ST. GONSALO GARCIA COLLEGE OF COMMERCE & ARTS

VASAI ROAD (W)

DIST-PALGHAR 401 202

Practical

On

Modern Networking

Submitted by
Shirke Omkar Suresh

Roll No. :07

Semester II

In partial fulfillment of the requirement for the Award Degree ofMaster of Science

(Information Technology)

University of Mumbai2022-

2023

ST. GONSALO GARCIA COLLEGE OF COMMERCE & ARTS

CERTIFICATE

This is to certify that **Mr. Omkar Shirke** of M. Sc.(IT) has successfully completed the Practical on **Modern Networking** for the year 2022 - 2023 under the guidance of **Prof. Brenza Cerejo.**

Subject Incharge **Head of Department**

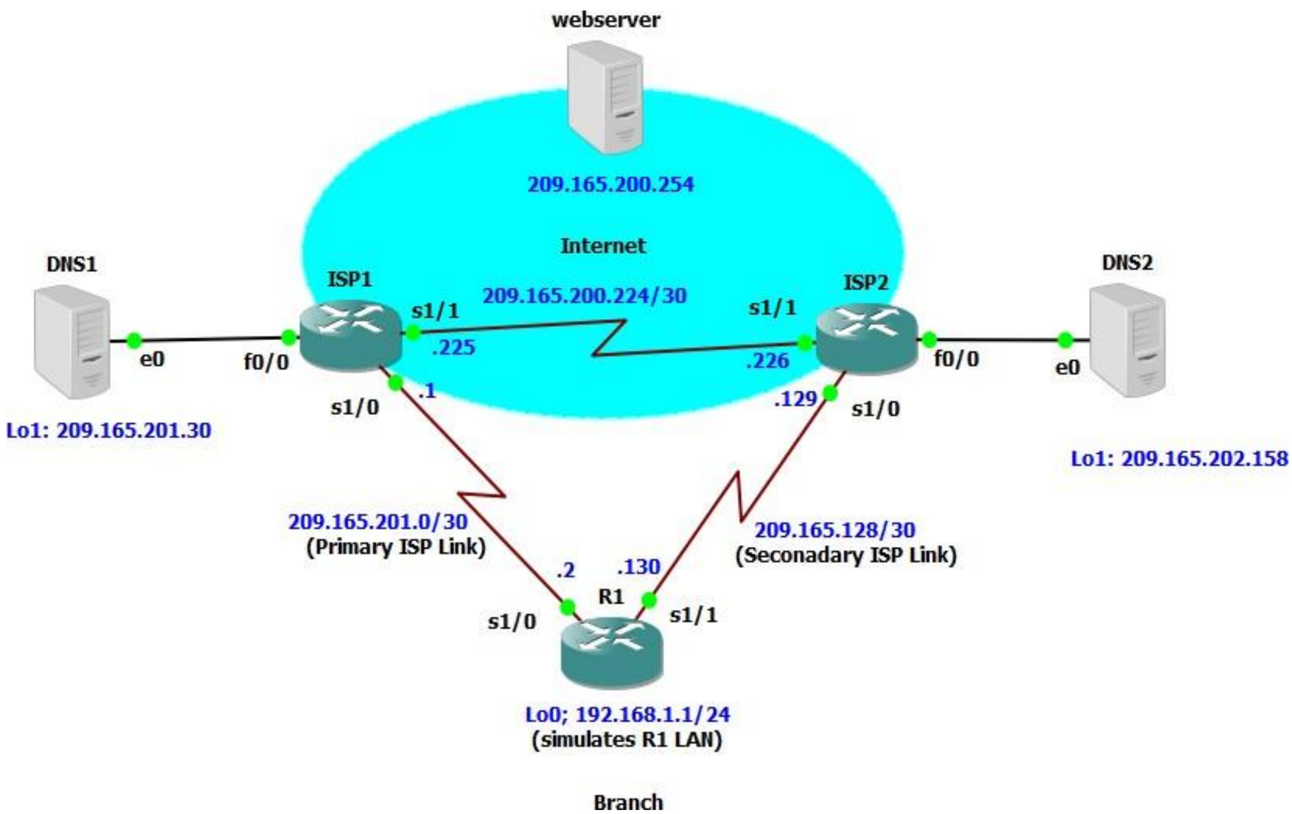
External Examiner **Principal**

Index

Practical No - 1

Aim: Configure IP SLA Tracking and Path Control Topology

Topolgy :



Objectives

- Configure and verify the IP SLA feature.
- Test the IP SLA tracking feature.
- Verify the configuration and operation using show and debug commands.

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

Router R1

interface Loopback 0

ip address 192.168.1.1 255.255.255.0

interface Serial0/0/0

ip address 209.165.201.2 255.255.255.252

no shutdown

interface Serial0/0/1

ip address 209.165.202.130 255.255.255.252

no shutdown

```
R1(config)#interface Loopback 0
```

```
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#
R1(config-if)#int s1/0
R1(config-if)#ip address 209.165.201.2 255.255.255.252
R1(config-if)#no shutdown
```

```
R1(config-if)#int s1/1
```

```
R1(config-if)#ip address 209.165.202.130 255.255.255.252
R1(config-if)#no shutdown
```

Router ISP1 (R2)

interface Loopback0

ip address 209.165.200.254 255.255.255.255

interface Loopback1

ip address 209.165.201.30 255.255.255.255

interface Serial0/0/0

ip address 209.165.201.1 255.255.255.252

no shutdown

interface Serial0/0/1

ip address 209.165.200.225 255.255.255.252

no shutdown

```
ISP1(config)#interface Loopback0
ISP1(config-if)#
*May 18 15:24:24.315: %LINEPROTO-5-UPDOWN: Line protocol or
ISP1(config-if)#ip address 209.165.200.254 255.255.255.255
ISP1(config-if)#interface Loopback1
ISP1(config-if)#
*May 18 15:24:36.915: %LINEPROTO-5-UPDOWN: Line protocol or
ISP1(config-if)#ip address 209.165.201.30 255.255.255.255
ISP1(config-if)#int s1/0
ISP1(config-if)#ip address 209.165.201.1 255.255.255.252
ISP1(config-if)#no shutdown
ISP1(config-if)#
*May 18 15:25:03.695: %LINK-3-UPDOWN: Interface Serial1/0,
ISP1(config-if)#
ISP1(config-if)#i
*May 18 15:25:04.699: %LINEPROTO-5-UPDOWN: Line protocol or
ISP1(config-if)#int s1/1
ISP1(config-if)#ip address 209.165.200.225 255.255.255.252
ISP1(config-if)#no shutdown
```

Router ISP2 (R3)

interface Loopback0

ip address 209.165.200.254 255.255.255.255

interface Loopback1

ip address 209.165.202.158 255.255.255.255

interface Serial0/0/0

description ISP2 --> R1

ip address 209.165.202.129 255.255.255.252

no shutdown

interface Serial0/0/1

ip address 209.165.200.226 255.255.255.252

no shutdown

```
ISP2(config)#interface Loopback0
ISP2(config-if)#
*May 18 15:25:22.219: %LINEPROTO-5-UPDOWN: Line protocol on
ISP2(config-if)#ip address 209.165.200.254 255.255.255.255
ISP2(config-if)#interface Loopback1
ISP2(config-if)#
*May 18 15:25:34.595: %LINEPROTO-5-UPDOWN: Line protocol on
ISP2(config-if)#ip address 209.165.202.158 255.255.255.255
ISP2(config-if)#int s1/0
ISP2(config-if)#ip address 209.165.202.129 255.255.255.252
ISP2(config-if)#no shutdown
ISP2(config-if)#
ISP2(config-if)#
*May 18 15:26:01.299: %LINK-3-UPDOWN: Interface Serial1/0,
ISP2(config-if)#i
*May 18 15:26:02.303: %LINEPROTO-5-UPDOWN: Line protocol on
ISP2(config-if)#int s1/1
ISP2(config-if)#ip address 209.165.200.226 255.255.255.252
ISP2(config-if)#no shutdown
```

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

R1# show interfaces description

```
R1#show interfaces description
Interface                      Status         Protocol Description
Fa0/0                          admin down     down
Se1/0                          up             up
Se1/1                          up             up
Se1/2                          admin down     down
Se1/3                          admin down     down
Lo0                            up             up
```

c. The current routing policy in the topology is as follows:

- Router R1 establishes connectivity to the Internet through ISP1 using a default static route.
- ISP1 and ISP2 have dynamic routing enabled between them, advertising their respective public address pools.
- ISP1 and ISP2 both have static routes back to the ISP LAN.

Router R1

ip route 0.0.0.0 0.0.0.0 209.165.201.1

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

Router ISP1 (R2)

router eigrp 1

network 209.165.200.224 0.0.0.3

network 209.165.201.0 0.0.0.31

no auto-summary

ip route 192.168.1.0 255.255.255.0 209.165.201.2

```
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)#ip route 192.168.1.0 255.255.255.0 209.165.201.2
```

Router ISP2 (R3)

router eigrp 1

network 209.165.200.224 0.0.0.3

network 209.165.202.128 0.0.0.31

no auto-summary

ip route 192.168.1.0 255.255.255.0 209.165.202.130

```
ISP2(config)#router eigrp 1
ISP2(config-router)#network 209.165.200.224 0.0.0.3
ISP2(config-router)#
*May 18 15:30:14.515: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 209.1
ISP2(config-router)#network 209.165.202.128 0.0.0.31
ISP2(config-router)#no auto-summary
ISP2(config-router)#
*May 18 15:30:28.971: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 209.1
ISP2(config-router)#ip route 192.168.1.0 255.255.255.0 209.165.202.130
```

Step 2: Verify server reachability.

a. Before implementing the Cisco IOS SLA feature, you must verify reachability to the Internet servers. From router R1, ping the web server, ISP1 DNS server, and ISP2 DNS server to verify connectivity. You can copy the following Tcl script and paste it intoR1.

R1(tcl)# foreach address {

+>(tcl)# 209.165.200.254

+>(tcl)# 209.165.201.30

+>(tcl)# 209.165.202.158

+>(tcl)# } { ping $address source 192.168.1.1 }

```
R1#tclsh
R1(tcl)#foreach address {
+>(tcl)#209.165.200.254
+>(tcl)#209.165.201.30
+>(tcl)#209.165.202.158
+>(tcl)#} {
+>(tcl)#ping $address source 192.168.1.1
+>(tcl)#}
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.254, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/78/96 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.201.30, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/31/48 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.202.158, timeout is 2 seconds:
Packet sent with a source address of 192.168.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/37/60 ms
```

b. 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(tcl)# foreach address {

+>(tcl)# 209.165.200.254

+>(tcl)# 209.165.201.30

+>(tcl)# 209.165.202.158

+>(tcl)# } { trace $address source 192.168.1.1 }

```
R1(tcl)#foreach address {
+>(tcl)#209.165.200.254
+>(tcl)#209.165.201.30
+>(tcl)#209.165.202.158
+>(tcl)#} {
+>(tcl)#trace $address source 192.168.1.1
+>(tcl)#}
```

```
Type escape sequence to abort.
Tracing the route to 209.165.200.254

  1 209.165.201.1 24 msec 24 msec 16 msec
Type escape sequence to abort.
Tracing the route to 209.165.201.30

  1 209.165.201.1 16 msec 24 msec 24 msec
Type escape sequence to abort.
Tracing the route to 209.165.202.158

  1 209.165.201.1 24 msec 24 msec 32 msec
  2 209.165.200.226 28 msec 44 msec 72 msec
```

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. the previous ip sla monitor command. In addition, the icmp-echo commandhas replaced the type echo protocol ipIcmpEcho command.

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

```
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

R1(config)#exit
```

b. Verify the IP SLAs configuration of operation 11 using the show ip sla configuration11 command.

R1# show ip sla configuration 11

```
R1(tcl)#show ip sla configuration 11
IP SLAs, Infrastructure Engine-II.
Entry number: 11
Owner:
Tag:
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
Operation timeout (milliseconds): 5000
Verify data: No
Vrf Name:
Schedule:
   Operation frequency (seconds): 10  (not considered if
   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
```

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

R1# show ip sla statistics

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

IPSLA operation id: 22
Type of operation: icmp-echo
        Latest RTT: 64 milliseconds
Latest operation start time: *15:40:40.823 UTC Tue May 18 2021
Latest operation return code: OK
Number of successes: 6
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. You can copy and paste the following commands on R1.

```
R1(config)#
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
```

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

R1# show ip sla configuration 22

```
R1#show ip sla configuration 22
IP SLAs, Infrastructure Engine-II.
Entry number: 22
Owner:
Tag:
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
Operation timeout (milliseconds): 5000
Verify data: No
```

R1# show ip sla statistics 22

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

IPSLA operation id: 22
Type of operation: icmp-echo
        Latest RTT: 64 milliseconds
Latest operation start time: *15:40:40.823 UTC Tue May 18 2021
Latest operation return code: OK
Number of successes: 6
Number of failures: 0
Operation time to live: Forever
```

Step 4: Configure tracking options.

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

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

```
R1(config)#
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.

R1# show ip route

```
R1#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 209.165.201.1 to network 0.0.0.0
```

c. Use the track 1 ip sla 11 reachability command to enter the config-track subconfiguration mode.

R1(config)# track 1 ip sla 11 reachability

R1(config-track)#

```
R1(config)#
R1(config)#track 1 ip sla 11 reachability
R1(config-track)#delay down 10 up 1
R1(config-track)#exit
```

d. Configure the floating static route that will be implemented when tracking object 1 is active. To view routing table changes as they happen, first enable the debug ip routing command. Next, 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.

R1# debug ip routing

```
R1#debug ip routing
IP routing debugging is on
```

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)#
*May 18 15:43:00.035: RT: closer admin distance for 0.0.0.0, flushing 1 routes
*May 18 15:43:00.035: RT: NET-RED 0.0.0.0/0
*May 18 15:43:00.035: RT: add 0.0.0.0/0 via 209.165.201.1, static metric [2/0]
*May 18 15:43:00.039: RT: NET-RED 0.0.0.0/0
*May 18 15:43:00.039: RT: default path is now 0.0.0.0 via 209.165.201.1
*May 18 15:43:00.039: RT: new default network 0.0.0.0
*May 18 15:43:00.043: RT: NET-RED 0.0.0.0/0
```

e. 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. track 2 ip sla 22 reachability delay down 10 up 1 exit

ip route 0.0.0.0 0.0.0.0 209.165.202.129 3 track 2

```
R1(config)#track 1 ip sla 22 reachability
R1(config-track)#delay down 10 up 1
R1(config-track)#exi
*May 18 15:43:56.339: RT: NET-RED 0.0.0.0/0
R1(config-track)#exit
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 0.0.0.0 209.165.202.129 3 track 2
```

f. Verify the routing table again.

R1# show ip route

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - B
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF i
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA extern
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2
       ia - IS-IS inter area, * - candidate default, U - per
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 209.165.201.1 to network 0.0.0.0

     209.165.201.0/30 is subnetted, 1 subnets
C       209.165.201.0 is directly connected, Serial1/0
     209.165.202.0/30 is subnetted, 1 subnets
C       209.165.202.128 is directly connected, Serial1/1
C    192.168.1.0/24 is directly connected, Loopback0
S*   0.0.0.0/0 [2/0] via 209.165.201.1
```

Step 5: Verify IP SLA operation.

The following summarizes the process:

- Disable the DNS loopback interface on ISP1 (R2).
- Observe the output of the debug command on R1.
- Verify the static route entries in the routing table and the IP SLA statistics of R1.
- Re-enable the loopback interface on ISP1 (R2) and again observe the operation ofthe IP SLA tracking feature.

ISP1(config)# interface loopback 1

ISP1(config-if)# shutdown

```
ISP1(config)#
ISP1(config)#interface loopback 1
ISP1(config-if)#shutdown
```

b. Verify the routing table.

R1# show ip route

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSP
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA ext
       E1 - OSPF external type 1, E2 - OSPF external type
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1,
       ia - IS-IS inter area, * - candidate default, U -
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 209.165.201.1 to network 0.0.0.

     209.165.201.0/30 is subnetted, 1 subnets
C       209.165.201.0 is directly connected, Serial1/0
     209.165.202.0/30 is subnetted, 1 subnets
C       209.165.202.128 is directly connected, Serial1/1
C    192.168.1.0/24 is directly connected, Loopback0
S*   0.0.0.0/0 [2/0] via 209.165.201.1
```

c. Verify the IP SLA statistics.

R1# show ip sla statistics

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

IPSLA operation id: 11
Type of operation: icmp-echo
        Latest RTT: NoConnection/Busy/Timeout
Latest operation start time: *15:47:41.887 UTC Tue May 18 2021
Latest operation return code: No connection
Number of successes: 51
Number of failures: 13
Operation time to live: Forever
```

d. Initiate a trace to the web server from the internal LAN IP address.

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

  1 209.165.201.1 4 msec 32 msec 32 msec
```

f. Again examine the IP SLA statistics.

R1# show ip sla statistics

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

IPSLA operation id: 11
Type of operation: icmp-echo
        Latest RTT: 57 milliseconds
Latest operation start time: *15:50:01.887 UTC Tue May 18 2021
Latest operation return code: OK
Number of successes: 61
Number of failures: 17
Operation time to live: Forever
```

g. Verify the routing table.

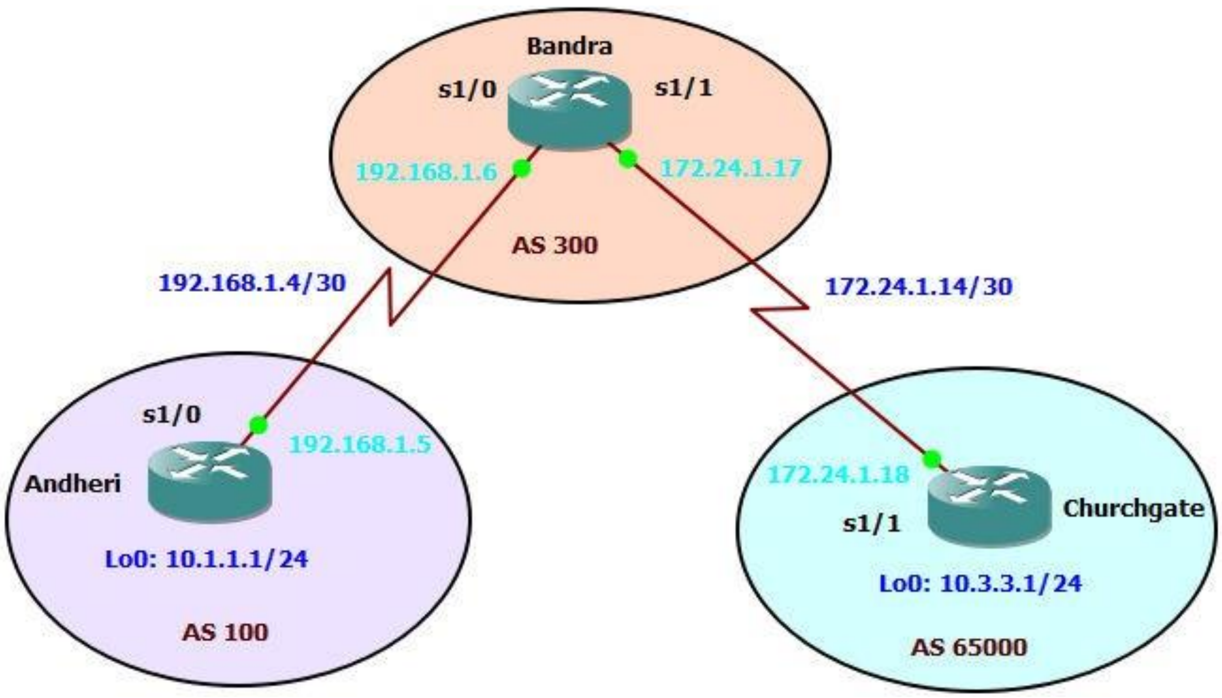
R1# show ip route

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BG
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF in
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA externa
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2
       ia - IS-IS inter area, * - candidate default, U - per-
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 209.165.201.1 to network 0.0.0.0

     209.165.201.0/30 is subnetted, 1 subnets
C       209.165.201.0 is directly connected, Serial1/0
     209.165.202.0/30 is subnetted, 1 subnets
C       209.165.202.128 is directly connected, Serial1/1
C    192.168.1.0/24 is directly connected, Loopback0
S*   0.0.0.0/0 [2/0] via 209.165.201.1
```

Practical No - 2

Aim: Using the AS_PATH Attribute

Topolgy :



Objective :

- Use BGP commands to prevent private AS numbers from being advertised to the outside world.
- Use the AS_PATH attribute to filter BGP routes based on their source AS number

Step 1 : Prepare the routers for the lab.

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

Step 2 : Configure the hostname and interface addresses.

a. You can copy and paste the following configurations into your routers to begin.

Router R1 (hostname Andheri)

Andheri(config)# interface Loopback0

Andheri(config-if)# ip address 10.1.1.1 255.255.255.0

Andheri(config-if)# exit

Andheri(config)# interface Serial0/0/0

Andheri(config-if)# ip address 192.168.1.5 255.255.255.252

Andheri(config-if)# no shutdown

Andheri(config-if)# end

Andheri#

```
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#hostname Andheri
Andheri(config)#int loopback 0
Andheri(config-if)#
*May  7 09:30:42.867: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0
Andheri(config-if)#ip address 10.1.1.1 255.255.255.0
Andheri(config-if)#exit
Andheri(config)#int s1/0
Andheri(config-if)#ip address 192.168.1.5 255.255.255.252
Andheri(config-if)#no shutdown
Andheri(config-if)#
*May  7 09:31:41.315: %LINK-3-UPDOWN: Interface Serial1/0, changed state to up
```

Router R2 (hostname Bandra)

Bandra(config)# interface Loopback0

Bandra(config-if)# ip address 10.2.2.1 255.255.255.0

Bandra(config-if)# interface Serial0/0/0

Bandra(config-if)# ip address 192.168.1.6 255.255.255.252

Bandra(config-if)# no shutdown

Bandra(config-if)# exit

Bandra(config)# interface Serial0/0/1

Bandra(config-if)# ip address 172.24.1.17 255.255.255.252

Bandra(config-if)# no shutdown

Bandra(config-if)# end

Bandra#

```
R2#
R2#conf t
Enter configuration commands, one per line.  End with CNT
R2(config)#hostname Bandra
Bandra(config)#int loopback 0
Bandra(config-if)#ip addr
*May  7 09:31:30.407: %LINEPROTO-5-UPDOWN: Line protocol
Bandra(config-if)#ip address 10.2.2.1 255.255.255.0
Bandra(config-if)#exit
Bandra(config)#int s1/0
Bandra(config-if)#ip address 192.168.1.6 255.255.255.252
Bandra(config-if)#no shutdown
Bandra(config-if)#exit
```

Router R3 (hostname ChurchGate)

Churchgate(config)# interface Loopback0

Churchgate(config-if)# ip address 10.3.3.1 255.255.255.0

Churchgate(config-if)# exit

Churchgate(config)# interface Serial0/0/1

Churchgate(config-if)# ip address 172.24.1.18 255.255.255.252

Churchgate(config-if)# no shutdown

Churchgate(config-if)# end

Churchgate#

```
Bandra(config)#int s1/1
Bandra(config-if)#ip address 172.24.1.17 255.255.255.252
Bandra(config-if)#no shutdown
Bandra(config-if)#
*May  7 09:33:39.591: %LINK-3-UPDOWN: Interface Serial1/1,
```

```
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#hostname Churchgate
Churchgate(config)#int loopback 0
Churchgate(config-if)#ip[
*May  7 09:33:31.243: %LINEPROTO-5-UPDOWN: Line protocol on I
Churchgate(config-if)#ip address 10.3.3.1 255.255.255.0
Churchgate(config-if)#exit
Churchgate(config)#int s1/1
Churchgate(config-if)#ip address 172.24.1.18 255.255.255.252
Churchgate(config-if)#no shutdown
Churchgate(config-if)#
*May  7 09:34:39.795: %LINK-3-UPDOWN: Interface Serial1/1, ch
```

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

```
Bandra#ping 192.168.1.5

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 = 20/32/40 ms
Bandra#
Bandra#
Bandra#ping 172.24.1.18

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 = 28/36/48 ms
Bandra#
```

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.

Andheri(config)# router bgp 100

Andheri(config-router)# neighbor 192.168.1.6 remote-as 300

Andheri(config-router)# network 10.1.1.0 mask 255.255.255.0

```
Andheri#conf t
Enter configuration commands, one per line.  End with CNTL/
Andheri(config)#router bgp 100
Andheri(config-router)#neighbor 192.168.1.6 remote-as 300
Andheri(config-router)#network 10.1.1.0 mask 255.255.255.0
```

Bandra(config)# router bgp 300

Bandra(config-router)# neighbor 192.168.1.5 remote-as 100

Bandra(config-router)# neighbor 172.24.1.18 remote-as 65000

Bandra(config-router)# network 10.2.2.0 mask 255.255.255.0

```
Bandra#conf t
Enter configuration commands, one per line.  End with CNTL/
Bandra(config)#router bgp 300
Bandra(config-router)#neighbor 192.168.1.5 remote-as 100
Bandra(config-router)#
*May  7 10:04:59.051: %BGP-5-ADJCHANGE: neighbor 192.168.1.
Bandra(config-router)#neighbor 172.24.1.18 remote-as 65000
Bandra(config-router)#network 10.2.2.0 mask 255.255.255.0
```

Churchgate(config)# router bgp 65000

Churchgate(config-router)# neighbor 172.24.1.17 remote-as 300

Churchgate(config-router)# network 10.3.3.0 mask 255.255.255.0

```
Churchgate#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Churchgate(config)#router bgp 65000
Churchgate(config-router)#neighbor 172.24.1.17 remote-as 300
Churchgate(config-router)#
*May  7 10:04:44.195: %BGP-5-ADJCHANGE: neighbor 172.24.1.17 U
Churchgate(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.

Bandra# show ip bgp neighbors

```
Bandra#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 00:01:30
  Last read 00:00:13, last write 00:00:44, hold time is 180, kee
econds
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    New ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0

                         Sent       Rcvd
    Opens:                  1          1
    Notifications:          0          0
    Updates:                3          1
    Keepalives:             2          3
    Route Refresh:          0          0
    Total:                  6          5
  Default minimum time between advertisement runs is 30 seconds
```

<u>Step 4</u> : Remove the private AS.

a. DBandralay the Andheri routing table using the show ip route command. Andherishould have a route to both 10.2.2.0 and 10.3.3.0. Troubleshoot if necessary.

Andheri#show ip route

```
Andheri#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

     10.0.0.0/24 is subnetted, 3 subnets
B       10.3.3.0 [20/0] via 192.168.1.6, 00:04:51
B       10.2.2.0 [20/0] via 192.168.1.6, 00:05:22
C       10.1.1.0 is directly connected, Loopback0
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.4 is directly connected, Serial1/0
```

b . Ping again, this time as an extended ping, sourcing from the Loopback0 interface address. **ping 10.3.3.1 source 10.1.1.1 or ping 10.3.3.1 source Lo0**

```
Andheri#ping 10.3.3.1 source 10.1.1.1

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 = 36/52/64 ms
Andheri#
Andheri#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 - in
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   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
```

c.Now check the BGP table on Andheri. The AS_ PATH to the 10.3.3.0 network shouldbe AS 300. It no longer has the private AS in the path.

Andheri# show ip bgp

```
Andheri#ping 10.3.3.1 source 10.1.1.1

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 = 20/60/9
Andheri#show ip bgp
BGP table version is 5, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   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 Bandra.

Bandra(config)# ip as-path access-list 1 deny ^100$

Bandra(config)# ip as-path access-list 1 permit .*

```
Bandra#
Bandra#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Bandra(config)#ip as-path access-list 1 deny ^100$
Bandra(config)#ip as-path access-list 1 permit .*
```

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

Bandra(config)# router bgp 300

Bandra (config-router)# neighbor 192.168.1.5 remove-private-as

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

c. Use the clear ip bgp * command to reset the routing information. Wait several seconds and then check the routing table for BANDRA. The route to 10.1.1.0 should bein the routing table.

Andheri# show ip route

```
Bandra#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

     172.24.0.0/30 is subnetted, 1 subnets
C       172.24.1.16 is directly connected, Serial1/1
     10.0.0.0/24 is subnetted, 3 subnets
B       10.3.3.0 [20/0] via 172.24.1.18, 00:13:19
C       10.2.2.0 is directly connected, Loopback0
B       10.1.1.0 [20/0] via 192.168.1.5, 00:13:20
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.4 is directly connected, Serial1/0
```

d. Return to BANDRA and verify that the filter is working as intended.

Bandra# show ip bgp regexp ^100$

```
Bandra#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
Origin codes: i - IGP, e - EGP, ? - incomplete

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

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

```
Bandra#tclsh
```

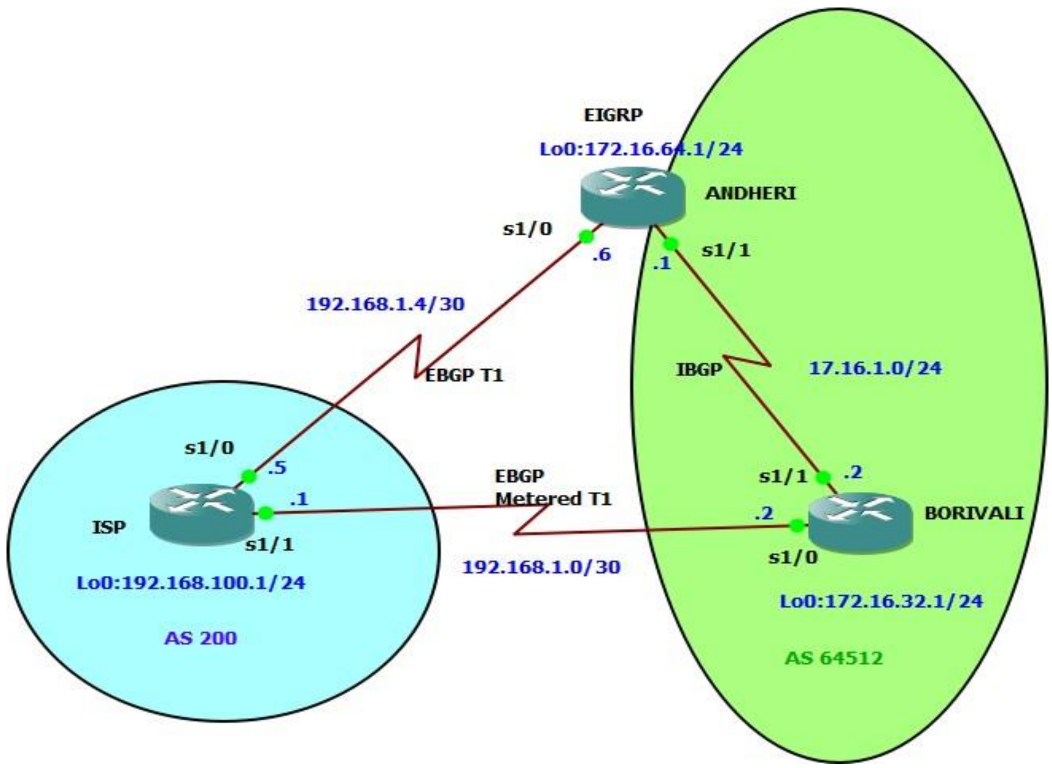
```
Bandra(tcl)#foreach address {
+>10.1.1.1
+>10.2.2.1
+>10.3.3.1
+>192.168.1.5
+>192.168.1.6
+>172.24.1.17
+>172.24.1.18
+>} { ping $address }
```

```
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 = 28/40/64 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 = 4/4/4 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 = 16/32/48 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 = 16/28/40 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 = 48/59/64 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 = 48/56/68 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 = 16/31/48 ms
```

Practical No - 3

Aim: Configuring IBGP and EBGP Sessions, Local Preference, and MED

Topolgy :



Objectives

- For IBGP peers to correctly exchange routing information, use the **next-hop-self** command with the **Local-Preference** and **MED** attributes.
- Ensure that the flat-rate, unlimited-use T1 link is used for sending and receiving data to and fromthe AS 200 on ISP and that the metered T1 only be used in the event that the primary T1 link has failed.

Step 1: Configure interface addresses.

Router R1 (hostname ISP)

ISP(config)# **interface Loopback0**

ISP(config-if)# **ip address 192.168.100.1 255.255.255.0**

ISP(config-if)# **exit**

ISP(config)# **interface Serial0/0/0**

ISP(config-if)# **ip address 192.168.1.5 255.255.255.252**

ISP(config-if)# **no shutdown**

ISP(config-if)# **exit**

ISP(config)# **interface Serial0/0/1**

ISP(config-if)# **ip address 192.168.1.1 255.255.255.252**

ISP(config-if)# **no shutdown**

ISP(config-if)# **end**

```
ISP(config-if)#interface Loopback0
ISP(config-if)#ip address 192.168.100.1 255.255.255.0
ISP(config-if)#exit
ISP(config)#
ISP(config)#int s1/0
ISP(config-if)#ip address 192.168.1.5 255.255.255.252
ISP(config-if)#no shutdown
ISP(config-if)#exit
ISP(config)#
*May 18 17:42:51.491: %LINK-3-UPDOWN: Interface Serial1
ISP(config)#
*May 18 17:42:52.495: %LINEPROTO-5-UPDOWN: Line protoco
ISP(config)#
ISP(config)#int s1/1
ISP(config-if)#ip address 192.168.1.1 255.255.255.252
ISP(config-if)#no shutdown
```

Router R2 (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 Serial0/0/0**

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 Serial0/0/1**

SanJose1(config-if)# **ip address 172.16.1.1 255.255.255.0**

SanJose1(config-if)# **no shutdown**

SanJose1(config-if)# **end**

```
ANDHERI(config)#interface Loopback0
ANDHERI(config-if)#
*May 18 17:42:40.167: %LINEPROTO-5-UPDOWN: Line protocol
ANDHERI(config-if)#ip address 172.16.64.1 255.255.255.0
ANDHERI(config-if)#exit
ANDHERI(config)#
ANDHERI(config)#int s1/0
ANDHERI(config-if)#ip address 192.168.1.6 255.255.255.252
ANDHERI(config-if)#no shutdown
ANDHERI(config-if)#exit
ANDHERI(config)#
*May 18 17:43:11.899: %LINK-3-UPDOWN: Interface Serial1/0
ANDHERI(config)#
*May 18 17:43:12.903: %LINEPROTO-5-UPDOWN: Line protocol
ANDHERI(config)#
ANDHERI(config)#int s1/1
ANDHERI(config-if)#ip address 172.16.1.1 255.255.255.0
ANDHERI(config-if)#no shutdown
```

Router R3 (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 Serial0/0/0**

SanJose2(config-if)# **ip address 192.168.1.2 255.255.255.252**

SanJose2(config-if)# **no shutdown**

SanJose2(config-if)# **exit**

SanJose2(config)# **interface Serial0/0/1**

SanJose2(config-if)# **ip address 172.16.1.2 255.255.255.0**

SanJose2(config-if)# **no shutdown**

SanJose2(config-if)# **end**

```
BORIVALI(config)#interface Loopback0
BORIVALI(config-if)#
*May 18 17:43:25.783: %LINEPROTO-5-UPDOWN: Line protocol o
BORIVALI(config-if)#ip address 172.16.32.1 255.255.255.0
BORIVALI(config-if)#exit
BORIVALI(config)#
BORIVALI(config)#int s1/0
BORIVALI(config-if)#ip address 192.168.1.2 255.255.255.252
BORIVALI(config-if)#no shutdown
BORIVALI(config-if)#exit
BORIVALI(config)#
*May 18 17:43:54.311: %LINK-3-UPDOWN: Interface Serial1/0,
BORIVALI(config)#
BORIVALI(config)#
*May 18 17:43:55.315: %LINEPROTO-5-UPDOWN: Line protocol o
BORIVALI(config)#int s1/1
BORIVALI(config-if)#ip address 172.16.1.2 255.255.255.0
```

Step 2: Configure EIGRP.

Configure EIGRP between the SanJose1 and SanJose2 routers. (Note: If using an IOS prior to 15.0, use theno auto-summary router configuration command to disable automatic summarization. This command is the default beginning with IOS 15.)

SanJose1(config)# **router eigrp 1**

SanJose1(config-router)# **network 172.16.0.0**

```
ANDHERI(config)# router eigrp 1
ANDHERI(config-router)#network 172.16.0.0
```

SanJose2(config)# **router eigrp 1**

SanJose2(config-router)# **network 172.16.0.0**

```
BORIVALI(config)#router eigrp 1
BORIVALI(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(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**

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

If multiple pathways to the BGP neighbor exist, the router can use multiple IP interfaces to communicatewith the neighbor. The source IP address therefore depends on the outgoing interface. The **update-source**

lo0 command instructs the router to use the IP address of the interface Loopback0 as the source IP address for all BGP messages sent to that neighbor.

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

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**

```
BORIVALI(config)#router bgp 64512
BORIVALI(config-router)#neighbor 172.16.64.1 remote-as 64512
BORIVALI(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 connection.

SanJose2# **show ip bgp neighbors**

```
BORIVALI#show ip bgp neighbors
BGP neighbor is 172.16.64.1,  remote AS 64512,
  BGP version 4, remote router ID 172.16.64.1
  BGP state = Established, up for 00:00:47
  Last read 00:00:47, last write 00:00:47, hol
  Neighbor capabilities:
    Route refresh: advertised and received(new
    New ASN Capability: advertised and receive
    Address family IPv4 Unicast: advertised an
  Message statistics:
    InQ depth is 0
    OutQ depth is 0

                         Sent       Rcvd
    Opens:                  1          1
    Notifications:          0          0
    Updates:                0          0
```

Step 4: Configure EBGP and verify BGP neighbors.

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

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**

```
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
```

e. 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 theISP. Later in this lab we will configure a default route to the ISP.

SanJose1(config)# **ip route 172.16.0.0 255.255.0.0 null0**

```
ANDHERI(config)#ip route 172.16.0.0 255.255.0.0 null0
ANDHERI(config)#
```

f. Configure SanJose1 as an EBGP peer to ISP.

SanJose1(config)# **router bgp 64512**

SanJose1(config-router)# **neighbor 192.168.1.5 remote-as 200**

SanJose1(config-router)# **network 172.16.0.0**

```
ANDHERI(config)#router bgp 64512
ANDHERI(config-router)#neighbor 192.168.1.5 remote-as 200
ANDHERI(config-router)#network 172.16.0.0
ANDHERI(config-router)#exit
```

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

SanJose1# **show ip bgp neighbors**

```
ANDHERI#show ip bgp neighbors
BGP neighbor is 172.16.32.1,  remote AS 64512, internal li
  BGP version 4, remote router ID 172.16.32.1
  BGP state = Established, up for 00:02:49
  Last read 00:00:56, last write 00:00:21, hold time is 18
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    New ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
```

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

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**

```
BORIVALI(config)#ip route 172.16.0.0 255.255.0.0 null0
BORIVALI(config)#router bgp 64512
BORIVALI(config-router)#neighbor 192.168.1.1 remote-as 200
BORIVALI(config-router)#
*May 18 18:00:01.031: %BGP-5-ADJCHANGE: neighbor 192.168.1.1 Up
BORIVALI(config-router)#network 172.16.0.0
```

Step 5: View BGP summary output.

In Step 4, the **show ip bgp neighbors** command was used to verify that SanJose1 and ISP had reached the established state. A useful alternative command is **show ip bgp summary**. The output should be similar to the following.

SanJose2# **show ip bgp summary**

```
BORIVALI# show ip bgp summary
BGP router identifier 172.16.32.1, local AS number 64512
BGP table version is 5, main routing table version 5
2 network entries using 264 bytes of memory
4 path entries using 208 bytes of memory
5/2 BGP path/bestpath attribute entries using 840 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
Bitfield cache entries: current 2 (at peak 2) using 64 bytes of memory
BGP using 1400 total bytes of memory
BGP activity 2/0 prefixes, 4/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/Pfx
172.16.64.1     4        64512       7       8        5    0    0 00:04:18        2
192.168.1.1     4          200       5       4        3    0    0 00:00:26        1
```

Step 6: Verify which path the traffic takes.

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

ISP# **clear ip bgp ***

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

g. 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# **ping 172.16.64.1**

```
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:

*May 18 18:02:42.575: %BGP-5-ADJCHANGE: neighbor 192.168.1.6 Up .....
Success rate is 0 percent (0/5)
```

ISP# **ping 172.16.1.1**

```
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)
```

h. 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# **ping 172.16.32.1**

```
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 = 12/31/48 ms
```

ISP# **ping 172.16.1.2**

```
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 = 16/24/32 ms
```

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

ISP# **show ip bgp**

```
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
Origin codes: i - IGP, e - EGP, ? - incomplete

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

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 specifythe source address of ISP Lo0 to test.

ISP# **ping 172.16.1.1 source 192.168.100.1**

```
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 = 16/38/48 ms
```

ISP# **ping 172.16.32.1 source 192.168.100.1**

```
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 = 4/32/48 ms
```

ISP# **ping 172.16.1.2 source 192.168.100.1**

```
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 = 16/28/48 ms
```

ISP# **ping 172.16.64.1 source 192.168.100.1**

```
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 = 24/35/56 ms
```

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

j. Issue the following commands on the ISP router.

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**

```
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
ISP(config-router)#exit
```

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

ISP# **show ip bgp**

```
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 -
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  172.16.0.0       192.168.1.6              0             0 64512 i
*>                  192.168.1.2              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
```

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

SanJose2# **show ip route**

```
BORIVALI#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
```

m. 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(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 serial 0/0/1**

ISP(config-if)# **shutdown**

```
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)#int s1/1
ISP(config-if)#shutdown
```

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

SanJose2# **show ip bgp**

```
BORIVALI#show ip bgp
BGP table version is 14, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
              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
* i192.168.100.0    192.168.1.5              0    100      0 200 i
*>                  192.168.1.1              0             0 200 i
```

SanJose2# **show ip route**

```
BORIVALI#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 lev
       ia - IS-IS inter area, * - candidate default, U - per-user static
       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, Serial1/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 00:11:47, Serial1/1
```

SanJose1(config)# **router bgp 64512**

SanJose1(config-router)# **neighbor 172.16.32.1 next-hop-self**

```
ANDHERI(config)#router bgp 64512
ANDHERI(config-router)#neighbor 172.16.32.1 next-hop-self
ANDHERI(config-router)#exit
```

SanJose2(config)# **router bgp 64512**

SanJose2(config-router)# **neighbor 172.16.64.1 next-hop-self**

```
BORIVALI(config-router)#router bgp 64512
BORIVALI(config-router)#neighbor 172.16.64.1 next-hop-self
```

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

SanJose1# **clear ip bgp ***

```
ANDHERI#clear ip bgp *
```

SanJose2# **clear ip bgp ***

```
BORIVALI#clear ip bgp *
```

p. After the routers have returned to established BGP speakers, issue the **show ip bgp** command onSanJose2 and notice that the next hop is now SanJose1 instead of ISP.

SanJose2# **show ip bgp**

```
BORIVALI#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
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i172.16.0.0       172.16.64.1              0    100      0 i
* i192.168.100.0    172.16.64.1              0    100      0 200 i
```

q. 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# **show ip route**

```
BORIVALI#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
       ia - IS-IS inter area, * - candidate default, U - per-user static r
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

r. 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 routingtable shows 192.168.100.0/24 will now have a better path through ISP.

ISP(config)# **interface serial 0/0/1**

ISP(config-if)# **no shutdown**

```
ISP(config)#int s1/1
ISP(config-if)#no shutdown
```

SanJose2# **show ip route**

```
BORIVALI#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - B
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF i
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA extern
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2
       ia - IS-IS inter area, * - candidate default, U - per
       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, Serial1/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 00:15:05
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial1/0
B    192.168.100.0/24 [20/0] via 192.168.1.1, 00:00:18
```

Step 8: Set BGP local preference.

s. 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(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**

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

SanJose2(config)# **route-map SECONDARY_T1_IN permit 10**

SanJose2(config-route-map)# **set local-preference 125**

SanJose1(config-route-map)# **exit**

SanJose2(config)# **router bgp 64512**

SanJose2(config-router)# **neighbor 192.168.1.1 route-map SECONDARY_T1_IN in**

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

t. 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# **clear ip bgp * soft**

```
ANDHERI#clear ip bgp * soft
```

SanJose2# **clear ip bgp * soft**

```
BORIVALI#clear ip bgp * soft
```

SanJose1# **show ip bgp**

```
ANDHERI#show ip bgp
BGP table version is 6, local router ID is 172.16.64.1
Status codes: s suppressed, d damped, h history, * valid, > best, i
              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# **show ip bgp**

```
BORIVALI#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
              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             0 200 i
*>i                 172.16.64.1              0    150      0 200 i
```

Step 9: Set BGP MED.

u. In the previous step we saw that SanJose1 and SanJose2 will route traffic for 192.168.100.0/24using 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# **show ip bgp**

```
ISP#show ip bgp
BGP table version is 11, local router ID is 192.168.100.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
              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# **show ip route**

```
ISP#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OS
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA ex
       E1 - OSPF external type 1, E2 - OSPF external typ
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1
       ia - IS-IS inter area, * - candidate default, U -
       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, 00:04:56
     192.168.1.0/30 is subnetted, 2 subnets
C       192.168.1.0 is directly connected, Serial1/1
C       192.168.1.4 is directly connected, Serial1/0
C    192.168.100.0/24 is directly connected, Loopback0
```

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

```
BORIVALI#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)
```

```
Reply to request 4 (8 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 = 8/47/68 ms
```

If you are unfamiliar with the **record** option, the important thing to note is that each IP address in brackets is an outgoing interface. The output can be interpreted as follows:

1. A ping that is sourced from 172.16.32.1 exits SanJose2 through s0/0/1, 172.16.1.2. It then arrivesat the s0/0/1 interface for SanJose1.
2. SanJose1 S0/0/0, 192.168.1.6, routes the packet out to arrive at the S0/0/0 interface of ISP.3.
   The target of 192.168.100.1 is reached: 192.168.100.1.
4. The packet is next forwarded out the S0/0/1, 192.168.1.1 interface for ISP and arrives at theS0/0/0 interface for SanJose2.
5. SanJose2 then forwards the packet out the last interface, loopback 0, 172.16.32.1.

Although the unlimited use of the T1 from SanJose1 is preferred here, ISP currently takes the link from SanJose2 for all return traffic.

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

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**

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

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**

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

v. 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# **clear ip bgp * soft**

```
ANDHERI#clear ip bgp * soft
```

SanJose2# **clear ip bgp * soft**

```
BORIVALI#clear ip bgp * soft
```

SanJose1# **show ip bgp**

```
ANDHERI#show ip bgp
BGP table version is 6, 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# **show ip bgp**

```
BORIVALI#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
              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
```

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

```
BORIVALI#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 sec
Packet sent with a source address of 172.16.32.1
Packet has IP options:  Total option bytes= 39, padded length=40
```

```
Reply to request 3 (60 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 (52 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 = 40/52/60 ms
```

ISP# **show ip bgp**

```
ISP#show ip bgp
BGP table version is 11, local router ID is 192.168.100.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
              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
```

Step 10: Establish a default route.

The final step is to establish a default route that uses a policy statement that adjusts to changes in the network.

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(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**

```
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-if)#ip address 10.0.0.1 255.255.255.0
ISP(config-if)#exit
```

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# **show ip route**

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

Gateway of last resort is 172.16.32.1 to network 0.0.0.0

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 ma
D       172.16.32.0/24 [90/2297856] via 172.16.1.2, 02:14
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial1/1
C       172.16.64.0/24 is directly connected, Loopback0
B    192.168.100.0/24 [200/0] via 172.16.32.1, 01:44:43
B*   0.0.0.0/0 [200/0] via 172.16.32.1, 01:44:43
```

SanJose2# **show ip route**

```
BORIVALI#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OS
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA ex
       E1 - OSPF external type 1, E2 - OSPF external typ
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1
       ia - IS-IS inter area, * - candidate default, U -
       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 m
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, Serial1/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 02:1
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial1/0
B    192.168.100.0/24 [20/0] via 192.168.1.1, 01:44:59
B*   0.0.0.0/0 [20/0] via 192.168.1.1, 01:45:00
```

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

SanJose2# **show ip bgp**

```
BORIVALI# show ip bgp
BGP table version is 9, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 0.0.0.0          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
```

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

SanJose2# **traceroute 10.0.0.1**

```
BORIVALI#traceroute 10.0.0.1

Type escape sequence to abort.
Tracing the route to 10.0.0.1

  1 192.168.1.1 [AS 200] 28 msec 32 msec 32 msec
  2 192.168.1.1 [AS 200] !H  !H  !H
```

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

ISP(config)# **interface serial 0/0/0**

ISP(config-if)# **shutdown**

```
ISP(config)#int s1/0
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# **show ip route**

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

Gateway of last resort is 172.16.32.1 to network 0.0.0.0

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 ma
D       172.16.32.0/24 [90/2297856] via 172.16.1.2, 02:15
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial1/1
C       172.16.64.0/24 is directly connected, Loopback0
B    192.168.100.0/24 [200/0] via 172.16.32.1, 01:45:58
B*   0.0.0.0/0 [200/0] via 172.16.32.1, 01:45:58
```

SanJose2# **show ip route**

```
BORIVALI#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B -
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA exte
       E1 - OSPF external type 1, E2 - OSPF external type
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1,
       ia - IS-IS inter area, * - candidate default, U - p
       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 mas
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, Serial1/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 02:15:
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial1/0
B    192.168.100.0/24 [20/0] via 192.168.1.1, 01:46:10
B*   0.0.0.0/0 [20/0] via 192.168.1.1, 01:46:10
```

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

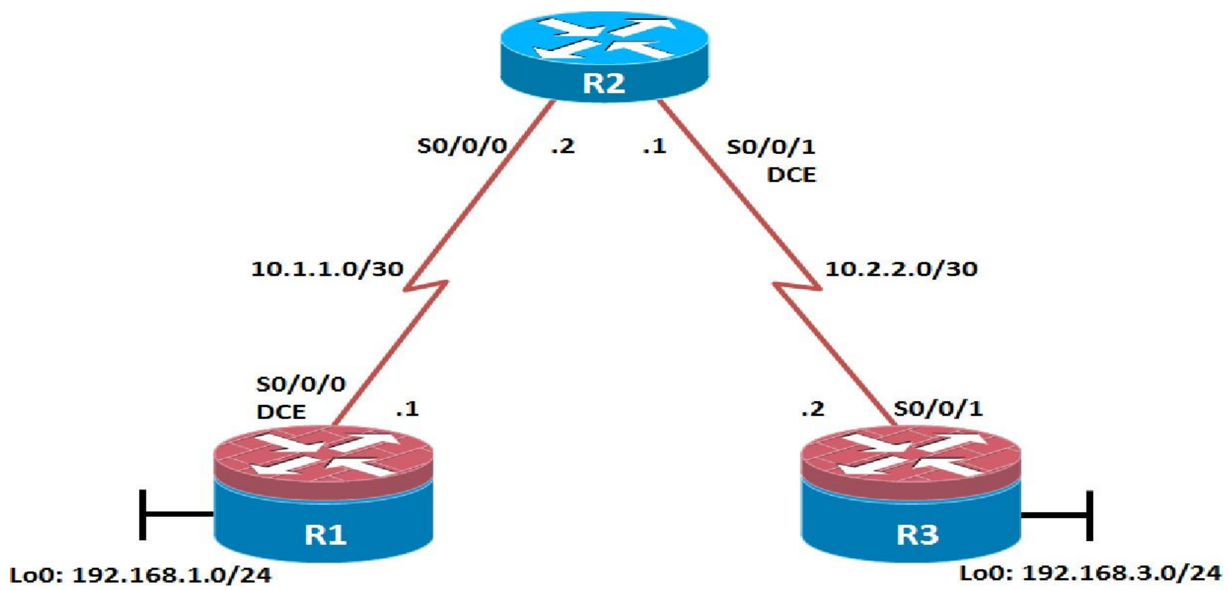
SanJose1# **trace 10.0.0.1**

```
ANDHERI#trace 10.0.0.1

Type escape sequence to abort.
Tracing the route to 10.0.0.1

  1 172.16.1.2 64 msec 32 msec 32 msec
  2 192.168.1.1 [AS 200] 56 msec 28 msec 64 msec
  3 192.168.1.1 [AS 200] !H  !H  !H
ANDHERI#
```

Practical No - 4

Aim: Secure the Management Plane

Topology:



Objectives :

- Secure management access.
- Configure enhanced username password security.
- Enable AAA RADIUS authentication.
- Enable secure remote management.

Step 1: Configure loopbacks and assign addresses.

Cable the network as shown in the topology diagram. Erase the startup configuration and reload eachrouter 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. You can copy and paste the following configurations into your routers to begin.

Router R1

interface Loopback 0

ip address 192.168.1.1 255.255.255.0

```
exit
interface Serial0/0/0
  ip address 10.1.1.1 255.255.255.252
  no shutdown
exit
end
```

Router R2

```
interface Serial0/0/0
 ip address 10.1.1.2 255.255.255.252
 no shutdown
exit
interface Serial0/0/1
 ip address 10.2.2.1 255.255.255.252
 no shutdown
exit
end
```

Router R3

```
interface Loopback0
  ip address 192.168.3.1 255.255.255.0
exit
interface Serial0/0/1
 ip address 10.2.2.2 255.255.255.252
 no shutdown
exit
end
```

Step 2: Configure static routes.

R1(config)# **ip route 0.0.0.0 0.0.0.0 10.1.1.2**

R3(config)# **ip route 0.0.0.0 0.0.0.0 10.2.2.1**

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**

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 = 1/1/1 msType

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 = 1/2/4 msType

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 = 1/1/4 msType

escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.2.2.1, timeout is 2 seconds:

!!!!!

Step 3: Secure management access.

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

R1(config)# **security passwords min-length 10**

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

R1(config)# **enable secret class12345**

3. 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(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**

Configure the password on the vty lines for router R1.

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**

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

R1(config)# **line aux 0**

R1(config-line)# **no exec**

R1(config-line)# **end**

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

____R1(config)

service password-encryption

R1(config)#

6. 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(config)# **banner motd $Unauthorized access strictly prohibited!$**

R1(config)# **exit**

Step 4: Configure enhanced username password security.

1. 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(config)# **username JR-ADMIN secret class12345**

R1(config)# **username ADMIN secret class54321**

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

R1(config)# **line console 0**

R1(config-line)# **login local**

R1(config-line)# **exit**

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

R1(config)# **line vty 0 4**

R1(config-line)# **login local**

R1(config-line)# **end**

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

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

R1# **telnet 10.2.2.2**

Trying 10.2.2.2 ... Open

Unauthorized access strictly prohibited!

User Access Verification

Username: **ADMIN**

Password:

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

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

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**

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

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**

2. Assign both RADIUS servers to a server group.

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**

3. 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(config)# **aaa authentication login default group RADIUS-GROUP local**

4. 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 sensitivelocal database.

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

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

R1(config)# **line vty 0 4**

R1(config-line)# **login authentication TELNET-LOGIN**

R1(config-line)# **exit**

Repeat the steps 5a to 5g on R3.

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

R1# **telnet 10.2.2.2**

Trying 10.2.2.2 ... Open

Unauthorized access strictly prohibited!

User Access Verification

Username: **admin**

Password:

Authentication failed

Username: **ADMIN**

Password:

Step 6: Enabling secure remote management using SSH.

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

R1(config)# **ip domain-name ccnasecurity.com**

2. 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(config)# **crypto key zeroize rsa**

3. 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(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]

R1(config)#

Jan 10 13:44:44.711: %SSH-5-ENABLED: SSH 1.99 has been enabled

4. Cisco routers support two versions of SSH:

- **SSH version 1 (SSHv1)**: Original version but has known vulnerabilities.

– **SSH version 2 (SSHv2)**: Provides better security using the Diffie-Hellman key exchange and the strong integrity-checking message authentication code (MAC).

Configure SSH version 2 on R1.

R1(config)# **ip ssh version 2** R1(config)#

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

R1(config)# **line vty 0 4**

R1(config-line)# **transport input ssh**

R1(config-line)# **end**

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

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

AAAAB3NzaC1yc2EAAAADAQABAAAAgQC3Lehh7ReYlgyDzls6wq+mFzxqzoaZFr9XGx +Q/ yio

dFYw00hQo80tZy1W1Ff3Pz6q7Qi0y00urwddHZ0kBZceZK9EzJ6wZ+9a87KKDETCWrGSLi6c8lE/y4K+

Z/oVrMMZk7bpTM1MFdP41YgkTf35utYv+TcqbsYo++KJiYk+xw==

7. Repeat the steps 6a to 6f on R3.
8. 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# **ssh -l ADMIN 10.2.2.2**

Password:

Unauthorized access strictly prohibited!

R3>

R3> **en**

Password:

R3#

Device Configurations
Router R1

```
service password-encryption
hostname R1
security passwords min-length 10
enable secret 5 $1$t6eK$FZ.JdmMLj8QSgNkpChyZz.
aaa new-model
aaa group server radius RADIUS-GROUP
 server name RADIUS-1
 server name RADIUS-2
aaa authentication login default group RADIUS-GROUP local
aaa authentication login TELNET-LOGIN group RADIUS-GROUP local-case
ip domain name ccnasecurity.com
username JR-ADMIN secret 5 $1$0u0q$lwimCZIAuQtV4C1ezXL1S0
username ADMIN secret 5 $1$NSVD$/YjzB7Auyes1sAt4qMfpd.
ip ssh version 2
interface Loopback0
description R1 LAN
 ip address 192.168.1.1 255.255.255.0
interface Serial0/0/0
 description R1 --> R2
 ip address 10.1.1.1 255.255.255.252
 no fair-queue
ip route 0.0.0.0 0.0.0.0 10.1.1.2
radius server RADIUS-1
 address ipv4 192.168.1.101 auth-port 1645 acct-port 1646
 key 7 107C283D2C2221465D493A2A717D24653017
radius server RADIUS-2
 address ipv4 192.168.1.102 auth-port 1645 acct-port 1646
 key 7 03367A2F2F3A12011C44090442471C5C162E
banner motd ^CUnauthorized access strictly prohibited!^C

line con 0
 exec-timeout 5 0
```

```
 password 7 070C285F4D061A0A19020A1F17
 logging synchronous
line aux 0
 no exec
password 7 060506324F411F0D1C0713181F
 login authentication TELNET-LOGIN
 transport input ssh
end
```

Router R2

```
hostname R2
enable secret 5 $1$DJS7$xvJDW87zLs8pSJDFUlCPB1
interface Serial0/0/0
 ip address 10.1.1.2 255.255.255.252
 no fair-queue

interface Serial0/0/1
 ip address 10.2.2.1 255.255.255.252
 clock rate 128000

ip route 192.168.1.0 255.255.255.0 10.1.1.1
ip route 192.168.3.0 255.255.255.0 10.2.2.2

line con 0
 exec-timeout 0 0
 logging synchronous

line vty 0 4
 password cisco
 login
end
```

Router R3

```
service password-encryption
```

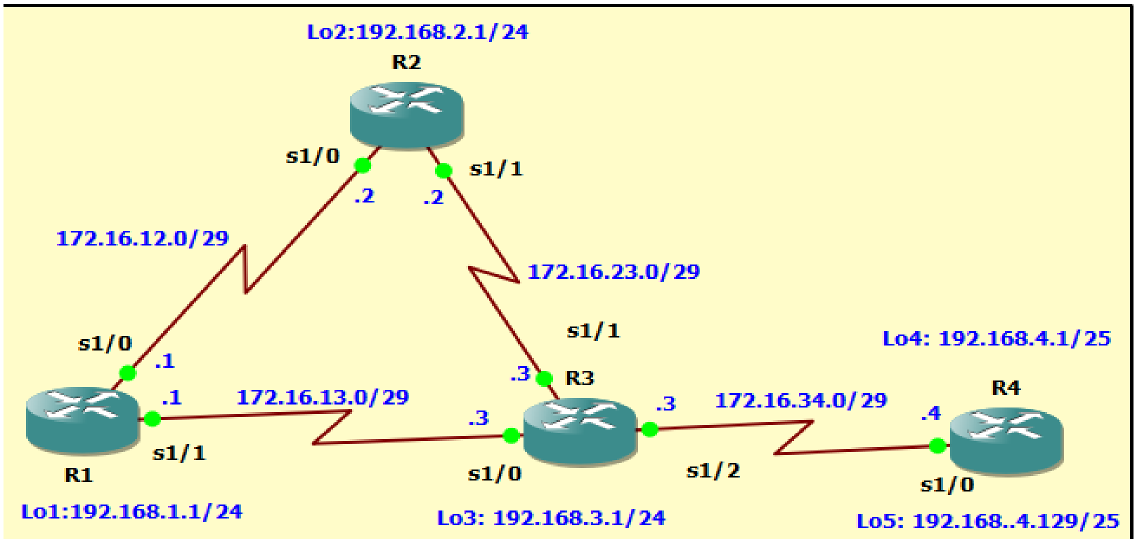
```
hostname R3
security passwords min-length 10
enable secret 5 $1$5OY4$4J6VFlvGNKjwQ8XtajgUk1
aaa new-model
aaa group server radius RADIUS-GROUP
 server name RADIUS-1
 server name RADIUS-2
aaa authentication login default group RADIUS-GROUP local
aaa authentication login TELNET-LOGIN group RADIUS-GROUP local-case
ip domain name ccnasecurity.com

username JR-ADMIN secret 5 $1$b4m1$RVmjL9S3gxKh1xr8qzNqr/username
ADMIN secret 5 $1$zGV7$pVgSEbinvXQ7f7uyxeKBj
ip ssh version 2

interface Loopback0
 description R3 LAN
 ip address 192.168.3.1 255.255.255.0

interface Serial0/0/1
 description R3 --> R2
 ip address 10.2.2.2 255.255.255.252

ip route 0.0.0.0 0.0.0.0 10.2.2.1

radius server RADIUS-1
 address ipv4 192.168.1.101 auth-port 1645 acct-port 1646
 key 7 01212720723E354270015E084C5000421908

radius server RADIUS-2
 address ipv4 192.168.1.102 auth-port 1645 acct-port 1646
```

```
 key 7 003632222D6E384B5D6C5C4F5C4C1247000F

banner motd ^CUnauthorized access strictly prohibited!^C

line con 0
 exec-timeout 5 0
 password 7 104D000A0618110402142B3837
 logging synchronous

line aux 0no
 exec

line vty 0 4
 exec-timeout 5 0
 password 7 070C285F4D060F110E020A1F17
 login authentication TELNET-LOGIN
 transport input ssh
end
```

Practical No - 5

Aim : **Configure and Verify Path Control Using PBR**

Topology :



Objectives

- Configure and verify policy-based routing.
- Select the required tools and commands to configure policy-based routing operations.
- Verify the configuration and operation by using the proper show and debug commands.

Step 1: Configure loopbacks and assign addresses.

x. Cable the network as shown in the topology diagram. Erase the startup configuration, and reloadeach router to clear previous configurations.

y. 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 toR3, specify the bandwidth as 128 Kb/s and set a clock rate on the DCE using the **clock rate 128000** command.

You can copy and paste the following configurations into your routers to begin.

Note: Depending on the router model, interfaces might be numbered differently than those listed. Youmight need to alter them accordingly.

Router R1

interface Lo1

ip address 192.168.1.1 255.255.255.0

nterface Serial0/0/0

ip address 172.16.12.1 255.255.255.248

no shutdown

interface Serial0/0/1

ip address 172.16.13.1 255.255.255.248

no shutdown

End

```
R1(config)#int Lo1
```

```
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#int s1/0
R1(config-if)#ip address 172.16.12.1 255.255.255.248
R1(config-if)#no shutdown
```

```
R1(config-if)#int s1/1
R1(config-if)#ip address 172.16.13.1 255.255.255.248
R1(config-if)#no shutdow
*May 19 23:06:21.987: %LINEPROTO-5-UPDOWN: Line proto
R1(config-if)#no shutdown
```

Router R2

interface Lo2

ip address 192.168.2.1 255.255.255.0

interface Serial0/0/0

ip address 172.16.12.2 255.255.255.248

no shutdown

interface Serial0/0/1

ip address 172.16.23.2 255.255.255.248

no shutdown

End

```
R2(config)#int Lo2
R2(config-if)#
*May 19 23:06:13.083: %LINEPROTO-5-UPDOWN: Line proto
R2(config-if)#ip address 192.168.2.1 255.255.255.0
R2(config-if)#int s1/0
R2(config-if)#ip address 172.16.12.2 255.255.255.248
R2(config-if)#no shutdown
```

```
R2(config)#int s1/1
R2(config-if)#ip address 172.16.23.2 255.255.255.248
R2(config-if)#no shutdown
```

Router R3

interface Lo3

ip address 192.168.3.1 255.255.255.0

interface Serial0/0/1

 ip address 172.16.23.3 255.255.255.248

 no shutdown

interface Serial0/1/0

 ip address 172.16.34.3 255.255.255.248

 no shutdown

End

```
R3(config)#int Lo3
R3(config-if)#
*May 19 23:07:08.351: %LINEPROTO-5-UPDOWN: Line proto
R3(config-if)#ip address 192.168.3.1 255.255.255.0
R3(config-if)#int s1/0
R3(config-if)#ip address 172.16.13.3 255.255.255.248
R3(config-if)#no shutdown
```

```
R3(config-if)#int s1/1
R3(config-if)#ip address 172.16.23.3 255.255.255.248
R3(config-if)#no shutdown
```

```
R3(config-if)#int s1/2
R3(config-if)#ip address 172.16.34.3 255.255.255.248
R3(config-if)#no shutdown
R3(config-if)#exit
```

Router R4

interface Lo4

 ip address 192.168.4.1 255.255.255.128

interface Lo5

 ip address 192.168.4.129 255.255.255.128

interface Serial0/0/0

 ip address 172.16.34.4 255.255.255.248

 no shutdown

End

```
R4(config)#int lo4
R4(config-if)#
*May 19 23:08:16.239: %LINEPROTO-5-UPDOWN: Line protoco
R4(config-if)#ip address 192.168.4.1 255.255.255.128
R4(config-if)#interface Lo5
R4(config-if)#
*May 19 23:08:32.527: %LINEPROTO-5-UPDOWN: Line protoco
R4(config-if)#ip address 192.168.4.129 255.255.255.128
R4(config-if)#int s1/0
R4(config-if)#ip address 172.16.34.4 255.255.255.248
R4(config-if)#no shutdown
```

z. 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.

R3# **show ip interface brief**

```
R3#show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            unassigned      YES unset  administratively down down
Serial1/0                  172.16.13.3     YES manual up                    up
Serial1/1                  172.16.23.3     YES manual up                    up
Serial1/2                  172.16.34.3     YES manual up                    up
Serial1/3                  unassigned      YES unset  administratively down down
Loopback3                  192.168.3.1     YES manual up                    up
```

R3# **show protocols**

```
R3#show protocols
Global values:
  Internet Protocol routing is enabled
FastEthernet0/0 is administratively down, line protocol i
Serial1/0 is up, line protocol is up
  Internet address is 172.16.13.3/29
Serial1/1 is up, line protocol is up
  Internet address is 172.16.23.3/29
Serial1/2 is up, line protocol is up
  Internet address is 172.16.34.3/29
Serial1/3 is administratively down, line protocol is down
Loopback3 is up, line protocol is up
  Internet address is 192.168.3.1/24
```

R3# **show interfaces description**

```
R3#show interfaces description
Interface                      Status         Protocol Description
Fa0/0                          admin down     down
Se1/0                          up             up
Se1/1                          up             up
Se1/2                          up             up
Se1/3                          admin down     down
```

Step 3: Configure basic EIGRP.

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

bb. 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.

You can copy and paste the following configurations into your routers.

Router R1

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

```
R1(config)#router eigrp 1
R1(config-router)#network 192.168.1.0
R1(config-router)#network 172.16.12.0 0.0.0.7
R1(config-router)#network 172.16.13.0 0.0.0.7
R1(config-router)#no auto-summary
```

Router R2

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

```
R2(config)#router eigrp 1
R2(config-router)#network 192.168.2.0
R2(config-router)#network 172.16.12.0 0.0.0.7
R2(config-router)#network 172.16.23.0 0.0.0.7
R2(config-router)#no auto-summary
```

Router R3

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

```
R3(config)#router eigrp 1
R3(config-router)#network 192.168.3.0
R3(config-router)#network 172.16.13.0 0.0.0.7
R3(config-router)#network 172.16.23.0 0.0.0.7
R3(config-router)#network 172.16.34.0 0.0.0.7
R3(config-router)#no auto-summary
```

Router R4

router eigrp 1

network 192.168.4.0

network 172.16.34.0 0.0.0.7

no auto-summary

```
R4(config)#router eigrp 1
R4(config-router)#network 192.168.4.0
R4(config-router)#network 172.16.34.0 0.0.0.7
R4(config-router)#no auto-summary
```

You should see EIGRP neighbor relationship messages being generated.

Step 4: Verify EIGRP connectivity.

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

R1# **show ip eigrp neighbors**

```
R1#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
1   172.16.13.3             Se1/1             11 00:00:31   26    200  0  18
0   172.16.12.2             Se1/0             12 00:00:44   37    222  0  13
```

R2# **show ip eigrp neighbors**

```
R2#show ip eigrp neighbors
*May 19 23:13:42.783: %SYS-5-CONFIG_I: Configured from console by console
R2#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
1   172.16.23.3             Se1/1             11 00:00:50   41    246  0  20
0   172.16.12.1             Se1/0             11 00:01:04   30    200  0  18
```

R3# **show ip eigrp neighbors**

```
R3#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
2   172.16.34.4             Se1/2             12 00:00:44   48    288  0  6
1   172.16.23.2             Se1/1             11 00:00:58   26    200  0  19
0   172.16.13.1             Se1/0             12 00:00:58  281   1686  0  20
```

R4# **show ip eigrp neighbors**

```
R4#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
0   172.16.34.3             Se1/0             10 00:00:55   23    200  0  26
```

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

R1# **tclsh**

```
R1#tclsh
R1(tcl)#foreach address {
+>(tcl)#172.16.12.1
+>(tcl)#172.16.12.2
+>(tcl)#172.16.13.1
+>(tcl)#172.16.13.3
+>(tcl)#172.16.23.2
+>(tcl)#172.16.23.3
+>(tcl)#172.16.34.3
+>(tcl)#172.16.34.4
+>(tcl)#192.168.1.1
+>(tcl)#192.168.2.1
+>(tcl)#192.168.3.1
+>(tcl)#192.168.4.1
+>(tcl)#192.168.4.129
+>(tcl)#} { ping $address }
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.12.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 44/61/76 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/27/40 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/58/80 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/31/44 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.23.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/28/32 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.23.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/40/48 ms
Type escape sequence to abort.
```

Step 5: Verify the current path.

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

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

R1# **show ip route**

```
R1#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 le
       ia - IS-IS inter area, * - candidate default, U - per-user stati
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/29 is subnetted, 4 subnets
D       172.16.34.0 [90/2681856] via 172.16.13.3, 00:01:50, Serial1/1
D       172.16.23.0 [90/2681856] via 172.16.13.3, 00:01:50, Serial1/1
                    [90/2681856] via 172.16.12.2, 00:01:50, Serial1/0
C       172.16.12.0 is directly connected, Serial1/0
C       172.16.13.0 is directly connected, Serial1/1
D    192.168.4.0/24 [90/2809856] via 172.16.13.3, 00:01:38, Serial1/1
C    192.168.1.0/24 is directly connected, Loopback1
D    192.168.2.0/24 [90/2297856] via 172.16.12.2, 00:01:50, Serial1/0
D    192.168.3.0/24 [90/2297856] via 172.16.13.3, 00:01:50, Serial1/1
```

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

  1 172.16.34.3 36 msec 32 msec 32 msec
  2 172.16.13.1 28 msec 56 msec 84 msec
```

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

  1 172.16.34.3 44 msec 28 msec 28 msec
  2 172.16.13.1 64 msec 28 msec 64 msec
```

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 S0/0/1.

R3# **show ip route**

```
R3#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 leve
       ia - IS-IS inter area, * - candidate default, U - per-user static
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

```
     172.16.0.0/29 is subnetted, 4 subnets
C       172.16.34.0 is directly connected, Serial1/2
C       172.16.23.0 is directly connected, Serial1/1
D       172.16.12.0 [90/2681856] via 172.16.23.2, 00:02:29, Serial1/1
                    [90/2681856] via 172.16.13.1, 00:02:29, Serial1/0
C       172.16.13.0 is directly connected, Serial1/0
D    192.168.4.0/24 [90/2297856] via 172.16.34.4, 00:02:17, Serial1/2
D    192.168.1.0/24 [90/2297856] via 172.16.13.1, 00:02:29, Serial1/0
D    192.168.2.0/24 [90/2297856] via 172.16.23.2, 00:02:29, Serial1/1
C    192.168.3.0/24 is directly connected, Loopback3
```

R3#

ff. On R3, use the **show interfaces serial 0/0/0** and **show interfaces s0/0/1** commands.

R3# **show interfaces serial0/0/0**

```
R3#show int s1/0
Serial1/0 is up, line protocol is up
  Hardware is M4T
  Internet address is 172.16.13.3/29
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
```

```
  Routing Descriptor Blocks:
  172.16.13.1 (Serial1/0), from 172.16.13.1, Send flag is 0x0
      Composite metric is (2297856/128256), Route is Internal
      Vector metric:
        Minimum bandwidth is 1544 Kbit
        Total delay is 25000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
  172.16.23.2 (Serial1/1), from 172.16.23.2, Send flag is 0x0
```

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

R3# **show ip eigrp topology 192.168.1.0**

```
R3#show ip eigrp topology 192.168.1.0
IP-EIGRP (AS 1): Topology entry for 192.168.1.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is
  Routing Descriptor Blocks:
  172.16.13.1 (Serial1/0), from 172.16.13.1, Send flag is 0x0
      Composite metric is (2297856/128256), Route is Internal
      Vector metric:
        Minimum bandwidth is 1544 Kbit
        Total delay is 25000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 1
  172.16.23.2 (Serial1/1), from 172.16.23.2, Send flag is 0x0
```

Step 6: Configure PBR to provide path control.

The steps required to implement path control include the following:

– Choose the path control tool to use. Path control tools manipulate or bypass the IP routing table. For PBR, **route-map** commands are used.
– Implement the traffic-matching configuration, specifying which traffic will be manipulated. The **match** commands are used within route maps.
– Define the action for the matched traffic using **set** commands within route maps.
– Apply the route map to incoming traffic.

As a test, you will configure the following policy on router R3:

– All traffic sourced from R4 LAN A must take the R3 --> R2 --> R1 path.
– All traffic sourced from R4 LAN B must take the R3 --> R1 path.

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

R3(config)# **ip access-list standard PBR-ACL**

R3(config-std-nacl)# **remark ACL matches R4 LAN B traffic**

R3(config-std-nacl)# **permit 192.168.4.128 0.0.0.127** R3(config-

std-nacl)# **exit**

```
R3(config)#ip access-list standard PBR-ACL
R3(config-std-nacl)#remark ACL matches R4 LAN B traffic
R3(config-std-nacl)#permit 192.168.4.128 0.0.0.127
R3(config-std-nacl)#exit
```

R3(config)#

ii. Create a route map called **R3-to-R1** that matches PBR-ACL and sets the next-hop interface to the R1 serial 0/0/1 interface.

R3(config)# **route-map R3-to-R1 permit**

R3(config-route-map)# **description RM to forward LAN B traffic to R1**

R3(config-route-map)# **match ip address PBR-ACL**

R3(config-route-map)# **set ip next-hop 172.16.13.1**

R3(config-route-map)# **exit**

```
R3(config)#route-map R3-to-R1 permit
R3(config-route-map)#match ip address PBR-ACL
R3(config-route-map)#set ip next-hop 172.16.13.1
R3(config-route-map)#exit
```

jj. Apply the R3-to-R1 route map to the serial interface on R3 that receives the traffic from R4. Use the **ip policy route-map** command on interface S0/1/0.

R3(config)# **interface s0/1/0**

R3(config-if)# **ip policy route-map R3-to-R1**

R3(config-if)# **end**

```
R3(config)#int s1/2
R3(config-if)#ip policy route-map R3-to-R1
R3(config-if)#end
```

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

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.

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

R3# **conf t**

Enter configuration commands, one per line. End with CNTL/Z.R3(config)#

access-list 1 permit 192.168.4.0 0.0.0.255

R3(config)# **exit**

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#access-list 1 permit 192.168.4.0 0.0.0.255
R3(config)#exit
```

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

R3# **debug ip policy ?**

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

R3# **debug ip policy 1**

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

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

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

  1 172.16.34.3 40 msec 12 msec 32 msec
  2 172.16.13.1 60 msec 48 msec 88 msec
```

```
R3#
*May 19 23:17:36.819: IP: s=192.168.4.1 (Serial1/2), d=192.168.1.1,
*May 19 23:17:36.851: IP: s=192.168.4.1 (Serial1/2), d=192.168.1.1,
*May 19 23:17:36.879: IP: s=192.168.4.1 (Serial1/2), d=192.168.1.1,
*May 19 23:17:36.915: IP: s=192.168.4.1 (Serial1/2), d=192.168.1.1,
g
*May 19 23:17:36.971: IP: s=192.168.4.1 (Serial1/2), d=192.168.1.1,
g
*May 19 23:17:37.031: IP: s=192.168.4.1 (Serial1/2), d=192.168.1.1
R3#, len 28, FIB policy rejected(no match) - normal forwarding
R3#
```

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

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

  1 172.16.34.3 40 msec 28 msec 32 msec
  2 172.16.13.1 60 msec 64 msec 32 msec
```

```
R3#.168.4.129 (Serial1/2), d=192.168.1.1, len 28, FIB policy match
*May 19 23:17:55.763: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
*May 19 23:17:55.763: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
*May 19 23:17:55.823: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
*May 19 23:17:55.823: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
*May 19 23:17:55.827: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
*May 19 23:17:55.883: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
*May 19 23:17:55.883: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
*May 19 23:17:55.887: IP: s=192.168.4.129 (Serial1/2), d=192.168.1.1,
```

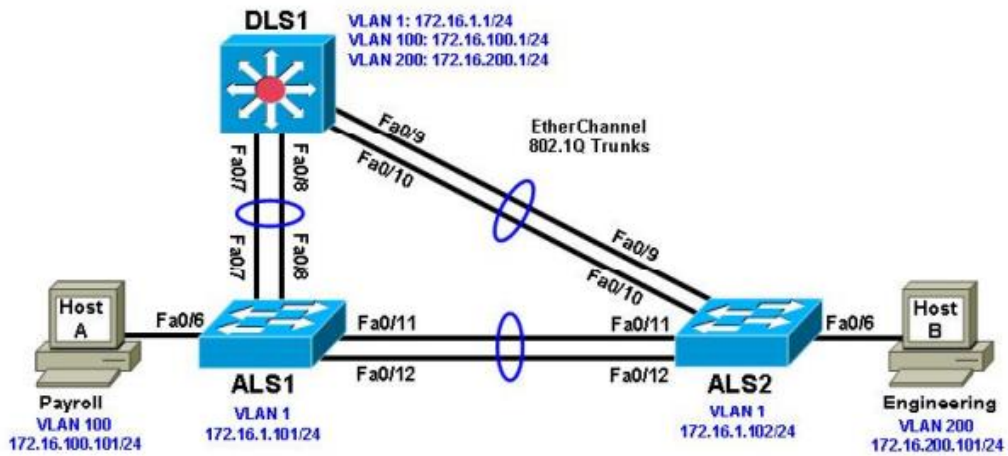
pp. On R3, display the policy and matches using the **show route-map** command.

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: 6 packets, 192 bytes
```

Practical no - 6

IP Service Level Agreements and Remote SPAN in a Campus Environment



Part 1: Prepare for the Lab

Step 1: Prepare the switches for the lab

Use the **reset.tcl** script you created in Lab 1 “Preparing the Switch” to set your switches up for this lab. Then load the file BASE.CFG into the running-config with the command **copy flash:BASE.CFG running-config**. An example from DLS1:

DLS1# tclsh reset.tcl

Erasing the nvram filesystem will remove all configuration files! Continue? [confirm]

[OK]

Erase of nvram: complete

Reloading the switch in 1 minute, type reload cancel to halt

Proceed with reload? [confirm]

*Mar 7 18:41:40.403: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram

*Mar 7 18:41:41.141: %SYS-5-RELOAD: Reload requested by console. Reload Reason:

Reload command.

<switch reloads - output omitted>

Would you like to enter the initial configuration dialog? [yes/no]: n

Switch> en

*Mar 1 00:01:30.915: %LINK-5-CHANGED: Interface Vlan1, changed state to

administratively down

Switch# copy BASE.CFG running-config

Destination filename [running-config]?

184 bytes copied in 0.310 secs (594 bytes/sec)

DLS1#

Step 2: Configure basic switch parameters.

Configure an IP address on the management VLAN according to the diagram. VLAN 1 is the default management VLAN, but following best practice, we will use a different VLAN. In this case, VLAN 99.

Enter basic configuration commands on each switch according to the diagram.

DLS1 example:

DLS1# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

DLS1(config)# interface vlan 99

DLS1(config-if)# ip address 172.16.99.1 255.255.255.0

DLS1(config-if)# no shutdown

The interface VLAN 99 will not come up immediately, because the broadcast domain it is associated with (VLAN 99) doesn’t exist on the switch. We will fix that in a few moments.

(Optional) On each switch, create an enable secret password and configure the VTY lines to allow remote access from other network devices.

DLS1 example:

DLS1(config)# enable secret class

DLS1(config)# line vty 0 15

DLS1(config-line)# password cisco

DLS1(config-line)# login

Note: The passwords configured here are required for NETLAB compatibility only and are NOT recommended for use in a live environment.
Note(2): For purely lab environment purposes, it is possible to configure the VTY lines so that they accept any Telnet connection immediately, without asking for a password, and place the user into the privileged EXEC mode directly. The configuration would be similar to the following example for DLS1:

DLS1(config)# enable secret class

DLS1(config)# line vty 0 15

DLS1(config-line)# no login

DLS1(config-line)# privilege level 15

Note: The %PKI-6-AUTOSAVE message tells you that your BASE.CFG has been saved as the startup-config, so a simple reload will revert the switch back to BASE configuration
a. Configure default gateways on ALS1 and ALS2. These are access layer switches operating as Layer 2 devices and need a default gateway to send traffic from their management interface to other networks. Configure both ALS1 and ALS2. An example from ALS1 is shown:

ALS1(config)# ip default-gateway 172.16.99.1

Step 3: Configure host PCs.

Configure PCs Host A and Host B with the IP address and subnet mask shown in the topology. Host A is in VLAN 100 with a default gateway of 172.16.100.1. Host B is in VLAN 200 with a default gateway of 172.16.200.1.

Step 4: Configure trunks and EtherChannels between switches.

Configure trunking according to the diagram. LACP is used for EtherChannel negotiation for these trunks. Examples from DLS1 and ALS1 are shown. Configure all the switches with the channel groups shown in the topology:

Configure the trunks and EtherChannel from DLS1 to ALS1 and ALS2.

DLS1(config)# vlan 666

DLS1(config-vlan)# name NATIVE_DO_NOT_USE

DLS1(config-vlan)# exit

DLS1(config)# int ran f0/7-10

DLS1(config-if-range)# switchport trunk encapsulation dot1q

DLS1(config-if-range)# switchport trunk native vlan 666

DLS1(config-if-range)# switchport nonegotiate

DLS1(config-if-range)# switchport mode trunk

DLS1(config-if-range)# exit

DLS1(config)# int ran f0/7-8

DLS1(config-if-range)# channel-group 1 mode active

DLS1(config-if-range)# description EtherChannel to ALS1

DLS1(config-if-range)# no shut

DLS1(config-if-range)# exit

DLS1(config)# int ran f0/9-10

DLS1(config-if-range)# channel-group 2 mode active

DLS1(config-if-range)# description EtherChannel to ALS2

DLS1(config-if-range)# no shut

DLS1(config-if-range)# exit

Configure the trunks and EtherChannel between ALS1 and ALS2.

ALS1(config)# interface range fastEthernet 0/11 - 12

ALS1(config-if-range)# switchport mode trunk

ALS1(config-if-range)# channel-group 3 mode active

ALS1(config-if-range)# no shut

Step 5: Configure VTP on ALS1 and ALS2.

Change the VTP mode of ALS1 and ALS2 to client.

ALS1(config)# vtp mode client

Setting device to VTP CLIENT mode.

ALS2(config)# vtp mode client

Setting device to VTP CLIENT mode.

Step 6: Configure VTP on DLS1.

Create the VTP domain on DLS1, and create VLANs 100 and 200 for the domain.

DLS1(config)# vtp domain SWPOD

DLS1(config)# vtp version 2

DLS1(config)# vlan 99

DLS1(config-vlan)# name Management

DLS1(config-vlan)# vlan 100

DLS1(config-vlan)# name Finance

DLS1(config-vlan)# vlan 200

DLS1(config-vlan)# name Engineering

DLS1(config-vlan)# exit

DLS1(config)#

Step 7: Configure access ports.

Configure the host ports for the appropriate VLANs according to the diagram.

ALS1(config)# interface fastEthernet 0/6

ALS1(config-if)# switchport mode access

ALS1(config-if)# switchport access vlan 100

ALS1(config-if)# no shut

ALS2(config)# interface fastEthernet 0/6

ALS2(config-if)# switchport mode access

ALS2(config-if)# switchport access vlan 200

ALS1(config-if)# no shut

Step 8: Configure VLAN interfaces and enable routing.

On DLS1, create the SVIs for VLANs 100 and 200. Note that the corresponding Layer 2 VLANs must be configured for the Layer 3 SVIs to activate. This was done in Step 6.

DLS1(config)# interface vlan 100

DLS1(config-if)# ip address 172.16.100.1 255.255.255.0

DLS1(config-if)# interface vlan 200

DLS1(config-if)# ip address 172.16.200.1 255.255.255.0

The **ip routing** command is also needed to allow the DLS1 switch to act as a Layer 3 device to route between these VLANs. Because the VLANs are all considered directly connected, a routing protocol is not needed at this time. The default configuration on 3560 switches is **no ip routing**.

DLS1(config)# ip routing

Verify the configuration using the **show ip route** command on DLS1.

DLS1# 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 172.16.99.0/24 is directly connected, Vlan99

L 172.16.99.1/32 is directly connected, Vlan99

C 172.16.100.0/24 is directly connected, Vlan100

L 172.16.100.1/32 is directly connected, Vlan100

C 172.16.200.0/24 is directly connected, Vlan200

L 172.16.200.1/32 is directly connected, Vlan200

DLS1#

Run the following Tcl script on DLS1 to verify full connectivity. If these pings are not successful, troubleshoot.

DLS1# tclsh

foreach address {

172.16.99.1

172.16.99.101

172.16.99.102

172.16.100.1

172.16.200.1

172.16.100.101

172.16.200.101

} {

ping $address }

Part 2: Configure Cisco IOS IP SLA

Step 1: Configure Cisco IOS IP SLA responders.

IP SLA responders are Cisco IOS devices that support the IP SLA control protocol. An IP SLA responder uses the Cisco IOS IP SLA Control Protocol for notification configuration and on which port to listen and respond. Some operations, such as ICMP echo, do not require a dedicated IP SLA responder.

Use the **ip sla responder** command on ALS1 and ALS2 to enable sending and receiving IP SLAs control packets.
Note: This command replaces the ip sla monitor responder command. All commands that used to begin with "ip sla monitor" now begin with "ip sla" (without "monitor"). Configure this on both ALS1 and ALS2. An example from ALS1:

ALS1(config)# ip sla responder

Configure ALS1 and ALS2 as IP SLA responders for UDP jitter using the **ip sla responder udp-echo ipaddress** command. Specify the IP address of DLS1 VLAN 1 to act as the destination IP address for the reflected UDP traffic on both ALS1 and ALS2. Configure this on both ALS1 and ALS2. An example from ALS1:

ALS1(config)# ip sla responder udp-echo ipaddress 172.16.99.1 port 5000

Step 2: Configure the Cisco IOS IP SLA source to measure network performance.

IP SLA uses generated traffic to measure network performance between two networking devices.

On DLS1, create an IP SLA operation and enter IP SLA configuration mode with the **ip sla operation-number** command.

DLS1(config)# ip sla 1

DLS1(config-ip-sla)#

Configure an IP SLA ICMP echo operation using the icmp-echo command in IP SLA configuration mode. The IP SLA ICMP echo operation does not require a dedicated Cisco IOS IP SLA responder (the destination device can be a non-Cisco device, such as a PC). By default, the ICMP operation repeats every 60 seconds. On DLS1, for ICMP echo operation 1, specify the IP address of Host A as the target. For ICMP echo operation 2, specify the IP address of Host B as the target.

DLS1(config-ip-sla)# icmp-echo 172.16.100.101

DLS1(config-ip-sla-echo)# exit

DLS1(config)# ip sla 2

DLS1(config-ip-sla)# icmp-echo 172.16.200.101

DLS1(config-ip-sla-echo)# exit

Jitter means inter-packet delay variance. UDP-based voice traffic associated with IP phone and PC softphone applications at the access layer require strict adherence to delay and jitter thresholds. To configure an IP SLA UDP jitter operation, use the udp-jitter command in IP SLA configuration mode. By default, the UDP jitter operation repeats every 60 seconds. For UDP jitter operation 3, specify the destination IP address of the ALS1 VLAN 99 interface as the target. For operation 4, specify the destination IP address of the ALS2 VLAN 99 interface as the target. The IP SLA communication port is 5000 for both operations.

DLS1(config)# ip sla 3

DLS1(config-ip-sla)# udp-jitter 172.16.99.101 5000

DLS1(config-ip-sla-jitter)# exit

DLS1(config)# ip sla 4

DLS1(config-ip-sla)# udp-jitter 172.16.99.102 5000

DLS1(config-ip-sla-jitter)# exit

Schedule the IP SLAs operations to run indefinitely beginning immediately using the ip sla schedule global configuration mode command.

DLS1(config)# ip sla schedule 1 life forever start-time now

DLS1(config)# ip sla schedule 2 life forever start-time now

DLS1(config)# ip sla schedule 3 life forever start-time now

DLS1(config)# ip sla schedule 4 life forever start-time now

Step 3: Monitor IP SLAs operations.

View the IP SLA configuration for IP SLA 1 on DLS1. The output for IP SLA 2 is similar.

DLS1# show ip sla configuration 1

IP SLAs Infrastructure Engine-III

Entry number: 1

Owner:

Tag:

Operation timeout (milliseconds): 5000

Type of operation to perform: icmp-echo

Target address/Source address: 172.16.100.101/0.0.0.0

Type Of Service parameter: 0x0

Request size (ARR data portion): 28

Verify data: No

Vrf Name:

Schedule:

Operation frequency (seconds): 60 (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

Number of statistic distribution buckets kept: 1

Statistic distribution interval (milliseconds): 20

Enhanced History:

History Statistics:

Number of history Lives kept: 0

Number of history Buckets kept: 15

History Filter Type: None

What type of operation is being performed with IP SLA 1?
____

View the IP SLA configuration for IP SLA 3 on DLS1. The output for IP SLA 4 is similar.

DLS1# show ip sla configuration 3

IP SLAs Infrastructure Engine-III

Entry number: 2

Owner:

Tag:

Operation timeout (milliseconds): 5000

Type of operation to perform: icmp-echo

Target address/Source address: 172.16.200.101/0.0.0.0

Type Of Service parameter: 0x0

Request size (ARR data portion): 28

Verify data: No

Vrf Name:

Schedule:

Operation frequency (seconds): 60 (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

Number of statistic distribution buckets kept: 1

Statistic distribution interval (milliseconds): 20

Enhanced History:

History Statistics:

Number of history Lives kept: 0

Number of history Buckets kept: 15

History Filter Type: None

What type of operation is being performed with IP SLA 3?

Display global information about Cisco IOS IP SLAs on DLS1.

DLS1# show ip sla application

IP Service Level Agreements

Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:

icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http

dns, udpJitter, dhcp, ftp, video, udpApp, wspApp

Supported Features:

IPSLAs Event Publisher

IP SLAs low memory water mark: 9359471

Estimated system max number of entries: 6855

Estimated number of configurable operations: 6817

Number of Entries configured : 4

Number of active Entries : 4

Number of pending Entries : 0

Number of inactive Entries : 0

Time of last change in whole IP SLAs: 13:54:00.025 CDT Fri Jul 31 2015

Display information about Cisco IOS IP SLA responders on ALS1. The ALS2 output is similar.

ALS1# show ip sla responder

General IP SLA Responder on Control port 1967

General IP SLA Responder is: Enabled

Number of control message received: 26 Number of errors: 0

Recent sources:

172.16.99.1 [14:17:28.775 CDT Fri Jul 31 2015]

172.16.99.1 [14:16:28.780 CDT Fri Jul 31 2015]

172.16.99.1 [14:15:28.776 CDT Fri Jul 31 2015]

172.16.99.1 [14:14:28.781 CDT Fri Jul 31 2015]

172.16.99.1 [14:13:28.777 CDT Fri Jul 31 2015]

Recent error sources:

Permanent Port IP SLA Responder

Permanent Port IP SLA Responder is: Enabled

udpEcho Responder:

IP Address Port

172.16.99.1 5000

Display IP SLA statistics on DLS1 for IP SLA 1. The IP SLA 2 output is similar.

DLS1# show ip sla statistics 1

IPSLAs Latest Operation Statistics

IPSLA operation id: 1

Latest RTT: 1 milliseconds

Latest operation start time: 14:17:00 CDT Fri Jul 31 2015

Latest operation return code: OK

Number of successes: 26

Number of failures: 0

Operation time to live: Forever

From this output, you can see that the latest round-trip time (RTT) for SLA operation Index 1 (icmp-echo) is 1 millisecond (ms). The number of packets sent successfully from DLS1 to PC Host A was 26, and there were no failures.

Display IP SLA statistics on DLS1 for IP SLA 3. The IP SLA 4 output is similar.

DLS1# show ip sla statistics 3

IPSLAs Latest Operation Statistics

IPSLA operation id: 3

Type of operation: udp-jitter

Latest RTT: 3 milliseconds

Latest operation start time: 14:18:01 CDT Fri Jul 31 2015

Latest operation return code: OK

RTT Values:

Number Of RTT: 10 RTT Min/Avg/Max: 3/3/5 milliseconds

Latency one-way time:

Number of Latency one-way Samples: 0

Source to Destination Latency one way Min/Avg/Max: 0/0/0 milliseconds

Destination to Source Latency one way Min/Avg/Max: 0/0/0 milliseconds

Jitter Time:

Number of SD Jitter Samples: 9

Number of DS Jitter Samples: 9

Source to Destination Jitter Min/Avg/Max: 0/1/1 milliseconds

Destination to Source Jitter Min/Avg/Max: 0/1/1 milliseconds

Packet Loss Values:

Loss Source to Destination: 0

Source to Destination Loss Periods Number: 0

Source to Destination Loss Period Length Min/Max: 0/0

Source to Destination Inter Loss Period Length Min/Max: 0/0

Loss Destination to Source: 0

Destination to Source Loss Periods Number: 0

Destination to Source Loss Period Length Min/Max: 0/0

Destination to Source Inter Loss Period Length Min/Max: 0/0

Out Of Sequence: 0 Tail Drop: 0

Packet Late Arrival: 0 Packet Skipped: 0

Voice Score Values:

Calculated Planning Impairment Factor (ICPIF): 0

Mean Opinion Score (MOS): 0

Number of successes: 27

Number of failures: 0

Operation time to live: Forever

From this output, you can see that the latest RTT for SLA operation Index 3 (udp-jitter) is 3 ms. Jitter time from source to destination and from destination to source is averaging 1 ms, which is acceptable for voice applications. The number of packets sent successfully from DLS1 to ALS1 was 27, and there were no failures.

Disable interface VLAN 99 on ALS1 using the **shutdown** command.

ALS1(config)# interface vlan 99

ALS1(config-if)# shutdown

Allow a few minutes to pass and then issue the **show ip sla statistics 3** command on DLS1. The output should look similar to the following.

DLS1# show ip sla statistics 3

IPSLAs Latest Operation Statistics

IPSLA operation id: 3

Type of operation: udp-jitter

Latest RTT: NoConnection/Busy/Timeout

Latest operation start time: 14:22:01 CDT Fri Jul 31 2015

Latest operation return code: No connection

RTT Values:

Number Of RTT: 0 RTT Min/Avg/Max: 0/0/0 milliseconds

Latency one-way time:

Number of Latency one-way Samples: 0

Source to Destination Latency one way Min/Avg/Max: 0/0/0 milliseconds

Destination to Source Latency one way Min/Avg/Max: 0/0/0 milliseconds

Jitter Time:

Number of SD Jitter Samples: 0

Number of DS Jitter Samples: 0

Source to Destination Jitter Min/Avg/Max: 0/0/0 milliseconds

Destination to Source Jitter Min/Avg/Max: 0/0/0 milliseconds

Packet Loss Values:

Loss Source to Destination: 0

Source to Destination Loss Periods Number: 0

Source to Destination Loss Period Length Min/Max: 0/0

Source to Destination Inter Loss Period Length Min/Max: 0/0

Loss Destination to Source: 0

Destination to Source Loss Periods Number: 0

Destination to Source Loss Period Length Min/Max: 0/0

Destination to Source Inter Loss Period Length Min/Max: 0/0

Out Of Sequence: 0 Tail Drop: 0

Packet Late Arrival: 0 Packet Skipped: 0

Voice Score Values:

Calculated Planning Impairment Factor (ICPIF): 0

Mean Opinion Score (MOS): 0

Number of successes: 29

Number of failures: 2

Operation time to live: Forever

If there is a connectivity problem between IP SLA source DLS1 and responder ALS1 or ALS2, the communication to the responder will be lost and statistics will cease to be collected, except for the number of failed tests.

Note: The IP SLA itself is an additional task that must be performed by the switch CPU. A large number of intensive SLAs could create a significant burden on the CPU, possibly interfering with other switch functions and having detrimental impact on the overall device performance. Therefore, you should carefully evaluate the benefits of running IP SLAs. The CPU load should be monitored after the SLAs are deployed to verify that they do not stress the device's CPU above safe limits. Re-enable ALS1's interface vlan 99 before continuing.

Part 3: Switch Port Analyzer (SPAN) Feature

SPAN is tool that allows for monitoring and troubleshooting a network. There are different variations of the SPAN tool. There is local SPAN, Remote Span, and VLAN span. Local Span allows an administrator to monitor traffic from a source and have it sent to a destination port on the same switch running a protocol analyzer on the same switch. The source and destination port used to create the monitor session must be on the same switch. Remote SPAN allows the source and destination ports to be on different switches. In order for this to work, it uses a VLAN configured only for remote span functionality. The source port then places the transmitted or received data onto the remote span VLAN. The remote span VLAN is carried across trunks. The receiving switch takes the data sourced from the remote VLAN and sends it to the destination port running the protocol analyzer.

In this lab, we will demonstrate the use of remote SPAN (RSPAN). VLAN 300 will be created and used as the remote span VLAN. We will set up a monitoring session for the host connected to port fa0/6 on switch ALS1. Ultimately, the destination port will be the host connected to fa0/6 of ALS2. The ALS2 host is collect the transmit and receive data using Wireshark.

Step 1: Configure Remote SPAN (RSPAN).

Create the RSPAN VLAN on DLS1 using the VLAN 300 command from global configuration mode.

DLS1(config)# vlan 300

DLS1(config-vlan)# name REMOTE_SPAN

DLS1(config-vlan)# remote-span

Use the **show vlan remote-span** command to verify the vlan 300 is configured correctly and is designated as the remote-span vlan. Ensure that the VLAN propagates across the VTP Domain with show vlan brief command. Use the **show interface trunk** command to ensure the RSPAN VLAN is allowed on the trunks. The RSPAN VLAN should not be a DATA VLAN. Its purpose is strictly for carrying the monitored traffic across trunk links from one switch to another.
Verify the output on DLS1.

DLS1# show vlan brief | include active

1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4

99 Management active

100 Finance active

200 Engineering active

300 REMOTE_SPAN active

666 NATIVE_DO_NOT_USE active

DLS1#

Verify the output on ALS1.

ALS1# show vlan brief | include active

1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4

99 Management active

100 Finance active Fa0/6

200 Engineering active

300 REMOTE_SPAN active

666 NATIVE_DO_NOT_USE active

ALS1#

Now configure the monitor session on ALS1 with a source interface of fa0/6 and a destination of remote vlan 300. Because the captured traffic must traverse the local switch to a remote switch, we must use the remote VLAN as the destination.

ALS1(config)# monitor session 1 source interface Fa0/6

ALS1(config)# monitor session 1 destination remote vlan 300

Verify the configuration using the show monitor command

ALS1# show monitor

Session 1

Type : Remote Source Session

Source Ports :

Both : Fa0/6

Dest RSPAN VLAN : 300

Move to the ALS2 switch and configure it to collect the desired traffic. The source port on ALS2 will be the remote span vlan 300 and the destination port will be the Engineering client connected to port fa0/6.

It is important to note that the PC-B host should be running a protocol analyzer to view the contents of the captured traffic and perform traffic analysis. Both transmit and receive traffic of the source port will be captured. The configuration can be modified to only capture transmit or receive traffic if necessary.

Configure ALS2 for the remote span session.

ALS2(config)# monitor session 10 source remote vlan 300

ALS2(config)# monitor session 10 destination interface Fa0/6

Our configuration shows the use of a different session number than the one used on ALS1. The session numbers do not have to match from device to device.

Verify the configuration using the show monitor command. The source port should show VLAN 300 and the destination port should be interface fa0/6.

ALS2# show monitor

Session 10

Type : Remote Destination Session

Source RSPAN VLAN : 300

Destination Ports : Fa0/6

Encapsulation : Native

Ingress : Disabled

Use the **show interfaces fa0/6** to command to view the status of the interface. Notice from the output the line protocol is down. When a port is used as a destination in monitoring session, it cannot be used to transmit and receive regular network traffic.

ALS2# show interface f0/6

FastEthernet0/6 is up, line protocol is down (monitoring)

Hardware is Fast Ethernet, address is 5017.ff84.0a86 (bia 5017.ff84.0a86)

<output omitted>

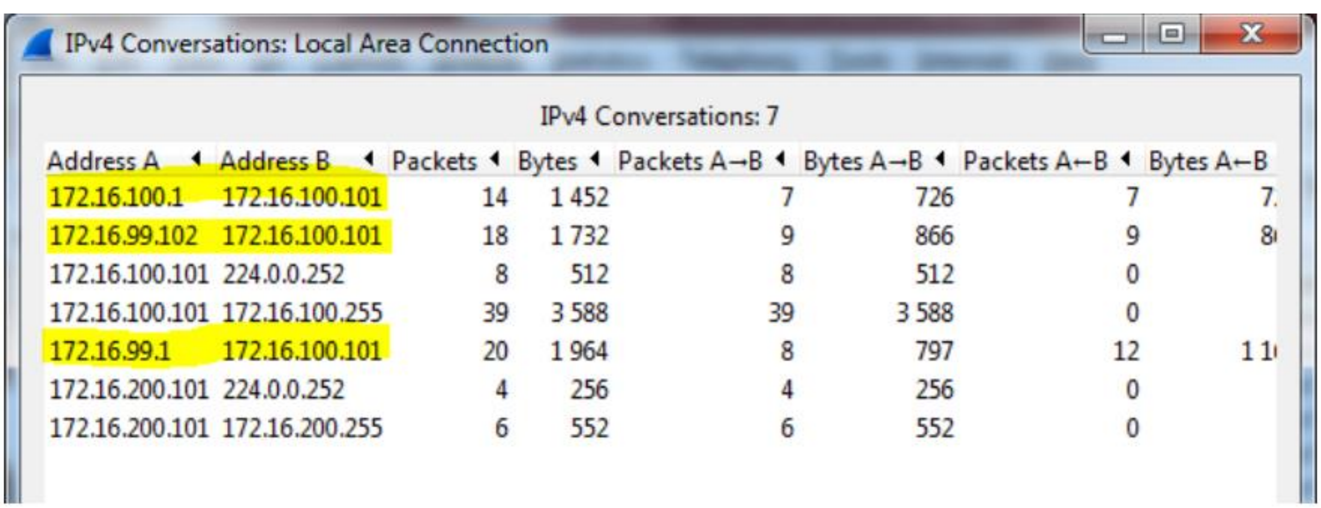
Step 2: Test RSPAN operation

On PC-B, turn on Wireshark and capture all interface traffic.

In order to test the RSPAN configuration implemented on ALS1 and ALS2, we need to generate traffic from the source host, PC-A.

o Initiate a ping from PC-A to the 172.16.99.102 address
o Open a web browser. Browse to the following url: http://172.16.99.1
o From ALS2, initiate a ping to PC-A, 172.16.100.101.
o From DLS1, initiate a ping to PC-A, 172.16.100.101.

In the Wireshark application that is running on PC-B, select the STOP button then use the Statistics > Converstion List > IPv4 menu to view the IPv4 conversations contained in the capture. You will see that 172.16.200.101 (the address of PC-B) is not involved in any conversations except for traffic to 224.0.0.252 and 172.16.200.255.



Address A	Address B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B
172.16.100.1	172.16.100.101	14	1 452	7	726	7	7.
172.16.99.102	172.16.100.101	18	1 732	9	866	9	8
172.16.100.101	224.0.0.252	8	512	8	512	0	
172.16.100.101	172.16.100.255	39	3 588	39	3 588	0	
172.16.99.1	172.16.100.101	20	1 964	8	797	12	1 1
172.16.200.101	224.0.0.252	4	256	4	256	0	
172.16.200.101	172.16.200.255	6	552	6	552	0	

Step 3: End of Lab

Do not save your configurations. The equipment will be reset for the next lab.

Practical no - 7

Inter-VLAN Routing

Normally, Routers are used to divide broadcast domain and switches (at layer 2) Operates in a single broadcast domain but Switches can also divide broadcast domain by using the concept of **VLAN (Virtual LAN)**.

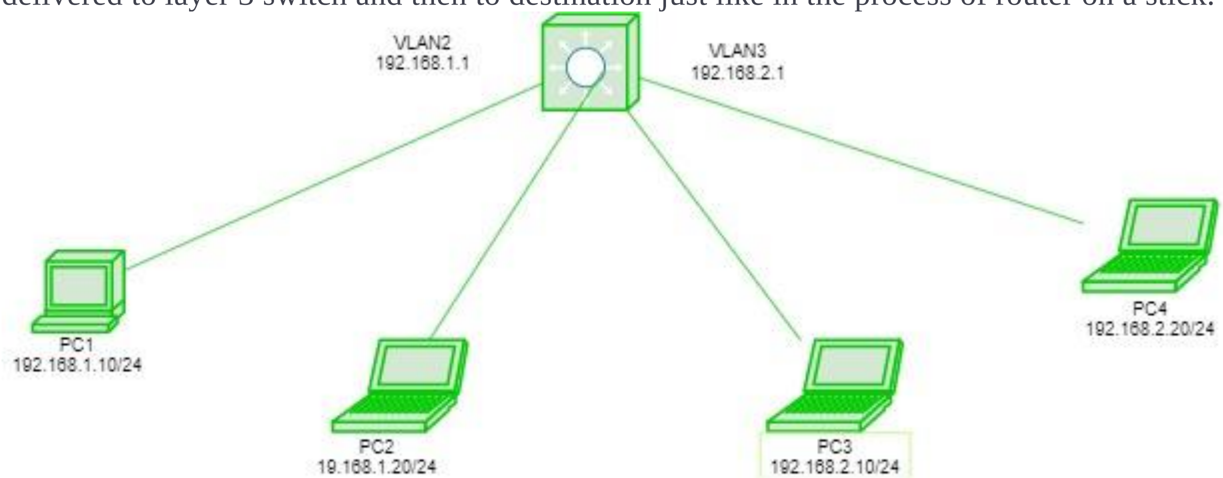
Vlan is logical grouping of devices in same or different broadcast domain. By default, all the switch ports are in Vlan 1. As the single broadcast domain is divided into multiple broadcast domains, Routers or layer 3 switches are used for intercommunication between the different Vlans.The process of intercommunication of the different Vlans is known as Inter Vlan Routing (IVR). Suppose we have made 2 logical group of devices (vlan) named sales and finance.If a device in sales department wants to communicate with a device in finance department, inter Vlan routing has to be performed. These can be performed by either router or layer 3 switches.

Switch Virtual Interface (SVI) –

SVI is a logical interface on a multilayer switch which provides layer 3 processing for packets to all switch ports associated with that VLAN.A single SVI can be created for a Vlan. SVI for layer 3 switch provides both management and routing services while SVI on layer 2 switch provides only management services like creating vlans or telnet/SSH services.

Process of Inter Vlan Routing by Layer 3 Switch –

The SVI created for the respective Vlan acts a default gateway for that Vlan just like the sub-interface of the router (in the process of Router On a stick). If the packet is to be delivered to different vlan i.e inter Vlan Routing is to be performed on layer 3 switch then first the packet is delivered to layer 3 switch and then to destination just like in the process of router on a stick.



Here is a topology in which we have a layer 3 switch connected to host devices namely PC1, PC2, PC3, PC4. The hosts PC1, PC2 will be in Vlan 10 and PC3, PC4 will be in Vlan 20. Giving IP address to All hosts. PC1-192.168.1.10/24, PC2-192.168.1.20/24, PC3 – 192.168.2.10/24, PC4-192.168.2.20/24.

Now creating vlans on layer 3 switch namely vlan 2 on the switch ports fa0/1, 2 and fa0/3, 4 for vlan 3.

Switch# vlan 2
Switch# vlan 3
Switch# int range fa0/1-2

Switch# switchport access vlan 2
Switch# int range fa0/3-4
Switch# switchport access vlan 3
Now creating SVI for vlan 2 giving it IP address 192.168.1.1/24 and SVI for vlan 3 giving IP address 192.168.2.1/24

Switch# ip routing
Switch# int vlan 2
Switch# ip address 192.168.1.1 255.255.255.0
Switch# int vlan 3
Switch# ip address 192.168.2.1 255.255.255.0
Now if we will try to ping PC1 to PC4.

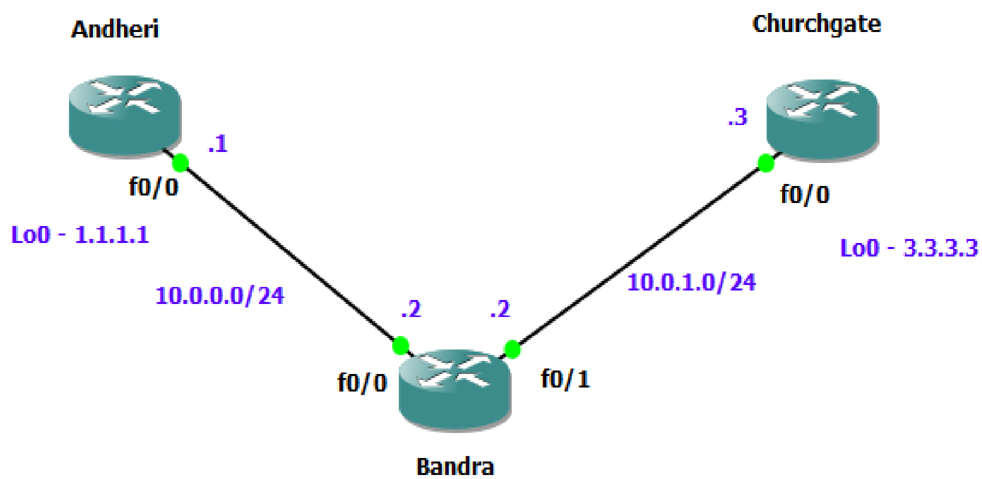
The packet is first delivered to switch then to the destination. As the destination is present in other networks, the packet will be first delivered to switch which has a SVI for both vlans (acts as gateway).

Advantages –
In **Router on a stick** method, both switch and router are needed but while using layer 3 switch, a single switch will perform inter-vlan routing as well as the layer 2 functions (Vlan), therefore this method is cost effective and also less configuration is needed.

Practical No - 8

Aim: Simulating MPLS environment

Topology :



Step 1 – IP addressing of MPLS Core and OSPF

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(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)#
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
```

```
Bandra(config)#int lo0
Bandra(config-if)#
Bandra(config-if)#ip add 2.2.2.2 255.255.255.255
Bandra(config-if)#ip ospf 1 are 0
Bandra(config-if)#
Bandra(config-if)#int f0/0
Bandra(config-if)#ip add 10.0.0.2 255.255.255.0
Bandra(config-if)#no shut
Bandra(config-if)#ip ospf 1 area 0
Bandra(config-if)#
Bandra(config-if)#int f0/1
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(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 f0/0
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 wecan ping between the loopbacks of R1 and R3

```
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 = 20/52/64 ms
```

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.

- At each interface enter the mpls ip command
- Under the ospf process use the mpls ldp autoconfig command

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

```
Bandra(config)#router ospf 1
Bandra(config-router)#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.

```
Bandra#
*May 29 17:03:09.559: %SYS-5-CONFIG_I: Configured from console by console
Bandra#
*May 29 17:03:28.631: %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 int
Interface              IP            Tunnel   BGP Static Operational
FastEthernet0/0        Yes (ldp)     No       No  No     Yes
FastEthernet0/1        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
    Peer LDP Ident: 1.1.1.1:0; Local LDP Ident 2.2.2.2:0
        TCP connection: 1.1.1.1.646 - 2.2.2.2.25712
        State: Oper; Msgs sent/rcvd: 9/9; Downstream
        Up time: 00:01:23
        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: 3.3.3.3.50470 - 2.2.2.2.646
        State: Oper; Msgs sent/rcvd: 8/8; Downstream
        Up time: 00:00:54
        LDP discovery sources:
          FastEthernet0/1, 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

Type escape sequence to abort.
Tracing the route to 3.3.3.3

  1 10.0.0.2 [MPLS: Label 17 Exp 0] 20 msec 60 msec 60 msec
  2 10.0.1.3 60 msec 60 msec 60 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 byconfiguring the vpnv4 address family as below

```
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)#!
Andheri(config-router)#address-family vpnv4
Andheri(config-router-af)#neighbor 3.3.3.3 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
*May 29 17:06:19.459: %BGP-5-ADJCHANGE: neighbor 1.1.1.1 Up
Churchgate(config-router)#neighbor 1.1.1.1 update-source loopback 0
Churchgate(config-router)#no auto-summary
Churchgate(config-router)#address-family vpnv4
Churchgate(config-router-af)#neighbor 1.1.1.1 activate
```

To verify the BGP session between R1 and R3 issue the command sh bgp vpnv4 unicast allsummary

```
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       6        1    0    0 00:00:30        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(config)#int lo0
Borivali(config-if)#ip ad
*May 29 17:13:47.223: %LINEPROTO-5-UPDOWN: Line protocol o
Borivali(config-if)#ip address 4.4.4.4 255.255.255.255
Borivali(config-if)#ip ospf 2 area 2
Borivali(config-if)#int f0/0
Borivali(config-if)#ip addresss 192.168.1.4 255.255.255.0
                              ^
% Invalid input detected at '^' marker.

Borivali(config-if)#ip address 192.168.1.4 255.255.255.0
Borivali(config-if)#ip ospf 2 area 2
Borivali(config-if)#no shut
```

```
Andheri(config)#int f0/1
Andheri(config-if)#no shut
Andheri(config-if)#ip address
*May 29 17:14:16.199: %LINK-3-UPDOWN: Interface FastEther
*May 29 17:14:17.199: %LINEPROTO-5-UPDOWN: Line protocol
Andheri(config-if)#ip address 192.168.1.1 255.255.255.0
```

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

```
Andheri(config-vrf)#int f0/1
Andheri(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 192.168.1.1 removed due to enabling VRF RED
```

```
Andheri#sh run int f0/1
Building configuration...

Current configuration : 119 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 ip address 192.168.1.1 255.255.255.0
 duplex auto
 speed auto
end
```

If you issue the command sh ip route this shows the routes in the global table and you willnotice that you do not see 192.168.1.0/24

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

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, 00:19:39, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/3] via 10.0.0.2, 00:18:35, FastEthernet0/0
     10.0.0.0/24 is subnetted, 2 subnets
C       10.0.0.0 is directly connected, FastEthernet0/0
O       10.0.1.0 [110/2] via 10.0.0.2, 00:18:45, FastEthernet0/0
```

```
Andheri#sh ip route vrf RED

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

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

We just need to enable OSPF on this interface and get the loopback address for R4 in the VRFRED routing table before proceeding.

```
Andheri(config)#int f0/1
Andheri(config-if)#ip ospf 2 area 2
```

If we now check the routes in the VRF RED routing table you should see 4.4.4.4 in there as well.

```
Andheri#sh ip route vrf RED

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

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, 00:00:11, FastEthernet0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

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

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, 00:28:18, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/3] via 10.0.0.2, 00:27:14, FastEthernet0/0
     10.0.0.0/24 is subnetted, 2 subnets
C       10.0.0.0 is directly connected, FastEthernet0/0
O       10.0.1.0 [110/2] via 10.0.0.2, 00:27:24, FastEthernet0/0
```

```
Andheri#sh ip route vrf RED

Routing Table: RED
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter are
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type
       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
       ia - IS-IS inter area, * - candidate default, U - per-user st
       o - ODR, P - periodic downloaded static route

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, 00:07:42, FastEthernet0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

We now need to repeat this process for R3 & R6 Router 6 will peer OSPF using process number2 to a VRF configured on R3. It will use the local site addressing to 192.168.2.0/24

```
Mahim(config)#INT LO0
Mahim(config-if)#
*May 29 17:18:58.903: %LINEPROTO-5-UPDOWN: Line pro
Mahim(config-if)#ip add 6.6.6.6 255.255.255.255
Mahim(config-if)#ip ospf 2 area 2
Mahim(config-if)#int f0/0
Mahim(config-if)#ip add 192.168.2.6 255.255.255.0
Mahim(config-if)#ip ospf 2 area 2
Mahim(config-if)#no shut
```

```
Churchgate(config)#int f0/1
Churchgate(config-if)#no shut
Churchgate(config-if)#ip add
*May 29 17:23:19.111: %LINK-3-UPDOWN: Interface FastEth
*May 29 17:23:20.111: %LINEPROTO-5-UPDOWN: Line protoco
Churchgate(config-if)#ip add 192.168.2.3 255.255.255.0
```

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

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

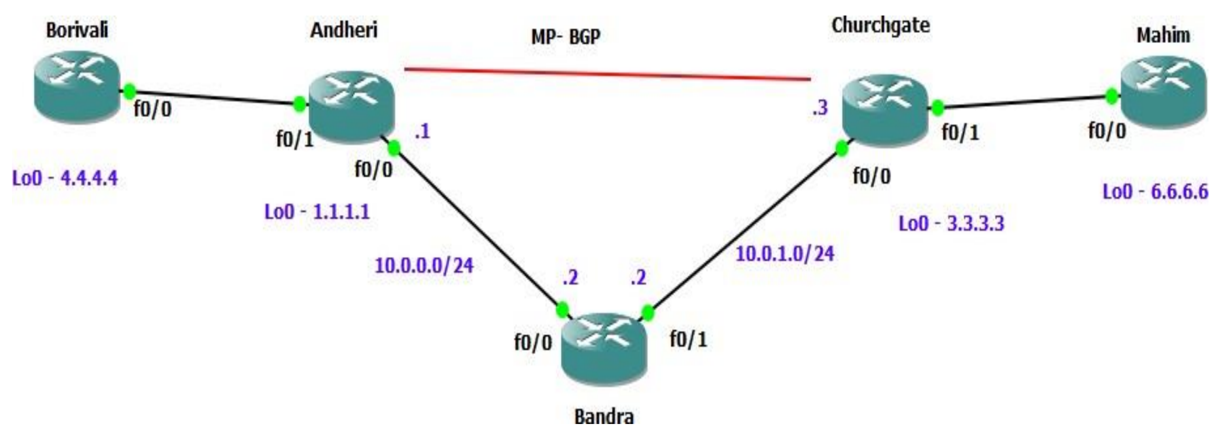
```
Churchgate(config-vrf)#int f0/1
Churchgate(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 192.168.2.3 removed due to enabling VRF RED
Churchgate(config-if)#int f0/1
Churchgate(config-if)#ip add 192.168.2.1 255.255.255.0
```

```
Churchgate#sh run int f0/1
Building configuration...

Current configuration : 119 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 ip address 192.168.2.1 255.255.255.0
 duplex auto
 speed auto
end
```

Check the router in vrf RED

```
Churchgate#sh ip route vrf RED

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

Gateway of last resort is not set

     6.0.0.0/32 is subnetted, 1 subnets
O       6.6.6.6 [110/2] via 192.168.2.6, 00:01:10, FastEthernet0/1
C    192.168.2.0/24 is directly connected, FastEthernet0/1
```

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




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

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
C    192.168.1.0/24 is directly connected, FastEthernet0/0
```

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

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

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, 00:28:18, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/3] via 10.0.0.2, 00:27:14, FastEthernet0/0
     10.0.0.0/24 is subnetted, 2 subnets
C       10.0.0.0 is directly connected, FastEthernet0/0
O       10.0.1.0 [110/2] via 10.0.0.2, 00:27:24, FastEthernet0/0
```

```
Andheri#sh ip route vrf RED

Routing Table: RED
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
       ia - IS-IS inter area, * - candidate default, U - per-user sta
       o - ODR, P - periodic downloaded static route

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, 00:07:42, FastEthernet0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

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

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

```
Andheri#sh ip bgp vpnv4 vrf RED
BGP table version is 9, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   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 ?
*>i6.6.6.6/32       3.3.3.3                  2    100      0 ?
*> 192.168.1.0      0.0.0.0                  0         32768 ?
*>i192.168.2.0      3.3.3.3                  0    100      0 ?
```

```
Churchgate#sh ip bgp vpnv4 vrf RED
BGP table version is 9, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 4:4 (default for vrf RED)
*>i4.4.4.4/32       1.1.1.1                  2    100      0 ?
*> 6.6.6.6/32       192.168.2.6              2         32768 ?
*>i192.168.1.0      1.1.1.1                  0    100      0 ?
*> 192.168.2.0      0.0.0.0                  0         32768 ?
```

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(config)#router ospf 2
Andheri(config-router)#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
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 l
       ia - IS-IS inter area, * - candidate default, U - per-user stat
       o - ODR, P - periodic downloaded static route

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
     6.0.0.0/32 is subnetted, 1 subnets
O IA    6.6.6.6 [110/3] via 192.168.1.1, 00:00:50, FastEthernet0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
O IA 192.168.2.0/24 [110/2] via 192.168.1.1, 00:00:50, FastEthernet0/0
```

Do the same step of on R6

```
Mahim#sh 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 lev
       ia - IS-IS inter area, * - candidate default, U - per-user static
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
O IA    4.4.4.4 [110/3] via 192.168.2.1, 00:00:22, FastEthernet0/0
     6.0.0.0/32 is subnetted, 1 subnets
C       6.6.6.6 is directly connected, Loopback0
O IA 192.168.1.0/24 [110/2] via 192.168.2.1, 00:00:22, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

Lets chevk ping command

```
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, letsdo a trace

```
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
```