

UNIVERSITY OF MUMBAI
DEPARTMENT OF COMPUTER SCIENCE



M.Sc. Computer Science – Semester I
SOFTWARE DEFINED NETWORKING
JOURNAL
2022-2023

Seat No. 10137



मुंबई विद्यापीठ
University of Mumbai
Re-accredited with A++ Grade
(CGPA 3.65) by NAAC (3rd Cycle 2021)



UNIVERSITY OF MUMBAI
DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE

This is to certify that the work entered in this journal was done in the University Department of Computer Science laboratory by Mr. **Kalpesh Ananda Koli** Seat No. **10137** for the course of M.Sc. Computer Science - Semester I (CBCS) (Revised) during the academic year 2022- 2023 in a satisfactory manner.

Subject In-charge

Head of Department

External Examiner

Index

Sr. no.	Name of the practical	Page No.	Date	Sign
1	Implement IP SLA (IP Service Level Agreement)	4-10		
2	Implement IPv4 ACLs 1.Standard 2.Extended	11-19		
3	Implement SPAN Technologies (Switch Port Analyzer)	20-28		
	Implement SNMP and SYSLOG			
	Implement Flexible NETFLOW			
4	Implement SNMP and SYSLOG	29-40		
	Implement Flexible NETFLOW			
	Implement a GRE Tunnel			
5	Implement Inter-VLAN Routing	41-58		
6	Observe STP Topology Changes and Implement RSTP	59-63		
7	Implement EtherChannel	64-73		
	Tune and Optimize EtherChannel Operations			
8	Implement Single channel OSPFv2	74-78		
	Implement Multi channel OSPFv2			
9	Implement BGP Communities	79-82		
10	Implement IPsec Site-to-Site VPNs	83-87		

PRACTICAL 1

AIM:- Implementing IP SLA (IP Service Level Agreement)

Objectives

- Configure and verify the IP SLA feature.
- Test the IP SLA tracking feature.
- Verify the configuration and operation using show and debug commands.

Background

You want to experiment with the Cisco IP Service Level Agreement (SLA) feature to study how it could be of value to your organization.

At times, a link to an ISP could be operational, yet users cannot connect to any other outside Internet resources. The problem might be with the ISP or downstream from them. Although policy-based routing (PBR) can be implemented to alter path control, you will implement the Cisco IOS SLA feature to monitor this behavior and intervene by injecting another default route to a backup ISP.

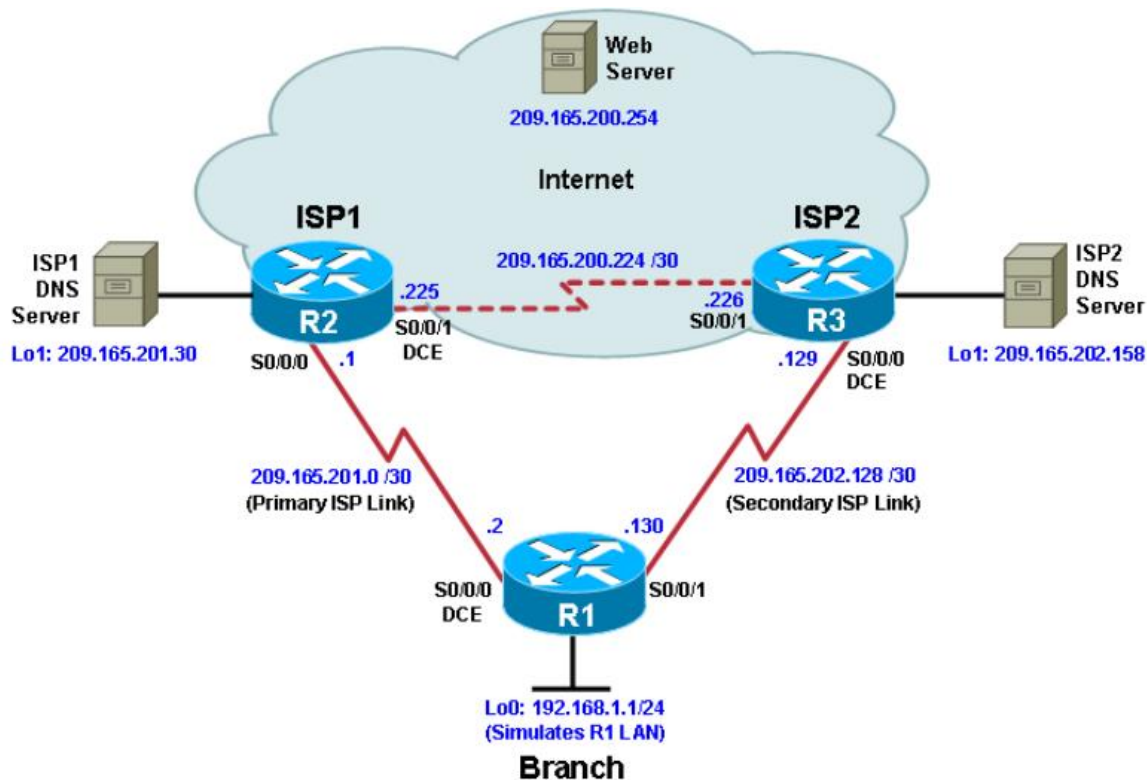
To test this, you have set up a three-router topology in a lab environment. Router R1 represents a branch office connected to two different ISPs. ISP1 is the preferred connection to the Internet, while ISP2 provides a backup link. ISP1 and ISP2 can also interconnect, and both can reach the web server. To monitor ISP1 for failure, you will configure IP SLA probes to track the reachability to the ISP1 DNS server. If connectivity to the ISP1 server fails, the SLA probes detect the failure and alter the default static route to point to the ISP2 server.

Note: This lab uses Cisco 1941 routers with Cisco IOS Release 15.2 with IP Base. Depending on the router or switch model and Cisco IOS Software version, the commands available and output produced might vary from what is shown in this lab.

Required Resources

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

Topology:



Step 1: Configure loopbacks and assign addresses.

a. Cable the network as shown in the topology diagram. Erase the startup configuration and reload each router to clear the previous configurations. Using the addressing scheme in the diagram, create the loopback interfaces and apply IP addresses to them as well as the serial interfaces on R1, ISP1, and ISP2.

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

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

b. Verify the configuration by using the show interfaces description command. The output from router R1 is shown here as an example.

```
R1# show interfaces description | include up
Se0/0/0          up      up      R1 --> ISP1
Se0/0/1          up      up      R1 --> ISP2
Lo0              up      up      R1 LAN
R1#
```

All three interfaces should be active. Troubleshoot if necessary.

Step 2: Configure static routing.

The current routing policy in the topology is as follows:

Router R1 establishes connectivity to the Internet through ISP1 using a default static route. ISP1 and ISP2 have dynamic routing enabled between them, advertising their respective public address pools.

ISP1 and ISP2 both have static routes back to the ISP LAN.

Note: For the purpose of this lab, the ISPs have a static route to an RFC 1918 private network address on the branch router R1. In an actual branch implementation, Network Address Translation (NAT) would be configured for all traffic exiting the branch LAN. Therefore, the static routes on the ISP routers would be pointing to the provided public pool of the branch office.

- a) Implement the routing policies on the respective routers. You can copy and paste the following configurations.
EIGRP neighbor relationship messages on ISP1 and ISP2 should be generated.
Troubleshoot if necessary.
- b) The Cisco IOS IP SLA feature enables an administrator to monitor network performance between Cisco devices (switches or routers) or from a Cisco device to a remote IP device. IP SLA probes continuously check the reachability of a specific destination, such as a provider edge router interface, the DNS server of the ISP, or any other specific destination, and can conditionally announce a default route only if the connectivity is verified.

Before implementing the Cisco IOS SLA feature, you must verify reachability to the Internet servers. From router R1, ping the web server, ISP1 DNS server, and ISP2 DNS server to verify connectivity. You can copy the following Tcl script and paste it into R1.

All pings should be successful. Troubleshoot if necessary.

- c) Trace the path taken to the web server, ISP1 DNS server, and ISP2 DNS server. You can copy the following Tcl script and paste it into R1.
All traffic is routed to the ISP1 router.

Step 3: Configure IP SLA probes.

When the reachability tests are successful, you can configure the Cisco IOS IP SLAs probes. Different types of probes can be created, including FTP, HTTP, and jitter probes. In this scenario, you will configure ICMP echo probes.

- a) Create an ICMP echo probe on R1 to the primary DNS server on ISP1 using the ip sla command.

The operation number of 11 is only locally significant to the router. The frequency 10 command schedules the connectivity test to repeat every 10 seconds. The probe is scheduled to start now and to run forever.

- b) Verify the IP SLAs configuration of operation 11 using the show ip sla configuration 11 command.

The output lists the details of the configuration of operation 11. The operation is an ICMP echo to 209.165.201.30, with a frequency of 10 seconds, and it has already started (the start time has already passed).

- c) Issue the show ip sla statistics command to display the number of successes, failures, and results of the latest operations.

You can see that operation 11 has already succeeded five times, has had no failures, and the last operation returned an OK result.

- d) Although not actually required because IP SLA session 11 alone could provide the desired fault tolerance, create a second probe, 22, to test connectivity to the second DNS server located on router ISP2.

- e) Verify the new probe using the show ip sla configuration and show ip sla statistics commands.

The output lists the details of the configuration of operation 22. The operation is an ICMP echo to 209.165.202.158, with a frequency of 10 seconds, and it has already started (the start time has already passed). The statistics also prove that operation 22 is active.

Step 4: Configure tracking options.

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

- a. On R1, remove the current default route and replace it with a floating static route having an administrative distance of 5.
- b. Verify the routing table.
Notice that the default static route is now using the route with the administrative distance of 5. The first tracking object is tied to IP SLA object 11.
- c. From global configuration mode on R1, use the **track 1 ip sla 11 reachability** command to enter the config-track subconfiguration mode.
- d. Specify the level of sensitivity to changes of tracked objects to 10 seconds of down delay and 1 second of up delay using the **delay down 10 up 1** command. The delay helps to alleviate the effect of flapping objects—objects that are going down and up rapidly. In this situation, if the DNS server fails momentarily and comes back up within 10 seconds, there is no impact.
- e. To view routing table changes as they happen, first enable the **debug ip routing** command.
- f. Configure the floating static route that will be implemented when tracking object 1 is active. Use the **ip route 0.0.0.0 0.0.0.0 209.165.201.1 2 track 1** command to create a floating static default route via 209.165.201.1 (ISP1). Notice that this command references the tracking object number 1, which in turn references IP SLA operation number 11.

Notice that the default route with an administrative distance of 5 has been immediately flushed because of a route with a better admin distance. It then adds the new default route with the admin distance of 2.
- g. Repeat the steps for operation 22, track number 2, and assign the static route an admin distance higher than track 1 and lower than 5. On R1, copy the following configuration, which sets an admin distance of 3.

- h. Verify the routing table again. Although a new default route was entered, its administrative distance is not better than 2. Therefore, it does not replace the previously entered default route.

Step 5: Verify IP SLA operation.

In this step you observe and verify the dynamic operations and routing changes when tracked objects fail. The following summarizes the process:

- Disable the DNS loopback interface on ISP1 (R2).
- Observe the output of the **debug** command on R1.
- Verify the static route entries in the routing table and the IP SLA statistics of R1.
- Re-enable the loopback interface on ISP1 (R2) and again observe the operation of the IP SLA tracking feature.

- a. On ISP1, disable the loopback interface 1.
- b. On R1, observe the **debug** output being generated. Recall that R1 will wait up to 10 seconds before initiating action therefore several seconds will elapse before the output is generated.

The tracking state of track 1 changes from up to down. This is the object that tracked reachability for IP SLA object 11, with an ICMP echo to the ISP1 DNS server at 209.165.201.30.

R1 then proceeds to delete the default route with the administrative distance of 2 and installs the next highest default route to ISP2 with the administrative distance of 3.

- c. On R1, verify the routing table.
The new static route has an administrative distance of 3 and is being forwarded to ISP2 as it should.
- d. Verify the IP SLA statistics.
Notice that the latest return code is **Timeout** and there have been 45 failures on IP SLA object 11.
- e. On R1, initiate a trace to the web server from the internal LAN IP address.

This confirms that traffic is leaving router R1 and being forwarded to the ISP2 router.

- f. On ISP1, re-enable the DNS address by issuing the **no shutdown** command on the loopback 1 interface to examine the routing behavior when connectivity to the ISP1 DNS is restored.

Now the IP SLA 11 operation transitions back to an up state and reestablishes the default static route to ISP1 with an administrative distance of 2.

- g. Again examine the IP SLA statistics.

The IP SLA 11 operation is active again, as indicated by the OK return code, and the number of successes is incrementing.

- h. Verify the routing table.

The default static through ISP1 with an administrative distance of 2 is reestablished.

There are many possibilities available with object tracking and Cisco IOS IP SLAs. As shown in this lab, a probe can be based on reachability, changing routing operations, and path control based on the ability to reach an object. However, Cisco IOS IP SLAs also allow paths to be changed based on network conditions such as delay, load, and other factors.

Before deploying a Cisco IOS IP SLA solution, the impact of the additional probe traffic being generated should be considered, including how that traffic affects bandwidth utilization, and congestion levels. Tuning the configuration (for example, with the **delay** and **frequency** commands) is critical to mitigate possible issues related to excessive transitions and route changes in the presence of flapping tracked objects.

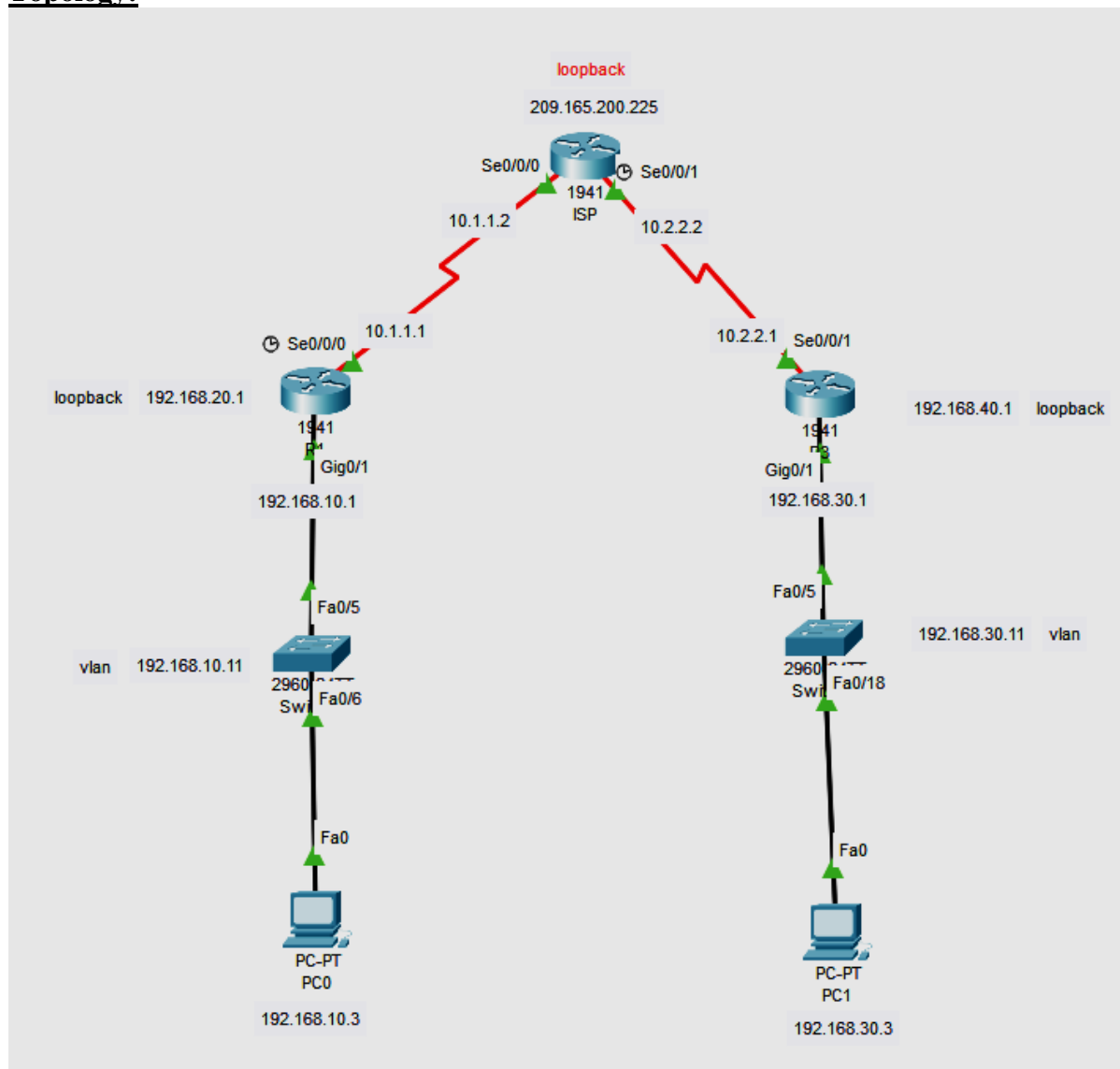
The benefits of running IP SLAs should be carefully evaluated. The IP SLA is an additional task that must be performed by the router's CPU. A large number of intensive SLAs could be a significant burden on the CPU, possibly interfering with other router functions and having detrimental impact on the overall router performance. The CPU load should be monitored after the SLAs are deployed to verify that they do not cause excessive utilization of the router CPU.

Practical 2

Aim:- Implement IPv4 ACLs

1.Standard

Topology:



Addressing Table:

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/1	192.168.10.1	255.255.255.0	N/A
	Lo0	192.168.20.1	255.255.255.0	N/A
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A
ISP	S0/0/0	10.1.1.2	255.255.255.252	N/A
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	N/A
	Lo0	209.165.200.225	255.255.255.224	N/A
R3	G0/1	192.168.30.1	255.255.255.0	N/A
	Lo0	192.168.40.1	255.255.255.0	N/A
	S0/0/1	10.2.2.1	255.255.255.252	N/A
S1	VLAN 1	192.168.10.11	255.255.255.0	192.168.10.1
S3	VLAN 1	192.168.30.11	255.255.255.0	192.168.30.1
PC-A	NIC	192.168.10.3	255.255.255.0	192.168.10.1
PC-C	NIC	192.168.30.3	255.255.255.0	192.168.30.1

Part 1: Set Up the Topology and Initialize Devices

In Part 1, you set up the network topology and clear any configurations, if necessary.

Step 1: Cable the network as shown in the topology.**Step 2: Initialize and reload the routers and switches.**

```
Router(config)#int Lo0
Router(config-if)#ip address 192.168.20.1 255.255.255.0
Router(config-if)#no shut
```

```
ISP(config)#int Lo0
ISP(config-if)#ip address 209.165.200.225 255.255.255.224
ISP(config-if)#no shut
ISP(config-if)#
R3(config)#int Lo0
R3(config-if)#ip address 192.168.40.1 255.255.255.0
R3(config-if)#no shut
R3(config-if)#
```

Step 2:

```
R1(config)# router rip
R1(config-router)# version 2
R1(config-router)# network 192.168.10.0
R1(config-router)# network 192.168.20.0
R1(config-router)# network 10.1.1.0
```

```
ISP(config)# router rip
ISP(config-router)# version 2
ISP(config-router)# network 209.165.200.224
ISP(config-router)# network 10.1.1.0
ISP(config-router)# network 10.2.2.0
```

```
R3(config)# router rip
R3(config-router)# version 2
R3(config-router)# network 192.168.30.0
R3(config-router)# network 192.168.40.0
R3(config-router)# network 10.1.1.0
```

```
S2(config)#int vlan 1
S2(config-if)#ip address 192.168.30.11 255.255.255.0
S2(config-if)#ip default-gateway 192.168.30.1
Switch(config)#int vlan 1
Switch(config-if)#ip address 192.168.10.11 255.255.255.0
Switch(config-if)#ip default-gateway 192.168.10.1
Switch(config)#
```

Step 3

```
R1(config)#access-list 1 remark Allow R3 LANs Access
R1(config)#access-list ?
<1-99> IP standard access list
<100-199> IP extended access list
R1(config)#access-list 1 permit 192.168.30.0 0.0.0.255
R1(config)#access-list 1 permit 192.168.40.0 0.0.0.255
R1(config)#access-list 1 deny?
deny
R1(config)#access-list 1 deny any
R1(config)#exit
```

```
R1(config)#int g0/1
R1(config-if)#ip access-group 1 out
R1(config-if)#exit
```

```
R1#show access-list 1
```

```
Standard IP access list 1
permit 192.168.30.0 0.0.0.255
permit 192.168.40.0 0.0.0.255
deny any
```

```
R1#show ip interface g0/1
GigabitEthernet0/1 is up, line protocol is up (connected)
Internet address is 192.168.10.1/24
Broadcast address is 255.255.255.255
Address determined by setup command
MTU is 1500 bytes
Helper address is not set
Directed broadcast forwarding is disabled
Outgoing access list is 1
Inbound access list is not set
Proxy ARP is enabled
Security level is default
Split horizon is enabled
ICMP redirects are always sent
ICMP unreachable are always sent
ICMP mask replies are never sent
IP fast switching is disabled
IP fast switching on the same interface is disabled
IP Flow switching is disabled
IP Fast switching turbo vector
IP multicast fast switching is disabled
IP multicast distributed fast switching is disabled
Router Discovery is disabled
```

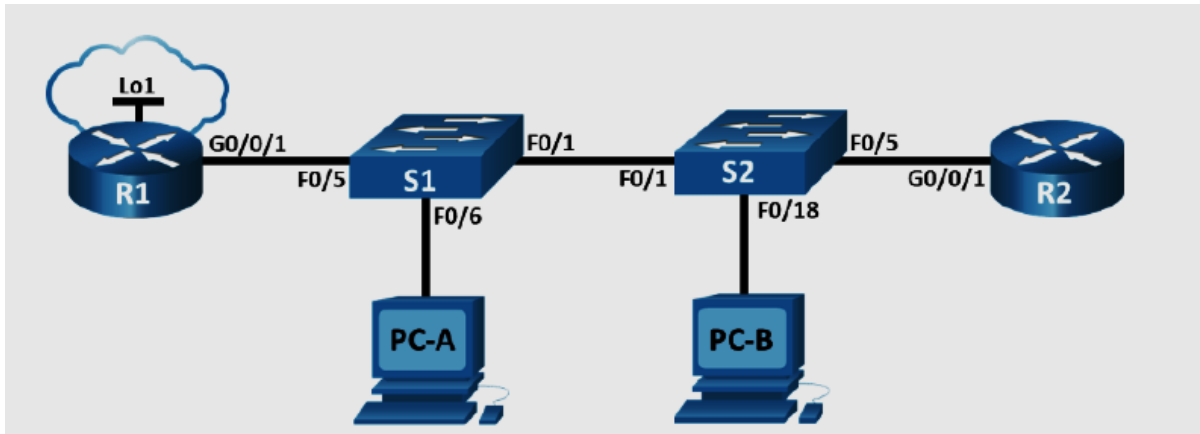
```
R3#ping
Protocol [ip]:
Target IP address: 192.168.10.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
```

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 2/16/23 ms

2.Extended

Topology:



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0/1	N/A	N/A	N/A
	G0/0/1.20	10.20.0.1	255.255.255.0	
	G0/0/1.30	10.30.0.1	255.255.255.0	
	G0/0/1.40	10.40.0.1	255.255.255.0	
	G0/0/1.1000	N/A	N/A	
	Loopback1	172.16.1.1	255.255.255.0	
R2	G0/0/1	10.20.0.4	255.255.255.0	N/A
S1	VLAN 20	10.20.0.2	255.255.255.0	10.20.0.1
S2	VLAN 20	10.20.0.3	255.255.255.0	10.20.0.1
PC-A	NIC	10.30.0.10	255.255.255.0	10.30.0.1
PC-B	NIC	10.40.0.10	255.255.255.0	10.40.0.1

VLAN Table

VLAN	Name	Interface Assigned
20	Management	S2: F0/5
30	Operations	S1: F0/6

VLAN	Name	Interface Assigned
40	Sales	S2: F0/18
999	ParkingLot	S1: F0/2-4, F0/7-24, G0/1-2 S2: F0/2-4, F0/6-17, F0/19-24, G0/1-2
1000	Native	N/A

Part 1: Build the Network and Configure Basic Device Settings.

Step 1: Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram, and cable as necessary.

Step 2: Configure basic settings for each router.

Step 3: Configure basic settings for each switch.

Part 2: Configure VLANs on the Switches

Step 1: Create VLANs on both switches

```

S1(config)# vlan 20
S1(config-vlan)# name Management
S1(config-vlan)# vlan 30
S1(config-vlan)# name Operations
S1(config-vlan)# vlan 40
S1(config-vlan)# name Sales
S1(config-vlan)# vlan 999
S1(config-vlan)# name ParkingLot
S1(config-vlan)# vlan 1000
S1(config-vlan)# name Native
S1(config-vlan)# exit

```

Do the same for s2

Assign all unused ports on the switch to the Parking Lot VLAN, configure them for static access mode, and administratively deactivate them.

Note: The interface range command is helpful to accomplish this task with as few commands as necessary.

Step 2: Assign VLANs to the correct switch interfaces.

Part 3: Configure Trunking

step 1: Manually configure trunk interface F0/1.

```
S1(config)# interface f0/1
```

```
S1(config-if)# switchport mode trunk
```



```
S1(config-if)# switchport trunk native vlan 1000
S1(config-if)# switchport trunk allowed vlan 20,30,40,1000
S1# show interfaces trunk
```

Step 2: Manually configure S1's trunk interface F0/5.

Configure S1's interface F0/5 with the same trunk parameters as F0/1. This is the trunk to the router.

```
S1(config)# interface f0/5
S1(config-if)#switchport mode trunk
S1(config-if)# switchport trunk native vlan 1000
S1(config-if)# switchport trunk allowed vlan 20,30,40,1000
Save the running configuration to the startup configuration file.
S1# copy running-config startup-config
Issue the show interfaces trunk command to verify trunking.
```

Part 4: Configure Routing

Step 1: Configure Inter-VLAN Routing on R1.

Activate interface G0/0/1 on the router.

```
R1(config)# interface g0/0/1
R1(config-if)# no shutdown
```

Configure sub-interfaces for each VLAN as specified in the IP addressing table. All sub-interfaces use 802.1Q encapsulation. Ensure the sub-interface for the native VLAN does not have an IP address assigned. Include a description for each sub-interface.

```
R1(config)# interface g0/0/1.20
R1(config-subif)# description Management Network
R1(config-subif)# encapsulation dot1q 20
R1(config-subif)# description Operations Network
R1(config-subif)# ip address 10.30.0.1 255.255.255.0
R1(config-subif)# interface g0/0/1.40
R1(config-subif)# encapsulation dot1q 40
R1(config-subif)# description Sales Network
R1(config-subif)# ip address 10.40.0.1 255.255.255.0
R1(config-subif)# interface g0/0/1.1000
R1(config-subif)# encapsulation dot1q 1000 native
R1(config-subif)# description Native VLAN
Configure interface Loopback 1 on R1 with addressing from the table above.
```

```
R1(config)# interface Loopback 1
R1(config-if)# ip address 172.16.1.1 255.255.255.0
Use the show ip interface brief command to verify the sub-interfaces are operational.
```

R1# show ip interface brief

Step 2: Configure the R2 interface g0/0/1 using the address from the table and a default route with the next hop 10.20.0.1

R2(config)# **interface g0/0/1**

R2(config-if)# ip address 10.20.0.4 255.255.255.0

R2(config-if)# **no shutdown**

R2(config-if)# **exit**

R2(config)# **ip route 0.0.0.0 0.0.0.0 10.20.0.1**

Part 5: Verify Connectivity

Step 1: Configure PC hosts.

Refer to the Addressing Table for PC host address information.

Part 6: Configure and Verify Extended Access Control Lists.

When basic connectivity is verified, the company requires the following security policies to be implemented

Policy 1: The Sales Network is not allowed to SSH to the Management Network (but other SSH is allowed).

Policy 2: The Sales Network is not allowed to access IP addresses in the Management network using any web protocol (HTTP/HTTPS). The Sales Network is also not allowed to access R1 interfaces using any web protocol. All other web traffic is allowed (note – Sales can access the Loopback 1 interface on R1).**Policy 3**

Policy 4: The Operations network is not allowed to send ICMP echo-requests to the Sales network. ICMP echo requests to other destinations are allowed.

Step 1: Analyze the network and the security policy requirements to plan ACL implementation. Answers may vary. The requirements listed above require two extended access lists to be implemented. Following the guidance of placing extended access lists as close to the source of the traffic to be filtered as possible, these ACLs will go on interfaces G0/0/0.30 and G0/0/0.40.

Step 2: Develop and apply extended access lists that will meet the security policy statements.

Answers may vary. The ACLs should be similar to the following:

R1(config)# access-list 101 remark ACL 101 fulfills policies 1, 2, and 3

R1(config)# access-list 100 deny tcp 10.40.0.0 0.0.0.255 10.20.0.0 0.0.0.255 eq 22

```

R1(config)# access-list 100 deny tcp 10.40.0.0 0.0.0.255 10.20.0.0 0.0.0.255 eq 80
R1(config)# access-list 100 deny tcp 10.40.0.0 0.0.0.255 10.30.0.0 0.0.0.0 eq 443
R1(config)# access-list 100 deny tcp 10.40.0.0 0.0.0.255 10.40.0.0 0.0.0.0 eq 80
R1(config)# access-list 100 deny tcp 10.40.0.0 0.0.0.255 10.20.0.0 0.0.0.255 eq 443
R1(config)# access-list 100 deny tcp 10.40.0.0 0.0.0.255 10.30.0.0 0.0.0.0 eq 80
R1(config)# access-list 100 deny tcp 10.40.0.0 0.0.0.255 10.40.0.0 0.0.0.0 eq 443
R1(config)# access-list 100 deny icmp 10.40.0.0 0.0.0.255 10.20.0.0 0.0.0.255 echo
R1(config)# access-list 100 deny icmp 10.40.0.0 0.0.0.255 10.30.0.0 0.0.0.255 echo
R1(config)# access-list 100 permit ip any any
R1(config)# interface g0/0/1.40
R1(config-subif)# ip access-group 100 in

R1(config)# access-list 102 deny icmp 10.30.0.0 0.0.0.255 10.40.0.0 0.0.0.255 echo
R1(config)# access-list 102 permit ip any any
R1(config)# interface g0/0/1.30
R1(config-subif)# ip access-group 101 in

```

Step 3: Verify security policies are being enforced by the deployed access lists.

Run the following tests. The expected results are shown in the table:

From	Protocol	Destination	Result
PC-A	Ping	10.40.0.10	Fail
PC-A	Ping	10.20.0.1	Success
PC-B	Ping	10.30.0.10	Fail
PC-B	Ping	10.20.0.1	Fail
PC-B	Ping	172.16.1.1	Success
PC-B	HTTPS	10.20.0.1	Fail
PC-B	HTTPS	172.16.1.1	Success

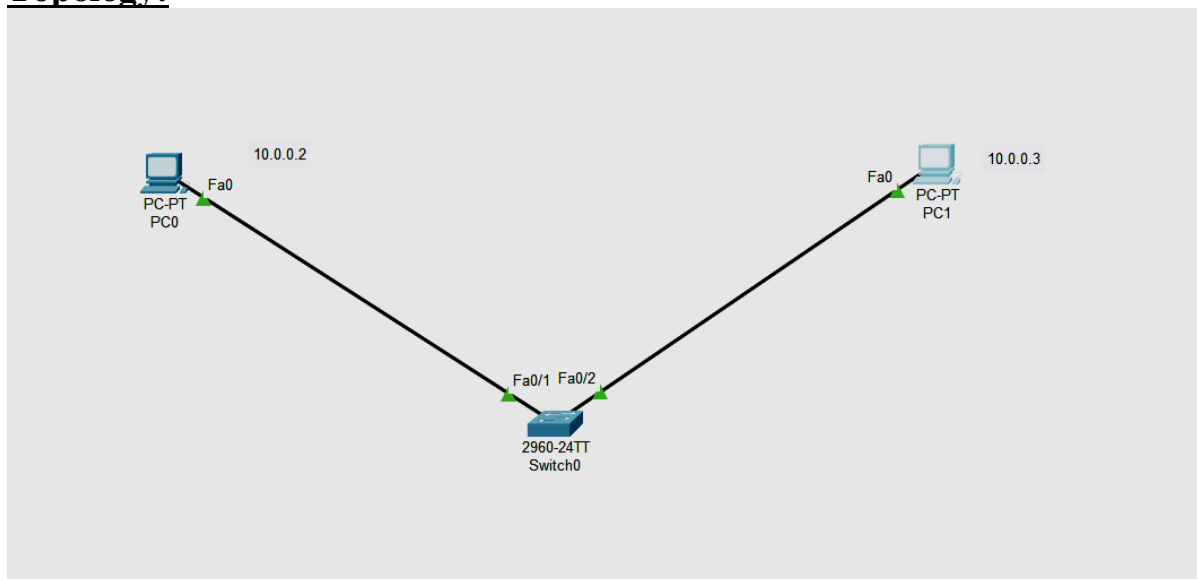
Practical 3

Aim:-

- IMPLEMENT SPAN TECHNOLOGIES (SWITCH PORT ANALYZER)
- IMPLEMENTATION OF SNMP AND SYSLOG
- IMPLEMENT FLEXIBLE NETFLOW

a) IMPLEMENT SPAN TECHNOLOGIES (SWITCH PORT ANALYZER)

Topology:



Addressing Table:

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0	10.0.0.3	255.0.0.0	N/A
PC0	Fa0	10.0.0.2	255.0.0.0	10.0.0.1
Switch0	Fa0/1	N/A	255..0.0.0	N/A
	Fa0/2	N/A	255..0.0.0	N/A

Part 1: Build the Network and verify Connectivity

In Part 1, you will set up the network topology and configure basic settings, such as the interface IP addresses.

Set ip address of PC0 as 10.0.0.2 and its default gateways as 10.0.0.1 and assign ip address of Router as 10.0.0.3

Part 2: monitoring switch port analyser working in switch 0:

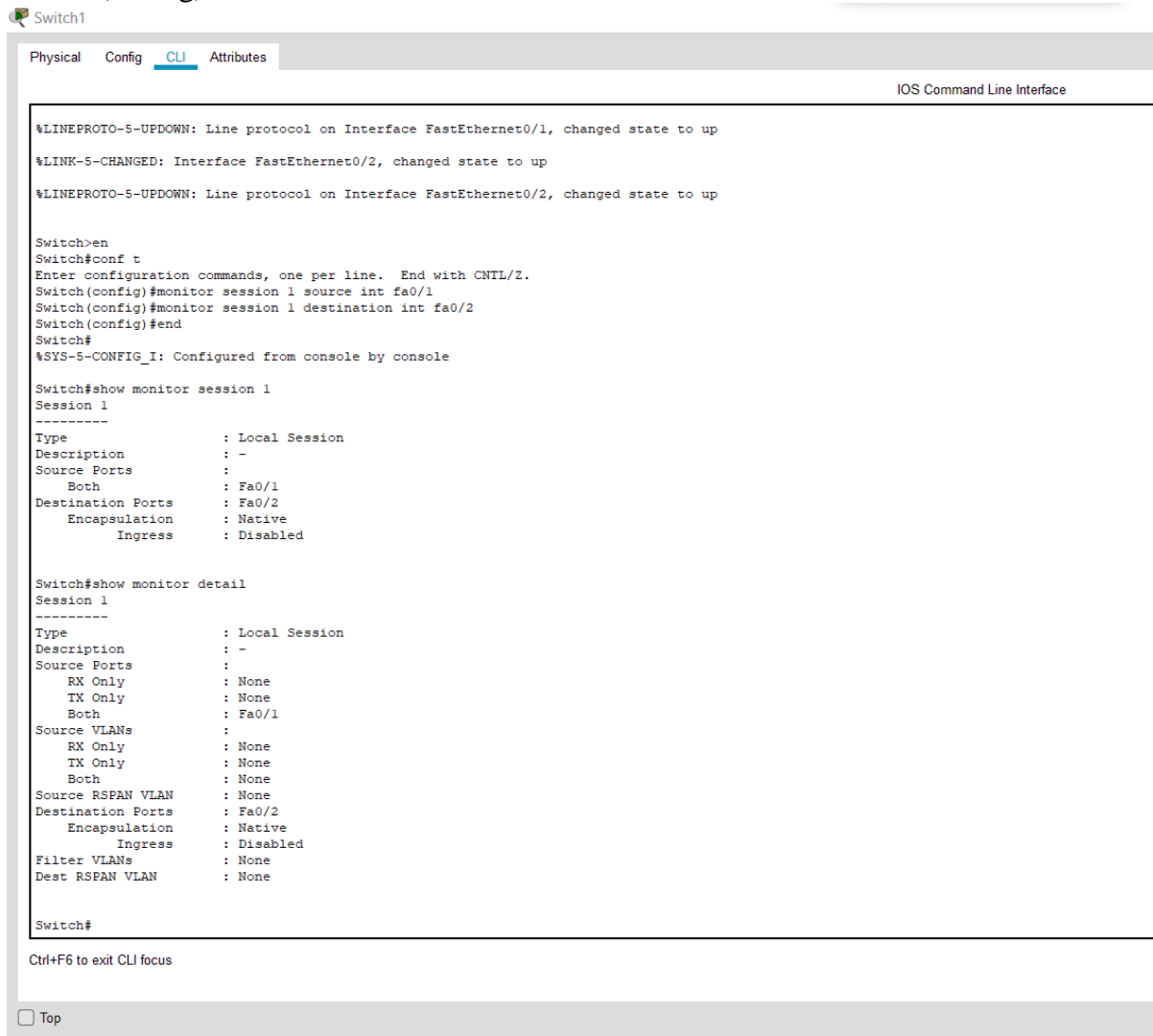
Switch>enable

Switch#config terminal

Switch(config)# monitor session 1 source int fa0/1

Switch(config)# monitor session 1 destination int fa0/2

Switch(config)#end



```

Switch1
Physical Config CLI Attributes
IOS Command Line Interface

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/2, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up

Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#monitor session 1 source int fa0/1
Switch(config)#monitor session 1 destination int fa0/2
Switch(config)#end
Switch#
%SYS-5-CONFIG_I: Configured from console by console

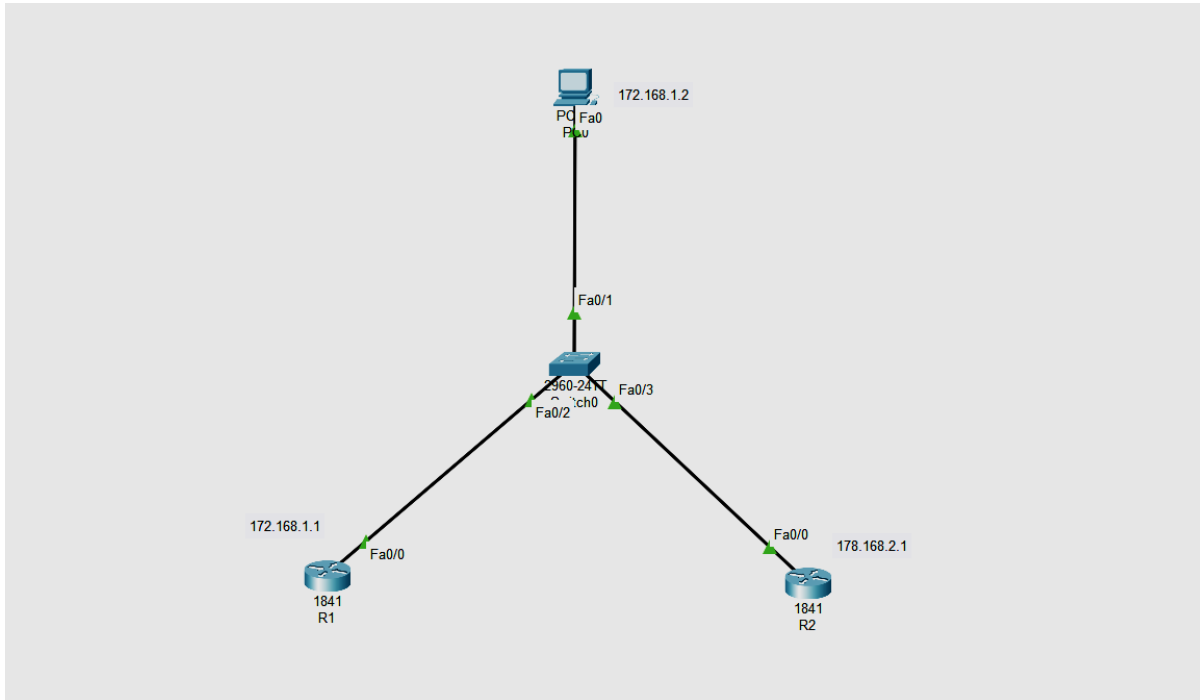
Switch#show monitor session 1
Session 1
-----
Type                : Local Session
Description          : -
Source Ports         :
    Both             : Fa0/1
Destination Ports    : Fa0/2
Encapsulation        : Native
    Ingress           : Disabled

Switch#show monitor detail
Session 1
-----
Type                : Local Session
Description          : -
Source Ports         :
    RX Only           : None
    TX Only           : None
    Both              : Fa0/1
Source VLANs         :
    RX Only           : None
    TX Only           : None
    Both              : None
Source RSPAN VLAN     : None
Destination Ports    : Fa0/2
Encapsulation        : Native
    Ingress           : Disabled
Filter VLANs         : None
Dest RSPAN VLAN       : None

Switch#
Ctrl+F6 to exit CLI focus
Top

```

b) IMPLEMENTATION OF SNMP AND SYSLOG



Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0	172.168.1.1	255.255.0.0	N/A
R2	Fa0	172.168.2.1	255.255.0.0	N/A
PC0	Fa0	172.168.1.2	255.255.0.0	172.168.1.1
Switch0	Fa0/1	N/A	255..0.0.0	N/A
	Fa0/2	N/A	255..0.0.0	N/A
	Fa0/3	N/A	255..0.0.0	N/A

Go to r1 or r2 any can be taken

R1>enable

R1#config terminal

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

R1(config)#int fa0/1

R1(config-if)#ip address 172.168.1.1 255.255.0.0

R1(config-if)#no shut

R1(config-if)#exit

R1(config)#snmp-server community read ro

R1(config)#snmp-server community write rw

PC0

Physical Config **Desktop** Programming Attributes

MIB Browser

Address: 172.168.1.1 OID: 1.3.6.1.2.1.1.5.0

Advanced... Operations: Set GO

SNMP MIBs

- ▼ MIB Tree
 - ▼ router_std MIBs
 - ▼ iso
 - ▼ org
 - ▼ dod
 - ▼ internet
 - ▼ mgmt
 - ▼ mib-2
 - ▼ system
 - sysName
 - sysObjectID
 - sysUpTime
 - sysContact
 - sysName
 - sysLocation
 - interfaces
 - ip
 - ospf
 - rip2
 - private
 - > router_advip MIBs
 - > switch_L2 MIBs
 - > switch_multiLayer MIBs

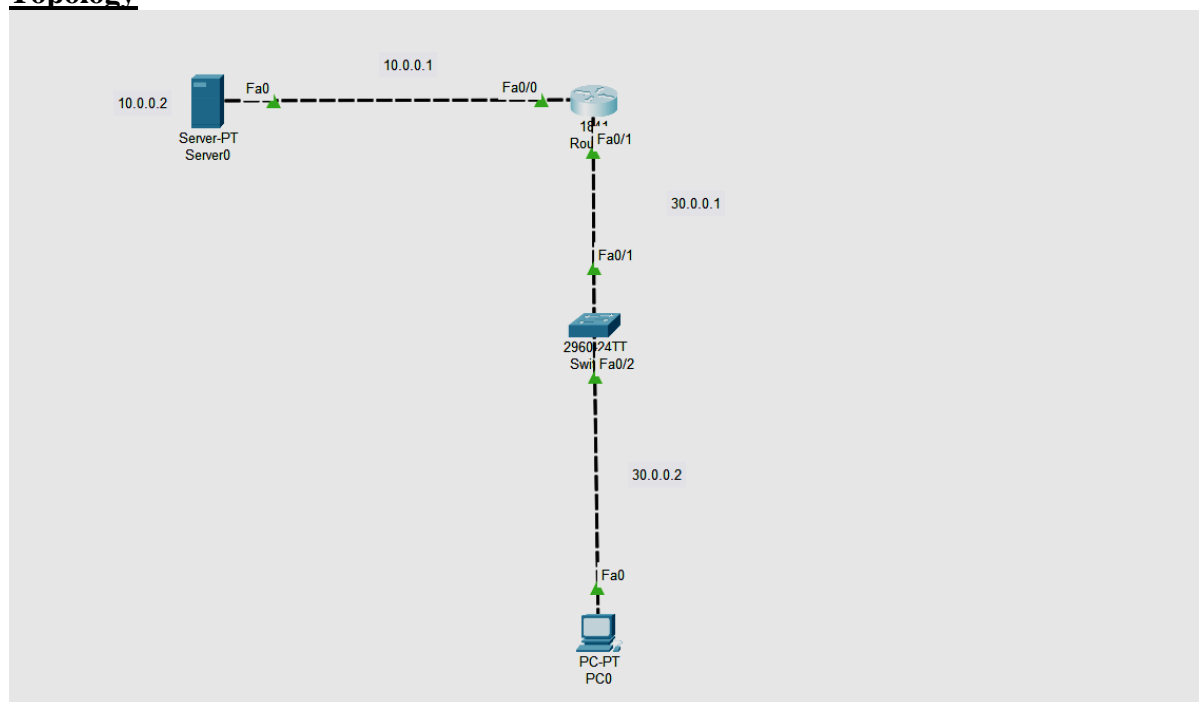
iso.org.dod.internet.mgmt.mib-2.system.sysName.0

☐ Top

| Name/OID | Value | Type |
|---|--------|-------------|
| 1.3.6.1.2.1.1.5.0
(iso.org.dod.internet.mgmt.mib-2.system.sysName.0) | my_sys | OctetString |

| | |
|---------------|-------------------|
| Name : | sysName |
| OID : | 1.3.6.1.2.1.1.5.0 |
| Syntax : | |
| Access : | |
| Description : | |

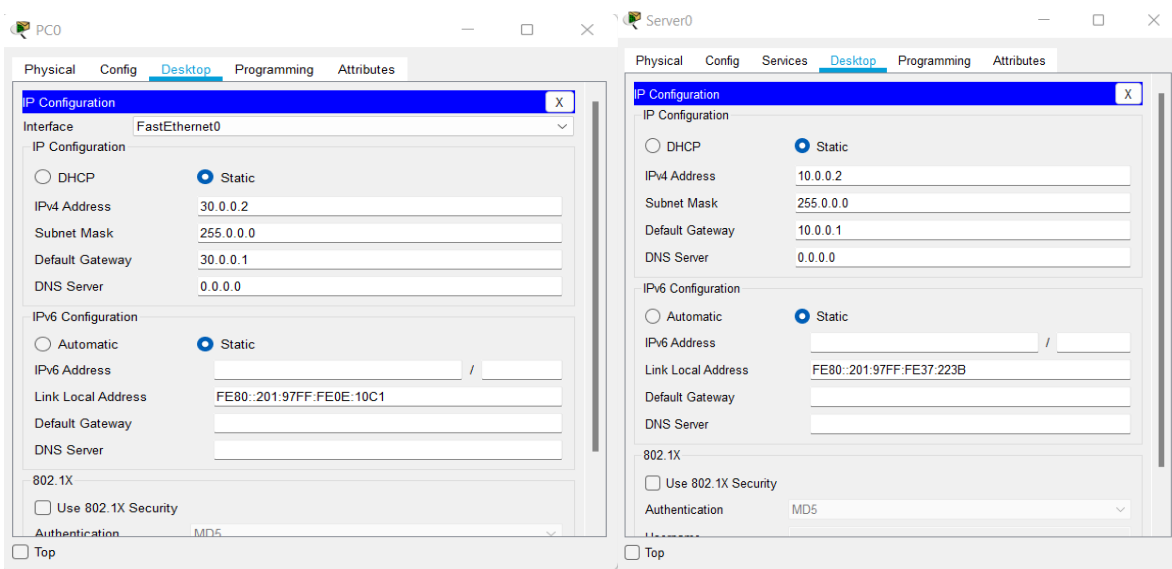
Syslog Topology



Addressing Table

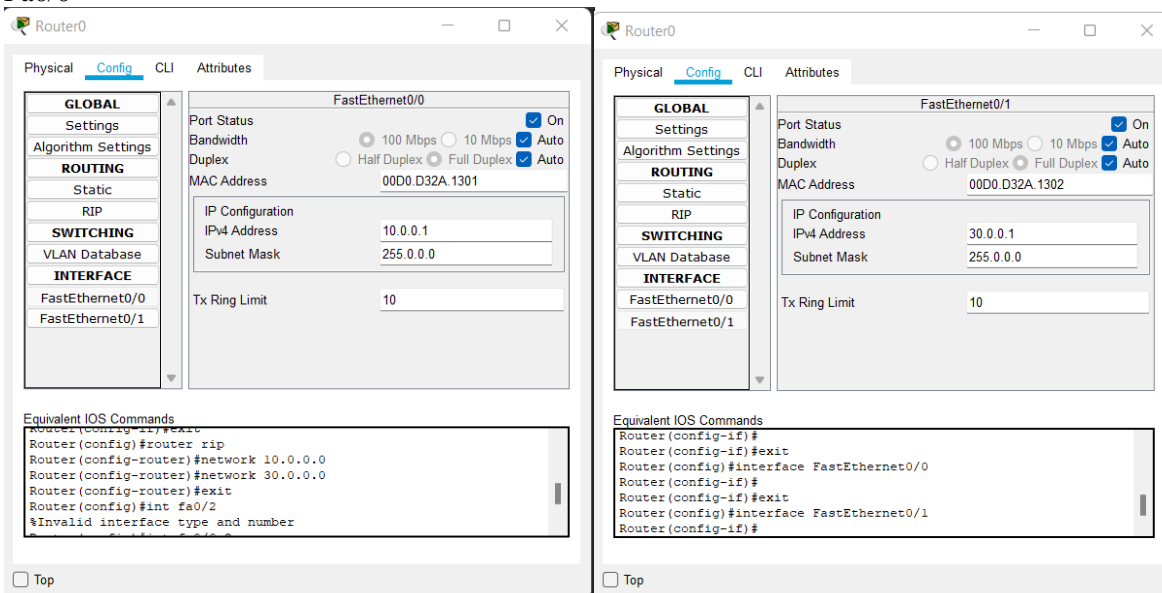
| Device | Interface | IP Address | Subnet Mask | Default Gateway |
|---------|-----------|------------|-------------|-----------------|
| R1 | Fa0/0 | 10.0.0.1 | 255.0.0.0 | N/A |
| | Fa0/1 | 30.0.0.1 | 255.0.0.0 | N/A |
| PC0 | Fa0 | 30.0.0.2 | 255.0.0.0 | 30.0.0.1 |
| Switch0 | Fa0/1 | N/A | 255..0.0.0 | N/A |
| | Fa0/2 | N/A | 255..0.0.0 | N/A |

Configure Pc0

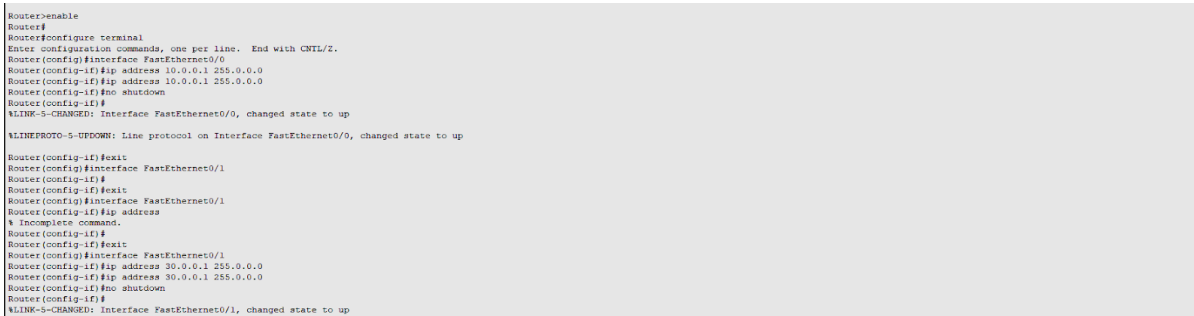


Go to r0 and configure network

Fa0/0



Fa0/1



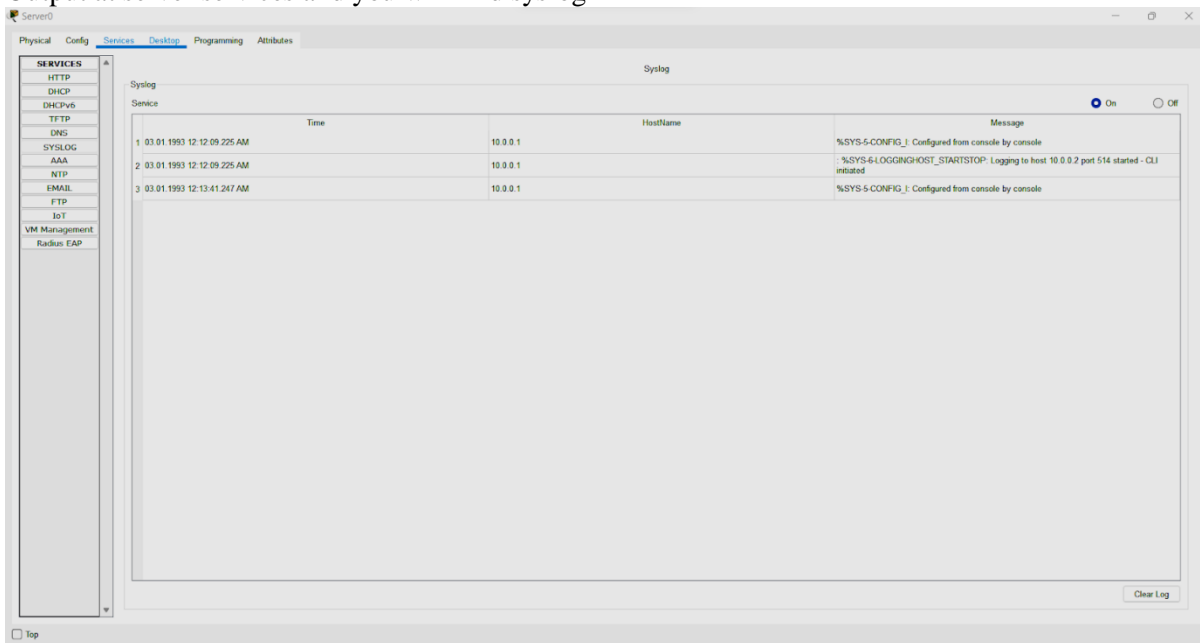
Go to r1 again and at cli put the following command

```

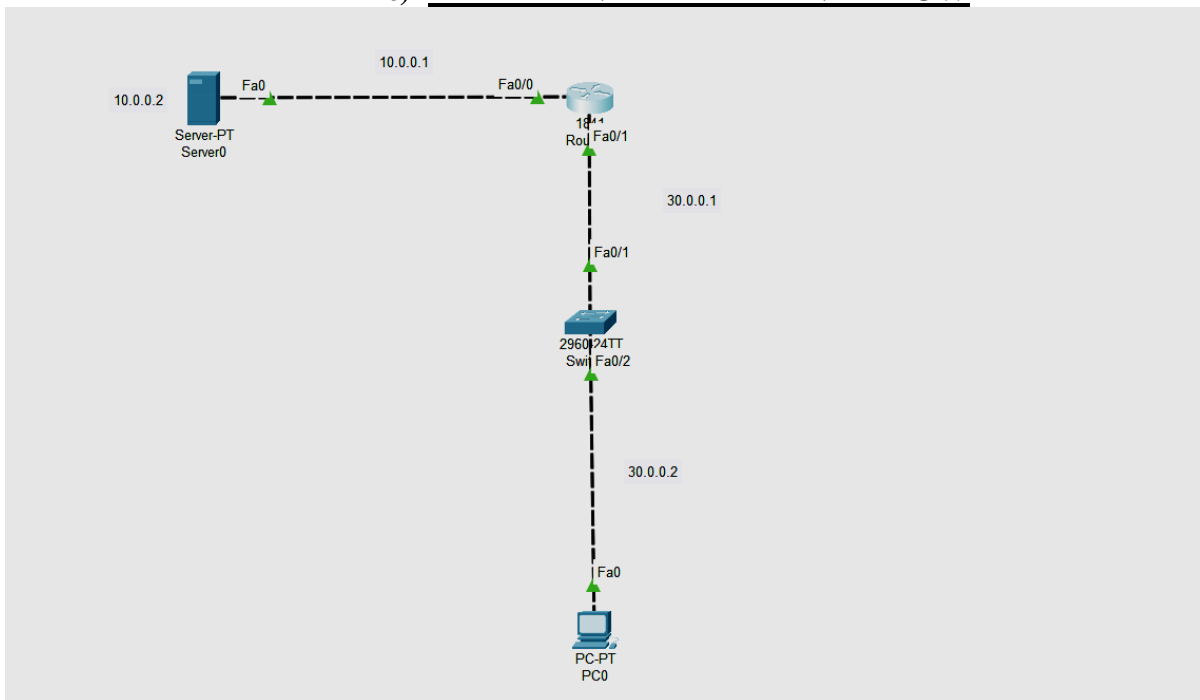
Router(config-if)#
Router(config-if)#exit
Router(config)#service timestamps log datetime msec
Router(config)#int fa0/0.1
Router(config-subif)#
*Mar 01, 00:06:26.066: %LINK-5-CHANGED: Interface FastEthernet0/0.1, changed state to up
*Mar 01, 00:06:26.066: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.1, changed state to up
Router(config-subif)#logging host 10.0.0.2
Router(config)#
Router(config)#interface FastEthernet0/1
Router(config-if)#
Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 30.0.0.0
Router(config-router)#exit
Router(config)#int fa0/2
%Invalid interface type and number
Router(config)#int fa0/0.2
Router(config-subif)#
*Mar 01, 00:11:31.1111: %LINK-5-CHANGED: Interface FastEthernet0/0.2, changed state to up
*Mar 01, 00:11:31.1111: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.2, changed state to up
Router(config-subif)#end
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

```

Output at server services and you will find syslog



c) IMPLEMENT FLEXIBLE NETFLOW



Do the configuration as above dig

Go to r1

R1>enable

R1#config terminal

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

R1(config)#int fa0/0

R1(config-if)#ip flow ingress

R1(config-if)#ip flow egress

R1(config-if)#ip flow-export source fa0/0

R1(config-if)#end

R1#show ip cache flow

```

Router#show ip cache flow
IP packet size distribution (0 total packets):
 1-32  64  96 128 160 192 224 256 288 320 352 384 416 448 480
.000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

 512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
.000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

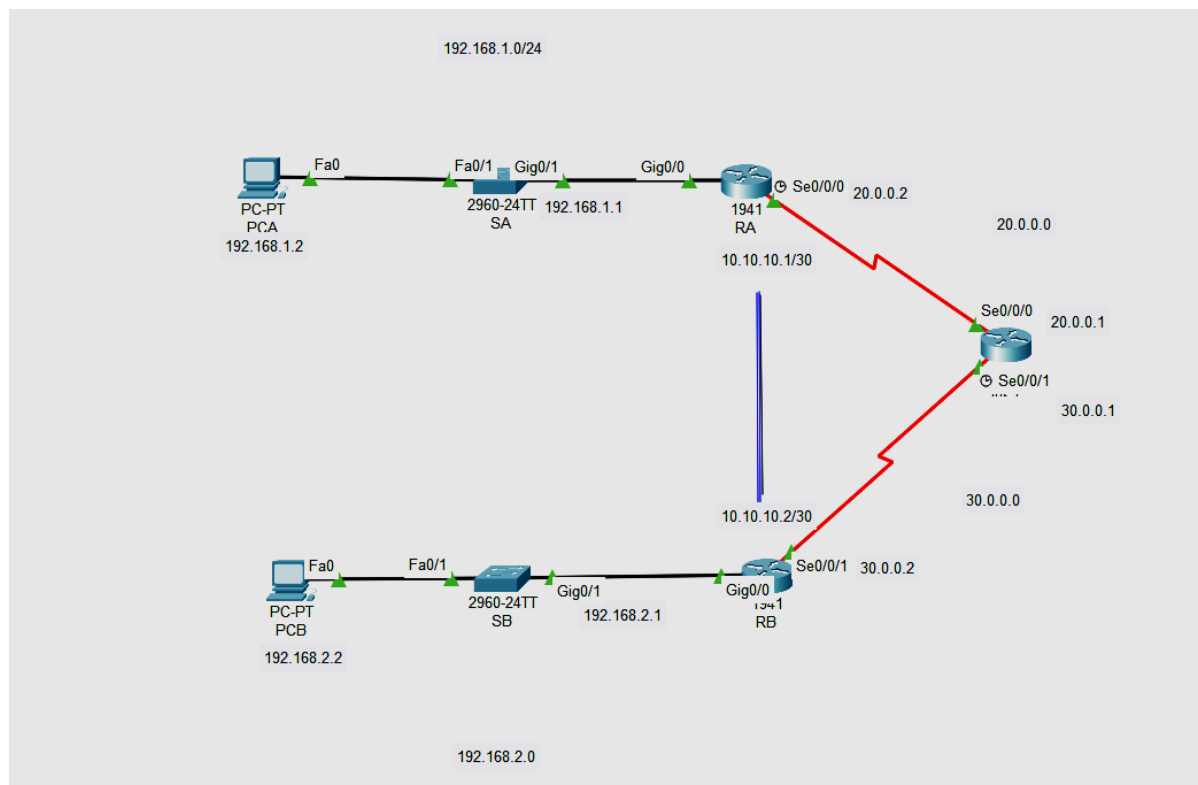
IP Flow Switching Cache, 278544 bytes
0 active, 4096 inactive, 0 added
1 ager polls, 0 flow alloc failures
Active flows timeout in 30 minutes
Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 34056 bytes
0 active, 1024 inactive, 0 added, 0 added to flow
0 alloc failures, 0 force free
1 chunk, 1 chunk added
last clearing of statistics never
Protocol      Total    Flows    Packets  Bytes    Packets  Active(Sec)  Idle(Sec)
-----
Flows         /Sec    /Flow   /Pkt     /Sec     /Flow      /Flow
Total:         0        0.0      0        0        0.0       0.0       0.0

SrcIf      SrcIPaddress  DstIf      DstIPaddress  Pr SrcP DstP  Pkts
Router#
  
```

PRACTICAL 4

Aim:-

1. IMPLEMENT A GRE TUNNEL



Addressing Table

| Device | Interface | IP Address | Subnet Mask | Default Gateway |
|--------|-----------|-------------|-----------------|-----------------|
| RA | G0/0 | 192.168.1.1 | 255.255.255.0 | N/A |
| | S0/0/0 | 20.0.0.2 | 255.255.255.252 | N/A |
| | Tunnel 0 | 10.10.10.1 | 255.255.255.252 | N/A |
| RB | G0/0 | 192.168.2.1 | 255.255.255.0 | N/A |
| | S0/0/0 | 30.0.0.2 | 255.255.255.252 | N/A |
| | Tunnel 0 | 10.10.10.2 | 255.255.255.252 | N/A |
| PC-A | NIC | 192.168.1.2 | 255.255.255.0 | 192.168.1.1 |
| PC-C | NIC | 192.168.2.2 | 255.255.255.0 | 192.168.2.1 |

Part 1: Verify Router Connectivity

Configuring RA

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface GigabitEthernet0/1
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown

Router(config)#interface Serial0/0/0
Router(config-if)#ip address 20.0.0.2 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown
```

Configuring RB

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface GigabitEthernet0/1
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#no shutdown
```

```
Router>enable
Router#configure terminal
Router(config)#interface Serial0/0/1
Router(config-if)#ip address 30.0.0.2 255.255.255.252
Router(config-if)#clock rate 64000
This command applies only to DCE interfaces
Router(config-if)#no shutdown
```

Configuring RC

```
Router>enable
Router#
Router#configure terminal
Router(config)#interface Serial0/0/0
Router(config-if)#ip address 20.0.0.1 255.255.255.252
Router(config-if)#clock rate 64000
This command applies only to DCE interfaces
Router(config-if)#no shutdown
```

```
Router>enable
Router#configure terminal
```

```
Router(config)#interface Serial0/0/0
Router(config-if)#ip address 20.0.0.1 255.255.255.252
Router(config-if)#clock rate 64000
This command applies only to DCE interfaces
Router(config-if)#no shutdown
```

NOW RIP ALL THE ROUTER

RA

```
Router(config)#router rip
Router(config-router)# network 20.0.0.0
```

RB

```
Router(config)#router rip
Router(config-router)#network 192.168.2.0
Router(config-router)#network 30.0.0.0
```

RC

```
Router(config)#router rip
Router(config-router)#network 20.0.0.0
Router(config-router)#network 30.0.0.0
```

Part 2: Configure GRE Tunnels

Step 1: Configure the Tunnel 0 interface of RA.

```
Router>EN
Router#conf t
Router(config)#int tunnel 0
Router(config-if)#ip address 10.10.10.1 255.255.255.252
Router(config-if)#tunnel source s0/0/0
Router(config-if)#tunnel destination 30.0.0.2
Router(config-if)#tunnel mode gre ip
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#ip route 192.168.2.0 255.255.255.0 10.10.10.2
```

Step 2: Configure the Tunnel 0 interface of RB.

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int tunnel 0
Router(config-if)#ip address 10.10.10.2 255.255.255.252
Router(config-if)#tunnel source s0/0/1
Router(config-if)#tunnel destination 20.0.0.2
Router(config-if)#tunnel mode gre ip
Router(config-if)#no shut
```

```
Router(config-if)#ip route 192.168.1.0 255.255.255.0 10.10.10.1
Router(config)#
```

Step 3: Configure a route for private IP traffic.

```
Router(config)#ip route 192.168.2.0 255.255.255.0 10.10.10.2
Router(config-if)#ip route 192.168.1.0 255.255.255.0 10.10.10.1
```

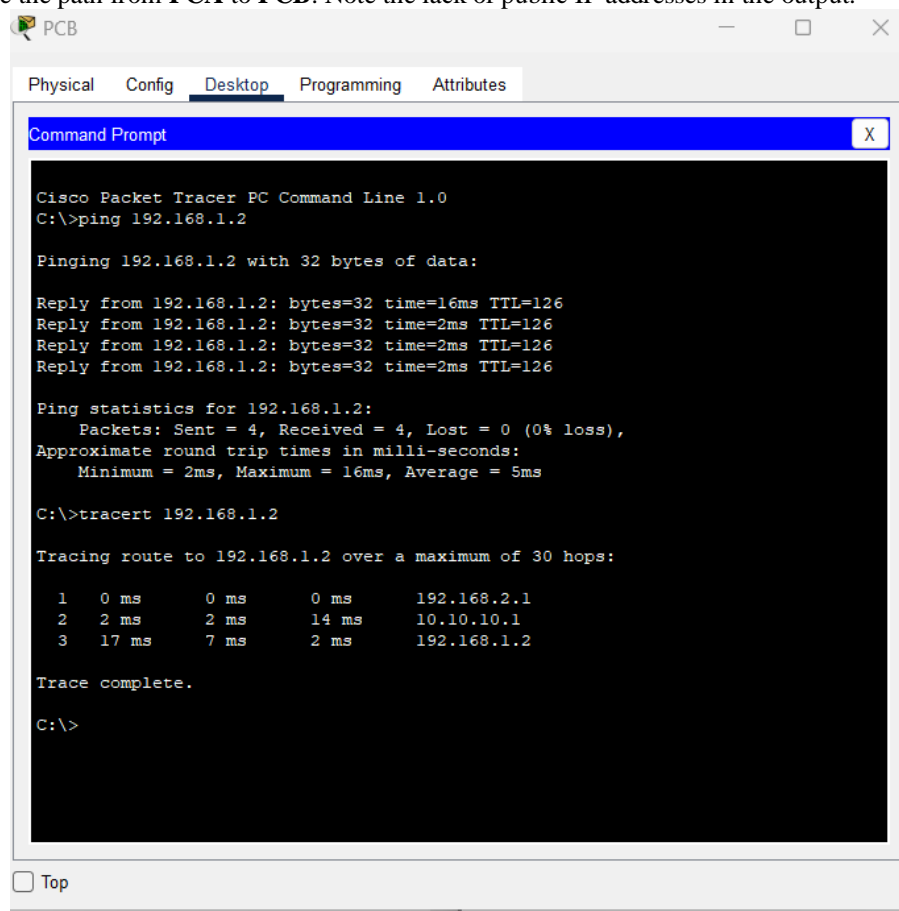
Part 3: Verify Router Connectivity

Step 1: Ping PCA from PCB.

Attempt to ping the IP address of **PCA** from **PCB**. The ping should be successful.

Step 2: Trace the path from PCA to PCB.

Attempt to trace the path from **PCA** to **PCB**. Note the lack of public IP addresses in the output.



The screenshot shows a Cisco Packet Tracer PC Command Line window for a device named PCB. The window has tabs for Physical, Config, Desktop, Programming, and Attributes. The Desktop tab is active, showing a Command Prompt window. The Command Prompt displays the output of a ping and a traceroute command.

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=16ms TTL=126
Reply from 192.168.1.2: bytes=32 time=2ms TTL=126
Reply from 192.168.1.2: bytes=32 time=2ms TTL=126
Reply from 192.168.1.2: bytes=32 time=2ms TTL=126

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 16ms, Average = 5ms

C:\>tracert 192.168.1.2

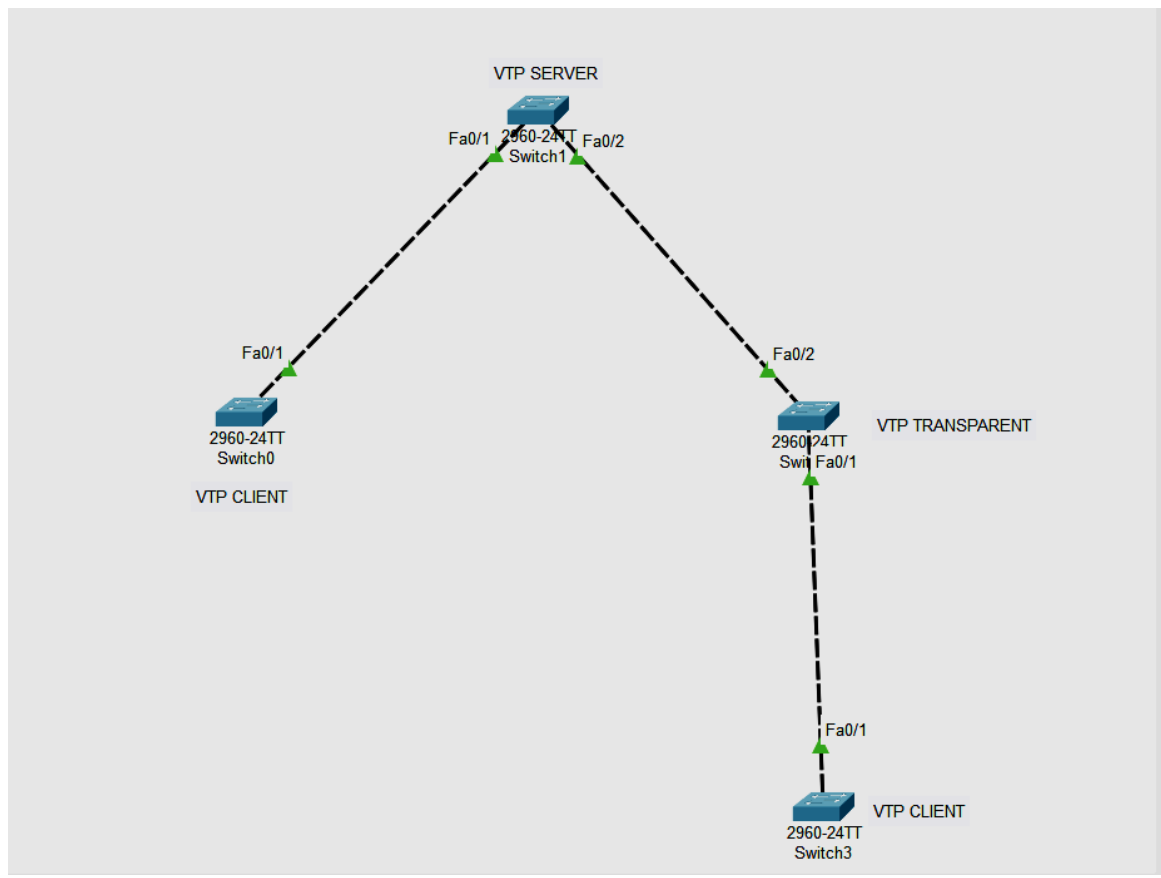
Tracing route to 192.168.1.2 over a maximum of 30 hops:

  0  0 ms    0 ms    0 ms    192.168.2.1
  1  2 ms    2 ms   14 ms   10.10.10.1
  2 17 ms    7 ms    2 ms   192.168.1.2

Trace complete.

C:\>
```


2. IMPLEMENT VTP



Part 1 : Configure the Switch and trunking mode

Switch 0

Switch>en

Switch#conf t

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

Switch(config)#int fa0/1

Switch(config-if)#switchport mode trunk

Switch(config-if)#exit

Switch 1

Switch>EN

Switch#conf t

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

Switch(config)#int fa0/1

Switch(config-if)#switchport mode trunk

Switch(config-if)#int fa0/2

Switch(config-if)#switchport mode trunk

Switch(config-if)#int fa0/2

Switch(config-if)#switchport mode trunk

Switch 2

Switch>en

Switch#conf t

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

Switch(config)#int fa0/1

Switch(config-if)#switchport mode trunk

Switch(config-if)#int fa0/2

Switch(config-if)#switchport mode trunk

Switch(config-if)#exit

Switch 3

Switch>en

Switch#conf t

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

Switch(config)#int fa0/1

Switch(config-if)#switchport mode trunk

Switch(config-if)#exit

Part 2: Configuring mode of switches**Switch 0 to client**

Switch(config)#vtp mode client

Setting device to VTP CLIENT mode.

Switch(config)#

Switch 2 to transparent

Switch(config)#vtp mode client

Setting device to VTP CLIENT mode.

Switch(config)#

Changing Switch 3 to client

Switch(config)#vtp mode client

Setting device to VTP CLIENT mode.

Switch(config)#

No need to change the mode of switch 1 since by default mode is server

Output to check status of each switch

Switch0

Physical Config **CLI** Attributes

IOS Command Line Interface

```
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#show vtp status
VTP Version                : 2
Configuration Revision      : 4
Maximum VLANs supported locally : 255
Number of existing VLANs    : 7
VTP Operating Mode          : Client
VTP Domain Name             : VTPServer
VTP Pruning Mode            : Disabled
VTP V2 Mode                 : Disabled
VTP Traps Generation        : Disabled
MD5 digest                  : 0x4E 0x70 0x55 0xD2 0x2F 0x8C
0x13 0x11
Configuration last modified by 0.0.0.0 at 3-1-93 00:26:01
Switch#show vlan

VLAN Name                Status      Ports
-----
1  default                active      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
2  production              active
3  object                  active
1002 fddi-default           active
1003 token-ring-default    active
1004 fddinet-default        active
1005 trnet-default          active
```

Ctrl+F6 to exit CLI focus

Copy Paste

☐ Top

Check the vtp status

Switch#Show vtp status

Switch0

Physical Config **CLI** Attributes

```
%SYS-5-CONFIG_I: Configured from console by console

Switch#show vtp status
VTP Version                : 2
Configuration Revision      : 4
Maximum VLANs supported locally : 255
Number of existing VLANs    : 7
VTP Operating Mode          : Client
VTP Domain Name             : VTPServer
VTP Pruning Mode            : Disabled
VTP V2 Mode                 : Disabled
VTP Traps Generation        : Disabled
MD5 digest                  : 0x4E 0x70 0x55 0xD2 0x2F 0x8C 0x13 0x11
Configuration last modified by 0.0.0.0 at 3-1-93 00:26:01
Switch#show vlan

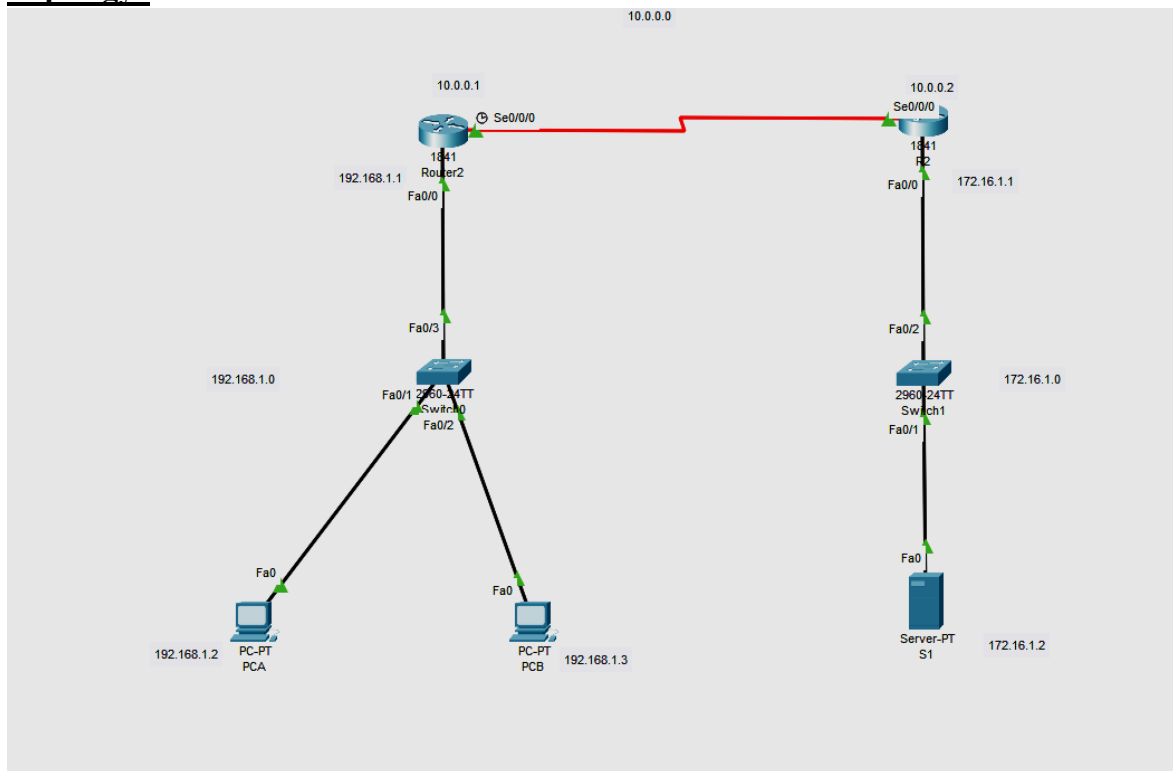
VLAN Name                Status      Ports
-----
1  default                active      Fa0/2, Fa0/3, Fa0/4, Fa0/5
                                           Fa0/6, Fa0/7, Fa0/8, Fa0/9
                                           Fa0/10, Fa0/11, Fa0/12, Fa0/13
                                           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
2  production              active
3  object                  active
1002 fddi-default           active
1003 token-ring-default    active
1004 fddinet-default        active
1005 trnet-default          active

VLAN Type  SAID      MTU    Parent RingNo BridgeNo Stp  BrdgMode Transl Trans2
-----
1  enet  100001   1500    -     -     -     -     -     0      0
2  enet  100002   1500    -     -     -     -     -     0      0
3  enet  100003   1500    -     -     -     -     -     0      0
--More--

Switch con0 is now available
```

3.IMPLEMENT NAT

Topology:



Part 1 : Configure the Router

| Device | Interface | IP Address | Subnet Mask | Default Gateway |
|--------|-----------|-------------|---------------|-----------------|
| R1 | F0/0 | 192.168.1.1 | 255.255.255.0 | N/A |
| | S0/0/0 | 10.0.0.1 | 255.0.0.0 | N/A |
| R2 | F0/0 | 172.16.1.1 | 255.255.0.0 | N/A |
| | S0/0/0 | 10.0.0.2 | 255.0.0.0 | N/A |
| PC-A | NIC | 192.168.1.2 | 255.255.255.0 | 192.168.1.1 |
| PC-C | NIC | 192.168.1.3 | 255.255.255.0 | 192.168.1.1 |
| SERVER | Fa0 | 172.16.1.2 | 255.255.0.0 | 172.16.1.1 |

In router r2

Router>enable

Router#configure terminal

Router(config)#interface Serial0/0/0

Router(config-if)#ip address 10.0.0.2 255.0.0.0

Router(config-if)#clock rate 64000

Router(config-if)#no shutdown

Router(config-if)#exit

```
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 172.16.1.1 255.255.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

```
Router(config)#router rip
Router(config-router)#network 172.16.0.0
Router(config-router)#network 10.0.0.0
```

In router r1

```
Router>enable
Router#configure terminal
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

```
Router(config)#interface Serial0/0/0
Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown
```

```
Router(config)#router rip
Router(config-router)#network 192.168.1.0
Router(config-router)#network 10.0.0.0
Router(config-router)#exit
```

```
Router(config)#int fa0/0
Router(config-if)#ip nat inside
```

```
Router(config-if)#int s0/0/0
Router(config-if)#ip nat outside
Router(config-if)#exit
```

```
Router(config)#ip nat inside source static 192.168.1.2 10.0.0.1
Router(config)#ip route 0.0.0.0 0.0.0.0 s0/0/0
```

```
Router(config)#ip nat inside source static 192.168.1.3 10.0.0.1
Router(config)#ip route 0.0.0.0 0.0.0.0 s0/0/0
Router(config)#exit
```

To check nat status

```
Router#show ip nat ?
statistics Translation statistics
```

translations Translation entries

Router#show ip nat statistics

Total translations: 1 (2 static, 4294967295 dynamic, 0 extended)

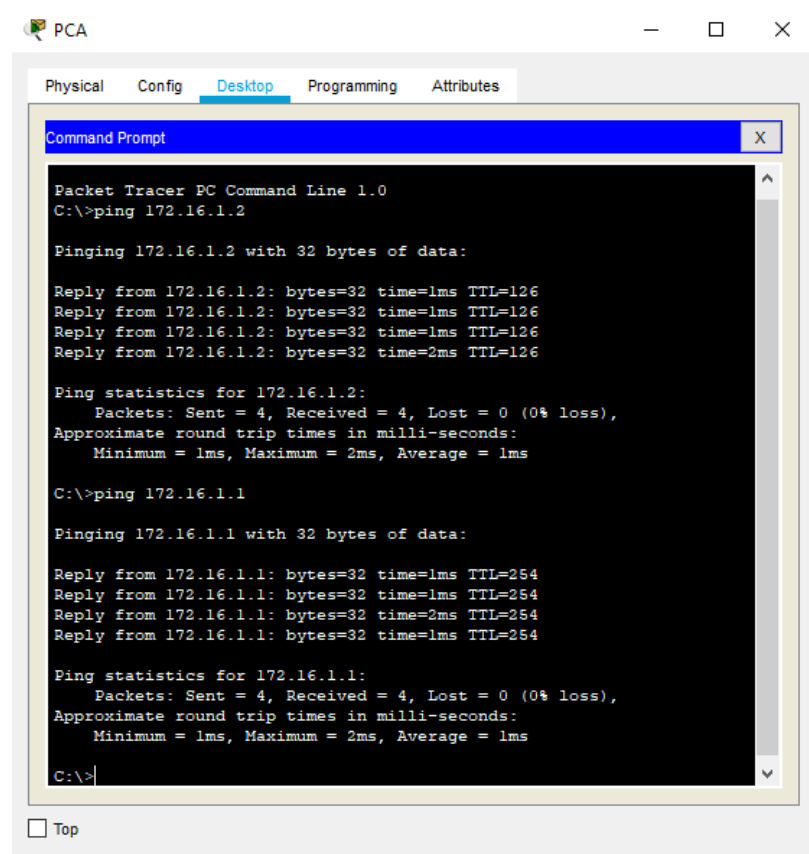
Outside Interfaces: Serial0/0/0

Inside Interfaces: FastEthernet0/0

Hits: 0 Misses: 16

Expired translations: 0

Dynamic mappings:



After pinging to server

Router#show ip nat statistics

Total translations: 5 (2 static, 3 dynamic, 4 extended)

Outside Interfaces: Serial0/0/0

Inside Interfaces: FastEthernet0/0

Hits: 4 Misses: 23

Expired translations: 0

Dynamic mappings:

After pinging to router 2 just after sending packet to server

Router#show ip nat statistics

Total translations: 9 (2 static, 7 dynamic, 8 extended)

Outside Interfaces: Serial0/0/0

Inside Interfaces: FastEthernet0/0

Hits: 8 Misses: 30

Expired translations: 0

Dynamic mappings:

Router#

The screenshot shows a Cisco IOS Command Line Interface window titled 'R1'. The 'CLI' tab is selected. The window displays the following commands and output:

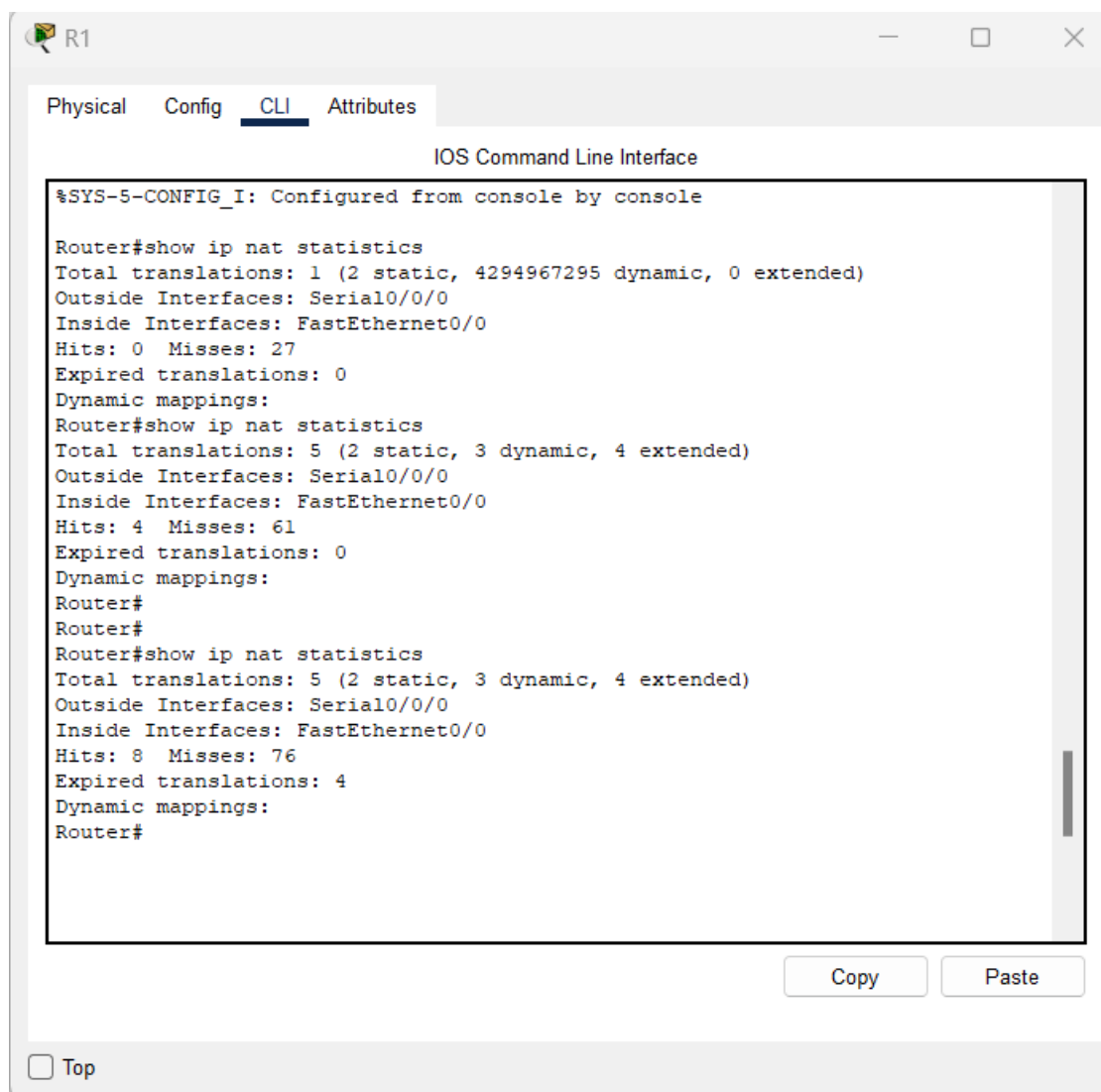
```

Router(config)#ip nat inside source static 192.168.1.2 10.0.0.1
Router(config)#ip route 0.0.0.0 0.0.0.0 se0/0/0
Router(config)#ip nat inside source static 192.168.1.3 10.0.0.1
Router(config)#ip route 0.0.0.0 0.0.0.0 se0/0/0
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show ip nat statistics
Total translations: 1 (2 static, 4294967295 dynamic, 0 extended)
Outside Interfaces: Serial0/0/0
Inside Interfaces: FastEthernet0/0
Hits: 0 Misses: 27
Expired translations: 0
Dynamic mappings:
Router#show ip nat statistics
Total translations: 5 (2 static, 3 dynamic, 4 extended)
Outside Interfaces: Serial0/0/0
Inside Interfaces: FastEthernet0/0
Hits: 4 Misses: 61
Expired translations: 0
Dynamic mappings:
Router#
Router#
Router#show ip nat statistics
Total translations: 5 (2 static, 3 dynamic, 4 extended)
Outside Interfaces: Serial0/0/0
Inside Interfaces: FastEthernet0/0
Hits: 8 Misses: 76
Expired translations: 0
Dynamic mappings:

```

At the bottom of the window, there are 'Copy' and 'Paste' buttons, and a 'Top' button with a checkbox.



Practical 5

Aim:- Implement Inter-VLAN Routing

Part 1: Build the Network and Configure Basic Device Settings

In Part 1, you will set up the network topology and configure basic settings on the PC hosts and switches.

Step 1: Cable the network as shown in the topology and Addressing table.

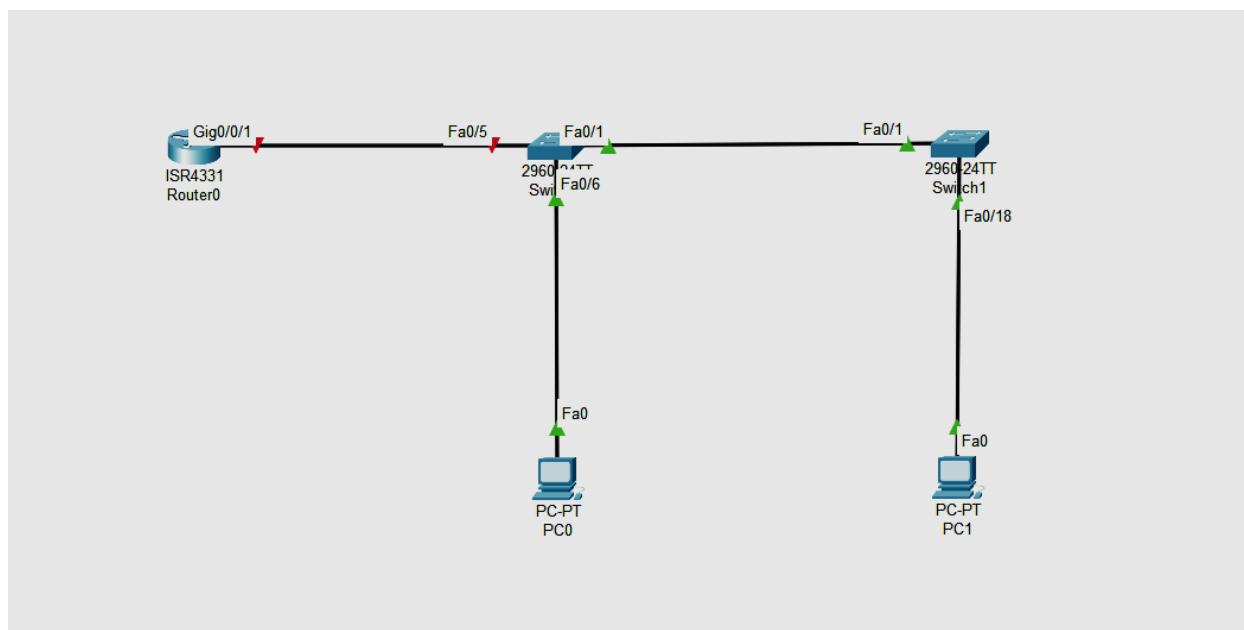
Attach the devices as shown in the topology diagram, and cable as necessary.

Addressing Table

| Device | Interface | IP Address | Subnet Mask | Default Gateway |
|--------|-------------|---------------|---------------|-----------------|
| R1 | G0/0/1.10 | 192.168.10.1 | 255.255.255.0 | N/A |
| R1 | G0/0/1.20 | 192.168.20.1 | 255.255.255.0 | N/A |
| R1 | G0/0/1.30 | 192.168.30.1 | 255.255.255.0 | N/A |
| R1 | G0/0/1.1000 | N/A | N/A | N/A |
| S1 | VLAN 10 | 192.168.10.11 | 255.255.255.0 | 192.168.10.1 |
| S2 | VLAN 10 | 192.168.10.12 | 255.255.255.0 | 192.168.10.1 |
| PC-A | NIC | 192.168.20.3 | 255.255.255.0 | 192.168.20.1 |
| PC-B | NIC | 192.168.30.3 | 255.255.255.0 | 192.168.30.1 |

VLAN Table

| VLAN | Name | Interface Assigned |
|------|-------------|--|
| 10 | Management | S1: VLAN 10
S2: VLAN 10 |
| 20 | Sales | S1: F0/6 |
| 30 | Operations | S2: F0/18 |
| 999 | Parking_Lot | S1: F0/2-4, F0/7-24, G0/1-2
S2: F0/2-17, F0/19-24, G0/1-2 |
| 1000 | Native | N/A |



Step 2: Configure basic settings for the router.

```
router> enable
router# config terminal
router(config)# hostname R1
```

Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names.

```
R1(config)# no ip domain lookup
```

Assign class as the privileged EXEC encrypted password.

```
R1(config)# enable secret class
```

Assign cisco as the console password and enable login.

```
R1(config)# line console 0
R1(config-line)# password cisco
R1(config-line)# login
```

Assign cisco as the vty password and enable login.

```
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
```

Encrypt the plaintext passwords.

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

Create a banner that warns anyone accessing the device that unauthorized access is prohibited.

```
R1(config)# banner motd $ Authorized Users Only! $
```

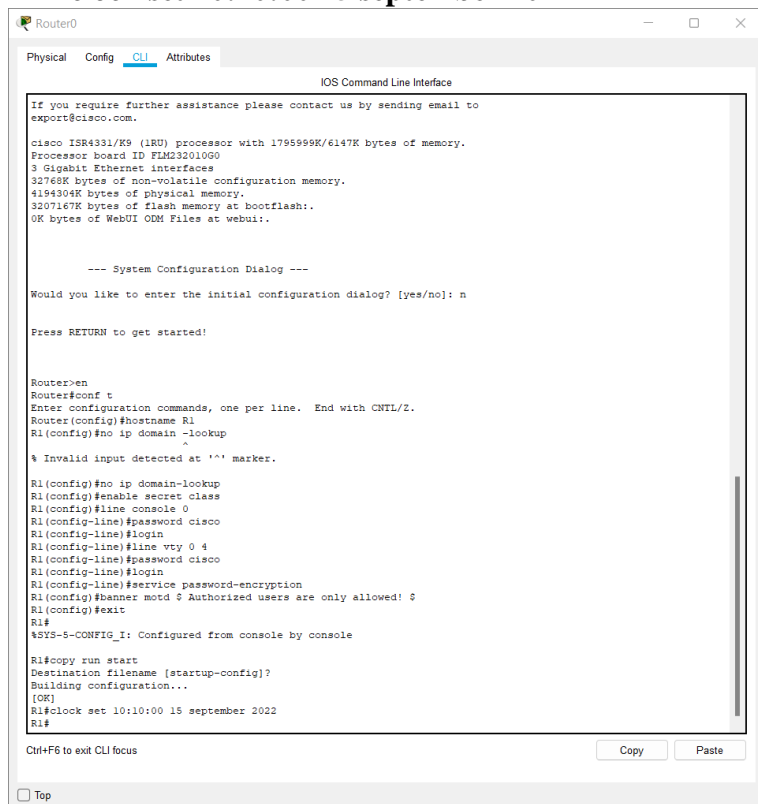
Save the running configuration to the startup configuration file.

R1(config)# exit

R1# copy running-config startup-config

Set the clock on the router.

R1# clock set 10:10:00 15 september 2022



Step 3: Configure basic settings for each switch.

1. Assign a device name to the switch.
 switch(config)# **hostname S1**

 switch(config)# **hostname S2**
2. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names.
 S1(config)# **no ip domain-lookup**

 S2(config)# **no ip domain-lookup**
3. Assign **class** as the privileged EXEC encrypted password.
 S1(config)# **enable secret class**

 S2(config)# **enable secret class**
4. Assign **cisco** as the console password and enable login.
 S1(config)# **line console 0**
 S1(config-line)# **password cisco**
 S1(config-line)# **login**

```
S2(config)# line console 0
S2(config-line)# password cisco
S2(config-line)# login
```

5. Assign **cisco** as the vty password and enable login.

```
S1(config)# line vty 0 4
S1(config-line)# password cisco
S1(config-line)# login
```

```
S2(config)# line vty 0 4
S2(config-line)# password cisco
S2(config-line)# login
```

6. Encrypt the plaintext passwords.

```
S1(config)# service password-encryption
S2(config)# service password-encryption
```

7. Create a banner that warns anyone accessing the device that unauthorized access is prohibited.

```
S1(config)# banner motd $ Authorized Users Only! $
S2(config)# exit
```

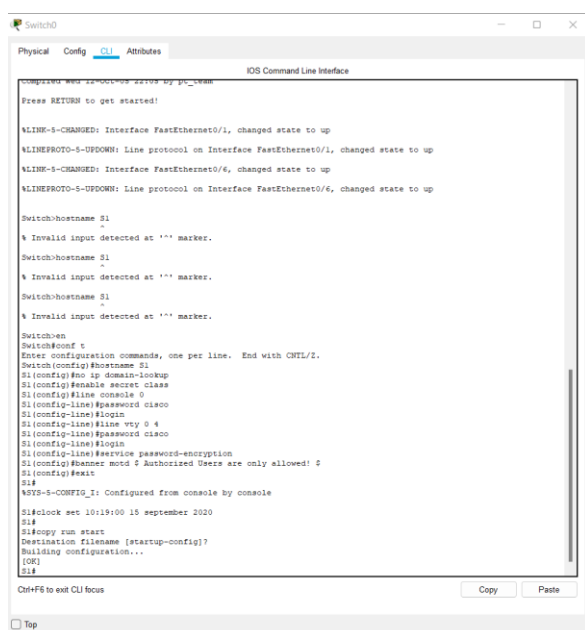
```
S2(config)# banner motd $ Authorized Users Only! $
S2(config)# exit
```

8. Set the clock on the switch.

```
S1# clock set 15:30:00 27 Aug 2019
S2# clock set 15:30:00 27 Aug 2019
```

9. Save the running configuration to the startup configuration.

```
S1# copy running-config startup-config
S2# copy running-config startup-config
```



```

Switch0
Physical Config CLI Attributes
IOS Command Line Interface

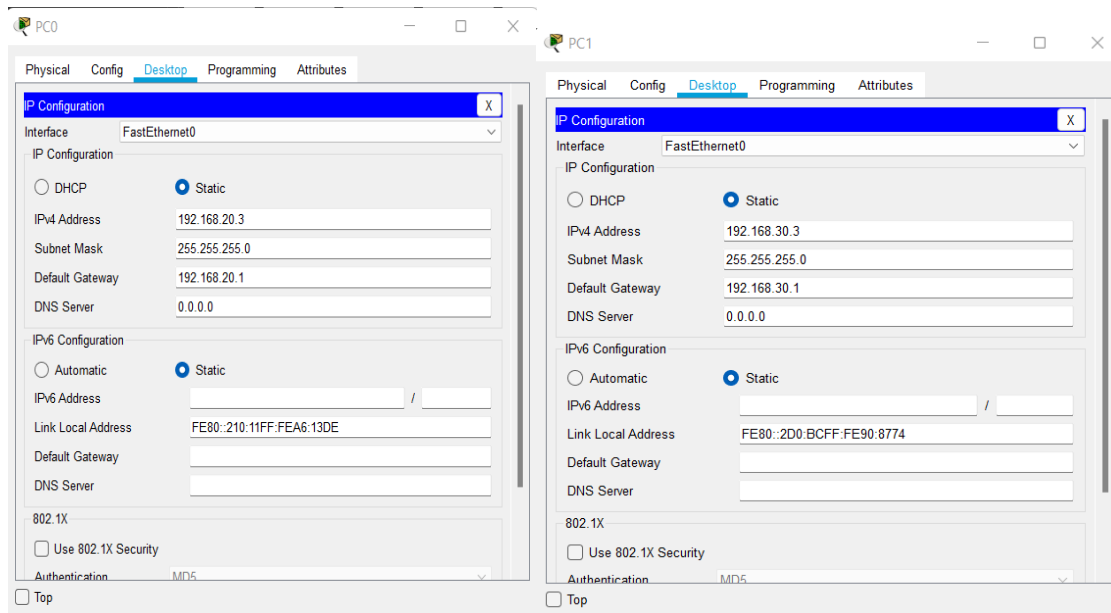
Compiled Wed 12-Nov-03 22:00 by pc_csm
Press RETURN to get started!

%LINE-5-CHANGED: Interface FastEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
%LINE-5-CHANGED: Interface FastEthernet0/6, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/6, changed state to up

Switch#hostname S1
^
% Invalid input detected at '^' marker.
Switch#hostname S1
^
% Invalid input detected at '^' marker.
Switch#hostname S1
^
% Invalid input detected at '^' marker.
Switch#en
Enter configuration commands, one per line. End with CTRL/Z.
Switch(config)#hostname S1
S1(config)#en ip domain-lookup
S1(config)#enable secret class
S1(config)#line console 0
S1(config-line)#password cisco
S1(config-line)#login
S1(config-line)#line vty 0 4
S1(config-line)#password cisco
S1(config-line)#login
S1(config-line)#service password-encryption
S1(config)#banner motd $ Authorized Users Only! $
S1(config)#exit
S1#
SYS-5-CONFIG_I: Configured from console by console
S1#clock set 10:19:00 15 september 2020
S1#
S1#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
S1#
Ctrl+F8 to exit CLI focus
Copy Paste

```

Refer to the Addressing Table for PC host address information.



In Part 2, you will create VLANs as specified in the table above on both switches. You will then assign the VLANs to the appropriate interface and verify your configuration settings. Complete the following tasks on each switch.

1. Create and name the required VLANs on each switch from the table above.

```
S1(config)# vlan 10
```

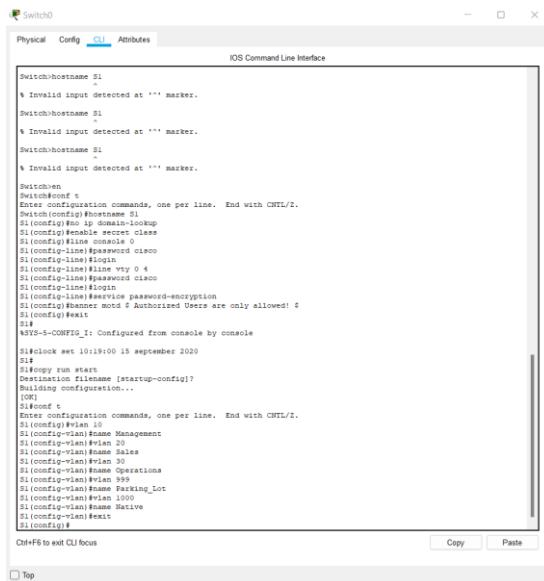
```
S1(config-vlan)# name Management
```

```
S1(config-vlan)# vlan 20
```

```

S1(config-vlan)# name Sales
S1(config-vlan)# vlan 30
S1(config-vlan)# name Operations
S1(config-vlan)# vlan 999
S1(config-vlan)# name Parking_Lot
S1(config-vlan)# vlan 1000
S1(config-vlan)# name Native
S1(config-vlan)# exit

```

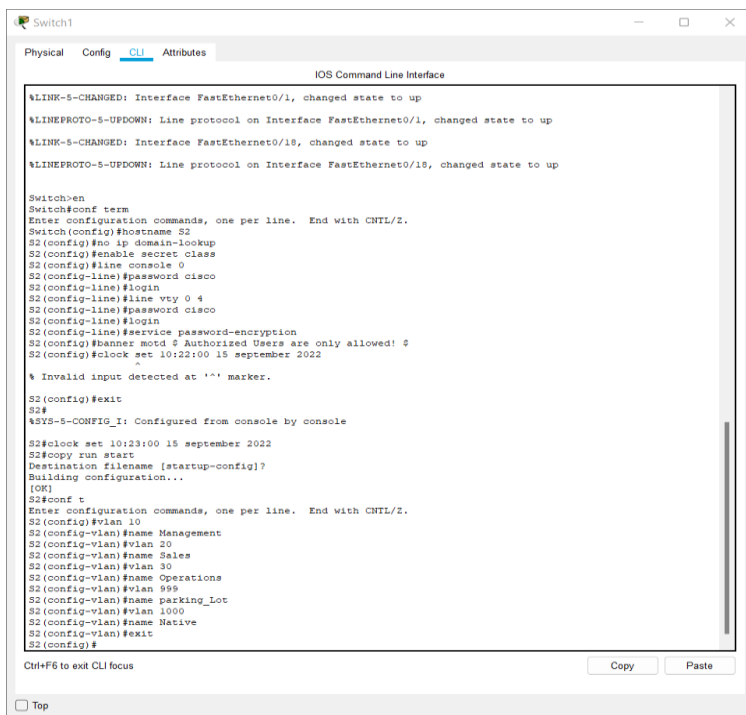


Same in switch s2

```

S2(config)# vlan 10
S2(config-vlan)# name Management
S2(config-vlan)# vlan 20
S2(config-vlan)# name Sales
S2(config-vlan)# vlan 30
S2(config-vlan)# name Operations
S2(config-vlan)# vlan 999
S2(config-vlan)# name Parking_Lot
S2(config-vlan)# vlan 1000
S2(config-vlan)# name Native
S2(config-vlan)# exit

```



2. Configure the management interface and default gateway on each switch using the IP address information in the Addressing Table.

```
S1(config)# interface vlan 10
S1(config-if)# ip address 192.168.10.11 255.255.255.0
S1(config-if)# no shutdown
S1(config-if)# exit
S1(config)# ip default-gateway 192.168.10.1
```

```
S2(config)# interface vlan 10
S2(config-if)# ip address 192.168.10.12 255.255.255.0
S2(config-if)# no shutdown
S2(config-if)# exit
S2(config)# ip default-gateway 192.168.10.1
```

Switch1

Physical Config CLI Attributes

IOS Command Line Interface

```
S2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#hostname S2
S2(config)#no ip domain-lookup
S2(config)#enable secret class
S2(config)#line console 0
S2(config-line)#password cisco
S2(config-line)#login
S2(config-line)#line vty 0 4
S2(config-line)#password cisco
S2(config-line)#login
S2(config-line)#service password-encryption
S2(config)#banner motd & Authorized Users are only allowed! &
S2(config)#clock set 10:22:00 15 september 2022
% Invalid input detected at '^' marker.

S2(config)#exit
S2#
\SYS-5-CONFIG_I: Configured from console by console

S2#clock set 10:23:00 15 september 2022
S2#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
S2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)#vlan 10
S2(config-vlan)#name Management
S2(config-vlan)#vlan 20
S2(config-vlan)#name Sales
S2(config-vlan)#vlan 30
S2(config-vlan)#name Operations
S2(config-vlan)#vlan 999
S2(config-vlan)#name parking_lot
S2(config-vlan)#vlan 1000
S2(config-vlan)#name Native
S2(config-vlan)#exit
S2(config)#
S2(config)#
S2(config)#
S2(config)#int vlan 10
S2(config-if)#
%LINK-5-CHANGED: Interface Vlan10, changed state to up

S2(config-if)#ip address 192.168.10.12 255.255.255.0
S2(config-if)#no shut
S2(config-if)#exit
S2(config)#ip default-gateway 192.168.10.1
S2(config)#
```

Ctrl+F6 to exit CLI focus

Copy Paste

☐ Top

3. Assign all unused ports on the switch to the Parking_Lot VLAN, configure them for static access mode, and administratively deactivate them.

Note: The interface range command is helpful to accomplish this task with as few commands as necessary.

S1(config)# interface range f0/2 – 4 , f0/7 – 24 , g0/1 – 2

S1(config-if-range)# **switchport mode access**

S1(config-if-range)# switchport access vlan 999

```
S1(config-if-range)# shutdown
```

The screenshot shows the Cisco Packet Tracer interface with the CLI window open. The user has entered the following commands:

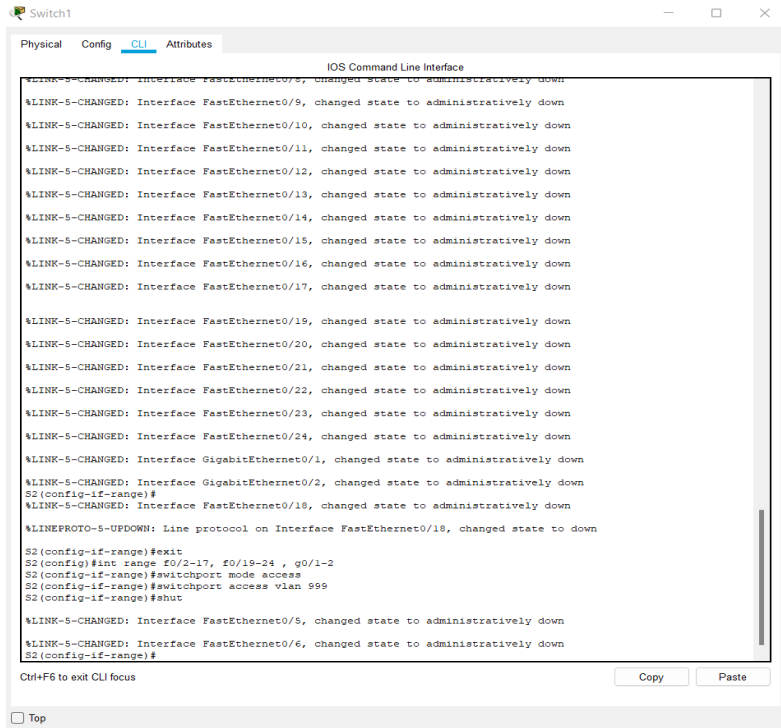
```
SI(config)#int range f0/2-4, f0/7-24, g0/1-2
SI(config-if-range)#switchport mode access
SI(config-if-range)#switchport access vlan 999
SI(config-if-range)#shut
```

The output shows that all interfaces have been successfully configured and are now administratively down:

```
%LINK-5-CHANGED: Interface FastEthernet0/2, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/3, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/4, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/7, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/8, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/9, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/10, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/11, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/12, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/13, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/14, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/15, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/16, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/17, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/18, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/19, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/20, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/21, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/22, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/23, changed state to administratively down
%LINK-5-CHANGED: Interface FastEthernet0/24, changed state to administratively down
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to administratively down
%LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to administratively down
```

The user has also entered the command `Ctrl+F6` to exit CLI focus.

S2(config)# interface range f0/2 – 17 , f0/19 – 24 , g0/1 – 2
S2(config-if-range)# switchport mode access
S2(config-if-range)# switchport access vlan 999
S2(config-if-range)# shutdown



Step 2: Assign VLANs to the correct switch interfaces.

1. Assign used ports to the appropriate VLAN (specified in the VLAN table above) and configure them for static access mode.

S1(config)# **interface f0/6**

S1(config-if)# **switchport mode access**

S1(config-if)# **switchport access vlan 20**

```
S1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
S1(config)#int f0/6
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 20
S1(config-if)#
```

S2(config)# **interface f0/18**

S2(config-if)# **switchport mode access**

S2(config-if)# **switchport access vlan 30**

```
S2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)#int f0/18
S2(config-if)#switchport mode access
S2(config-if)#switchport access vlan 30
S2(config-if)#
```

2. Verify that the VLANs are assigned to the correct interfaces.

S1# **show vlan brief**

| VLAN Name | Status | Ports |
|-------------------------|-----------|--|
| 1 default | active | Fa0/1, Fa0/5 |
| 10 Management | active | |
| 20 Sales | active | Fa0/6 |
| 30 Operations | active | |
| 999 Parking_Lot | active | Fa0/2, 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, Gi0/1, Gi0/2 |
| 1000 Native | active | |
| 1002 fddi-default | act/unsup | |
| 1003 token-ring-default | act/unsup | |
| 1004 fddinet-default | act/unsup | |
| 1005 trnet-default | act/unsup | |

```
S1#show vlan brief
```

| VLAN Name | Status | Ports |
|-------------------------|--------|--|
| 1 default | active | Fa0/1, Fa0/5 |
| 10 Management | active | |
| 20 Sales | active | Fa0/6 |
| 30 Operations | active | |
| 999 Parking_Lot | active | Fa0/2, 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 |
| 1000 Native | active | |
| 1002 fddi-default | active | |
| 1003 token-ring-default | active | |
| 1004 fddinet-default | active | |
| 1005 trnet-default | active | |

```
S1#
```

S2# show vlan brief

| VLAN Name | Status | Ports |
|-------------------------|-----------|--|
| 1 default | active | Fa0/1 |
| 10 Management | active | |
| 20 Sales | active | |
| 30 Operations | active | Fa0/18 |
| 999 Parking_Lot | active | 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
Fa0/19, Fa0/20, Fa0/21, Fa0/22
Fa0/23, Fa0/24, Gi0/1, Gi0/2 |
| 1000 Native | active | |
| 1002 fddi-default | act/unsup | |
| 1003 token-ring-default | act/unsup | |
| 1004 fddinet-default | act/unsup | |
| 1005 trnet-default | act/unsup | |

```
S2#show vlan brief
```

| VLAN Name | Status | Ports |
|-------------------------|--------|--|
| 1 default | active | Fa0/1 |
| 10 Management | active | |
| 20 Sales | active | |
| 30 Operations | active | Fa0/18 |
| 999 parking_Lot | active | 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
Fa0/19, Fa0/20, Fa0/21, Fa0/22
Fa0/23, Fa0/24, Gig0/1, Gig0/2 |
| 1000 Native | active | |
| 1002 fddi-default | active | |
| 1003 token-ring-default | active | |
| 1004 fddinet-default | active | |
| 1005 trnet-default | active | |

```
S2#
```

Part 3: Configure an 802.1Q Trunk Between the Switches

In Part 3, you will manually configure interface F0/1 as a trunk.

Step 1: Manually configure trunk interface F0/1 on switch S1 and S2.

1. Configure static trunking on interface F0/1 for both switches.

Open configuration window

```
S1(config)# interface f0/1
S1(config-if)# switchport mode trunk
```

```
S1(config)#int f0/1
S1(config-if)#switchport mode trunk

S1(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan10, changed state to up

S1(config-if)#
```

```
S2(config)# interface f0/1
S2(config-if)# switchport mode trunk
```

```
S2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)#int f0/1
S2(config-if)#switchport mode trunk
S2(config-if)#
```

2. Set the native VLAN to 1000 on both switches.

```
S1(config-if)# switchport trunk native vlan 1000
```

```
S2(config-if)# switchport trunk native vlan 1000
```

3. Specify that VLANs 10, 20, 30, and 1000 are allowed to cross the trunk.

```
S1(config-if)# switchport trunk allowed vlan 10,20,30,1000
```

```
S2(config-if)# switchport trunk allowed vlan 10,20,30,1000
```

4. Verify trunking ports, the Native VLAN and allowed VLANs across the trunk.

```
S1# show interfaces trunk
```

| Port | Mode | Encapsulation | Status | Native vlan |
|-------|------|---------------|----------|-------------|
| Fa0/1 | on | 802.1q | trunking | 1000 |

| Port | Vlans allowed on trunk |
|-------|------------------------|
| Fa0/1 | 10,20,30,1000 |

| Port | Vlans allowed and active in management domain |
|-------|---|
| Fa0/1 | 10,20,30,1000 |

| Port | Vlans in spanning tree forwarding state and not pruned |
|-------|--|
| Fa0/1 | 10,20,30,1000 |

S2# show interfaces trunk

| Port | Mode | Encapsulation | Status | Native vlan |
|-------|------|---------------|----------|-------------|
| Fa0/1 | on | 802.1q | trunking | 1000 |

| Port | Vlans allowed on trunk |
|-------|------------------------|
| Fa0/1 | 10,20,30,1000 |

| Port | Vlans allowed and active in management domain |
|-------|---|
| Fa0/1 | 10,20,30,1000 |

| Port | Vlans in spanning tree forwarding state and not pruned |
|-------|--|
| Fa0/1 | 10,20,30,1000 |

```
S2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)#int f0/1
S2(config-if)#switchport mode trunk
S2(config-if)#
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on FastEthernet0/1 (1), with S1 FastEthernet0/1 (1000).
%SPANTREE-2-RECV_EVID_ERR: Received BPDU with inconsistent peer vlan id 1000 on FastEthernet0/1 VLAN1.
%SPANTREE-2-BLOCK_PVID_LOCAL: Blocking FastEthernet0/1 on VLAN0001. Inconsistent local vlan.

S2(config-if)#switchport trunk native vlan 1000
S2(config-if)#%SPANTREE-2-UNBLOCK_CONSIST_PORT: Unblocking FastEthernet0/1 on VLAN1000. Port consistency restored.
%SPANTREE-2-UNBLOCK_CONSIST_PORT: Unblocking FastEthernet0/1 on VLAN0001. Port consistency restored.

S2(config-if)#switchport trunk allowed vlan 10,20,30,1000
S2(config-if)#end
S2#
%SYS-5-CONFIG_I: Configured from console by console

S2#show interface trunk
Port      Mode      Encapsulation  Status        Native vlan
Fa0/1     on        802.1q         trunking      1000

Port      Vlans allowed on trunk
Fa0/1     10,20,30,1000

Port      Vlans allowed and active in management domain
Fa0/1     10,20,30,1000

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/1     10,20,30,1000

S2#
```

Step 2: Manually configure S1's trunk interface F0/5

1. Configure S1's interface F0/5 with the same trunk parameters as F0/1. This is the trunk to the router.

```
S1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
S1(config)#int f0/5
S1(config-if)#switchport mode trunk
S1(config-if)#switchport trunk native vlan 1000
S1(config-if)#switchport trunk allowed vlan 10,20,30,1000
S1(config-if)#wnd
^
% Invalid input detected at '^' marker.

S1(config-if)#end
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#show interface trunk
Port      Mode      Encapsulation  Status        Native vlan
Fa0/1     on        802.1q         trunking      1000

Port      Vlans allowed on trunk
Fa0/1     10,20,30,1000

Port      Vlans allowed and active in management domain
Fa0/1     10,20,30,1000

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/1     10,20,30,1000

S1#
```

1. Save the running configuration to the startup configuration file.

S1# copy running-config startup-config

```
S1#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
S1#
```

S2# copy running-config startup-config

```
S2#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
S2#
```

3. Verify trunking.

Question:

What happens if G0/0/1 on R1 is down?

S1 F0/5 will not be displayed if the GigabitEthernet 0/0/1 interface status on the router is down.

Part 4: Configure Inter-VLAN Routing on the Router**Step 1: Configure the router.**

Activate interface G0/0/1 as necessary on the router.

R1(config)# **interface g0/0/1**R1(config-if)# **no shutdown**R1(config-if)# **exit**

```
R1>en
Password:
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int g0/0/1
R1(config-if)#no shut

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1, changed state to up

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

2. Configure sub-interfaces for each VLAN as specified in the IP addressing table. All sub-interfaces use 802.1Q encapsulation. Ensure the sub-interface for the native VLAN does not have an IP address assigned. Include a description for each sub-interface.

R1(config)# **interface g0/0/1.10**R1(config-subif)# **description Management Network**R1(config-subif)# **encapsulation dot1q 10**R1(config-subif)# **ip address 192.168.10.1 255.255.255.0**R1(config-subif)# **interface g0/0/1.20**R1(config-subif)# **encapsulation dot1q 20**R1(config-subif)# **description Sales Network**R1(config-subif)# **ip address 192.168.20.1 255.255.255.0**R1(config-subif)# **interface g0/0/1.30**R1(config-subif)# **encapsulation dot1q 30**

```

R1(config-subif)# description Operations Network
R1(config-subif)# ip address 192.168.30.1 255.255.255.0
R1(config-subif)# interface g0/0/1.1000
R1(config-subif)# encapsulation dot1q 1000 native
R1(config-subif)# description Native VLAN

```

```

R1(config)#int g0/0/1.10
R1(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1.10, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.10, changed state to up

R1(config-subif)#description Management Network
R1(config-subif)#encapsulation dot1q 10
R1(config-subif)#ip address 192.168.10.1 255.255.255.0
R1(config-subif)#int g0/0/1.20
R1(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1.20, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.20, changed state to up

R1(config-subif)#encapsulation dot1q 20
R1(config-subif)#ip address 192.168.20.1 255.255.255.0
R1(config-subif)#description Sales Network
R1(config-subif)#int g0/0/1.30
R1(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1.30, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.30, changed state to up

R1(config-subif)#encapsulation dot1q 30
R1(config-subif)#description Operations Network
R1(config-subif)#ip address 192.168.30.1 255.255.255.0
R1(config-subif)#int g0/0/1.1000
R1(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1.1000, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.1000, changed state to up

R1(config-subif)#encapsulation dot1q 1000 native
R1(config-subif)#description Native VLAN
R1(config-subif)#

```

3. Verify the sub-interfaces are operational

R1# **show ip interface brief**

| Interface | IP-Address | OK? | Method | Status | Protocol |
|----------------------|--------------|-----|--------|--------|----------|
| GigabitEthernet0/0/0 | unassigned | YES | NVRAM | down | down |
| GigabitEthernet0/0/1 | unassigned | YES | NVRAM | up | up |
| Gi0/0/1.10 | 192.168.10.1 | YES | manual | up | up |
| Gi0/0/1.20 | 192.168.20.1 | YES | manual | up | up |
| Gi0/0/1.30 | 192.168.30.1 | YES | manual | up | up |
| Gi0/0/1.1000 | unassigned | YES | unset | up | up |
| GigabitEthernet0 | unassigned | YES | NVRAM | down | down |

```

R1#show ip interface brief
Interface      IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0/0  unassigned      YES unset  administratively down down
GigabitEthernet0/0/1  unassigned      YES unset  up          up
GigabitEthernet0/0/1.10  192.168.10.1    YES manual up          up
GigabitEthernet0/0/1.20  192.168.20.1    YES manual up          up
GigabitEthernet0/0/1.30  192.168.30.1    YES manual up          up
GigabitEthernet0/0/1.1000 unassigned      YES unset  up          up
GigabitEthernet0/0/2  unassigned      YES unset  administratively down down
Vlan1          unassigned      YES unset  administratively down down
R1#

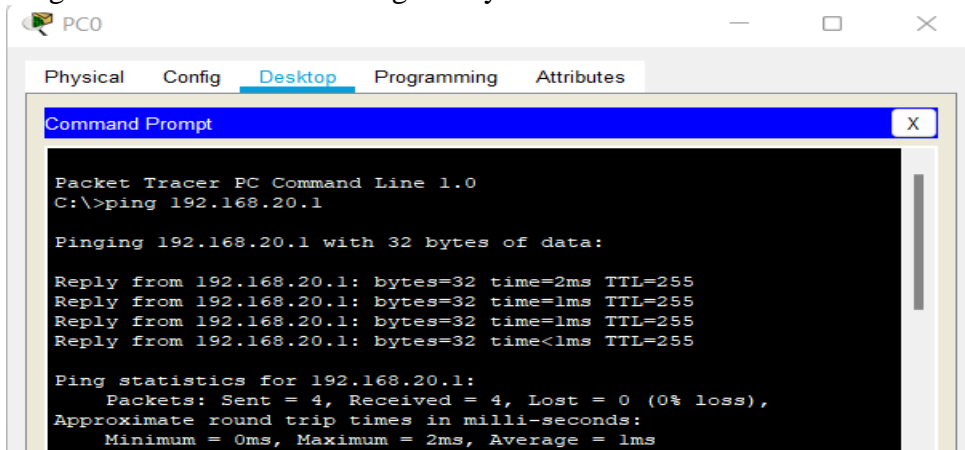
```

Part 5: Verify Inter-VLAN Routing is Working

Step 1: Complete the following tests from PC-A. All should be successful.

Note: You may have to disable the PC firewall for pings to work

1. Ping from PC-A to its default gateway. 192.168.20.1



```

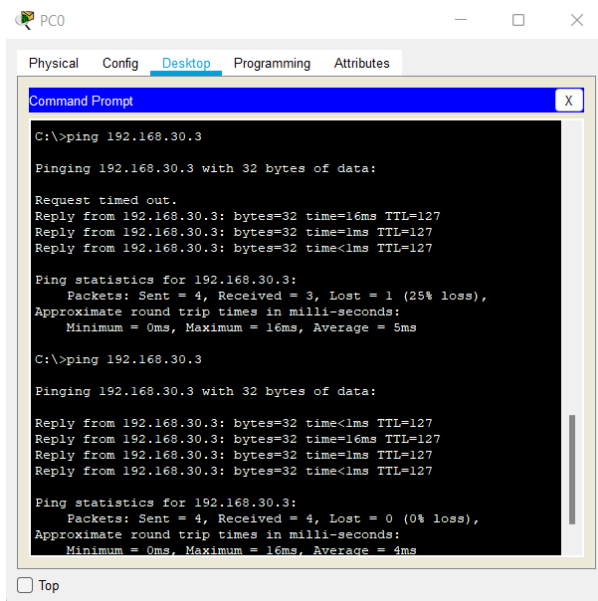
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.20.1

Pinging 192.168.20.1 with 32 bytes of data:

Reply from 192.168.20.1: bytes=32 time=2ms TTL=255
Reply from 192.168.20.1: bytes=32 time=1ms TTL=255
Reply from 192.168.20.1: bytes=32 time=1ms TTL=255
Reply from 192.168.20.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.20.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 1ms
  
```

1. Ping from PC-A to PC-B



```

PC0
Physical Config Desktop Programming Attributes
Command Prompt
C:\>ping 192.168.30.3

Pinging 192.168.30.3 with 32 bytes of data:

Request timed out.
Reply from 192.168.30.3: bytes=32 time=16ms TTL=127
Reply from 192.168.30.3: bytes=32 time=1ms TTL=127
Reply from 192.168.30.3: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.30.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 16ms, Average = 5ms

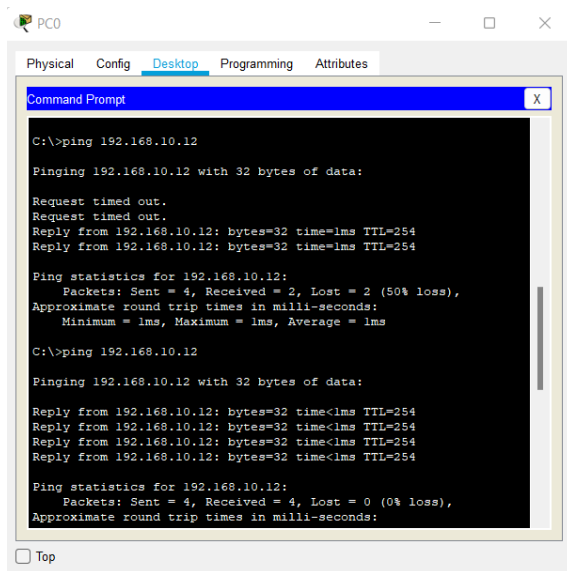
C:\>ping 192.168.30.3

Pinging 192.168.30.3 with 32 bytes of data:

Reply from 192.168.30.3: bytes=32 time<1ms TTL=127
Reply from 192.168.30.3: bytes=32 time=16ms TTL=127
Reply from 192.168.30.3: bytes=32 time=1ms TTL=127
Reply from 192.168.30.3: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.30.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 16ms, Average = 4ms
  
```

1. Ping from PC-A to S2



PC0

Physical Config Desktop Programming Attributes

Command Prompt

```
C:\>ping 192.168.10.12

Pinging 192.168.10.12 with 32 bytes of data:

Request timed out.
Request timed out.
Reply from 192.168.10.12: bytes=32 time=1ms TTL=254
Reply from 192.168.10.12: bytes=32 time=1ms TTL=254

Ping statistics for 192.168.10.12:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms

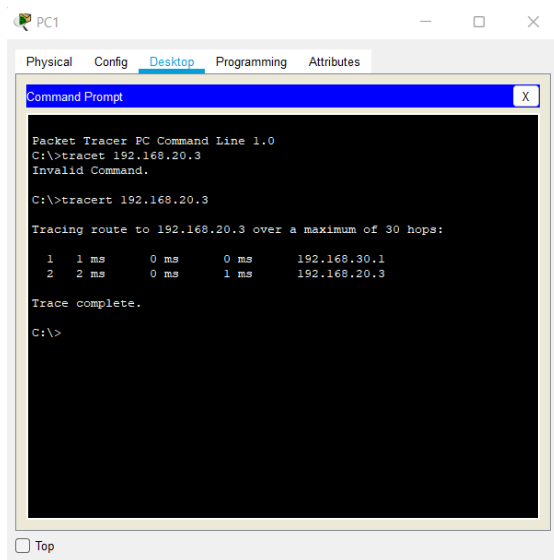
C:\>ping 192.168.10.12

Pinging 192.168.10.12 with 32 bytes of data:

Reply from 192.168.10.12: bytes=32 time<1ms TTL=254
Reply from 192.168.10.12: bytes=32 time<1ms TTL=254
Reply from 192.168.10.12: bytes=32 time<1ms TTL=254
Reply from 192.168.10.12: bytes=32 time<1ms TTL=254

Ping statistics for 192.168.10.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
```

☐ Top



PC1

Physical Config Desktop Programming Attributes

Packet Tracer PC Command Line 1.0

```
C:\>tracet 192.168.20.3
Invalid Command.

C:\>tracert 192.168.20.3

Tracing route to 192.168.20.3 over a maximum of 30 hops:

  1  1 ms    0 ms    0 ms    192.168.30.1
  2  2 ms    0 ms    1 ms    192.168.20.3

Trace complete.

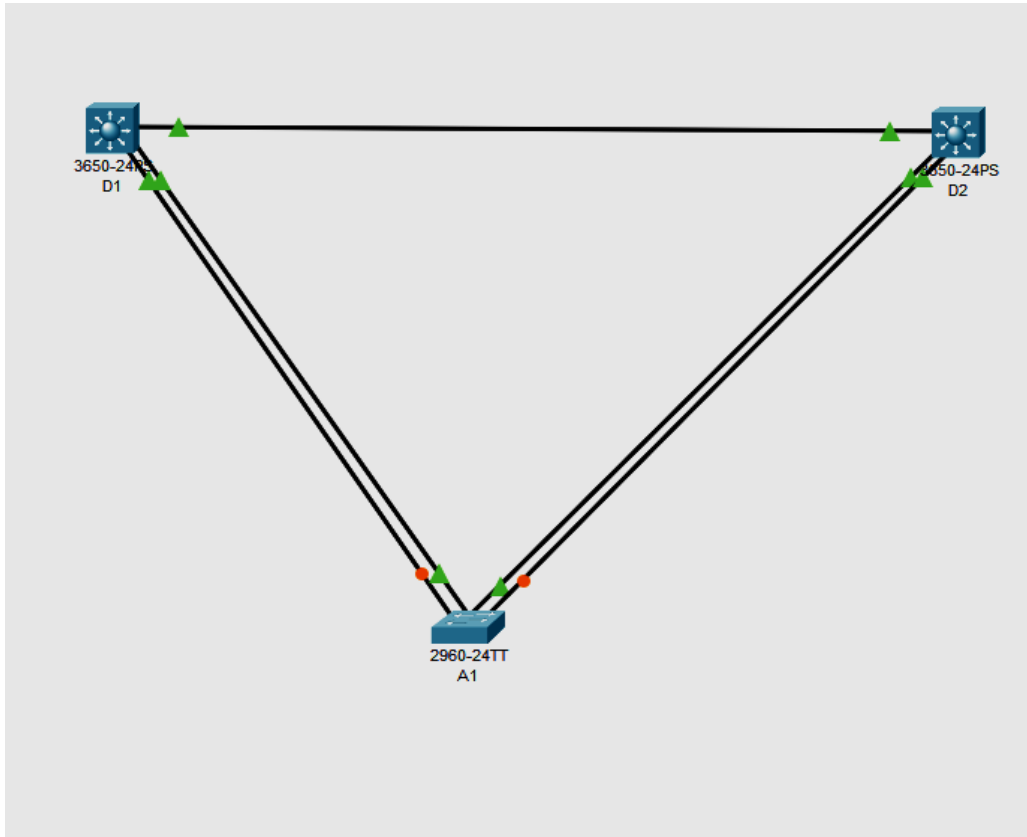
C:\>
```

☐ Top

Practical 6

Aim:- Observe STP Topology Changes and Implement RSTP

Topology



Addressing Table

| Device | Interface | IPv4 Address |
|--------|-----------|--------------|
| D1 | VLAN1 | 10.0.0.1/8 |
| D2 | VLAN1 | 10.0.0.2/8 |
| A1 | VLAN1 | 10.0.0.3/8 |

Part 1: Build the Network and Configure Basic Device Settings and Interface Addressing

In Part 1, you will set up the network topology and configure basic settings and interface addressing on routers.

Step 1: Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram, and cable as necessary.

Step 2: Configure basic settings for each switch.

Console into each switch, enter global configuration mode, and apply the basic settings and interface addressing. The startup configuration is provided below for each switch in the topology

Switch D1

```
hostname D1
```

```
spanning-tree mode pvst
line con 0
exec-timeout 0 0
logging synchronous
exit
interface range g1/0/1-24
shutdown
exit
```

```
interface range g1/0/1, g1/0/5-6
switchport mode trunk
no shutdown
exit
```

```
vlan 2
name SecondVLAN
exit
```

```
interface vlan 1
ip address 10.0.0.1 255.0.0.0
no shut
exit
```

Switch D2

```
hostname D2
spanning-tree mode pvst
line con 0
exec-timeout 0 0
logging synchronous
exit
```

```
interface range g1/0/1-24
shutdown
exit
```

```
interface range g1/0/1, g1/0/5-6
switchport mode trunk
no shutdown
exit
```

```
vlan 2
name SecondVLAN
exit
```

```
interface vlan 1
ip address 10.0.0.2 255.0.0.0
```

```
no shut
exit
```

Switch A1

```
hostname A1
spanning-tree mode pvst
line con 0
exec-timeout 0 0
logging synchronous
exit
```

```
interface range f0/1-24, g0/1-2
shutdown
exit
```

```
interface range f0/1-4
switchport mode trunk
no shutdown
exit
```

```
vlan 2
name SecondVLAN
exit
```

```
interface vlan 1
ip address 10.0.0.3 255.0.0.0
no shut
exit
```

Part 2: Discover the Default Spanning Tree

Step 1: Find the root bridge.

issue the command **show spanning-tree** command all switches

```
D1#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0000.0CAC.07A2
             Cost        4
             Port        1(GigabitEthernet1/0/1)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     000A.F3C7.3D8C
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface    Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/5      Desg FWD 19      128.5    P2p
Gi1/0/6      Desg FWD 19      128.6    P2p
Gi1/0/1      Root FWD 4       128.1    P2p

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32770
             Address     0000.0CAC.07A2
             Cost        4
             Port        1(GigabitEthernet1/0/1)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     000A.F3C7.3D8C
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface    Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/5      Desg FWD 19      128.5    P2p
Gi1/0/6      Desg FWD 19      128.6    P2p
Gi1/0/1      Root FWD 4       128.1    P2p
```

```

A1#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0000.0CAC.07A2
             Cost        19
             Port        3(FastEthernet0/3)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
             Address     00D0.97C0.327D
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface                Role Sts Cost      Prio.Nbr Type
-----
Fa0/4                    Altn BLK 19      128.4   P2p
Fa0/2                    Altn BLK 19      128.2   P2p
Fa0/3                    Root FWD 19      128.3   P2p
Fa0/1                    Altn BLK 19      128.1   P2p

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32770
             Address     0000.0CAC.07A2
             Cost        19
             Port        3(FastEthernet0/3)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770 (priority 32768 sys-id-ext 2)
             Address     00D0.97C0.327D
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface                Role Sts Cost      Prio.Nbr Type
-----
Fa0/4                    Altn BLK 19      128.4   P2p
Fa0/2                    Altn BLK 19      128.2   P2p
Fa0/3                    Root FWD 19      128.3   P2p
Fa0/1                    Altn BLK 19      128.1   P2p

D2#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0000.0CAC.07A2
             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     0000.0CAC.07A2
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface                Role Sts Cost      Prio.Nbr Type
-----
Gil/0/6                Desg FWD 19      128.6   P2p
Gil/0/1                Desg FWD 4       128.1   P2p
Gil/0/5                Desg FWD 19      128.5   P2p

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32770
             Address     0000.0CAC.07A2
             This bridge is the root
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770 (priority 32768 sys-id-ext 2)
             Address     0000.0CAC.07A2
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface                Role Sts Cost      Prio.Nbr Type
-----
Gil/0/6                Desg FWD 19      128.6   P2p
Gil/0/1                Desg FWD 4       128.1   P2p
Gil/0/5                Desg FWD 19      128.5   P2p

```

From above it is seen that D2 is the root Port as well as MAC address of D2 is minimum compared to D1 and A1.

Our topology does not really illustrate the difference between port cost and path cost very well, so we will introduce a change in the network to achieve this. At D2, shutdown the g1/0/1 interface. The result of this is that D1 will have to change the port it considers root, and we will then see the difference between port cost and path cost.

D2(config)# **interface g1/0/1** agar topolofy me a1 root hai to ye a1 me karna hai

D2(config-if)# **shutdown**

Now in d1,

D2# **show spanning-tree**

```
D1#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0000.0CAC.07A2
             Cost        38
             Port        5(GigabitEthernet1/0/5)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
             Address     000A.F3C7.3D8C
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface                Role Sts Cost      Prio.Nbr Type
-----
G11/0/5                  Root FWD 19        128.5   P2p
G11/0/6                  Altn BLK 19        128.6   P2p

VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32770
             Address     0000.0CAC.07A2
             Cost        38
             Port        5(GigabitEthernet1/0/5)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770 (priority 32768 sys-id-ext 2)
             Address     000A.F3C7.3D8C
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface                Role Sts Cost      Prio.Nbr Type
-----
G11/0/5                  Root FWD 19        128.5   P2p
G11/0/6                  Altn BLK 19        128.6   P2p
```

The root path cost is now 38, while the root port cost is 19. For D1 to reach the root bridge D2, it must traverse two FastEthernet links, and 19 times 2 is 38.

Part 3: Implement and Observe Rapid Spanning Tree Protocol

1. On D1, issue the **debug spanning-tree events** command, and then issue the **shutdown** command for interface g1/0/1 and observe the output.

D1# **debug spanning-tree events**

D1# **config t**

D1(config)# **interface g1/0/1**

D1(config-if)# **shutdown**

Now change the mode to rapid spanning tree mode in D2 and then run no shut command in D1 also observe the time taken to connect

D2(config)# **spanning-tree mode rapid-pvst**

D1(config-if)# **no shut**

2. Also change the mode of spanning tree in A1 and observe the changes

A1(config)# **spanning-tree mode rapid-pvst**

A1(config)#

Dec 24 13:31:51.023: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down

Dec 24 13:31:51.081: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up

A1(config)#

A1 was the last switch that was configured for RSTP. As you can see, interface VLAN1 was only down for 0.048 seconds. This is the “rapid” in rapid spanning tree.

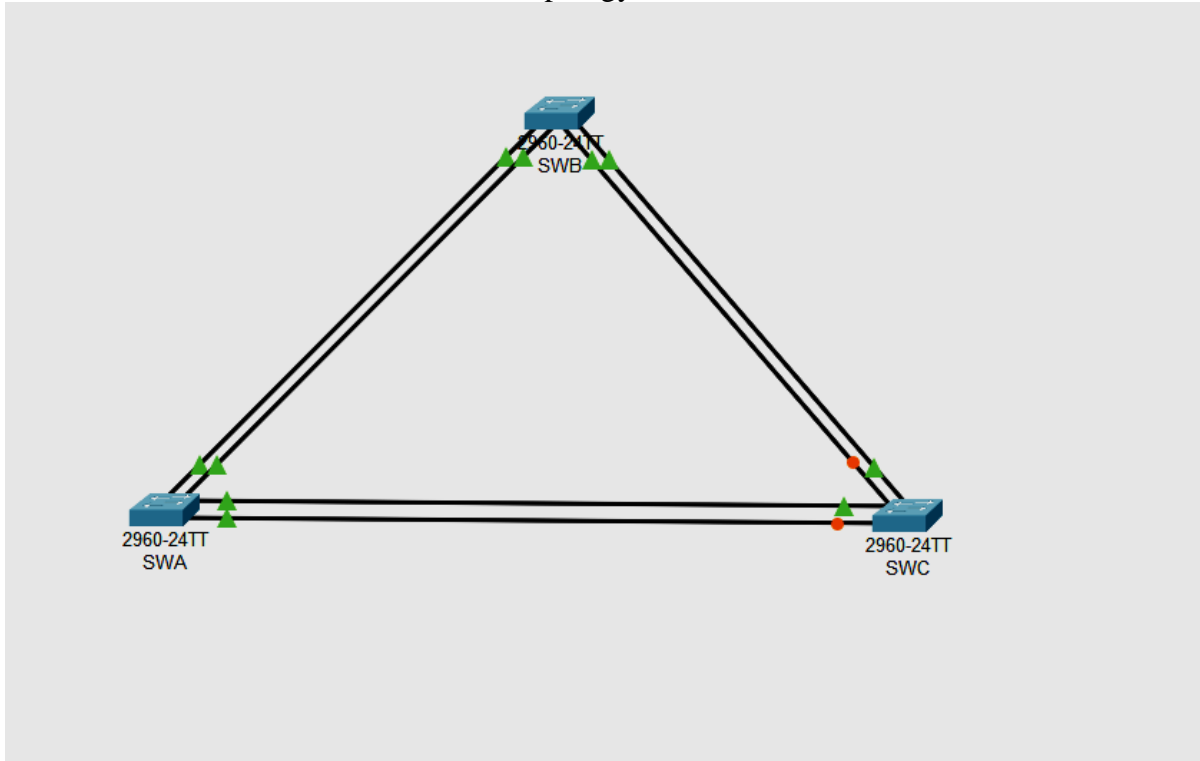
Practical 7

Aim:-

1. IMPLEMENT ETHERCHANNEL

Part 1: Build the network.

Use the table below to build the switch topology.



Step 2: Name the devices.

Step 3: Connect the devices.

d. Connect the devices as specified in the table below.

| Port Channel | Devices | Port Connections | Type |
|--------------|------------|------------------|------|
| 1 | SWA to SWB | G0/1 to G0/1 | PAgP |
| | | G0/2 to G0/2 | |
| 2 | SWA to SWC | F0/21 to F0/21 | LACP |
| | | F0/22 to F0/22 | |
| 3 | SWB to SWC | F0/23 to F0/23 | LACP |
| | | F0/24 to F0/24 | |

Part 2: Configure EtherChannel

Open configuration window On each switch, configure the ports that will be used in the Port Channels as static trunk ports.

Step 1: Configure a PAgP EtherChannel.

Follow the procedure that was used in previous activities to configure Port Channel 1 as a PAgP EtherChannel between SWA and SWB.

Both sides should negotiate the EtherChannel.

In Switch 1(swa):

```
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int range g0/1-2
Switch(config-if-range)#switchport mode trunk
Switch(config-if-range)#channel-group 1 mode desirable
Switch(config-if-range)#no shut
Switch(config-if-range)#exit
```

Switch 2(swb)

```
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int range g0/1-2
Switch(config-if-range)#switchport mode trunk
Switch(config-if-range)#channel-group 1 mode desirable
Switch(config-if-range)#no shut
Switch(config-if-range)#exit
```

Step 2: Configure a LACP EtherChannel.**Configure Port Channel 2 as an LACP channel between SWA and SWC.****Both sides should negotiate the EtherChannel.**

```
Switch 1 (swa)
Switch(config)#int range fa0/21-22
Switch(config-if-range)#switchport mode trunk
Switch(config-if-range)#channel-group 2 mode active
Switch(config-if-range)#no shut
Switch(config-if-range)#exit
```

```
Switch 3 (swc)
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int range fa0/21-22
Switch(config-if-range)#switchport mode trunk
Switch(config-if-range)#channel-group 2 mode active
Switch(config-if-range)#no shut
Switch(config-if-range)#exit
```

Step 3: Configure a Backup LACP

**EtherChannel Configure Port Channel 3 channel as an LACP channel between SWB and SWC.
In this case, SWC initiates negotiation with SWB.
SWB does not initiate negotiation of the channel**

```
Switch3(swc)
Switch(config)#int range fa0/23-24
Switch(config-if-range)#switchport mode trunk
Switch(config-if-range)#channel-group 3 mode active
Switch(config-if-range)#no shut
Switch(config-if-range)#exit
```

```
Switch2(swb)
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int range fa0/23-24
Switch(config-if-range)#switchport mode trunk
Switch(config-if-range)#channel-group 3 mode passive
Switch(config-if-range)#no shut
Switch(config-if-range)#exit
```

Output to see all the etherchannel connections
Go to any one switch and write the follow command

Swc
Switch#show etherchannel summary
Flags: D - down P - in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
u - unsuitable for bundling
w - waiting to be aggregated
d - default port

Number of channel-groups in use: 2
Number of aggregators: 2

Group Port-channel Protocol Ports

-----+-----+-----+-----

2 Po2(SU) LACP Fa0/21(P) Fa0/22(I)
3 Po3(SU) LACP Fa0/23(D) Fa0/24(P)

Swa

Switch#show etherchannel summary
 Flags: D - down P - in port-channel
 I - stand-alone s - suspended
 H - Hot-standby (LACP only)
 R - Layer3 S - Layer2
 U - in use f - failed to allocate aggregator
 u - unsuitable for bundling
 w - waiting to be aggregated
 d - default port

Number of channel-groups in use: 2
 Number of aggregators: 2

Group Port-channel Protocol Ports

-----+-----+-----+-----

1 Po1(SU) PAgP Gig0/1(P) Gig0/2(P)
 2 Po2(SU) LACP Fa0/21(P) Fa0/22(D)

Swb

Switch#show etherchannel summary
 Flags: D - down P - in port-channel
 I - stand-alone s - suspended
 H - Hot-standby (LACP only)
 R - Layer3 S - Layer2
 U - in use f - failed to allocate aggregator
 u - unsuitable for bundling
 w - waiting to be aggregated
 d - default port

Number of channel-groups in use: 2
 Number of aggregators: 2

Group Port-channel Protocol Ports

-----+-----+-----+-----

1 Po1(SU) PAgP Gig0/1(P) Gig0/2(P)
 3 Po3(SU) LACP Fa0/23(I) Fa0/24(P)

Checking trunking

Swc

Switch#show interface trunk
 Port Mode Encapsulation Status Native vlan
 Po2 on 802.1q trunking 1
 Po3 on 802.1q trunking 1

Port Vlans allowed on trunk
 Po2 1-1005
 Po3 1-1005

Port Vlans allowed and active in management domain
 Po2 1
 Po3 1

Port Vlans in spanning tree forwarding state and not pruned
 Po2 1
 Po3 1

Swb

Switch#show interface trunk
 Port Mode Encapsulation Status Native vlan
 Po1 on 802.1q trunking 1
 Po3 on 802.1q trunking 1

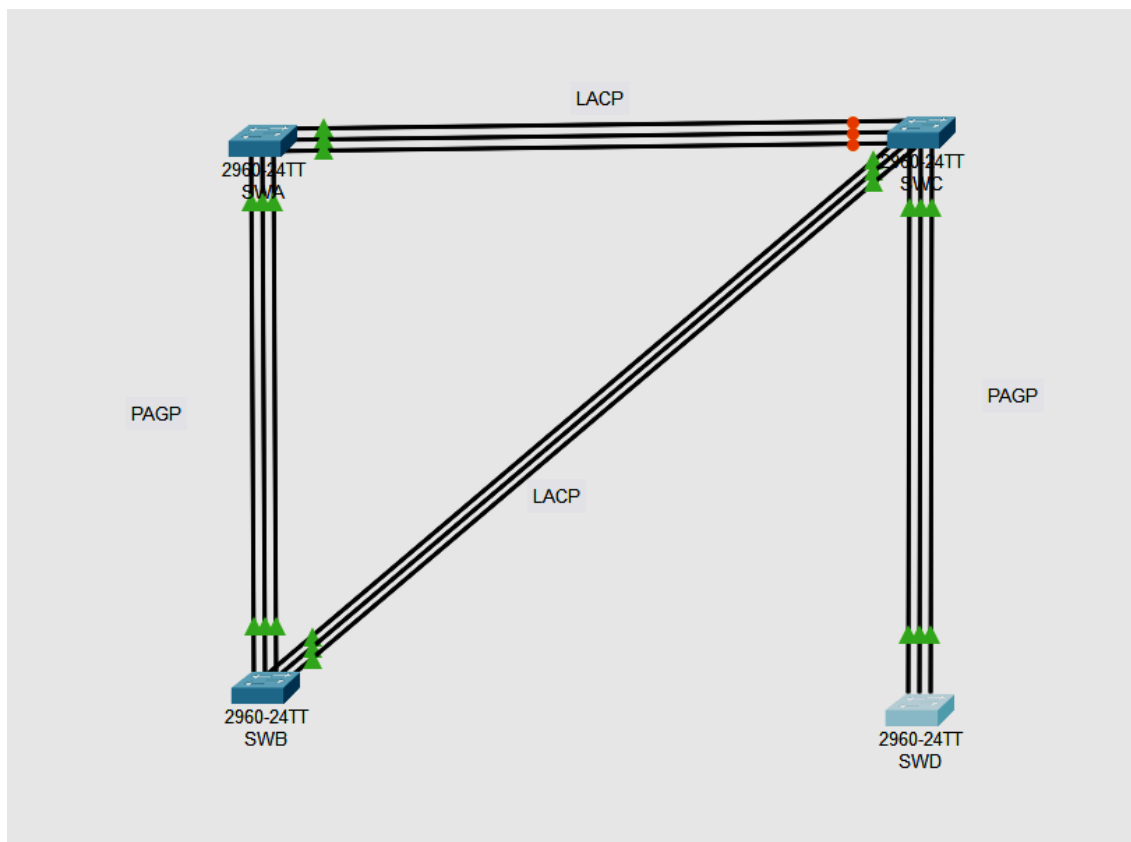
Port Vlans allowed on trunk
 Po1 1-1005
 Po3 1-1005

Port Vlans allowed and active in management domain
 Po1 1
 Po3 1

Port Vlans in spanning tree forwarding state and not pruned
 Po1 1
 Po3 1

2. Tune and optimize Etherchannel Operations

TOPOLOGY



CONNECTION TABLE

| Port Channel | Devices | Port Connections | Type |
|--------------|------------|------------------|------|
| 1 | SWA to SWB | F0/1 to F0/1 | PAgP |
| | | F0/2 to F0/2 | |
| | | F0/3 to F0/3 | |
| 2 | SWA to SWC | F0/10 to F0/10 | LACP |
| | | F0/11 to F0/11 | |
| | | F0/12 to F0/12 | |
| 3 | SWB to SWC | F0/15 to F0/15 | LACP |
| | | F0/16 to F0/16 | |
| | | F0/17 to F0/17 | |
| 4 | SWC to SWD | F0/22 to F0/22 | PAgP |
| | | F0/23 to F0/23 | |
| | | F0/24 to F0/24 | |
| | | | |

Part 1: Build the network.

Step 1: Obtain the devices that are required.

Step 2: Name the devices.

Step 3: Connect the devices. According to the connection table

Part 2: Configure EtherChannel

On each switch, configure the ports that will be used in the Port Channels as static trunk ports

Step 1: Configure a PAgP EtherChannel.

Configure Port Channel 1 as a PAgP EtherChannel between SWA and SWB. Both sides should negotiate the EtherChannel.

```
SWA>enable
SWA#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
SWA(config)#int range f0/1-3
SWA(config-if-range)#switchport mode trunk
SWA(config-if-range)#channel-group 1 mode desirable
SWA(config-if-range)#no shut
SWA(config-if-range)#exit
```

```
SWB>en
SWB#conf t
Enter configuration commands, one per line. End with CNTL/Z.
SWB(config)#int range f0/1-3
SWB(config-if-range)#switchport mode trunk
SWB(config-if-range)#channel-group 1 mode desirable
SWB(config-if-range)#no shut
SWB(config-if-range)#exit
```

Step 2: Configure a LACP EtherChannel.

Configure Port Channel 2 as an LACP channel between SWA and SWC. Both sides should negotiate the EtherChannel.

```
SWA>en
SWA#conf t
Enter configuration commands, one per line. End with CNTL/Z.
```

```
SWA(config)#int range fa0/10-12
SWA(config-if-range)#switchport mode trunk
SWA(config-if-range)#channel-group 2 mode active
SWA(config-if-range)#no shut
SWA(config-if-range)#exit
```

```
SWC>en
SWC#conf t
Enter configuration commands, one per line. End with CNTL/Z.
SWC(config)#int range fa0/10-12
SWC(config-if-range)#switchport mode trunk
SWC(config-if-range)#channel-group 2 mode active
SWC(config-if-range)#no shut
SWC(config-if-range)#exit
```

Step 3: Configure a Backup LACP EtherChannel

Configure Port Channel 3 channel as an LACP channel between SWB and SWC. In this case, SWB initiates negotiation with SWC. SWC does not initiate negotiation of the channel

```
SWB>en
SWB#conf t
Enter configuration commands, one per line. End with CNTL/Z.
SWB(config)#int range fa0/15-17
SWB(config-if-range)#switchport mode trunk
SWB(config-if-range)#channel-group 3 mode active
SWB(config-if-range)#no shut
SWB(config-if-range)#exit
```

```
SWC(config)#int range fa0/15-17
SWC(config-if-range)#switchport mode trunk
SWC(config-if-range)#channel-group 3 mode passive
SWC(config-if-range)#no shut
SWC(config-if-range)#exit
```

Step 4: Configure a Backup PAgP EtherChannel

Configure Port Channel 4 channel as an LACP channel between SWC and SWD. In this case, SWC initiates negotiation with SWD. SWD does not initiate negotiation of the channel.

```
SWC(config)#int range fa0/22-24
SWC(config-if-range)#switchport mode trunk
SWC(config-if-range)#channel-group 4 mode desirable
SWC(config-if-range)#no shut
SWC(config-if-range)#exit
```

```
SWD>en
```

SWD#conf t

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

SWD(config)#int range fa0/22-24

SWD(config-if-range)#switchport mode trunk

SWD(config-if-range)#channel-group 4 mode auto

SWD(config-if-range)#no shut

SWD(config-if-range)#exit

Part 4: Checking output

Output to see all the etherchannel connections

Go to any one switch and write the follow command

```
SWA>en
SWA#show etherchannel summary
Flags: D - down          P - in port-channel
       I - stand-alone s - suspended
       H - Hot-standby (LACP only)
       R - Layer3        S - Layer2
       U - in use        f - failed to allocate aggregator
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port

Number of channel-groups in use: 2
Number of aggregators:          2

Group  Port-channel  Protocol    Ports
-----+-----+-----+-----
1      Po1(SU)          PAgP        Fa0/1(P) Fa0/2(P) Fa0/3(P)
2      Po2(SU)          LACP        Fa0/10(P) Fa0/11(P) Fa0/12(P)
SWA#show interface trunk
Port      Mode          Encapsulation  Status      Native vlan
Po1       on            802.1q         trunking    1
Po2       on            802.1q         trunking    1

Port      Vlans allowed on trunk
Po1       1-1005
Po2       1-1005

Port      Vlans allowed and active in management domain
Po1       1
Po2       1

Port      Vlans in spanning tree forwarding state and not pruned
Po1       1
Po2       1
```

```

SWC>en
SWC#show etherchannel summary
Flags:  D - down          P - in port-channel
        I - stand-alone  s - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

Number of channel-groups in use: 3
Number of aggregators:          3

Group  Port-channel  Protocol    Ports
-----+-----+-----+-----
2      Po2(SU)        LACP       Fa0/10(P) Fa0/11(P) Fa0/12(P)
3      Po3(SU)        LACP       Fa0/15(P) Fa0/16(P) Fa0/17(P)
4      Po4(SU)        PAgP       Fa0/22(P) Fa0/23(P) Fa0/24(P)
SWC#show interface trunk
Port      Mode          Encapsulation  Status      Native vlan
Po2       on           802.1q         trunking    1
Po3       on           802.1q         trunking    1
Po4       on           802.1q         trunking    1

Port      Vlans allowed on trunk
Po2       1-1005
Po3       1-1005
Po4       1-1005

Port      Vlans allowed and active in management domain
Po2       1
Po3       1
Po4       1

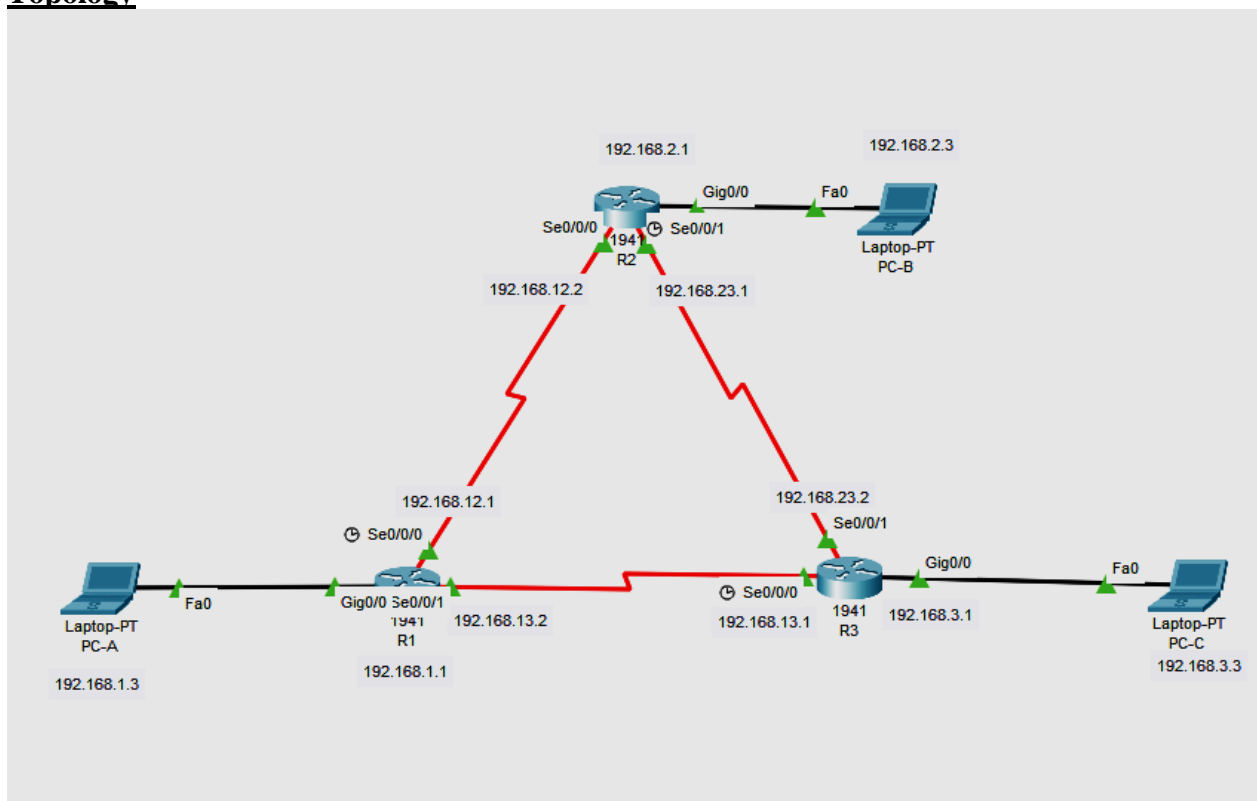
Port      Vlans in spanning tree forwarding state and not pruned
Po2       none
Po3       1
Po4       1

```


Practical 8

Aim:- OSPF Implementation

Topology



Addressing Table

| Device | Interface | IP Address | Subnet Mask | Default Gateway |
|--------|--------------|--------------|-----------------|-----------------|
| R1 | G0/0 | 192.168.1.1 | 255.255.255.0 | N/A |
| | S0/0/0 (DCE) | 192.168.12.1 | 255.255.255.252 | N/A |
| | S0/0/1 | 192.168.13.1 | 255.255.255.252 | N/A |
| R2 | G0/0 | 192.168.2.1 | 255.255.255.0 | N/A |
| | S0/0/0 | 192.168.12.2 | 255.255.255.252 | N/A |
| | S0/0/1 (DCE) | 192.168.23.1 | 255.255.255.252 | N/A |
| R3 | G0/0 | 192.168.3.1 | 255.255.255.0 | N/A |
| | S0/0/0 (DCE) | 192.168.13.2 | 255.255.255.252 | N/A |
| | S0/0/1 | 192.168.23.2 | 255.255.255.252 | N/A |
| PC-A | NIC | 192.168.1.3 | 255.255.255.0 | 192.168.1.1 |
| PC-B | NIC | 192.168.2.3 | 255.255.255.0 | 192.168.2.1 |
| PC-C | NIC | 192.168.3.3 | 255.255.255.0 | 192.168.3.1 |

Part 1: Build the Network and Configure Basic Device Settings

In Part 1, you set up the network topology and configure basic settings on the PC hosts and routers.

Step 1: Cable the network as shown in the topology.

Step 2: Configure basic settings for each router.

On R1:

```
Router>enable
```

```
Router#configure terminal
```

```
Router(config)#interface GigabitEthernet0/0
```

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

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

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

```
Router(config)#interface Serial0/0/0
```

```
Router(config-if)#ip address 192.168.12.1 255.255.255.252
```

```
Router(config-if)#clock rate 64000
```

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

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

```
Router(config)#interface Serial0/0/1
```

```
Router(config-if)#ip address 192.168.13.2 255.255.255.252
```

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

Do the same to configure the R2 and R3 router as well according to the addressing table. Assign the respective ip address to the PC

Part 2: Configure and Verify OSPF Routing

In Part 2, you will configure OSPFv2 routing on all routers in the network and then verify that routing tables are updated correctly. After OSPF has been verified, you will configure OSPF authentication on the links for added security.

Step 1: Configure OSPF on R1.

Use the **router ospf** command in global configuration mode to enable OSPF on R1.

```
Router(config)# router ospf 1
```

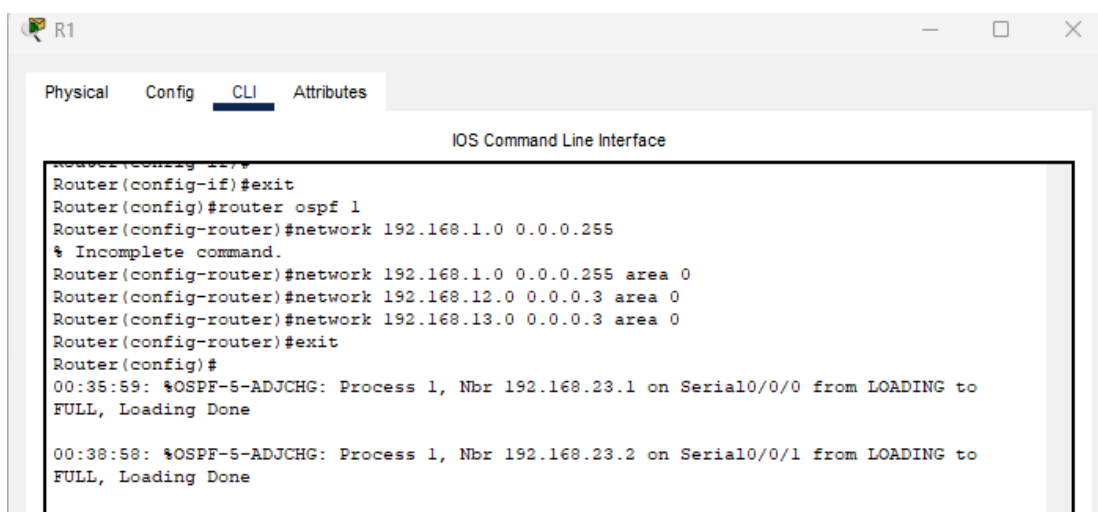
Note: The OSPF process id is kept locally and has no meaning to other routers on the network.

Configure the **network** statements for the networks on R1. Use an area ID of 0.

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

```
Router(config-router)# network 192.168.12.0 0.0.0.3 area 0
```

```
Router(config-router)# network 192.168.13.0 0.0.0.3 area 0
```



```

R1
Physical Config CLI Attributes
IOS Command Line Interface
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 192.168.1.0 0.0.0.255
% Incomplete command.
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Router(config-router)#network 192.168.12.0 0.0.0.3 area 0
Router(config-router)#network 192.168.13.0 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#
00:35:59: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.23.1 on Serial0/0/0 from LOADING to FULL, Loading Done
00:38:58: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.23.2 on Serial0/0/1 from LOADING to FULL, Loading Done

```

Step 1B: Configure OSPF on R2 and R3.

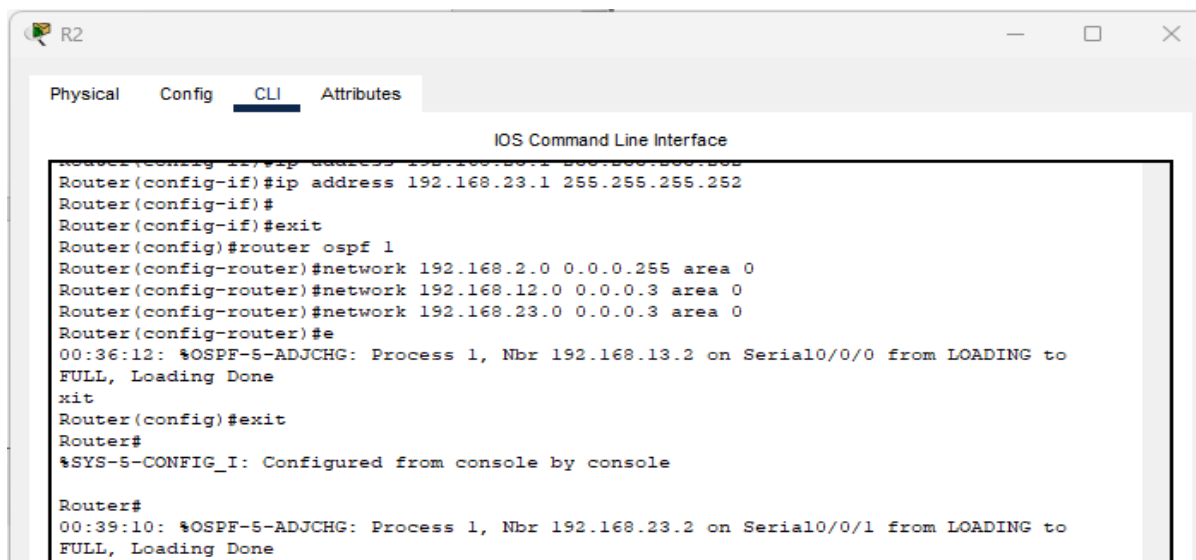
Use the **router ospf** command and add the **network** statements for the networks on R2 and R3. Neighbor adjacency messages display on R1 when OSPF routing is configured on R2 and R3.

On R2:

```

Router(config)#router ospf 1
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0
Router(config-router)#network 192.168.12.0 0.0.0.3 area 0
Router(config-router)#network 192.168.23.0 0.0.0.3 area 0
Router(config)#exit

```



```

R2
Physical Config CLI Attributes
IOS Command Line Interface
Router(config-if)#ip address 192.168.23.1 255.255.255.252
Router(config-if)#
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0
Router(config-router)#network 192.168.12.0 0.0.0.3 area 0
Router(config-router)#network 192.168.23.0 0.0.0.3 area 0
Router(config-router)#e
00:36:12: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.13.2 on Serial0/0/0 from LOADING to FULL, Loading Done
xit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#
00:39:10: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.23.2 on Serial0/0/1 from LOADING to FULL, Loading Done

```

On R3:

```

Router(config)#router ospf 1
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#network 192.168.13.0 0.0.0.3 area 0
Router(config-router)#network 192.168.23.0 0.0.0.3 area 0

```

Router(config-router)#exit

```

R3
Physical Config CLI Attributes
IOS Command Line Interface
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/0/1
Router(config-if)#ip address 192.168.23.2 255.255.255.252
Router(config-if)#ip address 192.168.23.2 255.255.255.252
Router(config-if)#
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#network 192.168.13.0 0.0.0.3 area 0
Router(config-router)#network 192.168.23.0 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#
00:40:24: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.23.1 on Serial0/0/1 from LOADING to FULL, Loading Done
00:40:24: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.13.2 on Serial0/0/0 from LOADING to FULL, Loading Done

```

Step 2: Check the connectivity

Ping from PC-B to PC-A after configuring OSPF and check the route with the help of tracert command
So see the route

```

PC-A
Physical Config Desktop Programming Attributes
Command Prompt
Ping statistics for 192.168.4.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 35ms, Average = 15ms

C:\>ping 192.168.3.3

Pinging 192.168.3.3 with 32 bytes of data:

Request timed out.
Reply from 192.168.3.3: bytes=32 time=3ms TTL=125
Reply from 192.168.3.3: bytes=32 time=4ms TTL=125
Reply from 192.168.3.3: bytes=32 time=4ms TTL=125

Ping statistics for 192.168.3.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 3ms, Maximum = 4ms, Average = 3ms

C:\>tracert 192.168.3.3

Tracing route to 192.168.3.3 over a maximum of 30 hops:

  0  0 ms    0 ms    0 ms    192.168.1.1
  1  1 ms    1 ms    1 ms    192.168.15.1
  2  1 ms    1 ms    1 ms    192.168.13.2
  3  2 ms    1 ms    1 ms    192.168.3.3

Trace complete.

C:\>

```

Step 3: Verify OSPF neighbors and routing information.

Issue the **show ip ospf neighbor** command to verify that each router lists the other routers in the network as neighbors.

Issue the **show ip route** command to verify that all networks display in the routing table on all routers.

```
R1>en
R1#show ip ospf neighbor
```

| Neighbor ID | Pri | State | Dead Time | Address | Interface |
|--------------|-----|---------|-----------|--------------|-------------|
| 192.168.13.1 | 0 | FULL/ - | 00:00:32 | 192.168.12.2 | Serial0/0/0 |
| 192.168.15.1 | 0 | FULL/ - | 00:00:31 | 192.168.15.1 | Serial0/0/1 |

```
R1#
```

Step 4: Verify OSPF protocol settings.

The **show ip protocols** command is a quick way to verify vital OSPF configuration information. This information includes the OSPF process ID, the router ID, networks the router is advertising, the neighbors the router is receiving updates from, and the default administrative distance, which is 110 for OSPF.

```
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.15.2
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.1.0 0.0.0.255 area 0
    192.168.15.0 0.0.0.3 area 0
    192.168.12.0 0.0.0.3 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.13.1      110          00:14:41
    192.168.15.1      110          00:11:57
    192.168.15.2      110          00:12:14
  Distance: (default is 110)

R1#show ip interface brief
```

Step 5: Verify OSPF interface settings.

Issue the **show ip ospf interface brief** command to display a summary of OSPF-enabled interfaces.

```
R1#show ip ospf interface brief
```

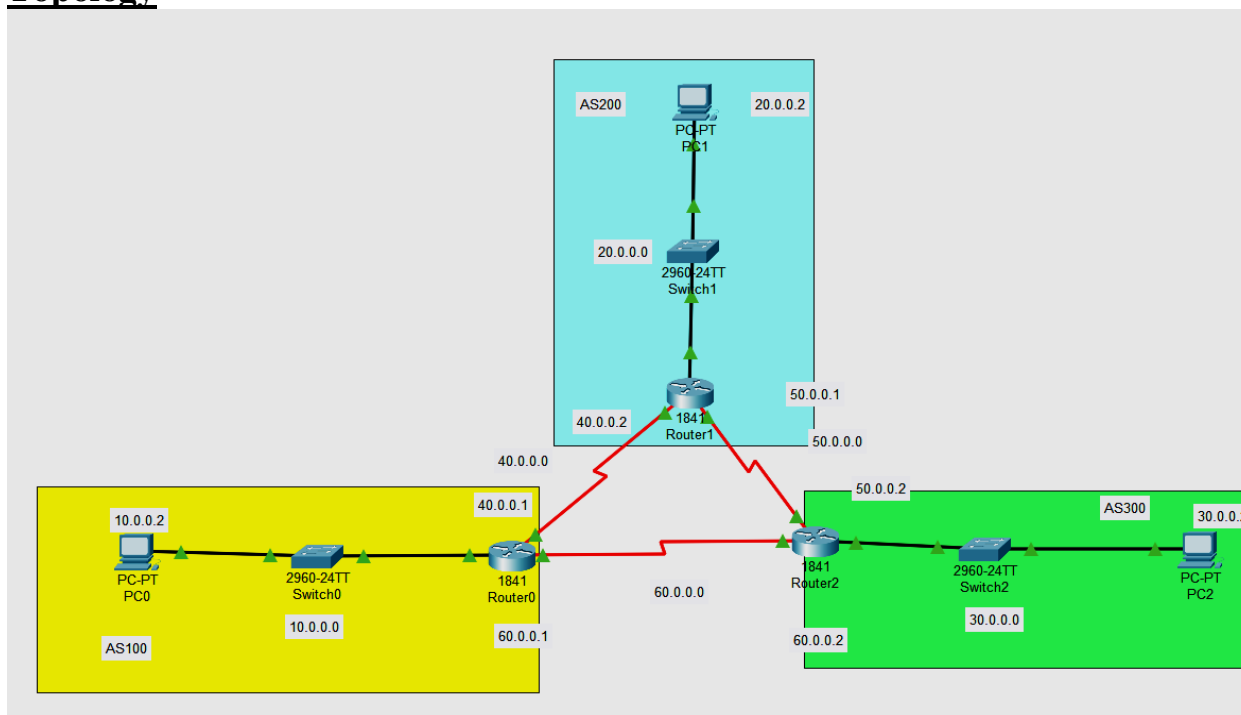
| Interface | IP-Address | OK? | Method | Status |
|--------------------|--------------|-----|--------|-----------------------|
| Protocol | | | | |
| GigabitEthernet0/0 | 192.168.1.1 | YES | manual | up |
| GigabitEthernet0/1 | unassigned | YES | unset | administratively down |
| Serial0/0/0 | 192.168.12.1 | YES | manual | up |
| Serial0/0/1 | 192.168.15.2 | YES | manual | up |
| Vlan1 | unassigned | YES | unset | administratively down |

```
R1#
```

Practical 9

Aim:- Implement BGP Communities

Topology



Addressing Table

| Device | Interface | IP Address | Subnet Mask | Default Gateway |
|--------|-----------|------------|-------------|-----------------|
| R1 | Fa0/0 | 10.0.0.1 | 255.0.0.0 | N/A |
| R1 | S0/0/0 | 40.0.0.1 | 255.0.0.0 | |
| R1 | S0/0/1 | 60.0.0.1 | 255.0.0.0 | |
| R2 | Fa0/0 | 20.0.0.1 | 255.0.0.0 | N/A |
| R2 | S0/0/0 | 40.0.0.2 | 255.0.0.0 | N/A |
| R2 | S0/0/1 | 50.0.0.1 | 255.0.0.0 | N/A |
| R3 | Fa0/0 | 30.0.0.1 | 255.0.0.0 | N/A |
| R3 | S0/0/1 | 50.0.0.2 | 255.0.0.0 | N/A |
| R3 | S0/0/0 | 60.0.0.2 | 255.0.0.0 | N/A |
| PC-A | Fa0/0 | 10.0.0.2 | 255.0.0.0 | 10.0.0.1 |
| PC-B | Fa0/0 | 20.0.0.2 | 255.0.0.0 | 20.0.0.1 |
| PC-C | Fa0/0 | 30.0.0.2 | 255.0.0.0 | 30.0.0.1 |

Part 1: Build the network.

Step 1: Obtain the devices that are required.

Step 2: Name the devices.

Step 3: Connect the devices. According to the connection table

Part 2: Configure BGP

Step 1: Configuring Bgp in each router with specific address

Go to each router and configure there neighbour and assign the area

For Router 0 , consider neighbor address 40.0.0.2 and 60.0.0.2

For Router 1 , consider neighbor address 40.0.0.1 and 50.0.0.2

For Router 2 , consider neighbor address 50.0.0.1 and 60.0.0.1

R1:

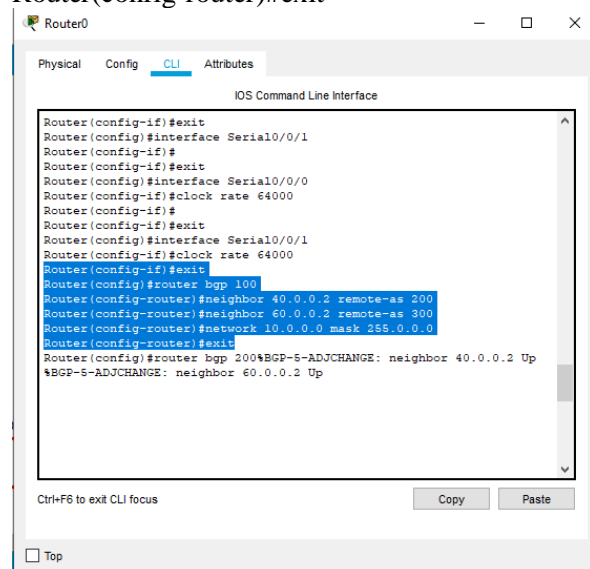
```
Router(config)#router bgp 100
```

```
Router(config-router)#neighbor 40.0.0.2 remote-as 200
```

```
Router(config-router)# neighbor 60.0.0.2 remote-as 300
```

```
Router(config-router)#network 10.0.0.0 mask 255.0.0.0
```

```
Router(config-router)#exit
```



R2

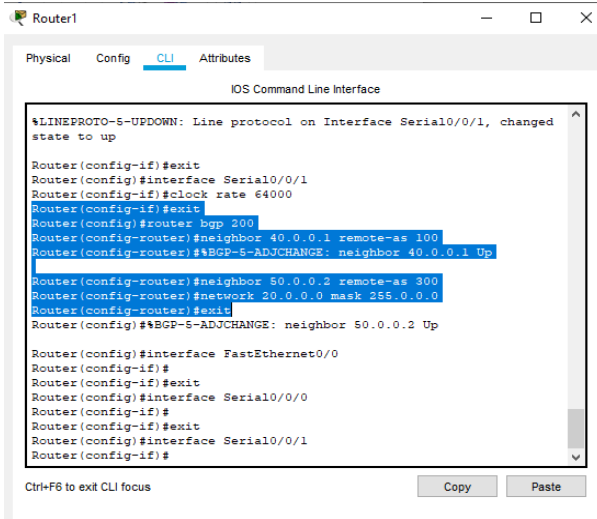
```
Router(config)#router bgp 200
```

```
Router(config-router)#neighbor 40.0.0.1 remote-as 100
```

```
Router(config-router)# neighbor 50.0.0.2 remote-as 300
```

```
Router(config-router)#network 20.0.0.0 mask 255.0.0.0
```

```
Router(config-router)#exit
```



Router1

Physical Config CLI Attributes

IOS Command Line Interface

```

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/1, changed
state to up
Router(config-if)#exit
Router(config)#interface Serial0/0/1
Router(config-if)#clock rate 64000
Router(config-if)#exit
Router(config)#router bgp 200
Router(config-router)#neighbor 40.0.0.1 remote-as 100
Router(config-router)#%BGP-5-ADJCHANGE: neighbor 40.0.0.1 Up
Router(config-router)#neighbor 50.0.0.2 remote-as 300
Router(config-router)#network 20.0.0.0 mask 255.0.0.0
Router(config-router)#exit
Router(config)#%BGP-5-ADJCHANGE: neighbor 50.0.0.2 Up
Router(config)#interface FastEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/0/1
Router(config-if)#

```

Ctrl+F6 to exit CLI focus

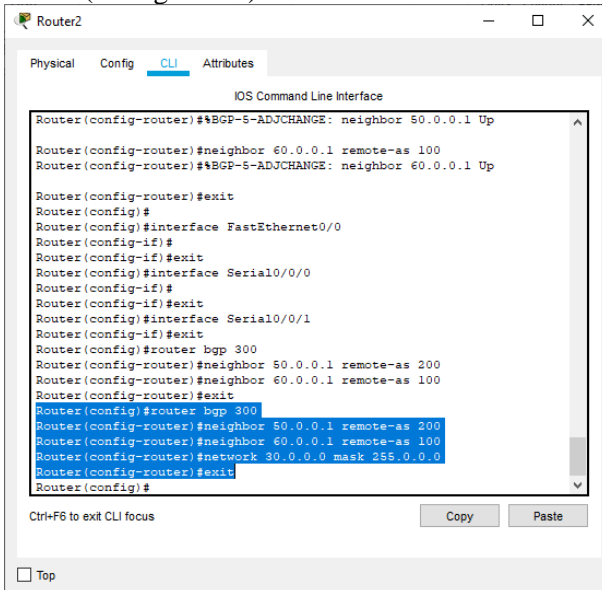
Copy Paste

R3

```

Router(config)#router bgp 300
Router(config-router)#neighbor 60.0.0.1 remote-as 100
Router(config-router)# neighbor 50.0.0.1 remote-as 200
Router(config-router)#network 30.0.0.0 mask 255.0.0.0
Router(config-router)#exit

```



Router2

Physical Config CLI Attributes

IOS Command Line Interface

```

Router(config-router)#%BGP-5-ADJCHANGE: neighbor 50.0.0.1 Up
Router(config-router)#neighbor 60.0.0.1 remote-as 100
Router(config-router)#%BGP-5-ADJCHANGE: neighbor 60.0.0.1 Up
Router(config-router)#exit
Router(config)#
Router(config)#interface FastEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/0/1
Router(config-if)#exit
Router(config)#router bgp 300
Router(config-router)#neighbor 50.0.0.1 remote-as 200
Router(config-router)#neighbor 60.0.0.1 remote-as 100
Router(config-router)#exit
Router(config)#router bgp 300
Router(config-router)#neighbor 50.0.0.1 remote-as 200
Router(config-router)#neighbor 60.0.0.1 remote-as 100
Router(config-router)#network 30.0.0.0 mask 255.0.0.0
Router(config-router)#exit
Router(config)#

```

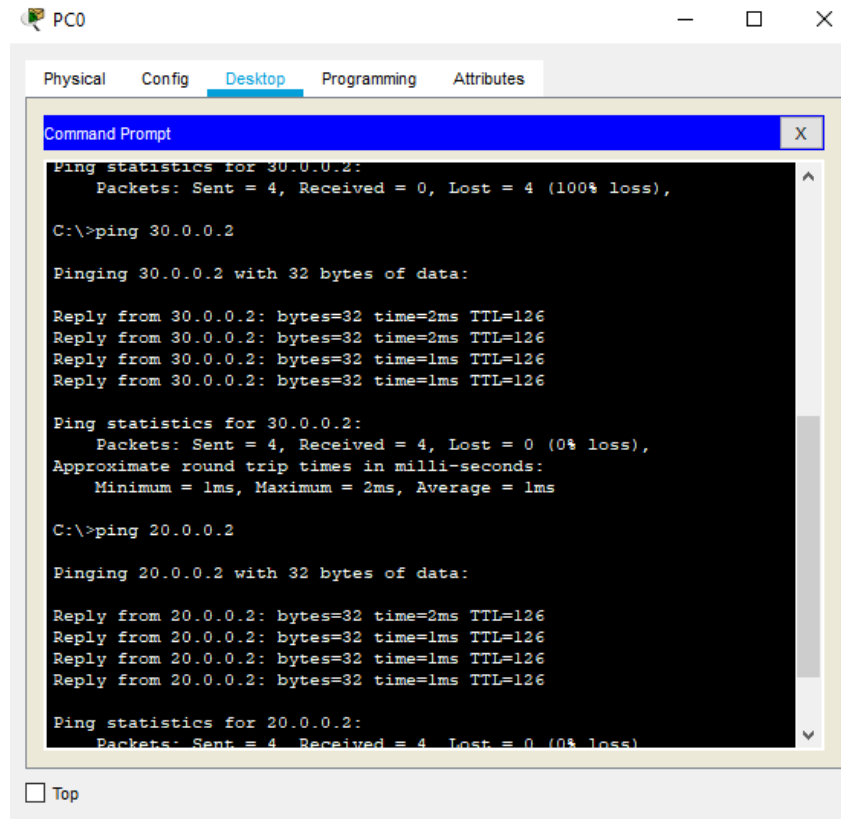
Ctrl+F6 to exit CLI focus

Copy Paste

☐ Top

Step 2 Ping from PC to check connection

Ping from PC-C to PC-B



The screenshot shows a Packet Tracer PC configuration window for PC0. The 'Desktop' tab is selected, displaying a Command Prompt window. The Command Prompt shows the results of two ping commands. The first command is 'ping 30.0.0.2', which initially shows 100% loss but then shows successful results after a second attempt. The second command is 'ping 20.0.0.2', which also shows successful results.

```
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Ping statistics for 30.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 30.0.0.2

Pinging 30.0.0.2 with 32 bytes of data:

Reply from 30.0.0.2: bytes=32 time=2ms TTL=126
Reply from 30.0.0.2: bytes=32 time=2ms TTL=126
Reply from 30.0.0.2: bytes=32 time=1ms TTL=126
Reply from 30.0.0.2: bytes=32 time=1ms TTL=126

Ping statistics for 30.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 2ms, Average = 1ms

C:\>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2: bytes=32 time=2ms TTL=126
Reply from 20.0.0.2: bytes=32 time=1ms TTL=126
Reply from 20.0.0.2: bytes=32 time=1ms TTL=126
Reply from 20.0.0.2: bytes=32 time=1ms TTL=126

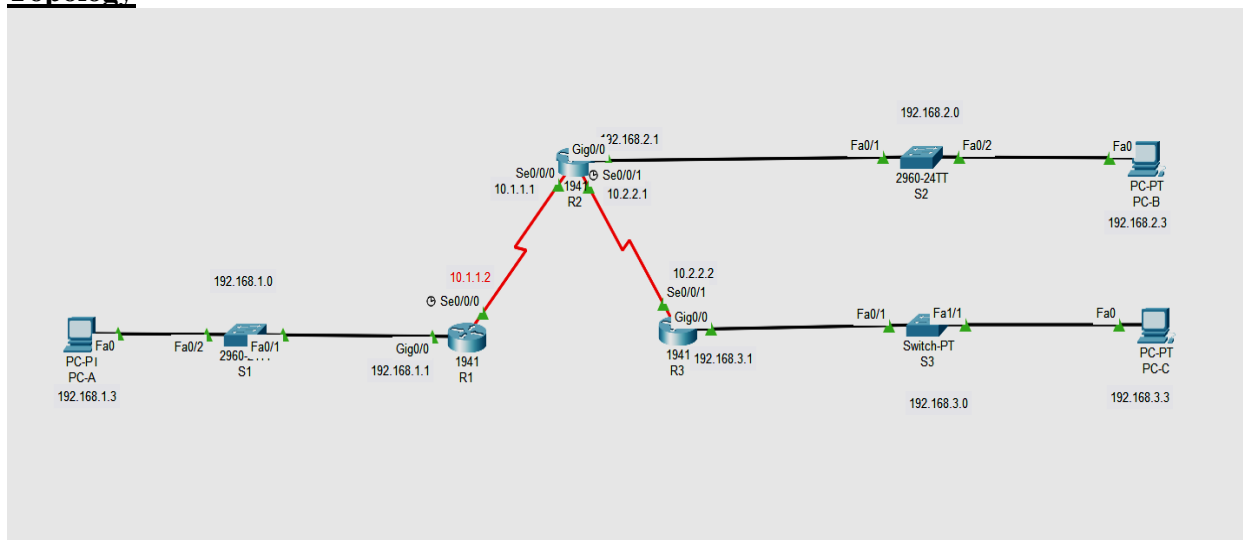
Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss)
```

☐ Top

Practical 10

Aim:- Implement IPsec Site-to-Site VPNs

Topology



Addressing Table

| Device | Interface | IP Address | Subnet Mask | Default Gateway | Switch Port |
|--------|--------------|-------------|-----------------|-----------------|-------------|
| R1 | G0/0 | 192.168.1.1 | 255.255.255.0 | N/A | S1 F0/1 |
| | S0/0/0 (DCE) | 10.1.1.2 | 255.255.255.252 | N/A | N/A |
| R2 | G0/0 | 192.168.2.1 | 255.255.255.0 | N/A | S2 F0/2 |
| | S0/0/0 | 10.1.1.1 | 255.255.255.252 | N/A | N/A |
| | S0/0/1 (DCE) | 10.2.2.1 | 255.255.255.252 | N/A | N/A |
| R3 | G0/0 | 192.168.3.1 | 255.255.255.0 | N/A | S3 F0/5 |
| | S0/0/1 | 10.2.2.2 | 255.255.255.252 | N/A | N/A |
| PC-A | NIC | 192.168.1.3 | 255.255.255.0 | 192.168.1.1 | S1 F0/2 |
| PC-B | NIC | 192.168.2.3 | 255.255.255.0 | 192.168.2.1 | S2 F0/1 |
| PC-C | NIC | 192.168.3.3 | 255.255.255.0 | 192.168.3.1 | S3 F0/18 |

Part 1 : Configure the Routers and pc according to Addressing Table

```

R1>enable
R1#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int G0/0
R1(config-if)#ip address 192.168.1.1 255.255.0.0
R1(config)#int S0/0/0
R1(config-if)#ip address 10.1.1.2 255.255.255.252
R1(config-if)#no shut
R1(config-if)#exit
R2>enable
R2#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#int G0/0
R2(config-if)#ip address 192.168.2.1 255.255.0.0
R2(config)#int S0/0/0
R2(config-if)#ip address 10.1.1.1 255.255.255.252
R2(config)#int S0/0/1
R2(config-if)#ip address 10.2.2.1 255.255.255.252
R2(config-if)#no shut
R2(config-if)#exit
R3>enable
R3#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#int G0/0
R3(config-if)#ip address 192.168.3.1 255.255.0.0
R3(config)#int S0/0/0
R3(config-if)#ip address 10.2.2.2 255.255.255.252
R3(config-if)#no shut
R3(config-if)#exit

```

NOW RIP ALL THE ROUTER

```

R1
Router(config)#router rip
Router(config-router)# network 10.0.0.0
Router(config-router)# network 192.168.0.0

R2
Router(config)#router rip
Router(config-router)# network 10.0.0.0
Router(config-router)# network 192.168.0.0

R3
Router(config)#router rip

```

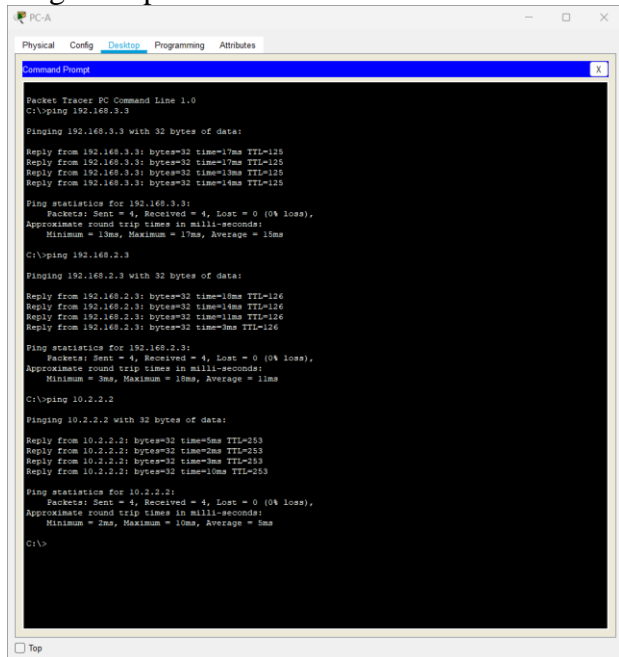
```
Router(config-router)# network 10.0.0.0
Router(config-router)# network 192.168.0.0
```

Part 2 Configure IPsec Parameters on R1

Step 1: Test connectivity.

Ping from PC-A to PC-C.

Ping from pcs



Step 2: Enable the Security Technology package.

```
R1(config)# license boot module c1900 technology-package securityk9
```

Step 3: Identify interesting traffic on R1.

```
R1(config)# access-list 110 permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255
```

Step 4 :Configure the IKE Phase 1 ISAKMP policy on R1.

```
R1(config)# crypto isakmp policy 10 R1(config-
isakmp)# encryption aes 256 R1(config-isakmp)#
authentication pre-share R1(config-isakmp)# group 5
R1(config-isakmp)# exit
```

```
R1(config)# crypto isakmp key vpnpa55 address 10.2.2.2
```

Step 5:Configure the IKE Phase 2 IPsec policy on R1.

Create the transform-set VPN-SET to use esp-aes and esp-sha-hmac.

```
R1(config)# crypto ipsec transform-set VPN-SET esp-aes esp-sha-hmac
```

Create the crypto map VPN-MAP that binds all of the Phase 2 parameters together. Use sequence number 10 and identify it as an ipsec-isakmp map.

```
R1(config)# crypto map VPN-MAP 10 ipsec-isakmp
R1(config-crypto-map)# description VPN connection to R3
R1(config-crypto-map)# set peer 10.2.2.2
R1(config-crypto-map)# set transform-set VPN-SET
R1(config-crypto-map)# match address 110 R1(config-
crypto-map)# exit
```

Step 6: Configure the crypto map on the outgoing interface.

Bind the **VPN-MAP** crypto map to the outgoing Serial 0/0/0 interface.

```
R1(config)# interface s0/0/0
R1(config-if)# crypto map VPN-MAP
```

```
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#access-list 110 permit ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255
Router(config)#crypto isakmp policy 10
Router(config-isakmp)#encryption aes 256
Router(config-isakmp)#authentication pre-share
Router(config-isakmp)#group 5
Router(config-isakmp)#exit
Router(config)#crypto isakmp key vpnpa55 address 10.1.1.2
Router(config)#
Router(config)#crypto ipsec transform-set VPN_SET esp-aes esp-sha-hmac
Router(config)#crypto map VPN-MAP 10 ipsec-isakmp
Router(config-crypto-map)#description VPN connection to R1
Router(config-crypto-map)#set peer 10.1.1.2
Router(config-crypto-map)#set transform-set VPN-SET
ERROR: transform set with tag VPN-SET does not exist.
Router(config-crypto-map)#exit
Router(config)#crypto ipsec transform-set VPN-SET esp-aes esp-sha-hmac
Router(config)#crypto map VPN-MAP 10 ipsec-isakmp
Router(config-crypto-map)#description VPN connection to R1
Router(config-crypto-map)#set peer 10.1.1.2
Router(config-crypto-map)#set transform-set VPN-SET
Router(config-crypto-map)#match address 110
Router(config-crypto-map)#exit
Router(config)#int s0/0/1
Router(config-if)#crypto map VPN-MAP
*Jan 3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
Router(config-if)#
```

```

Router#show crypto ipsec sa
interface: Serial0/0/0
  Crypto map tag: VPN-map, local addr 10.1.1.2

  protected vrf: (none)
  local ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
  remote ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
  current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 0
    #pkts decaps: 3, #pkts decrypt: 3, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 1, #recv errors 0

    local crypto endpt.: 10.1.1.2, remote crypto endpt.:10.2.2.2
    path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
    current outbound spi: 0x854C9428(2236388392)

  inbound esp sas:
    spi: 0xED35493B(3979692347)

Router#show crypto ipsec sa
interface: Serial0/0/0
  Crypto map tag: VPN-map, local addr 10.1.1.2

  protected vrf: (none)
  local ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
  remote ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
  current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 0
    #pkts decaps: 3, #pkts decrypt: 3, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 1, #recv errors 0

    local crypto endpt.: 10.1.1.2, remote crypto endpt.:10.2.2.2
    path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
    current outbound spi: 0x854C9428(2236388392)

  inbound esp sas:
    spi: 0xED35493B(3979692347)
--More--

```

```

spi: 0xED35493B(3979692347)

outer#show crypto ipsec sa
nterface: Serial0/0/0
  Crypto map tag: VPN-map, local addr 10.1.1.2

  protected vrf: (none)
  local ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
  remote ident (addr/mask/prot/port): (192.168.3.0/255.255.255.0/0/0)
  current_peer 10.2.2.2 port 500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 0
    #pkts decaps: 3, #pkts decrypt: 3, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 1, #recv errors 0

    local crypto endpt.: 10.1.1.2, remote crypto endpt.:10.2.2.2
    path mtu 1500, ip mtu 1500, ip mtu idb Serial0/0/0
    current outbound spi: 0x854C9428(2236388392)

  inbound esp sas:
    spi: 0xED35493B(3979692347)
--More--

```