Sadiq S.

Practical 4

AIM:

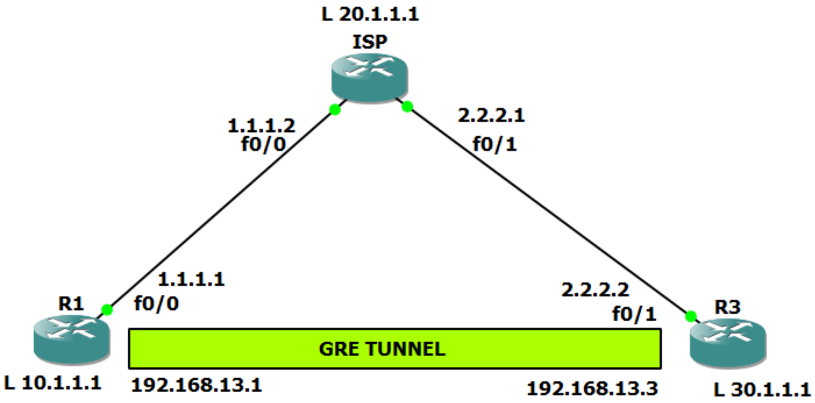
1. Implement a GRE Tunnel
2. Implement VTP
3. Implement NAT

1. Implement a GRE Tunnel:

What is GRE Tunnelling?

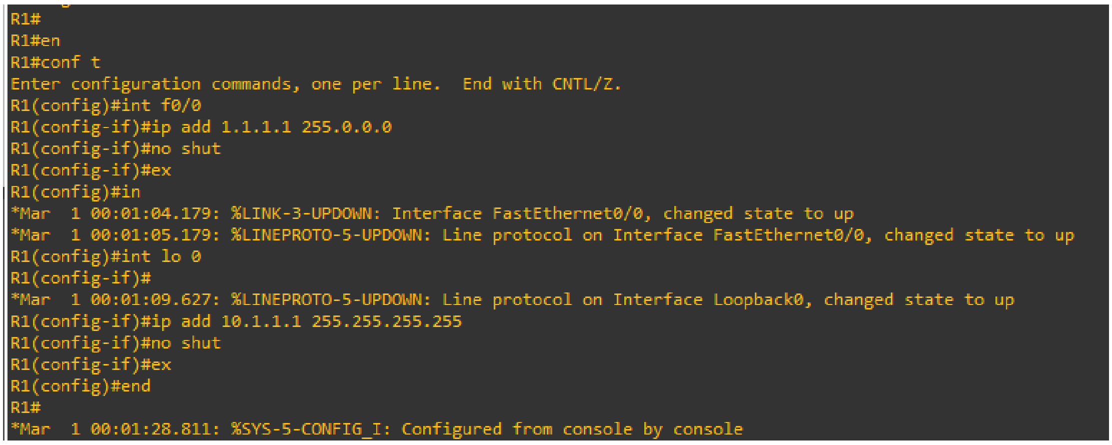
- Generic routing encapsulation (GRE) provides a private path for transporting packets through an otherwise public network by encapsulating (or tunnelling) the packets.
- GRE tunnelling is accomplished through tunnel endpoints that encapsulate or de-encapsulate traffic.
- GRE is one way to set up a direct point-to-point connection across a network, for the purpose of simplifying connections between separate networks. It works with a variety of network layer protocols.
- Encapsulating packets within other packets is called "tunnelling."
- GRE tunnels are usually configured between two routers, with each router acting like one end of the tunnel.
- The routers are set up to send and receive GRE packets directly to each other.
- Any routers in between those two routers will not open the encapsulated packets; they only reference the headers surrounding the encapsulated packets in order to forward them.
- Only routers between which GRE is configured can decrypt and encrypt the packet.

Step 1: Build the network:



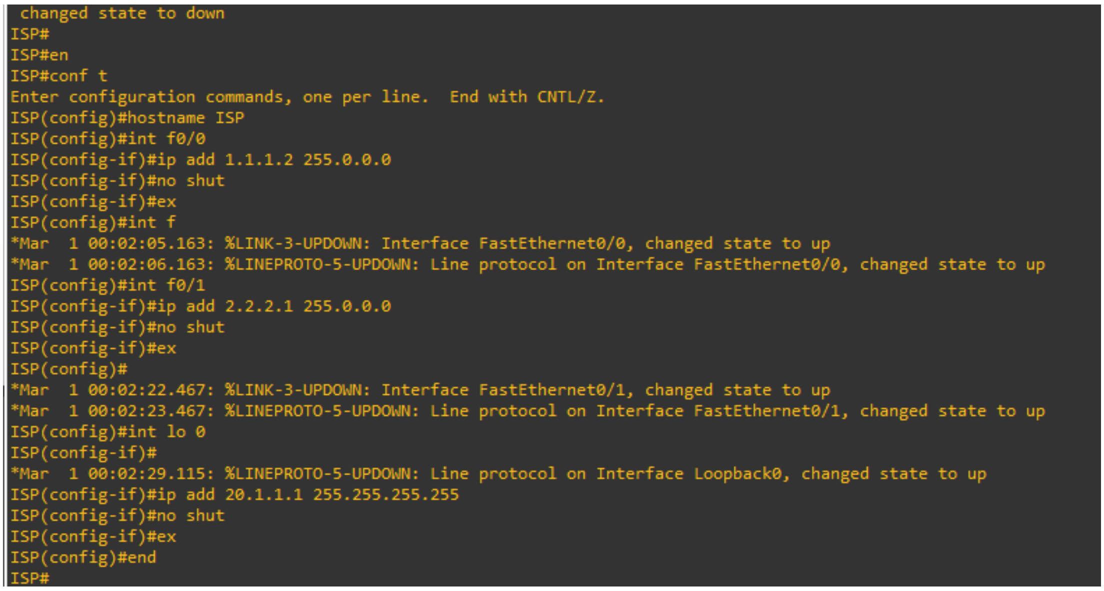
Step 2: Configure the router:

R1:



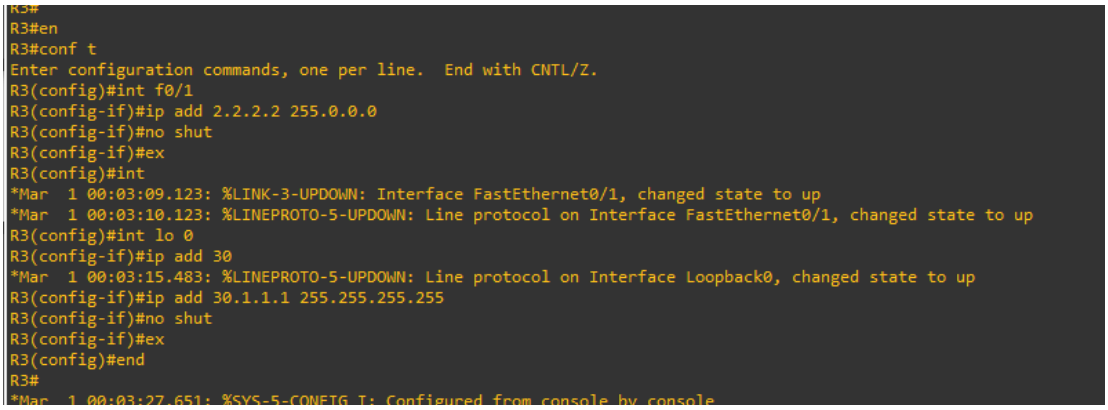
```
R1#
R1#en
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int f0/0
R1(config-if)#ip add 1.1.1.1 255.0.0.0
R1(config-if)#no shut
R1(config-if)#ex
R1(config)#in
*Mar  1 00:01:04.179: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:01:05.179: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config)#int lo 0
R1(config-if)#
*Mar  1 00:01:09.627: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R1(config-if)#ip add 10.1.1.1 255.255.255.255
R1(config-if)#no shut
R1(config-if)#ex
R1(config)#end
R1#
*Mar  1 00:01:28.811: %SYS-5-CONFIG_I: Configured from console by console
```

R2:



```
 changed state to down
ISP#
ISP#en
ISP#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP(config)#hostname ISP
ISP(config)#int f0/0
ISP(config-if)#ip add 1.1.1.2 255.0.0.0
ISP(config-if)#no shut
ISP(config-if)#ex
ISP(config)#int f
*Mar  1 00:02:05.163: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:02:06.163: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
ISP(config)#int f0/1
ISP(config-if)#ip add 2.2.2.1 255.0.0.0
ISP(config-if)#no shut
ISP(config-if)#ex
ISP(config)#
*Mar  1 00:02:22.467: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:02:23.467: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
ISP(config)#int lo 0
ISP(config-if)#
*Mar  1 00:02:29.115: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ISP(config-if)#ip add 20.1.1.1 255.255.255.255
ISP(config-if)#no shut
ISP(config-if)#ex
ISP(config)#end
ISP#
```

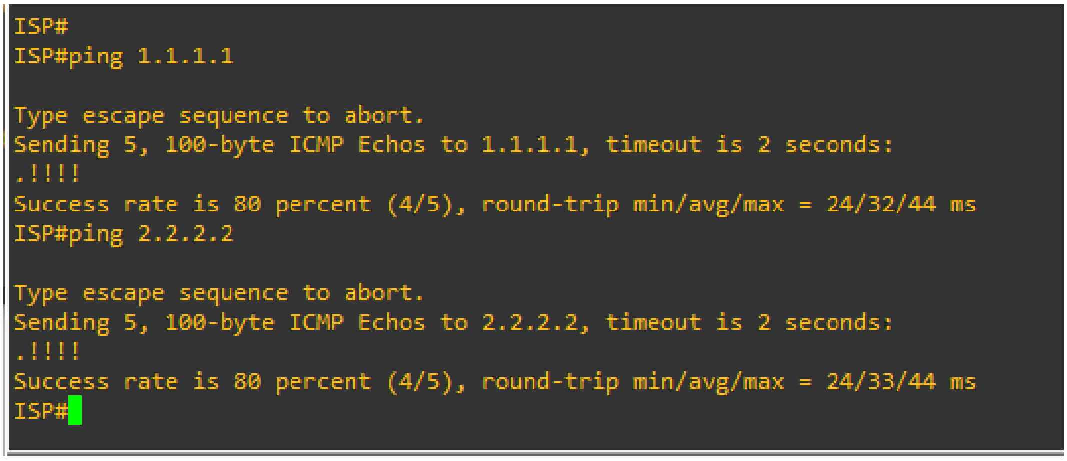
R3:



```
R3#
R3#en
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#int f0/1
R3(config-if)#ip add 2.2.2.2 255.0.0.0
R3(config-if)#no shut
R3(config-if)#ex
R3(config)#int
*Mar  1 00:03:09.123: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:03:10.123: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
R3(config)#int lo 0
R3(config-if)#ip add 30
*Mar  1 00:03:15.483: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R3(config-if)#ip add 30.1.1.1 255.255.255.255
R3(config-if)#no shut
R3(config-if)#ex
R3(config)#end
R3#
*Mar  1 00:03:27.651: %SYS-5-CONFIG_I: Configured from console by console
```

Step 3: Check the direct connection between them.

By pinging R1 and R3 by ISP:

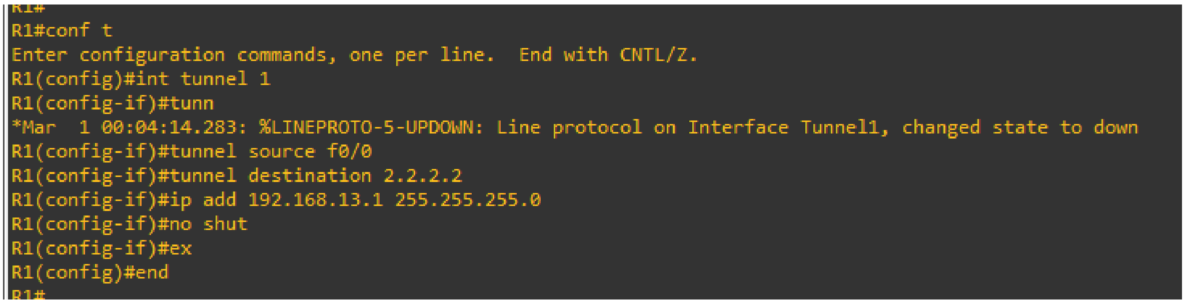


```
ISP#
ISP#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 24/32/44 ms
ISP#ping 2.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 24/33/44 ms
ISP#
```

Hence prove, direct connection is working.

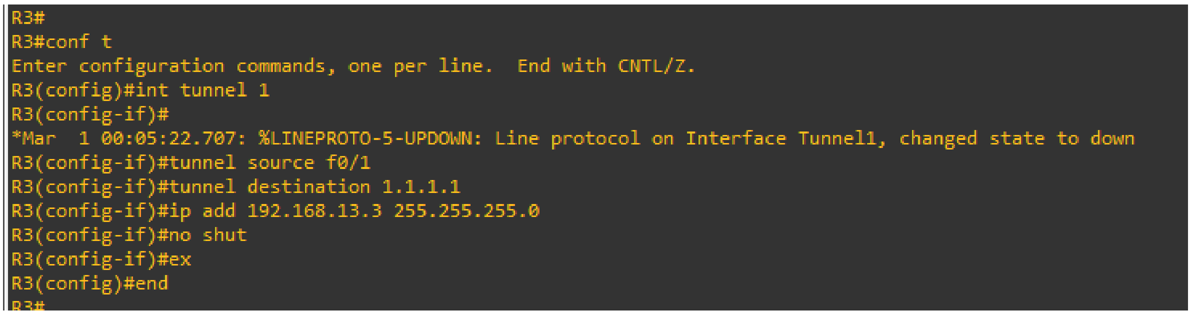
Step 4: Now we will create a GRE Tunnel between R1 and R3 according to the topology:

R1:



```
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int tunnel 1
R1(config-if)#tunn
*Mar  1 00:04:14.283: %LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel1, changed state to down
R1(config-if)#tunnel source f0/0
R1(config-if)#tunnel destination 2.2.2.2
R1(config-if)#ip add 192.168.13.1 255.255.255.0
R1(config-if)#no shut
R1(config-if)#ex
R1(config)#end
R1#
```

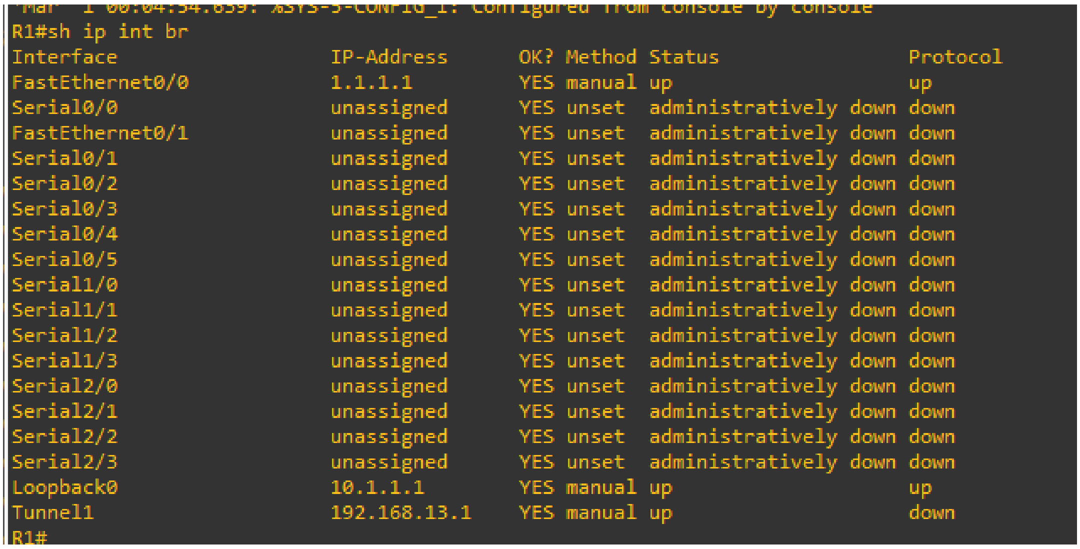
R3:



```
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#int tunnel 1
R3(config-if)#
*Mar  1 00:05:22.707: %LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel1, changed state to down
R3(config-if)#tunnel source f0/1
R3(config-if)#tunnel destination 1.1.1.1
R3(config-if)#ip add 192.168.13.3 255.255.255.0
R3(config-if)#no shut
R3(config-if)#ex
R3(config)#end
R3#
```

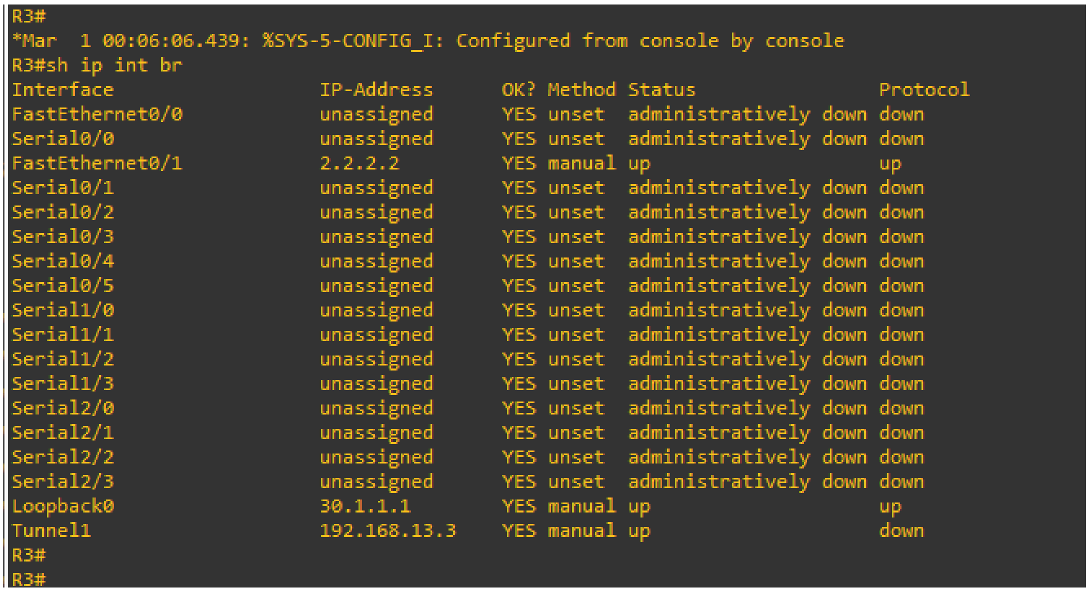
Step 5: Run ‘sh ip int br’ to confirm whether tunnel is made or not:

R1:



```
*Mar  1 00:04:34.639: %SYS-5-CONFIG_I: Configured from console by console
R1#sh ip int br
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            1.1.1.1         YES manual up                    up
Serial0/0                  unassigned      YES unset  administratively down down
FastEthernet0/1            unassigned      YES unset  administratively down down
Serial0/1                  unassigned      YES unset  administratively down down
Serial0/2                  unassigned      YES unset  administratively down down
Serial0/3                  unassigned      YES unset  administratively down down
Serial0/4                  unassigned      YES unset  administratively down down
Serial0/5                  unassigned      YES unset  administratively down down
Serial1/0                  unassigned      YES unset  administratively down down
Serial1/1                  unassigned      YES unset  administratively down down
Serial1/2                  unassigned      YES unset  administratively down down
Serial1/3                  unassigned      YES unset  administratively down down
Serial2/0                  unassigned      YES unset  administratively down down
Serial2/1                  unassigned      YES unset  administratively down down
Serial2/2                  unassigned      YES unset  administratively down down
Serial2/3                  unassigned      YES unset  administratively down down
Loopback0                  10.1.1.1        YES manual up                    up
Tunnel1                    192.168.13.1    YES manual up                    down
R1#
```

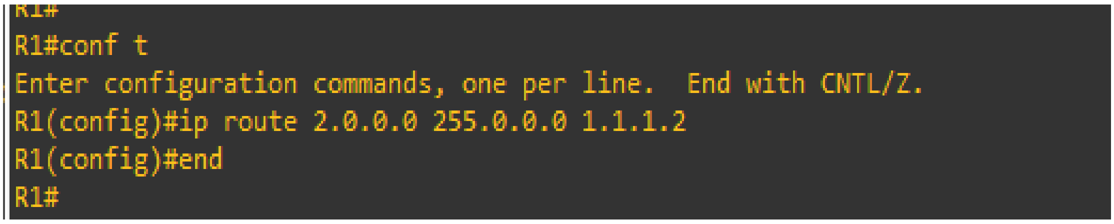
R3:



```
R3#
*Mar  1 00:06:06.439: %SYS-5-CONFIG_I: Configured from console by console
R3#sh ip int br
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            unassigned      YES unset  administratively down down
Serial0/0                  unassigned      YES unset  administratively down down
FastEthernet0/1            2.2.2.2         YES manual up                    up
Serial0/1                  unassigned      YES unset  administratively down down
Serial0/2                  unassigned      YES unset  administratively down down
Serial0/3                  unassigned      YES unset  administratively down down
Serial0/4                  unassigned      YES unset  administratively down down
Serial0/5                  unassigned      YES unset  administratively down down
Serial1/0                  unassigned      YES unset  administratively down down
Serial1/1                  unassigned      YES unset  administratively down down
Serial1/2                  unassigned      YES unset  administratively down down
Serial1/3                  unassigned      YES unset  administratively down down
Serial2/0                  unassigned      YES unset  administratively down down
Serial2/1                  unassigned      YES unset  administratively down down
Serial2/2                  unassigned      YES unset  administratively down down
Serial2/3                  unassigned      YES unset  administratively down down
Loopback0                  30.1.1.1        YES manual up                    up
Tunnel1                    192.168.13.3    YES manual up                    down
R3#
R3#
```

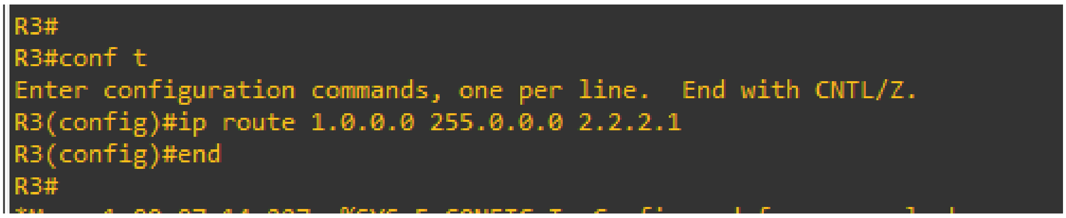
Step 6: Now we will create a static route for R1 and R3 to check whether our tunnel is working or not:

R1:



```
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip route 2.0.0.0 255.0.0.0 1.1.1.2
R1(config)#end
R1#
```

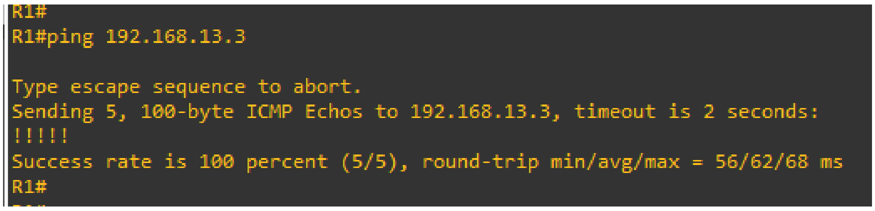
R3:



```
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#ip route 1.0.0.0 255.0.0.0 2.2.2.1
R3(config)#end
R3#
```

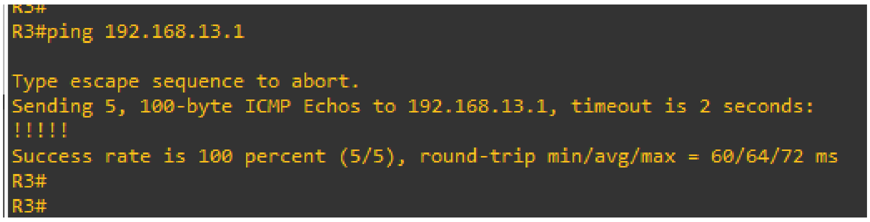
Step 7: Check whether the tunnel work or not:

R1:



```
R1#
R1#ping 192.168.13.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.13.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/62/68 ms
R1#
```

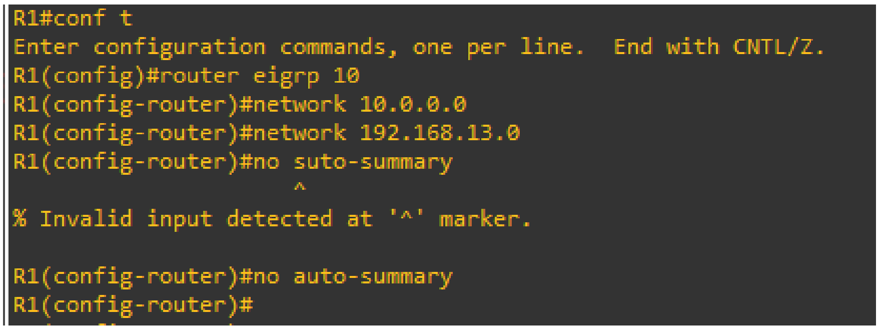
R3:



```
R3#
R3#ping 192.168.13.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.13.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/64/72 ms
R3#
R3#
```

Step 8: Now we will configure EIGRP protocol for R1 and R3:

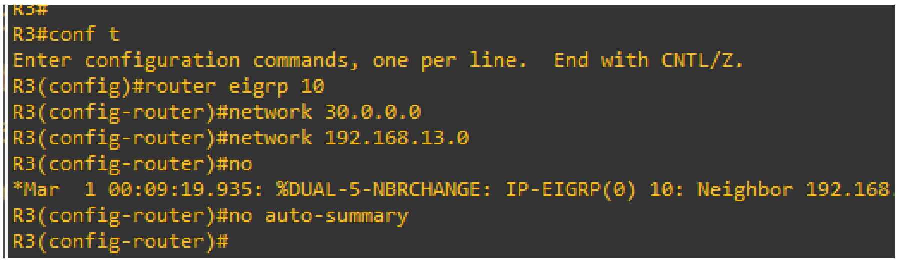
R1:



```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router eigrp 10
R1(config-router)#network 10.0.0.0
R1(config-router)#network 192.168.13.0
R1(config-router)#no suto-summary
                     ^
% Invalid input detected at '^' marker.

R1(config-router)#no auto-summary
R1(config-router)#
```

I haven't add 2.0.0.0's network in the network because they are already following Static Routing.

R3:

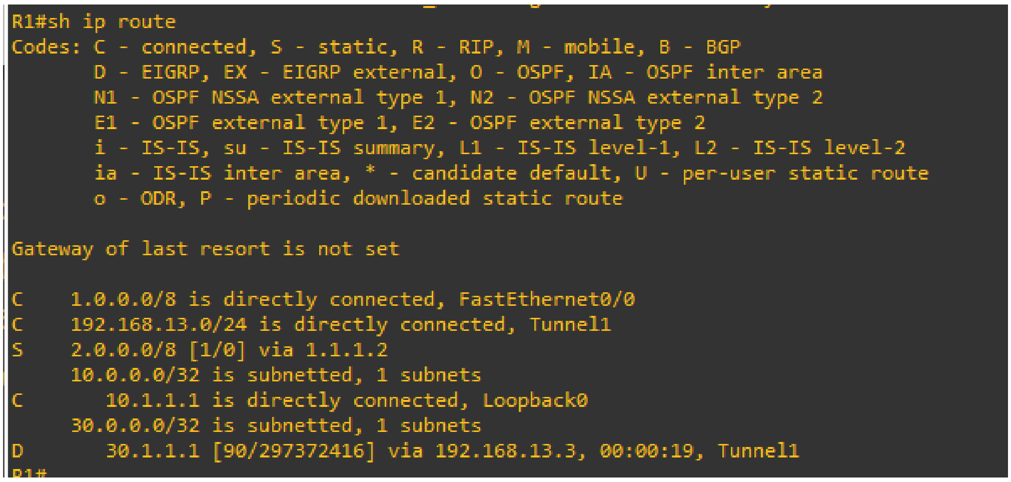


```
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router eigrp 10
R3(config-router)#network 30.0.0.0
R3(config-router)#network 192.168.13.0
R3(config-router)#no
*Mar  1 00:09:19.935: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 10: Neighbor 192.168.
R3(config-router)#no auto-summary
R3(config-router)#
```

I haven’t add 1.0.0.0’s network in the network because they are already following Static Routing.

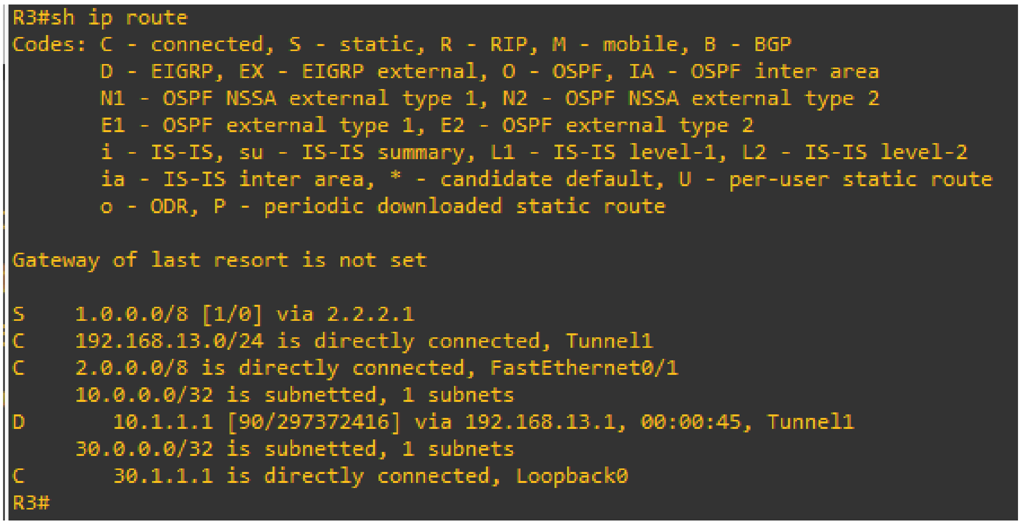
Step 9: Check whether EIGRP is configured properly:

R1:



```
R1#sh ip route
Codes: 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
       i - IS-IS, su - IS-IS summary, 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 not set

C    1.0.0.0/8 is directly connected, FastEthernet0/0
C    192.168.13.0/24 is directly connected, Tunnel1
S    2.0.0.0/8 [1/0] via 1.1.1.2
     10.0.0.0/32 is subnetted, 1 subnets
C       10.1.1.1 is directly connected, Loopback0
     30.0.0.0/32 is subnetted, 1 subnets
D       30.1.1.1 [90/297372416] via 192.168.13.3, 00:00:19, Tunnel1
R1#
```

R3:

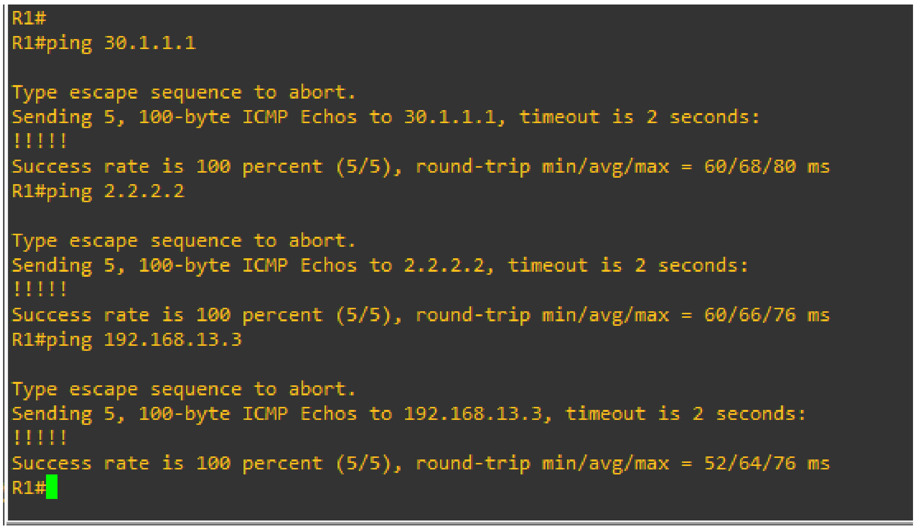


```
R3#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, 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 not set

S    1.0.0.0/8 [1/0] via 2.2.2.1
C    192.168.13.0/24 is directly connected, Tunnel1
C    2.0.0.0/8 is directly connected, FastEthernet0/1
     10.0.0.0/32 is subnetted, 1 subnets
D       10.1.1.1 [90/297372416] via 192.168.13.1, 00:00:45, Tunnel1
     30.0.0.0/32 is subnetted, 1 subnets
C       30.1.1.1 is directly connected, Loopback0
R3#
```

D stands for EIGRP and it is present in both router. So our EIGRP is successfully Configured.

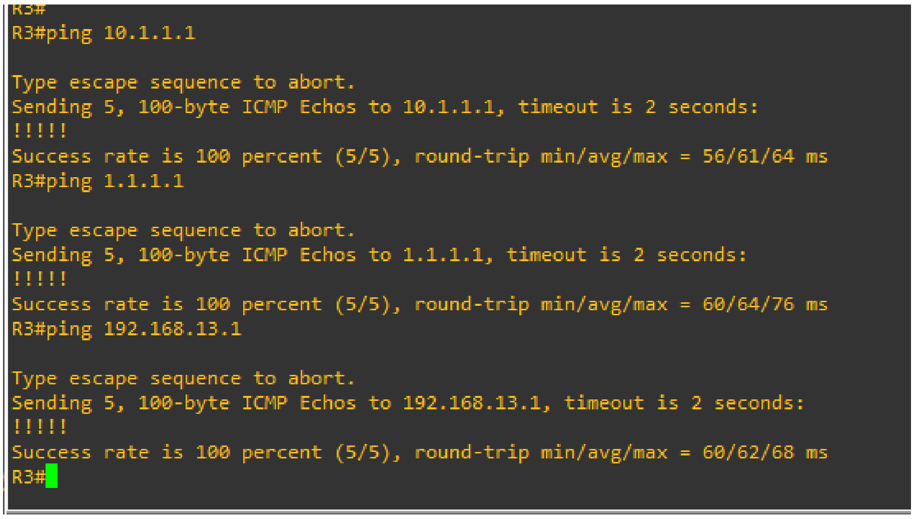
Step 10: You should be able to ping any network now because we have configured it:

R1:



```
R1#
R1#ping 30.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 30.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/68/80 ms
R1#ping 2.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/66/76 ms
R1#ping 192.168.13.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.13.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 52/64/76 ms
R1#
```

R3:



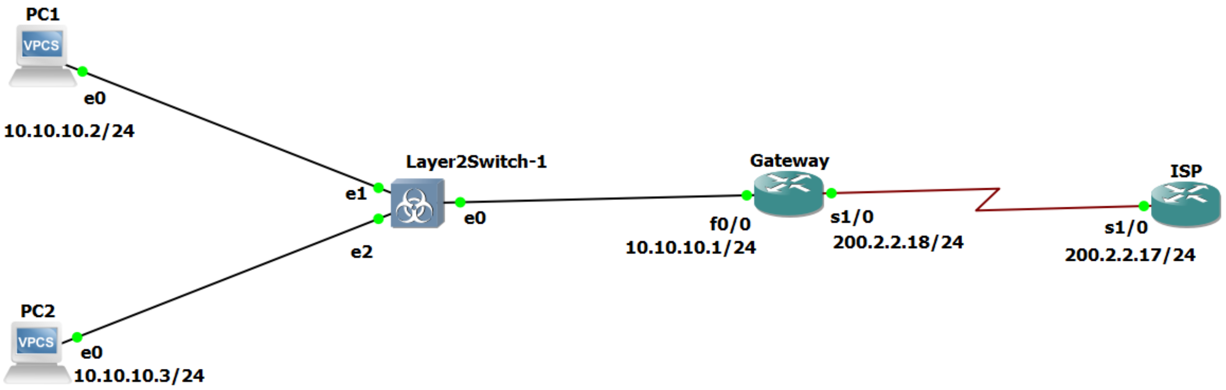
```
R3#
R3#ping 10.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/61/64 ms
R3#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/64/76 ms
R3#ping 192.168.13.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.13.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/62/68 ms
R3#
```

3. Implement NAT:

What is NAT?

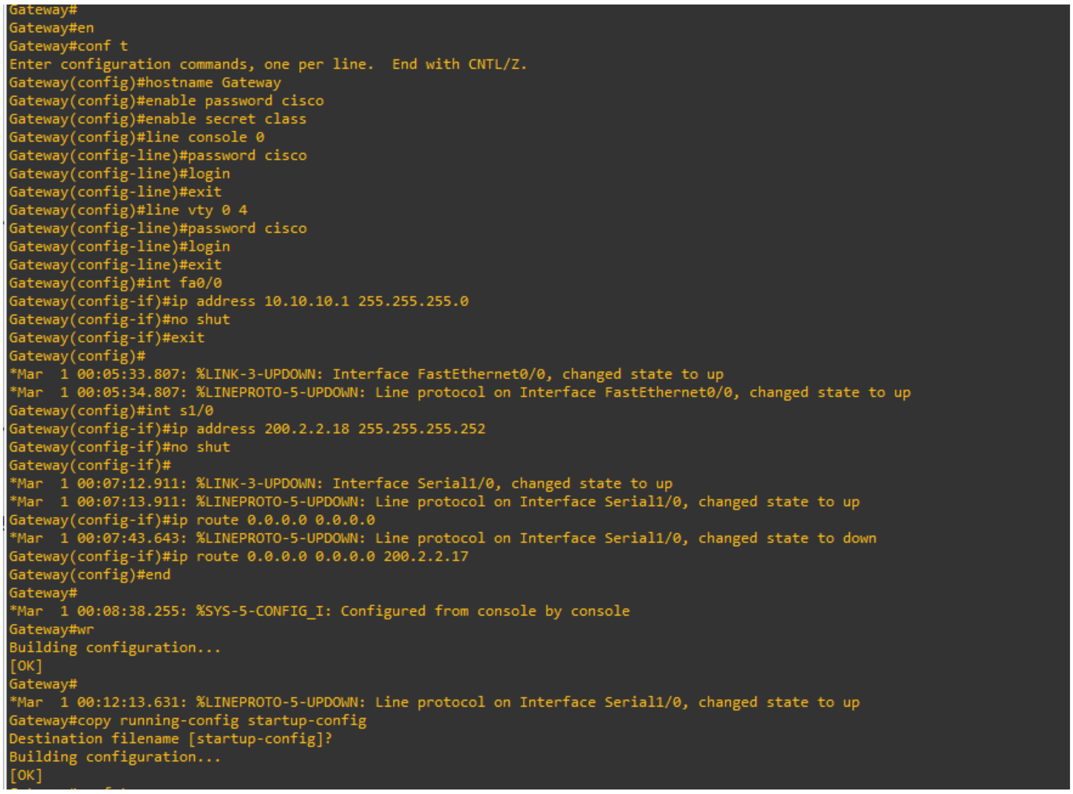
- Network Address Translation (NAT) is a process in which one or more local IP address is translated into one or more Global IP address and vice versa in order to provide Internet access to the local hosts.
- Also, it does the translation of port numbers i.e. masks the port number of the host with another port number, in the packet that will be routed to the destination.
- It then makes the corresponding entries of IP address and port number in the NAT table.
- NAT generally operates on a router or firewall.
- Generally, the border router is configured for NAT i.e the router which has one interface in the local (inside) network and one interface in the global (outside) network.
- When a packet traverse outside the local (inside) network, then NAT converts that local (private) IP address to a global (public) IP address.
- When a packet enters the local network, the global (public) IP address is converted to a local (private) IP address.

Step 1: Design the Network:



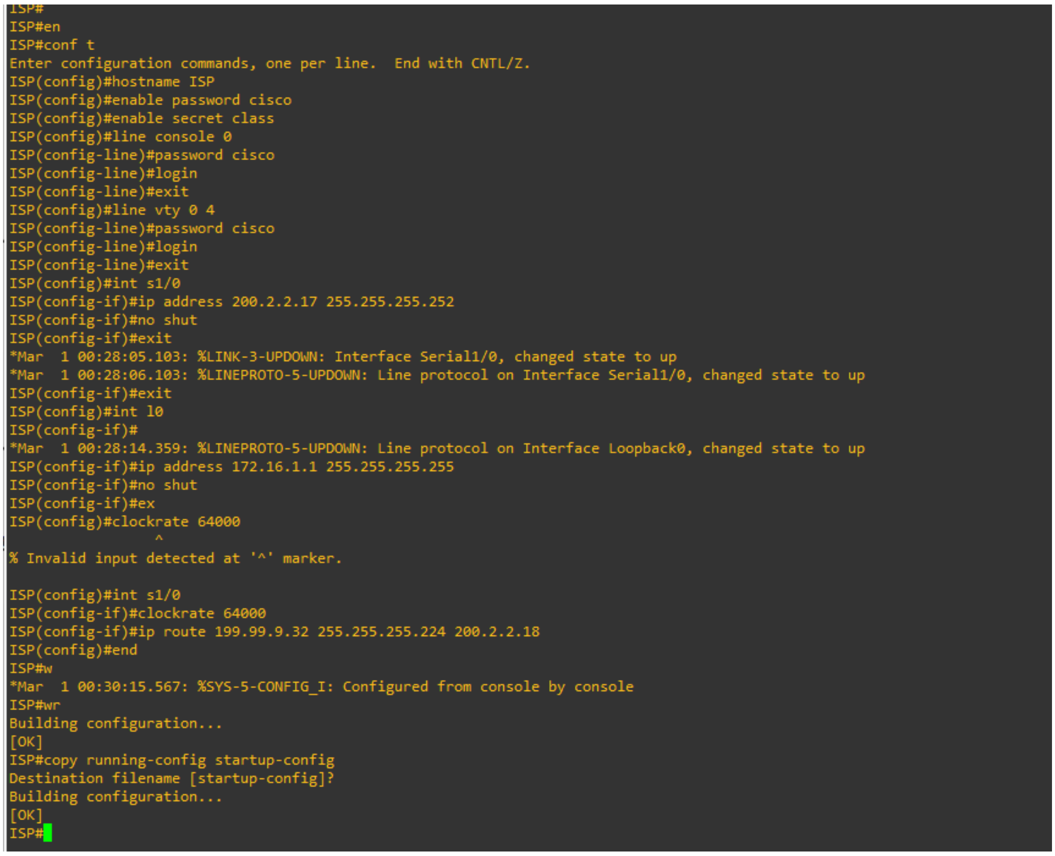
Step 2: Configure the network:

R1:



```
Gateway#
Gateway#en
Gateway#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Gateway(config)#hostname Gateway
Gateway(config)#enable password cisco
Gateway(config)#enable secret class
Gateway(config)#line console 0
Gateway(config-line)#password cisco
Gateway(config-line)#login
Gateway(config-line)#exit
Gateway(config)#line vty 0 4
Gateway(config-line)#password cisco
Gateway(config-line)#login
Gateway(config-line)#exit
Gateway(config)#int fa0/0
Gateway(config-if)#ip address 10.10.10.1 255.255.255.0
Gateway(config-if)#no shut
Gateway(config-if)#exit
Gateway(config)#
*Mar  1 00:05:33.807: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:05:34.807: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Gateway(config)#int s1/0
Gateway(config-if)#ip address 200.2.2.18 255.255.255.252
Gateway(config-if)#no shut
Gateway(config-if)#
*Mar  1 00:07:12.911: %LINK-3-UPDOWN: Interface Serial1/0, changed state to up
*Mar  1 00:07:13.911: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1/0, changed state to up
Gateway(config-if)#ip route 0.0.0.0 0.0.0.0
*Mar  1 00:07:43.643: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1/0, changed state to down
Gateway(config-if)#ip route 0.0.0.0 0.0.0.0 200.2.2.17
Gateway(config)#end
Gateway#
*Mar  1 00:08:38.255: %SYS-5-CONFIG_I: Configured from console by console
Gateway#wr
Building configuration...
[OK]
Gateway#
*Mar  1 00:12:13.631: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1/0, changed state to up
Gateway#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

R2:



```
ISP#
ISP#en
ISP#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ISP(config)#hostname ISP
ISP(config)#enable password cisco
ISP(config)#enable secret class
ISP(config)#line console 0
ISP(config-line)#password cisco
ISP(config-line)#login
ISP(config-line)#exit
ISP(config)#line vty 0 4
ISP(config-line)#password cisco
ISP(config-line)#login
ISP(config-line)#exit
ISP(config)#int s1/0
ISP(config-if)#ip address 200.2.2.17 255.255.255.252
ISP(config-if)#no shut
ISP(config-if)#exit
*Mar  1 00:28:05.103: %LINK-3-UPDOWN: Interface Serial1/0, changed state to up
*Mar  1 00:28:06.103: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1/0, changed state to up
ISP(config-if)#exit
ISP(config)#int l0
ISP(config-if)#
*Mar  1 00:28:14.359: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ISP(config-if)#ip address 172.16.1.1 255.255.255.255
ISP(config-if)#no shut
ISP(config-if)#ex
ISP(config)#clockrate 64000
                 ^
% Invalid input detected at '^' marker.

ISP(config)#int s1/0
ISP(config-if)#clockrate 64000
ISP(config-if)#ip route 199.99.9.32 255.255.255.224 200.2.2.18
ISP(config)#end
ISP#w
*Mar  1 00:30:15.567: %SYS-5-CONFIG_I: Configured from console by console
ISP#wr
Building configuration...
[OK]
ISP#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
ISP#
```

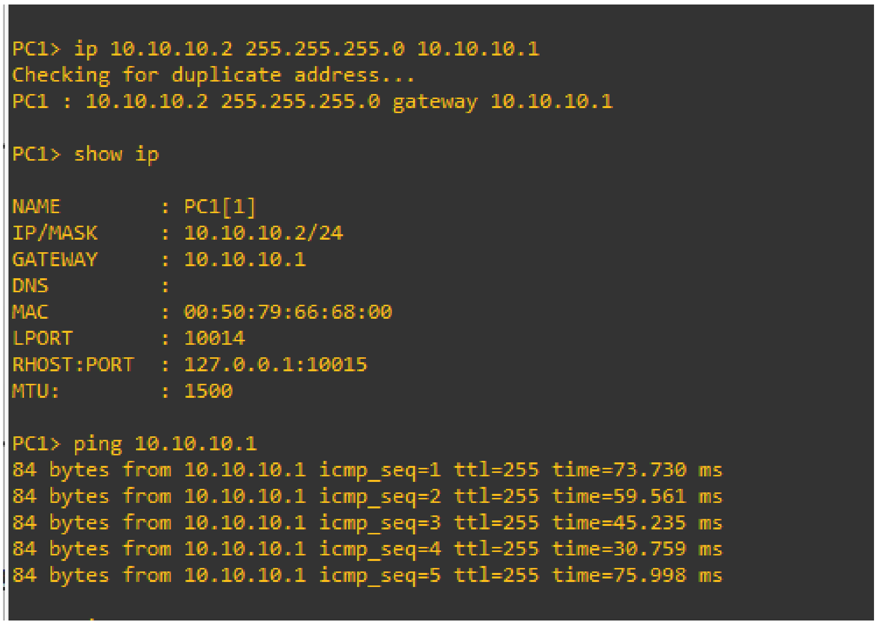
# enable password - it will enables a password that based on a clear text.

# enable secret - it will enables a password and password encryption that based on the md5 hashing algorithm.

The ISP has allocated a company the public CIDR IP address 199.99.9.32/30. So we are using as Static IP route in ISP.

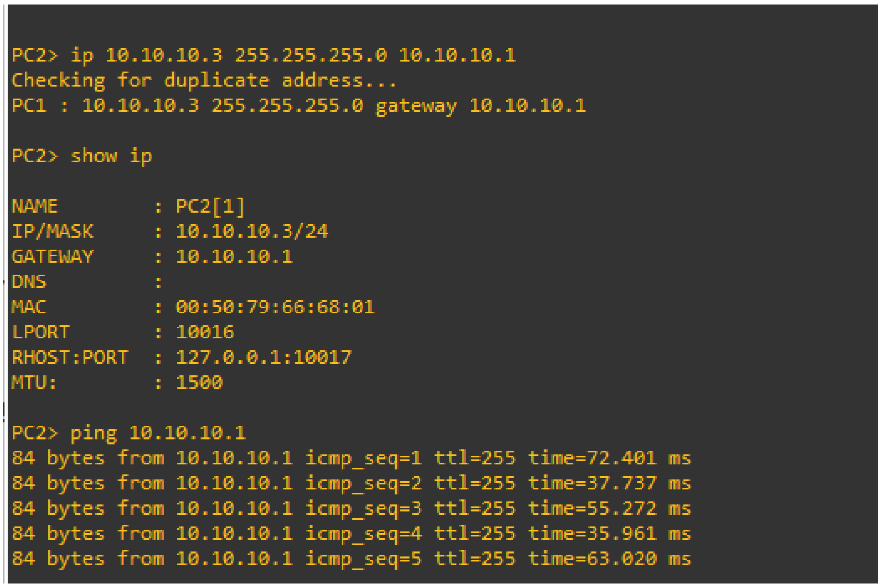
Step 3: Configure PC's and Ping the gateway router:

PC1:



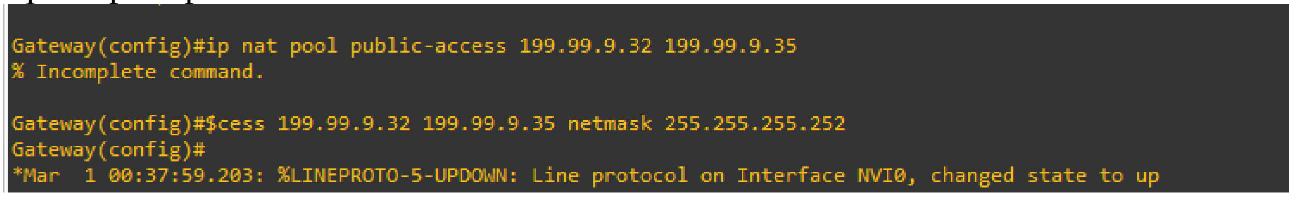
```
PC1> ip 10.10.10.2 255.255.255.0 10.10.10.1
Checking for duplicate address...
PC1 : 10.10.10.2 255.255.255.0 gateway 10.10.10.1

PC1> show ip

NAME        : PC1[1]
IP/MASK     : 10.10.10.2/24
GATEWAY     : 10.10.10.1
DNS         :
MAC         : 00:50:79:66:68:00
LPORT       : 10014
RHOST:PORT  : 127.0.0.1:10015
MTU:        : 1500

PC1> ping 10.10.10.1
84 bytes from 10.10.10.1 icmp_seq=1 ttl=255 time=73.730 ms
84 bytes from 10.10.10.1 icmp_seq=2 ttl=255 time=59.561 ms
84 bytes from 10.10.10.1 icmp_seq=3 ttl=255 time=45.235 ms
84 bytes from 10.10.10.1 icmp_seq=4 ttl=255 time=30.759 ms
84 bytes from 10.10.10.1 icmp_seq=5 ttl=255 time=75.998 ms
```

PC2:



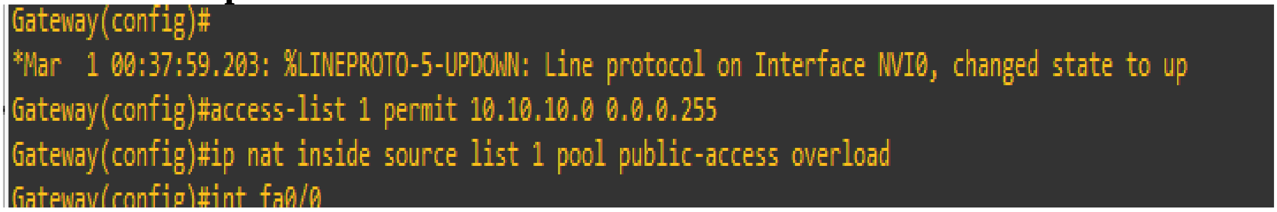
```
PC2> ip 10.10.10.3 255.255.255.0 10.10.10.1
Checking for duplicate address...
PC1 : 10.10.10.3 255.255.255.0 gateway 10.10.10.1

PC2> show ip

NAME        : PC2[1]
IP/MASK     : 10.10.10.3/24
GATEWAY     : 10.10.10.1
DNS         :
MAC         : 00:50:79:66:68:01
LPORT       : 10016
RHOST:PORT  : 127.0.0.1:10017
MTU:        : 1500

PC2> ping 10.10.10.1
84 bytes from 10.10.10.1 icmp_seq=1 ttl=255 time=72.401 ms
84 bytes from 10.10.10.1 icmp_seq=2 ttl=255 time=37.737 ms
84 bytes from 10.10.10.1 icmp_seq=3 ttl=255 time=55.272 ms
84 bytes from 10.10.10.1 icmp_seq=4 ttl=255 time=35.961 ms
84 bytes from 10.10.10.1 icmp_seq=5 ttl=255 time=63.020 ms
```

Step 4: Make a pool of IP Address which can be used as Public IP Address:
Run the command:
‘ip nat pool public-access 199.99.9.32 199.99.9.35 netmask 255.255.255.252’



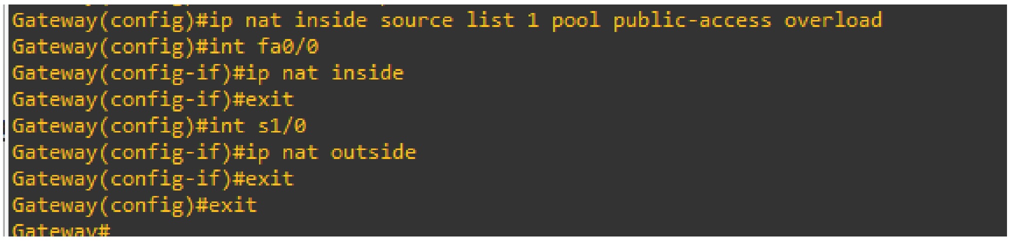
```
Gateway(config)#ip nat pool public-access 199.99.9.32 199.99.9.35
% Incomplete command.

Gateway(config)#$cess 199.99.9.32 199.99.9.35 netmask 255.255.255.252
Gateway(config)#
*Mar  1 00:37:59.203: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to up
```

Step 5: Make an access list that will map the public IP addresses to the inside private IP addresses. And Define the NAT translation from inside list to outside pool.



```
Gateway(config)#
*Mar  1 00:37:59.203: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to up
Gateway(config)#access-list 1 permit 10.10.10.0 0.0.0.255
Gateway(config)#ip nat inside source list 1 pool public-access overload
Gateway(config)#int fa0/0
```

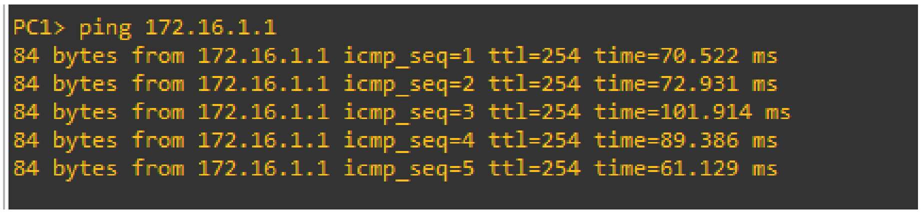
Step 6: Now we will define which interface is inside and which one is outside:



```
Gateway(config)#ip nat inside source list 1 pool public-access overload
Gateway(config)#int fa0/0
Gateway(config-if)#ip nat inside
Gateway(config-if)#exit
Gateway(config)#int s1/0
Gateway(config-if)#ip nat outside
Gateway(config-if)#exit
Gateway(config)#exit
Gateway#
```

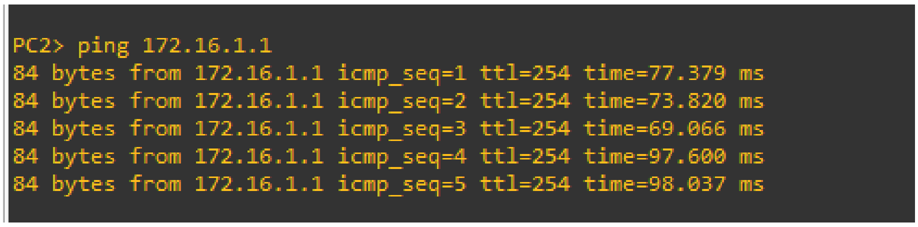
Step 7: Now ping from the PC to the loopback of ISP

PC1:



```
PC1> ping 172.16.1.1
84 bytes from 172.16.1.1 icmp_seq=1 ttl=254 time=70.522 ms
84 bytes from 172.16.1.1 icmp_seq=2 ttl=254 time=72.931 ms
84 bytes from 172.16.1.1 icmp_seq=3 ttl=254 time=101.914 ms
84 bytes from 172.16.1.1 icmp_seq=4 ttl=254 time=89.386 ms
84 bytes from 172.16.1.1 icmp_seq=5 ttl=254 time=61.129 ms
```

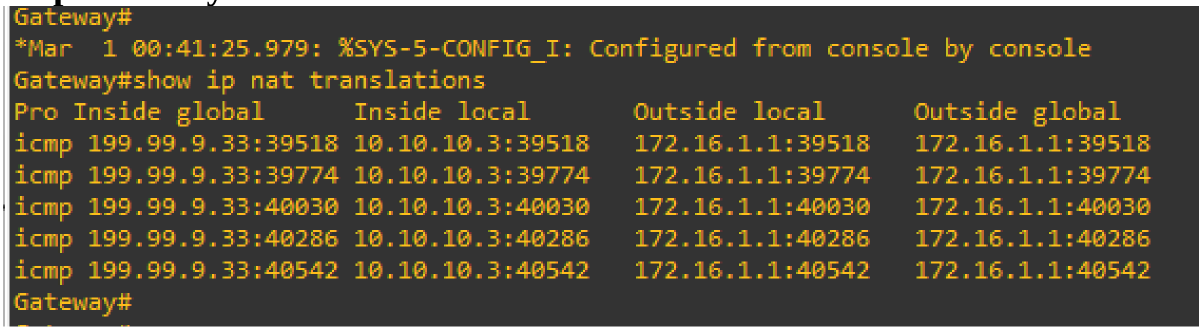
PC2:



```
PC2> ping 172.16.1.1
84 bytes from 172.16.1.1 icmp_seq=1 ttl=254 time=77.379 ms
84 bytes from 172.16.1.1 icmp_seq=2 ttl=254 time=73.820 ms
84 bytes from 172.16.1.1 icmp_seq=3 ttl=254 time=69.066 ms
84 bytes from 172.16.1.1 icmp_seq=4 ttl=254 time=97.600 ms
84 bytes from 172.16.1.1 icmp_seq=5 ttl=254 time=98.037 ms
```

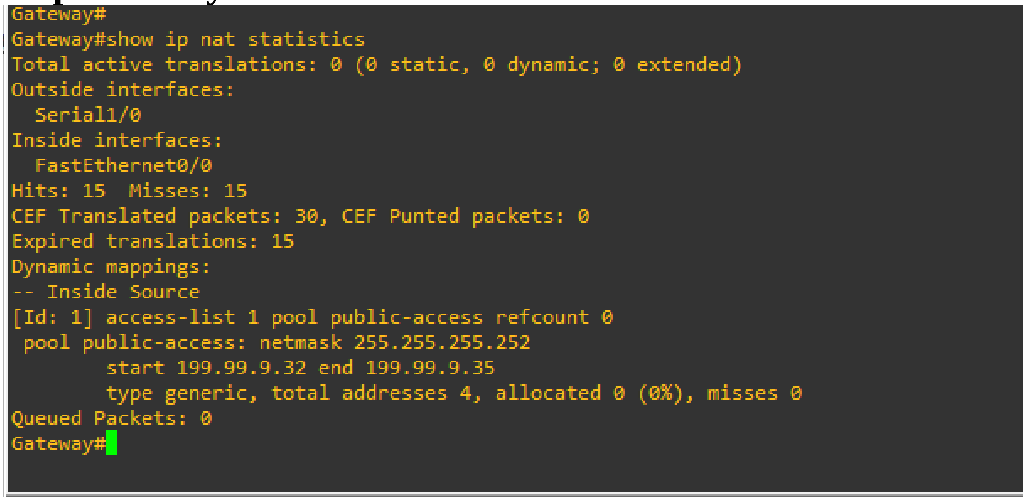
If we pinged before NAT, it won’t have worked.
Because they don’t know public or private IP Address.

Step 8: Verify NAT and PAT Translations:



```
Gateway#
*Mar  1 00:41:25.979: %SYS-5-CONFIG_I: Configured from console by console
Gateway#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
icmp 199.99.9.33:39518 10.10.10.3:39518   172.16.1.1:39518   172.16.1.1:39518
icmp 199.99.9.33:39774 10.10.10.3:39774   172.16.1.1:39774   172.16.1.1:39774
icmp 199.99.9.33:40030 10.10.10.3:40030   172.16.1.1:40030   172.16.1.1:40030
icmp 199.99.9.33:40286 10.10.10.3:40286   172.16.1.1:40286   172.16.1.1:40286
icmp 199.99.9.33:40542 10.10.10.3:40542   172.16.1.1:40542   172.16.1.1:40542
Gateway#
```

Step 9: Verify NAT and PAT Statistics:



```
Gateway#
Gateway#show ip nat statistics
Total active translations: 0 (0 static, 0 dynamic; 0 extended)
Outside interfaces:
  Serial1/0
Inside interfaces:
  FastEthernet0/0
Hits: 15  Misses: 15
CEF Translated packets: 30, CEF Punted packets: 0
Expired translations: 15
Dynamic mappings:
-- Inside Source
[Id: 1] access-list 1 pool public-access refcount 0
 pool public-access: netmask 255.255.255.252
        start 199.99.9.32 end 199.99.9.35
        type generic, total addresses 4, allocated 0 (0%), misses 0
Queued Packets: 0
Gateway#
```