Vivek Ponnala

COEN 146 Lab 8

Report

Introduction:

GNS is a network emulator which allows us to test connections between routers, PCs, cloud, switches, etc. It is useful in understanding how connections can be established between different devices on different networks.

Objective:

Create a network topology with an access network and ISP to understand how PCs in the access network connect to ISP.

Description:

Install the GNS software and select operating system for your computer to download by visiting http://www.gns3.com/software/download.

Get Cisco router IOS images from https://protechgurus.com/download-gns3-ios-images/. When adding a router, set the idle PC value as 0x6050b114.

To add switches, locate the vertical taskbar, which says browse switches. For most network topologies, ethernet switches are used.

To add PCs, simply locate the PC image on the vertical taskbar and select the virtual PC option.

To create a link, simply select the link option on the vertical taskbar and then click on the device and choose fast ethernet.

First, open custom console and add IP addresses.

To add IP addresses to PC, type ip (ip addr/mask) gateway

To add IP addresses to router, first config t, then locate to interface fa0/0 or fa0/1. Then type ip addr (ip address) mask (255.255.255.0).

To view the IP address in PC, type show ip.

To view the IP address in router, type show ip interface brief.

To ensure, IP address is up and running, type no shut.

To route between routers, type route 0.0.0.0 ip_addr

To ping between devices type ping ip_addr.

To create a nat interface for router, first type nat inside interface, then type nat outside the interface. Then create range of IP addresses and then type debug ip nat.

Part 1: IP addressing scheme

The next step is to create a network topology with an IP addressing scheme for all the PCs and routers in the network.

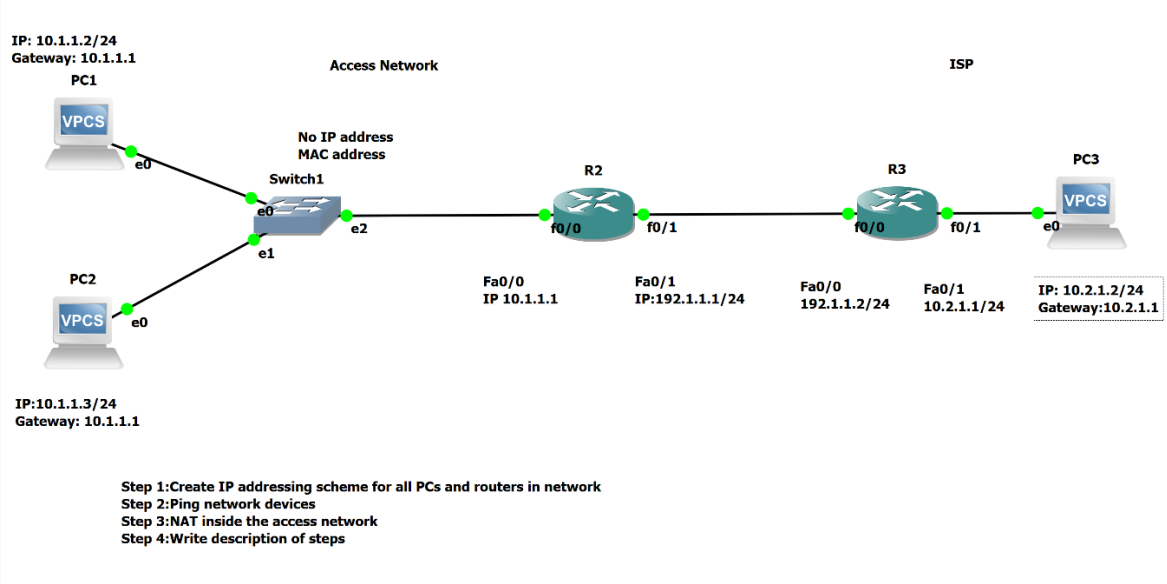

Figure 1: Network Topology

Then after the network topology is drawn, create an IP address and gateway for PC and ip address with mask for router.

```
PC1> show ip

NAME        : PC1[1]
IP/MASK     : 10.1.1.2/24
GATEWAY     : 10.1.1.1
DNS         :
MAC         : 00:50:79:66:68:00
LPORT       : 10020
RHOST:PORT  : 127.0.0.1:10021
MTU:        : 1500
```

Figure 2: PC1 IP addressing scheme

```
PC2> show ip

NAME        : PC2[1]
IP/MASK     : 10.1.1.3/24
GATEWAY     : 10.1.1.1
DNS         :
MAC         : 00:50:79:66:68:01
LPORT       : 10018
RHOST:PORT  : 127.0.0.1:10019
MTU:        : 1500
```

Figure 3: PC2 IP addressing scheme

```
PC3> show ip

NAME        : PC3[1]
IP/MASK     : 10.2.1.2/24
GATEWAY     : 10.2.1.1
DNS         :
MAC         : 00:50:79:66:68:02
LPORT       : 10022
RHOST:PORT  : 127.0.0.1:10023
MTU:        : 1500
```

Figure 4: PC3 IP addressing scheme

```
*Mar  1 00:29:44.687: %SYS-5-CONFIG_I: Configured from console by console
R2#show ip interface brief
Interface                  IP-Address      OK? Method Status                Prot
ocol
FastEthernet0/0            10.1.1.1        YES manual up                    up

Serial0/0                  unassigned      YES unset  administratively down down

FastEthernet0/1            192.1.1.1       YES manual up                    up

Serial0/1                  unassigned      YES unset  administratively down down
```

Figure 5: R2 IP addressing scheme

```
R3#show ip interface brief
Interface                  IP-Address      OK? Method Status                Prot
ocol
FastEthernet0/0            192.1.1.2       YES manual up                    up

Serial0/0                  unassigned      YES unset  administratively down down

FastEthernet0/1            10.2.1.1        YES manual up                    up

Serial0/1                  unassigned      YES unset  administratively down down
```

Figure 6: R3 IP addressing scheme

Part 2: Ping accessibility

The next step is to check the accessibility of all PCs from PC1. Essentially, in this step, we ping to the IP addresses of PC2, router 2, router 3 and PC3. Initially, packets can only be sent to PC2 and router 2 since there is no connection between router 2 and router 3 through which PC1 can connect.

```
PC1> ping 10.1.1.3
84 bytes from 10.1.1.3 icmp_seq=1 ttl=64 time=0.440 ms
84 bytes from 10.1.1.3 icmp_seq=2 ttl=64 time=0.963 ms
84 bytes from 10.1.1.3 icmp_seq=3 ttl=64 time=0.642 ms
84 bytes from 10.1.1.3 icmp_seq=4 ttl=64 time=0.575 ms
84 bytes from 10.1.1.3 icmp_seq=5 ttl=64 time=0.594 ms

PC1> ping 10.2.1.2
*10.1.1.1 icmp_seq=1 ttl=255 time=10.195 ms (ICMP type:3, code:1, Destination ho
st unreachable)
*10.1.1.1 icmp_seq=2 ttl=255 time=10.329 ms (ICMP type:3, code:1, Destination ho
st unreachable)
*10.1.1.1 icmp_seq=3 ttl=255 time=1.456 ms (ICMP type:3, code:1, Destination hos
t unreachable)
*10.1.1.1 icmp_seq=4 ttl=255 time=10.398 ms (ICMP type:3, code:1, Destination ho
st unreachable)
*10.1.1.1 icmp_seq=5 ttl=255 time=10.292 ms (ICMP type:3, code:1, Destination ho
st unreachable)

PC1> ping 10.1.1.1
84 bytes from 10.1.1.1 icmp_seq=1 ttl=255 time=2.500 ms
84 bytes from 10.1.1.1 icmp_seq=2 ttl=255 time=4.289 ms
84 bytes from 10.1.1.1 icmp_seq=3 ttl=255 time=5.410 ms
84 bytes from 10.1.1.1 icmp_seq=4 ttl=255 time=20.887 ms
84 bytes from 10.1.1.1 icmp_seq=5 ttl=255 time=2.318 ms

PC1> ping 192.1.1.1
84 bytes from 192.1.1.1 icmp_seq=1 ttl=255 time=5.361 ms
84 bytes from 192.1.1.1 icmp_seq=2 ttl=255 time=20.635 ms
84 bytes from 192.1.1.1 icmp_seq=3 ttl=255 time=20.434 ms
84 bytes from 192.1.1.1 icmp_seq=4 ttl=255 time=20.220 ms
84 bytes from 192.1.1.1 icmp_seq=5 ttl=255 time=3.415 ms

PC1> ping 192.1.1.2
192.1.1.2 icmp_seq=1 timeout
192.1.1.2 icmp_seq=2 timeout
192.1.1.2 icmp_seq=3 timeout
192.1.1.2 icmp_seq=4 timeout
192.1.1.2 icmp_seq=5 timeout

PC1> ping 10.2.1.1
*10.1.1.1 icmp_seq=1 ttl=255 time=10.547 ms (ICMP type:3, code:1, Destination ho
st unreachable)
*10.1.1.1 icmp_seq=2 ttl=255 time=5.257 ms (ICMP type:3, code:1, Destination hos
t unreachable)
*10.1.1.1 icmp_seq=3 ttl=255 time=3.429 ms (ICMP type:3, code:1, Destination hos
t unreachable)
*10.1.1.1 icmp_seq=4 ttl=255 time=8.429 ms (ICMP type:3, code:1, Destination hos
t unreachable)
*10.1.1.1 icmp_seq=5 ttl=255 time=2.198 ms (ICMP type:3, code:1, Destination hos
t unreachable)
```

Figure 7: Initial Ping accessibility from PC1

Due to this, we have to create a route between R2 and R3. The IP address for FastEthernet0/0 of R3 is 192.1.1.2 and the IP address for FastEthernet0/1 of R2 is 192.1.1.1.

```
R2(config)#interface fa0/1
R2(config-if)#ip route 0.0.0.0 0.0.0.0 192.1.1.2
```

Figure 8: R2 route to R3

```
R3(config)#interface fa0/0
R3(config-if)#ip route 0.0.0.0 0.0.0.0 192.1.1.1
```

Figure 9: R3 route to R2

Now, checking ping after configuration via R2 and R3 route.

```
PC1> ping 192.1.1.2
84 bytes from 192.1.1.2 icmp_seq=1 ttl=254 time=31.408 ms
84 bytes from 192.1.1.2 icmp_seq=2 ttl=254 time=25.288 ms
84 bytes from 192.1.1.2 icmp_seq=3 ttl=254 time=23.280 ms
84 bytes from 192.1.1.2 icmp_seq=4 ttl=254 time=31.331 ms
84 bytes from 192.1.1.2 icmp_seq=5 ttl=254 time=29.215 ms

PC1> ping 10.2.1.1
84 bytes from 10.2.1.1 icmp_seq=1 ttl=254 time=30.191 ms
84 bytes from 10.2.1.1 icmp_seq=2 ttl=254 time=31.449 ms
84 bytes from 10.2.1.1 icmp_seq=3 ttl=254 time=26.440 ms
84 bytes from 10.2.1.1 icmp_seq=4 ttl=254 time=30.410 ms
84 bytes from 10.2.1.1 icmp_seq=5 ttl=254 time=28.288 ms

PC1> ping 10.2.1.2
10.2.1.2 icmp_seq=1 timeout
84 bytes from 10.2.1.2 icmp_seq=2 ttl=62 time=43.340 ms
84 bytes from 10.2.1.2 icmp_seq=3 ttl=62 time=40.143 ms
84 bytes from 10.2.1.2 icmp_seq=4 ttl=62 time=43.260 ms
84 bytes from 10.2.1.2 icmp_seq=5 ttl=62 time=40.260 ms

PC1> ping 10.2.1.2
84 bytes from 10.2.1.2 icmp_seq=1 ttl=62 time=41.458 ms
84 bytes from 10.2.1.2 icmp_seq=2 ttl=62 time=35.248 ms
84 bytes from 10.2.1.2 icmp_seq=3 ttl=62 time=34.153 ms
84 bytes from 10.2.1.2 icmp_seq=4 ttl=62 time=40.320 ms
84 bytes from 10.2.1.2 icmp_seq=5 ttl=62 time=39.232 ms
```

Figure 10: Final Ping accessibility from PC1

Now, as you can see PC1 can connect to all devices in the ISP from the access network after the router connection has been established.

Part 3: NAT table of connection between PC1 or PC2 and PC3

The last step is to see how network address translation works from R2, which is a router in the access network when PC1 or PC2 ping to PC3. In the first part, we need to NAT inside FastEthernet0/0 of R2 and NAT outside FastEthernet0/1 of R3 to establish connection between access network and ISP.

```
R2(config)#interface fa0/0
R2(config-if)#ip nat inside
R2(config-if)#
*Mar  1 00:33:32.523: %LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, chan
ged state to up
R2(config-if)#interface fa0/1
R2(config-if)#ip nat outside
R2(config-if)#exit
```

Figure 11: NAT setup

Now, we need to the NAT table to know which IP addresses are permissible for transmission of packets from PC1 or PC2 to PC3 and then we turn NAT debugging on.

```
R2(config)#access-list 10 permit 10.1.1.0 0.0.0.255
R2(config)#ip nat inside source list 10 interface fa0/1 overload
R2(config)#debug ip nat
            ^
% Invalid input detected at '^' marker.

R2(config)#exit
R2#debug i
*Mar  1 00:35:58.607: %SYS-5-CONFIG_I: Configured from console by console
R2#debug ip nat
IP NAT debugging is on
```

Figure 12: Debug IP NAT

Then, we can ping PC1 to PC3 to view the NAT table. First, we ping PC1 to PC3.

```
PC1> ping 10.2.1.2
84 bytes from 10.2.1.2 icmp_seq=1 ttl=62 time=41.458 ms
84 bytes from 10.2.1.2 icmp_seq=2 ttl=62 time=35.248 ms
84 bytes from 10.2.1.2 icmp_seq=3 ttl=62 time=34.153 ms
84 bytes from 10.2.1.2 icmp_seq=4 ttl=62 time=40.320 ms
84 bytes from 10.2.1.2 icmp_seq=5 ttl=62 time=39.232 ms
```

Figure 13: PC1 to PC3 ping

In the last step, we see the NAT table from R2.

```
R2#
*Mar  1 00:37:55.975: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [21901]
*Mar  1 00:37:55.999: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [21901]
R2#
*Mar  1 00:37:57.011: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [21902]
*Mar  1 00:37:57.035: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [21902]
R2#
*Mar  1 00:37:58.047: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [21903]
*Mar  1 00:37:58.075: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [21903]
R2#
*Mar  1 00:37:59.091: NAT*: s=10.1.1.2->192.1.1.1, d=10.2.1.2 [21904]
*Mar  1 00:37:59.119: NAT*: s=10.2.1.2, d=192.1.1.1->10.1.1.2 [21904]
R2#
```

Figure 14: NAT table of R2 for connection between PC1 and PC3

As we can observe in the NAT table, when a ping connection is made, for any packet, the initial source is PC1 with IP address 10.1.1.2. Then, this passes through fast ethernet 0/1 of R2 and goes to PC3 with IP address 10.2.1.2. Then, PC3 acknowledges it has received the packet and sends this acknowledgement. In this case, PC3 is the source with IP address 10.2.1.2 which travels through the same fast ethernet cable of IP address 192.1.1.1 and goes back to the PC1 which is the initial source with IP address 10.1.1.2.

Conclusion:

This lab was helpful in understanding how GNS works. Primarily, I understood how to statically assign IP addresses to routers and PCs. Secondly, I understood how to enroute between routers and establish connection with devices outside of network. Lastly, I understood how NAT translation works and why it is required between different networks.