

CCNA SP ----- SPNG2 Workbook

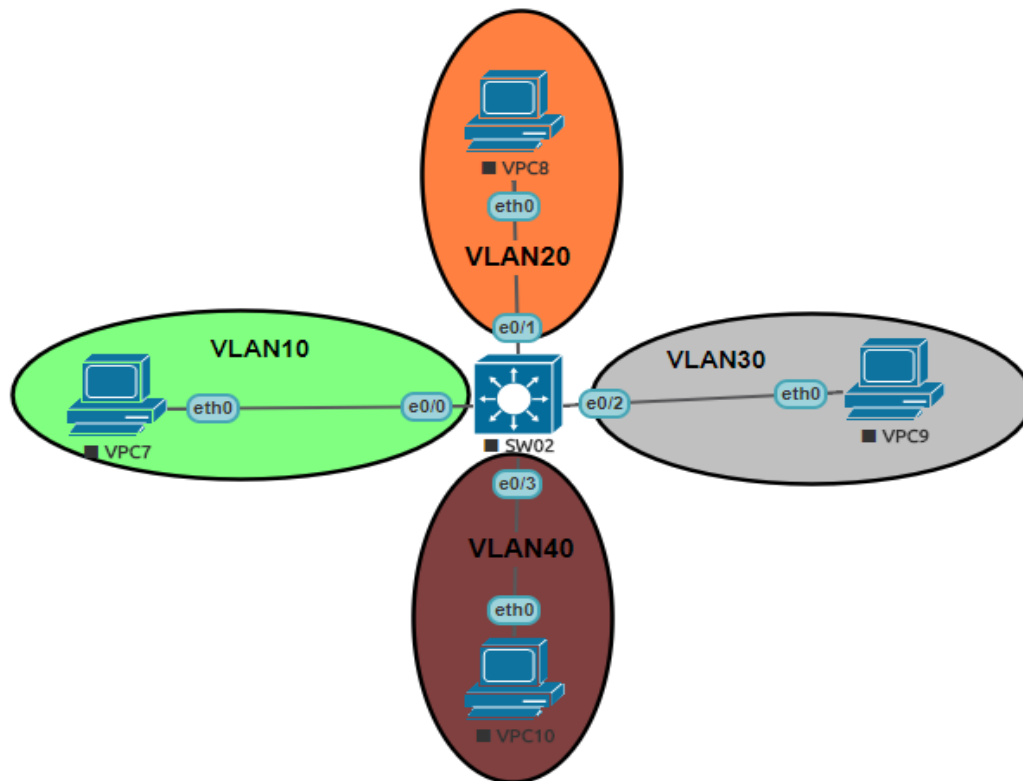
BY: ALA ALHABAHBA

2018

Index

Lab 1 : Configure VLANs	
Lab 2 : Configure Trunking 802.1Q	
Lab 3 : Configure QinQ	
Lab 4 : Configure STP & RSTP	
Lab5 : Configure MSTP	
Lab6 : Port fast	
Lab 7 : BPDU Filter	
Lab 8:BPDU Guard	
Lab 9 : configure REP	
Lab 10: Configure and Verify Inter-VLAN Routing (RouterOnStick)	
Lab 11: Configure and Verify Inter-VLAN Routing (SVI)	
Lab 12: Configure HSRP	
Lab 13: Configure VRRP	
Lab 14 : GLBP	
Lab 15: Configure OSPFv2	
Lab 16: OSPF Load Balance	
Lab 17 : ospfv3	
Lab 18 : OSPFV3 Authentication	
Lab 19: ospfv2 Authentication	
Lab 20: IS-IS configuration	
Lab 21: Configure IS-IS Load Balance	
Lab 22: Configure IS-IS Authentication	
Lab 23: route redistribution	
Lab 24 :Mpls Ldp	
Lab 25 : Configure BGP Process and BGP Peering Configure BGP to Advertise a Network Configure BGP Neighbor Authentication	
Lab 26: Configure IPv4 Filtering	
Lab 27: Configure IPv6 Filtering	
Lab 28: IPv6-in-IPv4 Configuration	
Lab 29 :IOS XR Software	

Lab 1 : Configure VLANs



Task :

- Create VLAN 10 For sales , VLAN 20 for market , VLAN 30 for eng , VLAN 40 for emp
- Verify all vlan created correctly with names
- Verify each vlan with id and name
- Check all switch port related to which vlan
- Assign switch port to particular vlan as diagram

E0/0 ---- vlan 10 ,E0/1 ----- vlan 20

E0/2 ----- vlan 30 ,E0/3 ----- vlan 40

- Verify switch ports if assigned correctly to their vlans
- Verify vlans membership
- Save your work

Solutions :

Create VLAN 10 For sales , VLAN 20 for market , VLAN 30 for eng , VLAN 40 for emp

```
Switch#configure terminal
```

```
Switch(config)#vlan 10
```

```
Switch(config-vlan)#name sales
```

```
Switch(config-vlan)#exit
```

```
Switch(config)#vlan 20
```

```
Switch(config-vlan)#name market
```

```
Switch(config-vlan)#exit
```

```
Switch(config)#vlan 30
```

```
Switch(config-vlan)#name eng
```

```
Switch(config-vlan)#exit
```

```
Switch(config)#vlan 40
```

```
Switch(config-vlan)#name emp
```

```
Switch(config-vlan)#exit
```

Verify all vlan created correctly with names

```
Switch#sh vlan or show vlan brief
```

```
VLAN Name          Status Ports
```

```

1 default          active Et0/0, Et0/1, Et0/2, Et0/3
10 sales           active
20 market         active
30 eng            active
40 emp            active

1002 fddi-default  act/unsup
1003 token-ring-default act/unsup
1004 fddinet-default act/unsup
1005 trnet-default act/unsup

```

Verify each vlan with id and name

Switch#sh vlan id 10

```

VLAN Name          Status Ports
-----
10 sales           active

```

Switch#sh vlan name sales

```

VLAN Name          Status Ports
-----
10 sales           active

```

For remaining vlans you can do same as vlan 10 sales

Check all switch port related to which vlan

Switch#show vlan brief

VLAN Name	Status	Ports

1 default	active	Et0/0, Et0/1, Et0/2, Et0/3
10 sales	active	
20 market	active	
30 eng	active	
40 emp	active	
1002 fddi-default	act/unsup	
1003 token-ring-default	act/unsup	
1004 fddinet-default	act/unsup	
1005 trnet-default	act/unsup	

Assign switch port to particular vlan as diagram

E0/0 ---- vlan 10 ,E0/1 ----- vlan 20

E0/2 ----- vlan 30 ,E0/3 ----- vlan 40

```
Switch(config)#interface ethernet 0/0
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
```

```
Switch(config)#interface ethernet 0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 20
```

```
Switch(config-if)#exit
```

```
Switch(config)#interface ethernet 0/2
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 30
```

```
Switch(config-if)#exit
```

```
Switch(config)#interface ethernet 0/3
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 40
```

```
Switch(config-if)#exit
```

Verify switch ports if assigned correctly to their vlans

```
Switch#sh vlan brief
```

VLAN Name	Status	Ports

1 default	active	
10 sales	active	Et0/0
20 market	active	Et0/1
30 eng	active	Et0/2
40 emp	active	Et0/3
1002 fddi-default	act/unsup	
1003 token-ring-default	act/unsup	
1004 fddinet-default	act/unsup	
1005 trnet-default	act/unsup	

Verify vlans membership

Switch#show interfaces ethernet 0/0 switchport

Name: Et0/0

Switchport: Enabled

Administrative Mode: static access

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: Off

Access Mode VLAN: 10 (sales)

Trunking Native Mode VLAN: 1 (default)

Administrative Native VLAN tagging: enabled

Switch#show interfaces ethernet 0/1 switchport

Name: Et0/1

Switchport: Enabled

Administrative Mode: static access

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: Off

Access Mode VLAN: 20 (market)

Trunking Native Mode VLAN: 1 (default)

Administrative Native VLAN tagging: enabled

Switch#show interfaces ethernet 0/2 switchport

Name: Et0/2

Switchport: Enabled

Administrative Mode: static access

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: Off

Access Mode VLAN: 30 (eng)

Trunking Native Mode VLAN: 1 (default)

Administrative Native VLAN tagging: enabled

Switch#show interfaces ethernet 0/3 switchport

Name: Et0/3

Switchport: Enabled

Administrative Mode: static access

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: Off

Access Mode VLAN: 40 (emp)

Trunking Native Mode VLAN: 1 (default)

Administrative Native VLAN tagging: enabled

Save your work

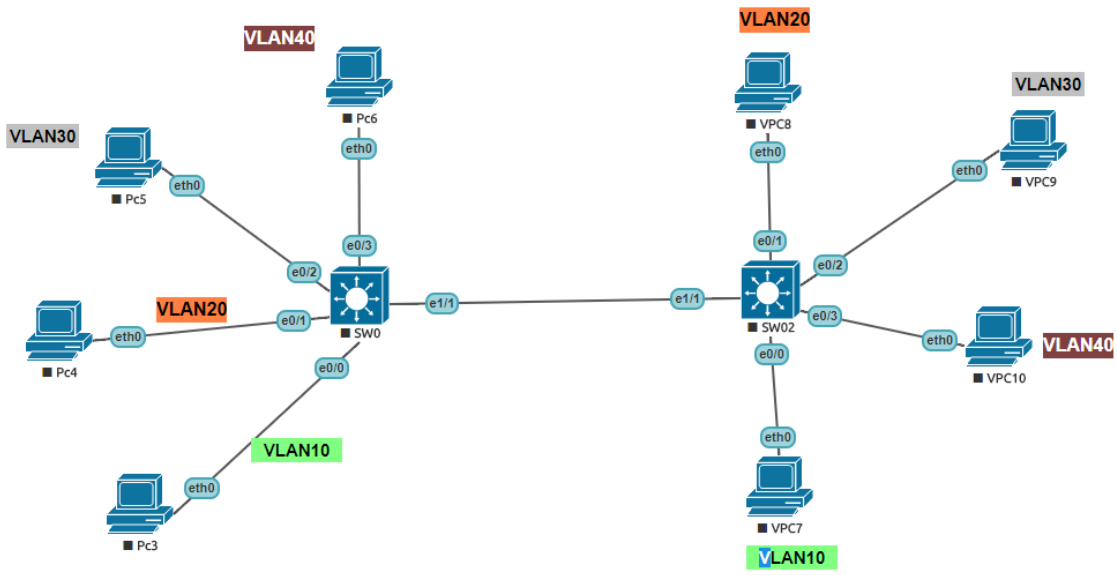
Switch#write memory

Building configuration...

Compressed configuration from 923 bytes to 595 bytes[OK]

End of lab 1

Lab 2 : Configure Trunking 802.1Q



Task:

Configure link between two switches as trunk

Verify trunk

Allow vlan 10, vlan20, vlan 30 to be exchange between switches through trunk

Verify allowed vlans on trunk

Check native vlan for trunk port

Create vlan 99 name mgmt. on both switches

Verify vlan 99 is created correctly

Change native vlan to vlan 99 on both switches

Check native vlan for trunk port on both switches

Save your work for both switches

Solution :

Configure link between two switches as trunk

On switch 0

```
Switch(config)#int e1/1
```

```
Switch(config-if)#switchport trunk encapsulation dot1q
```

```
Switch(config-if)#switchport mod trunk
```

Note : if you use this command at first : Switch(config-if)#switchport mod trunk

You will see error message :

Command rejected: An interface whose trunk encapsulation is "Auto" can not be configured to "trunk" mode.

Because of trunk encapsulation is auto , so we will make encapsulation dot1q

On switch 2 :

```
Switch(config)#interface ethernet 1/1
```

```
Switch(config-if)#switchport trunk encapsulation dot1q
```

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

Verify trunk

Switch#show interfaces ethernet 1/1 trunk

Port	Mode	Encapsulation	Status	Native vlan
Et1/1	on	802.1q	trunking	1

Port Vlans allowed on trunk

Et1/1 1-4094

Port Vlans allowed and active in management domain

Et1/1 1,10,20,30,40

Port Vlans in spanning tree forwarding state and not pruned

Et1/1 1,10,20,30,40

Switch#show interfaces ethernet 1/1 switchport

Name: Et1/1

Switchport: Enabled

Administrative Mode: trunk

Operational Mode: trunk

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q

Negotiation of Trunking: On

Access Mode VLAN: 1 (default)

Allow vlan 10, vlan20, vlan 30 to be exchange between switches through trunk

On both switches

```
Switch(config)#interface ethernet 1/1
```

```
Switch(config-if)#switchport trunk allowed vlan 10,20,30
```

Note : this command will allow these three vlans and ignore any allowed vlan before , in case if we want to add another vlan we must use command : switchport trunk allowed vlan add vlan-id or if we want to remove any vlan we use : switchport trunk allowed vlan remove vlan-id

Verify allowed vlans on trunk

```
Switch#sh int e1/1 trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Et1/1	on	802.1q	trunking	1

```
Port    Vlans allowed on trunk
```

```
Et1/1   10,20,30
```

```
Port    Vlans allowed and active in management domain
```

```
Et1/1   10,20,30
```

```
Port    Vlans in spanning tree forwarding state and not pruned
```

```
Et1/1   10,20,30
```

Or we can use this command to check all trunk port on switch

```
Switch#sh interfaces trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Et1/1	on	802.1q	trunking	1

Port Vlans allowed on trunk

Et1/1 10,20,30

Port Vlans allowed and active in management domain

Et1/1 10,20,30

Port Vlans in spanning tree forwarding state and not pruned

Et1/1 10,20,30

Check native vlan for trunk port

Switch#sh int e1/1 trunk

Port	Mode	Encapsulation	Status	Native vlan
Et1/1	on	802.1q	trunking	1

Create vlan 99 name mgmt. on both switches

On both switches

Switch(config)#vlan 99

Switch(config-vlan)#name mgmt

Switch(config-vlan)#exi

Verify vlan 99 is created correctly

On both switches

Switch#show vlan

VLAN Name	Status	Ports
1 default	active	Et1/0, Et1/2, Et1/3
10 sales	active	Et0/0
20 market	active	Et0/1
30 eng	active	Et0/2
40 emp	active	Et0/3
99 mgmt	active	
1002 fddi-default	act/unsup	
1003 trcrf-default	act/unsup	
1004 fddinet-default	act/unsup	
1005 trbrf-default	act/unsup	

Change native vlan to vlan 99 on both switches

On both switches

Switch(config)#interface ethernet 1/1

Switch(config-if)#switchport trunk native vlan 99

Note : if you configure onw switch with this command you will see error syslog message :

*Apr 12 14:35:28.510: %CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on Ethernet1/1 (1), with Switch Ethernet1/1 (99).

Because of native vlan mismatch between two switches

After you apply native vlan 99 on other switch you will not see this error syslog message and every thing will be ok

Check native vlan for trunk port on both switches

On switch 0

Switch#sh interfaces trunk

Port	Mode	Encapsulation	Status	Native vlan
Et1/1	on	802.1q	trunking	99

On switch 2

Switch#sh int trunk

Port	Mode	Encapsulation	Status	Native vlan
Et1/1	on	802.1q	trunking	99

So any traffic belong vlan 99 will be **untagged** through trunk port

Save your work for both switches

On both switches

Switch#wr

Building configuration...

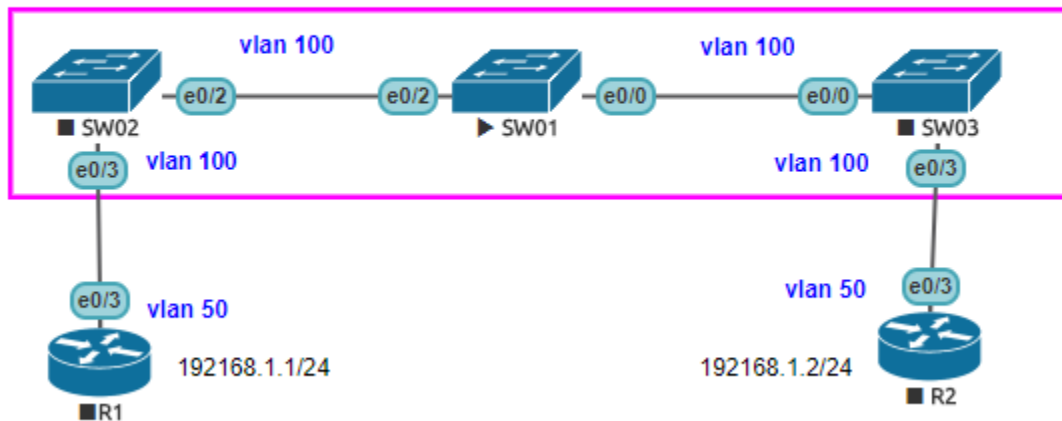
Compressed configuration from 1160 bytes to 703 bytes[OK]

End of lab 2

Lab 3 : Configure QinQ

SP

on



Tasks :

Create sub interface for vlan 50 on router 1 and router 2

Assign ip add for sub interface on router 1 and router 2 as diagram

Start up sub interface on router1 and router 2

Check interface status on both routers

Configure trunk port between switch 2, switch1 and switch 3 with interfaces connected together

Create vlan 100 on all switches

Verify vlan 100 is created on all switches

Configure QinQ in switch 2 on port face customer router R1 to tag customer vlan 50 into sp vlan 100

Verify QinQ is configured to correct port

Configure QinQ in switch 3 on port face customer router R2 to tag customer vlan 50 into sp vlan 100

Verify QinQ is configured to correct port

Verify allowed vlan on switch 2 and switch 3

Verify spanning tree in sw1 to be sure vlan 50 tagged into vlan 100

Force both switch 2 and switch 3 to tag native vlan

Verify native vlan tag on switch 2 and switch 3

Save your work

Solution :

Create sub interface for vlan 50 on router 1 and router 2

Assign ip add for sub interface on router 1 and router 2 as diagram

Start up sub interface on router1 and router 2

On R1 :

```
Router(config)#interface ethernet 0/3
```

```
Router(config-if)#no shutdown
```

```
Router(config-if)#interface ethernet 0/3.1
```

```
Router(config-subif)#encapsulation dot1Q 50
```

```
Router(config-subif)#ip address 192.168.1.1 255.255.255.0
```

```
Router(config-subif)#no shutdown
```

On R2 :

```
Router(config)#interface ethernet 0/3
```

```
Router(config-if)#no shutdown
```

```
Router(config-if)#interface ethernet 0/3.2
```

```
Router(config-subif)#encapsulation dot1Q 50
```

```
Router(config-subif)#ip address 192.168.1.2 255.255.255.0
```

```
Router(config-subif)#no shutdown
```

Check interface status on both routers

R1 :

```
Router#sh ip interface brief
```

```
Ethernet0/3.1      192.168.1.1  YES manual up      up
```

R2 :

```
Router#sh ip interface brief
```

```
Ethernet0/3.2      192.168.1.2  YES manual up      up
```

Configure trunk port between switch 2, switch1 and switch 3 with interfaces connected together



On switch2 :

```
Switch(config)#interface ethernet 0/2
Switch(config-if)#switchport trunk encapsulation dot1q
Switch(config-if)#switchport mode trunk
```

On switch 1 :

```
Switch(config)#interface ethernet 0/2
Switch(config-if)#sw tru encap dot
Switch(config-if)#sw mod tru
```

```
Switch(config)#int e0/0
Switch(config-if)#sw tru encap dot
Switch(config-if)#sw mod trunk
```

On switch 3 :

```
Switch(config)#int e0/0
Switch(config-if)#sw tru encap dot
```

```
Switch(config-if)#sw mod tru
```

Create vlan 100 on all switches

On all switches

```
Switch(config)#vlan 100
```

```
Switch(config-vlan)#exit
```

Verify vlan 100 is created on all switches

Issue this command on all switches you will see this output

```
Switch#sh vlan
```

VLAN Name	Status	Ports

1 default	active	Et0/0, Et0/1, Et0/3
100 VLAN0100	active	

Configure QinQ in switch 2 on port face customer router R1 to tag customer vlan 50 into sp vlan 100

```
Switch(config)#interface ethernet 0/3
```

```
Switch(config-if)#switchport access vlan 100
Switch(config-if)#switchport mode dot1q-tunnel
```

Verify QinQ is configured to correct port

```
Switch#sh dot1q-tunnel
```

```
dot1q-tunnel mode LAN Port(s)
```

```
-----
```

```
Et0/3
```

Configure QinQ in switch 3 on port face customer router R2 to tag customer vlan 50 into sp vlan 100

```
Switch(config)#int e0/3
Switch(config-if)#sw acc vlan 100
Switch(config-if)#sw mod dot1q-tunnel
```

Verify QinQ is configured to correct port

```
Switch#sh dot1q-tunnel
```

```
dot1q-tunnel mode LAN Port(s)
```

```
-----
```

```
Et0/3
```

Verify allowed vlan on switch 2 and switch 3

On switch 2 :

Switch#sh inter tru

Port	Mode	Encapsulation	Status	Native vlan
Et0/2	on	802.1q	trunking	1

Port Vlans allowed on trunk

Et0/2 1-4094

Port Vlans allowed and active in management domain

Et0/2 1,100

Port Vlans in spanning tree forwarding state and not pruned

Et0/2 1,100

On switch 3 :

Switch#sh int tru

Port	Mode	Encapsulation	Status	Native vlan
Et0/0	on	802.1q	trunking	1

Port Vlans allowed on trunk

Et0/0 1-4094

Port Vlans allowed and active in management domain

Et0/0 1,100

Port Vlans in spanning tree forwarding state and not pruned
Et0/0 1,100

Verify spanning tree in all switches to be sure vlan 50 tagged into vlan 100

On all switches

```
Switch#sh spanning-tree vlan 50
```

Spanning tree instance(s) for vlan 50 does not exist.

Force all switches to tag native vlan

On all switches

```
Switch(config)#vlan dot1q tag native
```

Verify native vlan tag on switch 2 and switch 3

```
Switch#sh int e0/3 switchport
```

Name: Et0/3

Switchport: Enabled

Administrative Mode: tunnel

Operational Mode: tunnel

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: Off

Access Mode VLAN: 100 (VLAN0100)

Trunking Native Mode VLAN: 1 (default)

Administrative Native VLAN tagging: enabled

Save your work

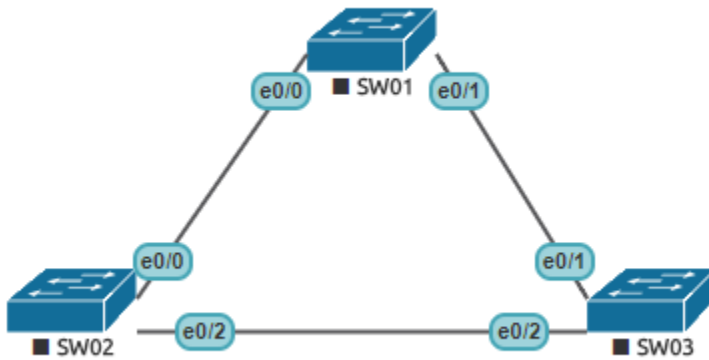
On all switches

Switch#wr

Building configuration...

End of lab 3

Lab 4 : Configure STP & RSTP



Tasks:

Verify spanning tree operation on Switches

Verify interface costs on Switch

Verify Spanning Tree Recalculation after switch1 interface shut down

Configure PVRST+.

Verify spanning-tree PVRST+

Solution :

Verify spanning tree operation on Switches

Verify interface costs on Switch

Switch 1

```
Switch#sh spanning-tree
```

```
VLAN0001
```

```
Spanning tree enabled protocol rstp
```

Root ID Priority 32769

Address aabb.cc00.0100

This bridge is the root

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address aabb.cc00.0100

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300 sec

Interface	Role	Sts	Cost	Prio.Nbr	Type
Et0/0	Desg	FWD	100	128.1	Shr
Et0/1	Desg	FWD	100	128.2	Shr
Et0/2	Desg	FWD	100	128.3	Shr
Et0/3	Desg	FWD	100	128.4	Shr

Switch 2:

Switch#show spanning-tree

VLAN0001

Spanning tree enabled protocol rstp

Root ID Priority 32769

Address aabb.cc00.0100

Cost 100

Port 1 (Ethernet0/0)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address aabb.cc00.0200

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300 sec

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Et0/0	Root	FWD	100	128.1	Shr
Et0/1	Desg	FWD	100	128.2	Shr
Et0/2	Desg	FWD	100	128.3	Shr
Et0/3	Desg	FWD	100	128.4	Shr

Switch 3:

Switch#show spanning-tree

VLAN0001

Spanning tree enabled protocol rstp

Root ID Priority 32769

Address aabb.cc00.0100

Cost 100

Port 2 (Ethernet0/1)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address aabb.cc00.0300

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 300 sec

Interface	Role	Sts	Cost	Prio.Nbr	Type
Et0/0	Desg	FWD	100	128.1	Shr
Et0/1	Root	FWD	100	128.2	Shr
Et0/2	Altn	BLK	100	128.3	Shr
Et0/3	Desg	FWD	100	128.4	Shr

Verify Spanning Tree Recalculation after switch1 interface shut down

Switch 1 Root Bridge

```
Switch(config)#int range e0/0-3
```

```
Switch(config-if-range)#sh
```

We will check stp recalculation after shut down interfaces on switch 1

On switch 2

```
Switch#show spanning-tree
```

```
VLAN0001
```

```
Spanning tree enabled protocol rstp
```

```
Root ID Priority 32769
```

Address aabb.cc00.0200

This bridge is the root

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address aabb.cc00.0200

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300 sec

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Et0/0	Desg	FWD	100	128.1		Shr
Et0/1	Desg	FWD	100	128.2		Shr
Et0/2	Desg	FWD	100	128.3		Shr
Et0/3	Desg	FWD	100	128.4		Sh

Switch 2 is root bridge now after recalculation

On switch 3

Switch#show spanning-tree

VLAN0001

Spanning tree enabled protocol rstp

Root ID Priority 32769

Address aabb.cc00.0200

Cost 100

Port 3 (Ethernet0/2)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address aabb.cc00.0300

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300 sec

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Et0/0	Desg	FWD	100	128.1		Shr
Et0/1	Desg	FWD	100	128.2		Shr
Et0/2	Root	FWD	100	128.3		Shr
Et0/3	Desg	FWD	100	128.4		Shr

Configure PVRST+.

On all switches

```
Switch(config)#spanning-tree mode rapid-pvst
```

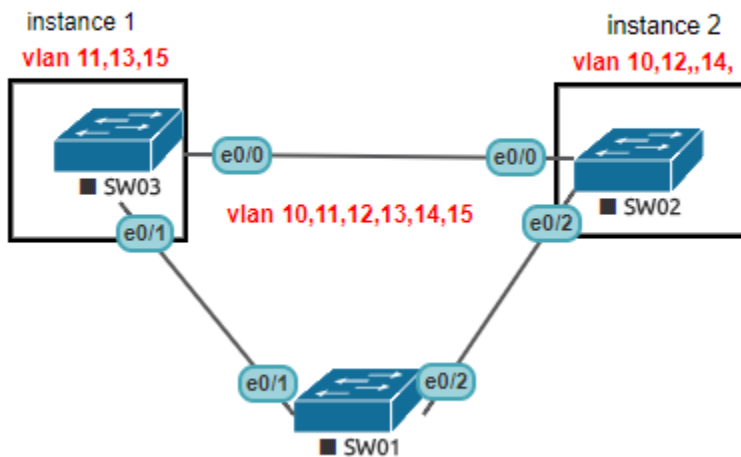
Verify spanning-tree PVRST+

```
Switch#sh spanning-tree
```

```
Spanning tree enabled protocol rstp
```

End of lab 4

Lab5 : Configure MSTP



TASKS:

Configure vlan 10,11,12,13,14,15, on switch 3 and switch 2

Configure switch 3 stp MSTP with :

Name : ccna revision :1

Configure two instance belong mstp ccna

Instance 1 for odd vlan

Instance 2 for even vlan

Configure mstp to be Root for instance 1

Configure switch 2 stp MSTP with :

Name : ccna revision :1

Configure two instance belong mstp ccna

Instance 1 for odd vlan

Instance 2 for even vlan

Configure mstp to be Root for instance 2

Verify mstp setup on switch 2 and switch 3

Save your work

Solution :

Configure vlan 10,11,12,13,14,15, on switch 3 and switch 2

On both switch

```
Switch(config)#vlan 10,11,12,13,14,15
```

```
Switch(config-vlan)#ex
```

```
Switch(config)#vlan 10,11,12,13,14,15
```

```
Switch(config-vlan)#ex
```

Configure switch 3 stp MSTP with :

Name : ccna revision :1

```
Switch(config)#spanning-tree mode mst
```

```
Switch(config)#spanning-tree mst configuration
```

```
Switch(config-mst)#name ccna
```

Switch(config-mst)#revision 1

Configure two instance belong mstp ccna

Instance 1 for odd vlan

Instance 2 for even vlan

Switch(config-mst)#instance 1 vlan 11,13,15

Switch(config-mst)#instance 2 vlan 10,12,14

Configure mstp to be Root for instance 1

Switch(config)#spanning-tree mst 1 root primary

Configure switch 2 stp MSTP with :

Name : ccna revision :1

Switch(config)#spanning-tree mode mst

Switch(config)#spanning-tree mst configuration

Switch(config-mst)#name ccna

Switch(config-mst)#revision 1

Configure two instance belong mstp ccna

Instance 1 for odd vlan

Instance 2 for even vlan

```
Switch(config-mst)#instance 1 vlan 11,13,15
```

```
Switch(config-mst)#instance 2 vlan 10,12,14
```

Configure mstp to be Root for instance 2

```
Switch(config)#spanning-tree mst 2 root primary
```

Verify mstp setup on switch 2 and switch 3

On switch 3 :

```
Switch#show spanning-tree mst configuration
```

```
Name [ccna]
```

```
Revision 1 Instances configured 3
```

```
Instance Vlans mapped
```

```
-----
```

```
0 1-9,16-4094
```

```
1 11,13,15
```

```
2 10,12,14
```

On switch 2 :

```
Switch#sh spanning-tree mst configuration
```

```
Name [ccna]
```

```
Revision 1 Instances configured 3
```

```
Instance Vlans mapped
```

```
-----
```

- 0 1-9,16-4094
- 1 11,13,15
- 2 10,12,14

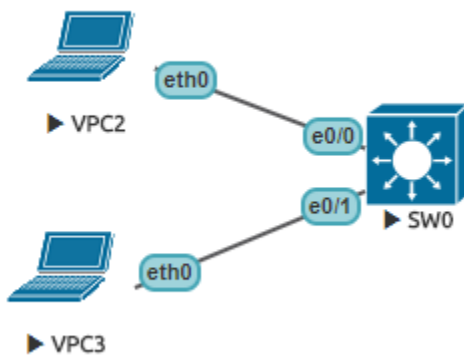
Save your work

On both switches

Switch#wr

End of lab 5

Lab6 : Port fast



Tasks:

Configure port fast on interface Ethernet 0/0

Configure port fast on all non-trunk interfaces

Solution

Configure port fast on interface Ethernet 0/0

```
Switch(config)#interface ethernet 0/0
```

```
Switch(config-if)#spanning-tree portfast
```

%Warning: portfast should only be enabled on ports connected to a single host. Connecting hubs, concentrators, switches, bridges, etc... to this interface when portfast is enabled, can cause temporary bridging loops.

Use with CAUTION

%Portfast has been configured on Ethernet0/0 but will only have effect when the interface is in a non-trunking mode.

Configure port fast on all non-trunk interfaces

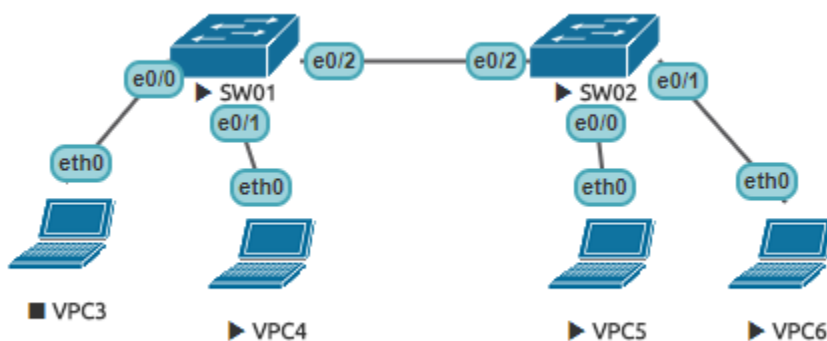
```
Switch(config)#spanning-tree portfast network default
```

Or

```
Switch(config)#spanning-tree portfast default
```

End of lab 6

Lab 7 : BPDU Filter



Tasks :

Configure bpdu filter on all end user interface on switch 1 and switch 2

Solution :

Configure bpdu filter on all end user interface on switch 1 and switch 2

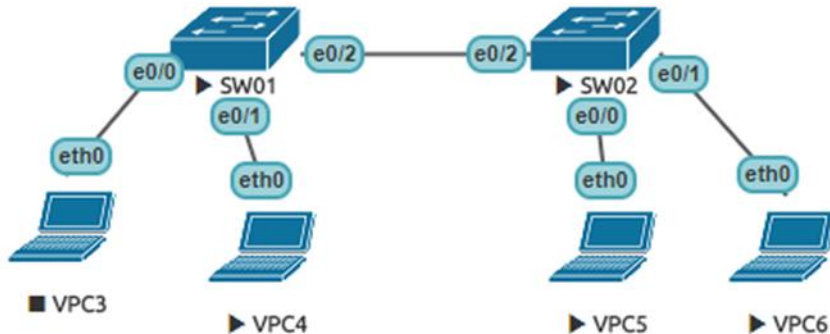
On both switches

```
Switch(config)#interface range ethernet 0/0-1
```

```
Switch(config-if-range)#spanning-tree bpdudfilter enable
```

End of lab 7

Lab 8:BPDU Guard



Tasks:

Configure bpduguard on interface Ethernet 0/0 on switch 1

Configure bpduguard on all interface

Verify bpduguard on interface ethernet 0/0

Solution

Configure bpduguard on interface Ethernet 0/0 on switch 1

```
Switch(config)#int e0/0
```

```
Switch(config-if)#spanning-tree bpduguard enable
```

Configure bpduguard on all interface

```
Switch#configure terminal
```

```
Switch(config)#spanning-tree portfast bpduguard default
```

Verify bpduguard on interface ethernet 0/0

```
Switch#sh run int e0/0
```

```
Building configuration...
```

```
Current configuration : 94 bytes
```

```
!  
interface Ethernet0/0  
  spanning-tree bpdufilter enable  
  spanning-tree bpduguard enable  
end
```

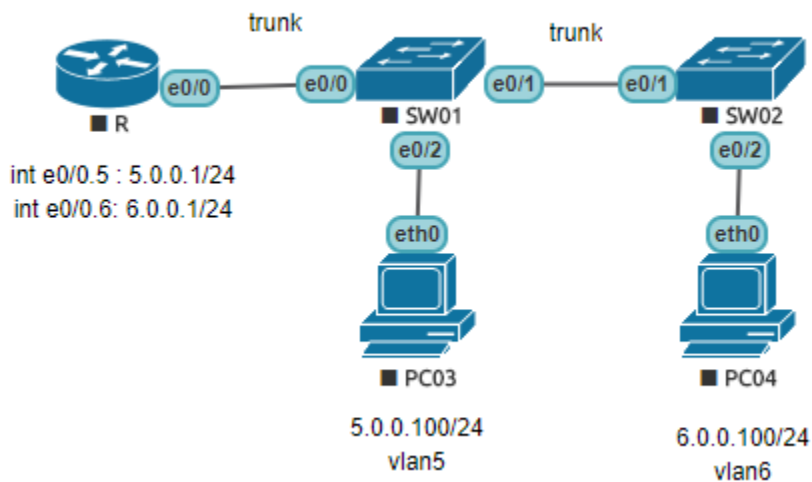
End of lab 8

Lab 9 : configure REP

```
Switch#conf t  
Switch(config)#interface fa0/1  
Switch(config-if)#port-type nni  
Switch(config-if)#switchport mode trunk  
Switch(config-if)#rep segment 1
```

End of lab 9

Lab 10: Configure and Verify Inter-VLAN Routing (RouterOnStick)



Tasks :

- Create sub interface for e0/0 on router as per diagram
- Configure pc03 ip add as per diagram
- Configure pc04 ip add as per diagram
- Create vlan5 , vlan 6 on both switches
- Configure interfaces between switches and switch to router as trunk port
- Assign port connected to pc03& pc04 to their vlan
- Ping from pc03 to pc04 ip address
- Save your work

Solution :

Create sub interface for e0/0 on router as per diagram

```
Router(config)#int e0/0
Router(config-if)#no sh
Router(config-if)#ex
```

```
Router(config)#int e0/0.5
Router(config-subif)#encapsulation dot1Q 5
Router(config-subif)#ip add 5.0.0.1 255.255.255.0
Router(config-subif)#ex
Router(config)#int e0/0.6
Router(config-subif)#enca
Router(config-subif)#encapsulation d
Router(config-subif)#encapsulation dot1Q 6
Router(config-subif)#ip add 6.0.0.1 255.255.255.0
Router(config-subif)#ex
```

Configure pc03 ip add as per diagram

```
VPCS> ip 5.0.0.100/24 5.0.0.1
Checking for duplicate address...
PC1 : 5.0.0.100 255.255.255.0 gateway 5.0.0.1
```

Configure pc04 ip add as per diagram

```
VPCS> ip 6.0.0.100/24 6.0.0.1
Checking for duplicate address...
PC1 : 6.0.0.100 255.255.255.0 gateway 6.0.0.1
```

Create vlan5 , vlan 6 on both switches

On both switches

```
Switch(config)#vlan 5,6
Switch(config-vlan)#ex
```

Configure interfaces between switches and switch to router as trunk port

On switch 1

```
Switch(config)#int ra e0/0-1
Switch(config-if-range)#sw tru encap dot
Switch(config-if-range)#sw mod tru
Switch(config-if-range)#ex
```

On switch 2

```
Switch(config)#int e0/1
Switch(config-if)#sw tru encap dot
Switch(config-if)#sw mod tr
Switch(config-if)#ex
```

Assign port connected to pc03& pc04 to their vlan

On switch 1

```
Switch(config)#int e0/2
Switch(config-if)#sw mod acc
Switch(config-if)#sw acc vlan 5
```

On switch 2

```
Switch(config)#int e0/2
Switch(config-if)#sw mod acc
Switch(config-if)#sw acc vlan 6
```

Ping from pc03 to pc04 ip address

```
VPCS> ping 6.0.0.100
```

```
84 bytes from 6.0.0.100 icmp_seq=1 ttl=63 time=5.678 ms
```

```
84 bytes from 6.0.0.100 icmp_seq=2 ttl=63 time=3.982 ms
```

```
84 bytes from 6.0.0.100 icmp_seq=3 ttl=63 time=2.708 ms
```

```
84 bytes from 6.0.0.100 icmp_seq=4 ttl=63 time=3.311 ms
```

```
84 bytes from 6.0.0.100 icmp_seq=5 ttl=63 time=3.582 ms
```

Save your work

On pc03&pc04

```
VPCS> save
```

```
Saving startup configuration to startup.vpc
```

```
. done
```

On sw1,sw2,router

```
Router#wr
```

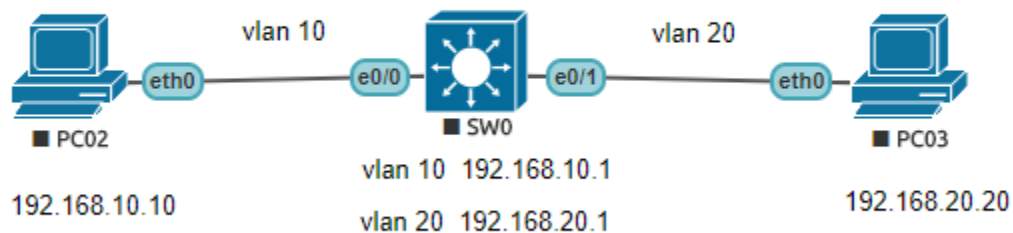
```
Building configuration...
```

```
[OK]
```

End of lab 10

Lab 11: Configure and Verify Inter-VLAN Routing

(SVI)



Tasks :

Configure ip address for pc02 and pc3 with /24 as diagram

Create vlan 10,vlan 20 on switch

Assign interface Ethernet 0/0 to vlan 10

Assign interface Ethernet 0/1 to vlan 20

Configure interface vlan 10, vlan 20 with ip as diagram on switch

Verify vlan interface status on switch

Solution :

Configure ip address for pc02 and pc03 with /24 as diagram

pc02

```
VPCS> ip 192.168.10.10/24 192.168.10.1
```

Checking for duplicate address...

```
PC1 : 192.168.10.10 255.255.255.0 gateway 192.168.10.1
```

pc03

```
VPCS> ip 192.168.20.20/24 192.168.20.1
```

Checking for duplicate address...

```
PC1 : 192.168.20.20 255.255.255.0 gateway 192.168.20.1
```

Create vlan 10,vlan 20 on switch

```
Switch(config)#vlan 10,20
```

```
Switch(config-vlan)#exit
```

Assign interface Ethernet 0/0 to vlan 10

```
Switch(config)#int e0/0
```

```
Switch(config-if)#sw mod acc
```

```
Switch(config-if)#sw acc vlan 10
```

```
Switch(config-if)#ex
```

Assign interface Ethernet 0/1 to vlan 20

```
Switch(config)#int e0/1
Switch(config-if)#sw mod acc
Switch(config-if)#sw acc vlan 20
Switch(config-if)#ex
```

Configure interface vlan 10, vlan 20 with ip as diagram on switch

```
Switch(config)#int vlan 10
Switch(config-if)#ip add 192.168.10.1 255.255.255.0
Switch(config)#int vlan 20
Switch(config-if)#ip add 192.168.20.1 255.255.255.0
```

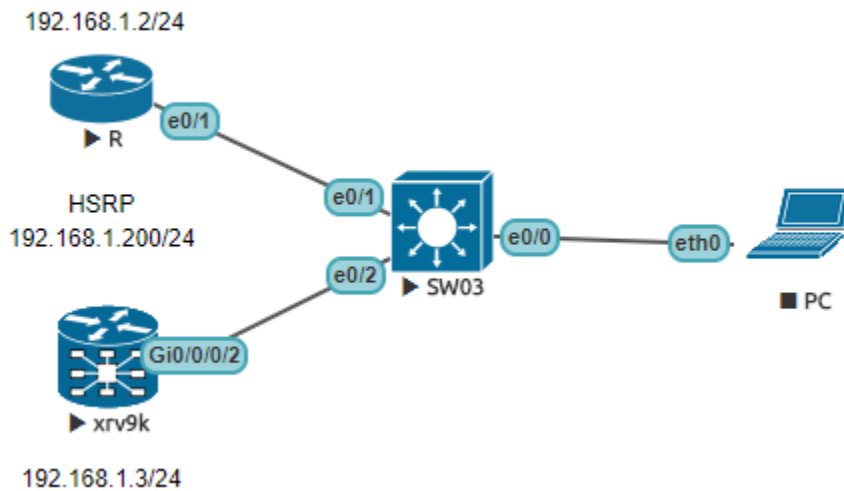
Verify vlan interface status on switch

```
Switch#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0/0	unassigned	YES	unset	up	up
Ethernet0/1	unassigned	YES	unset	up	up
Ethernet0/2	unassigned	YES	unset	up	up
Ethernet0/3	unassigned	YES	unset	up	up
Vlan10	192.168.10.1	YES	manual	up	up
Vlan20	192.168.20.1	YES	manual	up	up

End of lab 11

Lab 12: Configure HSRP



Tasks:

Configure ip address for router (ios) for interface e0/1

Configure ip address for router (xrv9k) for interface Gi0/0/0/2

Configure hsrp 1 on both routers

Ip address :192.168.1.200

Priority for ios router 105 , xrv9k router 95

Enable preempt on both router

Verify hsrp configuration on both routers

Configure load balance for two routers by configure hsrp2 on both

Ip address :192.168.1.222

Priority for ios router 95 , xrv9k router 105

Enable preempt on both router

Verify hsrp configuration on both routers

Solution :

Configure ip address for router (ios) for interface e0/1

```
Router(config)#int e0/1
Router(config-if)#ip add 192.168.1.2 255.255.255.0
Router(config-if)#no sh
Router(config-if)#ex
```

Configure ip address for router (xrv9k) for interface Gi0/0/0/2

```
RP/0/0/CPU0:ios(config)#int g0/0/0/2
RP/0/0/CPU0:ios(config-if)#ipv4 add 192.168.1.3/24
RP/0/0/CPU0:ios(config-if)#no sh
RP/0/0/CPU0:ios(config-if)#commit
```

Configure hsrp 1 on both routers

| Ip address :192.168.1.200 | Priority for ios router 105 , xrv9k router 95

Enable preempt on both router

On IOS Router:

```
Router(config)#int e0/1
Router(config-if)#standby 1 ip 192.168.1.200
*Apr 18 08:01:10.580: %HSRP-5-STATECHANGE: Ethernet0/1 Grp 1 state Standby -> Active
Router(config-if)#standby 1 priority 105
Router(config-if)#standby 1 preempt
```

On IOS XR :

```
RP/0/0/CPU0:ios(config)#router hsrp
RP/0/0/CPU0:ios(config-hsrp)#interface g0/0/0/2
RP/0/0/CPU0:ios(config-hsrp-if)#hsrp 1 ipv4 192.168.1.200
RP/0/0/CPU0:ios(config-hsrp-if)#hsrp 1 priority 95
RP/0/0/CPU0:ios(config-hsrp-if)#hsrp 1 preempt
RP/0/0/CPU0:ios(config-hsrp-if)#commit
```

Verify hsrp configuration on both routers

On ios router :

Router#sh standby br

P indicates configured to preempt.

|

Interface	Grp	Pri	P	State	Active	Standby	Virtual IP
Et0/1	1	105	P	Active	local	192.168.1.3	192.168.1.200

On ios XR router:

RP/0/0/CPU0:ios#sh hsrp

Wed Apr 18 08:13:06.483 UTC

IPv4 Groups:

P indicates configured to preempt.

|

Interface	Grp	Pri	P	State	Active addr	Standby addr	Group addr
Gi0/0/0/2	1	95	P	Standby	192.168.1.2	local	192.168.1.200

Configure load balance for two routers by configure hsrp2 on both

Ip address :192.168.1.222 | Priority for ios router 95 , xrv9k router 105

Enable preempt on both router

On ios XR router :

RP/0/0/CPU0:ios(config)#router hsrp

RP/0/0/CPU0:ios(config-hsrp)#interface gigabitEthernet 0/0/0/2

RP/0/0/CPU0:ios(config-hsrp-if)#hsrp 2 ipv4 192.168.1.222

RP/0/0/CPU0:ios(config-hsrp-if)#hsrp 2 prio 105

RP/0/0/CPU0:ios(config-hsrp-if)#hsrp 2 preem

```
RP/0/0/CPU0:ios(config-hsrp-if)#commit
```

On ios router:

```
Router(config)#int e0/1
Router(config-if)#standby 2 ip 192.168.1.222
Router(config-if)#standby 2 priority 95
Router(config-if)#standby 2 preempt
```

Verify hsrp configuration on both routers

On ios router

```
Router#sh stand br
          P indicates configured to preempt.
          |
Interface  Grp  Pri P State  Active      Standby      Virtual IP
Et0/1     1   105 P Active local      192.168.1.3  192.168.1.200
Et0/1     2   95  P Standby 192.168.1.3 local      192.168.1.222
```

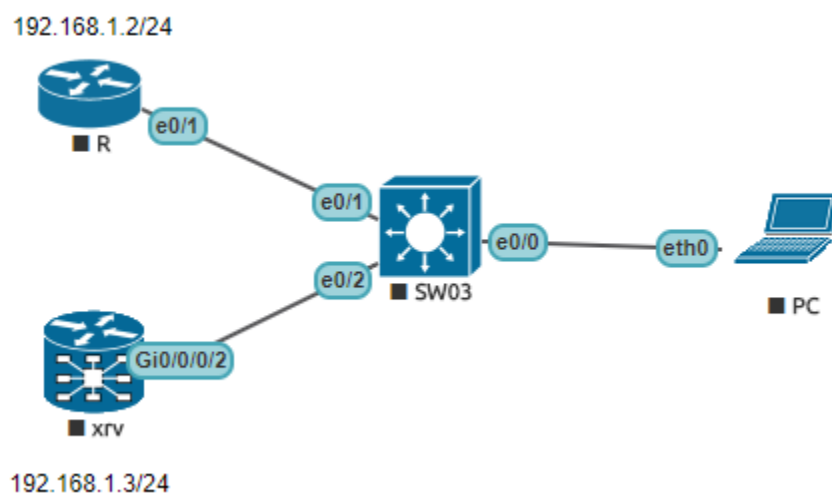
On ios XR router

```
RP/0/0/CPU0:ios#sh hsrp
Wed Apr 18 08:36:17.587 UTC
IPv4 Groups:
          P indicates configured to preempt.
          |
Interface  Grp  Pri P State  Active addr  Standby addr  Group addr
Gi0/0/0/2  1   95 P Standby 192.168.1.2 local      192.168.1.200
```

Gi0/0/0/2 2 105 P Active local 192.168.1.2 192.168.1.222

End lab 12

Lab 13: Configure VRRP



Tasks :

- Configure ip address 192.168.1.2/24 to ethernet0/1 on ios router
- Configure ip address 192.168.1.3/24 to Gi0/0/0/2 on ios XR router
- Configure VRRP 1 on ios router with :192.168.1.200 priority 105
- Configure VRRP 1 on iosXR router with :192.168.1.200 priority 95
- Verify VRRP configuration on both routers
- Configure load balance for two routers by configure VRRP2 on both
- IP address :192.168.1.220
- Priority on ios : 90, Priority on xr : 110

Verify VRRP configuration on both routers

Solution :

Configure ip address 192.168.1.2/24 to ethernet0/1 on ios router

```
Router(config)#int e0/1
Router(config-if)#ip add 192.168.1.2 255.255.255.0
Router(config-if)#no sh
Router(config-if)#ex
```

Configure ip address 192.168.1.3/24 to Gi0/0/0/2 on ios XR router

```
RP/0/0/CPU0:ios(config)#interface gigabitEthernet 0/0/0/2
RP/0/0/CPU0:ios(config-if)#ipv4 add 192.168.1.3/24
RP/0/0/CPU0:ios(config-if)#no sh
RP/0/0/CPU0:ios(config-if)# commit
```

Configure VRRP 1 on ios router with :192.168.1.200 priority 105

```
Router(config)#int e0/1
Router(config-if)#vrrp 1 ip 192.168.1.200
Router(config-if)#vrrp 1 priority 105
```

Configure VRRP 1 on iosXR router with :192.168.1.200 priority 95

```
RP/0/0/CPU0:ios(config)#router vrrp
RP/0/0/CPU0:ios(config-vrrp)#interface g0/0/0/2
RP/0/0/CPU0:ios(config-vrrp-if)#address-family ipv4
RP/0/0/CPU0:ios(config-vrrp-address-family)#vrrp 1
RP/0/0/CPU0:ios(config-vrrp-virtual-router)#address 192.168.1.200
RP/0/0/CPU0:ios(config-vrrp-virtual-router)#priority 95
RP/0/0/CPU0:ios(config-vrrp-virtual-router)#commit
```

Verify VRRP configuration on both routers

XR :

```
RP/0/0/CPU0:ios#sh vrrp
```

Sat Apr 21 09:25:32.460 UTC

IPv4 Virtual Routers:

A indicates IP address owner

| P indicates configured to preempt

| |

Interface	vrID	Prio	A	P	State	Master addr	VRouter addr
Gi0/0/0/2	1	95	P	Master	local	192.168.1.200	

IOS :

Router#SH VRRP BRlef

Interface	Grp	Pri	Time	Own	Pre	State	Master addr	Group addr
E0/1	1	105	3589	Y	Master	192.168.1.2	192.168.1.200	

Configure load balance for two routers by configure VRRP2 on both

IP address :192.168.1.220

Priority on ios : 90, Priority on xr : 110

ios :

Router(config)#INT E0/1

Router(config-if)#vrrp 2 ip 192.168.1.220

Router(config-if)#vrrp 2 priority 90

Xr :

RP/0/0/CPU0:ios(config)#router vrrp

RP/0/0/CPU0:ios(config-vrrp)#int g0/0/0/2

RP/0/0/CPU0:ios(config-vrrp-if)#add ipv4

RP/0/0/CPU0:ios(config-vrrp-address-family)#vrrp 2

RP/0/0/CPU0:ios(config-vrrp-virtual-router)#add 192.168.1.220

RP/0/0/CPU0:ios(config-vrrp-virtual-router)#priority 110

RP/0/0/CPU0:ios(config-vrrp-virtual-router)#commit

Verify VRRP configuration on both routers

ios :

```
Router#sh vrrp brief
```

Interface	Grp	Pri	Time	Own	Pre	State	Master addr	Group addr
Et0/1	1	105	3589	Y	Master		192.168.1.2	192.168.1.200
Et0/1	2	90	3648	Y	Master		192.168.1.2	192.168.1.220

Xr :

```
RP/0/0/CPU0:ios#sh vrrp
```

```
Sat Apr 21 09:31:42.845 UTC
```

```
IPv4 Virtual Routers:
```

```
    A indicates IP address owner
```

```
    | P indicates configured to preempt
```

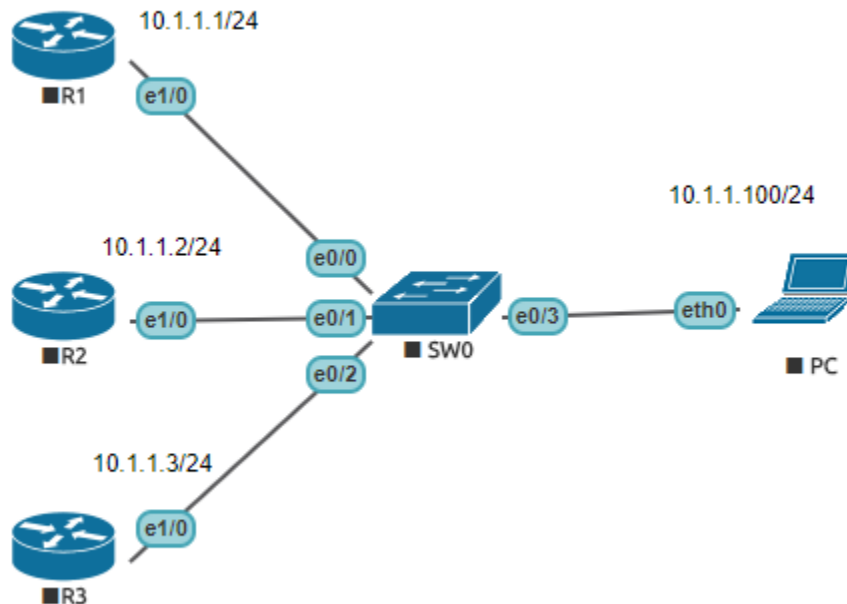
```
    | |
```

Interface	vrID	Prio	A	P	State	Master addr	VRouter addr
Gi0/0/0/2	1	95	P	Master	local	192.168.1.200	
Gi0/0/0/2	2	110	P	Master	local	192.168.1.220	

End lab 13

Lab 14 : GLBP

GLBP is not supported on the Cisco IOS XR routers



TASKS :

Configure ip address for both router R1/R2/R3

Configure GLBP on router R1/R2/R3

IP : 10.1.1.100 priority : R1 200 R2: 150 R3 : 50

Enable preempt

Verify GLBP on router R1

SOLUTION :

Configure ip address for both router R1/R2/R3

On router R1/R2/R3

```
Router(config)#int e1/0
```

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

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

Configure GLBP on both router R1/R2/R3

On router R1

```
Router(config)#int e1/0
```

```
Router(config-if)#glbp 1 ip 10.1.1.100
```

```
Router(config-if)#glbp 1 priority 200
```

```
*Apr 24 05:04:02.371: %GLBP-6-STATECHANGE: Ethernet1/0 Grp 1 state Speak -> Active
```

```
Router(config-if)#glbp 1 preempt
```

On router R2

```
Router(config)#int e1/0
```

```
Router(config-if)#glbp 1 ip 10.1.1.100
```

```
Router(config-if)#glbp 1 priority 100
```

```
*Apr 24 05:07:41.459: %GLBP-6-FWDSTATECHANGE: Ethernet1/0 Grp 1 Fwd 1 state Listen -> Active
```

```
Router(config-if)#glbp 1 preempt
```

On router R3

```
Router(config)#int e1/0
```

```
Router(config-if)#glbp 1 ip 10.1.1.100
```

```
Router(config-if)#glbp 1 priority 50
```

```
Router(config-if)#glbp 1 preempt
```

```
*Apr 24 05:09:51.415: %GLBP-6-FWDSTATECHANGE: Ethernet1/0 Grp 1 Fwd 1 state Listen -> Active
```

Verify GLBP ON router R1

Router#sh glbp br

Interface	Grp	Fwd	Pri	State	Address	Active router	Standby router
Et1/0	1	-	200	Active	10.1.1.100	local	unknown
Et1/0	1	1	-	Active	0007.b400.0101	local	-

Router#sh glbp

Ethernet1/0 - Group 1

State is Active

1 state change, last state change 00:07:38

Virtual IP address is 10.1.1.100

Hello time 3 sec, hold time 10 sec

Next hello sent in 2.144 secs

Redirect time 600 sec, forwarder time-out 14400 sec

Preemption enabled, min delay 0 sec

Active is local

Standby is unknown

Priority 200 (configured)

Weighting 100 (default 100), thresholds: lower 1, upper 100

Load balancing: round-robin

Group members:

ca01.29a8.001c (10.1.1.1) local

There is 1 forwarder (1 active)

Forwarder 1

State is Active

1 state change, last state change 00:07:27

MAC address is 0007.b400.0101 (default)

Owner ID is ca01.29a8.001c

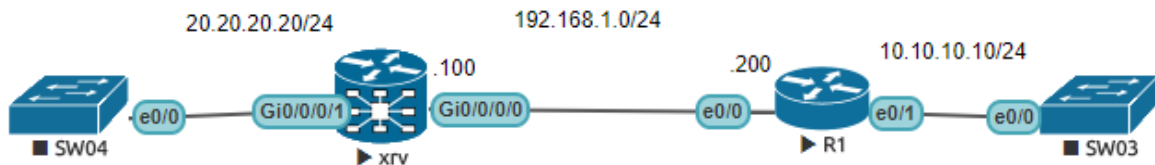
Redirection enabled

Preemption enabled, min delay 30 sec

Active is local, weighting 100

End lab 14

Lab 15: Configure OSPFv2



Tasks :

Configure ip address for interface shown

Enable OSPF process 1 on R1/xrv routers

Set the router id for R1 : 100.100.100.100 xrv : 200.200.200.200

Enable ospf on R1 interface e0/0,e0/1

Enable ospf on xrv router interface Gi0/0/0/0 , Gi0/0/0/1

Verify ospf if enabled on both routers

Verify ospf neighbor on both routers

Verify routing table on both routers

Solution :

Configure ip address for interface shown

R1:

```
Router(config)#int e0/0
Router(config-if)#ip add 192.168.1.200 255.255.255.0
Router(config-if)#no sh
Router(config-if)#ex
Router(config)#int e 0/1
Router(config-if)#ip add 10.10.10.10 255.255.255.0
Router(config-if)#no sh
Router(config-if)#ex
```

Xrv :

```
RP/0/0/CPU0:ios(config)#int g0/0/0/0
RP/0/0/CPU0:ios(config-if)#ipv4 add 192.168.1.100/24
RP/0/0/CPU0:ios(config-if)#no sh
RP/0/0/CPU0:ios(config-if)#ex
RP/0/0/CPU0:ios(config)#int g0/0/0/1
RP/0/0/CPU0:ios(config-if)#ipv4 add 20.20.20.20/24
RP/0/0/CPU0:ios(config-if)#no sh
RP/0/0/CPU0:ios(config-if)#commit
```

Enable OSPF process 1 on R1/xrv routers

Set the router id for R1 : 100.100.100.100 xrv : 200.200.200.200

XRv:

```
RP/0/0/CPU0:ios(config)#router ospf 1
RP/0/0/CPU0:ios(config-ospf)#router-id 200.200.200.200

R1 :
Router(config)#router ospf 1
Router(config-router)#router-id 100.100.100.100
```

Enable ospf on R1 interface e0/0,e0/1

```
Router(config)#router ospf 1
Router(config-router)#network 10.10.10.0 0.0.0.255 a 0
Router(config-router)#network 192.168.1.0 0.0.0.255 a 0
```

Enable ospf on xrv router interface Gi0/0/0/0 , Gi0/0/0/1

```
RP/0/0/CPU0:ios(config)#router ospf 1
RP/0/0/CPU0:ios(config-ospf)#area 0
RP/0/0/CPU0:ios(config-ospf-ar)#interface g0/0/0/0
RP/0/0/CPU0:ios(config-ospf-ar-if)#ex
RP/0/0/CPU0:ios(config-ospf-ar)#interface g0/0/0/1
RP/0/0/CPU0:ios(config-ospf-ar-if)#ex
RP/0/0/CPU0:ios(config-ospf-ar)#commit
```

Verify ospf if enabled on both routers

```
Xrv :
RP/0/0/CPU0:ios#sh protocols
Tue Apr 24 06:12:12.540 UTC
```

Routing Protocol OSPF 1

Router Id: 200.200.200.200

Distance: 110

Non-Stop Forwarding: Disabled

Redistribution:

None

Area 0

GigabitEthernet0/0/0/0

GigabitEthernet0/0/0/1

R1 :

Router#sh ip ospf

Routing Process "ospf 1" with ID 100.100.100.100

Verify ospf neighbor on both routers

R1 :

Router#sh ip ospf neighbor

Neighbor ID	Pri	State	Dead Time	Address	Interface
200.200.200.200	1	FULL/BDR	00:00:38	192.168.1.100	Ethernet0/0

XRV :

RP/0/0/CPU0:ios#sh ospf neighbor

Tue Apr 24 06:16:05.834 UTC

* Indicates MADJ interface

Indicates Neighbor awaiting BFD session up

Neighbors for OSPF 1

Neighbor ID	Pri	State	Dead Time	Address	Interface
100.100.100.100	1	FULL/DR	00:00:39	192.168.1.200	GigabitEthernet0/0/0/0

Neighbor is up for 00:04:55

Total neighbor count: 1

Verify routing table on both routers

R1:

Router#sh ip route

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks

C 10.10.10.0/24 is directly connected, Ethernet0/1

L 10.10.10.10/32 is directly connected, Ethernet0/1

20.0.0.0/24 is subnetted, 1 subnets

O 20.20.20.0 [110/11] via 192.168.1.100, 00:10:25, Ethernet0/0

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

C 192.168.1.0/24 is directly connected, Ethernet0/0

L 192.168.1.200/32 is directly connected, Ethernet0/0

XRV :

RP/0/0/CPU0:ios#sh route

Tue Apr 24 06:22:41.017 UTC

O 10.10.10.0/24 [110/11] via 192.168.1.200, 00:11:29, GigabitEthernet0/0/0/0

C 20.20.20.0/24 is directly connected, 00:17:06, GigabitEthernet0/0/0/1

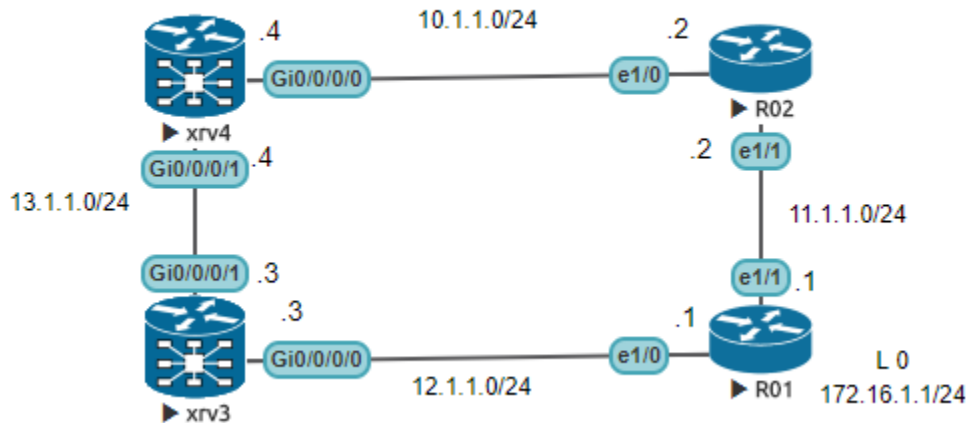
L 20.20.20.20/32 is directly connected, 00:17:06, GigabitEthernet0/0/0/1

C 192.168.1.0/24 is directly connected, 00:17:06, GigabitEthernet0/0/0/0

L 192.168.1.100/32 is directly connected, 00:17:06, GigabitEthernet0/0/0/0

End lab 15

Lab 16: OSPF Load Balance



Tasks :

Configure ip addressing for all routers as shown

Enable ospf 1 on all routers and advertise all interface link to other routers

Check ospf neighbor on all routers

Set ospf cost =10 for all routers

Set ospf maximum path for router xrv4 to 2

Verify routing table for xrv4

Solution :

Configure ip addressing for all routers as shown

R1	R2	Xrv3	Xrv4
R1(config)#int e1/0	R2(config)#int e1/0	RP/0/0/CPU0:ios(config)#int g0/0/0/0	RP/0/0/CPU0:ios(config)#int g0/0/0/0

R1(config-if)#ip add 12.1.1.1 255.255.255.0 R1(config-if)#no sh R1(config)#int e1/1 R1(config-if)#ip add 11.1.1.1 255.255.255.0 R1(config-if)#no sh R1(config)#int loo0 R1(config-if)#ip add 172.16.1.1 255.255.255.0	R2(config-if)#ip add 10.1.1.2 255.255.255.0 R2(config-if)#no sh R2(config-if)#ex R2(config)#int e1/1 R2(config-if)#ip add 11.1.1.2 255.255.255.0 R2(config-if)#no sh R2(config-if)#ex	RP/0/0/CPU0:ios(config-if)#ipv4 add 12.1.1.3/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/1 RP/0/0/CPU0:ios(config-if)#ipv4 add 13.1.1.3/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#commit	RP/0/0/CPU0:ios(config-if)#ipv4 add 10.1.1.4/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/1 RP/0/0/CPU0:ios(config-if)#ipv4 add 13.1.1.4/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#commit
---	---	--	--

Enable ospf 1 on all routers and advertise all interface link to other routers

R1(config)#router os 1 R1(config-router)#net 11.1.1.0 0.0.0.255 a 0 R1(config-router)#net 12.1.1.0 0.0.0.255 a 0	R2(config)#router os 1 R2(config-router)#net 10.1.1.0 0.0.0.255 a 0 R2(config-router)#net 11.1.1.0 0.0.0.255 a 0	RP/0/0/CPU0:ios(config)#router ospf 1 RP/0/0/CPU0:ios(config-ospf)#area 0 RP/0/0/CPU0:ios(config-ospf-ar)#inter g0/0/0/0 RP/0/0/CPU0:ios(config-ospf-ar-if)#ex RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/1	RP/0/0/CPU0:ios(config-ospf)#area 0 RP/0/0/CPU0:ios(config-ospf-ar)#inter g0/0/0/0 RP/0/0/CPU0:ios(config-ospf-ar-if)#ex RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/1 RP/0/0/CPU0:ios(config-ospf-ar-if)#ex
--	--	---	---

R1(config-router)#net 172.16.1.0 0.0.0.255 a 0		RP/0/0/CPU0:ios(config-ospf-ar-if)#ex RP/0/0/CPU0:ios(config-ospf-ar)#commit	RP/0/0/CPU0:ios(config-ospf-ar)#commit
--	--	---	--

Check ospf neighbor on all routers

R1(config-router)#do sh ip osp nei

Neighbor ID	Pri	State	Dead Time	Address	Interface
12.1.1.3	1	FULL/BDR	00:00:38	12.1.1.3	Ethernet1/0
11.1.1.2	1	FULL/BDR	00:00:35	11.1.1.2	Ethernet1/1

R2#sh ip ospf nei

Neighbor ID	Pri	State	Dead Time	Address	Interface
12.1.1.1	1	FULL/DR	00:00:35	11.1.1.1	Ethernet1/1
10.1.1.4	1	FULL/BDR	00:00:36	10.1.1.4	Ethernet1/0

Xrv3:

RP/0/0/CPU0:ios#sh ospf neighbor

Neighbor ID	Pri	State	Dead Time	Address	Interface
12.1.1.1	1	FULL/DR	00:00:31	12.1.1.1	GigabitEthernet0/0/0/0

Neighbor is up for 00:04:32

10.1.1.4	1	FULL/BDR	00:00:39	13.1.1.4	GigabitEthernet0/0/0/1
----------	---	----------	----------	----------	------------------------

Neighbor is up for 00:03:25

Total neighbor count: 2

Xrv4

RP/0/0/CPU0:ios#sh ospf neighbor

Neighbor ID	Pri	State	Dead Time	Address	Interface
-------------	-----	-------	-----------	---------	-----------

11.1.1.2 1 FULL/DR 00:00:39 10.1.1.2 GigabitEthernet0/0/0/0
Neighbor is up for 00:04:36

12.1.1.3 1 FULL/DR 00:00:31 13.1.1.3 GigabitEthernet0/0/0/1
Neighbor is up for 00:04:42

Total neighbor count: 2

Set ospf cost =10 for all routers

Set ospf maximum path for router xrv4 to 2

```
RP/0/0/CPU0:ios(config)#router ospf 1
RP/0/0/CPU0:ios(config-ospf)#area 0
RP/0/0/CPU0:ios(config-ospf-ar)#interface g0/0/0/0
RP/0/0/CPU0:ios(config-ospf-ar-if)#cost 10
RP/0/0/CPU0:ios(config-ospf-ar-if)#ex
RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/1
RP/0/0/CPU0:ios(config-ospf-ar-if)#cost 10
RP/0/0/CPU0:ios(config-ospf-ar-if)#ex
RP/0/0/CPU0:ios(config-ospf-ar)#ex
RP/0/0/CPU0:ios(config-ospf)#maximum paths 2
RP/0/0/CPU0:ios(config-ospf)#commit
XRV 3
RP/0/0/CPU0:ios(config)#router ospf 1
```

```
RP/0/0/CPU0:ios(config-ospf)#area 0
RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/0
RP/0/0/CPU0:ios(config-ospf-ar-if)#cost 10
RP/0/0/CPU0:ios(config-ospf-ar-if)#int g0/0/0/1
RP/0/0/CPU0:ios(config-ospf-ar-if)#cost 10
RP/0/0/CPU0:ios(config-ospf-ar-if)#ex
RP/0/0/CPU0:ios(config-ospf-ar)#commit
```

R1:

```
R1(config)#int e1/0
R1(config-if)#ip ospf cost 10
R1(config-if)#int e1/1
R1(config-if)#ip ospf cost 10
R1(config-if)#ex
```

R2:

```
int e1/0 , e1/1
ip ospf cost 10
```

Verify routing table for xrv4

RP/0/0/CPU0:ios#show route ospf

Tue Apr 24 09:21:47.257 UTC

O 11.1.1.0/24 [110/20] via 10.1.1.2, 00:14:04, GigabitEthernet0/0/0/0

- O 12.1.1.0/24 [110/20] via 13.1.1.3, 00:14:04, GigabitEthernet0/0/0/1
- O 172.16.1.1/32 [110/21] via 13.1.1.3, 00:14:04, GigabitEthernet0/0/0/1
- [110/21] via 10.1.1.2, 00:14:04, GigabitEthernet0/0/0/0

End lab 16

Lab 17 : ospfv3



TASKS :

Configure ipv6 address for all interfaces as shown in diagram
 Configure ospfv3 on both routers with process 1 , router id for

R: 2.2.2.2 xrv:1.1.1.1

Advertise both interfaces on both router into ospfv3

Verify ospfv3 neighbors

Verify routing table for both routers

Solution

R1:

Configure ipv6 address for all interfaces as shown in diagram

```
Router(config)#int e0/0
```

```
Router(config-if)#ipv6 add 2001:db8:0:1::2/64
```

```
Router(config-if)#no sh
Router(config)#int lo0
Router(config-if)#ipv6 add 2001:db9:0:1::2/64
Router(config-if)#no sh
```

Xrv :

```
RP/0/0/CPU0:ios(config)#int g0/0/0/0
RP/0/0/CPU0:ios(config-if)#ipv6 add 2001:db8:0:1::1/64
RP/0/0/CPU0:ios(config-if)#no sh
RP/0/0/CPU0:ios(config-if)#ex
RP/0/0/CPU0:ios(config)#int lo0
RP/0/0/CPU0:ios(config-if)#ipv6 add 2001:db7:0:1::1/64
RP/0/0/CPU0:ios(config-if)#no sh
RP/0/0/CPU0:ios(config-if)#ex
RP/0/0/CPU0:ios(config)#commit
```

Configure ospfv3 on both routers with process 1 , router id for

R: 2.2.2.2 xrv:1.1.1.1

R1:

```
Router(config)#ipv6 unicast-routing
Router(config)#router ospfv3 1
Router(config-router)#router-id 2.2.2.2
Router(config-router)#ex
Router(config)#int e0/0
Router(config-if)#ospfv3 1 ipv6 area 0
Router(config-if)#int lo0
Router(config-if)#ospfv3 1 ipv6 area 0
```

Xrv :

```
RP/0/0/CPU0:ios(config)#router ospfv3 1
RP/0/0/CPU0:ios(config-ospfv3)#router-id 1.1.1.1
RP/0/0/CPU0:ios(config-ospfv3)#address-family ipv6 unicast
RP/0/0/CPU0:ios(config-ospfv3)#area 0
RP/0/0/CPU0:ios(config-ospfv3-ar)#int lo0
RP/0/0/CPU0:ios(config-ospfv3-ar)#int g0/0/0/0
```

```
ORP/0/0/CPU0:ios(config)#commit
```

Verify ospfv3 neighbors

R:

```
Router#sh ipv6 ospf neighbor
```

```
OSPFv3 Router with ID (2.2.2.2) (Process ID 1)
```

Neighbor ID	Pri	State	Dead Time	Interface ID	Interface
1.1.1.1	1	FULL/BDR	00:00:31	4	Ethernet0/0

Xrv :

```
RP/0/0/CPU0:ios#sh ospfv3 neighbor
```

```
Thu Apr 26 02:34:33.340 UTC
```

```
# Indicates Neighbor awaiting BFD session up
```

```
Neighbors for OSPFv3 1
```

Neighbor ID	Pri	State	Dead Time	Interface ID	Interface
2.2.2.2	1	FULL/DR	00:00:33	3	GigabitEthernet0/0/0/0

```
Neighbor is up for 00:03:01
```

```
Total neighbor count: 1
```

Verify routing table for both routers

Xrv :

```
RP/0/0/CPU0:ios#show route ipv6 ospf
```

Thu Apr 26 02:36:59.900 UTC

```
O 2001:db9:0:1::2/128
```

```
[110/1] via fe80::a8bb:ccff:fe00:100, 00:05:26, GigabitEthernet0/0/0/0
```

R:

```
Router#sh ipv6 route ospf
```

```
O 2001:DB7:0:1::1/128 [110/10]
```

```
via FE80::5200:FF:FE02:1, Ethernet0/0
```

End lab 17

Lab 18 : OSPFV3 Authentication



Tasks :

Enable OSPFv3 SHA-1 authentication between both routers

Verify authentication

Solution :

Xrv :

```
RP/0/0/CPU0:ios(config)#router ospfv3 1
```

```
RP/0/0/CPU0:ios(config-ospfv3)#area 0
```

```
RP/0/0/CPU0:ios(config-ospfv3-ar)#inter g0/0/0/0
```

```
RP/0/0/CPU0:ios(config-ospfv3-ar-if)#authentication ipsec spi 256 sha1
1234567891011121314151617181920212223242
```

(40 hex digits)

R :

```
Router(config)#int e0/0
```

```
Router(config-if)#ipv6 ospf authentication ipsec spi 256 sha1
1234567891011121314151617181920212223242
```

```
*Apr 26 08:44:53.664: %CRYPTO-6-ISA_KMP_ON_OFF: ISAKMP is ON
```

Verify authentication

R1:

```
Router#sh ipv6 ospf inter e0/0
```

```
Ethernet0/0 is up, line protocol is up
```

```
Link Local Address FE80::A8BB:CCFF:FE00:100, Interface ID 3
```

```
Area 0, Process ID 1, Instance ID 0, Router ID 2.2.2.2
```

```
Network Type BROADCAST, Cost: 10
```

```
SHA-1 authentication SPI 256, secure socket UP (errors: 0)
```

Xrv :

```
RP/0/0/CPU0:ios#sh ospfv3 interface g0/0/0/0
```

```
Thu Apr 26 09:08:04.907 UTC
```

```
GigabitEthernet0/0/0/0 is up, line protocol is up, ipsec is up
```

```
Link Local address fe80::5200:ff:fe02:1, Interface ID 4
```

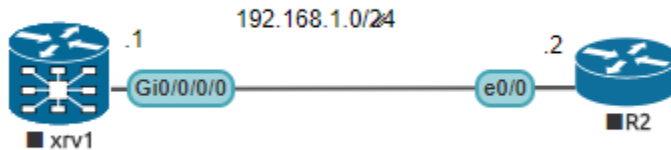
```
Area 0, Process ID 1, Instance ID 0, Router ID 1.1.1.1
```

```
Network Type BROADCAST, Cost: 1
```

```
AH Authentication SHA1, SPI 256
```

End lab 18

Lab 19: ospfv2 Authentication



Tasks:

Configure ip address for both routers as shown

Configure ospf process 10 between both routers and advertise internal link between routers within area 0

Configure MD5 authentication routers

Verify neighbor ship after authentication applied

solution:

Configure ip address for both routers as shown

R2

```
Router(config)#int e0/0
```

```
Router(config-if)#ip add 192.168.1.2 255.255.255.0
```

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

Xrv2 :

```
RP/0/0/CPU0:ios(config)#int g0/0/0/0
```

```
RP/0/0/CPU0:ios(config-if)#ipv4 add 192.168.1.1/24
```

```
RP/0/0/CPU0:ios(config-if)#no sh
```

```
RP/0/0/CPU0:ios(config-if)#commit
```

Configure ospf process 10 between both routers and advertise internal link between routers within area 0

R2:

```
Router(config)#router ospf 10
```

```
Router(config-router)#net 192.168.1.0 0.0.0.255 area 0
```

Xrv2:

```
RP/0/0/CPU0:ios(config)#router ospf 10
```

```
RP/0/0/CPU0:ios(config-ospf)#area 0
```

```
RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/0
```

Configure MD5 authentication routers

R2

```
Router(config)#int e0/0
```

```
Router(config-if)#ip ospf authentication message-digest
```

```
Router(config-if)#ip ospf message-digest-key 1 md5 spng2
```

Xrv2:

```
RP/0/0/CPU0:ios(config)#router ospf 10
```

```
RP/0/0/CPU0:ios(config-ospf)#area 0
```

```
RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/0
```

```
RP/0/0/CPU0:ios(config-ospf-ar-if)#authentication message-digest
```

```
RP/0/0/CPU0:ios(config-ospf-ar-if)#message-digest-key 1 md5 spng2
```

```
RP/0/0/CPU0:ios(config-ospf-ar-if)#commit
```

Verify neighbor ship after authentication applied

Xrv2:

```
RP/0/0/CPU0:ios#sh ospf neighbor
```

```
Thu Apr 26 11:16:14.326 UTC
```

* Indicates MADJ interface

Indicates Neighbor awaiting BFD session up

Neighbors for OSPF 10

Neighbor ID	Pri	State	Dead Time	Address	Interface
192.168.1.2	1	FULL/DR	00:00:38	192.168.1.2	GigabitEthernet0/0/0/0

Neighbor is up for 00:00:20

Total neighbor count: 1

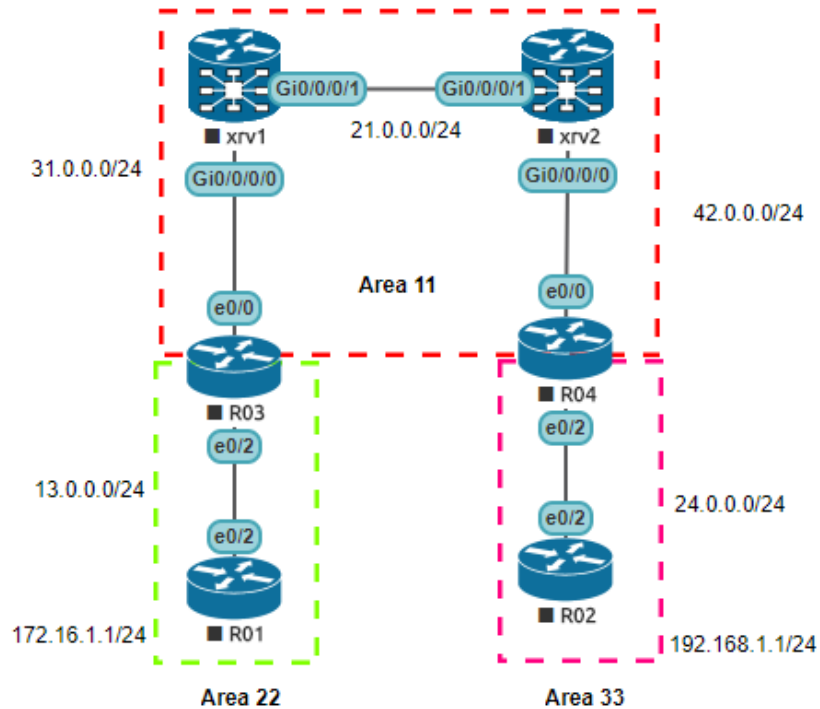
R2:

```
Router#sh ip osp nei
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
192.168.1.1	1	FULL/BDR	00:00:38	192.168.1.1	Ethernet0/0

End lab 19

Lab 20: IS-IS configuration



Tasks:

Set ip address for all router interface as shown, please note router no will use for end of interface ip

Configure is-is protocol in all routers use name SPNG2

Set is is level 1 for R1/R2

Set is is level1-2 for R3/R4 as per interface

Set is is level-2 for xrv1/xrv2

Add interfaces to is is for every router

Enable wide metric on R1/R2

Verify is is configuration on xrv1/R3

Verify is is neighbors

Verify route table for R2

Trace route from R1 to R2

Solution :

Set ip address for all router interface as shown, please note router no will use for end of interface ip

R1	R2	R3	R4	XRV1	XRV2
R1(config) #int e0/2 R1(config- if)#ip add 13.0.0.1 255.255.25 5.0 R1(config- if)#no sh	R2(config) #int e0/2 R2(config- if)#ip add 24.0.0.2 255.255.25 5.0 R2(config- if)#no sh	R3(config) #int e0/2 R3(config- if)#ip add 13.0.0.3 255.255.25 5.0 R3(config- if)#no sh R3(config) #int e0/0 R3(config- if)#ip add 31.0.0.3 255.255.25 5.0 R3(config- if)#no sh	R4(config) #int e0/2 R4(config- if)#ip add 24.0.0.4 255.255.25 5.0 R4(config- if)#no sh R4(config) #int e0/0 R4(config- if)#ip add 42.0.0.4 255.255.25 5.0 R4(config- if)#no sh	RP/0/0/CPU0:ios(conf ig)#int g0/0/0/0 RP/0/0/CPU0:ios(conf ig-if)#ipv4 add 42.0.0.2/24 RP/0/0/CPU0:ios(conf ig-if)#no sh RP/0/0/CPU0:ios(conf ig-if)#int g0/0/0/1 RP/0/0/CPU0:ios(conf ig-if)#ipv4 add 21.0.0.2/24 RP/0/0/CPU0:ios(conf ig-if)#no sh RP/0/0/CPU0:ios(conf ig-if)#commit	RP/0/0/CPU0:ios(conf ig)#int g0/0/0/0 RP/0/0/CPU0:ios(conf ig-if)#ipv4 add 31.0.0.1/24 RP/0/0/CPU0:ios(conf ig-if)#no sh RP/0/0/CPU0:ios(conf ig-if)#int g0/0/0/1 RP/0/0/CPU0:ios(conf ig-if)#ipv4 add 21.0.0.1/24 RP/0/0/CPU0:ios(conf ig-if)#no sh RP/0/0/CPU0:ios(conf ig-if)#commit

Configure is-is protocol in all routers use name SPNG2

Set is is level 1 for R1/R2

Set is is level1-2 for R3/R4 as per interface

Set is is level-2 for xrv1/xrv2

Add interfaces to is is for every router

R1	R1(config)#router isis R1(config-router)#net 49.0022.11111.11111.1111.00
----	---

	<pre> R1(config-router)#is-type level-1 R1(config)#int e0/2 R1(config-if)#ip router isis R1(config)#int lo0 R1(config-if)#ip add 172.16.1.1 255.255.255.0 R1(config-if)#ip router isis </pre>
R2	<pre> R2(config)#router isis R2(config-router)#net 49.0033.2222.2222.2222.00 R2(config-router)#is-type level-1 R2(config)#int e0/2 R21(config-if)#ip router isis R2(config)#int lo0 R2(config-if)#ip add 192.168.1.1 255.255.255.0 R2(config-if)#ip router isis </pre>
R3	<pre> R3(config)#router isis R3(config-router)#net 49.0022.3333.3333.3333.00 R3(config)#int e0/0 R3(config-if)#ip router isis R3(config-if)#isis circuit-type level-2-only R3(config-if)#int e0/2 R3(config-if)#ip router isis R3(config-if)#isis circuit-type level-1 </pre>
R4	<pre> R4(config)#router isis R4(config-router)#net 49.0033.4444.4444.4444.00 R4(config)#int e0/0 R4(config-if)#ip router isis R4(config-if)#isis circuit-type level-2-only R4(config-if)#ex R4(config)#int e0/2 R4(config-if)#ip router isis R4(config-if)#isis circuit-type level-1 </pre>
XRV1	<pre> RP/0/0/CPU0:ios(config)#router isis spng2 RP/0/0/CPU0:ios(config-isis)#net 49.0011.11111.11111.1111.00 RP/0/0/CPU0:ios(config-isis)#is-type level-2-only RP/0/0/CPU0:ios(config-isis)#interface g0/0/0/0 RP/0/0/CPU0:ios(config-isis-if)#add ipv4 unicast RP/0/0/CPU0:ios(config-isis-if-af)#ex RP/0/0/CPU0:ios(config-isis-if)#interface g0/0/0/1 RP/0/0/CPU0:ios(config-isis-if)#add ipv4 unicast </pre>
XRV2	<pre> RP/0/0/CPU0:ios(config)#router isis spng2 RP/0/0/CPU0:ios(config-isis)#net 49.0011.2222.2222.2222.00 RP/0/0/CPU0:ios(config-isis)#is-type level-2-only RP/0/0/CPU0:ios(config-isis)#int g0/0/0/0 RP/0/0/CPU0:ios(config-isis-if)#add ipv4 unicast RP/0/0/CPU0:ios(config-isis-if-af)#int g0/0/0/0 RP/0/0/CPU0:ios(config-isis-if)#add ipv4 unicast </pre>

Enable wide metric on R1/R2

On both routers :

```
R2(config)#router isis spng2
```

```
R2(config-router)#metric-style wide
```

Verify is is configuration on xrv1/R3

Xrv1

```
RP/0/0/CPU0:ios#sh run
```

```
router isis spng2
```

```
is-type level-2-only
```

```
net 49.0011.1111.1111.1111.00
```

```
interface GigabitEthernet0/0/0/0
```

```
address-family ipv4 unicast
```

```
!
```

```
!
```

```
interface GigabitEthernet0/0/0/1
```

```
address-family ipv4 unicast
```

R3 :

```
R3#sh run
```

```
router isis
```

```
net 49.0022.3333.3333.3333.00
```

Verify is is neighbors

R1	R1#SH ISIS NEIghbors System Id Type Interface IP Address State Holdtime Circuit Id R3 L1 Et0/2 13.0.0.3 UP 7 R3.02
R2	R2#SH ISIS NEIghbors System Id Type Interface IP Address State Holdtime Circuit Id R4 L1 Et0/2 24.0.0.4 UP 8 R4.02
R3	System Id Type Interface IP Address State Holdtime Circuit Id R1 L2 Et0/0 31.0.0.1 UP 27 R3.01 R1 L1 Et0/2 13.0.0.1 UP 25 R3.02
R4	R4#sh isis neighbors System Id Type Interface IP Address State Holdtime Circuit Id R2 L2 Et0/0 42.0.0.2 UP 27 R4.01 R2 L1 Et0/2 24.0.0.2 UP 23 R4.02
XRV1	RP/0/0/CPU0:ios#sh isis nei Fri Apr 27 15:13:58.171 UTC IS-IS spng2 neighbors: System Id Interface SNPA State Holdtime Type IETF-NSF 2222.2222.2222 Gi0/0/0/1 5000.0006.0002 Up 9 L2 Capable R3 Gi0/0/0/0 aabb.cc00.0300 Up 9 L2 Capable
XRV2	RP/0/0/CPU0:ios#sh isis nei Fri Apr 27 15:12:45.096 UTC IS-IS spng2 neighbors: System Id Interface SNPA State Holdtime Type IETF-NSF 4444.4444.4444 Gi0/0/0/0 aabb.cc00.0400 Up 24 L2 Capable 1111.1111.1111 Gi0/0/0/1 5000.0005.0002 Up 26 L2 Capable

Verify route table for R3

R3#sh ip route

```

13.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C   13.0.0.0/24 is directly connected, Ethernet0/2
L   13.0.0.3/32 is directly connected, Ethernet0/2
    21.0.0.0/24 is subnetted, 1 subnets
i L2 21.0.0.0 [115/20] via 31.0.0.1, 00:36:42, Ethernet0/0
    24.0.0.0/24 is subnetted, 1 subnets
i L2 24.0.0.0 [115/40] via 31.0.0.1, 00:00:18, Ethernet0/0
    31.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C   31.0.0.0/24 is directly connected, Ethernet0/0
L   31.0.0.3/32 is directly connected, Ethernet0/0
  
```

```
42.0.0.0/24 is subnetted, 1 subnets
i L2  42.0.0.0 [115/30] via 31.0.0.1, 00:00:18, Ethernet0/0
      172.16.0.0/24 is subnetted, 1 subnets
i L1  172.16.1.0 [115/20] via 13.0.0.1, 00:07:58, Ethernet0/2
i L2  192.168.1.0/24 [115/50] via 31.0.0.1, 00:00:18, Ethernet0/0
```

Trace route from R1 to R2

```
R1#trace 192.168.1.1
```

```
Type escape sequence to abort.
```

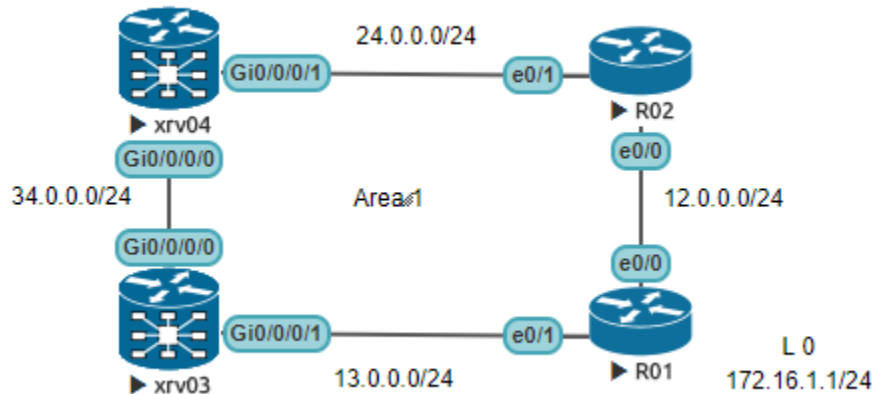
```
Tracing the route to 192.168.1.1
```

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

```
 1 13.0.0.3 1 msec 6 msec 7 msec
 2 31.0.0.1 8 msec 3 msec 2 msec
 3 21.0.0.2 15 msec 5 msec 4 msec
 4 42.0.0.4 5 msec 10 msec 8 msec
 5 24.0.0.2 10 msec 11 msec *
```

End lab 20

Lab 21: Configure IS-IS Load Balance



Tasks :

- Configure ip address for all interfaces as per diagram
- Enable isis on all routers use name :spng2 for xrv router
- Check routing table for xrv4

Solution :

Configure ip address for all interfaces as per diagram

R1	<pre>R1(config)#int lo0 R1(config-if)#ip add 172.16.1.1 255.255.255.0 R1(config-if)#no sh R1(config)#int e0/0 R1(config-if)#ip add 12.0.0.1 255.255.255.0 R1(config-if)#no sh R1(config)#int e0/1 R1(config-if)#ip add 13.0.0.1 255.255.255.0 R1(config-if)#no sh</pre>
R2	<pre>R2(config)#int e0/0</pre>

	<pre>R2(config-if)#ip add 12.0.0.2 255.255.255.0 R2(config-if)#no sh R2(config)#int e0/1 R2(config-if)#ip add 24.0.0.2 255.255.255.0 R2(config-if)#no sh</pre>
XRV3	<pre>RP/0/0/CPU0:XRV3(config)#int g0/0/0/1 RP/0/0/CPU0:XRV3(config-if)#ipv4 add 13.0.0.3 255.255.255.0 RP/0/0/CPU0:XRV3(config-if)#no sh RP/0/0/CPU0:XRV3(config-if)#ex RP/0/0/CPU0:XRV3(config)#int g0/0/0/0 RP/0/0/CPU0:XRV3(config-if)#ipv4 add 34.0.0.3/24 RP/0/0/CPU0:XRV3(config-if)#no sh RP/0/0/CPU0:XRV3(config-if)#commit</pre>
XRV4	<pre>RP/0/0/CPU0:XRV4(config)#int g0/0/0/0 RP/0/0/CPU0:XRV4(config-if)#ipv4 add 34.0.0.4/24 RP/0/0/CPU0:XRV4(config-if)#no sh RP/0/0/CPU0:XRV4(config-if)#ex RP/0/0/CPU0:XRV4(config)#int g0/0/0/1 RP/0/0/CPU0:XRV4(config-if)#ipv4 add 24.0.0.4/24 RP/0/0/CPU0:XRV4(config-if)#no sh RP/0/0/CPU0:XRV4(config-if)#commit</pre>

Enable isis on all routers use name :spng2 for xrv router

R1	<pre>R1(config)#router isis R1(config-router)#net 49.0001.1111.1111.1111.00 R1(config)#int lo0 R1(config-if)#ip router isis R1(config-if)#ex R1(config)#int ra e0/0-1 R1(config-if-range)#ip router isis R1(config-if-range)#ex</pre>
R2	<pre>R2(config)#router isis R2(config-router)#net 49.0001.2222.2222.2222.00 R2(config)#int ra e0/0-1 R2(config-if-range)#ip router isis R2(config-if-range)#ex</pre>
XRV3	<pre>RP/0/0/CPU0:XRV3(config)#router isis spng2 RP/0/0/CPU0:XRV3(config-isis)#net 49.0001.3333.3333.3333.00 RP/0/0/CPU0:XRV3(config-isis)#interface g0/0/0/0</pre>

	<pre>RP/0/0/CPU0:XR3(config-isis-if)#add ipv4 unicast RP/0/0/CPU0:XR3(config-isis-if-af)#ex RP/0/0/CPU0:XR3(config-isis-if)#ex RP/0/0/CPU0:XR3(config-isis)#interface g0/0/0/1 RP/0/0/CPU0:XR3(config-isis-if)#add ipv4 unicast RP/0/0/CPU0:XR3(config-isis-if-af)#ex RP/0/0/CPU0:XR3(config-isis-if)#ex RP/0/0/CPU0:XR3(config-isis)#commit</pre>
XR4	<pre>RP/0/0/CPU0:XR4(config)#router isis spng2 RP/0/0/CPU0:XR4(config-isis)#net 49.0001.4444.4444.00 RP/0/0/CPU0:XR4(config-isis)#interface g0/0/0/0 RP/0/0/CPU0:XR4(config-isis-if)#add ipv4 unicast RP/0/0/CPU0:XR4(config-isis-if-af)#ex RP/0/0/CPU0:XR4(config-isis-if)#ex RP/0/0/CPU0:XR4(config-isis)#interface g0/0/0/1 RP/0/0/CPU0:XR4(config-isis-if)#add ipv4 unicast RP/0/0/CPU0:XR4(config-isis-if-af)#ex RP/0/0/CPU0:XR4(config-isis-if)#ex RP/0/0/CPU0:XR4(config-isis)#commit</pre>

Check routing table for xrv4

RP/0/0/CPU0:XR4#sh route

```
i L1 12.0.0.0/24 [115/20] via 24.0.0.2, 00:00:47, GigabitEthernet0/0/0/1
i L1 13.0.0.0/24 [115/20] via 34.0.0.3, 00:02:05, GigabitEthernet0/0/0/0
C 24.0.0.0/24 is directly connected, 00:09:06, GigabitEthernet0/0/0/1
L 24.0.0.4/32 is directly connected, 00:09:06, GigabitEthernet0/0/0/1
C 34.0.0.0/24 is directly connected, 00:09:06, GigabitEthernet0/0/0/0
L 34.0.0.4/32 is directly connected, 00:09:06, GigabitEthernet0/0/0/0
i L1 172.16.1.0/24 [115/30] via 34.0.0.3, 00:00:47, GigabitEthernet0/0/0/0
[115/30] via 24.0.0.2, 00:00:47, GigabitEthernet0/0/0/1
```

If you couldn't see load balance you can do this command on xrv4 :

```
router isis spng2
```

```
address-family ipv4 unicast
```

```
maximum-paths 2
```

```
!
```

```
interface GigabitEthernet0/0/0/0
```

```
address-family ipv4 unicast
```

```
metric 100
```

```
!
```

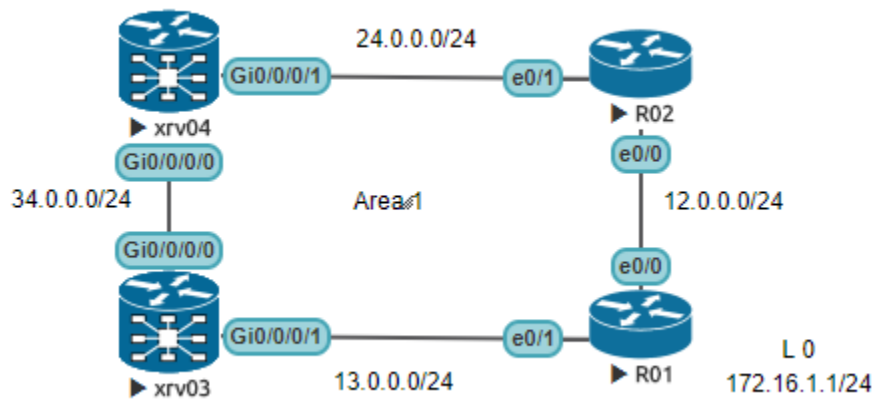
```
interface GigabitEthernet0/0/0/1
```

```
address-family ipv4 unicast
```

```
metric 100
```

End lab 21

Lab 22: Configure IS-IS Authentication



Tasks :

- Enable IS-IS MD5 hello packet authentication between xrv4 and R2
- Enable IS-IS MD5 LSP packet authentication on xrv4 and R2

Solution :

XR4 :

```
RP/0/0/CPU0:XR4(config)#router isis spng2
```

```
RP/0/0/CPU0:XR4(config-isis)#lsp-password hmac-md5 cisco123
```

```
RP/0/0/CPU0:XR4(config-isis)#int g0/0/0/1
RP/0/0/CPU0:XR4(config-isis-if)#hello-password hmac-md5 cisco123
RP/0/0/CPU0:XR4(config-isis-if)#commit

R2:

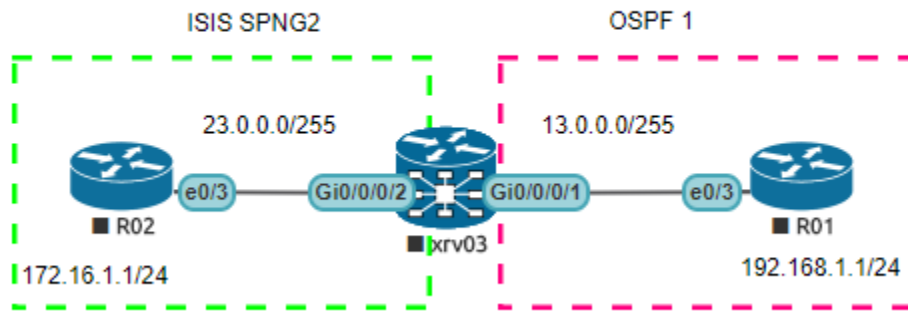
R2(config)#key chain ala
R2(config-keychain)#key 1
R2(config-keychain-key)#key
R2(config-keychain-key)#key-string cisco123
R2(config-keychain-key)#ex

R2(config)#int e0/1
R2(config-if)#isis authentication mod md5
R2(config-if)#isis authentication key-chain ala
R2(config-if)#ex

R2(config)#router isis
R2(config-router)#authentication mod md5
R2(config-router)#authentication key-chain ala
```

End lab 22

Lab 23: route redistribution



Tasks:

Configure ip address for all router as per diagram

Configure ospf 1 on R1 and advertise Lo – E0/3 interface

Configure ospf 1 on xrv3 and advertise G0/0/0/1

Configure isis area 0 on R2 and enable isis on E0/3

Configure isis spng2 on xrv3 and enable G0/0/0/2

Redistribute ospf rout into isis (R2 will learn network 192.168.1.1)

Redistribute isis route into ospf (R1 will learn network 172.16.1.1)

Solution

Configure ip address for all router as per diagram

R1	<pre>R1(config)#int lo0 R1(config-if)#ip add 192.168.1.1 255.255.255.0 R1(config-if)#no sh R1(config)#int e0/3 R1(config-if)#ip add 13.0.0.1 255.255.255.0 R1(config-if)#no sh</pre>
R2	<pre>R2(config)#int lo0 R2(config-if)#ip add 172.16.1.1 255.255.255.0 R2(config-if)#no sh R2(config)#int e0/3 R2(config-if)#ip add 23.0.0.2 255.255.255.0 R2(config-if)#no sh R2(config-if)#ex</pre>

Xrv3	<pre> RP/0/0/CPU0:ios(config)#int g0/0/0/1 RP/0/0/CPU0:ios(config-if)#ipv4 add 13.0.0.3/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/2 RP/0/0/CPU0:ios(config-if)#ipv4 add 23.0.0.3/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#commit </pre>
------	--

Configure ospf 1 on R1 and advertise Lo – E0/3 interface

```

R1(config)#router os 1
R1(config-router)#net 192.168.1.0 0.0.0.255 a 0
R1(config-router)#net 13.0.0.0 0.0.0.255 a 0

```

Configure ospf 1 on xrv3 and advertise G0/0/0/1

```

RP/0/0/CPU0:ios(config)#router ospf 1
RP/0/0/CPU0:ios(config-ospf)#area 0
RP/0/0/CPU0:ios(config-ospf-ar)#interface g0/0/0/1
RP/0/0/CPU0:ios(config-ospf-ar-if)#commit

```

Configure isis area 0 on R2 and enable isis on E0/3

```

R2(config)#router isis
R2(config-router)#net 49.0000.2222.2222.2222.00
R2(config-router)#ex
R2(config)#int lo0
R2(config-if)#ip router isis
R2(config-if)#ex
R2(config)#int e0/3
R2(config-if)#ip router isis

```

R2(config-if)#ex

Configure isis spng2 on xrv3 and enable G0/0/0/2

```
RP/0/0/CPU0:ios(config)#router isis spng2
RP/0/0/CPU0:ios(config-isis)#net 49.0000.3333.3333.00
RP/0/0/CPU0:ios(config-isis)#int g0/0/0/2
RP/0/0/CPU0:ios(config-isis-if)#address-family ipv4 unicast
RP/0/0/CPU0:ios(config-isis-if-af)#ex
RP/0/0/CPU0:ios(config-isis-if)#commit
```

Redistribute ospf rout into isis (R2 will learn network 192.168.1.1)

```
RP/0/0/CPU0:ios(config)#router isis spng2
RP/0/0/CPU0:ios(config-isis)#address-family ipv4 unicast
RP/0/0/CPU0:ios(config-isis-af)#redistribute ospf 1 metric ?
<0-16777215> Default metric: <0-63> for narrow, <0-16777215> for wide
RP/0/0/CPU0:ios(config-isis-af)#redistribute ospf 1 metric 20
RP/0/0/CPU0:ios(config-isis-af)#commit
```

R2#sh ip route

```
i L2 192.168.1.1 [115/30] via 23.0.0.3, 00:00:49, Ethernet0/3
```

Redistribute isis route into ospf (R1 will learn network 172.16.1.1)

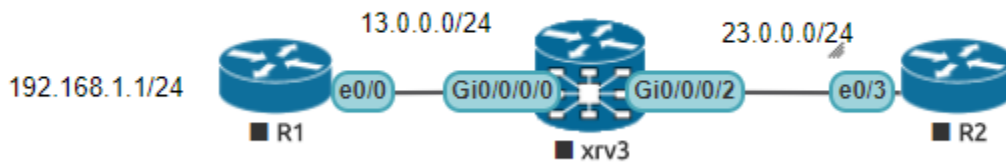
```
RP/0/0/CPU0:ios(config)#router ospf 1
RP/0/0/CPU0:ios(config-ospf)#redistribute isis spng2 metric 30
RP/0/0/CPU0:ios(config-ospf)#commit
```

R1#sh ip route

```
O E2 172.16.1.0 [110/30] via 13.0.0.3, 00:00:19, Ethernet0/3
```

End lab 23

Lab 24 :Mpls Ldp



TASKS :

Configure ip address for all routers as per diagram

Configure to all routers loopback 0 with x.x.x.x/24 x= router number

Configure ospf on all routers including loopback interface

Check loopback interface learned and reachable via ospf route

Enable mpls ldp on all routers interfaces

Verify LDP neighbors on R2

Displays content of the LIB table on xrv3

Displays content of the LFIB table on xrv3

Displays content of the FIB(CEF) on xrv3

Solution

Configure ip address for all routers as per diagram

Configure to all routers loopback 0 with x.x.x.x/24 x= router number

R1	<pre>R1(config)#int loo 0 R1(config-if)#ip add 1.1.1.1 255.255.255.0 R1(config)#int lo 1 R1(config-if)#ip add 192.168.1.1 255.255.255.0 R1(config-if)#ex R1(config)#int e0/0</pre>
----	--

	R1(config-if)#ip add 13.0.0.1 255.255.255.0 R1(config-if)#no sh
R2	R2(config)#int e0/3 R2(config-if)#ip add 23.0.0.2 255.255.255.0 R2(config-if)#no sh R2(config-if)#ex R2(config)#int lo0 R2(config-if)#ip add 2.2.2.2 255.255.255.0
XRV3	RP/0/0/CPU0:ios(config)#int g0/0/0/0 RP/0/0/CPU0:ios(config-if)#ipv4 add 13.0.0.3/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/2 RP/0/0/CPU0:ios(config-if)#ipv4 add 23.0.0.3/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#commit

Configure ospf on all routers including loopback interface

R1	R1(config)#router ospf 1 R1(config-router)#net 192.168.1.0 0.0.0.255 a 0 R1(config-router)#net 1.1.1.0 0.0.0.255 a 0 R1(config-router)#net 13.0.0.0 0.0.0.255 a 0
R2	R2(config)#router os 1 R2(config-router)#net 2.2.2.0 0.0.0.255 a 0 R2(config-router)#net 23.0.0.0 0.0.0.255 a 0
XRV3	RP/0/0/CPU0:ios(config)#router ospf 1 RP/0/0/CPU0:ios(config-ospf)#router-id 3.3.3.3 RP/0/0/CPU0:ios(config-ospf)#area 0 RP/0/0/CPU0:ios(config-ospf-ar)#int lo0 RP/0/0/CPU0:ios(config-ospf-ar-if)#ex RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/0 RP/0/0/CPU0:ios(config-ospf-ar-if)#ex RP/0/0/CPU0:ios(config-ospf-ar)#int g0/0/0/2 RP/0/0/CPU0:ios(config-ospf-ar-if)#commit

Check loopback interface learned and reachable via ospf route

R1	R1#sh ip route osp 2.0.0.0/32 is subnetted, 1 subnets O 2.2.2.2 [110/12] via 13.0.0.3, 00:04:11, Ethernet0/0 3.0.0.0/32 is subnetted, 1 subnets
----	---

	<pre>O 3.3.3.3 [110/11] via 13.0.0.3, 00:04:21, Ethernet0/0</pre>
R2	<pre>R2#sh ip route osp 1.0.0.0/32 is subnetted, 1 subnets O 1.1.1.1 [110/12] via 23.0.0.3, 00:00:48, Ethernet0/3 3.0.0.0/32 is subnetted, 1 subnets O 3.3.3.3 [110/11] via 23.0.0.3, 00:00:48, Ethernet0/3</pre>
XRV3	<pre>RP/0/0/CPU0:ios#sh route ospf O 1.1.1.1/32 [110/2] via 13.0.0.1, 00:00:38, GigabitEthernet0/0/0/0 O 2.2.2.2/32 [110/2] via 23.0.0.2, 00:00:29, GigabitEthernet0/0/0/2</pre>

Enable mpls ldp on all routers interfaces

R1	<pre>R1(config)#int e0/0 R1(config-if)#mpls ip R1(config-if)#ex *May 1 07:42:23.046: %LDP-5-NBRCHG: LDP Neighbor 3.3.3.3:0 (1) is UP</pre>
R2	<pre>R2(config)#mpls ip R2(config)#ip cef R2(config)#int e0/3 R2(config-if)#mpls ip R2(config-if)#ex R2(config)# *May 1 07:42:50.202: %LDP-5-NBRCHG: LDP Neighbor 3.3.3.3:0 (1) is UP</pre>
Xrv3	<pre>RP/0/0/CPU0:ios(config)#mpls ldp RP/0/0/CPU0:ios(config-ldp)#int g0/0/0/0 RP/0/0/CPU0:ios(config-ldp-if)#ex RP/0/0/CPU0:ios(config-ldp)#int g0/0/0/2 RP/0/0/CPU0:ios(config-ldp-if)#commit</pre>

Verify LDP neighbors on R2

R2#sh mpls ldp neighbor

Peer LDP Ident: 3.3.3.3:0; Local LDP Ident 2.2.2.2:0

TCP connection: 3.3.3.3.26088 - 2.2.2.2.646

State: Oper; Msgs sent/rcvd: 10/11; Downstream

Up time: 00:01:03

LDP discovery sources:

Ethernet0/3, Src IP addr: 23.0.0.3

Addresses bound to peer LDP Ident:

13.0.0.3 23.0.0.3 3.3.3.3

Displays content of the LIB table on XRv3

RP/0/0/CPU0:ios#sh mpls ldp bindings

Tue May 1 07:45:26.646 UTC

1.1.1.0/24, rev 0

No local binding

Remote bindings: (1 peers)

Peer	Label
192.168.1.1:0	ImpNull

1.1.1.1/32, rev 12

Local binding: label: 24000

Remote bindings: (1 peers)

Peer	Label
2.2.2.2:0	16

2.2.2.0/24, rev 0

No local binding

Remote bindings: (1 peers)

Peer	Label
-----	-----
2.2.2.2:0	ImpNull

2.2.2.2/32, rev 14

Local binding: label: 24002

Remote bindings: (1 peers)

Peer	Label
-----	-----
192.168.1.1:0	16

3.3.3.0/24, rev 4

Local binding: label: ImpNull

No remote bindings

3.3.3.3/32, rev 2

Local binding: label: ImpNull

Remote bindings: (2 peers)

Peer	Label
-----	-----
192.168.1.1:0	17
2.2.2.2:0	17

13.0.0.0/24, rev 6

Local binding: label: ImpNull

Remote bindings: (2 peers)

Peer	Label
-----	-----
192.168.1.1:0	ImpNull
2.2.2.2:0	18

23.0.0.0/24, rev 8

Local binding: label: ImpNull

Remote bindings: (2 peers)

Peer	Label
-----	-----
192.168.1.1:0	18
2.2.2.2:0	ImpNull

192.168.1.0/24, rev 0

No local binding

Remote bindings: (1 peers)

Peer	Label
-----	-----
192.168.1.1:0	ImpNull

192.168.1.1/32, rev 13

Local binding: label: 24001

Remote bindings: (1 peers)

Peer	Label
-----	-----
2.2.2.2:0	19

Displays content of the LFIB table on xrv3

RP/0/0/CPU0:ios#sh mpls ldp forwarding

Tue May 1 07:47:08.259 UTC

Codes:

- = GR label recovering, (!) = LFA FRR pure backup path
- { } = Label stack with multi-line output for a routing path
- G = GR, S = Stale, R = Remote LFA FRR backup

Prefix	Label	Label(s)	Outgoing	Next Hop	Flags
	In	Out	Interface	G S R	

```

1.1.1.1/32  24000  Unlabelled  Gi0/0/0/0  13.0.0.1
2.2.2.2/32  24002  Unlabelled  Gi0/0/0/2  23.0.0.2
192.168.1.1/32  24001  Unlabelled  Gi0/0/0/0  13.0.0.1

```

Displays content of the FIB(CEF) on xrv3

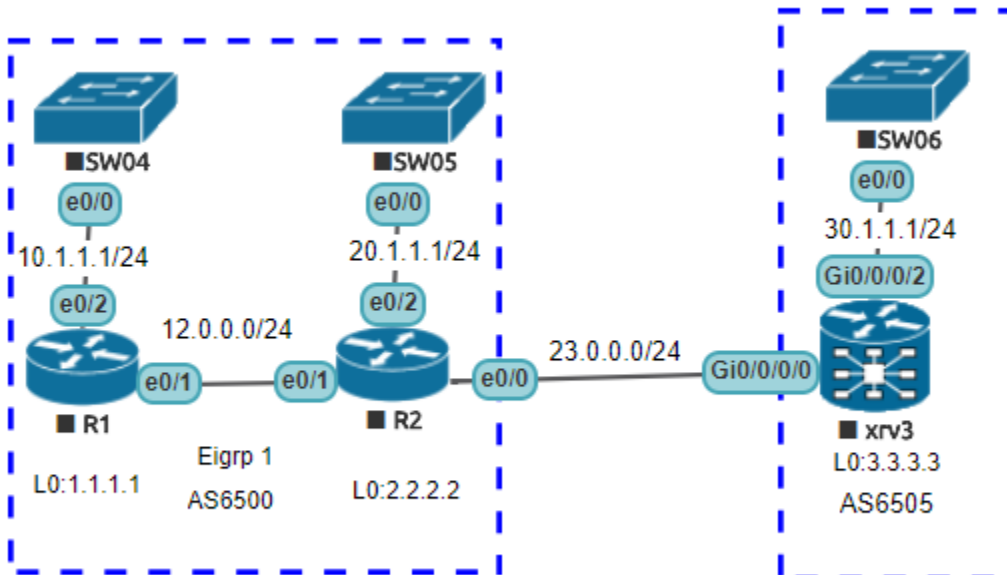
```

RP/0/0/CPU0:ios#sh cef 192.168.1.0/24
local label 24001  labels imposed {None}

```

End lab 24

Lab 25 : Configure BGP Process and BGP Peering
 Configure BGP to Advertise a Network
 Configure BGP Neighbor Authentication



Tasks :

Set interfaces ip address for all routers as per diagram

Configure eigrp 1 between R1/R2 including loopback interface

Configure BGP Peers between all routers

Configure peer authentication between R2/XRV3

Advertise Local Routes for network (10,20,30)

Set route policy to pass all traffic for both direction in EBGp routers

Verify bgp route table

Verify Learned Prefixes

Solution

Set interfaces ip address for all routers as per diagram

R1	Router(config)#int lo0 Router(config-if)#ip add 1.1.1.1 255.255.255.255 Router(config-if)#ex Router(config)#int e0/1 Router(config-if)#ip add 12.0.0.1 255.255.255.0 Router(config-if)#no sh Router(config-if)#ex R1(config)#int e0/2 R1(config-if)#ip add 10.1.1.1 255.255.255.0 R1(config-if)#no sh
R2	Router(config)#int lo0 Router(config-if)#ip add Router(config-if)#ip add 2.2.2.2 255.255.255.255 Router(config-if)#ex Router(config)#int e0/0 Router(config-if)#ip add 23.0.0.2 255.255.255.0 Router(config-if)#no sh Router(config-if)#ex Router(config)#int e0/1 Router(config-if)#ip add 12.0.0.2 255.255.255.0 Router(config-if)#no sh Router(config-if)#ex R2(config)#int e0/2 R2(config-if)#ip add 20.1.1.1 255.255.255.0 R2(config-if)#no sh
XRV3	RP/0/0/CPU0:ios(config)#int lo0 RP/0/0/CPU0:ios(config-if)#ipv4 add 3.3.3.3/32 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/0 RP/0/0/CPU0:ios(config-if)#ipv4 add 23.0.0.3/24

	<pre>RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/2 RP/0/0/CPU0:ios(config-if)#ipv4 add 30.1.1.1/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config)#commit</pre>
--	--

Configure eigrp 1 between R1/R2 including loopback interface

R1	<pre>Router(config)#router ei 1 Router(config-router)#net 1.1.1.1 0.0.0.0 Router(config-router)#net 12.0.0.0 0.0.0.255 Router(config-router)#no au</pre>
R2	<pre>Router(config)#router ei 1 Router(config-router)#net 2.2.2.2 0.0.0.0 Router(config-router)#net 12.0.0.0 0.0.0.255 Router(config-router)#no au</pre>

*May 4 07:35:59.075: %DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 12.0.0.1 (Ethernet0/1) is up: new adjacency

Configure BGP Peers between all routers

IBGP between R1/R2

R1	<pre>Router(config)#router bgp 6500 Router(config-router)#neighbor 2.2.2.2 remote 6500 Router(config-router)#neighbor 2.2.2.2 update-source lo0 Router(config-router)#address-family ipv4 unicast Router(config-router-af)#neighbor 2.2.2.2 activate</pre>
R2	<pre>Router(config)#router bgp 6500 Router(config-router)#nei 1.1.1.1 remote 6500 Router(config-router)#nei 1.1.1.1 update-source lo0 Router(config-router)#address-family ipv4 unicast Router(config-router-af)#nei 1.1.1.1 activate</pre>

*May 4 07:42:03.177: %BGP-5-ADJCHANGE: neighbor 1.1.1.1 Up

EBGP between R2/XRV3

R2	<pre>Router(config)#router bgp 6500 Router(config-router)#nei 23.0.0.3 remote-as 6505 Router(config-router)#address-family ipv4 unicast Router(config-router-af)#nei 23.0.0.3 activate</pre>
XRV3	<pre>router bgp 6505 address-family ipv4 unicast</pre>

	! neighbor 23.0.0.2 remote-as 6500 address-family ipv4 unicast
--	---

Configure peer authentication between R2/XRV3

R2	R2(config)#router bgp 6500 R2(config-router)#neighbor 23.0.0.3 password spng2
XRV3	RP/0/0/CPU0:ios(config)#router bgp 6505 RP/0/0/CPU0:ios(config-bgp)#neighbor 23.0.0.2 RP/0/0/CPU0:ios(config-bgp-nbr)#password clear spng2 RP/0/0/CPU0:ios(config-bgp-nbr)#commit

Advertise Local Routes

R1	R1(config)#router bgp 6500 R1(config-router)#address-family ipv4 unicast R1(config-router-af)#net 10.1.1.0 mask 255.255.255.0
R2	R2(config)#router bgp 6500 R2(config-router)#add ipv4 unicast R2(config-router-af)#net 20.1.1.0 mask 255.255.255.0
Xrv3	RP/0/0/CPU0:ios(config)#router bgp 6505 RP/0/0/CPU0:ios(config-bgp)#address-family ipv4 unicast RP/0/0/CPU0:ios(config-bgp-af)#net 30.1.1.0/24 RP/0/0/CPU0:ios(config-bgp-af)#commit

Set route policy to pass all traffic for both direction in EBGp routers

RP/0/0/CPU0:ios(config)#router bgp 6505

RP/0/0/CPU0:ios(config-bgp)#nei 23.0.0.2

RP/0/0/CPU0:ios(config-bgp-nbr)#address-family ipv4 unicast

RP/0/0/CPU0:ios(config-bgp-nbr-af)#route-policy spng2 in

RP/0/0/CPU0:ios(config-bgp-nbr-af)#route-policy spng2 out

RP/0/0/CPU0:ios(config-bgp-nbr-af)#commit

```
R2(config)#router bgp 6500
R2(config-router)#address-family ipv4 unicast
R2(config-router-af)#neighbor 1.1.1.1 next-hop-self
```

```
R1(config)#do sh ip rout bgp
B 30.1.1.0 [200/0] via 2.2.2.2, 00:00:15
```

Verify bgp route table

R1 :

```
R1(config)#do sh ip rout bgp
20.0.0.0/24 is subnetted, 1 subnets
B 20.1.1.0 [200/0] via 2.2.2.2, 00:07:54
30.0.0.0/24 is subnetted, 1 subnets
B 30.1.1.0 [200/0] via 2.2.2.2, 00:00:15
```

R2 :

```
R2(config-router)#do sh ip rout bgp
10.0.0.0/24 is subnetted, 1 subnets
B 10.1.1.0 [200/0] via 1.1.1.1, 00:07:05
30.0.0.0/24 is subnetted, 1 subnets
B 30.1.1.0 [20/0] via 23.0.0.3, 00:02:38
```

XRV3

```
RP/0/0/CPU0:ios(config)#do sh route bgp
```

Sun May 6 07:44:46.084 UTC

```
B 10.1.1.0/24 [20/0] via 23.0.0.2, 00:04:44
```

B 20.1.1.0/24 [20/0] via 23.0.0.2, 00:04:44

Verify Learned Prefixes

XRV3

RP/0/0/CPU0:ios#sh bgp

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 10.1.1.0/24	23.0.0.2		0	6500	i
*> 20.1.1.0/24	23.0.0.2	0		0 6500	i
*> 30.1.1.0/24	0.0.0.0	0	32768		i

Processed 3 prefixes, 3 paths

R2

R2#sh ip bgp

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i 10.1.1.0/24	1.1.1.1	0	100	0	i
*> 20.1.1.0/24	0.0.0.0	0	32768		i
*> 30.1.1.0/24	23.0.0.3	0		0 6505	i

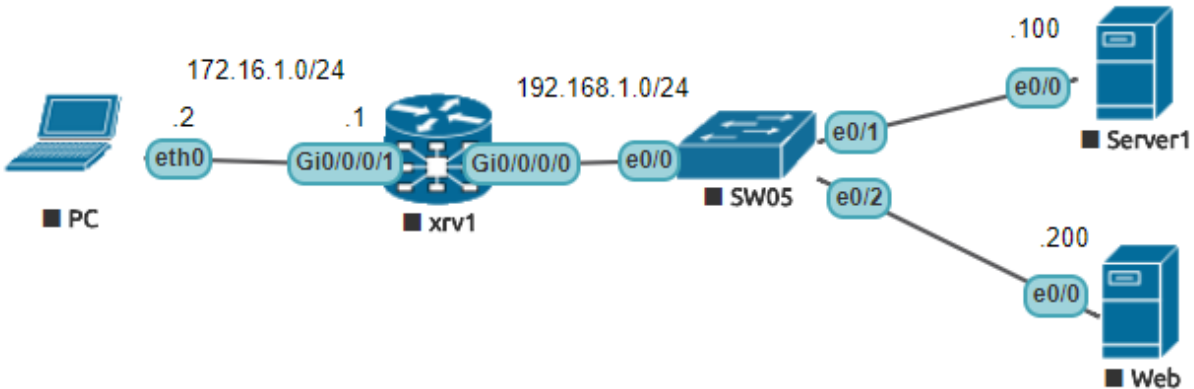
R1:

R1#sh ip bgp

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 10.1.1.0/24	0.0.0.0	0	32768		i
*>i 20.1.1.0/24	2.2.2.2	0	100	0	i
*>i 30.1.1.0/24	2.2.2.2	0	100	0 6505	i

End lab 25

Lab 26: Configure IPv4 Filtering



Tasks:

- Configure ip address for all interfaces as per diagram
- Configure access list to permit network 172.16.1.0 through R1 only
- Verify access list and interface policy
- Configure access list to deny pc to connect to web server through http
- Verify access list and interface policy

Solution:

Configure ip address for all interfaces as per diagram

pc	<pre>VPCS> ip 172.16.1.2/24 172.16.1.1 Checking for duplicate address... PC1 : 172.16.1.2 255.255.255.0 gateway 172.16.1.1</pre>
Xrv1	<pre>RP/0/0/CPU0:ios(config)#int g0/0/0/1 RP/0/0/CPU0:ios(config-if)#ipv4 add 172.16.1.1/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/0 RP/0/0/CPU0:ios(config-if)#ipv4 add 192.168.1.1/24 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#commit</pre>
Server 1	<pre>Server1(config)#int e0/0 Server1(config-if)#ip add 192.168.1.100 255.255.255.0 Server1(config-if)#no sh</pre>

web	WEB(config)#int e0/0 WEB(config-if)#ip add 192.168.1.200 255.255.255.0 WEB(config-if)#no sh
-----	---

```
RP/0/0/CPU0:ios(config)#ipv4 access-list ccna
RP/0/0/CPU0:ios(config-ipv4-acl)#permit 172.16.1.0/24
RP/0/0/CPU0:ios(config-ipv4-acl)#deny any
RP/0/0/CPU0:ios(config)#int g0/0/0/1
RP/0/0/CPU0:ios(config-if)#ipv4 access-group ccna in
RP/0/0/CPU0:ios(config-if)#commit
```

```
RP/0/0/CPU0:ios#sh access-lists ipv4
```

Mon May 7 11:39:03.442 UTC

```
ipv4 access-list ccna
```

```
10 permit ipv4 172.16.1.0 0.0.0.255 any
```

```
20 deny ipv4 any any
```

Verify access list and interface policy

```
RP/0/0/CPU0:ios#sh run int g0/0/0/1
```

Mon May 7 11:39:39.919 UTC

```
interface GigabitEthernet0/0/0/1
```

```
ipv4 address 172.16.1.1 255.255.255.0
```

```
ipv4 access-group ccna ingress
```

Configure access list to deny pc to connect to web server through http

```
RP/0/0/CPU0:ios(config)#ipv4 access-list permitweb
```

```
RP/0/0/CPU0:ios(config-ipv4-acl)#permit tcp host 172.16.1.2 host 192.168.1.200 eq www
```

```
RP/0/0/CPU0:ios(config-ipv4-acl)#deny ipv4 any any
```

```
RP/0/0/CPU0:ios(config)#int g0/0/0/0
```

```
RP/0/0/CPU0:ios(config-if)#ipv4 access-group permitweb egress
```

```
RP/0/0/CPU0:ios(config-if)#commit
```

Verify access list and interface policy

```
RP/0/0/CPU0:ios#sh access-lists ipv4 permitweb
```

```
Mon May 7 11:48:04.915 UTC
```

```
ipv4 access-list permitweb
```

```
10 permit tcp host 172.16.1.2 host 192.168.1.200 eq www
```

```
20 deny ipv4 any any
```

```
RP/0/0/CPU0:ios#sh run int g0/0/0/0
```

```
Mon May 7 11:48:41.242 UTC
```

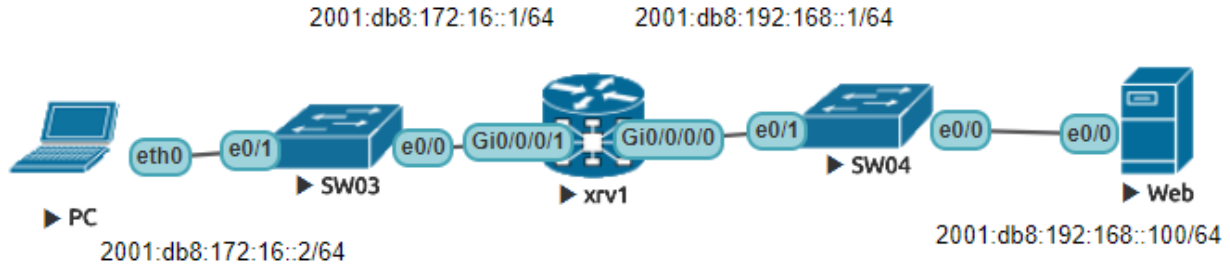
```
interface GigabitEthernet0/0/0/1
```

```
ipv4 address 172.16.1.1 255.255.255.0
```

```
ipv4 access-group permitweb egress
```

End lab 26

Lab 27: Configure IPv6 Filtering



Tasks :

Configure ip address for interface as per diagram

Configure aces list to deny pc to connect to web server through http

Verify access list and interface policy

Solution :

Configure ip address for interface as per diagram

pc	<pre>VPCS> ip 2001:db8:172:16::2/64 PC1 : 2001:db8:172:16::2/64 VPCS> save Saving startup configuration to startup.vpc . done</pre>
Xrv1	<pre>RP/0/0/CPU0:ios(config)#int g0/0/0/0 RP/0/0/CPU0:ios(config-if)#ipv6 add 2001:db8:192:168::1/64 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#ex RP/0/0/CPU0:ios(config)#int g0/0/0/1 RP/0/0/CPU0:ios(config-if)#ipv6 add 2001:db8:172:16::1/64 RP/0/0/CPU0:ios(config-if)#no sh RP/0/0/CPU0:ios(config-if)#commit</pre>
web	<pre>Web(config)#int e0/0 Web(config-if)#ipv6 add 2001:db8:192:168::100/64 Web(config-if)#no sh</pre>

Configure access list to deny PC to connect to web server through HTTP

```
RP/0/0/CPU0:ios(config)#ipv6 access-list permitweb
```

```
RP/0/0/CPU0:ios(config-ipv6-acl)#permit tcp 2001:db8:172:16:1::2/64 2001:db8:192:168:1::100/64 eq www
```

```
RP/0/0/CPU0:ios(config-ipv6-acl)#deny ipv6 any any
```

```
RP/0/0/CPU0:ios(config-ipv6-acl)#commit
```

```
RP/0/0/CPU0:ios(config)#int g0/0/0/1
```

```
RP/0/0/CPU0:ios(config-if)#ipv6 access-group permitweb ingress
```

```
RP/0/0/CPU0:ios(config-if)#commit
```

Verify access list and interface policy

```
RP/0/0/CPU0:ios#sh access-lists ipv6
```

Mon May 7 15:43:28.213 UTC

```
ipv6 access-list permitweb
```

```
10 permit tcp 2001:db8:172:16::/64 2001:db8:192:168::/64 eq www
```

```
20 deny ipv6 any any
```

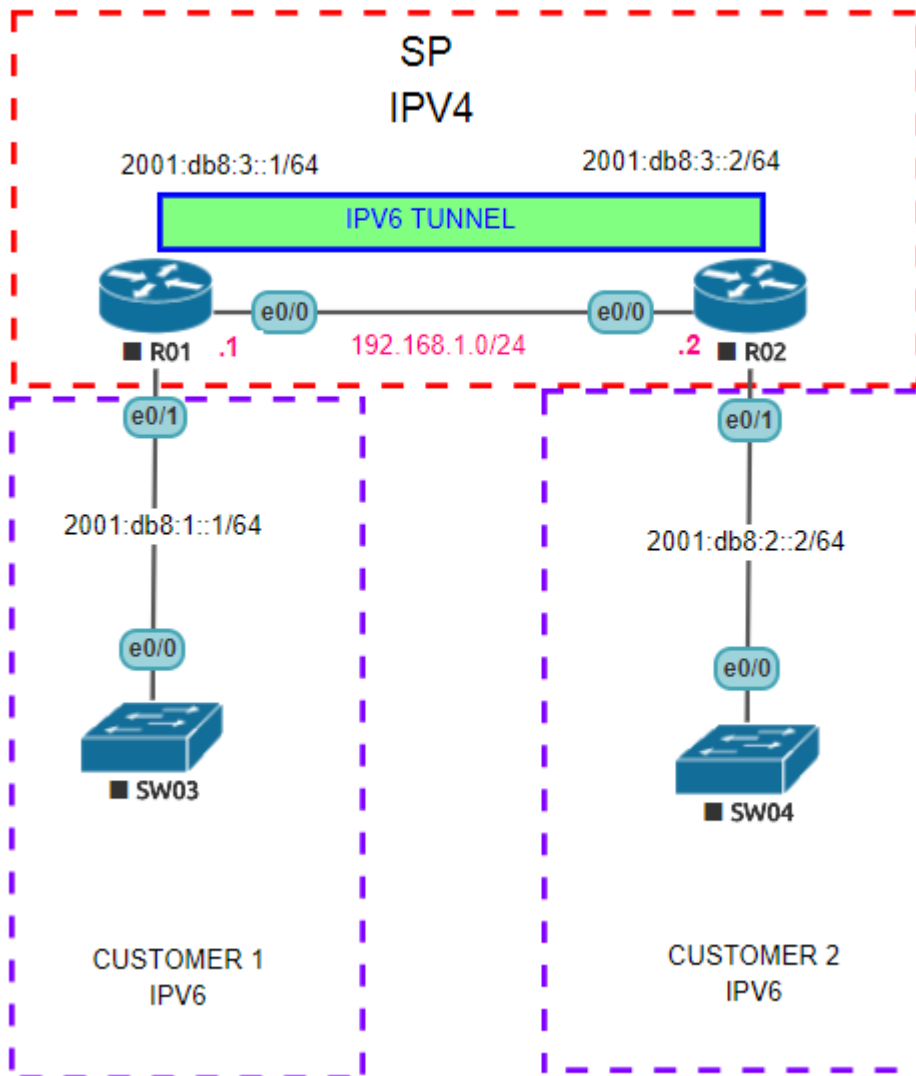
```
RP/0/0/CPU0:ios#sh ipv6 interface g0/0/0/1
```

Outgoing access list is not set

Inbound common access list is not set, access list is **permitweb**

End lab 27

Lab 28: IPv6-in-IPv4 Configuration



Tasks :

Configure ip address for connected interface for R1/R2

Create tunnel 0 between R1/R2 for ipv6 traffic between customer1 and customer 2 network

Verify connectivity

Solution :

Configure ip address for connected interface for R1/R2

R1	R1(config)#int e0/1 R1(config-if)#ipv6 add 2001:db8:1::1/64
----	--

	<pre>R1(config-if)#no sh R1(config-if)#ex R1(config)#int e0/0 R1(config-if)#ip add 192.168.1.1 255.255.255.0 R1(config-if)#no sh</pre>
R2	<pre>R2(config)#int e0/1 R2(config-if)#ipv6 add 2001:db8:2::2/64 R2(config-if)#no sh R2(config)#int e0/0 R2(config-if)#ip add 192.168.1.2 255.255.255.0 R2(config-if)#no sh</pre>

Create tunnel 0 between R1/R2 for ipv6 traffic between customer1 and customer 2 network

	<pre>R1(config)#int tunnel 0 R1(config-if)#ipv6 add 2001:db8:3::1/64 R1(config-if)#tunnel source e0/0 R1(config-if)#tunnel destination 192.168.1.2 R1(config-if)#tunnel mod ipv6ip R1(config)#ipv6 route 2001:db8:2::/64 Tunnel0 2001:db8:3::2</pre>
	<pre>R2(config)#int tunnel 0 R2(config-if)#ipv6 add 2001:db8:3::2/64 R2(config-if)#tunne source e0/0 R2(config-if)#tunnel destination 192.168.1.1 R2(config-if)#tunnel mode ipv6ip R2(config)#ipv6 route 2001:db8:1::/64 Tunnel0 2001:db8:3::1</pre>

Verify connectivity

R1#ping 2001:DB8:2::2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2001:DB8:2::2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

R1#trace 2001:DB8:2::2

Type escape sequence to abort.

Tracing the route to 2001:DB8:2::2

1 2001:DB8:3::2 7 msec 5 msec 5 msec

End lab 28

Lab 29 :IOS XR Software



Task :

Deactivated software package xrvr-mgbl-x-6.0.1

Install software package

Change the hostname

Roll back configuration to the last commit made

Solution:

Deactivated software package xrvr-mgbl-x-6.0.1

```
RP/0/0/CPU0:ios(admin)#install deactivate disk0:xrvr-mgbl-x-6.0.1
```

```
RP/0/0/CPU0:ios(admin)#install commit
```

```
RP/0/0/CPU0:ios(admin)#install remove disk0:xrvr-mgbl-x-6.0.1
```

Install software package

```
RP/0/0/CPU0:ios(admin)#install add disk0::xrvr-mgbl-x-6.0.1
```

```
RP/0/0/CPU0:ios(admin)#install commit
```

```
RP/0/0/CPU0:ios(admin)#install activate disk0::xrvr-mgbl-x-6.0.1
```

Change the hostname

```
RP/0/0/CPU0:ios(config)#host SPNG2
```

```
RP/0/0/CPU0:ios(config)#commit
```

Tue May 8 15:26:29.120 UTC

```
RP/0/0/CPU0:SPNG2(config)#
```

Roll back configuration to the last commit made

```
RP/0/0/CPU0:SPNG2#rollback configuration last 1
```

Tue May 8 15:27:38.335 UTC

Loading Rollback Changes.

Loaded Rollback Changes in 1 sec

Committing.

1 items committed in 1 sec (0)items/sec

Updating.

Updated Commit database in 1 sec

Configuration successfully rolled back 1 commits

End lab 29