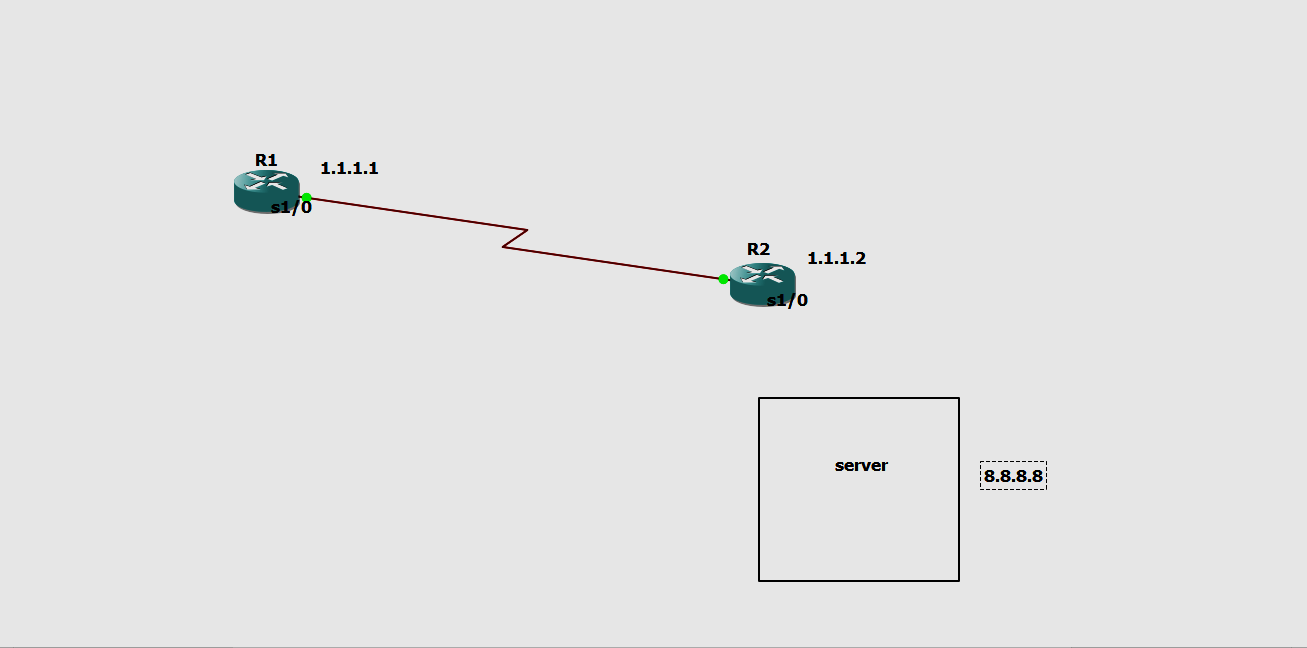
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |
|  |
|  |  |  |  |  |
|  |
|  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |
|  |  |  |  |  |
|  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Practical 1**

Aim:- Implement IP SLA (IP Service Level Agreement)

Topology:



Addressing Table:

| **Device** | **Interface** | **IP Address** | **Subnet Mask** | **Default Gateway** |
| --- | --- | --- | --- | --- |
| R1 | S0/0/0 | 1.1.1.1 | 255.0.0.0 | N/A |
| ISP | S0/0/0 | 1.1.1.2 | 255..0.0.0 | N/A |
| Lo0 | 8.8.8.8 | 255..0.0.0 | N/A |

**Part 1: Build the Network and Verify Connectivity**

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

**Step 2**: **Configure basic settings for R1**

**Code in R1**

R1>enable

R1#config terminal

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

R1(config)#int s1/0

R1(config-if)#ip address 1.1.1.1 255.0.0.0

R1(config-if)#no shut

R1(config-if)#ip route 0.0.0.0 0.0.0.0 1.1.1.2

R1(config)#exit

**Step 3: Configure basic settings for R2(ISP ROUTER)**

**Code in R2**

R2>enable

R2#config terminal

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

R2(config)#int s1/0

R2(config-if)#ip address 1.1.1.2 255.0.0.0

R2(config-if)#clock rate 64000

R2(config-if)#no shut

R2(config)#exit

**Step 4: IN R2 TO CONFIGURE THE ISP ROUTER**

R2(config)#int loopback 1

R2(config)#int s1/0

R2(config-if)#no ip domain-lookup

R2(config-if)#ip address 8.8.8.8 255.0.0.0

R2(config-if)#no shut

R2(config)#exit

**Step 5: Ping 8.8.8.8 and 1.1.1.2 from R1 AND R2 To check whether the connection is proper**

**PING FROM R1**

R1#ping 8.8.8.8

Type escape sequence to abort

Sending 5, 100-bytes ICMP Echo to 8.8.8.8, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/30/40 ms

R1#ping 1.1.1.2

Type escape sequence to abort

Sending 5, 100-bytes ICMP Echo to 8.8.8.8, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/30/40 ms

**PINGING FROM R2**

R2#ping 8.8.8.8

Type escape sequence to abort

Sending 5, 100-bytes ICMP Echo to 8.8.8.8, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/30/40 ms

R2#ping 1.1.1.2

Type escape sequence to abort

Sending 5, 100-bytes ICMP Echo to 8.8.8.8, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/30/40 ms

**PART 2: Configure IP SLA ICMP Echo on R1**

**Code in R1**

R1#config terminal

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

R1(config)#ip sla 1

R1(config-ip-sla)#icmp-echo 8.8.8.8

R1(config-ip-sla-echo)#frequency 10

R1(config-ip-sla-echo)#ip sla schedule 1 start-time now life forever

R1(config)#end

**PART 3: Monitor the IP SLA Operation to check the output**

**Issue the command used to display the IP SLA operation statistics on R1.**

R1# **show ip sla statistics**

IPSLAs Latest Operation Statistics

IPSLA operation id: 22

Latest RTT: 1 milliseconds

Latest operation start time: 18:44:45 UTC Thu Sep 22 2022

Latest operation return code: OK

Number of successes: 103

Number of failures: 10

Operation time to live: Forever

R1# **show ip sla configuration**

IP SLA Infrastructure Engine-III

Entry number: 1

Owner:

Tag:

Operation timeout (milliseconds): 5000

Type of operation to perform: icmp-echo

Target address/Source address :8.8.8.8/0.0.0.0

Type of service parameter : 0x0

Request size (ARR data portion):28

Verify data: No

Vrf Name:

Schedule:

Operation frequency (seconds): 10 (not considered if randomly scheduled)

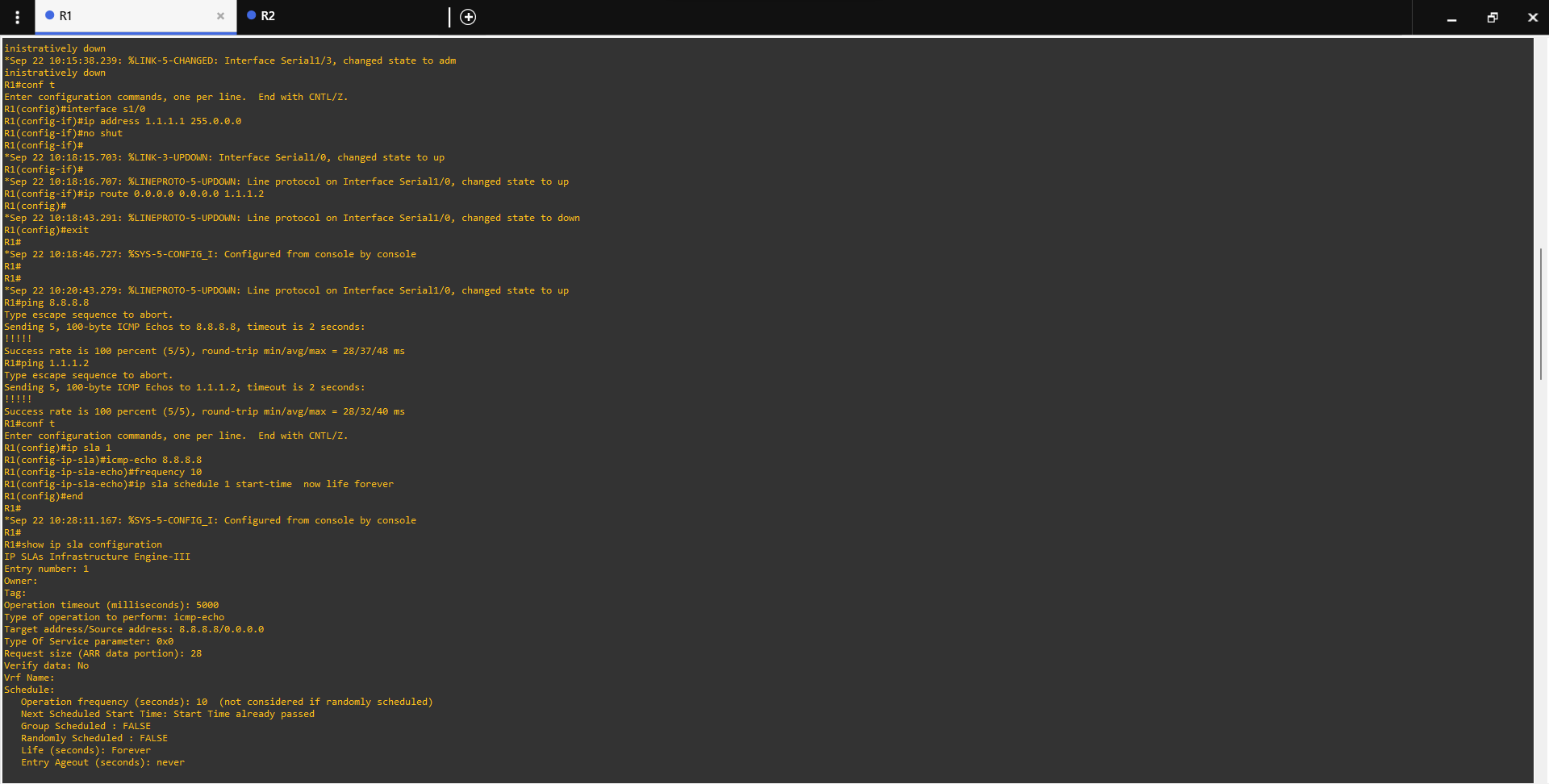
Next Scheduled start time: Start Time already passed

Group Scheduled: FALSE  
Randomly scheduled: FALSE

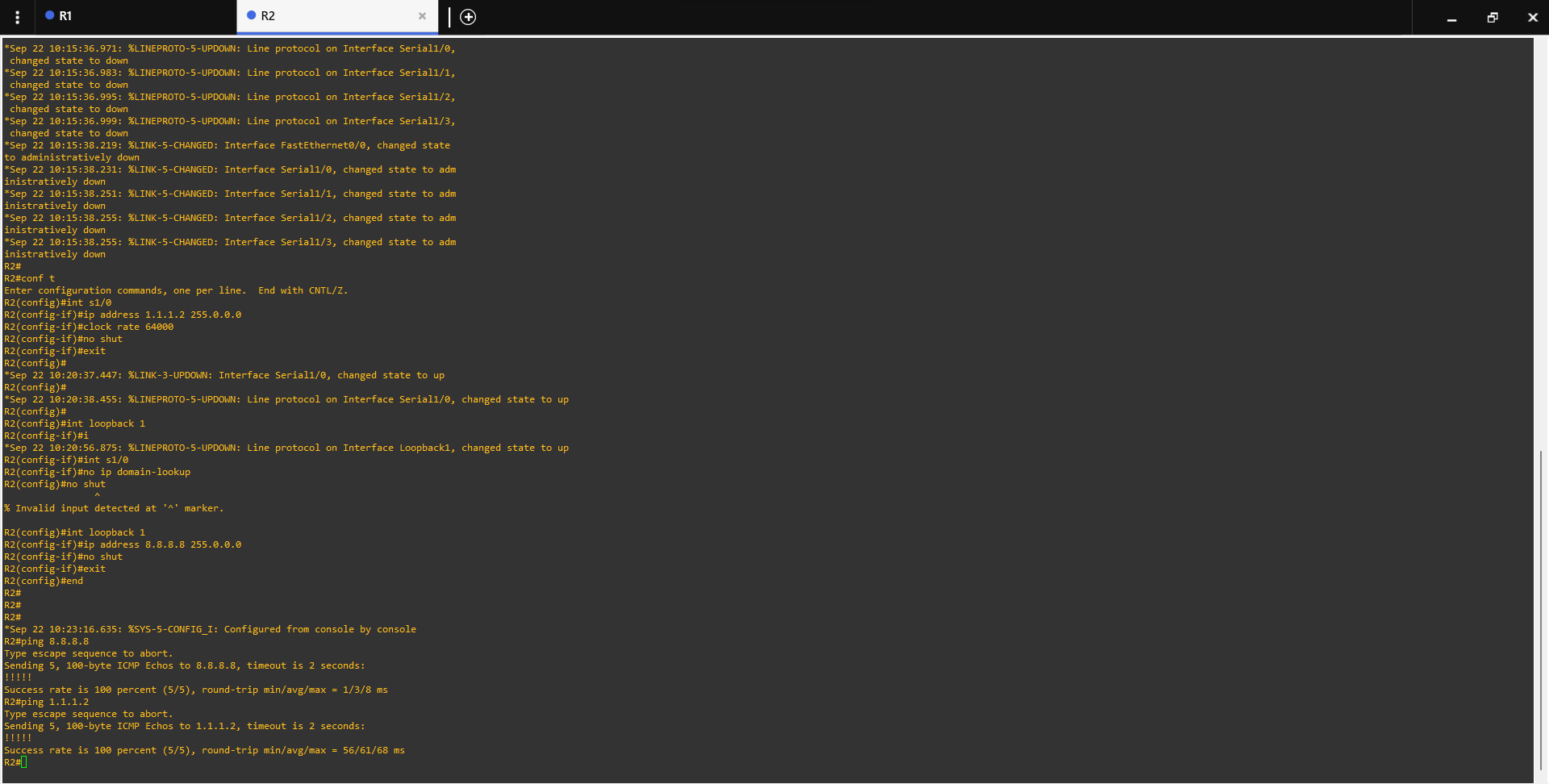
Life (seconds): FALSE

Entry Agent (seconds): never

**Output R1:**



**Output R2:**



**Practical 2**

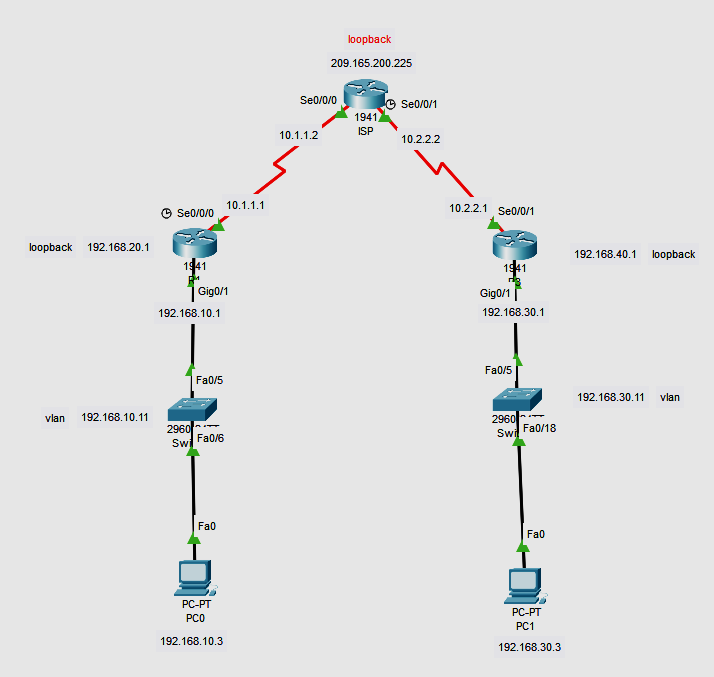
Implement IPv4 ACLs

1.Standard

2.Extended

**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 |

1. **Set Up the Topology and Initialize Devices**
   1. **Cable the network as shown in the topology.**
   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 unreachables 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.

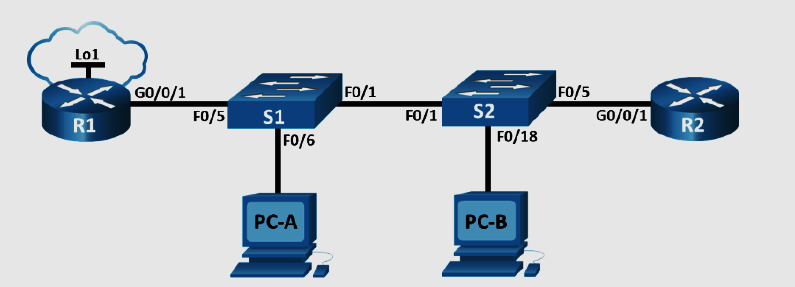
Sending 5, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:

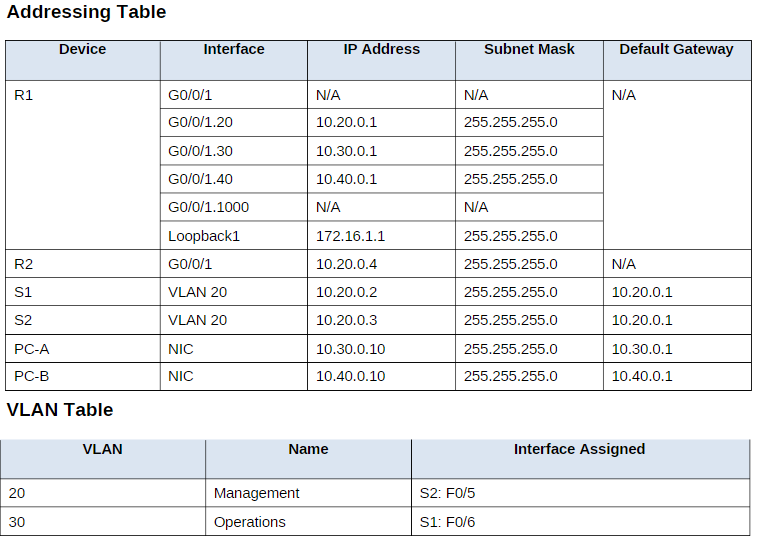
!!!!!

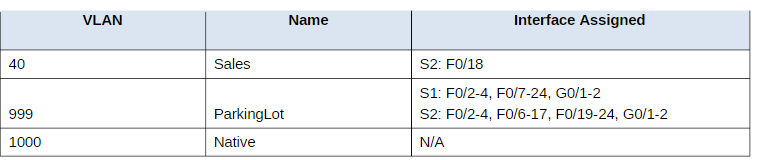
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/16/23 ms

**2.Extended**

**Topology:**







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

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

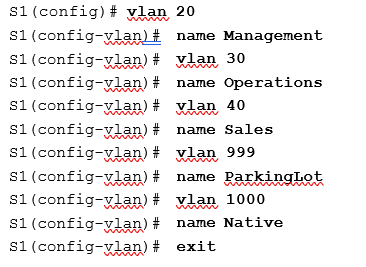
Attach the devices as shown in the topology diagram, and cable as necessary.

#### 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**

****

**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

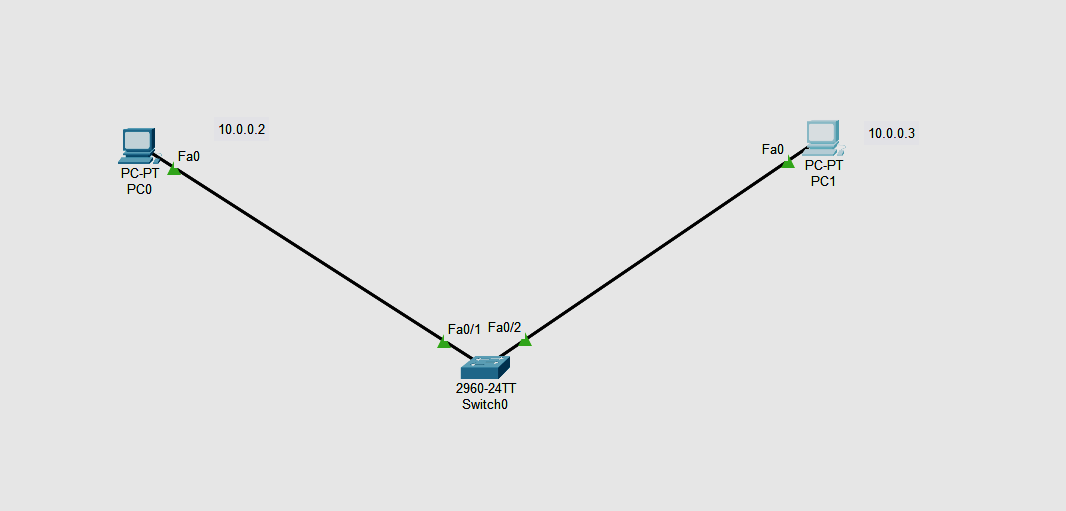
| PC-A | Ping | 10.40.0.10 | Fail |
| --- | --- | --- | --- |
| PC-A | Ping | 10.20.0.1 | Success |
| PC-B | Ping | 10.30.0.10 | Fail |
| PC-B | Ping | 10.20.0.1 | Fail |
| PC-B | Ping | 172.16.1.1 | Success |
| PC-B | HTTPS | 10.20.0.1 | Fail |
| **PC-B** | **HTTPS** | **172.16.1.1** | **Success** |

Practical 3

Implement span technologies (switch port analyzer)

3A Implement span technologies (switch port analyzer)

Topology:



1. **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:

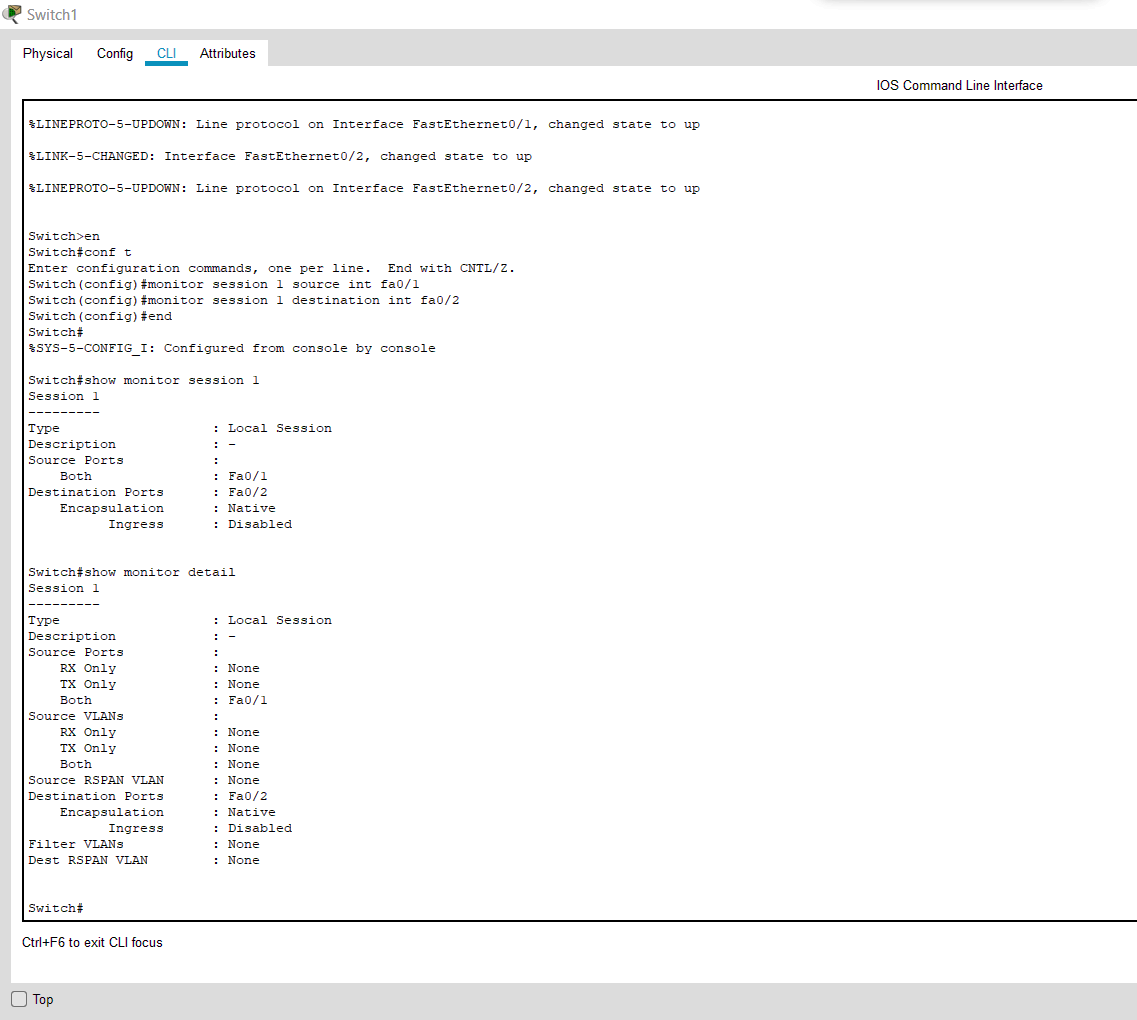
Switch>enable

Switch#config terminal

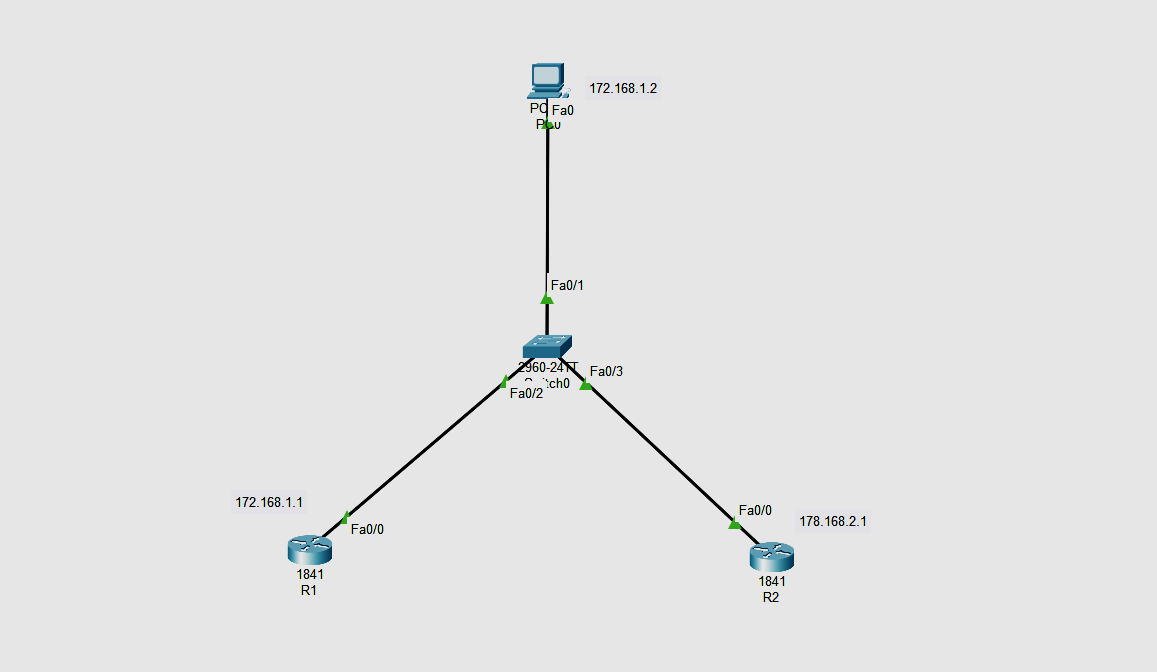
Switch(config)# moniter session 1 source int fa0/1

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

Switch(config)#end



**3B 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

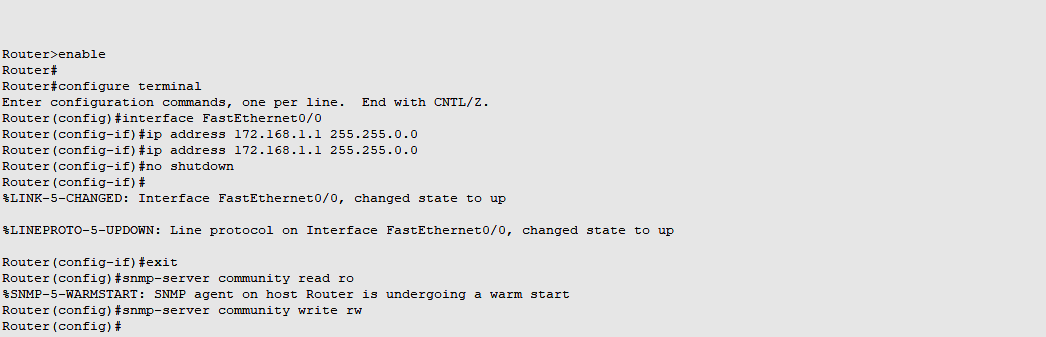
R1(config-if)#ip address 172.168.1.1 255.255.0.0

R1(config-if)#no shut

R1(config-if)#exit

R1(config)#snmp-server community read ro

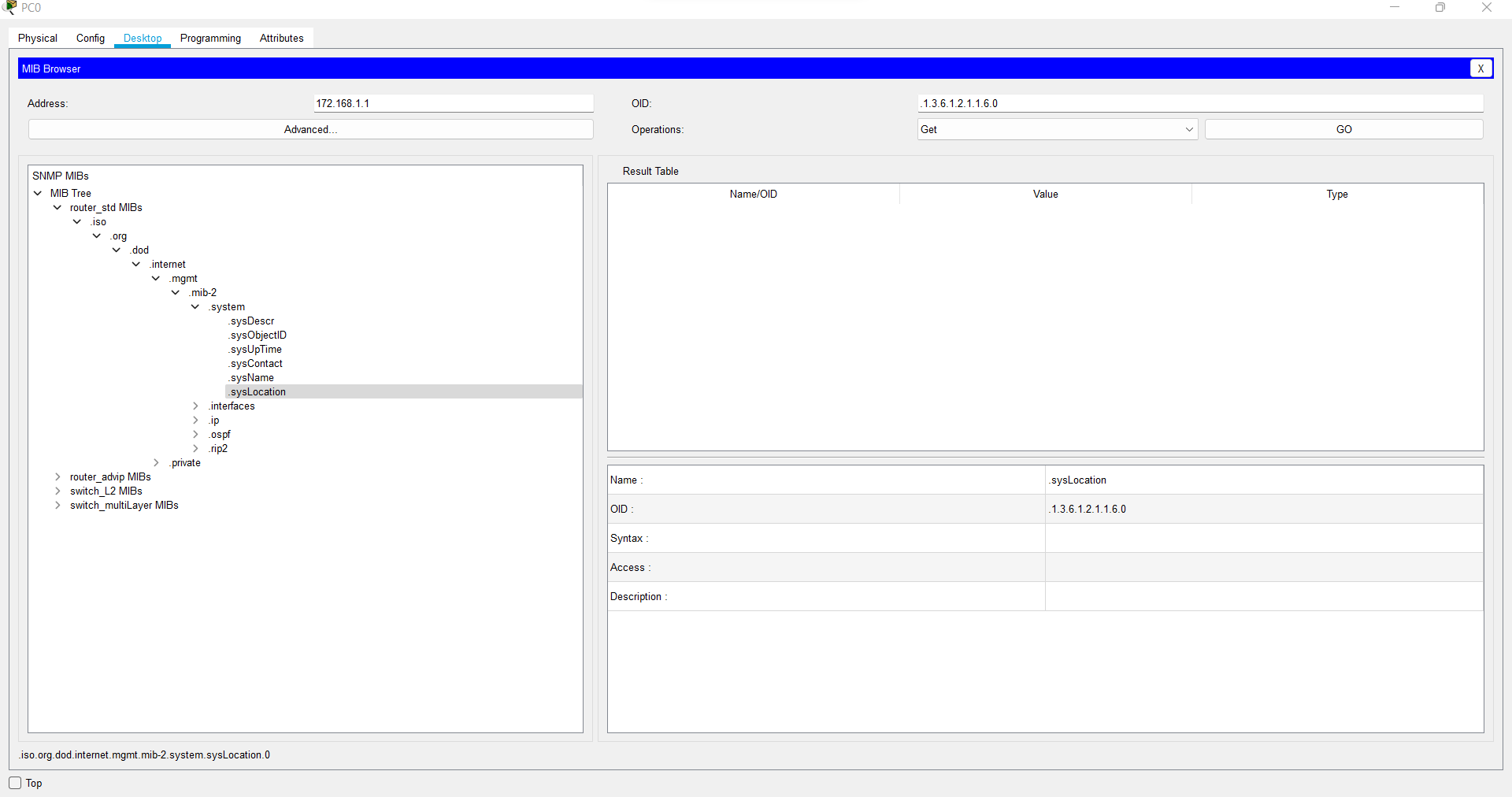
R1(config)#snmp-server community write rw

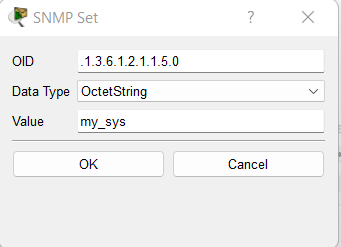


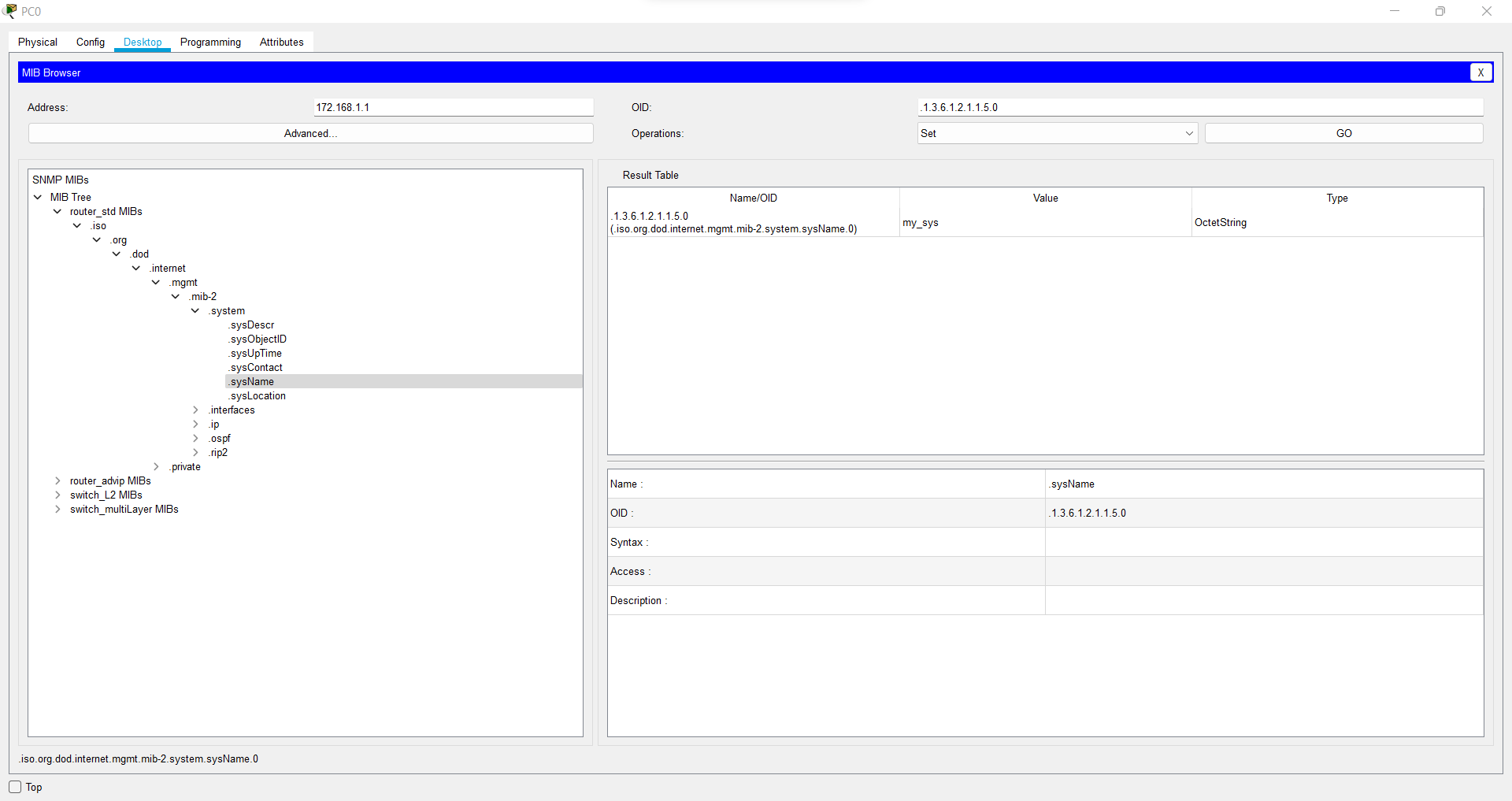
Step 2

Go to pc o then

Mib browser in desktop

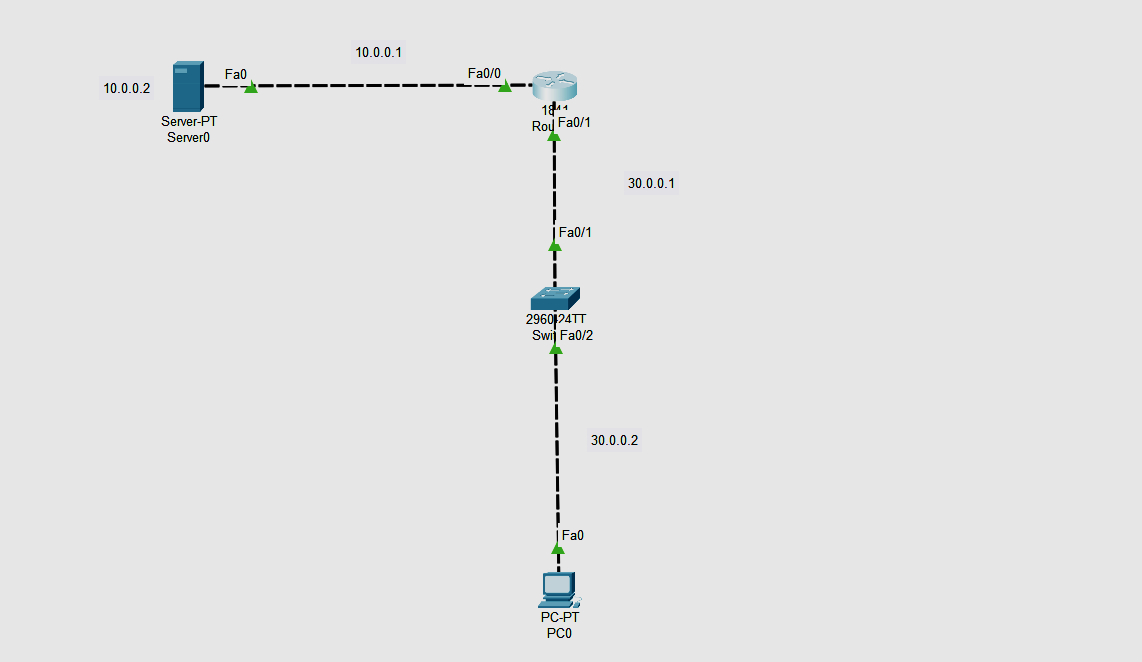






**b)Syslog**

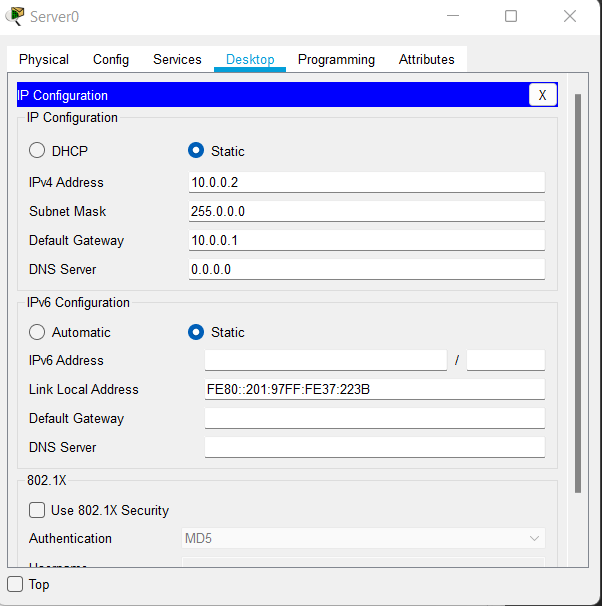
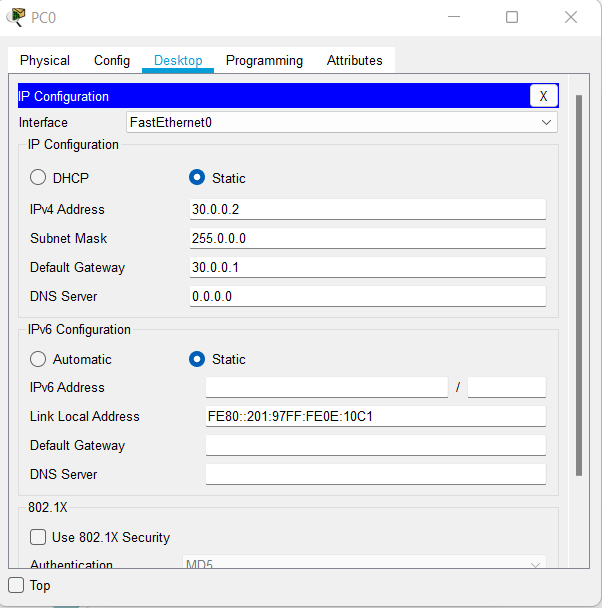
**Topology**



Addressing Table

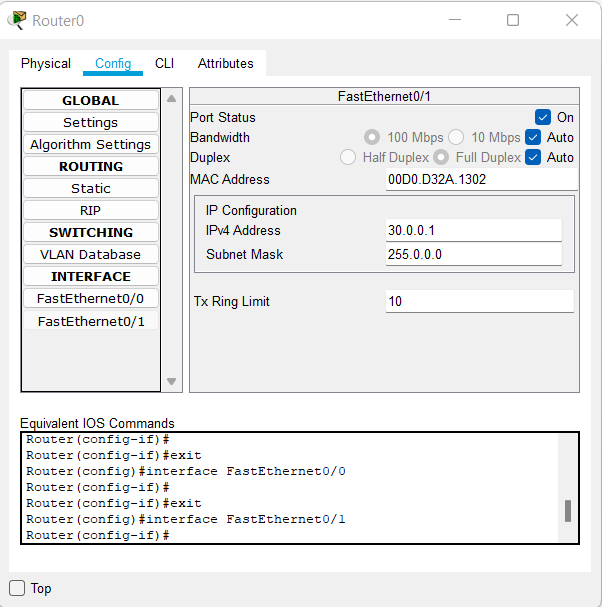
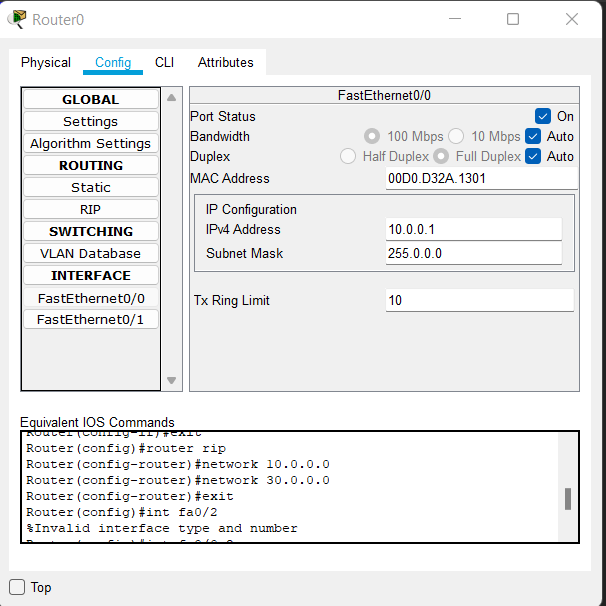
| **Device** | **Interface** | **IP Address** | **Subnet Mask** | **Default Gateway** |
| --- | --- | --- | --- | --- |
| R1 | Fa0/0 | 10.0.0.1 | 255.0.0.0 | N/A |
| Fa0/1 | 30.0.0.1 | 255.0.0.0 | N/A |
| PC0 | Fa0 | 30.0.0.2 | 255.0.0.0 | 30.0.0.1 |
| Switch0 | Fa0/1 | N/A | 255..0.0.0 | N/A |
| Fa0/2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

Configure Pc0

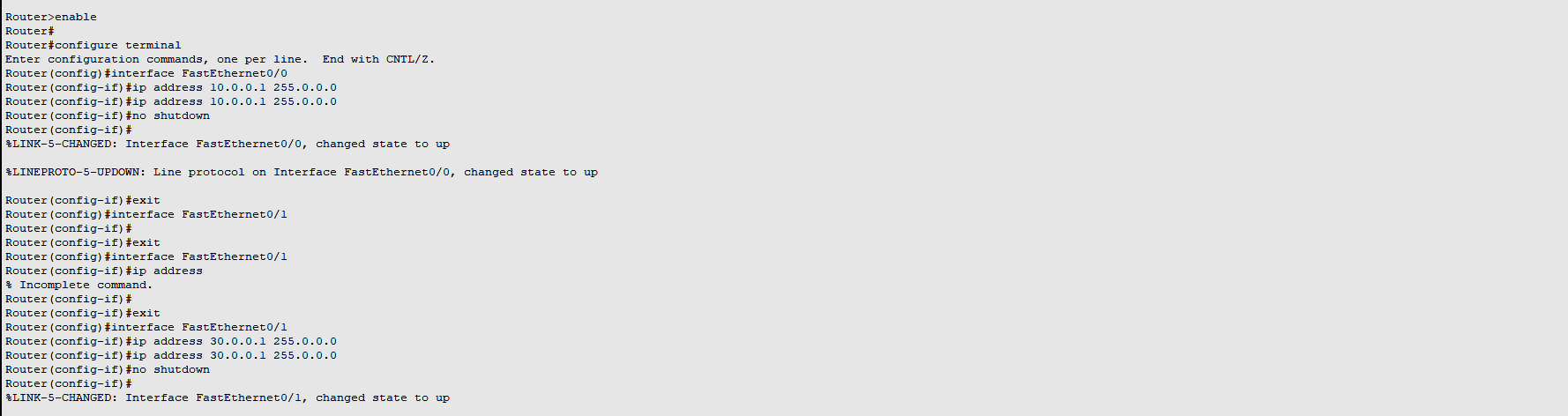


Go to r0 and configure network

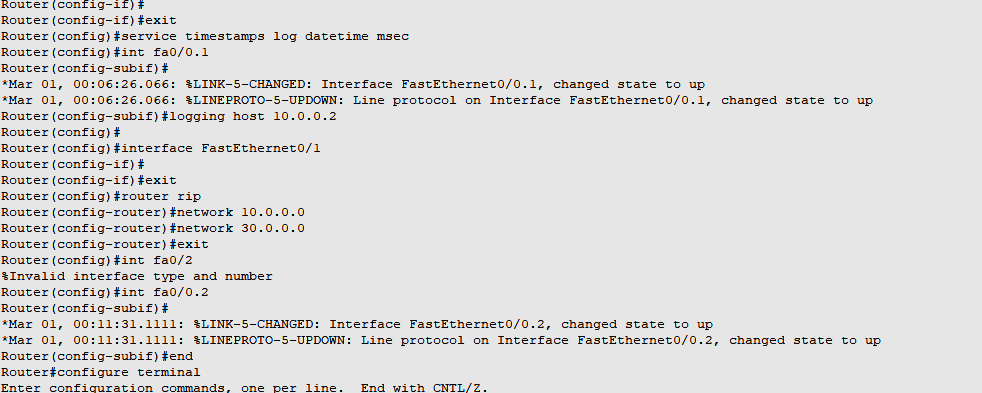
Fa0/0



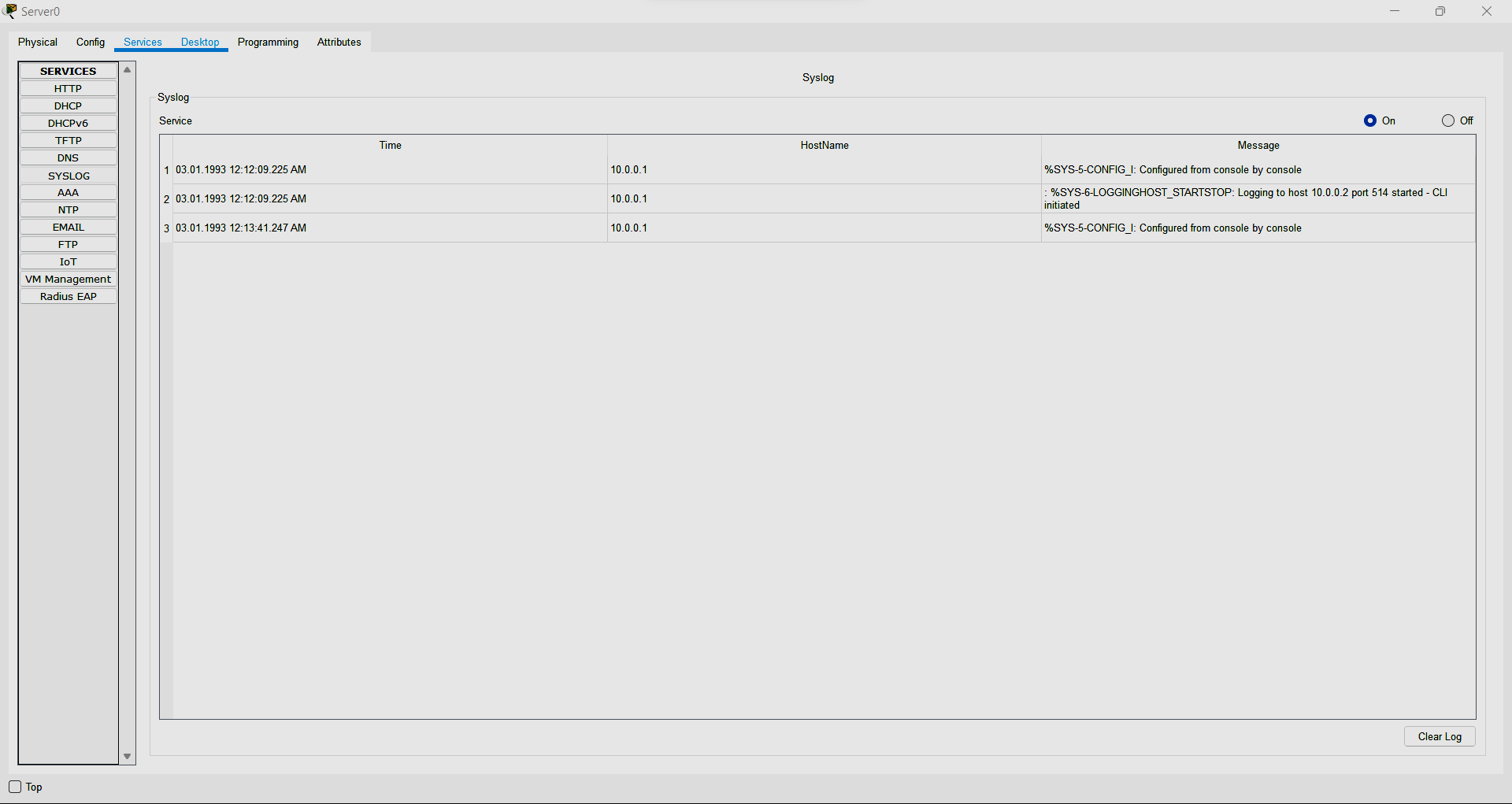
Fa0/1



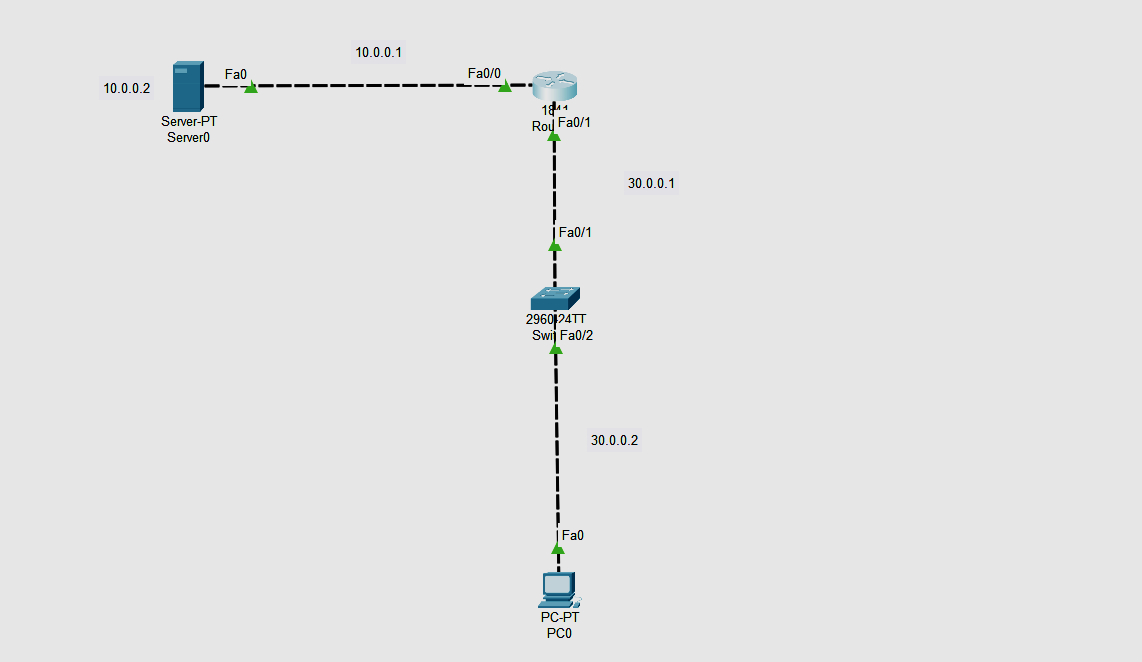
Go to r1 again and at cli put the following command



Output at server services and you will find syslog



**3.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

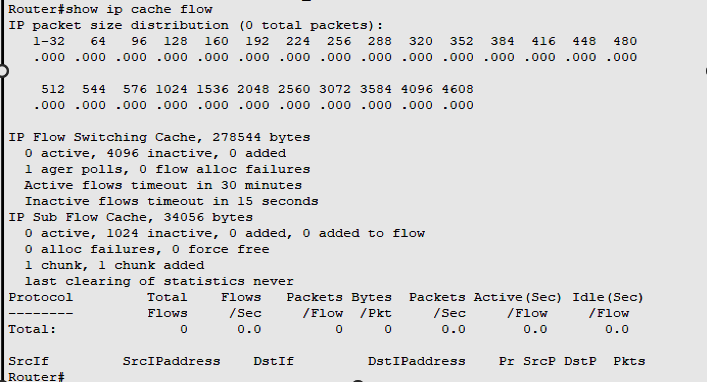
R1(config-if)#ip flow ingress

R1(config-if)#ip flow engress

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

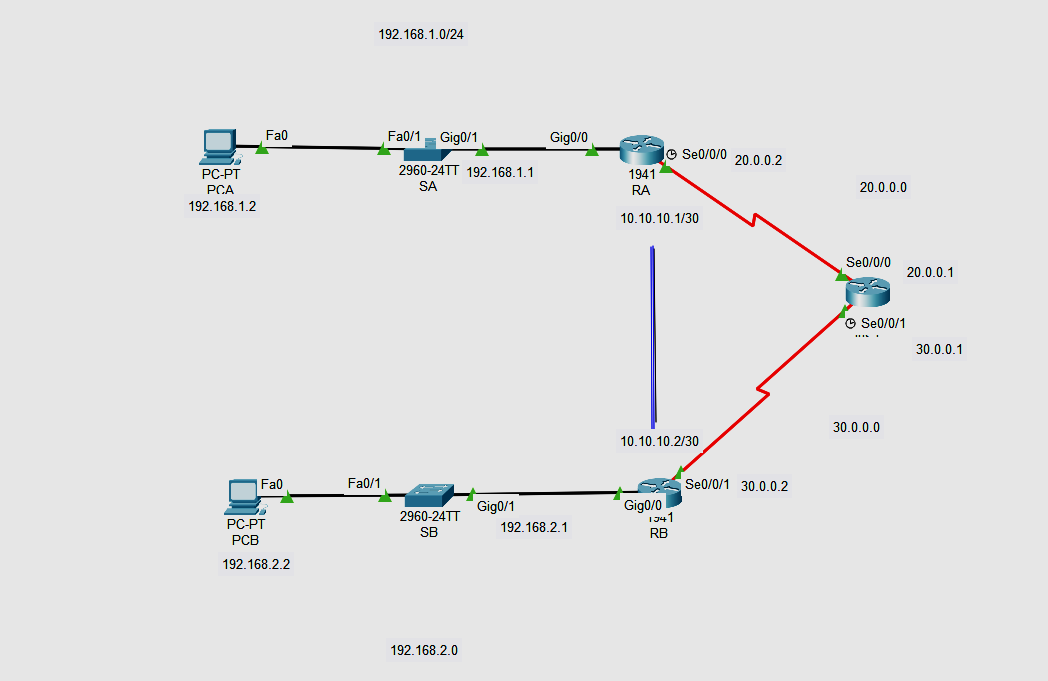
R1(config-if)#end

R1#show ip cache flow



Practical 4

1.IMPLEMENT A GRE TUNNEL

Topology: 

**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

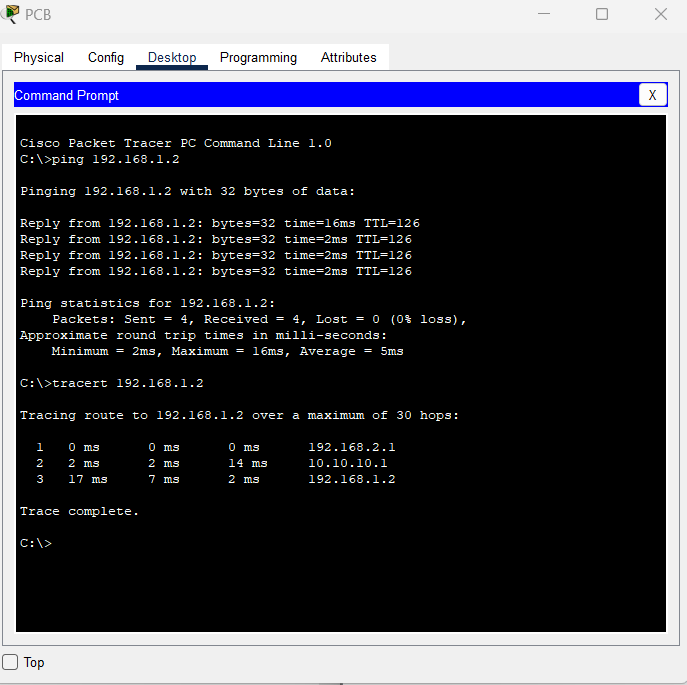
**Part** **3:** **Verify** **Router** **Connectivity**

**Step** **1:** **Ping** **PCA** **from** **PCB.**

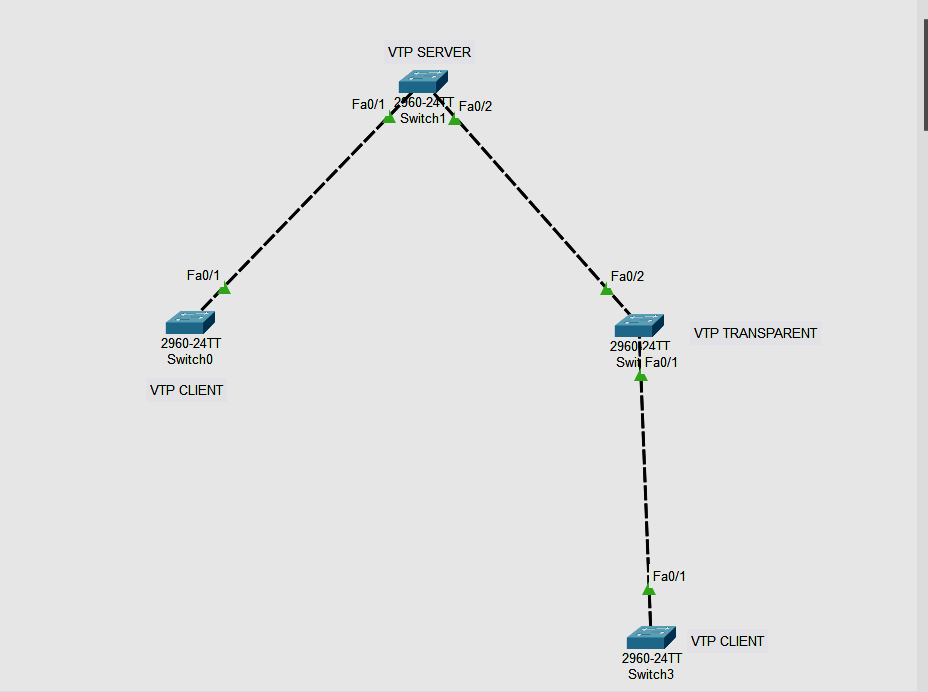
Attempt to ping the IP address of **PCA** from **PCB**. The ping should be successful.

**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.



**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**

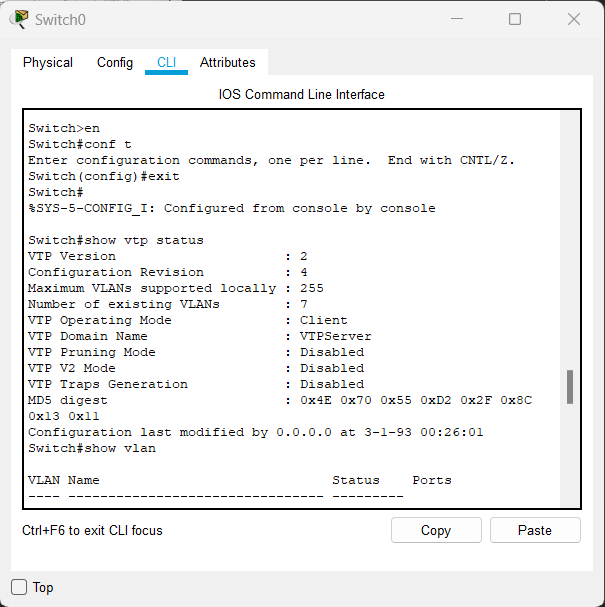
Switch(config)#vtp mode client

Setting device to VTP CLIENT mode.

Switch(config)#

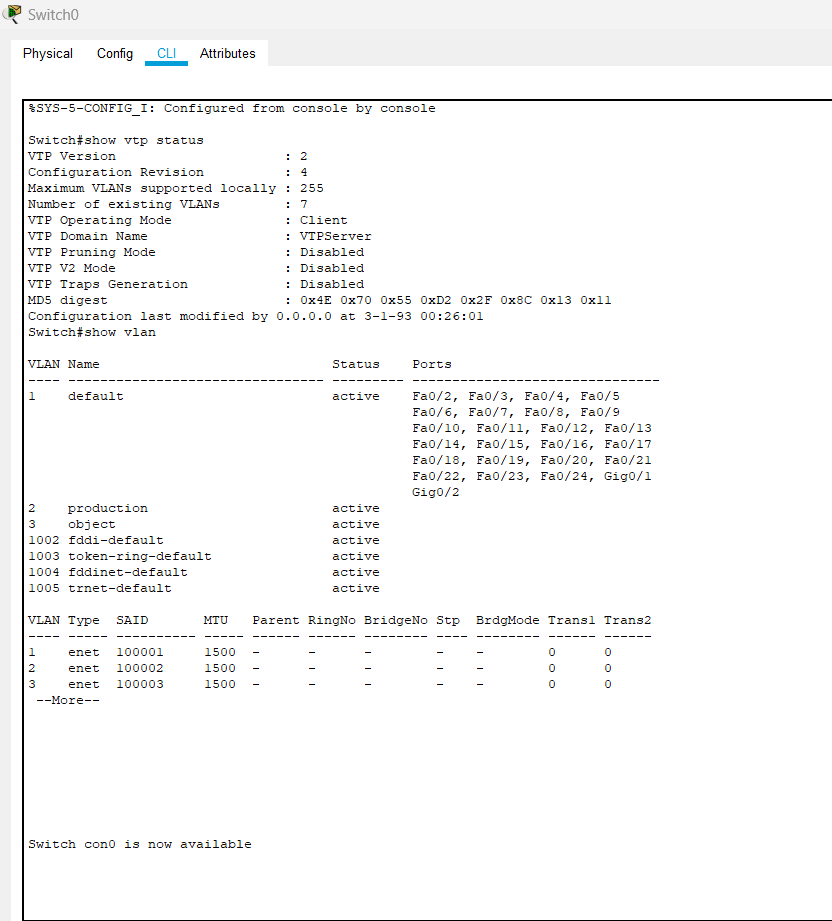
**No need to change the mode of switch 1 since by default mode is server**

**Output to check status of each switch**



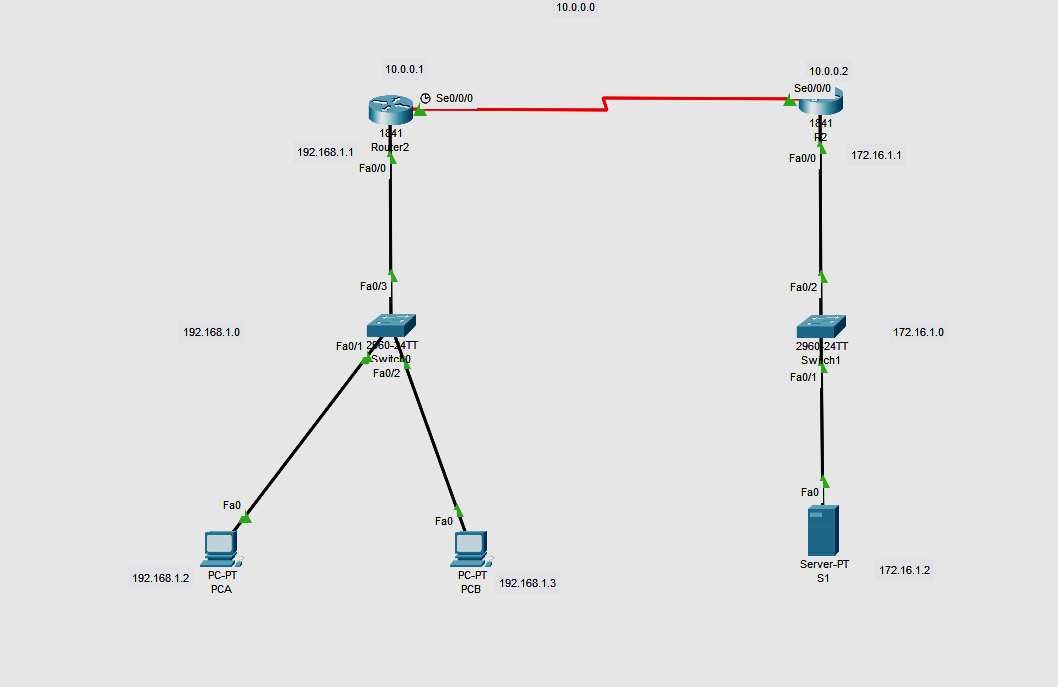
**Check the vtp status**

Switch#Show vtp status



**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 | 172.168.1.1 | 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 statu**s

Router#show ip nat ?

statistics Translation statistics

translations Translation entries

Router#show ip nat statistics

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

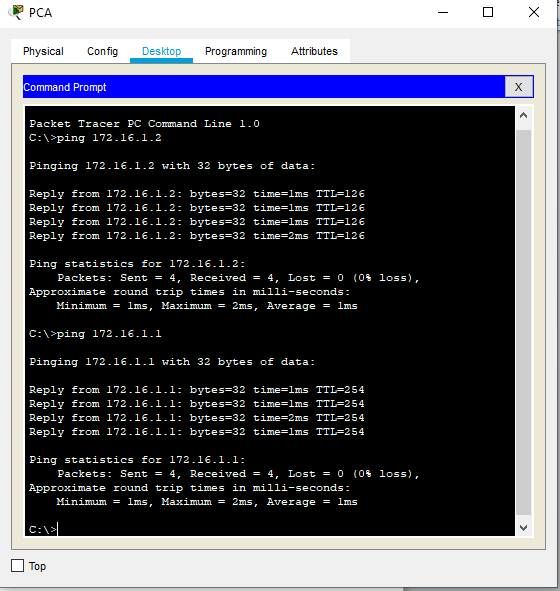
Outside Interfaces: Serial0/0/0

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

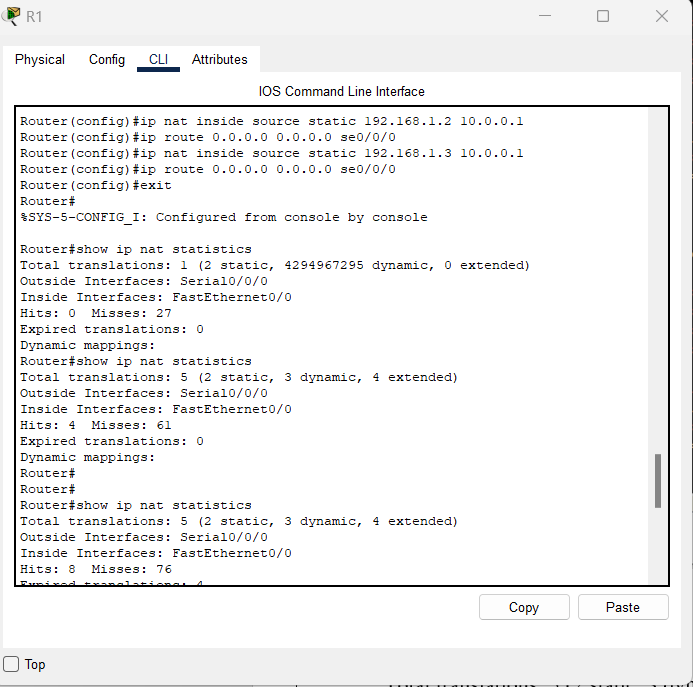
Inside Interfaces: FastEthernet0/0

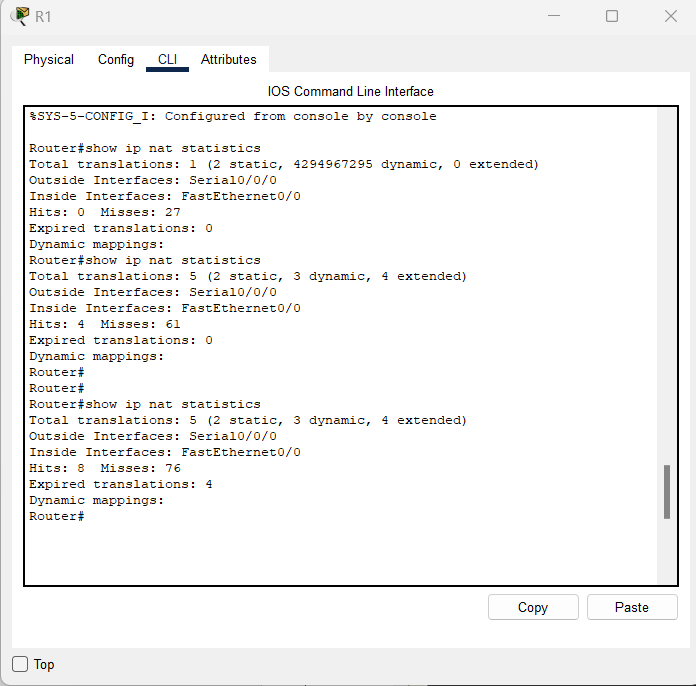
Hits: 8 Misses: 30

Expired translations: 0

Dynamic mappings:

Router#





**Practical 5**

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.

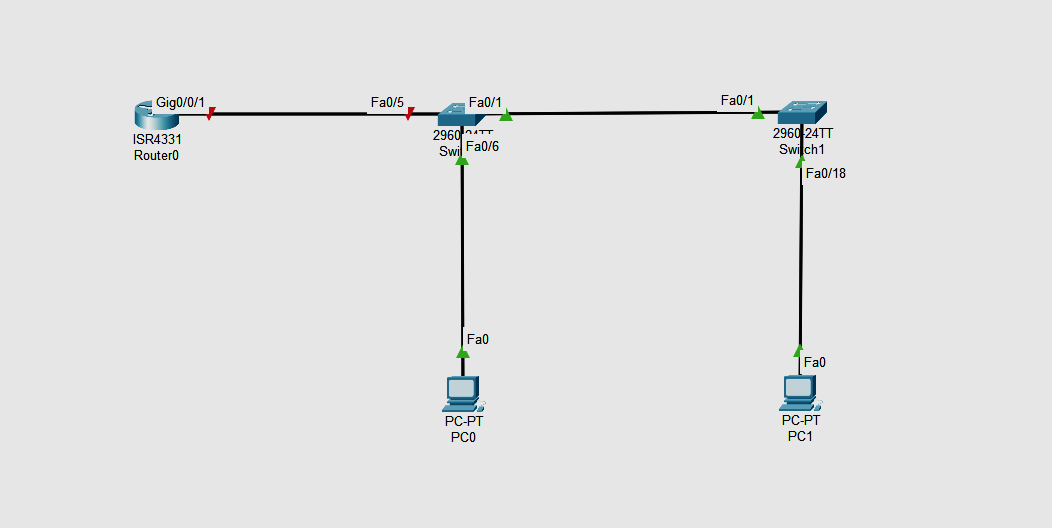
# Addressing Table

| **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 |

**Topology**



**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! $**

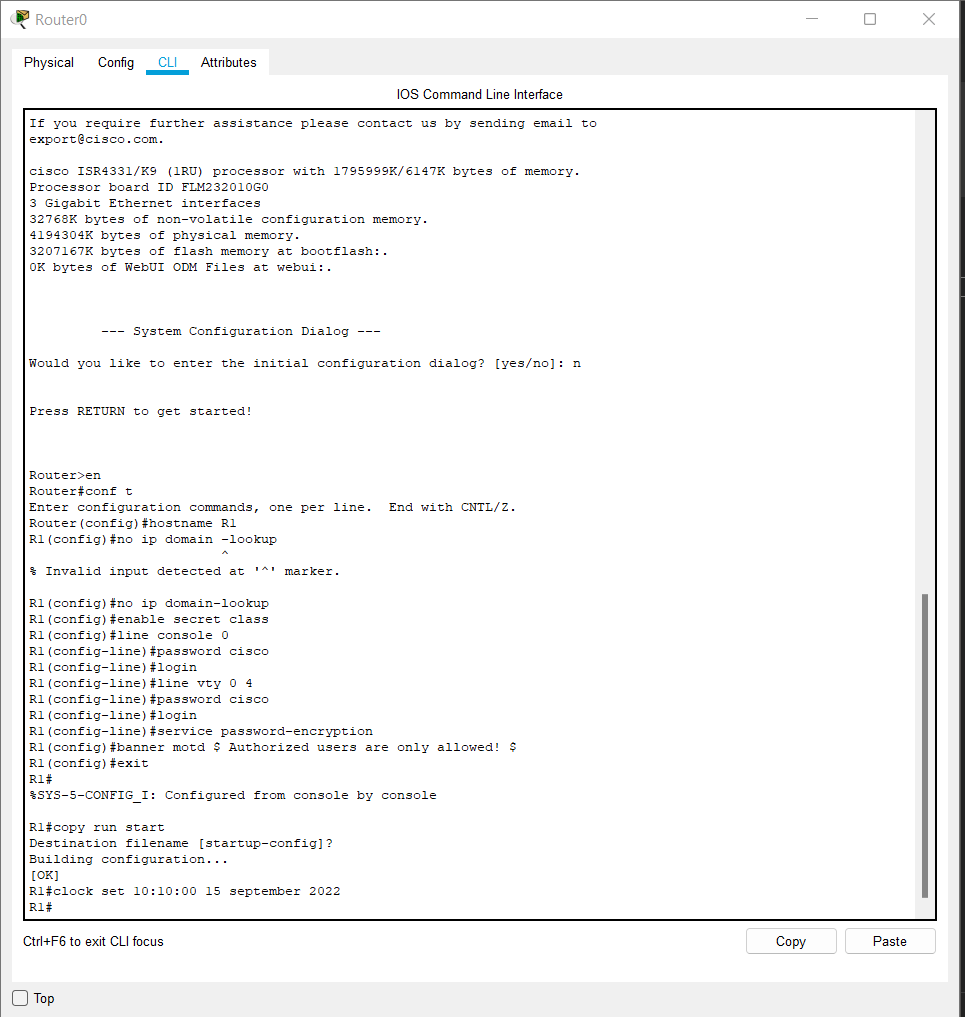
**Save the running configuration to the startup configuration file.**

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**

1. 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**

1. Assign **class** as the privileged EXEC encrypted password.

S1(config)# **enable secret class**

S2(config)# **enable secret class**

1. 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**

1. 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**

1. Encrypt the plaintext passwords.

S1(config)# **service password-encryption**

S2(config)# **service password-encryption**

1. 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**

1. Set the clock on the switch.

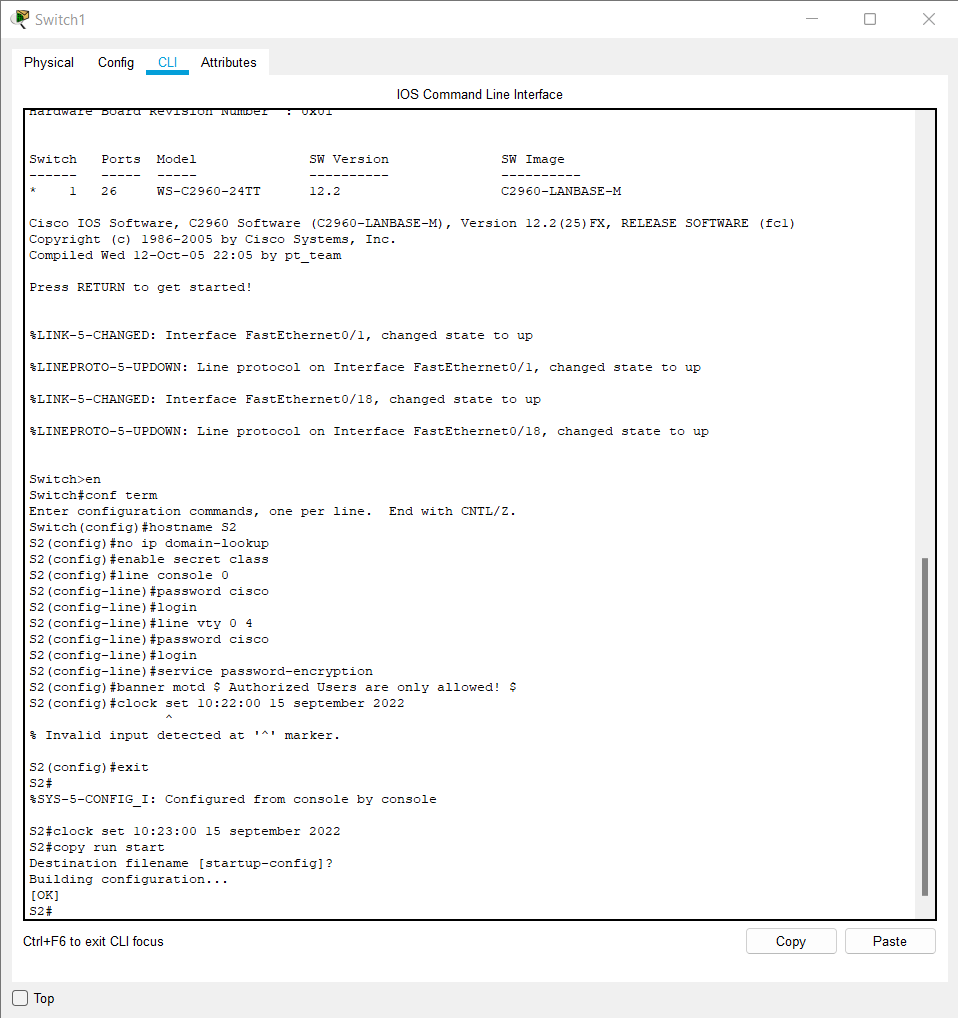
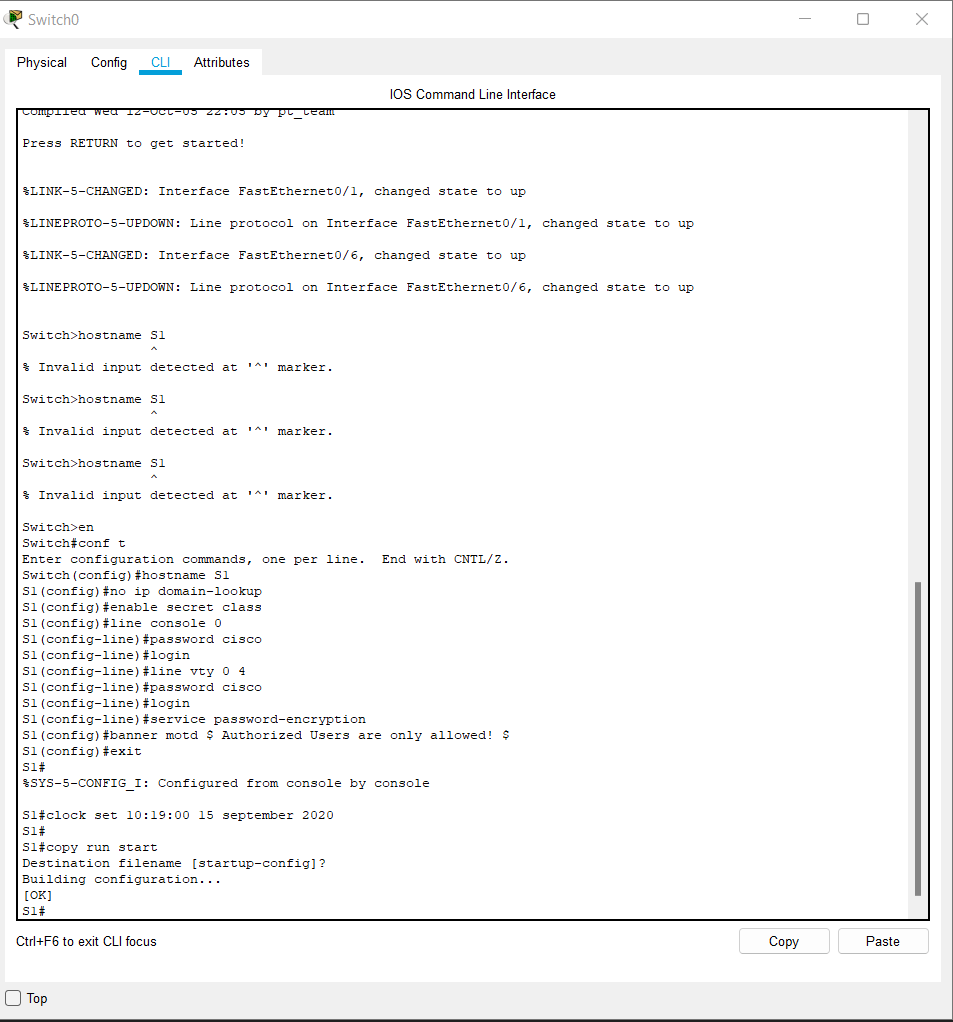
S1# **clock set 15:30:00 27 Aug 2019**

S2# **clock set 15:30:00 27 Aug 2019**

1. Save the running configuration to the startup configuration.

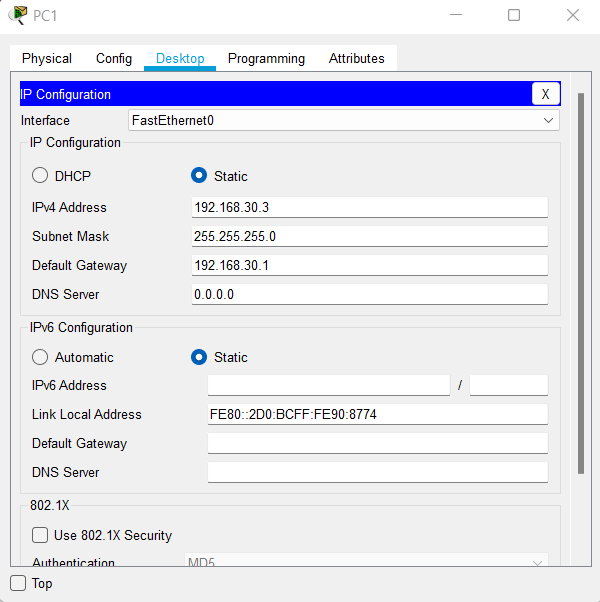
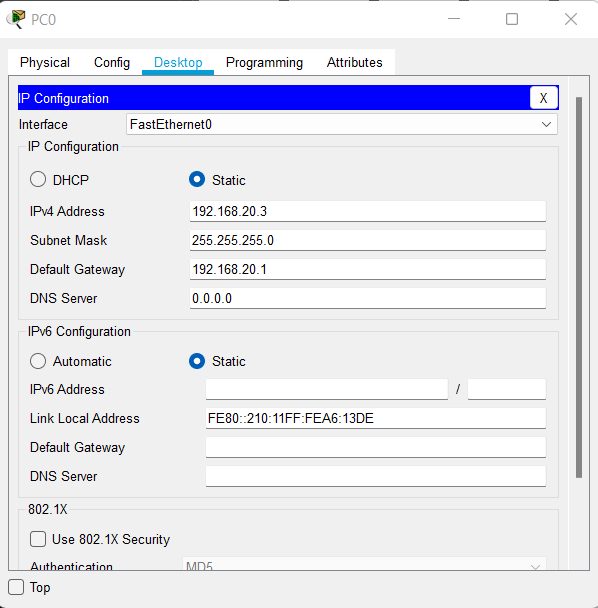
S1# **copy running-config startup-config**

S2# **copy running-config startup-config**



**Step 4:  Configure PC hosts.**

Refer to the Addressing Table for PC host address information.



**Part 2:   Create VLANs and Assign Switch Ports**

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.

**Step 1:  Create VLANs on both switches.**

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

*Open configuration window*

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**

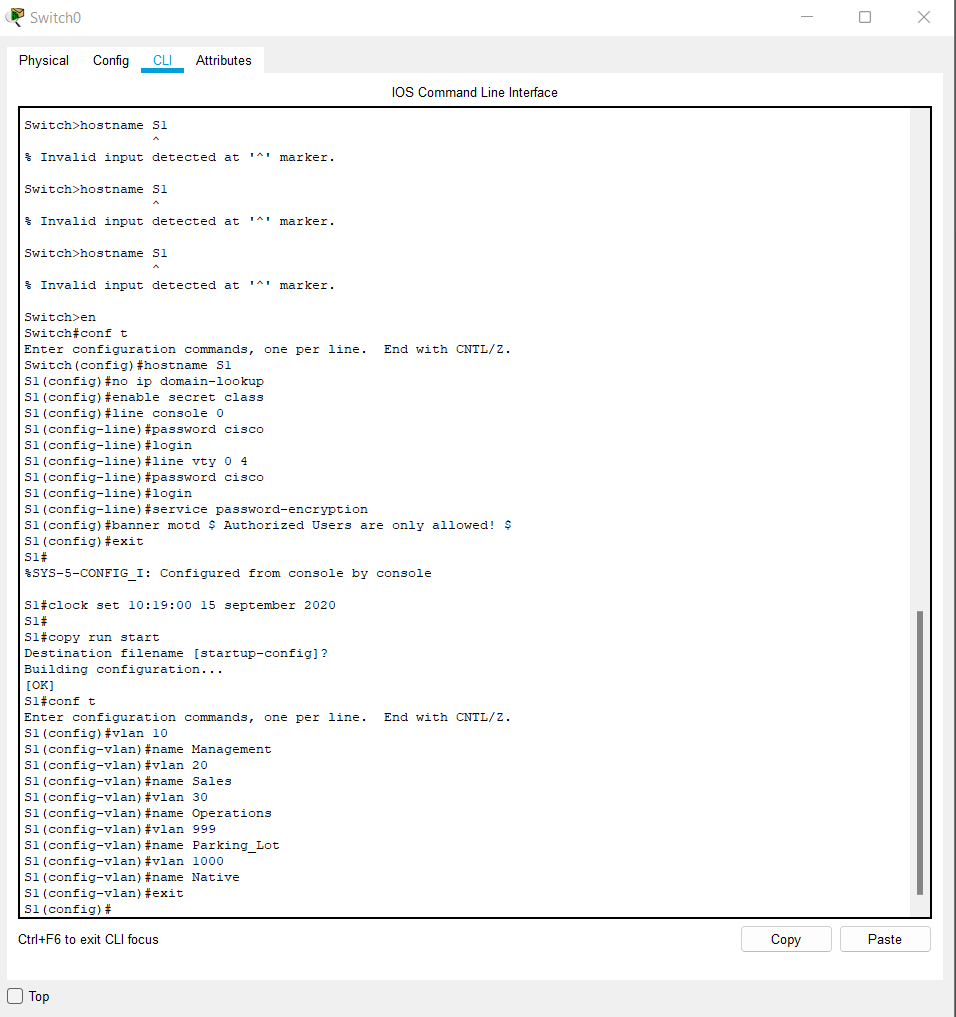
S1(config-vlan)# **vlan 999**

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**

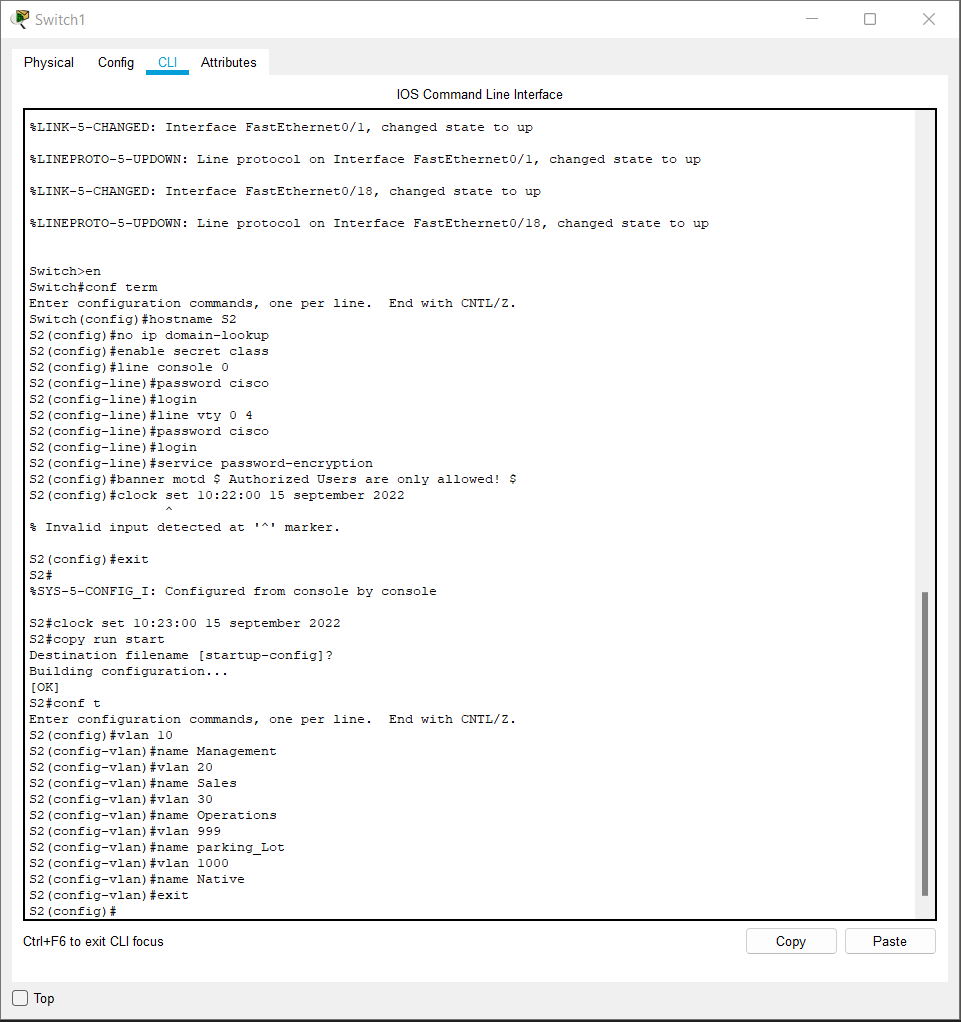
S2(config-vlan)# **vlan 999**

S2(config-vlan)# **name Parking\_Lot**

S2(config-vlan)# **vlan 1000**

S2(config-vlan)# **name Native**

S2(config-vlan)# **exit**



1. 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**

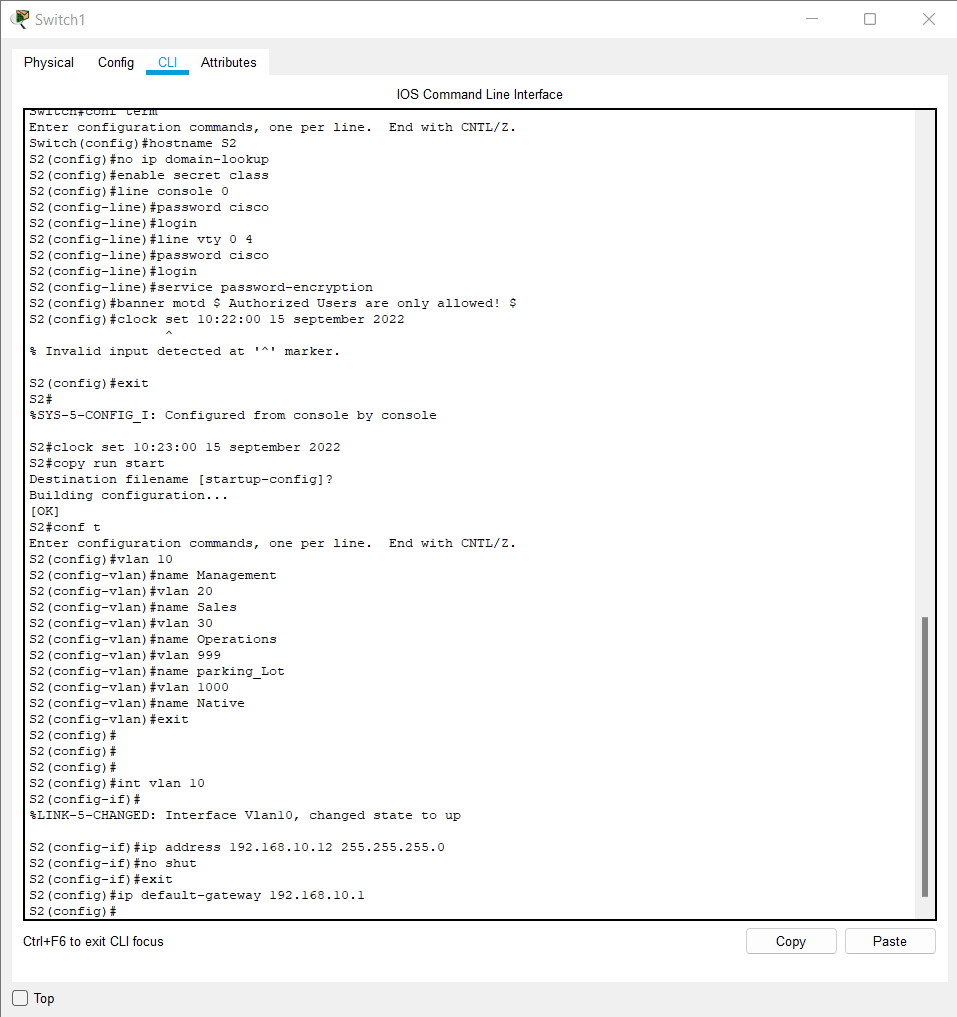
S2(config)# **interface vlan 10**

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**



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

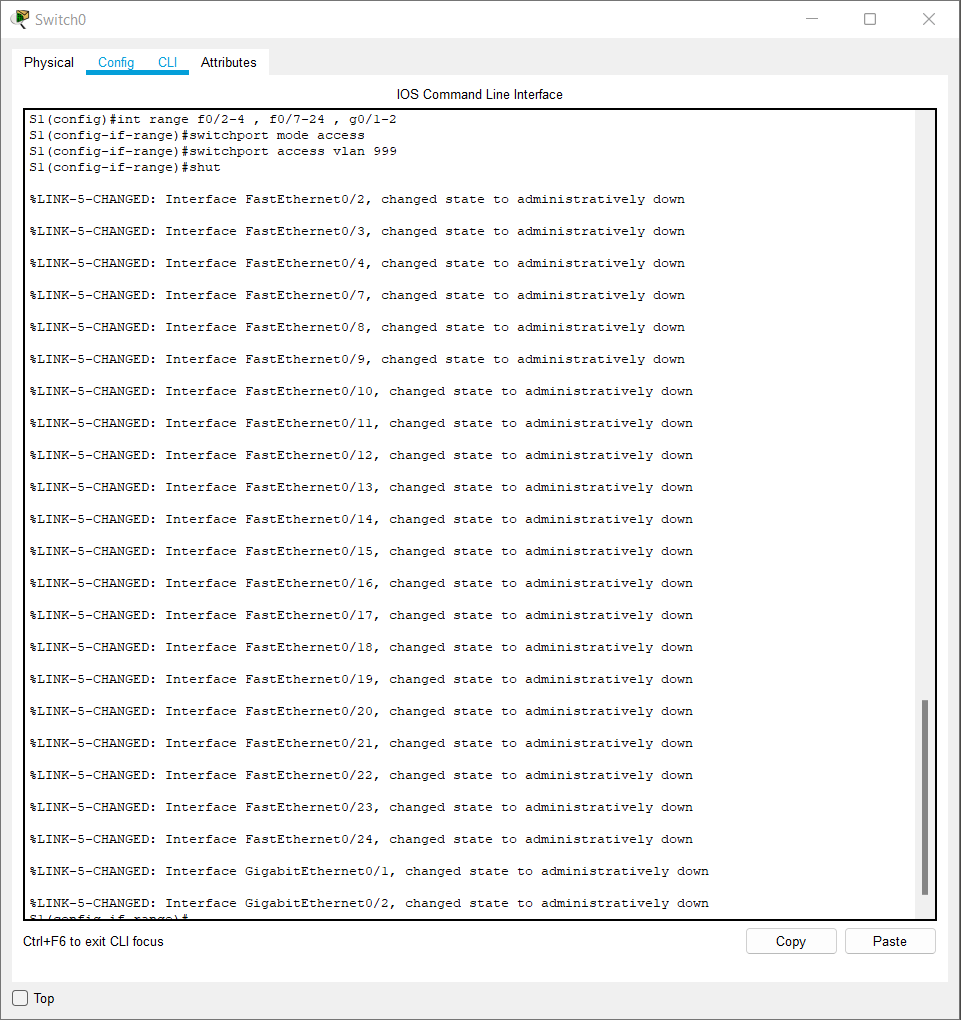
**Note**: The interface range command is helpful to accomplish this task with as few commands as necessary.

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**

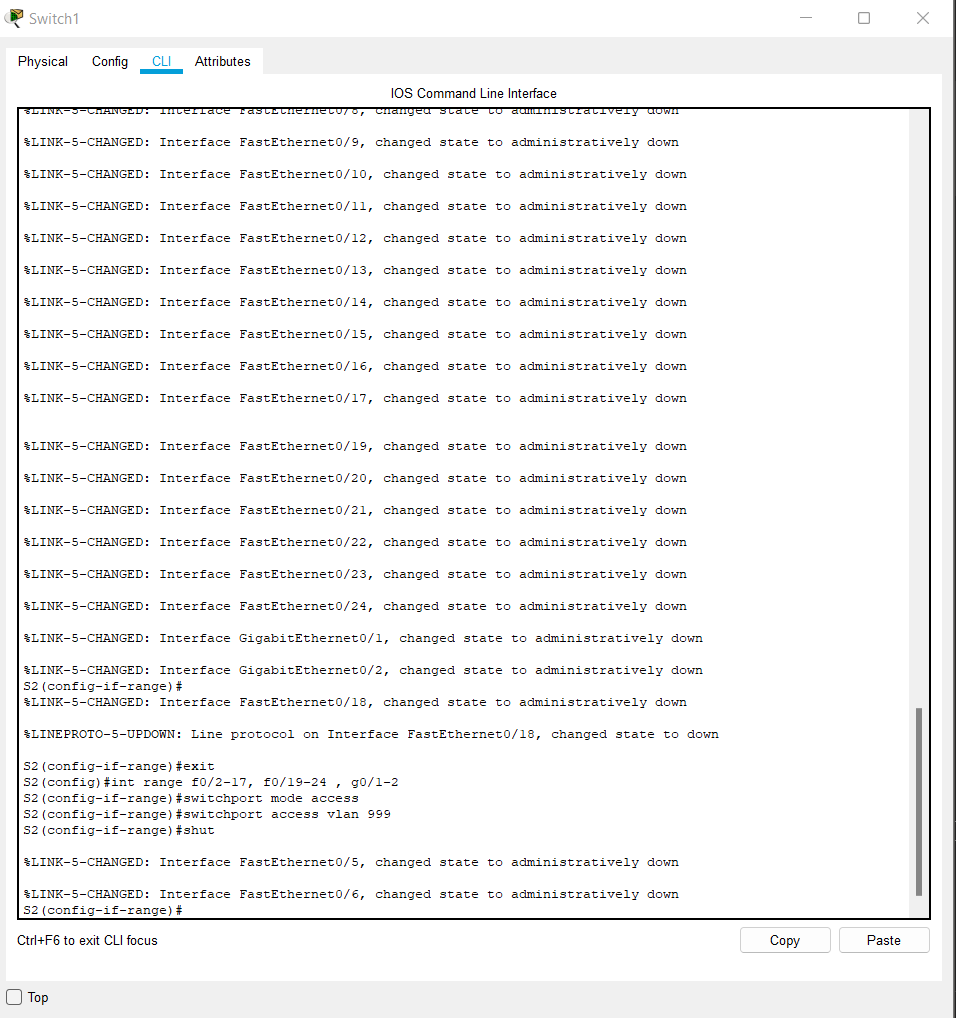


S2(config)#**interface range f0/2 – 17 , f0/19 – 24 , g0/1 – 2**

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

S2(config-if-range)# **switchport access vlan 999**

S2(config-if-range)# **shutdown**



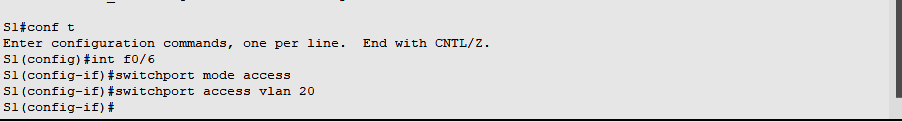
**Step 2:  Assign VLANs to the correct switch interfaces.**

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**

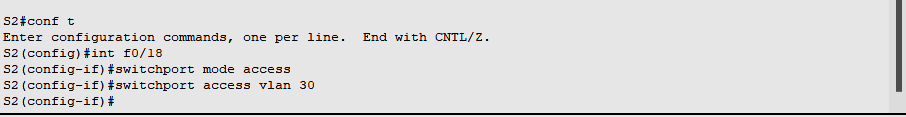
S1(config-if)# **switchport access vlan 20**



S2(config)# **interface f0/18**

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

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



1. 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

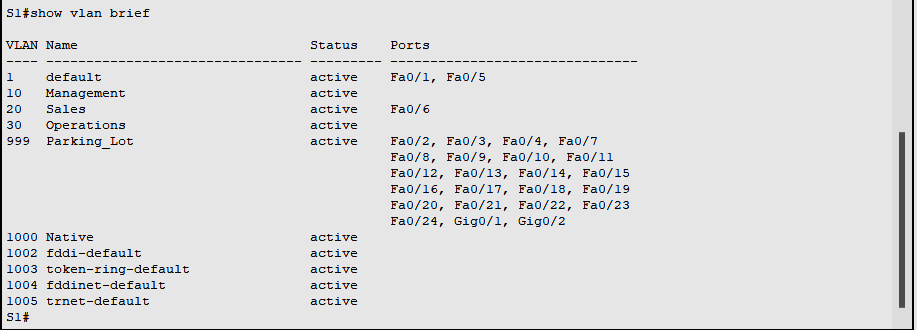
1000 Native                           active

1002 fddi-default                     act/unsup

1003 token-ring-default               act/unsup

1004 fddinet-default                  act/unsup

1005 trnet-default                    act/unsup



**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**

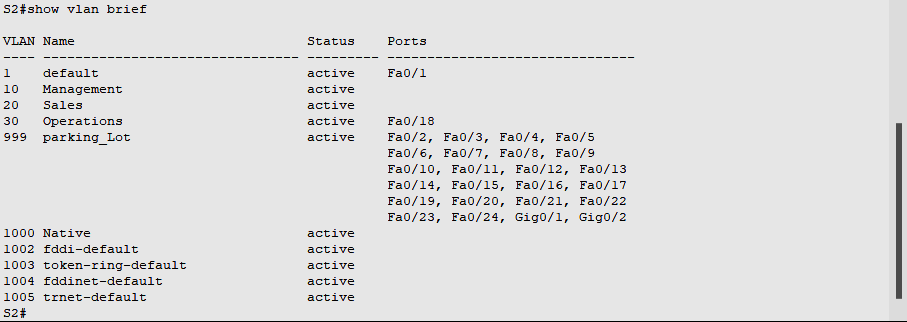
**1000 Native                           active**

**1002 fddi-default                     act/unsup**

**1003 token-ring-default               act/unsup**

**1004 fddinet-default                  act/unsup**

**1005 trnet-default                    act/unsup**



**Part 3:   Configure an 802.1Q Trunk Between the Switches**

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

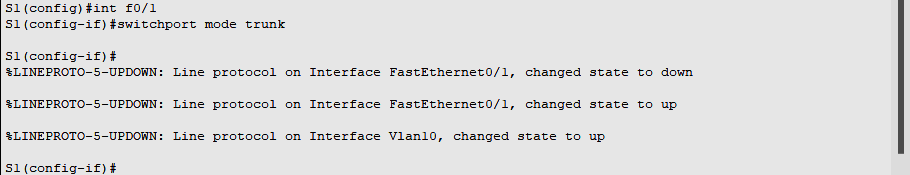
**Step 1:  Manually configure trunk interface F0/1 on switch S1 and S2.**

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

*Open configuration window*

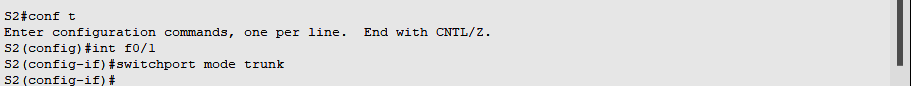
S1(config)# **interface f0/1**

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



S2(config)# **interface f0/1**

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



1. 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**

1. 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**

1. 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

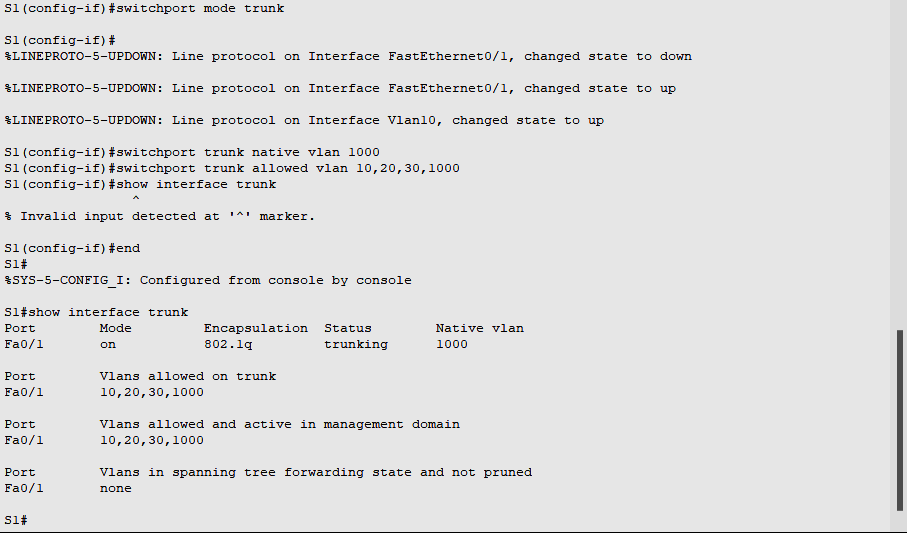
Fa0/1       10,20,30,1000

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

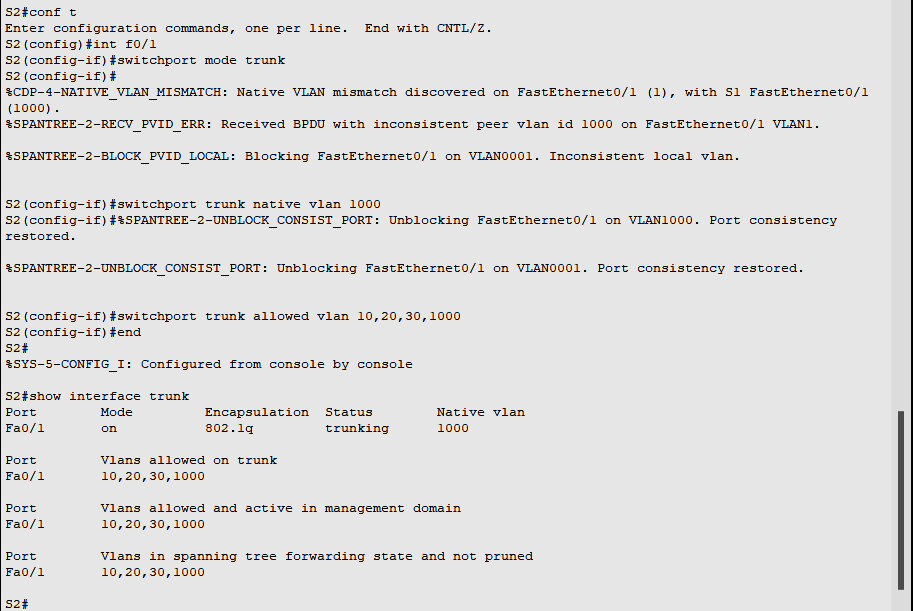
Fa0/1       10,20,30,1000

Port        Vlans allowed and active in management domain

Fa0/1       10,20,30,1000

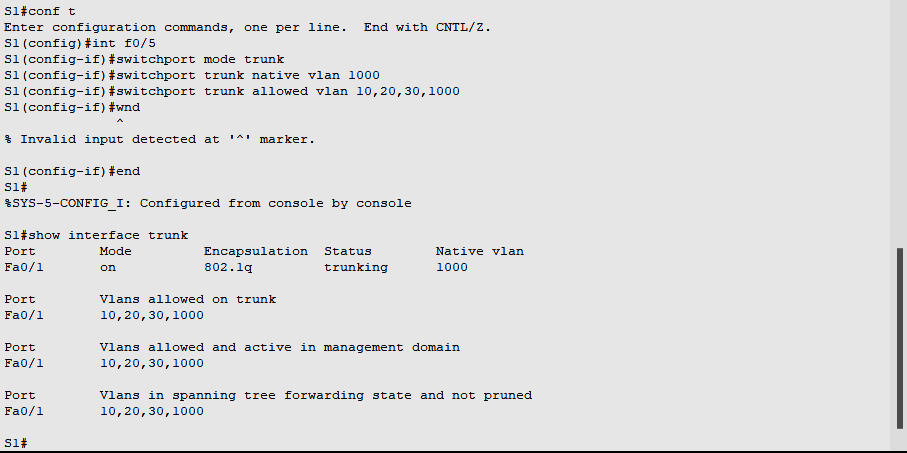
Port        Vlans in spanning tree forwarding state and not pruned

Fa0/1       10,20,30,1000



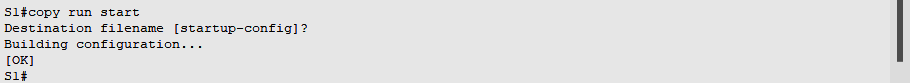
**Step 2:  Manually configure S1’s trunk interface F0/5**

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

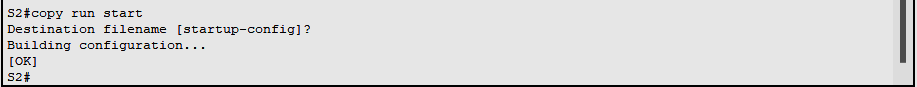


1. Save the running configuration to the startup configuration file.

S1# **copy running-config startup-config**



S2# **copy running-config startup-config**



1. 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**

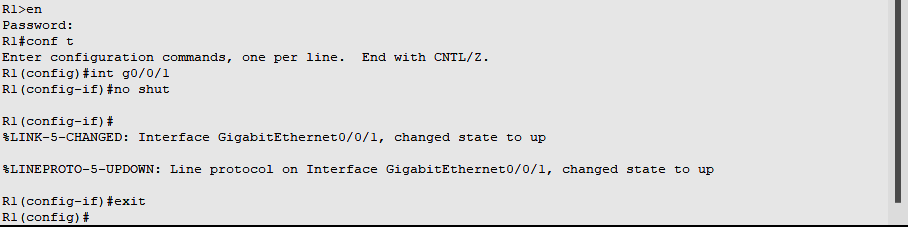
**Step 1:  Configure the router.**

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**



1. 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**

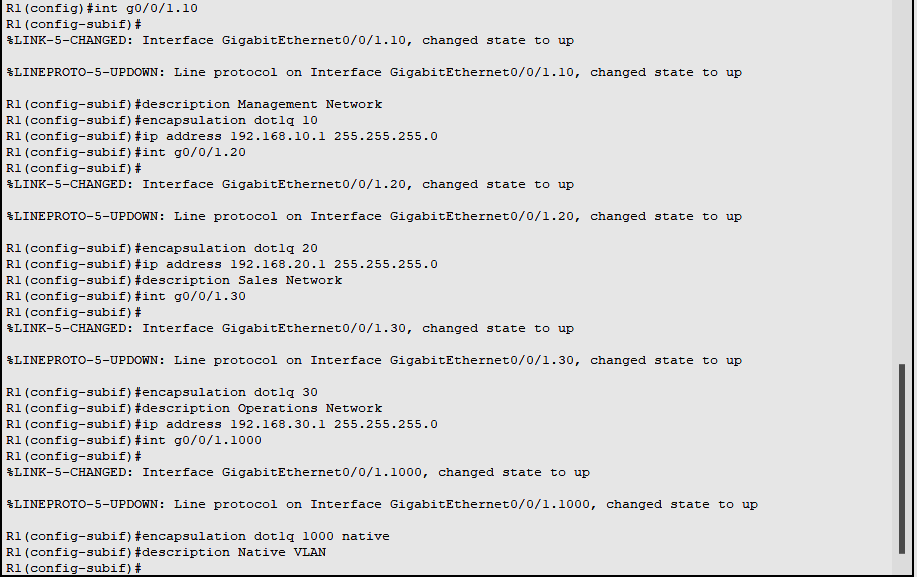
R1(config-subif)# **description Operations Network**

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**



1. 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

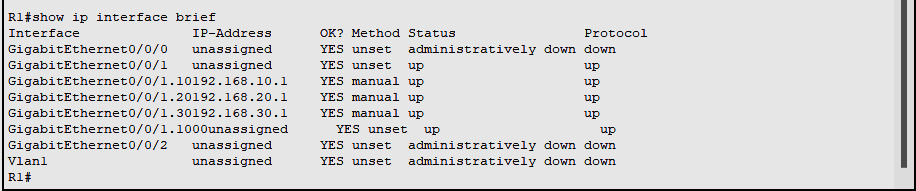
Gi0/0/1.10             192.168.10.1    YES manual up                    up

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

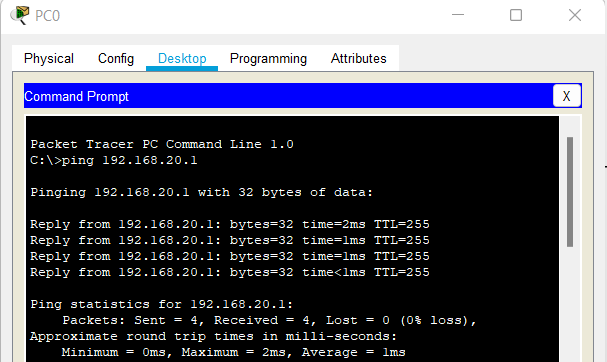


**Part 5:   Verify Inter-VLAN Routing is Working**

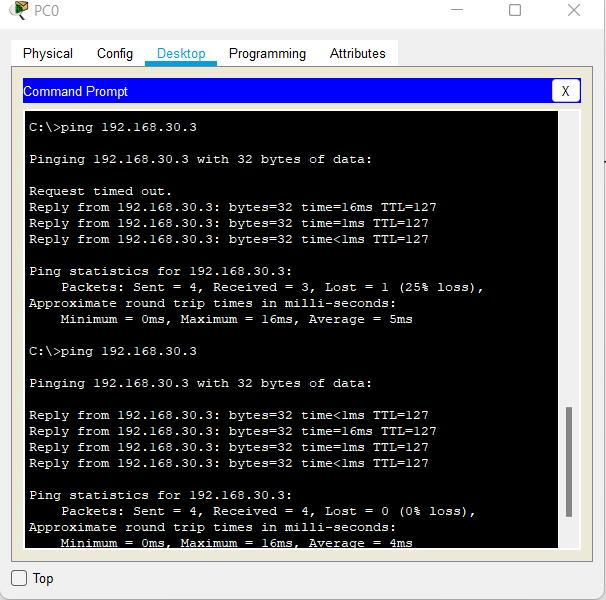
**Step 1:  Complete the following tests from PC-A. All should be successful.**

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

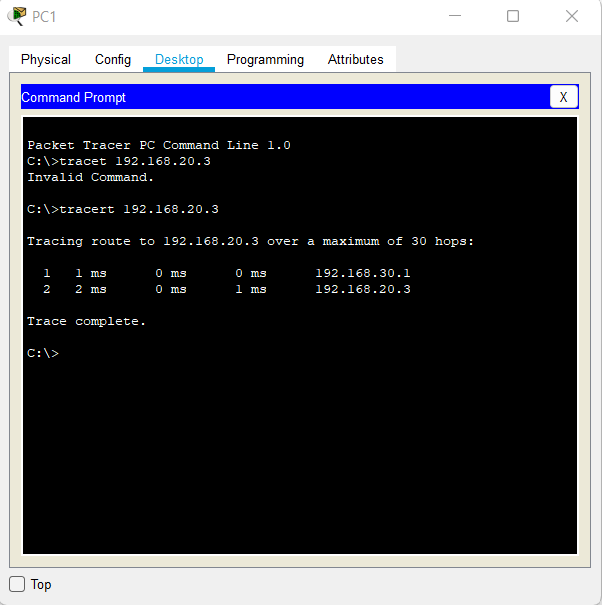
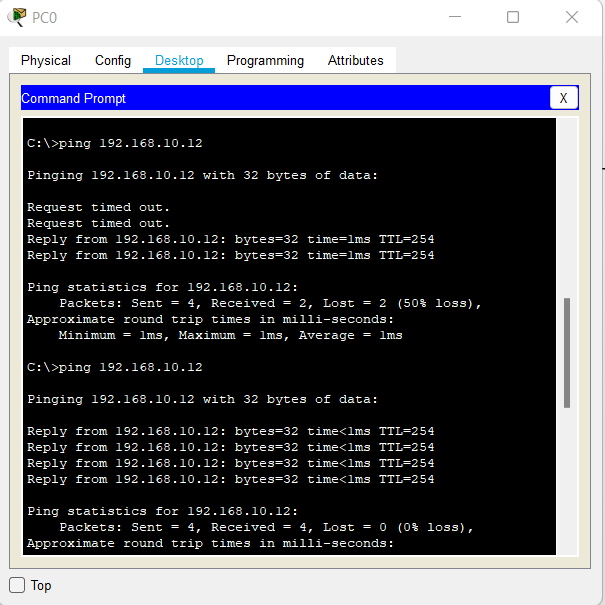
1. Ping from PC-A to its default gateway. 192.168.20.1



1. Ping from PC-A to PC-B



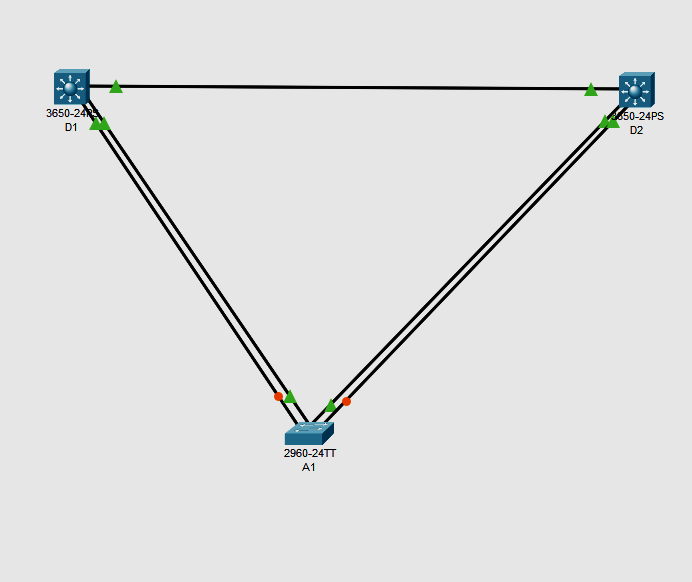
1. Ping from PC-A to S2



**Practical 6**

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

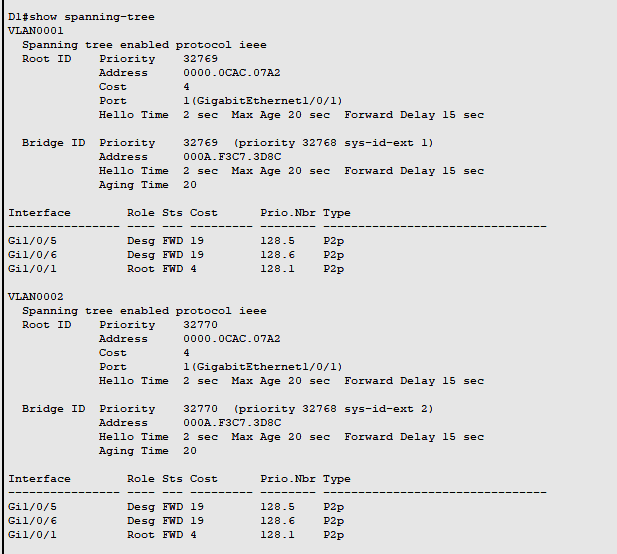
no shut

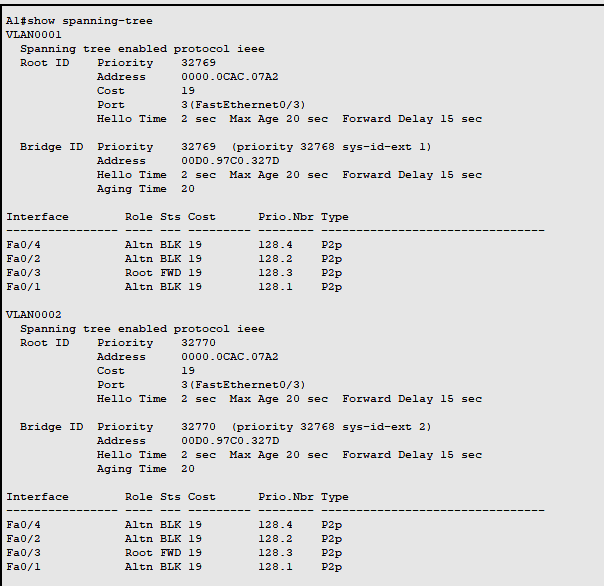
exit

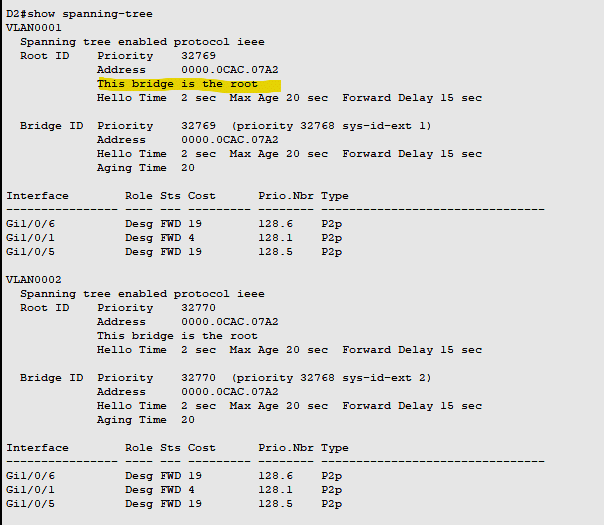
**Part 2:  Discover the Default Spanning Tree**

**Step 1:  Find the root bridge.**

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







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.

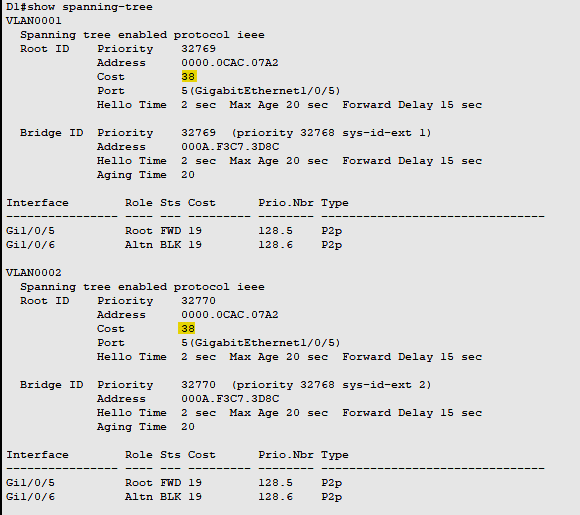
Our topology does not really illustrate the difference between port cost and path cost very well, so we will introduce a change in the network to achieve this. At D2, shutdown the g1/0/1 interface. The result of this is that D1 will have to change the port it considers root, and we will then see the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



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 tp rapid spanning tree mode in D2 and then run no shut comman 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)#

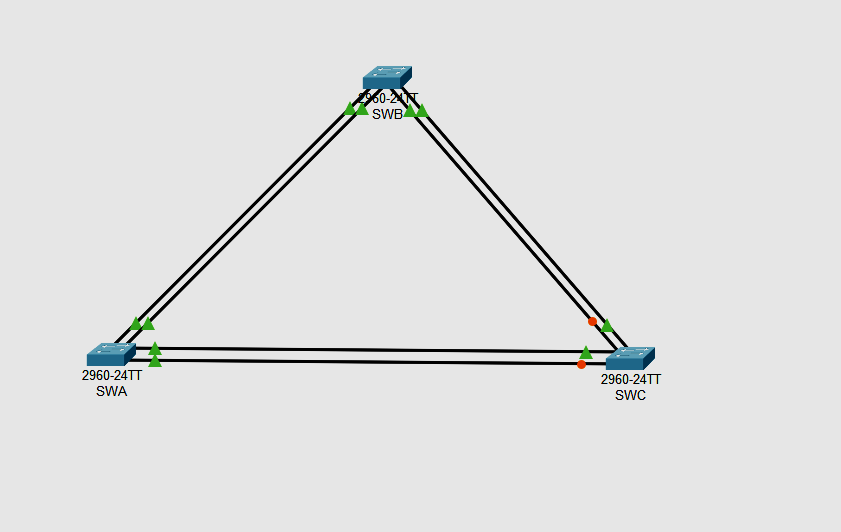
A1 was the last switch that was configured for RSTP. As you can see, interface VLAN1 was only down for 0.048 seconds. This is the “rapid” in rapid spanning tree.

Practical 7

1.IMPLEMENT ETHERCHANNEL

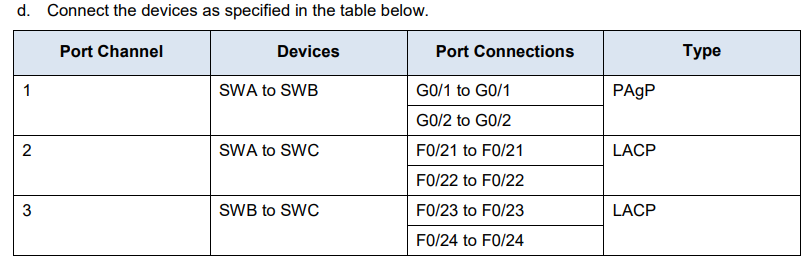
**Part 1: Build the network.**

Use the table below to build the switch topology.



**Step 2: Name the devices.**

**Step 3: Connect the devices.**



**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

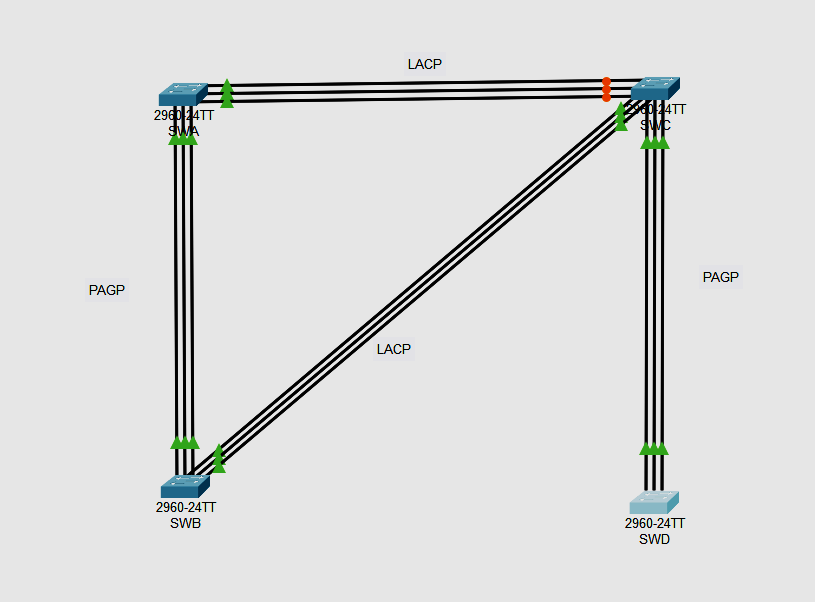
Port Vlans in spanning tree forwarding state and not pruned

Po1 1

Po3 1

**7B.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 |  |
| *1* | *SWA to SWB* | F0/3 to F0/3 |  |
| 2  *2* | SWA to SWC  *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 |  |
| *3* |  | 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**

*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.**

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

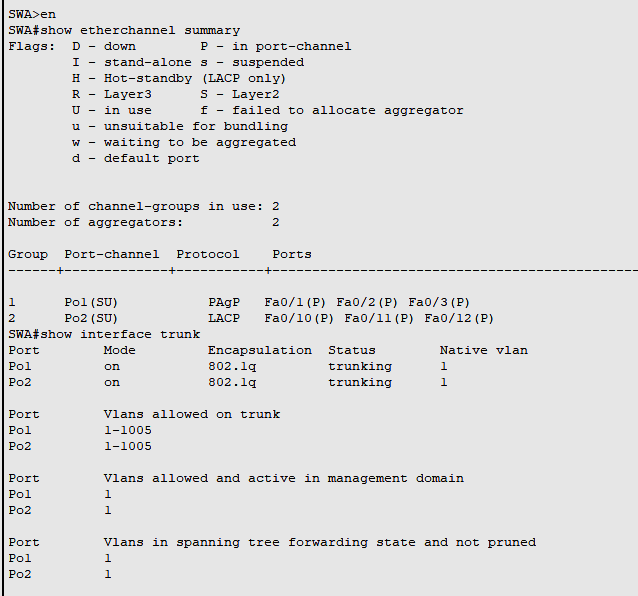
SWD(config-if-range)#no shut

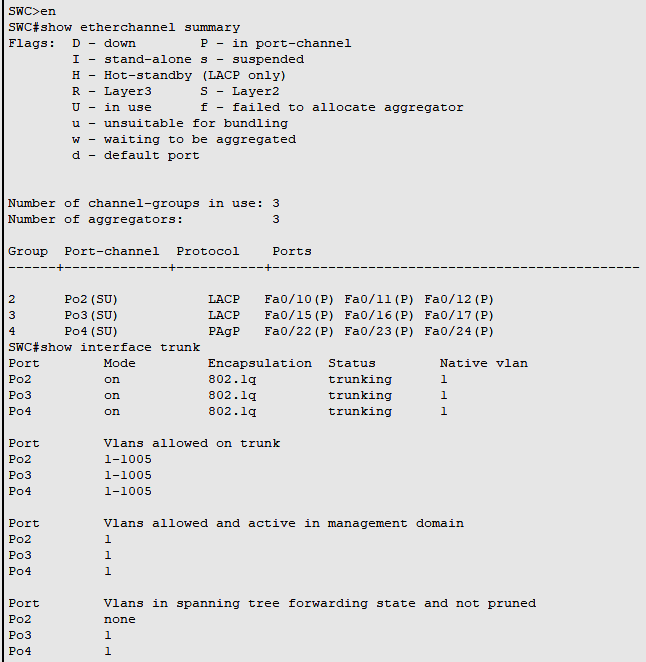
SWD(config-if-range)#exit

**Part 4: Checking output**

Output to see all the etherchannel connections

Go to any one switch and write the follow command

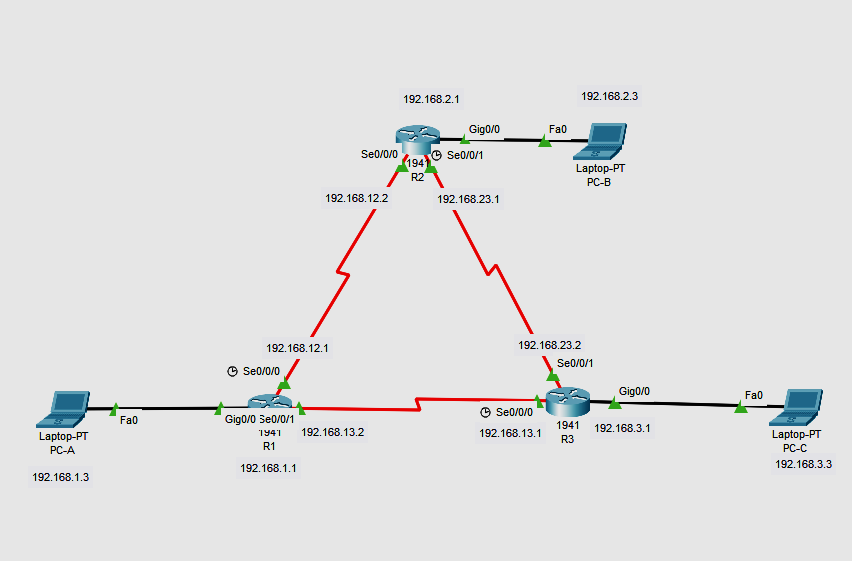




PRACTICAL 8

Implement Single channel OSPFv2

**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**

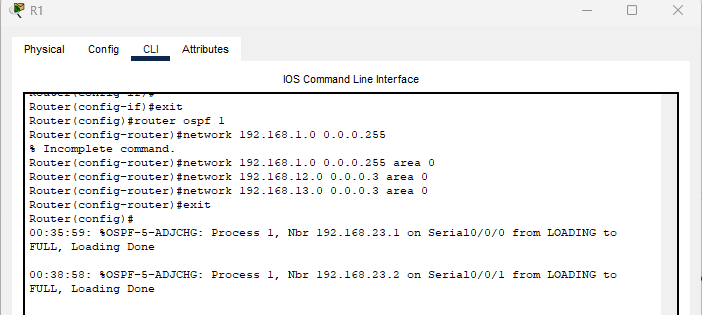
**Note**: The OSPF process id is kept locally and has no meaning to other routers on the network.

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**



### **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:**

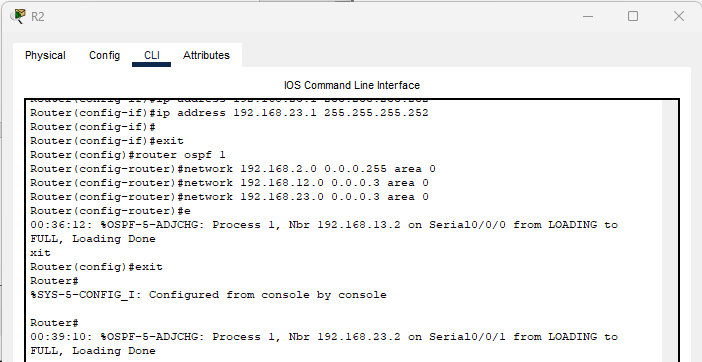
Router(config)#router ospf 1

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



**On R3 :**

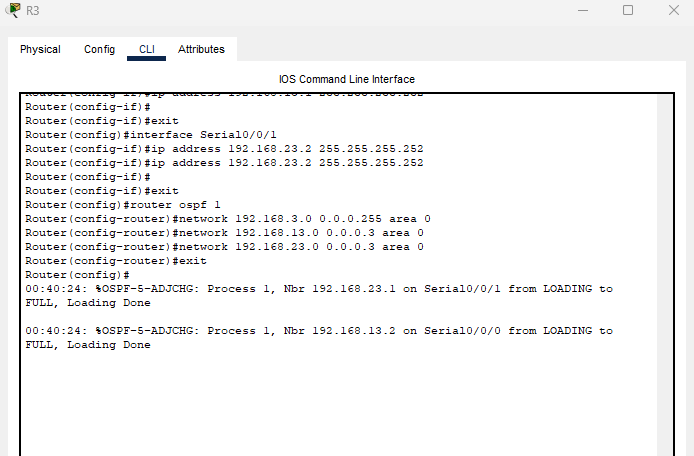
Router(config)#router ospf 1

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

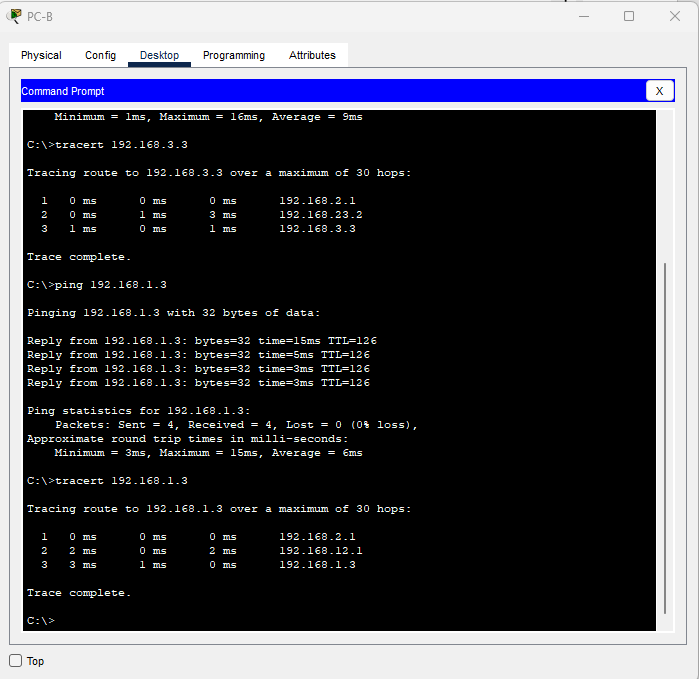
Router(config-router)#exit



### **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

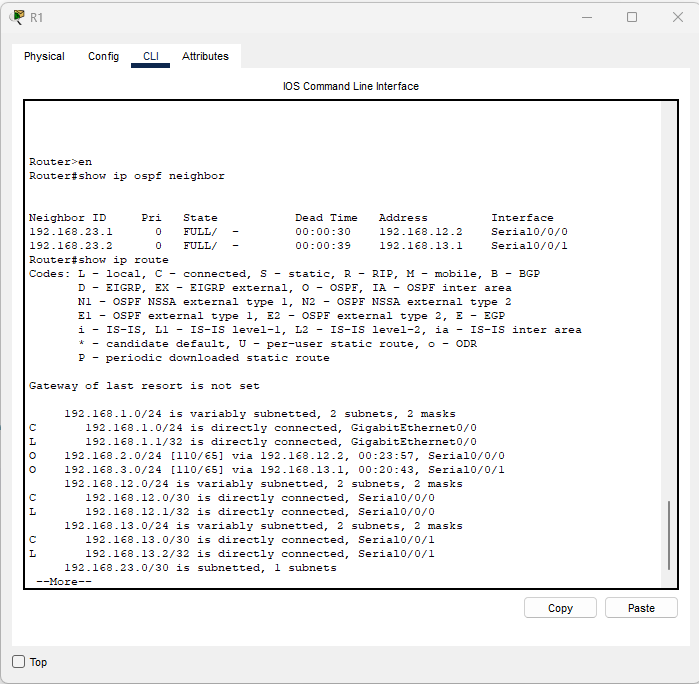
So see the route



### Step 3:Verify OSPF neighbors and routing information.

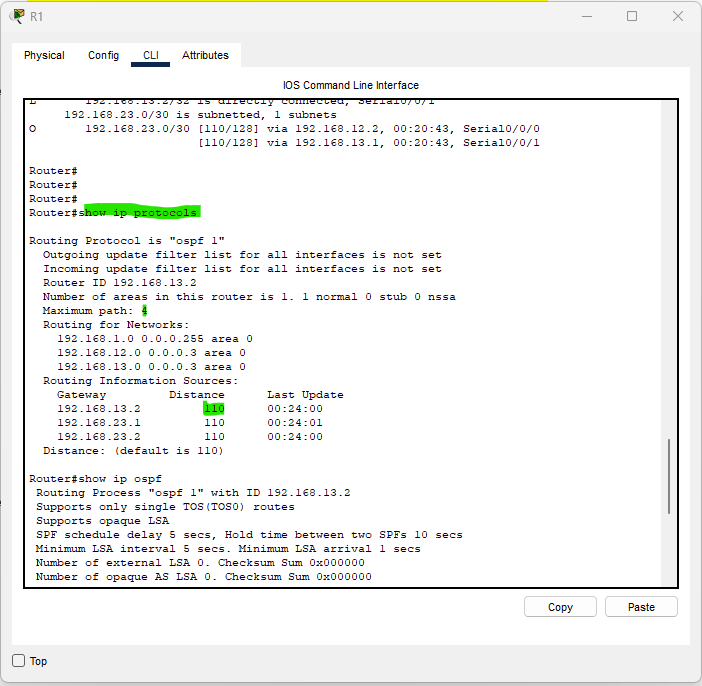
Issue the **show ip ospf neighbor** command to verify that each router lists the other routers in the network as neighbors.

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



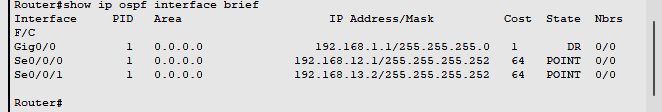
### **Step 4: Verify OSPF protocol settings.**

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.



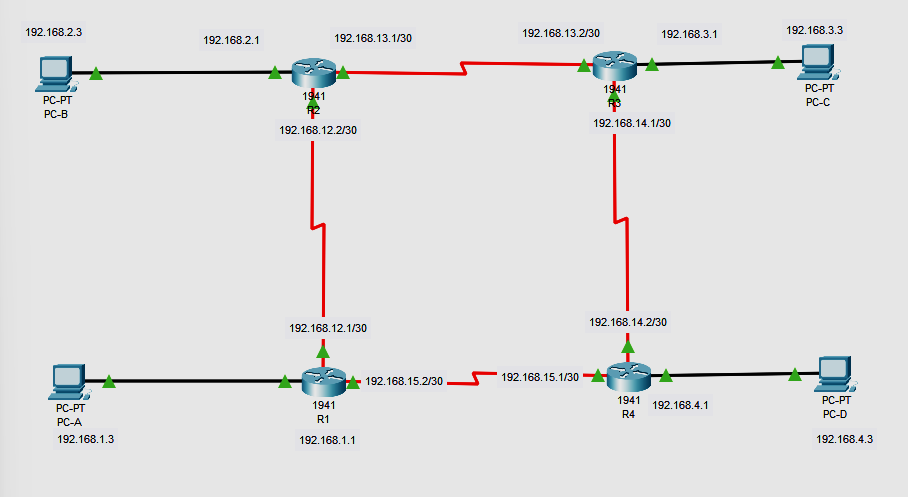
### **Step 5:Verify OSPF interface settings.**

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



**Implement Multi channel OSPFv2**

**Topology**



**Addressing Table**

| **Device** | **Interface** | **IP Address** | **Subnet Mask** | **Default Gateway** |
| --- | --- | --- | --- | --- |
| R1 | G0/0 | 192.168.1.1/24 | 255.255.255.0 | N/A |
| S0/0/0 (DCE) | 192.168.12.1/30 | 255.255.255.252 | N/A |
| S0/0/1 | 192.168.15.2/30 | 255.255.255.252 | N/A |
| R2 | G0/0 | 192.168.2.1/24 | 255.255.255.0 | N/A |
| S0/0/0 | 192.168.12.2/30 | 255.255.255.252 | N/A |
| S0/0/1 (DCE) | 192.68.13.1/30 | 255.255.255.252 | N/A |
| R3 | G0/0 | 192.168.3.1/24 | 255.255.255.0 | N/A |
| S0/0/0 (DCE) | 192.168.14.1/30 | 255.255.255.252 | N/A |
| S0/0/1 | 192.168.13.2/30 | 255.255.255.252 | N/A |
| R4 | G0/0 | 192.168.4.1/24 | 255.255.255.0 | N/A |
| S0/0/0 | 192.168.14.2/30 | 255.255.255.252 | N/A |
| S0/0/1 (DCE) | 192.168.15.1/30 | 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 |
| PC-D | NIC | 192.168.4.3 | 255.255.255.0 | 192.168.4.1 |

# **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 192.168.15.2 255.255.255.252

Router(config-if)#no shutdown

**Do the same to configure the R2 ,R3 and R4 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.

On R1:

R1(config)#router ospf 1

R1(config-router)#network 192.168.1.0 0.0.0.255 area 0

R1(config-router)#network 192.168.15.0 0.0.0.3 area 0

R1(config-router)#network 192.168.12.0 0.0.0.3 area 0

On R2

R2(config)#router ospf 1

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

R2(config-router)#network 192.168.12.0 0.0.0.3 area 0

R2(config-router)#network 192.168.13.0 0.0.0.3 area 1

R2(config-router)#network 192.168.14.0 0.0.0.3 area 2

On R3:

R3(config)#router ospf 1

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

R3(config-router)#network 192.168.13.0 0.0.0.3 area 1

R3(config-router)#network 192.168.12.0 0.0.0.3 area 0

R3(config-router)#network 192.168.14.0 0.0.0.3 area 2

On R4:

R4(config)#router ospf 1

R4(config-router)#network 192.168.4.0 0.0.0.255 area 0

R4(config-router)#network 192.168.15.0 0.0.0.3 area 0

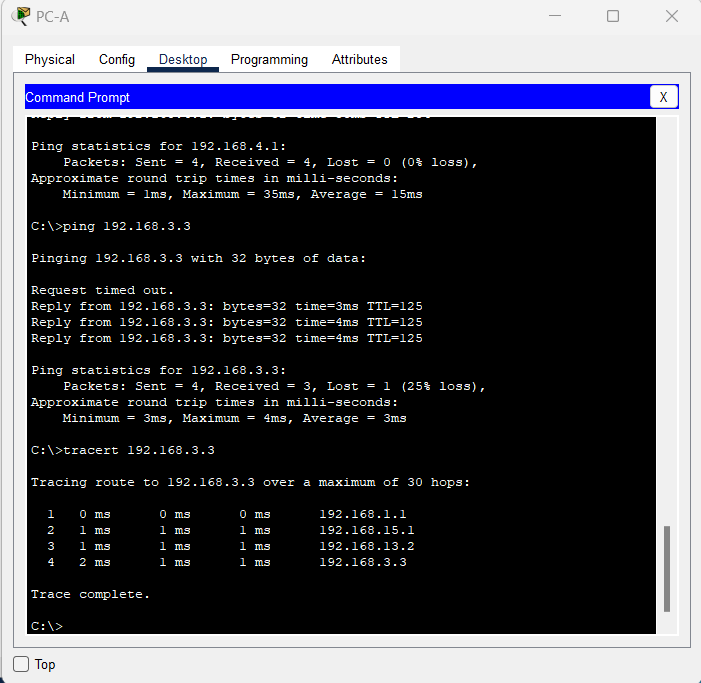
R4(config-router)#network 192.168.14.0 0.0.0.3 area 2

R4(config-router)#network 192.168.13.0 0.0.0.3 area 1

### **Step 2: Check the connectivity**

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

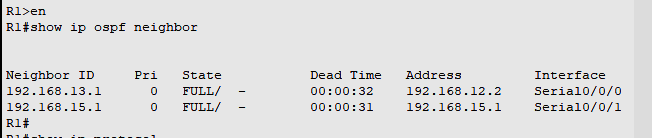
So see the route



### **Step 3:Verify OSPF neighbors and routing information.**

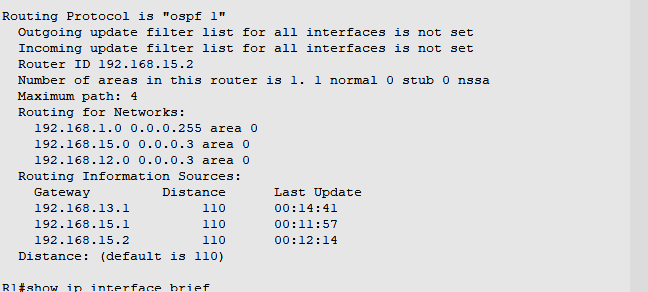
Issue the **show ip ospf neighbor** command to verify that each router lists the other routers in the network as neighbors.

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



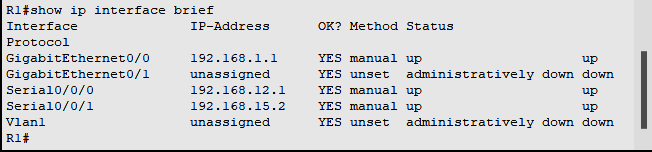
### **Step 4:Verify OSPF protocol settings.**

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.



### **Step 5:Verify OSPF interface settings.**

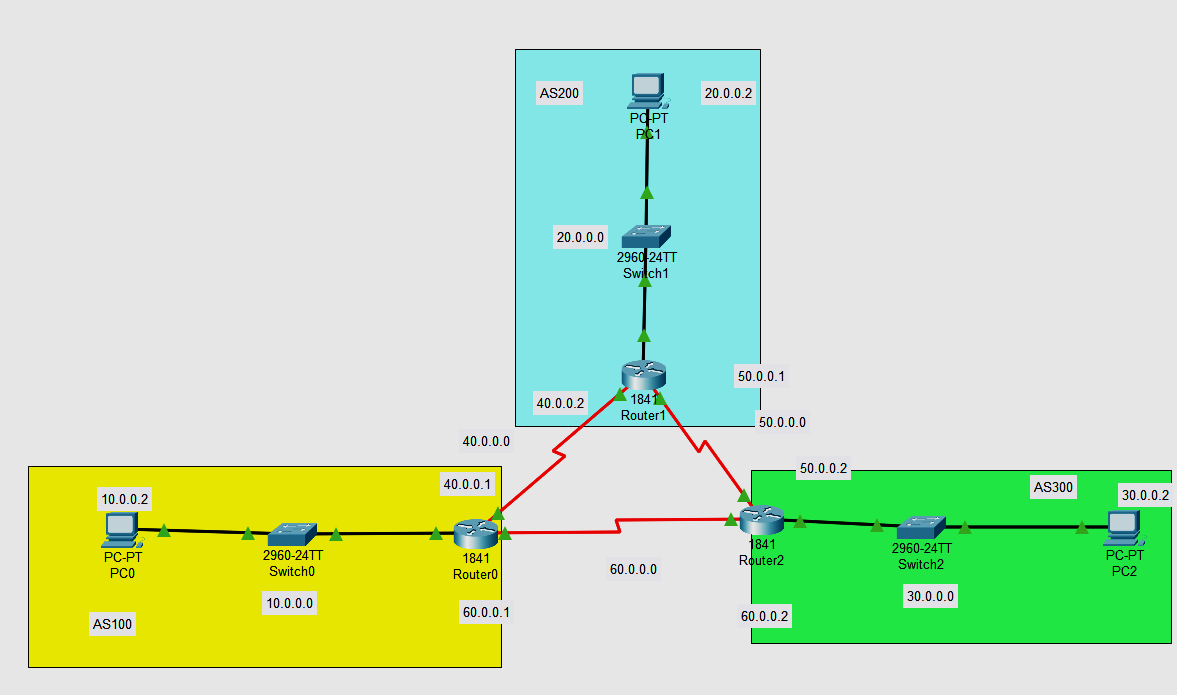
Issue the **show ip ospf interface brief** command to display a summary of OSPF-enabled interfaces.



**Practical 9**

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 | *N/A* |
| *R1* | S0/0/1 | 60.0.0.1 | 255.0.0.0 | *N/A* |
| R2  R2  R2 | Fa0/0 | 20.0.0.1 | 255.0.0.0 | N/A |
| S0/0/0 | 40.0.0.2 | 255.0.0.0 | N/A |
| S0/0/1 | 50.0.0.1 | 255.0.0.0 | N/A |
| R3  R3  R3 | Fa0/0 | 30.0.0.1 | 255.0.0.0 | N/A |
| S0/0/1 | 50.0.0.2 | 255.0.0.0 | N/A |
| 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:**

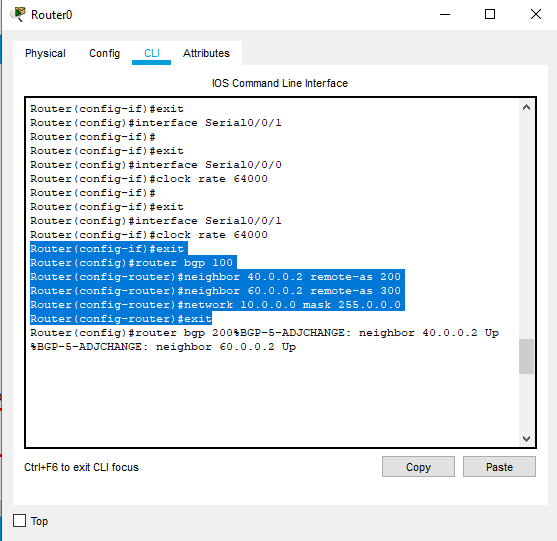
Router(config)#router bgp 100

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**

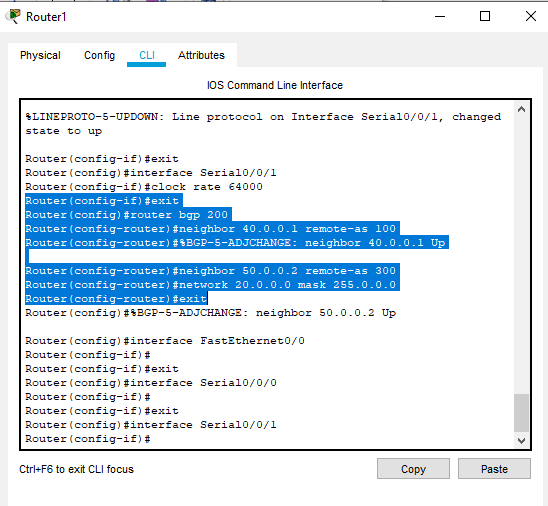
Router(config)#router bgp 200

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



**R3**

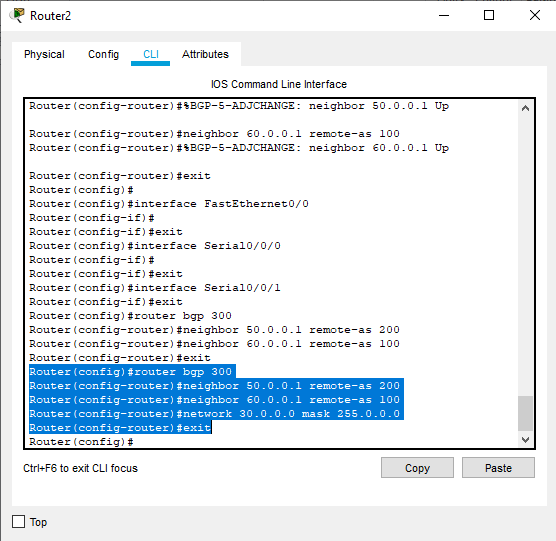
Router(config)#router bgp 300

Router(config-router)#neighbor 60.0.0.1 remote-as 100

Router(config-router)# neighbor 50.0.0.1 remote-as 200

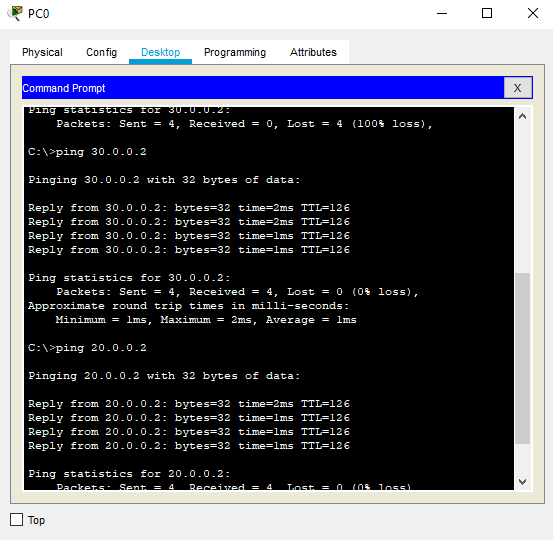
Router(config-router)#network 30.0.0.0 mask 255.0.0.0

Router(config-router)#exit



Step 2 Ping from PC to check connection

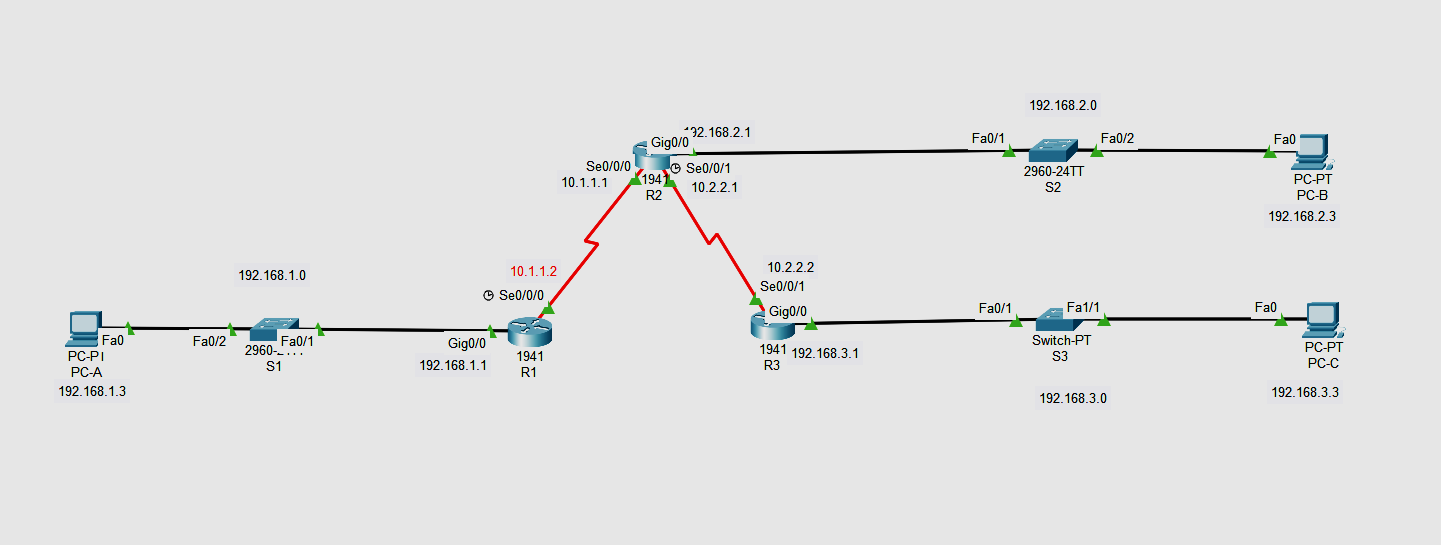
Ping from PC-C to PC-B



**Practical 10**

Implement IPsec Site-to-Site VPNs

**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

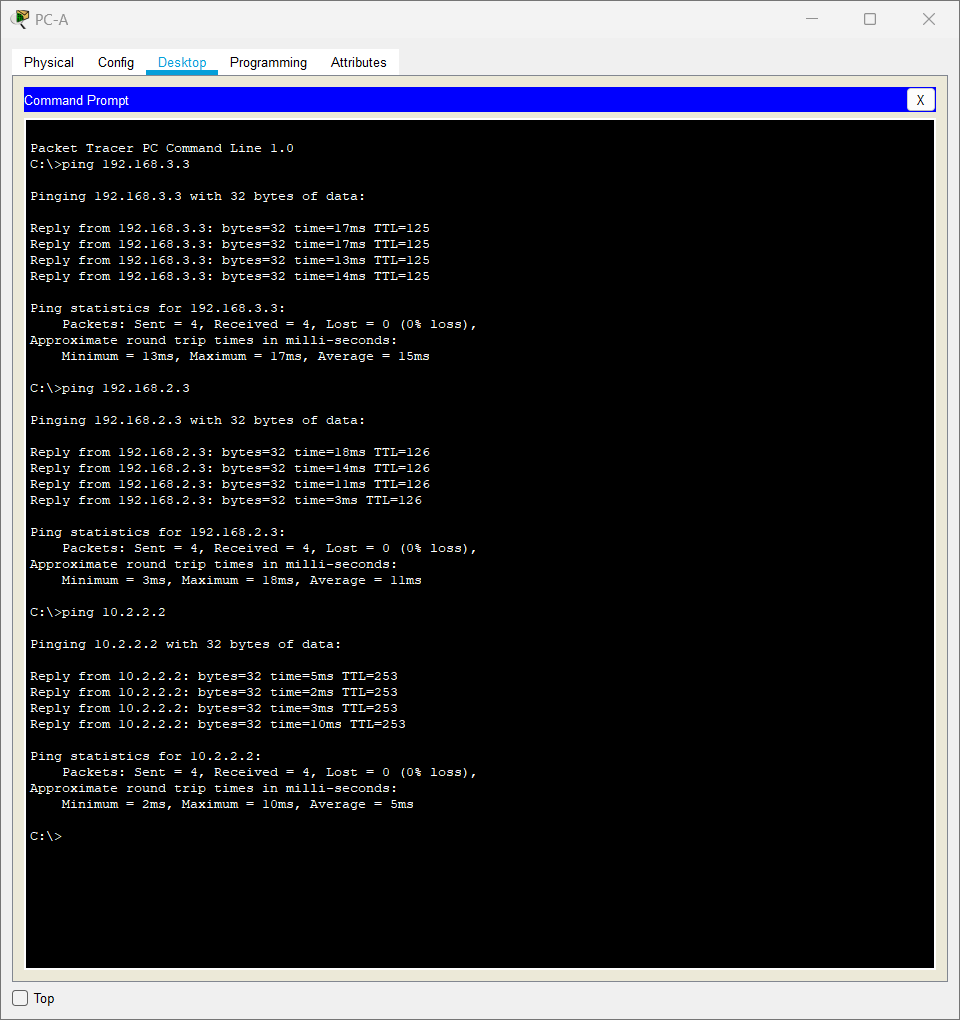
Router(config-router)# network 192.168.0.0

**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**

