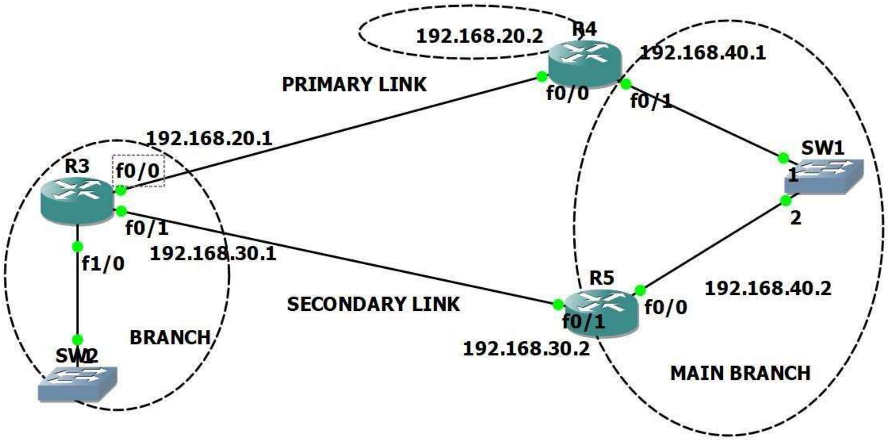
INDEX

SR No.	Practical Name	Date	Remark
1	Configure IP SLA Tracking and Path Control Topology		
2	Using the AS_PATH Attribute		
3	Configuring IBGP and EBGP Sessions, Local Preference, and MED		
4	Secure the Management Plane		
5	Configure and Verify Path Control Using PBR		

Practical No.1

- ❖ Aim: Configure IP SLA Tracking and Path Control Topology



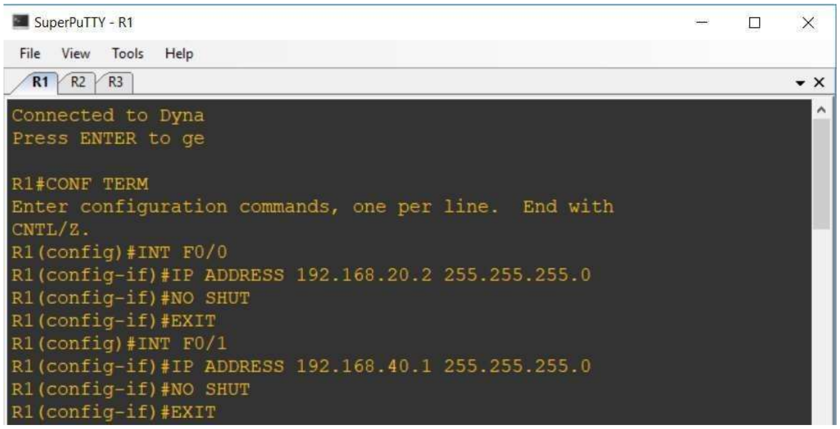
In the topology branch site connected to main site using two link primary and secondary all the data go through primary link. If the primary link fail then router automatically change the link i.e secondary link. R3 send the icmp echo messages to send packet to R1.

If the R3 will not receive icmp echo response then it will take secondary link on R3 configure.

IP ADDRESS CONFIGURATION –

R1 Commands –

```
R1#CONF TERM
R1(config)#INT F0/0
R1(config-if)#IP ADDRESS 192.168.20.2 255.255.255.0
R1(config-if)#NO SHUT
R1(config-if)#EXIT
R1(config)#INT F0/1
R1(config-if)#IP ADDRESS 192.168.40.1 255.255.255.0
R1(config-if)#NO SHUT
R1(config-if)#EXIT
```



```
Connected to Dyna
Press ENTER to ge

R1#CONF TERM
Enter configuration commands, one per line.  End with
CNTL/Z.
R1(config)#INT F0/0
R1(config-if)#IP ADDRESS 192.168.20.2 255.255.255.0
R1(config-if)#NO SHUT
R1(config-if)#EXIT
R1(config)#INT F0/1
R1(config-if)#IP ADDRESS 192.168.40.1 255.255.255.0
R1(config-if)#NO SHUT
R1(config-if)#EXIT
```

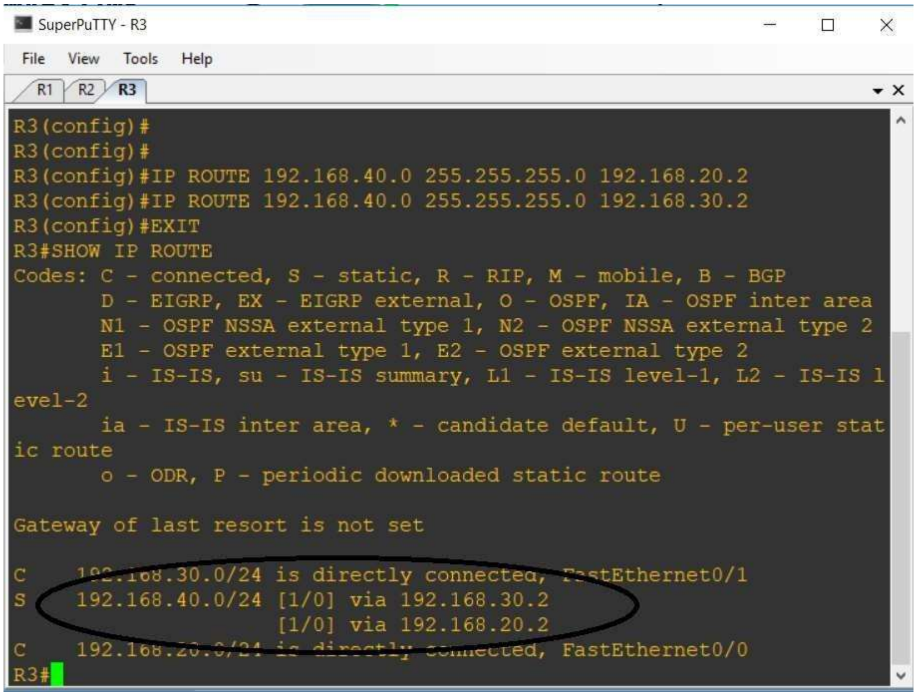
R2 COMMANDS –

```
R2#CONF TERM
```

R2(config)#INT F0/1
R2(config-if)#IP ADDRESS 192.168.30.2 255.255.255.0
R2(config-if)#NO SHUT
R2(config-if)#EXIT
R2(config)#INT F0/0
R2(config-if)#IP ADDRESS 192.168.40.2 255.255.255.0
R2(config-if)#NO SHUT
R2(config-if)#EXIT

R3 COMMANDS –
R3#CONF TERM
R3(config)#INT F0/0
R3(config-if)#IP ADDRESS 192.168.20.1 255.255.255.0
R3(config-if)#NO SHUT
R3(config-if)#EXIT
R3(config)#INT F0/1
R3(config-if)#IP ADDRESS 192.168.30.1 255.255.255.0
R3(config-if)#NO SHUT
R3(config-if)#EXIT

STATIC ROUTING –
R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.20.2
R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.30.2
R3(config)#EXIT
R3#SHOW IP ROUTE



```
R3(config)#
R3(config)#
R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.20.2
R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.30.2
R3(config)#EXIT
R3#SHOW IP ROUTE
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS l
evel-2
       ia - IS-IS inter area, * - candidate default, U - per-user stat
ic route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.30.0/24 is directly connected, FastEthernet0/1
S    192.168.40.0/24 [1/0] via 192.168.30.2
                     [1/0] via 192.168.20.2
C    192.168.20.0/24 is directly connected, FastEthernet0/0
R3#
```

IP SLA Commands –
R3#CONF TERM
R3(config)#IP SLA 1
R3(config-ip-sla)#ICMP-ECHO 192.168.20.2 SOURCE-INTERFACE F0/0
R3(config-ip-sla-echo)#FREQUENCY 10

R3(config-ip-sla-echo)#TIMEOUT 6000

R3(config-ip-sla-echo)#EXIT

R3(config)#IP SLA SCHEDULE 1 START-TIME NOW LIFE FOREVER

The operation number of 1 is only locally significant to the router. The frequency 10 command schedules the connectivity test to repeat every 10 seconds. The probe is scheduled to start now and to run forever. Verify the IP SLAs configuration of operation 1 using the show ip sla configuration 1 command.

R1# show ip sla configuration 1

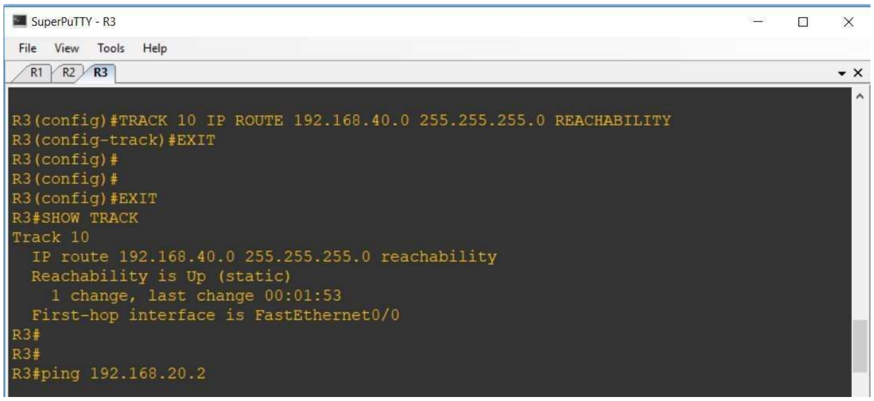
Configure tracking options.

Although PBR could be used, you will configure a floating static route that appears or disappears depending on the success or failure of the IP SLA.

R3(config)#TRACK 10 IP ROUTE 192.168.40.0 255.255.255.0 REACHABILITY R3(config-track)#EXIT

Use the show track command to check the track configuration. you will get reachability up output in this command –

```
R3#SHOW TRACK
Track 10
  IP route 192.168.40.0 255.255.255.0 reachability
  Reachability is Up (static)
    1 change, last change 00:01:53
  First-hop interface is FastEthernet0/0
R3#
```



```
R3(config)#TRACK 10 IP ROUTE 192.168.40.0 255.255.255.0 REACHABILITY
R3(config-track)#EXIT
R3(config)#
R3(config)#
R3(config)#EXIT
R3#SHOW TRACK
Track 10
  IP route 192.168.40.0 255.255.255.0 reachability
  Reachability is Up (static)
    1 change, last change 00:01:53
  First-hop interface is FastEthernet0/0
R3#
R3#
R3#ping 192.168.20.2
```

Again configure the static route for ip sla with your track configuration.

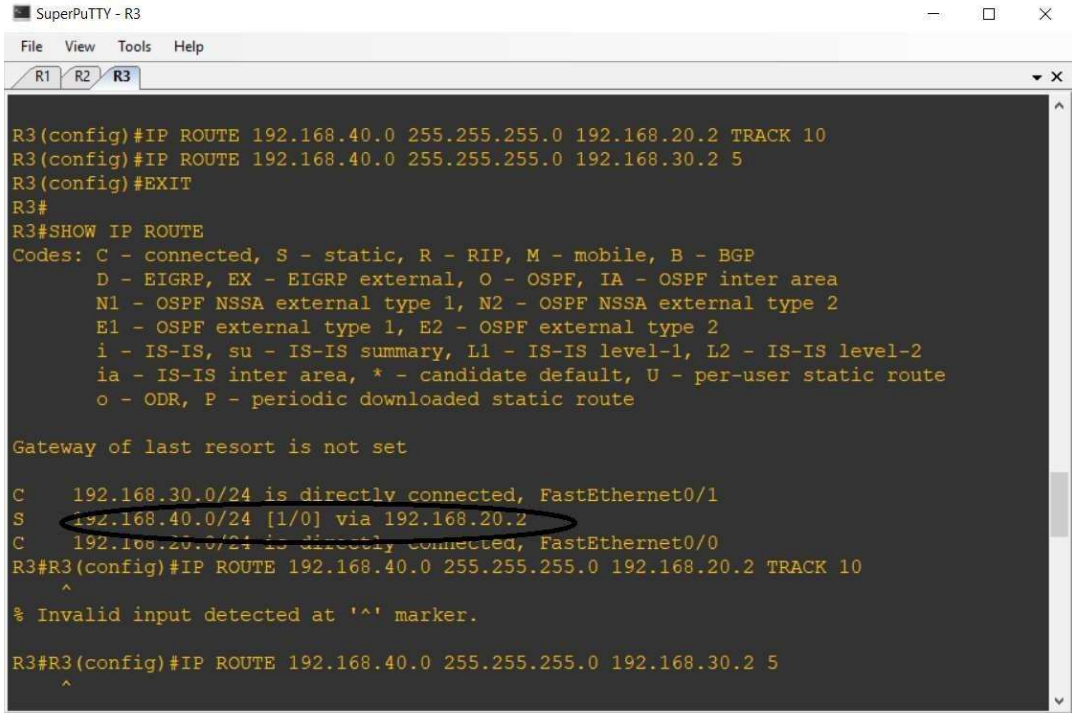
R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.20.2 TRACK 10

R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.30.2 5

R3(config)#EXIT

R3#

R3#SHOW IP ROUTE

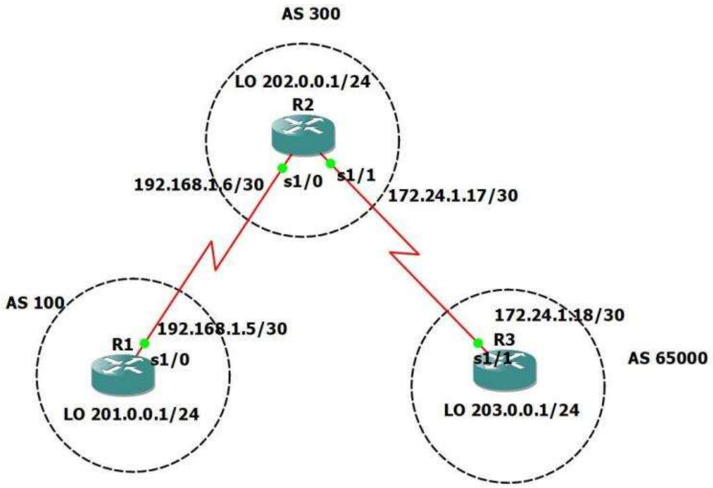


```
R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.20.2 TRACK 10
R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.30.2 5
R3(config)#EXIT
R3#
R3#SHOW IP ROUTE
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.30.0/24 is directly connected, FastEthernet0/1
S    192.168.40.0/24 [1/0] via 192.168.20.2
C    192.168.20.0/24 is directly connected, FastEthernet0/0
R3#R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.20.2 TRACK 10
    ^
% Invalid input detected at '^' marker.

R3#R3(config)#IP ROUTE 192.168.40.0 255.255.255.0 192.168.30.2 5
    ^
```

In this final configuration we will get only one link to reach to network 192.168.40.0 as per ip sla created on network.

Practical No.2

Aim: Using the AS_PATH Attribute –



In this lab, the student will use BGP commands to prevent private AS numbers from being advertised to the outside world. The student will also use the AS_PATH attribute to filter BGP routes based on their source AS numbers.

Scenario

The International Travel Agency's Internet service provider ISP2 has been assigned an AS number of 300. This provider uses BGP to exchange routing information with several customer networks. Each customer network is assigned an AS number from the private range, such as AS 65000. Configure ISP2 to remove the private AS numbers within the AS_Path information from the CusRtr. In addition, Provider ISP2 would like to prevent its customer networks from receiving route information from International Travel Agency's AS 100. Use the AS_PATH attribute to implement this policy.

IP ADDRESS CONFIGURATION –

R1 Commands –

```
R1#Conf Term
R1(config)#INT S1/0
R1(config-if)#NO IP ADDRESS
R1(config-if)#IP ADDRESS 192.168.1.5 255.255.255.252
R1(config-if)#NO SHUT
R1(config-if)#EXIT
R1(config)#INT LO0
R1(config-if)#IP ADDRESS 201.0.0.1 255.255.255.0
R1(config-if)#NO SHUT
R1(config-if)#EXIT
```

R2 COMMANDS –

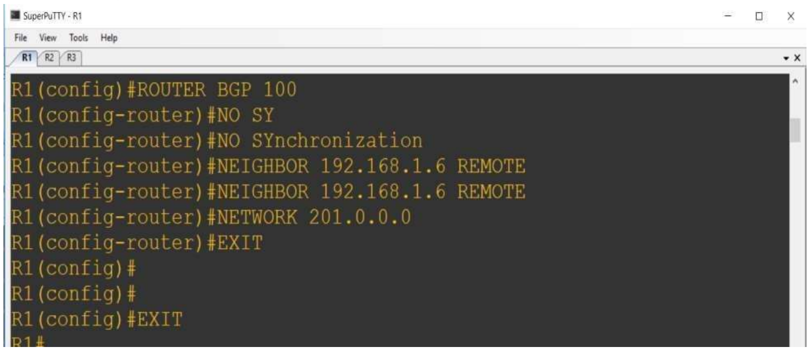
```
R2#CONF TERM
R2(config)#INT S1/0
R2(config-if)#IP ADDRESS 192.168.1.6 255.255.255.252
R2(config-if)#NO SHUT
R2(config-if)#EXIT
```

R2(config)#INT S1/1
R2(config-if)#IP ADDRESS 172.24.1.17 255.255.255.252
R2(config-if)#NO SHUT
R2(config-if)#EXIT
R2(config)#INT LO0
R2(config-if)#IP ADDRESS 202.0.0.1 255.255.255.0
R2(config-if)#NO SHUT
R2(config-if)#EXIT

R3 COMMANDS –
R3#CONF TERM
R3(config)#INT S1/1
R3(config-if)#IP ADDRESS 172.24.1.18 255.255.255.252
R3(config-if)#NO SHUT
R3(config-if)#EXIT
R3(config)#
R3(config)#INT LO0
R3(config-if)#IP ADDRESS 203.0.0.1 255.255.255.0
R3(config-if)#NO SHUT
R3(config-if)#EXIT

Build and configure the network according to the diagram, but do not configure a routing protocol. Use ping to test connectivity between the directly connected routers.
Configure BGP for normal operation. Enter the appropriate BGP commands on each router so that they will identify their BGP neighbors and advertise their Ethernet networks –

R1 (config)#router bgp 100
R1 (config-router)#no synchronization
R1 (config-router)#neighbor 192.168.1.6 remote-as 300
R1 (config-router)#network 201.0.0.0



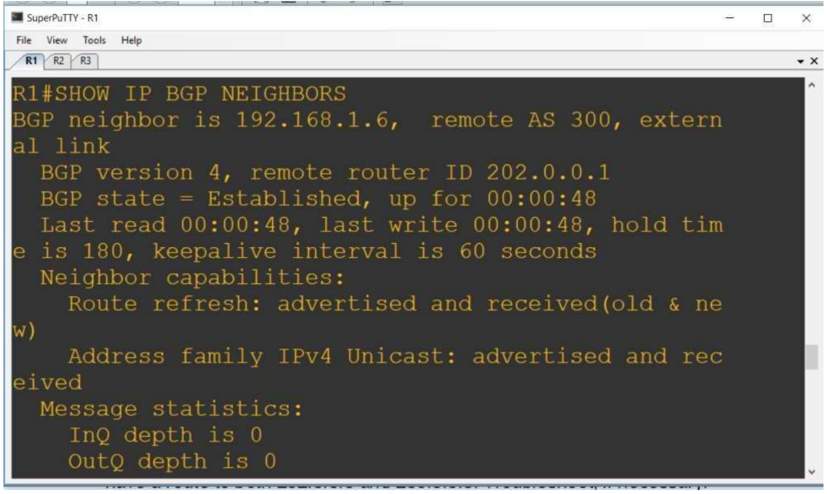
```
R1(config)#ROUTER BGP 100
R1(config-router)#NO SY
R1(config-router)#NO SYnchronization
R1(config-router)#NEIGHBOR 192.168.1.6 REMOTE
R1(config-router)#NEIGHBOR 192.168.1.6 REMOTE
R1(config-router)#NETWORK 201.0.0.0
R1(config-router)#EXIT
R1(config)#
R1(config)#
R1(config)#EXIT
R1#
```

R2 (config)#router bgp 300
R2(config-router)#no synchronization
R2 (config-router)#neighbor 192.168.1.5 remote-as 100
R2 (config-router)#neighbor 172.24.1.18 remote-as 65000
R2 (config-router)#network 202.0.0.0

R3 (config)#router bgp 65000

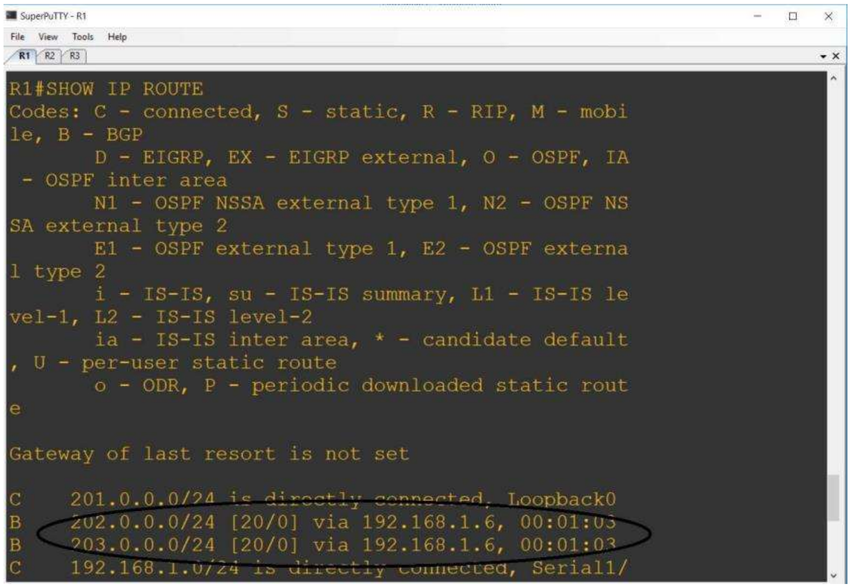
```
R3 (config-router)#no synchronization
R3 (config-router)#neighbor 172.24.1.17 remote-as 300
R3 (config-router)#network 203.0.0.0
```

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



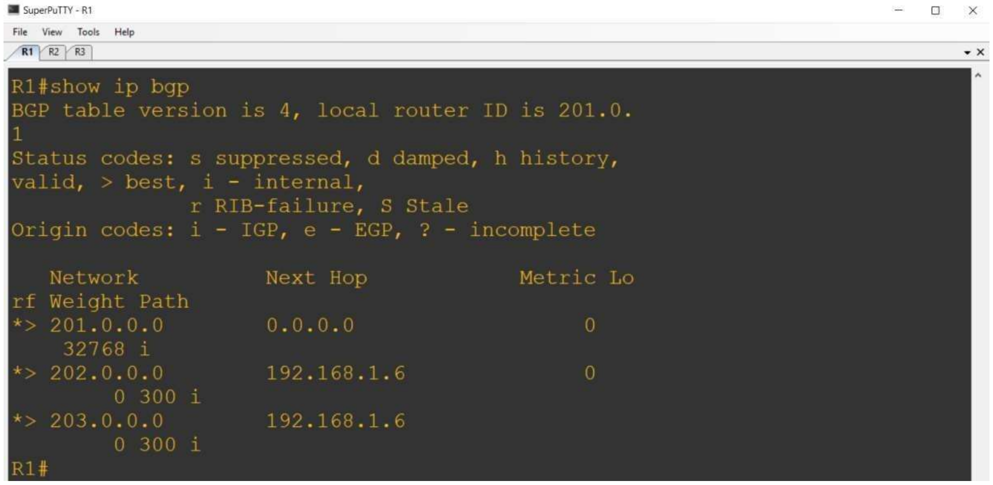
```
R1#SHOW IP BGP NEIGHBORS
BGP neighbor is 192.168.1.6,  remote AS 300, extern
al link
  BGP version 4, remote router ID 202.0.0.1
  BGP state = Established, up for 00:00:48
  Last read 00:00:48, last write 00:00:48, hold tim
e is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received(old & ne
w)
    Address family IPv4 Unicast: advertised and rec
eived
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
```

Check the routing table from R1 by using the show ip route command. R1 should have a route to both 202.0.0.0 and 203.0.0.0.



```
R1#SHOW IP ROUTE
Codes: C - connected, S - static, R - RIP, M - mobi
le, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA
 - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NS
SA external type 2
       E1 - OSPF external type 1, E2 - OSPF externa
l type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS le
vel-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default
, U - per-user static route
       o - ODR, P - periodic downloaded static rout
e

Gateway of last resort is not set

C    201.0.0.0/24 is directly connected, Loopback0
B    202.0.0.0/24 [20/0] via 192.168.1.6, 00:01:03
B    203.0.0.0/24 [20/0] via 192.168.1.6, 00:01:03
C    192.168.1.0/24 is directly connected, Serial1/
```

Check the BGP table from R1 by using the show ip bgp command. Note the AS path for the 203.0.0.0 network. The AS 65000 should be listed in the path to 203.0.0.0.

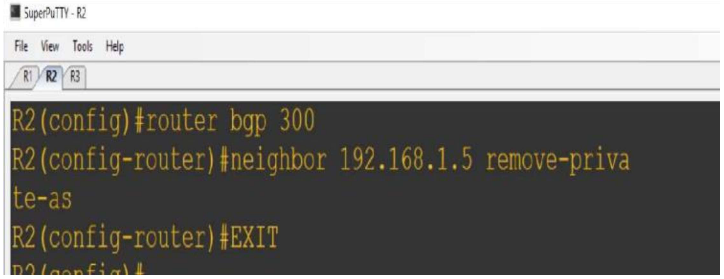


```
R1#show ip bgp
BGP table version is 4, local router ID is 201.0.
1
Status codes: s suppressed, d damped, h history,
valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric Lo
rf Weight Path
*> 201.0.0.0        0.0.0.0                  0
    32768 i
*> 202.0.0.0        192.168.1.6              0
        0 300 i
*> 203.0.0.0        192.168.1.6
        0 300 i
R1#
```

Configure R2 to strip the private AS numbers from BGP routes exchanged with R1. Use the following commands:

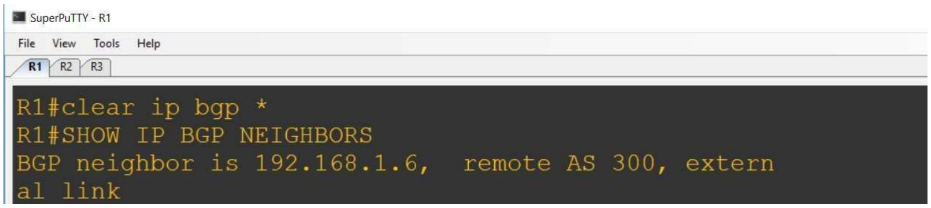
R2(config)#router bgp 300

R2(config-router)#neighbor 192.168.1.5 remove-private-as



```
R2(config)#router bgp 300
R2(config-router)#neighbor 192.168.1.5 remove-priva
te-as
R2(config-router)#EXIT
R2(config)#
```

After issuing these commands, use the clear ip bgp * command on R1 to re-establish the BGP relationships between the three routers.



```
R1#clear ip bgp *
R1#SHOW IP BGP NEIGHBORS
BGP neighbor is 192.168.1.6,  remote AS 300, extern
al link
```

Wait several seconds, and then return to R1 to check its routing table.

R1 should be able to ping 203.0.0.0.

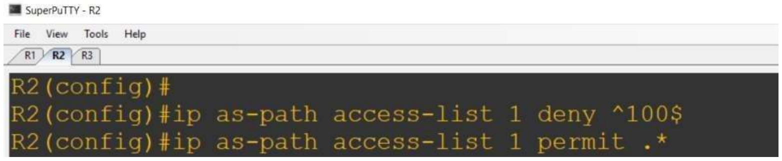
Now check the BGP table on R1. The AS_PATH to the 203.0.0.0 network should be AS 300.

As a final configuration, use the AS_PATH attribute to filter routes based on their origin. In a complex environment, this attribute can be used to enforce routing policy. In this case, the provider router R2, must be configured so that it does not propagate routes that originate from AS 100 to the customer router, R3.

First, 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 R2:

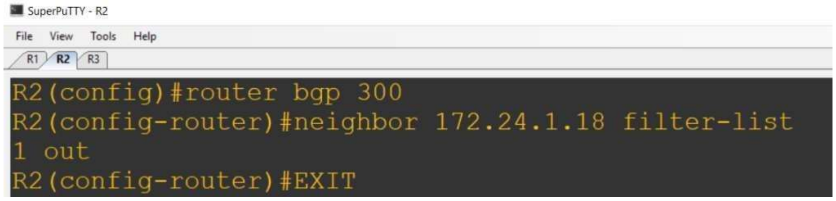
R2(config)#ip as-path access-list 1 deny ^100$

R2(config)#ip as-path access-list 1 permit .*



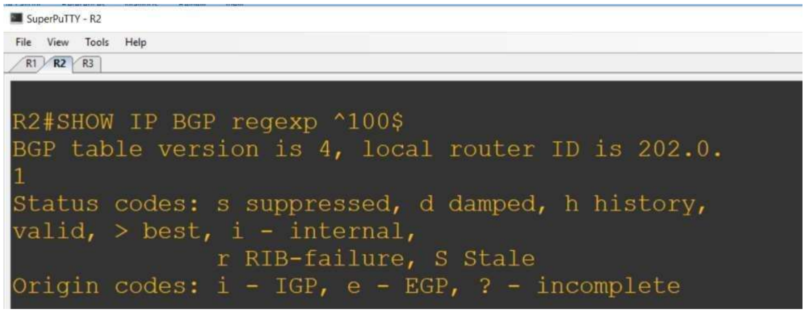
```
R2(config)#
R2(config)#ip as-path access-list 1 deny ^100$
R2(config)#ip as-path access-list 1 permit .*
```

Now that the access list has been configured, apply it as follows:
R2(config)#router bgp 300
R2(config-router)#neighbor 172.24.1.18 filter-list 1 out



```
R2(config)#router bgp 300
R2(config-router)#neighbor 172.24.1.18 filter-list
1 out
R2(config-router)#EXIT
```

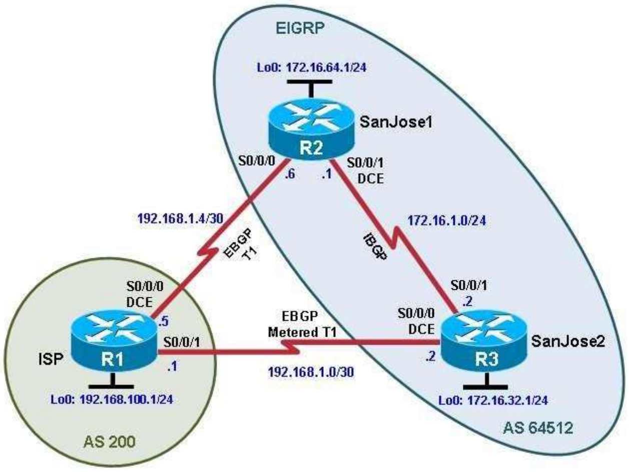
The out keyword specifies that the list should be applied to routing information sent to this neighbor. Use the clear ip bgp * command to reset the routing information. Wait several seconds, and then check the routing table for R2. The route to 201.0.0.0 should be in the routing table. Check the routing table for R3. It should not have a route to 201.0.0.0 in its routing table. Return to R2 and verify that the filter is working as intended. Issue the command show ip bgp regexp ^100$.
The output of this command shows all matches for the regular expressions that were used in the access list. The path to 201.0.0.0 matches the access list and is filtered out of updates to R3.



```
R2#SHOW IP BGP regexp ^100$
BGP table version is 4, local router ID is 202.0.
1
Status codes: s suppressed, d damped, h history,
valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Practical No. 3

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



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.

Required Resources

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

Step 0: Suggested starting configurations.

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

Router(config)# no ip domain-lookup Router(config)# line con 0
Router(config-line)# logging synchronous Router(config-line)# exec-timeout 0 0 Step 1: Configure interface addresses.

a. 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#
```

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#
```

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#
```

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

Step 2: Configure EIGRP.

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

```
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.

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

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

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.

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

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

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

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

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.

a. 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
```

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.

b. 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
```

c. 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
```

d. 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
```

e. 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
```

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#
```

Step 6: Verify which path the traffic takes.

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

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

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

```
ISP# ping 172.16.64.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.64.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
ISP#
ISP# ping 172.16.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
ISP#
```

h. Now ping from ISP to the loopback 0 address of 172.16.32.1 on 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
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#
```

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?

____

____

i. 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 backuppath, f RT-Filter,
          x best-external, a additional-path, c RIBcompressed,
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


ISP#
i

ISP# show ip bgp
```

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).

j. 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
```

```
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
```

```
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#
```

```
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
```

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.

a. 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
```

b. 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 backuppath, f RT-Filter,
            x best-external, a additional-path, c RIBcompressed,
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


 *>  192.168.1.4/30   0.0.0.0                 0        32768


 *>  192.168.100.0    0.0.0.0                 0        32768


i i i
```

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

```
SanJose2# 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 - peruser 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
```

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 nexthop-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.

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

```
ISP(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)#
```

e. 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 backuppath, f RT-Filter,
           x best-external, a additional-path, c RIBcompressed,
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# 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 - peruser 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
```

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.

f. 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

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

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

```
SanJose1# clear ip bgp *

SanJose2# clear ip bgp *
```

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

```
SanJose2# 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 backuppath, f RT-Filter,
              x best-external, a additional-path, c RIBcompressed,
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
```

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

```
SanJose2# 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 - peruser 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
```

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

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

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 - peruser 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
```

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.

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

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

SanJose2(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
```

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

```
SanJose1# 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 backuppath, f RT-Filter,
              x best-external, a additional-path, c RIBcompressed,
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
```

```
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 backuppath, f RT-Filter,
          x best-external, a additional-path, c RIBcompressed,
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
```

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.

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

```
ISP# 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 backuppath, f RT-Filter,
          x best-external, a additional-path, c RIBcompressed,
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
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 - peruser 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#
```

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?

____

____

____

a. 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
   (172.16.32.1) <*>           Total option bytes= 40, padded length=40
   (0.0.0.0)                   Record route:
   (0.0.0.0)                   (172.16.1.2)
   (0.0.0.0)                   (192.168.1.6)
   (0.0.0.0)
 End of list
     (192.168.1001)
     (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)
```

```
Reply to request 2 (20 ms).  Received packet has options  Total option bytes= 40,
padded length=40  Record route:
      (172.16.1.2)


  (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)


  (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
```

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

1. A ping that is sourced from 172.16.32.1 exits SanJose2 through s0/0/1, 172.16.1.2. It then 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.

b. 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

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
```

c. 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 Status codes: s suppressed, d
damped, h history, * valid, > best, i - internal,
           r RIB-failure, S Stale, m multipath, b backuppath, f RT-Filter,
           x best-external, a additional-path, c RIBcompressed,
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

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 backuppath, f RT-Filter,
           x best-external, a additional-path, c RIBcompressed,
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
```

d. 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]:
  (172.16.1.1)                     Data pattern [0xABCD]:
  (172.16.32.1) <*>                Loose, Strict, Record, Timestamp, Verbose[none]: record
  (0.0.0.0)                        Number of hops [ 9 ]:
  (0.0.0.0)                        Loose, Strict, Record, Timestamp,
  (0.0.0.0)                        Verbose[RV]:  Sweep range of sizes [n]:
 End of list
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=40 168.100.1)
 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)
Reply to request 1 (28 ms).  Received packet has options  Total option bytes= 40,
padded length=40  Record route:
  (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)

  (172.16.1.1)        5)
  (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)

  (172.16.1.1)        5)
  (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)

  (172.16.1.1)        5)
  (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
```

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 backuppath, f RT-Filter,
           x best-external, a additional-path, c RIBcompressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found


 *              192.168.1.2         75          0
64512 i
 *>  192.168.100.0    0.0.0.0             0       32768


ISP#
    Network         Next Hop          Metric LocPrf Weight
Path
 *>  172.16.0.0       192.168.1.6          50          0
64512 i
```

Step 10: Establish a default route.

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

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

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

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

```
SanJose1# show ip route
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 - peruser 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#

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 - peruser 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#
```

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

```
SanJose2# show ip bgp
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 backuppath, f RT-Filter,
              x best-external, a additional-path, c RIBcompressed,
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

 *>                0.0.0.0                 0         32768

 *>i 192.168.100.0    172.16.64.1              0    150      0
200 i

 *                192.168.1.1              0    125      0
200 i

SanJose2#
*>i 0.0.0.0          172.16.64.1              0    150      0 200 i
 *                192.168.1.1                  125      0 200 i
i i
```

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

```
SanJose2# traceroute 10.0.0.1 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#
```

e. 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)#
```

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

```
SanJose1# show ip route
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 - peruser 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#

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 - peruser 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#
```

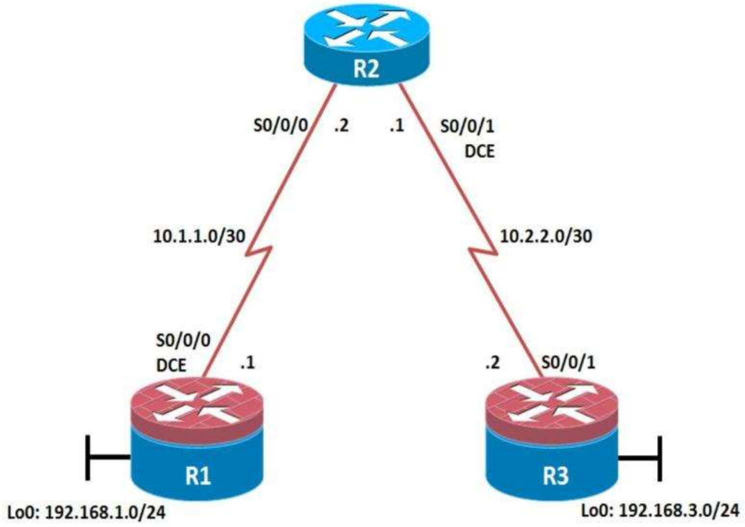
g. Verify the new path using the traceroute command to 10.0.0.1 from SanJose1. Notice the default route is now through SanJose2.

```
SanJose1# 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#
```

Practical No.4

Aim: Secure the Management Plane

Topology



Objectives

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

Background

The management plane of any infrastructure device should be protected as much as possible. Controlling access to routers and enabling reporting on routers are critical to network security and should be part of a comprehensive security policy.

In this lab, you build a multi-router network and secure the management plane of routers R1 and R3.

Required Resources

- 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.

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

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

R1

```
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)# exit

R1(config)# interface Serial0/0/0
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)#
exit
```

R2

```
R2(config)# exit
R2(config)# hostname R2
R2(config)# interface Serial0/0/0
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 Serial0/0/1
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
```

R3

```
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 Serial0/0/1
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)#
exit
```

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
```

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

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

```
R1# tclsh
R1(tcl)#foreach address {
+>(tcl)#192.168.1.1
+>(tcl)#10.1.1.1
+>(tcl)#10.1.1.2
+>(tcl)#10.2.2.1
+>(tcl)#10.2.2.2
+>(tcl)#192.168.3.1 +>(tcl)#} { ping
$address } Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
1/2/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
12/14/16 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max =
12/15/16 ms
R1(tcl)#
```

Are the pings now successful?

____

____

Yes. If not, troubleshoot.

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

How does configuring an enable secret password help protect a router from being compromised by an attack?

____
____

____
____

____
____

The goal is to always prevent unauthorized users from accessing a device using Telnet, SSH, or via the console. If attackers are able to penetrate this first layer of defense, using an enable secret password prevents them from being able to alter the configuration of the device. Unless the enable secret password is known, a user cannot go into privileged EXEC mode where they can display the running config and enter various configuration commands to make changes to the router. This provides an additional layer of security.

Passwords in this task are set to a minimum of 10 characters but are relatively simple for the benefit of performing the lab. More complex passwords are recommended in a production network.

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

Note: To avoid repetitive logins during this lab, the exec-timeout command can be set to 0 0, which prevents it from expiring. However, this is not considered a good security practice.

```
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)#
```

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)#
```

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
R1#
```

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

____
____

The enable secret password is encrypted automatically using the MD5 or SHA hash algorithm. . IOS 15.0(1)S and later default to SHA256 hashing algorithm. SHA256 which is considered to be a very strong hashing algorithm and is extremely difficult to reverse. Earlier IOS versions use the weaker MD5 hashing algorithm.

If the enable secret password command is lost or forgotten, it must be replaced using the Cisco router password recovery procedure. Refer to cisco.com for more information.

Can you read the console, aux, and vty passwords? Why or why not?

____
____
____

Yes. They are all in clear text.

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

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

Note: Password encryption is applied to all the passwords, including the username passwords, the authentication key passwords, the privileged command password, the console and the virtual terminal line access passwords, and the BGP neighbor passwords.

h. Issue the show run command.

h. Can you read the console, aux, and vty passwords? Why or why not?

____
____

____
____

No. The passwords are now encrypted.

Note: Type 7 passwords are encrypted using a Vigenère cipher which can be easily reversed. Therefore this command primarily protects from shoulder surfing attacks.

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

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

j. Issue the show run command.

i.

j.

k. What does the $ convert to in the output?

____
____

____
____

The $ is converted to ^C when the running-config is displayed.

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

l.

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

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
```

Note: An older method for creating local database entries is to use the username name password password command.

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
R1(config)#
```

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
R1(config)#
```

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

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

```
R1# telnet 10.2.2.2
Trying 10.2.2.2 ... Open
Unauthorized access strictly prohibited!
User Access Verification

Username: ADMIN Password:
R3>
```

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).

Users must authenticate against an authentication database which can be stored:

- Locally: Users are authenticated against the local device database which is created using the username secret command. Sometimes referred to self-contained AAA.
- Centrally: A client-server model where users are authenticated against AAA servers. This provides improved scalability, manageability and control. Communication between the device and AAA servers is secured using either the RADIUS or TACACS+ protocols.

In this step, we will configure AAA authentication to use a RADIUS server and the local database as a backup. Specifically, the authentication will be validated against one of two RADIUS servers. If the servers are not available, then authentication will be validated against the local database.

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
```

Note: Although the following configuration refers to two RADIUS servers, the actual RADIUS server implementation is beyond the scope. Therefore, the goal of this step is to provide an example of how to configure a router to access the servers.

b. Configure the specifics for the first RADIUS server located at 192.168.1.101. Use RADIUS1-pa55w0rd as the server password.

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

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

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

d. Assign both RADIUS servers to a server group.

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

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 RADIUSGROUP local
```

Note: Once this command is configured, all line access methods default to the default authentication method. The local option enables AAA to refer to the local database. Only the password is case sensitive.

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
```

Note: Unlike the local option that makes the password is case sensitive, local-case makes the username and password case sensitive.

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
```

```
R1(config)#
```

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

m.

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

```
R1# telnet 10.2.2.2
Trying 10.2.2.2 ... Open
Unauthorized access strictly prohibited!

User Access Verification

Username: admin Password:

% Authentication failed

Username: ADMIN Password:

R3>
```

Note: The first login attempt did not use the correct username (i.e., ADMIN) which is why it failed.

Note: The actual login time is longer since the RADIUS servers are not available.

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.

Note: For a router to support SSH, it must be configured with local authentication, (AAA services, or username) or password authentication. In this task, you configure an SSH username and local authentication.

In this step, you will enable R1 and R3 to support SSH instead of Telnet.

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
```

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
```

Note: If no keys exist, you might receive this message: % No Signature RSA Keys found in configuration.

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
The name for the keys will be: R1.ccnasecurity.com

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-
exportable...[OK]

R1(config)#
Jan 10 13:44:44.711: %SSH-5-ENABLED: SSH 1.99 has been enabled n.
```

d. Cisco routers support two versions of SSH:

- SSH version 1 (SSHv1): Original version but has known vulnerabilities.
- SSH version 2 (SSHv2): Provides better security using the Diffie-Hellman key exchange and the strong integrity-checking message authentication code (MAC).

o. The default setting for SSH is SSH version 1.99. This is also known as compatibility mode and is merely an indication that the server supports both SSH version 2 and SSH version 1. However, best practices are to enable version 2 only.

p.

q.

r.

s. Configure SSH version 2 on R1.

```
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
```

Note: SSH requires that the login local command be configured. However, in the previous step we enabled AAA authentication using the TELNET-LOGIN authentication method, therefore login local is not necessary.

Note: If you add the keyword telnet to the transport input command, users can log in using Telnet as well as SSH. However, the router will be less secure. If only SSH is specified, the connecting host must have an SSH client installed.

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

```
R1# show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits IOS Keys in SECSH
format(ssh-rsa, base64 encoded):
ssh-rsa
```

```
AAAAB3NzaC1yc2EAAAADAQABAAAAgQC3Lehh7ReYlgyDzls6wq+mFzxqzoaZFr
9XGx+Q/yio
dFYw00hQo80tZy1W1Ff3Pz6q7Qi0y00urwddHZ0kBZceZK9EzJ6wZ+9a87KKDE
TCWrGSLi6c8lE/y4K+
Z/oVrMMZk7bpTM1MFdP41YgkTf35utYv+TcqbsYo++KJiYk+xw==
R1#
```

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

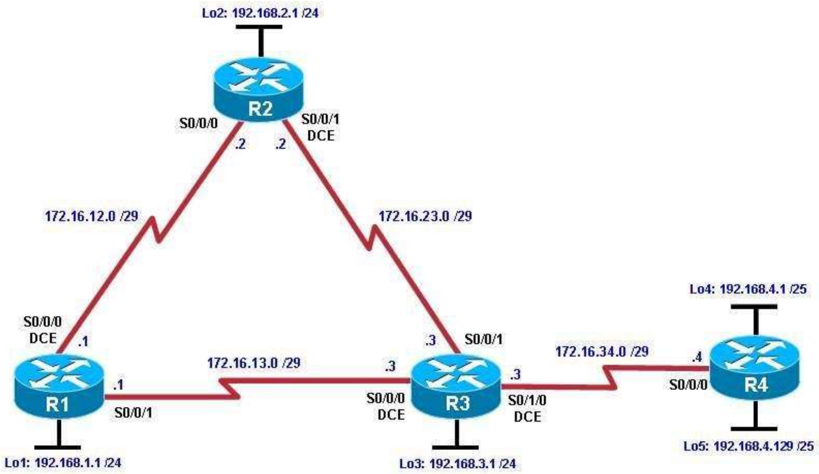
t.

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:
Unauthorized access strictly prohibited!
R3> R3>
en
Password:
R3#
```

Practical No.5

Aim: Configure and Verify Path Control Using PBR Topology



Objectives

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

Background

You want to experiment with policy-based routing (PBR) to see how it is implemented and to study how it could be of value to your organization. To this end, you have interconnected and configured a test network with four routers. All routers are exchanging routing information using EIGRP.

Note: This lab uses Cisco 1941 routers with Cisco IOS Release 15.2 with IP Base. 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

- 4 routers (Cisco IOS Release 15.2 or comparable)
- Serial and Ethernet cables

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.

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

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

Router R1

```
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)#exit
R1(config)#interface Serial0/0/0
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)#exit
R1(config)#interface Serial0/0/1
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)#exit
```

Router R2

```
R2(config)#hostname R2
R2(config-if)#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 Serial0/0/0
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 Serial0/0/1
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
```

Router R3

```
R3(config)#hostname R3
R3(config-if)#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 Serial0/0/0
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 Serial0/0/1
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 Serial0/1/0
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
```

Router R4

```
R4(config)#hostname R4

R4(config-if)#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 Serial0/0/0

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
```

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 | include up

Serial0/0/0          172.16.13.3    YES manual up              up

Serial0/0/1          172.16.23.3    YES manual up              up

Serial0/1/0          172.16.34.3    YES manual up              up

Loopback3              192.168.3.1    YES manual up              up


R3# show protocols

Global values:

  Internet Protocol routing is enabled

Embedded-Service-Engine0/0 is administratively down, line protocol is down

GigabitEthernet0/0 is administratively down, line protocol is down GigabitEthernet0/1 is

administratively down, line protocol is down

Serial0/0/0 is up, line protocol is up   Internet address is

172.16.13.3/29

Serial0/0/1 is up, line protocol is up

  Internet address is 172.16.23.3/29
```

```
Serial0/1/0 is up, line protocol is up

  Internet address is 172.16.34.3/29

Serial0/1/1 is administratively down, line protocol is down

Loopback3 is up, line protocol is up   Internet address is

192.168.3.1/24
```

```
R3# show interfaces description | include up

Se0/0/0                    up            up        R3 --> R1

Se0/0/1                    up            up        R3 --> R2

Se0/1/0                    up            up        R3 --> R4

Lo3                        up            up        R3 LAN
```

Step 3: Configure basic EIGRP.

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

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

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

Router R1

```
R1(config)#router eigrp 1

 R1(config-router)#network 192.168.1.0

 R1(config-router)#network 172.16.12.0 0.0.0.7

 R1(config-router)#network 172.16.13.0 0.0.0.7

 R1(config-router)#no auto-summary
```

Router R2

```
R2(config)#router eigrp 1

R2(config-router)#network 192.168.2.0

R2(config-router)#network 172.16.12.0 0.0.0.7

R2(config-router)#network 172.16.23.0 0.0.0.7

R2(config-router)#no auto-summary
```

Router R3

```
R3(config)#router eigrp 1

R3(config-router)#network 192.168.3.0

R3(config-router)#network 172.16.13.0 0.0.0.7
```

```
R3(config-router)#network 172.16.23.0 0.0.0.7 R3(config-router)#network 172.16.34.0 0.0.0.7

R3(config-router)#no auto-summary
```

Router R4

```
R4(config)#router eigrp 1

R4(config-router)#network 192.168.4.0

R4(config-router)#network 172.16.34.0 0.0.0.7

R4(config-router)#no auto-summary
```

You should see EIGRP neighbor relationship messages being generated.

Step 4: Verify EIGRP connectivity.

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

```
R1# show ip eigrp neighbors

EIGRP-IPv4 Neighbors for AS(1)

H   Address              Interface        Hold Uptime   SRTT
RTO  Q  Seq

                                    (sec)         (ms)       Cnt Num

1   172.16.13.3            Se0/0/1            10 00:01:55   27

2340  0  9

0       172.16.12.2            Se0/0/0            13 00:02:07    8
1170  0  11



R2# show ip eigrp neighbors

EIGRP-IPv4 Neighbors for AS(1)

H   Address              Interface        Hold Uptime   SRTT
RTO  Q  Seq

                                    (sec)         (ms)       Cnt Num

1       172.16.23.3            Se0/0/1            12 00:02:15   12  1170  0  10

0   172.16.12.1            Se0/0/0            11 00:02:27    9
1170  0  13



R3# show ip eigrp neighbors

EIGRP-IPv4 Neighbors for AS(1)

H   Address              Interface        Hold Uptime   SRTT
RTO  Q  Seq

                                    (sec)         (ms)       Cnt Num
```

```
2   172.16.34.4            Se0/1/0            12 00:02:14   44

2340  0  3

1   172.16.23.2            Se0/0/1            11 00:02:23   10

1170  0  10

0   172.16.13.1            Se0/0/0            10 00:02:23 1031
5000  0  12



R4# show ip eigrp neighbors

EIGRP-IPv4 Neighbors for AS(1)

H   Address                Interface         Hold Uptime   SRTT
RTO  Q  Seq

                                     (sec)         (ms)       Cnt Num

0   172.16.34.3            Se0/0/0            10 00:02:22   37  2340  0  11
```

Did you receive the output you expected?

---

b. 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 }
```

You should get ICMP echo replies for every address pinged. Make sure to run the Tcl script on each router.

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 Gateway of last resort is

not set



     172.16.0.0/16 is variably subnetted, 6 subnets, 2 masks

C       172.16.12.0/29 is directly connected, Serial0/0/0

L       172.16.12.1/32 is directly connected, Serial0/0/0

C       172.16.13.0/29 is directly connected, Serial0/0/1

L       172.16.13.1/32 is directly connected, Serial0/0/1

D       172.16.23.0/29 [90/21024000] via 172.16.12.2, 00:07:22, Serial0/0/0

D       172.16.34.0/29 [90/41024000] via 172.16.13.3, 00:07:22,
Serial0/0/1

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

C       192.168.1.0/24 is directly connected, Loopback1

L       192.168.1.1/32 is directly connected, Loopback1

D    192.168.2.0/24 [90/20640000] via 172.16.12.2, 00:07:22, Serial0/0/0

D    192.168.3.0/24 [90/21152000] via 172.16.12.2, 00:07:22,
Serial0/0/0

     192.168.4.0/25 is subnetted, 2 subnets

D       192.168.4.0 [90/41152000] via 172.16.13.3, 00:07:14,
Serial0/0/1

D       192.168.4.128 [90/41152000] via 172.16.13.3, 00:07:14,
Serial0/0/1
```

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

Note: You can specify the source as the interface address (for example 192.168.4.1) or the interface designator (for example, Fa0/0).

```
R4# traceroute 192.168.1.1 source 192.168.4.1

Type escape sequence to abort.

Tracing the route to 192.168.1.1

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

```
1   172.16.34.3 12 msec 12 msec 16 msec

2   172.16.23.2 20 msec 20 msec 20 msec

3   172.16.12.1 24 msec *  24 msec
```

```
R4# traceroute 192.168.1.1 source 192.168.4.129

Type escape sequence to abort.

Tracing the route to 192.168.1.1

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

1   172.16.34.3 12 msec 16 msec 12 msec

2   172.16.23.2 28 msec 20 msec 16 msec

3   172.16.12.1 24 msec *  24 msec
```

Notice that the path taken for the packets sourced from the R4 LANs are going through R3 --> R2 --> R1.

Why are the R4 interfaces not using the R3 --> R1 path?

____

____

____

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 S0/0/1.

```
R3# show ip route | begin Gateway Gateway of last resort is

not set


     172.16.0.0/16 is variably subnetted, 7 subnets, 2 masks

D        172.16.12.0/29 [90/21024000] via 172.16.23.2, 00:10:54,
Serial0/0/1

C        172.16.13.0/29 is directly connected, Serial0/0/0

L        172.16.13.3/32 is directly connected, Serial0/0/0

C        172.16.23.0/29 is directly connected, Serial0/0/1

L        172.16.23.3/32 is directly connected, Serial0/0/1

C        172.16.34.0/29 is directly connected, Serial0/1/0

L        172.16.34.3/32 is directly connected, Serial0/1/0

D     192.168.1.0/24 [90/21152000] via 172.16.23.2, 00:10:54, Serial0/0/1

D     192.168.2.0/24 [90/20640000] via 172.16.23.2, 00:10:54,
Serial0/0/1
```

```
    192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.3.0/24 is directly connected, Loopback3
L       192.168.3.1/32 is directly connected, Loopback3
    192.168.4.0/25 is subnetted, 2 subnets
D       192.168.4.0 [90/40640000] via 172.16.34.4, 00:10:47,
Serial0/1/0
D       192.168.4.128 [90/40640000] via 172.16.34.4, 00:10:47,
Serial0/1/0
```

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

```
R3# show interfaces serial0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is WIC MBRD Serial
  Description: R3 --> R1
  Internet address is 172.16.13.3/29
  MTU 1500 bytes, BW 64 Kbit/sec, DLY 20000 usec,     reliability 255/255, txload 1/255,
rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:01, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     399 packets input, 29561 bytes, 0 no buffer
     Received 186 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     393 packets output, 29567 bytes, 0 underruns
     0 output errors, 0 collisions, 3 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
```

```
    0 carrier transitions

    DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
```

```
R3# show interfaces serial0/0/0 | include BW

  MTU 1500 bytes, BW 64 Kbit/sec, DLY 20000 usec,

R3# show interfaces serial0/0/1 | include BW

  MTU 1500 bytes, BW 128 Kbit/sec, DLY 20000 usec,
```

Notice that the bandwidth of the serial link between R3 and R1 (S0/0/0) is set to 64 Kb/s, while the bandwidth of the serial link between R3 and R2 (S0/0/1) is set to 128 Kb/s.

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

```
R3# show ip eigrp topology 192.168.1.0

EIGRP-IPv4 Topology Entry for AS(1)/ID(192.168.3.1) for
192.168.1.0/24

  State is Passive, Query origin flag is 1, 1 Successor(s), FD is
21152000

  Descriptor Blocks:

  172.16.23.2 (Serial0/0/1), from 172.16.23.2, Send flag is 0x0      Composite metric is

(21152000/20640000), route is Internal

      Vector metric:

        Minimum bandwidth is 128 Kbit

        Total delay is 45000 microseconds

        Reliability is 255/255

        Load is 1/255

        Minimum MTU is 1500

        Hop count is 2

        Originating router is 192.168.1.1

  172.16.13.1 (Serial0/0/0), from 172.16.13.1, Send flag is 0x0

      Composite metric is (40640000/128256), route is Internal

      Vector metric:

        Minimum bandwidth is 64 Kbit

        Total delay is 25000 microseconds

        Reliability is 255/255

        Load is 1/255
```

Minimum MTU is 1500

Hop count is 1

Originating router is 192.168.1.1

As indicated, R4 has two routes to reach 192.168.1.0. However, the metric for the route to R1 (172.16.13.1) is much higher (40640000) than the metric of the route to R2 (21152000), making the route through R2 the successor route.

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

R3(config)#
```

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

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

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

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

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

R3(config)#
```

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 S0/1/0.

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

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

R3(config-if)# end
```

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

```
map route-map R3-to-R1, permit, sequence 10   Match clauses:

   ip address (access-lists): PBR-ACL

  Set clauses:

   ip next-hop 172.16.13.1

  Policy routing matches: 0 packets, 0 bytes
```

Note: There are currently no matches because no packets matching the ACL have passed through R3 S0/1/0.

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
```

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

```
R3# debug ip policy ?   <1-199>

Access list    dynamic   dynamic

PBR

  <cr>
```

```
R3# debug ip policy 1

Policy routing debugging is on for access list 1
```

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

```
R4# traceroute 192.168.1.1 source 192.168.4.1


Type escape sequence to abort.

Tracing the route to 192.168.1.1


1   172.16.34.3 0 msec 0 msec 4 msec

2   172.16.23.2 0 msec 0 msec 4 msec

3   172.16.12.1 4 msec 0 msec *
```

Notice the path taken for the packet sourced from R4 LAN A is still going through R3 --> R2 --> R1.

As the traceroute was being executed, router R3 should be generating the following debug output.

```
R3#

Jan 10 10:49:48.411: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, policy rejected --
normal forwarding

Jan 10 10:49:48.427: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, policy rejected --
normal forwarding

Jan 10 10:49:48.439: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, policy rejected --
normal forwarding

Jan 10 10:49:48.451: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy rejected(no
match) - normal forwarding

Jan 10 10:49:48.471: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy rejected(no
match) - normal forwarding

Jan 10 10:49:48.491: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy rejected(no
match) - normal forwarding

Jan 10 10:49:48.511: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy rejected(no
match) - normal forwarding

Jan 10 10:49:48.539: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy rejected(no
match) - normal forwarding

Jan 10 10:49:51.539: IP: s=192.168.4.1 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy rejected(no
match) - normal forwarding

R3#
```

Why is the traceroute traffic not using the R3 --> R1 path as specified in the R3-to-R1 policy?

____

____

d. 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

Type escape sequence to abort.

Tracing the route to 192.168.1.1

1   172.16.34.3 12 msec 12 msec 16 msec

2   172.16.13.1 28 msec 28 msec *
```

Now the path taken for the packet sourced from R4 LAN B is R3 --> R1, as expected.

The debug output on R3 also confirms that the traffic meets the criteria of the R3-to-R1 policy.

```
Jan 10 10:50:04.283: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, policy match

Jan 10 10:50:04.283: IP: route map R3-to-R1, item 10, permit

Jan 10 10:50:04.283: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1 (Serial0/0/0), len 28, policy routed

Jan 10 10:50:04.283: IP: Serial0/1/0 to Serial0/0/0 172.16.13.1

Jan 10 10:50:04.295: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, policy match

Jan 10 10:50:04.295: IP: route map R3-to-R1, item 10, permit

Jan 10 10:50:04.295: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1 (Serial0/0/0), len 28, policy routed

Jan 10 10:50:04.295: IP: Serial0/1/0 to Serial0/0/0 172.16.13.1

Jan 10 10:50:04.311: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, policy match

Jan 10 10:50:04.311: IP: route map R3-to-R1, item 10, permit

Jan 10 10:50:04.311: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1 (Serial0/0/0), len 28, policy routed

Jan 10 10:50:04.311: IP: Serial0/1/0 to Serial0/0/0 172.16.13.1

Jan 10 10:50:04.323: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy match

Jan 10 10:50:04.323: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, PBR Counted

Jan 10 10:50:04.323: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, g=172.16.13.1, len 28, FIB policy routed

Jan 10 10:50:04.351: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy match

Jan 10 10:50:04.351: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, PBR Counted

Jan 10 10:50:04.351: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, g=172.16.13.1, len 28, FIB policy routed

Jan 10 10:50:07.347: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, FIB policy match
Jan 10 10:50:07.347: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, len 28, PBR Counted

Jan 10 10:50:07.347: IP: s=192.168.4.129 (Serial0/1/0), d=192.168.1.1, g=172.16.13.1, len 28, FIB
policy routed
```

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

```
R3# show route-map    route-map R3-to-R1, permit, sequence 10   Match clauses:

    ip address (access-lists): PBR-ACL

  Set clauses:

    ip next-hop 172.16.13.1

Nexthop tracking current: 0.0.0.0

172.16.13.1, fib_nh:0,oce:0,status:0

  Policy routing matches: 12 packets, 384 bytes

R3#
```

Note: There are now matches to the policy because packets matching the ACL have passed through R3 S0/1/0.