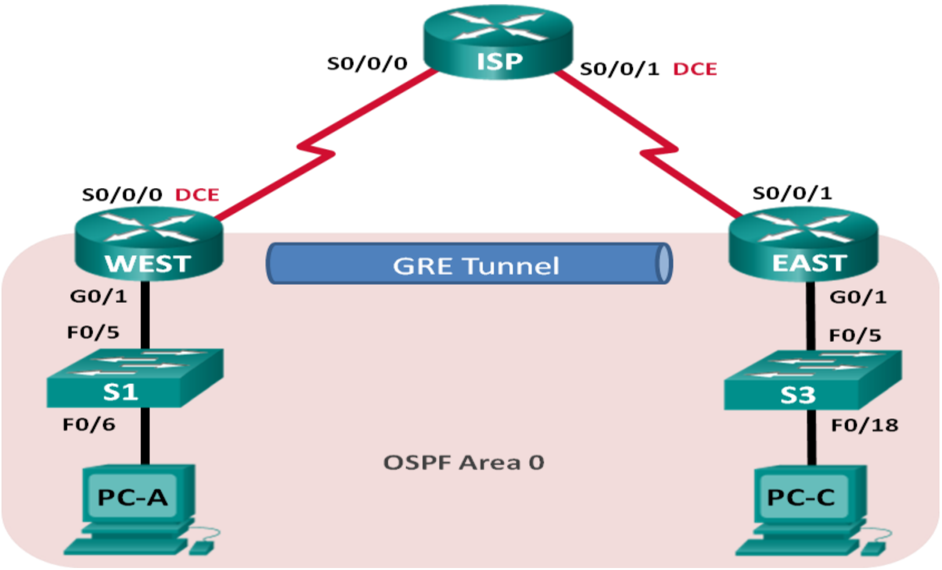
Practical 4

1. Aim : Configuring a Point-to-Point GRE VPN Tunnel



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
WEST	G0/1	172.16.1.1	255.255.255.0	N/A
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	N/A
	Tunnel0	172.16.12.1	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
EAST	G0/1	172.16.2.1	255.255.255.0	N/A
	S0/0/1	10.2.2.1	255.255.255.252	N/A
	Tunnel0	172.16.12.2	255.255.255.252	N/A
PC-A	NIC	172.16.1.3	255.255.255.0	172.16.1.1
PC-C	NIC	172.16.2.3	255.255.255.0	172.16.2.1

What is GRE?

→ Generic Routing Encapsulation (GRE) is a tunneling protocol developed by Cisco that can encapsulate a wide variety of network layer protocols into point-to-point connections. It allows the creation of a virtual point-to-point link between two network nodes over an existing network, such as the Internet. GRE is often used in conjunction with other protocols, such as IPsec, to provide a secure and encrypted communication channel.

Here are some key features and aspects of GRE:

1. Tunneling: GRE encapsulates a payload (which could be of any network layer protocol) within a GRE header. This encapsulated packet is then transmitted over an existing network.
2. Compatibility: GRE is protocol-independent, which means it can carry a variety of network layer protocols, including IP, IPX, and even non-IP protocols.
3. Routing: GRE does not have inherent encryption or security features. However, it is often used with IPsec to secure the data being transmitted through the tunnel. GRE itself focuses on providing a virtual point-to-point connection and encapsulation.
4. Dynamic Routing: GRE can carry multicast traffic and supports dynamic routing protocols, allowing it to participate in routing decisions.
5. Point-to-Point Connections: GRE is typically used to create point-to-point connections between two network nodes. It is not designed for point-to-multipoint or multipoint-to-multipoint connections.
6. Header Format: The GRE header includes information such as the key, which can be used for simple packet filtering, and the protocol type of the payload being carried.

While GRE has been widely used in various networking scenarios, it's worth noting that newer technologies, such as IPsec, have gained popularity for secure tunneling due to their built-in encryption features. Nonetheless, GRE remains relevant in certain contexts, especially when used in conjunction with other protocols to fulfill specific networking requirements.

Objectives

Part 1: Configure Basic Device Settings

Part 2: Configure a GRE Tunnel

Part 3: Enable Routing over the GRE Tunnel

Background / Scenario

Generic Routing Encapsulation (GRE) is a tunneling protocol that can encapsulate a variety of network layer protocols between two locations over a public network, such as the Internet.

GRE can be used with:

- Connecting IPv6 networks over IPv4 networks
- Multicast packets, such as OSPF, EIGRP, and streaming applications

In this lab, you will configure an unencrypted point-to-point GRE VPN tunnel and verify that network traffic is using the tunnel. You will also configure the OSPF routing protocol inside the GRE VPN tunnel. The GRE tunnel is between the WEST and EAST routers in OSPF area 0. The ISP has no knowledge of the GRE tunnel. Communication between the WEST and EAST routers and the ISP is accomplished using default static routes.

Note: The routers used with CCNA hands-on labs are Cisco 1941 Integrated Services Routers (ISRs) with Cisco IOS Release 15.2(4)M3 (universalk9 image). The switches used are Cisco Catalyst 2960s with Cisco IOS Release 15.0(2) (lanbasek9 image). Other routers, switches, and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and output produced might vary from what is shown in the labs. Refer to the Router Interface Summary Table at the end of this lab for the correct interface identifiers.

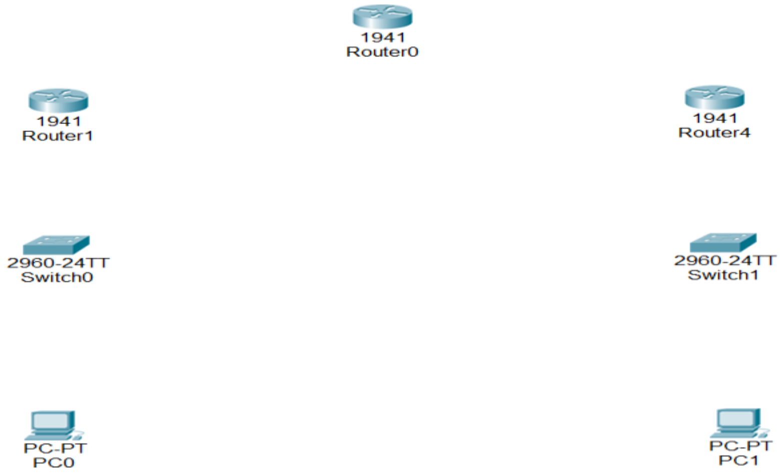
Required Resources

- 3 Routers (Cisco 1941 with Cisco IOS Release 15.2(4)M3 universal image or comparable)
- 2 Switches (Cisco 2960 with Cisco IOS Release 15.0(2) lanbasek9 image or comparable)

- 2 PCs (Windows 7, Vista, or XP with terminal emulation program, such as Tera Term)
- Console cables to configure the Cisco IOS devices via the console ports
- Ethernet and serial cables as shown in the topology

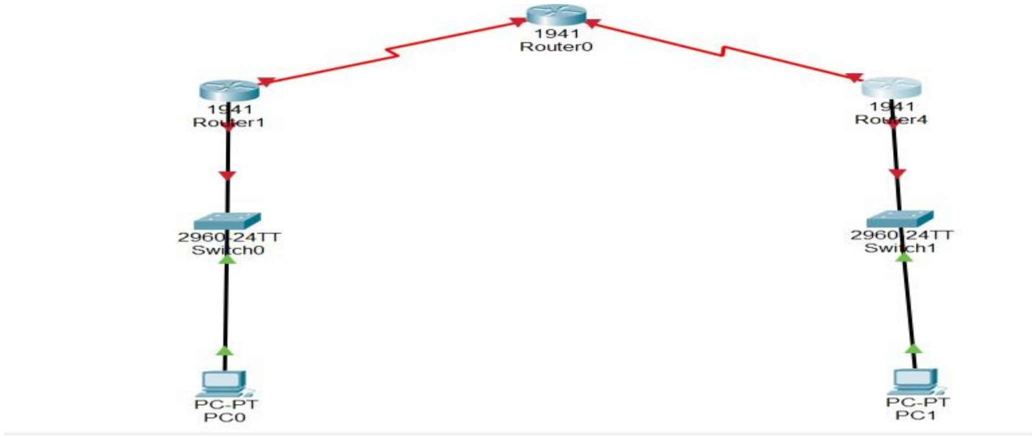
Part 1: Configure Basic Device Settings

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



sss

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



Step 2: Initialize and reload the routers and switches.

Step 3: Configure basic settings for each router.

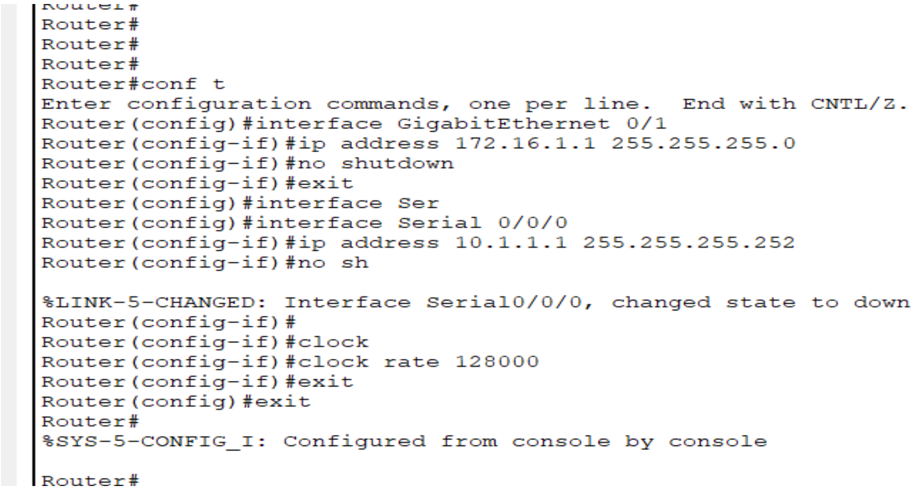
a. Disable DNS lookup.

b. Configure the device names.

c. Apply IP addresses to Serial and Gigabit Ethernet interfaces according to the Addressing Table and activate the physical interfaces. Do NOT configure the Tunnel0 interfaces at this time.

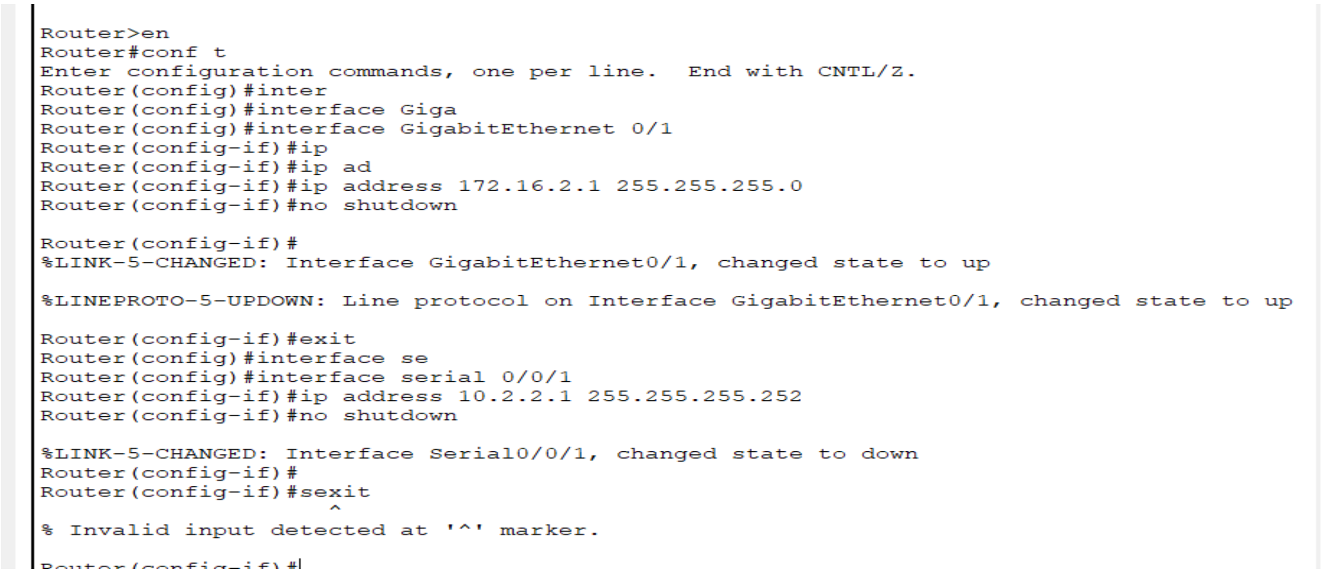
d. Set the clock rate to **128000** for DCE serial interfaces.

West Router :



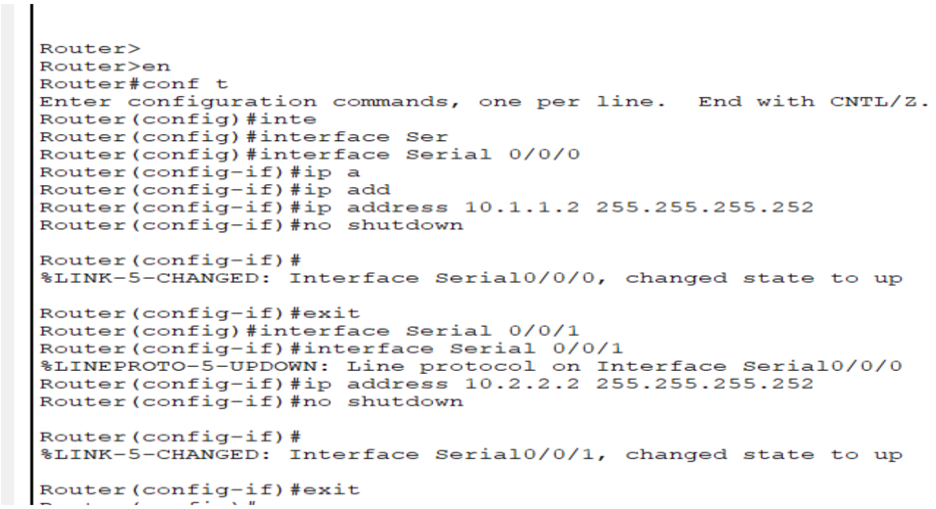
```
Router#
Router#
Router#
Router#
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface GigabitEthernet 0/1
Router(config-if)#ip address 172.16.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface Ser
Router(config)#interface Serial 0/0/0
Router(config-if)#ip address 10.1.1.1 255.255.255.252
Router(config-if)#no sh

%LINK-5-CHANGED: Interface Serial0/0/0, changed state to down
Router(config-if)#
Router(config-if)#clock
Router(config-if)#clock rate 128000
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
```

East Router :

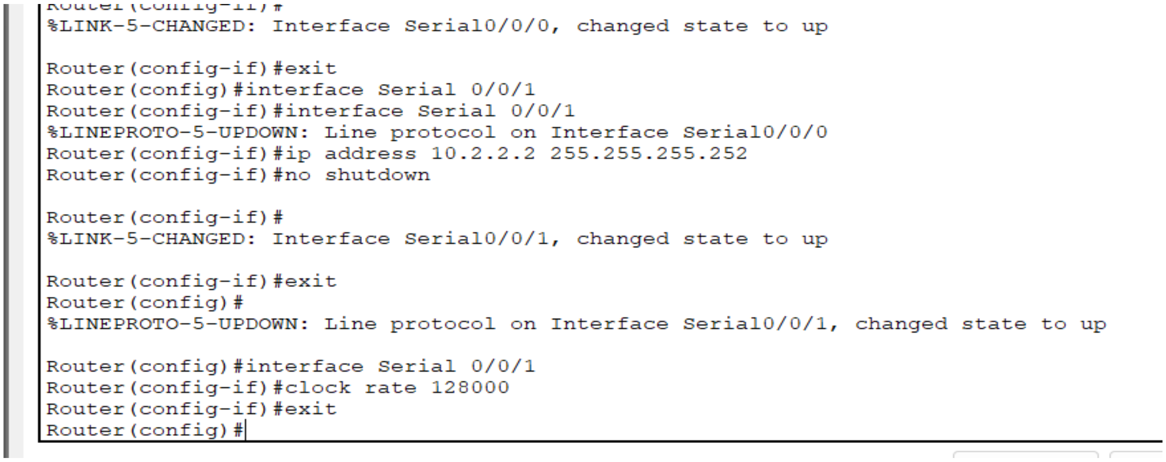


```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#inter
Router(config)#interface Giga
Router(config)#interface GigabitEthernet 0/1
Router(config-if)#ip
Router(config-if)#ip ad
Router(config-if)#ip address 172.16.2.1 255.255.255.0
Router(config-if)#no shutdown

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

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

Router(config-if)#exit
Router(config)#interface se
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 10.2.2.1 255.255.255.252
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/0/1, changed state to down
Router(config-if)#
Router(config-if)#sexit
                   ^
% Invalid input detected at '^' marker.

Router(config-if)#
```

ISP Router :



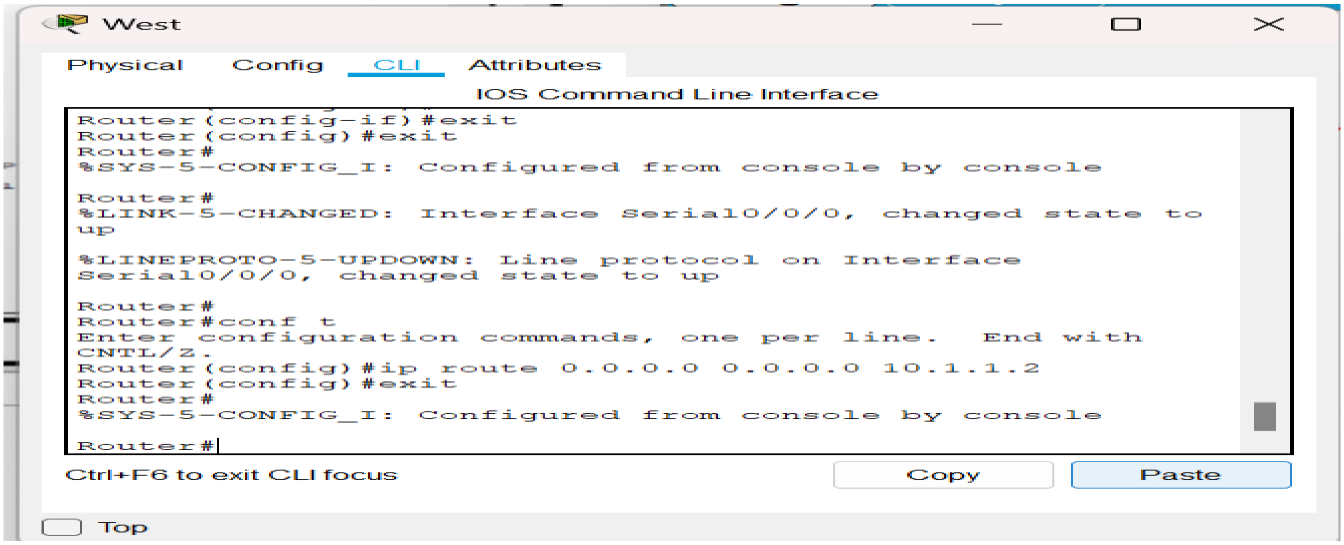
```
Router>
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#inte
Router(config)#interface Ser
Router(config)#interface Serial 0/0/0
Router(config-if)#ip a
Router(config-if)#ip add
Router(config-if)#ip address 10.1.1.2 255.255.255.252
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

Router(config-if)#exit
Router(config)#interface Serial 0/0/1
Router(config-if)#interface Serial 0/0/1
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0
Router(config-if)#ip address 10.2.2.2 255.255.255.252
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/1, changed state to up

Router(config-if)#exit
```



```
Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

Router(config-if)#exit
Router(config)#interface Serial 0/0/1
Router(config-if)#interface Serial 0/0/1
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0
Router(config-if)#ip address 10.2.2.2 255.255.255.252
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/1, changed state to up

Router(config-if)#exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/1, changed state to up

Router(config)#interface Serial 0/0/1
Router(config-if)#clock rate 128000
Router(config-if)#exit
Router(config)#
```

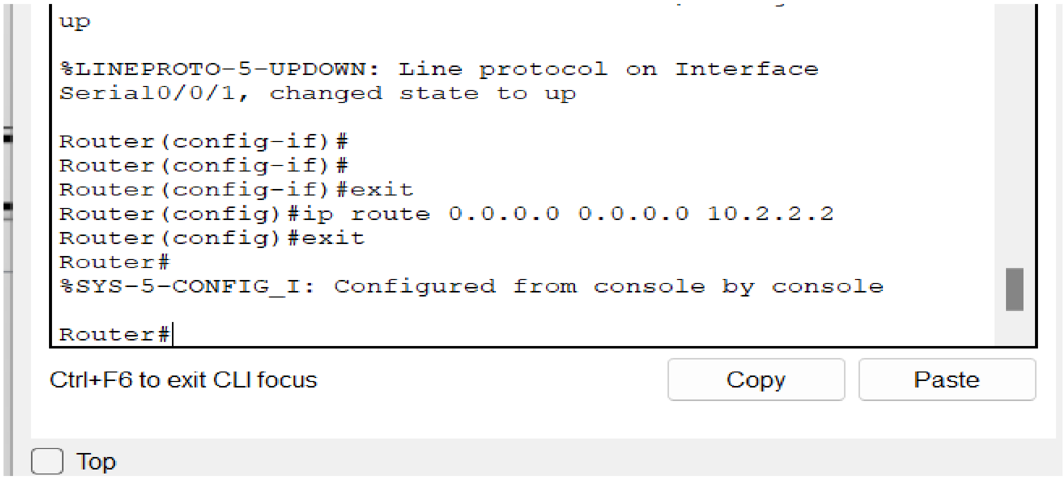
Step 4: Configure default routes to the ISP router.

WEST(config)# **ip route 0.0.0.0 0.0.0.0 10.1.1.2**



```
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to
up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to up

Router#
Router#conf t
Enter configuration commands, one per line.  End with
CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 10.1.1.2
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
```

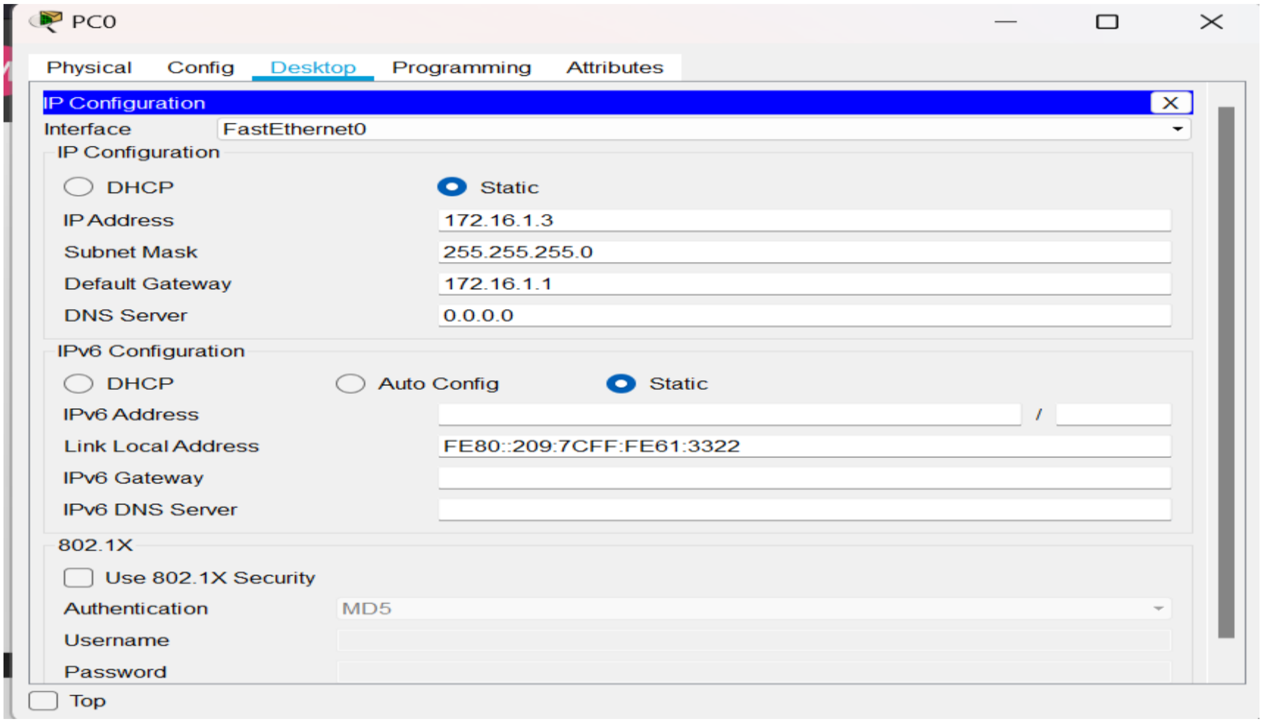
EAST(config)# **ip route 0.0.0.0 0.0.0.0 10.2.2.2**



```
up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up

Router(config-if)#
Router(config-if)#
Router(config-if)#exit
Router(config)#ip route 0.0.0.0 0.0.0.0 10.2.2.2
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
```

Step 5: Configure the PCs.

Assign IP addresses and default gateways to the PCs according to the Addressing Table.



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IP Address: 172.16.1.3

Subnet Mask: 255.255.255.0

Default Gateway: 172.16.1.1

DNS Server: 0.0.0.0

IPv6 Configuration

○ DHCP ○ Auto Config ● Static

IPv6 Address: /

Link Local Address: FE80::209:7CFF:FE61:3322

IPv6 Gateway:

IPv6 DNS Server:

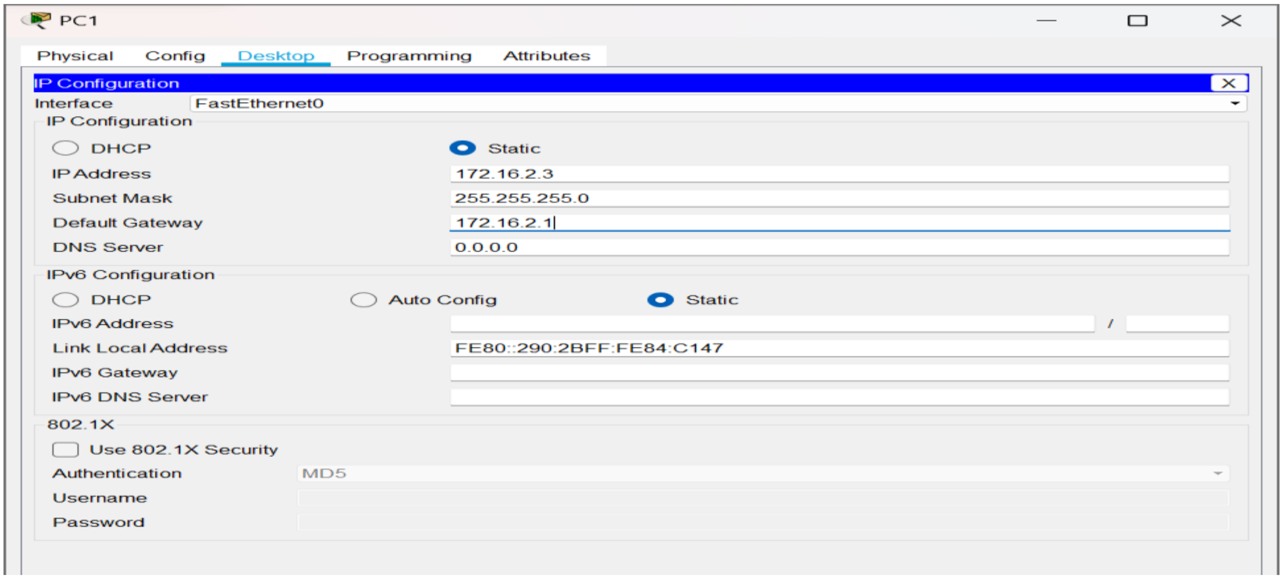
802.1X

□ Use 802.1X Security

Authentication: MD5

Username:

Password:



IP Configuration

Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IP Address: 172.16.2.3

Subnet Mask: 255.255.255.0

Default Gateway: 172.16.2.1

DNS Server: 0.0.0.0

IPv6 Configuration

○ DHCP ○ Auto Config ● Static

IPv6 Address: /

Link Local Address: FE80::290:2BFF:FE84:C147

IPv6 Gateway:

IPv6 DNS Server:

802.1X

□ Use 802.1X Security

Authentication: MD5

Username:

Password:

Step 6: Verify connectivity.

Step 7: Save your running configuration.

Part 2: Configure a GRE Tunnel

In Part 2, you will configure a GRE tunnel between the WEST and EAST routers.

Step 1: Configure the GRE tunnel interface.

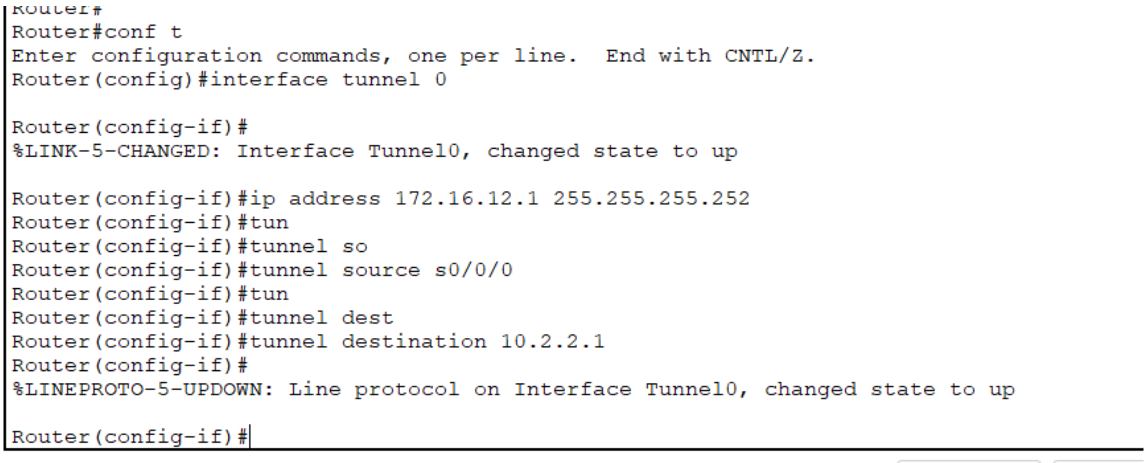
a. Configure the tunnel interface on the WEST router. Use S0/0/0 on WEST as the tunnel source interface and 10.2.2.1 as the tunnel destination on the EAST router.

WEST(config)# **interface tunnel 0**

WEST(config-if)# **ip address 172.16.12.1 255.255.255.252**

WEST(config-if)# **tunnel source s0/0/0**

WEST(config-if)# **tunnel destination 10.2.2.1**



```
Router#
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface tunnel 0

Router(config-if)#
%LINK-5-CHANGED: Interface Tunnel0, changed state to up

Router(config-if)#ip address 172.16.12.1 255.255.255.252
Router(config-if)#tun
Router(config-if)#tunnel so
Router(config-if)#tunnel source s0/0/0
Router(config-if)#tun
Router(config-if)#tunnel dest
Router(config-if)#tunnel destination 10.2.2.1
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to up

Router(config-if)#
```

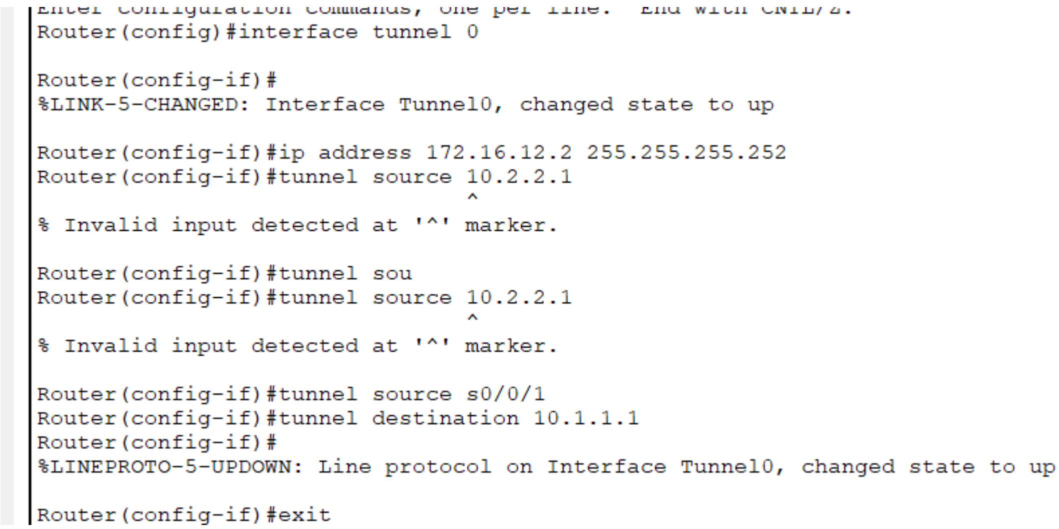
b. Configure the tunnel interface on the EAST router. Use S0/0/1 on EAST as the tunnel source interface and 10.1.1.1 as the tunnel destination on the WEST router.

EAST(config)# **interface tunnel 0**

EAST(config-if)# **ip address 172.16.12.2 255.255.255.252**

EAST(config-if)# **tunnel source s0/0/1**

EAST(config-if)# **tunnel destination 10.1.1.1**



```
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface tunnel 0

Router(config-if)#
%LINK-5-CHANGED: Interface Tunnel0, changed state to up

Router(config-if)#ip address 172.16.12.2 255.255.255.252
Router(config-if)#tunnel source 10.2.2.1
                                ^
% Invalid input detected at '^' marker.

Router(config-if)#tunnel sou
Router(config-if)#tunnel source 10.2.2.1
                                ^
% Invalid input detected at '^' marker.

Router(config-if)#tunnel source s0/0/1
Router(config-if)#tunnel destination 10.1.1.1
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel0, changed state to up

Router(config-if)#exit
```

Part 3: Enable Routing over the GRE Tunnel

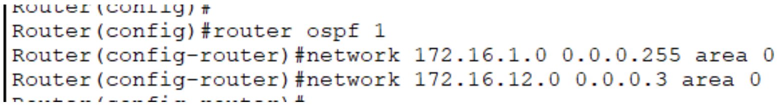
Step 1: Configure OSPF routing for area 0 over the tunnel.

a. Configure OSPF process ID 1 using area 0 on the WEST router for the 172.16.1.0/24 and 172.16.12.0/24 networks.

WEST(config)# **router ospf 1**

WEST(config-router)# **network 172.16.1.0 0.0.0.255 area 0**

WEST(config-router)# **network 172.16.12.0 0.0.0.3 area 0**



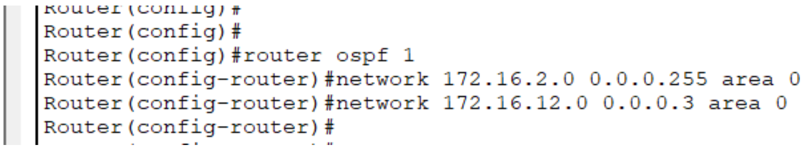
```
Router(config)#
Router(config)#router ospf 1
Router(config-router)#network 172.16.1.0 0.0.0.255 area 0
Router(config-router)#network 172.16.12.0 0.0.0.3 area 0
Router(config-router)#
```

b. Configure OSPF process ID 1 using area 0 on the EAST router for the 172.16.2.0/24 and 172.16.12.0/24 networks.

EAST(config)# **router ospf 1**

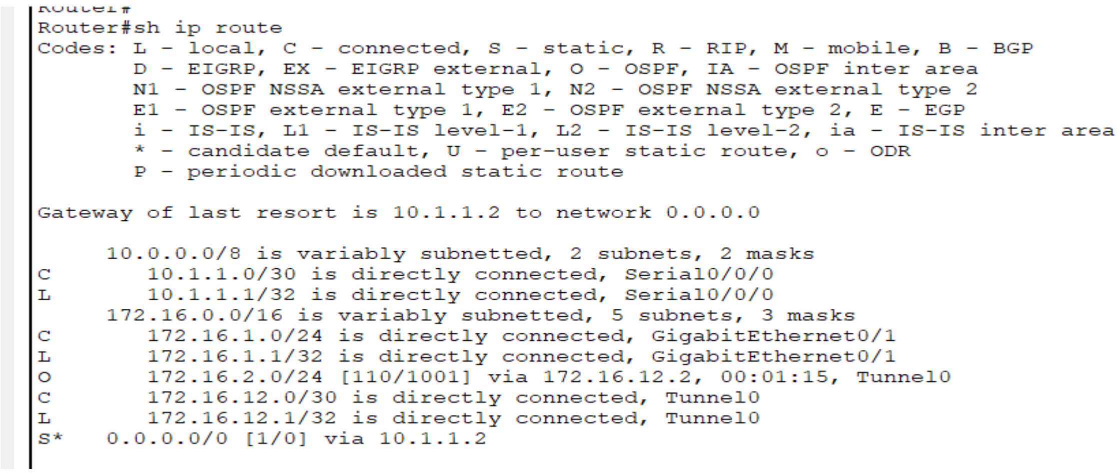
EAST(config-router)# **network 172.16.2.0 0.0.0.255 area 0**

EAST(config-router)# **network 172.16.12.0 0.0.0.3 area 0**



```
Router(config)#
Router(config)#
Router(config)#router ospf 1
Router(config-router)#network 172.16.2.0 0.0.0.255 area 0
Router(config-router)#network 172.16.12.0 0.0.0.3 area 0
Router(config-router)#
```

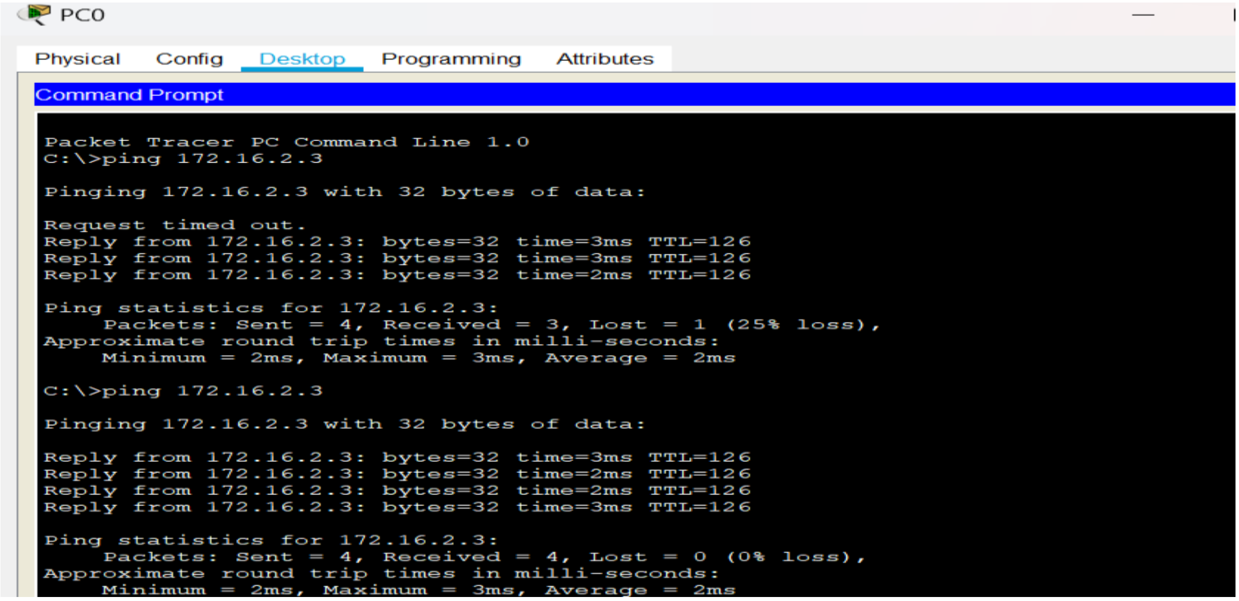
Check Route :



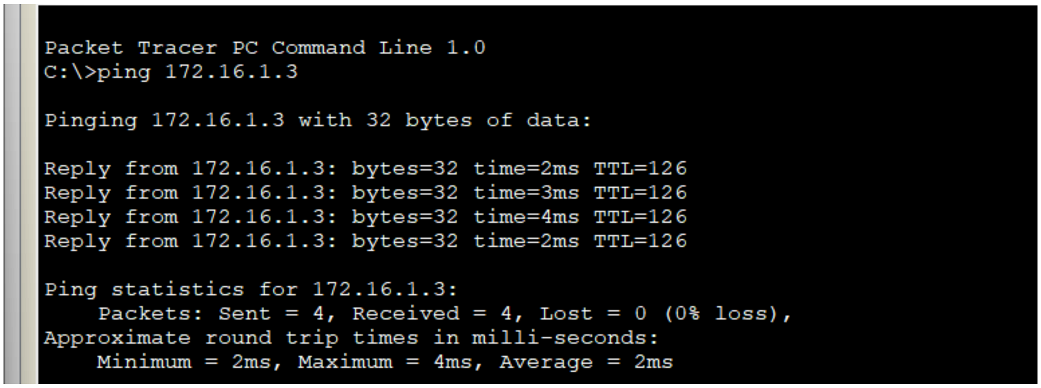
```
Router#
Router#sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type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

Gateway of last resort is 10.1.1.2 to network 0.0.0.0

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.1.1.0/30 is directly connected, Serial0/0/0
L       10.1.1.1/32 is directly connected, Serial0/0/0
     172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
C       172.16.1.0/24 is directly connected, GigabitEthernet0/1
L       172.16.1.1/32 is directly connected, GigabitEthernet0/1
O       172.16.2.0/24 [110/1001] via 172.16.12.2, 00:01:15, Tunnel0
C       172.16.12.0/30 is directly connected, Tunnel0
L       172.16.12.1/32 is directly connected, Tunnel0
S*   0.0.0.0/0 [1/0] via 10.1.1.2
```

Step 2: Verify end-to-end connectivity.

a. Ping from PC-A to PC-C. It should be successful. If not, troubleshoot until you have end-to-end connectivity.

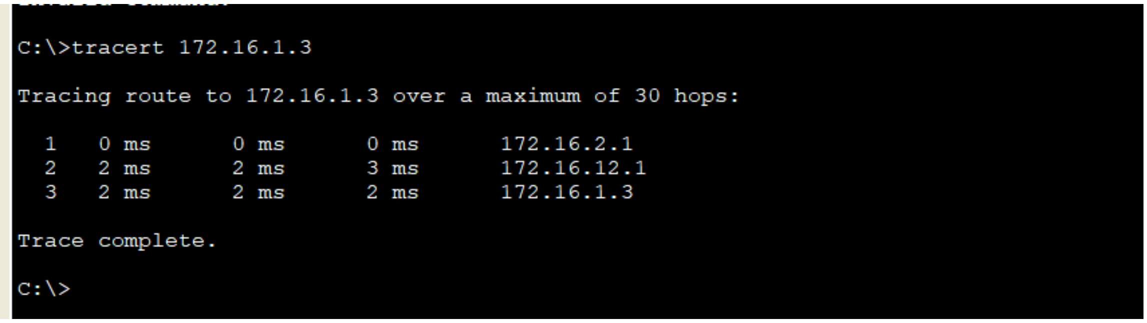


```
Packet Tracer PC Command Line 1.0
C:\>ping 172.16.2.3

Pinging 172.16.2.3 with 32 bytes of data:

Request timed out.
Reply from 172.16.2.3: bytes=32 time=3ms TTL=126
Reply from 172.16.2.3: bytes=32 time=3ms TTL=126
Reply from 172.16.2.3: bytes=32 time=2ms TTL=126

Ping statistics for 172.16.2.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 3ms, Average = 2ms

C:\>ping 172.16.2.3

Pinging 172.16.2.3 with 32 bytes of data:

Reply from 172.16.2.3: bytes=32 time=3ms TTL=126
Reply from 172.16.2.3: bytes=32 time=2ms TTL=126
Reply from 172.16.2.3: bytes=32 time=2ms TTL=126
Reply from 172.16.2.3: bytes=32 time=3ms TTL=126

Ping statistics for 172.16.2.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 3ms, Average = 2ms
```



```
Packet Tracer PC Command Line 1.0
C:\>ping 172.16.1.3

Pinging 172.16.1.3 with 32 bytes of data:

Reply from 172.16.1.3: bytes=32 time=2ms TTL=126
Reply from 172.16.1.3: bytes=32 time=3ms TTL=126
Reply from 172.16.1.3: bytes=32 time=4ms TTL=126
Reply from 172.16.1.3: bytes=32 time=2ms TTL=126

Ping statistics for 172.16.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 4ms, Average = 2ms
```

b. Traceroute from PC-A to PC-C. What is the path from PC-A to PC-C?



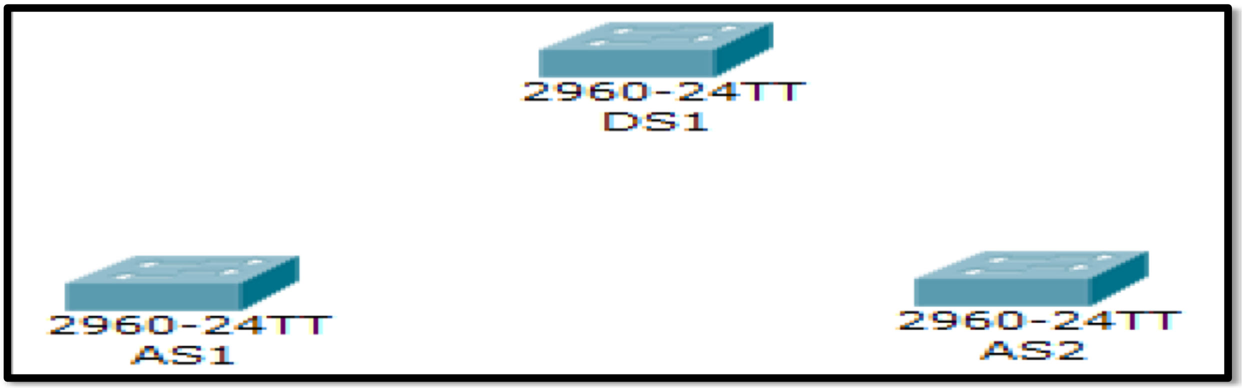
```
C:\>tracert 172.16.1.3

Tracing route to 172.16.1.3 over a maximum of 30 hops:

  1   0 ms      0 ms      0 ms      172.16.2.1
  2   2 ms      2 ms      3 ms      172.16.12.1
  3   2 ms      2 ms      2 ms      172.16.1.3

Trace complete.

C:\>
```

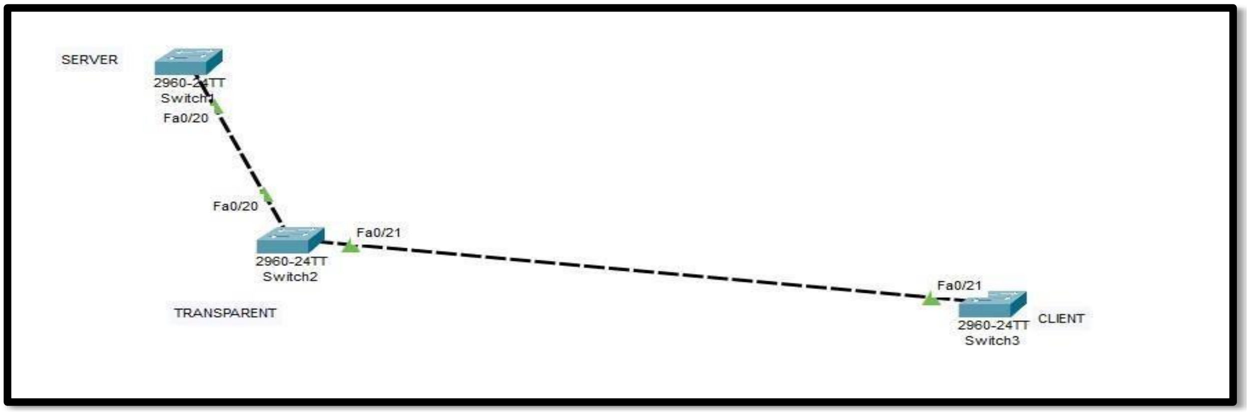
Router Interface Summary Table

Router Interface Summary				
Router Model	**Ethernet Interface #1**	**Ethernet Interface #2**	**Serial Interface #1**	**Serial Interface #2**
1800	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
2811	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

2. **Aim : Implement VTP**

Topology





What is VTP?

VTP is a Layer 2 messaging protocol that was designed to manage the creation and deletion of VLANs and maintain network-wide VLAN database consistency. Using this protocol, a network administrator can add or delete VLANs and have those changes automatically propagated to all other switches in the network. Without VTP, switches do not exchange information about VLANs. The protocol has been designed around the centralized management idea. One or more switches are assigned the role of VTP Server. Any updates made on these switches are sent through VTP to the other switches, which are VTP Clients

What are the modes of VTP in networking?

There are three modes of VTP in networking, namely-

- VTP Server Mode
- VTP Client Mode
- VTP Transparent Mode

Objectives

In this activity, you will configure VTP

Background / Scenario

Scalability and management are two crucial considerations in large networks. VTP and DTP are technologies that improve management and scalability. VLAN Trunking Protocol (VTP) allows the switches to communicate over VLANs automatically, improving management and scalability. Dynamic Trunking Protocol (DTP) allows the switches to automatically negotiate and establish trunk links. DTP also improves scalability.

Step 1: Using Dynamic Trunk Protocol (DTP) to form trunk links

Access links transport single VLAN frames and trunk links carry frames belonging to multiple VLANs. While trunk links can be manually configured, DTP can be used to allow the switches to negotiate and establish trunk links automatically. DTP is very helpful in large networks

Configure Switches:- show vtp status en conf t hostname s1 int f0/20 switchport mode trunk

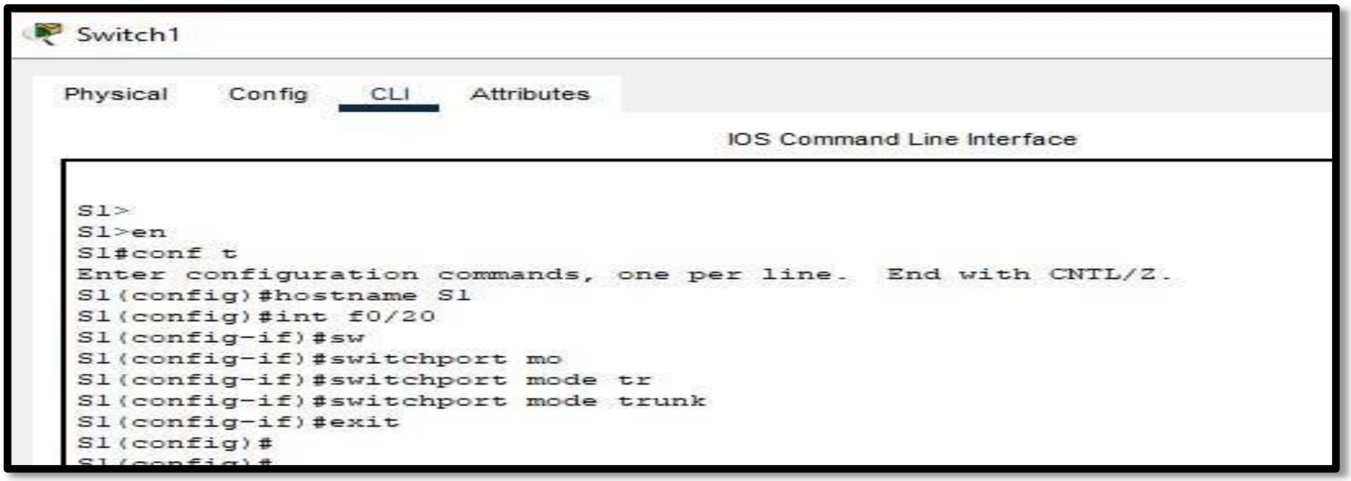
exit

vtp domain CCIE vtp password

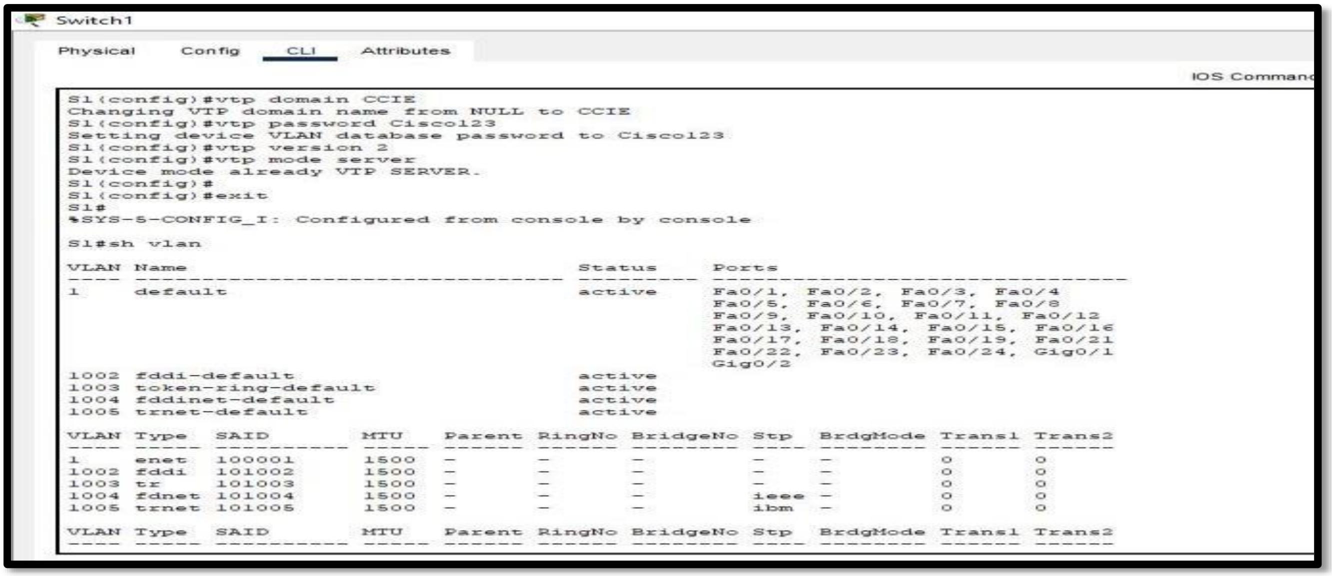
Cisco123 vtp version 2 vtp mode

server

exit



```
S1>
S1>en
S1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S1(config)#hostname S1
S1(config)#int f0/20
S1(config-if)#sw
S1(config-if)#switchport mo
S1(config-if)#switchport mode tr
S1(config-if)#switchport mode trunk
S1(config-if)#exit
S1(config)#
S1(config)#
```



```
S1(config)#vtp domain CCIE
Changing VTP domain name from NULL to CCIE
S1(config)#vtp password Cisco123
Setting device VLAN database password to Cisco123
S1(config)#vtp version 2
S1(config)#vtp mode server
Device mode already VTP SERVER.
S1(config)#
S1(config)#exit
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#sh vlan

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

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0
1003 tr    101003     1500  -      -      -        -    -        0      0
1004 fdnet 101004     1500  -      -      -        ieee -        0      0
1005 trnet 101005     1500  -      -      -        ibm  -        0      0

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
```

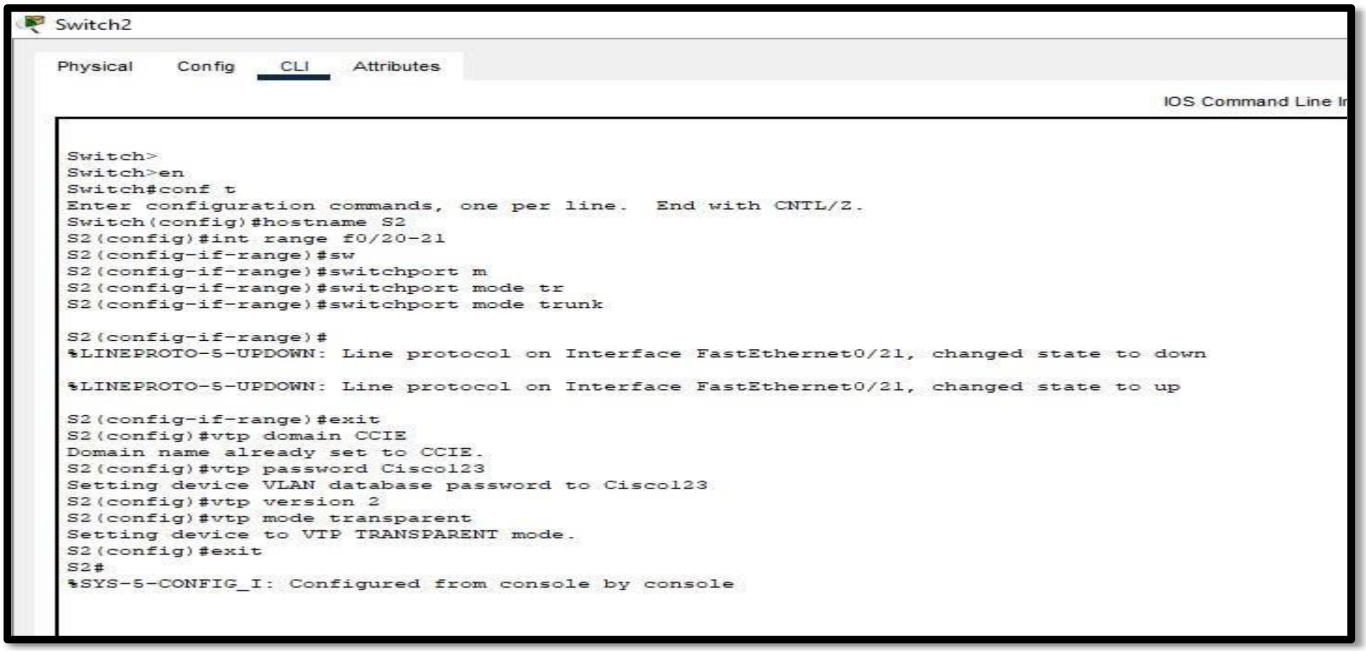
en conf t hostname s2 int f0/20-21

switchport mode trunk exit

vtp domain CCIE vtp password

Cisco123 vtp version 2 vtp mode server

exit



```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S2
S2(config)#int range f0/20-21
S2(config-if-range)#sw
S2(config-if-range)#switchport m
S2(config-if-range)#switchport mode tr
S2(config-if-range)#switchport mode trunk

S2(config-if-range)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/21, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/21, changed state to up

S2(config-if-range)#exit
S2(config)#vtp domain CCIE
Domain name already set to CCIE.
S2(config)#vtp password Cisco123
Setting device VLAN database password to Cisco123
S2(config)#vtp version 2
S2(config)#vtp mode transparent
Setting device to VTP TRANSPARENT mode.
S2(config)#exit
S2#
%SYS-5-CONFIG_I: Configured from console by console
```

en

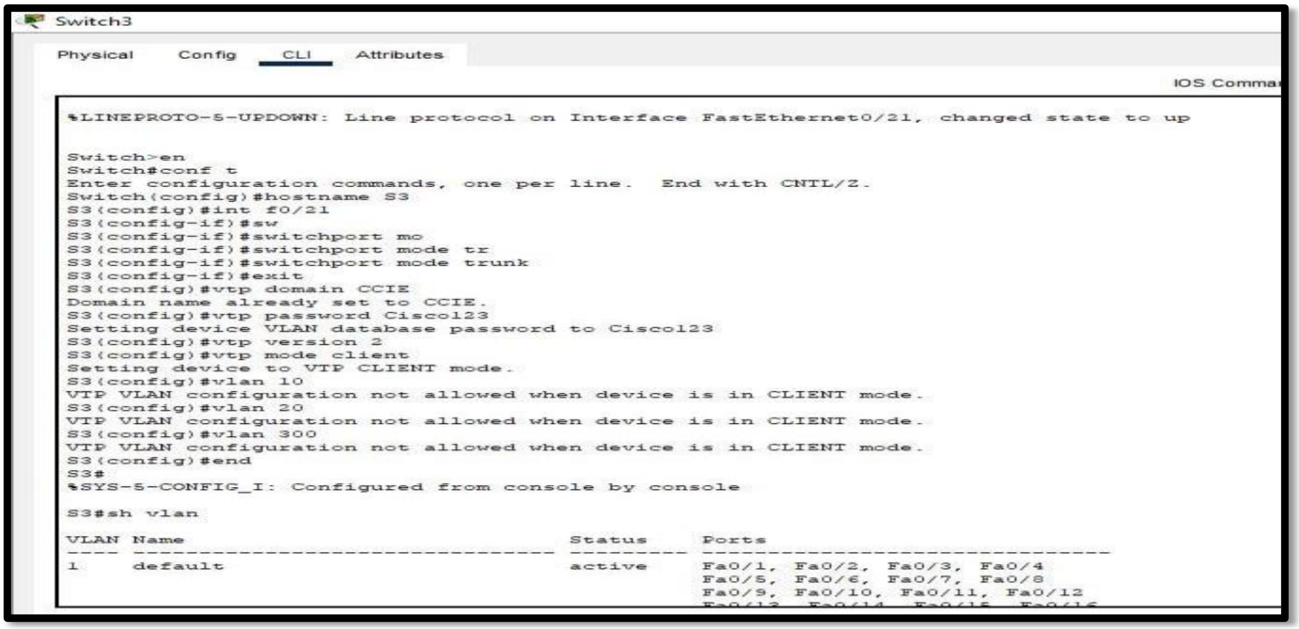
conf t hostname s3 int f0/21 switchport

mode trunk exit

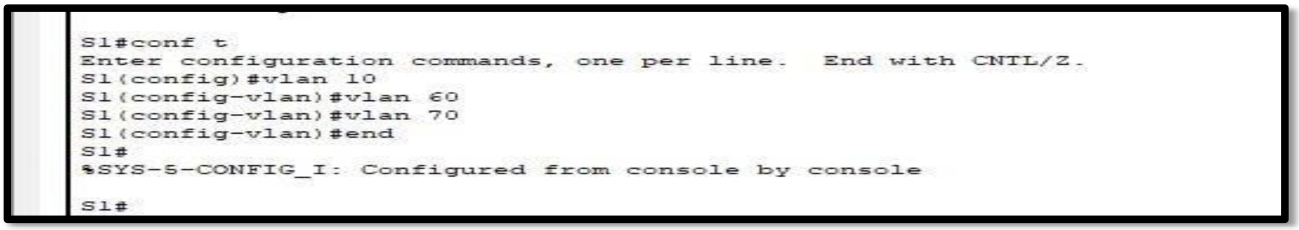
vtp domain CCIE vtp password

Cisco123 vtp version 2 vtp mode server

exit



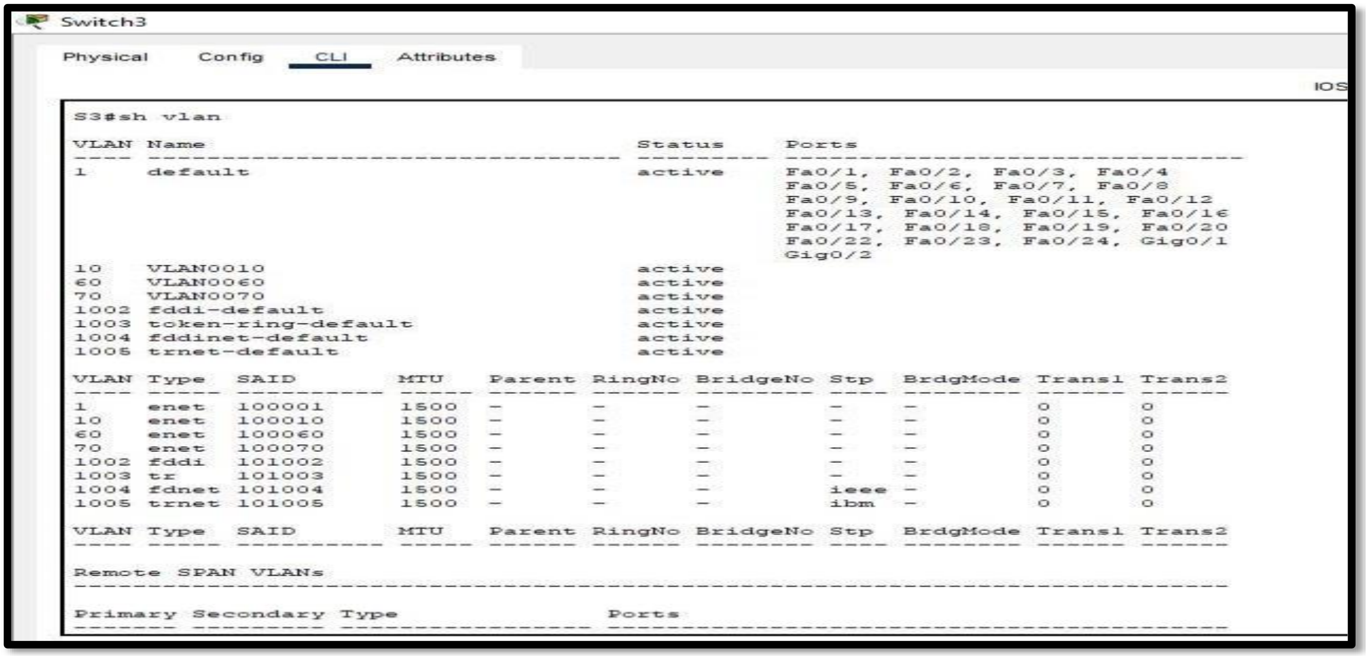
```
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/21, changed state to up


Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S3
S3(config)#int f0/21
S3(config-if)#sw
S3(config-if)#switchport mo
S3(config-if)#switchport mode tr
S3(config-if)#switchport mode trunk
S3(config-if)#exit
S3(config)#vtp domain CCIE
Domain name already set to CCIE.
S3(config)#vtp password Cisco123
Setting device VLAN database password to Cisco123
S3(config)#vtp version 2
S3(config)#vtp mode client
Setting device to VTP CLIENT mode.
S3(config)#vlan 10
VTP VLAN configuration not allowed when device is in CLIENT mode.
S3(config)#vlan 20
VTP VLAN configuration not allowed when device is in CLIENT mode.
S3(config)#vlan 300
VTP VLAN configuration not allowed when device is in CLIENT mode.
S3(config)#end
S3#
%SYS-5-CONFIG_I: Configured from console by console

S3#sh vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
```



```
S1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S1(config)#vlan 10
S1(config-vlan)#vlan 60
S1(config-vlan)#vlan 70
S1(config-vlan)#end
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#
```

Check VLAN'S are created or not

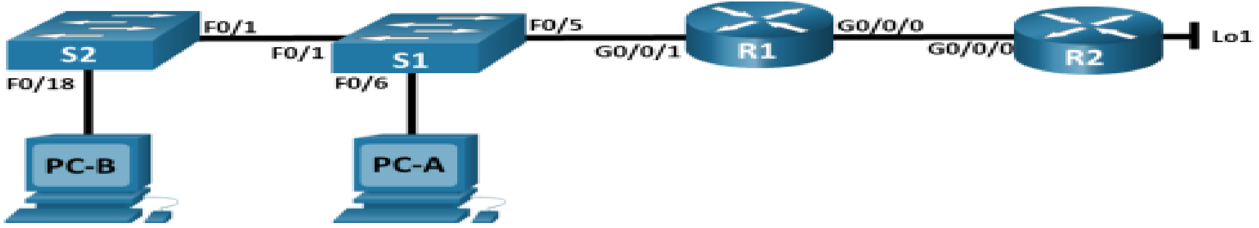
Sh vlan



```
S3#sh vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                                Fa0/22, Fa0/23, Fa0/24, Gig0/1
                                                Gig0/2
10   VLAN0010                         active
60   VLAN0060                         active
70   VLAN0070                         active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
10   enet  100010     1500  -      -      -        -    -        0      0
60   enet  100060     1500  -      -      -        -    -        0      0
70   enet  100070     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0
1003 tr    101003     1500  -      -      -        -    -        0      0
1004 fdnet 101004     1500  -      -      -        ieee -        0      0
1005 trnet 101005     1500  -      -      -        ibm  -        0      0

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------

Remote SPAN VLANs
------------------------------------------------------------------------------

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```

3. Aim : Implement NAT

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask
R1	G0/0/0	209.165.200.230	255.255.255.248
	G0/0/1	192.168.1.1	255.255.255.0
R2	G0/0/0	209.165.200.225	255.255.255.248
	Lo1	209.165.200.1	255.255.255.224
S1	VLAN 1	192.168.1.11	255.255.255.0
S2	VLAN 1	192.168.1.12	255.255.255.0
PC-A	NIC	192.168.1.2	255.255.255.0
PC-B	NIC	192.168.1.3	255.255.255.0

What is NAT?

Network Address Translation (NAT) is a service that enables private IP networks to use the internet and cloud. NAT translates private IP addresses in an internal network to a public IP address before packets are sent to an external network.

What is Static NAT?

Static NAT (Network Address Translation) - Static NAT (Network Address Translation) is one-to-one mapping of a private IP address to a public IP address.

What is Dynamic NAT?

Dynamic NAT (Network Address Translation) - Dynamic NAT can be defined as mapping of a private IP address to a public IP address from a group of public IP addresses called as NAT pool. The public to private mapping may vary based on the available public IP address in NAT pool.

What is PAT?

PAT (Port Address Translation) - Port Address Translation (PAT) is another type of dynamic NAT which can map multiple private IP addresses to a single public IP address by using a technology known as Port Address Translation.

Objectives

Part 1: Build the Network and Configure Basic Device Settings

Part 2: Configure and verify NAT for IPv4

Part 3: Configure and verify PAT for IPv4

Part 4: Configure and verify Static NAT for IPv4

Background / Scenario

Network Address Translation (NAT) is the process where a network device, such as a Cisco router, assigns a public address to host devices inside a private network . An ISP has allocated the public IP address space of 209.165.200.224/29 to a company. This network is used to address the link between the ISP router (R2) and the company gateway (R1). The first address (209.165.200.225) is assigned to the g0/0/0 interface on R2 and the last address (209.165.200.230) is assigned to the g0/0/0 interface on R1. The remaining addresses (209.165.200.226-209.165.200.229) will be used to provide internet access to the company hosts. A default route is used from R1 to R2. The internet is simulated by a loopback address on R2.

Required Resources

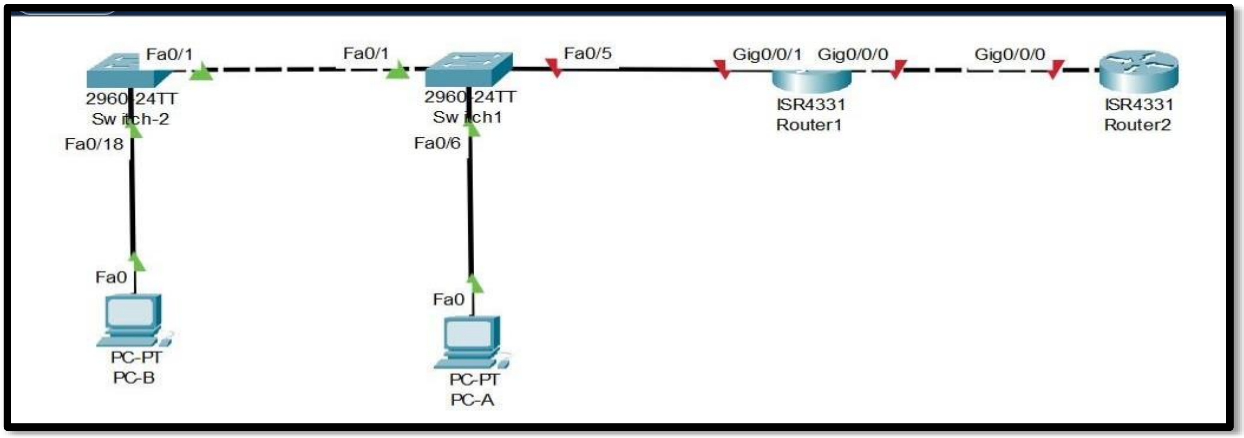
- 2 Routers (Cisco 4221 with Cisco IOS XE Release 16.9.4 universal image or comparable)
- 2 Switches (Cisco 2960 with Cisco IOS Release 15.2(2) lanbasek9 image or comparable)
- 2 PCs (Windows with a terminal emulation program, such as Tera Term)
- Console cables to configure the Cisco IOS devices via the console ports
- Ethernet cables as shown in the topology

Instructions

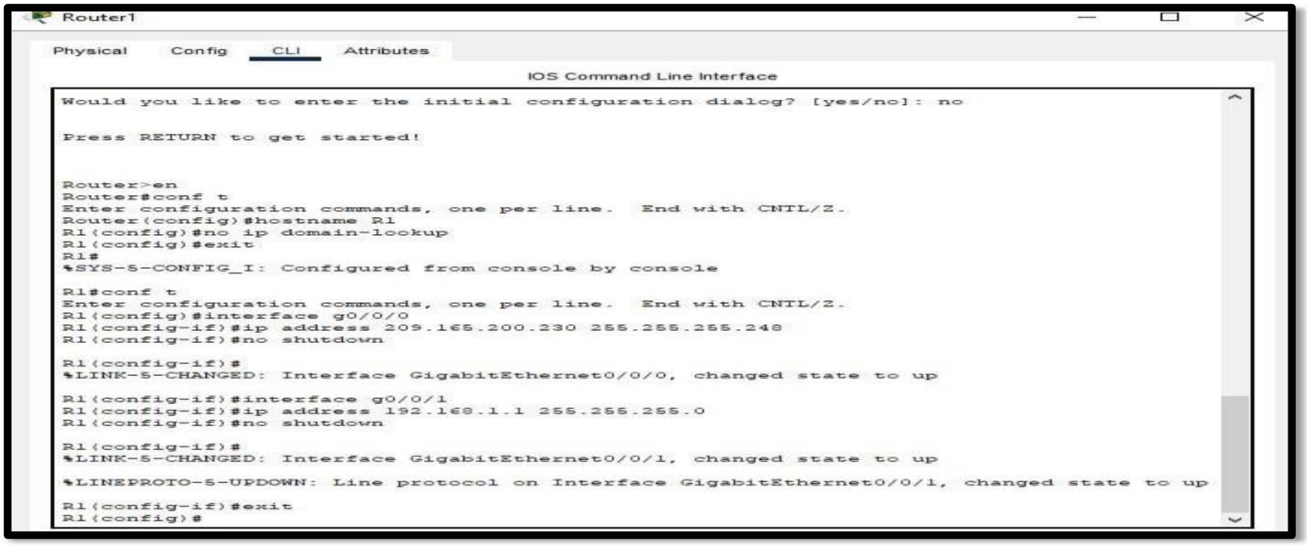
Part 1: Build the Network and Configure Basic Device Settings

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

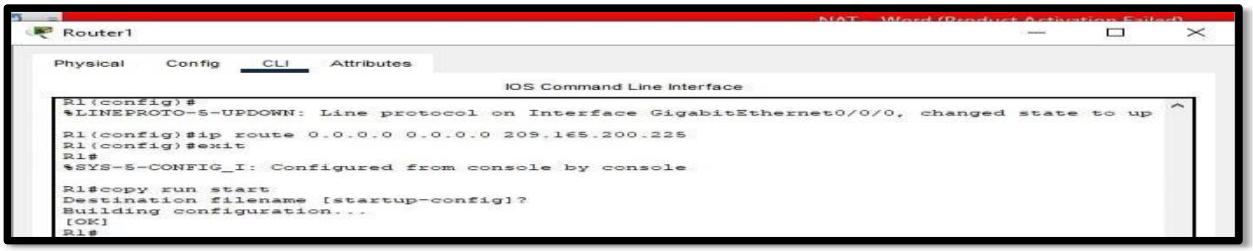
Attach the devices as shown in the topology diagram and cable as necessary.



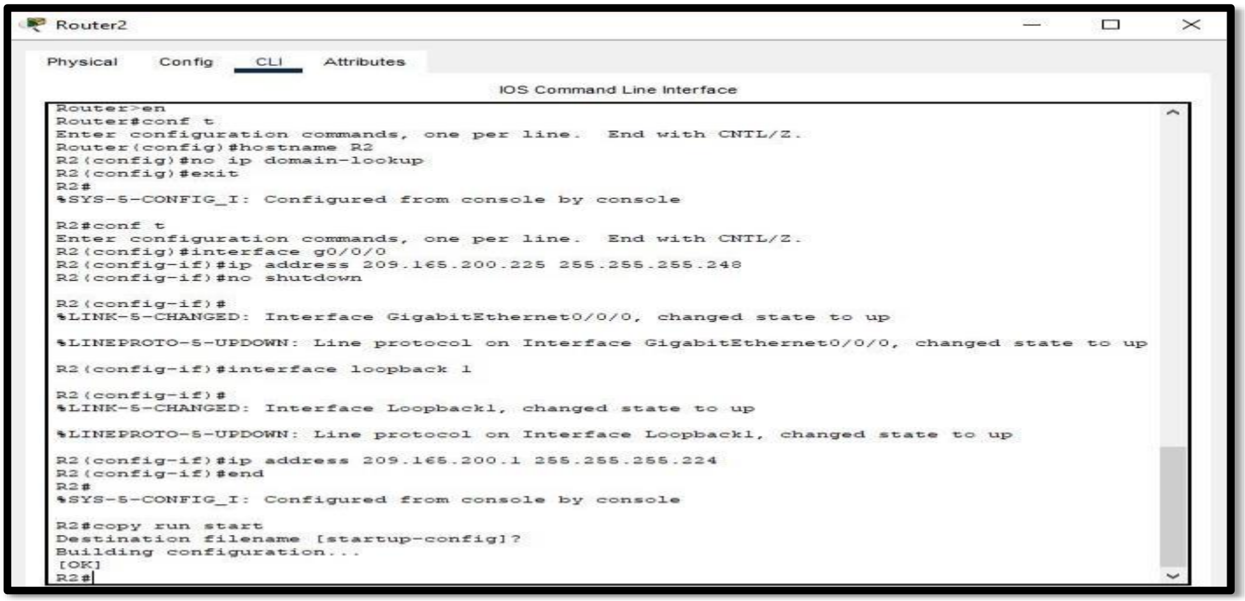
Step 2: Configure basic settings for each router.



```
Would you like to enter the initial configuration dialog? [yes/no]: no

Press RETURN to get started!


Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R1
R1(config)#no ip domain-lookup
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface g0/0/0
R1(config-if)#ip address 209.165.200.230 255.255.255.248
R1(config-if)#no shutdown

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

R1(config-if)#interface g0/0/1
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shutdown

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

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

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

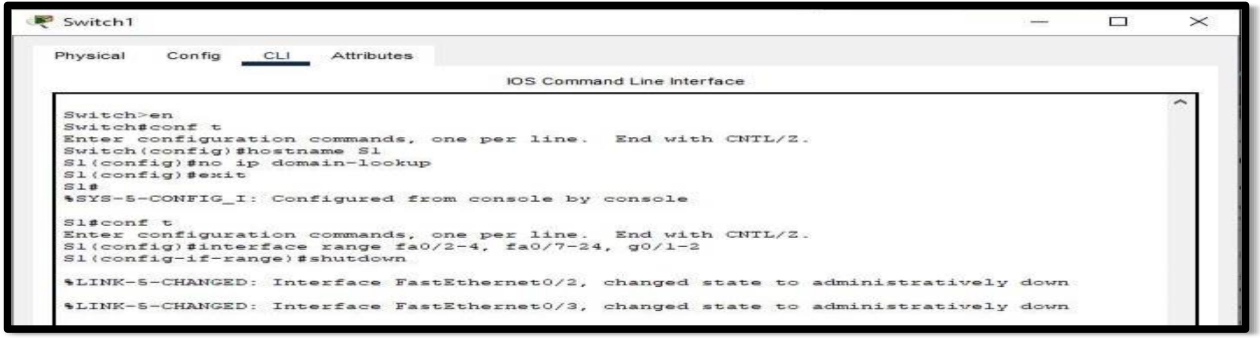


```
R1(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/0, changed state to up

R1(config)#ip route 0.0.0.0 0.0.0.0 209.165.200.225
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

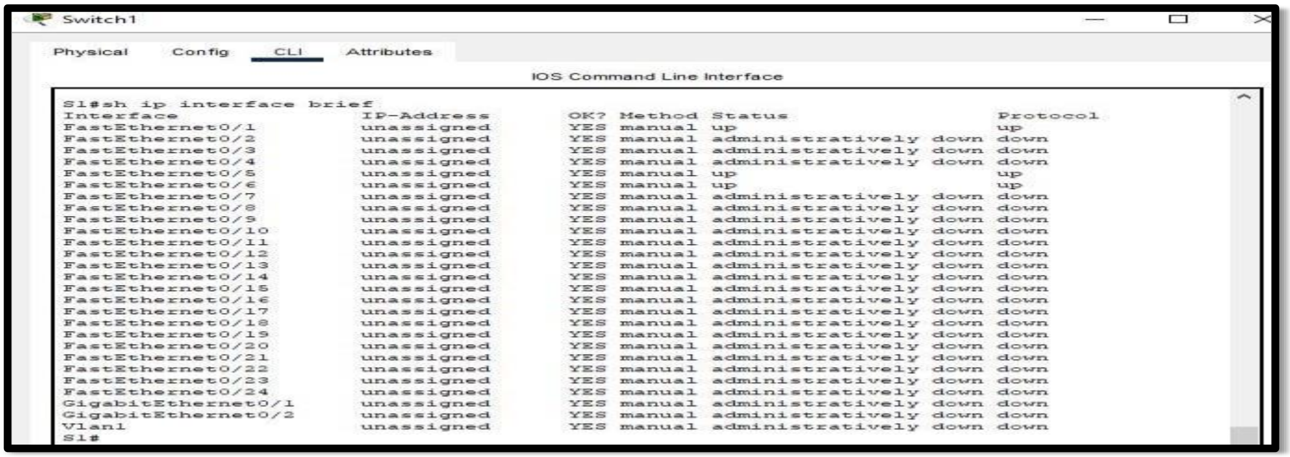


```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R2
R2(config)#no ip domain-lookup
R2(config)#exit
R2#
%SYS-5-CONFIG_I: Configured from console by console

R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface g0/0/0
R2(config-if)#ip address 209.165.200.225 255.255.255.248
R2(config-if)#no shutdown

R2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state to up

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

R2(config-if)#interface loopback 1

R2(config-if)#
%LINK-5-CHANGED: Interface Loopback1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback1, changed state to up

R2(config-if)#ip address 209.165.200.1 255.255.255.224
R2(config-if)#end
R2#
%SYS-5-CONFIG_I: Configured from console by console

R2#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
R2#
```

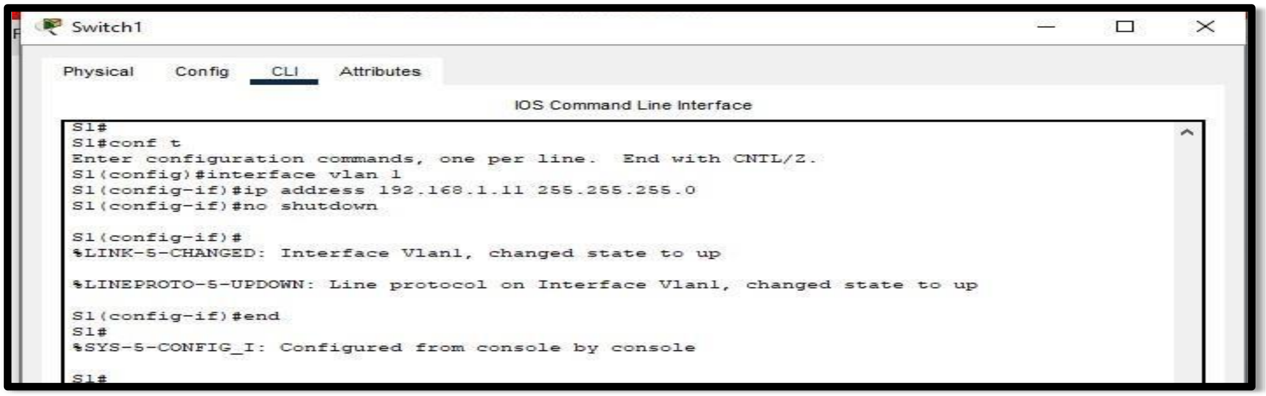
Step 3: Configure basic settings for each switch and PC



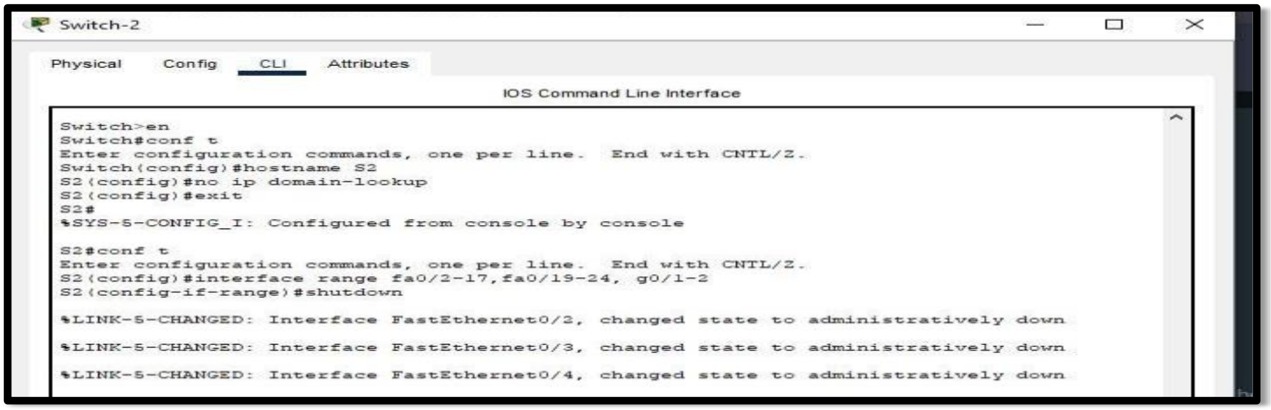
```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S1
S1(config)#no ip domain-lookup
S1(config)#exit
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S1(config)#interface range fa0/2-4, fa0/7-24, g0/1-2
S1(config-if-range)#shutdown

%LINK-5-CHANGED: Interface FastEthernet0/2, changed state to administratively down

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



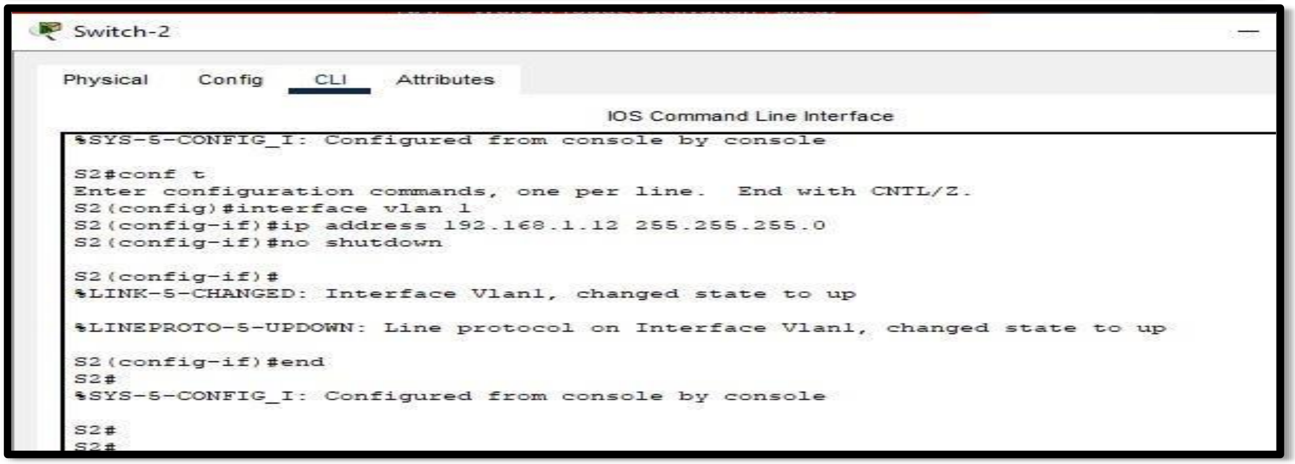
```
S1#sh ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/1            unassigned      YES manual up                    up
FastEthernet0/2            unassigned      YES manual administratively down down
FastEthernet0/3            unassigned      YES manual administratively down down
FastEthernet0/4            unassigned      YES manual administratively down down
FastEthernet0/5            unassigned      YES manual up                    up
FastEthernet0/6            unassigned      YES manual up                    up
FastEthernet0/7            unassigned      YES manual administratively down down
FastEthernet0/8            unassigned      YES manual administratively down down
FastEthernet0/9            unassigned      YES manual administratively down down
FastEthernet0/10           unassigned      YES manual administratively down down
FastEthernet0/11           unassigned      YES manual administratively down down
FastEthernet0/12           unassigned      YES manual administratively down down
FastEthernet0/13           unassigned      YES manual administratively down down
FastEthernet0/14           unassigned      YES manual administratively down down
FastEthernet0/15           unassigned      YES manual administratively down down
FastEthernet0/16           unassigned      YES manual administratively down down
FastEthernet0/17           unassigned      YES manual administratively down down
FastEthernet0/18           unassigned      YES manual administratively down down
FastEthernet0/19           unassigned      YES manual administratively down down
FastEthernet0/20           unassigned      YES manual administratively down down
FastEthernet0/21           unassigned      YES manual administratively down down
FastEthernet0/22           unassigned      YES manual administratively down down
FastEthernet0/23           unassigned      YES manual administratively down down
FastEthernet0/24           unassigned      YES manual administratively down down
GigabitEthernet0/1         unassigned      YES manual administratively down down
GigabitEthernet0/2         unassigned      YES manual administratively down down
Vlan1                      unassigned      YES manual administratively down down
S1#
```



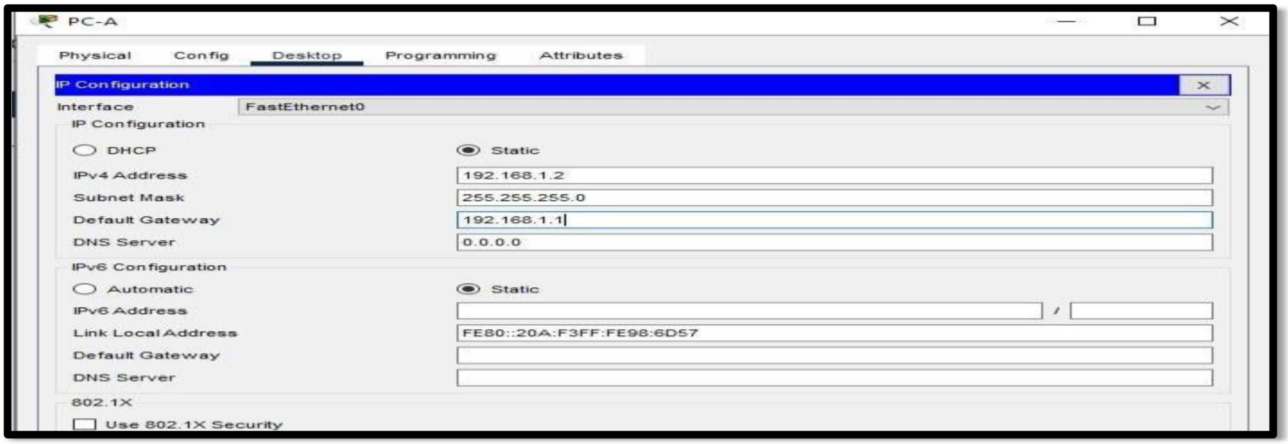
```
S1#
S1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S1(config)#interface vlan 1
S1(config-if)#ip address 192.168.1.11 255.255.255.0
S1(config-if)#no shutdown

S1(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up

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

S1#
```



```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S2
S2(config)#no ip domain-lookup
S2(config)#exit
S2#
%SYS-5-CONFIG_I: Configured from console by console

S2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S2(config)#interface range fa0/2-17,fa0/19-24, g0/1-2
S2(config-if-range)#shutdown

%LINK-5-CHANGED: Interface FastEthernet0/2, changed state to administratively down

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

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



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

S2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S2(config)#interface vlan 1
S2(config-if)#ip address 192.168.1.12 255.255.255.0
S2(config-if)#no shutdown

S2(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up

S2(config-if)#end
S2#
%SYS-5-CONFIG_I: Configured from console by console

S2#
S2#
```



IP Configuration

Interface FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address 192.168.1.2

Subnet Mask 255.255.255.0

Default Gateway 192.168.1.1

DNS Server 0.0.0.0

IPv6 Configuration

○ Automatic ● Static

IPv6 Address /

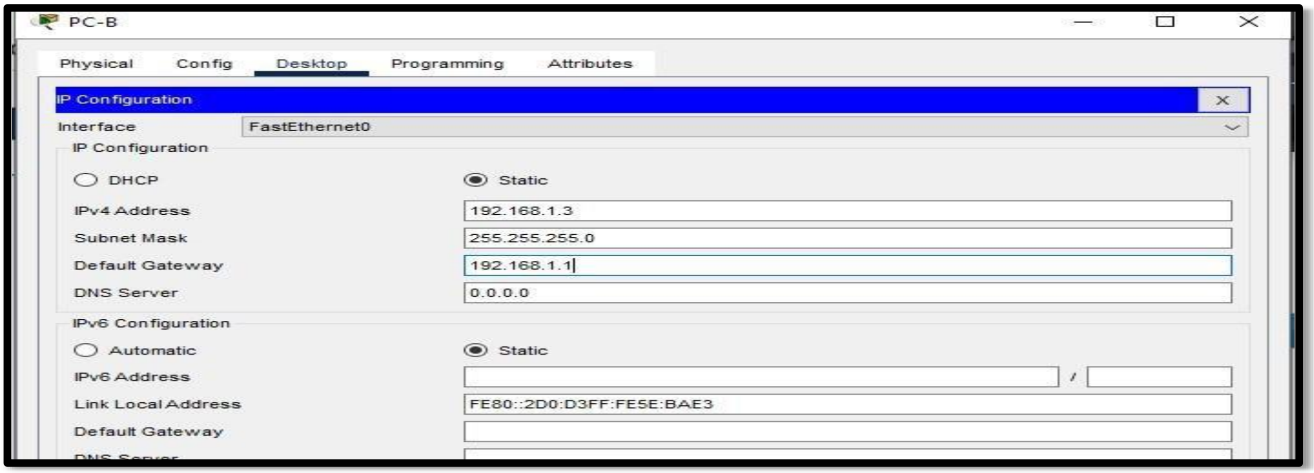
Link Local Address FE80::20A:F3FF:FE98:6D57

Default Gateway

DNS Server

802.1X

☐ Use 802.1X Security



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address: 192.168.1.3

Subnet Mask: 255.255.255.0

Default Gateway: 192.168.1.1

DNS Server: 0.0.0.0

IPv6 Configuration

○ Automatic ● Static

IPv6 Address: /

Link Local Address: FE80::2D0:D3FF:FE5E:BAE3

Default Gateway:

DNS Server

Part 2: Configure and verify NAT for IPv4

Step 1: Configure NAT on R1 using a pool of three addresses, 209.165.200.226- 209.165.200.228.

a. Configure a simple access list that defines what hosts are going to be allowed for translation. In this case,all devices on the R1 LAN are eligible for translation.

R1(config)# **access-list 1 permit 192.168.1.0 0.0.0.255**

b. Create the NAT pool, and give it a name and a range of addresses to use.

R1(config)# **ip nat pool PUBLIC_ACCESS 209.165.200.226 209.165.200.228 netmask 255.255.255.248**

c. Configure the translation, associating the ACL and Pool to the translation process.
R1(config)# **ip nat inside source list 1 pool PUBLIC_ACCESS**

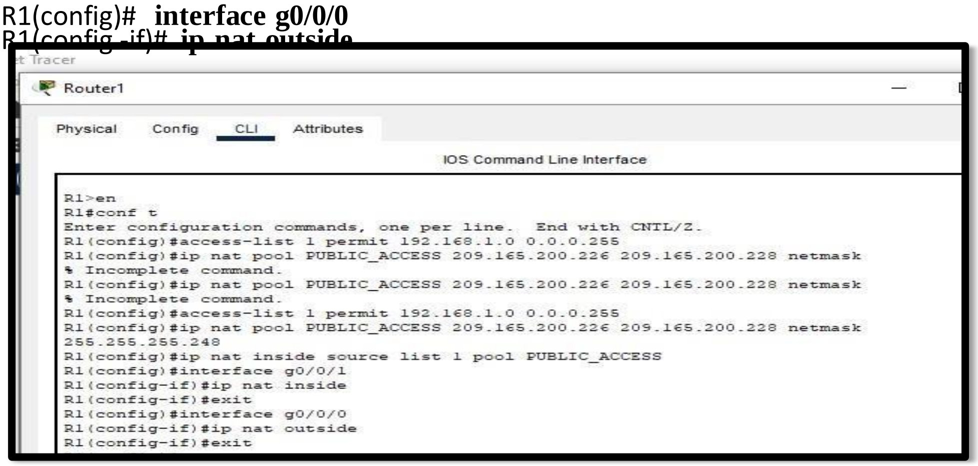
d. Define the inside interface.

R1(config)# **interface g0/0/1**

R1(config-if)# **ip nat inside**

e. Define the outside interface.

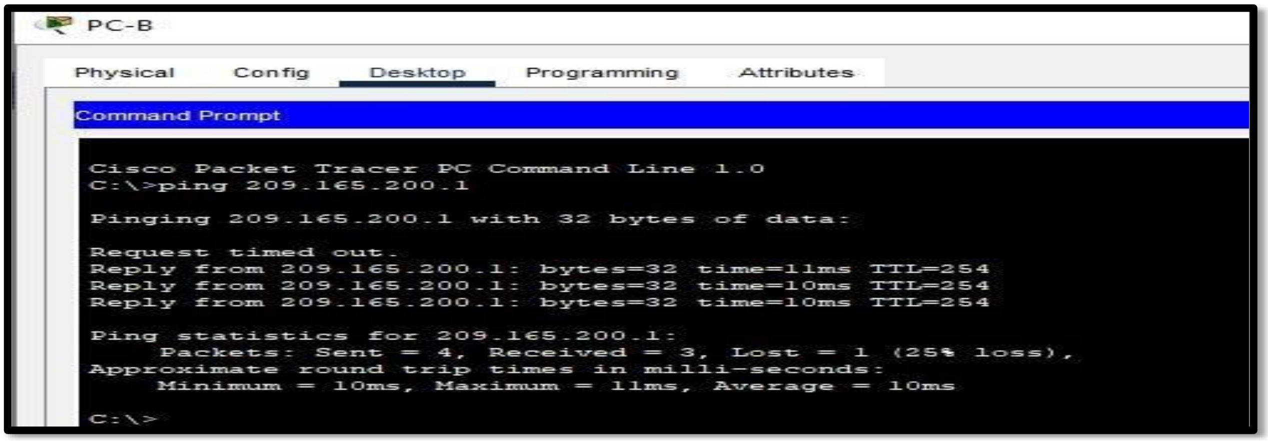
R1(config)# **interface g0/0/0**
R1(config-if)# **ip nat outside**



```
R1>en
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
R1(config)#ip nat pool PUBLIC_ACCESS 209.165.200.226 209.165.200.228 netmask
% Incomplete command.
R1(config)#ip nat pool PUBLIC_ACCESS 209.165.200.226 209.165.200.228 netmask
% Incomplete command.
R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
R1(config)#ip nat pool PUBLIC_ACCESS 209.165.200.226 209.165.200.228 netmask
255.255.255.248
R1(config)#ip nat inside source list 1 pool PUBLIC_ACCESS
R1(config)#interface g0/0/1
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#interface g0/0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
```

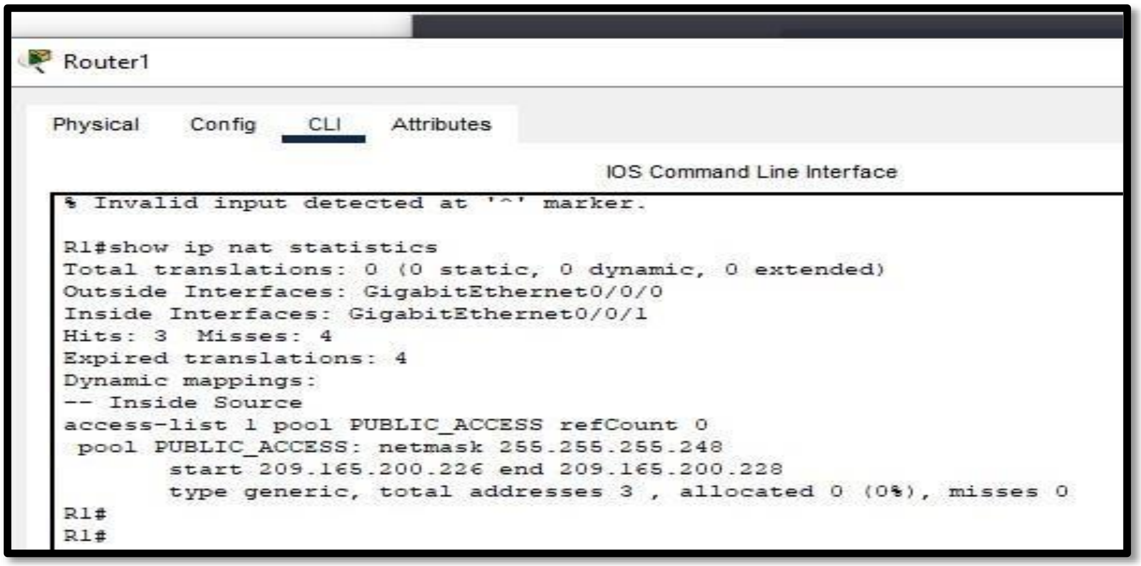
Step 2: Test and Verify the configuration.

a. From PC-B, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on
R1 with the command **show ip nat translations.**



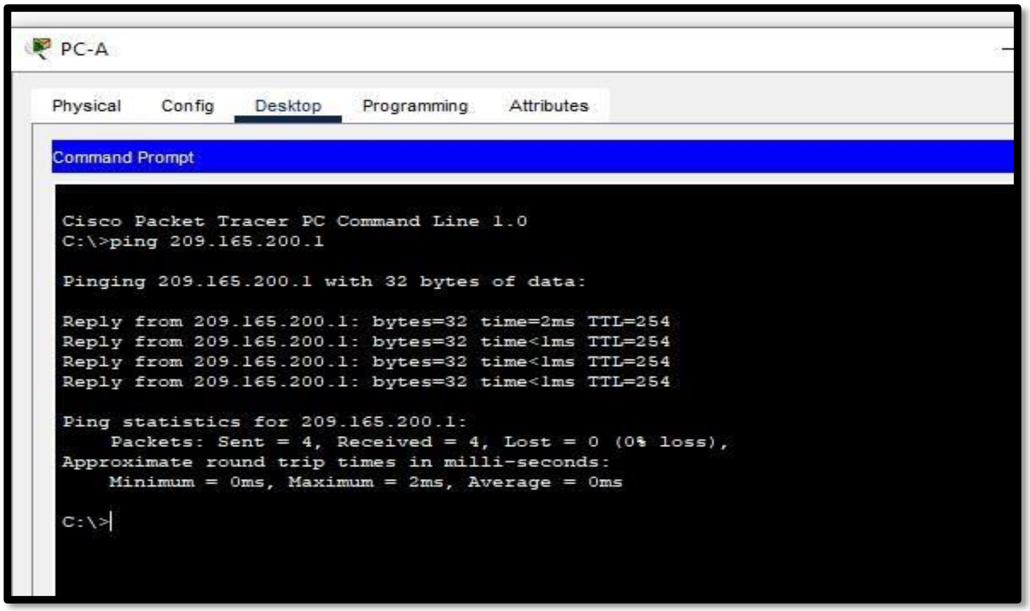
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Request timed out.
Reply from 209.165.200.1: bytes=32 time=11ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 10ms, Maximum = 11ms, Average = 10ms

C:\>
```

R1# **show ip nat translations**



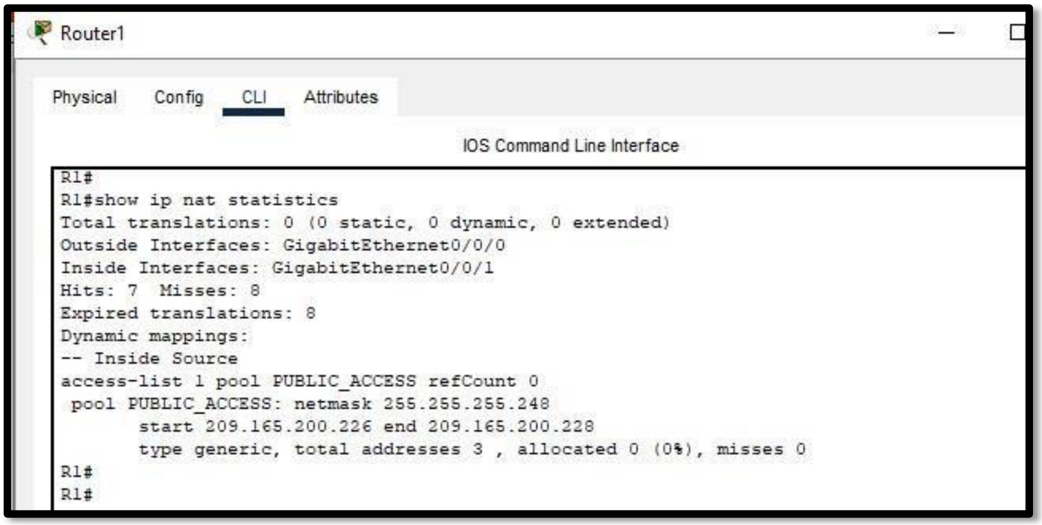
```
% Invalid input detected at '^' marker.

R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 3  Misses: 4
Expired translations: 4
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 0
 pool PUBLIC_ACCESS: netmask 255.255.255.248
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%), misses 0
R1#
R1#
```

b. From PC-A, ping the Lo1 interface (**209.165.200.1**) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations.**



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Reply from 209.165.200.1: bytes=32 time=2ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

C:\>
```

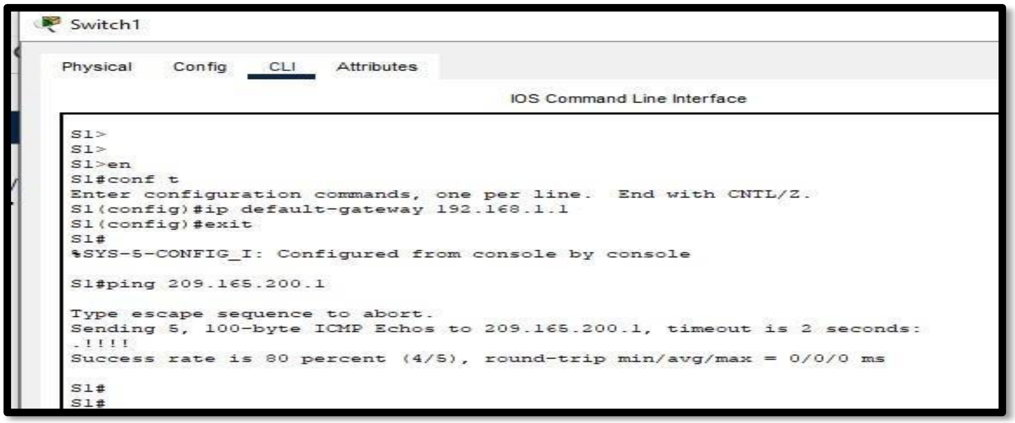
R1# **show ip nat translations**



```
R1#
R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 7  Misses: 8
Expired translations: 8
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 0
 pool PUBLIC_ACCESS: netmask 255.255.255.248
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%), misses 0
R1#
R1#
```

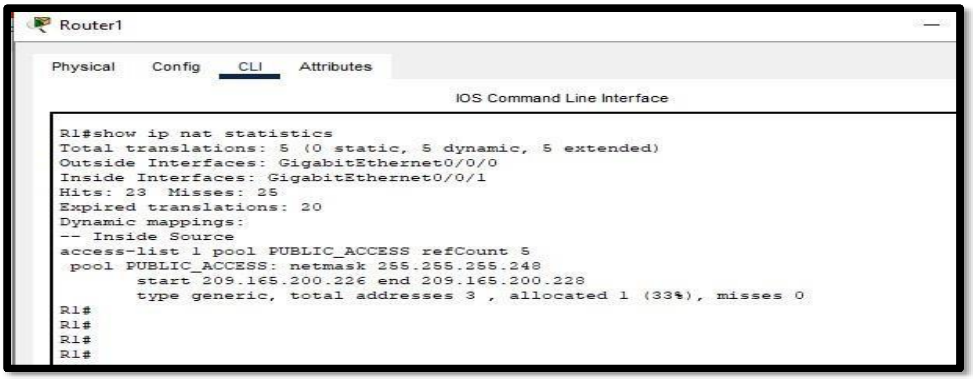
c. Notice that the previous translation for PC-B is still in the table. From S1, ping the Lo1 interface (**209.165.200.1**) on R2.

If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations**.



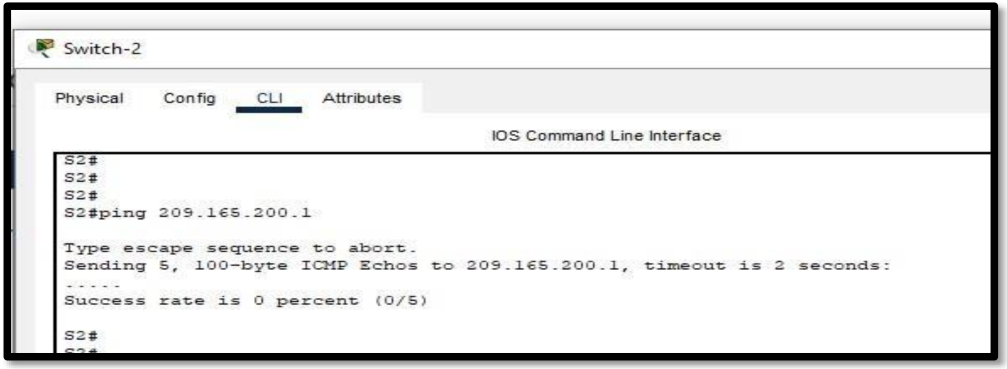
```
S1>
S1>
S1>en
S1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
S1(config)#ip default-gateway 192.168.1.1
S1(config)#exit
S1#
%SYS-5-CONFIG_I: Configured from console by console

S1#ping 209.165.200.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 0/0/0 ms

S1#
S1#
```

R1# **show ip nat translations**



```
R1#show ip nat statistics
Total translations: 5 (0 static, 5 dynamic, 5 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 23  Misses: 25
Expired translations: 20
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 5
 pool PUBLIC_ACCESS: netmask 255.255.255.248
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 1 (33%), misses 0
R1#
R1#
R1#
R1#
```

d. Now try and ping R2 Lo1 from S2. This time, the translations fail, and you get these messages (or similar) on the R1 console:



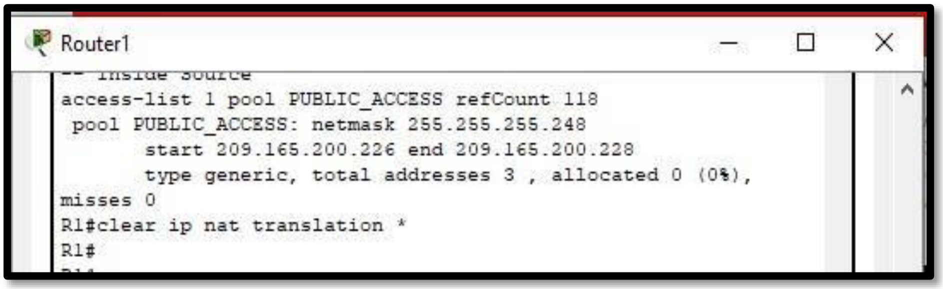
```
S2#
S2#
S2#
S2#ping 209.165.200.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

S2#
```

e. This is an expected result, because only 3 addresses are allocated, and we tried to ping Lo1 from four devices. Recall that NAT is a one-to-one translation. So how long are the translations allocated? Issue the command **show ip nat translations verbose** and you will see that the answer is for 24 hours.

R1# **show ip nat translations verbose**

f. Given that the pool is limited to three addresses, NAT to a pool of addresses is not adequate for our application. Clear the NAT translations and statistics and we will move on to PAT.

R1# **clear ip nat translations**



```
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 118
 pool PUBLIC_ACCESS: netmask 255.255.255.248
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#clear ip nat translation *
R1#
```

Part 3: Configure and verify PAT for IPv4

In Part 3, you will configure replace NAT with PAT to a pool of addresses, and then with PAT using an interface.

Step 1: Remove the translation command on R1.

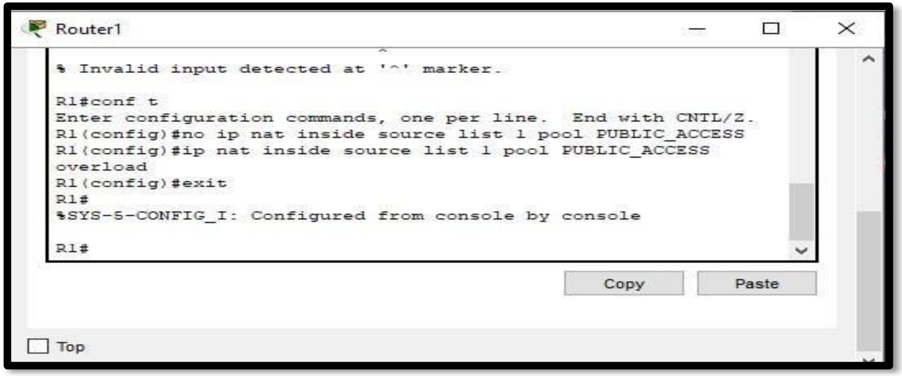
The components of an Address Translation configuration are basically the same; something (an access-list) to identify addresses eligible to be translated, an optionally configured pool of addresses to translate them to, and the commands necessary to identify the inside and outside interfaces. From Part 1, our access-list (access-list 1) is still correct for the network scenario, so there is no need to recreate it. We are going to use the same pool of addresses, so there is no need to recreate that configuration either. Also, the inside and outside interfaces are not changing. To get started in Part 3, remove the command that ties the ACL and pool together.

R1(config)# **no ip nat inside source list 1 pool PUBLIC_ACCESS**

Step 2: Add the PAT command on R1.

Now, configure for PAT translation to a pool of addresses (remember, the ACL and Pool are already configured, so this is the only command we need to change from NAT to PAT).

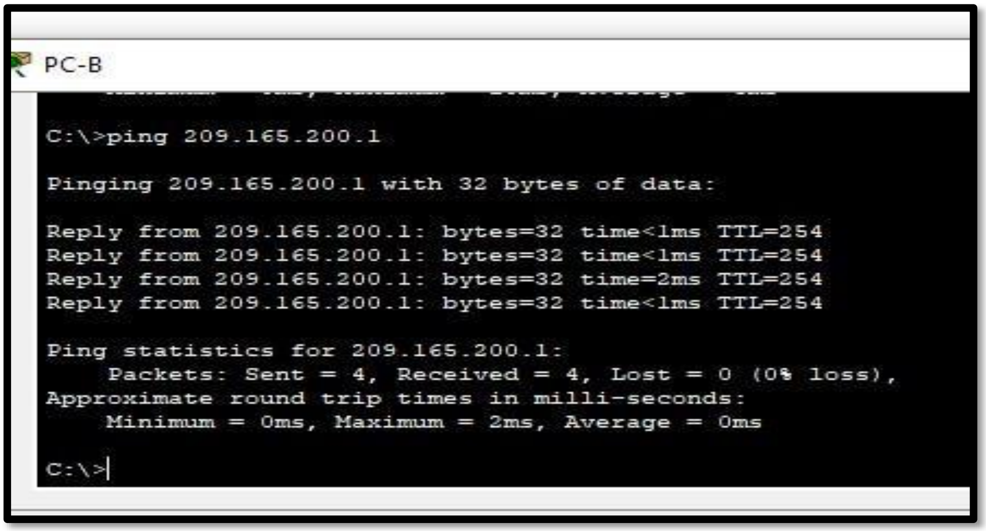
R1(config)# **ip nat inside source list 1 pool PUBLIC_ACCESS overload**



```
% Invalid input detected at '^' marker.

R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#no ip nat inside source list 1 pool PUBLIC_ACCESS
R1(config)#ip nat inside source list 1 pool PUBLIC_ACCESS
overload
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#
```

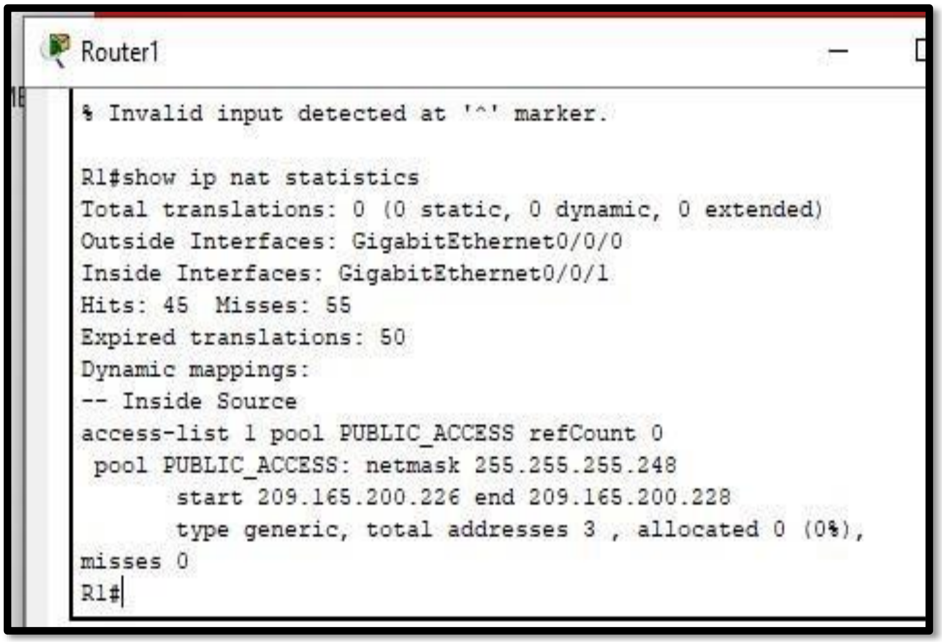
Step 3: Test and Verify the configuration.

a. Let's verify PAT is working. From PC-B, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations.**



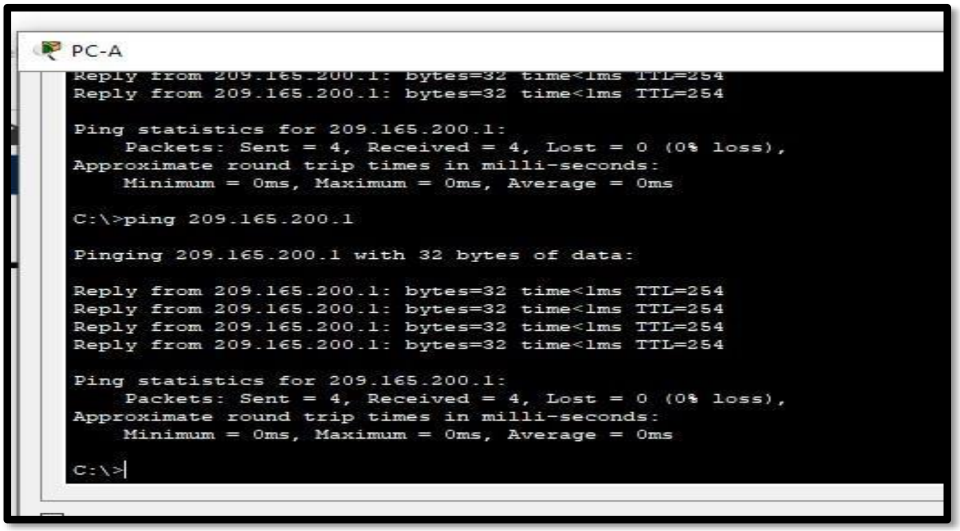
```
C:\>ping 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=2ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

C:\>
```

R1# **show ip nat translations**



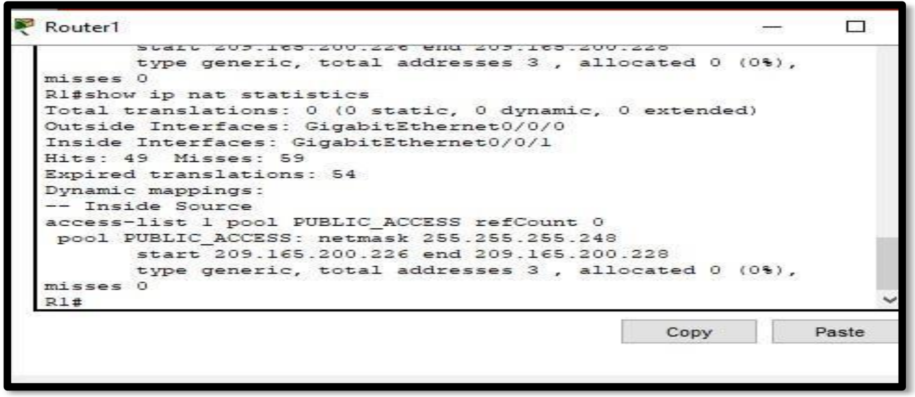
```
% Invalid input detected at '^' marker.

R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 45  Misses: 55
Expired translations: 50
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 0
 pool PUBLIC_ACCESS: netmask 255.255.255.248
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#
```

b. From PC-A, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues. On R1, display the NAT table on R1 with the command **show ip nat translations.**



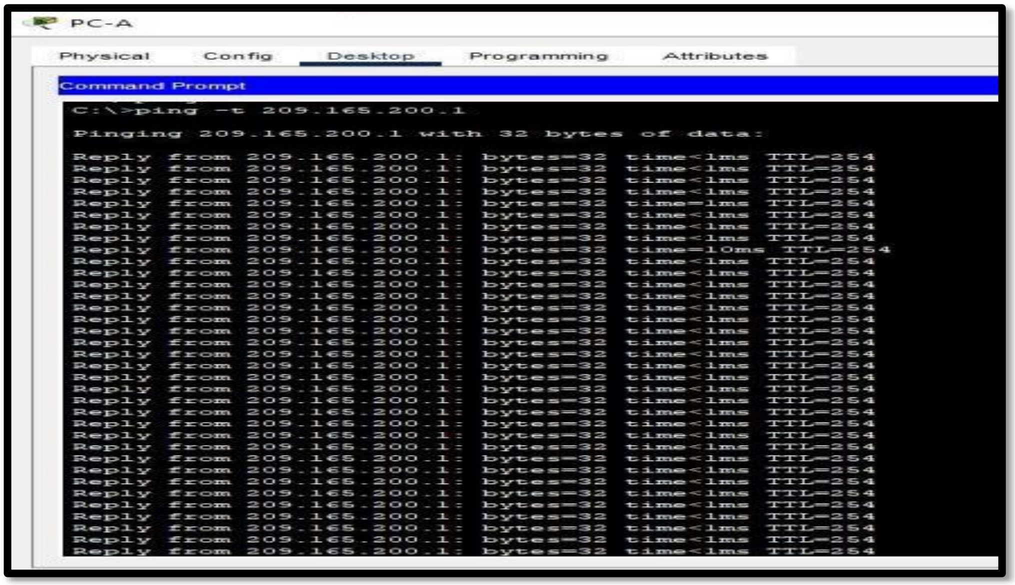
```
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

R1# **show ip nat translations**

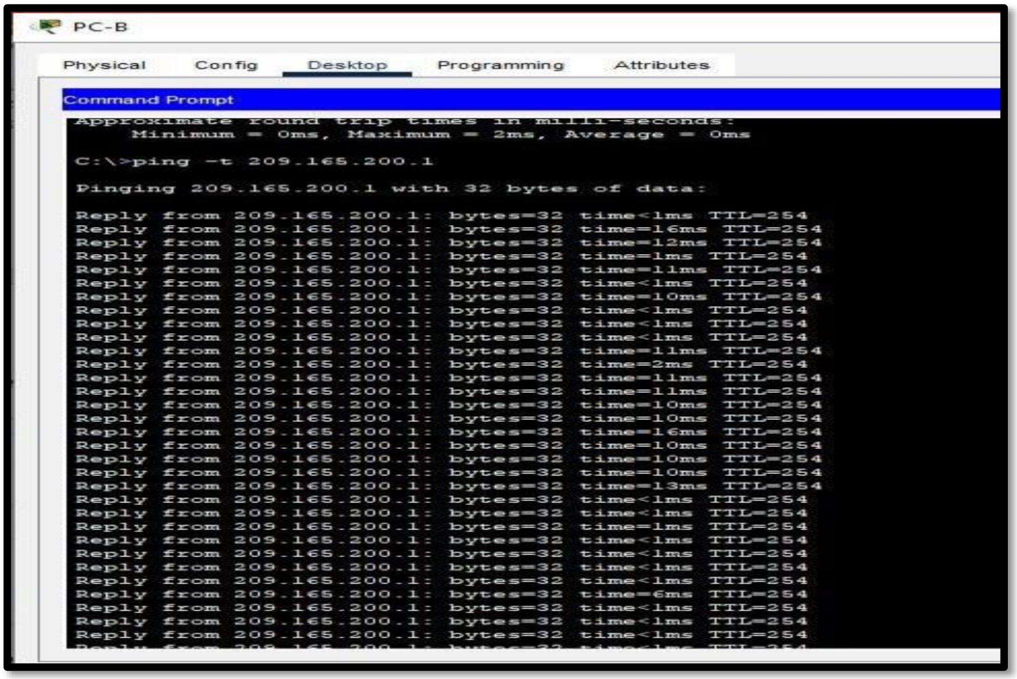


```
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 49  Misses: 59
Expired translations: 54
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 0
 pool PUBLIC_ACCESS: netmask 255.255.255.248
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#
```

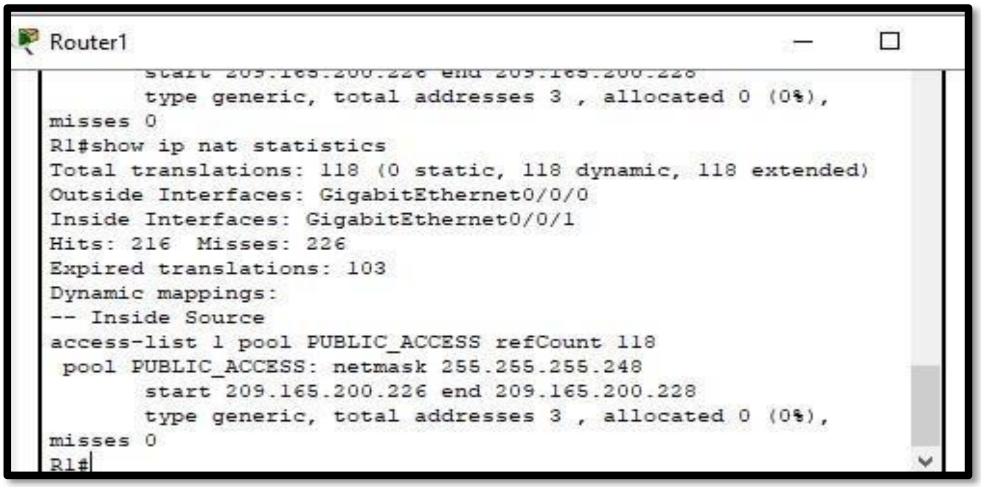
c. Generate traffic from multiple devices to observe PAT. On PC-A and PC-B, use the -t parameter with the ping command to send a non-stop ping to R2's Lo1 interface (**ping -t 209.165.200.1**), then go back to R1 and issue the **show ip nat translations** command:



```
C:\>ping -t 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
```



```
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

C:\>ping -t 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=16ms TTL=254
Reply from 209.165.200.1: bytes=32 time=12ms TTL=254
Reply from 209.165.200.1: bytes=32 time=1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=11ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=11ms TTL=254
Reply from 209.165.200.1: bytes=32 time=2ms TTL=254
Reply from 209.165.200.1: bytes=32 time=11ms TTL=254
Reply from 209.165.200.1: bytes=32 time=11ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time=16ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time=13ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=6ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
```

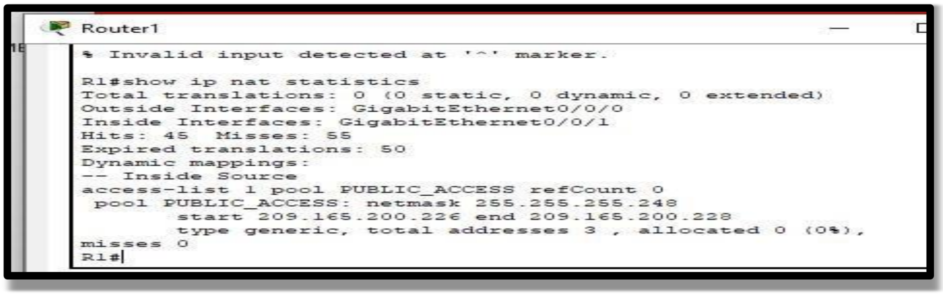


```
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#show ip nat statistics
Total translations: 118 (0 static, 118 dynamic, 118 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 216  Misses: 226
Expired translations: 103
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 118
 pool PUBLIC_ACCESS: netmask 255.255.255.248
       start 209.165.200.226 end 209.165.200.228
       type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#
```

d. PAT to a pool is a very effective solution for small-to-midsize organizations. However, there are unused IPv4 addresses involved in this scenario. We will move to PAT with interface overload to eliminate this waste of IPv4 addresses. Stop the pings on PC-A and PC-B with the Control-C key combination, then clear translations and translation statistics:

R1# **clear ip nat translations ***

R1# **clear ip nat statistics**



```
% Invalid input detected at '^' marker.

R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 45  Misses: 55
Expired translations: 50
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 0
 pool PUBLIC_ACCESS: netmask 255.255.255.248
        start 209.165.200.226 end 209.165.200.228
        type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#
```

Step 4: On R1, remove the nat pool translation commands.

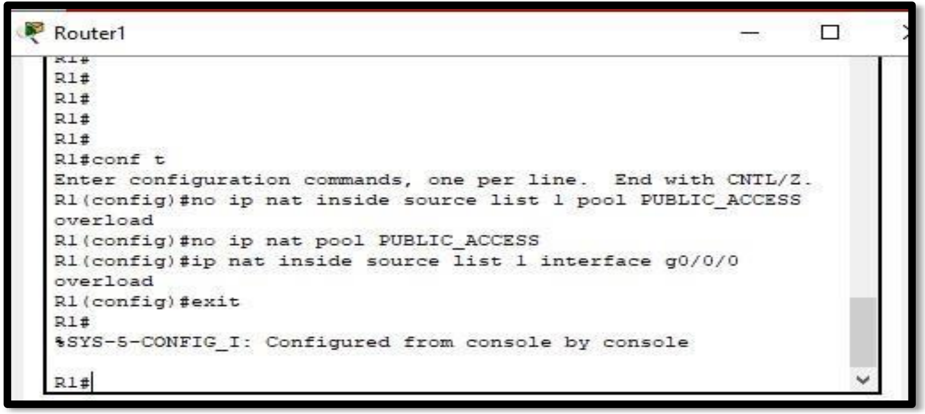
Once again, our access-list (access-list 1) is still correct for the network scenario, so there is no need to recreate it. Also, the inside and outside interfaces are not changing.

R1(config)# **no ip nat inside source list 1 pool PUBLIC_ACCESS overload**

R1(config)# **no ip nat pool PUBLIC_ACCESS**

Step 5: Add the PAT overload command by specifying the outside interface.

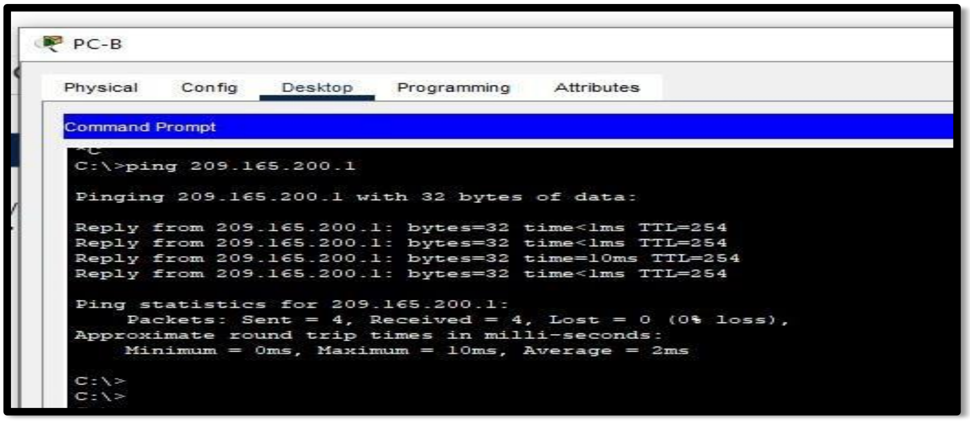
R1(config)# **ip nat inside source list 1 interface g0/0/0 overload**



```
R1#
R1#
R1#
R1#
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#no ip nat inside source list 1 pool PUBLIC_ACCESS
overload
R1(config)#no ip nat pool PUBLIC_ACCESS
R1(config)#ip nat inside source list 1 interface g0/0/0
overload
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#
```

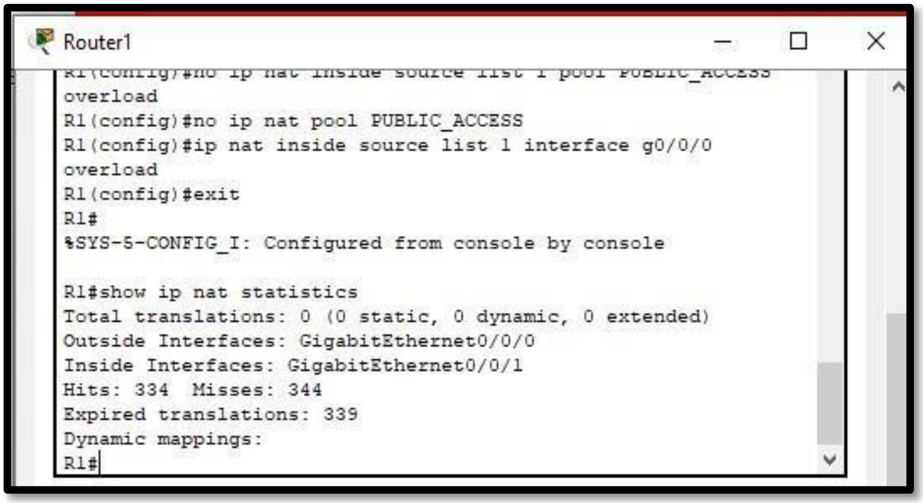
Step 6: Test and Verify the configuration.

a. Let's verify PAT to the interface is working. From PC-B, ping the Lo1 interface (209.165.200.1) on R2. If the ping was unsuccessful, troubleshoot and correct the issues



```
^C
C:\>ping 209.165.200.1

Pinging 209.165.200.1 with 32 bytes of data:

Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254
Reply from 209.165.200.1: bytes=32 time=10ms TTL=254
Reply from 209.165.200.1: bytes=32 time<1ms TTL=254

Ping statistics for 209.165.200.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 10ms, Average = 2ms

C:\>
C:\>
```

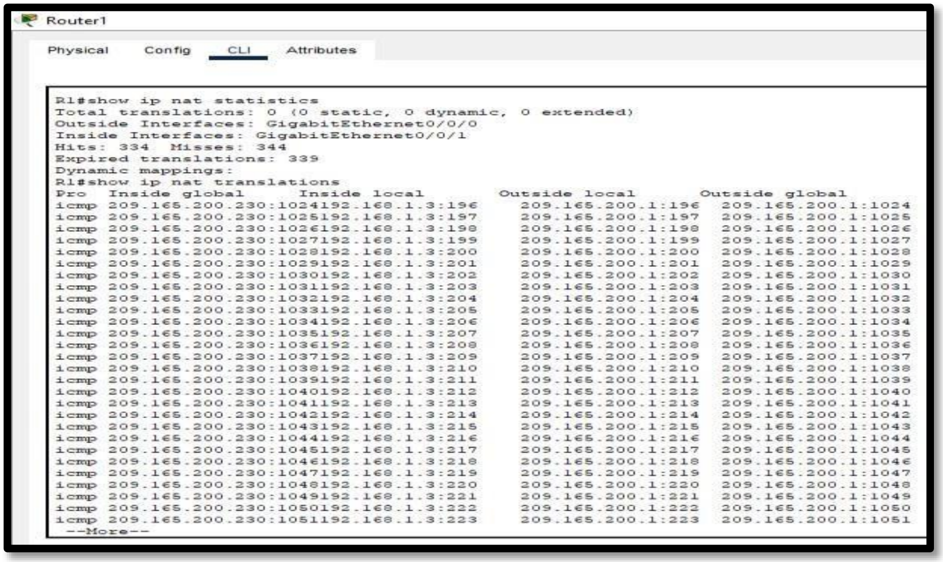
R1# **show ip nat translations**



```
R1(config)#no ip nat inside source list 1 pool PUBLIC_ACCESS
overload
R1(config)#no ip nat pool PUBLIC_ACCESS
R1(config)#ip nat inside source list 1 interface g0/0/0
overload
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 334  Misses: 344
Expired translations: 339
Dynamic mappings:
R1#
```

b. Generate traffic from multiple devices to observe PAT. On PC-A and PC-B, use the -t parameter with the ping command to send a non-stop ping to R2’s Lo1 interface **(ping -t 209.165.200.1**). On S1 and S2, issue the privileged exec command ping 209.165.200.1 repeat 2000. Then go back to R1 and issue the **show ip nat translations** command.

R1# **show ip nat translations**



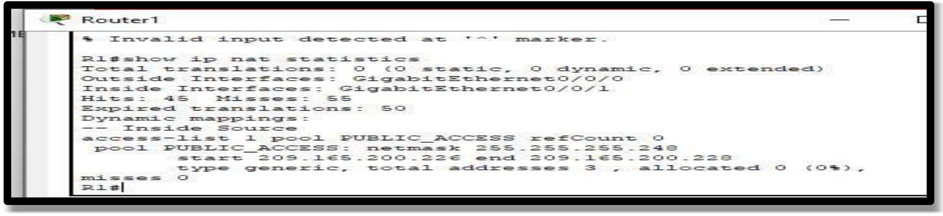
```
R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 334  Misses: 344
Expired translations: 339
Dynamic mappings:
R1#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
icmp 209.165.200.230:1024192.168.1.3:196    209.165.200.1:196  209.165.200.1:1024
icmp 209.165.200.230:1025192.168.1.3:197    209.165.200.1:197  209.165.200.1:1025
icmp 209.165.200.230:1026192.168.1.3:198    209.165.200.1:198  209.165.200.1:1026
icmp 209.165.200.230:1027192.168.1.3:199    209.165.200.1:199  209.165.200.1:1027
icmp 209.165.200.230:1028192.168.1.3:200    209.165.200.1:200  209.165.200.1:1028
icmp 209.165.200.230:1029192.168.1.3:201    209.165.200.1:201  209.165.200.1:1029
icmp 209.165.200.230:1030192.168.1.3:202    209.165.200.1:202  209.165.200.1:1030
icmp 209.165.200.230:1031192.168.1.3:203    209.165.200.1:203  209.165.200.1:1031
icmp 209.165.200.230:1032192.168.1.3:204    209.165.200.1:204  209.165.200.1:1032
icmp 209.165.200.230:1033192.168.1.3:205    209.165.200.1:205  209.165.200.1:1033
icmp 209.165.200.230:1034192.168.1.3:206    209.165.200.1:206  209.165.200.1:1034
icmp 209.165.200.230:1035192.168.1.3:207    209.165.200.1:207  209.165.200.1:1035
icmp 209.165.200.230:1036192.168.1.3:208    209.165.200.1:208  209.165.200.1:1036
icmp 209.165.200.230:1037192.168.1.3:209    209.165.200.1:209  209.165.200.1:1037
icmp 209.165.200.230:1038192.168.1.3:210    209.165.200.1:210  209.165.200.1:1038
icmp 209.165.200.230:1039192.168.1.3:211    209.165.200.1:211  209.165.200.1:1039
icmp 209.165.200.230:1040192.168.1.3:212    209.165.200.1:212  209.165.200.1:1040
icmp 209.165.200.230:1041192.168.1.3:213    209.165.200.1:213  209.165.200.1:1041
icmp 209.165.200.230:1042192.168.1.3:214    209.165.200.1:214  209.165.200.1:1042
icmp 209.165.200.230:1043192.168.1.3:215    209.165.200.1:215  209.165.200.1:1043
icmp 209.165.200.230:1044192.168.1.3:216    209.165.200.1:216  209.165.200.1:1044
icmp 209.165.200.230:1045192.168.1.3:217    209.165.200.1:217  209.165.200.1:1045
icmp 209.165.200.230:1046192.168.1.3:218    209.165.200.1:218  209.165.200.1:1046
icmp 209.165.200.230:1047192.168.1.3:219    209.165.200.1:219  209.165.200.1:1047
icmp 209.165.200.230:1048192.168.1.3:220    209.165.200.1:220  209.165.200.1:1048
icmp 209.165.200.230:1049192.168.1.3:221    209.165.200.1:221  209.165.200.1:1049
icmp 209.165.200.230:1050192.168.1.3:222    209.165.200.1:222  209.165.200.1:1050
icmp 209.165.200.230:1051192.168.1.3:223    209.165.200.1:223  209.165.200.1:1051
 --More--
```

Part 4: Configure and verify Static NAT for IPv4

In Part 4, you will configure static NAT so that PC-A is directly reachable from the internet. PC-A will be reachable from R2 via the address 209.165.200.229.

Step 1: On R1, clear current translations and statistics. R1# **clear ip nat translations ***

R1# **clear ip nat statistics**



```
% Invalid input detected at '^' marker.

R1#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/0/0
Inside Interfaces: GigabitEthernet0/0/1
Hits: 45  Misses: 55
Expired translations: 50
Dynamic mappings:
-- Inside Source
access-list 1 pool PUBLIC_ACCESS refCount 0
 pool PUBLIC_ACCESS: netmask 255.255.255.248
      start 209.165.200.226 end 209.165.200.228
      type generic, total addresses 3 , allocated 0 (0%),
misses 0
R1#
```

Step 2: On R1, configure the NAT command required to statically map an inside address to an outside address.

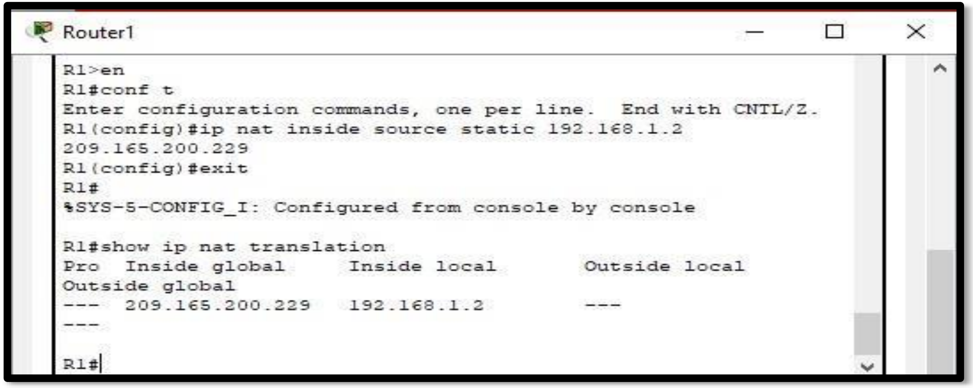
For this step, configure a static mapping between 192.168.1.11 and 209.165.200.1 using the following command:

R1(config)# **ip nat inside source static 192.168.1.2 209.165.200.229**

Step 3: Test and Verify the configuration.

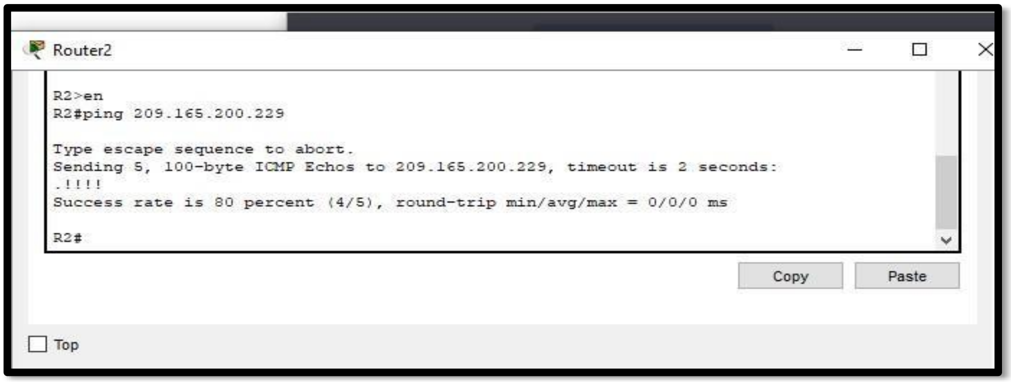
a. Let’s verify the Static NAT is working. On R1, display the NAT table on R1 with the command **show ip nat translations**, and you should see the static mapping.

R1# **show ip nat translations**



```
R1>en
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip nat inside source static 192.168.1.2
209.165.200.229
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show ip nat translation
Pro  Inside global     Inside local       Outside local
Outside global
---  209.165.200.229   192.168.1.2        ---
---

R1#
```

b. The translation table shows the static translation is in effect. Verify this by pinging from R2 to 209.165.200.229. The pings should work.

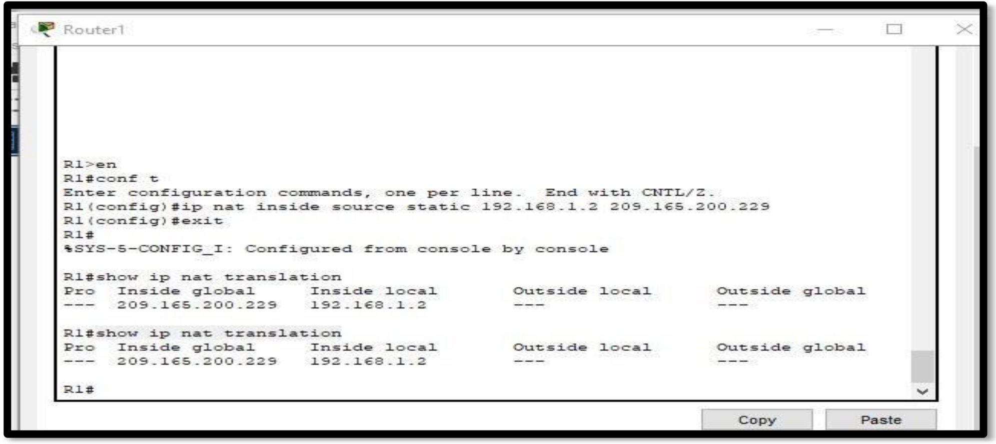


```
R2>en
R2#ping 209.165.200.229

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.229, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 0/0/0 ms

R2#
```

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

c. On R1, display the NAT table on R1 with the command **show ip nat translations**, and you should see the static mapping and the port-level translation for the inbound pings.

R1# **show ip nat translations**



```
R1>en
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip nat inside source static 192.168.1.2 209.165.200.229
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show ip nat translation
Pro  Inside global     Inside local       Outside local      Outside global
---  209.165.200.229   192.168.1.2        ---                ---

R1#show ip nat translation
Pro  Inside global     Inside local       Outside local      Outside global
---  209.165.200.229   192.168.1.2        ---                ---

R1#
```

This validates that the Static NAT is working.