INDEX

Practical No-1

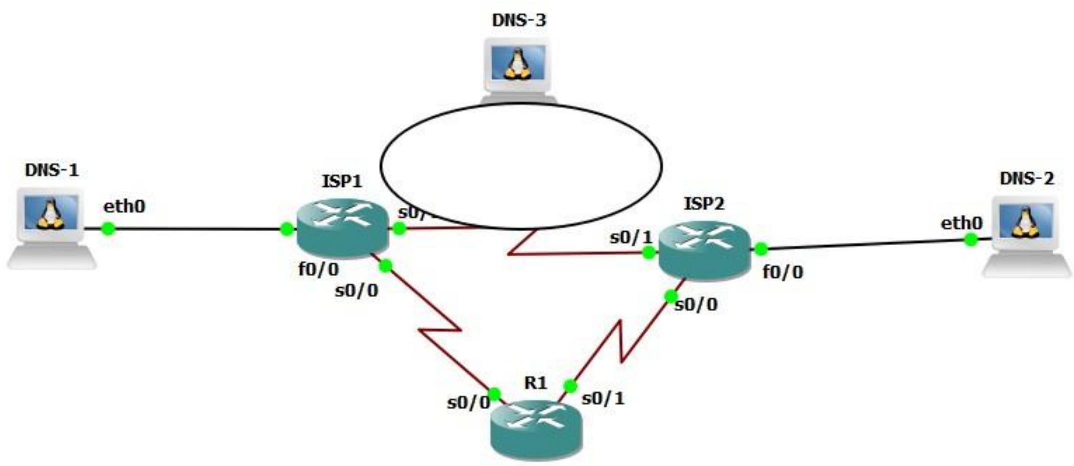
Aim: Configure IP SLA Tracking and Path Control

Theory: IP SLA is a Cisco IOS feature that allows active monitoring of network performance by generating traffic and measuring parameters like delay, jitter, and packet loss. This proactive monitoring helps in assessing the quality of network paths.

Path Control involves directing traffic over specific network paths based on performance metrics. By integrating IP SLA with tracking objects, routers can make intelligent routing decisions, ensuring optimal path selection and network resilience.

Tracking Objects are used to monitor the status of IP SLA operations. If an IP SLA operation fails (e.g., due to high latency or packet loss), the tracking object reflects this state, allowing the router to adjust its routing decisions accordingly.

Topology :



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.

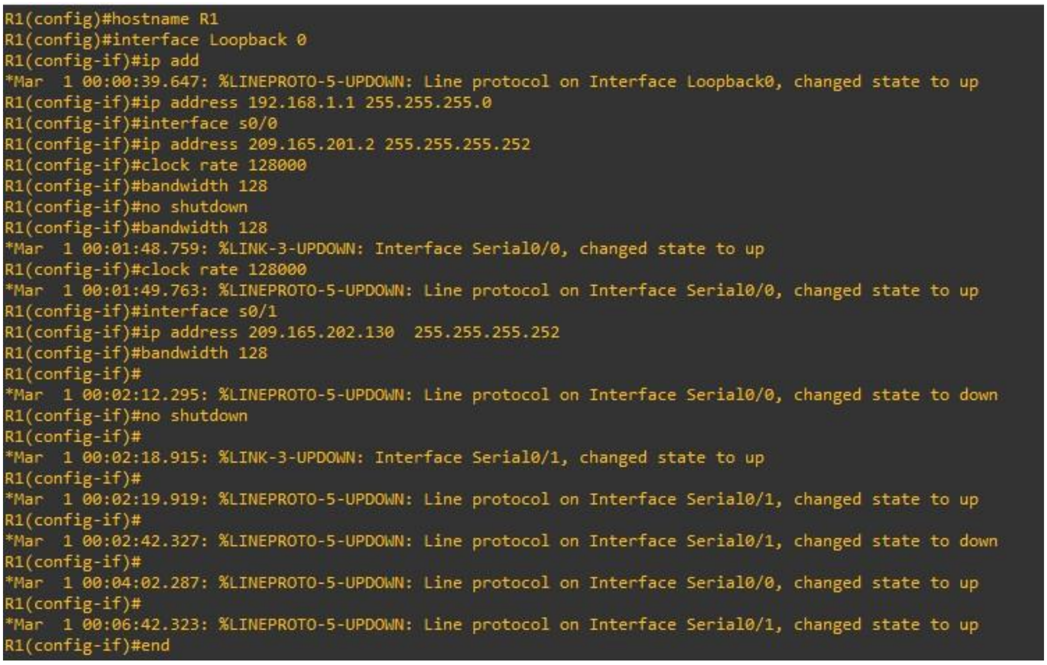
Router R1

Interface Loopback 0
Ip address 223.168.1.4 255.255.255.0

Interface serial 0/0/0
Ip address 223.165.201.2 255.255.255.252

no shutdown interface serial 0/0/1 ip

address 223.165.202.130 255.255.255.252



```
R1(config)#hostname R1
R1(config)#interface Loopback 0
R1(config-if)#ip add
*Mar  1 00:00:39.647: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#interface s0/0
R1(config-if)#ip address 209.165.201.2 255.255.255.252
R1(config-if)#clock rate 128000
R1(config-if)#bandwidth 128
R1(config-if)#no shutdown
R1(config-if)#bandwidth 128
*Mar  1 00:01:48.759: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R1(config-if)#clock rate 128000
*Mar  1 00:01:49.763: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R1(config-if)#interface s0/1
R1(config-if)#ip address 209.165.202.130  255.255.255.252
R1(config-if)#bandwidth 128
R1(config-if)#
*Mar  1 00:02:12.295: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to down
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 00:02:18.915: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
R1(config-if)#
*Mar  1 00:02:19.919: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
R1(config-if)#
*Mar  1 00:02:42.327: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to down
R1(config-if)#
*Mar  1 00:04:02.287: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R1(config-if)#
*Mar  1 00:06:42.323: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
R1(config-if)#end
```

Router ISP1(R2)

Interface Loopback 0
Ip address 223.165.200.254 255.255.255.255
Interface Loopback 1
Ip address 223.165.201.30 255.255.255.255
Int s0/0
Ip address 223.165.201.1 255.255.255.252

no shutdown int s0/1 ip address

223.165.200.225 255.255.255.252 no

shutdown

```
ISP1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP1(config)#hostname ISP1
ISP1(config)#interface Loopback0
ISP1(config-if)#ip a
*Mar  1 00:01:38.955: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to upd
ISP1(config-if)#ip address 209.165.200.254 255.255.255.255
ISP1(config-if)#interface Loopback1
ISP1(config-if)#ip address 209.165.200.254 255.255.255.255
*Mar  1 00:02:03.959: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1, changed state to up
ISP1(config-if)#ip address 209.165.201.30 255.255.255.255
ISP1(config-if)#interface s0/0
ISP1(config-if)#ip address 209.165.201.1 255.255.255.252
ISP1(config-if)#bandwidth 128
ISP1(config-if)#no shutdown
ISP1(config-if)#interface s0/0
*Mar  1 00:02:47.979: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
ISP1(config-if)#interface s0/0
*Mar  1 00:02:48.983: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
ISP1(config-if)#interface s0/1
ISP1(config-if)#ip address 209.165.200.225 255.255.255.252
ISP1(config-if)#clock rate 128000
ISP1(config-if)#bandwidth 128
ISP1(config-if)#no shutdown
ISP1(config-if)#
```

ISP2 Router 3: Interface Loopback 0 # ip

address 223.165.200.254 255.255.255.255

Interface Loopback 1 # ip address

223.165.202.158 255.255.255.255

Int s0/0
Ip address 223.165.202.129 255.255.255.252 no

shutdown #

int s0/1 ip address 223.165.200.226

255.255.255.255

```
ISP2(config)#hostname ISP2
ISP2(config)#interface Loopback0
ISP2(config-if)#ip
*Mar  1 00:02:46.623: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ISP2(config-if)#ip address 209.165.200.254 255.255.255.255
ISP2(config-if)#interface Loopback1
ISP2(config-if)#ip address 209.165.200.254 255.255.255.255
*Mar  1 00:03:17.855: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1, changed state to up
ISP2(config-if)#ip address 209.165.202.158 255.255.255.255
ISP2(config-if)#interface s0/0
ISP2(config-if)#ip address 209.165.202.129 255.255.255.252
ISP2(config-if)#clock rate 128000
ISP2(config-if)#bandwidth 128
ISP2(config-if)#no shutdown
ISP2(config-if)#
*Mar  1 00:04:05.611: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
ISP2(config-if)#no shutdown
*Mar  1 00:04:06.615: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
ISP2(config-if)#interface s0/1
ISP2(config-if)#ip address 209.165.200.226 255.255.255.252
ISP2(config-if)#bandwidth 128
ISP2(config-if)#no shutdown
ISP2(config-if)#
```

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

R1 show interface description

```
R1(config-if)#end
R1#
*Mar  1 00:07:05.451: %SYS-5-CONFIG_I: Configured from console by console
R1#show interfaces description | include up
Se0/0                          up             up
Se0/1                          up             up
Lo0                            up             up
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
```

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. **C) Router R1 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
R1(config)#exit
R1#
```

ISP1(R2) Router eigrp 1 network 223.165.200.224

0.0.03 network 223.165.201.4 0.0.0.31 no auto-

summary ip route 223.168.1.0 255.255.255.0

223.165.201.2

```
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)#exit
ISP1(config)#ip route 192.168.1.0 255.255.255.0 209.165.201.2
ISP1(config)#
*Mar  1 00:08:31.895: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 209.165.200.226 (Serial0/1) is up: new adjacency
ISP1(config)#
*Mar  1 00:08:47.411: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 209.165.200.226 (Serial0/1) is resync: peer graceful-restar
t
ISP1(config)#
```

ISP2(R3)

Router eigrp 1

Network 223.165.200.224 0.0.0.0

#

Network 223.165.202.128 0.0.031 no auto-summary

 # ip route 223.168.1.0 255.255.255.0

223.165.202.130

```
ISP2(config)#router eigrp 1
ISP2(config-router)#network 209.165.200.224 0.0.0.3
ISP2(config-router)#network 209.165.200.224 0.0.0.3
*Mar  1 00:07:15.307: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 209.165.200.225 (Serial0/1) is up: new adjacency
ISP2(config-router)#network 209.165.202.128 0.0.0.31
ISP2(config-router)#no auto-summary
ISP2(config-router)#exit
*Mar  1 00:07:33.355: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 209.165.200.225 (Serial0/1) is resync: summary configured
ISP2(config-router)#exit
ISP2(config)#ip route 192.168.1.0 255.255.255.0 209.165.202.130
ISP2(config)#
```

Step 2: Verify server reachability.

a. Before Implementing the Cisco IOS SLA feature, you must verify reachabilty 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 into R1.

R1(tcl) # foreach address {
+>(tcl) #223.165.200.254
+>(tcl) #223.165.201.30
+>(tcl) #223.165.202.158
+>(tcl) #} {
+>(tcl) #ping $address source 223.168.1.4
+>(tcl) #}

```
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 = 36/41/52 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 = 40/40/44 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 = 40/41/44 ms
R1(tcl)#foreach  address {
+>(tcl)#209.165.201.3
```

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) #223.165.200.254
+>(tcl) #223.165.201.30
+>(tcl) #223.165.202.158
+>(tcl) #} {
+>(tcl) # trace $address source 223.168.1.4 +>(tcl) #}

```
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 40 msec 36 msec 40 msec
Type escape sequence to abort.
Tracing the route to 209.165.201.30

  1 209.165.201.1 40 msec 36 msec 40 msec
Type escape sequence to abort.
Tracing the route to 209.165.202.158

  1 209.165.201.1 40 msec 40 msec 36 msec
  2 209.165.200.226 36 msec 40 msec 40 msec
R1(tcl)#
```

Step 3: Configure IP SLA probes.

a. Create and 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 command has replaced the type echo protocol Ip ICMP Echo command.

R1(config)# ip sla 11
R1(config-ip-sla)# icmp-echo 223.165.201.30
R1(config-ip-sla-echo) # frequency 10
R1(config-ip-sla-echo) #exit

```
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)#end
```

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

R1# show ip sla configuration 11

```
R1#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
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Vrf Name:
Request size (ARR data portion): 28
Verify data: No
Schedule:
   Operation frequency (seconds): 10  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
   Number of statistic hours kept: 2
   Number of statistic distribution buckets kept: 1
   Statistic distribution interval (milliseconds): 4294967295
History Statistics:
   Number of history Lives kept: 0
   Number of history Buckets kept: 15
   History Filter Type: None
Enhanced History:
```

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

```
R1#show ip sla statistics

Round Trip Time (RTT) for       Index 11
        Latest RTT: 40 milliseconds
Latest operation start time: *00:37:28.259 UTC Fri Mar 1 2002
Latest operation return code: OK
Number of successes: 19
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)#ip sla 22
R1(config-ip-sla)#icmp-echo 209
                            ^
% Invalid input detected at '^' marker.

R1(config-ip-sla)#icmp-echo 209.165.202.158
R1(config-ip-sla-echo)#frequency 10
R1(config-ip-sla-echo)#exit
R1(config)#ip sla schedule 22 life forever start-time now
R1(config)#end
R1#
*Mar  1 00:39:07.895: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

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
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Vrf Name:
Request size (ARR data portion): 28
Verify data: No
Schedule:
   Operation frequency (seconds): 10  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
   Number of statistic hours kept: 2
   Number of statistic distribution buckets kept: 1
   Statistic distribution interval (milliseconds): 4294967295
History Statistics:
   Number of history Lives kept: 0
   Number of history Buckets kept: 15
   History Filter Type: None
Enhanced History:

R1#
```

R1# show ip sla statistics 22

```
R1#show ip sla statistics 22

Round Trip Time (RTT) for       Index 22
        Latest RTT: 44 milliseconds
Latest operation start time: *00:41:25.547 UTC Fri Mar 1 2002
Latest operation return code: OK
Number of successes: 15
Number of failures: 0
Operation time to live: Forever


R1#
```

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 0.0.0.0 223.165.201.1

```
R1(config)#no ip route 0.0.0.0 0.0.0.0 223.165.201.1
%No matching route to delete
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#
*Mar  1 00:43:14.203: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

b. Verify the Routing table.

```
R1#show ip route | begin Gateway
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, Serial0/0
     209.165.202.0/30 is subnetted, 1 subnets
C       209.165.202.128 is directly connected, Serial0/1
C    192.168.1.0/24 is directly connected, Loopback0
S*   0.0.0.0/0 [5/0] via 209.165.201.1
R1#
```

Conclusion:

By configuring IP SLA tracking and path control, the network can dynamically adjust to changing conditions, ensuring optimal path selection and improved network resilience. This setup is particularly beneficial in scenarios where multiple ISPs are used, allowing for automatic failover in case of link degradation or failure.

Practical No- 2

Aim: Using AS_PATH Attribute

Theory: The Border Gateway Protocol (BGP) is a standardized exterior gateway protocol designed to exchange routing and reachability information among autonomous systems (ASes) on the Internet. BGP is classified as a path-vector routing protocol, and it makes routing decisions based on paths, network policies, or rule-sets configured by a network administrator .

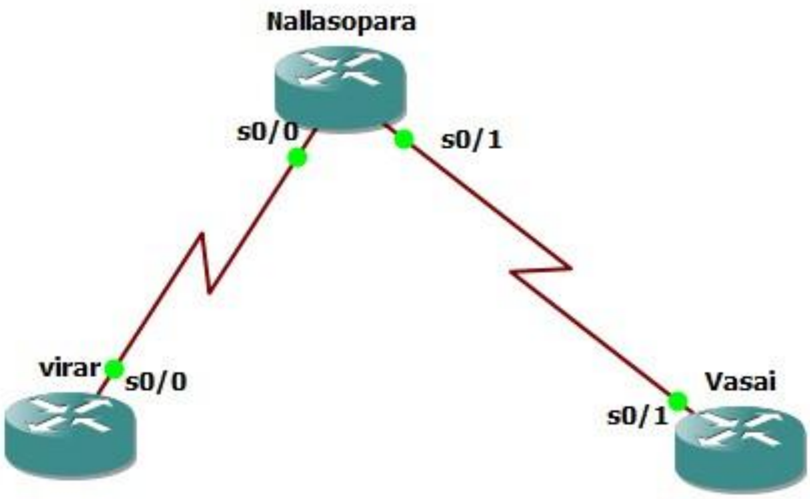
One of the key attributes in BGP is the AS_PATH attribute. This attribute lists the sequence of ASes that routing information has traversed. It serves two primary purposes:

Loop Prevention: By examining the AS_PATH, a BGP router can detect routing loops. If a router sees its own AS number in the AS_PATH of a received route, it will reject that route to prevent a loop .

Path Selection: BGP prefers routes with shorter AS_PATHs, assuming all other attributes are equal. This preference helps in selecting the most efficient path to a destination .

Additionally, network administrators can manipulate the AS_PATH attribute using a technique called AS Path Prepending. This involves adding multiple instances of an AS number to the AS_PATH to make a route less attractive, thereby influencing the path selection process.

Topology:



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 sources AS number.

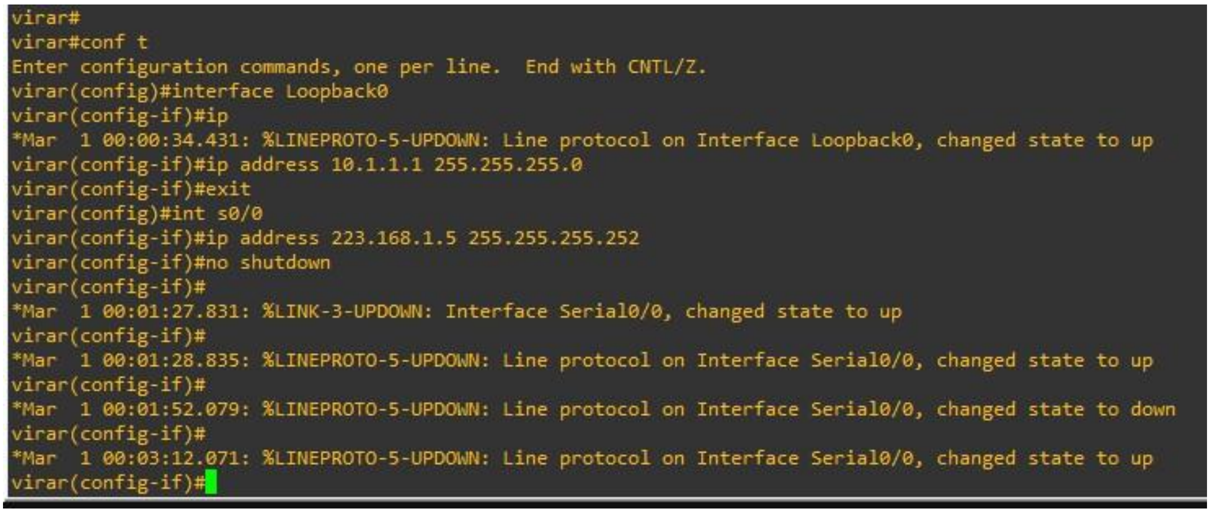
Step 1: Prepare the routers for the lab.

Cable the network as shown in the topology diagram. Erase the Startup configuration and reload each router to clear previous configurations. **Step 2: Configure the hostname and interface addresses.**

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

Router R1(hostname Virar)

Virar(config)# interface Loopback 0
Virar(config-if) #ip address 10.1.1.1 255.255.255.255
Virar(config-if) #exit
Virar(config-if) #int s0/0
Virar(config-if) #ip address 223.168.1.5 255.255.255.252 Virar(config-if) #no shutdown
Virar(config-if) #end
Virar#
Virar(R1)



```
virar#
virar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
virar(config)#interface Loopback0
virar(config-if)#ip
*Mar  1 00:00:34.431: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
virar(config-if)#ip address 10.1.1.1 255.255.255.0
virar(config-if)#exit
virar(config)#int s0/0
virar(config-if)#ip address 223.168.1.5 255.255.255.252
virar(config-if)#no shutdown
virar(config-if)#
*Mar  1 00:01:27.831: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
virar(config-if)#
*Mar  1 00:01:28.835: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
virar(config-if)#
*Mar  1 00:01:52.079: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to down
virar(config-if)#
*Mar  1 00:03:12.071: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
virar(config-if)#
```

Router R2(hostname Nallasopara)

Nallasopra(config) #interface Loopback 0
Nallasopara(config-if) #ip address 10.2.2.1 255.255.255.0
Nallasopara(config-if) #exit
Nallasopara(config-if) #int s0/0
Nallasopara(config-if) #ip address 223.168.1.6 255.255.255.252
Nallasopara(config-if) #no shutdown
Nallasopara(config-if) #exit
Nallasopara(R2)

Nallasopara(config-if) #int s0/1
Nallasopara(config-if) #ip address 172.24.1.17 255.255.255.252
Nallasopara(config-if) #no shutdown
Nallasopara(config-if) #end
Nallasopara#

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#interface Loopback0
Nallasopara(config-if)#
*Mar  1 00:01:14.715: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Nallasopara(config-if)#ip address 10.2.2.1 255.255.255.0
Nallasopara(config-if)#exit
Nallasopara(config)#int s0/0
Nallasopara(config-if)#ip address 223.168.1.6 255.255.255.252
Nallasopara(config-if)#no shutdown
Nallasopara(config-if)#exit
Nallasopara(config)#
*Mar  1 00:02:19.055: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
Nallasopara(config)#
*Mar  1 00:02:20.059: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
Nallasopara(config)#int s0/1
Nallasopara(config-if)#ip address 172.24.1.17 255.255.255.252
Nallasopara(config-if)#no shutdown
Nallasopara(config-if)#
*Mar  1 00:04:59.387: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
Nallasopara(config-if)#
*Mar  1 00:05:00.391: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
Nallasopara(config-if)#exit
```

Router 3(hostname Vasai)

Vasai(config)# interface Loopback 0
Vasai(config-if) #ip address 10.3.3.1 255.255.255.0
Vasai(config-if) #exit
Vasai(config-if) #int s0/1
Vasai(config-if) #ip address 172.24.1.18 255.255.255.252
Vasai(config-if) #no shutdown
Vasai(config-if) #end
Vasai#
Vasai(R3)

```
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#interface Loopback0
Vasai(config-if)#i
*Mar  1 00:02:12.907: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Vasai(config-if)#ip address 10.3.3.1 255.255.255.0
Vasai(config-if)#exit
Vasai(config)#int s0/1
Vasai(config-if)#ip address 172.24.1.18 255.255.255.252
Vasai(config-if)#no shutdown
Vasai(config-if)#
*Mar  1 00:02:59.967: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
Vasai(config-if)#end
*Mar  1 00:03:00.971: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
Vasai(config-if)#end
Vasai#
*Mar  1 00:03:03.919: %SYS-5-CONFIG_I: Configured from console by console
Vasai#
*Mar  1 00:03:22.147: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to down
Vasai#
*Mar  1 00:04:12.139: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#interface Loopback0
Vasai(config-if)#ip address 10.3.3.1 255.255.255.0
Vasai(config-if)#exit
Vasai(config)#int s0/1
Vasai(config-if)#ip address 172.24.1.18 255.255.255.252
Vasai(config-if)#no shutdown
Vasai(config-if)#
```

B. Use Ping to test the Connectivity between the directly connected routers

Nallasopara(R2)

```
Nallasopara#
*Mar  1 00:06:52.091: %SYS-5-CONFIG_I: Configured from console by console
Nallasopara#ping 223.168.1.5

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 223.168.1.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/48/80 ms
Nallasopara#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 = 40/42/44 ms
Nallasopara#
```

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.
Virar(config)# router bgp 100
Virar(config-router) #neighbor 223.168.1.6 remote-as 300
Virar(config-router) #network 10.1.1.0 mask 255.255.255.0
Virar(config-router) #

Virar

```
virar(config-if)#exit
virar(config)#router bgp 100
virar(config-router)#neighbor 223.168.1.6 remote-as 300
virar(config-router)#network 10.1.1.0 mask 255.255.255.0
virar(config-router)#
*Mar  1 00:16:22.255: %BGP-5-ADJCHANGE: neighbor 223.168.1.6 Up
virar(config-router)#
```

Nallasopara(config)# router bgp 300
Nallasopara(config-router) #neighbor 223.168.1.5 remote-as 100
Nallasopara(config-router) #neighbor 172.24.1.18 remote-as 65000
Nallasopara(config-router) #10.20.2.0 mask 255.255.255.0

Nallasopara

```
Nallasopara(config)#router bgp 300
Nallasopara(config-router)#neighbor 223.168.1.5 remote-as 100
Nallasopara(config-router)#
*Mar  1 00:15:32.199: %BGP-5-ADJCHANGE: neighbor 223.168.1.5 Up
Nallasopara(config-router)#neighbor 172.24.1.18 remote-as 65000
Nallasopara(config-router)#network 10
% Incomplete command.

Nallasopara(config-router)#network 10.2.2.0 mask 255.255.255.0
Nallasopara(config-router)#end
*Mar  1 00:17:36.731: %BGP-5-ADJCHANGE: neighbor 172.24.1.18 Up
Nallasopara(config-router)#end
Nallasopara#
```

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

```
Vasai(config)#router bgp 65000
Vasai(config-router)#neighbor 172.24.1.17 remote-as 300
Vasai(config-router)#network 10.3.3.0 mask 255.255.255.0
Vasai(config-router)#
*Mar  1 00:16:06.731: %BGP-5-ADJCHANGE: neighbor 172.24.1.17 Up
Vasai(config-router)#
```

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

Nallasopara# show ip bgp neighbors

```
Nallasopara#
*Mar  1 00:17:38.611: %SYS-5-CONFIG_I: Configured from console by console
Nallasopara#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 00:00:17
  Last read 00:00:17, last write 00:00:17, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received(old & new)
    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:                2          1
    Keepalives:             3          3
    Route Refresh:          0          0
    Total:                  6          5
  Default minimum time between advertisement runs is 30 seconds

 For address family: IPv4 Unicast
  BGP table version 4, neighbor version 3/0
  Output queue size: 0
  Index 1, Offset 0, Mask 0x2
 --More--
```

Step 4: Remove the Private AS.

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

Virar# Show ip route

```
virar#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

     223.168.1.0/30 is subnetted, 1 subnets
C       223.168.1.4 is directly connected, Serial0/0
     10.0.0.0/24 is subnetted, 3 subnets
B       10.3.3.0 [20/0] via 223.168.1.6, 00:03:53
B       10.2.2.0 [20/0] via 223.168.1.6, 00:05:22
C       10.1.1.0 is directly connected, Loopback0
virar#
```

b. Ping Again, this time as an extended ping, sourcing from the Loopback 0 interface address. Ping 10.3.3.1 source 10.1.1.1 or ping 10.3.3.1 source Lo0

```
virar#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 = 1/3/12 ms
virar#show ip bgp
BGP table version is 4, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
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      223.168.1.6              0             0 300 i
*> 10.3.3.0/24      223.168.1.6                            0 300 65000 i
virar#
```

c. Now check the BGP table on Virar. The AS_PATH to the 10.3.3.0 network should be AS 300. It no longer has the private AS in the path.

```
virar#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 = 1/1/1 ms
virar#show ip bgp
BGP table version is 4, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
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      223.168.1.6              0             0 300 i
*> 10.3.3.0/24      223.168.1.6                            0 300 65000 i
virar#
```

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 Nallasopara.

Nallasopara(config) #ip as_path access-list 1 deny ^100$

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

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

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

Nallasopara(config) #router bgp 300

Nallasopara(config-router) #neighbor 223.168.1.5 remove-private-as

```
Nallasopara(config)#router bgp 300
Nallasopara(config-router)#neighbors 172.24.1.18 filter-list 1 out
                                    ^
% Invalid input detected at '^' marker.

Nallasopara(config-router)#neighbor 172.24.1.18 filter-list 1 out
Nallasopara(config-router)#exit
Nallasopara(config)#
```

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

Virar# show ip route

```
virar#show ip bgp
BGP table version is 4, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
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      223.168.1.6              0             0 300 i
*> 10.3.3.0/24      223.168.1.6                            0 300 65000 i
virar#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

     223.168.1.0/30 is subnetted, 1 subnets
C       223.168.1.4 is directly connected, Serial0/0
     10.0.0.0/24 is subnetted, 3 subnets
B       10.3.3.0 [20/0] via 223.168.1.6, 00:12:08
B       10.2.2.0 [20/0] via 223.168.1.6, 00:13:38
C       10.1.1.0 is directly connected, Loopback0
virar#
```

d. Return to Nallasopara and Verify that the filter is working as intended.

Nallasopara# show ip bgp regexp ^100$

```
Nallasopara#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      223.168.1.5              0             0 100 i
Nallasopara#
```

e. Run the following TCL scripts on all routers to verify whether there is connectivity. All pings from Nallasopara should be successful. Virar should not be able to ping the

Vasai loopback 10.3.3.1 or The WAN link 172.24.1.6/30. Vasai should not be able to ping Virar Loopback 10.1.1.1 or the WAN link 223.168.1.4/30

```
Nallasopara#tclsh
Nallasopara(tcl)#foreach address {
+>10.1.1.1
+>10.2.2.1
+>10.3.3.1
+>223.168.1.5
+>223.168.1.6
+>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 = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 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 = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 223.168.1.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 223.168.1.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/16 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 = 1/1/1 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 = 1/7/20 ms
Nallasopara(tcl)#
```

Conclusion:

This practical demonstrates the significance of the AS_PATH attribute in BGP for both loop prevention and path selection. By manipulating the AS_PATH using techniques like AS Path Prepending, network administrators can influence routing decisions to achieve desired traffic engineering outcomes. The GNS3 simulation effectively illustrates how BGP routers use the AS_PATH attribute to determine the best path to a destination network.

Practical No – 3

Aim: Configure IBGP and EBGP Sessions, Local Preference, and MED.

Theory:
Border Gateway Protocol (BGP) Overview

BGP (Border Gateway Protocol) is a routing protocol used to exchange routing information between different networks on the internet. It is classified into two types:

- **External BGP (EBGP)** – Used between different Autonomous Systems (AS).
- **Internal BGP (IBGP)** – Used within the same Autonomous System.

External BGP (EBGP)

- EBGP is used to exchange routes between different Autonomous Systems (ASes).
- It is commonly used by Internet Service Providers (ISPs) and large enterprises to communicate with external networks.
- The default Time-To-Live (TTL) value is 1, meaning that EBGP peers must be directly connected unless explicitly configured otherwise.
- AS-path attribute is used in EBGP to prevent routing loops.
- It prefers shorter AS paths when selecting the best route.
- **Example Scenario:**
  If AS100 wants to exchange routes with AS200, they establish an EBGP connection between their routers.

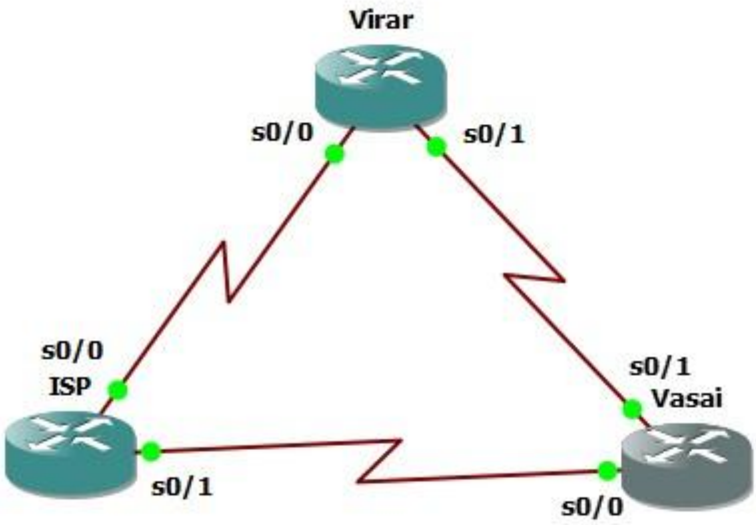
Internal BGP (IBGP)

- IBGP is used for routing within the same Autonomous System.
- It ensures that all routers in an AS have a consistent view of external routes learned via EBGP.
- Unlike EBGP, IBGP does not modify the AS-path attribute.
- IBGP requires a full mesh of connections (or Route Reflectors/Confederations to reduce overhead).
- Next-hop attribute must be reachable within the AS for proper routing.
- **Example Scenario:**
  If AS100 has multiple routers, they must use IBGP to share routes learned from EBGP peers.

Key Differences Between EBGP and IBGP

Feature	EBGP	IBGP
Used for	Between different ASes	Within the same AS
AS-Path Modification	Yes	No
Next-Hop Change	Yes	No (next-hop must be reachable)
Default TTL	1	255
Full Mesh Required?	No	Yes (or use Route Reflectors)

Topology:



Objectives:

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

Step 1: Configure interface addresses. Router R1(hostname ISP)
ISP (config)# interface Loopback 0
ISP (config-if) #ip address 223.168.100.1 255.255.255.0
ISP (config-if) #exit
ISP (config) #interface Serial 0/0/0
ISP (config-if) #ip address 223.168.1.5 255.255.255.252
ISP (config-if) #exit
ISP (config) #interface Serial 0/0/1
ISP (config-if) #ip address 223.168.1.1 255.255.255.252
ISP (config-if) #no shutdown
ISP (config-if) #end

```
ISP#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP(config)#interface Loopback 0
ISP(config-if)#
*Mar  1 00:03:13.175: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ISP(config-if)#ip address 223.168.100.1 255.255.255.0
ISP(config-if)#exit
ISP(config)#interface s0/0
ISP(config-if)#ip address 223.168.1.5 255.255.255.252
ISP(config-if)#exit
ISP(config)#interface s0/1
ISP(config-if)#ip address 223.168.1.1 255.255.255.252
ISP(config-if)#no shutdown
ISP(config-if)#end
ISP#
```

Router R2(hostname Virar)

Virar(config) #interface Loopback 0
Virar(config-if) #ip address 172.16.64.1 255.255.255.0
Virar(config) #exit
Virar(config) #interface Serial 0/0/0
Virar(config-if) #ip address 223.168.1.6 255.255.255.252
Virar(config-if) #no shutdown
Virar(config-if) #exit
Virar(config) interface Serial 0/0/1
Virar(config-if) #ip address 172.16.1.1 255.255.255.0
Virar(config-if) #no shutdown
Virar(config-if) #end

```
Virar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Virar(config)#interface Loopback 0
Virar(config-if)#ip
*Mar  1 00:05:27.387: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Virar(config-if)#ip address 172.16.64.1 255.255.255.0
Virar(config-if)#exit
Virar(config)#interface s0/0
Virar(config-if)#ip address 223.168.1.6 255.255.255.252
Virar(config-if)#no shutdown
Virar(config-if)#exit
Virar(config)#
*Mar  1 00:06:40.707: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
*Mar  1 00:06:41.743: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
Virar(config)#interface s0/1
Virar(config-if)#ip address 172
*Mar  1 00:07:02.147: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to down
Virar(config-if)#ip address 172.16.1.1 255.255.255.0
Virar(config-if)#no shutdown
Virar(config-if)#no shutdown
*Mar  1 00:07:20.399: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
Virar(config-if)#no shutdown
*Mar  1 00:07:21.403: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
Virar(config-if)#end
Virar#
*Mar  1 00:07:26.491: %SYS-5-CONFIG_I: Configured from console by console
Virar#
```

Router R3(hostname Vasai)

Vasai(config) #interface Loopback 0
Vasai(config-if) #ip address 172.16.32.1 255.255.255.0
Vasai(config-if) #exit
Vasai(config) #interface Serial 0/0/0
Vasai(config-if) #ip address 223.168.1.2 255.255.255.252
Vasai(config-if) #no shutdown
Vasai(config-if) #exit
Vasai(config) #interface Serial 0/0/1
Vasai(config-if) #ip address 172.16.1.2 255.255.255.0
Vasai(config-if) #no shutdown
Vasai(config-if) #end

```
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#interface Loopback 0
Vasai(config-if)#ip
*Mar  1 00:07:13.483: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Vasai(config-if)#ip address 172.16.32.1 255.255.255.0
Vasai(config-if)#exit
Vasai(config)#interface s0/0
Vasai(config-if)#ip address 223.168.1.2 255.255.255.252
Vasai(config-if)#no shutdown
Vasai(config-if)#exit
Vasai(config)#
*Mar  1 00:08:19.475: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
Vasai(config)#
*Mar  1 00:08:20.479: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
Vasai(config)#interface s0/1
Vasai(config-if)#ip address 172.16.1.2 255.255.255.0
Vasai(config-if)#no shutdown
Vasai(config-if)#no sh
*Mar  1 00:08:51.031: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
Vasai(config-if)#
*Mar  1 00:08:52.035: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
Vasai(config-if)#end
Vasai#
*Mar  1 00:08:55.863: %SYS-5-CONFIG_I: Configured from console by console
Vasai#
```

Step 2: Configure EIGRP.

Confiigure EIGRP between the Virar and Vasai routers. (Note: if using an IOS prior to 15.0, use the no auto-summary router configuration command to disable automatic summary this command is the default beginning with IOS 15)
Virar(config) #router eigrp 1
Virar(config-router) #network 172.16.0.0

```
Virar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Virar(config)#router eigrp 1
Virar(config-router)#network 172.16.0.0
Virar(config-router)#
```

Vasai(config) #router eigrp 1
Vasai(config-router) #network 172.16.0.0

```
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#router eigrp 1
Vasai(config-router)#network 172.16.0.0
Vasai(config-router)#
*Mar  1 00:11:30.947: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16.1.1 (Serial0/1) is up: new adjacency
Vasai(config-router)#
```

Step 3: Configure IBGP and Verify BGP neighbors.

a. Configure IBGP between the Virar and Vasai routers. On the Virar router, enter the following configuration.

Virar(config) #router bgp 64512
Virar(config-router) #neighbor 172.16.32.1 remote-as 64512
Virar(config-router) #neighbor 172.16.32.1 update-source Lo0

```
Virar(config)#router bgp 64512
Virar(config-router)#neighour 172.16.32.1 remote-as 64512
                                   ^
% Invalid input detected at '^' marker.

Virar(config-router)#neighbor 172.16.32.1 remote-as 64512
Virar(config-router)#neighbor 172.16.32.1 update-source Lo0
Virar(config-router)#
```

If multiple pathways to the BGP neighbor exist, the router can use multiple IP interfaces to communicate with 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 Loopback 0 as the source IP address for all BGP messages sent to that neighbor.

b. Complete the IBGP configuration on Vasai using the Following commands. Vasai(config) #router bgp 64512

Vasai(config-router) # neighbor 172.16.64.1 remote-as 64512
Vasai(config-router) #neighbor 172.16.64.1 update-source Lo0

```
Vasai(config)#router bgp 64512
Vasai(config-router)#neighbor 172.16.64.1 remote-as 64512
Vasai(config-router)#neighbor 172.16.64.1 update-source L
*Mar  1 00:17:28.243: %BGP-5-ADJCHANGE: neighbor 172.16.64.1 Up
Vasai(config-router)#neighbor 172.16.64.1 update-source Lo0
Vasai(config-router)#
```

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

```
Vasai#show ip bgp neighbors
BGP neighbor is 172.16.64.1,  remote AS 64512, internal link
  BGP version 4, remote router ID 172.16.64.1
  BGP state = Established, up for 00:01:11
  Last read 00:00:10, last write 00:00:10, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received(old & new)
    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:                0          0
    Keepalives:             3          3
    Route Refresh:          0          0
    Total:                  4          4
```

Step 4: Configure EBGP and Verify BGP neighbors.

d. Configure ISP to run EBGP with Virar and Vasai. Enter the following commands on ISP.

ISP (config) #router bgp 200
ISP (config-router) #neighbor 223.168.1.6 remote-as 64512
ISP (config-router) #neighbor 223168.1.2 remote-as 64512
ISP (config-router) #network 223.168.100.0

```
ISP#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP(config)#router bgp 200
ISP(config-router)#neighbor 223.168.1.6 remote-as 64512
ISP(config-router)#neighbor 223.168.1.2 remote-as 64512
ISP(config-router)#network 223.168.100.0
ISP(config-router)#
```

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 the

ISP. Later in this lab we will configure a default route to the ISP. Virar(config) #ip route 172.16.0.0 255.255.0.0 null0

```
Virar(config-router)#exit
Virar(config)#ip route 172.16.0.0 255.255.0.0 null0
Virar(config)#
```

f. Configure Virar as an EBGP peer to ISP Virar(config) #router bgp 64512

Virar(config-router) #neighbor 223.168.1.5 remote-as 200
Virar(config-router) #network 172.16.0.0

```
Virar(config-router)#neighbor 223.168.1.5 remote-as 200
Virar(config-router)#network 172.16.0.0
Virar(config-router)#exit
Virar(config)#
```

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

Virar# **show ip bgp neighbors**

```
Virar#show ip bgp neighbors
BGP neighbor is 172.16.32.1,  remote AS 64512, internal link
  BGP version 4, remote router ID 172.16.32.1
  BGP state = Established, up for 00:10:24
  Last read 00:00:23, last write 00:00:23, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received(old & new)
    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:                1          0
    Keepalives:            12         12
    Route Refresh:          0          0
    Total:                 14         13
  Default minimum time between advertisement runs is 0 seconds
```

Configure a discard static route for 172.16.0.0/16 on Vasai and an EBGP peer to ISP.
Vasai(config) #ip route 172.16.0.0 255.255.0.0 null0
Vasai(config) #router bgp 64512
Vasai(config-router) #neighbor 223.168.1.1 remote-as 200
Vasai(config-router) #network 172.16.0.0

```
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#ip route 172.16.0.0 255.255.0.0 null0
Vasai(config)#router bgp 64512
Vasai(config-router)#neighbor 223.168.1.1 remote-as 200
Vasai(config-router)#network 172.1
*Mar  1 00:31:26.455: %BGP-5-ADJCHANGE: neighbor 223.168.1.1 Up
Vasai(config-router)#network 172.16.0.0
Vasai(config-router)#
```

Step 5: View BGP summary output.

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

Vasai# **show ip bgp summary**

```
Vasai#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 240 bytes of memory
3 path entries using 156 bytes of memory
4/2 BGP path/bestpath attribute entries using 496 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 3) using 64 bytes of memory
BGP using 980 total bytes of memory
BGP activity 2/0 prefixes, 3/0 paths, scan interval 60 secs

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.64.1     4 64512      19      20        5    0    0 00:15:53        1
223.168.1.1     4   200       8       7        5    0    0 00:01:55        1
Vasai#
```

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 conversation to re-establish with each Virar router.

ISP# clear ip bgp *

```
ISP#clear ip bgp *
ISP#
*Mar  1 00:39:28.035: %BGP-5-ADJCHANGE: neighbor 223.168.1.2 Down User reset
ISP#
*Mar  1 00:39:30.427: %BGP-5-ADJCHANGE: neighbor 223.168.1.2 Up
ISP#
ISP#
```

g. Test whether ISP can ping the Loopback 0 address of 172.16.64.1 on Virar and the serial link between Virar and Vasai, 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:
.....
Success rate is 0 percent (0/5)
ISP#
```

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)
ISP#
```

h. Now ping from ISP to the Loopback 0 address of 172.16.32.1 on Vasai and the Serial link between Virar and Vasai, 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 = 1/1/1 ms
ISP#
```

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 = 1/3/8 ms
ISP#
```

l. 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 223.168.100.1
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       223.168.1.2              0             0 64512 i
*> 223.168.100.0    0.0.0.0                  0         32768 i
ISP#
```

i. At this point, the ISP router should be able to get to each network connected to Virar and Vasai from the Loopback address 223.168.100.1. Use the extended ping command and specify the source address of ISP Lo0 to test.

ISP# ping 172.16.1.1 source 223.168.100.1

```
ISP#ping 172.16.1.1 source 223.168.100.1

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 223.168.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/16 ms
ISP#
```

ISP# ping 172.16.32.1 source 223.168.100.1

```
ISP#ping 172.16.32.1 source 223.168.100.1

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 223.168.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/16 ms
```

ISP# ping 172.16.1.2 source 223.168.100.1

```
ISP#ping 172.16.1.2 source 223.168.100.1

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 223.168.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
ISP#
```

ISP# ping 172.16.64.1 source 223.168.100.1

```
ISP#ping 172.16.64.1 source 223.168.100.1

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 223.168.100.1
.....
Success rate is 0 percent (0/5)
ISP#
```

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 223.168.1.0 mask 255.255.255.252 ISP

(config-router) # network 223.168.1.4 mask 255.255.255.252

```
ISP#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP(config)#router bgp 200
ISP(config-router)#network 223.168.1.0 mask 255.255.255.252
ISP(config-router)#network 223.168.1.4 mask 255.255.255.252
ISP(config-router)#exit
ISP(config)#
```

k. Issue the show ip bgp command to verify that ISP is Correctly injecting its own WAN links int BGP.

ISP# show ip bgp

```
ISP#show ip bgp
BGP table version is 4, local router ID is 223.168.100.1
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       223.168.1.2              0             0 64512 i
*> 223.168.1.0/30   0.0.0.0                  0         32768 i
*> 223.168.100.0    0.0.0.0                  0         32768 i
ISP#
```

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

Vasai# show ip route

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

Gateway of last resort is not set

B    223.168.100.0/24 [20/0] via 223.168.1.1, 02:53:49
     223.168.1.0/30 is subnetted, 1 subnets
C       223.168.1.0 is directly connected, Serial0/0
     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 03:18:59, Serial0/1
Vasai#
```

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 Vasai. The only possible path from to ISP’s 223.168.100.0/24 through Virar.

ISP (config) #router bgp 200
ISP (config-router) #no network 223.168.1.0 mask 255.255.255.252 ISP

(config-router) #no network 223.168.1.4 mask 255.255.255.252

ISP (config-router) #exit
ISP (config) #interface serial 0/0/1
ISP (config-if) #shutdown

```
ISP#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP(config)#router bgp 200
ISP(config-router)#no network 223.168.1.0 mask 255.255.255.252
ISP(config-router)#no network 223.168.1.4 mask 255.255.255.252
ISP(config-router)#exit
ISP(config)#interface s0/1
ISP(config-if)#shutdown
ISP(config-if)#
*Mar  1 03:38:31.967: %BGP-5-ADJCHANGE: neighbor 223.168.1.2 Down Interface flap
ISP(config-if)#
*Mar  1 03:38:33.947: %LINK-5-CHANGED: Interface Serial0/1, changed state to administratively down
*Mar  1 03:38:34.947: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to down
ISP(config-if)#
```

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

Vasai# show ip bgp

```
Vasai#show ip bgp
BGP table version is 11, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 172.16.0.0       0.0.0.0                  0         32768 i
* i                 172.16.64.1              0    100      0 i
Vasai#
```

Vasai# show ip route

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

Gateway of last resort is not set

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 03:25:19, Serial0/1
Vasai#
```

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

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

```
Virar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Virar(config)#router bgp 64512
Virar(config-router)#neighbor 172.16.32.1 next-hop-self
Virar(config-router)#
Virar(config-router)#exit
Virar(config)#
```

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

```
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#router bgp 64512
Vasai(config-router)#neighbor 172.16.64.1 next-hop-self
Vasai(config-router)#
```

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

```
Virar#clear ip bgp *
Virar#
*Mar  1 03:45:14.915: %BGP-5-ADJCHANGE: neighbor 172.16.32.1 Down User reset
Virar#
*Mar  1 03:45:16.287: %BGP-5-ADJCHANGE: neighbor 172.16.32.1 Up
Virar#
```

Vasai# clear ip bgp *

```
Vasai#clear ip bgp *
Vasai#
*Mar  1 03:44:38.427: %BGP-5-ADJCHANGE: neighbor 172.16.64.1 Down User reset
Vasai#
*Mar  1 03:44:39.659: %BGP-5-ADJCHANGE: neighbor 172.16.64.1 Up
Vasai#
```

p. After the routers have returned to established BGP Speakers, issues the show ip bgp command on Vasai and notice that the next hop is now Virar instead of ISP.

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

   Network          Next Hop            Metric LocPrf Weight Path
* i172.16.0.0       172.16.64.1              0    100      0 i
*                   0.0.0.0                  0         32768 i
Vasai#
```

q. The show ip route command on Vasai now displays the 223.168.100.0/24 network because Virar is the next hop, 172.16.64.1, which is reachable from Vasai. **Vasai# show ip route**

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

Gateway of last resort is not set

     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 03:35:01, Serial0/1
Vasai#
```

r. Before configuring the next BGP attribute, restore the WAN link between ISP and Vasai. This will change the BGP table and routing table on both routers. For example, Vasai routing table shows 223.168.100.0/24 will now have a better path through ISP. **Vasai # show ip route**

```
Vasai#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

     223.168.1.0/30 is subnetted, 1 subnets
C       223.168.1.0 is directly connected, Serial0/0
     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 03:38:50, Serial0/1
Vasai#[~
*Mar  1 03:50:34.647: %BGP-5-ADJCHANGE: neighbor 223.168.1.1 Up
Vasai#[~
```

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 Virar and Vasai. This policy adjusts outbound traffic to prefer the link off the Virar router instead of the metered T1 off Vasai.

Virar(config) # route-map PRIMARY_T1_IN permit 10

Virar(config-route-map) # set local-preference 150

Virar(config-route-map) #exit

Virar(config) # router bgp 64512

Virar(config-router) #neighbor 223.168.1.5 route-map PRIMARY_T1_IN in

```
Virar(config)#route-map PRIMARY_T1_IN permit 10
Virar(config-route-map)#set local-preference 150
Virar(config-route-map)#exit
Virar(config)#router bgp 64512
Virar(config-router)#neighbor 223.168.1.5 route-map PRIMARY_T1_IN in
Virar(config-router)#
```

Vasai(config) # route-map SECONDARY_T1_IN permit 10

Vasai(config-route-map) # set local-preference 125

Vasai(config-router-map) #exit

Vasai(config) # router bgp 64512

Vasai(config-router) #neighbor 223.168.1.1 route-map SECONDARY_T1_IN in

```
Vasai(config)#route-map SECONDARY_T1_IN permit 10
Vasai(config-route-map)#set local-preference 125
Vasai(config-route-map)#exit
Vasai(config)#route bgp 64512
% Ambiguous command:  "route bgp 64512"
Vasai(config)#router bgp 64512
Vasai(config-router)#neighbor 223.168.1.1 route-map SECONDARY_T1_IN int
                                                                      ^
% Invalid input detected at '^' marker.

Vasai(config-router)#neighbor 223.168.1.1 route-map SECONDARY_T1_IN in
Vasai(config-router)#
```

t. Use the clear ip bgp * soft command after configuring this new policy. When the Conversations have been re-established, issue the show ip bgp command on Virar and Vasai.

Virar# clear ip bgp * soft

```
Virar#clear ip bgp * soft
Virar#
```

Vasai# clear ip bgp * soft

```
Vasai#clear ip bgp * soft
Vasai#
```

Virar# show ip bgp

```
Virar#show ip bgp
BGP table version is 4, 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
*>i223.168.100.0    172.16.32.1              0    125      0 200 i
Virar#
```

Vasai# show ip bgp

```
Vasai#show ip bgp
BGP table version is 4, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i172.16.0.0       172.16.64.1              0    100      0 i
*>                  0.0.0.0                  0         32768 i
*> 223.168.100.0    223.168.1.1              0    125      0 200 i
Vasai#
```

Step 9: BGP MED.

u. In this previous step we saw that Virar and Vasai will route traffic for 223.168.100.0/24 using the link between Virar and ISP. Examine what the return path ISP takes to reach AS 64512. Notice that the 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 7, local router ID is 223.168.100.1
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       223.168.1.2              0             0 64512 i
*> 223.168.100.0    0.0.0.0                  0         32768 i
ISP#
```

ISP# show ip route.

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

Gateway of last resort is not set

C    223.168.100.0/24 is directly connected, Loopback0
     223.168.1.0/30 is subnetted, 1 subnets
C       223.168.1.0 is directly connected, Serial0/1
B    172.16.0.0/16 [20/0] via 223.168.1.2, 00:17:14
ISP#
```

a. Use an extended ping command to verify this situation. Specify the record option and compare your output to following. Notice the return path using the exit interface 223.168.1.1 to Vasai.

```
Vasai#ping
Protocol [ip]:
Target IP address: 223.168.100.1
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 223.168.100.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.32.1
Packet has IP options:  Total option bytes= 39, padded length=40
 Record route: <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
```

```
   (0.0.0.0)
   (0.0.0.0)

Reply to request 0 (16 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (223.168.1.2)
   (223.168.100.1)
   (223.168.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list

Reply to request 1 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (223.168.1.2)
   (223.168.100.1)
   (223.168.1.1)
   (172.16.32.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list
```

```
 End of list

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

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 Vasai through s0/0/1, 172.16.1.2. then it at the s0/0/1 interface for Virar.

2. Virar s0/0/0, 223.168.1.6, routes the packet out to arrive at the s0/0/0 interface of ISP.

3. The target of 223.168.100.1 is reached: 223.168.100.1.

4. The packet is next forwarded out the s0/0/1, s0/0/1, 223.168.1.1 interface for ISP and arrives at the s0/0/0 interface for Vasai.
5. Vasai then forwards the packet out the last interface, loopback 0, 172.16.32.1.

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

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

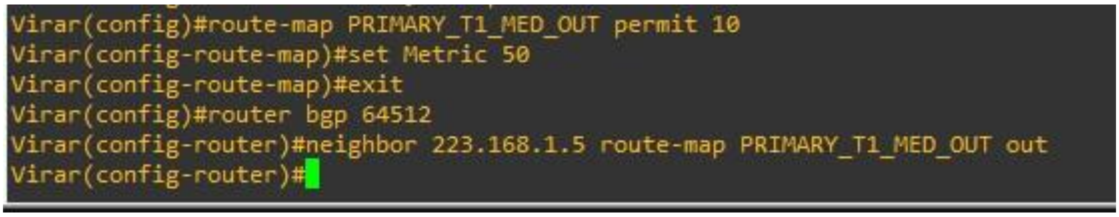
Virar(config) #route-map PRIMARY_T1_MED_OUT permit 10

Virar(config-route-map) #set Metric 50

Virar(config-router-map) # exit

Virar(config) #router bgp 64512

Virar(config-router) #neighbor 223.168.1.5 route-map PRIMARY_T1_MED_OUT out



```
Virar(config)#route-map PRIMARY_T1_MED_OUT permit 10
Virar(config-route-map)#set Metric 50
Virar(config-route-map)#exit
Virar(config)#router bgp 64512
Virar(config-router)#neighbor 223.168.1.5 route-map PRIMARY_T1_MED_OUT out
Virar(config-router)#
```

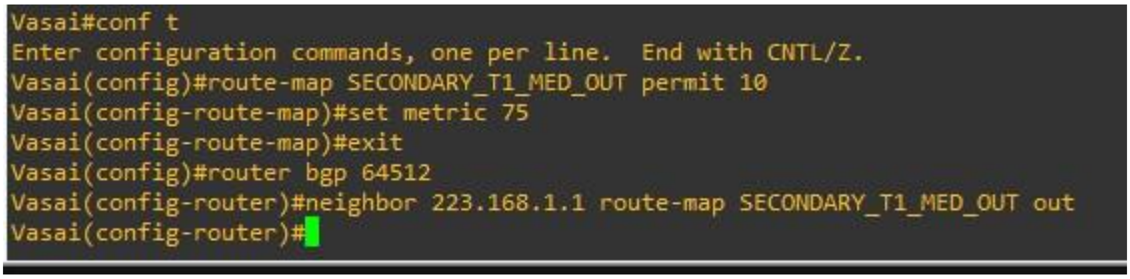
Vasai(config) #route-map SECONDARY_T1_MED_OUT permit 10

Vasai(config-route-map) # set metric 75

Vasai(config-route-map) # exit

Vasai(config) #router bgp 64512

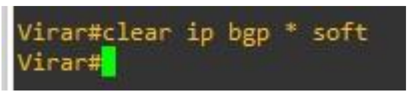
Vasai(config-router) #neighbor 223.168.1.1 route-map SECONDARY_T1_MED_OUT out



```
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#route-map SECONDARY_T1_MED_OUT permit 10
Vasai(config-route-map)#set metric 75
Vasai(config-route-map)#exit
Vasai(config)#router bgp 64512
Vasai(config-router)#neighbor 223.168.1.1 route-map SECONDARY_T1_MED_OUT out
Vasai(config-router)#
```

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

Virar# clear ip bgp * soft



```
Virar#clear ip bgp * soft
Virar#
```

Vasai# clear ip bgp * soft

```
Vasai#clear ip bgp * soft
Vasai#
```

Virar# show ip bgp

```
Virar#show ip bgp
BGP table version is 4, 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
*>i223.168.100.0    172.16.32.1              0    125      0 200 i
Virar#
```

Vasai# show ip bgp

```
Vasai#show ip bgp
BGP table version is 4, local router ID is 172.16.32.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i172.16.0.0       172.16.64.1              0    100      0 i
*>                  0.0.0.0                  0         32768 i
*> 223.168.100.0    223.168.1.1              0    125      0 200 i
Vasai#
```

Reissue an extended ping command with the record command. Notice the change in return path using the exit interface 223.168.1.5 to Virar.

```
Virar#ping
Protocol [ip]:
Target IP address:
% Bad IP address
Virar#ping
Protocol [ip]:
Target IP address: 223.168.100.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 172.16.32.1
% Invalid source. Must use IP address or full interface name without spaces (e.g. Serial0/1)
Source address or interface: 172.16.32.1
% Invalid source. Must use IP address or full interface name without spaces (e.g. Serial0/1)
Source address or interface: 172.16.32.1
% Invalid source. Must use IP address or full interface name without spaces (e.g. Serial0/1)
Source address or interface:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
```

```
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 223.168.100.1, timeout is 2 seconds:
Packet has IP options:  Total option bytes= 39, padded length=40
 Record route: <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)

Reply to request 0 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.1)
   (223.168.1.2)
   (223.168.100.1)
   (223.168.1.1)
   (172.16.1.2)
   (172.16.1.1) <*>
```

```
   (172.16.1.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list

Reply to request 4 (1 ms).  Received packet has options
 Total option bytes= 40, padded length=40
 Record route:
   (172.16.1.1)
   (223.168.1.2)
   (223.168.100.1)
   (223.168.1.1)
   (172.16.1.2)
   (172.16.1.1) <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
 End of list

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/28 ms
Virar#
```

ISP# show ip bgp

```
ISP#show ip bgp
BGP table version is 8, local router ID is 223.168.100.1
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       223.168.1.2             75             0 64512 i
*> 223.168.100.0    0.0.0.0                  0         32768 i
ISP#
```

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 Virar and Vasai 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 will be used to test the default route on Virar and Vasai.

ISP (config)# router bgp 200

ISP (config-router) #neighbor 223.168.1.6 default-originate

ISP (config-router) #neighbor 223.168.1.2 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#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP(config)#router bgp 200
ISP(config-router)#neighbor 223.168.1.6 default-originate
ISP(config-router)#neighbor 223.168.1.2 default-originate
ISP(config-router)#exit
ISP(config)#interface Loopback 10
ISP(config-if)#
*Mar  1 04:43:41.206: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback10, changed state to up
ISP(config-if)#ip address 10.0.0.1 255.255.255.0
ISP(config-if)#
```

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

Virar# show ip route

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

Gateway of last resort is 172.16.32.1 to network 0.0.0.0

B    223.168.100.0/24 [200/0] via 172.16.32.1, 00:49:10
     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
D       172.16.32.0/24 [90/2297856] via 172.16.1.2, 04:28:38, Serial0/1
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
C       172.16.64.0/24 is directly connected, Loopback0
B*   0.0.0.0/0 [200/0] via 172.16.32.1, 00:01:07
Virar#
```

Vasai# show ip route

```
Vasai#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 223.168.1.1 to network 0.0.0.0

B    223.168.100.0/24 [20/0] via 223.168.1.1, 00:50:52
     223.168.1.0/30 is subnetted, 1 subnets
C       223.168.1.0 is directly connected, Serial0/0
     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 04:29:55, Serial0/1
B*   0.0.0.0/0 [20/0] via 223.168.1.1, 00:02:49
Vasai#
```

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

Vasai# show ip bgp

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

   Network          Next Hop            Metric LocPrf Weight Path
*> 0.0.0.0          223.168.1.1              0    125      0 200 i
* i172.16.0.0       172.16.64.1              0    100      0 i
*>                  0.0.0.0                  0         32768 i
*> 223.168.100.0    223.168.1.1              0    125      0 200 i
Vasai#
```

d. Using the traceroute command verify that packets to 10.0.01 is using the default route through Virar.

Vasai# traceroute 10.0.0.1

```
Vasai#traceroute 10.0.0.1

Type escape sequence to abort.
Tracing the route to 10.0.0.1

  1 223.168.1.1 [AS 200] 0 msec 0 msec 0 msec
Vasai#
```

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

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

```
ISP(config)#interface s0/0
ISP(config-if)#
```

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

Virar# show ip route

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

Gateway of last resort is 172.16.32.1 to network 0.0.0.0

B    223.168.100.0/24 [200/0] via 172.16.32.1, 00:55:13
     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
D       172.16.32.0/24 [90/2297856] via 172.16.1.2, 04:34:40, Serial0/1
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
C       172.16.64.0/24 is directly connected, Loopback0
B*   0.0.0.0/0 [200/0] via 172.16.32.1, 00:07:09
Virar#
```

Vasai# show ip route

```
Vasai#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 223.168.1.1 to network 0.0.0.0

B    223.168.100.0/24 [20/0] via 223.168.1.1, 00:55:57
     223.168.1.0/30 is subnetted, 1 subnets
C       223.168.1.0 is directly connected, Serial0/0
     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.32.0/24 is directly connected, Loopback0
S       172.16.0.0/16 is directly connected, Null0
C       172.16.1.0/24 is directly connected, Serial0/1
D       172.16.64.0/24 [90/2297856] via 172.16.1.1, 04:35:01, Serial0/1
B*   0.0.0.0/0 [20/0] via 223.168.1.1, 00:07:56
Vasai#
```

g. **Verify the new path using the traceroute command to 10.0.0.1 from Virar. Notice default route is now through Vasai.**

Vasai# trace 10.0.0.1

```
Vasai#trace 10.0.0.1

Type escape sequence to abort.
Tracing the route to 10.0.0.1

  1 223.168.1.1 [AS 200] 0 msec 0 msec 0 msec
Vasai#
```

Practical No 4

Aim: Secure Management Plane

Theory: The management plane is responsible for managing a network device — such as configuring settings, monitoring status, and maintaining security. It allows administrators to communicate with the router or switch through command-line interfaces (CLI) or graphical interfaces.

Examples of management plane protocols:
- Telnet (insecure, plaintext)
- SSH (secure)
- SNMP
- HTTP/HTTPS
- Console/VTY access

The management plane is a critical attack surface. If compromised, attackers can:
- View or change device configurations
- Shut down interfaces
- Redirect traffic
- Launch attacks on other devices

Hence, securing it is essential for protecting the entire network infrastructure.

Methods to Secure the Management Plane

1. Use Secure Access Protocols
   Replace insecure protocols like Telnet with SSH, which provides encrypted access to the CLI.

2. Implement User Authentication
   Configure local usernames and passwords, and set privilege levels to limit what users can do.

3. Use Access Control Lists (ACLs)
   Limit access to management interfaces by allowing only authorized IP addresses to connect.

4. Configure Strong Passwords & Encryption
   Use `enable secret`, `service password-encryption`, and avoid weak or default credentials.

5. Disable Unused Services
   Turn off services like CDP, HTTP, FTP, and bootp that are not in use to reduce the attack surface.

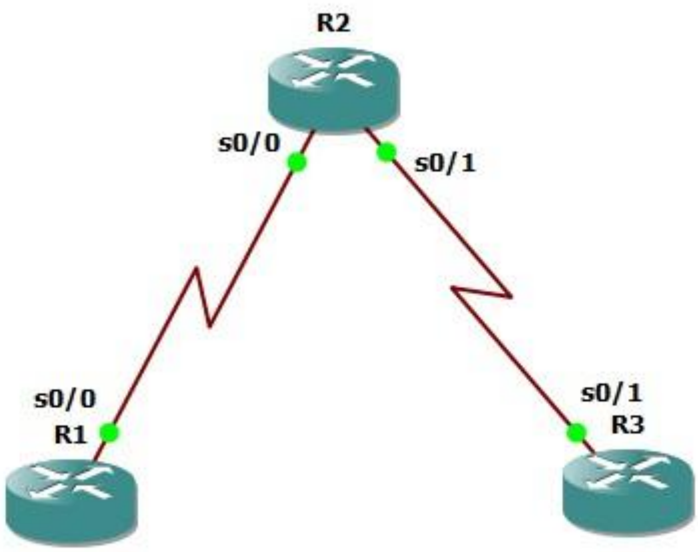
6. Display Banner Warnings
   Configure `banner motd` and `banner login` to display legal notices or warnings to deter unauthorized users.

7. Verify and Test the Setup
   Always test your configuration using commands like `show running-config`, `show ip ssh`, `show access-lists`, and attempt authorized and unauthorized access.

Real-World Importance

In enterprise networks, securing the management plane ensures:
- Only authorized personnel can make changes
- Devices are not compromised remotely
- Regulatory compliance with security standards (like ISO, NIST, etc.)
- Reduced risk of configuration tampering, insider threats, and cyber attacks

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 each router to clear previous configurations. Using the addressing scheme in the diagram, apply the IP addresses to the interfaces on the R1, R2, and R3 routers. You can copy and paste the following configurations into your routers to begin. **Router 1 interface Loopback 0 ip address 223.168.1.1 255.255.255.0**

exit interface Serial0/0/0 ip address

10.1.1.1 255.255.255.252 no shutdown

exit

end

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface Loopback 0
R1(config-if)#
*Mar  1 00:01:05.851: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R1(config-if)#ip address 223.168.1.1 255.255.255.0
R1(config-if)#exit
R1(config)#interface s0/0
R1(config-if)#ip address 10.1.1.1 255.255.255.252
R1(config-if)#no shutdown
R1(config-if)#
*Mar  1 00:02:19.879: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R1(config-if)#
*Mar  1 00:02:20.883: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R1(config-if)#exit
R1(config)#end
R1#
*Mar  1 00:02:28.359: %SYS-5-CONFIG_I: Configured from console by console
R1#
*Mar  1 00:02:42.187: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to down
R1#
```

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

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface s0/0
R2(config-if)#ip address 10.1.1.2 255.255.255.252
R2(config-if)#no shutdown
R2(config-if)#
*Mar  1 00:02:45.767: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R2(config-if)#ex
*Mar  1 00:02:46.771: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0, changed state to up
R2(config-if)#exit
R2(config)#interface s0/1
R2(config-if)#ip address 10.2.2.1 255.255.255.252
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#
*Mar  1 00:03:53.599: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
R2(config)#
*Mar  1 00:03:54.603: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
R2(config)#end
R2#
*Mar  1 00:03:58.319: %SYS-5-CONFIG_I: Configured from console by console
R2#
```

Router R3 interface Loopback0 ip address

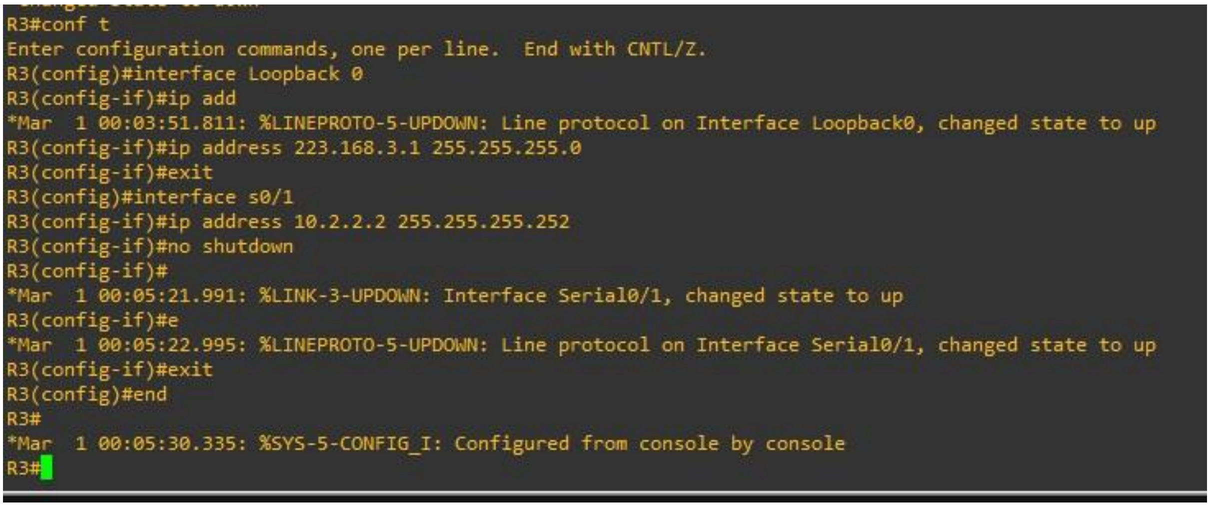
223.168.3.1 255.255.255.0

exit interface Serial0/0/1 ip address

10.2.2.2 255.255.255.252 no shutdown

exit

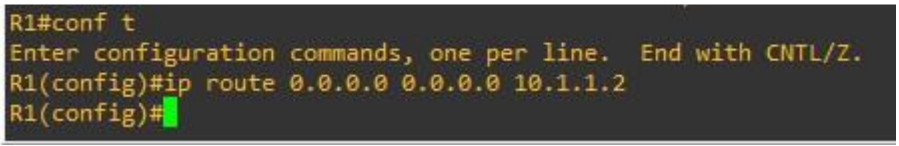
end



```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#interface Loopback 0
R3(config-if)#ip add
*Mar  1 00:03:51.811: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R3(config-if)#ip address 223.168.3.1 255.255.255.0
R3(config-if)#exit
R3(config)#interface s0/1
R3(config-if)#ip address 10.2.2.2 255.255.255.252
R3(config-if)#no shutdown
R3(config-if)#
*Mar  1 00:05:21.991: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
R3(config-if)#e
*Mar  1 00:05:22.995: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1, changed state to up
R3(config-if)#exit
R3(config)#end
R3#
*Mar  1 00:05:30.335: %SYS-5-CONFIG_I: Configured from console by console
R3#
```

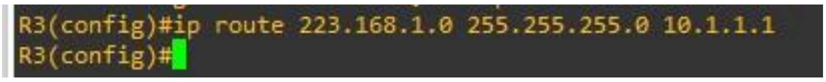
Step 2: Configure static routes.

R1(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.2



```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip route 0.0.0.0 0.0.0.0 10.1.1.2
R1(config)#
```

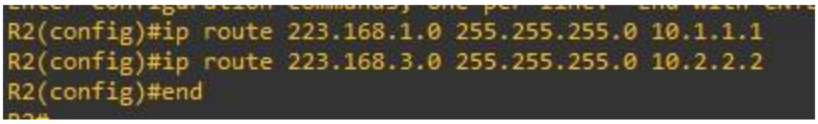
R3(config)# ip route 0.0.0.0 0.0.0.0 10.2.2.1



```
R3(config)#ip route 223.168.1.0 255.255.255.0 10.1.1.1
R3(config)#
```

R2(config)# ip route 223.168.1.0 255.255.255.0 10.1.1.1 R2(config)# ip route

223.168.3.0 255.255.255.0 10.2.2.2



```
R2(config)#ip route 223.168.1.0 255.255.255.0 10.1.1.1
R2(config)#ip route 223.168.3.0 255.255.255.0 10.2.2.2
R2(config)#end
```

tclsh foreach address {

223.168.1.1

10.1.1.1

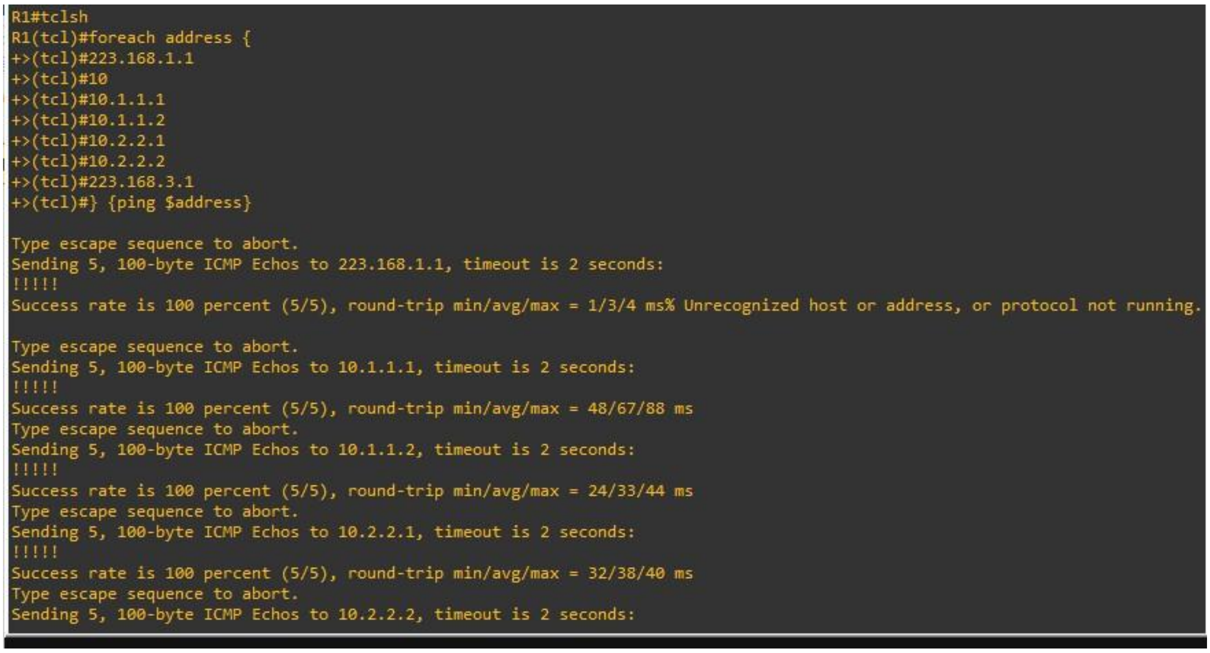
10.1.1.2

10.2.2.1

10.2.2.2

223.168.3.1}

{ping $address}



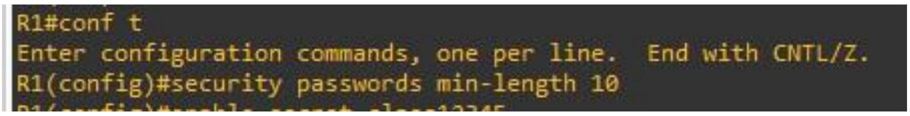
```
R1#tclsh
R1(tcl)#foreach address {
+>(tcl)#223.168.1.1
+>(tcl)#10
+>(tcl)#10.1.1.1
+>(tcl)#10.1.1.2
+>(tcl)#10.2.2.1
+>(tcl)#10.2.2.2
+>(tcl)#223.168.3.1
+>(tcl)#} {ping $address}

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 223.168.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms% Unrecognized host or address, or protocol not running.

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/67/88 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/33/44 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 = 32/38/40 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
```

Step 3: Secure management access.

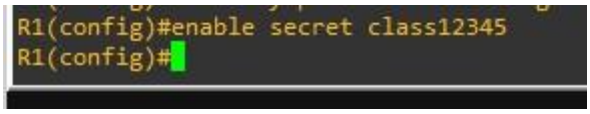
1. On R1, use the security passwords command to set a minimum password length of 10 characters.
2. **R1(config)# security passwords min-length 10**



```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#security passwords min-length 10
```

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

R1(config)# enable secret class12345.



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

4. Configure a console password and enable login for routers. For additional security, the exectimeout 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

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

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

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

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

R1(config)# line aux 0

R1(config-line) # no exec

R1(config-line) # end

```
R1(config)#line aux 0
R1(config-line)#no exec
R1(config-line)#end
R1#
*Mar  1 00:30:15.283: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

6. 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)#service password-encryption
R1(config)#
```

7. 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.

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 Shivam secret class12345

R1(config)# username Shivam secret class54321

```
R1(config)#username MScIT secret class12345
R1(config)#username MScIT secret class5432
% Password too short - must be at least 10 characters. Password configuration failed
R1(config)#username MScIT secret class543212
R1(config)#
```

4. 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

R1(config)# line vty 0 4

R1(config-line) # login local

R1(config-line) # end

```
R1(config)#line console 0
R1(config-line)#login local
R1(config-line)#exit
R1(config)#line vty 0 4
R1(config-line)#login local
R1(config-line)#end
R1#
*Mar  1 00:37:11.083: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

Practical No-5

Aim: Configure and Verify Path Control Using PBR

Theory:
In traditional IP routing, routers forward packets based only on the destination IP address, using the longest prefix match from their routing table.

Policy-Based Routing (PBR) allows you to override this behavior by defining custom routing policies based on other criteria like:

1) Source IP address
2) Protocol
3) Packet size
4) Incoming interface

This gives network administrators greater flexibility in managing traffic flow based on business or technical policies.

PBR is commonly used when you need to:

1) Route specific users or devices through different ISPs or firewalls
2) Apply different Quality of Service (QoS) or bandwidth policies to selected traffic
3) Bypass certain links for sensitive data
4) Perform load balancing or failover routing between multiple paths

PBR uses three main configuration components on a router:

1) Access Control List (ACL): Used to match the traffic that should be treated differently (e.g., match source IPs).
2) Route Map: Defines the policy for matched traffic, such as setting a specific next-hop IP.
3) IP Policy Statement: Applied to an interface using the ip policy route-map command, telling the router to evaluate and apply the route map to incoming traffic on that interface.

Example Use Case
You have two exit routers (R2 and R3).
You want a specific PC (e.g., 192.168.1.2) to reach the internet only through R2, even though R3 is the default route.
PBR allows you to match traffic from that PC and force it through R2, regardless of the routing table.

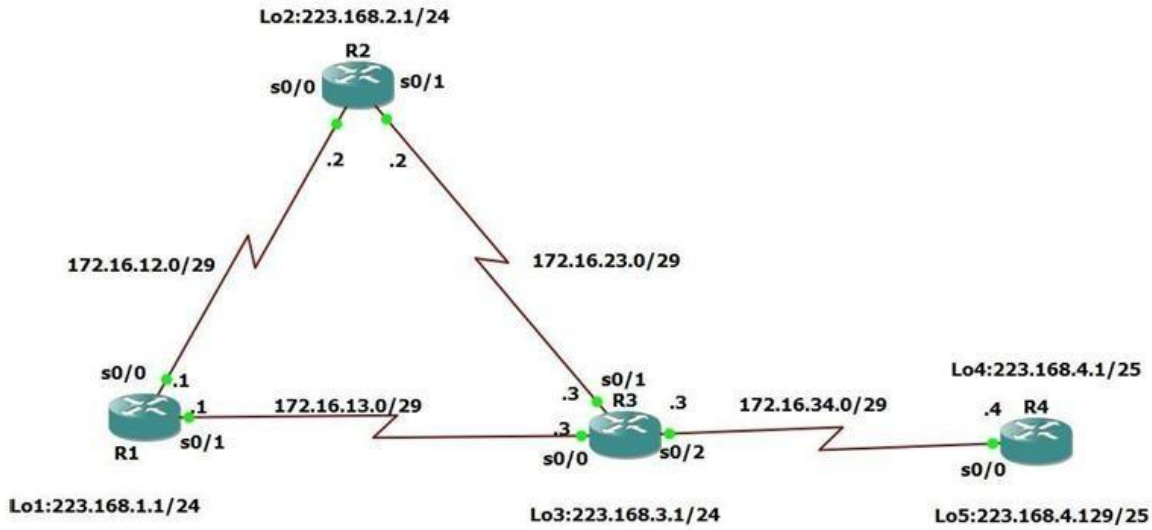
Benefits of PBR

1) Granular control over traffic flows
2) Can enforce security, compliance, and routing policies
3) Useful in multi-homed networks (with multiple ISPs)

Limitations of PBR

1) It only applies to incoming packets on the interface where the policy is configured.
2) Requires careful design to avoid routing loops or black holes
3) Adds some processing overhead on routers (not recommended on low-end devices for high-volume traffic)

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 reload each 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 to R3, specify the bandwidth as 128 Kb/s and set a clock rate on the DCE using the clock rate 128000 command**.**

Router R1

Interface Lo1
ip address 223.168.1.1 255.255.255.0 interface
Serial 0/0
ip address 172.16.12.1 255.255.255.248
no shutdown interface Serial 0/1
ip address 172.16.13.1 255.255.255.248
no shutdown
output

R1

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int lo1
R1(config-if)#i
*Mar  1 00:00:32.471: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1,
 changed state to up
R1(config-if)#ip address 223.168.1.1 255.255.255.0
R1(config-if)#int s0/0
R1(config-if)#ip add 172.16.12.1 255.255.255.248
R1(config-if)#no sh
R1(config-if)#exit
R1(config)#
*Mar  1 00:01:13.391: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R1(config)#
*Mar  1 00:01:14.395: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0,
 changed state to up
R1(config)#int s0/1
R1(config-if)#ip add 172.16.13.1 255.255.255.248
R1(config-if)#no sh
```

Router R2

Interface Lo2
ip address 223.168.2.1 255.255.255.0 interface
Serial 0/0
ip address 172.16.12.2 255.255.255.248
no shutdown
interface Serial 0/1
ip address 172.16.23.2 255.255.255.248
no shutdown

R2

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int lo1
R2(config-if)#ip add
*Mar  1 00:03:26.759: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1,
 changed state to up
R2(config-if)#int lo2
R2(config-if)#int lo2
*Mar  1 00:03:41.319: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback2,
 changed state to up
R2(config-if)#ip add 223.168.2.1 255.255.255.0
R2(config-if)#int s0/0
R2(config-if)#ip add 172.16.12.2 255.255.255.248
R2(config-if)#no sh
R2(config-if)#
*Mar  1 00:04:36.063: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R2(config-if)#
*Mar  1 00:04:37.067: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0,
 changed state to up
R2(config-if)#int s0/1
R2(config-if)#ip add 172.16.23.2 255.255.255.248
R2(config-if)#no sh
```

Router R3

Interface Lo3
ip address 223.168.3.1 255.255.255.0 interface
Serial 0/0
ip address 172.16.13.3 255.255.255.248 no
shutdown

interface Serial 0/1
ip address 172.16.23.3 255.255.255.248 interface
Serial 0/2
ip address 172.16.34.3 255.255.255.248 no
shutdown

R3

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#int lo3
R3(config-if)#
*Mar  1 00:03:33.671: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback3,
 changed state to up
R3(config-if)#ip add 223.168.3.1 255.255.255.0
R3(config-if)#int s0/0
R3(config-if)#ip add 172.16.13.3 255.255.255.248
R3(config-if)#no sh
R3(config-if)#int s0/1
R3(config-if)#
*Mar  1 00:04:12.335: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
R3(config-if)#ip
*Mar  1 00:04:13.339: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0,
 changed state to up
R3(config-if)#ip add 172.16.23.3 255.255.255.248
R3(config-if)#no sh
R3(config-if)#int s0/2
*Mar  1 00:04:29.483: %LINK-3-UPDOWN: Interface Serial0/1, changed state to up
R3(config-if)#int s0/2
R3(config-if)#
*Mar  1 00:04:30.487: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1,
 changed state to up
R3(config-if)#ip add 172.16.34.3 255.255.255.248
R3(config-if)#no sh
R3(config-if)#ex
*Mar  1 00:04:52.815: %LINK-3-UPDOWN: Interface Serial0/2, changed state to up
R3(config-if)#exit
```

Router R4

Interface Lo4
ip address 223.168.4.1 255.255.255.128 interface
Lo5
ip address 223.168.4.129 255.255.255.128 interface
Serial 0/0
ip address 172.16.34.4 255.255.255.248 no
shutdown

R4

```
 changed state to down
R4#
R4#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R4(config)#int lo4
R4(config-if)#
*Mar  1 00:04:14.627: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback4,
 changed state to up
R4(config-if)#ip add 223.168.4.1 255.255.255.128
R4(config-if)#exit
R4(config)#int lo5
R4(config-if)#ip add
*Mar  1 00:04:56.379: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback5,
 changed state to up
R4(config-if)#ip add 223.168.4.129 255.255.255.128
R4(config-if)#exit
R4(config)#int s0/0
R4(config-if)#ip add 172.16.34.4 255.255.255.248
R4(config-if)#no sh
```

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                Prot
ocol
FastEthernet0/0            unassigned      YES unset  administratively down down

Serial0/0                  172.16.13.3     YES manual up                    up

FastEthernet0/1            unassigned      YES unset  administratively down down

Serial0/1                  172.16.23.3     YES manual up                    up

Serial0/2                  172.16.34.3     YES manual up                    up

Serial0/3                  unassigned      YES unset  administratively down down

Serial0/4                  unassigned      YES unset  administratively down down

Serial0/5                  unassigned      YES unset  administratively down down

Serial1/0                  unassigned      YES unset  administratively down down

Serial1/1                  unassigned      YES unset  administratively down down

Serial1/2                  unassigned      YES unset  administratively down down

Serial1/3                  unassigned      YES unset  administratively down down

Serial2/0                  unassigned      YES unset  administratively down down

Serial2/1                  unassigned      YES unset  administratively down down

Serial2/2                  unassigned      YES unset  administratively down down

Serial2/3                  unassigned      YES unset  administratively down down

Loopback3                  223.168.3.1     YES manual up                    up
```

R3# show protocols

```
R3#show protocols
Global values:
  Internet Protocol routing is enabled
FastEthernet0/0 is administratively down, line protocol is down
Serial0/0 is up, line protocol is up
  Internet address is 172.16.13.3/29
FastEthernet0/1 is administratively down, line protocol is down
Serial0/1 is up, line protocol is up
  Internet address is 172.16.23.3/29
Serial0/2 is up, line protocol is up
  Internet address is 172.16.34.3/29
Serial0/3 is administratively down, line protocol is down
Serial0/4 is administratively down, line protocol is down
Serial0/5 is administratively down, line protocol is down
Serial1/0 is administratively down, line protocol is down
Serial1/1 is administratively down, line protocol is down
Serial1/2 is administratively down, line protocol is down
Serial1/3 is administratively down, line protocol is down
Serial2/0 is administratively down, line protocol is down
Serial2/1 is administratively down, line protocol is down
Serial2/2 is administratively down, line protocol is down
Serial2/3 is administratively down, line protocol is down
Loopback3 is up, line protocol is up
  Internet address is 223.168.3.1/24
```

R3# show interface description

```
R3#show interfaces description
Interface                      Status         Protocol Description
Fa0/0                          admin down     down
Se0/0                          up             up
Fa0/1                          admin down     down
Se0/1                          up             up
Se0/2                          up             up
Se0/3                          admin down     down
Se0/4                          admin down     down
Se0/5                          admin down     down
Se1/0                          admin down     down
Se1/1                          admin down     down
Se1/2                          admin down     down
Se1/3                          admin down     down
Se2/0                          admin down     down
Se2/1                          admin down     down
Se2/2                          admin down     down
Se2/3                          admin down     down
Lo3                            up             up
```

Step 3: Configure basic EIGRP.

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

b. Advertise networks 172.16.12.0/29, 172.16.13.0/29, 172.16.23.0/29, 172.16.34.0/29, 223.168.1.0/24, 223.168.2.0/24, 223.168.3.0/24, and 223.168.4.0/24 from their respective routers. **Router R1** Router eigrp 1
network 223.168.1.0 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  223.168.1.0
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 223.168.2.0
network 172.16.12.0
0.0.0.7 network
172.16.23.0 0.0.07 no
auto-summary

```
R2(config)#router eigrp 1
R2(config-router)#network 223.168.2.0
R2(config-router)#network 172.16.12.0 0.0.0.7
R2(config-router)#
*Mar  1 00:13:26.275: %DUAL-5-NBRCHANGE: IP-EIG
acency
R2(config-router)#network 172.16.23.0 0.0.0.7
R2(config-router)#no auto-summary
```

Router R3 Router eigrp 1
network 223.168.3.0
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#conf t
Enter configuration commands, one per line.  Er
R3(config)#router eigrp 1
R3(config-router)#network 223.168.3.0
R3(config-router)#network 172.16.13.0 0.0.0.7
R3(config-router)#
*Mar  1 00:12:43.067: %DUAL-5-NBRCHANGE: IP-EIG
w adjacency
R3(config-router)#network 172.16.23.0 0.0.0.7
R3(config-router)#
*Mar  1 00:12:50.371: %DUAL-5-NBRCHANGE: IP-EIG
w adjacency
R3(config-router)#network 172.16.34.0 0.0.0.7
R3(config-router)#no auto-summary
```

Router R4 router eigrp 1
network 223.168.4.0
network 172.16.34.0
0.0.07 no auto-summary

```
R4(config-if)#router eigrp 1
R4(config-router)#network 223.168.4.0
R4(config-router)#network 172.16.34.0 0.0.0.7
R4(config-router)#
*Mar  1 00:12:45.423: %DUAL-5-NBRCHANGE: IP-EI
R4(config-router)#no auto-summary
```

Step 4: Verify EIGRP connectivity.

c. 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             Se0/1             11 00:01:56   34   204  0  29
0   172.16.12.2             Se0/0             13 00:03:23   26   200  0  24
```

R2# show ip eigrp neighbors

```
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             Se0/1             13 00:02:39   36   216  0  28
0   172.16.12.1             Se0/0             10 00:04:15   32   200  0  26
```

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             Se0/2             12 00:01:55   40   240  0  7
1   172.16.23.2             Se0/1             11 00:02:57   34   204  0  25
0   172.16.13.1             Se0/0             10 00:03:04   33   200  0  27
```

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             Se0/0             12 00:02:08   33   200  0  32
```

d. Run the following TCL Scripts on all routers to verify full connectivity.

```
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)#223.168.1.1
+>(tcl)#223.168.2.1
+>(tcl)#223.168.3.1
+>(tcl)#223.168.4.1
+>(tcl)#223.168.4.129
+>(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 = 64/72/104 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 = 32/32/32 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.1, timeout is 2 seconds:
!!!!!
```

Step 5: Verify the current path.

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

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

```
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 not set

D    223.168.2.0/24 [90/2297856] via 172.16.12.2, 00:08:22, Serial0/0
D    223.168.3.0/24 [90/2297856] via 172.16.13.3, 00:08:22, Serial0/1
C    223.168.1.0/24 is directly connected, Loopback1
     223.168.4.0/25 is subnetted, 2 subnets
D       223.168.4.0 [90/2809856] via 172.16.13.3, 00:07:16, Serial0/1
D       223.168.4.128 [90/2809856] via 172.16.13.3, 00:07:16, Serial0/1
     172.16.0.0/29 is subnetted, 4 subnets
D       172.16.34.0 [90/2681856] via 172.16.13.3, 00:08:13, Serial0/1
D       172.16.23.0 [90/2681856] via 172.16.13.3, 00:08:24, Serial0/1
                    [90/2681856] via 172.16.12.2, 00:08:24, Serial0/0
C       172.16.12.0 is directly connected, Serial0/0
C       172.16.13.0 is directly connected, Serial0/1
```

R4# traceroute 223.168.1.1 source 223.168.4.1

```
R4#traceroute 223.168.1.1 source 223.168.4.1

Type escape sequence to abort.
Tracing the route to 223.168.1.1

  1 172.16.34.3 32 msec 28 msec 28 msec
  2 172.16.13.1 32 msec 32 msec 32 msec
```

R4# traceroute 223.168.1.1 source 223.168.4.129

```
R4#traceroute 223.168.1.1 source 223.168.4.129

Type escape sequence to abort.
Tracing the route to 223.168.1.1

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

On R3, use the show ip route command and note that the preferred route from R3 to R1 LAN 223.168.1.0/24 is via R2 using the R3 exit interface S0/0/1

R3# show ip route

```
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             Se0/2             12 00:01:55   40    240  0  7
1   172.16.23.2             Se0/1             11 00:02:57   34    204  0  25
0   172.16.13.1             Se0/0             10 00:03:04   33    200  0  27
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 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

D    223.168.2.0/24 [90/2297856] via 172.16.23.2, 00:10:08, Serial0/1
C    223.168.3.0/24 is directly connected, Loopback3
D    223.168.1.0/24 [90/2297856] via 172.16.13.1, 00:10:08, Serial0/0
     223.168.4.0/25 is subnetted, 2 subnets
D       223.168.4.0 [90/2297856] via 172.16.34.4, 00:08:59, Serial0/2
D       223.168.4.128 [90/2297856] via 172.16.34.4, 00:08:59, Serial0/2
     172.16.0.0/29 is subnetted, 4 subnets
C       172.16.34.0 is directly connected, Serial0/2
C       172.16.23.0 is directly connected, Serial0/1
D       172.16.12.0 [90/2681856] via 172.16.23.2, 00:10:10, Serial0/1
                    [90/2681856] via 172.16.13.1, 00:10:10, Serial0/0
C       172.16.13.0 is directly connected, Serial0/0
```

f. On R3, use the show interfaces serial 0/0/0 and show interfaces s0/0/1 command.

```
R3#show int s0/0
Serial0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 172.16.13.3/29
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:03, output 00:00:01, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/1/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1158 kilobits/sec
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     416 packets input, 28570 bytes, 0 no buffer
     Received 173 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     347 packets output, 24818 bytes, 0 underruns
     0 output errors, 0 collisions, 6 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
     DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
```

g. Confirm that R3 has a valid route to reach R1 from its serial 0/0/0 interface using the show ip eigrp topology 223.168.1.0 command.

R3# show ip eigrp topology 223.168.1.0

```
R3#show ip eigrp topology 223.168.1.0
IP-EIGRP (AS 1): Topology entry for 223.168.1.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2297856
  Routing Descriptor Blocks:
  172.16.13.1 (Serial0/0), 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 (Serial0/1), from 172.16.23.2, Send flag is 0x0
      Composite metric is (2809856/2297856), Route is Internal
      Vector metric:
        Minimum bandwidth is 1544 Kbit
        Total delay is 45000 microseconds
        Reliability is 255/255
        Load is 1/255
        Minimum MTU is 1500
        Hop count is 2
```

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.

h. 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 223.168.4.128 0.0.0.127
R3(config-std-nacl) #exit

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#ip access-list standard PBR-ACL
R3(config-std-nacl)#remark ACL matches R4 LAN B traffic
R3(config-std-nacl)#permit 223.168.4.128 0.0.0.127
R3(config-std-nacl)#exit
```

i. 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)#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
```

j. 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 s0/2
R3(config-if)#ip policy route-map R3-to-R1
R3(config-if)#end
```

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

```
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

1. 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 223.168.4.0 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 223.168.4.0 0.0.0.255
R3(config)#exit
```

2. Enable PBR debugging only for traffic that matches the R4 LANs. **R3# debug ip policy?**

```
R3#debug ip policy ?
  dynamic  dynamic PBR
```

R3# debug ip policy 1

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

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

R4# traceroute 223.168.1.1 source 223.168.4.1

```
R4#traceroute 223.168.1.1 source 223.168.4.1

Type escape sequence to abort.
Tracing the route to 223.168.1.1

  1 172.16.34.3 32 msec 28 msec 28 msec
  2 172.16.13.1 32 msec 32 msec 32 msec
```

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

R4# traceroute 223.168.1.1 source 223.168.4.129

```
R4#traceroute 223.168.1.1 source 223.168.4.129

Type escape sequence to abort.
Tracing the route to 223.168.1.1

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

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

```
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: 3 packets, 96 bytes
```

Practical No 6

Aim: Simulating MPLS environment

and Simulating VRF

Theory: 1. IP SLA (IP Service Level Agreement)
IP SLA is a Cisco feature used to measure network performance and monitor service levels by generating synthetic traffic between network devices. It helps in:

Measuring latency, jitter, packet loss, and availability

Verifying QoS (Quality of Service)

Troubleshooting network paths

Tracking reachability to trigger dynamic routing changes

How it works:
IP SLA sends test traffic (like ICMP, UDP, or HTTP) to a target IP address.

It records statistics like round-trip time and availability.

You can schedule these tests and track trends over time.

Remote SPAN (RSPAN)
Remote SPAN extends the Switched Port Analyzer (SPAN) feature to span traffic across multiple switches in a campus environment. It is used to:

Monitor traffic on remote switches

Capture packets from a source port/VLAN to a central analyzer

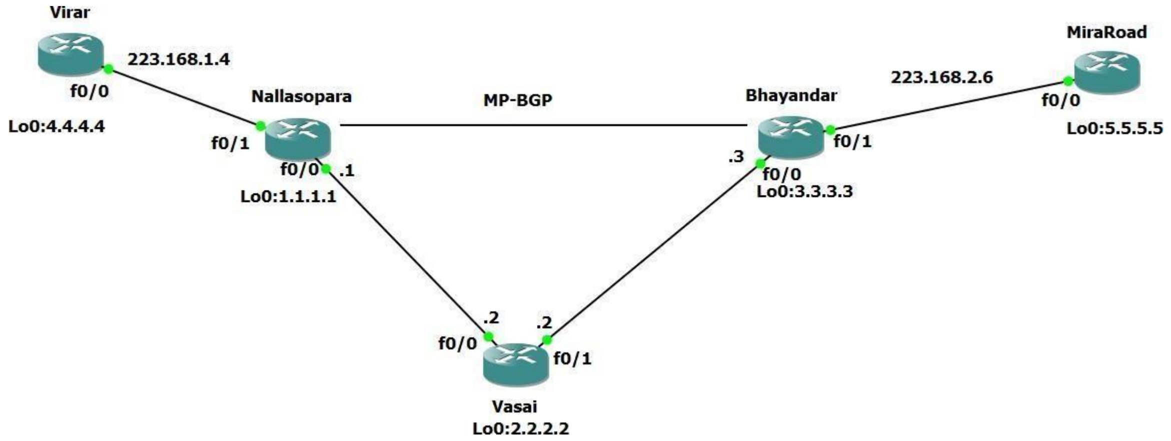
Troubleshoot performance or security issues centrally

How it works:
You configure a special RSPAN VLAN to carry mirrored traffic.

Source switches send mirrored traffic to this VLAN.

The destination switch receives mirrored traffic on an RSPAN port where an analyzer or packet capture tool is connected.

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 R1 and R3

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#int lo0
Nallasopara(config-if)#
*Mar  1 00:02:57.655: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
Nallasopara(config-if)#ip add 1.1.1.1 255.255.255
% Incomplete command.

Nallasopara(config-if)#ip add 1.1.1.1 255.255.255.255
Nallasopara(config-if)#ip ospf 1 area 0
Nallasopara(config-if)#int f0/0
Nallasopara(config-if)#ip add 10.0.01 255.255.255.0
                                     ^
% Invalid input detected at '^' marker.

Nallasopara(config-if)#ip add 10.0.01 255.255.255.0
                                     ^
% Invalid input detected at '^' marker.

Nallasopara(config-if)#ip add 10.0.0.1 255.255.255.0
Nallasopara(config-if)#no shut
Nallasopara(config-if)#
*Mar  1 00:09:04.275: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 00:09:05.275: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
Nallasopara(config-if)#ospf 1 area 0
                       ^
% Invalid input detected at '^' marker.

Nallasopara(config-if)#ip ospf 1 area 0
Nallasopara(config-if)#
```

```
Vasai#
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#int lo0
Vasai(config-if)#ip add 2.2.2.2 255.255.255.255
Vasai(config-if)#ip ospf 1 area 0
Vasai(config-if)#
Vasai(config-if)#int f0/0
Vasai(config-if)#ip add 10.0.0.2 255.255.255.0
Vasai(config-if)#no shut
Vasai(config-if)#
*Mar  1 00:26:15.331: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 00:26:16.331: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
Vasai(config-if)#ip ospf 1 area 0
Vasai(config-if)#
*Mar  1 00:27:18.279: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on FastEthernet0/0
from LOADING to FULL, Loading Done
Vasai(config-if)#int f0/1
Vasai(config-if)#ip add 10.0.1.2 255.255.255.0
Vasai(config-if)#no shut
Vasai(config-if)#
*Mar  1 00:27:59.663: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Mar  1 00:28:00.663: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
Vasai(config-if)#ip ospf 1 area 0
Vasai(config-if)#
```

```
Bhayandar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Bhayandar(config)#int lo0
Bhayandar(config-if)#
*Mar  1 00:29:51.463: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
Bhayandar(config-if)#ip add 3.3.3.3 255.255.255.255
Bhayandar(config-if)#ip ospf 1 area 0
Bhayandar(config-if)#int f0/0
Bhayandar(config-if)#ip add 10.0.1.3 255.255.255.0
Bhayandar(config-if)#no shut
Bhayandar(config-if)#
*Mar  1 00:33:02.535: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 00:33:03.535: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
Bhayandar(config-if)#ip ospf 1 area 0
Bhayandar(config-if)#
*Mar  1 00:33:35.195: %OSPF-5-ADJCHG: Process 1, Nbr 10.0.0.2 on FastEthernet0/0
 from LOADING to FULL, Loading Done
Bhayandar(config-if)#
```

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

```
Nallasopara(config-if)#
Nallasopara(config-if)#exit
Nallasopara(config)#exit
Nallasopara#
*Mar  1 00:36:45.995: %SYS-5-CONFIG_I: Configured from console by console
Nallasopara#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 = 52/65/84 ms
Nallasopara#
```

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.

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

```
Vasai(config-if)#exit
Vasai(config)#router ospf 1
Vasai(config-router)#mpls ldp autoconfig
Vasai(config-router)#
*Mar  1 00:49:29.567: %LDP-5-NBRCHG: LDP Neighbor 1.1.1.1:0 (1) is UP
```

You should see log messages coming up showing the LDP neighbors are up. To verify the mpls interfaces the command is very simple – sh mpls interface

```
Bhayandar(config-if)#exit
Bhayandar(config)#router ospf 1
Bhayandar(config-router)#mpls ldp autoconfig
Bhayandar(config-router)#
*Mar  1 00:57:16.555: %LDP-5-NBRCHG: LDP Neighbor 2.2.2.2:0 (1) is UP
Bhayandar(config-router)#
```

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

```
Vasai#sh mpls int
Interface              IP            Tunnel   Operational
FastEthernet0/0        Yes (ldp)     No       Yes
FastEthernet0/1        Yes (ldp)     No       Yes
Vasai#
```

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

```
Vasai#
Vasai#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 - 2.2.2.2.14789
        State: Oper; Msgs sent/rcvd: 24/24; Downstream
        Up time: 00:14:38
        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: 3.3.3.3.32506 - 2.2.2.2.646
        State: Oper; Msgs sent/rcvd: 15/15; Downstream
        Up time: 00:06:51
        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
Vasai#
```

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.

```
Nallasopara#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] 52 msec 60 msec 60 msec
  2 10.0.1.3 64 msec 60 msec 64 msec
Nallasopara#
```

Step 3 – MPLS BGP Configuration between R1 and R3

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

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#router bgp 1
Nallasopara(config-router)#neighbor 3.3.3.3 remote-as 1
Nallasopara(config-router)#neighbor 3.3.3.3 update-source Loopback0
Nallasopara(config-router)#no auto-summary
Nallasopara(config-router)#address-family vpnv4
Nallasopara(config-router-af)#neighbor 3.3.3.3 activate
Nallasopara(config-router-af)#
Nallasopara(config-router-af)#
```

```
Bhayandar(config)#router bgp 1
Bhayandar(config-router)#neighbor 1.1.1.1 remote-as 1
Bhayandar(config-router)#neighbor 1.1.1.1
*Mar  1 01:11:12.011: %BGP-5-ADJCHANGE: neighbor 1.1.1.1 Up
Bhayandar(config-router)#neighbor 1.1.1.1 update-source loopback 0
Bhayandar(config-router)#no auto summary
                                 ^
% Invalid input detected at '^' marker.

Bhayandar(config-router)#no auto-summary
Bhayandar(config-router)#address-family vpnv4
Bhayandar(config-router-af)#neighbor 1.1.1.1 activate
Bhayandar(config-router-af)#
*Mar  1 01:12:59.559: %BGP-5-ADJCHANGE: neighbor 1.1.1.1 Down Address family act
ivated
Bhayandar(config-router-af)#
*Mar  1 01:13:01.711: %BGP-5-ADJCHANGE: neighbor 1.1.1.1 Up
Bhayandar(config-router-af)#
```

To verify the BGP session between R1 and R3 issue the command sh bgp vpnv4 unicast all summary.

```
Nallasopara#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      10      10        1    0    0 00:02:12        0
Nallasopara#
```

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.

```
Virar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Virar(config)#int lo0
Virar(config-if)#
*Mar  1 01:17:59.067: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
Virar(config-if)#ip address 4.4.4.4 255.255.255.255
Virar(config-if)#ip ospf 2 area 2
Virar(config-if)#int f0/0
Virar(config-if)#address 223.168.1.4 255.255.255.0
                  ^
% Invalid input detected at '^' marker.

Virar(config-if)#ip address 223.168.1.4 255.255.255.0
Virar(config-if)#ip ospf 2 area 2
Virar(config-if)#no shut
Virar(config-if)#
*Mar  1 01:21:20.275: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 01:21:21.275: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
Virar(config-if)#
```

```
Nallasopara#
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#int f0/1
Nallasopara(config-if)#no shut
Nallasopara(config-if)#
*Mar  1 01:24:44.151: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Mar  1 01:24:45.151: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
Nallasopara(config-if)#ip address 223.168.1.1 255.255.255.0
Nallasopara(config-if)#
```

```
Nallasopara(config)#ip vrf RED
Nallasopara(config-vrf)#rd 4:4
Nallasopara(config-vrf)#router-target both 4:4
                            ^
% Invalid input detected at '^' marker.

Nallasopara(config-vrf)#route-target both 4:4
Nallasopara(config-vrf)#
Nallasopara(config-vrf)#int f0/1
Nallasopara(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 223.168.1.1 removed due to enabling VRF R
ED
Nallasopara(config-if)#ip add 223.168.1.1 255.255.255.0
Nallasopara(config-if)#
```

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

Current configuration : 119 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 ip address 223.168.1.1 255.255.255.0
 duplex auto
 speed auto
end

Nallasopara#
```

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

```
Nallasopara#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 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

     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 [110/11] via 10.0.0.2, 01:14:05, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/21] via 10.0.0.2, 01:07:38, 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 [110/20] via 10.0.0.2, 01:12:47, FastEthernet0/0
Nallasopara#
```

```
Nallasopara(config)#ip vrf RED
Nallasopara(config-vrf)#rd 4:4
Nallasopara(config-vrf)#route-target both 4:4
Nallasopara(config-vrf)#
Nallasopara(config-vrf)#int f0/1
Nallasopara(config-if)#ip vrf forwarding RED
Nallasopara(config-if)#ip add 223.168.1.1 255.255.255.0
Nallasopara(config-if)#
```

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

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

```
Nallasopara#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/1
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/11] via 223.168.1.4, 00:00:35, FastEthernet0/1
Nallasopara#
```

We now need to repeat this process for R3 & R6 Router 6 will peer OSPF using process number 2 to a VRF configured on R3. It will use the local site addressing to 223.168.2.0/24.

```
Nallasopara#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 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

     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 [110/11] via 10.0.0.2, 01:27:31, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/21] via 10.0.0.2, 01:21:04, 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 [110/20] via 10.0.0.2, 01:26:13, FastEthernet0/0
Nallasopara#
```

```
MiraRoad#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
MiraRoad(config)#int lo0
MiraRoad(config-if)#ip
*Mar  1 01:55:49.163: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
MiraRoad(config-if)#ip add 6.6.6.6 255.255.255.255
MiraRoad(config-if)#ip ospf 2 area 2
MiraRoad(config-if)#int f0/0
MiraRoad(config-if)#ip add 223.168.2.6 255.255.255.0
MiraRoad(config-if)#ip ospf 2 area 2
MiraRoad(config-if)#no shut
MiraRoad(config-if)#
*Mar  1 01:58:08.587: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 01:58:09.587: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
MiraRoad(config-if)#
```

```
Bhayandar(config)#int f0/1
Bhayandar(config-if)#no shut
Bhayandar(config-if)#
*Mar  1 01:59:27.655: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Mar  1 01:59:28.655: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
Bhayandar(config-if)#ip add 233.168.2.3 255.255.255.0
Not a valid host address - 233.168.2.3
Bhayandar(config-if)#ip add 223.168.2.3 255.255.255.0
Bhayandar(config-if)#
```

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

```
Bhayandar(config)#ip vrf RED
Bhayandar(config-vrf)#int f0/1
Bhayandar(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 223.168.2.3 removed due to enabling VRF R
ED
Bhayandar(config-if)#ip add 223.168.2.1 255.255.255.0
Bhayandar(config-if)#ip vrf RED
Bhayandar(config-vrf)#rd 4:4
Bhayandar(config-vrf)#route-target both 4:4
Bhayandar(config-vrf)#int f0/1
Bhayandar(config-if)#ip vrf forwarding RED
Bhayandar(config-if)#ip add 223.168.2.1 255.255.255.0
Bhayandar(config-if)#
```

Check the Router in VRF RED

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

Current configuration : 119 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 ip address 223.168.2.1 255.255.255.0
 duplex auto
 speed auto
end

Bhayandar#
```

```
Bhayandar#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 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

C    223.168.2.0/24 is directly connected, FastEthernet0/1
Bhayandar#
```

```
Virar#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/0
     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loopback0
Virar#
```

Done till page 82.

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

```
Nallasopara#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 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

     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 [110/11] via 10.0.0.2, 00:33:12, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/21] via 10.0.0.2, 00:29:27, 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 [110/20] via 10.0.0.2, 00:32:17, FastEthernet0/0
```

```
Nallasopara#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/1
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/11] via 223.168.1.4, 00:13:07, FastEthernet0/1
```

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#
Nallasopara(config)#router bgp 1
Nallasopara(config-router)#address-family ipv4 vrf RED
Nallasopara(config-router-af)#redistribute ospf 2
Nallasopara(config-router-af)#exit
Nallasopara(config-router)#end
```

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

```
Nallasopara#sh ip bgp vpnv4 vrf RED
BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
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       223.168.1.4             11         32768 ?
*> 223.168.1.0      0.0.0.0                  0         32768 ?
```

```
Bhayandar#sh ip bgp vpnv4 vrf RED
BGP table version is 5, local router ID is 3.3.3.3
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
Route Distinguisher: 4:4 (default for vrf RED)
*>i4.4.4.4/32       1.1.1.1                 11    100      0 ?
*>i223.168.1.0      1.1.1.1                  0    100      0 ?
```

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

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

```
Bhayandar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Bhayandar(config)#router ospf 2
Bhayandar(config-router)#redistribute bgp 1 subnets
Bhayandar(config-router)#
```

Before we do let's see what the routing table look like on R.

```
Virar#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/0
     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loopback0
```

```
Bhayandar#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

     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/21] via 10.0.1.2, 04:33:27, FastEthernet0/0
     2.0.0.0/32 is subnetted, 1 subnets
O       2.2.2.2 [110/11] via 10.0.1.2, 04:33:27, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
C       3.3.3.3 is directly connected, Loopback0
     10.0.0.0/24 is subnetted, 2 subnets
O       10.0.0.0 [110/20] via 10.0.1.2, 04:33:27, FastEthernet0/0
C       10.0.1.0 is directly connected, FastEthernet0/0
Bhayandar#
```

Do the Same Step in R6.

```
MiraRoad#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 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

C    223.168.2.0/24 is directly connected, FastEthernet0/0
     6.0.0.0/32 is subnetted, 1 subnets
C       6.6.6.6 is directly connected, Loopback0
```

```
Nallasopara#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 = 48/60/84 ms
```

Practical No - 7

Aim: Inter-VLAN Routing

Theory: VLANs are used to segment switched Layer 2 networks for a variety of reasons. Regardless of the reason, hosts in one VLAN cannot communicate with hosts in another VLAN unless there is a router or a Layer 3 switch to provide routing services.

Inter-VLAN routing is the process of forwarding network traffic from one VLAN to another VLAN.
There are three inter-VLAN routing options:

1) Legacy Inter-VLAN routing: This is a legacy solution. It does not scale well.
2) Router-on-a-Stick: This is an acceptable solution for a small- to medium-sized network.
3) Layer 3 switch using switched virtual interfaces (SVIs): This is the most scalable solution for medium to large organizations.

Inter-VLAN Routing on a Layer 3 Switch

The modern method of performing inter-VLAN routing is to use Layer 3 switches and switched virtual interfaces (SVI). An SVI is a virtual interface that is configured on a Layer 3 switch

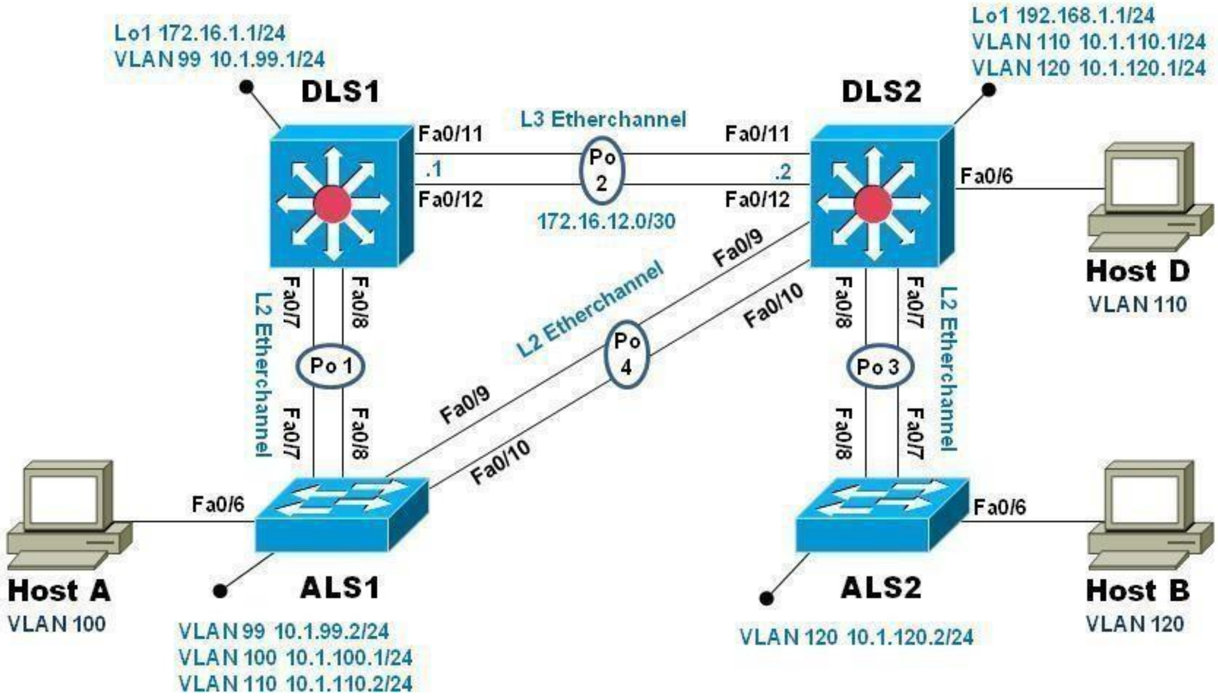
Inter-VLAN SVIs are created the same way that the management VLAN interface is configured. The SVI is created for a VLAN that exists on the switch. Although virtual, the SVI performs the same functions for the VLAN as a router interface would. Specifically, it provides Layer 3 processing for packets that are sent to or from all switch ports associated with that VLAN.

The following are advantages of using Layer 3 switches for inter-VLAN routing:

1) They are much faster than router-on-a-stick because everything is hardware switched and routed.
2) There is no need for external links from the switch to the router for routing.
3) They are not limited to one link because Layer 2 Ether Channels can be used as trunk links between the switches to increase bandwidth.
4) Latency is much lower because data does not need to leave the switch to be routed to a different network.
5) They are more commonly deployed in a campus LAN than routers.

The only disadvantage is that Layer 3 switches are more expensive than Layer 2 switches, but they can be less expensive than a separate Layer 2 switch and router.

Topology



Objectives

- Implement a Layer 3 EtherChannel
- Implement Static Routing
- Implement Inter-VLAN Routing

Required Resources

- 2 Cisco 2960 with the Cisco IOS Release 15.0(2)SE6 C2960-LANBASEK9-M or comparable
- 2 Cisco 3560v2 with the Cisco IOS Release 15.0(2)SE6 C3560-IPSERVICESK9-M or comparable
- Computer with terminal emulation software
- Ethernet and console cables
- 3 PCs with appropriate software

Part 1: Configure Multilayer Switching using Distribution Layer Switches

Step 1: **Load base config**

Use the reset.tcl script you created in Lab 1 “Preparing the Switch” to set your switches up for this lab. Thenload 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: **Verify switch management database configuration**

At each switch, use the show sdm prefer command to verify the appropriate template is chosen. The DLS switches should be using the "dual ipv4-and-ipv6 routing" template and the ALS switches should be using the"lanbase-routing" template. If any of the switches are using the wrong template, make the necessary change and reboot the switch with the **reload** command. An example from ALS1 is below:

ALS1# **sho sdm pref**

The current template is "default" template.

<output

omitted> ALS1#

conf t

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

ALS1(config)# **sdm pref lanbase-routing**

Changes to the running SDM preferences have been stored, but cannot take effect until the next reload.

Use 'show sdm prefer' to see what SDM preference is currently active. ALS1(config)# **end**

ALS1# **reload**

System configuration has been modified. Save? [yes/no]: **y**

*Mar 1 02:12:00.699: %SYS-5-CONFIG_I: Configured from console by console Building configuration...

[OK]

Proceed with reload? [confirm]

Step 3: **Configure layer 3 interfaces on the DLS switches**

Enable IP Routing, create broadcast domains (VLANs), and configure the DLS switches with the layer 3interfaces and addresses shown:

Switch	Interface	Address/Mask
DLS1	VLAN 99	10.1.99.1/24
DLS1	Loopback 1	172.16.1.1/24
DLS2	VLAN 110	10.1.110.1/24
DLS2	VLAN 120	10.1.120.1/24
DLS2	Loopback 1	192.168.2.1/24

An example from DLS2:

DLS2(config)# **ip routing**

DLS2(config)# **vlan 110** DLS2(config-vlan)# **name Management**

DLS2(config-vlan)# **exit**

DLS2(config)# **vlan 120** DLS2(config-vlan)# **name Local** DLS2(config-vlan)# **exit** DLS2(config)# **int vlan**

110

DLS2(config-if)# **ip address 10.1.110.1 255.255.255.0**

DLS2(config-if)# **no shut**

DLS2(config-if)# **exit**

DLS2(config)# **int vlan 120**

DLS2(config-if)# **ip address 10.1.120.1 255.255.255.0**

DLS2(config-if)# **no shut**

DLS2(config-if)# **exit**

DLS2(config)# **int loopback 1**

DLS2(config-if)# **ip address 223.168.1.1 255.255.255.0**

DLS2(config-if)# **no shut**

DLS2(config-if)# **exit**

DLS2(config)#

At this point, basic intervlan routing can be demonstrated using an attached host. Host D is attached to DLS2via interface Fa0/6. On DLS2, assign interface Fa0/6 to VLAN 110 and configure the host with the address 10.1.110.50/24 and default gateway of 10.1.110.1. Once you have done that, try and ping Loopback 1's IP address (192.168.1.1). This should work just like a hardware router; the switch will provide connectivity between two directly connected interfaces. In the output below, the **switchport host** macro was used to quickly configure interface Fa0/6 with host-relative commands:

DLS2(config)# **int f0/6**

DLS2(config-if)# **switchport host**

switchport mode will be set to access

spanning-tree portfast will be enabled

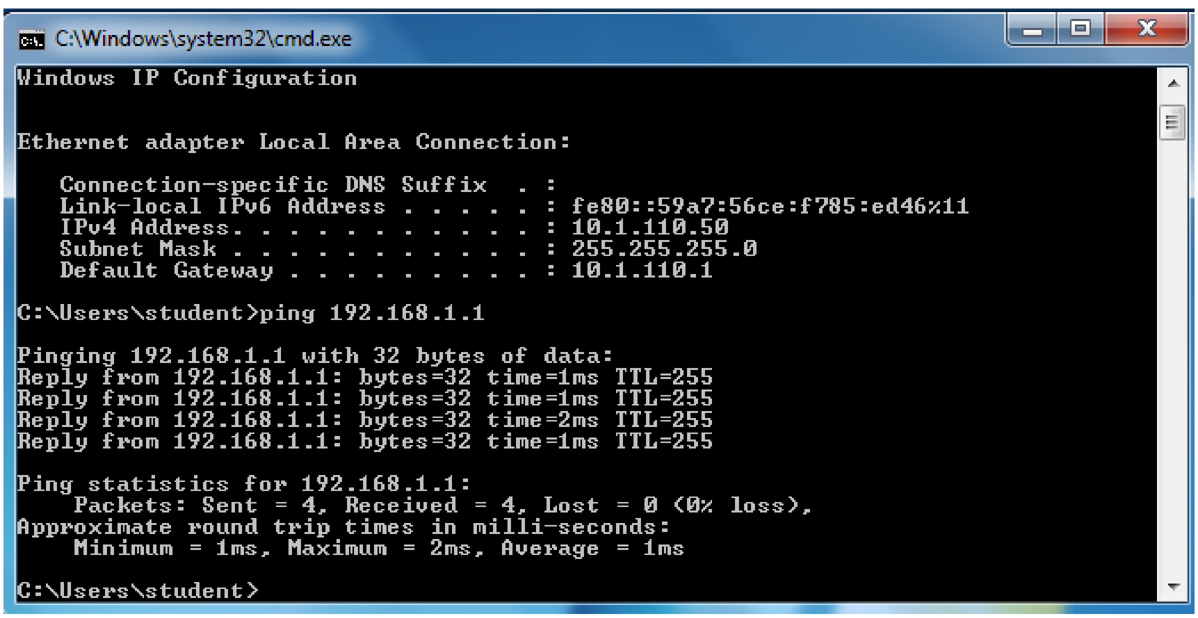
channel group will be disabled

DLS2(config-if)# **switchport access vlan 110**

DLS2(config-if)# **no shut**

DLS2(config-if)# **exit**

DLS2(config)#



Step 4: **Configure a Layer 3 Etherchannel between DLS1 and DLS2**

Now you will interconnect the multilayer switches in preparation to demonstrate other routing capabilities. Configure a layer 3 EtherChannel between the DLS switches. This will provide the benefit of increased available bandwidth between the two multilayer switches. To convert the links from layer 2 to layer 3, issue the **no switchport** command. Then, combine interfaces F0/11 and F0/12 into a single PAgP EtherChanneland then assign an IP address as shown.

DLS1	172.16.12.1/30	DLS2	172.16.12.2/30

Example from DLS1:

DLS1(config)# **interface range f0/11-12**

DLS1(config-if-range)# **no switchport**

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

Creating a port-channel interface Port-channel 2

DLS1(config-if-range)# **no shut** DLS1(config-if-range)#

exit DLS1(config)# **interface port-channel 2**

DLS1(config-if)# **ip address 172.16.12.1 255.255.255.252**

DLS1(config-if)# **no shut**

DLS1(config-if)# **exit** DLS1(config)#

Once you have configured both sides, verify that the EtherChannel link is up

DLS2# **show etherchannel summary**

Flags: D - down P - bundled in port-channel I - stand-

alone s - suspended

H - Hot-standby (LACP only) R - Layer3 S - Layer2

U - in use f - failed to allocate aggregator

M - not in use, minimum links not met u - unsuitable for bundling

w - waiting to be aggregated d - default port

Number of channel-groups in use: 1 Number of

aggregators: 1

```
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
2      Po2(RU)       PAgP        Fa0/11(P)   Fa0/12(P)
```

DLS2# **ping 172.16.12.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.12.1, timeout is 2 seconds:

.!!!!

Success rate is 80 percent (4/5), round-trip min/avg/max = 1/3/9 ms DLS2#

Step 5: **Configure default routing between DLS switches**

At this point, local routing is support at each distribution layer switch. Now to provide reachability across the layer 3 EtherChannel trunk, configure fully qualified static default routes at DLS1 and DLS2 that point to eachother. From DLS1:

DLS1(config)# **ip route 0.0.0.0 0.0.0.0 port-channel 2**

%Default route without gateway, if not a point-to-point interface, may impact performance

DLS1(config)# **ip route 0.0.0.0 0.0.0.0 port-channel 2 172.16.12.2**

DLS1(config)#

Once done at both ends, verify connectivity by pinging from one switch to the other. In the example below,DLS2 pings the Loopback 1 interface at DLS1.

DLS2# **show ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

<output omitted>

Gateway of last resort is 172.16.12.1 to network 0.0.0.0 S*

0.0.0.0/0 [1/0] via 172.16.12.1, Port-channel2

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks C

10.1.110.0/24 is directly connected, Vlan110

L 10.1.110.1/32 is directly connected, Vlan110

172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks

C 172.16.12.0/30 is directly connected, Port-channel2 L

172.16.12.2/32 is directly connected, Port-channel2

192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks C

192.168.1.0/24 is directly connected, Loopback1

L 192.168.1.1/32 is directly connected, Loopback1

DLS2# **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 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms DLS2#

Step 6: **Configure the remaining EtherChannels for the topology**

Configure the remaining EtherChannel links as layer 2 PagP trunks using VLAN 1 as the native VLAN.

Endpoint 1	Channel number	Endpoint 2	VLANs Allowed
ALS1 F0/7-8	1	DLS1 F0/7-8	All except 110
ALS1 F0/9-10	4	DLS2 F0/9-10	110 Only
ALS2 F0/7-8	3	DLS2 F0/7-8	All

Example from ALS1:

ALS1(config)# **interface range f0/7-8**

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

ALS1(config-if-range)# **switchport trunk allowed vlan except 110**

ALS1(config-if-range)# **channel-group 1 mode desirable**

Creating a port-channel interface Port-channel 1

ALS1(config-if-range)# **no shut**

ALS1(config-if-range)# **exit**

ALS1(config)# **interface range f0/9-10**

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

ALS1(config-if-range)# **switchport trunk allowed vlan 110**

ALS1(config-if-range)# **channel-group 4 mode desirable**

Creating a port-channel interface Port-channel 4

ALS1(config-if-range)# **no shut**

ALS1(config-if-range)# **exit**

ALS1(config)#end

ALS1# **show etherchannel summary**

Flags: D - down P - bundled in port-channel I -

stand-alone s - suspended

H - Hot-standby (LACP only) R - Layer3

S - Layer2

U - in use f - failed to allocate aggregator

M - not in use, minimum links not met u -
unsuitable for bundling
w - waiting to be aggregated d - default
port

Number of channel-groups in use: 2
Number of aggregators: 2

Group	Port-channel	Protocol	Ports	
1	Po1(SU)	PAgP	Fa0/7(P)	Fa0/8(P)
4	Po4(SU)	PAgP	Fa0/9(P)	Fa0/10(P)

Port	Vlans allowed on trunk
Po1	1-109,111-4094
Po4	110

<output
omitted> ALS1#

Step 7: **Enable and Verify Layer 3 connectivity across the network**

In this step we will enable basic connectivity from the management VLANs on both sides of the network.

- Create the management VLANs (99 at ALS1, 120 at ALS2)
- Configure interface VLAN 99 at ALS1 and interface VLAN 120 at ALS2
- Assign addresses (refer to the diagram) and default gateways (at DLS1/DLS2 respectively).

Once that is all done, pings across the network should work, flowing across the layer 3 EtherChannel. Anexample from ALS2:

ALS2(config)# **vlan 120** ALS2(config-
vlan)# **name Management**
ALS2(config-vlan)# **exit**
ALS2(config)# **int vlan 120**
ALS2(config-if)# **ip address 10.1.120.2 255.255.255.0**
ALS2(config-if)# **no shut**
ALS2(config-if)# **exit**

ALS2(config)# **ip default-gateway 10.1.120.1**

ALS2(config)# **end**

ALS2# **ping 10.1.99.2**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.99.2, timeout is 2 seconds:

..!!!

Success rate is 60 percent (3/5), round-trip min/avg/max = 1/3/8 ms ALS2#
ALS2# **traceroute 10.1.99.2**

Type escape sequence to abort.

Tracing the route to 10.1.99.2

VRF info: (vrf in name/id, vrf out name/id)

1 10.1.120.1 0 msec 0 msec 8 msec

2 172.16.12.1 0 msec 0 msec 8 msec

3 10.1.99.2 0 msec 0 msec *

ALS2#

Part 2: Configure Multilayer Switching at ALS1

At this point all routing is going through the DLS switches, and the port channel between ALS1 and DLS2 isnot passing anything but control traffic (BPDUs, etc).

The Cisco 2960 is able to support basic routing when it is using the LANBASE IOS. In this step you will configure ALS1 to support multiple SVIs and configure it for basic static routing. The objectives of this stepare:

- Enable intervlan routing between two VLANs locally at ALS1
- Enable IP Routing
- Configure a static route for DLS2's Lo1 network travel via Port-Channel 4.

Step 1: **Configure additional VLANs and VLAN interfaces**

At ALS1, create VLAN 100 and VLAN 110 and then create SVIs for those VLANs

ALS1(config)# **ip routing**

ALS1(config)# **vlan 100** ALS1(config-

vlan)# **name Local** ALS1(config-

vlan)# **exit** ALS1(config)# **vlan 110**

ALS1(config-vlan)# **name InterNode**

ALS1(config-vlan)# **exit**

ALS1(config)# **int vlan 100**

ALS1(config-if)# **ip address 10.1.100.1 255.255.255.0**

ALS1(config-if)# **no shut**

ALS1(config-if)# **exit**

ALS1(config)# **int vlan 110**

ALS1(config-if)# **ip address 10.1.110.2 255.255.255.0**

ALS1(config-if)# **no shut**

ALS1(config-if)# **exit**

ALS1(config)#

Step 2: **Configure and test Host Access**

Assign interface Fa0/6 to VLAN 100. On the attached host (Host A) configure the IP address 10.1.100.50/24with a default gateway of 10.1.100.1. Once configured, try a traceroute from the host to 10.1.99.2 and observe the results.

In the output below, the **switchport host**macro was used to quickly configure interface Fa0/6 with host-relative commands.

ALS1(config)# **interface f0/6** ALS1(config-

if)# **switchport host** switchport mode

will be set to access spanning-tree

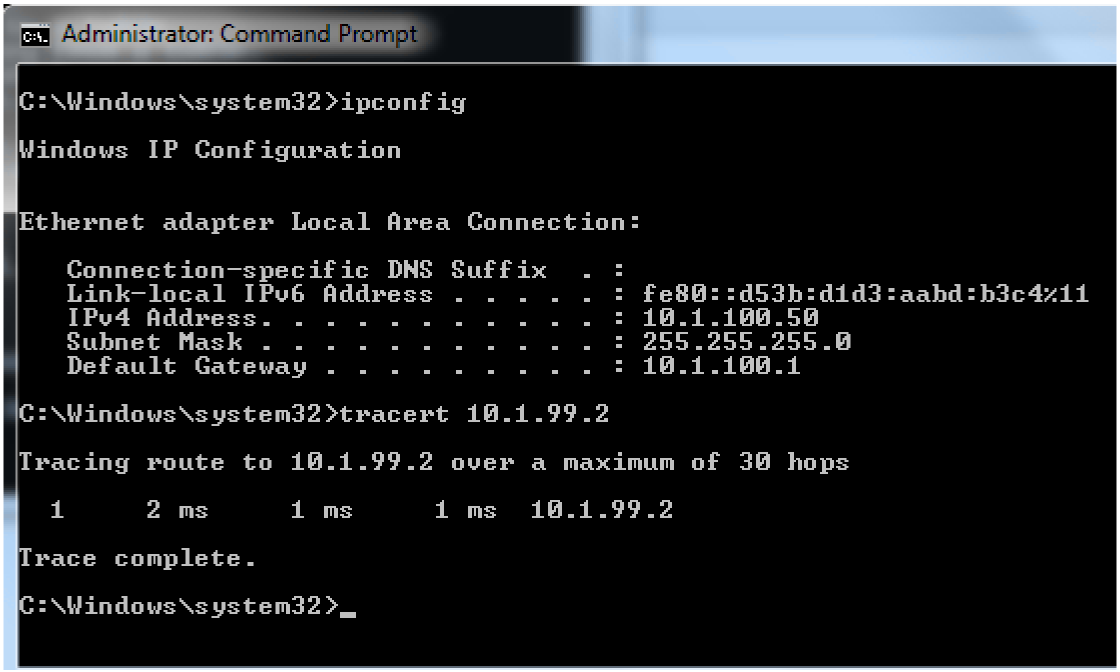
portfast will be enabled channel group

will be disabled

ALS1(config-if)# **switchport access vlan 100**

ALS1(config-if)# **no shut**

ALS1(config-if)# **exit**



Administrator: Command Prompt

```
C:\Windows\system32>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::d53b:d1d3:aabd:b3c4%11
   IPv4 Address. . . . . . . . . . . : 10.1.100.50
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.1.100.1

C:\Windows\system32>tracert 10.1.99.2

Tracing route to 10.1.99.2 over a maximum of 30 hops

  1     2 ms     1 ms     1 ms  10.1.99.2

Trace complete.

C:\Windows\system32>_
```

The output from the host shows that attempts to communicate with interface VLAN 99 at ALS1 were fulfilledlocally, and not sent to DLS1 for routing.

Step 3: **Configure and verify static routing across the network**

At this point, local routing (at ALS1) works, and off-net routing (outside of ALS1) will not work, because DLS1doesn't have any knowledge of the 10.1.100.0 subnet. In this step you will configure routing on several different switches:

- At DLS1, configure:

The output from the host shows that attempts to communicate with interface VLAN 99 at ALS1 were fulfilledlocally, and not sent to DLS1 for routing.

Step 3: **Configure and verify static routing across the network**

At this point, local routing (at ALS1) works, and off-net routing (outside of ALS1) will not work, because DLS1doesn't have any knowledge of the 10.1.100.0 subnet. In this step you will configure routing on several different switches:

- At DLS1, configure:
  - a static route to the 10.1.100.0/24 network via VLAN 99
- At DLS2, configure
  - a static route to the 10.1.100.0/24 network via VLAN 110
- At ALS1, configure
  - a static route to the 192.168.1.0/24 network via VLAN 110
  - a default static route to use 10.1.99.1

Here is an example from ALS1:

ALS1(config)# **ip route 192.168.1.0 255.255.255.0 vlan 110**
ALS1(config)# **ip route 0.0.0.0 0.0.0.0 10.1.99.1**
ALS1(config)# **end**

ALS1# **show ip route**

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2 E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2 ia - IS-IS inter area, * - candidate default, U - per-user static route o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP

+ - replicated route, % - next hop override

Gateway of last resort is 10.1.99.1 to network 0.0.0.0

S* 0.0.0.0/0 [1/0] via 10.1.99.1

10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks C

10.1.99.0/24 is directly connected, Vlan99

L 10.1.99.2/32 is directly connected, Vlan99 C

10.1.100.0/24 is directly connected, Vlan100 L

10.1.100.1/32 is directly connected, Vlan100 C

10.1.110.0/24 is directly connected, Vlan110 L

10.1.110.2/32 is directly connected, Vlan110 S

192.168.1.0/24 is directly connected, Vlan110

After configuring all of the required routes, test to see that the network behaves as expected.

From ALS1, a traceroute to 10.1.120.2 should take three hops:

ALS1# **traceroute 10.1.120.2**

Type escape sequence to abort.

Tracing the route to 10.1.120.2

VRF info: (vrf in name/id, vrf out name/id)

1 10.1.99.1 0 msec 0 msec 0 msec

2 172.16.12.2 9 msec 0 msec 0 msec

3 10.1.120.2 0 msec 8 msec *

ALS1#

From ALS1, a traceroute to 192.168.1.1 should take one hop:

ALS1# **traceroute 192.168.1.1**

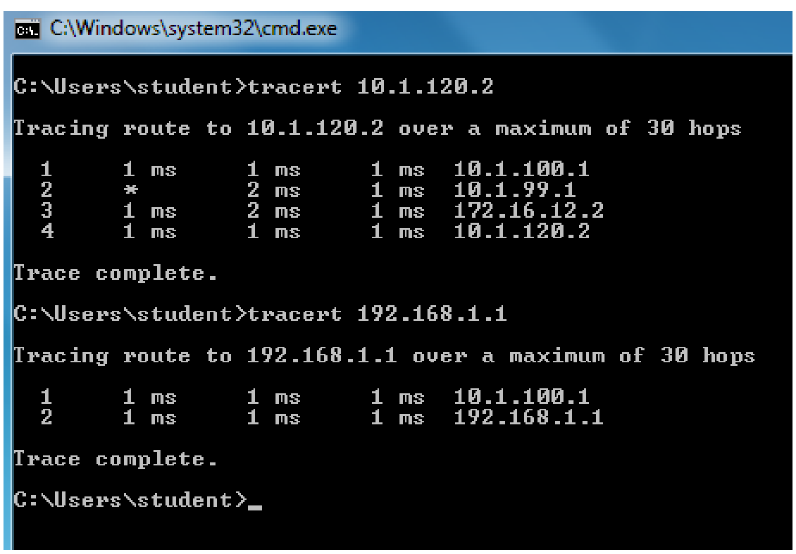
Type escape sequence to abort.

Tracing the route to 192.168.1.1

VRF info: (vrf in name/id, vrf out name/id)

1 10.1.110.1 0 msec 0 msec *

ALS1#



Traces from Host A show an additional hop, but follow the appointed path:

Step 4: **End of Lab**

Save your configurations. The switches will be used as configured now for lab 5-2, DHCP.

Practical No 8

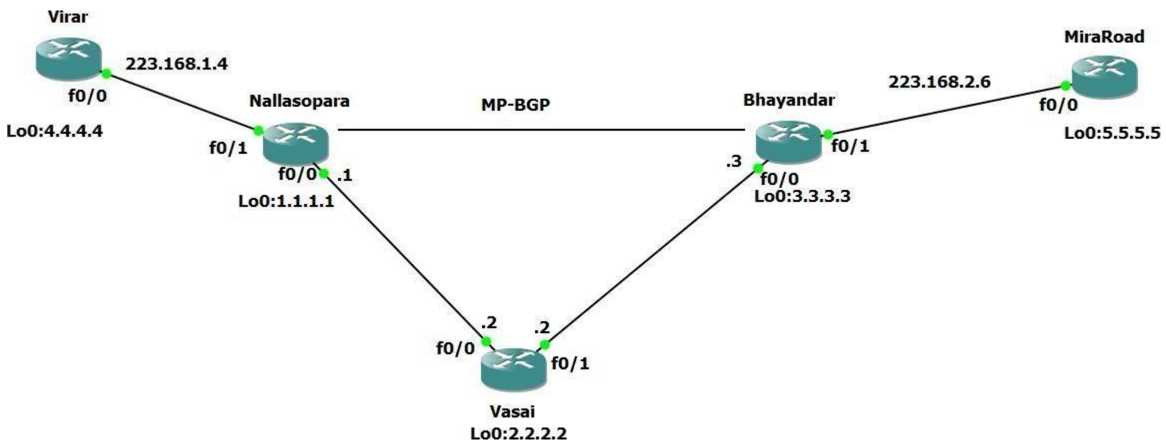
Aim: IP Service Level Agreements and Remote SPAN in a Campus Environment

Theory
:MPLS is a high-performance network technique that directs data from one node to another using short path labels rather than long network addresses. It operates between Layer 2 and Layer 3 and improves speed and control in large-scale networks. MPLS helps service providers build scalable and efficient networks capable of supporting VPNs, traffic engineering, and Quality of Service (QoS).

VRF (Virtual Routing and Forwarding) is a method that allows multiple instances of routing tables to coexist on the same router, isolating traffic without needing multiple physical networks. VRFs are crucial in MPLS VPNs where different customers need private routing domains while sharing the same MPLS backbone.

Together, MPLS and VRF provide the foundation for MPLS Layer 3 VPNs, enabling secure, scalable, and efficient data transport across multi-customer environments.

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 R1 and R3

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#int lo0
Nallasopara(config-if)#
*Mar  1 00:02:30.835: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
Nallasopara(config-if)#ip add 1.1.1.1 255.255.255.255
Nallasopara(config-if)#ip ospf 1 area 0
Nallasopara(config-if)#
Nallasopara(config-if)#int f0/0
Nallasopara(config-if)#ip add 10.0.0.1 255.255.255.0
Nallasopara(config-if)#no shut
Nallasopara(config-if)#ip ospf
*Mar  1 00:04:00.111: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 00:04:01.111: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
Nallasopara(config-if)#ip ospf 1 area 0
Nallasopara(config-if)#
```

```
Vasai#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Vasai(config)#int lo0
Vasai(config-if)#
*Mar  1 00:03:11.943: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
Vasai(config-if)#ip add 2.2.2.2 255.255.255.255
Vasai(config-if)#ip ospf 1 area 0
Vasai(config-if)#
Vasai(config-if)#int f0/0
Vasai(config-if)#ip add 10.0.0.2 255.255.255.0
Vasai(config-if)#no shut
Vasai(config-if)#ip
*Mar  1 00:04:28.359: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 00:04:29.359: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
Vasai(config-if)#ip ospf 1 area 0
Vasai(config-if)#
Vasai(config-if)#int f0/
*Mar  1 00:04:47.391: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on FastEthernet0/0
from LOADING to FULL, Loading Done
Vasai(config-if)#int f0/1
Vasai(config-if)#ip add 10.0.1.2 255.255.255.0
Vasai(config-if)#no shut
Vasai(config-if)#ip ospf
*Mar  1 00:05:32.339: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Mar  1 00:05:33.339: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
Vasai(config-if)#ip ospf 1 area 0
```

```
Bhayandar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Bhayandar(config)#int lo0
Bhayandar(config-if)#
*Mar  1 00:04:46.155: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
Bhayandar(config-if)#ip add 3.3.3.3 255.255.255.255
Bhayandar(config-if)#ip ospf 1 area 0
Bhayandar(config-if)#
Bhayandar(config-if)#int f0/0
Bhayandar(config-if)#ip add 10.0.1.3 255.255.255.0
Bhayandar(config-if)#no shut
Bhayandar(config-if)#ip ospf
*Mar  1 00:06:07.163: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Mar  1 00:06:08.163: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
Bhayandar(config-if)#ip ospf 1 area 0
```

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

```
Nallasopara#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 = 48/60/84 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.

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

```
Vasai(config)#router ospf 1
Vasai(config-router)#mpls ldp autoconfig
Vasai(config-router)#router
*Mar  1 00:11:28.635: %LDP-5-NBRCHG: LDP Neighbor 1.1.1.1:0 (1) is UP
Vasai(config-router)#
*Mar  1 00:12:49.523: %LDP-5-NBRCHG: LDP Neighbor 3.3.3.3:0 (2) is UP
```

You should see log messages coming up showing the LDP neighbors are up. To verify the mpls interfaces the command is very simple – sh mpls interface

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

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

```
Vasai#sh mpls int
Interface              IP            Tunnel   Operational
FastEthernet0/0        Yes (ldp)     No       Yes
FastEthernet0/1        Yes (ldp)     No       Yes
```

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

```
Vasai#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 - 2.2.2.2.24585
        State: Oper; Msgs sent/rcvd: 11/11; Downstream
        Up time: 00:03:06
        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: 3.3.3.3.42628 - 2.2.2.2.646
        State: Oper; Msgs sent/rcvd: 10/10; Downstream
        Up time: 00:01:46
        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.

```
Nallasopara#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] 44 msec 76 msec 60 msec
  2 10.0.1.3 64 msec 72 msec 28 msec
```

Step 3 – MPLS BGP Configuration between R1 and R3

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

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#router bgp 1
Nallasopara(config-router)#neighbor 3.3.3.3 remote-as 1
Nallasopara(config-router)#neighbor 3.3.3.3 update-source Loopback0
Nallasopara(config-router)#no auto-summary
Nallasopara(config-router)#address-family vpnv4
Nallasopara(config-router-af)#neighbor 3.3.3.3 activate
```

```
Bhayandar(config)#router bgp 1
Bhayandar(config-router)#neighbor 1.1.1.1 remote-as 1
Bhayandar(config-router)#neighbor 1.1.1.1
% Incomplete command.

Bhayandar(config-router)#
*Mar  1 00:18:56.079: %BGP-5-ADJCHANGE: neighbor 1.1.1.1 Up
Bhayandar(config-router)#neighbor 1.1.1.1 update-source loopback 0
Bhayandar(config-router)#no auto-summary
Bhayandar(config-router)#address-family vpnv4
Bhayandar(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 all summary.

```
Nallasopara#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       8       8        1    0    0 00:00:59        0
Nallasopara#
```

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.

```
Virar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Virar(config)#int lo0
Virar(config-if)#
*Mar  1 00:11:25.043: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
Virar(config-if)#ip address 4.4.4.4 255.255.255.255
Virar(config-if)#ip ospf 2 area 2
Virar(config-if)#int f0/0
Virar(config-if)#ip address 223.168.1.4 255.255.255.0
Virar(config-if)#ip ospf 2 area 2
Virar(config-if)#no shut
```

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#int f0/1
Nallasopara(config-if)#no shut
Nallasopara(config-if)#
*Mar  1 00:29:01.831: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Mar  1 00:29:02.831: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
Nallasopara(config-if)#ip address 223.168.1.1 255.255.255.0
```

```
Nallasopara(config)#ip vrf RED
Nallasopara(config-vrf)#rd 4:4
Nallasopara(config-vrf)#route-target both 4:4
Nallasopara(config-vrf)#
Nallasopara(config-vrf)#
Nallasopara(config-vrf)#int f0/1
Nallasopara(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 223.168.1.1 removed due to enabling VRF R
ED
Nallasopara(config-if)#ip add 223.168.1.1 255.255.255.0
Nallasopara(config-if)#
```

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

Current configuration : 119 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 ip address 223.168.1.1 255.255.255.0
 duplex auto
 speed auto
end
```

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

```
Nallasopara#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 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

     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 [110/11] via 10.0.0.2, 00:30:41, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/21] via 10.0.0.2, 00:26:56, 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 [110/20] via 10.0.0.2, 00:29:46, FastEthernet0/0
Nallasopara#
```

```
Nallasopara(config)#ip vrf RED
Nallasopara(config-vrf)#rd 4:4
Nallasopara(config-vrf)#route-target both 4:4
Nallasopara(config-vrf)#
Nallasopara(config-vrf)#
Nallasopara(config-vrf)#int f0/1
Nallasopara(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 223.168.1.1 removed due to enabling VRF R
ED
Nallasopara(config-if)#ip add 223.168.1.1 255.255.255.0
Nallasopara(config-if)#
```

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

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

```
Nallasopara#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/1
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/11] via 223.168.1.4, 00:00:20, FastEthernet0/1
```

We now need to repeat this process for R3 & R6 Router 6 will peer OSPF using process number 2 to a VRF configured on R3. It will use the local site addressing to 223.168.2.0/24.

```
Nallasopara#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 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

     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 [110/11] via 10.0.0.2, 00:33:12, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/21] via 10.0.0.2, 00:29:27, 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 [110/20] via 10.0.0.2, 00:32:17, FastEthernet0/0
Nallasopara#
```

```
MiraRoad#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
MiraRoad(config)#int lo0
MiraRoad(config-if)#ip
*Mar  1 00:17:16.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0,
 changed state to up
MiraRoad(config-if)#ip add 6.6.6.6 255.255.255.255
MiraRoad(config-if)#ip ospf 2 area 2
MiraRoad(config-if)#int f0/0
MiraRoad(config-if)#ip address 223.168.2.6 255.255.255.0
MiraRoad(config-if)#ip ospf 2 area 2
MiraRoad(config-if)#no shut
```

```
Bhayandar(config)#int f0/1
Bhayandar(config-if)#no shut
Bhayandar(config-if)#
Bhayandar(config-if)#ip add
*Mar  1 00:43:46.895: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Mar  1 00:43:47.895: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
Bhayandar(config-if)#ip add 223.168.2.3 255.255.255.0
```

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

Check the Router in VRF RED

```
Bhayandar(config-vrf)#int f0/1
Bhayandar(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 223.168.2.3 removed due to enabling VRF R
ED
Bhayandar(config-if)#ip add 223.168.2.1 255.255.255.0
```

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

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

Current configuration : 119 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 ip address 223.168.2.1 255.255.255.0
 duplex auto
 speed auto
end
```

```
Bhayandar#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 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

C    223.168.2.0/24 is directly connected, FastEthernet0/1
```

```
Virar#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/0
     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loopback0
```

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

```
Nallasopara#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 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

     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 [110/11] via 10.0.0.2, 00:33:12, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/21] via 10.0.0.2, 00:29:27, 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 [110/20] via 10.0.0.2, 00:32:17, FastEthernet0/0
```

```
Nallasopara#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/1
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/11] via 223.168.1.4, 00:13:07, FastEthernet0/1
```

```
Nallasopara#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Nallasopara(config)#
Nallasopara(config)#router bgp 1
Nallasopara(config-router)#address-family ipv4 vrf RED
Nallasopara(config-router-af)#redistribute ospf 2
Nallasopara(config-router-af)#exit
Nallasopara(config-router)#end
```

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

```
Nallasopara#sh ip bgp vpnv4 vrf RED
BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
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       223.168.1.4             11         32768 ?
*> 223.168.1.0      0.0.0.0                  0         32768 ?
```

```
Bhayandar#sh ip bgp vpnv4 vrf RED
BGP table version is 5, local router ID is 3.3.3.3
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
Route Distinguisher: 4:4 (default for vrf RED)
*>i4.4.4.4/32       1.1.1.1                 11    100      0 ?
*>i223.168.1.0      1.1.1.1                  0    100      0 ?
```

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

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

```
Bhayandar#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Bhayandar(config)#router ospf 2
Bhayandar(config-router)#redistribute bgp 1 subnets
Bhayandar(config-router)#
```

Before we do let's see what the routing table look like on R**.**

```
Virar#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 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

C    223.168.1.0/24 is directly connected, FastEthernet0/0
     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loopback0
```

```
Bhayandar#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

     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/21] via 10.0.1.2, 04:33:27, FastEthernet0/0
     2.0.0.0/32 is subnetted, 1 subnets
O       2.2.2.2 [110/11] via 10.0.1.2, 04:33:27, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
C       3.3.3.3 is directly connected, Loopback0
     10.0.0.0/24 is subnetted, 2 subnets
O       10.0.0.0 [110/20] via 10.0.1.2, 04:33:27, FastEthernet0/0
C       10.0.1.0 is directly connected, FastEthernet0/0
Bhayandar#
```

Do the Same Step in R6.

```
MiraRoad#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 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

C    223.168.2.0/24 is directly connected, FastEthernet0/0
     6.0.0.0/32 is subnetted, 1 subnets
C       6.6.6.6 is directly connected, Loopback0
```

```
Nallasopara#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 = 48/60/84 ms
```

Conclusion:

Through this simulation, we understood how MPLS facilitates faster packet forwarding using labels and how VRF enables traffic segmentation in a multi-customer network. Implementing MPLS with VRF offers a scalable solution for service providers and enterprises, supporting multiple clients over a shared infrastructure while maintaining complete data isolation and independent routing.