 from R1 and login using the **ADMIN** local database account.

R1

```
R1#telnet 10.2.2.2
```



Step 5: Enabling AAA RADIUS Authentication with Local User for Backup.

a. Always have local database accounts created before enabling AAA. Since we created two local database accounts in the previous step, then we can proceed and enable AAA on R1

R1

```
R1#conf t
R1(config)#aaa new-model
```

b. Configure the specifics for the first RADIUS server located at 192.168.1.101. Use **RADIUS-1-pa55w0rd** as the server password.

R1

```
R1#conf t
R1(config)#radius server RADIUS-1
R1(config-radius-server)#address ipv4 192.168.1.101
R1(config-radius-server)#key RADIUS-1-pa55w0rd
R1(config-radius-server)#exit
```

c. Configure the specifics for the second RADIUS server located at 192.168.1.102. Use **RADIUS-2-pa55w0rd** as the server password.

R1

```
R1# conf t
R1(config)#radius server RADIUS-2
R1(config-radius-server)#address ipv4 192.168.1.102
R1(config-radius-server)#key RADIUS-2-pa55w0rd
R1(config-radius-server)#exit
```

d. Assign both RADIUS servers to a server group.

R1

```
conf t
R1(config)#aaa group server radius RADIUS-GROUP
R1(config-sg-radius)#server name RADIUS-1
R1(config-sg-radius)#server name RADIUS-2
R1(config-sg-radius)#exit
```

e. Enable the default AAA authentication login to attempt to validate against the server group. If they are not available, then authentication should be validated against the local database..

R1

```
conf t
R1(config)#aaa authentication login default group RADIUS-GROUP local
```

f. Enable the default AAA authentication Telnet login to attempt to validate against the server group. If they are not available, then authentication should be validated against a case sensitive local database.

R1

```
conf t
R1(config)#aaa authentication login TELNET-LOGIN group RADIUS-GROUP local-
case
```

g. Alter the VTY lines to use the TELNET-LOGIN AAA authentiaito0n method.

R1

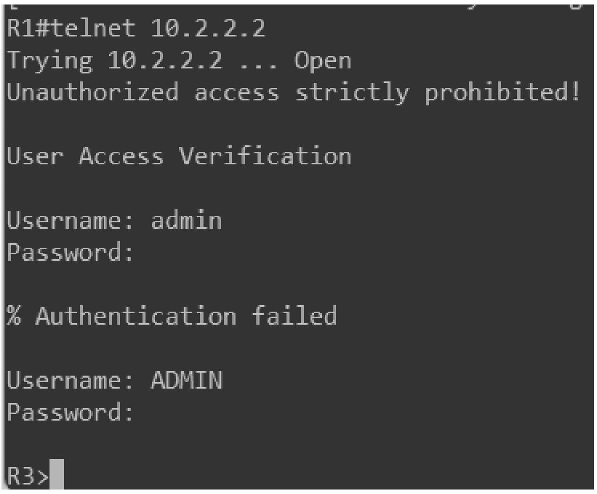
```
conf t
R1(config)#line vty 0 4
R1(config-line)#login authentication TELNET-LOGIN
R1(config-line)#exit
```

h. Repeat the steps 5a to 5g on R3.

i. To verify the configuration, telnet to R3 from R1 and login using the ADMIN local database account.

R1

```
R1#telnet 10.2.2.2
```



Step 6: Enabling secure remote management using SSH.

a. SSH requires that a device name and a domain name be configured. Since the router already has a name assigned, configure the domain name.

R1

```
conf t
R1(config)#ip domain-name ccnasecurity.com
```

b. The router uses the RSA key pair for authentication and encryption of transmitted SSH data. Although optional it may be wise to erase any existing key pairs on the router.

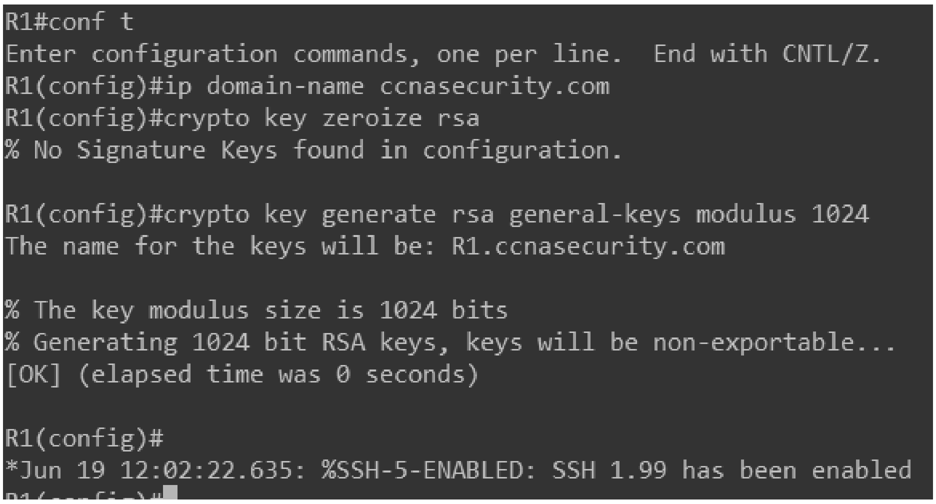
R1

```
conf t
R1(config)#crypto key zeroize rsa
```

c. Generate the RSA encryption key pair for the router. Configure the RSA keys with **1024** for the number of modulus bits. The default is 512, and the range is from 360 to 2048.

R1

```
conf t
R1(config)#crypto key generate rsa general-keys modulus 1024
```



```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip domain-name ccnasecurity.com
R1(config)#crypto key zeroize rsa
% No Signature Keys found in configuration.

R1(config)#crypto key generate rsa general-keys modulus 1024
The name for the keys will be: R1.ccnasecurity.com

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)

R1(config)#
*Jun 19 12:02:22.635: %SSH-5-ENABLED: SSH 1.99 has been enabled
```

d. Cisco routers support two versions of SSH

Configure SSH version 2 on R1.

R1

```
conf t
R1(config)#ip ssh version 2
```

e. Configure the vty lines to use only SSH connections.

R1

```
conf t
R1(config)#line vty 0 4
R1(config-line)#transport input ssh
R1(config-line)#end
```

f. Verify the SSH configuration using the **show ip ssh** command.

R1

```
R1#Show ip ssh
```

```
R1#Show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
IOS Keys in SECSH format(ssh-rsa, base64 encoded):
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgQC02r0aUSfV82Ry4VXIWSL/Sr4oO6MeP1e2+N9eardD
j5+KR0rQMJ+xfux1FanEW3r5IpJuuN7eHt7gJVbPimwAZqTTWtGSfgw7tWRvQN4N+5U+DI2p2JUkbrul
zEj9eQJg3zMidSDhAIRAII1sd+UVJao8beOw2Ord8QYbrpSUUQ==
R1#
```

g. **Repeat the steps 6a to 6f on R3.**

h. Although a user can SSH from a host using the SSH option of TeraTerm of PuTTY, a router can also SSH to another SSH enabled device. SSH to R3 from R1.

R1

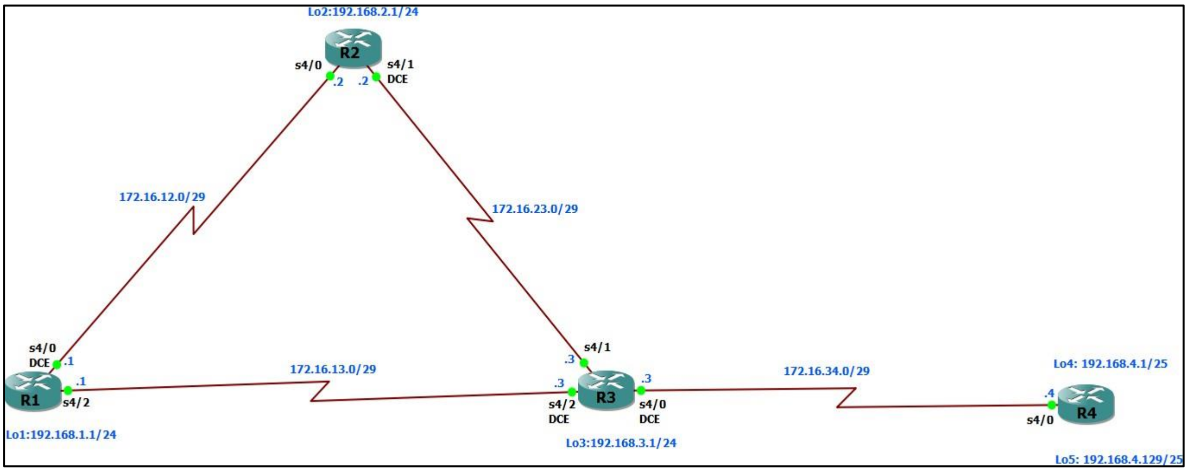
```
R1# ssh -l ADMIN 10.2.2.2
R3> en
```

```
R1#ssh -l ADMIN 10.2.2.2
Password:
Unauthorized access strictly prohibited!R3>
R3>
R3>
R3>
R3>en
Password:
```

Practical -5

Title: - Configure and Verify Path Control Using PBR.

Topology: -



Topology for PBR

Step 1: Configure loopbacks and assign addresses.

a. Cable the network as shown in the topology diagram. Erase the startup configuration, and reload each router to clear previous configurations.

b. Using the addressing scheme in the diagram, create the loopback interfaces and apply IP addresses to these and the serial interfaces on R1, R2, R3, and R4. On the serial interfaces connecting R1 to R3 and R3 to R4, specify the bandwidth as 64 Kb/s and set a clock rate on the DCE using the **clock rate 64000** command. On the serial interfaces connecting R1 to R2 and R2 to R3, specify the bandwidth as 128 Kb/s and set a clock rate on the DCE using the **clock rate 128000** command.

Router R1

```
conf t
hostname R1
interface Lo1
description R1 LAN
ip address 192.168.1.1 255.255.255.0
interface Serial4/0
description R1 --> R2
ip address 172.16.12.1 255.255.255.248
clock rate 128000
bandwidth 128
no shutdown
interface Serial4/2
description R1 --> R3
ip address 172.16.13.1 255.255.255.248
```

```
bandwidth 64
no shutdown
end
```

Router R2

```
conf t
hostname R2
interface Lo2
description R2 LAN
ip address 192.168.2.1 255.255.255.0
interface Serial4/0
description R2 --> R1
ip address 172.16.12.2 255.255.255.248
bandwidth 128
no shutdown
interface Serial4/1
description R2 --> R3
ip address 172.16.23.2 255.255.255.248
clock rate 128000
bandwidth 128
no shutdown
end
```

Router R3

```
conf t
hostname R3
interface Lo3
description R3 LAN
ip address 192.168.3.1 255.255.255.0
interface Serial4/2
description R3 --> R1
ip address 172.16.13.3 255.255.255.248
clock rate 64000
bandwidth 64
no shutdown
interface Serial4/1
description R3 --> R2
ip address 172.16.23.3 255.255.255.248
bandwidth 128
no shutdown
interface Serial4/0
description R3 --> R4
ip address 172.16.34.3 255.255.255.248
clock rate 64000
bandwidth 64
no shutdown
end
```

Router R4

```
conf t
hostname R4
interface Lo4
description R4 LAN A
ip address 192.168.4.1 255.255.255.128
interface Lo5
```

```
description R4 LAN B
ip address 192.168.4.129 255.255.255.128
interface Serial4/0
description R4 --> R3
ip address 172.16.34.4 255.255.255.248
bandwidth 64
no shutdown
end
```

c. Verify the configuration with the **show ip interface brief**, **show protocols,** and **show interfaces description** commands. The output from router R3 is shown here as an example.

Router R3

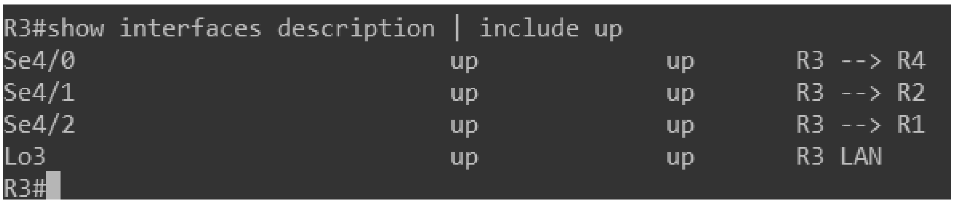
```
R3#show ip interface brief | include up
```

```
R3#show ip interface brief | include up
Serial4/0              172.16.34.3     YES manual up                    up
Serial4/1              172.16.23.3     YES manual up                    up
Serial4/2              172.16.13.3     YES manual up                    up
Loopback3              192.168.3.1     YES manual up                    up
R3#
*Jun 19 20:05:39.823: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial4/0, changed state to up
R3#
```

```
R3#show protocols
```

```
R3#show protocols
Global values:
  Internet Protocol routing is enabled
FastEthernet0/0 is administratively down, line protocol is down
FastEthernet1/0 is administratively down, line protocol is down
FastEthernet1/1 is administratively down, line protocol is down
Ethernet2/0 is administratively down, line protocol is down
Ethernet2/1 is administratively down, line protocol is down
Ethernet2/2 is administratively down, line protocol is down
Ethernet2/3 is administratively down, line protocol is down
Ethernet2/4 is administratively down, line protocol is down
Ethernet2/5 is administratively down, line protocol is down
Ethernet2/6 is administratively down, line protocol is down
Ethernet2/7 is administratively down, line protocol is down
GigabitEthernet3/0 is administratively down, line protocol is down
Serial4/0 is up, line protocol is up
  Internet address is 172.16.34.3/29
Serial4/1 is up, line protocol is up
  Internet address is 172.16.23.3/29
Serial4/2 is up, line protocol is up
  Internet address is 172.16.13.3/29
Serial4/3 is administratively down, line protocol is down
FastEthernet5/0 is administratively down, line protocol is down
Serial6/0 is administratively down, line protocol is down
```

```
R3#show interfaces description | include up
```



```
R3#show interfaces description | include up
Se4/0                          up             up       R3 --> R4
Se4/1                          up             up       R3 --> R2
Se4/2                          up             up       R3 --> R1
Lo3                            up             up       R3 LAN
R3#
```

Step 3: Configure basic EIGRP.

a. Implement EIGRP AS 1 over the serial and loopback interfaces as you have configured it for the other EIGRP labs.

b. Advertise networks 172.16.12.0/29, 172.16.13.0/29, 172.16.23.0/29, 172.16.34.0/29, 192.168.1.0/24, 192.168.2.0/24, 192.168.3.0/24, and 192.168.4.0/24 from their respective routers.

Router R1

```
conf t
router eigrp 1
network 192.168.1.0
network 172.16.12.0 0.0.0.7
network 172.16.13.0 0.0.0.7
no auto-summary
```

Router R2

```
conf t
router eigrp 1
network 192.168.2.0
network 172.16.12.0 0.0.0.7
network 172.16.23.0 0.0.0.7
no auto-summary
```

Router R3

```
conf t
router eigrp 1
network 192.168.3.0
network 172.16.13.0 0.0.0.7
network 172.16.23.0 0.0.0.7
network 172.16.34.0 0.0.0.7
no auto-summary
```

Router R4

```
conf t
router eigrp 1
network 192.168.4.0
network 172.16.34.0 0.0.0.7
no auto-summary
```

Step 4: Verify EIGRP connectivity.

a. Verify the configuration by using the **show ip eigrp neighbors** command to check which routers have EIGRP adjacencies.

Router R1

```
show ip eigrp neighbors
```

```
R1#show ip eigrp neighbors
*Jun 19 20:10:10.399: %SYS-5-CONFIG_I: Configured from console by console
R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   172.16.13.3             Se4/2                    12 00:00:26   48  2340  0  10
0   172.16.12.2             Se4/0                    13 00:00:40   61  1170  0  9
R1#
```

Router R2

```
show ip eigrp neighbors
```

```
R2#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
1   172.16.23.3             Se4/1                    14 00:01:05   48  1170  0  11
0   172.16.12.1             Se4/0                    13 00:01:18   53  1170  0  11
R2#
```

Router R3

```
show ip eigrp neighbors
```

```
R3#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
2   172.16.34.4             Se4/0                    10 00:01:36   65  2340  0  3
1   172.16.23.2             Se4/1                    14 00:01:47   45  1170  0  8
0   172.16.13.1             Se4/2                    11 00:01:47   47  2340  0  10
R3#
```

Router R4

```
show ip eigrp neighbors
```

```
R4#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
0   172.16.34.3             Se4/0                    11 00:02:21   58  2340  0  9
R4#
```

b. Run the following Tcl script on all routers to verify full connectivity.

Router R1

```
tclsh
foreach address {
172.16.12.1
172.16.12.2
172.16.13.1
172.16.13.3
172.16.23.2
172.16.23.3
172.16.34.3
172.16.34.4
192.168.1.1
192.168.2.1
192.168.3.1
192.168.4.1
192.168.4.129
} { ping $address }
```

Step 5: Verify the current path.

Before you configure PBR, verify the routing table on R1.

a. On R1, use the **show ip route** command. Notice the next-hop IP address for all networks discovered by EIGRP.

Router R1

```
show ip route | begin Gateway
```

```
R1#show ip route | begin Gateway
Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 6 subnets, 2 masks
C        172.16.12.0/29 is directly connected, Serial4/0
L        172.16.12.1/32 is directly connected, Serial4/0
C        172.16.13.0/29 is directly connected, Serial4/2
L        172.16.13.1/32 is directly connected, Serial4/2
D        172.16.23.0/29 [90/21024000] via 172.16.12.2, 00:04:47, Serial4/0
D        172.16.34.0/29 [90/41024000] via 172.16.13.3, 00:04:47, Serial4/2
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Loopback1
L        192.168.1.1/32 is directly connected, Loopback1
D     192.168.2.0/24 [90/20640000] via 172.16.12.2, 00:04:46, Serial4/0
D     192.168.3.0/24 [90/21152000] via 172.16.12.2, 00:04:47, Serial4/0
      192.168.4.0/25 is subnetted, 2 subnets
D        192.168.4.0 [90/41152000] via 172.16.13.3, 00:04:35, Serial4/2
D        192.168.4.128 [90/41152000] via 172.16.13.3, 00:04:35, Serial4/2
```

b. On R4, use the **traceroute** command to the R1 LAN address and source the ICMP packet from R4 LAN A and LAN B.

Router R4

```
traceroute 192.168.1.1 source 192.168.4.1
```

```
R4#traceroute 192.168.1.1 source 192.168.4.1
Type escape sequence to abort.
Tracing the route to 192.168.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.34.3 48 msec 20 msec 28 msec
  2 172.16.23.2 68 msec 52 msec 56 msec
  3 172.16.12.1 80 msec 72 msec 80 msec
```

```
traceroute 192.168.1.1 source 192.168.4.129
```

```
R4#traceroute 192.168.1.1 source 192.168.4.129
Type escape sequence to abort.
Tracing the route to 192.168.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.34.3 20 msec 24 msec 24 msec
  2 172.16.23.2 60 msec 72 msec 52 msec
  3 172.16.12.1 72 msec 76 msec 76 msec
```

c. On R3, use the **show ip route** command and note that the preferred route from R3 to R1 LAN 192.168.1.0/24 is via R2 using the R3 exit interface S4/1.

Router R3

```
show ip route | begin Gateway
```

```
R3#show ip route | begin Gateway
Gateway of last resort is not set

      172.16.0.0/16 is variably subnetted, 7 subnets, 2 masks
D        172.16.12.0/29 [90/21024000] via 172.16.23.2, 00:07:29, Serial4/1
C        172.16.13.0/29 is directly connected, Serial4/2
L        172.16.13.3/32 is directly connected, Serial4/2
C        172.16.23.0/29 is directly connected, Serial4/1
L        172.16.23.3/32 is directly connected, Serial4/1
C        172.16.34.0/29 is directly connected, Serial4/0
L        172.16.34.3/32 is directly connected, Serial4/0
D     192.168.1.0/24 [90/21152000] via 172.16.23.2, 00:07:29, Serial4/1
D     192.168.2.0/24 [90/20640000] via 172.16.23.2, 00:07:29, Serial4/1
      192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.3.0/24 is directly connected, Loopback3
L        192.168.3.1/32 is directly connected, Loopback3
      192.168.4.0/25 is subnetted, 2 subnets
D        192.168.4.0 [90/40640000] via 172.16.34.4, 00:07:19, Serial4/0
D        192.168.4.128 [90/40640000] via 172.16.34.4, 00:07:19, Serial4/0
```

d. On R3, use the **show interfaces serial 4/2** and **show interfaces s4/1** commands.

Router R3

```
show interfaces serial4/2
```

```
R3#show interfaces serial4/2
Serial4/2 is up, line protocol is up
  Hardware is M4T
  Description: R3 --> R1
  Internet address is 172.16.13.3/29
  MTU 1500 bytes, BW 64 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Restart-Delay is 0 secs
  Last input 00:00:03, output 00:00:04, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/1/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 48 kilobits/sec
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     271 packets input, 20592 bytes, 0 no buffer
     Received 94 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
 --More--
```

```
show interfaces serial4/2 | include BW
```

```
R3#show interfaces serial4/2 | include BW
  MTU 1500 bytes, BW 64 Kbit/sec, DLY 20000 usec,
```

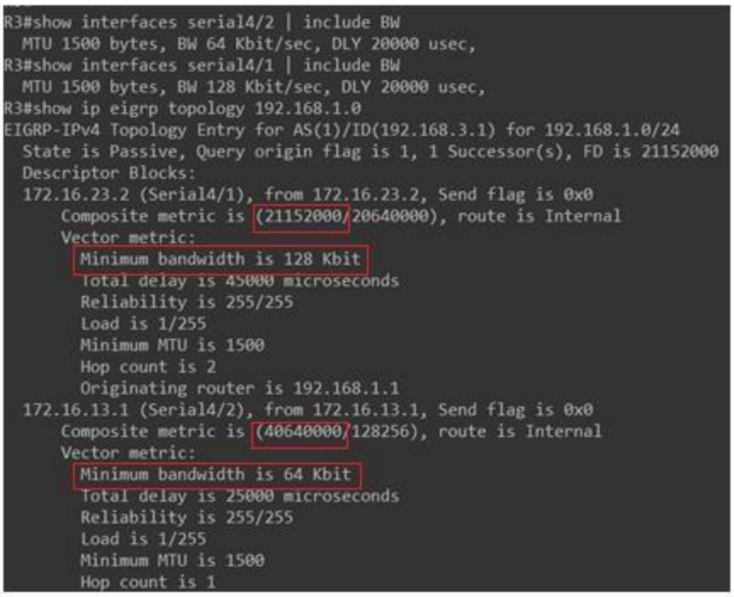
```
show interfaces serial4/1 | include BW
```

```
R3#show interfaces serial4/1 | include BW
  MTU 1500 bytes, BW 128 Kbit/sec, DLY 20000 usec,
```

e. Confirm that R3 has a valid route to reach R1 from its serial 0/0/0 interface using the **show ip eigrp topology 192.168.1.0** command.

Router R3

```
show ip eigrp topology 192.168.1.0
```



```
R3#show interfaces serial4/2 | include BW
  MTU 1500 bytes, BW 64 Kbit/sec, DLY 20000 usec,
R3#show interfaces serial4/1 | include BW
  MTU 1500 bytes, BW 128 Kbit/sec, DLY 20000 usec,
R3#show ip eigrp topology 192.168.1.0
EIGRP-IPv4 Topology Entry for AS(1)/ID(192.168.3.1) for 192.168.1.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 21152000
  Descriptor Blocks:
  172.16.23.2 (Serial4/1), from 172.16.23.2, Send flag is 0x0
      Composite metric is (21152000/20640000), route is Internal
      Vector metric:
        Minimum bandwidth is 128 Kbit
        Total delay is 45000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 2
        Originating router is 192.168.1.1
  172.16.13.1 (Serial4/2), from 172.16.13.1, Send flag is 0x0
      Composite metric is (40640000/128256), route is Internal
      Vector metric:
        Minimum bandwidth is 64 Kbit
        Total delay is 25000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
```

Step 6: Configure PBR to provide path control.

a. On router R3, create a standard access list called **PBR-ACL** to identify the R4 LAN B network.

Router R3

```
conf t
ip access-list standard PBR-ACL
remark ACL matches R4 LAN B traffic
permit 192.168.4.128 0.0.0.127
exit
```

b. Create a route map called **R3-to-R1** that matches PBR-ACL and sets the next-hop interface to the R1 serial 4/2 interface.

Router R3

```
conf t
route-map R3-to-R1 permit
description RM to forward LAN B traffic to R1
match ip address PBR-ACL
set ip next-hop 172.16.13.1
exit
```

c. **policy route-map** command on interface S4/0.

Router R3

```
conf t
interface s4/0
ip policy route-map R3-to-R1
end
```

d. On R3, display the policy and matches using the **show route-map** command.

Router R3

```
show route-map
```

```
R3#show route-map
route-map R3-to-R1, permit, sequence 10
  Match clauses:
    ip address (access-lists): PBR-ACL
  Set clauses:
    ip next-hop 172.16.13.1
  Policy routing matches: 0 packets, 0 bytes
```

Step 7: Test the policy.

a. On R3, create a standard ACL which identifies all of the R4 LANs.

Router R3

```
conf t
access-list 1 permit 192.168.4.0 0.0.0.255
exit
```

b. Enable PBR debugging only for traffic that matches the R4 LANs.

Router R3

```
debug ip policy ?
```

```
R3#debug ip policy ?
  <1-199>  Access list
  dynamic  dynamic PBR
  early    Early PBR
  <cr>
```

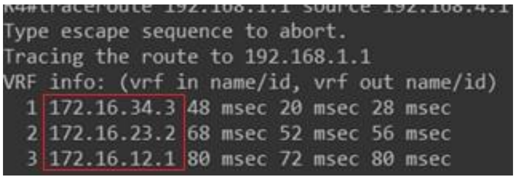
```
debug ip policy 1
```

```
R3#debug ip policy 1
Policy routing debugging is on for access list 1
```

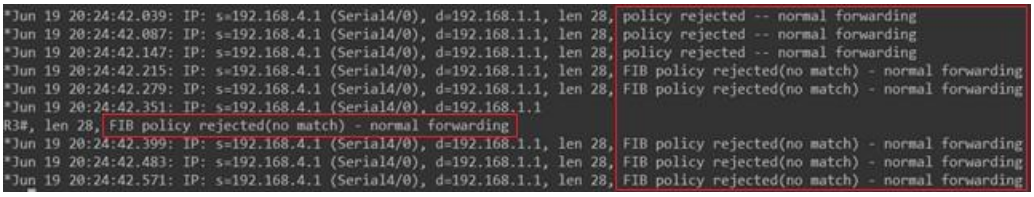
c. Test the policy from R4 with the **traceroute** command, using R4 LAN A as the source network.

Router R4

`traceroute 192.168.1.1 source 192.168.4.1`



```
R4#traceroute 192.168.1.1 source 192.168.4.1
Type escape sequence to abort.
Tracing the route to 192.168.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.34.3 48 msec 20 msec 28 msec
  2 172.16.23.2 68 msec 52 msec 56 msec
  3 172.16.12.1 80 msec 72 msec 80 msec
```

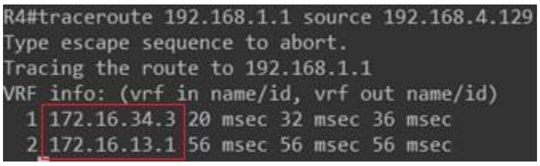


```
*Jun 19 20:24:42.039: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, policy rejected -- normal forwarding
*Jun 19 20:24:42.087: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, policy rejected -- normal forwarding
*Jun 19 20:24:42.147: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, policy rejected -- normal forwarding
*Jun 19 20:24:42.215: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, FIB policy rejected(no match) - normal forwarding
*Jun 19 20:24:42.279: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, FIB policy rejected(no match) - normal forwarding
*Jun 19 20:24:42.351: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1
R3#, len 28, FIB policy rejected(no match) - normal forwarding
*Jun 19 20:24:42.399: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, FIB policy rejected(no match) - normal forwarding
*Jun 19 20:24:42.483: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, FIB policy rejected(no match) - normal forwarding
*Jun 19 20:24:42.571: IP: s=192.168.4.1 (Serial4/0), d=192.168.1.1, len 28, FIB policy rejected(no match) - normal forwarding
```

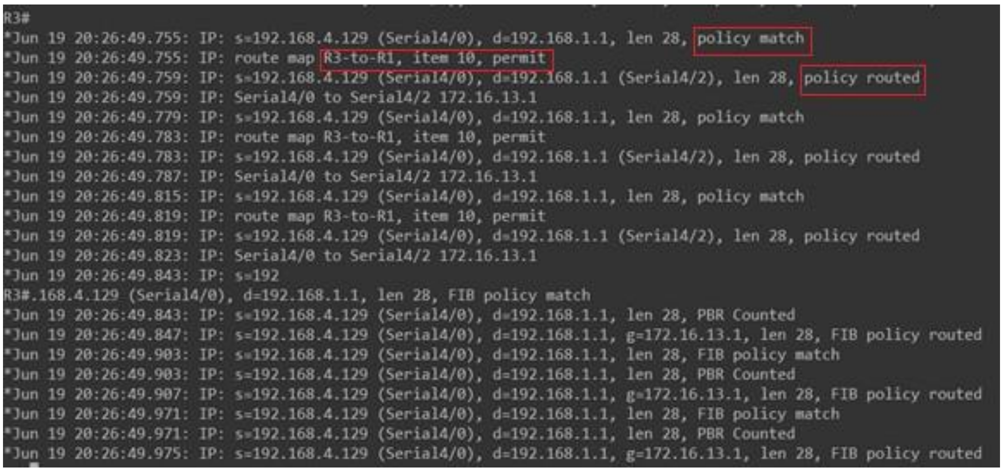
d. Test the policy from R4 with the **traceroute** command, using R4 LAN B as the source network.

Router R4

`traceroute 192.168.1.1 source 192.168.4.129`



```
R4#traceroute 192.168.1.1 source 192.168.4.129
Type escape sequence to abort.
Tracing the route to 192.168.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.34.3 20 msec 32 msec 36 msec
  2 172.16.13.1 56 msec 56 msec 56 msec
```

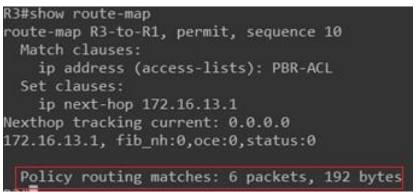


```
R3#
*Jun 19 20:26:49.755: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, policy match
*Jun 19 20:26:49.755: IP: route map R3-to-R1, item 10, permit
*Jun 19 20:26:49.759: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1 (Serial4/2), len 28, policy routed
*Jun 19 20:26:49.759: IP: Serial4/0 to Serial4/2 172.16.13.1
*Jun 19 20:26:49.779: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, policy match
*Jun 19 20:26:49.783: IP: route map R3-to-R1, item 10, permit
*Jun 19 20:26:49.783: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1 (Serial4/2), len 28, policy routed
*Jun 19 20:26:49.787: IP: Serial4/0 to Serial4/2 172.16.13.1
*Jun 19 20:26:49.815: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, policy match
*Jun 19 20:26:49.819: IP: route map R3-to-R1, item 10, permit
*Jun 19 20:26:49.819: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1 (Serial4/2), len 28, policy routed
*Jun 19 20:26:49.823: IP: Serial4/0 to Serial4/2 172.16.13.1
*Jun 19 20:26:49.843: IP: s=192
R3#.168.4.129 (Serial4/0), d=192.168.1.1, len 28, FIB policy match
*Jun 19 20:26:49.843: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, PBR Counted
*Jun 19 20:26:49.847: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, g=172.16.13.1, len 28, FIB policy routed
*Jun 19 20:26:49.903: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, FIB policy match
*Jun 19 20:26:49.903: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, PBR Counted
*Jun 19 20:26:49.907: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, g=172.16.13.1, len 28, FIB policy routed
*Jun 19 20:26:49.971: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, FIB policy match
*Jun 19 20:26:49.971: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, len 28, PBR Counted
*Jun 19 20:26:49.975: IP: s=192.168.4.129 (Serial4/0), d=192.168.1.1, g=172.16.13.1, len 28, FIB policy routed
```

e. On R3, display the policy and matches using the **show route-map** command.

Router R3

`show route-map`

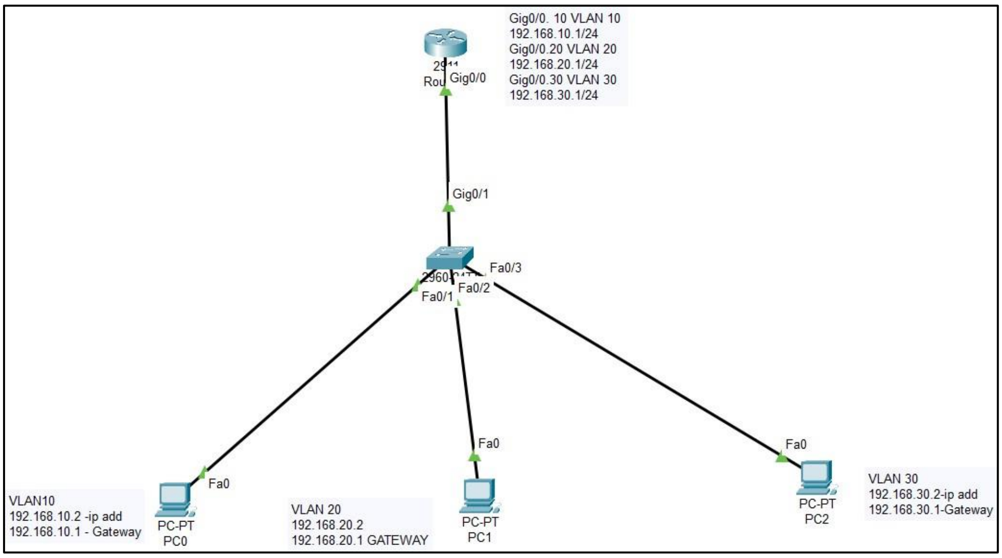


```
R3#show route-map
route-map R3-to-R1, permit, sequence 10
  Match clauses:
    ip address (access-lists): PBR-ACL
  Set clauses:
    ip next-hop 172.16.13.1
Nexthop tracking current: 0.0.0.0
172.16.13.1, fib_nh:0,oce:0,status:0

  Policy routing matches: 6 packets, 192 bytes
```

Practical -6

Title: - Inter V-LAN Routing

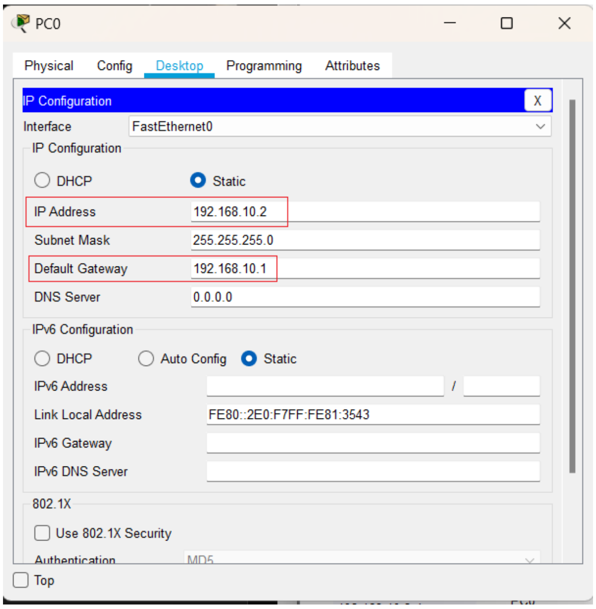
Topology: -



Topology for V-LAN

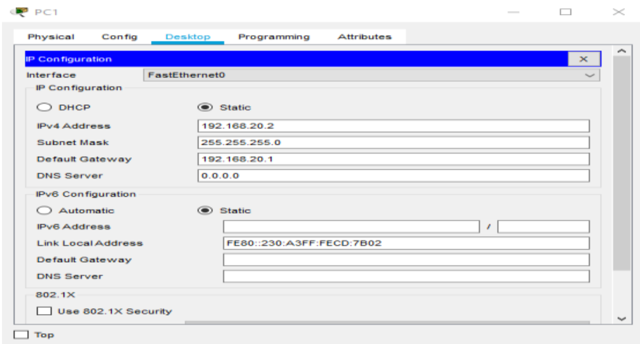
Step 1:- Assign IP address to PC0, PC1 and PC2.

a. To assign the IP address to PC click on PC, go to Desktop tab then click on IP Configuration and enter IP address and Default Gateway.



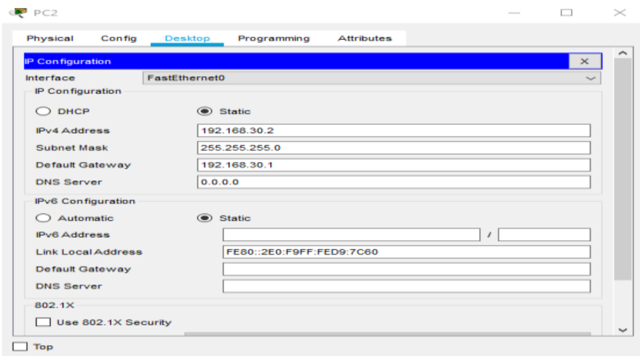
IP address for PC0- `192.168.10.2`

Default Gateway for PC0- `192.168.10.1`



IP address for PC1- `192.168.20.2`

Default Gateway for PC1- `192.168.20.1`



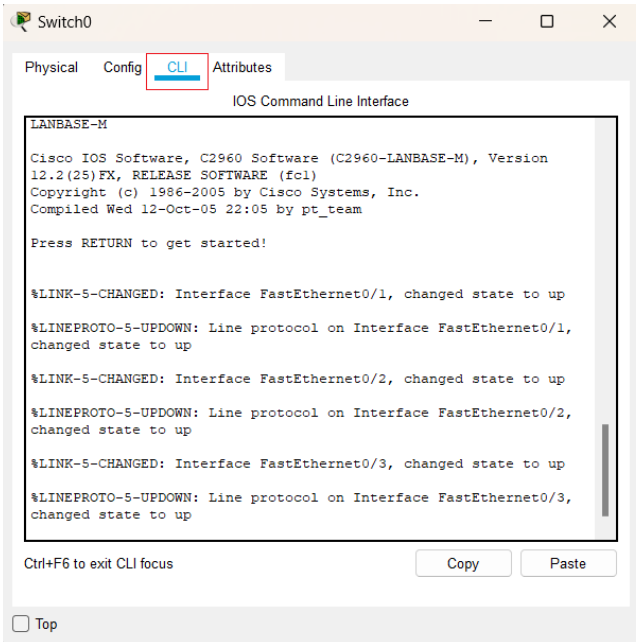
IP address for PC2- `192.168.30.2`

Default Gateway for PC2- `192.168.30.1`

b. Assign VLAN-10 to PC0, VLAN-20 to PC1 and VLAN-30 to PC2.

Step 2:- Configure within the Switch.

a. Create VLAN 10, 20 and 30 using command below in switch CLI.




```
Switch>enable
Switch#conf t
Switch(config)#vlan 10
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#exit
Switch(config)#vlan 30
Switch(config-vlan)#exit
```

To check whether VLAN's are created or not type following command.

```
Switch#show vlan brief
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2,
Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6,
Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10,
Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14,
Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18,
Fa0/19, Fa0/20
                                                Fa0/21, Fa0/22,
Fa0/23, Fa0/24
                                                Gig0/1, Gig0/2
10   VLAN0010                         active
20   VLAN0020                         active
30   VLAN0030                         active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch#
```

b. Assign the ports for VLAN.

```
Switch#conf t
Switch(config)#int f0/1
Switch(config-if)#switchport access vlan 10
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#int f0/2
Switch(config-if)#switchport access vlan 20
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#int f0/3
Switch(config-if)#switchport access vlan 30
Switch(config-if)#switchport mode access
Switch(config-if)#exit
```

Check whether the ports are assigned. Use following command

```
Switch#show vlan brief
```

```
Switch#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- ---------
-------------------------------
1    default                          active    Fa0/4, Fa0/5,
Fa0/6, Fa0/7
                                                Fa0/8, Fa0/9,
Fa0/10, Fa0/11
                                                Fa0/12, Fa0/13,
Fa0/14, Fa0/15
                                                Fa0/16, Fa0/17,
Fa0/18, Fa0/19
                                                Fa0/20, Fa0/21,
Fa0/22, Fa0/23
                                                Fa0/24, Gig0/1,
Gig0/2
10   VLAN0010                         active    Fa0/1
20   VLAN0020                         active    Fa0/2
30   VLAN0030                         active    Fa0/3
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
```

c. Configure G0/1 port as trunk port.

```
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int g0/1
Switch(config-if)#no shut
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
```

Check whether port is set as trunk port using command,

```
Switch#show run
```

```
Switch#show run
Building configuration...

Current configuration : 1256 bytes
!
version 12.2
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Switch
!
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
interface FastEthernet0/1
 switchport access vlan 10
 switchport mode access

interface FastEthernet0/2
 switchport access vlan 20
 switchport mode access
!
interface FastEthernet0/3
 switchport access vlan 30
 switchport mode access

interface GigabitEthernet0/1
 switchport mode trunk
!
interface GigabitEthernet0/2
!
interface Vlan1
 no ip address
 shutdown
!
```

Step 3:- Configure router with dot1Q encapsulation by making each of sub-interfaces.

```
Router>enable
Router#conf t
Router(config)#int G0/0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int g0/0.10
Router(config-subif)#encapsulation dot1Q 10
Router(config-subif)#ip add 192.168.10.1 255.255.255.0
Router(config-subif)#exit

Router(config)#int G0/0.20
Router(config-subif)#encapsulation dot1Q 20
Router(config-subif)#ip add 192.168.20.1 255.255.255.0
Router(config-subif)#exit


Router(config)#int g0/0.30
Router(config-subif)#encapsulation dot1Q 30
Router(config-subif)#ip add 192.168.30.1 255.255.255.0
Router(config-subif)#exit
```

Check whether all the ports for VLAN are configured using following command.

```
Router#show run
interface GigabitEthernet0/0
 no ip address
 duplex auto
 speed auto
!
interface GigabitEthernet0/0.10
 encapsulation dot1Q 10
 ip address 192.168.10.1 255.255.255.0
!
interface GigabitEthernet0/0.20
 encapsulation dot1Q 20
 ip address 192.168.20.1 255.255.255.0
!
interface GigabitEthernet0/0.30
 encapsulation dot1Q 30
 ip address 192.168.30.1 255.255.255.0
!
interface GigabitEthernet0/1
 no ip address
 duplex auto
 speed auto
 shutdown
```

```
Router#show ip route
```

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

Gateway of last resort is not set

     192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0.10
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0.10
     192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.20.0/24 is directly connected, GigabitEthernet0/0.20
L       192.168.20.1/32 is directly connected, GigabitEthernet0/0.20
     192.168.30.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.30.0/24 is directly connected, GigabitEthernet0/0.30
L       192.168.30.1/32 is directly connected, GigabitEthernet0/0.30
```

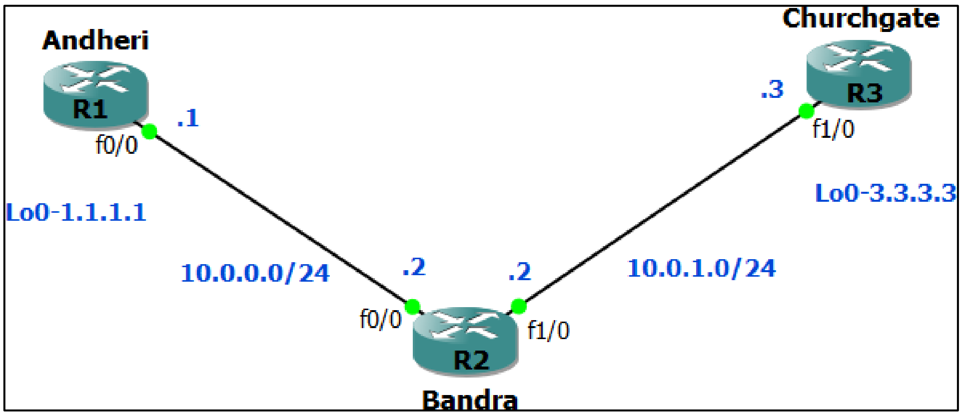
Step 4:- Try to ping from one PC to another to check whether VLAN configuration is successful.

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Failed	PC0	PC2	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC0	PC2	ICMP		0.000	N	1	(edit)	(delete)

Practical -7

Title: - Simulating MPLS Environment

Topology: -



Topology for MPLS 7.1

Step 1 – IP addressing of MPLS Core and OSPF.

a. First bring 3 routers into your **Topology** R1, R2, R3 position them as below. We are going to address the routers and configure ospf to ensure loopback to loopback connectivity between R1and R3

Andheri

```
conf t
hostname Andheri
int lo0
ip add 1.1.1.1 255.255.255.255
ip ospf 1 area 0
int f0/0
ip add 10.0.0.1 255.255.255.0
no shut
ip ospf 1 area 0
```

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#hostname Andheri
Andheri(config)#int lo0
Andheri(config-if)#ip add 1.1.1.1 255.255.255.255
Andheri(config-if)#ip ospf 1 area 0
Andheri(config-if)#int f0/0
Andheri(config-if)#ip add 10.0.0.1 255.255.255.0
Andheri(config-if)#no shut
Andheri(config-if)#ip ospf 1 area 0
Andheri(config-if)#
*Jun 21 10:27:23.735: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Andheri(config-if)#
*Jun 21 10:27:26.739: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Jun 21 10:27:27.739: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Andheri(config-if)#
*Jun 21 10:28:39.247: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on FastEthernet0/0 from LOADING to FULL, Loadin
g Done
```

Bandra

```
conf  t
hostname Bandra
int lo0
ip add 2.2.2.2 255.255.255.255
ip ospf 1 are 0
int f0/0
ip add 10.0.0.2 255.255.255.0
no shut
ip ospf 1 area 0
int f1/0
ip add 10.0.1.2 255.255.255.0
no shut
ip ospf 1 area 0
```

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#hostname Bandra
Bandra(config)#int lo0
Bandra(config-if)#ip add 2.2.2.2 255.255.255.255
Bandra(config-if)#ip ospf 1 are 0
Bandra(config-if)#int f0/0
Bandra(config-if)#ip add 10.0.0.2 255.255.255.0
Bandra(config-if)#
*Jun 21 10:27:38.575: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Bandra(config-if)#no shut
Bandra(config-if)#ip ospf 1 area 0
Bandra(config-if)#int f1/0
Bandra(config-if)#ip add 10.0.1.2 255.255.255.0
Bandra(config-if)#no shut
Bandra(config-if)#ip ospf 1 area 0
```

Churchgate

```
conf t
hostname Churchgate
int lo0
ip add 3.3.3.3 255.255.255.255
ip ospf 1 are 0

int f1/0
ip add 10.0.1.3 255.255.255.0
no shut
ip ospf 1 area 0
```

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#hostname Churchgate
Churchgate(config)#int lo0
Churchgate(config-if)#ip add 3.3.3.3 255.255.255.255
Churchgate(config-if)#ip ospf 1 are 0
Churchgate(config-if)#
Churchgate(config-if)#int f1/0
Churchgate(config-if)#ip add 10.0.1.3 255.255.255.0
Churchgate(config-if)#no shut
Churchgate(config-if)#ip ospf 1 area 0
```

You should now have full ip connectivity between R1, R2, R3 to verify this we need to see if we can ping between the loopbacks of R1 and R3

Andheri

```
ping 3.3.3.3 source lo0
```

```
Andheri#ping 3.3.3.3 source lo0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/64/68 ms
Andheri#
```

Step 2 – Configure LDP on all the interfaces in the MPLS Core

In order to run MPLS you need to enable it, there are two ways to do this.

Andheri

```
conf t
router ospf 1
mpls ldp autoconfig
```

```
Andheri(config)#router ospf 1
Andheri(config-router)#mpls ldp autoconfig
```

Bandra

```
conf t
router ospf 1
mpls ldp autoconfig
```

```
Bandra(config)#router ospf 1
Bandra(config-router)#mpls ldp autoconfig
```

Churchgate

```
conf t
router ospf 1
mpls ldp autoconfig
```

```
Churchgate(config)#router ospf 1
Churchgate(config-router)#mpls ldp autoconfig
```

You should see log messages coming up showing the LDP neighbors are up.

```
*Jun 21 10:32:59.019: %LDP-5-NBRCHG: LDP Neighbor 1.1.1.1:0 (1) is UP
Bandra(config-router)#
*Jun 21 10:33:10.275: %LDP-5-NBRCHG: LDP Neighbor 3.3.3.3:0 (2) is UP
```

To verify the mpls interfaces the command is very simple – sh mpls interface

This is done on R2 and you can see that both interfaces are running mpls and using LDP

Bandra

```
sh mpls interface
```

```
Bandra#sh mpls interface
Interface              IP            Tunnel   BGP Static Operational
FastEthernet0/0        Yes (ldp)     No       No  No     Yes
FastEthernet1/0        Yes (ldp)     No       No  No     Yes
Bandra#
```

You can also verify the LDP neighbors with the sh mpls ldp neighbors command.

Bandra

```
sh mpls ldp neigh
```

```
Bandra#sh mpls ldp neigh
    Peer LDP Ident: 1.1.1.1:0; Local LDP Ident 2.2.2.2:0
        TCP connection: 1.1.1.1.646 - 2.2.2.2.20412
        State: Oper; Msgs sent/rcvd: 11/12; Downstream
        Up time: 00:03:26
        LDP discovery sources:
          FastEthernet0/0, Src IP addr: 10.0.0.1
        Addresses bound to peer LDP Ident:
          10.0.0.1        1.1.1.1
    Peer LDP Ident: 3.3.3.3:0; Local LDP Ident 2.2.2.2:0
        TCP connection: 3.3.3.3.13190 - 2.2.2.2.646
        State: Oper; Msgs sent/rcvd: 11/11; Downstream
        Up time: 00:03:15
        LDP discovery sources:
          FastEthernet1/0, Src IP addr: 10.0.1.3
        Addresses bound to peer LDP Ident:
          10.0.1.3        3.3.3.3
```

One more verification to confirm LDP is running ok is to do a trace between R1 and R3 and verify if you get MPLS Labels show up in the trace.

Andheri

```
trace 3.3.3.3
```

```
Andheri#trace 3.3.3.3
Type escape sequence to abort.
Tracing the route to 3.3.3.3
VRF info: (vrf in name/id, vrf out name/id)
  1 10.0.0.2 [MPLS: Label 17 Exp 0] 44 msec 16 msec 28 msec
  2 10.0.1.3 72 msec 60 msec 64 msec
```

Step 3 – MPLS BGP Configuration between R1 and R3

We need to establish a Multi Protocol BGP session between R1 and R3 this is done by configuring the vpnv4 address family as below

Andheri

```
conf t
router bgp 1
neighbor 3.3.3.3 remote-as 1
neighbor 3.3.3.3 update-source Loopback0
no auto-summary
address-family vpnv4
neighbor 3.3.3.3 activate
```

```
Andheri(config)#router bgp 1
Andheri(config-router)#neighbor 3.3.3.3 remote-as 1
Andheri(config-router)#neighbor 3.3.3.3 update-source Loopback0
Andheri(config-router)#no auto-summary
Andheri(config-router)#address-family vpnv4
Andheri(config-router-af)#neighbor 3.3.3.3 activate
```

Churchgate

```
conf t
router bgp 1
neighbor 1.1.1.1 remote-as 1
neighbor 1.1.1.1 update-source Loopback0
no auto-summary
address-family vpnv4
neighbor 1.1.1.1 activate
```

```
Churchgate(config)#router bgp 1
Churchgate(config-router)#neighbor 1.1.1.1 remote-as 1
Churchgate(config-router)#neighbor 1.1.1.1 update-source Loopback0
Churchgate(config-router)#no auto-summary
Churchgate(config-router)#address-family vpnv4
Churchgate(config-router-af)#neighbor 1.1.1.1 activate
```

Andheri

```
sh bgp vpnv4 unicast all summary
```

```
Andheri#sh bgp vpnv4 unicast all summary
BGP router identifier 1.1.1.1, local AS number 1
BGP table version is 1, main routing table version 1

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
3.3.3.3         4            1       5       5        1    0    0 00:01:24        0
```

Step 4 – Add two more routers, create VRFs We will add two more routers into the **Topology** so it now looks like the final **Topology**

Borivali(R4)

```
conf t
int lo0
ip add 4.4.4.4 255.255.255.255
ip ospf 2 area 2

int f0/0
ip add 192.168.1.4 255.255.255.0
ip ospf 2 area 2
no shut
```

```
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#int lo0
R4(config-if)#ip add 4.4.4.4 255.255.255.255
R4(config-if)#ip ospf 2 area 2
R4(config-if)#int f0/0
R4(config-if)#ip add 192.168.1.4 255.255.255.0
R4(config-if)#ip ospf 2 area 2
R4(config-if)#no shut
```

Andheri

```
conf t
int f1/0
no shut
ip add 192.168.1.1 255.255.255.0
```

```
Andheri(config)#int f1/0
Andheri(config-if)#no shut
Andheri(config-if)#ip add 192.168.1.1 255.255.255.0
```

Andheri

```
conf t
ip vrf RED
rd 4:4
route-target both 4:4
```

Andheri

```
conf t
int f1/0
ip vrf forwarding RED
```

```
Andheri(config)#int f1/0
Andheri(config-if)#ip vrf forwarding RED
% Interface FastEthernet1/0 IPv4 disabled and address(es) removed due to enabling VRF RED
```

Andheri

```
conf  t
ip vrf fo
ip vrf forwarding RED
```

Andheri

```
conf t
int f1/0
ip address 192.168.1.1 255.255.255.0
```

```
Andheri(config)#int f1/0
Andheri(config-if)#ip address 192.168.1.1 255.255.255.0
```

Andheri

```
sh run int f1/0
```

```
Andheri#sh run int f1/0
Building configuration...

Current configuration : 119 bytes
!
interface FastEthernet1/0
 ip vrf forwarding RED
 ip address 192.168.1.1 255.255.255.0
 speed auto
 duplex auto
end
```

Andheri

sh ip route

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

Gateway of last resort is not set

      1.0.0.0/32 is subnetted, 1 subnets
C        1.1.1.1 is directly connected, Loopback0
      2.0.0.0/32 is subnetted, 1 subnets
O        2.2.2.2 [110/2] via 10.0.0.2, 00:38:17, FastEthernet0/0
      3.0.0.0/32 is subnetted, 1 subnets
O        3.3.3.3 [110/3] via 10.0.0.2, 00:37:47, FastEthernet0/0
      10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C        10.0.0.0/24 is directly connected, FastEthernet0/0
L        10.0.0.1/32 is directly connected, FastEthernet0/0
O        10.0.1.0/24 [110/2] via 10.0.0.2, 00:37:57, FastEthernet0/0
```

sh ip route vrf red

```
Andheri#sh ip route vrf red
% IP routing table vrf red does not exist
```

sh ip route vrf RED

```
Andheri#sh ip route vrf RED

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

Gateway of last resort is not set

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet1/0
L        192.168.1.1/32 is directly connected, FastEthernet1/0
```

```
conf t
int f1/0
ip ospf 2 area 2
```

```
Andheri(config)#int f1/0
Andheri(config-if)#ip ospf 2 area 2
```

```
sh ip route vrf RED
```

```
Andheri#sh ip route vrf RED

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

Gateway of last resort is not set

      4.0.0.0/32 is subnetted, 1 subnets
O        4.4.4.4 [110/2] via 192.168.1.4, 00:00:40, FastEthernet1/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet1/0
L        192.168.1.1/32 is directly connected, FastEthernet1/0
```

We now need to repeat this process for R3 & R6 Router 6 will peer OSPF using process number2to a VRF configured on R3. It will use the local site addressing to 192.168.2.0/24

Mahim(R6)

```
conf t
int lo0
ip add 6.6.6.6 255.255.255.255
ip ospf 2 area 2
int f0/0
ip add 192.168.2.6 255.255.255.0
ip ospf 2 area 2
no shut
```

Churchgate

```
conf t
int f0/0
no shutdown
ip address 192.168.2.3 255.255.255.0
```

```
Churchgate(config)#int f0/0
Churchgate(config-if)#no shutdown
Churchgate(config-if)#ip address 192.168.2.3 255.255.255.0
Churchgate(config-if)#
```

We also need to configure a VRF onto R3 as well.

Churchgate

```
conf t
ip vrf RED
rd 4:4
route-target both 4:4
```

```
Churchgate(config)#ip vrf RED
Churchgate(config-vrf)#rd 4:4
Churchgate(config-vrf)#route-target both 4:4
Churchgate(config-vrf)#
```

```
conf t
int f0/0
ip vrf forwarding RED
```

```
Churchgate(config)#int f0/0
Churchgate(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/0 IPv4 disabled and address(es) removed due to enabling VRF RED
Churchgate(config-if)#
```

```
conf t
ip vrf forwarding RED
```

```
conf t
int f1/0
ip address 192.168.2.1 255.255.255.0
sh run int f1/0
```

```
Churchgate#sh run int f1/0
Building configuration...

Current configuration : 114 bytes
!
interface FastEthernet1/0
 ip address 192.168.2.1 255.255.255.0
 ip ospf 1 area 0
 speed auto
 duplex auto
end
```

```
conf t
int f1/0
ip ospf 2 area 2
```

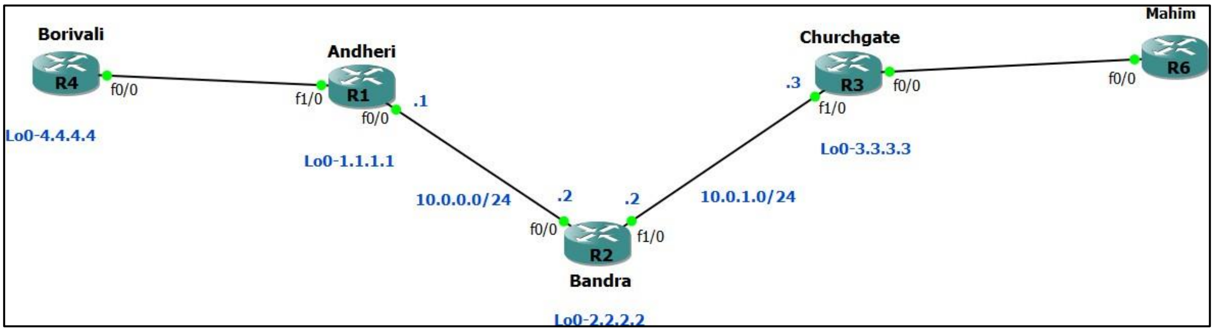
```
Churchgate#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Churchgate(config)#int f1/0
Churchgate(config-if)#ip ospf 2 area 2
Churchgate(config-if)#
```

Check the router in vrf RED

```
sh ip route vrf RED
```

```
Churchgate#sh ip route vrf RED

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

Gateway of last resort is not set

Churchgate#
*Jun 21 12:11:26.731: %OSPF-4-ERRRCV: Received invalid packet: mismatched area I
D, from backbone area must be virtual-link but not found from 10.0.1.2, FastEthe
rnet1/0
```

Ok so we have come a long way now let’s review the current situation. We now have this setup



Topology for MPLS 7.2

Borivali

```
sh ip route
```

```
R4#sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      4.0.0.0/32 is subnetted, 1 subnets
C        4.4.4.4 is directly connected, Loopback0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet0/0
L        192.168.1.4/32 is directly connected, FastEthernet0/0
```

As expected we have the local interface and the loopback address. When we are done we want to see 6.6.6.6 in there so we can ping across the MPLS Check the routes on R1

Andheri

```
sh ip route
```

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

Gateway of last resort is not set

      1.0.0.0/32 is subnetted, 1 subnets
C        1.1.1.1 is directly connected, Loopback0
      2.0.0.0/32 is subnetted, 1 subnets
O        2.2.2.2 [110/2] via 10.0.0.2, 00:15:54, FastEthernet0/0
      10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C        10.0.0.0/24 is directly connected, FastEthernet0/0
L        10.0.0.1/32 is directly connected, FastEthernet0/0
O        10.0.1.0/24 [110/2] via 10.0.0.2, 00:15:54, FastEthernet0/0
```

```
sh ip route vrf RED
```

```
Andheri#sh ip route vrf RED

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

Gateway of last resort is not set

      4.0.0.0/32 is subnetted, 1 subnets
O        4.4.4.4 [110/2] via 192.168.1.4, 00:09:57, FastEthernet1/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet1/0
L        192.168.1.1/32 is directly connected, FastEthernet1/0
```

```
conf t
router bgp 1
address-family ipv4 vrf RED
redistribute ospf 2
```

```
Andheri(config)#router bgp 1
Andheri(config-router)#address-family ipv4 vrf RED
Andheri(config-router-af)#redistribute ospf 2
Andheri(config-router-af)#
```

Churchgate

```
conf t
router bgp 1
address-family ipv4 vrf RED
redistribute ospf 2
```

```
Churchgate#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Churchgate(config)#router bgp 1
Churchgate(config-router)#address-family ipv4 vrf RED
Churchgate(config-router-af)#redistribute ospf 2
%VRF specified does not match this router
Churchgate(config-router-af)#
```

Andheri

```
sh ip bgp vpnv4 vrf RED
```

```
Andheri#sh ip bgp vpnv4 vrf RED
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 4:4 (default for vrf RED)
 *>  4.4.4.4/32       192.168.1.4              2         32768 ?
 *>  192.168.1.0      0.0.0.0                  0         32768 ?
```

Churchgate

```
sh ip bgp vpnv4 vrf RED
```

```
Churchgate#sh ip bgp vpnv4 vrf RED
Churchgate#
*Jun 21 12:18:04.851: %OSPF-4-ERRRCV: Received invalid packet: mismatched area I
D, from backbone area must be virtual-link but not found from 10.0.1.2, FastEthe
rnet1/0
Churchgate#
*Jun 21 12:18:14.331: %OSPF-4-ERRRCV: Received invalid packet: mismatched area I
D, from backbone area must be virtual-link but not found from 10.0.1.2, FastEthe
rnet1/0
```

Which it is! 6.6.6.6 is now in the BGP table in VRF RED on R3 with a next hop of 192.168.2.6 (R6)and also 4.4.4 is in there as well with a next hop of 1.1.1.1 (which is the loopback of R1 – showing that it is going over the MPLS and R2 is not in the picture)

Andheri

```
conf  t
router ospf 2
redistribute bgp 1 subnets
```

```
Andheri(config)#router ospf 2
Andheri(config-router)#redistribute bgp 1 subnets
```

Churchgate

```
conf  t
router ospf 2
redistribute bgp 1 subnets
```

```
Churchgate(config)#router ospf 2
Churchgate(config-router)#redistribute bgp 1 subnets
```

Before we do let’s see what the routing table look like on R4

Borivali

```
sh ip route
```

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

Gateway of last resort is not set

      4.0.0.0/32 is subnetted, 1 subnets
C        4.4.4.4 is directly connected, Loopback0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, FastEthernet0/0
L        192.168.1.4/32 is directly connected, FastEthernet0/0
```

Do the same step of on R6

Mahim

```
sh ip route
```

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

Gateway of last resort is not set

      6.0.0.0/32 is subnetted, 1 subnets
C        6.6.6.6 is directly connected, Loopback0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.2.0/24 is directly connected, FastEthernet0/0
L        192.168.2.6/32 is directly connected, FastEthernet0/0
```

Lets check ping command

Borivali

```
ping 6.6.6.6
```

```
Borivali#ping 6.6.6.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 6.6.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 112/120/128 ms
```

Which we can – to prove this is going over the mpls and be label switched and not routed,let do a trace

```
trace 6.6.6.6
```

```
Borivali#trace 6.6.6.6

Type escape sequence to abort.
Tracing the route to 6.6.6.6

  1 192.168.1.1 20 msec 32 msec 24 msec
  2 10.0.0.2 [MPLS: Labels 17/19 Exp 0] 112 msec 136 msec 124 msec
  3 192.168.2.1 [MPLS: Label 19 Exp 0] 72 msec 92 msec 92 msec
  4 192.168.2.6 140 msec 124 msec 124 msec
```