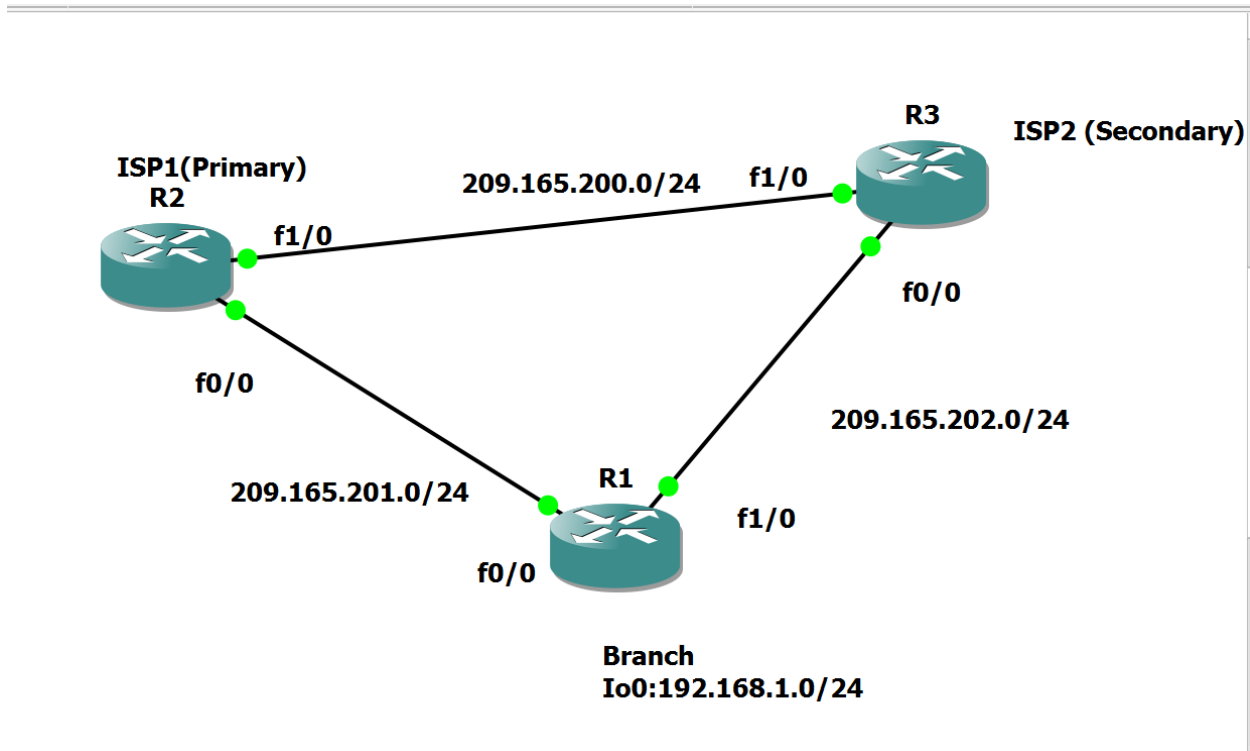


INDEX

SR.NO.	TITLE	DATE	SIGN
1.	Configure IP SLA tracking and path control topology		
2.	Implementation of BGP using AS_path attribute		
3.	Configure IGBP and EGBP Sessions		
4.	Secure management plane		
5.	Configure and verify path control using PBR (Policy Based Routing)		

Practical No: 1

Configure IP SLA tracking and path control topology.



Take 3 routers -> Configure -> slots -> PA-FE_TX

R1 configuration

General	Memories and disks	Slots	Advanced	Environment	Usage
Adapters					
slot 0: C7200-IO-FE					
slot 1: PA-FE-TX					
slot 2:					

Task 1: Configure IP SLA using GNS3**On router 1 console**

R1#conf t

R1(config)#interface f0/0

R1(config-if)#ip add 209.165.201.1 255.255.255.0

R1(config-if)#no sh

```
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#interface f0/0
R1(config-if)#ip add 209.165.201.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

R1(config-if) # interface f1/0

R1(config-if) # ip add 209.165.202.1 255.255.255.0

R1(config-if) # no sh

```
R1(config)#interface f1/0
R1(config-if)#ip add 209.165.202.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

R1(config-if) # int lo0

R1(config-if) # ip add 192.168.1.1 255.255.255.0

R1(config-if) # no sh

```
Loopback0 192.168.1.1 255.255.255.0
R1(config-if)#int lo0
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

R1(config-if) # do sh ip int br | include up

```
R1(config-if)#do sh ip int br | include up
FastEthernet0/0      209.165.201.1    YES manual up
FastEthernet1/0      209.165.202.1    YES manual up
Loopback0            192.168.1.1      YES manual up
R1(config-if)#
```

On router 2 console

R2 # conf t

R2(config) # int f0/0

R2(config-if) # ip add 209.165.201.2 255.255.255.0

R2(config-if) # no sh

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int f0/0
R2(config-if)#ip add 209.165.201.2 255.255.255.0
R2(config-if)#no sh
```

R2(config-if) # int f1/0

R2(config-if) # ip add 209.165.200.2 255.255.255.0

R2(config-if) # no sh

```
R2(config-if)#int f1/0
R2(config-if)#ip add 209.165.200.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

R2(config-if) # do sh ip int br | include up

```
R2(config-if)#do sh ip int br | include up
FastEthernet0/0      209.165.201.2    YES manual up
FastEthernet1/0      209.165.200.2    YES manual up
R2(config-if)#
```

On router 3 console

R3 # conf t

R3(config) # int f0/0

R3(config-if) # ip add 209.165.202.3 255.255.255.0

R3(config-if) # no sh

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#int f0/0
R3(config-if)#ip add 209.165.202.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

R3(config) # int f1/0

R3(config-if) # ip add 209.165.200.3 255.255.255.0

R3(config-if) # no sh

```
R3(config-if)#int f1/0
R3(config-if)#ip add 209.165.200.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

R3(config-if) # do sh ip int br | include up

```
R3(config-if)#do sh ip int br | include up
FastEthernet0/0      209.165.202.3    YES manual up
FastEthernet1/0      209.165.200.3    YES manual up
R3(config-if)#
```

Task 2: Configure static routing on branch router and dynamic routing using eigrp

On router 1 console

R1 # conf t

R1(config) # ip route 0.0.0.0 0.0.0.0 209.165.201.2

R1(config) #

```
R1(config-if)#ip route 0.0.0.0 0.0.0.0 209.165.201.2
R1(config)#
```

On router 2 console

R2(config) # router eigrp 1

R2(config-router) # network 209.165.200.0 0.0.0.255

R2(config-router) # network 209.165.201.0 0.0.0.255

R2(config-router) # no auto-summary

```
R2(config-if)#router eigrp 1
R2(config-router)#network 209.165.200.0 0.0.0.255
R2(config-router)#network 209.165.201.0 0.0.0.255
R2(config-router)#no auto-summary
R2(config-router)#
```

On router 3 console

R3(config) # router eigrp 1

R3(config-router) # network 209.165.200.0 0.0.0.255

R3(config-router) # network 209.165.202.0 0.0.0.255

R3(config-router) # no auto-summary

```
R3(config)#  
R3(config)#router eigrp 1  
R3(config-router)#network 209.165.200.0 0.0.0.255  
R3(config-router)#network 209.165.202.0 0.0.0.255  
R3(config-router)#no auto-summary  
R3(config-router)#
```

On router 2 console

R2(config-router) # exit

R2(config) # ip route 192.168.1.0 255.255.255.0 209.165.201.1

```
R2(config-router)#  
R2(config-router)#exit  
R2(config)#ip route 192.168.1.0 255.255.255.0 209.165.201.1  
R2(config)#
```

On router 3 console

R3(config-router) # exit

R3(config) # ip route 192.168.1.0 255.255.255.0 209.165.202.1

```
R3(config-router)#  
R3(config-router)#exit  
R3(config)#ip route 192.168.1.0 255.255.255.0 209.165.202.1  
R3(config)#
```

Ping other routers

R1(config) # do ping 209.165.200.3

R3(config) # do ping 209.165.201.1

```
R1(config)#do ping 209.165.200.3  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 209.165.200.3, timeout is 2 seconds:  
!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/23/40 ms  
R1(config)#
```

```
R3(config)#do ping 209.165.201.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.201.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/21/28 ms
R3(config)#
```

Ping other routers

R2(config) # do ping 192.168.1.1

R3(config) # do ping 192.168.1.1

```
R2(config)#do ping 192.168.1.1
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 = 1/60/132 ms
R2(config)#
```

```
R3(config)#
R3(config)#do ping 192.168.1.1
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 = 8/11/16 ms
R3(config)#
```

Give hostname

R1(config) # hostname r1-branch

R2(config) # hostname r2-isp1

R3(config) # hostname r3-isp2

```
R1(config)#
R1(config)#hostname r1-branch
r1-branch(config)#
```



```
R2(config)#  
R2(config)#hostname r2-isp1  
r2-isp1(config)#
```

```
R3(config)#  
R3(config)#hostname r3-isp2  
r3-isp2(config)#
```

Task 3: Configure IP SLA probes at branch router

On router 1 console

```
r1-branch(config) # ip sla 11  
r1-branch(config-ip-sla) # icmp-echo 209.165.201.2  
r1-branch(config-ip-sla-echo) # frequency 10  
r1-branch(config-ip-sla-echo) # exit  
r1-branch(config) #  
r1-branch(config) # ip sla schedule 11 life forever start-time now  
r1-branch(config) #  
r1-branch(config) # do sh ip sla configuration 11
```

```
r1-branch(config)#ip sla 11  
r1-branch(config-ip-sla)# icmp-echo 209.165.201.2  
r1-branch(config-ip-sla-echo)#frequency 10  
r1-branch(config-ip-sla-echo)#exit  
r1-branch(config)#  
r1-branch(config)#ip sla schedule 11 life forever start-time now  
r1-branch(config)#
```

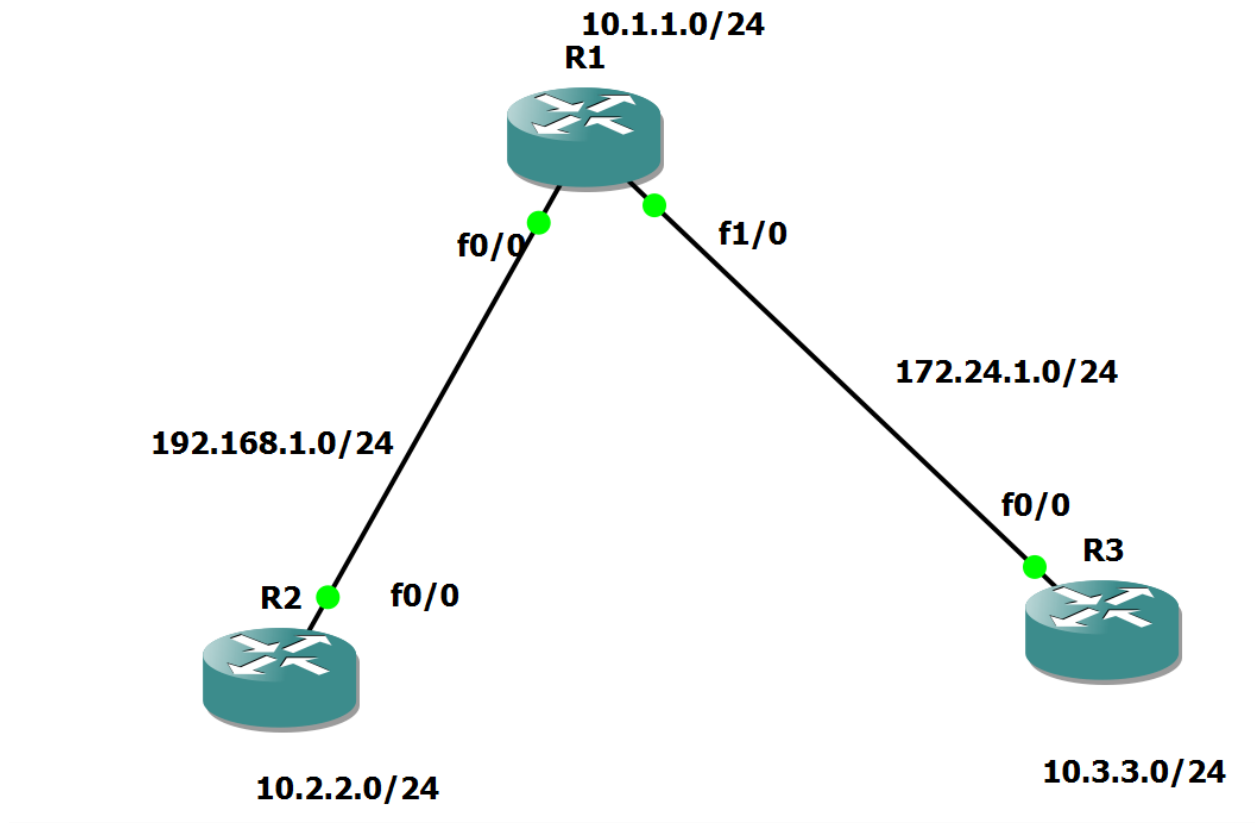
```
r1-branch(config)#do sh ip sla configuration 11
IP SLAs Infrastructure Engine-III
Entry number: 11
Owner:
Tag:
Operation timeout (milliseconds): 5000
Type of operation to perform: icmp-echo
Target address/Source address: 209.165.201.2/0.0.0.0
Type Of Service parameter: 0x0
Request size (ARR data portion): 28
Verify data: No
Vrf Name:
Schedule:
  Operation frequency (seconds): 10 (not considered if randomly scheduled)
  Next Scheduled Start Time: Start Time already passed
  Group Scheduled : FALSE
  Randomly Scheduled : FALSE
  Life (seconds): Forever
  Entry Ageout (seconds): never
  Recurring (Starting Everyday): FALSE
  Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
  Number of statistic hours kept: 2
  Number of statistic distribution buckets kept: 1
  Statistic distribution interval (milliseconds): 20
Enhanced History:

r1-branch(config)#ip sla 11
Entry already running and cannot be modified
  (only can delete (no) and start over)
  (check to see if the probe has finished exiting)

r1-branch(config)#
```

Practical No: 2

Implementation of BGP using AS_path attribute.



On Router console type following commands one by one.

R1 Console

```
R1(config)#conf t
```

```
R1(config)#int f0/0
```

```
R1(config-if)#ip add 192.168.1.1 255.255.255.0
```

```
R1(config-if)#no sh
```

R1(config-if)#int f1/0

R1(config-if)#ip add 172.24.1.1 255.255.255.0

R1(config-if)#no sh

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int f0/0
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh
```

```
R1(config-if)#int f1/0
R1(config-if)#ip add 172.24.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

R2 Console

R2#conf t

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

R2(config)#int f0/0

R2(config-if)#ip add 192.168.1.2 255.255.255.0

R2(config-if)#no sh

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int f0/0
R2(config-if)#ip add 192.168.1.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

R3 Console

R3#conf t

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

R3(config)#int f0/0

```
R3(config-if)#ip add 172.24.1.3 255.255.255.0
```

```
R3(config-if)#no sh
```

```
R3#  
R3#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
R3(config)#int f0/0  
R3(config-if)#ip add 172.24.1.3 255.255.255.0  
R3(config-if)#no sh  
R3(config-if)#
```

To add loopback address ,On Router console type following commands one by one.

R1 Console

```
R1#conf t
```

```
R1(config)#int lo0
```

```
R1(config-if)#ip add 10.1.1.1 255.255.255.0
```

```
R1(config-if)#
```

```
R1#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
R1(config)#int lo0  
R1(config-if)#ip add 10.1.1.1 255.255.255.0  
R1(config-if)#
```

R2 Console

```
R2#conf t
```

```
R2(config)#int lo0
```

```
R2(config-if)#ip add 10.2.2.2 255.255.255.0
```

```
R2(config-if)#
```

```
R2#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
R2(config)#int lo0  
R2(config-if)#ip add 10.2.2.2 255.255.255.0  
R2(config-if)#  
R2(config-if)#
```

R3 Console

R3#conf t

R3(config)#int lo0

R3(config-if)#ip add 10.3.3.3 255.255.255.0

```
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#int lo0
R3(config-if)#ip add 10.3.3.3 255.255.255.0
R3(config-if)#
```

To add bgp protocol, On Router console type following commands one by one.

R1 Console

R1(config-if)#router bgp 100

R1(config-router)#neighbor 192.168.1.2 remote-as 200

R1(config-router)#neighbor 172.24.1.3 remote-as 300

R1(config-router)#network 10.1.1.0 mask 255.255.255.0

```
R1(config-if)#router bgp 100
R1(config-router)#neighbor 192.168.1.2 remote-as 200
R1(config-router)#neighbor 172.24.1.3 remote-as 300
R1(config-router)#network 10.1.1.0 mask 255.255.255.0
R1(config-router)#
```

R2 Console

R2(config-if)#router bgp 200

R2(config-router)#neighbor 192.168.1.1 remote-as 100

R2(config-router)#network 10.2.2.0 mask 255.255.255.0

```
R2(config-if)#router bgp 200
R2(config-router)#neighbor 192.168.1.1 remote-as 100
R2(config-router)#network 10.2.2.0 mask 255.255.255.0
R2(config-router)#
```

R3 Console

```
R3(config-if)#router bgp 300
```

```
R3(config-router)#neighbor 172.24.1.1 remote-as 100
```

```
R3(config-router)#network 10.3.3.0 mask 255.255.255.0
```

```
R3(config-router)#
```

```
R3(config-if)#
R3(config-if)#router bgp 300
R3(config-router)#neighbor 172.24.1.1 remote-as 100
R3(config-router)#network 10.3.3.0 mask 255.255.255.0
R3(config-router)#
```

To show ip route type following command in each router console

```
R3#conf t
```

```
R3(config)#do sh ip route
```

```
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

    10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
B       10.1.1.0/24 [20/0] via 172.24.1.1, 00:00:48
B       10.2.2.0/24 [20/0] via 172.24.1.1, 00:00:48
C       10.3.3.0/24 is directly connected, Loopback0
L       10.3.3.3/32 is directly connected, Loopback0
    172.24.0.0/16 is variably subnetted, 2 subnets, 2 masks
C       172.24.1.0/24 is directly connected, FastEthernet0/0
L       172.24.1.3/32 is directly connected, FastEthernet0/0
R3(config)#
```

To verify output type following commands: (OUTPUT)

R2(config-router)#do ping 10.3.3.3 source lo0

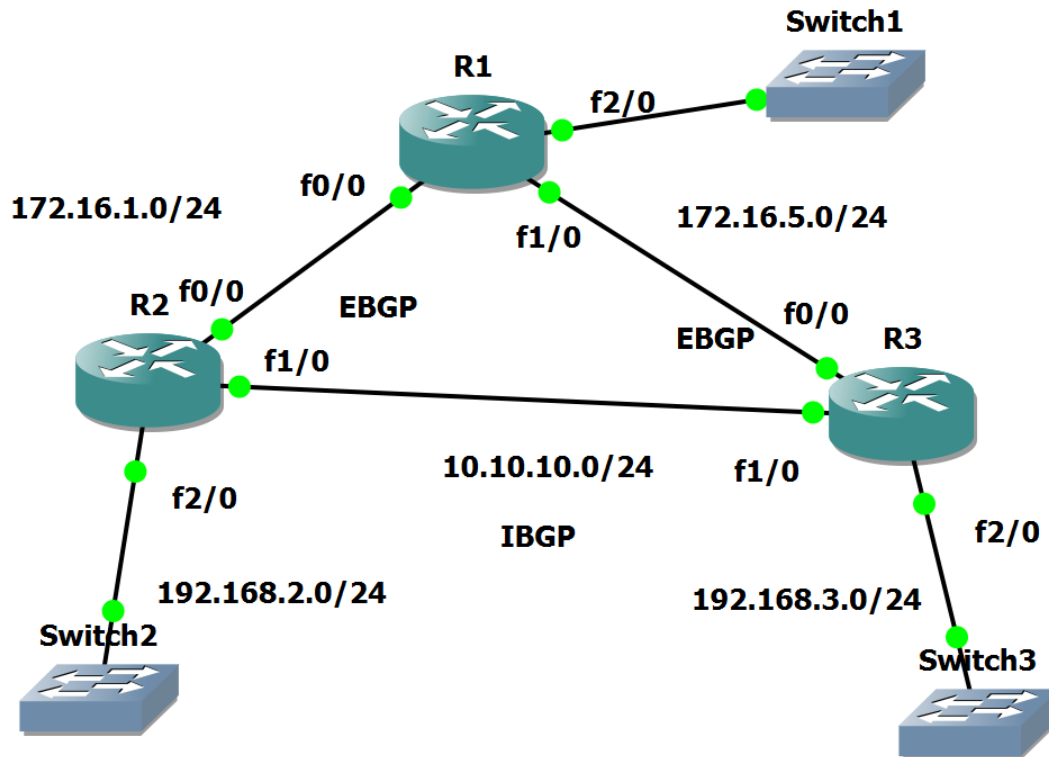
```
R2(config)#router bgp 200
R2(config-router)#do ping 10.3.3.3 source lo0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.3.3.3, timeout is 2 seconds:
Packet sent with a source address of 10.2.2.2
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/43/48 ms
R2(config-router)#
```

R3(config)#do ping 10.2.2.2 source lo0

```
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#do ping 10.2.2.2 source lo0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.3.3.3
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/40/44 ms
R3(config)#
```


Practical No: 3

Configure IGBP and EGBP Sessions



Step 1: Configure IP addresses on the given routers

On Router R1:

```
R1 # conf t
R1(config) # int f2/0
R1(config-if) # ip add 192.168.1.1 255.255.255.0
R1(config-if) # no sh
R1(config-if) #
```

```
R1(config-if) # int f0/0
R1(config-if) # ip add 172.16.1.1 255.255.255.0
R1(config-if) # no sh
R1(config-if) #
R1(config-if) # int f1/0
R1(config-if) # ip add 172.16.5.1 255.255.255.0
R1(config-if) # no sh
```

```
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int f2/0
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

```
R1(config-if)#int f0/0
R1(config-if)#ip add 172.16.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

```
Apr 23 23:23:54.399: %LINEPROTO-5-UPDOWN: Line protocol is
R1(config-if)#int f1/0
R1(config-if)#ip add 172.16.5.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

On Router R2:

```
R2 # conf t
R2(config) # int f1/0
R2(config-if) # ip add 10.10.10.2 255.255.255.0
R2(config-if) # no sh
R2(config-if) #
R2(config-if) # int f2/0
```

R2(config-if) # ip add 192.168.2.2 255.255.255.0

R2(config-if) # no sh

R2(config-if) #

R2(config-if) # int f0/0

R2(config-if) # ip add 172.16.1.2 255.255.255.0

R2(config-if) # no sh

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int f1/0
R2(config-if)#ip add 10.10.10.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

```
R2(config-if)#int f2/0
R2(config-if)#ip add 192.168.2.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

```
R2(config-if)#int f0/0
R2(config-if)#ip add 172.16.1.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

On Router 3:

R3 # conf t

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

R3(config) # int f1/0

R3(config-if) # ip add 10.10.10.3 255.255.255.0

R3(config-if) # no sh

R3(config-if) #

```
R3(config-if) # int f2/0
```

```
R3(config-if) # ip add 192.168.3.3 255.255.255.0
```

```
R3(config-if) # no sh
```

```
R3(config-if) #
```

```
R3(config-if) # int f0/0
```

```
R3(config-if) # ip add 172.16.5.3 255.255.255.0
```

```
R3(config-if) # no sh
```

```
R3(config)#int f1/0
R3(config-if)#ip add 10.10.10.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

```
R3(config-if)#int f2/0
R3(config-if)#ip add 192.168.3.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

```
R3(config-if)#int f0/0
R3(config-if)#ip add 172.16.5.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

On all routers:

```
do sh ip int br | include up
```

```
R1(config-if)#
R1(config-if)#do sh ip int br | include up
FastEthernet0/0      172.16.1.1      YES manual up
FastEthernet1/0      172.16.5.1      YES manual up
FastEthernet2/0      192.168.1.1     YES manual up
R1(config-if)#
```

```
R2(config-if)#  
R2(config-if)#do sh ip int br | include up  
FastEthernet0/0      172.16.1.2      YES manual up  
FastEthernet1/0      10.10.10.2      YES manual up  
FastEthernet2/0      192.168.2.2     YES manual up  
R2(config-if)#
```

```
R3(config-if)#  
R3(config-if)#do sh ip int br | include up  
FastEthernet0/0      172.16.5.3      YES manual up  
FastEthernet1/0      10.10.10.3      YES manual up  
FastEthernet2/0      192.168.3.3     YES manual up  
R3(config-if)#
```

Step 2: Configure IRP in autonomous system 65200

On Router R2:

```
R2(config-if) # router ospf 1
```

```
R2(config-router) # network 10.10.10.0 0.0.0.255 area 0
```

```
R2(config-router) # network 192.168.2.0 0.0.0.255 area 1
```

```
R2(config-if)#router ospf 1  
R2(config-router)#network 10.10.10.0 0.0.0.255 area 0  
R2(config-router)#network 192.168.2.0 0.0.0.255 area 1  
R2(config-router)#
```

On Router R3:

```
R3(config-if) # router ospf 1
```

```
R3(config-router) # network 10.10.10.0 0.0.0.255 area 0
```

```
R3(config-router) # network 192.168.3.0 0.0.0.255 area 2
```

```
R3(config-if)#  
R3(config-if)#router ospf 1  
R3(config-router)#network 10.10.10.0 0.0.0.255 area 0  
R3(config-router)#network 192.168.3.0 0.0.0.255 area 2  
R3(config-router)#
```

Step 3: IBGP & EBGp configuration:

On Router R1:

```
R1(config) # router bgp 65100
```

```
R1(config-router) # network 192.168.1.0
```

```
R1(config-router) # network 172.16.1.0 mask 255.255.255.0
```

```
R1(config-router) # network 172.16.5.0 mask 255.255.255.0
```

```
R1(config-router) # neighbor 172.16.1.2 remote-as 65200
```

```
R1(config-router) # neighbor 172.16.5.3 remote-as 65200
```

```
R1(config-router) # do sh ip route
```

```

R1(config-if)#
R1(config-if)#router bgp 65100
R1(config-router)#network 192.168.1.0
R1(config-router)#network 172.16.1.0 mask 255.255.255.0
R1(config-router)#network 172.16.5.0 mask 255.255.255.0
R1(config-router)#neighbor 172.16.1.2 remote-as 65200
R1(config-router)#neighbor 172.16.5.3 remote-as 65200
R1(config-router)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

    172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.1.0/24 is directly connected, FastEthernet0/0
L       172.16.1.1/32 is directly connected, FastEthernet0/0
C       172.16.5.0/24 is directly connected, FastEthernet1/0
L       172.16.5.1/32 is directly connected, FastEthernet1/0
    192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.0/24 is directly connected, FastEthernet2/0
L       192.168.1.1/32 is directly connected, FastEthernet2/0
R1(config-router)#

```

On Router R2:

```

R2(config-router) # router bgp 65200

R2(config-router) # redistribute ospf 1

R2(config-router) # network 172.16.1.0 mask 255.255.255.0

R2(config-router) # neighbor 172.16.1.1 remote-as 65100

R2(config-router) # neighbor 10.10.10.3 remote-as 65200

R2(config-router) # do sh ip route

```

```

R2(config-router)#
R2(config-router)#router bgp 65200
R2(config-router)#redistribute ospf 1
R2(config-router)#network 172.16.1.0 mask 255.255.255.0
R2(config-router)#neighbor 172.16.1.1 remote-as 65100
R2(config-router)#
*Apr 23 23:47:14.803: %BGP-5-ADJCHANGE: neighbor 172.16.1.1 Up
R2(config-router)#neighbor 10.10.10.3 remote-as 65200
R2(config-router)#

```

```

R2(config-router)#
R2(config-router)#neighbor 172.16.1.1 remote-as 65100
R2(config-router)#neighbor 10.10.10.3 remote-as 65200
R2(config-router)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

    10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.10.10.0/24 is directly connected, FastEthernet1/0
L       10.10.10.2/32 is directly connected, FastEthernet1/0
    172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
C       172.16.1.0/24 is directly connected, FastEthernet0/0
L       172.16.1.2/32 is directly connected, FastEthernet0/0
B       172.16.5.0/24 [20/0] via 172.16.1.1, 00:02:47
B       192.168.1.0/24 [20/0] via 172.16.1.1, 00:02:47
    192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.2.0/24 is directly connected, FastEthernet2/0
L       192.168.2.2/32 is directly connected, FastEthernet2/0
--More--

```

On Router R3:

```
R3(config-router) #
```

```
R3(config-router) # router bgp 65200
```

```
R3(config-router) # redistribute ospf 1
```

```
R3(config-router) # network 172.16.5.0 mask 255.255.255.0
```

```
R3(config-router) # neighbor 172.16.5.1 remote-as 65100
```

```
R3(config-router) # neighbor 10.10.10.2 remote-as 65200
```

```
R3(config-router) # do sh ip route
```

```

R3(config-router)#router bgp 65200
R3(config-router)#redistribute ospf 1
R3(config-router)#network 172.16.5.0 mask 255.255.255.0
R3(config-router)#neighbor 172.16.5.1 remote-as 65100
R3(config-router)#

```



```

R3(config-router)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

    10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.10.10.0/24 is directly connected, FastEthernet1/0
L       10.10.10.3/32 is directly connected, FastEthernet1/0
    172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
B       172.16.1.0/24 [200/0] via 10.10.10.2, 00:00:16
C       172.16.5.0/24 is directly connected, FastEthernet0/0
L       172.16.5.3/32 is directly connected, FastEthernet0/0
B       192.168.1.0/24 [20/0] via 172.16.5.1, 00:00:27
O IA    192.168.2.0/24 [110/2] via 10.10.10.2, 00:09:00, FastEthernet1/0
        192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.3.0/24 is directly connected, FastEthernet2/0
--More--

```

On Router R1:

do ping 192.168.3.3

do ping 192.168.2.2

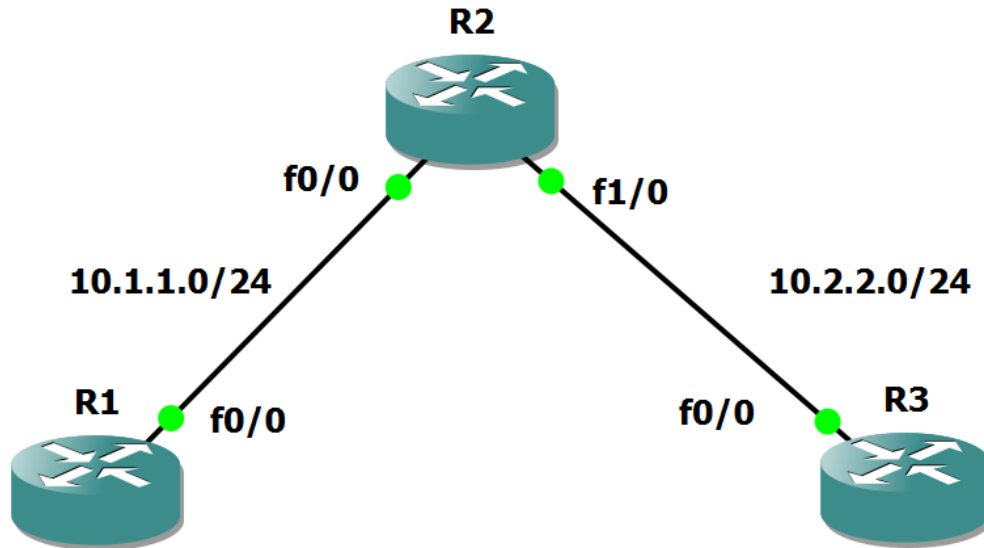
```

R1(config-router)#
R1(config-router)#do ping 192.168.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/24/32 ms
R1(config-router)#do ping 192.168.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/24 ms
R1(config-router)#

```


Practical No: 4

Secure management plane



On Router R1 Console:

```
R1#conf t
```

```
R1(config)#int f0/0
```

```
R1(config-if)#ip add 10.1.1.1 255.255.255.0
```

```
R1(config-if)#no sh
```

```
R1(config-if)#int lo0
```

```
R1(config-if)#ip add 192.168.1.1 255.255.255.0
```

```
R1(config-if)#no sh
```

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int f0/0
R1(config-if)#ip add 10.1.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

```
R1(config-if)#
R1(config-if)#int lo0
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

On Router R2 Console

R2 # conf t

R2(config) # int f0/0

R2(config-if) # ip add 10.1.1.2 255.255.255.0

R2(config-if) # no sh

R2(config-if) # int f1/0

R2(config-if) # ip add 10.2.2.2 255.255.255.0

R2(config-if) # no sh

```
R2(config)#int f0/0
R2(config-if)#ip add 10.1.1.2 255.255.255.0
R2(config-if)#no sh
```

```
R2(config-if)#
R2(config-if)#ip add 10.2.2.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

On Router R3 Console

R3 # conf t

R3(config) # int f0/0

R3(config-if) # ip add 10.2.2.3 255.255.255.0

R3(config-if) # no sh

R3(config-if) #

R3(config-if) # int lo0

R3(config-if) # ip add 192.168.3.3 255.255.255.0

```
R3(config)#int f0/0
R3(config-if)#ip add 10.2.2.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

```
R3(config-if)#no sh
R3(config-if)#int lo0
R3(config-if)#ip add 192.168.3.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

Part 2 : Routing

R1 Console

R1(config-if) # exit

R1(config) #

R1(config) # ip route 0.0.0.0 0.0.0.0 10.1.1.2

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

R2 Console

R2(config-if) # exit

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 10.2.2.3

```
R2(config-if)#exit
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 10.2.2.3
```

R3 Console

R3(config-if) # exit

R3(config) # ip route 0.0.0.0 0.0.0.0 10.2.2.2

```
R3(config-if)#exit
R3(config)#ip route 0.0.0.0 0.0.0.0 10.2.2.2
R3(config)#
```

Ping

R1 Console

R1(config) # do ping 192.168.3.3

```
R1(config)#ip route 0.0.0.0 0.0.0.0 10.1.1.2
R1(config)#do ping 192.168.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/48/76 ms
R1(config)#
```

R3 Console

R3(config) # do ping 192.168.1.1

```
R3(config)#do ping 192.168.1.1
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 = 40/45/60 ms
R3(config)#
```

Part 3: Security Management Access

R1 Console

r1(config) # hostname r1

r1(config) # security password min-length 10

```
r1(config) # enable secret class12345
r1(config) #
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) #
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) #
r1(config) # line aux 0
r1(config-line) # no exec
r1(config-line) # end
r1(config) # service password-encryption
r1(config) # banner motd $Unauthorized access not allowed$
r1(config) # exit
```

```

R1(config)#
R1(config)#hostname r1
r1(config)#security password min-length 10
r1(config)#enable secret class12345
r1(config)#
r1(config)#
r1(config)#line console 0
r1(config-line)#password ciscoconpass
r1(config-line)#exec-timeout 5 0
r1(config-line)#login
r1(config-line)#logging synchronous
r1(config-line)#exit
r1(config)#line vty 0 4
r1(config-line)#password ciscovtypass
r1(config-line)#
r1(config-line)#exec-timeout 5 0
r1(config-line)#
r1(config-line)#login
r1(config-line)#exit
r1(config)#
r1(config)#
r1(config)#line aux 0
r1(config-line)#
r1(config-line)#no exec
r1(config-line)#end
r1#

```

```

r1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
r1(config)#service password-encryption
r1(config)#banner motd $Unauthorized access not allowed$
r1(config)#exit
r1#

```

R3 Console (Same as R1)

```

R3(config) # hostname r3

r3(config) # security password min-length 10

r3(config) # enable secret class12345

r3(config) # line console 0

r3(config-line) # password ciscoconpass

r3(config-line) # exec-timeout 5 0

r3(config-line) # login

r3(config-line) # logging synchronous

```



```
r3(config-line) # exit
r3(config) # line vty 0 4
r3(config-line) # password ciscovtypass
r3(config-line) #
r3(config-line) #
r3(config-line) #
r3(config-line) # exec-timeout 5 0
r3(config-line) # login
r3(config-line) # exit
r3(config) #
r3(config) # line aux 0
r3(config-line) # no exec
r3(config-line) # end
r3 #
r3 # conf t
r3(config) # service password-encryption
r3(config) # banner motd $Unauthorized access not allowed$
r3(config) # exit
```

```

R3(config)#hostname r3
r3(config)#security password min-length 10
r3(config)#enable secret class12345
r3(config)#line console 0
r3(config-line)#password ciscoconpass
r3(config-line)#exec-timeout 5 0
r3(config-line)#login
r3(config-line)#
r3(config-line)#logging synchronous
r3(config-line)#exit
r3(config)#
r3(config)#line vty 0 4
r3(config-line)#password ciscovtypass
r3(config-line)#exec-timeout 5 0
r3(config-line)#login
r3(config-line)#exit
r3(config)#line aux 0
r3(config-line)#no exec
r3(config-line)#
r3(config-line)#end
r3#

```

```

r3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
r3(config)#
r3(config)#
r3(config)#service password-encryption
r3(config)#banner motd $Unauthorized access not allowed$
r3(config)#exit
r3#

```

R3 Console

r3 # telnet 10.1.1.1

(password-> ciscovtypass)

```

r3#telnet 10.1.1.1
Trying 10.1.1.1 ... Open
Unauthorized access not allowed

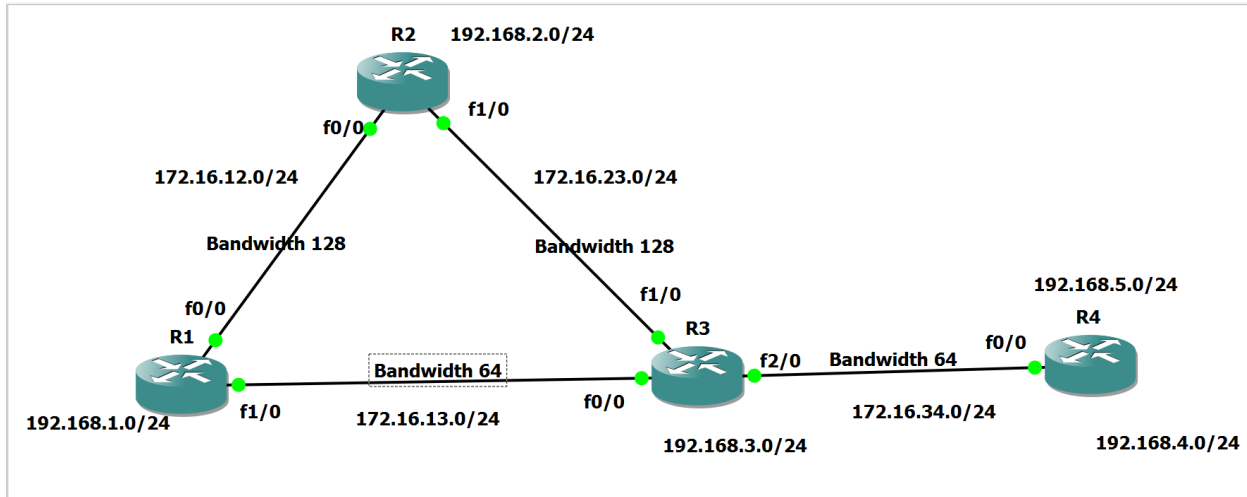
User Access Verification

Password:
Password:
r1>

```

Practical No: 5

Configure and verify path control using PBR (Policy Based Routing).



STEP 1: Perform IP configuration

On router 1 console

```
R1 # conf t
R1(config) # hostname r1
r1(config) # int f0/0
r1(config-if) # ip add 172.16.12.1 255.255.255.0
r1(config-if) # bandwidth 128
r1(config-if) # no sh
```

```
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#hostname r1
r1(config)#int f0/0
r1(config-if)#ip add 172.16.12.1 255.255.255.0
r1(config-if)#bandwidth 128
r1(config-if)#no sh
r1(config-if)#
```

```
r1(config-if) # int f1/0
r1(config-if) # ip add 172.16.13.1 255.255.255.0
```

r1(config-if) # bandwidth 64

r1(config-if) # no sh

```
r1(config-if)#
r1(config-if)#int f1/0
r1(config-if)#ip add 172.16.13.1 255.255.255.0
r1(config-if)#bandwidth 64
r1(config-if)#no sh
r1(config-if)#
```

r1(config-if) # int lo0

r1(config-if) # ip add 192.168.1.1 255.255.255.0

r1(config-if) # do sh ip int br | include up

```
r1(config-if)#int lo0
r1(config-if)#ip add 192.168.1.1 255.255.255.0
r1(config-if)#
r1(config-if)#do sh ip int br | include up
FastEthernet0/0      172.16.12.1      YES manual up
FastEthernet1/0      172.16.13.1      YES manual up
Loopback0            192.168.1.1      YES manual up
r1(config-if)#
```

On router 2 console

R2 # conf t

R2(config) # hostname r2

r2(config) #

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#hostname r2
```

r2(config) # int f0/0

r2(config-if) # ip add 172.16.12.2 255.255.255.0

r2(config-if) # bandwidth 128

r2(config-if) # no sh

```

r2(config-if)#int f0/0
r2(config-if)# ip add 172.16.12.2 255.255.255.0
r2(config-if)#bandwidth 128
r2(config-if)#no sh
r2(config-if)#

```

r2(config-if) # int f1/0

r2(config-if) # ip add 172.16.23.2 255.255.255.0

r2(config-if) # bandwidth 128

r2(config-if) # no sh

```

r2(config-if)#
r2(config-if)#int f1/0
r2(config-if)#ip add 172.16.23.2 255.255.255.0
r2(config-if)#bandwidth 128
r2(config-if)#no sh
r2(config-if)#

```

r2(config-if) # int lo0

r2(config-if) # ip add 192.168.2.2 255.255.255.0

r2(config-if) #

r2(config-if) # do sh ip int br | include up

```

r2(config-if)# int lo0
r2(config-if)#ip add 192.168.2.2 255.255.255.0
r2(config-if)#do sh ip int br | include up
FastEthernet0/0      172.16.12.2      YES manual up
FastEthernet1/0      172.16.23.2      YES manual up
Loopback0            192.168.2.2      YES manual up
r2(config-if)#

```

On router 3 console

R3 # conf t

R3(config) # hostname r3

```

R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#hostname r3

```

r3(config) # int f0/0

r3(config-if) # ip add 172.16.13.3 255.255.255.0

r3(config-if) # bandwidth 64

r3(config-if) # no sh

r3(config-if) #

```
r3(config-if)#  
r3(config-if)#int f0/0  
r3(config-if)#ip add 172.16.13.3 255.255.255.0  
r3(config-if)#bandwidth 64  
r3(config-if)#no sh  
r3(config-if)#
```

r3(config-if) # int f1/0

r3(config-if) # ip add 172.16.23.3 255.255.255.0

r3(config-if) # bandwidth 128

r3(config-if) # no sh

r3(config-if) #

```
r3(config-if)#  
r3(config-if)#int f1/0  
r3(config-if)#ip add 172.16.23.3 255.255.255.0  
r3(config-if)#bandwidth 128  
r3(config-if)#no sh  
r3(config-if)#  
r3(config-if)#
```

r3(config-if) # int f2/0

r3(config-if) # ip add 172.16.34.3 255.255.255.0

r3(config-if) # bandwidth 64

r3(config-if) # no sh

```
r3(config-if)#  
r3(config-if)# int f2/0  
r3(config-if)#ip add 172.16.34.3 255.255.255.0  
r3(config-if)#bandwidth 64  
r3(config-if)#no sh  
r3(config-if)#
```

r3(config-if) # int lo0

r3(config-if) # ip add 192.168.3.3 255.255.255.0

```
r3(config-if) #
```

```
r3(config-if) # do sh ip int br | include up
```

```
r3(config-if)#int lo0
r3(config-if)#ip add 192.168.3.3 255.255.255.0
r3(config-if)#do sh ip int br | include up
FastEthernet0/0      172.16.13.3      YES manual up
FastEthernet1/0      172.16.23.3      YES manual up
FastEthernet2/0      172.16.34.3      YES manual up
Loopback0            192.168.3.3      YES manual up
r3(config-if)#
```

On router 4 console

```
R4 # conf t
```

```
R4(config) # hostname r4
```

```
r4(config) #
```

```
R4#
R4#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R4(config)#hostname r4
r4(config)#
```

```
r4(config) # int f0/0
```

```
r4(config-if) # ip add 172.16.34.4 255.255.255.0
```

```
r4(config-if) # bandwidth 64
```

```
r4(config-if) # no sh
```

```
r4(config)#
r4(config)#int f0/0
r4(config-if)#ip add 172.16.34.4 255.255.255.0
r4(config-if)#bandwidth 64
r4(config-if)#no sh
r4(config-if)#
```

```
r4(config-if) # int lo0
```

```
r4(config-if) # ip add 192.168.4.1 255.255.255.0
```

```
r4(config-if) #
```

```

r4(config-if)#
r4(config-if)#int lo0
r4(config-if)#ip add 192.168.4.1 255.255.255.0
r4(config-if)#

```

r4(config-if) # int lo1

r4(config-if) # ip add 192.168.5.1 255.255.255.0

r4(config-if) #

r4(config-if) # do sh ip int br | include up

```

r4(config-if)#int lo1
r4(config-if)#ip add 192.168.4.1 255.255.255.0
% 192.168.4.0 overlaps with Loopback0
r4(config-if)#ip add 192.168.5.1 255.255.255.0
r4(config-if)#do sh ip int br | include up
FastEthernet0/0      172.16.34.4      YES manual up
Loopback0            192.168.4.1      YES manual up
Loopback1            192.168.5.1      YES manual up
r4(config-if)#

```

STEP 2 : Configure eigrp on all routers On router 1 console

On router 1 console

r1(config) # router eigrp 1

r1(config-router) # network 172.16.12.0 0.0.0.255

r1(config-router) # network 172.16.13.0 0.0.0.255

r1(config-router) # network 192.168.1.0

r1(config-router) # no auto-summary

```

r1(config-if)#
r1(config-if)#router eigrp 1
r1(config-router)#network 172.16.12.0 0.0.0.255
r1(config-router)#network 172.16.13.0 0.0.0.255
r1(config-router)#network 192.168.1.0
r1(config-router)#no auto-summary
r1(config-router)#

```


On router 2 console

```
r2(config) # router eigrp 1
r2(config-router) # network 172.16.12.0 0.0.0.255
r2(config-router) #
r2(config-router) # network 172.16.23.0 0.0.0.255
r2(config-router) # network 192.168.2.0
r2(config-router) # no auto-summary
```

```
r2(config-router)#
r2(config-router)#router eigrp 1
r2(config-router)#network 172.16.12.0 0.0.0.255
r2(config-router)#network 172.16.23.0 0.0.0.255
r2(config-router)#network 192.168.2.0
r2(config-router)#no auto-summary
r2(config-router)#
```

On router 3 console

```
r3(config-if) # router eigrp 1
r3(config-router) # network 172.16.13.0 0.0.0.255
r3(config-router) # network 172.16.13.0 0.0.0.255
r3(config-router) # network 172.16.23.0 0.0.0.255
r3(config-router) # network 172.16.34.0 0.0.0.255
r3(config-router) # network 192.168.3.0
r3(config-router) # no auto-summary
```

```
r3(config-router)#router eigrp 1
r3(config-router)#network 172.16.13.0 0.0.0.255
r3(config-router)#network 172.16.13.0 0.0.0.255
r3(config-router)#network 172.16.23.0 0.0.0.255
```

```
r3(config-router)#network 172.16.34.0 0.0.0.255
r3(config-router)#network 192.168.3.0
r3(config-router)#no auto-summary
r3(config-router)#
```

On router 4 console

```
r4(config) # router eigrp 1
r4(config-router) # network 172.16.34.0 0.0.0.255
r4(config-router) #
r4(config-router) # network 192.168.4.0
r4(config-router) # network 192.168.5.0
r4(config-router) # no auto-summary
```

```
r4(config-router)#
r4(config-router)#router eigrp 1
r4(config-router)#network 172.16.34.0 0.0.0.255
r4(config-router)#
```

```
r4(config-router)#network 192.168.4.0
r4(config-router)#network 192.168.5.0
r4(config-router)#network 192.168.5.0
r4(config-router)#
```

STEP 3: Command on all routers

do sh ip route

r4(config) # do ping 192.168.1.1

```
r4(config-router)#do ping 192.168.1.1
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 = 568/800/992 ms
r4(config-router)#
```

r1(config) # do ping 192.168.4.1

```
r1(config-router)#do ping 192.168.4.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.4.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 392/684/820 ms
r1(config-router)#
```

On Router R4

r4(config) # do traceroute 192.168.1.1 source 192.168.4.1

r4(config) #

r4(config) # do traceroute 192.168.1.1 source 192.168.5.1

```
r4(config-router)#
r4(config-router)#do 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)
  1 172.16.34.3 400 msec 560 msec 12 msec
  2 172.16.23.2 776 msec 748 msec 1024 msec
  3 172.16.12.1 592 msec 624 msec 1204 msec
r4(config-router)#do traceroute 192.168.1.1 source 192.168.5.1
Type escape sequence to abort.
Tracing the route to 192.168.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.34.3 408 msec 392 msec 540 msec
  2 172.16.23.2 596 msec 936 msec 964 msec
  3 172.16.12.1 976 msec 1160 msec 1140 msec
r4(config-router)#
```

Configure PBR to provide path control

- All traffic from source 192.168.5.1 should take route R4 -> R3 -> R1
- All traffic from source 192.168.4.1 should take route R4 -> R3 -> R2 -> R1

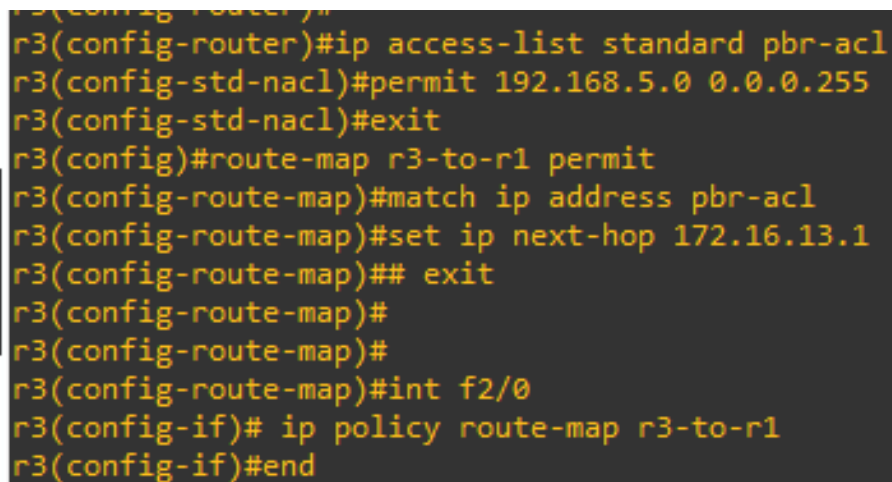
On router 3 console

r3(config) # ip access-list standard pbr-acl

r3(config-std-nacl) # permit 192.168.5.0 0.0.0.255

r3(config-std-nacl) # exit

```
r3(config) #  
r3(config) #  
r3(config) # route-map r3-to-r1 permit  
r3(config-route-map)#match ip address pbr-acl  
r3(config-route-map)#set ip next-hop 172.16.13.1  
r3(config-route-map)## exit  
r3(config-route-map)#  
r3(config-route-map)#  
r3(config-route-map)#int f2/0  
r3(config-if)# ip policy route-map r3-to-r1  
r3(config-if)#end
```

A screenshot of a terminal window showing the configuration of router 3. The text is yellow on a dark background. The commands are: r3(config-router)#ip access-list standard pbr-acl, r3(config-std-nacl)#permit 192.168.5.0 0.0.0.255, r3(config-std-nacl)#exit, r3(config)#route-map r3-to-r1 permit, r3(config-route-map)#match ip address pbr-acl, r3(config-route-map)#set ip next-hop 172.16.13.1, r3(config-route-map)## exit, r3(config-route-map)#, r3(config-route-map)#, r3(config-route-map)#int f2/0, r3(config-if)# ip policy route-map r3-to-r1, and r3(config-if)#end.

```
r3(config-router)#ip access-list standard pbr-acl  
r3(config-std-nacl)#permit 192.168.5.0 0.0.0.255  
r3(config-std-nacl)#exit  
r3(config)#route-map r3-to-r1 permit  
r3(config-route-map)#match ip address pbr-acl  
r3(config-route-map)#set ip next-hop 172.16.13.1  
r3(config-route-map)## exit  
r3(config-route-map)#  
r3(config-route-map)#  
r3(config-route-map)#int f2/0  
r3(config-if)# ip policy route-map r3-to-r1  
r3(config-if)#end
```

On router 4 console

```
r4(config) # do traceroute 192.168.1.1 source 192.168.4.1  
r4(config) # do traceroute 192.168.1.1 source 192.168.5.1
```

```
r4(config-router)#do 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)
  1 172.16.34.3 584 msec 384 msec 388 msec
  2 172.16.23.2 784 msec 752 msec 920 msec
  3 172.16.12.1 592 msec 1044 msec 1048 msec
r4(config-router)#do traceroute 192.168.1.1 source 192.168.5.1
Type escape sequence to abort.
Tracing the route to 192.168.1.1
VRF info: (vrf in name/id, vrf out name/id)
  1 172.16.34.3 396 msec 400 msec 568 msec
  2 172.16.13.1 972 msec 944 msec 936 msec
r4(config-router)#
```