**Practical No: 01**

**Title**: Configure IP SLA Tracking & Path Control Topology

A diagram of a network

Description automatically generated

**Objectives**:

Configure and verify the IP SLA feature.

Test the IP SLA tracking feature.

Verify the configuration and operation using show and debug commands.

Use IO router c7200

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

**Router R1**

hostname R1

interface Loopback 0

description R1 LAN

ip address 192.168.1.1 255.255.255.0

interface Serial3/0

description R1 → ISP1

ip address 209.165.201.2 255.255.255.252

clock rate 128000

bandwidth 128

no shutdown

interface Serial3/1

description R1 → ISP2

ip address 209.165.202.130 255.255.255.252

bandwidth 128

no shutdown

A screenshot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated



A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

**Router ISP1 (R2)**

hostname ISP1

interface Loopback0

description Simulated Internet Web Server

ip address 209.165.200.254 255.255.255.255

interface Loopback1

description ISP1 DNS Server

ip address 209.165.201.30 255.255.255.255

interface Serial3/0

description ISP1 → R1

ip address 209.165.201.1 255.255.255.252

bandwidth 128

no shutdown

interface Serial3/1

description ISP1 → ISP2

ip address 209.165.200.225 255.255.255.252

clock rate 128000

bandwidth 128

no shutdown

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

**Router ISP2 (R3)**

hostname ISP2

interface Loopback0

description Simulated Internet Web Server

ip address 209.165.200.254 255.255.255.255

interface Loopback1

description ISP2 DNS Server

ip address 209.165.202.158 255.255.255.255

interface Serial

description ISP2 → R1

ip address 209.165.202.129 255.255.255.252

clock rate 128000

bandwidth 128

no shutdown

interface Serial3/1

description ISP2 → ISP1

ip address 209.165.200.226 255.255.255.252

bandwidth 128

no shutdown

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

**Practical No: 02**

**Title**: Using the AS\_Path Attribute

**A diagram of a network

Description automatically generated**

A diagram of a network

Description automatically generated

**Objectives**: Use BGP commands to prevent private AS numbers from being advertised to the outside world. Use the AS\_PATH attribute to filter BGP routes based on their source AS number

**Step 1**: Prepare the routers for the lab.

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

**Step 2**: Configure the hostname and interface addresses.

**Router R1 (hostname SanJose)**

R1#CONF T

R1(config)#hostname SanJose

SanJose(config)#interface Loopback0

SanJose(config-if)#ip address 10.1.1.1 255.255.255.0

SanJose(config-if)#interface Serial3/0

SanJose(config-if)#ip address 192.168.1.5 255.255.255.252

SanJose(config-if)#clock rate 128000

SanJose(config-if)#no shutdown

SanJose(config-if)#

A screenshot of a computer program

Description automatically generated

**Router 2**

**Router R2 (hostname ISP)**

R2#CONF T

R2(config)#hostname ISP

ISP(config)#interface Loopback0

ISP(config-if)#ip address 10.2.2.1 255.255.255.0

ISP(config-if)#interface Serial3/0

ISP(config-if)#ip address 192.168.1.6 255.255.255.252

ISP(config-if)#no shutdown

ISP(config-if)#interface Serial3/1

ISP(config-if)#ip address 172.24.1.17 255.255.255.252

ISP(config-if)#clock rate 128000

ISP(config-if)#no shutdown

ISP(config-if)#

A screenshot of a computer

Description automatically generated

**Router 3**

R3#CONF T

R3(config)#hostname CustRtr

CustRtr(config)#interface Loopback0

CustRtr(config-if)#ip address 10.3.3.1 255.255.255.0

CustRtr(config-if)#interface Serial3/1

CustRtr(config-if)#ip address 172.24.1.18 255.255.255.252

CustRtr(config-if)#no shutdown

CustRtr(config-if)#

**Step 3: Configure BGP.**

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.

SanJose(config)# router bgp 100

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

SanJose(config-router)# network 10.1.1.0 mask 255.255.255.0

A screenshot of a computer program

Description automatically generated

ISP(config)# router bgp 300

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

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

ISP(config-router)# network 10.2.2.0 mask 255.255.255.0

**R1**

CustRtr(config)# router bgp 65000

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

CustRtr(config-router)# network 10.3.3.0 mask 255.255.255.0

A screen shot of a computer

Description automatically generated

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

ISP# show ip bgp neighbors

R2

Step 4: Remove the private AS.

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

SanJose# show ip route

A computer screen shot of a black screen

Description automatically generated

R3

A screenshot of a computer screen

Description automatically generated

R1

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

ping 10.3.3.1 source 10.1.1.1

A screenshot of a computer

Description automatically generated

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

SanJose# ping

Protocol [ip]:

Target IP address: 10.3.3.1

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: y

Source address or interface: 10.1.1.1

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort

Check the BGP table from SanJose by using the show ip bgp command. Note the AS path for the

10.3.3.0 network. The AS 65000 should be listed in the path to 10.3.3.0.

SanJose# show ip bgp

Configure ISP to strip the private AS numbers from BGP routes exchanged with SanJose using the

following commands.

ISP(config)# router bgp 300

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

R2

After issuing these commands, use the clear ip bgp \* command on ISP to reestablish the BGP relationship between the three routers. Wait several seconds and then return to SanJose to check its routing table.

SanJose# ping 10.3.3.1 source lo0

A black screen with white text

Description automatically generated

R3

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

SanJose# show ip bgp

A computer screen with white text

Description automatically generated

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

R1

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

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

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

A screen shot of a computer

Description automatically generated

R2

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

ISP(config)# router bgp 300

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

A computer screen with white text

Description automatically generated

R3

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

Note: To force the local router to resend its BGP table, a less disruptive option is to use the clear ip bgp \* out or clear ip bgp \* soft command (the second command performs both outgoing and incoming route resync).

ISP#clear ip bgp \*

ISP# show ip route

A screen shot of a computer

Description automatically generated

R2

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

A computer screen with white text

Description automatically generated

R1

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

ISP# show ip bgp regexp ^100$

A screenshot of a computer screen

Description automatically generated

R1

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

ISP# tclsh

foreach address {

10.1.1.1

10.2.2.1

10.3.3.1

192.168.1.5

192.168.1.6

172.24.1.17

172.24.1.18

} {

ping $address }

A black screen with white text

Description automatically generated

R1

A computer screen shot of a black screen

Description automatically generated

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

R1

A screenshot of a computer screen

Description automatically generated

A screen shot of a computer

Description automatically generated

A computer screen with white text

Description automatically generated

R2

A computer screen with white text

Description automatically generated

R2

A screenshot of a computer screen

Description automatically generated

R3

A screenshot of a computer program

Description automatically generated

R2

A screenshot of a computer

Description automatically generated

R2

A screenshot of a computer program

Description automatically generated

**Practical No: 03**

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

**Topology**

A diagram of a network

Description automatically generated

**Objectives**

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

**Background**

The International Travel Agency runs BGP on its SanJose1 and SanJose2 routers externally with the ISP router in AS 200. IBGP is run internally between SanJose1 and SanJose2. Your job is to configure both EBGP and IBGP for this internetwork to allow for redundancy. The metered T1 should only be used in the event that the primary T1 link has failed. Traffic sent across the metered T1 link offers the same bandwidth of the primary link but at a huge expense. Ensure that this link is not used unnecessarily.

**Note:** This lab uses Cisco 1941 routers with Cisco IOS Release 15.4 with IP Base. The switches are Cisco WS-C2960-24TT-L with Fast Ethernet interfaces, therefore the router will use routing metrics associated with a 100 Mb/s interface. Depending on the router or switch model and Cisco IOS Software version, the commands available and output produced might vary from what is shown in this lab.

**Required Resources**

* 3 routers (Cisco IOS Release 15.2 or comparable)
* Serial and Ethernet cables

A diagram of a network

Description automatically generated

**Step 0: Suggested starting configurations.**

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

Router(config)# **no ip domain-lookup**

Router(config)# **line con 0**

Router(config-line)# **logging synchronous**

Router(config-line)# **exec-timeout 0 0**

A screenshot of a computer screen

Description automatically generated

**Step 1: Configure interface addresses.**

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

**Router R1 (hostname ISP)**

ISP(config)# **interface Loopback0**

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

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

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

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

ISP(config-if)# **clock rate 128000**

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

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

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

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

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

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

ISP#

A screenshot of a computer screen

Description automatically generated

**Router R2 (hostname SanJose1)**

SanJose1(config)# **interface Loopback0**

SanJose1(config-if)# **ip address 172.16.64.1 255.255.255.0**

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

SanJose1(config)# **interface Serial0/0/0**

SanJose1(config-if)# **ip address 192.168.1.6 255.255.255.252**

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

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

SanJose1(config)# **interface Serial0/0/1**

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

SanJose1(config-if)# **clock rate 128000**

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

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

SanJose1#

A screenshot of a computer program

Description automatically generated

**Router R3 (hostname SanJose2)**

SanJose2(config)# **interface Loopback0**

SanJose2(config-if)# **ip address 172.16.32.1 255.255.255.0**

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

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

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

SanJose2(config-if)# **clock rate 128000**

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

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

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

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

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

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

SanJose2#

A screenshot of a computer program

Description automatically generated

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

**Step 2: Configure EIGRP.**

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

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

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



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

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



**Step 3: Configure IBGP and verify BGP neighbors.**

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

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

SanJose1(config-router)# **neighbor 172.16.32.1 remote-as 64512**

SanJose1(config-router)# **neighbor 172.16.32.1 update-source lo0**

A number on a black background

Description automatically generated

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 Loopback0 as the source IP address for all BGP messages sent to that neighbor.

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

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

SanJose2(config-router)# **neighbor 172.16.64.1 remote-as 64512**

SanJose2(config-router)# **neighbor 172.16.64.1 update-source lo0**

F,{708eae6a-97aa-4fe2-a450-bcbcd30b4e4c}{197},10.45833,1.0625

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

SanJose2# **show ip bgp neighbors**

BGP neighbor is 172.16.64.1, remote AS 64512, internal link

BGP version 4, remote router ID 172.16.64.1

BGP state = Established, up for 00:00:22

Last read 00:00:22, last write 00:00:22, hold time is 180, keepalive interval is 60 seconds

<output omitted>

A screenshot of a computer program

Description automatically generated

The link between SanJose1 and SanJose2 should be identified as an internal link indicating an IBGP peering relationship, as shown in the output.

**Step 4: Configure EBGP and verify BGP neighbors.**

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

ISP(config)# **router bgp 200**

ISP(config-router)# **neighbor 192.168.1.6 remote-as 64512**

ISP(config-router)# **neighbor 192.168.1.2 remote-as 64512**

ISP(config-router)# **network 192.168.100.0**

A screenshot of a computer

Description automatically generated

Because EBGP sessions are almost always established over point-to-point links, there is no reason to use the **update-source** keyword in this configuration. Only one path exists between the peers. If this path goes down, alternative paths are not available.

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.

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



1. Configure SanJose1 as an EBGP peer to ISP.

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

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

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



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

SanJose1# **show ip bgp neighbors**

BGP neighbor is 172.16.32.1, remote AS 64512, internal link

BGP version 4, remote router ID 172.16.32.1

BGP state = Established, up for 00:12:43

<output omitted>

BGP neighbor is 192.168.1.5, remote AS 200, external link

BGP version 4, remote router ID 192.168.100.1

BGP state = Established, up for 00:06:49

Last read 00:00:42, last write 00:00:45, hold time is 180, keepalive interval is 60 seconds

<output omitted>

Notice that the “external link” indicates that an EBGP peering session has been established. You should also see an informational message indicating the establishment of the BGP neighbor relationship.

\*Sep 8 21:09:59.699: %BGP-5-ADJCHANGE: neighbor 192.168.1.5 Up

A screenshot of a computer program

Description automatically generated

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

SanJose2(config)# **ip route 172.16.0.0 255.255.0.0 null0**

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

SanJose2(config-router)# **neighbor 192.168.1.1 remote-as 200**

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

A screenshot of a computer

Description automatically generated

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

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

SanJose2# **show ip bgp summary**

BGP router identifier 172.16.32.1, local AS number 64512

BGP table version is 6, main routing table version 6

2 network entries using 288 bytes of memory

4 path entries using 320 bytes of memory

4/2 BGP path/bestpath attribute entries using 640 bytes of memory

1 BGP AS-PATH entries using 24 bytes of memory

0 BGP route-map cache entries using 0 bytes of memory

0 BGP filter-list cache entries using 0 bytes of memory

BGP using 1272 total bytes of memory

BGP activity 2/0 prefixes, 4/0 paths, scan interval 60 secs

Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd

172.16.64.1 4 64512 27 26 6 0 0 00:18:15 2

192.168.1.1 4 200 10 7 6 0 0 00:01:42 1

SanJose2#

A computer screen shot of a black screen

Description automatically generated

**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 conversations to reestablish with each SanJose router.

ISP# **clear ip bgp \***

ISP#

\*Nov 9 22:05:32.427: %BGP-5-ADJCHANGE: neighbor 192.168.1.2 Down User reset

\*Nov 9 22:05:32.427: %BGP\_SESSION-5-ADJCHANGE: neighbor 192.168.1.2 IPv4 Unicast topology base removed from session User reset

\*Nov 9 22:05:32.427: %BGP-5-ADJCHANGE: neighbor 192.168.1.6 Down User reset

\*Nov 9 22:05:32.427: %BGP\_SESSION-5-ADJCHANGE: neighbor 192.168.1.6 IPv4 Unicast topology base removed from session User reset

\*Nov 9 22:05:32.851: %BGP-5-ADJCHANGE: neighbor 192.168.1.2 Up

\*Nov 9 22:05:32.851: %BGP-

ISP#5-ADJCHANGE: neighbor 192.168.1.6 Up

ISP#

A screen shot of a computer

Description automatically generated

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

ISP# **ping 172.16.64.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.64.1, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

ISP#

A screen shot of a computer

Description automatically generated

ISP# **ping 172.16.1.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

ISP#

A computer screen with numbers and text

Description automatically generated

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

ISP# **ping 172.16.32.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.32.1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 12/14/16 ms

A computer screen with white text

Description automatically generated

ISP# **ping 172.16.1.2**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.1.2, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 12/13/16 ms

ISP#

A screen shot of a computer

Description automatically generated

You should see successful pings to each IP address on SanJose2 router. Ping attempts to 172.16.64.1 and 172.16.1.1 should fail. Why does this happen?

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

ISP# **show ip bgp**

BGP table version is 3, local router ID is 192.168.100.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* 172.16.0.0 192.168.1.6 0 0 64512 i

\*> 192.168.1.2 0 0 64512 i

\*> 192.168.100.0 0.0.0.0 0 32768 i

ISP#

ISP# **show ip bgp**

A screenshot of a computer

Description automatically generated

Notice that ISP has two valid routes to the 172.16.0.0 network, as indicated by the **.** However, the link to SanJose2 has been selected as the best path, indicated by the inclusion of the “>”. Why did the ISP prefer the link to SanJose2 over SanJose1?

Would changing the bandwidth metric on each link help to correct this issue? Explain

BGP operates differently than all other protocols. Unlike other routing protocols that use complex algorithms involving factors such as bandwidth, delay, reliability, and load to formulate a metric, BGP is policy-based. BGP determines the best path based on variables, such as AS path, weight, local preference, MED, and so on. If all things are equal, BGP prefers the route leading to the BGP speaker with the lowest BGP router ID. The SanJose2 router with BGP router ID 172.16.32.1 was preferred to the higher BGP router ID of the SanJose1 router (172.16.64.1).

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

ISP# **ping 172.16.1.1 source 192.168.100.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:

Packet sent with a source address of 192.168.100.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 20/21/24 ms

A screen shot of a computer

Description automatically generated

ISP# **ping 172.16.32.1 source 192.168.100.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.32.1, timeout is 2 seconds:

Packet sent with a source address of 192.168.100.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 12/15/16 ms

A computer screen shot of a black screen

Description automatically generated

ISP# **ping 172.16.1.2 source 192.168.100.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.1.2, timeout is 2 seconds:

Packet sent with a source address of 192.168.100.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 12/15/16 ms

ISP#

A computer screen with white text

Description automatically generated

ISP# **ping 172.16.64.1 source 192.168.100.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.64.1, timeout is 2 seconds:

Packet sent with a source address of 192.168.100.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 20/21/24 ms

A computer screen shot of numbers

Description automatically generated

You can also use the extended ping dialogue to specify the source address, as shown in this example.

ISP# **ping**

Protocol [ip]:

Target IP address: **172.16.64.1**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: **y**

Source address or interface: **192.168.100.1**

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.64.1, timeout is 2 seconds:

Packet sent with a source address of 192.168.100.1

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/24 ms

ISP#

A computer screen shot of a black screen

Description automatically generated

Complete reachability has been demonstrated between the ISP router and both SanJose1 and SanJose2.

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

SanJose1 is unaware of the link between ISP and SanJose2, and SanJose2 is unaware of the link between ISP and SanJose1. Before ISP can successfully ping all the internal serial interfaces of AS 64512, these serial links should be advertised via BGP on the ISP router. This can also be resolved via EIGRP on each SanJose router. One method is for ISP to advertise these links.

1. Issue the following commands on the ISP router.

ISP(config)# **router bgp 200**

ISP(config-router)# **network 192.168.1.0 mask 255.255.255.252**

ISP(config-router)# **network 192.168.1.4 mask 255.255.255.252**

A black background with white numbers

Description automatically generated

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

ISP# **show ip bgp**

BGP table version is 5, local router ID is 192.168.100.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* 172.16.0.0 192.168.1.6 0 0 64512 i

\*> 192.168.1.2 0 0 64512 i

\*> 192.168.1.0/30 0.0.0.0 0 32768 i

\*> 192.168.1.4/30 0.0.0.0 0 32768 i

\*> 192.168.100.0 0.0.0.0 0 32768 i

ISP#

A computer screen with white text

Description automatically generated

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

SanJose2# **show ip route**

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

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.2/32 is directly connected, Serial0/0/1

C 172.16.32.0/24 is directly connected, Loopback0

L 172.16.32.1/32 is directly connected, Loopback0

D 172.16.64.0/24 [90/2297856] via 172.16.1.1, 00:52:03, Serial0/0/1

192.168.1.0/24 is variably subnetted, 3 subnets, 2 masks

C 192.168.1.0/30 is directly connected, Serial0/0/0

L 192.168.1.2/32 is directly connected, Serial0/0/0

B 192.168.1.4/30 [20/0] via 192.168.1.1, 00:01:03

B 192.168.100.0/24 [20/0] via 192.168.1.1, 00:25:20

SanJose2#

A screenshot of a computer screen

Description automatically generated

The next issue to consider is BGP policy routing between autonomous systems. The next-hop attribute of a route in a different AS is set to the IP address of the border router in the next AS toward the destination, and this attribute is not modified by default when advertising this route through IBGP. Therefore, for all IBGP peers, it is either necessary to know the route to that border router (in a different neighboring AS), or our own border router needs to advertise the foreign routes using the next-hop-self feature, overriding the next-hop address with its own IP address. The SanJose2 router is passing a policy to SanJose1 and vice versa. The policy for routing from AS 64512 to AS 200 is to forward packets to the 192.168.1.1 interface. SanJose1 has a similar yet opposite policy: it forwards requests to the 192.168.1.5 interface. If either WAN link fails, it is critical that the opposite router become a valid gateway. This is achieved if the **next-hop-self** command is configured on SanJose1 and SanJose2.

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 SanJose2. The only possible path from SanJose2 to ISP’s 192.168.100.0/24 is through SanJose1.

ISP(config)# **router bgp 200**

ISP(config-router)# **no** **network 192.168.1.0 mask 255.255.255.252**

ISP(config-router)# **no** **network 192.168.1.4 mask 255.255.255.252**

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

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

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

ISP(config-if)#

A black screen with white text

Description automatically generated

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

SanJose2# **show ip bgp**

BGP table version is 1, local router ID is 172.16.32.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* i 172.16.0.0 172.16.64.1 0 100 0 i

\* i 192.168.100.0 192.168.1.5 0 100 0 200 i

SanJose2#

A screenshot of a computer

Description automatically generated

SanJose2# **show ip route**

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

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

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

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.2/32 is directly connected, Serial0/0/1

C 172.16.32.0/24 is directly connected, Loopback0

L 172.16.32.1/32 is directly connected, Loopback0

D 172.16.64.0/24 [90/2297856] via 172.16.1.1, 02:41:46, Serial0/0/1

SanJose2#

A screenshot of a computer screen

Description automatically generated

Notice that SanJose2 has 192.168.100.0 in it’s BGP table but not in its routing table. The BGP table shows the next hop to 192.168.100.0 as 192.168.1.5. Because SanJose2 does not have a route to this next hop address of 192.168.1.5 in its routing table, it will not install the 192.168.100.0 network into the routing table. It won’t install a route if it doesn’t know how to get to the next hop.

EBGP next hop addresses are carried into IBGP unchanged. As we saw previously, we could advertise the WAN link using BGP, but this is not always desirable. It means advertising additional routes when we are usually trying to minimize the size of the routing table. Another option is to have the routers within the IGP domain advertise themselves as the next hop router using the **next-hop-self** command.

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

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

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

A black screen with white text

Description automatically generated

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

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



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

SanJose1# **clear ip bgp \***

SanJose1#

A screen shot of a computer

Description automatically generated

SanJose2# **clear ip bgp \***

SanJose2#

A black screen with white text

Description automatically generated

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

SanJose2# **show ip bgp**

BGP table version is 5, local router ID is 172.16.32.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\*> 172.16.0.0 0.0.0.0 0 32768 i

\* i 172.16.64.1 0 100 0 i

\*>i 192.168.100.0 172.16.64.1 0 100 0 200 i

SanJose2#

A screenshot of a computer

Description automatically generated

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

SanJose2# **show ip route**

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

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.2/32 is directly connected, Serial0/0/1

C 172.16.32.0/24 is directly connected, Loopback0

L 172.16.32.1/32 is directly connected, Loopback0

D 172.16.64.0/24 [90/2297856] via 172.16.1.1, 04:27:19, Serial0/0/1

B 192.168.100.0/24 [200/0] via 172.16.64.1, 00:00:46

SanJose2#

A screenshot of a computer screen

Description automatically generated

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

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

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

ISP(config-if)#

A black background with white text

Description automatically generated

SanJose2# **show ip route**

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

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

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

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.2/32 is directly connected, Serial0/0/1

C 172.16.32.0/24 is directly connected, Loopback0

L 172.16.32.1/32 is directly connected, Loopback0

D 172.16.64.0/24 [90/2297856] via 172.16.1.1, 04:37:34, Serial0/0/1

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

C 192.168.1.0/30 is directly connected, Serial0/0/0

L 192.168.1.2/32 is directly connected, Serial0/0/0

B 192.168.100.0/24 [20/0] via 192.168.1.1, 00:01:35

SanJose2#

A screenshot of a computer screen

Description automatically generated

**Step 8: Set BGP local preference.**

At this point, everything looks good, with the exception of default routes, the outbound flow of data, and inbound packet flow.

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

SanJose1(config)# **route-map PRIMARY\_T1\_IN permit 10**

SanJose1(config-route-map)# **set local-preference 150**

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

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

SanJose1(config-router)# **neighbor 192.168.1.5 route-map PRIMARY\_T1\_IN in**

A computer screen with white text

Description automatically generated

SanJose2(config)# **route-map SECONDARY\_T1\_IN permit 10**

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

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

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

SanJose2(config-router)# **neighbor 192.168.1.1 route-map SECONDARY\_T1\_IN in**

A screen shot of a computer

Description automatically generated

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

SanJose1# **clear ip bgp \* soft**



SanJose2# **clear ip bgp \* soft**



SanJose1# **show ip bgp**

BGP table version is 3, local router ID is 172.16.64.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* i 172.16.0.0 172.16.32.1 0 100 0 i

\*> 0.0.0.0 0 32768 i

\*> 192.168.100.0 192.168.1.5 0 150 0 200 i

SanJose1#

A computer screen shot of a black background

Description automatically generated

SanJose2# **show ip bgp**

BGP table version is 7, local router ID is 172.16.32.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* i 172.16.0.0 172.16.64.1 0 100 0 i

\*> 0.0.0.0 0 32768 i

\*>i 192.168.100.0 172.16.64.1 0 150 0 200 i

\* 192.168.1.1 0 125 0 200 i

SanJose2#

A screenshot of a computer

Description automatically generated

This now indicates that routing to the loopback segment for ISP 192.168.100.0 /24 can be reached only through the link common to SanJose1 and ISP. SanJose2’s next hop to 192.168.100.0/24 is SanJose1 because both routers have been configured using the **next-hop-self** command.

**Step 9: Set BGP MED.**

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

ISP# **show ip bgp**

BGP table version is 22, local router ID is 192.168.100.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* 172.16.0.0 192.168.1.6 0 0 64512 i

\*> 192.168.1.2 0 0 64512 i

\*> 192.168.100.0 0.0.0.0 0 32768 i

ISP# **show ip route**

A computer screen shot of a black screen

Description automatically generated

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

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is not set

B 172.16.0.0/16 [20/0] via 192.168.1.2, 00:12:45

192.168.1.0/24 is variably subnetted, 4 subnets, 2 masks

C 192.168.1.0/30 is directly connected, Serial0/0/1

L 192.168.1.1/32 is directly connected, Serial0/0/1

C 192.168.1.4/30 is directly connected, Serial0/0/0

L 192.168.1.5/32 is directly connected, Serial0/0/0

192.168.100.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.100.0/24 is directly connected, Loopback0

L 192.168.100.1/32 is directly connected, Loopback0

ISP#

A screenshot of a computer screen

Description automatically generated

How will traffic from network 192.168.100.0 /24 on ISP return to SanJose1 or SanJose2? Will it be routed through SanJose1 or SanJose2?

To verify this, the simplest solution is to issue the **show ip bgp** command on the ISP router as was done above. What if access was not given to the ISP router? Traffic returning from the Internet should not be passed across the metered T1. Is there a simple way to verify before receiving the monthly bill? How can it be checked instantly?

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

SanJose2# **ping**

Protocol [ip]:

Target IP address: **192.168.100.1**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: **y**

Source address or interface: **172.16.32.1**

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]: **record**

Number of hops [ 9 ]:

Loose, Strict, Record, Timestamp, Verbose[RV]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.100.1, timeout is 2 seconds:

Packet sent with a source address of 172.16.32.1

Packet has IP options: Total option bytes= 39, padded length=40

Record route: <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

Reply to request 0 (20 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 1 (20 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 2 (20 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 3 (24 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 4 (20 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/24 ms

SanJose2#

A screenshot of a computer program

Description automatically generated

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

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

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

SanJose1(config)#**route-map PRIMARY\_T1\_MED\_OUT permit 10**

SanJose1(config-route-map)#**set Metric 50**

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

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

SanJose1(config-router)#**neighbor 192.168.1.5 route-map PRIMARY\_T1\_MED\_OUT out**

A screen shot of a computer

Description automatically generated

SanJose2(config)#**route-map SECONDARY\_T1\_MED\_OUT permit 10**

SanJose2(config-route-map)#**set Metric 75**

SanJose2(config-route-map)#**exit**

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

SanJose2(config-router)#**neighbor 192.168.1.1 route-map SECONDARY\_T1\_MED\_OUT out**

A computer screen shot of a black screen

Description automatically generated

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

SanJose1# **clear ip bgp \* soft**



SanJose2# **clear ip bgp \* soft**



SanJose1# **show ip bgp**

BGP table version is 4, local router ID is 172.16.64.1

A computer screen shot of a black screen

Description automatically generated

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* i 172.16.0.0 172.16.32.1 0 100 0 i

\*> 0.0.0.0 0 32768 i

\*> 192.168.100.0 192.168.1.5 0 150 0 200 i

SanJose1#

A computer screen shot of a black screen

Description automatically generated

SanJose2# **show ip bgp**

BGP table version is 8, local router ID is 172.16.32.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\* i 172.16.0.0 172.16.64.1 0 100 0 i

\*> 0.0.0.0 0 32768 i

\*>i 192.168.100.0 172.16.64.1 0 150 0 200 i

\* 192.168.1.1 0 125 0 200 i

SanJose2#

A screenshot of a computer

Description automatically generated

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

SanJose2# **ping**

Protocol [ip]:

Target IP address: **192.168.100.1**

Repeat count [5]:

Datagram size [100]:

Timeout in seconds [2]:

Extended commands [n]: **y**

Source address or interface: **172.16.32.1**

Type of service [0]:

Set DF bit in IP header? [no]:

Validate reply data? [no]:

Data pattern [0xABCD]:

Loose, Strict, Record, Timestamp, Verbose[none]: record

Number of hops [ 9 ]:

Loose, Strict, Record, Timestamp, Verbose[RV]:

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.100.1, timeout is 2 seconds:

Packet sent with a source address of 172.16.32.1

Packet has IP options: Total option bytes= 39, padded length=40

Record route: <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

Reply to request 0 (28 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.5)

(172.16.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 1 (28 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.5)

(172.16.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 2 (28 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.5)

(172.16.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 3 (28 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.5)

(172.16.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Reply to request 4 (28 ms). Received packet has options

Total option bytes= 40, padded length=40

Record route:

(172.16.1.2)

(192.168.1.6)

(192.168.100.1)

(192.168.1.5)

(172.16.1.1)

(172.16.32.1) <\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

End of list

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/28 ms

SanJose2#

A screenshot of a computer program

Description automatically generated

Does the output look correct? Does the 192.168.1.5 above mean that the ISP now prefers SanJose1 for return traffic?

The newly configured policy MED shows that the lower MED value is considered best. The ISP now prefers the route with the lower MED value of 50 to AS 64512. This is just opposite from the **local-preference** command configured earlier.

ISP# **show ip bgp**

BGP table version is 24, local router ID is 192.168.100.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\*> 172.16.0.0 192.168.1.6 50 0 64512 i

\* 192.168.1.2 75 0 64512 i

\*> 192.168.100.0 0.0.0.0 0 32768 i

ISP#

A screenshot of a computer

Description automatically generated

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

ISP(config)# **router bgp 200**

ISP(config-router)# **neighbor 192.168.1.6 default-originate**

ISP(config-router)# **neighbor 192.168.1.2 default-originate**

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

ISP(config)# **interface loopback 10**

ISP(config-if)# **ip address 10.0.0.1 255.255.255.0**

ISP(config-if)#

A screen shot of a computer

Description automatically generated

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

SanJose1# **show ip route**

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

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is 192.168.1.5 to network 0.0.0.0

B\* 0.0.0.0/0 [20/0] via 192.168.1.5, 00:00:36

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.1/32 is directly connected, Serial0/0/1

D 172.16.32.0/24 [90/2297856] via 172.16.1.2, 05:47:24, Serial0/0/1

C 172.16.64.0/24 is directly connected, Loopback0

L 172.16.64.1/32 is directly connected, Loopback0

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

C 192.168.1.4/30 is directly connected, Serial0/0/0

L 192.168.1.6/32 is directly connected, Serial0/0/0

SanJose1#

A screenshot of a computer program

Description automatically generated

SanJose2# **show ip route**

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

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

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

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is 172.16.64.1 to network 0.0.0.0

B\* 0.0.0.0/0 [200/0] via 172.16.64.1, 00:00:45

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.2/32 is directly connected, Serial0/0/1

C 172.16.32.0/24 is directly connected, Loopback0

L 172.16.32.1/32 is directly connected, Loopback0

D 172.16.64.0/24 [90/2297856] via 172.16.1.1, 05:47:33, Serial0/0/1

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

C 192.168.1.0/30 is directly connected, Serial0/0/0

L 192.168.1.2/32 is directly connected, Serial0/0/0

SanJose2#

A computer screen shot of a black screen

Description automatically generated

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

SanJose2# **show ip bgp**

BGP table version is 38, local router ID is 172.16.32.1

Status codes: s suppressed, d damped, h history, \* valid, > best, i - internal,

r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,

x best-external, a additional-path, c RIB-compressed,

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

RPKI validation codes: V valid, I invalid, N Not found

Network Next Hop Metric LocPrf Weight Path

\*>i 0.0.0.0 172.16.64.1 0 150 0 200 i

\* 192.168.1.1 125 0 200 i

\* i 172.16.0.0 172.16.64.1 0 100 0 i

\*> 0.0.0.0 0 32768 i

\*>i 192.168.100.0 172.16.64.1 0 150 0 200 i

\* 192.168.1.1 0 125 0 200 i

SanJose2#

A computer screen shot of a black screen

Description automatically generated

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

SanJose2# **traceroute 10.0.0.1**

Type escape sequence to abort.

Tracing the route to 10.0.0.1

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

1 172.16.1.1 8 msec 4 msec 8 msec

2 192.168.1.5 [AS 200] 12 msec \* 12 msec

SanJose2#

A screen shot of a computer

Description automatically generated

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

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

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

ISP(config-if)#

A black background with a black square

Description automatically generated with medium confidence

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

SanJose1# **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

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is 172.16.32.1 to network 0.0.0.0

B\* 0.0.0.0/0 [200/0] via 172.16.32.1, 00:00:06

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.1/32 is directly connected, Serial0/0/1

D 172.16.32.0/24 [90/2297856] via 172.16.1.2, 05:49:25, Serial0/0/1

C 172.16.64.0/24 is directly connected, Loopback0

L 172.16.64.1/32 is directly connected, Loopback0

B 192.168.100.0/24 [200/0] via 172.16.32.1, 00:00:06

SanJose1#

A computer screen shot of a black screen

Description automatically generated

SanJose2# **show ip route**

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

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

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

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, \* - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP

a - application route

+ - replicated route, % - next hop override

Gateway of last resort is 192.168.1.1 to network 0.0.0.0

B\* 0.0.0.0/0 [20/0] via 192.168.1.1, 00:00:30

172.16.0.0/16 is variably subnetted, 6 subnets, 3 masks

S 172.16.0.0/16 is directly connected, Null0

C 172.16.1.0/24 is directly connected, Serial0/0/1

L 172.16.1.2/32 is directly connected, Serial0/0/1

C 172.16.32.0/24 is directly connected, Loopback0

L 172.16.32.1/32 is directly connected, Loopback0

D 172.16.64.0/24 [90/2297856] via 172.16.1.1, 05:49:49, Serial0/0/1

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

C 192.168.1.0/30 is directly connected, Serial0/0/0

L 192.168.1.2/32 is directly connected, Serial0/0/0

B 192.168.100.0/24 [20/0] via 192.168.1.1, 00:00:30

SanJose2#

A screenshot of a computer

Description automatically generated

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

SanJose1# **trace 10.0.0.1**

Type escape sequence to abort.

Tracing the route to 10.0.0.1

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

1 172.16.1.2 8 msec 8 msec 8 msec

2 192.168.1.1 [AS 200] 12 msec \* 12 msec

SanJose1#

A screen shot of a computer

Description automatically generated

**Practical No: 04**

**Title**: Secure the Management Plane

A diagram of a network

Description automatically generated

Objective:

• Secure management access.

• Configure enhanced username password security.

• Enable AAA RADIUS authentication.

• Enable secure remote management.

Required Resource

• 3 routers (Cisco IOS Release 15.2 or comparable)

• Serial and Ethernet cables

**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.

**Router 1**

R1#conf t

R1(config)#hostname R1

R1(config)#interface Loopback 0

R1(config-if)#description R1 LAN

R1(config-if)#ip address 192.168.1.1 255.255.255.0

R1(config-if)#

R1(config-if)#EXIT

R1(config)#interface Serial0/0

R1(config-if) #description R1 --> R2

R1(config-if) #ip address 10.1.1.1 255.255.255.252

R1(config-if) #clock rate 128000

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#end

A screenshot of a computer program

Description automatically generated

**Router 2**

R2#conf t

R2(config)#hostname R2

R2(config)#interface Serial0/0

R2(config-if)#description R2 --> R1

R2(config-if)#ip address 10.1.1.2 255.255.255.252

R2(config-if)#no shutdown

R2(config-if)#exit

R2(config)#interface Serial0/1

R2(config-if)#description R2 --> R3

R2(config-if)#ip address 10.2.2.1 255.255.255.252

R2(config-if)#clock rate 128000

R2(config-if)#no shutdown

R2(config-if)#exit

R2(config)#end

A computer screen shot of a black screen

Description automatically generated

**Router 3**

R3#conf t

R3(config)#hostname R3

R3(config)#interface Loopback0

R3(config-if)#description R3 LAN

R3(config-if)#ip address 192.168.3.1 255.255.255.0

R3(config-if)#exit

R3(config)#interface Serial0/1

R3(config-if)#description R3 --> R2

R3(config-if)#ip address 10.2.2.2 255.255.255.252

R3(config-if)#no shutdown

R3(config-if)#exit

R3(config)#end

A screenshot of a computer program

Description automatically generated

**Step 2 : Configure static routes.**

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

R1(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.2



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

R3(config)# ip route 0.0.0.0 0.0.0.0 10.2.2.1



**c. On R2, configure two static routes.**

R2(config)# ip route 192.168.1.0 255.255.255.0 10.1.1.1

R2(config)# ip route 192.168.3.0 255.255.255.0 10.2.2.2

A black background with white text

Description automatically generated

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

R1#tclsh

R1(tcl)#foreach address {

+>(tcl)#192.168.1.1

+>(tcl)#10.1.1.1

+>(tcl)#10.1.1.2

+>(tcl)#10.2.2.1

+>(tcl)#10.2.2.2

+>(tcl)#192.168.3.1

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

A computer screen shot of a number

Description automatically generated

A computer screen with white text

Description automatically generated

**Step 3: Secure management access.**

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

R1(config)# security passwords min-length 10

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

R1(config)# enable secret class12345

A black background with white text

Description automatically generated

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

R1(config)# line console 0

R1(config-line)# password ciscoconpass

R1(config-line)# exec-timeout 5 0

R1(config-line)# login

R1(config-line)# logging synchronous

R1(config-line)# exit

R1(config)#

A computer screen shot of white text

Description automatically generated

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

R1(config)# line vty 0 4

R1(config-line)# password ciscovtypass

R1(config-line)# exec-timeout 5 0

R1(config-line)# login

R1(config-line)# exit

R1(config)#

A screen shot of a computer code

Description automatically generated

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

R1(config)# line aux 0

R1(config-line)# no exec

R1(config-line)# end

A black background with white text

Description automatically generated

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

R1(config)# service password-encryption

A black background with white text

Description automatically generated

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

R1(config)# banner motd $Unauthorized access strictly prohibited! $

R1(config)# exit

A black background with white text

Description automatically generated

**Step 4: Configure enhanced username password security.**

To increase the encryption level of console and VTY lines, it is recommended to enable authentication using the local database. The local database consists of usernames and password combinations that are created locally on each device. The local and VTY lines are configured to refer to the local database when authenticating a user.

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

R1(config)# username JR-ADMIN secret class12345

R1(config)# username ADMIN secret class54321

A black screen with white text

Description automatically generated

**b. 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

A black background with white text

Description automatically generated

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

R1(config)# line vty 0 4

R1(config-line)# login local

R1(config-line)# end

A black background with white text

Description automatically generated

**d. 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:**

R3(config)# username JR-ADMIN secret class12345

R3(config)# username ADMIN secret class54321

A black background with white text

Description automatically generated

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

R3(config)# line console 0

R3(config-line)# login local

R3(config-line)# exit

A black background with white text

Description automatically generated

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

R3(config)# line vty 0 4

R3(config-line)# login local

R3(config-line)# end

A black background with white text

Description automatically generated

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

R1# telnet 10.2.2.2

Trying 10.2.2.2 ... Open

Unauthorized access strictly prohibited!

User Access Verification

Username: ADMIN

Password:

R3>

A black screen with white text

Description automatically generated

A black background with white text

Description automatically generated

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

Authentication, authorization, and accounting (AAA) is a standards-based framework that can be implemented to control who is permitted to access a network (authenticate), what they can do on that network (authorize), and audit what they did while accessing the network (accounting).

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

R1(config)# aaa new-model



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

A black screen with white text

Description automatically generated

**d. Assign both RADIUS servers to a server group.**

A black screen with white text

Description automatically generated

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

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



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

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



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

R1(config)# line vty 0 4

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

R1(config-line)# exit

A black background with white text

Description automatically generated

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

R3(config)# aaa new-model

A black background with white text

Description automatically generated

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



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

A black background with white text

Description automatically generated

**Assign both RADIUS servers to a server group.**

A black screen with white text

Description automatically generated

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

R3(config)# aaa authentication login default group RADIUS-GROUP local



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

R3(config)# aaa authentication login TELNET-LOGIN group RADIUS-GROUP local-case



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

R3(config)# line vty 0 4

R3(config-line)# login authentication TELNET-LOGIN

R3(config-line)# exit

A black background with white text

Description automatically generated

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

R1# telnet 10.2.2.2

Username: ADMIN

Password: class54321

A screen shot of a computer

Description automatically generated

A black background with white text

Description automatically generated

**Step 6: Enabling secure remote management using SSH.**

Traditionally, remote access on routers was configured using Telnet on TCP port 23. However, Telnet was developed in the days when security was not an issue; therefore, all Telnet traffic is forwarded in plaintext. Secure Shell (SSH) is a network protocol that establishes a secure terminal emulation connection to a router or other networking device. SSH encrypts all information that passes over the network link and provides authentication of the remote computer. SSH is rapidly replacing Telnet as the remote login tool of choice for network professionals.

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

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

A black background with white text

Description automatically generated

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

R1(config)# crypto key zeroize rsa



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

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

A computer screen with white text

Description automatically generated

**d. Cisco routers support two versions of SSH:**

R1(config)# ip ssh version 2



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

R1(config)# line vty 0 4

R1(config-line)# transport input ssh

R1(config-line)# end

A black background with white text

Description automatically generated

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

R1# show ip ssh

A black background with white text

Description automatically generated

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

R3(config)# ip domain-name ccnasecurity.com



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

R3(config)# crypto key zeroize rsa

A black background with white text

Description automatically generated

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

R3(config)# crypto key generate rsa general-keys modulus 1024

A computer screen with white text

Description automatically generated

**Cisco routers support two versions of SSH:**

R3(config)# ip ssh version 2



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

R3(config)# line vty 0 4

R3(config-line)# transport input ssh

R3(config-line)# end

A black screen with white text

Description automatically generated

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

R3# show ip ssh



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

R1# ssh -l ADMIN 10.2.2.2

Password:

R3>

R3>en

% Error in authentication.

R3>

A screen shot of a computer

Description automatically generated

**Practical No: 05**

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

A diagram of a network

Description automatically generated

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

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

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

**Router R1**

R1#conf t

R1(config)#hostname R1

R1(config)#interface Lo1

R1(config-if)#description R1 LAN

R1(config-if)#ip address 192.168.1.1 255.255.255.0

R1(config-if)#interface Serial0/0

R1(config-if)#description R1 --> R2

R1(config-if)#ip address 172.16.12.1 255.255.255.248

R1(config-if)#clock rate 128000

R1(config-if)#bandwidth 128

R1(config-if)#no shutdown

R1(config-if)#interface Serial0/1

R1(config-if)#description R1 --> R3

R1(config-if)#ip address 172.16.13.1 255.255.255.248

R1(config-if)#bandwidth 64

R1(config-if)#no shutdown

R1(config-if)#end

A screenshot of a computer program

Description automatically generated

**Router R2**

R2#conf t

R2(config)#hostname R2

R2(config)#interface Lo2

R2(config-if)#description R2 LAN

R2(config-if)#ip address 192.168.2.1 255.255.255.0

R2(config-if)#interface Serial0/0

R2(config-if)#description R2 --> R1

R2(config-if)#ip address 172.16.12.2 255.255.255.248

R2(config-if)#bandwidth 128

R2(config-if)#no shutdown

R2(config-if)#interface Serial1/1

R2(config-if)#description R2 --> R3

R2(config-if)#ip address 172.16.23.2 255.255.255.248

R2(config-if)#clock rate 128000

R2(config-if)#bandwidth 128

R2(config-if)#no shutdown

R2(config-if)#end

A computer screen with white text

Description automatically generated

**Router R3**

R3#conf t

R3(config)#hostname R3

R3(config)#interface Lo3

R3(config-if)#description R3 LAN

R3(config-if)#ip address 192.168.3.1 255.255.255.0

R3(config-if)#interface Serial0/0

R3(config-if)#description R3 --> R1

R3(config-if)#ip address 172.16.13.3 255.255.255.248

R3(config-if)#clock rate 64000

R3(config-if)#bandwidth 64

R3(config-if)#no shutdown

R3(config-if)#interface Serial0/1

R3(config-if)#description R3 --> R2

R3(config-if)#ip address 172.16.23.3 255.255.255.248

R3(config-if)#bandwidth 128

R3(config-if)#no shutdown

R3(config-if)#interface Serial0/2

R3(config-if)#description R3 --> R4

R3(config-if)#ip address 172.16.34.3 255.255.255.248

R3(config-if)#clock rate 64000

R3(config-if)#bandwidth 64

R3(config-if)#no shutdown

R3(config-if)#end

A computer screen shot of a black screen

Description automatically generated

**Router R4**

R4#conf t

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

R4(config)#hostname R4

R4(config)#interface Lo4

R4(config-if)#description R4 LAN A

R4(config-if)#ip address 192.168.4.1 255.255.255.128

R4(config-if)#interface Lo5

R4(config-if)#description R4 LAN B

R4(config-if)#ip address 192.168.4.129 255.255.255.128

R4(config-if)#interface Serial0/2

R4(config-if)#description R4 --> R3

R4(config-if)#ip address 172.16.34.4 255.255.255.248

R4(config-if)#bandwidth 64

R4(config-if)#no shutdown

R4(config-if)#end

A screenshot of a computer program

Description automatically generated

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

R3# show ip interface brief

A black background with white text

Description automatically generated

R3# show protocols

A screenshot of a computer

Description automatically generated

R3# show interfaces description

A black screen with white text

Description automatically generated

**Step 3: Configure basic EIGRP.**

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

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

R1(config)#router eigrp 1

R1(config-router)#network 192.168.1.0

R1(config-router)#network 172.16.12.0 0.0.0.7

R1(config-router)#network 172.16.13.0 0.0.0.7

R1(config-router)#no auto-summary

A screenshot of a computer screen

Description automatically generated

R2#conf t

R2(config)#router eigrp 1

R2(config-router)#network 192.168.2.0

R2(config-router)#network 172.16.12.0 0.0.0.7

R2(config-router)#network 172.16.23.0 0.0.0.7

R2(config-router)#no auto-summary

A computer screen with white text

Description automatically generated

R3(config)#router eigrp 1

R3(config-router)#network 192.168.3.0

R3(config-router)#network 172.16.13.0 0.0.0.7

R3(config-router)#network 172.16.23.0 0.0.0.7

R3(config-router)#network 172.16.34.0 0.0.0.7

R3(config-router)#no auto-summary

A computer screen with white text

Description automatically generated

R4#conf t

R4(config)#router eigrp 1

R4(config-router)#network 192.168.4.0

R4(config-router)#network 172.16.34.0 0.0.0.7

R4(config-router)#no auto-summary

A black screen with white text

Description automatically generated

**Step 4: Verify EIGRP connectivity.**

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

R1# show ip eigrp neighbors

A black background with white text

Description automatically generated

R2# show ip eigrp neighbors

A black background with white text

Description automatically generated

R3# show ip eigrp neighbors

A screen shot of a computer

Description automatically generated

R4# show ip eigrp neighbors

A black background with white text

Description automatically generated

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

R1# tclsh

foreach address {

172.16.12.1

172.16.12.2

172.16.13.1

172.16.13.3

172.16.23.2

172.16.23.3

172.16.34.3

172.16.34.4

192.168.1.1

192.168.2.1

192.168.3.1

192.168.4.1

192.168.4.129

} { ping $address }

A screenshot of a computer program

Description automatically generated

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

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

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

R1# show ip route | begin Gateway

A computer screen shot of a computer code

Description automatically generated

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

R4# traceroute 192.168.1.1 source 192.168.4.1

A screenshot of a computer

Description automatically generated

R4# traceroute 192.168.1.1 source 192.168.4.129

A screenshot of a computer

Description automatically generated

c. On R3, use the show ip route command and note that the preferred route from R3 to R1 LAN 192.168.1.0/24 is via R2 using the R3 exit interface S0/1.

R3# show ip route | begin Gateway

A computer screen shot of a computer code

Description automatically generated

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

R3# show interfaces serial0/0 | include BW

A screenshot of a computer program

Description automatically generated

R3# show interfaces serial0/0 | include BW



R3# show interfaces serial0/1 | include BW



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

R3# show ip eigrp topology 192.168.1.0

A computer screen shot of a black screen

Description automatically generated

**Step 6: Configure PBR to provide path control.**

Now you will deploy source-based IP routing by using PBR. You will change a default IP routing decision based on the EIGRP-acquired routing information for selected IP source-to-destination flows and apply a different next-hop router. Recall that routers normally forward packets to destination addresses based on information in their routing table.

By using PBR, you can implement policies that selectively cause packets to take different paths based on source address, protocol type, or application type. Therefore, PBR overrides the router’s normal routing behavior. Configuring PBR involves configuring a route map with match and set commands and then applying the route map to the interface.

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 --&gt; R2 --&gt; R1 path.
* All traffic sourced from R4 LAN B must take the R3 --&gt; R1 path.

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

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

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

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

R3(config-std-nacl)# exit

A screen shot of a computer

Description automatically generated

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

A black screen with white text

Description automatically generated

c. 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 S3/2

R3(config)# interface s0/2

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

R3(config-if)# end

A black background with white text

Description automatically generated

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

R3# show route-map

A screen shot of a computer program

Description automatically generated

**Step 7: Test the policy.**

Now you are ready to test the policy configured on R3. Enable the debug ip policy command on R3 so that you can observe the policy decision-making in action. To help filter the traffic, first create a standard ACL that identifies all traffic from the R4 LANs.

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

R3# conf t

R3(config)# access-list 1 permit 192.168.4.0 0.0.0.255

R3(config)# exit

A black background with white text

Description automatically generated

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

R3# debug ip policy ?



R3# debug ip policy 1



Policy routing debugging is on for access list 1 m. Test the policy from R4 with the traceroute command, using R4 LAN A as the source network.

R4# traceroute 192.168.1.1 source 192.168.4.1

A screenshot of a computer

Description automatically generated

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

R4# traceroute 192.168.1.1 source 192.168.4.129

A screen shot of a computer

Description automatically generated

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

R3# show route-map

A screen shot of a computer code

Description automatically generated

**Practical No: 06**

**Title**: IP Service Level Agreements & Remote SPAN in a campus environment

**Requirement:** Switch (2960-24TT),Router (2620XM),PCs (PC1& PC2),3 copper straight-through wires,2 PC-PT.

#### **Topology:**

A computer screen shot of a computer

Description automatically generated

**Configuration: 1 Assign IP addresses**

#### 10.0.0.1 for PC0

A screenshot of a computer

Description automatically generated

**20.0.0.1 for PC1**

A screenshot of a computer

Description automatically generated

1. For server side connection.

A screenshot of a computer

Description automatically generated

PC0 is added in the **SERVER VLAN** as shown above.

**Switch#**conf t

**Switch(config)#**interface FastEthernet0/0 **Switch(config-if)#**switchport access vlan 2 **Switch(config-if)#**exit

A screenshot of a computer

Description automatically generated

1. For client side connection.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

PC1 is added in the **CLIENT VLAN** as shown above.

**Switch#**conf t

**Switch(config)#**interface FastEthernet0/1 **Switch(config-if)#**switchport access vlan 3 **Switch(config-if)#**exit

1. Connect a router to switch at FastEthernet 0/3 as shown in the topology diagram To UP the status of fastEthernet 0/0

A close-up of a computer screen

Description automatically generated

**Switch#**conf t

**Switch(config)#**interface fastEthernet 0/3

**Switch(config-if)#**switchport mode trunk

1. Open Router CLI to check the status of the fastEthernet 0/0 port

**Router>**en

**Router#**show ip interface brief

A. To create sub interfaces on Router

A screenshot of a computer

Description automatically generated

To create sub interfaces(0.1) on Router

**Router#**conf t

**Router(config)#**interface fastEthernet 0/0 **Router(config-if)#**no shutdown **Router#**conf t

**Router(config)#**interface fastEthernet 0/0.1

**Router(config-subif)#**encapsulation dot1Q 2

**Router(config-subif)#**ip address 10.0.0.100 255.0.0.0

#### Router(config-subif)#exit

A screenshot of a computer program

Description automatically generated

To create sub interfaces(0.2) on Router

**Router#**conf t

**Router(config)#**interface fastEthernet 0/0.2

**Router(config-subif)#**encapsulation dot1Q 3

**Router(config-subif)#**ip address 20.0.0.100 255.0.0.0

#### Router(config-subif)#^z Router(config-subif)#exit

A screenshot of a computer

Description automatically generated

To provide route for communicating between PCs and Router.

**Router#**conf t

**Router(config)#**router rip

**Router(config-router)#**network 10.0.0.0

**Router(config-router)#**network 20.0.0.0

1. Assign Default Gateway to PC0 10.0.0.100 & Default Gateway to PC1 10.0.0.100.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

#### Output:

Open Command Prompt of PC0 and check the ip addresses by the command ipconfig.

Now using the ping command check if the connection is correct and the messages are sent properly without any error.

A computer screen shot of a computer program

Description automatically generated

**Commands: C:\>**ipconfig **C:\>ping** 20.0.0.1 **Repeat same on PC1**

A computer screen shot of a computer program

Description automatically generated

C:\>ipconfig

C:\>ping 10.0.0.1

**Practical No: 07**

**Title**: Inter-VLAN Routing

Topology :

A diagram of a computer network

Description automatically generated

Step 1:

Switch# configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
Switch(config)#interface fastethernet 0/1  
Switch(config-if)#switchport access vlan 10  
Switch(config-if)#switchport mode access  
Switch(config-if)#exit  
Switch(config)#  
Switch(config)#interface fastethernet 0/2  
Switch(config-if)#switchport access vlan 20  
Switch(config-if)#switchport mode access  
Switch(config-if)#exit  
Switch(config)#  
Switch(config)#interface fastethernet 0/3  
Switch(config-if)#switchport access vlan 10  
Switch(config-if)#switchport mode access  
Switch(config-if)#end   
Switch#  
%SYS-5-CONFIG\_I: Configured from console by console

A screenshot of a computer program

Description automatically generated

Step 2:

Switch#config t  
Enter configuration commands, one per line. End with CNTL/Z.  
Switch(config)#interface gigabitethernet 0/1  
Switch(config-if)#no shutdown  
Switch(config-if)#  
Switch(config-if)#switchport mode trunk  
Switch(config-if)#exit  
Switch(config)#exit  
Switch#  
%SYS-5-CONFIG\_I: Configured from console by console  
Switch#show run

A white background with black text

Description automatically generated

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated

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

Router>enable

Router#conf t

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

Router(config)#interface gigabitEthernet 0/0

Router(config-if)#no shutdown

Router(config-if)#

%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up  
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up  
Router(config-if)#exit  
Router(config)#interface gigabitEthernet 0/0.10  
Router(config-subif)#

%LINK-5-CHANGED: Interface GigabitEthernet0/0.10, changed state to up  
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.10, changed state to up  
Router(config-subif)#encapsulation dot1Q 10  
Router(config-subif)#ip address 192.168.10.1 255.255.255.0  
Router(config-subif)#exit  
Router(config)#interface gigabitEthernet 0/0.20  
Router(config-subif)#  
%LINK-5-CHANGED: Interface GigabitEthernet0/0.20, changed state to up  
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.20, changed state to up  
Router(config-subif)#encapsulation dot1Q 20  
Router(config-subif)#ip address 192.168.20.1 255.255.255.0  
Router(config-subif)#exit  
Router(config)#interface gigabitEthernet 0/0.30  
Router(config-subif)#  
%LINK-5-CHANGED: Interface GigabitEthernet0/0.30, changed state to up  
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.30, changed state to up  
Router(config-subif)#encapsulation dot1Q 30  
Router(config-subif)#ip address 192.168.30.1 255.255.255.0  
Router(config-subif)#  
Router(config-subif)#encapsulation dot1Q 10  
%Configuration of multiple subinterfaces of the same main  
interface with the same VID (10) is not permitted.  
This VID is already configured on GigabitEthernet0/0.10.  
Router(config-subif)#shutdown  
Router(config-subif)#  
%LINK-5-CHANGED: Interface GigabitEthernet0/0.30, changed state to administratively down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.30, changed state to down  
Router(config-subif)#  
Router(config-subif)#exit  
Router(config)#no interface gigabitethernet 0/0.30  
Router(config)#end  
Router#  
%SYS-5-CONFIG\_I: Configured from console by console  
Router#show run

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

Router#show ip route

A white text with black text

Description automatically generated

Switch#show vlan brief

A screenshot of a computer

Description automatically generated

Switch#conf t

Enter configuration commands, one per line. End with CNTL/Z.  
Switch(config)#vlan 30  
Switch(config-vlan)#exit  
Switch(config)#  
Switch(config)#interface fastethernet 0/3  
Switch(config-if)#switchport access vlan 30  
Switch(config-if)#switchport mode access  
Switch(config-if)#exit  
Switch(config)#exit  
Switch#  
Switch#show vlan brief

A white screen with black text

Description automatically generated

Switch#show vlan brief

A screenshot of a computer

Description automatically generated

Router#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
Router(config)#interface gigabitethernet 0/0.30  
Router(config-subif)#  
Router(config-subif)#encapsulation dot1Q 30  
Router(config-subif)#ip address 192.168.30.1 255.255.255.0  
Router(config-subif)#exit  
Router(config)#exit  
Router#  
%SYS-5-CONFIG\_I: Configured from console by console  
Router#show ip route

A screenshot of a computer program

Description automatically generated

Switch#copy running-config startup-config

A computer screen shot of a program

Description automatically generated

**Practical No: 08**

**Title**: Simulating MPLS Environment

**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

A diagram of a network

Description automatically generated

**Router R1**

R1#conf t

R1(config)#int lo0

R1(config-if)#ip add 1.1.1.1 255.255.255.255

R1(config-if)#ip ospf 1 area 0

R1(config-if)#int f0/0

R1(config-if)#ip add 10.0.0.1 255.255.255.0

R1(config-if)#no shut

R1(config-if)#ip ospf 1 area 0

A computer screen with white text

Description automatically generated

**Router R2**

R2#CONF T

R2(config)#int lo0

R2(config-if)#ip add 2.2.2.2 255.255.255.255

R2(config-if)#ip ospf 1 area 0

R2(config-if)#

R2(config-if)#int f0/0

R2(config-if)#ip add 10.0.0.2 255.255.255.0

R2(config-if)#no shut

R2(config-if)#ip ospf 1 area 0

R2(config)#int f0/1

R2(config-if)#ip add 10.0.1.2 255.255.255.0

R2(config-if)#no shut

ip ospf 1 area 0

A screenshot of a computer program

Description automatically generated

Router R3

R3(config)#CONF T

R3(config)#int lo0

R3(config-if)#ip add 3.3.3.3 255.255.255.255

R3(config-if)#ip ospf 1 area 0

R3(config-if)#

R3(config-if)#int f0/0

R3(config-if)#ip add 10.0.1.3 255.255.255.0

R3(config-if)#no shut

R3(config-if)#ip ospf 1 area 0

A computer screen with text and numbers

Description automatically generated

R1#ping 3.3.3.3 source lo0

A screen shot of a computer

Description automatically generated

**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

**Router R1**

R1#conf t

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

R1(config)#router ospf 1

R1(config-router)#mpls ldp autoconfig

A black screen with white text

Description automatically generated

**Router R2**

R2(config)#router ospf 1

R2(config-router)#mpls ldp autoconfig

A black background with white text

Description automatically generated

**Router R3**

R3(config)#router ospf 1

R3(config-router)#mpls ldp autoconfig

A black screen with white text

Description automatically generated

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

A black background with white text

Description automatically generated

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

A black screen with white text

Description automatically generated

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

R2#sh mpls ldp neigh

A screenshot of a computer screen

Description automatically generated

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. R1#trace 3.3.3.3A screenshot of a computer

Description automatically generated

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. 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 lif

**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(config)#router bgp 1

R1(config-router)#neighbor 3.3.3.3 remote-as 1

R1(config-router)#neighbor 3.3.3.3 update-source Loopback0

R1(config-router)#no auto-summary

R1(config-router)#address-family vpnv4

R1(config-router-af)#neighbor 3.3.3.3 activate

A screenshot of a computer program

Description automatically generated

R3(config)#router bgp 1

R3(config-router)#neighbor 1.1.1.1 remote-as 1

R3(config-router)#neighbor 1.1.1.1 update-source Loopback0

R3(config-router)#no auto-summary

R3(config-router)#address-family vpnv4

R3(config-router-af)#neighbor 1.1.1.1 activate

A screenshot of a computer program

Description automatically generated

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

R1#sh bgp vpnv4 unicast all summary

A black screen with white text

Description automatically generated

You can see here that we do have a bgp vpnv4 peering to R3 – looking at the PfxRcd you can see it says 0 this is because we have not got any routes in BGP. We are now going to add two more routers to the topology. These will be the customer sites connected to R1 and R3. We will then create a VRF on each router and put the interfaces connected to each site router into that VRF

**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

Router 4 will peer OSPF using process number 2 to a VRF configured on R1. It will use the local site addressing of 192.168.1.0/24.

A diagram of a network

Description automatically generated

R4#conf t

R4(config)#int lo0

R4(config-if)#ip add 4.4.4.4 255.255.255.255

R4(config-if)#ip ospf 2 area 2

R4(config-if)#int f0/0

R4(config-if)#ip add 192.168.1.4 255.255.255.0

R4(config-if)#ip ospf 2 area 2

R4(config-if)#no shut

A screenshot of a computer program

Description automatically generated

R1#conf t

R1(config)#int f0/1

R1(config-if)#no shut

R1(config-if)#ip add 192.168.1.1 255.255.255.0

A screen shot of a computer

Description automatically generated

R1(config)#ip vrf RED

R1(config-vrf)#rd 4:4

R1(config-vrf)#route-target both 4:4

A screenshot of a computer

Description automatically generated

R1(config)#int f0/1

R1(config-if)#ip vrf forwarding RED

R1(config-if)#ip vrf fo

R1(config-if)#ip vrf forwarding RED



R1(config)#int f0/1

R1(config-if)#ip address 192.168.1.1 255.255.255.0

A black background with white numbers

Description automatically generated

R1#sh run int f0/1

A screenshot of a computer

Description automatically generated

R1#sh ip route

A computer screen shot of a black screen

Description automatically generated

R1#sh ip route vrf red

A black screen with white text

Description automatically generated

R1#sh ip route vrf RED

A screenshot of a computer program

Description automatically generated

R1(config)#int f0/1

R1(config-if)#ip ospf 2 area 2

A black screen with white text

Description automatically generated

R1#sh ip route vrf RED

A screenshot of a computer program

Description automatically generated

R5(config)#int lo0

R5(config-if)#ip add 6.6.6.6 255.255.255.255

R5(config-if)#ip ospf 2 area 2

R5(config-if)#int f0/0

R5(config-if)#ip add 192.168.2.6 255.255.255.0

R5(config-if)#ip ospf 2 area 2

R5(config-if)#no shut

A computer screen with white text

Description automatically generated

R3(config)#int f0/1

R3(config-if)#no shut

R3(config-if)#ip add 192.168.2.3 255.255.255.0A black screen with white text

Description automatically generated

R3(config)#ip vrf RED

R3(config-vrf)#rd 4:4

R3(config-vrf)#route-target both 4:4A screen shot of a computer

Description automatically generated

R3(config)#int f0/1

R3(config-if)#ip vrf forwarding RED

A black background with white numbers

Description automatically generated

R3(config)#int f0/1

R3(config-if)#ip address 192.168.2.1 255.255.255.0

A black background with white numbers

Description automatically generated

R3#sh run int f0/1

A screenshot of a computer

Description automatically generated

R3(config)#int f0/1

R3(config-if)#ip ospf 2 area 2A black background with white text

Description automatically generated

R3#sh ip route vrf RED

A screen shot of a computer

Description automatically generated

R4#sh ip route

A screenshot of a computer screen

Description automatically generated

R1#sh ip route

A screenshot of a computer program

Description automatically generated

R1#sh ip route vrf RED

A screenshot of a computer screen

Description automatically generated

R1(config)#router bgp 1

R1(config-router)#address-family ipv4 vrf RED

R1(config-router-af)#redistribute ospf 2

A black screen with white text

Description automatically generated

R3(config)#router bgp 1

R3(config-router)#address-family ipv4 vrf RED

R3(config-router-af)#redistribute ospf 2

A black background with white text

Description automatically generated

R1#sh ip bgp vpnv4 vrf RED

A screenshot of a computer

Description automatically generated

R3#sh ip bgp vpnv4 vrf RED

A screenshot of a computer program

Description automatically generated

R1(config)#router ospf 2

R1(config-router)#redistribute bgp 1 subnets

A black screen with white text

Description automatically generated

R3(config)#router ospf 2

R3(config-router)#redistribute bgp 1 subnets

A black screen with white text

Description automatically generated

R4#sh ip route

A screen shot of a computer

Description automatically generated

R5#sh ip route

A screenshot of a computer screen

Description automatically generated

R4#ping 6.6.6.6

A screen shot of a computer

Description automatically generated

R4#trace 6.6.6.6

A screenshot of a computer

Description automatically generated