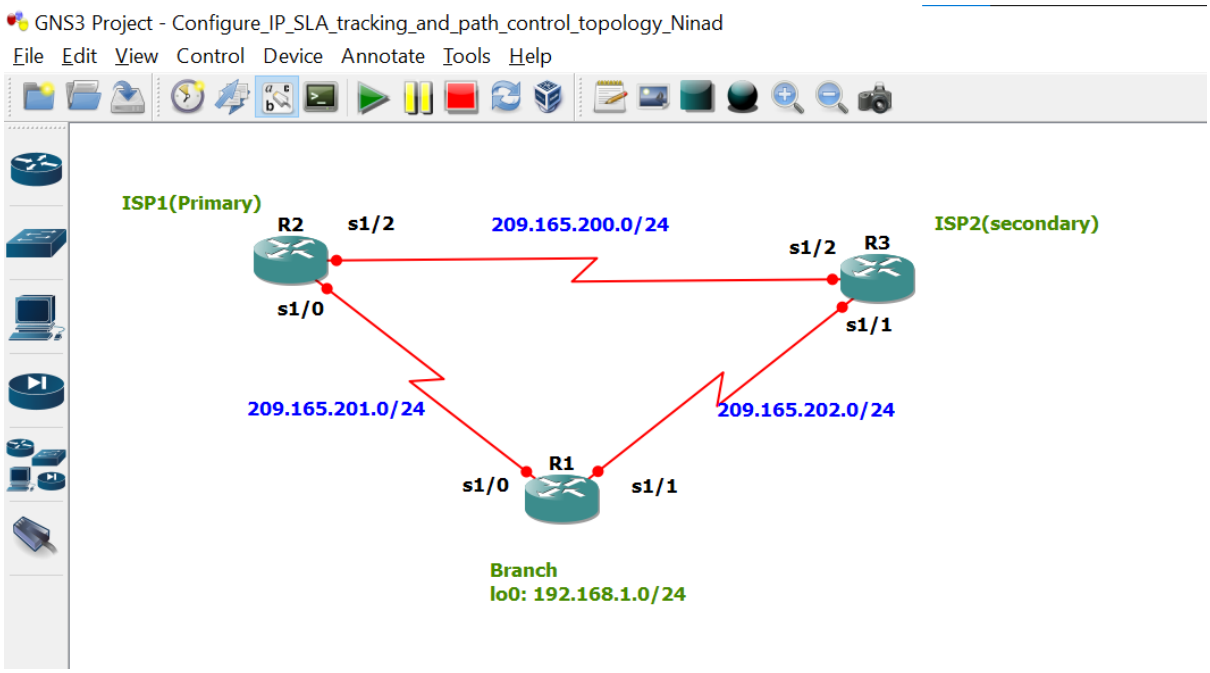
223060A1012

INDEX

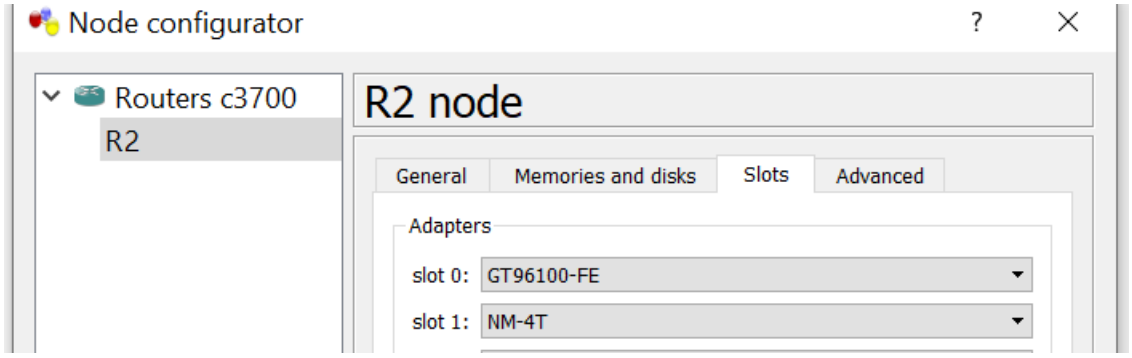
Sr. No.	Title/Aim of The Practical	Page No.	Date	Signature
1	Configure IP SLA tracking and path control topology.		17/04/2023	
2	Implementation of BGP using AS_path attribute.		13/03/2023	
3	Configuring IBGP and EBGP sessions.		08/05/2023	
4	Secure management plane.		14/03/2023	
5	Configure and verify path control using PBR (Policy Based Routing).		18/04/2023	
6	Demonstrate inter vlan routing.		11/04/2023	
7	Simulating MP LS environment.		08/05/2023	
8	Simulating VRF (Virtual Routing and Forwarding).		08/05/2023	

Practical No: 1 Ninad Karlekar 22306A1012 Date: 17/04/2023

Aim: Configure IP SLA tracking and path control topology.



Take 3 routers -> Configure -> slots -> NM-4T



Task 1: Configure IP SLA using GNS3

On router 1 console

R1 #
R1 # conf t
R1(config) # int s1/0
R1(config-if) # ip add 209.165.201.1 255.255.255.0
R1(config-if) # no sh
R1(config-if) #

R1(config-if) # int s1/1
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) #
R1(config-if) # do sh ip int br | include up

```
R1#
R1#conf t
Enter configuration commands, one per line.  End wi
R1(config)#int s1/0
R1(config-if)#ip add 209.165.201.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

```
R1(config-if)#
R1(config-if)#int s1/1
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)#
R1(config-if)#do sh ip int br | include up
Serial1/0                  209.165.201.1   YES manual up                    down
Serial1/1                  209.165.202.1   YES manual up                    down
Loopback0                  192.168.1.1     YES manual up                    up
R1(config-if)#
```

On router 2 console

R2 # conf t
R2(config) # int s1/0
R2(config-if) # ip add 209.165.201.2 255.255.255.0
R2(config-if) # no sh
R2(config-if) #
R2(config-if) # int s1/2
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#
R2#conf t
Enter configuration commands, one per line.  End wit
R2(config)#int s1/0
R2(config-if)#ip add 209.165.201.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

```
R2(config-if)#
R2(config-if)#int s1/2
R2(config-if)#ip add 209.165.200.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

```
R2(config-if)#
R2(config-if)#do sh ip int br | include up
Serial1/0                  209.165.201.2   YES manual up                    up
Serial1/2                  209.165.200.2   YES manual up                    up
R2(config-if)#exit
```

On router 3 console

R3 # conf t
R3(config) # int s1/1
R3(config-if) # ip add 209.165.202.3 255.255.255.0
R3(config-if) # no sh
R3(config-if) #
R3(config-if) # int s1/2
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#
R3#conf t
Enter configuration commands, one per line.  End wi
R3(config)#int s1/1
R3(config-if)#ip add 209.165.202.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

```
R3(config-if)#
R3(config-if)#int s1/2
R3(config-if)#ip add 209.165.200.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
```

```
R3(config-if)#
R3(config-if)#do sh ip int br | include up
Serial1/1                  209.165.202.3   YES manual up                    up
Serial1/2                  209.165.200.3   YES manual up                    up
R3(config-if)#
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#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip route 0.0.0.0 0.0.0.0 209.165.201.2
R1(config)#
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)#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)#exit
R2(config)#
R2(config)#ip route 192.168.1.0 255.255.255.0 209.165.201.1
```

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

Ping other routers

R1(config) # do ping 209.165.200.3
R3(config) # do ping 209.165.201.1

```
R1(config)#
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 = 36/55/64 ms
R1(config)#
R1(config)#
```

```
R3(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 = 48/58/64 ms
R3(config)#
```

Ping other routers

R2(config) # do ping 192.168.1.1
R3(config) # do ping 192.168.1.1

```
R2(config)#
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 = 16/28/36 ms
R2(config)#
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 = 20/28/36 ms
R3(config)#
```

Give hostname

R1(config) # hostname r1-branch
R2(config) # hostname r2-isp1
R3(config) # hostname r3-isp2

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

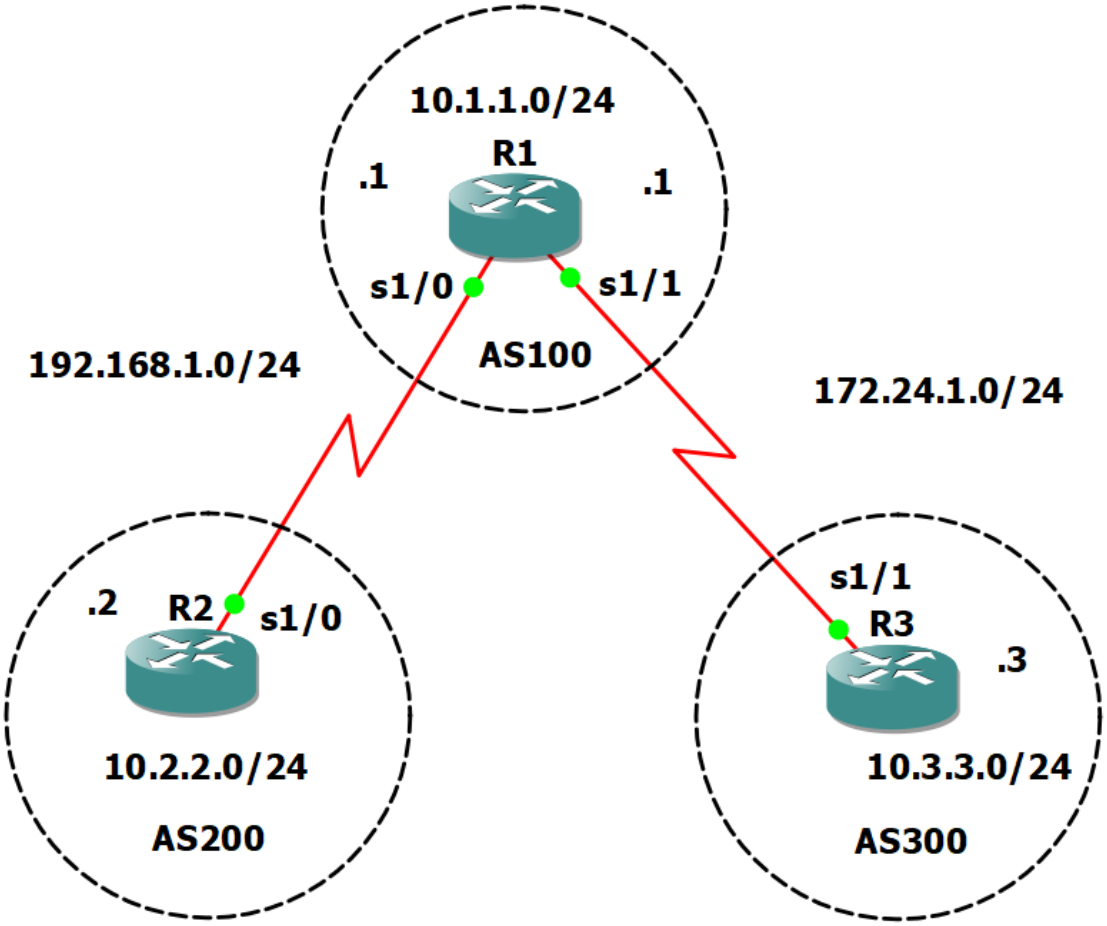
```
r1-branch(config)#
r1-branch(config)#ip sla 11
r1-branch(config-ip-sla)#icmp-echo 209.165.201.2
r1-branch(config-ip-sla-echo)#frequency 10
r1-branch(config-ip-sla-echo)#exit
r1-branch(config)#
r1-branch(config)#ip sla schedule 11 life forever start-time now
r1-branch(config)#
```

```
r1-branch(config)#
r1-branch(config)#do sh ip sla configuration 11
IP SLAs, Infrastructure Engine-II.
Entry number: 11
Owner:
Tag:
Type of operation to perform: icmp-echo
Target address/Source address: 209.165.201.2/0.0.0.0
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Vrf Name:
Request size (ARR data portion): 28
Verify data: No
Schedule:
   Operation frequency (seconds): 10  (not considered i
   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): 4294
History Statistics:
   Number of history Lives kept: 0
   Number of history Buckets kept: 15
   History Filter Type: None
Enhanced History:
```

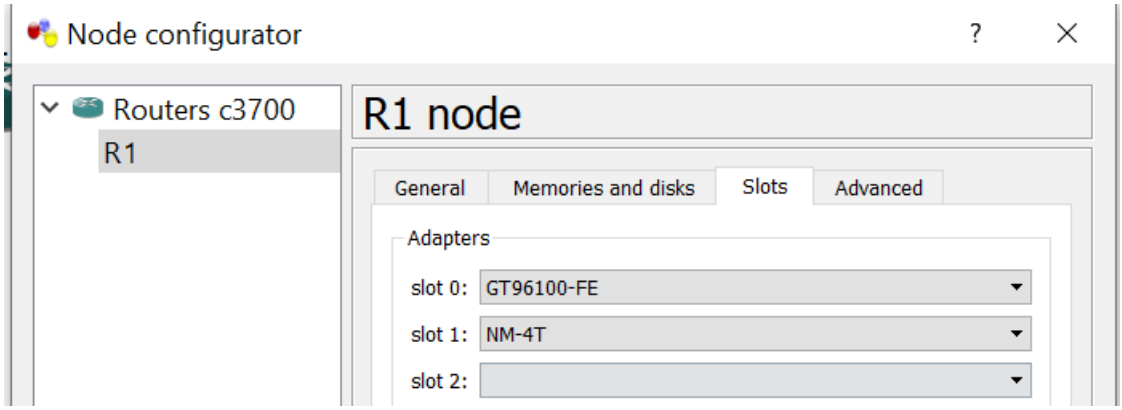
```
r1-branch(config)#
r1-branch(config)#do sh ip sla statistics

Round Trip Time (RTT) for       Index 11
        Latest RTT: 44 milliseconds
Latest operation start time: *00:41:49.903
Latest operation return code: OK
Number of successes: 6
Number of failures: 0
Operation time to live: Forever


r1-branch(config)#
r1-branch(config)#do wr
Building configuration...
[OK]
r1-branch(config)#
```

Practical No: 2 Ninad Karlekar 22306A1012 Date: 13/03/2023

Aim: Implementation of BGP using AS_path attribute.



Take 3 routers -> Configure -> slots -> NM-4T



On Router console type following commands one by one.

R1 Console

conf t
int s1/0
ip add 192.168.1.1 255.255.255.0
no sh
int s1/1
ip add 172.24.1.1 255.255.255.0
no sh

```
R1#conf t
Enter configuration commands, one per line.  End with
R1(config)#int s1/0
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
*Mar  1 00:11:55.495: %LINK-3-UPDOWN: Interface Serial1/0, changed
 state to up
R1(config-if)#
*Mar  1 00:11:56.499: %LINEPROTO-5-UPDOWN: Line protocol on Interf
ace Serial1/0, changed state to up
R1(config-if)#int s1/1
R1(config-if)#ip add 172.24.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

R2 Console

conf t
int s1/0
ip add 192.168.1.2 255.255.255.0
no sh

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int s1/0
R2(config-if)#ip add 192.168.1.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

R3 Console

conf t
int s1/1
ip add 172.24.1.3 255.255.255.0
no sh

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#int s1/1
R3(config-if)#ip add 172.24.1.3 255.255.255.0
R3(config-if)#no sh
```

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

R1 Console

int lo0
ip add 10.1.1.1 255.255.255.0

```
R1(config-if)#
R1(config-if)#int lo0
R1(config-if)#
*Mar  1 00:26:35.995: %LINEPROTO-5-UPDOWN: Line p
ace Loopback0, changed state to up
R1(config-if)#ip add 10.1.1.1 255.255.255.0
R1(config-if)#
```

R2 Console

int lo0
ip add 10.2.2.2 255.255.255.0

```
R2(config-if)#int lo0
R2(config-if)#ip a
*Mar  1 00:27:31.011: %LINEPROTO-5-UPDOWN: Line
ace Loopback0, changed state to up
R2(config-if)#ip add 10.2.2.2 255.255.255.0
R2(config-if)#
```

R3 Console

int lo0
ip add 10.3.3.3 255.255.255.0

```
R3(config-if)#int lo0
R3(config-if)#
*Mar  1 00:23:08.683: %LINEPROTO-5-UPDOWN: Line pro
ace Loopback0, changed state to up
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

router bgp 100
neighbor 192.168.1.2 remote-as 200
neighbor 172.24.1.3 remote-as 300
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
*Mar  1 00:39:51.291: %BGP-5-ADJCHANGE: neighbor 192.168
1.2 Up
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

router bgp 200
neighbor 192.168.1.1 remote-as 100
network 10.2.2.0 mask 255.255.255.0

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

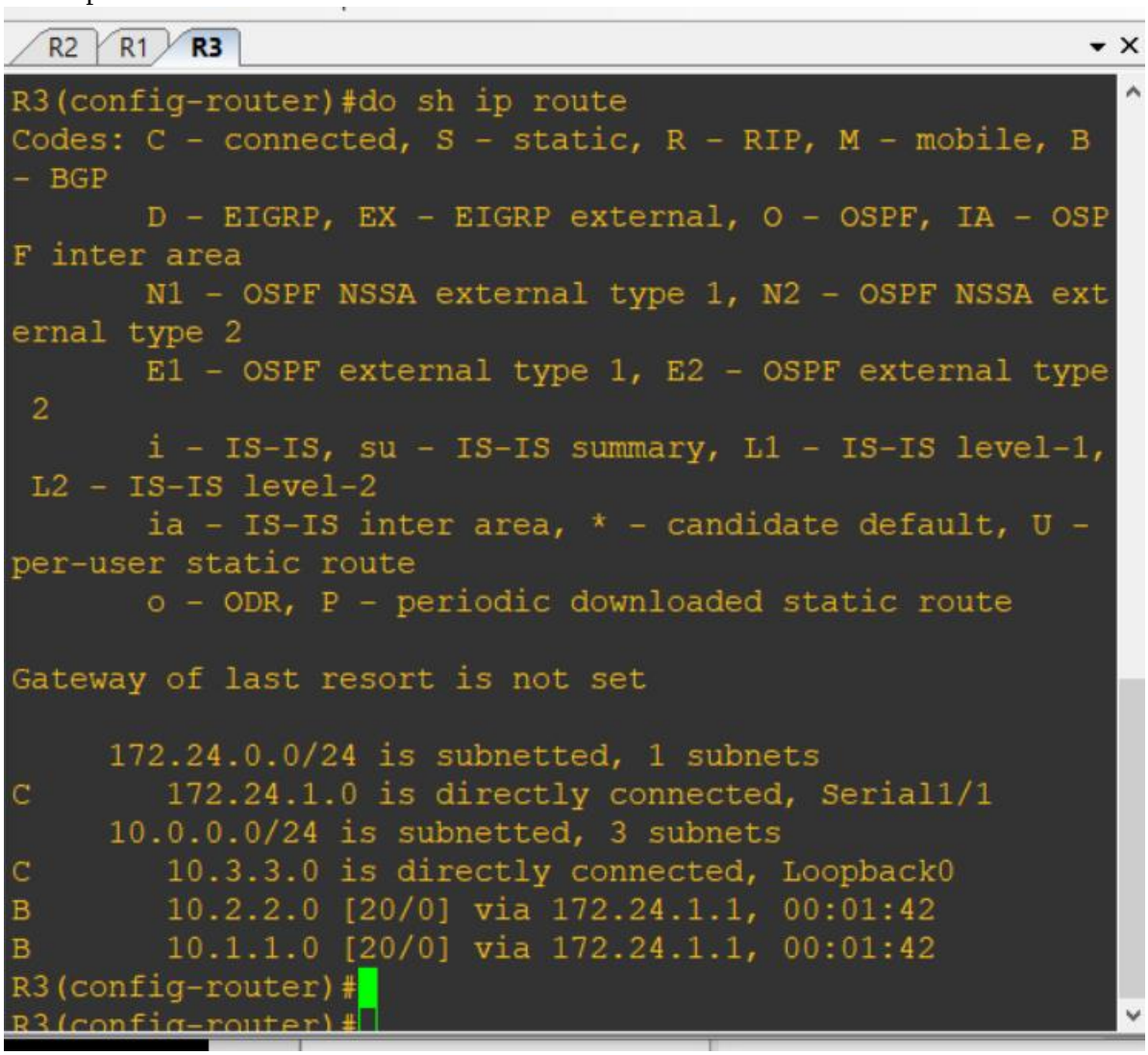
R3 Console

router bgp 300
neighbor 172.24.1.1 remote-as 100
network 10.3.3.0 mask 255.255.255.0

```
R3(config-if)#router bgp 300
R3(config-router)#neighbor 172.24.1.1 remote-as 100
R3(config-router)#network
*Mar  1 00:42:31.635: %BGP-5-ADJCHANGE: neighbor 172.24
.1 Up
R3(config-router)#network 10.3.3.0 mask 255.255.255.0
R3(config-router)#
```

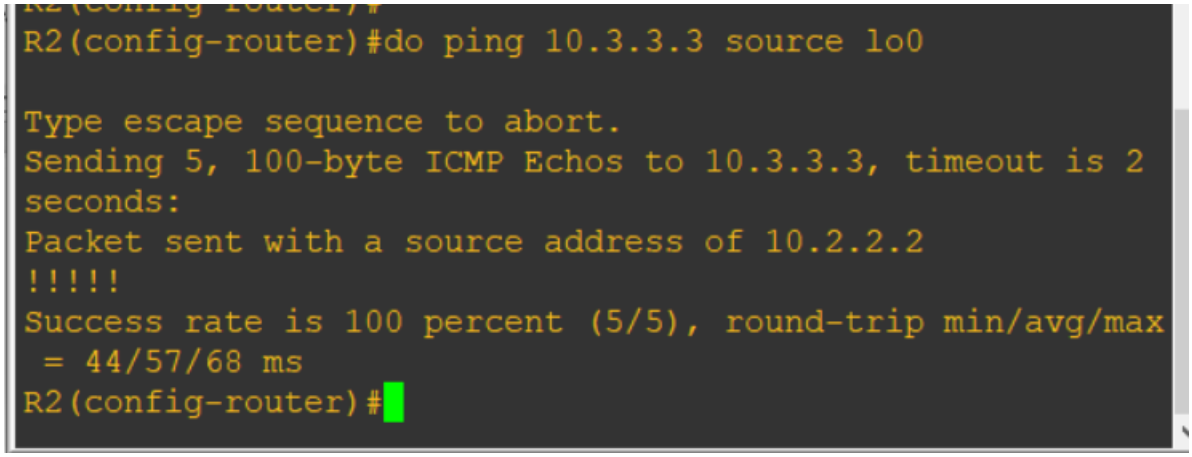
To show ip route type following command in each router console

do sh ip route

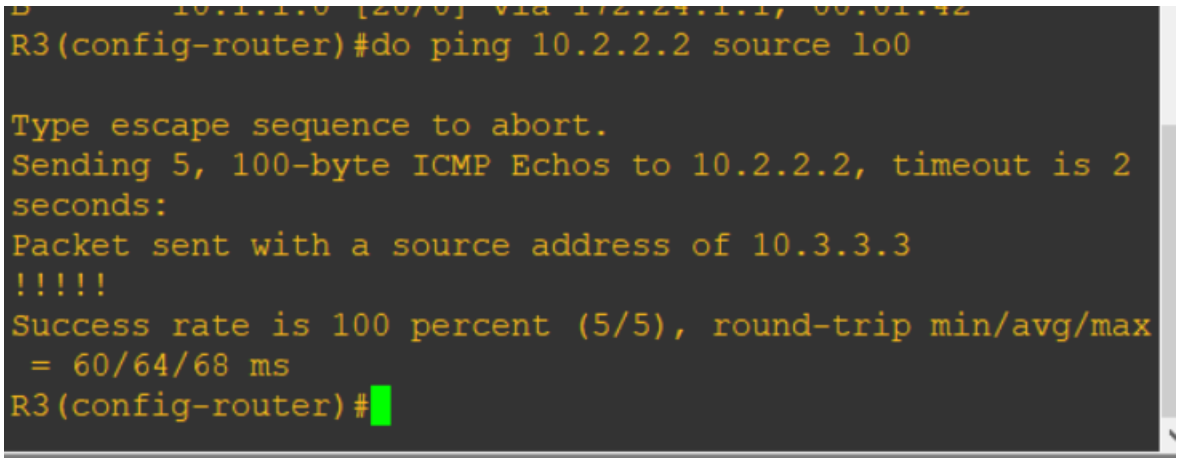


To verify output type following commands: (OUTPUT)

do ping 10.3.3.3 source lo0

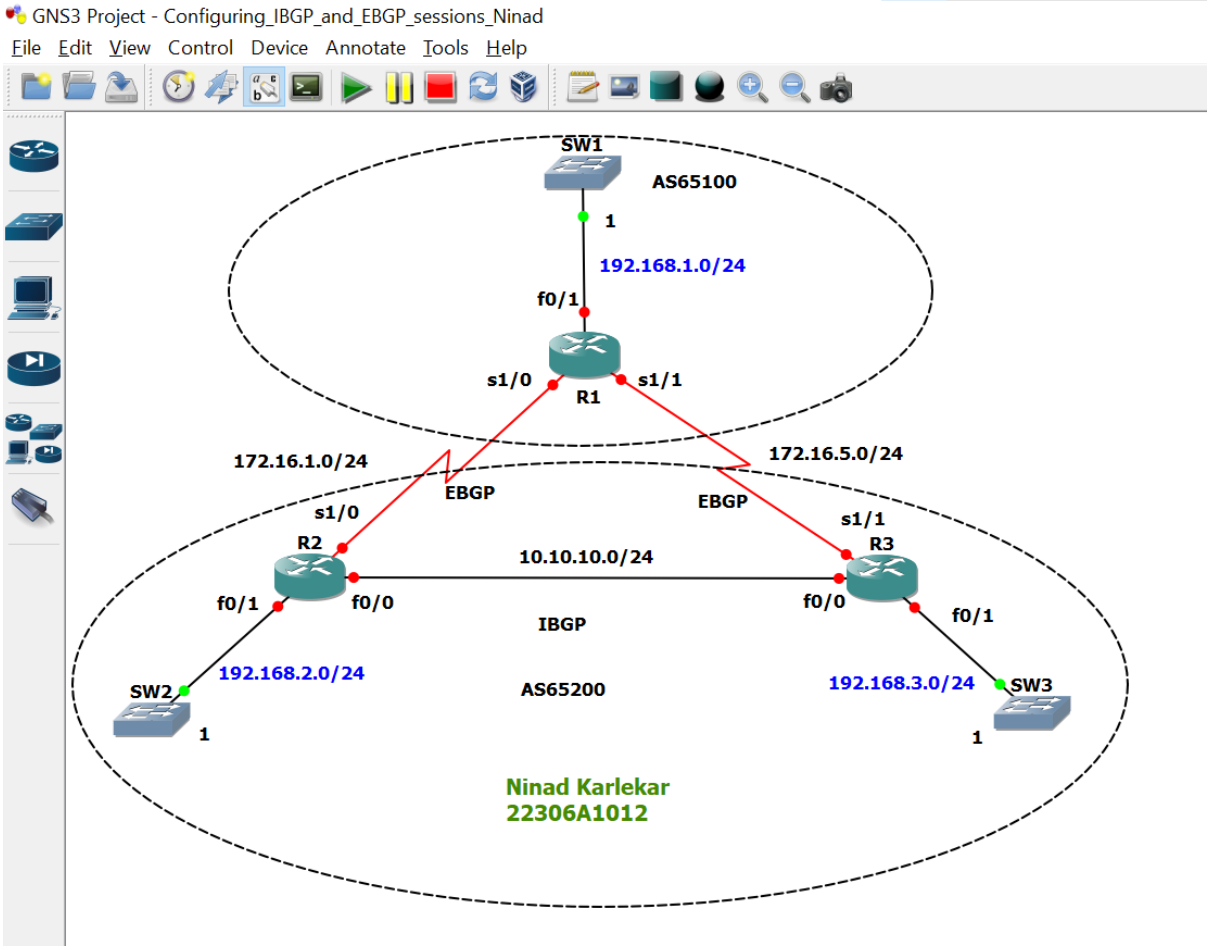


do ping 10.2.2.2 source lo0



Practical No: 3 Ninad Karlekar 22306A1012 Date: 08/05/2023

Aim: Configure IP SLA tracking and path control topology.



Step 1: Configure IP addresses on the given routers

R1:

```
R1#conf t
R1(config)#int f0/1
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 s1/0
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 s1/1
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
R1(config)#int f0/1
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
R1(config-if)#
*Mar  1 00:02:43.203: %LINK-3-UPDOWN: Interface
tate to up
*Mar  1 00:02:44.203: %LINEPROTO-5-UPDOWN: Line
Ethernet0/1, changed state to up
R1(config-if)#int s1/0
R1(config-if)#ip add 172.16.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
R1(config-if)#
*Mar  1 00:03:07.383: %LINK-3-UPDOWN: Interface
o up
R1(config-if)#
*Mar  1 00:03:08.387: %LINEPROTO-5-UPDOWN: Line
al1/0, changed state to up
R1(config-if)#int s1/1
R1(config-if)#ip add 172.16.5.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
```

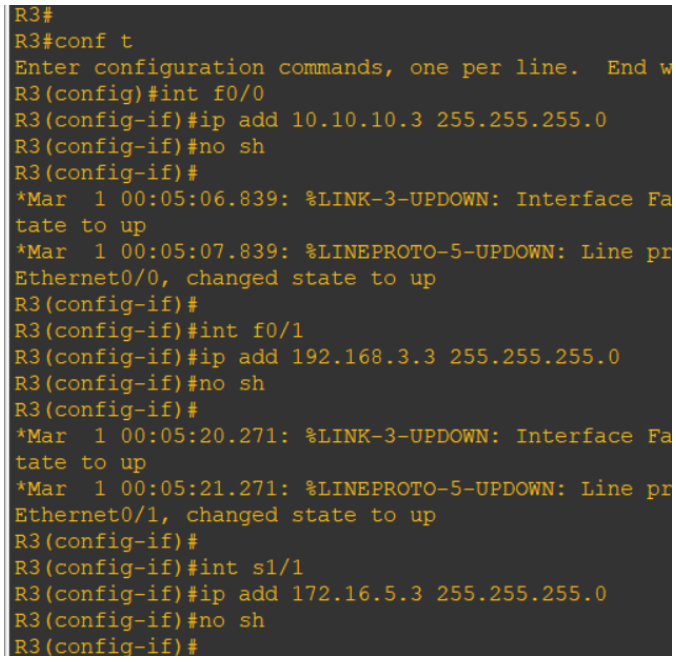
R2:

R2#conf t
R2(config)#int f0/0
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 f0/1
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 s1/0
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
R2(config)#int f0/0
R2(config-if)#ip add 10.10.10.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
R2(config-if)#
R2(config-if)#
*Mar  1 00:04:25.311: %LINK-3-UPDOWN: Interface
tate to up
*Mar  1 00:04:26.311: %LINEPROTO-5-UPDOWN: Line
Ethernet0/0, changed state to up
R2(config-if)#int f0/1
R2(config-if)#ip add 192.168.2.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
*Mar  1 00:04:39.655: %LINK-3-UPDOWN: Interface
tate to up
*Mar  1 00:04:40.655: %LINEPROTO-5-UPDOWN: Line
Ethernet0/1, changed state to up
R2(config-if)#
R2(config-if)#int s1/0
R2(config-if)#ip add 172.16.1.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

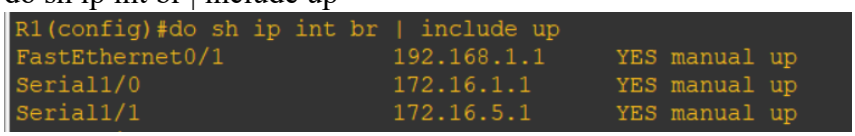
R3:

R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#int f0/0
R3(config-if)#ip add 10.10.10.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
*Mar 1 00:05:06.839: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar 1 00:05:07.839: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R3(config-if)#
R3(config-if)#int f0/1
R3(config-if)#ip add 192.168.3.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
*Mar 1 00:05:20.271: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar 1 00:05:21.271: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
R3(config-if)#
R3(config-if)#int s1/1
R3(config-if)#ip add 172.16.5.3 255.255.255.0
R3(config-if)#no sh



On all routers:

do sh ip int br | include up



```
R2(config-if)#do sh ip int br | include up
FastEthernet0/0            10.10.10.2      YES manual up
FastEthernet0/1            192.168.2.2     YES manual up
Serial1/0                  172.16.1.2      YES manual up
```

```
R3(config-if)#do sh ip int br | include up
FastEthernet0/0            10.10.10.3      YES manual up
FastEthernet0/1            192.168.3.3     YES manual up
Serial1/1                  172.16.5.3      YES manual up
```

Step 2: Configure IRP in autonomous system 65200

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
```

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)#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
```

do ping 192.168.2.2

```
R3(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 = 24/29/32 ms
R3(config-router)#
```

Step 3: IBGP & EBGP configuration

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)#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: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter are
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type
       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
-2
       ia - IS-IS inter area, * - candidate default, U - per-user st
oute
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.5.0 is directly connected, Serial1/1
C       172.16.1.0 is directly connected, Serial1/0
C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

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)#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)#
*Mar  1 01:32:53.123: %BGP-5-ADJCHANGE: neighbor 172.16.1.1 Up
R2(config-router)#do sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external ty
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS
-2
       ia - IS-IS inter area, * - candidate default, U - per-user
oute
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.1.0 is directly connected, Serial1/0
     10.0.0.0/24 is subnetted, 1 subnets
C       10.10.10.0 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/1
O IA 192.168.3.0/24 [110/20] via 10.10.10.3, 01:23:07, FastEthern
R2(config-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)#neighbor 10.10.10.2 remote-as 65200
R3(config-router)#do sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS lev
-2
       ia - IS-IS inter area, * - candidate default, U - per-user static
oute
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.5.0 is directly connected, Serial1/1
     10.0.0.0/24 is subnetted, 1 subnets
C       10.10.10.0 is directly connected, FastEthernet0/0
O IA 192.168.2.0/24 [110/20] via 10.10.10.2, 01:23:58, FastEthernet0/0
C    192.168.3.0/24 is directly connected, FastEthernet0/1
R3(config-router)#
*Mar  1 01:33:22.495: %BGP-5-ADJCHANGE: neighbor 10.10.10.2 Up
R3(config-router)#
```

R1:

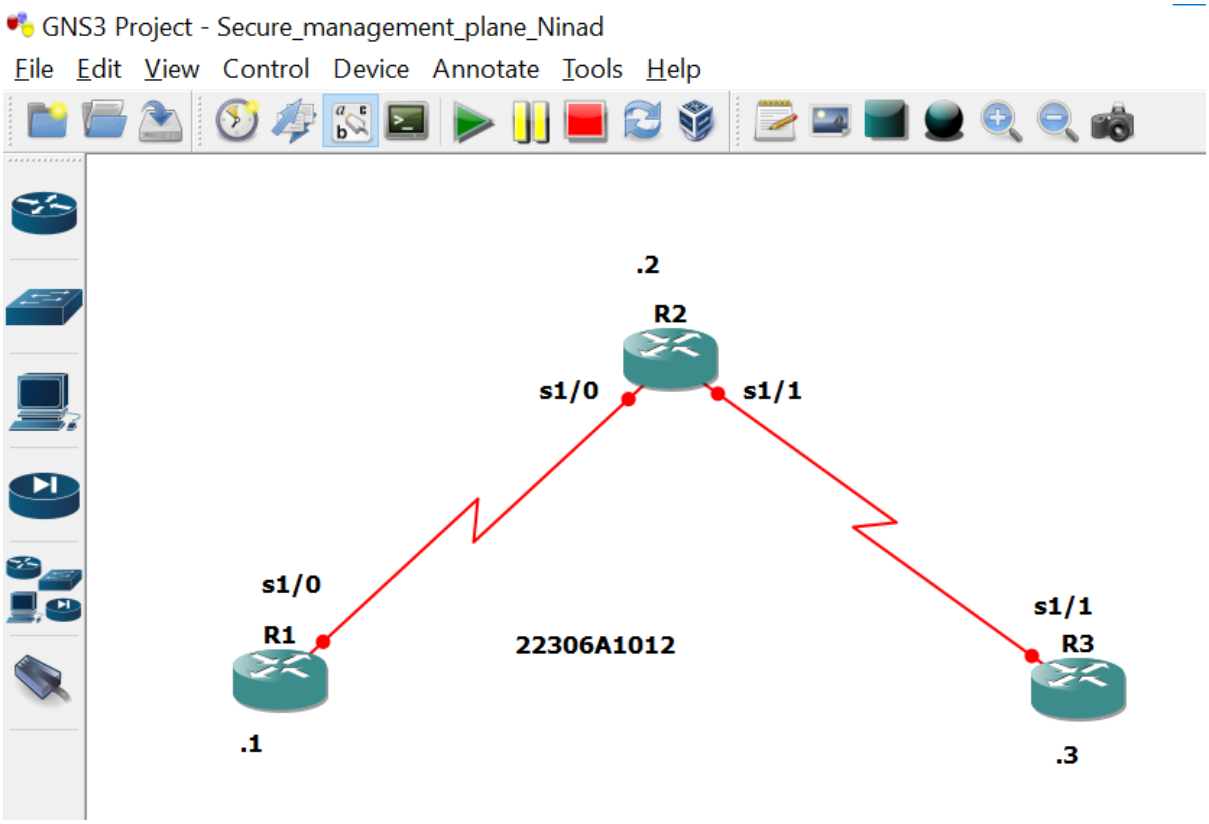
do ping 192.168.3.3

do ping 192.168.2.2

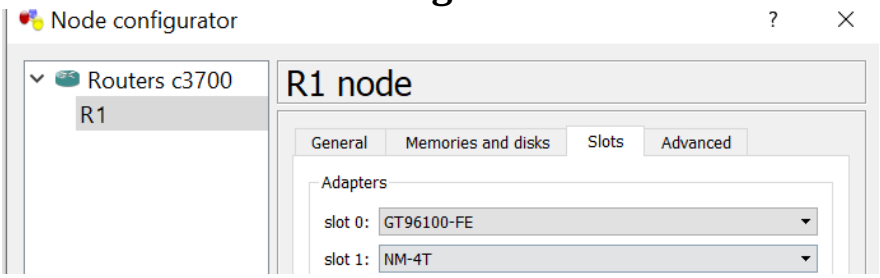
```
R1(config-router)#
R1(config-router)#do ping 192.168.3.3

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 = 20/28/32 ms
R1(config-router)#
R1(config-router)#
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 = 28/29/32 ms
R1(config-router)#
R1#
*Mar  1 01:53:58.735: %SYS-5-CONFIG_I: Configured from console by console
R1#
```

Practical No: 4 Ninad Karlekar 22306A1012 Date: 14/03/2023

Aim: Secure management plane.



Take 3 routers -> Configure -> slots -> NM-4T



R1 Console

R1 #
R1 # conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config) # int s1/0
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#
R1#conf t
Enter configuration commands, one per line.  End with CNTI
R1(config)#int s1/0
R1(config-if)#ip add 10.1.1.1 255.255.255.0
R1(config-if)#no sh
R1(config-if)#
R1(config-if)#
*Mar  1 00:02:09.687: %LINK-3-UPDOWN: Interface Serial1/0,
R1(config-if)#
*Mar  1 00:02:10.691: %LINEPROTO-5-UPDOWN: Line protocol c
up
R1(config-if)#int lo0
R1(config-if)#
*Mar  1 00:02:24.167: %LINEPROTO-5-UPDOWN: Line protocol c
up
R1(config-if)#ip add 192.168.1.1 255.255.255.0
*Mar  1 00:02:33.059: %LINEPROTO-5-UPDOWN: Line protocol c
down
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#
```

R2 Console

R2 # conf t
R2(config) # int s1/0
R2(config-if) # ip add 10.1.1.2 255.255.255.0
R2(config-if) # no sh

R2(config-if) # int s1/1
R2(config-if) # ip add 10.2.2.2 255.255.255.0
R2(config-if) # no sh
R2(config-if) #

```
R2#
R2#conf t
Enter configuration commands, one per line.  End w
R2(config)#int s1/0
R2(config-if)#ip add 10.1.1.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
R2(config-if)#
*Mar  1 00:03:41.095: %LINK-3-UPDOWN: Interface Se
R2(config-if)#int
*Mar  1 00:03:42.099: %LINEPROTO-5-UPDOWN: Line pr
up
R2(config-if)#int s1/1
R2(config-if)#ip add 10.2.2.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
```

R3 Console

R3 # conf t
R3(config) # int s1/1
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#conf t
Enter configuration commands, one per line.  End wit
R3(config)#int s1/1
R3(config-if)#ip add 10.2.2.3 255.255.255.0
R3(config-if)#no sh
R3(config-if)#
R3(config-if)#
*Mar  1 00:10:15.255: %LINK-3-UPDOWN: Interface Seri
R3(config-if)#
*Mar  1 00:10:16.259: %LINEPROTO-5-UPDOWN: Line prot
up
R3(config-if)#int lo0
R3(config-if)#
*Mar  1 00:10:23.203: %LINEPROTO-5-UPDOWN: Line prot
up
R3(config-if)#ip add 192.168.3.3 255.255.255.0
```

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

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

```
R2(config)#
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)#
```

R3 Console

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

```
R3(config-if)#exit
R3(config)#
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)#
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 = 40/56/64 ms
R1(config)#
```

R3 Console

R3(config) # do ping 192.168.1.1

```
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 = 20/50/80 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 # 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

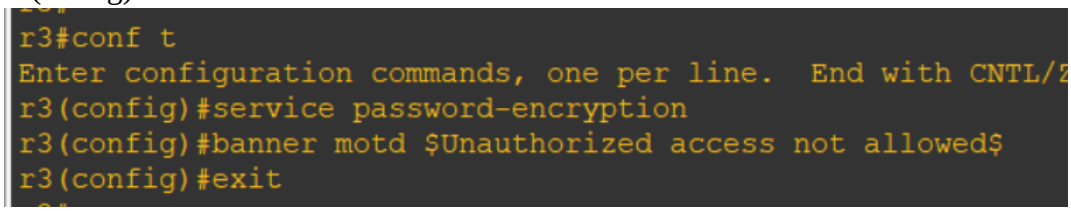
```
R1(config)#
R1(config)#hostname r1
r1(config)#security password min-length 10
r1(config)#enable secret class12345
r1(config)#
r1(config)#line console 0
r1(config-line)#password ciscoconpass
r1(config-line)#exec-timeout 5 0
r1(config-line)#login
r1(config-line)#logging synchronous
r1(config-line)#exit
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#
```

```
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

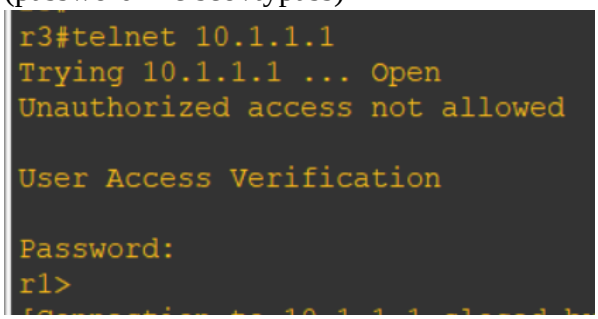
r3(config) # security password min-length 10
r3(config) # enable secret class12345
r3(config) # line console 0
r3(config-line) # password ciscoconpass
r3(config-line) # exec-timeout 5 0
r3(config-line) # login
r3(config-line) # logging synchronous
r3(config-line) # exit
r3(config) # line vty 0 4
r3(config-line) # 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 Console

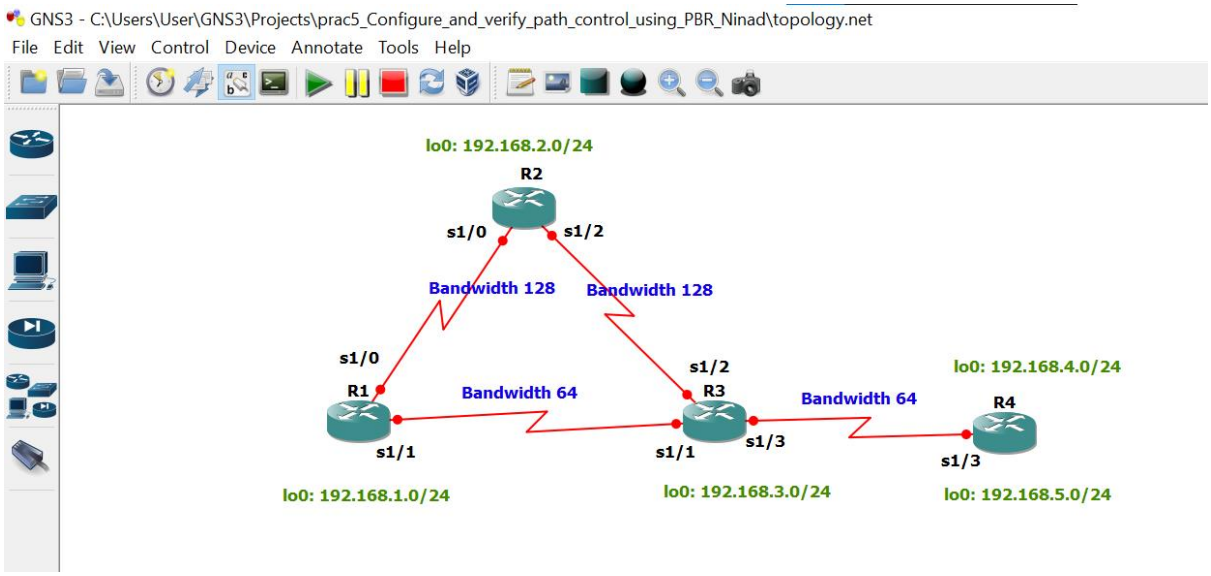
r3 # telnet 10.1.1.1

(password-> ciscovtypass)

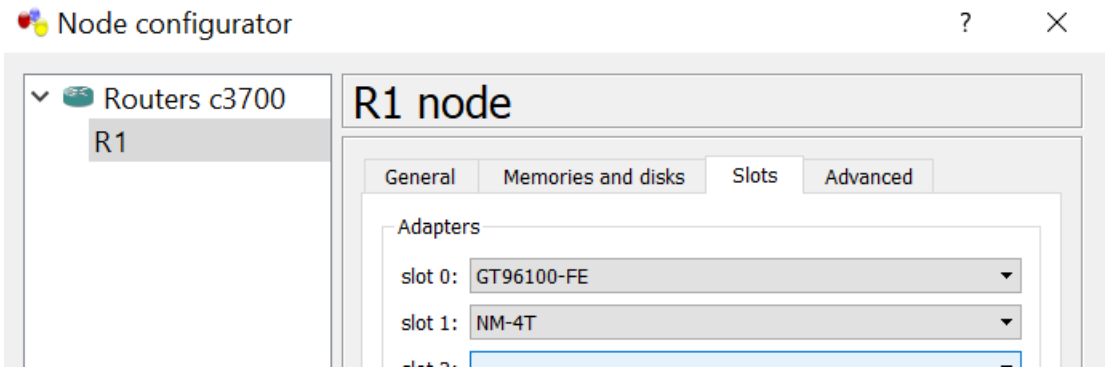


Practical No: 5 Ninad Karlekar 22306A1012 Date: 18/04/2023

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



Take 4 routers -> Configure -> slots -> NM-4T



STEP 1: Perform IP configuration

On router 1 console

R1 # conf t
R1(config) # hostname r1

```
R1#conf t
Enter configuration commands,
R1(config)#hostname r1
r1(config)#
```

r1(config) # int s1/0
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)#
r1(config)#int s1/0
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 s1/1
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 s1/1
r1(config-if)#no sh
*Mar  1 00:05:22.339: %LINEPROTO-5-UPDOWN: Line p
down
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

```
down
r1(config-if)#int lo0
r1(config-if)#ip ad
*Mar  1 00:06:47.047: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, cha
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
Serial1/0                  172.16.12.1     YES manual up                    down
Serial1/1                  172.16.13.1     YES manual up                    down
Loopback0                  192.168.1.1     YES manual up                    up
r1(config-if)#
```

On router 2 console

R2 # conf t
R2(config) # hostname r2

```
R2#
R2#conf t
Enter configuration commands, one pe
R2(config)#hostname r2
r2(config)#
```

r2(config) #
r2(config) # int s1/0
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)#
r2(config)#int s1/0
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 s1/2
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)#
r2(config-if)#int s1/2
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)#
*Mar  1 00:16:58.211: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, cha
r2(config-if)#ip add
*Mar  1 00:17:02.759: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1/2, cha
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
Serial1/0                  172.16.12.2     YES manual up                    up
Serial1/2                  172.16.23.2     YES manual up                    down
Loopback0                  192.168.2.2     YES manual up                    up
r2(config-if)#
r2(config-if)#
```

On router 3 console

R3 # conf t
R3(config) # hostname r3
r3(config) #

```
R3#
R3#conf t
Enter configuration commands,
R3(config)#hostname r3
r3(config)#
```

r3(config) # int s1/1
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)#
r3(config)#int s1/1
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 s1/2
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 s1/2
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) # int s1/3
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)#
r3(config-if)#int s1/3
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 a
*Mar  1 00:27:31.031: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, cha
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
Serial1/1                  172.16.13.3     YES manual up                    up
Serial1/2                  172.16.23.3     YES manual up                    up
Serial1/3                  172.16.34.3     YES manual up                    down
Loopback0                  192.168.3.3     YES manual up                    up
r3(config-if)#
```

On router 4 console

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

```
R4#
R4#conf t
Enter configuration commands, one per
R4(config)#hostname r4
r4(config)#
```

r4(config) # int s1/3
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)#
r4(config)#int s1/3
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) # int lo1
r4(config-if) # ip add 192.168.4.1 255.255.255.0
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)#
r4(config-if)#int lo0
r4(config-if)#ip add
*Mar  1 00:31:06.559: %LINEPROTO-5-UPDOWN: Line pro
r4(config-if)#ip add 192.168.4.1 255.255.255.0
r4(config-if)#
```

```
r4(config-if)#
r4(config-if)#int lo1
r4(config-if)#ip add 192.168.4.1 255.255.255.0
*Mar  1 00:31:38.855: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1, ch
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
Serial1/3                  172.16.34.4     YES manual up                    up
Loopback0                  192.168.4.1     YES manual up                    up
Loopback1                  192.168.5.1     YES manual up                    up
r4(config-if)#
```

STEP 2 : Configure eigrp on all routers

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)#exit
r1(config)#
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-router)#
r1(config-router)#
*Mar  1 00:42:43.707: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbo
```

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)#
r2(config)#router eigrp 1
r2(config-router)#network 172.16.12.0 0.0.0.255
r2(config-router)#
*Mar  1 00:44:06.927: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighb
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)#
*Mar  1 00:44:54.415: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighb
```

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-if)#
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
*Mar  1 00:45:23.543: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16
r3(config-router)#network 172.16.23.0 0.0.0.255
r3(config-router)#
*Mar  1 00:45:32.191: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16
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)#
*Mar  1 00:46:07.631: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16
```

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)#
r4(config)#router eigrp 1
r4(config-router)#network 172.16.34.0 0.0.0.255
r4(config-router)#
*Mar  1 00:44:34.247: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neig
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)#
```

STEP 3: Command on all routers

do sh ip route

r4(config) # do ping 192.168.1.1

```
r4(config)#
r4(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 = 44/66/96 ms
r4(config)#
r4(config)#
```

r1(config) # do ping 192.168.4.1

```
r1(config)#
r1(config)#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 = 60/61/64 ms
r1(config)#
r1(config)#
```

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)#
r4(config)#do 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 32 msec 28 msec 28 msec
  2 172.16.23.2 60 msec 60 msec 64 msec
  3 172.16.12.1 72 msec 88 msec 80 msec
r4(config)#
r4(config)#
```

```
  3 172.16.12.1 72 msec 88 msec 80 msec
r4(config)#
r4(config)#do traceroute 192.168.1.1 source 192.168.5.1

Type escape sequence to abort.
Tracing the route to 192.168.1.1

  1 172.16.34.3 12 msec 48 msec 24 msec
  2 172.16.23.2 68 msec 52 msec 68 msec
  3 172.16.12.1 56 msec 88 msec 68 msec
r4(config)#
r4(config)#
```

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) #
r3(config-route-map) # set ip next-hop 172.16.13.1
r3(config-route-map) # exit
r3(config) #
r3(config) # int s1/3
r3(config-if) # ip policy route-map r3-to-r1
r3(config-if) # end

```
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)#route-map r3-to-r1 permit
r3(config-route-map)#match ip address pbr-acl
r3(config-route-map)#
r3(config-route-map)#set ip next-hop 172.16.13.1
r3(config-route-map)#exit
r3(config)#
r3(config)#int s1/3
r3(config-if)#ip policy route-map r3-to-r1
r3(config-if)#end
r3#
```

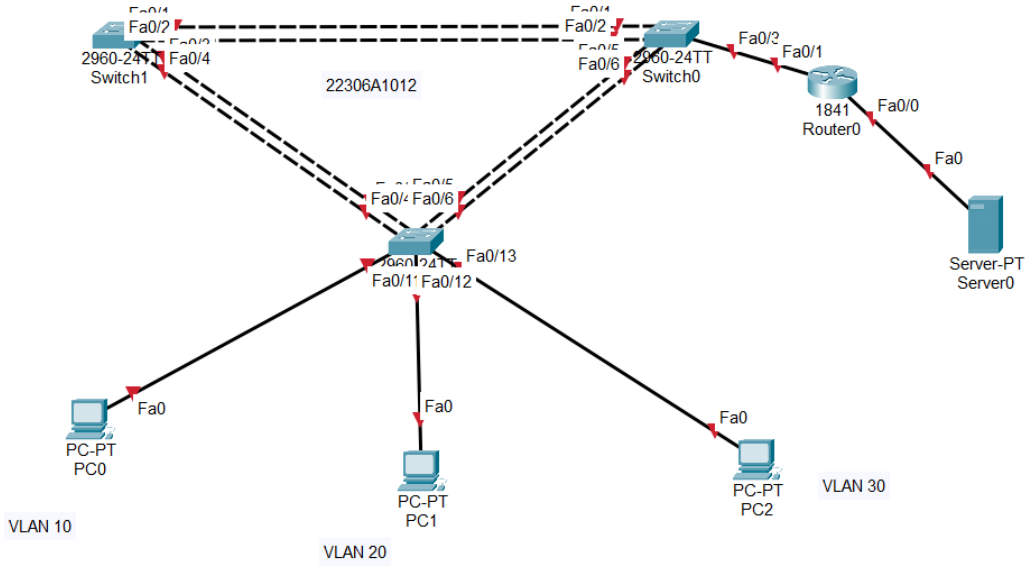
On router 4 console

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)#do 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 36 msec 44 msec 28 msec
  2 172.16.23.2 28 msec 44 msec 48 msec
  3 172.16.12.1 84 msec 64 msec 92 msec
r4(config)#
r4(config)#do traceroute 192.168.1.1 source 192.168.5.1

Type escape sequence to abort.
Tracing the route to 192.168.1.1

  1 172.16.34.3 20 msec 28 msec 32 msec
  2 172.16.13.1 32 msec 60 msec 64 msec
r4(config)#
r4(config)#
r4(config)#Ninad Karlekar 22306A1012
           ^
% Invalid input detected at '^' marker.
```

Practical No: 6 Ninad Karlekar 22306A1012 Date: 11/04/2023

AIM: Demonstrate inter vlan routing.



Task1: check VLAN config in each switch

type command for all switches:

en
show vlan br
CHECK IF ALL SWITCHES HAVE SAME VLAN (1002,1003,1004,1005...)

```
Switch>en
Switch#sh vlan br

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                 Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                 Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                 Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                 Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                 Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                                 Gig0/1, Gig0/2
1002 fddi-default                      active
1003 token-ring-default                active
1004 fddinet-default                   active
1005 trnet-default                     active
Switch#
Switch#
```

Task2: disable all ports on all the switches

commands for all switches:

conf t
interface range fa0/1-24
shutdown
interface range gi0/1-2
shutdown

```
Switch#
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#interface range fa0/1-24
Switch(config-if-range)#shutdown



%LINK-5-CHANGED: Interface FastEthernet0/3, changed state to administratively down

%LINK-5-CHANGED: Interface FastEthernet0/4, changed state to administratively down
```

```
Switch(config-if-range)#
Switch(config-if-range)#interface range gi0/1-2
Switch(config-if-range)#shutdown

%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to administratively down

%LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to administratively down
Switch(config-if-range)#
Switch(config-if-range)#
```

Task3: Perform basic switch configurations like assign name to switches, password to switches as well as gateways.

commands for all switches:

exit
(config)
hostname s0
enable secret class
no ip domain-lookup
ip default-gateway 172.17.99.1
line console 0
(config-line)
password cisco
login
line vty 0 15
password cisco
login
end

```
Switch(config-if-range)#
Switch(config-if-range)#exit
Switch(config)#hostname s0
s0(config)#enable secret class
s0(config)#no ip domain-lookup
s0(config)#ip default-gateway 172.17.99.1
s0(config)#line console 0
s0(config-line)#password cisco
s0(config-line)#login
s0(config-line)#line vty 0 15
s0(config-line)#password cisco
s0(config-line)#login
s0(config-line)#end
s0#
```

Task4: On the interfaces of the switch 2 connect it to the PCs, configure access mode and enable

commands for s2:

(config)
int fa0/11
(config-if)
switchport mode access
no shutdown

int fa0/12
switchport mode access
no shutdown

int fa0/13
switchport mode access
no shutdown

```
s2#conf t
Enter configuration commands, one per line.  End with CN
s2(config)#int fa0/11
s2(config-if)#switchport mode access
s2(config-if)#no shutdown
```

```
s2(config-if)#int fa0/12
s2(config-if)#switchport mode access
s2(config-if)#no shutdown

s2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet
```

```
s2(config-if)#int fa0/13
s2(config-if)#switchport mode access
s2(config-if)#no shutdown

s2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0

%LINEPROTO-5-UPDOWN: Line protocol on In
state to up
```

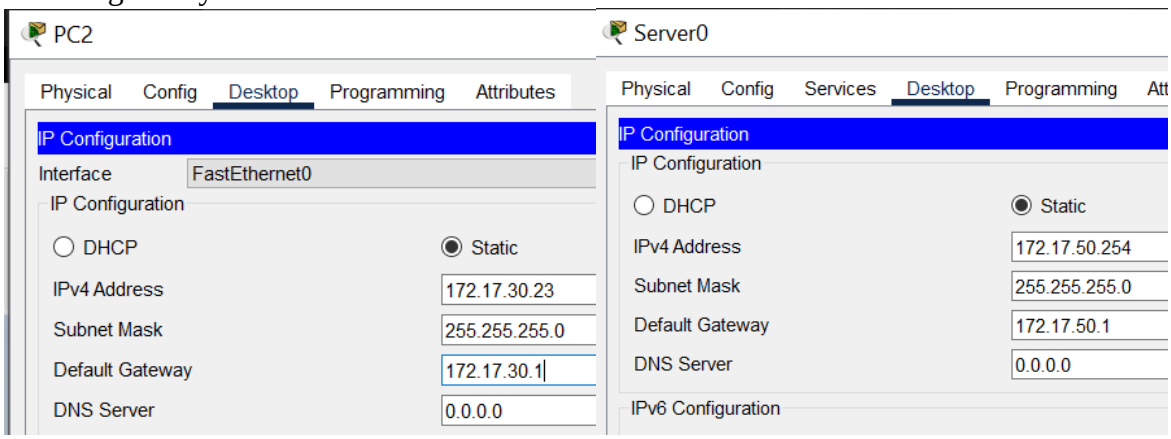
Task5: Configure IP addresses on the three PCs and the server

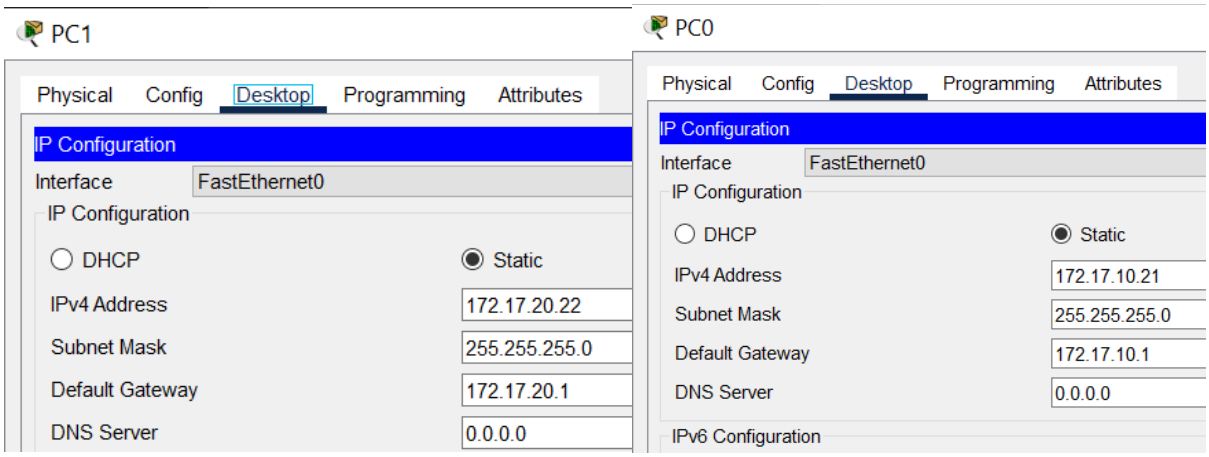
PC0-> Desktop -> IP config
IP: 172.17.10.21 255.255.255.0
Default gateway: 172.17.10.1

PC1-> Desktop -> IP config
IP: 172.17.20.22 255.255.255.0
Default gateway: 172.17.20.1

PC2-> Desktop -> IP config
IP: 172.17.30.23 255.255.255.0
Default gateway: 172. 17.30.1

Server -> Desktop -> IP config
IP: 172.17.50.254 255.255.255.0
Default gateway: 172.17.50.1



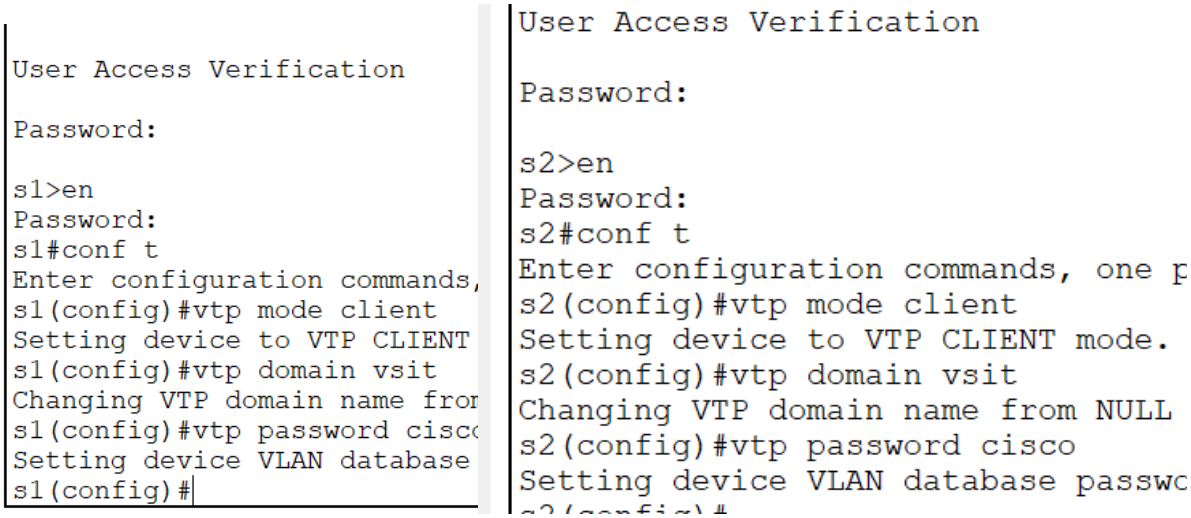


Task6: Configure VTP protocol on the switches.

s0 will be VTP server, s1 & s2 will be VTP client

s0:
Password: class
en
(#)
Password:
conf t
(config)
vtp mode server
vtp domain vsit
vtp password cisco
s1:
Password:
en
#
Password:
conf t
(config)
vtp mode client
vtp domain vsit
vtp password cisco
s2:
Password:
en
#
Password:
conf t
(config)
vtp mode client
vtp domain vsit
vtp password cisco

```
s0>en
Password:
s0#conf t
Enter configuration commands, c
s0(config)#vtp mode server
Device mode already VTP SERVER.
s0(config)#vtp domain vsit
Changing VTP domain name from N
s0(config)#vtp password cisco
Setting device VLAN database pa
s0(config)#
```



Task7: Configure trunking codes on all connections between switches and enable them

s0:
(config)
int range fa0/1-3
(config-if)
switchport mode trunk
switchport trunk native vlan 99
no shutdown

int range fa0/5-6
switchport mode trunk
switchport trunk native vlan 99
no shutdown

s2:
(config)
int range fa0/3-6
(config-if)
switchport mode trunk
switchport trunk native vlan 99
no sh

s1:
(config)
int range fa0/1-4
(config-if)
switchport mode trunk
switchport trunk native vlan 99

no sh

s0:
(config-if-range)
exit
(config)
vlan 99
name management
(config-vlan)
vlan 10
name staff
vlan 20
name students
vlan 30
name guests

exit
do sh vlan br (On s0 and s2)

```
s0(config)#
s0(config)#
s0(config)#int range fa0/1-3
s0(config-if-range)#switchport mode trunk
s0(config-if-range)#switchport trunk native vlan 99
s0(config-if-range)#no shutdown
```

```
s1(config)#
s1(config)#int range fa0/1-4
s1(config-if-range)#switchport mode trunk
s1(config-if-range)#switchport trunk native vlan 99
s1(config-if-range)#no sh
```

```
Setting device VLAN database password to cisco
s2(config)#
s2(config)#
s2(config)#int range fa0/3-6
s2(config-if-range)#switchport mode trunk
s2(config-if-range)#switchport trunk native vlan 99
s2(config-if-range)#no sh
```

```
s0(config)#vlan 99
s0(config-vlan)#name management
s0(config-vlan)#vlan 10
s0(config-vlan)#name staff
s0(config-vlan)#vlan 20
s0(config-vlan)#name students
s0(config-vlan)#vlan 30
s0(config-vlan)#name guests
s0(config-vlan)#exit
s0(config)#do sh vlan br
```

```
s0(config)#do sh vlan br

VLAN Name                              Status    Ports
---- --------------------------------- ---------
------------------------------
1    default                           active    Fa0/3, Fa0/4, Fa0/7,
Fa0/8
                                                 Fa0/9, Fa0/10, Fa0/11,
Fa0/12
                                                 Fa0/13, Fa0/14, Fa0/15,
Fa0/16
                                                 Fa0/17, Fa0/18, Fa0/19,
Fa0/20
                                                 Fa0/21, Fa0/22, Fa0/23,
Fa0/24
                                                 Gig0/1, Gig0/2
10   staff                             active
20   students                          active
30   guests                            active
99   management                        active
1002 fddi-default                      active
1003 token-ring-default                active
1004 fddinet-default                   active
1005 trnet-default                     active
s0(config)#
```

```
s2(config-if-range)#exit
s2(config)#do sh vlan br

VLAN Name                              Status    Ports
---- --------------------------------- ---------
------------------------------
1    default                           active    Fa0/1, Fa0/2, Fa0/7,
Fa0/8
                                                 Fa0/9, Fa0/10, Fa0/11,
Fa0/12
                                                 Fa0/13, Fa0/14, Fa0/15,
Fa0/16
                                                 Fa0/17, Fa0/18, Fa0/19,
Fa0/20
                                                 Fa0/21, Fa0/22, Fa0/23,
Fa0/24
                                                 Gig0/1, Gig0/2
10   staff                             active
20   students                          active
30   guests                            active
99   management                        active
1002 fddi-default                      active
1003 token-ring-default                active
1004 fddinet-default                   active
1005 trnet-default                     active
s2(config)#
```

Task8: Configure interface vlan 99 on all the switches

s0:
(config)
int vlan 99
(config-if)
ip add 172.17.99.11 255.255.255.0

end

s2:
(config)
int vlan 99
(config-if)
ip add 172.17.99.12 255.255.255.0
end

s1:
(config)
int vlan 99
(config-if)
ip add 172.17.99.13 255.255.255.0
end

```
s0(config)#
s0(config)#int vlan 99
s0(config-if)#
%LINK-5-CHANGED: Interface Vlan99, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, ch
up

s0(config-if)#ip add 172.17.99.11 255.255.255.0
s0(config-if)#end
s0#
%SYS-5-CONFIG_I: Configured from console by console

s0#
```

```
1005 trnet-default                   active
s2(config)#
s2(config)#int vlan 99
s2(config-if)#
%LINK-5-CHANGED: Interface Vlan99, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state to
up

s2(config-if)#ip add 172.17.99.12 255.255.255.0
s2(config-if)#end
s2#
%SYS-5-CONFIG_I: Configured from console by console

s2#
```

```
s1(config)#
s1(config)#
s1(config)#int vlan 99
s1(config-if)#
%LINK-5-CHANGED: Interface Vlan99, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99,
up

s1(config-if)#ip add 172.17.99.13 255.255.255.0
s1(config-if)#end
s1#
%SYS-5-CONFIG_I: Configured from console by console

s1#
```

Task9: Configure vlan 10, vlan 20 and vlan 30 on switch 2

s2:

(config)
int fa0/11
(config-if)
switchport access vlan 10
int fa0/12
switchport access vlan 20
int fa0/13
switchport access vlan 30

```
s2#conf t
Enter configuration commands, one per line
s2(config)#int fa0/11
s2(config-if)#switchport access vlan 10
s2(config-if)#int fa0/12
s2(config-if)#switchport access vlan 20
s2(config-if)#int fa0/13
s2(config-if)#switchport access vlan 30
s2(config-if)#
```

Task10: perform configuration on router

Router:
en
conf t
hostname r1
no ip domain-lookup
line console 0
(config-line)
password cisco
login
line vty 0 15
password cisco
login
end

conf t
(config)
enable secret class
int fa0/1
no sh

int fa0/1.1
(config-subif)
encapsulation dot1q 1
ip add 172.17.1.1 255.255.255.0
int fa0/1.10
encapsulation dot1q 10
ip add 172.17.10.1 255.255.255.0
int fa0/1.20
encapsulation dot1q 20

ip add 172.17.20.1 255.255.255.0
int fa0/1.30
encapsulation dot1q 30
ip add 172.17.30.1 255.255.255.0
int fa0/1.99
encapsulation dot1q 99 native
ip add 172.17.99.1 255.255.255.0

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/
Router(config)#hostname r1
r1(config)#no ip domain-lookup
r1(config)#line console 0
r1(config-line)#password cisco
r1(config-line)#login
r1(config-line)#line vty 0 15
r1(config-line)#password cisco
r1(config-line)#login
r1(config-line)#end
```

```
r1#
r1#conf t
Enter configuration commands, one per l
r1(config)#enable secret class
r1(config)#int fa0/1
r1(config-if)#no sh
```

```
r1(config-if)#int fa0/1.1
r1(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/1.1, change

%LINEPROTO-5-UPDOWN: Line protocol on Interface Fastl
changed state to up

r1(config-subif)#encapsulation dot1q 1
r1(config-subif)#ip add 172.17.1.1 255.255.255.0
r1(config-subif)#int fa0/1.10
r1(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/1.10, chang

%LINEPROTO-5-UPDOWN: Line protocol on Interface Fastl
changed state to up

r1(config-subif)#encapsulation dot1q 10
r1(config-subif)#ip add 172.17.10.1 255.255.255.0
r1(config-subif)#int fa0/1.20
r1(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/1.20, chang

%LINEPROTO-5-UPDOWN: Line protocol on Interface Fastl
changed state to up

r1(config-subif)#encapsulation dot1q 20
r1(config-subif)#ip add 172.17.20.1 255.255.255.0
```

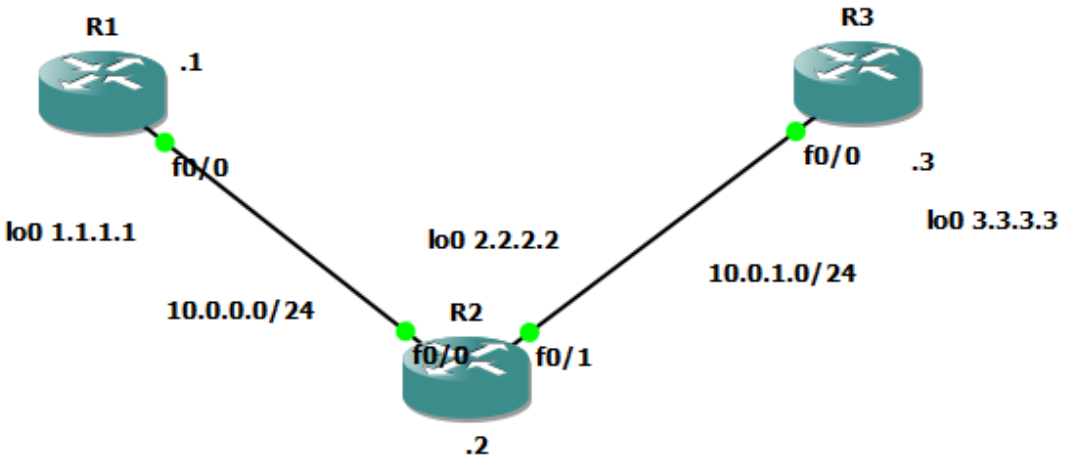
```
r1(config-subif)#encapsulation dot1q 30
r1(config-subif)#ip add 172.17.30.1 255.255.255.0
r1(config-subif)#int fa0/1.99
r1(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/1.99, changed s

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthe
changed state to up

r1(config-subif)#encapsulation dot1q 99 native
r1(config-subif)#ip add 172.17.99.1 255.255.255.0
r1(config-subif)#exit
r1(config)#
```

```
r1(config)#do sh ip int br
Interface              IP-Address      OK? Method Status
Protocol
FastEthernet0/0        unassigned      YES unset  administratively down
down
FastEthernet0/1        unassigned      YES unset  up
up
FastEthernet0/1.1      172.17.1.1      YES manual up
up
FastEthernet0/1.10     172.17.10.1     YES manual up
up
FastEthernet0/1.20     172.17.20.1     YES manual up
up
FastEthernet0/1.30     172.17.30.1     YES manual up
up
FastEthernet0/1.99     172.17.99.1     YES manual up
up
Vlan1                  unassigned      YES unset  administratively down
down
r1(config)#
```

Practical No: 7 **Date: 08/05/2023**

Aim: Simulating MP LS environment.

Topology:



Task 1: IP addressing of MPLS Core and OSPF

R1 console:

```
R1(config)#int lo0
R1(config-if)#ip add
*Mar  1 00:02:00.015: %LINEPROTO-5-UPDOWN: Line proto
R1(config-if)#ip add 1.1.1.1 255.255.255.255
R1(config-if)#ip ospf 1 area 0
R1(config-if)#int f0/0
R1(config-if)#ip add 10.0.0.1 255.255.255.0
R1(config-if)#no shut
R1(config-if)#ip ospf 1 aera
```

R2 Console:

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL
R2(config)#hostname R2
R2(config)#int lo0
R2(config-if)#ip
*Mar  1 00:03:33.783: %LINEPROTO-5-UPDOWN: Line protocol o
R2(config-if)#ip add 2.2.2.2 255.255.255.255
R2(config-if)#ip ospf 1 area 0
R2(config-if)#int f0/0
R2(config-if)#ip add 10.0.0.2 255.255.255.0
R2(config-if)#no shut
R2(config-if)#ip ospf 1 a
*Mar  1 00:04:03.991: %LINK-3-UPDOWN: Interface FastEtherr
*Mar  1 00:04:04.991: %LINEPROTO-5-UPDOWN: Line protocol o
R2(config-if)#ip ospf 1 area 0
R2(config-if)#int f0/1
R2(config-if)#ip add 10.0.
*Mar  1 00:04:19.831: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1
R2(config-if)#ip add 10.0.1.2 255.255.255.0
R2(config-if)#no shut
R2(config-if)#ip ospf 1 aea
*Mar  1 00:04:29.655: %LINK-3-UPDOWN: Interface FastEtherr
*Mar  1 00:04:30.655: %LINEPROTO-5-UPDOWN: Line protocol o
R2(config-if)#ip ospf 1 area 0
```

R3 Console:

```
R3#conf t
Enter configuration commands, one per line.  End
R3(config)#hostname R3
R3(config)#int lo0
R3(config-if)#ip ad
*Mar  1 00:04:54.151: %LINEPROTO-5-UPDOWN: Line p
R3(config-if)#ip add 3.3.3.3 255.255.255.255
R3(config-if)#ip ospf 1 area 0
R3(config-if)#int f0/0
R3(config-if)#ip add 10.0.1.3 255.255.255.0
R3(config-if)#no shut
R3(config-if)#ip ospf 1 aera
*Mar  1 00:05:21.451: %LINK-3-UPDOWN: Interface F
*Mar  1 00:05:22.451: %LINEPROTO-5-UPDOWN: Line p
R3(config-if)#ip ospf 1 area 0
```

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

```
R1(config-if)#do ping 3.3.3.3 source lo0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 s
Packet sent with a source address of 1.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max
R1(config-if)#
```

Task 2: Configure LDP on all the interfaces in the MPLS Core

R1 Console:

```
R1(config-if)#router ospf 1
R1(config-router)#mpls ldp autoconfig
R1(config-router)#
```

R2 Console:

```
R2(config-if)#router ospf 1
R2(config-router)#mpls ldp autoconfig
R2(config-router)#
```

R3 Console:

```
R3(config-if)#router ospf 1
R3(config-router)#mpls ldp autoconfig
R3(config-router)#
```

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

- sh mpls interface

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

and using LDP

```
R2#sh mpls interface
Interface              IP            Tunnel   Operational
FastEthernet0/0        Yes (ldp)     No       Yes
FastEthernet0/1        Yes (ldp)     No       Yes
R2#
```

Task 3: MPLS BGP Configuration between R1 and R3

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

R1 Console:

```
R1(config)#router bgp 1
R1(config-router)#neighbor 3.3.3.3 remote-as 1
R1(config-router)#neighbor 3.3.3.3 update-source Loopback
R1(config-router)#no auto-summary
R1(config-router)#!
R1(config-router)#address-family vpnv4
R1(config-router-af)#neighbor 3.3.3.3 activate
R1(config-router-af)#
```

R3 Console:

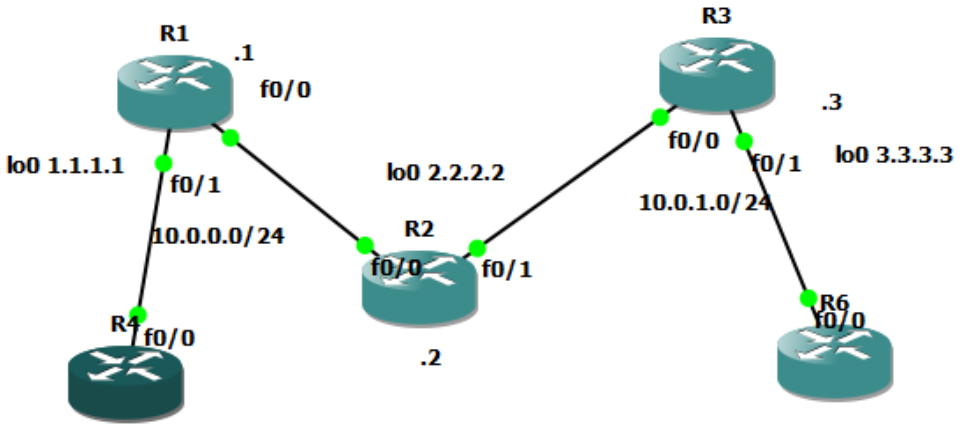
```
R3(config-router)#router bgp 1
R3(config-router)#neighbor 1.1.1.1 remote-as 1
R3(config-router)#neighbor 1.1.1.1 update-source Loopback0
R3(config-router)#no auto-summary
R3(config-router)#!
R3(config-router)#address-family vpnv4
R3(config-router-af)#
*Mar  1 00:25:46.883: %BGP-5-ADJCHANGE: neighbor 1.1.1.1 Up
R3(config-router-af)#neighbor 1.1.1.1 activate
R3(config-router-af)#
```

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

```
R1#sh bgp vpnv4 unicast all summary
BGP router identifier 1.1.1.1, local AS number 1
BGP table version is 1, main routing table version 1

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
3.3.3.3         4     1       7       7        1    0    0 00:01:11        0
R1#
```

Task 4: – Add two more routers, create VRFs

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



R4 Console:

```
R4#conf t
Enter configuration commands, one per line.  End
R4(config)#int lo0
R4(config-if)#ip add
*Mar  1 00:01:01.999: %LINEPROTO-5-UPDOWN: Line
R4(config-if)#ip add 4.4.4.4 255.255.255.255
R4(config-if)#ip ospf 2 area 2
R4(config-if)#int f0/0
R4(config-if)#ip add 192.168.1.4 255.255.255.0
R4(config-if)#ip ospf 2 area 2
R4(config-if)#no shut
R4(config-if)#
```

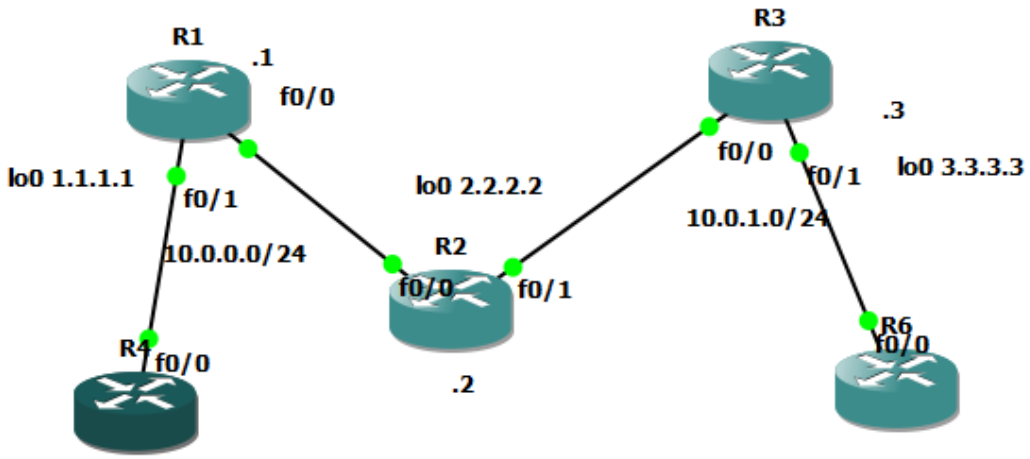
R1 Console:

```
R1(config)#int f0/1
R1(config-if)#no shut
R1(config-if)#ip add 192.16
*Mar  1 00:03:43.271: %LINK-3-UPDOWN: Interface Fas
*Mar  1 00:03:44.271: %LINEPROTO-5-UPDOWN: Line pro
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#
```

Practical No: 8 **Date: 08/05/2023**

Aim: Simulating VRF (Virtual Routing and Forwarding).

Topology:



Task 1: create a VRF on R1

R1 console:

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

```
R1(config-if)#ip vrf RED
R1(config-vrf)#rd 4:4
R1(config-vrf)#route-target bot 4:4
R1(config-vrf)#
```

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

```
R1(config-vrf)#int f0/1
R1(config-if)#ip vrf forwarding RED
% Interface FastEthernet0/1 IP address 192.168.1.1 removed due to enabling VRF RED
R1(config-if)#
```

You just need to re-apply it

```
R1(config-if)#int f0/1
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#
```

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

```
R1#sh run int f0/1
Building configuration...

Current configuration : 119 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 ip address 192.168.1.1 255.255.255.0
 duplex auto
 speed auto
end
```

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

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

```
R1#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static rout
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R1#sh ip route vrf RED

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

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

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

```
Enter configuration commands, one per line.  End with CN
R1(config)#int f0/1
R1(config-if)#ip ospf 2 area 2
R1(config-if)#
*Mar  1 00:22:22.963: %OSPF-5-ADJCHG: Process 2, Nbr 4.4
ne
R1(config-if)#
```


check the routes in the VRF RED routing table you should see

4.4.4.4 in there as well.

```
R1#sh ip route vrf RED

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

Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/11] via 192.168.1.4, 00:01:09, FastEthernet0
C    192.168.1.0/24 is directly connected, FastEthernet0/1
R1#
```


We now need to repeat this process for R3 & R6
Router 6 will peer OSPF using process number 2 to a VRF configured on R3. It will use the local site addressing of 192.168.2.0/24

R6 Console:

```
R6#conf t
Enter configuration commands, one per line.  End with CN
R6(config)#int lo0
R6(config-if)#ip add 6
*Mar  1 00:25:01.455: %LINEPROTO-5-UPDOWN: Line protocol
R6(config-if)#ip add 6.6.6.6 255.255.255.255
R6(config-if)#ip ospf 2 area 2
R6(config-if)#int f0/0
R6(config-if)#ip add 192.168.2.6 255.255.255.0
R6(config-if)#ip ospf 2 area 2
R6(config-if)#no shut
R6(config-if)#
```

R3 Console:

```
R3(config)#ip vrf RED
R3(config-vrf)#rd 4:4
R3(config-vrf)#route-target both 4:4
R3(config-vrf)#
```

```
R3#sh run int f0/1
Building configuration...

Current configuration : 106 bytes
!
interface FastEthernet0/1
 ip vrf forwarding RED
 no ip address
 shutdown
 duplex auto
 speed auto
end

R3#
```

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

```
R3(config-if)#int f0/1
R3(config-if)#ip ospf 2 area 2
R3(config-if)#
```

```
R3#sh ip route vrf RED

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

Gateway of last resort is not set

     6.0.0.0/32 is subnetted, 1 subnets
O       6.6.6.6 [110/11] via 192.168.2.6, 00:00:32, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

Task 2: Get full connectivity across the MPLS core is to redistribute

the routes in OSPF on R1 and R3 into MP-BGP and MP-BGP into OSPF

R4 Console:

```
R4#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loopback0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
```

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

Check the routes on R1:

R1 Console:

```
R1#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     1.0.0.0/32 is subnetted, 1 subnets
C       1.1.1.1 is directly connected, Loopback0
     2.0.0.0/32 is subnetted, 1 subnets
O       2.2.2.2 [110/11] via 10.0.0.2, 00:48:37, FastEthernet0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/21] via 10.0.0.2, 00:45:47, FastEthernet0/0
     10.0.0.0/24 is subnetted, 2 subnets
C       10.0.0.0 is directly connected, FastEthernet0/0
O       10.0.1.0 [110/20] via 10.0.0.2, 00:47:26, FastEthernet0/0
```

we have a VRF configured on this router so this command will show routes in the global routing table (the MPLS Core) and it will not show the 192.168.1.0/24 route as that is in VRF RED – to see that we run the following command

R1 Console:

```
R1#sh ip route vrf RED

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

Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/11] via 192.168.1.4, 00:17:34, FastEthernet0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

Here you can see Routing Table: RED is shown and the routes to R4 are

now visible with 4.4.4.4 being in OSPF.

So we need to do the following;

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

Redistribute OSPF into MP-BGP on R1

```
R1(config)#router bgp 1
R1(config-router)#address-family ipv4 vrf RED
R1(config-router-af)#redistribute ospf 2
```

Redistribute OSPF into MP-BGP on R3

```
R3(config)#router bgp 1
R3(config-router)#address-family ipv4 vrf RED
R3(config-router-af)#redistribute ospf 2
```

This has enabled redistribution of the OSPF routes into BGP. We can check the routes from R4 and R6 are now showing in the BGP table for their VRF with this command:

〉 sh ip bgp vpnv4 vrf RED

```
R1#sh ip bgp vpnv4 vrf RED
BGP table version is 9, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 4:4 (default for vrf RED)
*> 4.4.4.4/32       192.168.1.4             11         32768 ?
*>i6.6.6.6/32       3.3.3.3                 11    100      0 ?
*> 192.168.1.0      0.0.0.0                  0         32768 ?
*>i192.168.2.0      3.3.3.3                  0    100      0 ?
```

Here we can see that 4.4.4.4 is now in the BGP table in VRF RED on R1 with a next hop of 192.168.1.4 (R4) and also 6.6.6.6 is in there as well with a next hop of 3.3.3.3 (which is the loopback of R3 – showing that it is going over the MPLS and R1 is not in the picture)

The same should be true on R3

```
R3#sh ip bgp vpnv4 vrf RED
BGP table version is 9, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 4:4 (default for vrf RED)
*>i4.4.4.4/32       1.1.1.1                 11    100      0 ?
*> 6.6.6.6/32       192.168.2.6             11         32768 ?
*>i192.168.1.0      1.1.1.1                  0    100      0 ?
*> 192.168.2.0      0.0.0.0                  0         32768 ?
```

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

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

R1 Console:

```
R1(config)#router ospf 2
R1(config-router)#redistribute bgp 1 subnets
```

R3 Console:

```
R3(config)#router ospf 2
R3(config-router)#redistribute bgp 1 subnets
```

If all has worked we should be now able to ping 6.6.6.6 from R4
Before we do let's see what the routing table looks like on R4

```
R4#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loopback0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
R4#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loopback0
     6.0.0.0/32 is subnetted, 1 subnets
O IA    6.6.6.6 [110/21] via 192.168.1.1, 00:01:44, FastEthernet0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
O IA 192.168.2.0/24 [110/11] via 192.168.1.1, 00:01:44, FastEthernet0/0
R4#
```

Great we have 6.6.6.6 in there Also, check the routing table on R6

```
R6#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     4.0.0.0/32 is subnetted, 1 subnets
O IA    4.4.4.4 [110/21] via 192.168.2.1, 00:01:25, FastEthernet0/0
     6.0.0.0/32 is subnetted, 1 subnets
C       6.6.6.6 is directly connected, Loopback0
O IA 192.168.1.0/24 [110/11] via 192.168.2.1, 00:01:25, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
```

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

```
R4#ping 6.6.6.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 6.6.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 120/126/144 ms
R4#
```

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

```
R4#trace 6.6.6.6

Type escape sequence to abort.
Tracing the route to 6.6.6.6

  1 192.168.1.1 20 msec 24 msec 32 msec
  2 10.0.0.2 [MPLS: Labels 17/19 Exp 0] 128 msec 120 msec 120 msec
  3 192.168.2.1 [MPLS: Label 19 Exp 0] 100 msec 88 msec 96 msec
  4 192.168.2.6 124 msec 140 msec 156 msec
R4#
```