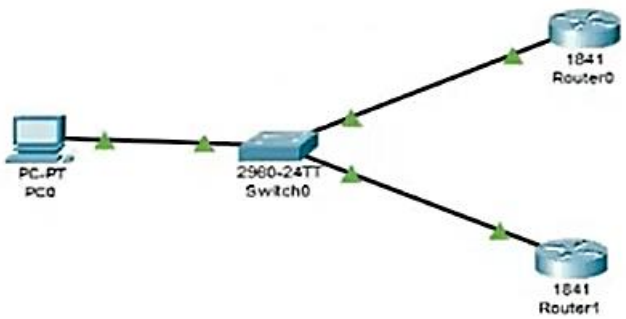
Implement SPAN Technologies (Switch Port Analyzer)

Address Table

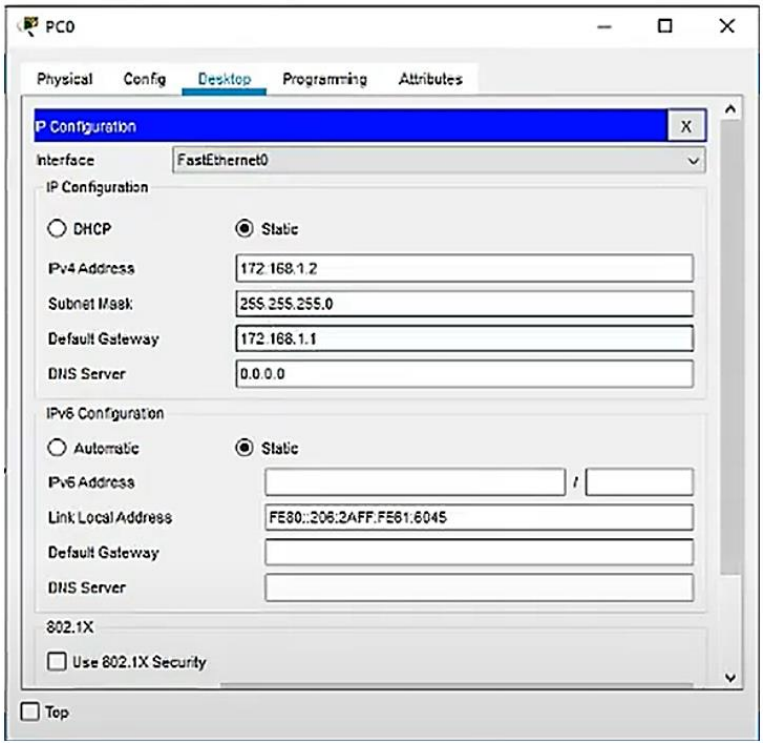
Device	IP Address
PC0	10.0.0.2
PC1	10.0.0.3

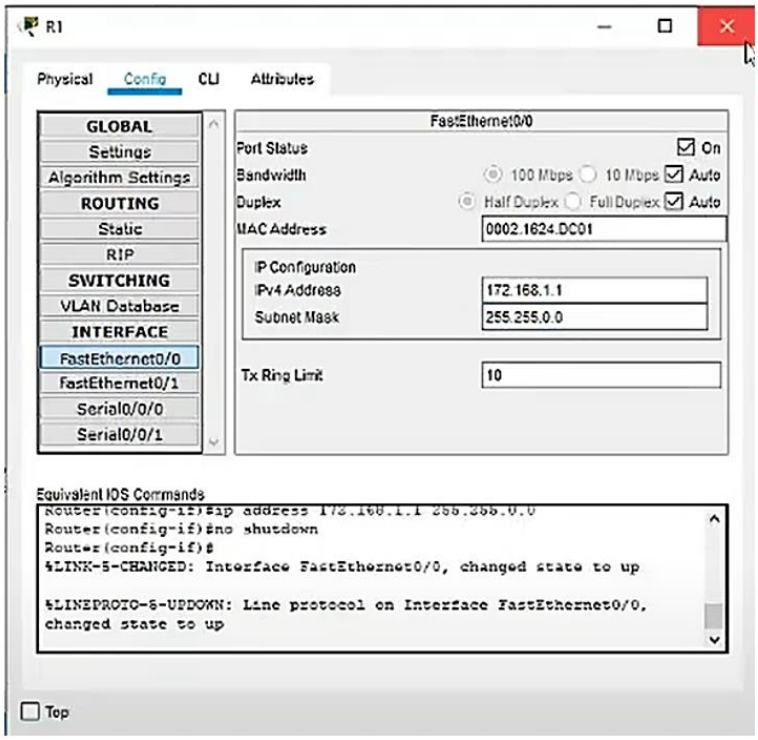
```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#monitor session 1 source int f0/1
Switch(config)#monitor session 1 destination int f0/2
Switch(config)#end
Switch#show monitor session 1
Session 1
---------
Type                   : Local Session
Description            : -
Source Ports           :
    Both               : Fa0/1
Destination Ports      : Fa0/2
    Encapsulation      : Native
          Ingress      : Disabled
Switch#show monitor detail
```

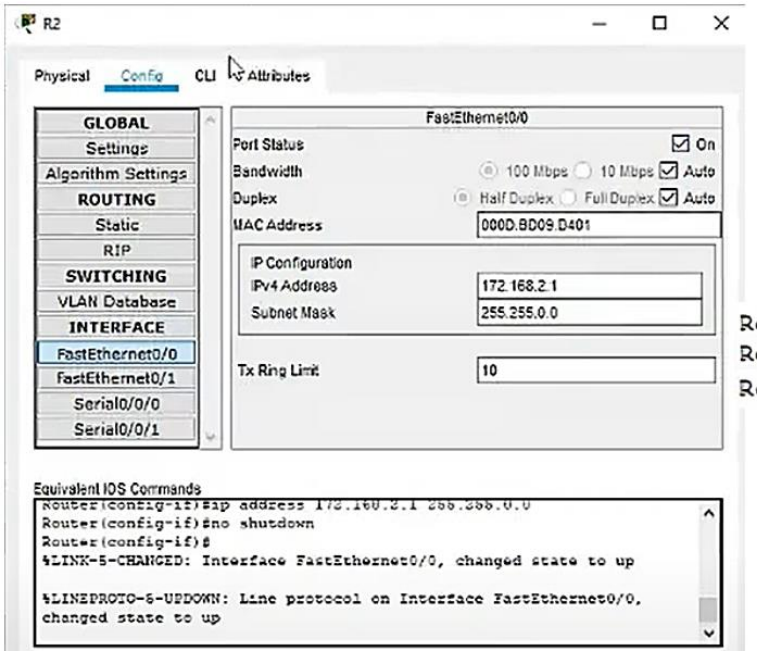
Implement SNMP



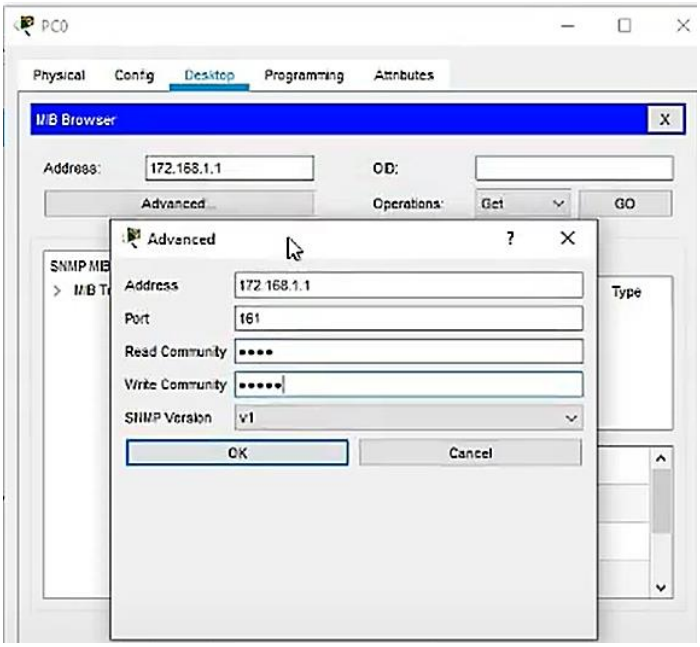
Device	Interface	IP Address
PC0	PC0	172.168.1.2 /172.168.1.1
R1	F0/0	172.168.1.1
R2	F0/0	172.168.2.1

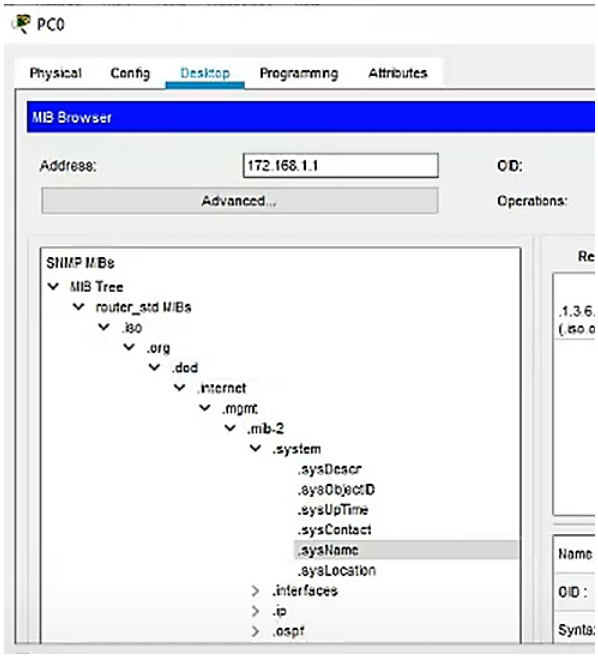




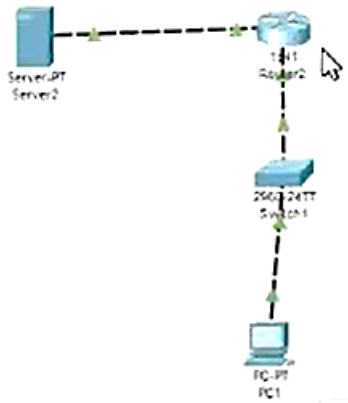



```
Router(config-if)#exit
Router(config)#snmp-server community read ro
Router(config)#snmp-server community write rw
```





Implement Syslog

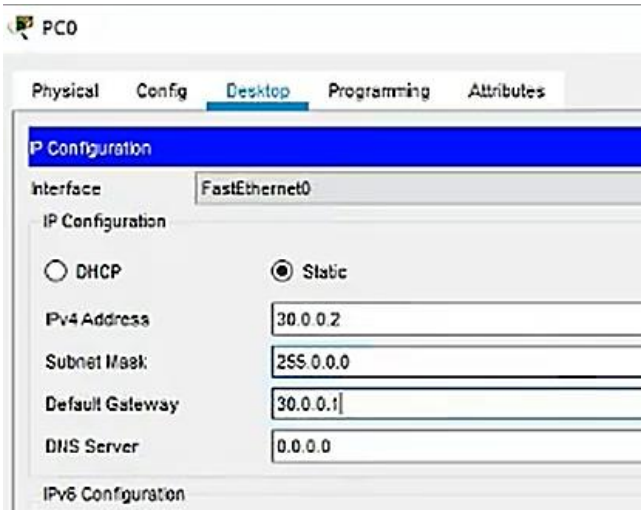


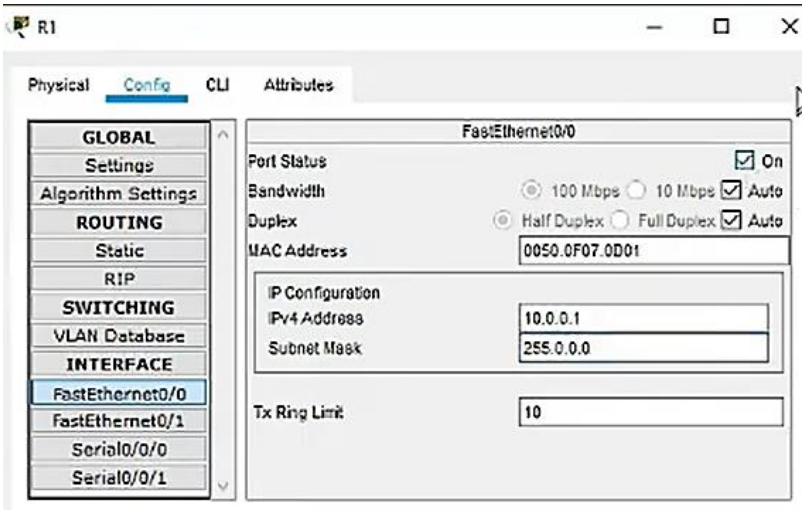
Device	Interface	IP Address
PC1		30.0.0.2/30.0.0.1
R2	F0/0 server	10.0.0.1
	F0/1 PC1	30.0.0.1
server		10.0.0.2/10.0.0.1

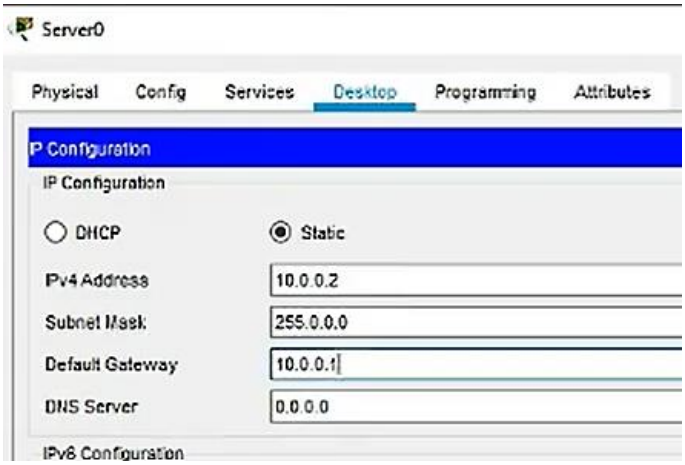
service timestamps log datetime msec

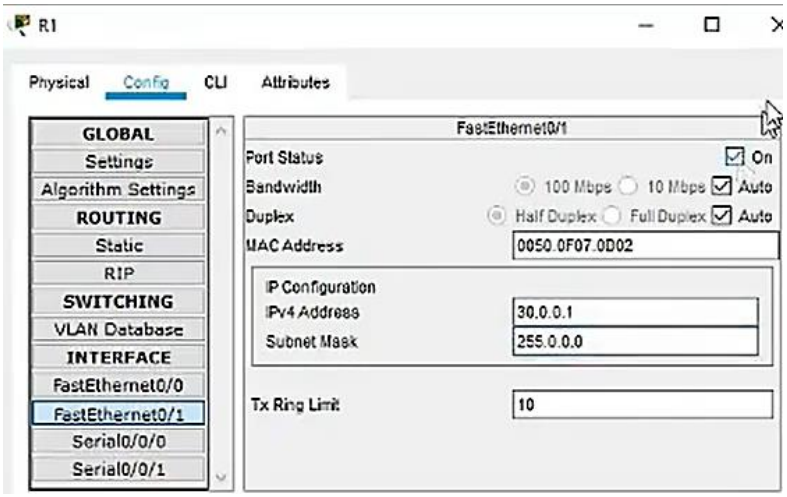
int f0/0.1

check syslog of server

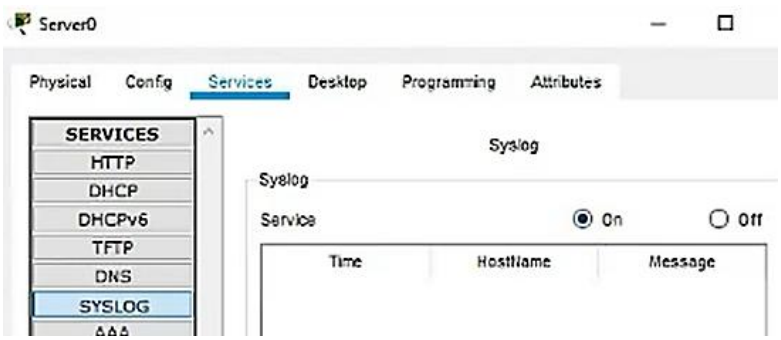


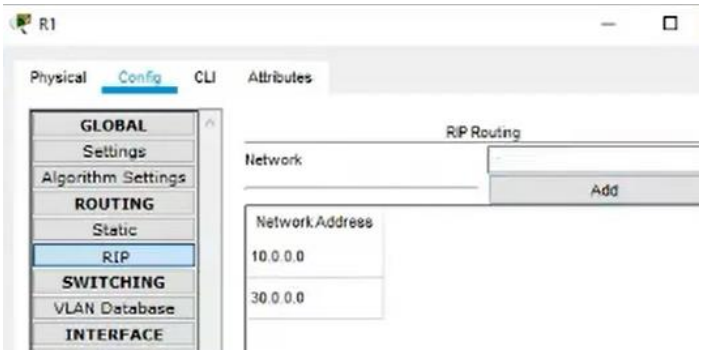







```
Router(config-if)#exit
Router(config)#service timestamps log datetime msec
Router(config)#int f0/0.1
```





Flow Simulator

int f0/0

ip flow ingress

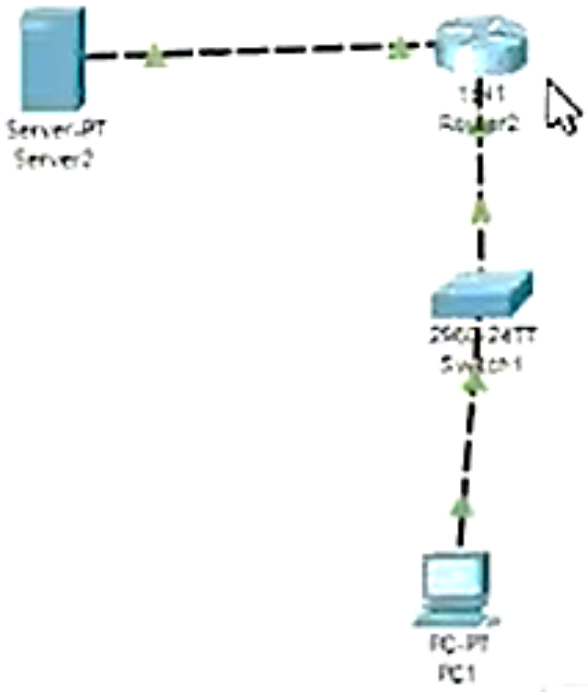
ip flow egress

ip flow-export destination 10.0.0.2 99

ip flow-export source f0/0

show ip cache flow

```
Router(config-if)#exit
Router(config)#int f0/0
Router(config-if)#ip flow ingress
Router(config-if)#ip flow egress
Router(config-if)#ip flow-export destination 10.0.0.2 99
Router(config)#ip flow-export source f0/0
Router(config)#end
Router#show ip cache flow
```



Practical 5: Implement Inter-VLAN Routing

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
	G0/0/1.20	192.168.20.1	255.255.255.0	
	G0/0/1.30	192.168.30.1	255.255.255.0	
	G0/0/1.1000	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

Objectives

Part 1: Build the Network and Configure Basic Device Settings
Step 1: Cable the network as shown in the topology.
Step 2: Configure basic settings for the router.
Step 3: Configure basic settings for each switch.
Step 4: Configure PC hosts.

Part 2: Create VLANs and Assign Switch Ports
Step 1: Create VLANs on both switches.
Step 2: Assign VLANs to the correct switch interfaces.

Part 3: Configure an 802.1Q Trunk between the Switches
Step 1: Manually configure trunk interface F0/1 on switch S1 and S2.
Step 2: Manually configure S1's trunk interface F0/5

Part 4: Configure Inter-VLAN Routing on the Router
Step 1: Configure the router.
Step 2: Complete the following tests from PC-A. All should be successful.
Step 3: Complete the following test from PC-B

Part 5: Verify Inter-VLAN Routing is working

Background:

VLAN trunks are used to span VLANs across multiple devices. Trunks allow the traffic from multiple VLANs to travel over a single link, while keeping the VLAN identification and segmentation intact. A particular kind of inter-VLAN routing, called "Router-on-a-Stick", uses a trunk from the router to the switch to enable all VLANs to pass to the router.

In this lab, you will create VLANs on both switches in the topology, assign VLANs to switch access ports, verify that VLANs are working as expected, create VLAN trunks between the two switches and between S1 and R1, and configure Inter-VLAN routing on R1 to allow hosts in different VLANs to communicate, regardless of which subnet the host resides.

ON S1 -> CLI

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNT
Switch(config)#hostname S1
S1(config)#vlan 10
S1(config-vlan)#name management
S1(config-vlan)#exit
S1(config)#vlan 20
S1(config-vlan)#name Sales
S1(config-vlan)#exit
S1(config)#vlan 30
S1(config-vlan)#name Operations
S1(config-vlan)#exit
S1(config)#vlan 999
S1(config-vlan)#name Parking_Lot
S1(config-vlan)#exit
S1(config)#vlan 1000
S1(config-vlan)#name Native
S1(config-vlan)#exit
S1(config)#end
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#show vlan br
S1#conf t
Enter configuration commands, one per line.  End with CN
S1(config)#int vlan 10
S1(config-if)#
%LINK-5-CHANGED: Interface Vlan10, changed state to up

S1(config-if)#ip address 192.168.10.11 255.255.255.0
S1(config-if)#exit
S1(config)#ip default-gateway 192.168.10.1
S1(config)#no shutdown
% Invalid input detected at '^' marker.

S1(config)#int vlan 10
S1(config-if)#no shutdown
S1(config-if)#exit
S1(config)#end
S1#show vlan br
S1>en
S1#conf t
Enter configuration commands, one per line.  End with CNTL/Z
S1(config)#int range f0/4-24,g0/1-2
S1(config-if-range)#switchport mode access
S1(config-if-range)#switchport access vlan 999
S1(config-if-range)#shutdown
S1(config-if-range)#exit
S1(config)#end
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#show vlan br
```

ON S2 -> CLI

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNT
Switch(config)#hostname S2
S2(config)#vlan 10
S2(config-vlan)#name Management
S2(config-vlan)#exi
S2(config)#exit
S2#
%SYS-5-CONFIG_I: Configured from console by console

S2#conf t
Enter configuration commands, one per line.  End with CNT
S2(config)#vlan 20
S2(config-vlan)#name Sales
S2(config-vlan)#exit
S2(config)#vlan 30
S2(config-vlan)#name Operations
S2(config-vlan)#exit
S2(config)#vlan 999
S2(config-vlan)#name Parking_Lot
S2(config-vlan)#exit
S2(config)#vlan 1000
S2(config-vlan)#name Native
S2(config-vlan)#exit
S2(config)#end
S2#
%SYS-5-CONFIG_I: Configured from console by console

S2#show vlan br
S2#conf t
Enter configuration commands, one per line.  End with CNTL/
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)#exit
S2(config)#ip default-gateway 192.168.10.1
S2(config)#int vlan 10
S2(config-if)#no shutdown
S2(config-if)#exit
S2(config)#int range f0/3-24,g0/1-2
S2(config-if-range)#switchport mode access
S2(config-if-range)#switchport access vlan 999
S2(config-if-range)#shutdown
S2(config-if-range)#end
S2#
%SYS-5-CONFIG_I: Configured from console by console

S2#show vlan br
```

ON S1 -> CLI

```
S1#conf t
Enter configuration commands, one per l
S1(config)#int f0/3
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 30
S1(config-if)#exit
S1(config)#end
```

ON S2 -> CLI

```
S2#conf t
Enter configuration commands, one per l:
S2(config)#int f0/2
S2(config-if)#switchport mode access
S2(config-if)#switchport access vlan 30
S2(config-if)#exit
S2(config)#end
S2#show vlan br
```

ON S1 -> CLI

```
S1#conf t
Enter configuration commands, one pe:
S1(config)#int f0/2
S1(config-if)#switchport mode access
S1(config-if)#switchport mode trunk
S1(config-if)#switchport trunk native vlan 1000
S1(config-if)#switchport trunk allowed vlan 1000
S1(config-if)#switchport trunk allowed vlan 10,20,30,1000
```

ON S2 -> CLI

```
S2#conf t
Enter configuration commands, one pe
S2(config)#int f0/1
S2(config-if)#switchport mode trunk
S2(config-if)#switchport trunk native vlan 1000
S2(config-if)#switchport trunk allowed vlan 10,20,30,1000
S2(config-if)#exit
S2(config)#end
S2#show vlan br
S2#show int trunk
```

ON R1 -> CLI

```
Router>en
Router#conf t
Enter configuration commands, one per lin
Router(config)#int f0/0.10
Router(config-subif)#description vlan 10
Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#ip address 192.168.10.1 255.255.255.0
Router(config-subif)#exit
Router(config)#int f0/0.20
Router(config-subif)#description vlan 20
Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip address 192.168.20.1 255.255.255.0
Router(config-subif)#exit
Router(config)#int f0/0.30
Router(config-subif)#description vlan 30
Router(config-subif)#encapsulation dot1q 30
Router(config-subif)#ip address 192.168.30.1 255.255.255.0
Router(config-subif)#exit
Router(config)#int f0/0.1000
Router(config-subif)#description NATIVE
Router(config-subif)#encapsulation dot1q 1000 native
Router(config-subif)#exit
Router(config)#int f0/0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#end
Router#show ip int br
```

Practical 6: Observe STP Topology Changes and Implement RSTP

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

Cable the network as shown in the topology.
Configure basic settings for each switch.

Switch D1

```
Switch>en
Switch#conf t
Enter configuration commands, one per
Switch(config)#hostname D1
spanning-tree mode pvst
banner motd # D1, STP Topology Change and RSTP Lab #
line con 0
exec-timeout 0 0
logging synchronous
exit
interface range g1/0/1-24, g1/1/1-4, g0/0
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 shutdown
exit
end
```

Switch D2

```
en
conf t
hostname D2
banner motd # D2, STP Topology Change and RSTP Lab #
spanning-tree mode pvst
line con 0
exec-timeout 0 0
logging synchronous
exit
interface range g1/0/1-24, g1/1/1-4, g0/0
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 shutdown
exit
end
```

Switch A1

```
en
conf t
hostname A1
banner motd # A1, STP Topology Change and RSTP Lab #
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 shutdown
exit
end
show ip int br
```

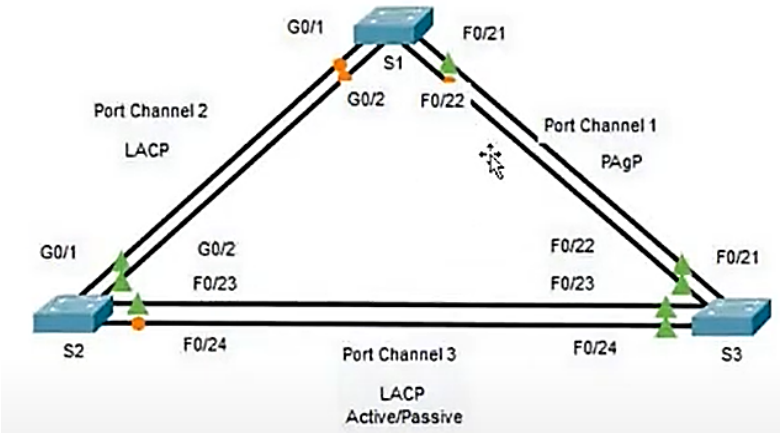
Find the root bridge.

```
A1# show spanning-tree root
A1# show version | include MAC

D1# show spanning-tree root active
D1# show version | include MAC

D2# show spanning-tree root active
D2# show version | include MAC
```

PRACTICAL 7: IMPLEMENT ETHERCHANNEL



Objectives

Part 1: Configure Basic Switch Settings

Part 2: Configure an EtherChannel with Cisco PAgP

Part 3: Configure an 802.3ad LACP EtherChannel

Part 4: Configure a Redundant EtherChannel Link

Background

Three switches have just been installed. There are redundant uplinks between the switches. As configured, only one of these links can be used; otherwise, a bridging loop might occur. However, using only one link utilizes only half of the available bandwidth. EtherChannel allows up to eight redundant links to be bundled together into one logical link. In this lab, you will configure Port Aggregation Protocol (PAgP), a Cisco EtherChannel protocol, and Link Aggregation Control Protocol (LACP), an IEEE 802.3ad open standard version of EtherChannel.

Port Channel Table

Channel Group	Ports	Protocol
1	S1 F0/21. F0/22 S3 F0/21, F0/22	PAgP
2	S1 G0/1, G0/2 S2 G0/1, G0/2	LACP
3	S2 F0/23, F0/24 S3 F0/23, F0/24	Negotiated LACP

PART 2

ON S1

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.
Switch(config)#hostname S1
S1(config)#int range g0/1-2,f0/21-22
S1(config-if-range)#switchport mode trunk
S1(config-if-range)#switchport nonegotiate
```

ON S2

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  En
Switch(config)#hostname S1
S1(config)#hostname S2
S2(config)#int range f0/23-24,g0/1-2
S2(config-if-range)#switchport mode trunk
S2(config-if-range)#switchport nonegotiate
S2(config-if-range)#exit
```

ON S3

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  En
Switch(config)#int range f0/21-24
Switch(config-if-range)#switchport mode trunk
Switch(config-if-range)#switchport nonegotiate
Switch(config-if-range)#exit
```

```
S1#show int trunk
```

ON S1

```
S1(config)# interface range f0/21 - 22
S1(config-if-range)# shutdown
S1(config-if-range)# channel-group 1 mode desirable
S1(config-if-range)# no shutdown
```

ON S3

```
S3(config)# interface range f0/21 - 22
S3(config-if-range)# shutdown
S3(config-if-range)# channel-group 1 mode desirable
S3(config-if-range)# no shutdown
```

The message “Creating a port-channel interface Port-channel 1” should appear on both switches when the channel-group is configured.

Configure the logical interface to become a trunk by first entering the **interface port-channel** *number* command and then the **switchport mode trunk** command. Add this configuration to both switches.

```
S1(config)# interface port-channel 1
S1(config-if)# switchport mode trunk

S3(config)# interface port-channel 1
S3(config-if)# switchport mode trunk
```

ON S1 & S3

```
S1(config)#interface port-channel 1
S1(config-if)#switchport mode trunk
S1(config-if)#exit
S1(config)#end
```

ON S1

```
S1#show etherchannel summary
```

PART 3

ON S1 & S2

```
S1(config)# interface range g0/1 - 2
S1(config-if-range)# shutdown
S1(config-if-range)# channel-group 2 mode active
S1(config-if-range)# no shutdown
S1(config-if-range)#exit
S1(config-if-range)# interface port-channel 2
S1(config-if)# switchport mode trunk
S1(config-if)#exit
S1(config)#end
```

ON S1

```
S1#show etherchannel summary
PAgP    Fa0/21(P) Fa0/22(P)
LACP    Gig0/1(P) Gig0/2(P)
```

PART 4

```
S2(config)# interface range f0/23 - 24
S2(config-if-range)# shutdown
S2(config-if-range)# channel-group 3 mode passive
S2(config-if-range)# no shutdown
S2(config-if-range)# exit
S2(config-if-range)# interface port-channel 3
S2(config-if)# switchport mode trunk
S2(config-if)# exit
```

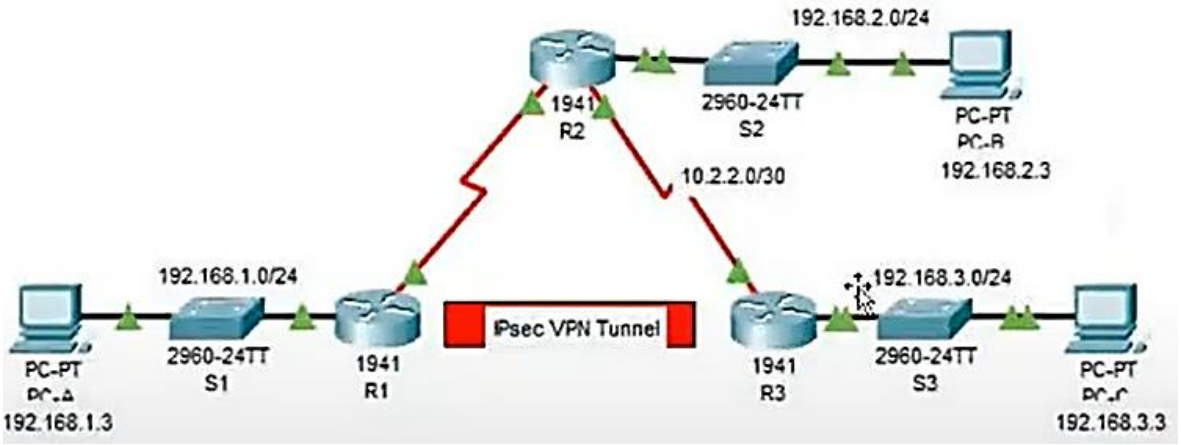
ON S2

```
S2(config-if)# end
S2# show etherchannel summary
```

Creating EtherChannel links does not prevent Spanning Tree from detecting switching loops. View the spanning tree status of the active ports on **S1**.

```
S1# show spanning-tree active
```

PRACTICAL 10: IPSEC SITE-TO-SITE VPN



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

Objectives

- Verify connectivity throughout the network.
- Configure R1 to support a site-to-site IPsec VPN with R3.

Part 1: Configure IPsec Parameters on R1

Step 1: Test connectivity.

Ping from PC-A to PC-C.

Step 2: Enable the Security Technology package.

a. On R1, issue the **show version** command to view the Security Technology package license information.

b. If the Security Technology package has not been enabled, use the following command to enable the package.

```
R1(config)# license boot module c1800 technology-package securityk9
```

c. Accept the end-user license agreement.

d. Save the running-config and reload the router to enable the security license.

e. Verify that the Security Technology package has been enabled by using the **show version** command.

Step 3: Identify interesting traffic on R1.

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

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

```
Router(config-router)#exit
Router(config)#access-list 110 permit ip 192.168.1.0 0.0.0.255
192.168.3.0 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 10.2.2.2
Router(config)#crypto ipsec transform-set VPN-SET esp-aes esp-sha-
hmac
Router(config)#hostname R1
R1(config)#crypto map VPN-MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
        and a valid access list have been configured.
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
```

Part 2: Configure IPsec Parameters on R3

Step 1: Enable the Security Technology package.

Step 2: Configure router R3 to support a site-to-site VPN with R1.

Step 3: Configure the IKE Phase 1 ISAKMP properties on R3.

Step 4: Configure the IKE Phase 2 IPsec policy on R3.

FIRST exit

```
R3(config)# access-list 110 permit ip 192.168.3.0 0.0.0.255 192.168.1.0
0.0.0.255
R3(config)# crypto isakmp policy 10
R3(config-isakmp)# encryption aes 256
R3(config-isakmp)# authentication pre-share
R3(config-isakmp)# group 5
R3(config-isakmp)# exit
R3(config)# crypto isakmp key vpnpa55 address 10.1.1.2
R3(config)# hostname R3
R3(config)# crypto ipsec transform-set VPN-SET esp-aes esp-sha-hmac
R3(config)# crypto map VPN-MAP 10 ipsec-isakmp
R3(config-crypto-map)# description VPN connection to R1
R3(config-crypto-map)# set peer 10.1.1.2
R3(config-crypto-map)# set transform-set VPN-SET
R3(config-crypto-map)# match address 110
R3(config-crypto-map)# exit
```

Step 5: Configure the crypto map on the outgoing interface.

Bind the VPN-MAP crypto map to the outgoing Serial 0/0/1 interface. **Note:** This is not graded.

```
R3(config)# interface s0/0/1
R3(config-if)# crypto map VPN-MAP
```

Part 3: Verify the IPsec VPN

```
R1(config-if)#end
R1#show crypto ipsec sa

interface: Serial0/0/0
    Crypto map tag: VPN-MAP, local addr 10.1.1.2
```