

## CN Lab 7 & 8

Name – Vedant Rishi Das  
Sec – D  
Batch – D2  
Roll No. -25  
Reg No. - 190905160

Q1) Make Vlan for the given network topology where PC1 and PC2 are part of Vlan 2, PC3 and PC4 are part of Vlan 3, router is part of Vlan1 and has sub-interfaces.

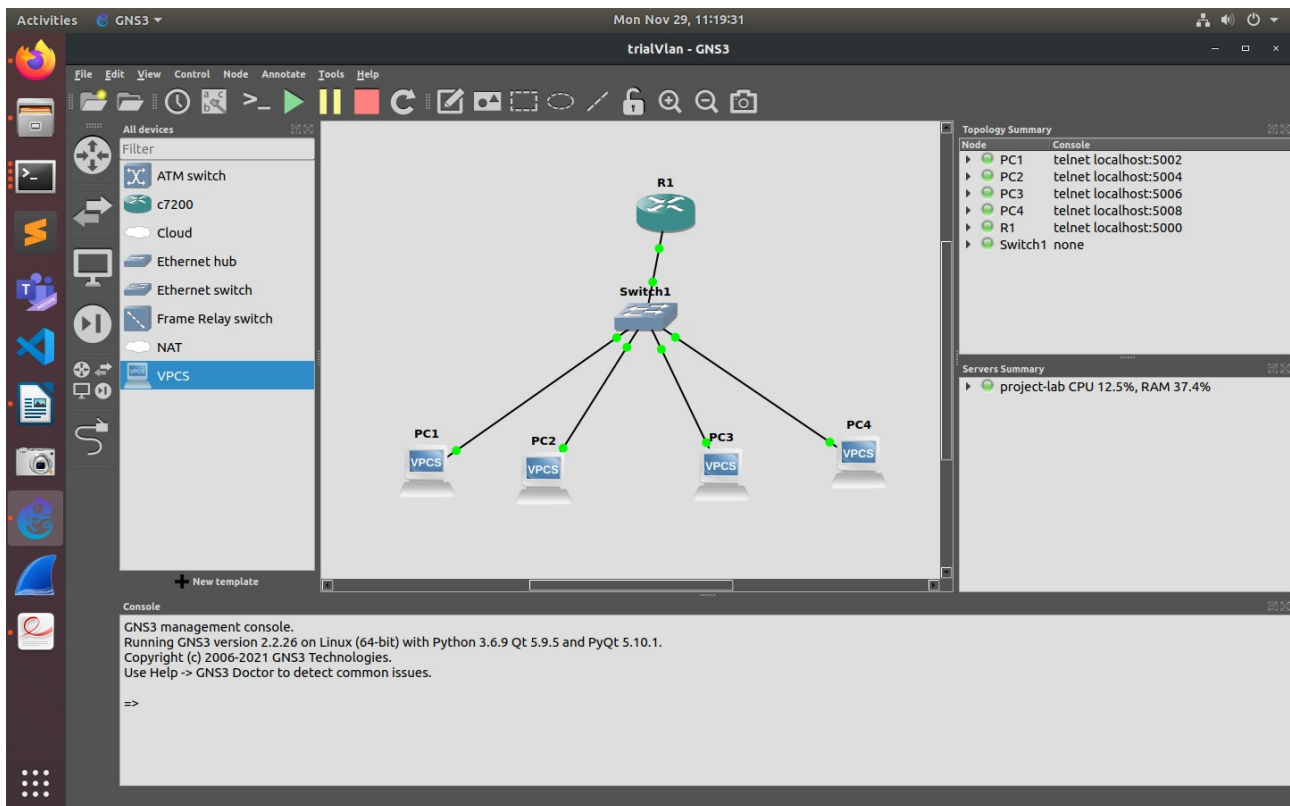


Fig1. Network topology

First in the switch we configure the required ports as seen in Fig.2. Type dot1q is assigned for vlan1 to enable it to have sub-interfaces for multiple vlans. Rest all are set as access to let only one at a time.

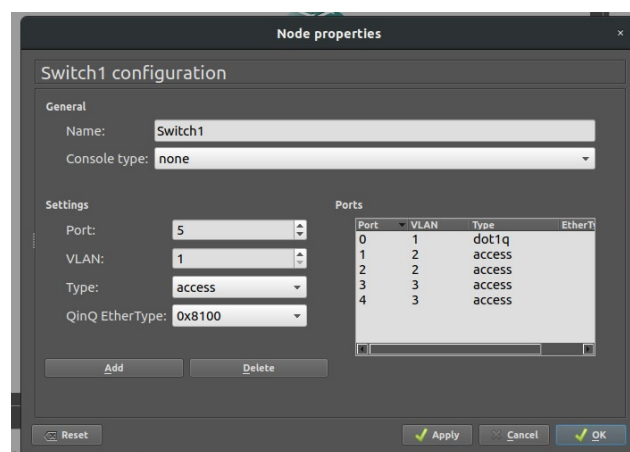


Fig.2 Switch configuration

```

R1#enable
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int f0/0
R1(config-if)#ip address 192.168.10.1
% Incomplete command.

R1(config-if)#ip address 192.168.10.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#int
*Nov 29 05:44:04.975: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
R1(config-if)#int f0
*Nov 29 05:44:04.975: %ENTITY_ALARM-6-INFO: CLEAR INFO Fa0/0 Physical Port Administrative State Down
*Nov 29 05:44:05.975: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config-if)#int f0/0.2
R1(config-subif)#encapsulation dot1q 2
R1(config-subif)#ip address 192.168.1.65 255.255.255.192
R1(config-subif)#no shut
R1(config-subif)#int f0/0.3
R1(config-subif)#encapsulation dot1q 3
R1(config-subif)#ip address 192.168.1.129 255.255.255.224
R1(config-subif)#no shut
R1(config-subif)#exit
R1(config)#exit
R1#
*Nov 29 05:45:44.067: %SYS-5-CONFIG_I: Configured from console by console
R1#

```

Fig.3 Configuring router for main interface f0/0 and subinterfaces f0/0.2 and f0/0.3

In Fig.3 we can see the configuration of router to have a main interface f0/0 and multiple sub-interfaces f0/0.2 and f0/0.3.

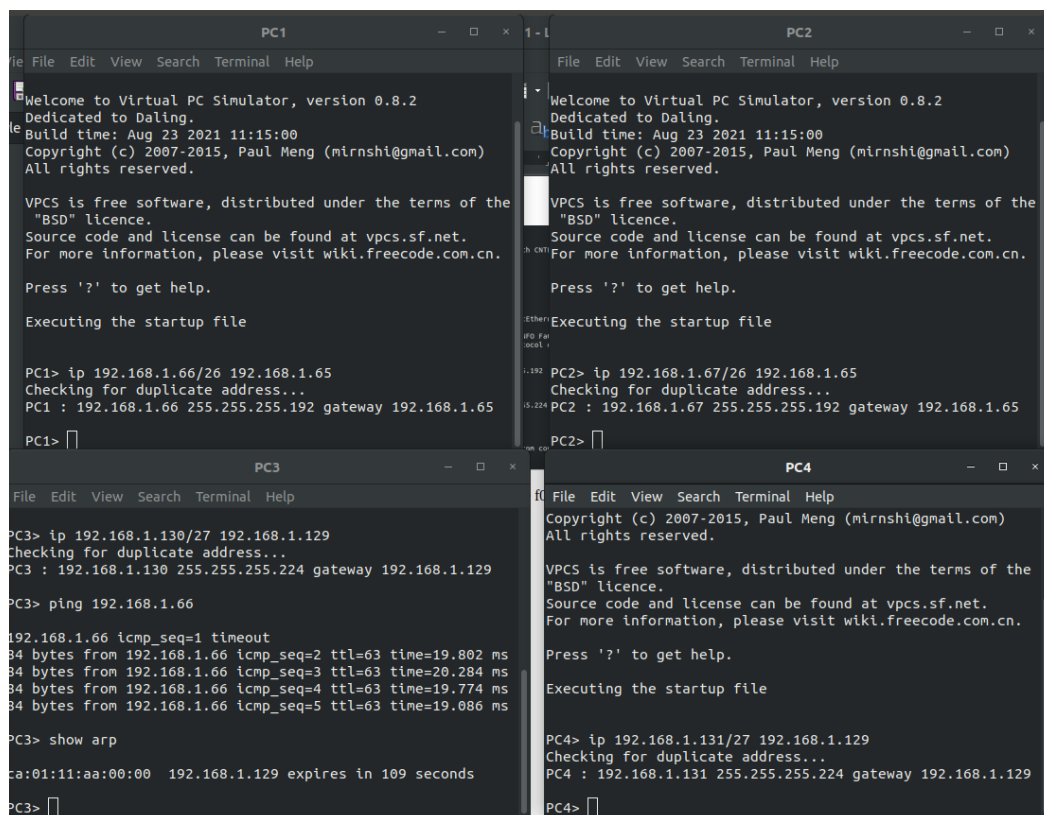


Fig.4 Configuring PCs

Default interface f0/0 = 192.168.10.1  
 Default gateway for vlan id 2 = 192.168.1.65  
 Default gateway for vlan id 3 = 192.168.1.129

```

PC3
File Edit View Search Terminal Help

PC3> ip 192.168.1.130/27 192.168.1.129
Checking for duplicate address...
PC3 : 192.168.1.130 255.255.255.224 gateway 192.168.1.129

PC3> ping 192.168.1.66

192.168.1.66 icmp_seq=1 timeout
84 bytes from 192.168.1.66 icmp_seq=2 ttl=63 time=19.802 ms
84 bytes from 192.168.1.66 icmp_seq=3 ttl=63 time=20.284 ms
84 bytes from 192.168.1.66 icmp_seq=4 ttl=63 time=19.774 ms
84 bytes from 192.168.1.66 icmp_seq=5 ttl=63 time=19.086 ms

PC3> show arp

ca:01:11:aa:00:00 192.168.1.129 expires in 109 seconds

PC3>

```

Fig.5 Pinging PC1 from PC3 which are on separate vlans

PC1 and PC3 are on separate vlan networks, but since they share a common router and the router supports multiple sub-interfaces we are able to ping them across vlans. As we can see in the ARP table, the default gateway for PC3 is saved since that helps it resolve to other vlans.

```

R1#ping 192.168.1.66

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.66, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/47/64 ms
R1#ping 192.168.1.130

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.130, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/46/60 ms
R1#show arp

```

Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	192.168.1.65	-	ca01.11aa.0000	ARPA	FastEthernet0/0.2
Internet	192.168.1.66	10	0050.7966.6800	ARPA	FastEthernet0/0.2
Internet	192.168.10.1	-	ca01.11aa.0000	ARPA	FastEthernet0/0
Internet	192.168.1.129	-	ca01.11aa.0000	ARPA	FastEthernet0/0.3
Internet	192.168.1.130	10	0050.7966.6802	ARPA	FastEthernet0/0.3

```

R1#

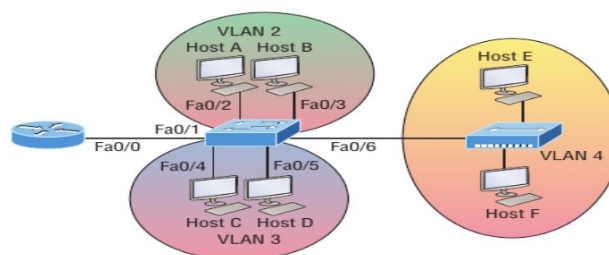
```

Fig.6 Ping from Router to PC1 and PC3

In Fig.6 we can see the router pinging to PC1 and PC3. The arp table is more extensive here as it has to resolve between the multiple sub-interfaces that it has. It contains the PC IP addresses directly and also the default gateway addresses.

Q2) (from exercise)

Configure the given network topology in GNS-3 and show it's working



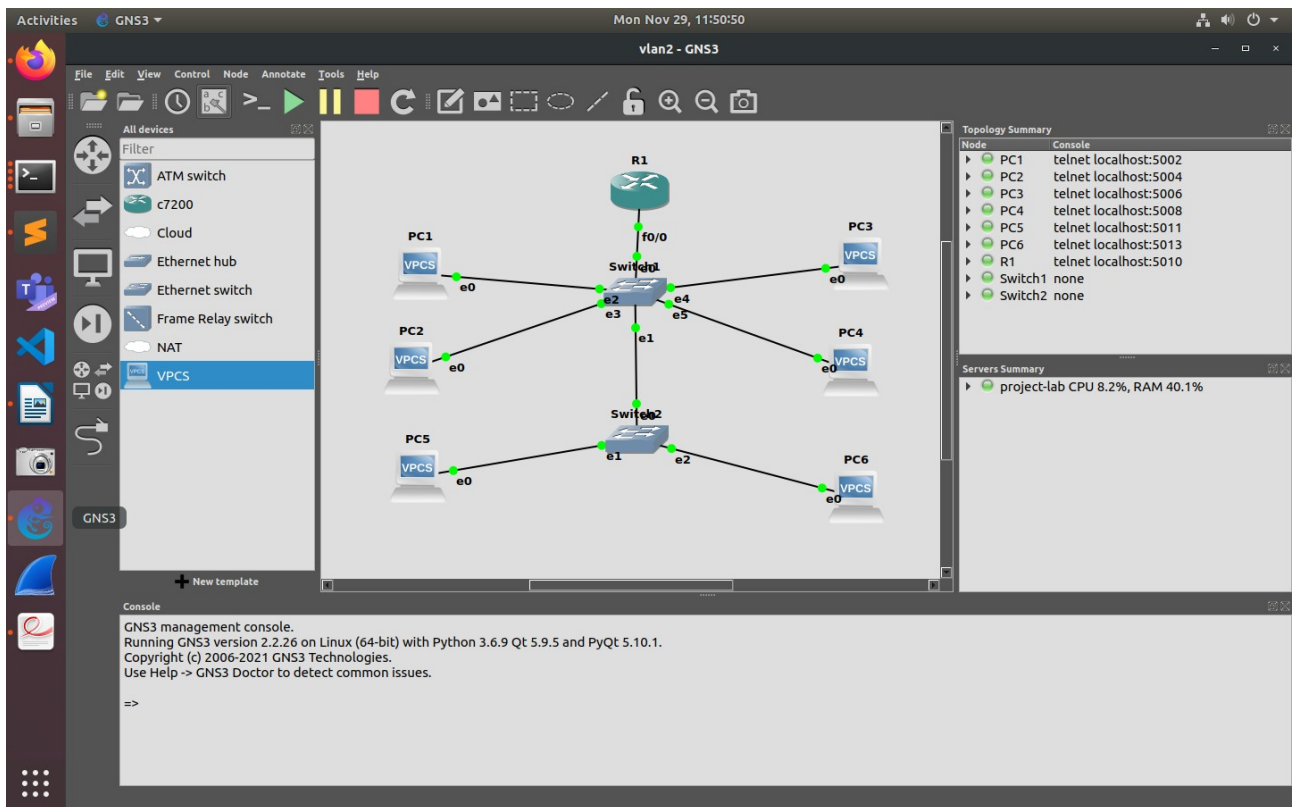


Fig.1 Network topology on GNS

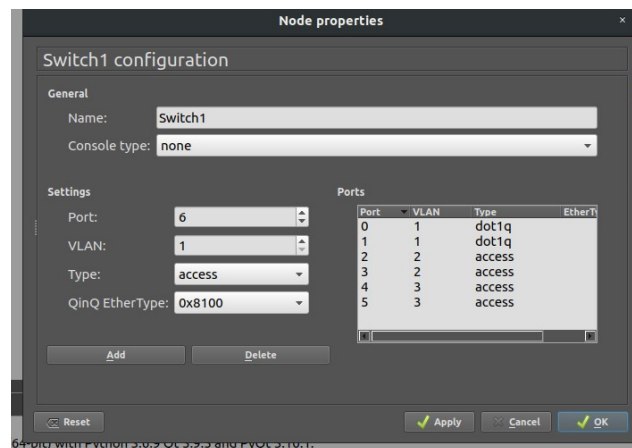


Fig.2 Switch1 Configuration

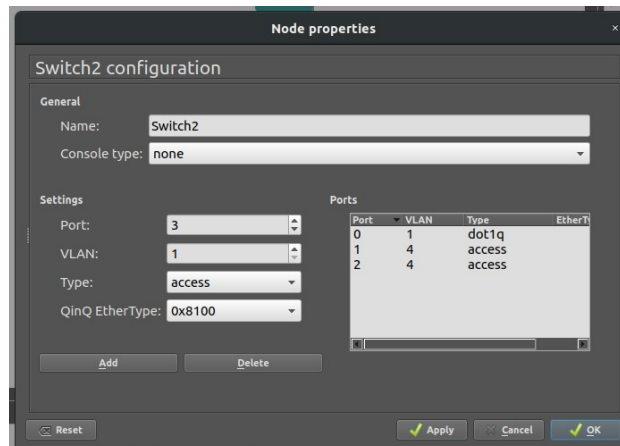


Fig.3 Switch2 Configuration

In the given network topology as in Fig.1, we have 2 switches connected to multiple vlans and 1 router. So an additional dot1q port is added to switch 1 with vlan id 1 so that it can be connected to switch 2 and they are part of the same vlan id network. Now as seen in Fig.3 there are 3 sub-interfaces for 3 different vlans. So all have to be configured according to a chosen block of addresses.

```

R1#enable
R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int f0/0
R1(config-if)#ip address 192.168.10.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#!
*Nov 29 06:06:48.011: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
R1(config-if)#int
*Nov 29 06:06:48.011: %ENTITY_ALARM-6-INFO: CLEAR INFO Fa0/0 Physical Port Administrative State Down
*Nov 29 06:06:49.011: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config-if)#int f0/0.2
R1(config-subif)#encapsulation dot1q 2
R1(config-subif)#ip address 192.168.1.65 255.255.255.192
R1(config-subif)#no shut
R1(config-subif)#int f0/0.3
R1(config-subif)#encapsulation dot1q 3
R1(config-subif)#ip address 192.168.1.129 255.255.255.224
R1(config-subif)#no shut
R1(config-subif)#int f0/0.4
R1(config-subif)#encapsulation dot1q 4
R1(config-subif)#ip address 192.168.1.161 255.255.255.240
R1(config-subif)#no shut
R1(config-subif)#exit
R1(config-subif)#exit
R1#
*Nov 29 06:15:01.599: %SYS-5-CONFIG_I: Configured from console by console
R1#

```

Fig. 3 Configuring R1 with main interface and 3 sub interfaces

Fig.4 contains the configuration of all PCs in the network topology with their respective default gateways. Fig.5 contains the ping from PC6 which is on vlan id 4 to PC1 which is on vlan id 2. It is a successful ping with the ARP table showing PC6 has to access its default gateway.

Default interface f0/0 = 192.168.10.1  
 Default gateway for vlan id 2 = 192.168.1.65  
 Default gateway for vlan id 3 = 192.168.1.129  
 Default gateway for vlan id 4 = 192.168.1.161



```
190905160 lab7 CN.odt - LibreOffice Writer
PC1> ip 192.168.1.66/26 192.168.1.65
Checking for duplicate address...
PC1 : 192.168.1.66 255.255.255.192 gateway 192.168.1.65
PC1>

PC2> ip 192.168.1.67/26 192.168.1.65
Checking for duplicate address...
PC2 : 192.168.1.67 255.255.255.192 gateway 192.168.1.65
PC2>

PC3> ip 192.168.1.130/27 192.168.1.129
Checking for duplicate address...
PC3 : 192.168.1.130 255.255.255.224 gateway 192.168.1.129
PC3>

PC4> ip 192.168.1.131/27 192.168.1.129
Checking for duplicate address...
PC4 : 192.168.1.131 255.255.255.224 gateway 192.168.1.129
PC4>

PC5> ip 192.168.1.162/28 192.168.1.161
Checking for duplicate address...
PC5 : 192.168.1.162 255.255.255.240 gateway 192.168.1.161
PC5>

PC6> ip 192.168.1.163/28 192.168.1.161
Checking for duplicate address...
PC6 : 192.168.1.163 255.255.255.240 gateway 192.168.1.161
PC6> ping 192.168.1.66
192.168.1.66 icmp_seq=1 timeout
84 bytes from 192.168.1.66 icmp_seq=2 ttl=63 time=17.726 ms
84 bytes from 192.168.1.66 icmp_seq=3 ttl=63 time=19.730 ms
84 bytes from 192.168.1.66 icmp_seq=4 ttl=63 time=19.826 ms
84 bytes from 192.168.1.66 icmp_seq=5 ttl=63 time=18.631 ms
PC6> show arp
ca:01:13:dc:00:00 192.168.1.161 expires in 23 seconds
PC6>
```

Fig. 4 Configuring all PCs on the network

```
PC6> ping 192.168.1.66

192.168.1.66 icmp_seq=1 timeout
84 bytes from 192.168.1.66 icmp_seq=2 ttl=63 time=17.726 ms
84 bytes from 192.168.1.66 icmp_seq=3 ttl=63 time=19.730 ms
84 bytes from 192.168.1.66 icmp_seq=4 ttl=63 time=19.826 ms
84 bytes from 192.168.1.66 icmp_seq=5 ttl=63 time=18.631 ms

PC6> show arp

ca:01:13:dc:00:00 192.168.1.161 expires in 23 seconds
PC6>
```

Fig. 5 Pinging PC1 from PC6 and showing ARP table

```
IP Multicast Fast switching is enabled
R1#show ip inter brief
Interface      IP-Address      OK? Method Status      Protocol
FastEthernet0/0 192.168.10.1    YES manual up          up
FastEthernet0/0.2 192.168.1.65    YES manual up          up
FastEthernet0/0.3 192.168.1.129   YES manual up          up
FastEthernet0/0.4 192.168.1.161   YES manual up          up
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
R1#
```

Fig.6 Interface IP address for router 1

Mon Nov 29, 12:35:12

Capturing from Standard Input [Switch1 Ethernet0 to R1 FastEthernet0/0]

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/> Expression...

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	ca:01:13:dc:00:00	ca:01:13:dc:00:00	LOOP	60	Reply
2	4.696391	ca:01:13:dc:00:00	ca:01:13:dc:00:00	CDP	357	Device ID: R1 Port ID: FastEthernet0/0
3	7.089410	Private_66:68:05	Broadcast	ARP	68	Who has 192.168.1.161? Tell 192.168.1.163
4	7.090621	ca:01:13:dc:00:00	Private_66:68:05	ARP	64	192.168.1.161 is at ca:01:13:dc:00:00
5	7.091473	192.168.1.163	192.168.1.66	ICMP	192	Echo (ping) request id=0x937b, seq=1/256, ttl=64 (no response found!)
6	7.100775	192.168.1.163	192.168.1.66	ICMP	192	Echo (ping) request id=0x937b, seq=1/256, ttl=63 (reply in 9)
7	7.101153	Private_66:68:00	Broadcast	ARP	68	Who has 192.168.1.65? Tell 192.168.1.66
8	7.110811	ca:01:13:dc:00:00	Private_66:68:00	ARP	64	192.168.1.65 is at ca:01:13:dc:00:00
9	7.112090	192.168.1.66	192.168.1.163	ICMP	192	Echo (ping) reply id=0x937b, seq=1/256, ttl=64 (request in 6)
10	7.121025	192.168.1.66	192.168.1.163	ICMP	192	Echo (ping) reply id=0x937b, seq=1/256, ttl=63
11	8.122160	192.168.1.163	192.168.1.66	ICMP	192	Echo (ping) request id=0x947b, seq=2/512, ttl=64 (no response found!)
12	8.130804	192.168.1.163	192.168.1.66	ICMP	192	Echo (ping) request id=0x947b, seq=2/512, ttl=63 (reply in 13)
13	8.131075	192.168.1.66	192.168.1.163	ICMP	192	Echo (ping) reply id=0x947b, seq=2/512, ttl=64 (request in 12)
14	8.140859	192.168.1.66	192.168.1.163	ICMP	192	Echo (ping) reply id=0x947b, seq=2/512, ttl=63
15	9.142252	192.168.1.163	192.168.1.66	ICMP	192	Echo (ping) request id=0x957b, seq=3/768, ttl=64 (no response found!)
16	9.151129	192.168.1.163	192.168.1.66	ICMP	192	Echo (ping) request id=0x957b, seq=3/768, ttl=63 (reply in 17)
17	9.151570	192.168.1.66	192.168.1.163	ICMP	192	Echo (ping) reply id=0x957b, seq=3/768, ttl=64 (request in 16)
18	9.161174	192.168.1.66	192.168.1.163	ICMP	192	Echo (ping) reply id=0x957b, seq=3/768, ttl=63
19	10.010770	ca:01:13:dc:00:00	ca:01:13:dc:00:00	LOOP	60	Reply

Frame 7: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface 0

Ethernet II, Src: Private\_66:68:00 (00:50:79:66:68:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 2

Address Resolution Protocol (request)

Hardware type: Ethernet (1)

Protocol type: IPv4 (0x0800)

Hardware size: 6

Protocol size: 4

Opcode: request (1)

Sender MAC address: Private\_66:68:00 (00:50:79:66:68:00)

Sender IP address: 192.168.1.66

Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)

Target IP address: 192.168.1.65

0000 ff ff ff ff ff ff 00 50 79 66 68 00 81 00 00 02 .....P yfh....

0010 08 06 00 01 08 00 06 04 00 01 00 50 79 66 68 00 .....Pyfh....

0020 c0 a8 01 42 ff ff ff ff ff ff c0 a8 01 41 00 00 ...B.....A...

0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....A.....

0040 00 00 00 00 .....B.....

Ready to load or capture Packets: 28 - Displayed: 28 (100.0%) Profile: Default

Fig.7 Packets captured when ping is made from PC6 to PC1

We can see in Fig.7 the packets captured while pinging PC6 to PC1. An ARP request is made to find is visible in the image.

Q3 (RIP) Configure the below topology to setup connectivity using RIPv2. R1, R2, and R3 will use dynamic routing protocol (RIPv2).

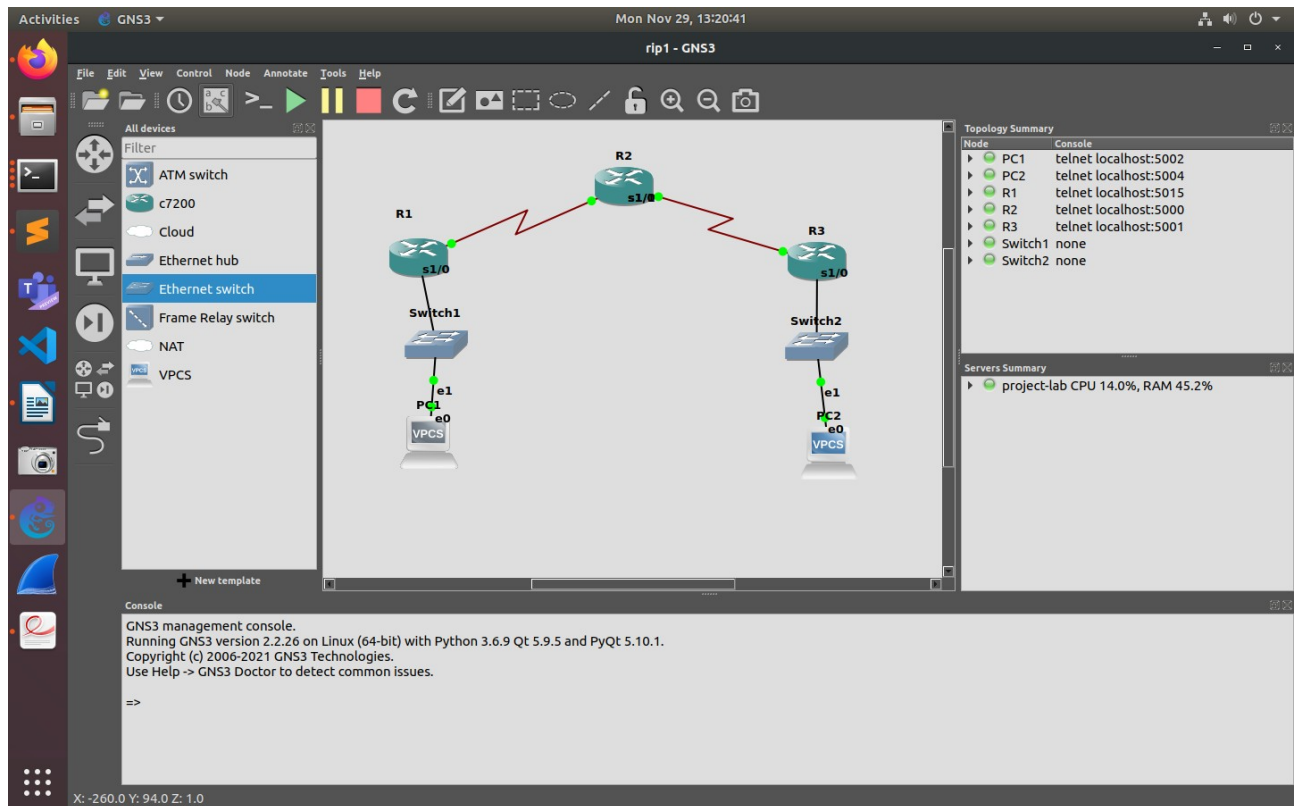
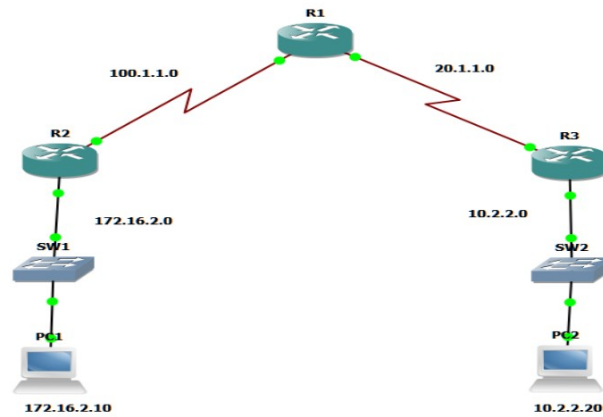


Fig.1 Network topology on GNS-3



```

R3#show 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

    20.0.0.0/24 is subnetted, 1 subnets
C       20.1.1.0 is directly connected, Serial1/0
    10.0.0.0/24 is subnetted, 1 subnets
C       10.2.2.0 is directly connected, FastEthernet0/0
R3#

```

Fig. 2 After configuring PCs we are able to see the nearest neighbours

Before the RIP protocol is set up, the neighbors it can view are only the next neighbors as seen in Fig. 2. In Fig. 3 we can see that after RIP protocol is setup it is able to see other networks via it's neighbors.

```

R3#show 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

R       100.0.0.0/8 [120/1] via 20.1.1.1, 00:00:22, Serial1/0
    20.0.0.0/24 is subnetted, 1 subnets
C       20.1.1.0 is directly connected, Serial1/0
R       172.16.0.0/16 [120/1] via 20.1.1.1, 00:00:06, Serial1/0
    10.0.0.0/24 is subnetted, 1 subnets
C       10.2.2.0 is directly connected, FastEthernet0/0
R3#

```

Fig. 3 Neighbors visible after RIP on all the three routers

```

PC2> ping 172.16.2.10

84 bytes from 172.16.2.10 icmp_seq=1 ttl=61 time=43.675 ms
84 bytes from 172.16.2.10 icmp_seq=2 ttl=61 time=40.214 ms
84 bytes from 172.16.2.10 icmp_seq=3 ttl=61 time=40.694 ms
84 bytes from 172.16.2.10 icmp_seq=4 ttl=61 time=40.393 ms
84 bytes from 172.16.2.10 icmp_seq=5 ttl=61 time=40.373 ms

PC2>

```

Fig. 4 Successful ping from PC2 to PC1 in separate networks

RIP uses distance vector algorithm. Once all routers have RIP setup we are able to see successful ping from one network to another as seen in Fig. 4.