PRACTICAL BOOK

MSCIT (PART I) SEMESTER - II 2023-24

SUBJECT

MORDERN NETWORKING

SUBMITTED BY

**Prathamesh Zore**

**Seat No.1312689**

Submitted in partial fulfillment of the requirement for Qualifying

M.Sc. Part I Semester II Examination 2023-24

University of Mumbai Department of Information Technology

Shailendra Degree College

S.V. Road, Dahisar (E), Mumbai-68

# Shailendra Education Society’s

**ARTS, COMMERCE & SCIENCE COLLEGE**

**NAAC Re-accreditated ‘B++’ grade with CGPA score 2.98 Shailendra Nagar, Dahisar (East), Mumbai - 400 068.**



CERTIFICATE

This is to certify that **Mr. Prathamesh Zore** of M.Sc. (I.T.) Part I Semester II has completed the practical work in the subject of **“ Mordern Networking ”** during the academic year 2023- 24 under the guidance of **Asst. Prof. Sandeep Vishwakarma** being the partial requirement for the fulfillment of the curriculum of Master of Science in Information Technology, University of Mumbai.

**Place: Mumbai**

**Date: / /2024**

Internal Examiner External Examiner

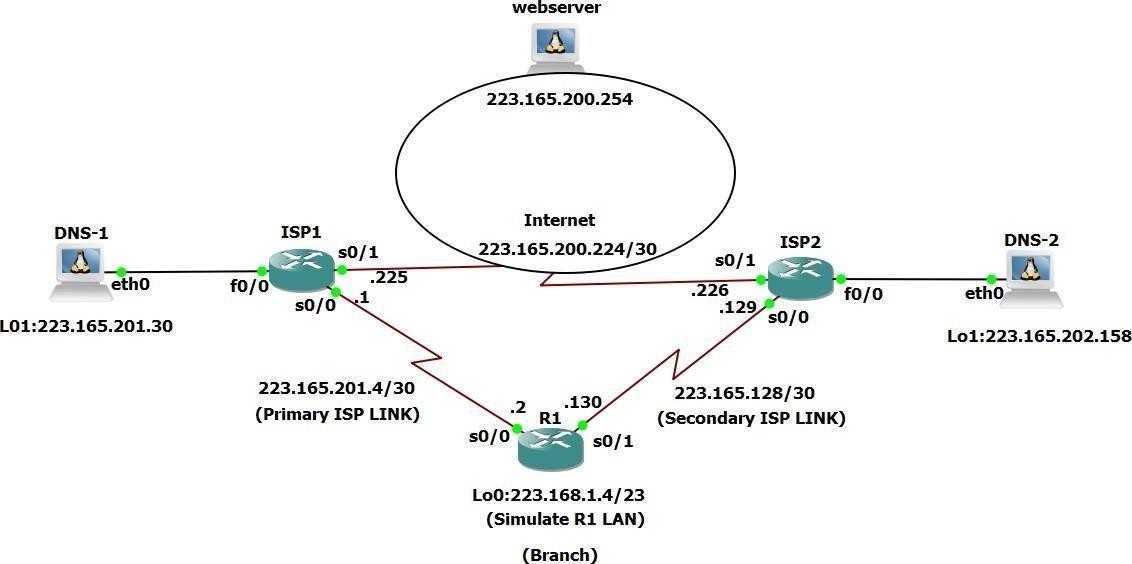
M.Sc. (IT) Coordinator College Seal

|  |  |  |
| --- | --- | --- |
| **INDEX** | | |
| **PRAC NO.** | **PRACTICAL** | **SIGN** |
| **1** | **Configure IP SLA Tracking and Path Control Topology** |  |
| **2** | **Using AS\_PATH Attribute** |  |
| **3** | **Configure IBGP and EBGP Sessions, Local Preference, and MED.** |  |
| **4** | **Secure Management Plane** |  |
| **5** | **Configure and Verify Path Control Using PBR** |  |
| **6** | **Cisco MPLS Configuration** |  |
| **7** | **Inter-VLAN Routing** |  |
| **8** | **Cisco MPLS Configuration** |  |

# Practical No-1

**Aim:** Configure IP SLA Tracking and Path Control Topology

###### 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

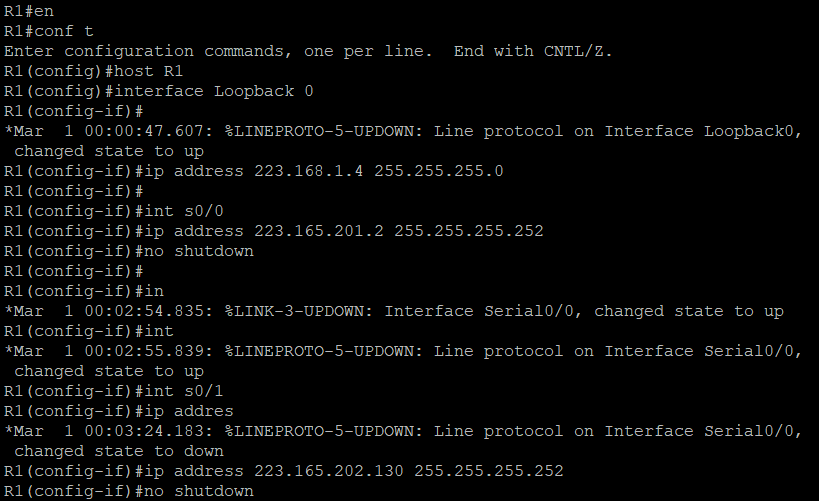
Ip address 223.168.1.4 255.255.255.0

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



###### 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

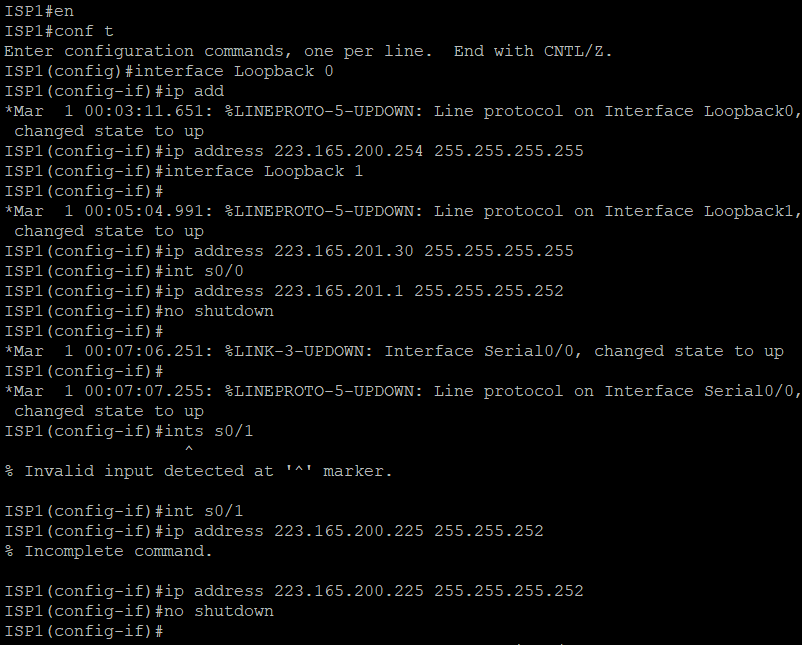
Int s0/0

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



**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

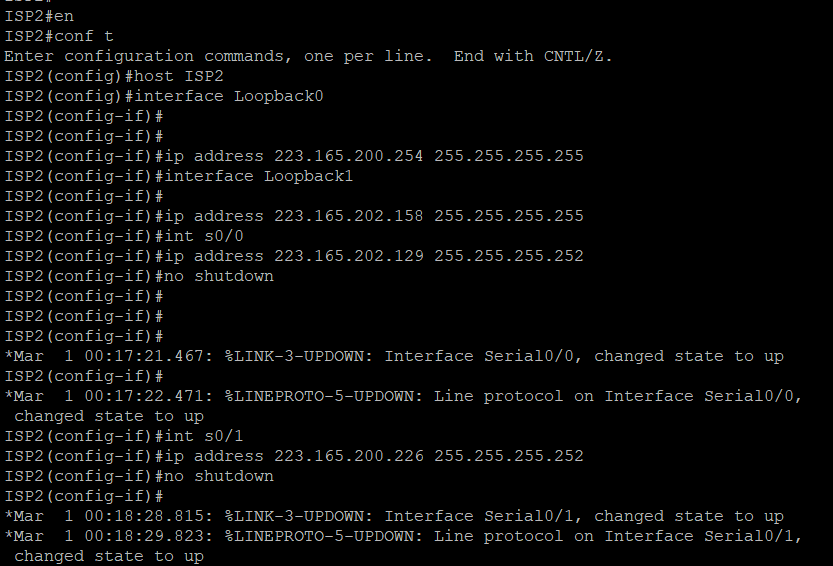
Int s0/0

Ip address 223.165.202.129 255.255.255.252

no shutdown #

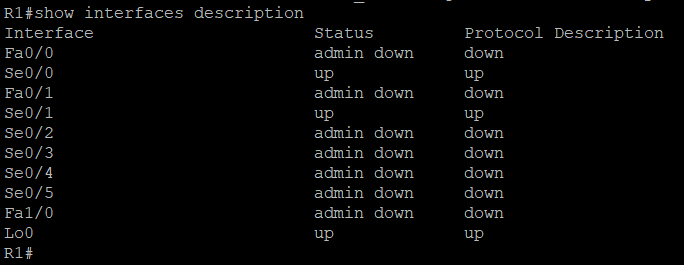
int s0/1

ip address 223.165.200.226 255.255.255.255



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

###### R1# show interfaces description



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

-- Router R1 establishes connectivity to the Internet through ISP1 using a default static route.

-- ISP1 and ISP2 have dynamic routing enabled between them, advertising their respective public address pools.

-- ISP1 and ISP2 both have static routes back to the ISP LAN.

###### Router R1 ip route 0.0.0.0 0.0.0.0 223.165.201.



**Router ISP1 (R2)**

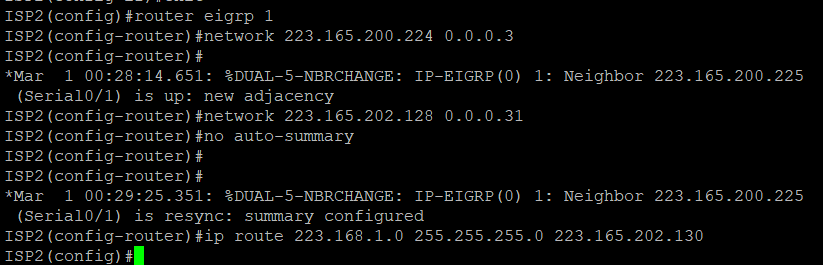
Router eigrp 1

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



###### Router ISP2 (R3)

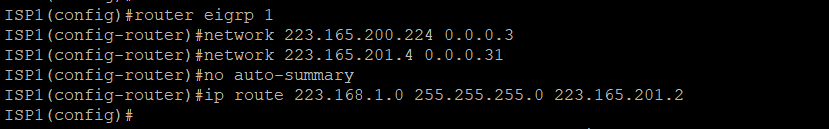
Router eigrp 1

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

###### Step 2: Verify server reachability.

1. 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

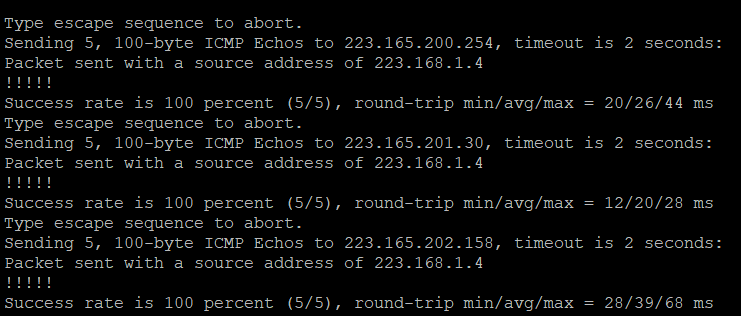
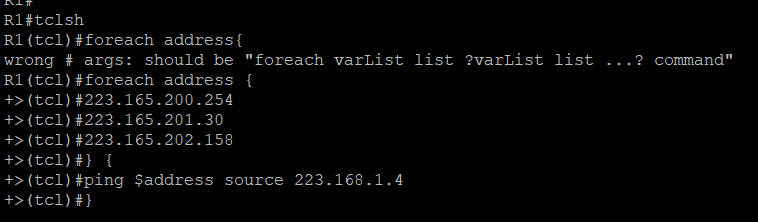
+>(tcl) #223.165.201.30

+>(tcl) #223.165.202.158

+>(tcl) #} {

+>(tcl) #ping $address source 223.168.1.4

+>(tcl) #}



1. 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

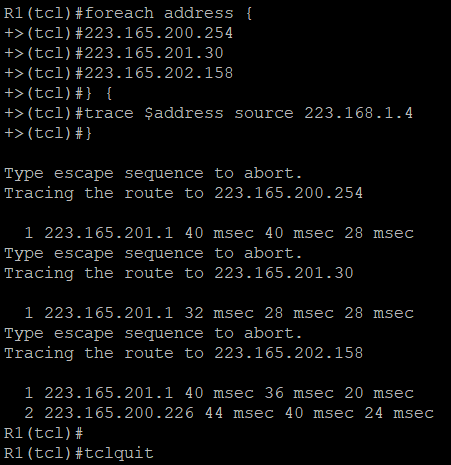
+>(tcl) #223.165.201.30

+>(tcl) #223.165.202.158

+>(tcl) #} {

+>(tcl) # trace $address source 223.168.1.4

+>(tcl) #}

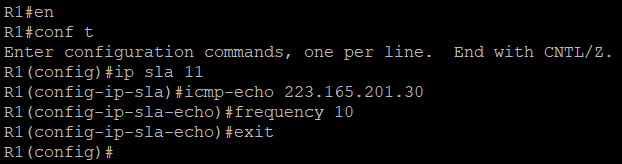


###### Step 3: Configure IP SLA probes.

1. 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.

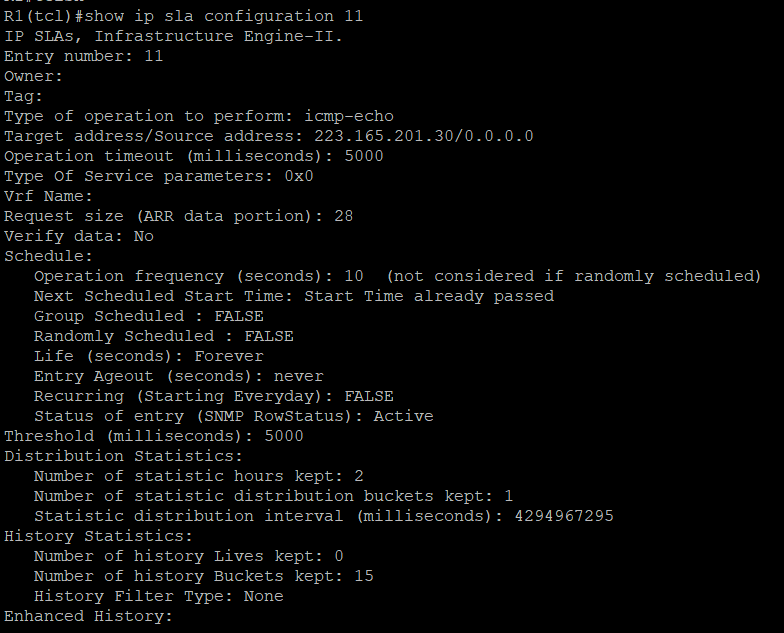
R1(config)# ip sla 11

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



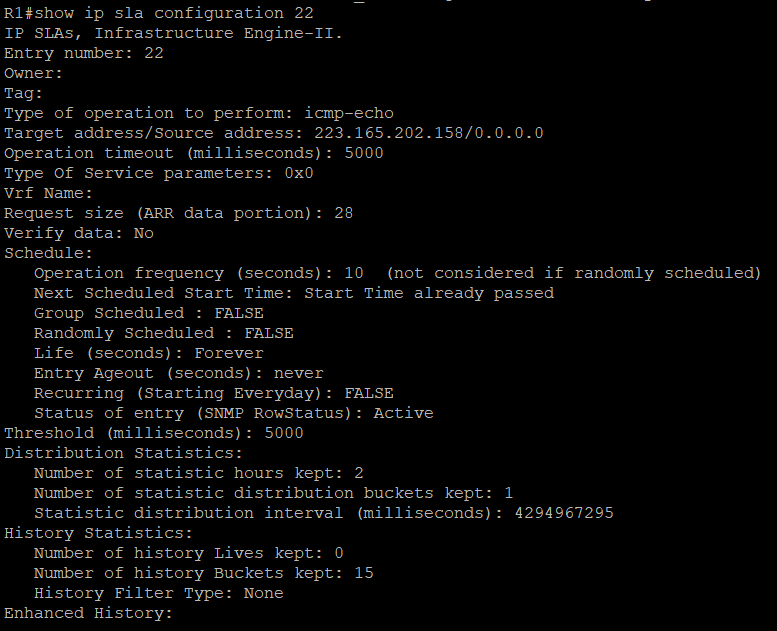
1. Verify the IP SLAs Configuration of operation 11 using the show ip sla configuration 11 command.

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

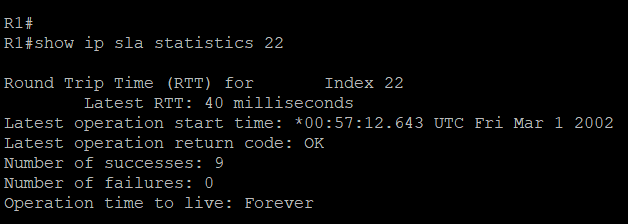


1. Issue the show ip sla statistic command to display the number of successes, failures, and results of the latest operations.
2. Although not actually required because IP SLA session 11 alone could provide the desired fault tolerance, create a second probe,22, to test connectivity to the second DNS server located on router ISP2. You can copy and paste the following commands on R1.
3. Verify the new probe using the show ip sla configuration and show ip sla statistics commands.

###### R1# show ip sla configuration 22



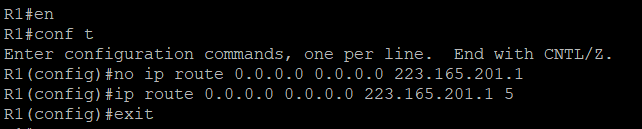
**R1# show ip sla statistics 22**



###### Step 4: Configure tracking options.

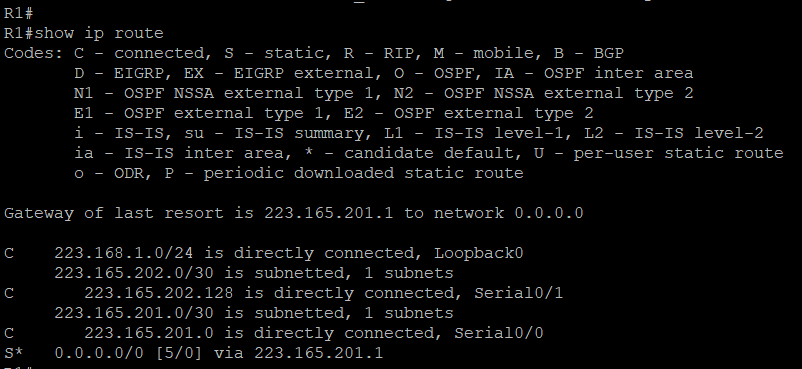
1. Remove the current default route on R1, and replace it with a floating static route having an administrative distance of 5.

R1(config) # no ip route 0.0.0.0 0.0.0.0 223.165.201.1



R1(config) #ip route 0.0.0.0 0.0.0.0 223.165.201.1 5

###### Verify the Routing table.



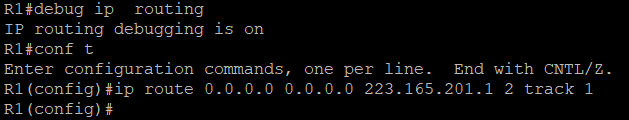
**R1# show ip route**

1. Use the trace 1 ip sla 11 reachability command to enter the config-track sub- Configuration mode.

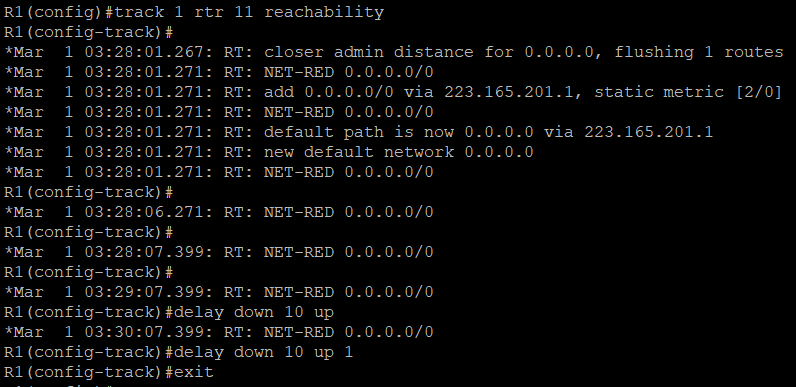
R1(config)# track 1 ip sla 11 reachability. R1(config-track) #

1. Configure the floating static route that will be implemented with when tracking object 1 is active. To view routing table changes as they happen, first enable the debug ip routing command. Next, use the ip route 0.0.0.0 0.0.0.0 223.165.201.1 2 track 1 command to create a floating static default route via 223.165.201.1 (ISP1). Notice that this command references the tracking object number 1, which in turn references IP SLA operation number 11.

###### R1# debug ip routing:

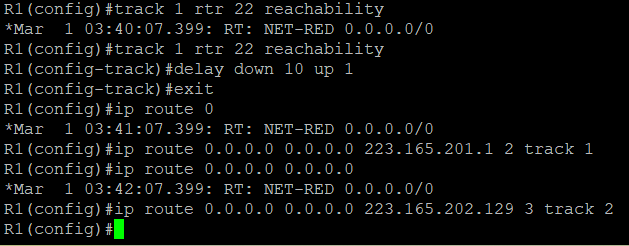


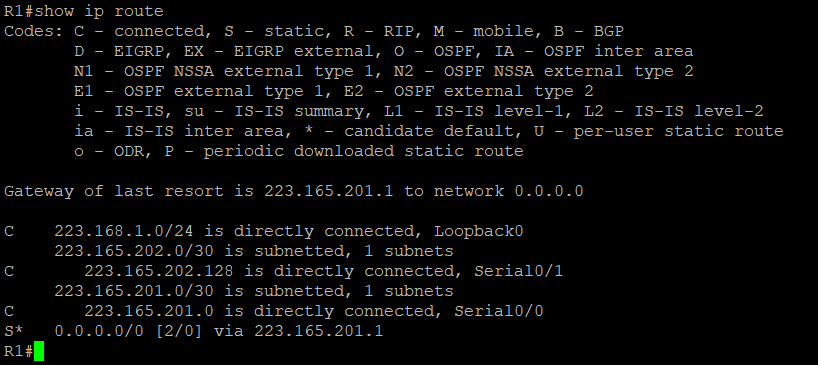
**R1(config)# ip route 0.0.0.0 0.0.0.0 223.165.201.1 2 track 1**



1. Repeat the steps for operation 22, track number 2, and assign the static route an admin distance higher than track 1 and lower than 5. On R1, copy the following configuration, which sets an admin distance of 3. Track 2 ip sla 22 reachability delay down 10 up 1 exit.

###### ip route 0.0.0.0 0.0.0.0 223.165.202.129 3 track 2

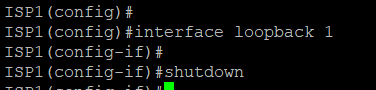


1. **Verify the Routing table again. R1# show ip route**

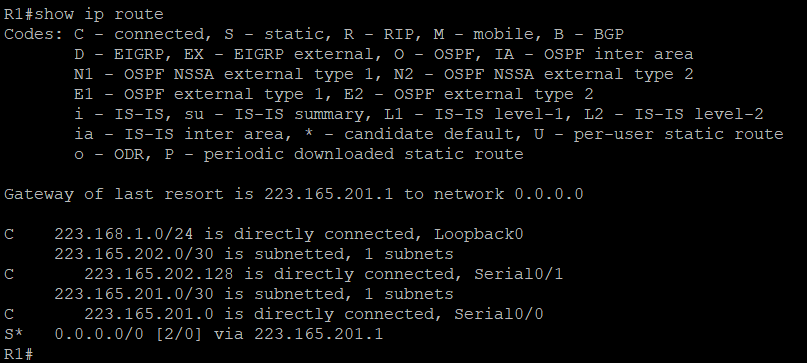
###### Step 5: Verify IP SLA operation.

The following summarizes the process:

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

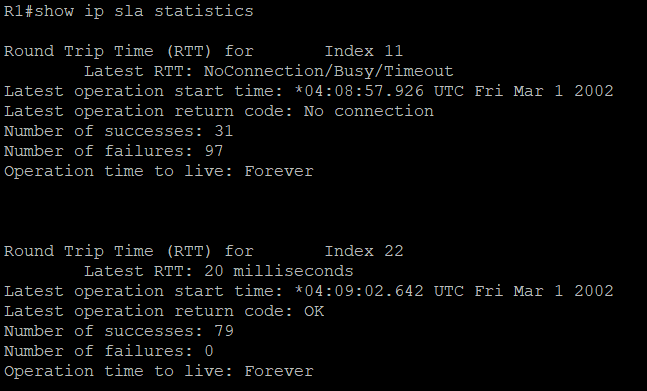
ISP1(config)# interface loopback 1 ISP1(config-if) #shutdown

1. Verify the routing table.

**R1# show ip route**

1. Verify the SLA statistics.

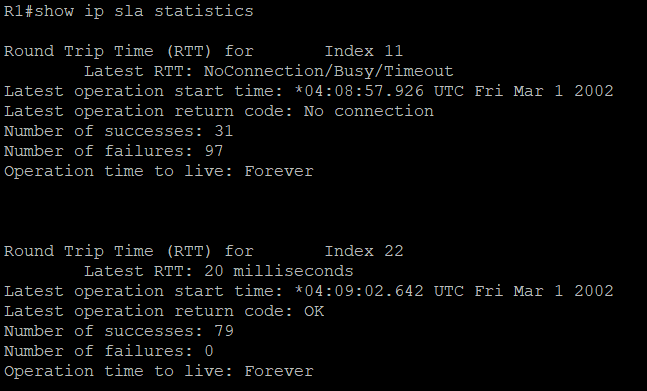
###### R1# show ip sla statistics



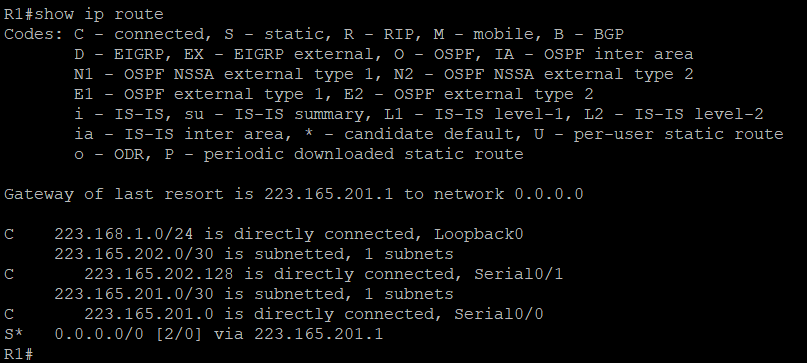
1. Initiate a trace to the web server from the internal LAN IP address.

###### R1# trace 223.165.200.254 source 223.168.1.4

1. **Again Examine the IP SLA statistics. R1# show ip sla statistics**



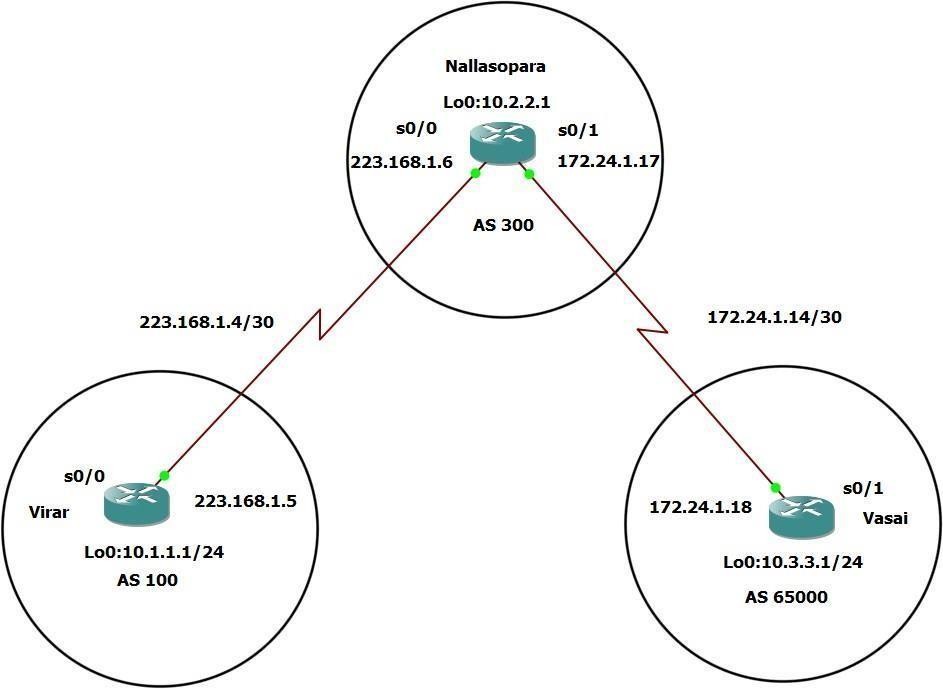
**g. Verify the Routing Table. R1# show ip route**



# Practical No-2

**Aim:** Using AS\_PATH Attribute

**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.

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

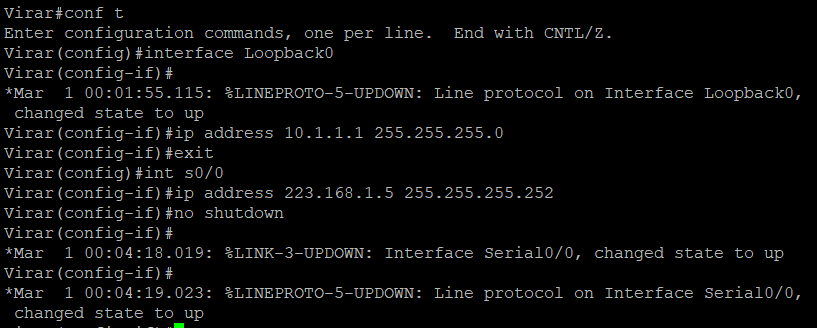
Router R1(hostname Virar) Virar(config)# interface Loopback 0

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#



###### 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**

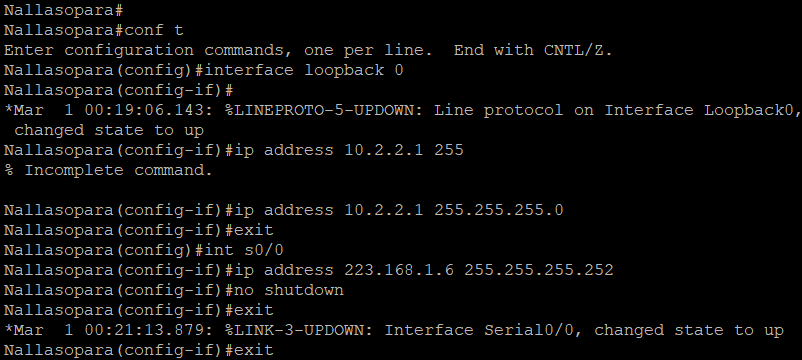
###### Nallasopara(config-if) #int s0/0

**Nallasopara(config-if) #ip address 223.168.1.6 255.255.255.252 Nallasopara(config-if) #no shutdown**

###### Nallasopara(config-if) #exit

**Nallasopara(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#**

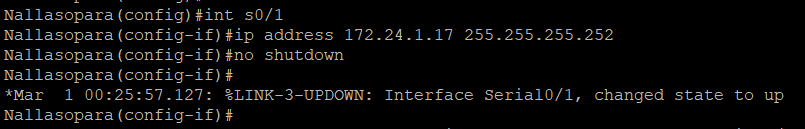
###### Router 3(hostname Vasai) Vasai(config)# interface Loopback 0

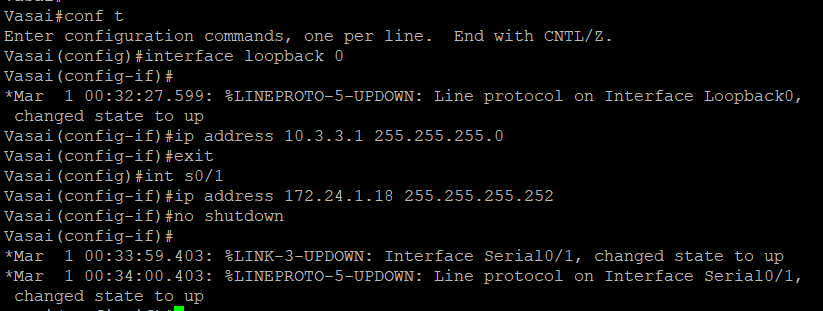
**Vasai(config-if) #ip address 10.3.3.1 255.255.255.0 Vasai(config-if) #exit**

###### Vasai(config-if) #int s0/1

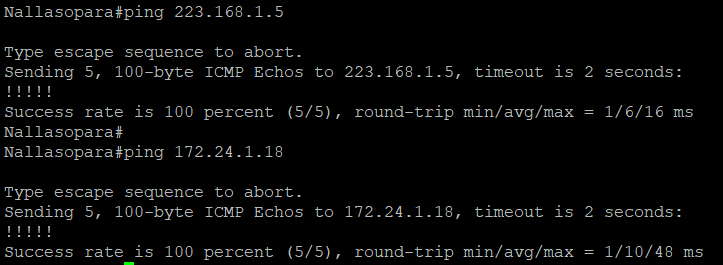
**Vasai(config-if) #ip address 172.24.1.18 255.255.255.252 Vasai(config-if) #no shutdown**

###### Vasai(config-if) #end Vasai#





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

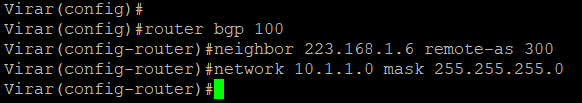


###### Step 3: Configure BGP.

1. 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

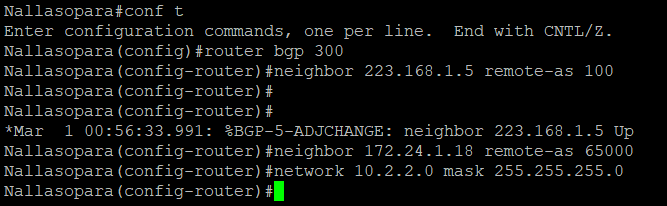
Virar(config-router) #network 10.1.1.0 mask 255.255.255.0 Virar(config-router) #

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

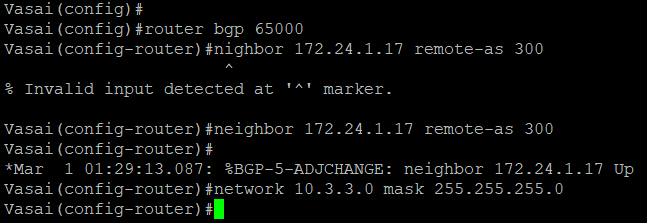
Nallasopara(config-router) #10.20.2.0 mask 255.255.255.0



Vasai(config) #router bgp 65000

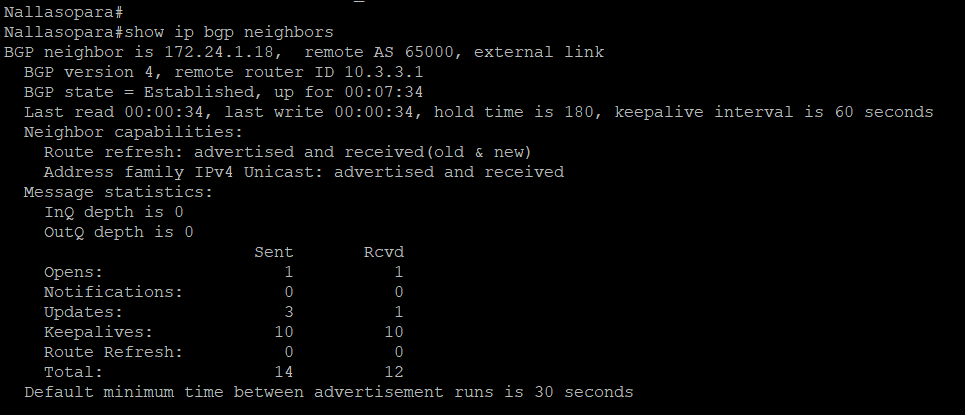
Vasai(config-router) #neighbor 172.24.1.17 remote-as 300

Vasai(config-router) #network 10.3.3.0 mask 255.255.255.0



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

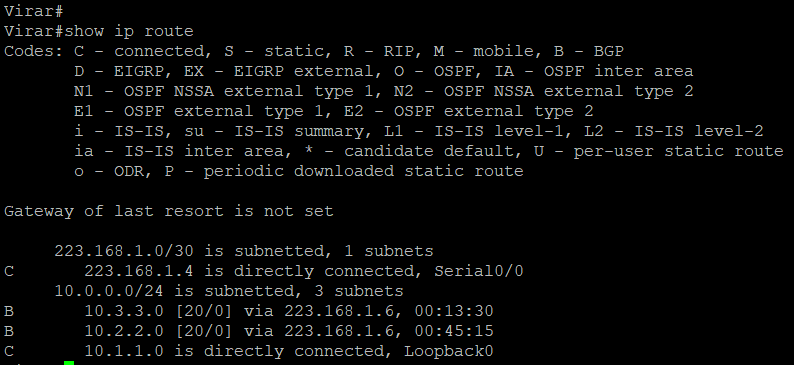
###### Nallasopara# show ip bgp neighbors



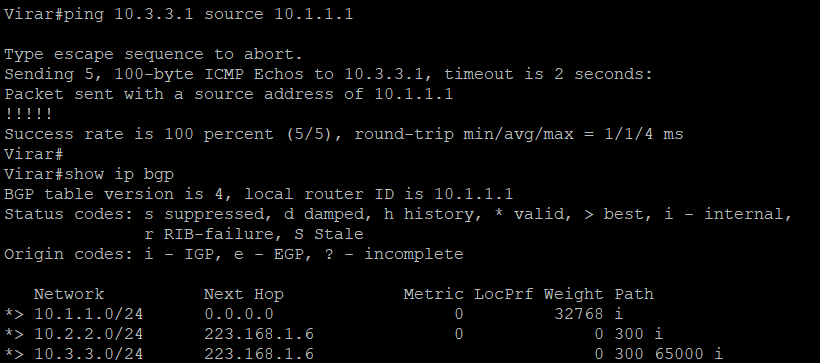
**Step 4: Remove the Private AS.**

* 1. 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

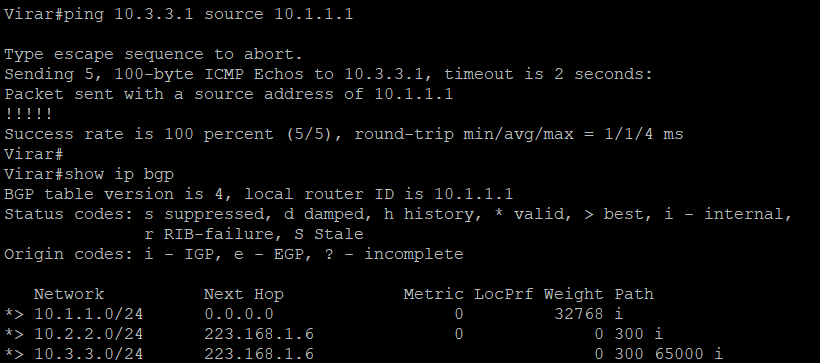


* 1. Ping Again, this time as an extended ping, sourcing from the Loopback 0 interface address. Ping 10.3.3.1 source 10.1.1.1 or ping 10.3.3.1 source Lo0



* 1. 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.

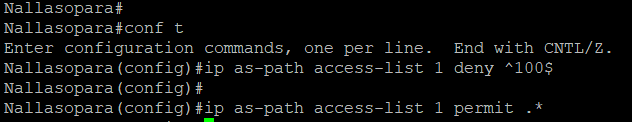
###### Virar# show ip bgp



**Step 5: Use the AS\_PATH attribute to filter routes.**

* + 1. 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.

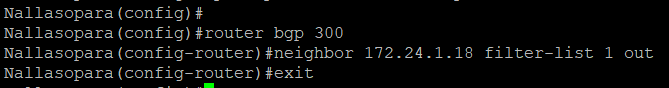
###### Nallasopara(config) #ip as\_path access-list 1 deny ^100$ Nallasopara(config) #ip as-path access-list 1 permit .\*



* + 1. Apply the Configured access list using the neighbor command with the filter-list option.

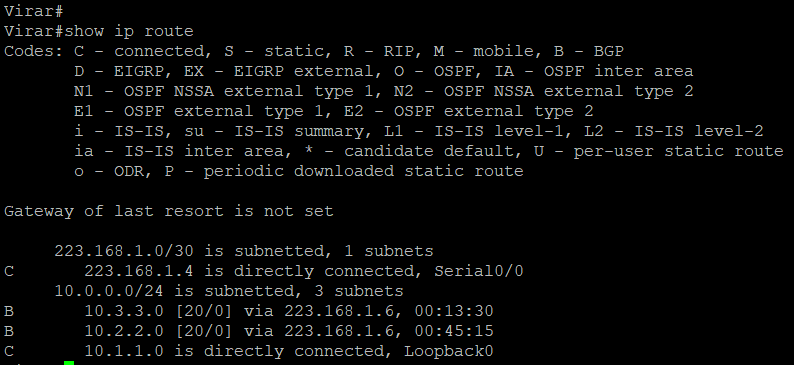
###### Nallasopara(config) #router bgp 300

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



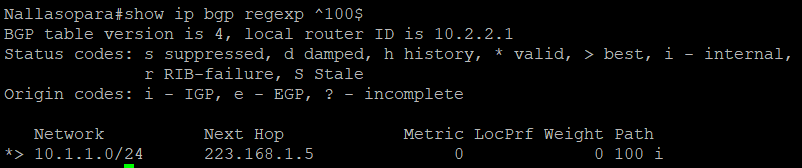
* + 1. Use the clear ip bgp \* command to reset the routing information. Wait several seconds and then check the routing table for Nallasopara. The route 10.1.1.0 should be in the routing table.

###### Virar# show ip route



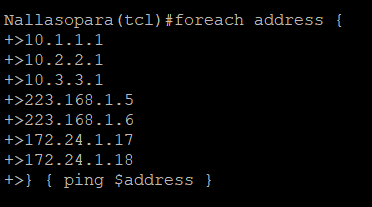
* + 1. Return to Nallasopara and Verify that the filter is working as intended.

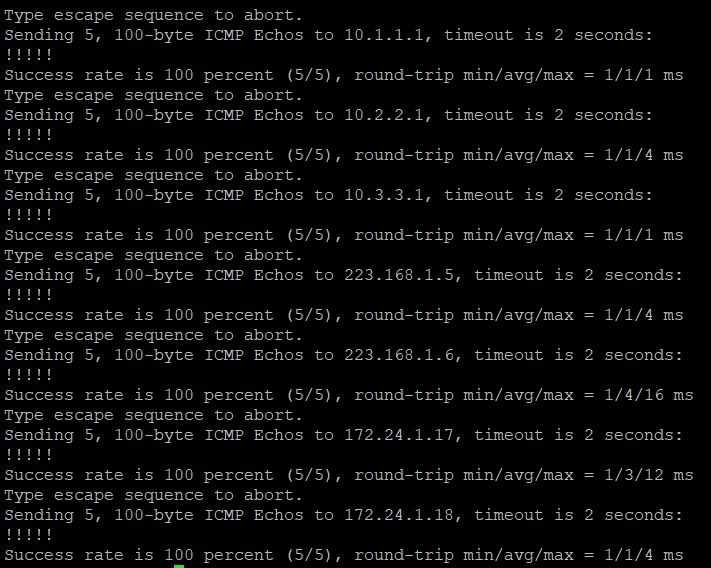
###### Nallasopara# show ip bgp regexp ^100$



* + 1. 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.

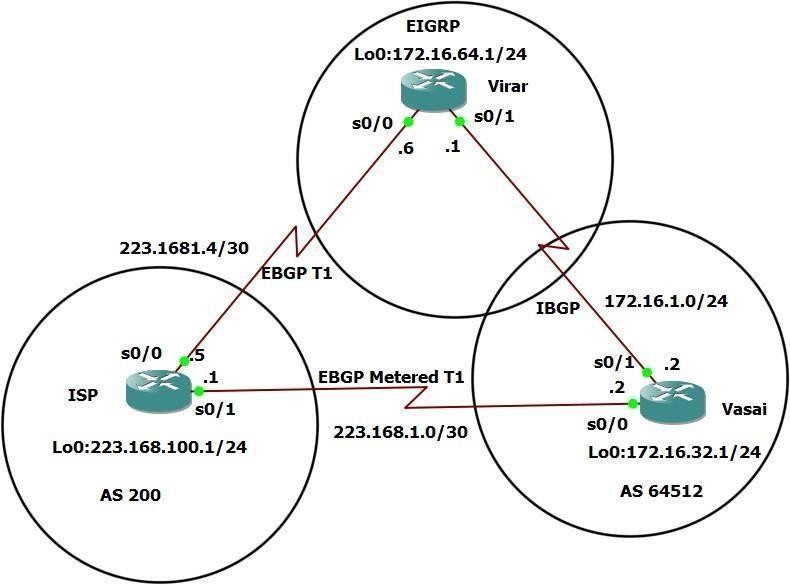




Practical No – 3

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

**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

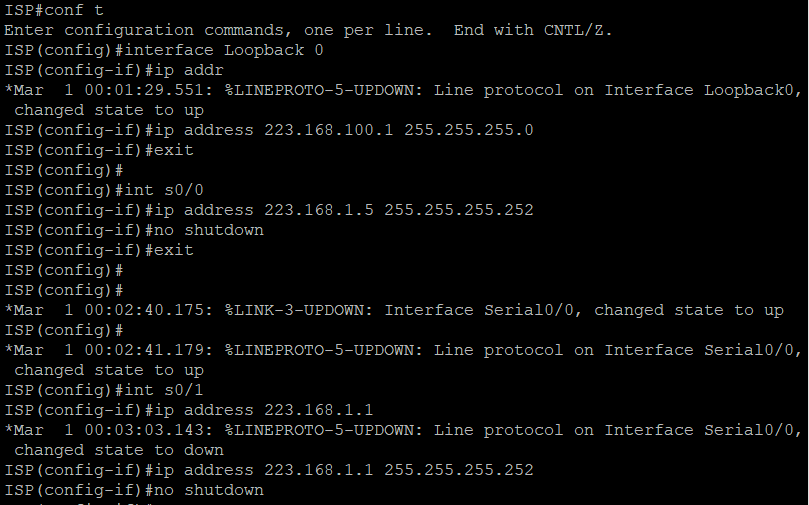
ISP (config) #interface Serial 0/0/0

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



###### 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

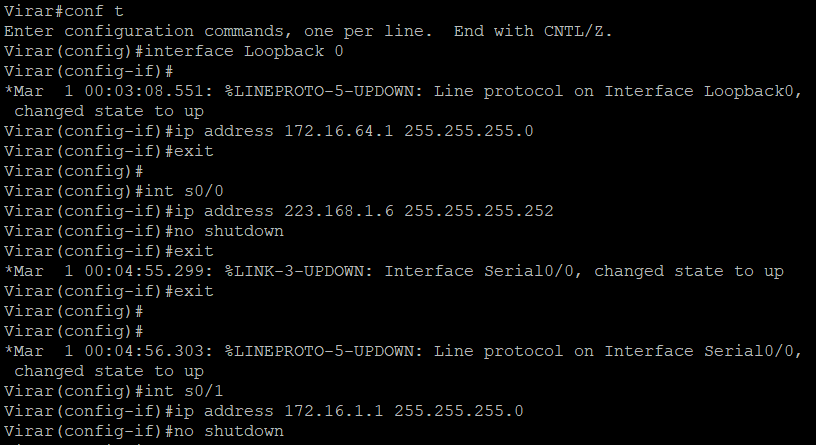
Virar(config-if) #ip address 223.168.1.6 255.255.255.252 Virar(config-if) #no shutdown

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



###### 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

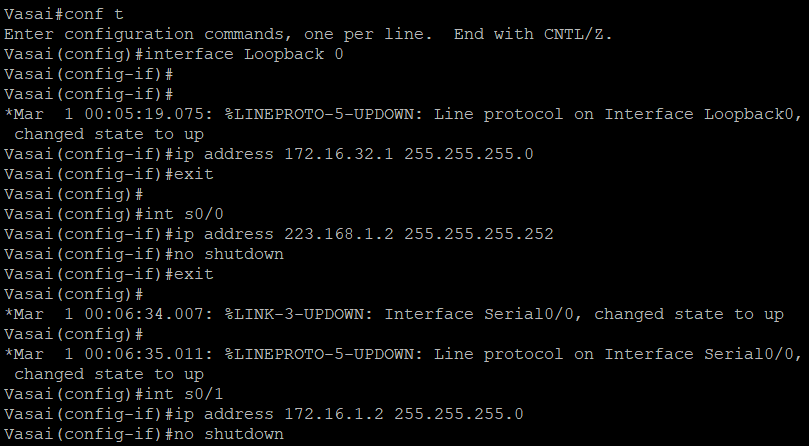
Vasai(config-if) #ip address 223.168.1.2 255.255.255.252 Vasai(config-if) #no shutdown

Vasai(config-if) #exit

Vasai(config) #interface Serial 0/0/1

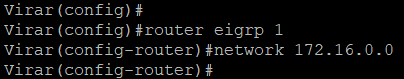
Vasai(config-if) #ip address 172.16.1.2 255.255.255.0 Vasai(config-if) #no shutdown

Vasai(config-if) #end

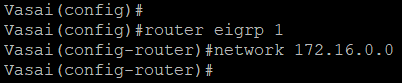


###### Step 2: Configure EIGRP.

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

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



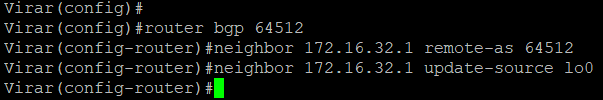
###### Step 3: Configure IBGP and Verify BGP neighbors.

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

Virar(config) #router bgp 64512

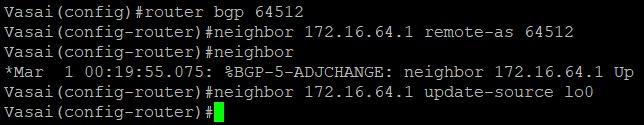
Virar(config-router) #neighbor 172.16.32.1 remote-as 64512

Virar(config-router) #neighbor 172.16.32.1 update-source Lo0

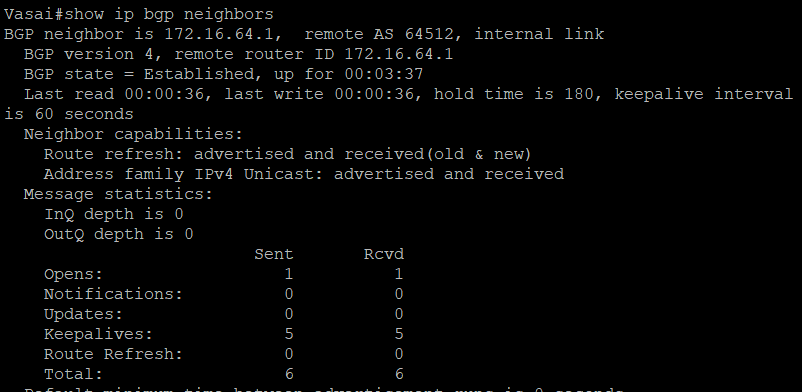


If multiple pathways to the BGP neighbor exist, the router can use multiple IP interfaces to communicate with the neighbor. The source IP address therefore depends on the outgoing interface. The update-source Lo0 command instructs the router to use the IP address of the interface Loopback 0 as the source IP address for all BGP messages sent to that neighbor.

1. 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

1. 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.

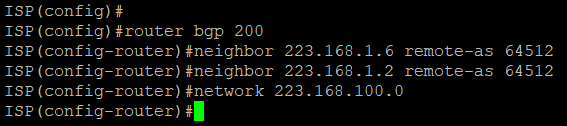


###### Step 4: Configure EBGP and Verify BGP neighbors.

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

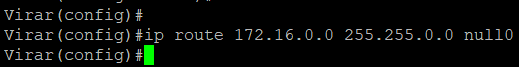
ISP (config) #router bgp 200

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

1. 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.

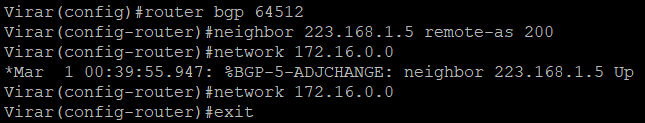
Virar(config) #ip route 172.16.0.0 255.255.0.0 null0



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

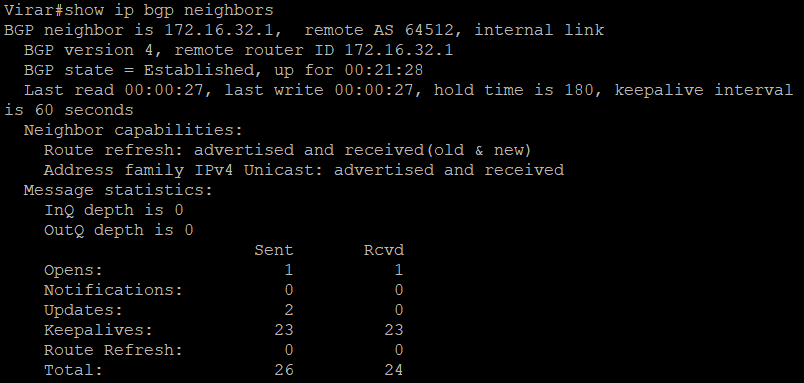
Virar(config-router) #neighbor 223.168.1.5 remote-as 200

Virar(config-router) #network 172.16.0.0



1. 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**

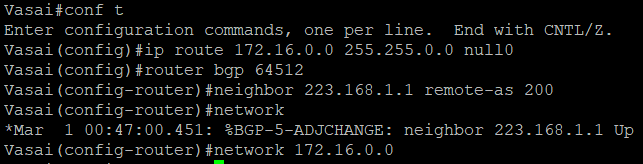


Configure a discard static route for 172.16.0.0/16 on Vasai and an EBGP peer to ISP. Vasai(config) #ip route 172.16.0.0 255.255.0.0 null0

Vasai(config) #router bgp 64512

Vasai(config-router) #neighbor 223.168.1.1 remote-as 200

Vasai(config-router) #network 172.16.0.0

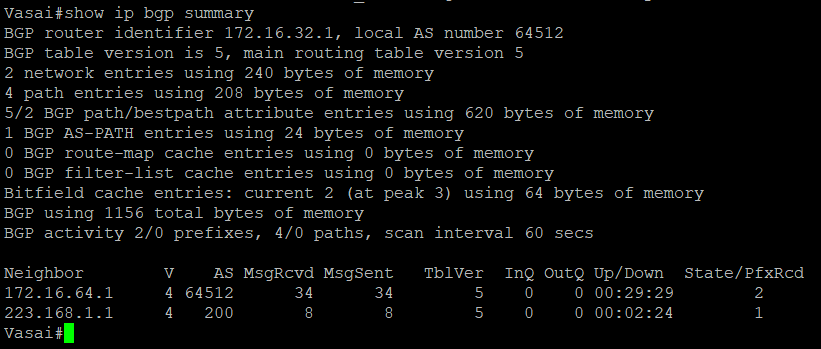


###### Step 5: View BGP summary output.

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.

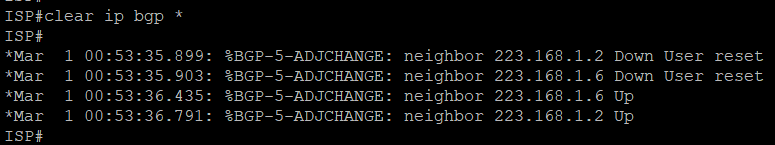
Vasai# **show ip bgp summary**



###### Step 6: Verify which path the traffic takes.

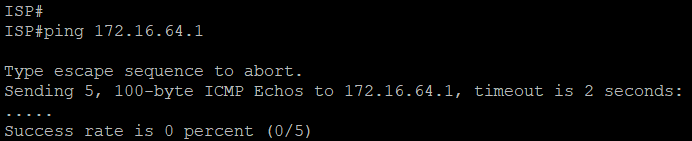
1. Clear the IP BGP conversation with the clear ip bgp \* command on ISP. Wait for the conversation to re-establish with each Virar router.

###### ISP# clear ip bgp \*

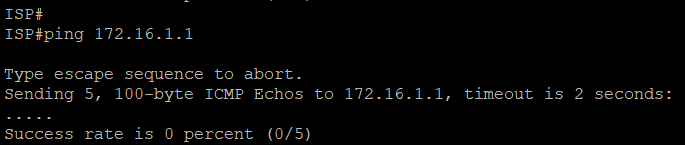


1. Test whether ISP can ping the Loopback 0 address of 172.16.64.1 on Virar and the serial link between Virar and Vasai, 172.16.1.1

###### ISP# ping 172.16.64.1

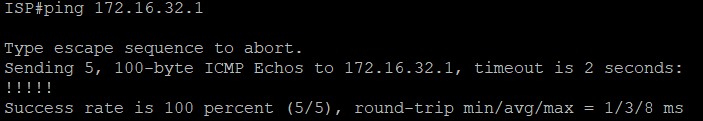


**ISP# ping 172.16.1.1**

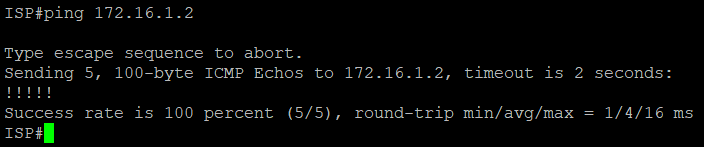


1. Now ping from ISP to the Loopback 0 address of 172.16.32.1 on Vasai and the Serial link between Virar and Vasai, 172.16.1.2.

ISP# **ping 172.16.32.1**

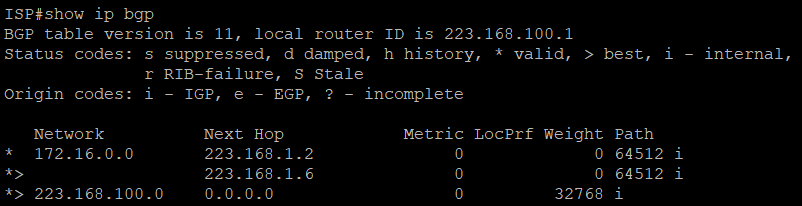


###### ISP# ping 172.16.1.2



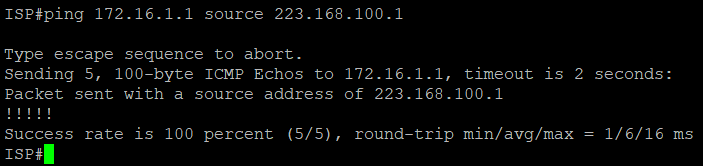
**I.** Issue the show ip bgp command on ISP to verify BGP routes and metrics.

###### ISP# show ip bgp

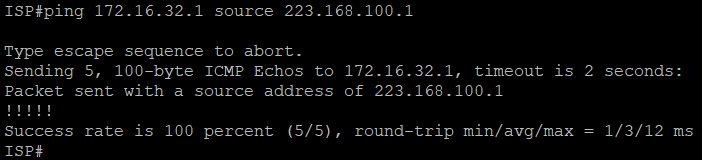


**i.** At this point, the ISP router should be able to get to each network connected to Virar and Vasai from the Loopback address 223.168.100.1. Use the extended ping command and specify the source address of ISP Lo0 to test.

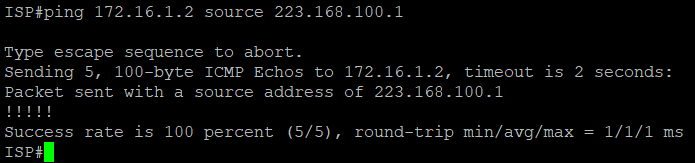
###### ISP# ping 172.16.1.1 source 223.168.100.1



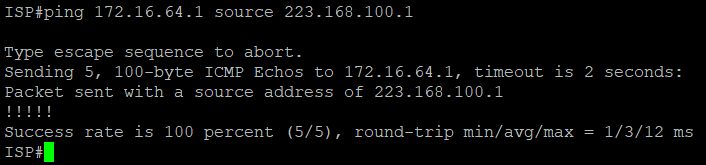
**ISP# ping 172.16.32.1 source 223.168.100.1**



###### ISP# ping 172.16.1.2 source 223.168.100.1



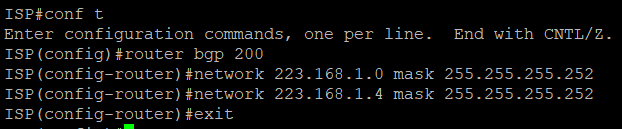
**ISP# ping 172.16.64.1 source 223.168.100.1**



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

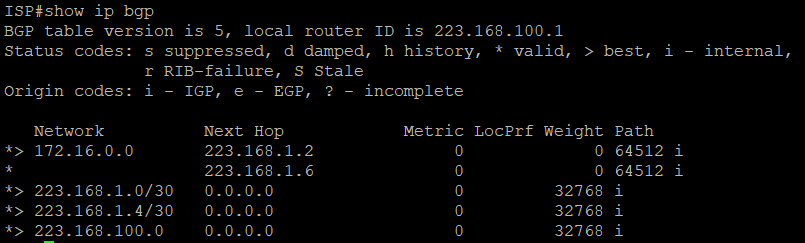
1. Issue the following commands on the ISP router. ISP (config) # router bgp 200

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



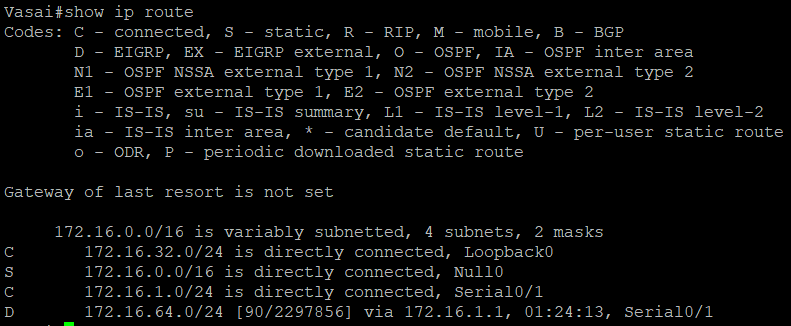
1. Issue the show ip bgp command to verify that ISP is Correctly injecting its own WAN links int BGP.

###### ISP# show ip bgp



1. 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



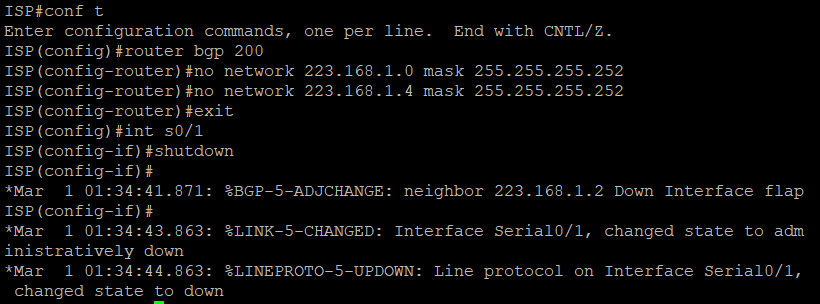
1. To better understand the **next-hop-self** command we will remove ISP advertising its two WAN links and shutdown the WAN link between ISP and Vasai. The only possible path from to ISP’s 223.168.100.0/24 through Virar.

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**

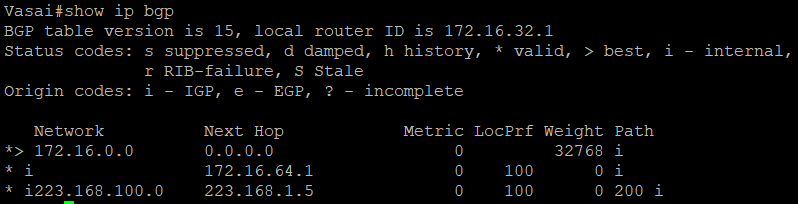
ISP (config) #**interface serial 0/0/1**

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

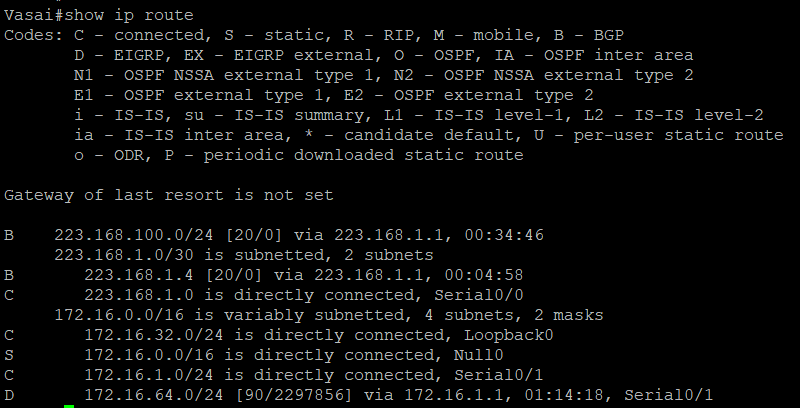


1. Display Vasai BGP table using the **show ip bgp** command and the IPv4 routing table with **show ip route.**

###### Vasai# show ip bgp

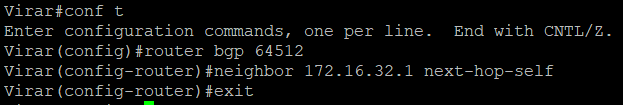


**Vasai# show ip route**



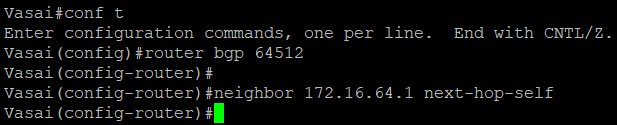
Virar(config) #**router bgp 64512**

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

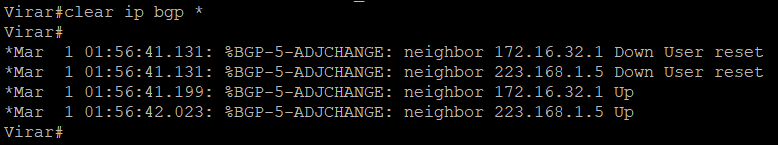


###### Vasai(config) # router bgp 64512

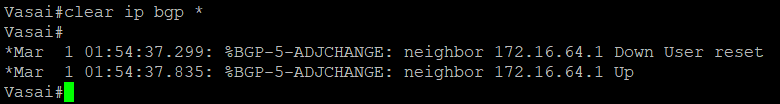
**Vasai(config-router) #neighbor 172.16.64.1 next-hop-self**



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

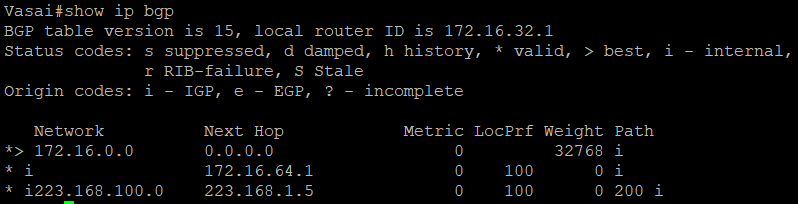


Vasai# **clear ip bgp \***



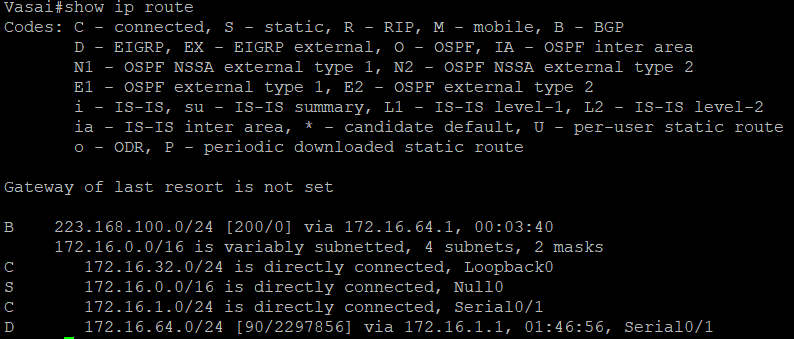
1. After the routers have returned to established BGP Speakers, issues the **show ip bgp**

command on Vasai and notice that the next hop is now Virar instead of ISP.



1. 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.

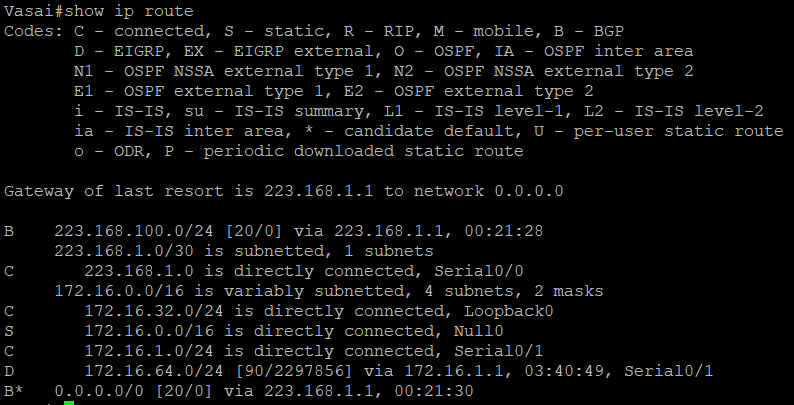
Vasai# **show ip route**



1. 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.

ISP (config)# interface serial 0/0/1 ISP (config-if) #no shutdown

Vasai # **show ip route**



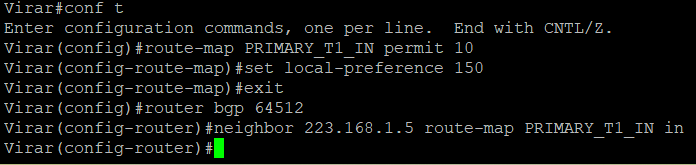
###### Step 8: Set BGP local Preference.

1. 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**

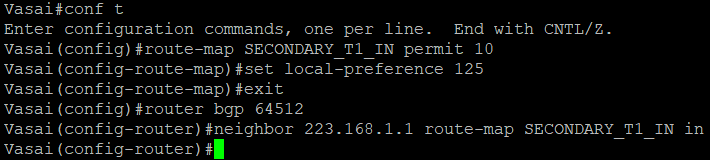
Virar(config-router) #**neighbor 223.168.1.5 route-map PRIMARY\_T1\_IN in**



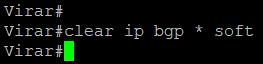
Vasai(config) # **route-map SECONDARY\_T1\_IN permit 10** Vasai(config-route-map) # **set local-preference 125** Vasai(config-router-map) #**exit**

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

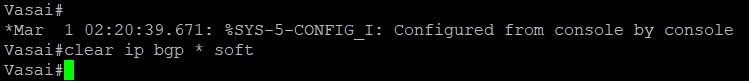
Vasai(config-router) #**neighbor 223.168.1.1 route-map SECONDARY\_T1\_IN in**



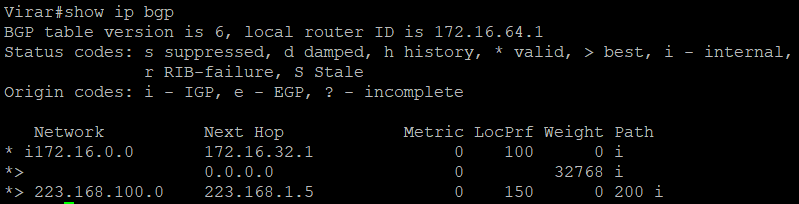
1. Use the clear ip bgp \* soft command after configuring this new policy. When the Conversations have been re-established, issue the show ip bgp command on Virar and Vasai.

Virar# **clear ip bgp \* soft**

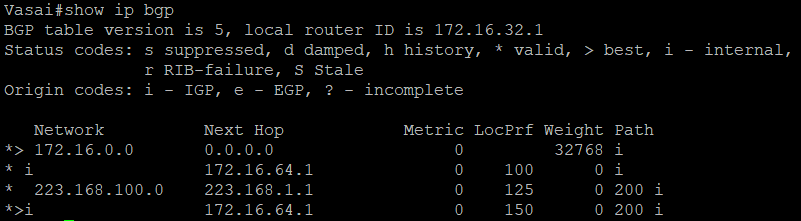
###### Vasai# clear ip bgp \* soft



Virar# **show ip bgp**



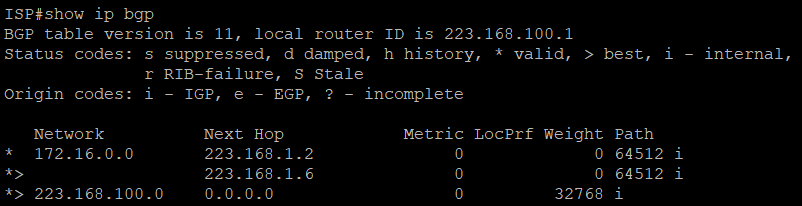
###### Vasai# show ip bgp



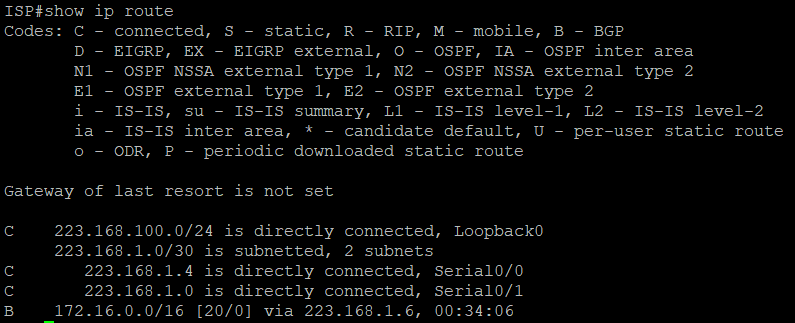
**Step 9: BGP MED.**

1. 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.

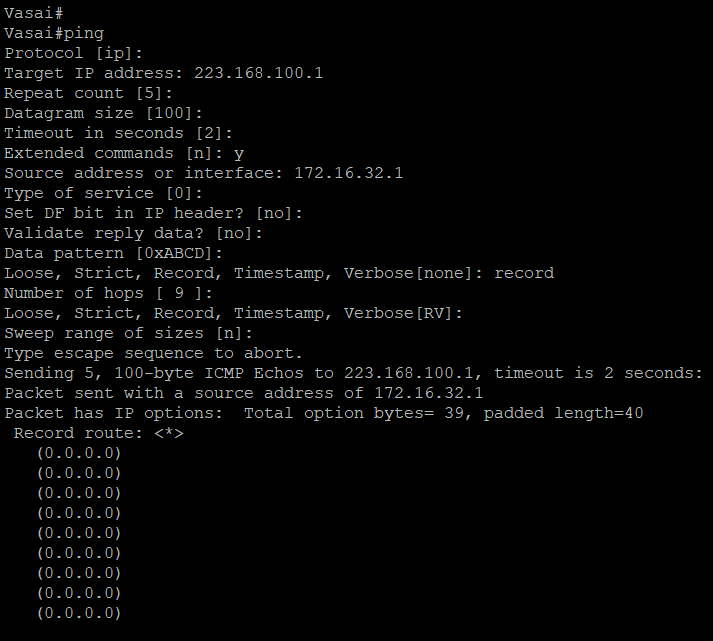
###### ISP# show ip bgp

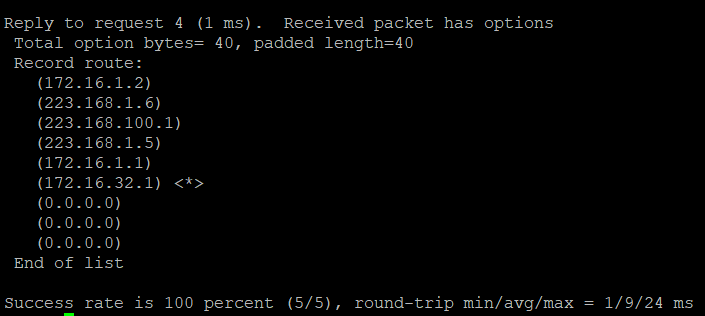


**ISP# show ip route.**



1. 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.





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.

1. 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.
2. Vasai then forwards the packet out the last interface, loopback 0, 172.16.32.1.

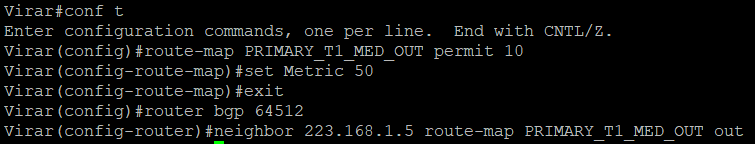
Although the unlimited use of the T1 from Virar is preferred here, ISP currently takes the link from Vasai for all return traffic.

1. 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

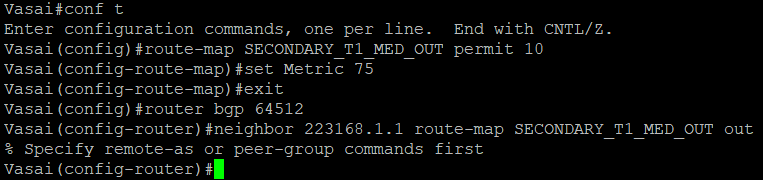
Virar(config-router) #neighbor 223.168.1.5 route-map PRIMARY\_T1\_MED\_OUT out



Vasai(config) #route-map SECONDARY\_T1\_MED\_OUT permit 10 Vasai(config-route-map) # set metric 75

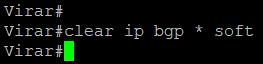
Vasai(config-route-map) # exit Vasai(config) #router bgp 64512

Vasai(config-router) #neighbor 223.168.1.1 route-map SECONDARY\_T1\_MED\_OUT out

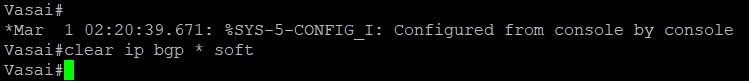


1. Use the clear ip bgp \* soft command after issuing this new policy. Issuing the show ip bgp command as follows on Virar or Vasai does not indicate anything about this newly defined policy.

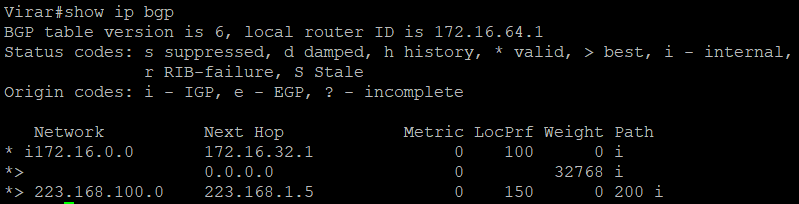
###### Virar# show ip bgp \* soft



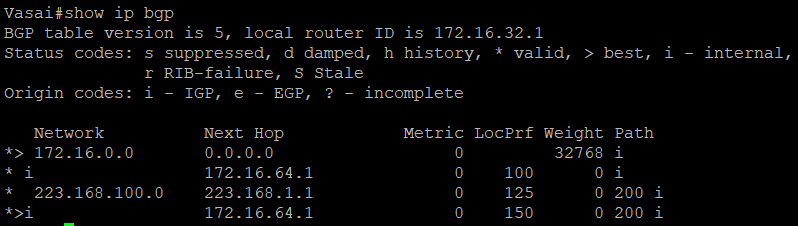
**Vasai# show ip bgp \* soft**



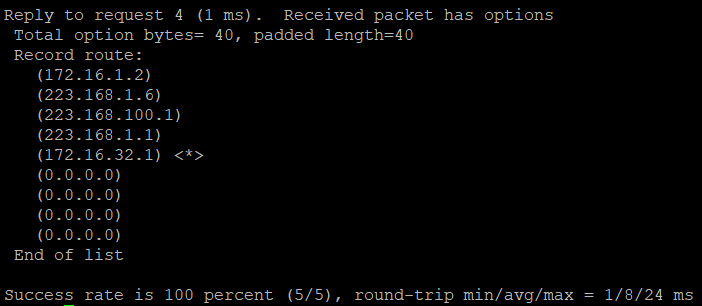
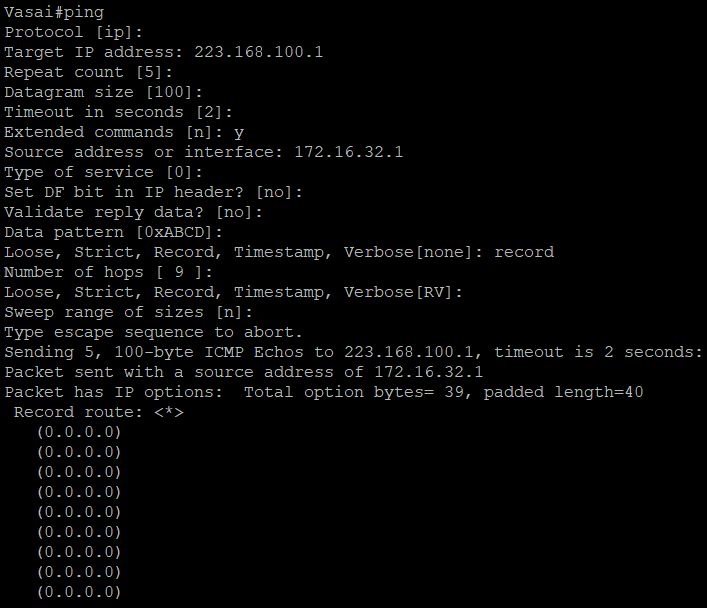
###### Virar# show ip bgp



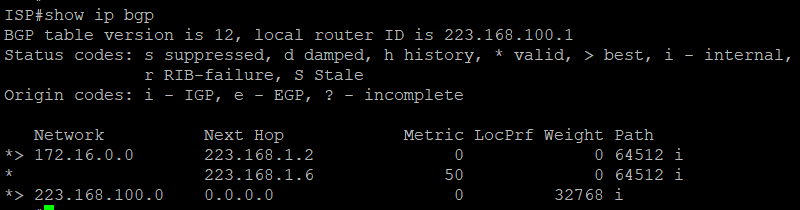
**Vasai# show ip bgp**



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.



###### ISP# show ip bgp



**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.

1. Configure ISP to inject a default route to both Virar and Vasai using BGP using the

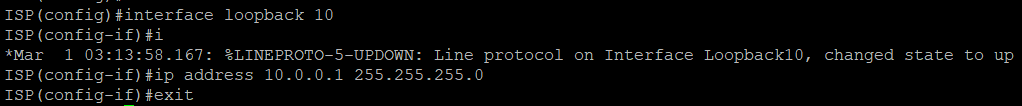
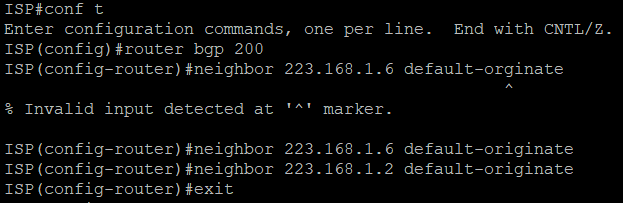
**Default-originate** command. This command does not require the presence of 0.0.0.0 in the ISP router. Configure the 10.0.0.0/8 network which will not be advertised using BGP. This will be used to test the default route on Virar and Vasai.

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

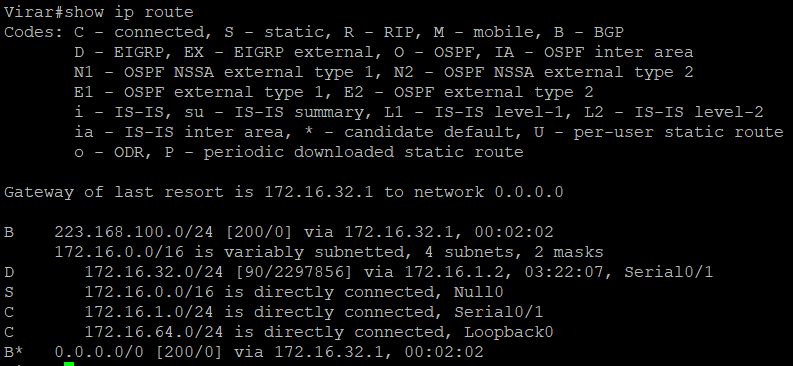
ISP (config) # interface Loopback 10

ISP (config-if) #ip address 10.0.0.1 255.255.255.0

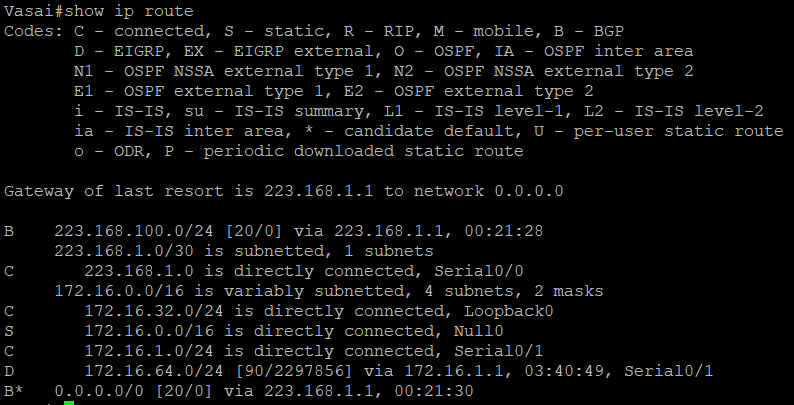


1. Verify that both routers have received the default route by examining the routing tables on Virar and Vasai. Notice that both routers prefer the route between Virar and ISP.

Virar# show ip route

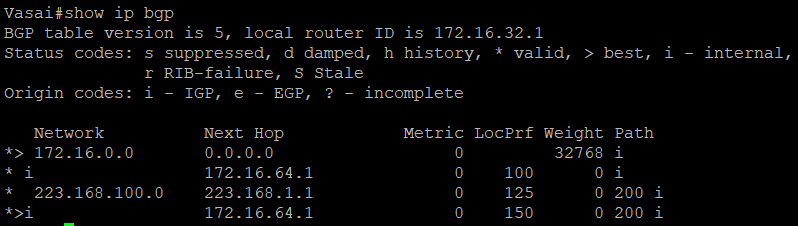


###### Vasai# show ip route



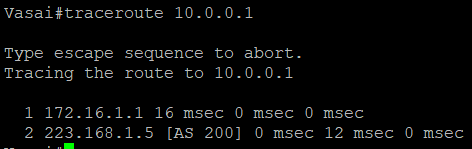
1. The preferred default route is by way of Virar because of the higher local preference attribute configured on Virar earlier.

###### Vasai# show ip bgp



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

###### Vasai# traceroute 10.0.0.1



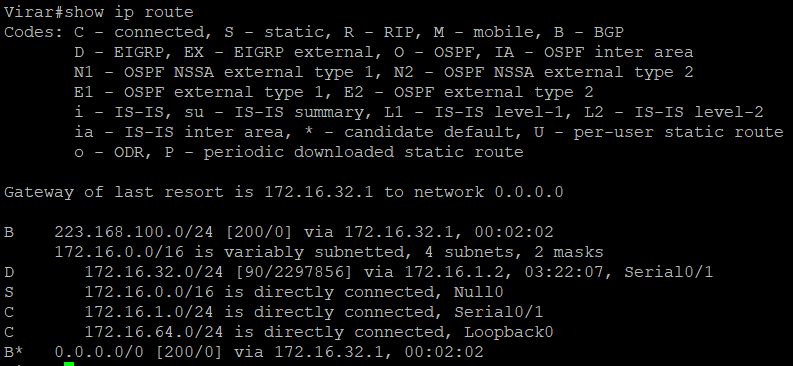
1. Next, test how BGP adapts to using a different default route when the path between Virar and ISP goes down.

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

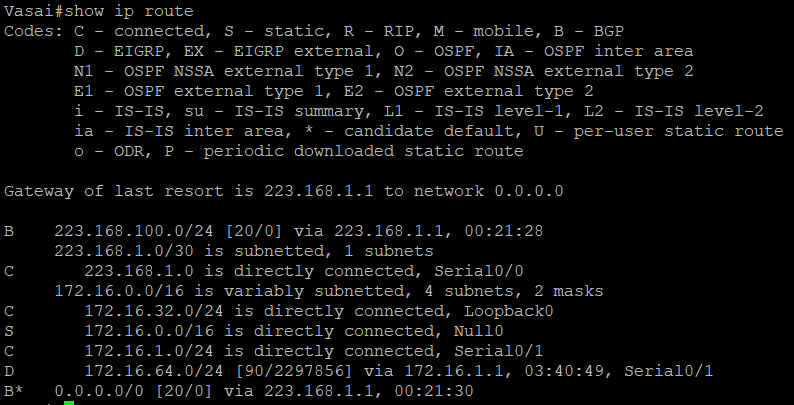


1. Verify that both routers are modified their routing tables with the default route using the path between Vasai and ISP.

###### Virar# show ip route

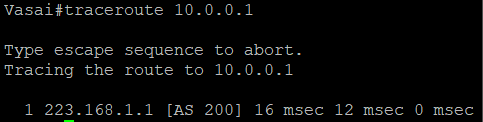


**Vasai# show ip route**



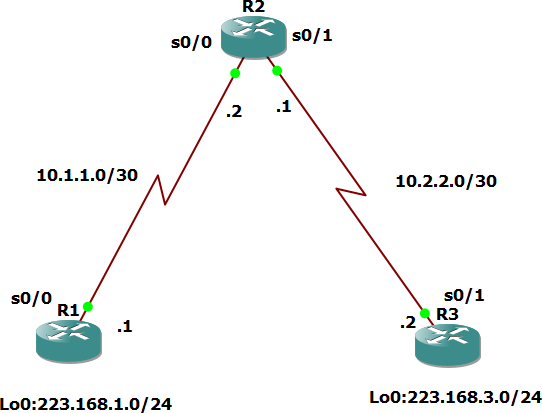
1. Verify the new path using the traceroute command to 10.0.0.1 from Virar. Notice default route is now through Vasai.

**Vasai# trace 10.0.0.1**



# Practical No 4

#### Aim: Secure Management Plane Topology:



Objectives:

Secure management access.

− Configure enhanced username password security.

− Enable AAA RADIUS authentication.

− Enable secure remote management.

#### Step 1: Configure loopbacks and assign addresses.

Cable the network as shown in the topology diagram. Erase the startup configuration and reload 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

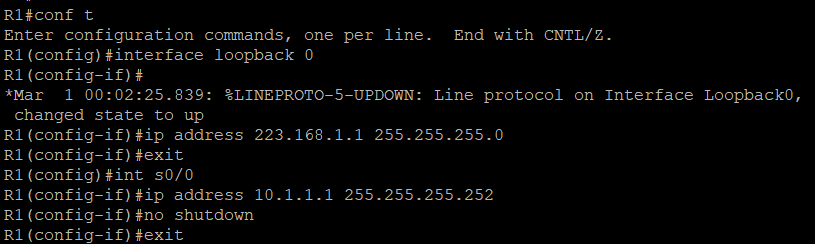
exit

interface Serial0/0/0

ip address 10.1.1.1 255.255.255.252

no shutdown exit

end



#### Router R2

interface Serial0/0/0

ip address 10.1.1.2 255.255.255.252

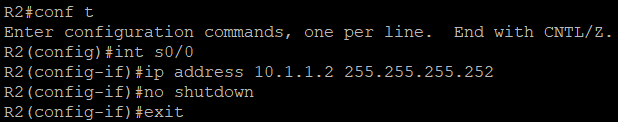
no shutdown

exit interface Serial0/0/1

ip address 10.2.2.1 255.255.255.252

no shutdown exit

end



#### Router R3

interface Loopback0

ip address 223.168.3.1 255.255.255.0

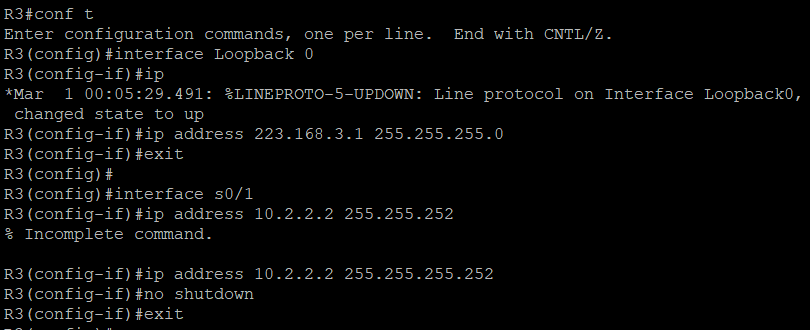
exit

interface Serial0/0/1

ip address 10.2.2.2 255.255.255.252

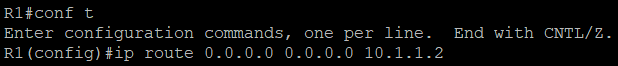
no shutdown exit

end

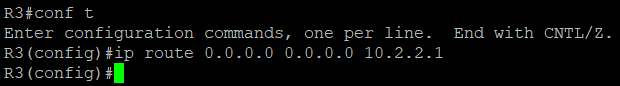


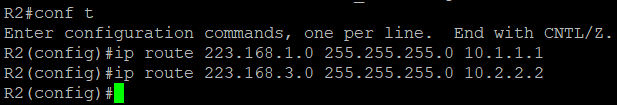
#### Step 2: Configure static routes.

R1(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.2



R3(config)# ip route 0.0.0.0 0.0.0.0 10.2.2.1





R2(config)# ip route 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

foreach address { 223.168.1.1

10.1.1.1

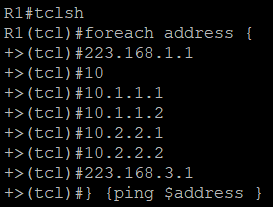
10.1.1.2

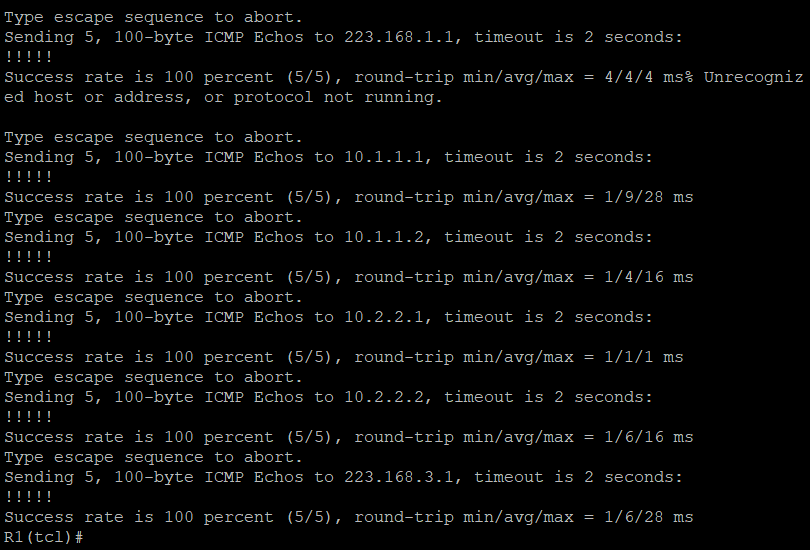
10.2.2.1

10.2.2.2

223.168.3.1}

{ping $address}





#### Step 3: Secure management access.

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

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

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

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

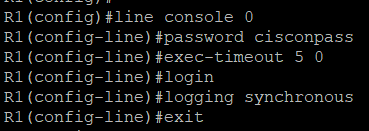
1. Configure a console password and enable login for routers. For additional security, the exectimeout command causes the line to log out after 5 minutes of inactivity. The logging synchronous command prevents console messages from interrupting command entry.

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) #

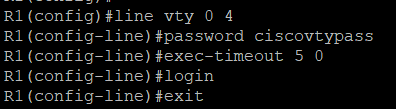
exit



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



1. 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

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

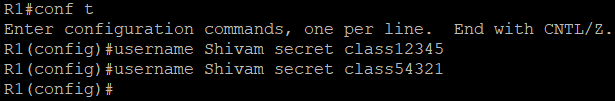
R1(Config)# service password-encryption

1. 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.



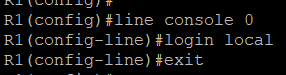
Step 4: Configure enhanced username password security.

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:
2. R1(config)# username Shivam secret class12345
3. R1(config)# username Shivam secret class5432



1. 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

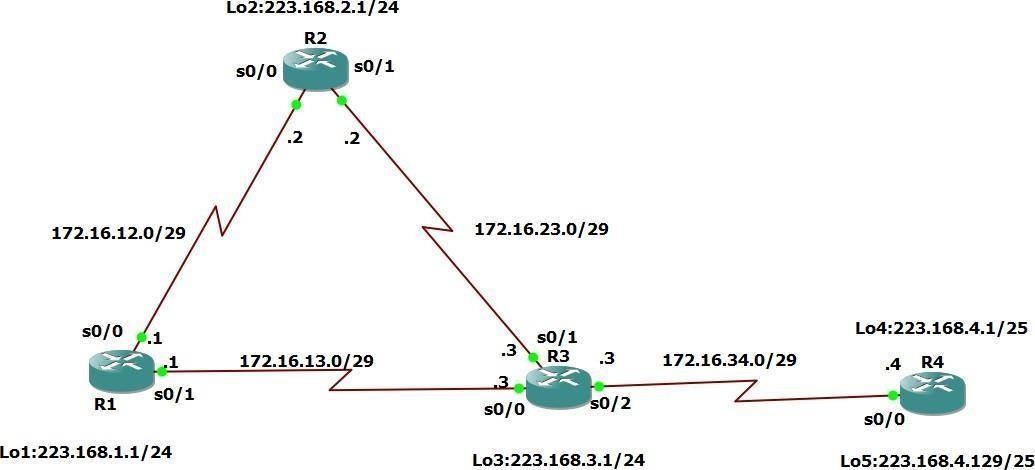


R1(config)# line vty 0 4 R1(config-line) # login local R1(config-line) # end



# Practical No-5

###### Aim: Configure and Verify Path Control Using PBR Topology:



**Objectives:**

− Configure and verify policy-based routing.

− Select the required tools and commands to configure policy-based routing operations.

− Verify the configuration and operation by using the proper show and debug commands.

###### Step 1: Configure loopbacks and assign addresses.

1. Cable the network as shown in the topology diagram. Erase the startup configuration, and reload each router to clear previous configurations.
2. 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.

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

###### Router R1

Interface Lo1

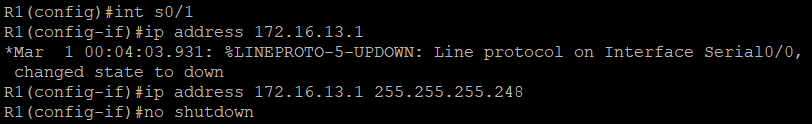
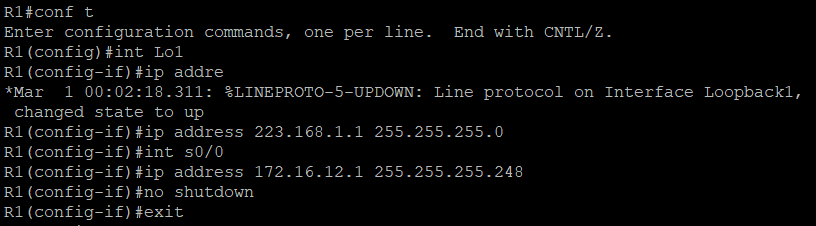
ip address 223.168.1.1 255.255.255.0

interface Serial 0/0/0

ip address 172.16.12.1 255.255.255.248

no shutdown interface Serial 0/0/1

ip address 172.16.13.1 255.255.255.248

no shutdown end

###### Router R2

Interface Lo2

ip address 223.168.2.1 255.255.255.0

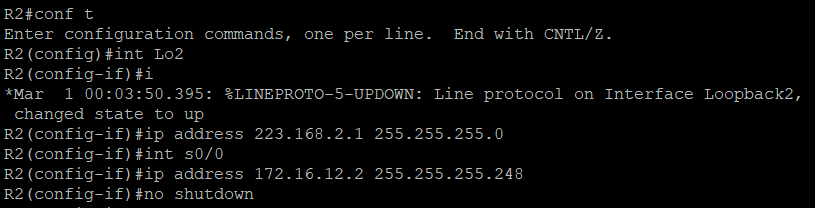
interface Serial 0/0/0

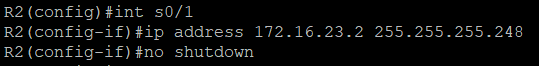
ip address 172.16.12.2 255.255.255.248

no shutdown interface Serial 0/0/1

ip address 172.16.23.2 255.255.255.248

no shutdown end





###### Router R3

Interface Lo3

ip address 223.168.3.1 255.255.255.0

interface Serial 0/0/0

ip address 172.16.13.3 255.255.255.248

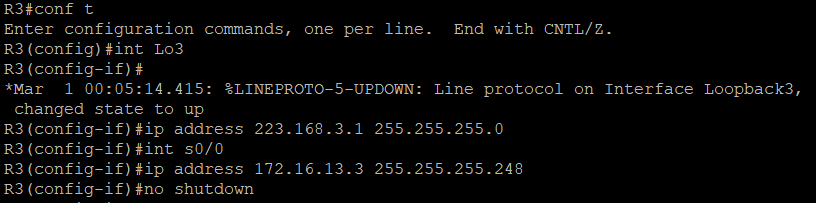
no shutdown interface Serial 0/0/1

ip address 172.16.23.3 255.255.255.248

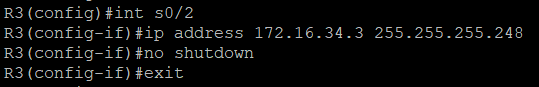
interface Serial 0/1/0

ip address 172.16.34.3 255.255.255.248

no shutdown end







###### Router R4

Interface Lo4

ip address 223.168.4.1 255.255.255.128

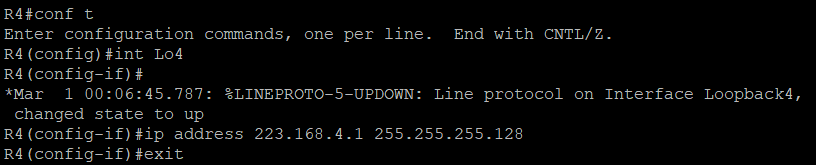
interface Lo5

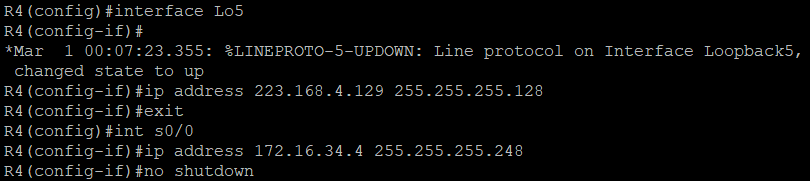
ip address 223.168.4.129 255.255.255.128

interface Serial 0/0/0

ip address 172.16.34.4 255.255.255.248

no shutdown end



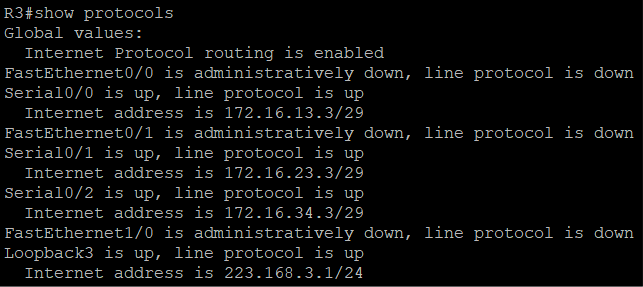


1. Verify the configuration with the show ip interface brief, show protocols, and show interfaces description commands. The output from router R3 is shown here as an example.

###### R3# show ip interface brief



**R3# show protocols**



###### R3# show interface description

**Step 3: Configure basic EIGRP.**

* 1. 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

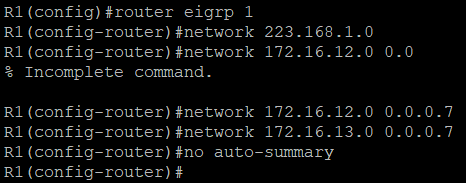
Router eigrp 1

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



###### Router R2

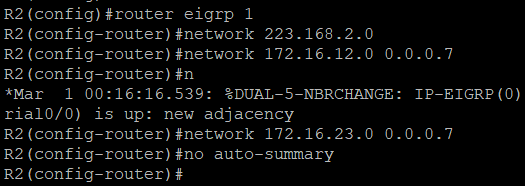
Router eigrp 1

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



###### Router R3

Router eigrp 1

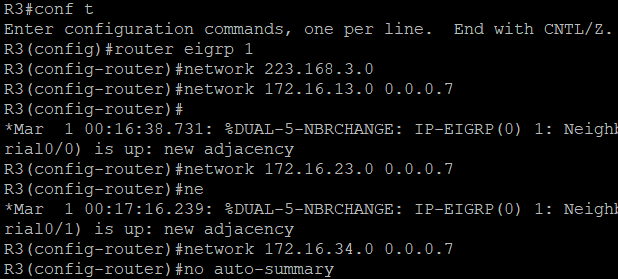
network 223.168.3.0

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



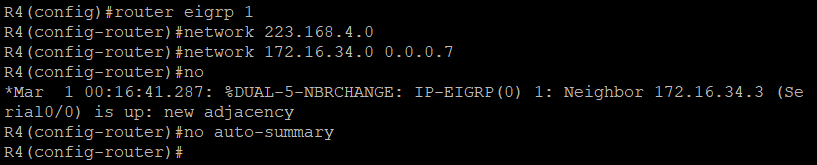
###### Router R4

router eigrp 1

network 223.168.4.0

network 172.16.34.0 0.0.07

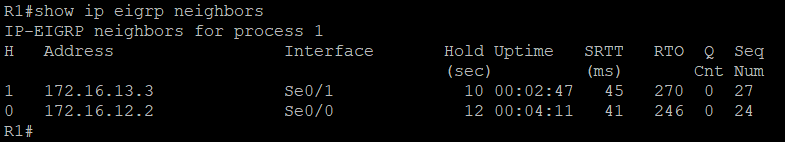
no auto-summary



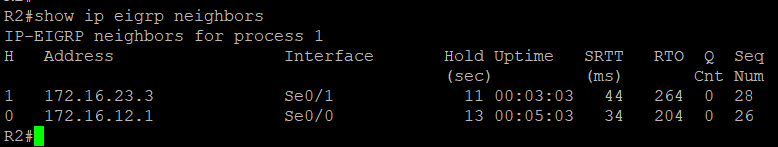
###### Step 4: Verify EIGRP connectivity.

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

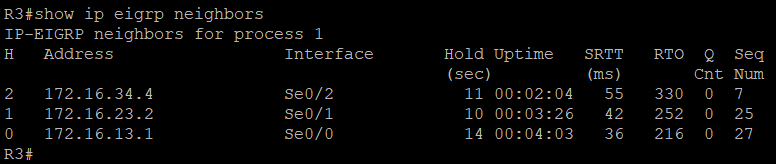
**R1# show ip eigrp neighbors**



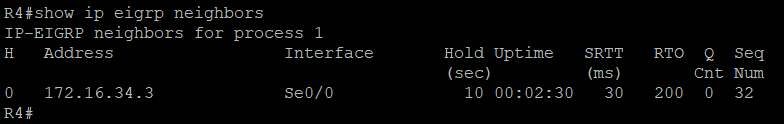
**R2# show ip eigrp neighbors**



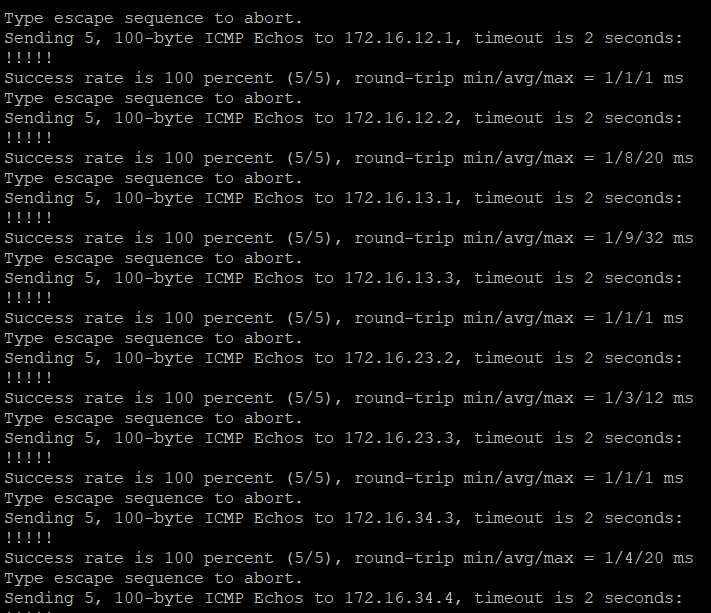
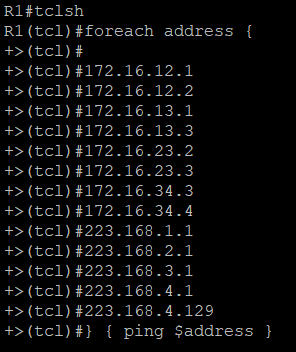
#### R3# show ip eigrp neighbors



**R4# show ip eigrp neighbors**



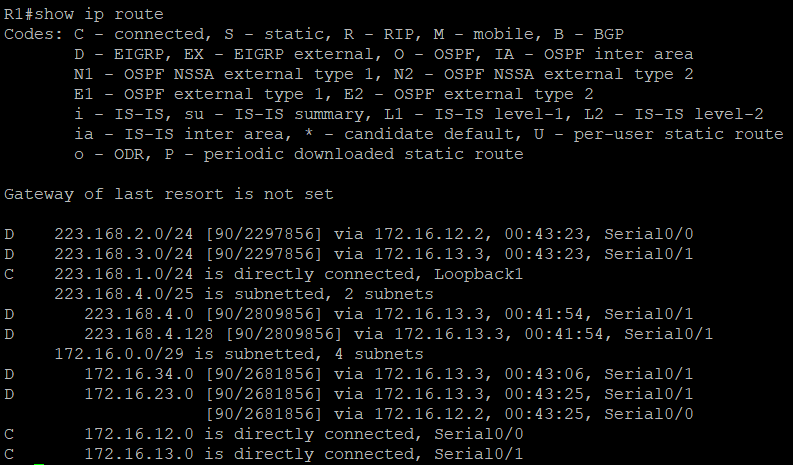
###### Run the following TCL Scripts on all routers to verify full connectivity.



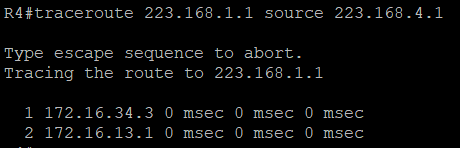
**Step 5: Verify the current path**.

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

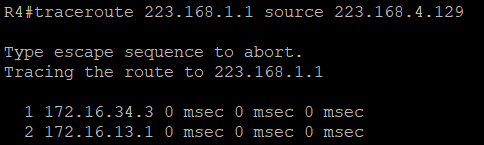
1. On R1, use the show ip route command. Notice the next-hop IP address for all networks discovered by EIGRP.



###### R4# traceroute 223.168.1.1 source 223.168.4.1

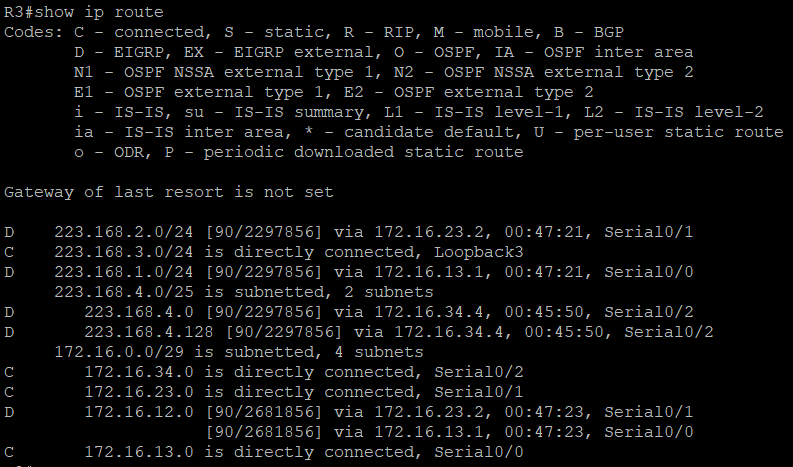


**R4# traceroute 223.168.1.1 source 223.168.4.129**

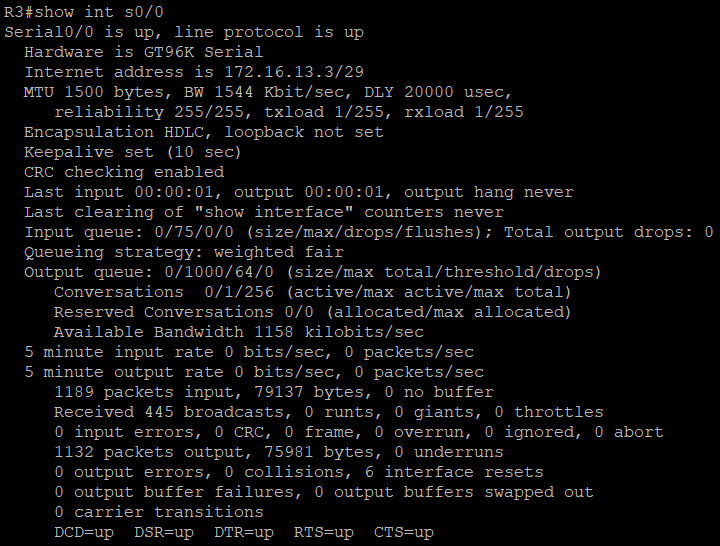


On R3, use the show ip route command and note that the preferred route from R3 to R1 LAN 223.168.1.0/24 is via R2 using the R3 exit interface S0/0/1

###### R3# show ip route



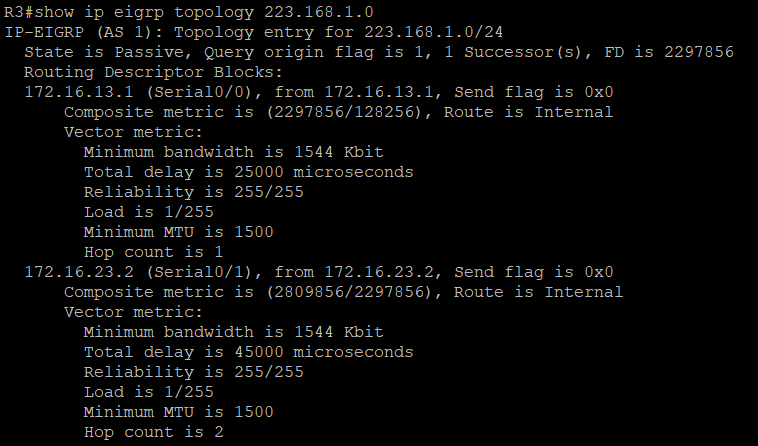
1. On R3, use the **show interfaces serial 0/0/0** and show **interfaces s0/0/1** command.



1. 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



###### 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.

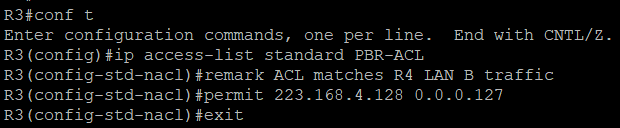
As a test, you will configure the following policy on router R3:

− 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.

1. 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**

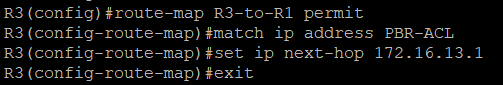
R3(config-std-nacl) #**remark ACL matches R4 LAN B traffic** R3(config-std-nacl) #**permit 223.168.4.128 0.0.0.127** R3(config-std-nacl) #**exit**

1. 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**

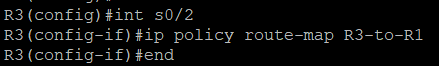


1. Apply the R3-to-R1 route map to the serial interface on R3 that receives the traffic from R4. Use the ip policy route-map command on interface S0/1/0.

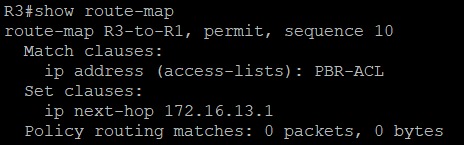
R3(config)# **interface s0/1/0**

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

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



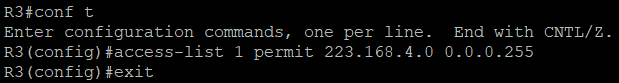
1. On R3, display the policy and matches using the show route-map command.



#### 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**

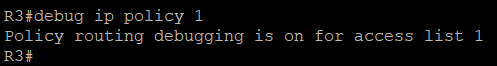


1. Enable PBR debugging only for traffic that matches the R4 LANs.

###### R3# debug ip policy?

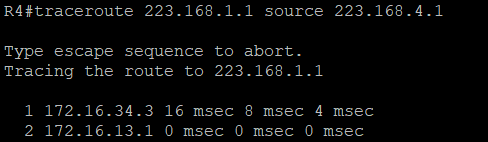


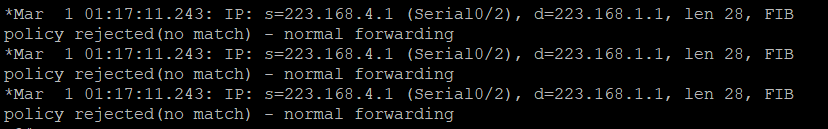
**R3# debug ip policy 1**



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

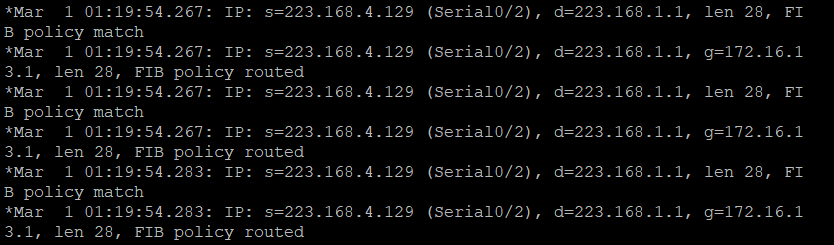
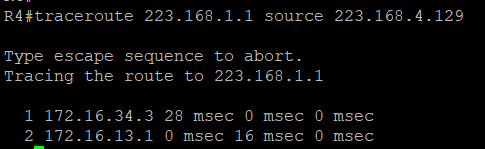
###### R4# traceroute 223.168.1.1 source 223.168.4.1



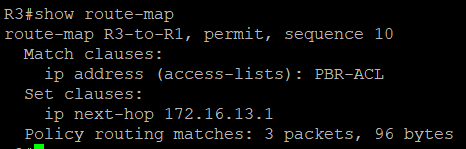


1. Test the policy from R4 with the traceroute command, using R4 LAN B as the source network.

###### R4# traceroute 223.168.1.1 source 223.168.4.129

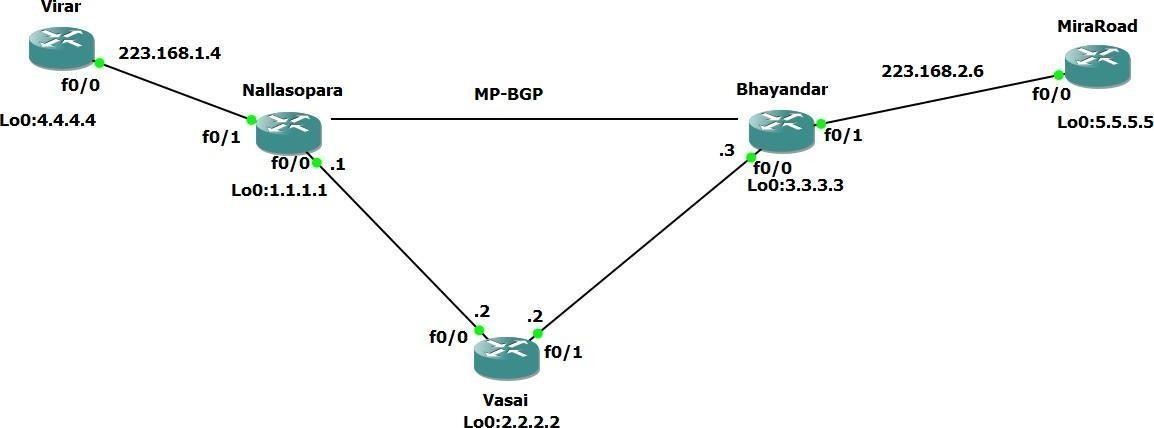


1. On R3, display the policy and matches using the show route-map command.



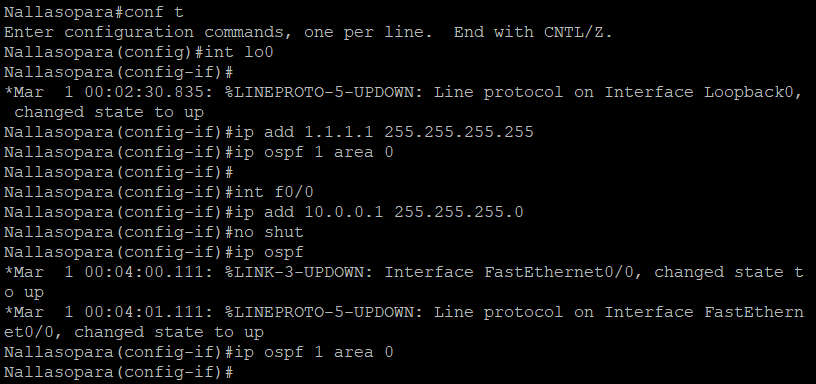
Practical No 6

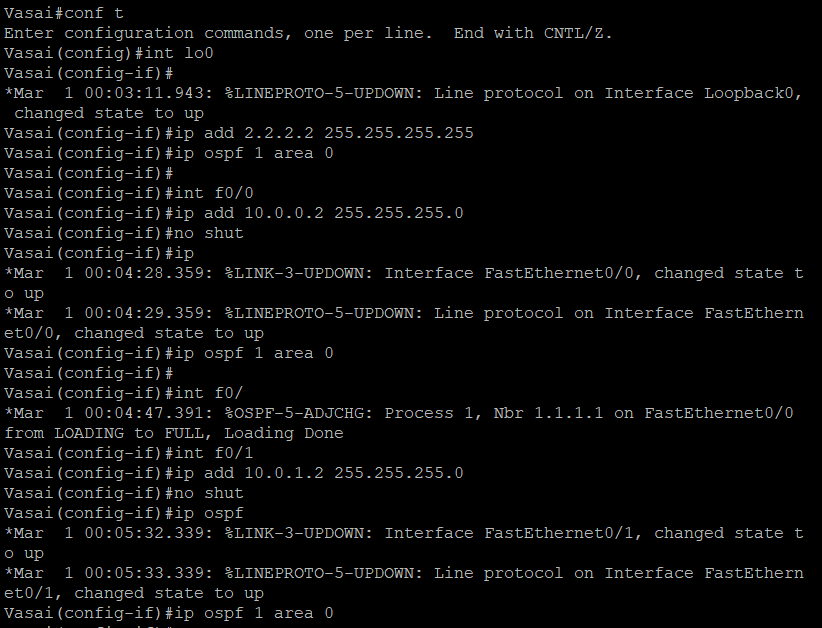
Aim: Cisco MPLS Configuration Topology:

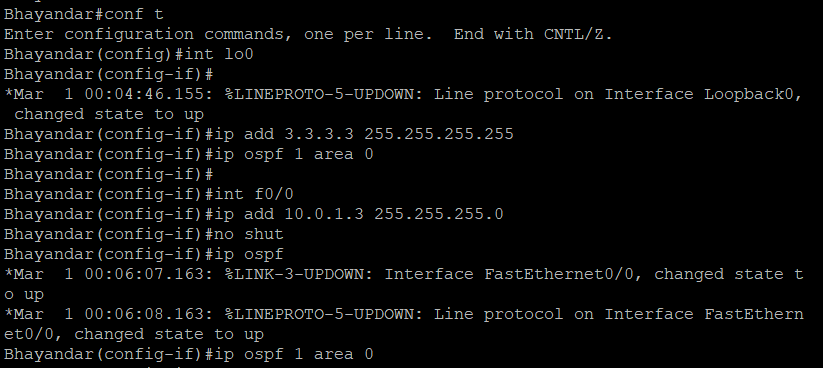


Step 1 – IP addressing of MPLS Core and OSPF

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

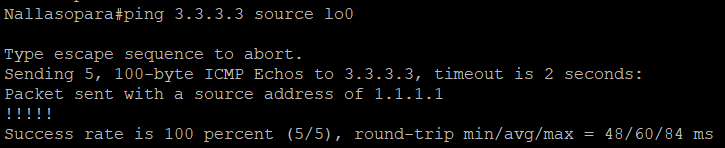






You should now have full ip connectivity between R1, R2, R3 to verify this we

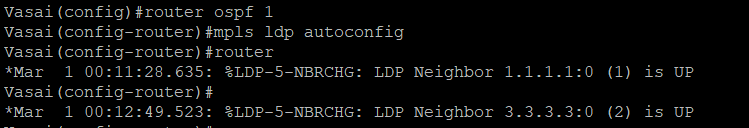
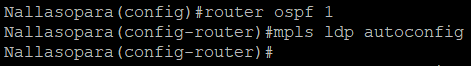
need to see if we can ping between the loopbacks of R1 and R3



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.

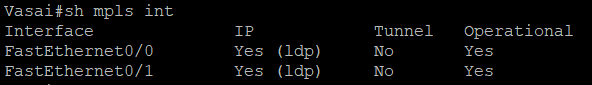
At each interface enter the mpls ip command

Under the ospf process use the mpls ldp autoconfig command.

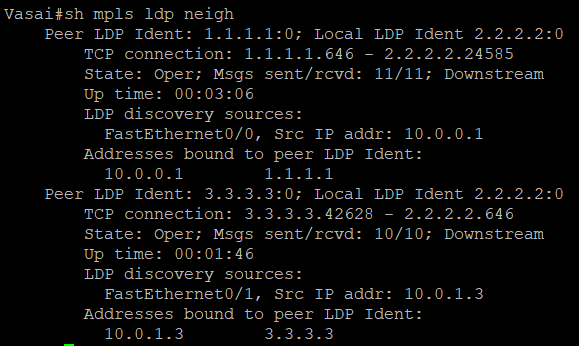


You should see log messages coming up showing the LDP neighbors are up. To verify the mpls interfaces the command is very simple – sh mpls interface

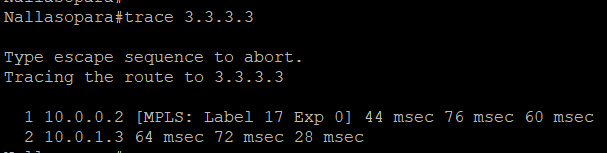
This is done on R2 and you can see that both interfaces are running mpls and using LDP.



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

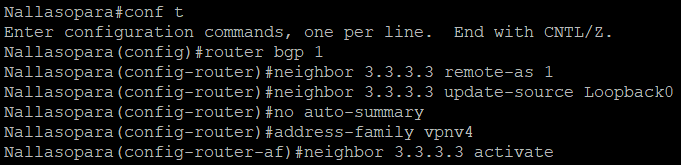


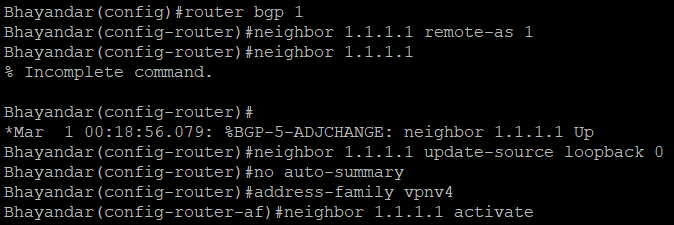
One more verification to confirm LDP is running ok is to do a trace between R1 and R3 and verify if you get MPLS Labels show up in the trace.



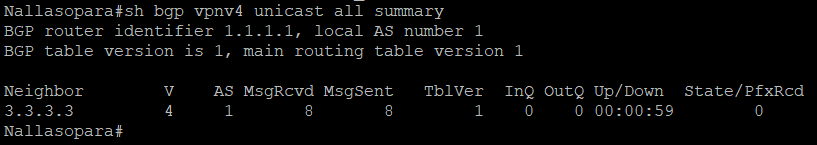
**Step 3 – MPLS BGP Configuration between R1 and R3**

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



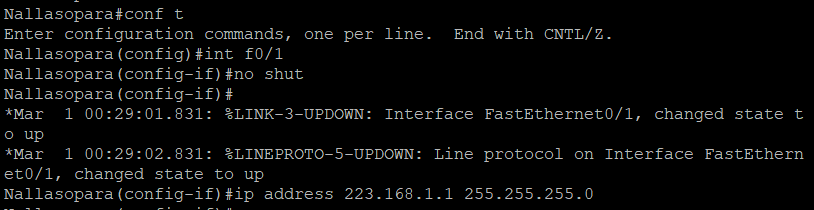
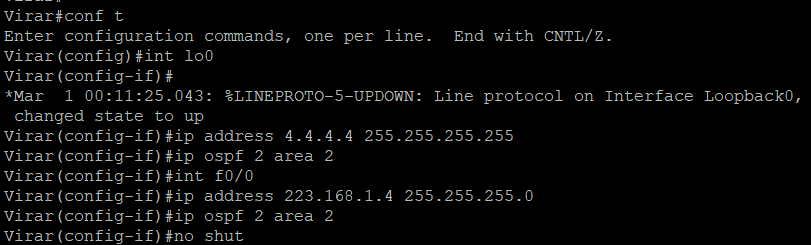


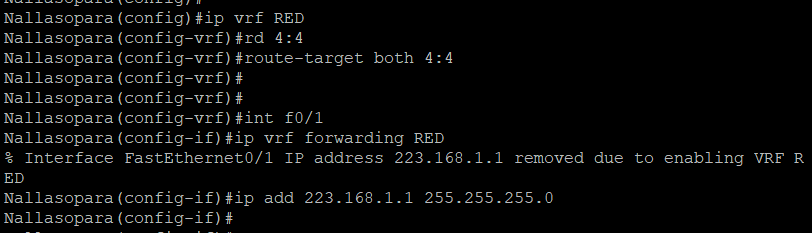
To verify the BGP session between R1 and R3 issue the command sh bgp vpnv4 unicast all summary.

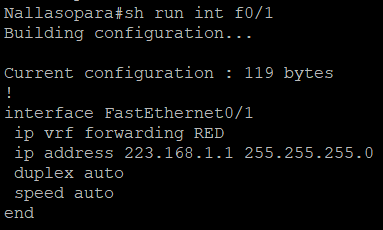


#### Step 4 – Add two more routers, create VRFs

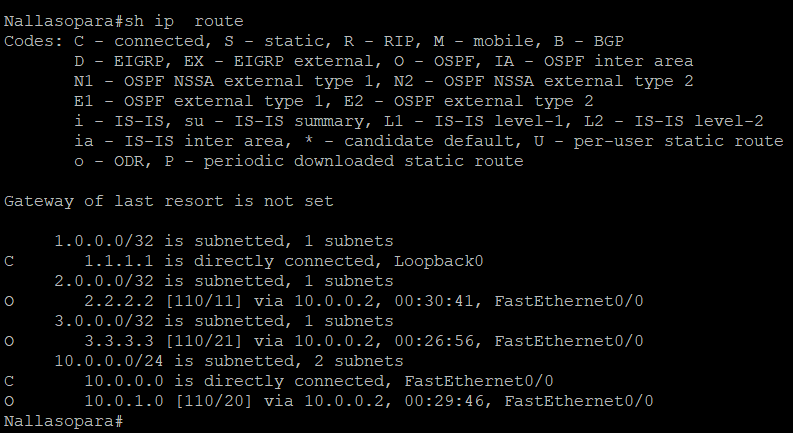
We will add two more routers into the topology so it now looks like the final topology.

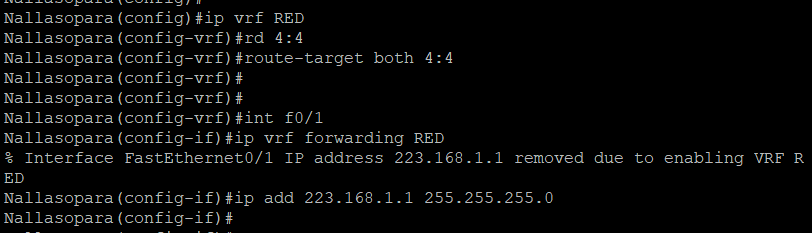




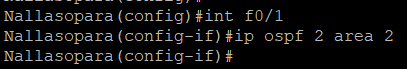


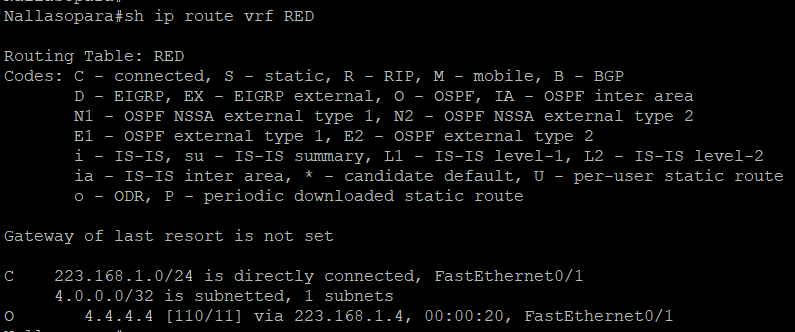
If you issue the command sh ip route this shows the routes in the global table and you will notice that you do not see 192.168.1.0/24.

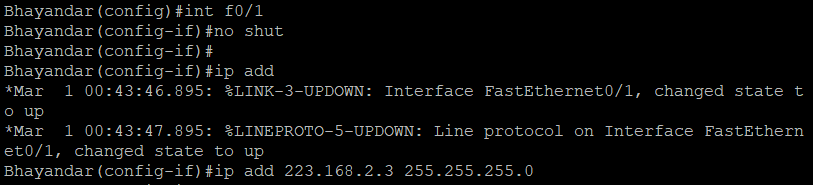
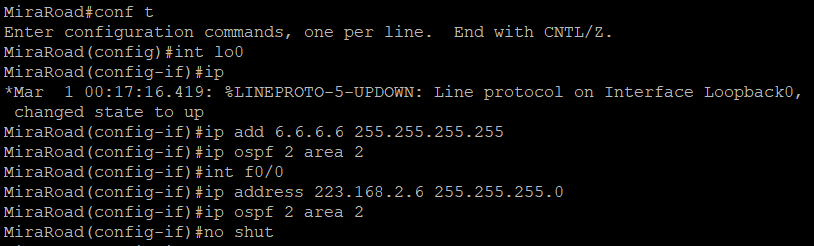
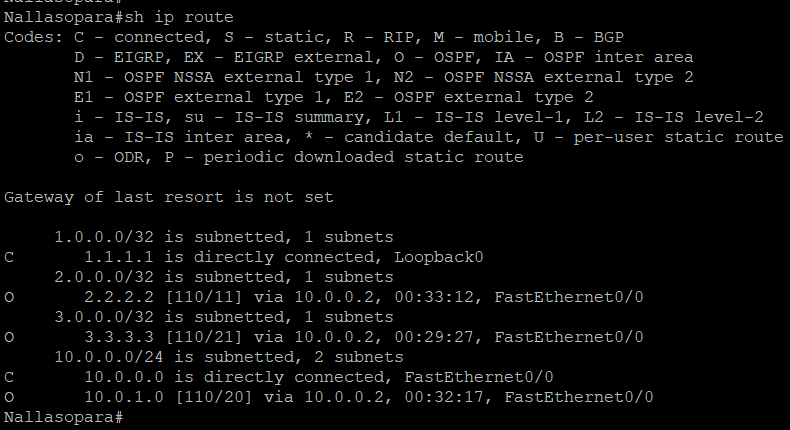




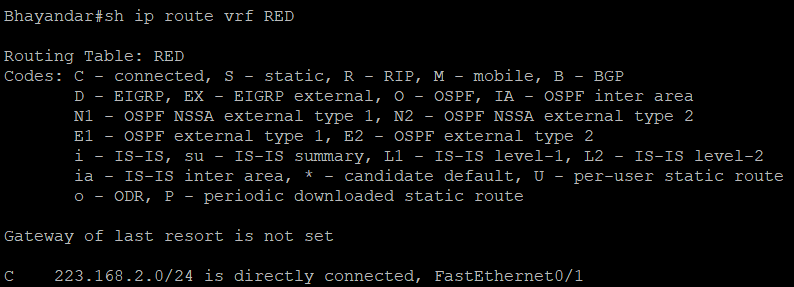
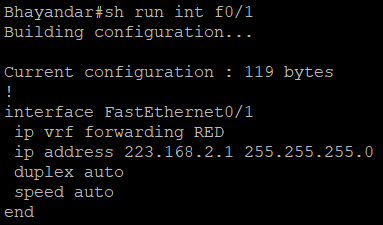
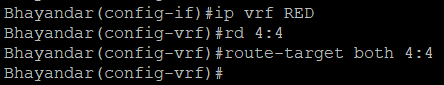
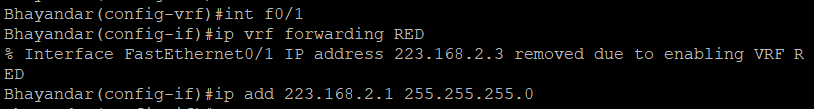
We just need to enable OSPF on this interface and get the loopback address for R4 in the VRF RED routing table before proceeding.

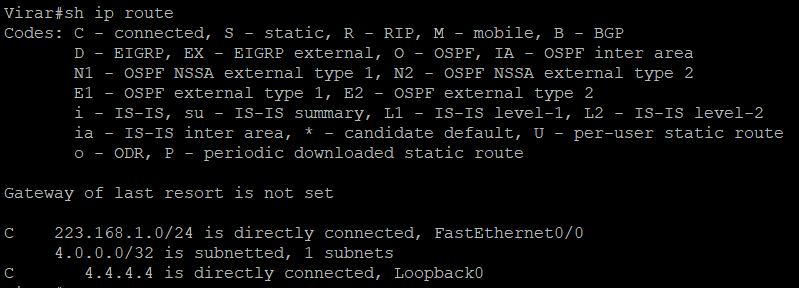




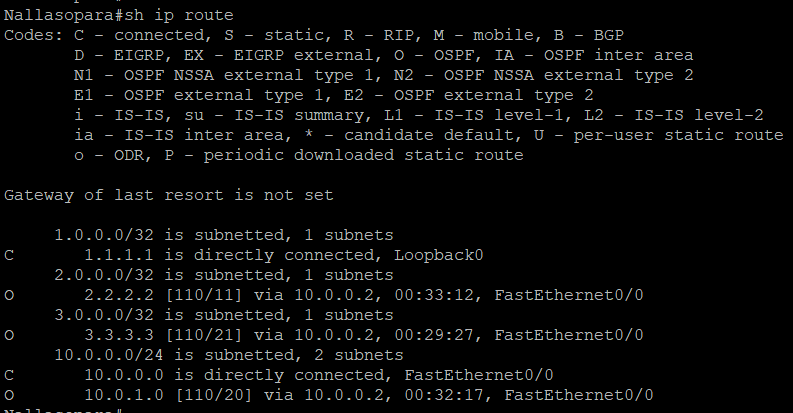
We now need to repeat this process for R3 & R6 Router 6 will peer OSPF using process number 2 to a VRF configured on R3. It will use the local site addressing to 223.168.2.0/24.

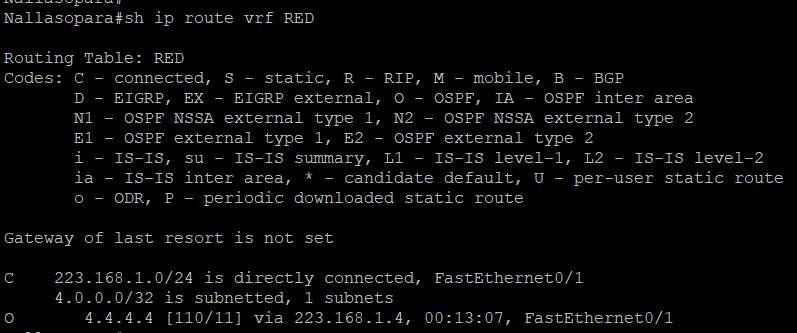
We also need to configure a VRF onto R3 as well.

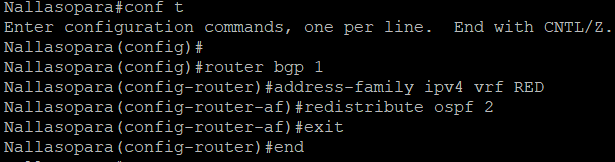
Check the Router in VRF RED

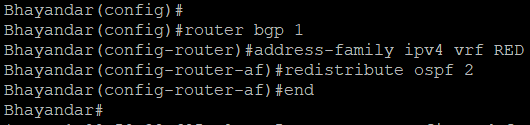


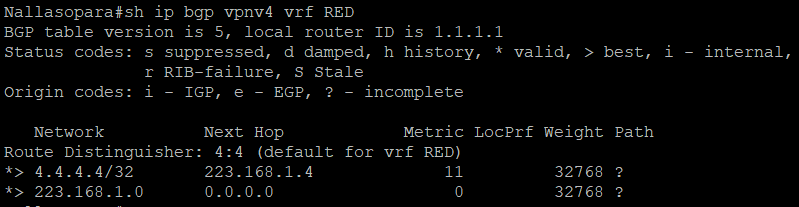
As expected we have the local interface and the loopback address. When we are done we want to see 6.6.6.6 in there so we can ping across the MPLS Check the routes on R1.

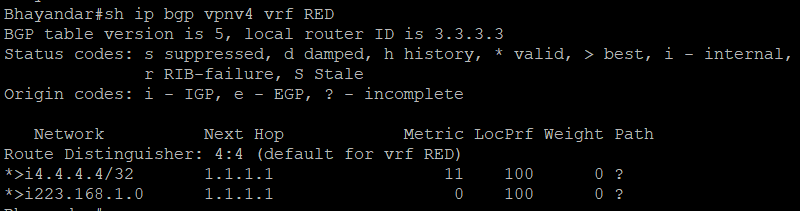




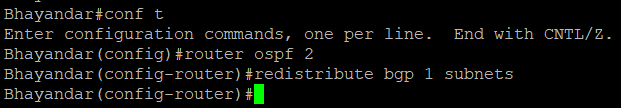
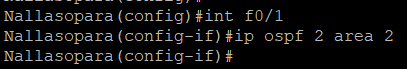




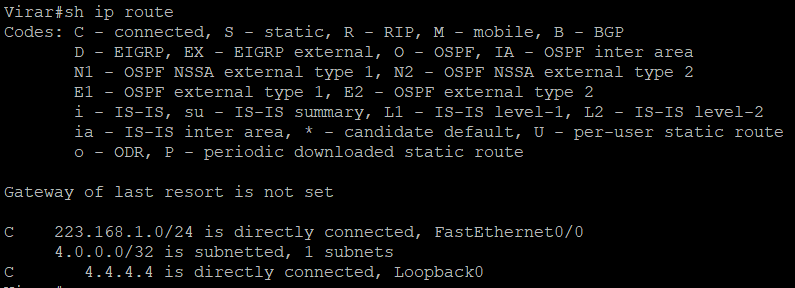


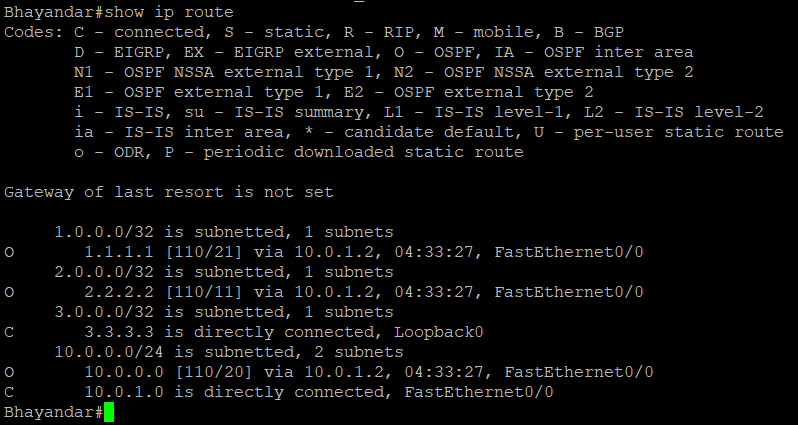


Which it is! 6.6.6.6 is now in the BGP table in VRF RED on R3 with a next hop of 192.168.2.6 (R6) and also 4.4.4 is in there as well with a next hop of 1.1.1.1 (which is the loopback of R1 – showing that it is going over the MPLS and R2 is not in the picture).

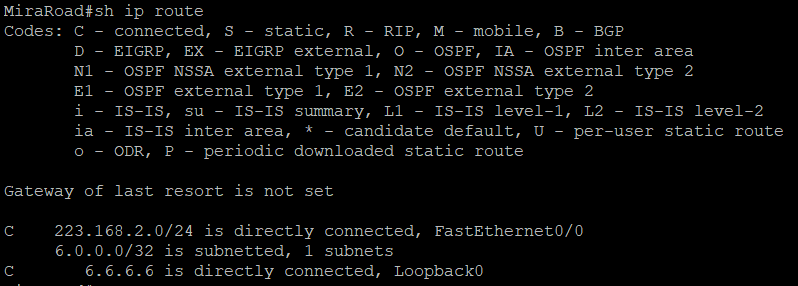
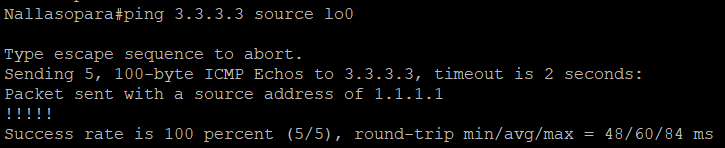


Before we do let’s see what the routing table look like on R.





Do the Same Step in R6.



# Practical No 7

#### Aim: Inter-VLAN Routing

###### 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)#

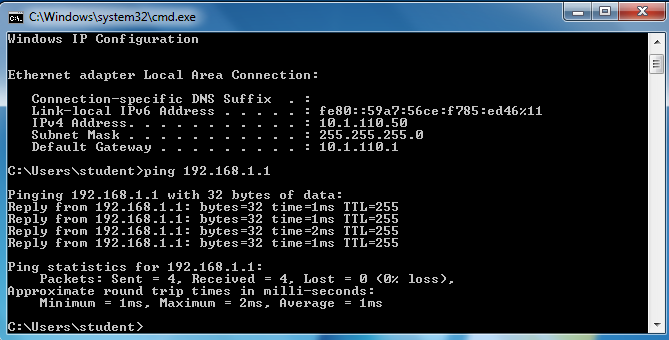
At this point, basic intervlan routing can be demonstrated using an attached host. Host D is attached to DLS2via interface Fa0/6. On DLS2, assign interface Fa0/6 to VLAN 110 and configure the host with the address 10.1.110.50/24 and default gateway of 10.1.110.1. Once you have done that, try and ping 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)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ALS1# | **show** | **interface** | **trunk** |  |  |
| Port |  | Mode | Encapsulation | Status | Native vlan |
| Po1 |  | on | 802.1q | trunking | 1 |
| Po4 |  | on | 802.1q | trunking | 1 |

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.

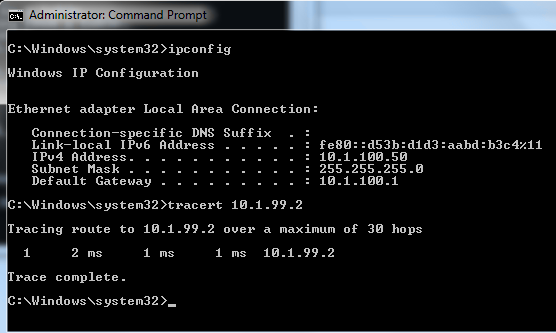
In the output below, the **switchport host** macro was used to quickly configure interface Fa0/6 with host-relative commands.

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**



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#

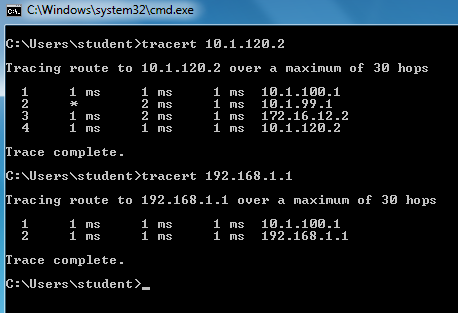
From ALS1, a traceroute to 192.168.1.1 should take one hop:

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.

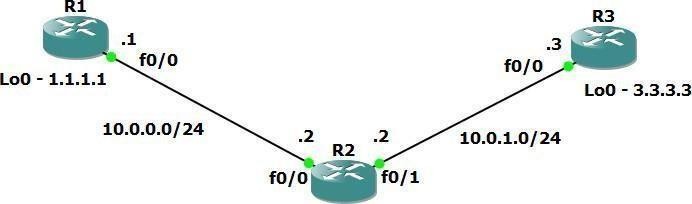
#### MSC-IT Part 1 Sem 2 Modern Networking

**Practical No 8**

**Aim: Cisco MPLS Configuration**

**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



R1

hostname R1intlo0

ipadd1.1.1.1255.255.255.255

ipospf1area0

intf0/0 ipadd10.0.0.1255.255.255.0

no shut ipospf1area0

R2

hostname R2intlo0

ip add 2.2.2.2 255.255.255.255

ip ospf 1 are 0

int f0/0

ip add 10.0.0.2 255.255.255.0

no shut

ip ospf 1 area 0 int f0/1

ip add 10.0.1.2 255.255.255.0

no shut

ip ospf 1 area 0 R3

hostname R3 int lo0

ip add 3.3.3.3 255.255.255.255

ip ospf 1 are 0 int f0/0

ip add 10.0.1.3 255.255.255.0

no shut

ip ospf 1 area 0

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

R1#ping3.3.3.3sourcelo0

Typeescapesequencetoabort.

Sending5,100-byteICMPEchosto3.3.3.3,timeoutis2seconds: Packetsentwithasourceaddressof1.1.1.1

!!!!!

Success rate is 100 percent (5/5), round- tripmin/avg/max=40/52/64ms

R1#

You could show the routing table here, but the fact that you can ping between the loopbacks is verification enough and it is safe to move on.

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

For this tutorial we will be using the second option, so go int the ospf process and enter mpls ldp autoconfig – this will enable mpls label distribution protocol on every interface running ospf under that specific process.

R1

routerospf 1 mplsldpautoconfig

R2

routerospf 1

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

mplsldpautoconfig

R3

routerospf 1 mplsldpautoconfig

R2#

\*Mar1 00:31:53.643: %SYS-5-CONFIG\_I: Configured fromconsole

\*Mar100:31:54.423:%LDP-5-NBRCHG:LDPNeighbor 1.1.1.1:0 (1) is UPR2#

\*Mar100:36:09.951:%LDP-5-NBRCHG:LDPNeighbor 3.3.3.3:0(2)isUP

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

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| R2#sh mpls interface | | | | |
| Interface Operational | IP |  | Tunnel | |
| 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.

R2#shmplsldpneigh

PeerLDPIdent:1.1.1.1:0;LocalLDPIdent2.2.2.2:0TCPconnection:1.1.1.1.646

-2.2.2.2.37909

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.

State:Oper;Msgssent/rcvd:16/17;DownstreamUptim e:00:07:46

LDPdiscoverysources:

FastEthernet0/0,SrcIPaddr:10.0.0.1Addr

essesboundtopeerLDPIdent:

10.0.0.1

1.1.1.1

PeerLDPIdent:3.3.3.3:0;LocalLDPIdent2.2.2.2:0TCPcon nection:3.3.3.3.22155-2.2.2.2.646

State:Oper;Msgssent/rcvd:12/11;DownstreamUptim e:00:03:30

LDPdiscoverysources:

FastEthernet0/1,SrcIPaddr:10.0.1.3Addr essesboundtopeerLDPIdent:

10.0.1.3

3.3.3.3

R1#trace3.3.3.3

Type escape sequence to abort.Tracingtherouteto3.3.3.3

110.0.0.2[MPLS:Label17Exp0]84msec72msec44msec 210.0.1.368msec60msec\*

As you can see the trace to R2 used an MPLS Label in the path, as this is a very small MPLS core only one label was used as R3 was the final hop.

**MSC-IT Part 1 Sem 2 Modern Networking**

##### So to review we have now configured IP addresses on the MPLS core, enabled OSPF and full IP connectivity between all routers and finally enabled mpls on all the interfaces in the core and have established ldp neighbors between all routers.

The next step is to configure MP-BGP between R1 and R3

##### This is when you start to see the layer 3 vpn configuration come to life

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

R1#

routerbgp1 neighbor3.3.3.3remote-as1

neighbor 3.3.3.3 update-source Loopback0noauto-summary

!

address-family vpnv4neighbor3.3.3.3activate

R3#

routerbgp1 neighbor1.1.1.1remote-as1

neighbor 1.1.1.1 update-source Loopback0noauto-summary

!

address-family vpnv4neighbor1.1.1.1activate

2022-2023 Page:103

You should see log messages showing the BGP sessions coming up.

\*Mar100:45:01.047:%BGP-5-ADJCHANGE:neighbor1.1.1.1

Up

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



R1Y#osuhbcganpvsepne vh4eurenithcaatswteadlolshuamvemaarbygp vpnv4 peering to R3 – looking at

BGtPheroPfuxtRecrdiydoeuncatnifseieeitrs1a.y1s .01th.i1s ,islboeccaaulsAeSwneuhmabveern1ot got any routes

in BGP. We are now going to add two more routers to the topology. These

BGwPitllabbeltehveecrussitoomneirs1sit,emsacionnrnoeuctteidntgotRab1 laenvdeRr3s.iWone1will then create a VRF on each router and put the interfaces connected to each site router intothat VRF.

Neighbo rOutQ

V

ASMsgRcvdMsgSent

TblVerInQ

Up/Down State/PfxRcd

3.f3in.a3l .to3pology 0 03:17:48

We resolve add two more routers into the topology so it now looks like the

4 1 218 218 1 0

0

Router 4 will peer OSPF using process number 2 to a VRF configured on R1. It will use the local site addressing of 223.168.1.0/24.

R4

int lo0 ipadd4.4.4.4255.255.255.255

ip ospf 2 area

2intf0/0

Now at this point we have R4 peering to R1 but in the global routing table of R1 which is not what we want.

ipadd192.168.1.4255.255.255.0

ip ospf 2 area 2noshut

R1

int f0/1no shut

ipadd223.168.1.1255.255.255.0

We are now going to start using VRF’s

What is a VRF in networking?

Virtual routing and forwarding (**VRF**) is a technology included in IP (Internet Protocol) that allows multiple instances of a routing table to co-exist in a router and work together but not interfere with each other.. This increases functionality by allowing network paths to be segmented without using multiple devices.

As an example if R1 was a PE Provider Edge router of an ISP and it had two customers that were both addressed locally with the 223.168.1.0/24 address space it could accommodate both their routing tables in different VRFs – it distinguishes between the two of them using a Route Distinguisher

So back to the topology – we now need to create a VRF on R1 For this mpls tutorial I will

be using VRF RED

R1

ip vrf REDrd4:4

route-targetboth4:4

The RD and route-target do not need to be the same – and for a full explanation please read this post on Route Distinguishers

[Route Distinguisher vs Route Target](https://www.rogerperkin.co.uk/ccie/mpls/route-distinguisher-vs-route-target/) before proceeding.

So now we have configured the VRF on R1 we need to move the interface F0/1 into that VRF

R1

intf0/1 ipvrfforwardingRED

Now notice what happens when you do that – the IP address is removed

R1(config-if)#ipvrffo

R1(config-if)#ipvrfforwardingRED

% Interface FastEthernet0/1 IP address 223.168.1.1removedduetoenablingVRFRED

You just need to re-apply it

R1

intf0/1 ipaddress223.168.1.1255.255.255.0

Now if we view the config on R1 int f0/1 you can see the VRF configured.

R1

R1#sh run int f0/1Buildingconfiguration... Currentconfiguration:119bytes

Now we can start to look int VRF’s and how they operate – you need to understand now that there are 2 routing tables within R1

interface FastEthernet0/1ipvrfforwardingRE D ipaddress223.168.1.1255.255.255.0

duplexautospeed auto

end

* The Global Routing Table
* The Routing Table for VRF RED

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 223.168.1.0/24

R1#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

If you now issue the command sh ip route vrf red – this will show the routes in the routing table for VRF RED

1.0.0.0/32issubnetted,1subnets

C 1.1.1.1 is directly connected, Loopback02.0.0.0/32issubnetted,1subnets O 2.2.2.2[110/11]via10.0.0.2,01:03:48,

FastEthernet0/0 3.0.0.0/32issubnetted,1subnets

O 3.3.3.3[110/21]via10.0.0.2,01:02:29,

FastEthernet0/0 10.0.0.0/24issubnetted,2subnets

C10.0.0.0isdirectlyconnected,FastEthernet0/0O10.0.1.0[110/20]via10.0.0.2,01:02:39, FastEthernet0/0R1#

R1#shiproutevrfred

%IProutingtablereddoesnotexist

**NOTE**: The VRF name is case sensitive!

R1#shiproutevrfRED

RoutingTable:RED

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

-BGP

D- EIGRP,EX- EIGRPexternal,O-OSPF,IA-OSPF

interarea

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA externaltype2

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

E1-OSPFexternaltype1,E2-OSPFexternaltype2

i- IS-IS,su - IS-ISsummary,L1-IS-ISlevel-1,L2-

IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-userstaticroute o-ODR,P -periodicdownloadedstaticroute

Gatewayoflastresortisnotset

C223.168.1.0/24isdirectlyconnected,FastEthernet0/1R1#

R1

intf0/1 ipospf2area2

You should see a log message showing the OSPF neighbor come up

R1(config-if)#

\*Mar101:12:54.323:%OSPF-5-ADJCHG:Process2,Nbr

4.4.4.4

onFastEthernet0/1fromLOADINGtoFULL,LoadingDone

If we now check the routes in the VRF RED routing table you should see

* + - 1. in there as well.

R1#shiproutevrfRED

RoutingTable:RED

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

We now need to repeat this process for R3 & R6

D- EIGRP,EX- EIGRPexternal,O-OSPF,IA-OSPF

interarea

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA externaltype2 E1-OSPFexternaltype1,E2-OSPFexternaltype2

i- IS-IS,su - IS-ISsummary,L1-IS-ISlevel-1,L2-

IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-userstaticroute o-ODR,P -periodicdownloadedstaticroute

Gatewayoflastresortisnotset

4.0.0.0/32issubnetted,1subnets

**O4.4.4.4[110/11]via223.168.1.4,00:00:22,**

**FastEthernet0/1**

C223.168.1.0/24isdirectlyconnected,FastEthernet0/1R1#

Router 6 will peer OSPF using process number 2 to a VRF configured on R3. It will use the local site addressing of 223.168.2.0/24.

R6

int lo0 ipadd6.6.6.6255.255.255.255

ip ospf 2 area 2intf0/0

ipadd223.168.2.6255.255.255.0

ipospf2area2

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

no shut

R3

int f0/1no shut

ipadd223.168.2.3255.255.255.0

R3

ip vrf REDrd4:4

route-targetboth4:4

So now we have configured the VRF on R3 we need to move the interface F0/1 into that VRF

R3

intf0/1 ipvrfforwardingRED

Now notice what happens when you do that – the IP address is removed

R3(config-if)#ipvrfforwardingRED

% Interface FastEthernet0/1 IP address 223.168.2.1removedduetoenablingVRFRED

### You just need to re-apply it

R3

intf0/1

ip address 223.168.2.1255.255.255.0

Now if we view the config on R3 int f0/1 you can see the VRF configured.

Finally, we just need to enable OSPF on that interface and verify the routes are in the RED routing table.

R3#sh run int f0/1Buildingconfiguration...

Currentconfiguration:119bytes

!

interface FastEthernet0/1ipvrfforwardingRED ip address 223.168.2.1255.255.255.0

duplex autospeed autoend

R3

intf0/1 ipospf2area2

Check the routes in vrf RED

R3

R3#shiproutevrfRED RoutingTable:RED

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

-BGP

Gatewayoflastresortisnotset

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

O

6.0.0.0/32issubnetted,1subnets

6.6.6.6[110/11]via 223.168.2.6,00:02:44,

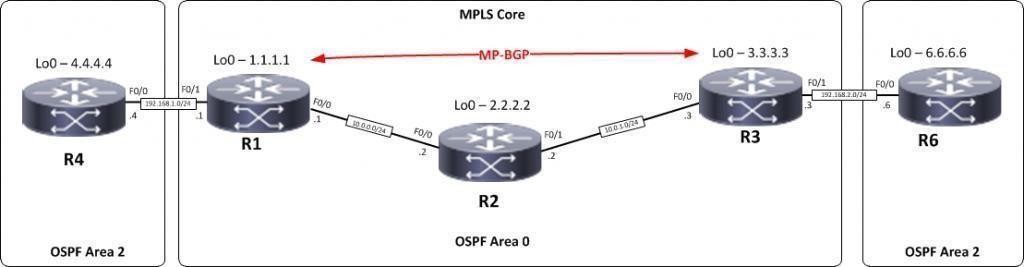
FastEthernet0/1

C

223.168.2.0/24isdirectlyconnected,

FastEthernet0/1

R3#



R1,R2,R3 form the MPLS Core and are running OSPF with all loopbacks running a /32 address and all have full connectivity. R1 and R3 are peering with MP-BGP. LDP is enabled on all the internal interfaces. The external interfaces of the MPLS core have been placed into a VRF called RED and then a site router has been joined to that VRF on each side of the MPLS core – (These represent a small office)

The final step to get full connectivity across the MPLS core is to redistribute the routes in OSPF on R1 and R3 into MP-BGP and MP-BGP into OSPF, this is what we are going to do now.

We need to redistribute the OSPF routes from R4 into BGP in the VRF on R1, the OSPF routes from R6 into MP-BGP in the VRF on R3 and then the routes in MP-BGP in R1 and R3 back out to OSPF

Before we start lets do some verifications

**Check the routes on R4**

R4#shiproute 4.0.0.0/32issubnetted,1subnets

C 4.4.4.4isdirectlyconnected,Loopback0

As expected we have the local interface and the loopback address.

C223.168.1.0/24isdirectlyconnected,FastEthernet0/0

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**

R1#shiproute

1.0.0.0/32issubnetted,1subnets C1.1.1.1isdirectlyconnected,Loopback02.0.0.0/32issubnette

d,1subnets

O 2.2.2.2[110/11]via10.0.0.2,00:01:04,

FastEthernet0/0 3.0.0.0/32issubnetted,1subnets

O 3.3.3.3[110/21]via10.0.0.2,00:00:54,

FastEthernet0/0 10.0.0.0/24issubnetted,2subnets

C10.0.0.0isdirectlyconnected,FastEthernet0/0O10.0.1.0[110/20]via10

.0.0.2,00:00:54,

FastEthernet0/0

Remember we have a VRF configured on this router so this command will

show routes in the global routing table (the MPLS Core) and it will not show the 223.168.1.0/24 route as that is in VRF RED – to see that we run the following command.

R1#shiproutevrfRED RoutingTable:RED 4.0.0.0/32issubnetted,1subnets

Here you can see Routing Table: RED is shown and the routes to R4 are now visible with 4.4.4.4 being in OSPF. So we need

O 4.4.4.4[110/11]via192.168.1.4,00:02:32,

FastEthernet0/1 C223.168.1.0/24isdirectlyconnected,FastEthernet0/1

to do the following;

* + - * + Redistribute OSPF into MP-BGP on R1
        + Redistribute MP-BGP into OSPF on R1
        + Redistribute OSPF into MP-BGP on R3
        + Redistribute MP-BGP into OSPF on R3

Redistribute OSPF into MP-BGP on R1

R1

routerbgp1

address-family ipv4 vrf REDredistributeospf2

Redistribute OSPF into MP-BGP on R3

R3

routerbgp1

address-family ipv4 vrf REDredistributeospf2

This has enabled redistribution of the OSPF routes into BGP. We can check

the routes from R4 and R6 are now showing in the BGP table for their VRF with this command

## sh ip bgp vpnv4 vrf RED

R1#shipbgpvpnv4vrfRED BGPtableversionis9,localrouterIDis1.1.1.1 Statuscodes:ssuppressed,ddamped,hhistory,\*valid,>best,

|  |  |  |  |
| --- | --- | --- | --- |
| r RIB-failure, S | Stale |  |  |
| Origin codes: i - | IGP, e | - EGP, ? - incomplete | |
| Network Next Hop | Metric | LocPrf Weight Path | |
| Route Distinguisher: 4:4 (default for vrf RED)  **\*> 4.4.4.4/32 223.168.1.4 11 32768 ?** | | | |
| **\*>i6.6.6.6/32 3.3.3.3 11** |  | **100 0** | **?** |
| \*> 223.168.1.0 0.0.0.0 0 |  | 32768 | ? |
| \*>i223.168.2.0 3.3.3.3 0 |  | 100 0 | ? |

Here we can see that 4.4.4.4 is now in the BGP table in VRF RED on

R1 with a next hop of 223.168.1.4 (R4) and also 6.6.6.6 is in there as well with a next hop of 3.3.3.3 (which is the loopback of R3 – showing that it is going over the MPLS and R1 is not in the picture)

The same should be true on R3

R3#sh ip bgp vpnv4 vrf RED

BGP table version is 9, local router ID is 3.3.3.3

Status codes: s suppressed, d damped, h history, \* valid, > best, 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 ?**

**\*> 6.6.6.6/32 223.168.2.6 11 32768 ?**

\*>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 223.168.2.6 (R6) and also 4.4.4 is in there as well with a next hop of

\*>223.168.2.00.0.0.0032768?

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)

The final step is to get the routes that have come across the MPLS back into OSPF and then we can get end to end connectivity

If all has worked we should be now able to ping 6.6.6.6 from R4

R1

routerospf2redistributebgp1subnets R3

routerospf2redistributebgp1subnets

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

R4#shiproute

4.0.0.0/32issubnetted,1subnets C

4.4.4.4isdirectlyconnected,Loopba ck06.0.0.0/32issubnetted,1subnets

**O IA6.6.6.6[110/21] via 223.168.1.1,00:01:31,**

**FastEthernet0/0**

Great we have 6.6.6.6 in there

O E2223.168.2.0/24[110/1]via223.168.1.1,00:01:31,

FastEthernet0/0

## Also check the routing table on R6

R6#shiproute

4.0.0.0/32issubnetted,1subnets

**O IA4.4.4.4[110/21]via225.168.2.1,00:01:22,**

**FastEthernet0/0**

6.0.0.0/32is subnetted, 1 subnets C

6.6.6.6isdirectlyconnected,Loopback0OIA223.168.1.0/24[110

/11]via 223.168.2.1,00:01:22,FastEthernet0/0

C223.168.2.0/24isdirectlyconnected,FastEthernet0/0

Brilliant we have 4.4.4.4 in there so we should be able to ping across the MPLS

R4#ping6.6.6.6

Typeescapesequencetoabort.

Sending5,100-byteICMPEchosto6.6.6.6,timeoutis2seconds:

!!!!!

Success rate is 100 percent (5/5), round- tripmin/avg/max=40/48/52ms

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

R4#trace6.6.6.6

Type escape sequence to abort.Tracingtherouteto6.6.6.6

223.168.1.120msec8msec8msec 210.0.0.2[MPLS:Labels17/20Exp0]36msec40msec

36 msec 223.168.2.1[MPLS:Label20Exp0]16msec40msec16msec 223.168.2.644msec40msec56msecR4#