Networking Assessment 5 - Module 7 & 8

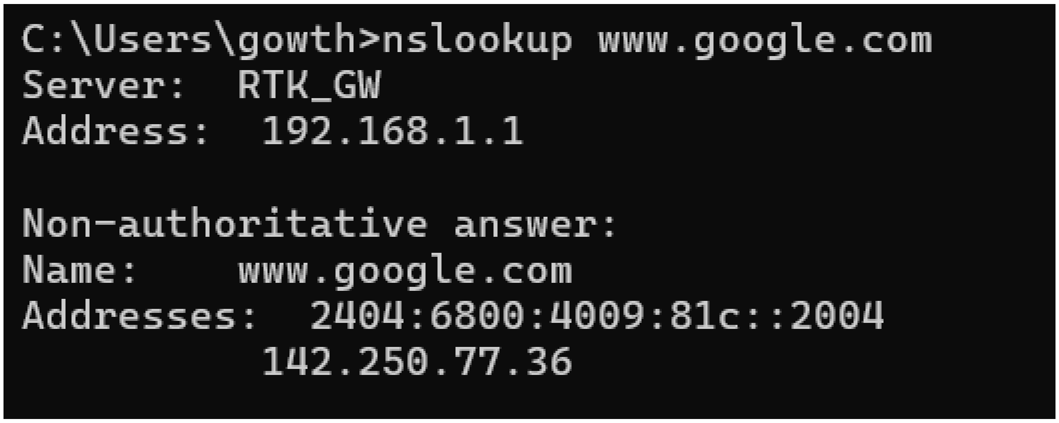
Name	T K Gowtham
Email ID	gowthamkamalasekar@gmail.com
College	VIT Chennai

Here are the questions from the image converted into text:

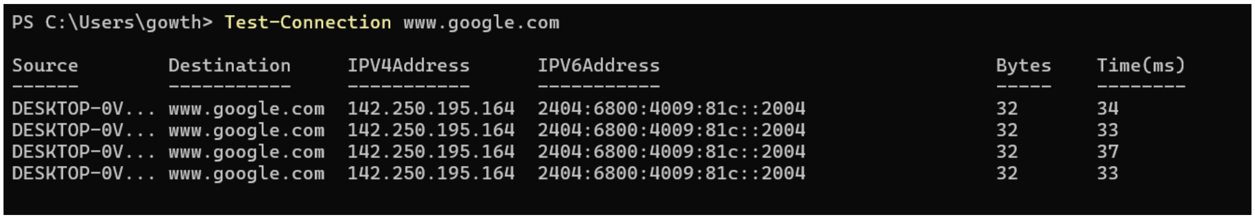
1. Try Test-Connection and nslookup commands for below websites:
    - www.google.com
    - www.facebook.com
    - www.amazon.com
    - www.github.com
    - www.cisco.com

Trying nslookup and test connection for each domain as follows :

Google.com :

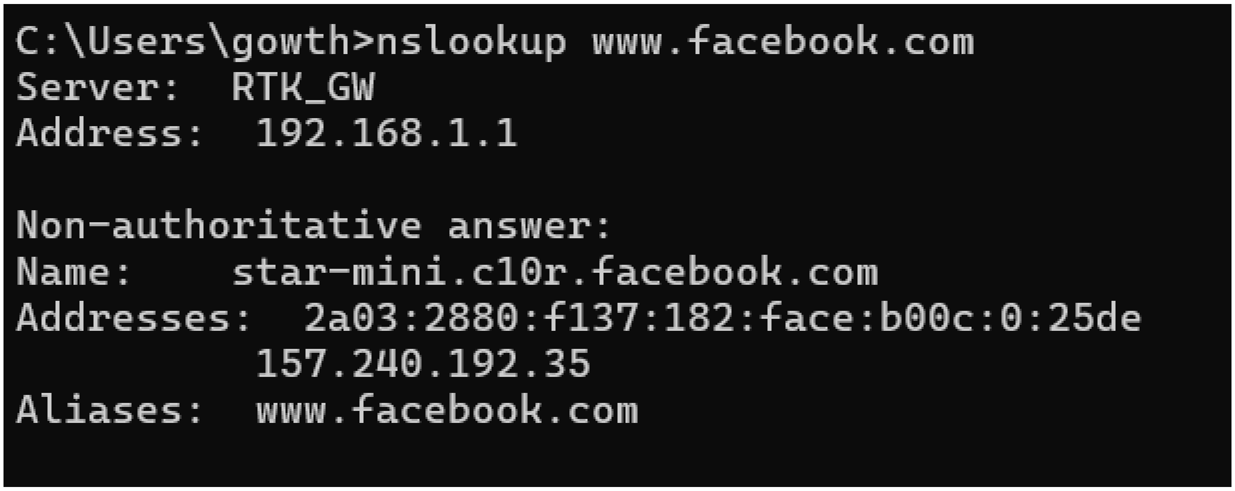


```
C:\Users\gowth>nslookup www.google.com
Server:  RTK_GW
Address:  192.168.1.1

Non-authoritative answer:
Name:    www.google.com
Addresses:  2404:6800:4009:81c::2004
          142.250.77.36
```

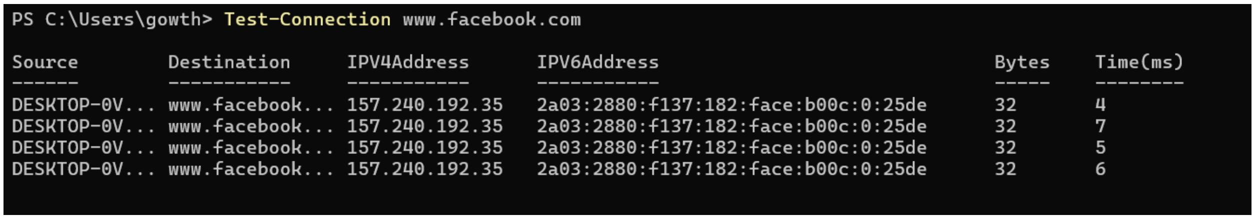


```
PS C:\Users\gowth> Test-Connection www.google.com

Source        Destination     IPV4Address      IPV6Address                                  Bytes    Time(ms)
------        -----------     -----------      -----------                                  -----    --------
DESKTOP-0V... www.google.com  142.250.195.164  2404:6800:4009:81c::2004                     32       34
DESKTOP-0V... www.google.com  142.250.195.164  2404:6800:4009:81c::2004                     32       33
DESKTOP-0V... www.google.com  142.250.195.164  2404:6800:4009:81c::2004                     32       37
DESKTOP-0V... www.google.com  142.250.195.164  2404:6800:4009:81c::2004                     32       33
```

www.facebook.com :

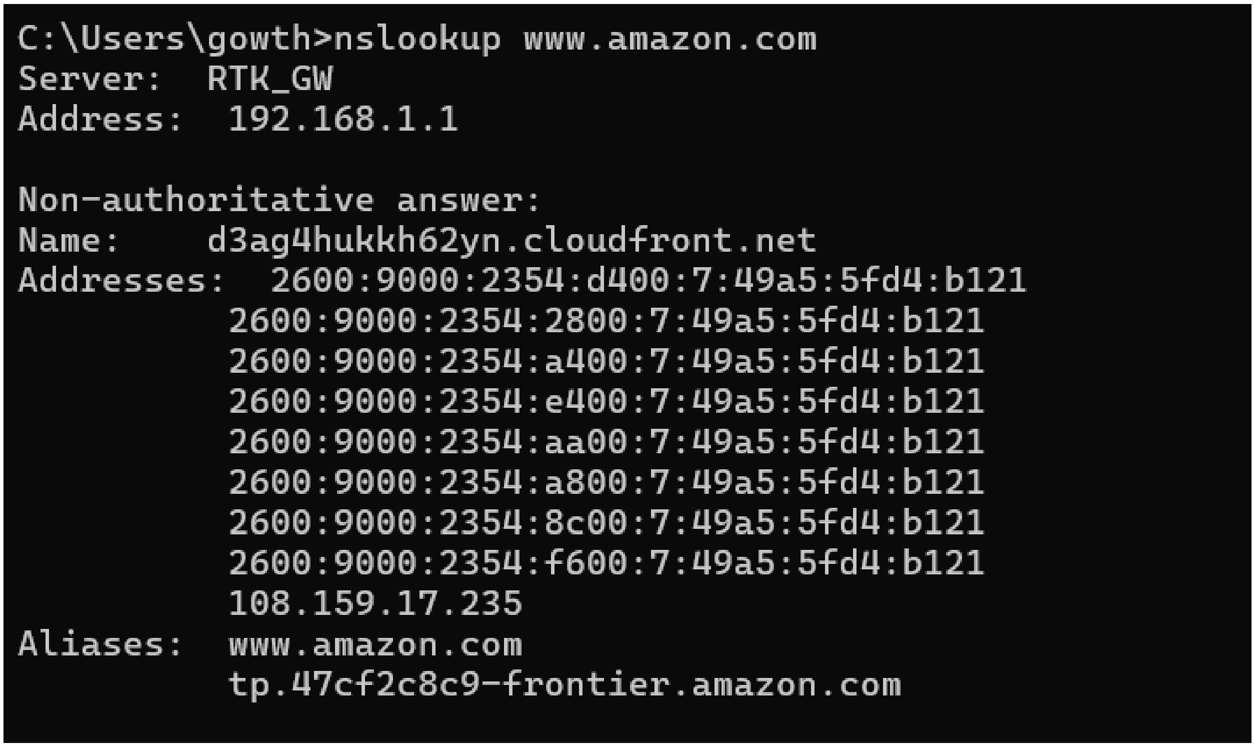


```
C:\Users\gowth>nslookup www.facebook.com
Server:  RTK_GW
Address:  192.168.1.1

Non-authoritative answer:
Name:    star-mini.c10r.facebook.com
Addresses:  2a03:2880:f137:182:face:b00c:0:25de
          157.240.192.35
Aliases:  www.facebook.com
```

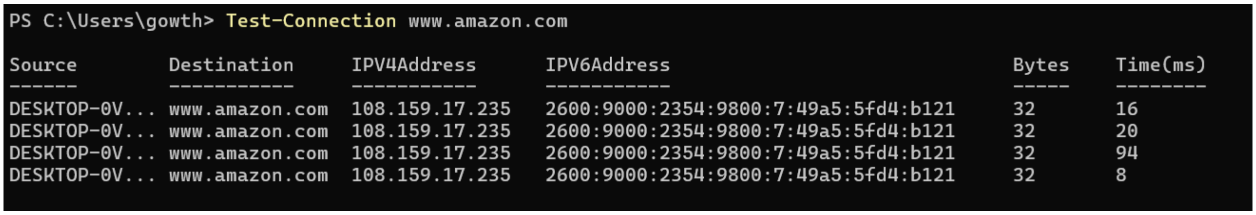


```
PS C:\Users\gowth> Test-Connection www.facebook.com

Source        Destination     IPV4Address     IPV6Address                                   Bytes    Time(ms)
------        -----------     -----------     -----------                                   -----    --------
DESKTOP-0V... www.facebook... 157.240.192.35  2a03:2880:f137:182:face:b00c:0:25de           32       4
DESKTOP-0V... www.facebook... 157.240.192.35  2a03:2880:f137:182:face:b00c:0:25de           32       7
DESKTOP-0V... www.facebook... 157.240.192.35  2a03:2880:f137:182:face:b00c:0:25de           32       5
DESKTOP-0V... www.facebook... 157.240.192.35  2a03:2880:f137:182:face:b00c:0:25de           32       6
```

www.amazon.com :

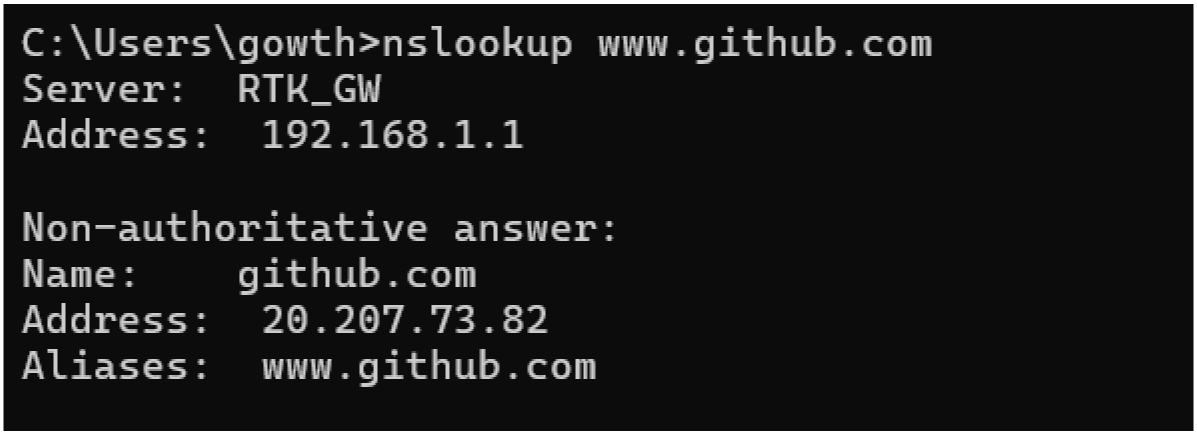


```
C:\Users\gowth>nslookup www.amazon.com
Server:  RTK_GW
Address:  192.168.1.1

Non-authoritative answer:
Name:    d3ag4hukkh62yn.cloudfront.net
Addresses:  2600:9000:2354:d400:7:49a5:5fd4:b121
          2600:9000:2354:2800:7:49a5:5fd4:b121
          2600:9000:2354:a400:7:49a5:5fd4:b121
          2600:9000:2354:e400:7:49a5:5fd4:b121
          2600:9000:2354:aa00:7:49a5:5fd4:b121
          2600:9000:2354:a800:7:49a5:5fd4:b121
          2600:9000:2354:8c00:7:49a5:5fd4:b121
          2600:9000:2354:f600:7:49a5:5fd4:b121
          108.159.17.235
Aliases:  www.amazon.com
          tp.47cf2c8c9-frontier.amazon.com
```

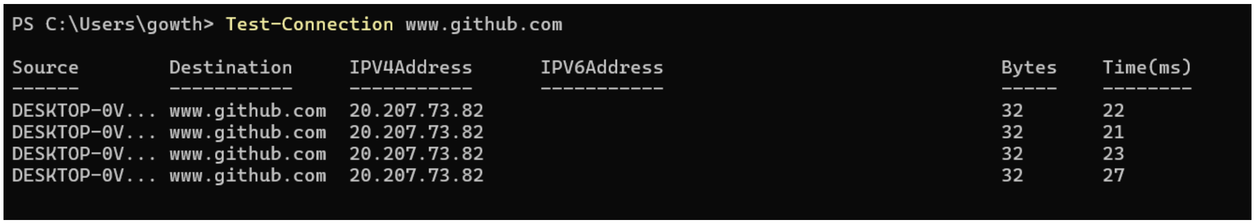


```
PS C:\Users\gowth> Test-Connection www.amazon.com

Source        Destination     IPV4Address      IPV6Address                              Bytes    Time(ms)
------        -----------     -----------      -----------                              -----    --------
DESKTOP-0V... www.amazon.com  108.159.17.235   2600:9000:2354:9800:7:49a5:5fd4:b121     32       16
DESKTOP-0V... www.amazon.com  108.159.17.235   2600:9000:2354:9800:7:49a5:5fd4:b121     32       20
DESKTOP-0V... www.amazon.com  108.159.17.235   2600:9000:2354:9800:7:49a5:5fd4:b121     32       94
DESKTOP-0V... www.amazon.com  108.159.17.235   2600:9000:2354:9800:7:49a5:5fd4:b121     32       8
```

www.github.com :

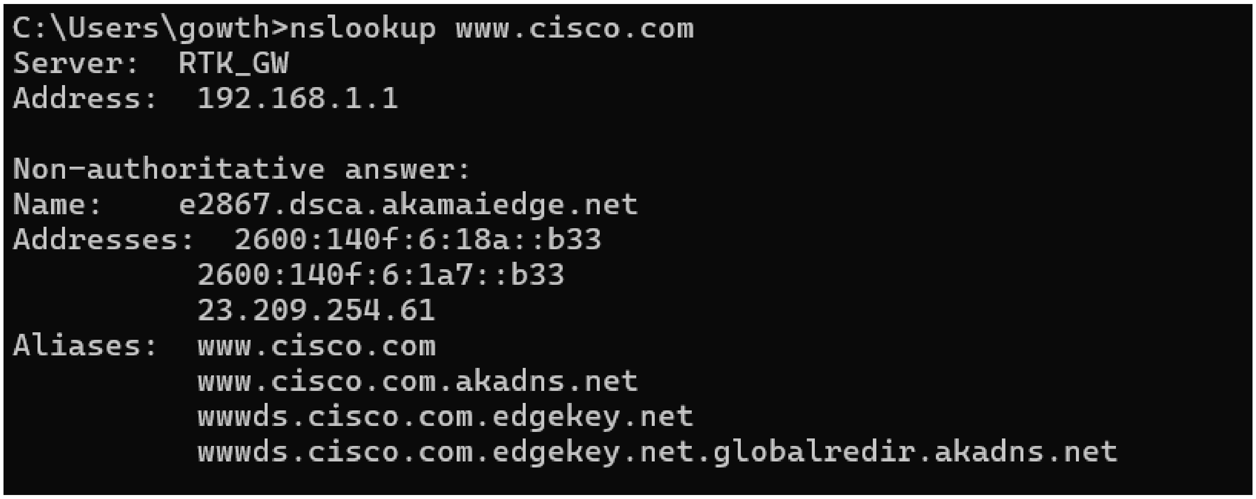


```
C:\Users\gowth>nslookup www.github.com
Server:  RTK_GW
Address:  192.168.1.1

Non-authoritative answer:
Name:    github.com
Address:  20.207.73.82
Aliases:  www.github.com
```

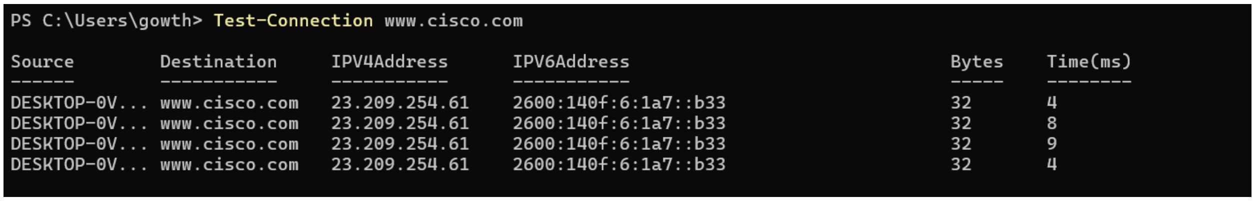


```
PS C:\Users\gowth> Test-Connection www.github.com

Source        Destination     IPV4Address      IPV6Address                                     Bytes    Time(ms)
------        -----------     -----------      -----------                                     -----    --------
DESKTOP-0V... www.github.com  20.207.73.82                                                     32       22
DESKTOP-0V... www.github.com  20.207.73.82                                                     32       21
DESKTOP-0V... www.github.com  20.207.73.82                                                     32       23
DESKTOP-0V... www.github.com  20.207.73.82                                                     32       27
```

www.cisco.com :



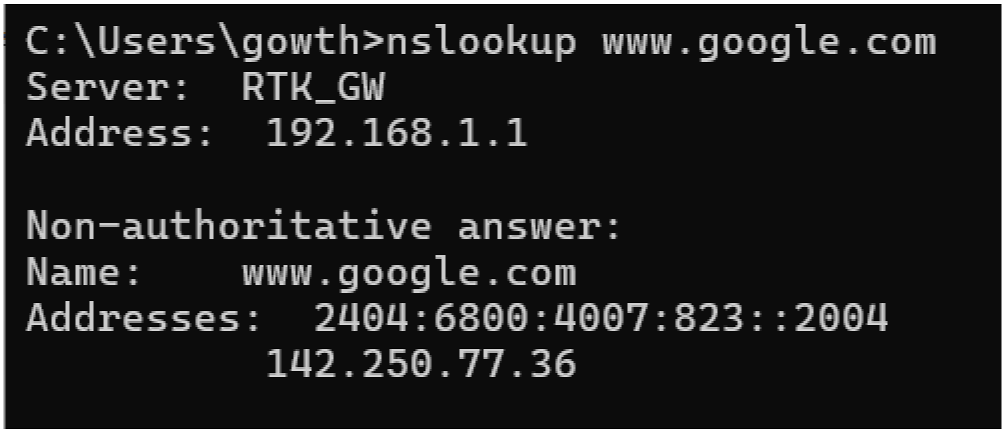
```
C:\Users\gowth>nslookup www.cisco.com
Server:  RTK_GW
Address:  192.168.1.1

Non-authoritative answer:
Name:    e2867.dsca.akamaiedge.net
Addresses:  2600:140f:6:18a::b33
          2600:140f:6:1a7::b33
          23.209.254.61
Aliases:  www.cisco.com
          www.cisco.com.akadns.net
          wwwds.cisco.com.edgekey.net
          wwwds.cisco.com.edgekey.net.globalredir.akadns.net
```



```
PS C:\Users\gowth> Test-Connection www.cisco.com

Source        Destination     IPV4Address      IPV6Address                                     Bytes    Time(ms)
------        -----------     -----------      -----------                                     -----    --------
DESKTOP-0V... www.cisco.com   23.209.254.61    2600:140f:6:1a7::b33                            32       4
DESKTOP-0V... www.cisco.com   23.209.254.61    2600:140f:6:1a7::b33                            32       8
DESKTOP-0V... www.cisco.com   23.209.254.61    2600:140f:6:1a7::b33                            32       9
DESKTOP-0V... www.cisco.com   23.209.254.61    2600:140f:6:1a7::b33                            32       4
```

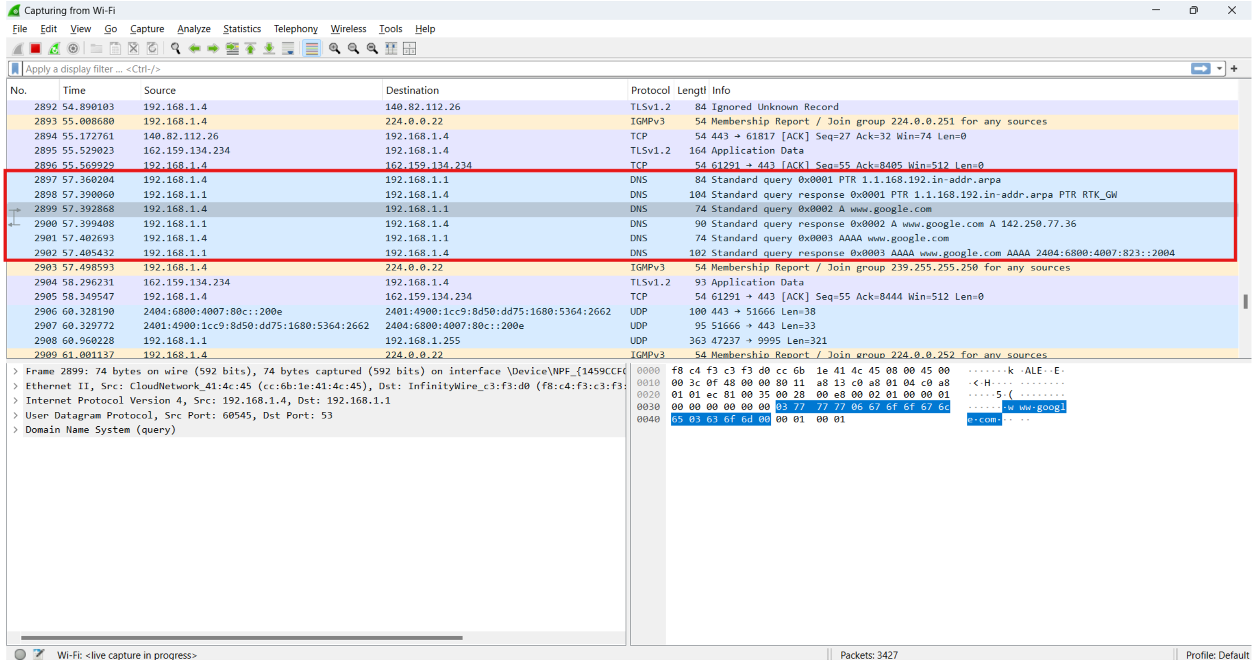
2. Use Wireshark to capture and analyze DNS, TCP, UDP traffic and packet header, packet flow, options and flags.

DNS :



```
C:\Users\gowth>nslookup www.google.com
Server:   RTK_GW
Address:  192.168.1.1

Non-authoritative answer:
Name:     www.google.com
Addresses:  2404:6800:4007:823::2004
          142.250.77.36
```

Capturing Packets in wireshark of DNS :

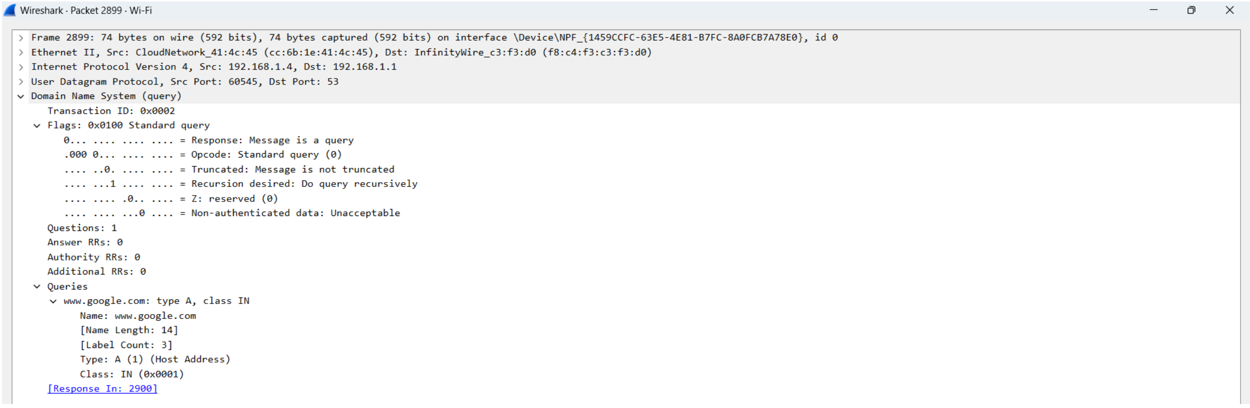


No.	Time	Source	Destination	Protocol	Length	Info
2892	54.890103	192.168.1.4	140.82.112.26	TLSv1.2	84	Ignored Unknown Record
2893	55.008680	192.168.1.4	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.251 for any sources
2894	55.172761	140.82.112.26	192.168.1.4	TCP	54	443 → 61817 [ACK] Seq=27 Ack=32 Win=74 Len=0
2895	55.529023	162.159.134.234	192.168.1.4	TLSv1.2	164	Application Data
2896	55.569929	192.168.1.4	162.159.134.234	TCP	54	61291 → 443 [ACK] Seq=55 Ack=8405 Win=512 Len=0
2897	57.360204	192.168.1.4	192.168.1.1	DNS	84	Standard query 0x0001 PTR 1.1.168.192.in-addr.arpa
2898	57.390060	192.168.1.1	192.168.1.4	DNS	104	Standard query response 0x0001 PTR 1.1.168.192.in-addr.arpa PTR RTK_GW
2899	57.392868	192.168.1.4	192.168.1.1	DNS	74	Standard query 0x0002 A www.google.com
2900	57.399408	192.168.1.1	192.168.1.4	DNS	90	Standard query response 0x0002 A www.google.com A 142.250.77.36
2901	57.402693	192.168.1.4	192.168.1.1	DNS	74	Standard query 0x0003 AAAA www.google.com
2902	57.405432	192.168.1.1	192.168.1.4	DNS	102	Standard query response 0x0003 AAAA www.google.com AAAA 2404:6800:4007:823::2004
2903	57.498593	192.168.1.4	224.0.0.22	IGMPv3	54	Membership Report / Join group 239.255.255.250 for any sources
2904	58.296231	162.159.134.234	192.168.1.4	TLSv1.2	93	Application Data
2905	58.349547	192.168.1.4	162.159.134.234	TCP	54	61291 → 443 [ACK] Seq=55 Ack=8444 Win=512 Len=0
2906	60.328190	2404:6800:4007:80c::200e	2401:4900:1cc9:8d50:dd75:1680:5364:2662	UDP	100	443 → 51666 Len=38
2907	60.329772	2401:4900:1cc9:8d50:dd75:1680:5364:2662	2404:6800:4007:80c::200e	UDP	95	51666 → 443 Len=33
2908	60.960228	192.168.1.1	192.168.1.255	UDP	363	47237 → 9995 Len=321
2909	61.001137	192.168.1.4	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.252 for any sources

```
> Frame 2899: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{1459CCF(
> Ethernet II, Src: CloudNetwork_41:4c:45 (cc:6b:1e:41:4c:45), Dst: InfinityWire_c3:f3:d0 (f8:c4:f3:c3:f3:
> Internet Protocol Version 4, Src: 192.168.1.4, Dst: 192.168.1.1
> User Datagram Protocol, Src Port: 60545, Dst Port: 53
> Domain Name System (query)
```

```
0000  f8 c4 f3 c3 f3 d0 cc 6b  1e 41 4c 45 08 00 45 00   ·······k ·ALE··E·
0010  00 3c 0f 48 00 00 80 11  a8 13 c0 a8 01 04 c0 a8   ·<·H···· ········
0020  01 01 ec 81 00 35 00 28  00 e8 00 02 01 00 00 01   ·····5·( ········
0030  00 00 00 00 00 00 03 77  77 77 06 67 6f 6f 67 6c   ·······w ww·googl
0040  65 03 63 6f 6d 00 00 01  00 01                     e·com··· ··
```

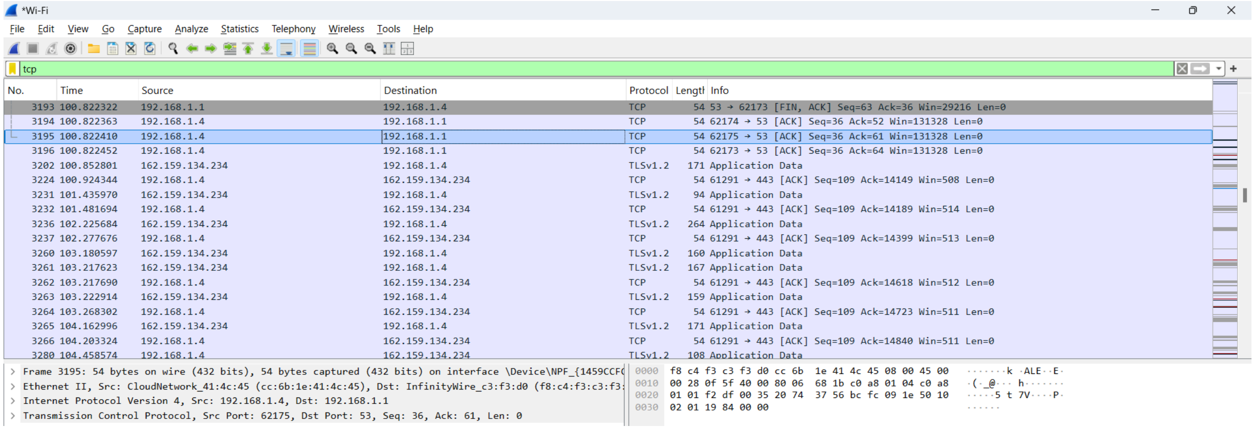
The DNS contains the information of the website like domain ID, type, class and flags.



Wireshark · Packet 2899 · Wi-Fi

```
> Frame 2899: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{1459CCFC-63E5-4E81-B7FC-8A0FCB7A78E0}, id 0
> Ethernet II, Src: CloudNetwork_41:4c:45 (cc:6b:1e:41:4c:45), Dst: InfinityWire_c3:f3:d0 (f8:c4:f3:c3:f3:d0)
> Internet Protocol Version 4, Src: 192.168.1.4, Dst: 192.168.1.1
> User Datagram Protocol, Src Port: 60545, Dst Port: 53
v Domain Name System (query)
    Transaction ID: 0x0002
  v Flags: 0x0100 Standard query
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  v Queries
    v www.google.com: type A, class IN
        Name: www.google.com
        [Name Length: 14]
        [Label Count: 3]
        Type: A (1) (Host Address)
        Class: IN (0x0001)
    [Response In: 2900]
```

TCP :



*Wi-Fi

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

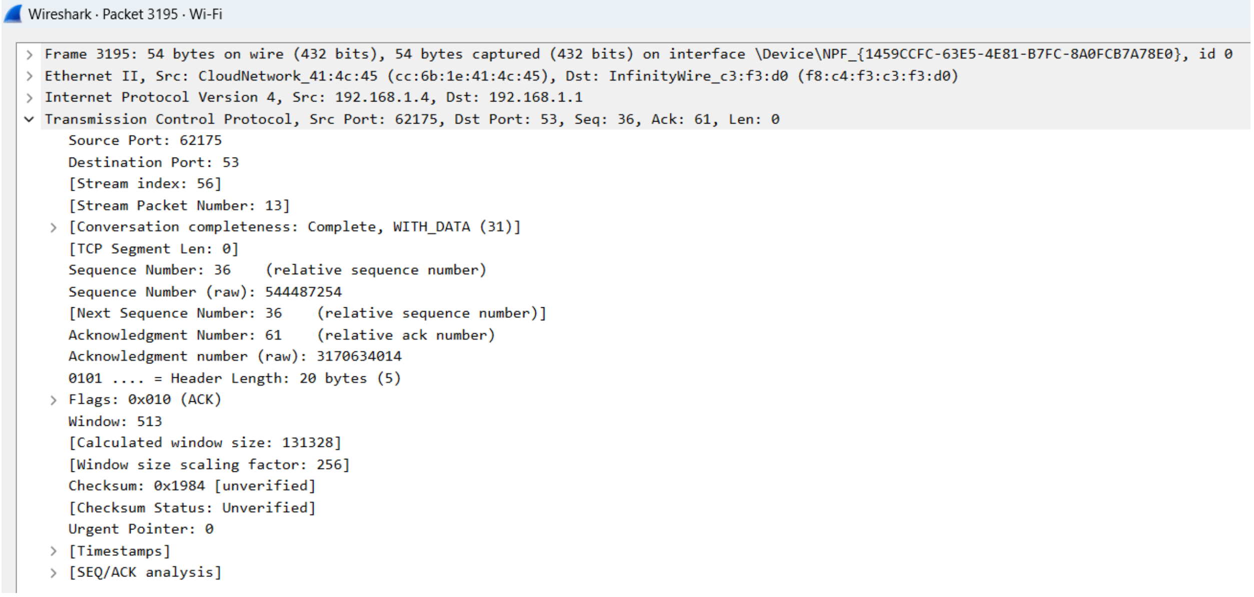
tcp

```
No.   Time        Source           Destination      Protocol Lengtl Info
 3193 100.822322  192.168.1.1      192.168.1.4      TCP        54 53 → 62173 [FIN, ACK] Seq=63 Ack=36 Win=29216 Len=0
 3194 100.822363  192.168.1.4      192.168.1.1      TCP        54 62174 → 53 [ACK] Seq=36 Ack=52 Win=131328 Len=0
 3195 100.822410  192.168.1.4      192.168.1.1      TCP        54 62175 → 53 [ACK] Seq=36 Ack=61 Win=131328 Len=0
 3196 100.822452  192.168.1.4      192.168.1.1      TCP        54 62173 → 53 [ACK] Seq=36 Ack=64 Win=131328 Len=0
 3202 100.852801  162.159.134.234  192.168.1.4      TLSv1.2   171 Application Data
 3224 100.924344  192.168.1.4      162.159.134.234  TCP        54 61291 → 443 [ACK] Seq=109 Ack=14149 Win=508 Len=0
 3231 101.435970  162.159.134.234  192.168.1.4      TLSv1.2    94 Application Data
 3232 101.481694  192.168.1.4      162.159.134.234  TCP        54 61291 → 443 [ACK] Seq=109 Ack=14189 Win=514 Len=0
 3236 102.225684  162.159.134.234  192.168.1.4      TLSv1.2   264 Application Data
 3237 102.277676  192.168.1.4      162.159.134.234  TCP        54 61291 → 443 [ACK] Seq=109 Ack=14399 Win=513 Len=0
 3260 103.180597  162.159.134.234  192.168.1.4      TLSv1.2   160 Application Data
 3261 103.217623  162.159.134.234  192.168.1.4      TLSv1.2   167 Application Data
 3262 103.217690  192.168.1.4      162.159.134.234  TCP        54 61291 → 443 [ACK] Seq=109 Ack=14618 Win=512 Len=0
 3263 103.222914  162.159.134.234  192.168.1.4      TLSv1.2   159 Application Data
 3264 103.268302  192.168.1.4      162.159.134.234  TCP        54 61291 → 443 [ACK] Seq=109 Ack=14723 Win=511 Len=0
 3265 104.162996  162.159.134.234  192.168.1.4      TLSv1.2   171 Application Data
 3266 104.203324  192.168.1.4      162.159.134.234  TCP        54 61291 → 443 [ACK] Seq=109 Ack=14840 Win=511 Len=0
 3280 104.458574  192.168.1.4      162.159.134.234  TLSv1.2   108 Application Data
```

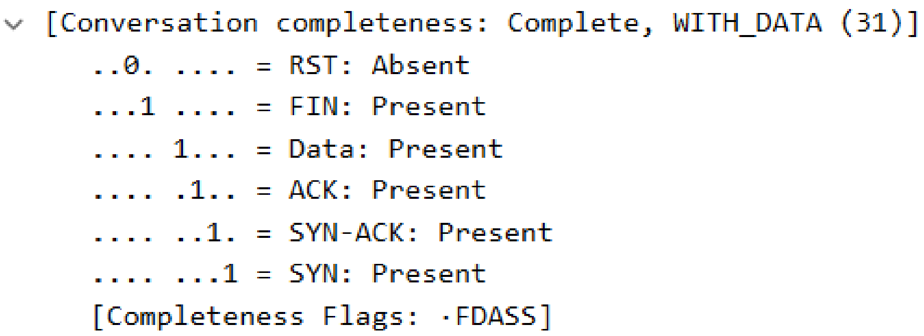
```
> Frame 3195: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{1459CCF(
> Ethernet II, Src: CloudNetwork_41:4c:45 (cc:6b:1e:41:4c:45), Dst: InfinityWire_c3:f3:d0 (f8:c4:f3:c3:f3:
> Internet Protocol Version 4, Src: 192.168.1.4, Dst: 192.168.1.1
> Transmission Control Protocol, Src Port: 62175, Dst Port: 53, Seq: 36, Ack: 61, Len: 0
```

```
0000  f8 c4 f3 c3 f3 d0 cc 6b  1e 41 4c 45 08 00 45 00   ·······k ·ALE··E·
0010  00 28 0f 5f 40 00 80 06  68 1b c0 a8 01 04 c0 a8   ·(·_@··· h·······
0020  01 01 f2 df 00 35 20 74  37 56 bc fc 09 1e 50 10   ·····5 t 7V····P·
0030  02 01 19 84 00 00                                  ······
```

TCP contains a lot of information like source and destination port, sequence number, flags, window size, checksum. Since this reliable connection, it uses flags with ACK in it.



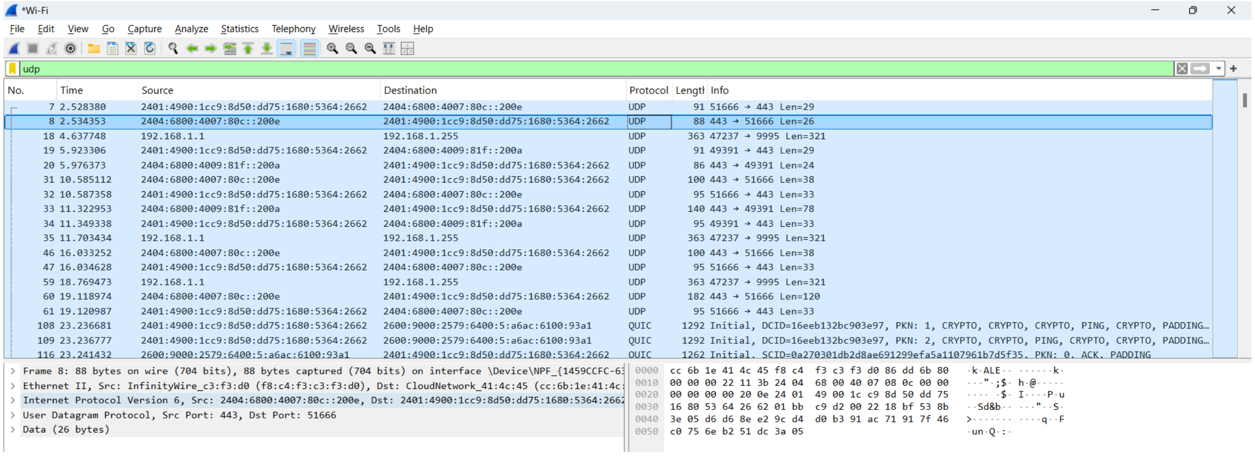
```
> Frame 3195: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{1459CCFC-63E5-4E81-B7FC-8A0FCB7A78E0}, id 0
> Ethernet II, Src: CloudNetwork_41:4c:45 (cc:6b:1e:41:4c:45), Dst: InfinityWire_c3:f3:d0 (f8:c4:f3:c3:f3:d0)
> Internet Protocol Version 4, Src: 192.168.1.4, Dst: 192.168.1.1
v Transmission Control Protocol, Src Port: 62175, Dst Port: 53, Seq: 36, Ack: 61, Len: 0
     Source Port: 62175
     Destination Port: 53
     [Stream index: 56]
     [Stream Packet Number: 13]
   > [Conversation completeness: Complete, WITH_DATA (31)]
     [TCP Segment Len: 0]
     Sequence Number: 36    (relative sequence number)
     Sequence Number (raw): 544487254
     [Next Sequence Number: 36    (relative sequence number)]
     Acknowledgment Number: 61    (relative ack number)
     Acknowledgment number (raw): 3170634014
     0101 .... = Header Length: 20 bytes (5)
   > Flags: 0x010 (ACK)
     Window: 513
     [Calculated window size: 131328]
     [Window size scaling factor: 256]
     Checksum: 0x1984 [unverified]
     [Checksum Status: Unverified]
     Urgent Pointer: 0
   > [Timestamps]
   > [SEQ/ACK analysis]
```



```
v [Conversation completeness: Complete, WITH_DATA (31)]
     ..0. .... = RST: Absent
     ...1 .... = FIN: Present
     .... 1... = Data: Present
     .... .1.. = ACK: Present
     .... ..1. = SYN-ACK: Present
     .... ...1 = SYN: Present
     [Completeness Flags: ·FDASS]
```

```
∨ Flags: 0x010 (ACK)
     000. .... .... = Reserved: Not set
     ...0 .... .... = Accurate ECN: Not set
     .... 0... .... = Congestion Window Reduced: Not set
     .... .0.. .... = ECN-Echo: Not set
     .... ..0. .... = Urgent: Not set
     .... ...1 .... = Acknowledgment: Set
     .... .... 0... = Push: Not set
     .... .... .0.. = Reset: Not set
     .... .... ..0. = Syn: Not set
     .... .... ...0 = Fin: Not set
     [TCP Flags: ·······A····]


∨ [Timestamps]
    [Time since first frame in this TCP stream: 0.079134000 seconds]
    [Time since previous frame in this TCP stream: 0.000088000 seconds]
∨ [SEQ/ACK analysis]
    [This is an ACK to the segment in frame: 3192]
    [The RTT to ACK the segment was: 0.000088000 seconds]
    [iRTT: 0.002776000 seconds]
```

UDP :

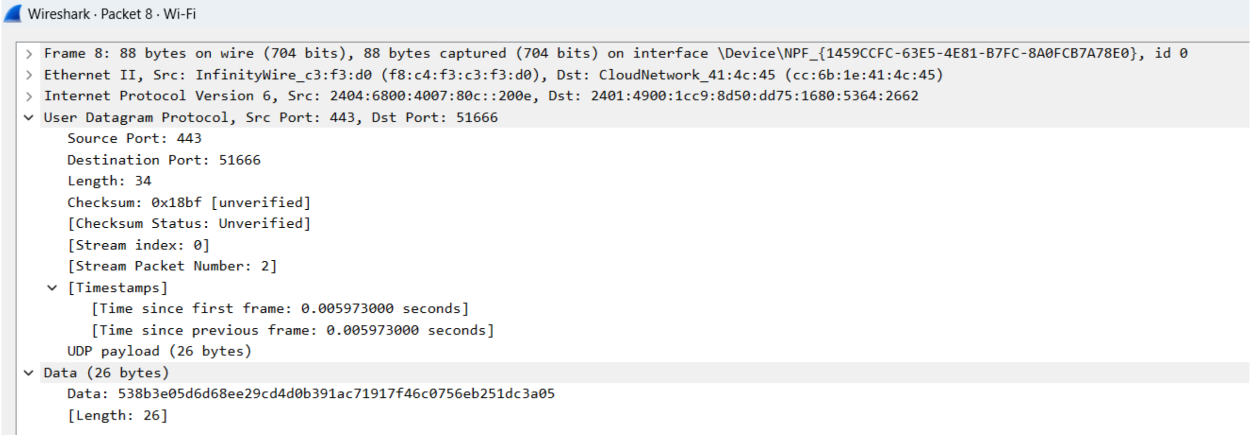

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
udp

```
No.   Time       Source                                  Destination                             Protocol Lengtl Info
    7 2.528380   2401:4900:1cc9:8d50:dd75:1680:5364:2662 2404:6800:4007:80c::200e                UDP        91 51666 → 443 Len=29
    8 2.534353   2404:6800:4007:80c::200e                2401:4900:1cc9:8d50:dd75:1680:5364:2662 UDP        88 443 → 51666 Len=26
   18 4.637748   192.168.1.1                             192.168.1.255                           UDP       363 47237 → 9995 Len=321
   19 5.923306   2401:4900:1cc9:8d50:dd75:1680:5364:2662 2404:6800:4009:81f::200a                UDP        91 49391 → 443 Len=29
   20 5.976373   2404:6800:4009:81f::200a                2401:4900:1cc9:8d50:dd75:1680:5364:2662 UDP        86 443 → 49391 Len=24
   31 10.585112  2404:6800:4007:80c::200e                2401:4900:1cc9:8d50:dd75:1680:5364:2662 UDP       100 443 → 51666 Len=38
   32 10.587358  2401:4900:1cc9:8d50:dd75:1680:5364:2662 2404:6800:4007:80c::200e                UDP        95 51666 → 443 Len=33
   33 11.322953  2404:6800:4009:81f::200a                2401:4900:1cc9:8d50:dd75:1680:5364:2662 UDP       140 443 → 49391 Len=78
   34 11.349338  2401:4900:1cc9:8d50:dd75:1680:5364:2662 2404:6800:4009:81f::200a                UDP        95 49391 → 443 Len=33
   35 11.703434  192.168.1.1                             192.168.1.255                           UDP       363 47237 → 9995 Len=321
   46 16.033252  2404:6800:4007:80c::200e                2401:4900:1cc9:8d50:dd75:1680:5364:2662 UDP       100 443 → 51666 Len=38
   47 16.034628  2401:4900:1cc9:8d50:dd75:1680:5364:2662 2404:6800:4007:80c::200e                UDP        95 51666 → 443 Len=33
   59 18.769473  192.168.1.1                             192.168.1.255                           UDP       363 47237 → 9995 Len=321
   60 19.118974  2404:6800:4007:80c::200e                2401:4900:1cc9:8d50:dd75:1680:5364:2662 UDP       182 443 → 51666 Len=120
   61 19.120987  2401:4900:1cc9:8d50:dd75:1680:5364:2662 2404:6800:4007:80c::200e                UDP        95 51666 → 443 Len=33
  108 23.236681  2401:4900:1cc9:8d50:dd75:1680:5364:2662 2600:9000:2579:6400:5:a6ac:6100:93a1    QUIC     1292 Initial, DCID=16eeb132bc903e97, PKN: 1, CRYPTO, CRYPTO, CRYPTO, PING, CRYPTO, PADDING…
  109 23.236777  2401:4900:1cc9:8d50:dd75:1680:5364:2662 2600:9000:2579:6400:5:a6ac:6100:93a1    QUIC     1292 Initial, DCID=16eeb132bc903e97, PKN: 2, CRYPTO, CRYPTO, PING, CRYPTO, CRYPTO, PADDING…
  116 23.241432  2600:9000:2579:6400:5:a6ac:6100:93a1    2401:4900:1cc9:8d50:dd75:1680:5364:2662 QUIC     1262 Initial, SCID=0a270301db2d8ae691299efa5a1107961b7d5f35, PKN: 0, ACK, PADDING
```

```
> Frame 8: 88 bytes on wire (704 bits), 88 bytes captured (704 bits) on interface \Device\NPF_{1459CCFC-63
> Ethernet II, Src: InfinityWire_c3:f3:d0 (f8:c4:f3:c3:f3:d0), Dst: CloudNetwork_41:4c:45 (cc:6b:1e:41:4c:
> Internet Protocol Version 6, Src: 2404:6800:4007:80c::200e, Dst: 2401:4900:1cc9:8d50:dd75:1680:5364:2662
> User Datagram Protocol, Src Port: 443, Dst Port: 51666
> Data (26 bytes)
```

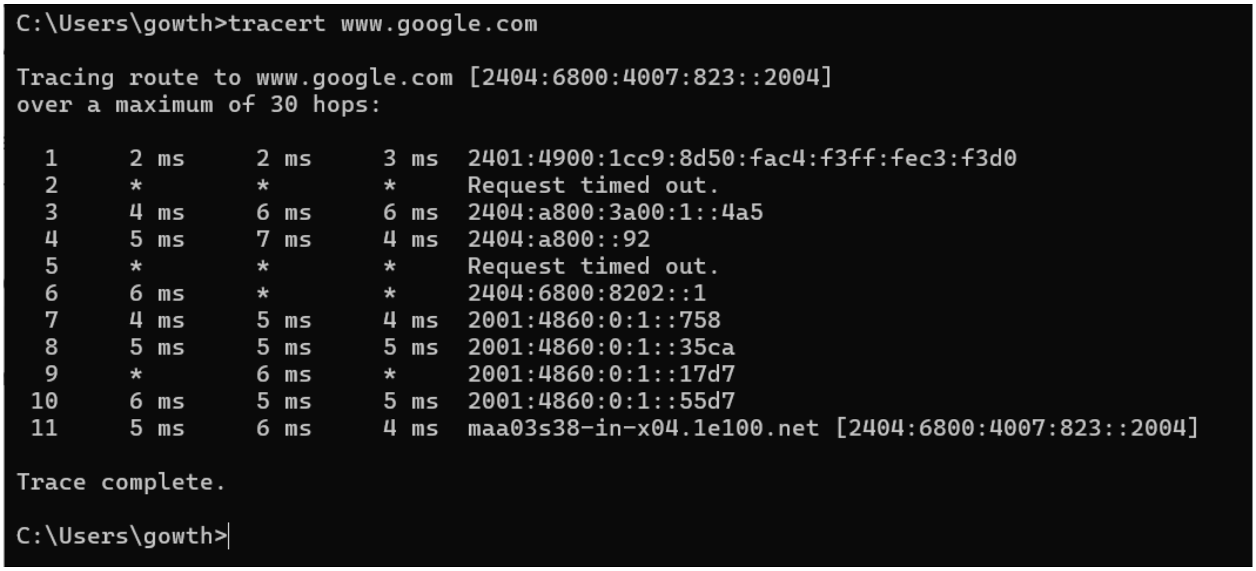
```
0000  cc 6b 1e 41 4c 45 f8 c4  f3 c3 f3 d0 86 dd 6b 80   ·k·ALE·· ······k·
0010  00 00 00 22 11 3b 24 04  68 00 40 07 08 0c 00 00   ···"·;$· h·@·····
0020  00 00 00 00 20 0e 24 01  49 00 1c c9 8d 50 dd 75   ···· ·$· I····P·u
0030  16 80 53 64 26 62 01 bb  c9 d2 00 22 18 bf 53 8b   ··Sd&b·· ···"··S·
0040  3e 05 d6 d6 8e e2 9c d4  d0 b3 91 ac 71 91 7f 46   >······· ····q··F
0050  c0 75 6e b2 51 dc 3a 05                            ·un·Q·:·
```

Unlike TCP, UDP is unreliable but fast so I doesn’t contain ACK or Flags, just source and destination ports and comparatively smaller than TCP.



```
> Frame 8: 88 bytes on wire (704 bits), 88 bytes captured (704 bits) on interface \Device\NPF_{1459CCFC-63E5-4E81-B7FC-8A0FCB7A78E0}, id 0
> Ethernet II, Src: InfinityWire_c3:f3:d0 (f8:c4:f3:c3:f3:d0), Dst: CloudNetwork_41:4c:45 (cc:6b:1e:41:4c:45)
> Internet Protocol Version 6, Src: 2404:6800:4007:80c::200e, Dst: 2401:4900:1cc9:8d50:dd75:1680:5364:2662
v User Datagram Protocol, Src Port: 443, Dst Port: 51666
    Source Port: 443
    Destination Port: 51666
    Length: 34
    Checksum: 0x18bf [unverified]
    [Checksum Status: Unverified]
    [Stream index: 0]
    [Stream Packet Number: 2]
  v [Timestamps]
      [Time since first frame: 0.005973000 seconds]
      [Time since previous frame: 0.005973000 seconds]
    UDP payload (26 bytes)
v Data (26 bytes)
    Data: 538b3e05d6d68ee29cd4d0b391ac71917f46c0756eb251dc3a05
    [Length: 26]
```

3. Explore traceroute/tracert for different websites eg: google.com and analyse the parameters in the output and explore different options for traceroute command.



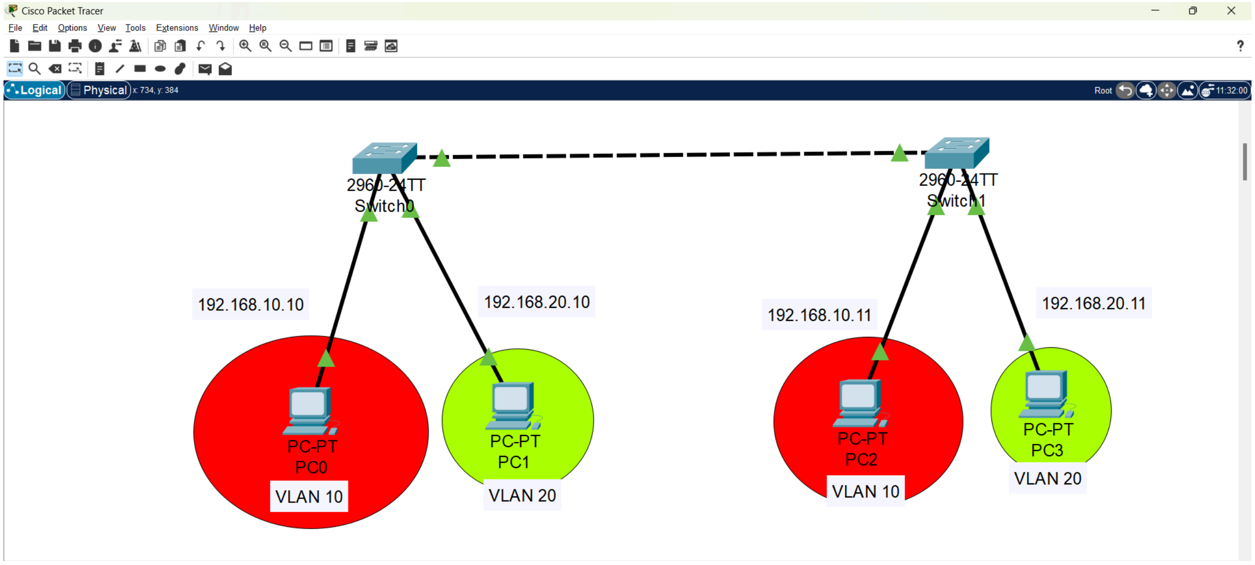
```
C:\Users\gowth>tracert www.google.com

Tracing route to www.google.com [2404:6800:4007:823::2004]
over a maximum of 30 hops:

  1     2 ms     2 ms     3 ms  2401:4900:1cc9:8d50:fac4:f3ff:fec3:f3d0
  2     *        *        *     Request timed out.
  3     4 ms     6 ms     6 ms  2404:a800:3a00:1::4a5
  4     5 ms     7 ms     4 ms  2404:a800::92
  5     *        *        *     Request timed out.
  6     6 ms     *        *     2404:6800:8202::1
  7     4 ms     5 ms     4 ms  2001:4860:0:1::758
  8     5 ms     5 ms     5 ms  2001:4860:0:1::35ca
  9     *        6 ms     *     2001:4860:0:1::17d7
 10     6 ms     5 ms     5 ms  2001:4860:0:1::55d7
 11     5 ms     6 ms     4 ms  maa03s38-in-x04.1e100.net [2404:6800:4007:823::2004]

Trace complete.

C:\Users\gowth>
```

The tracert command gives output of the every router the packets pass through. It attempts three times to go to a router and it shows the time it takes. It also shows the IP address. So to reach google server it takes 11 routers to reach there.

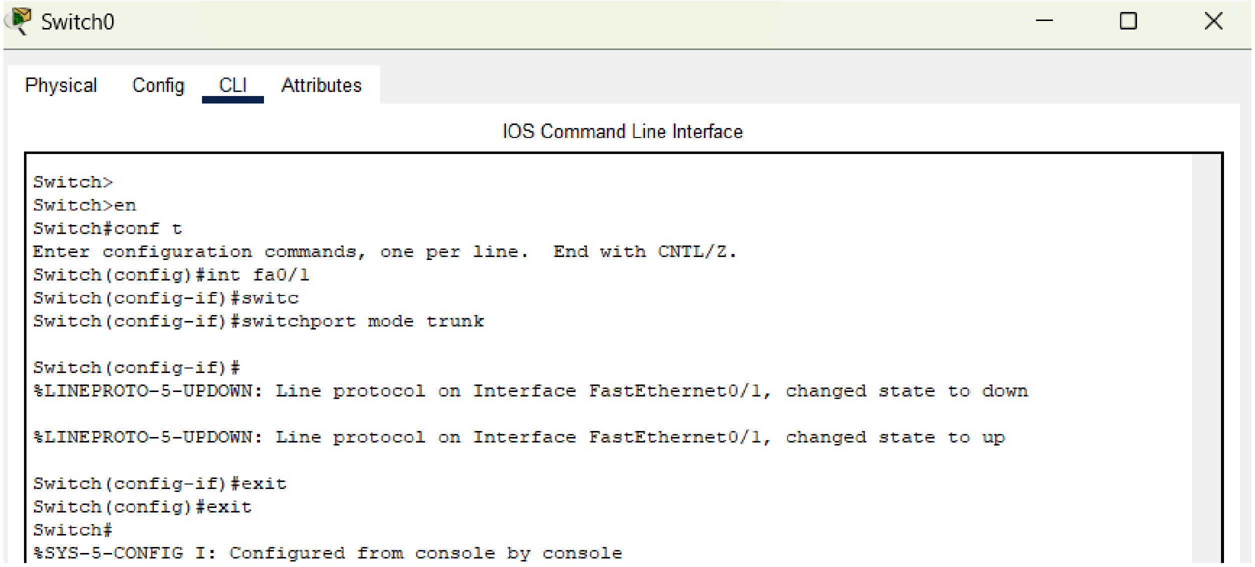
4. Use Cisco packet tracer for the below:

5. Set up trunk ports between switches and try ping between different VLANs.

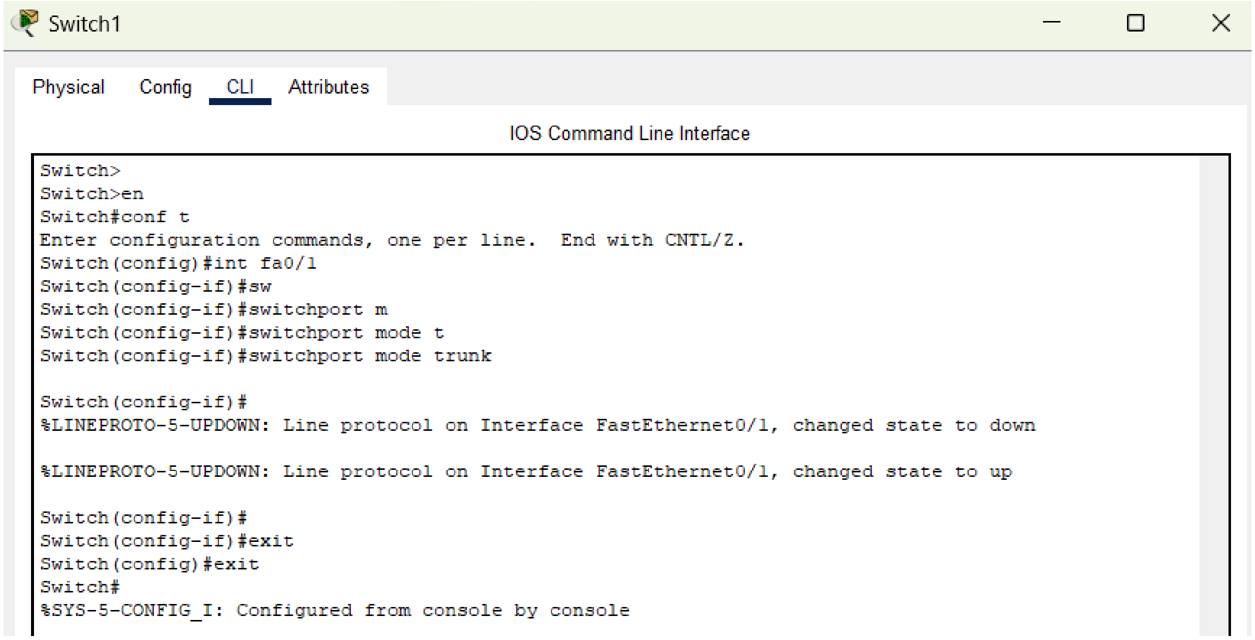
Configure this network in Packet Tracer :



In Switches, configure the interface that connecting switches as the truck port.

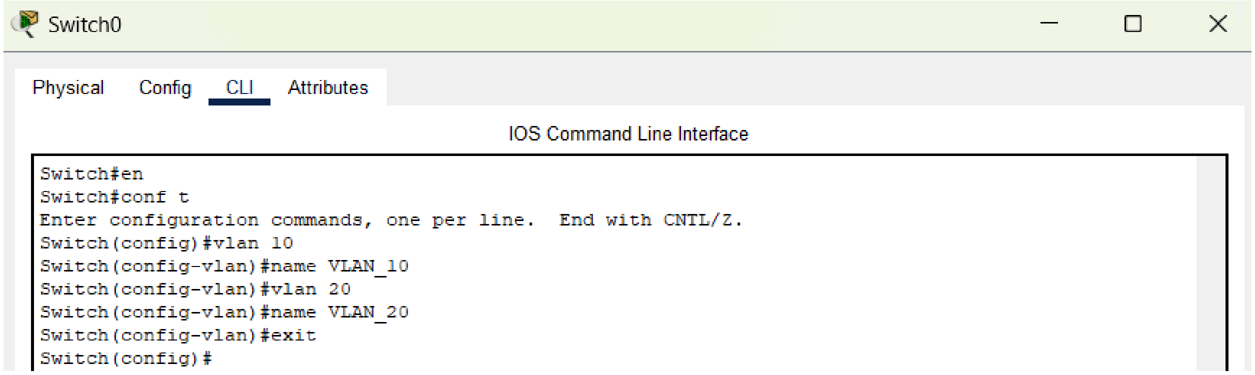


```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#switc
Switch(config-if)#switchport mode trunk

Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG I: Configured from console by console
```

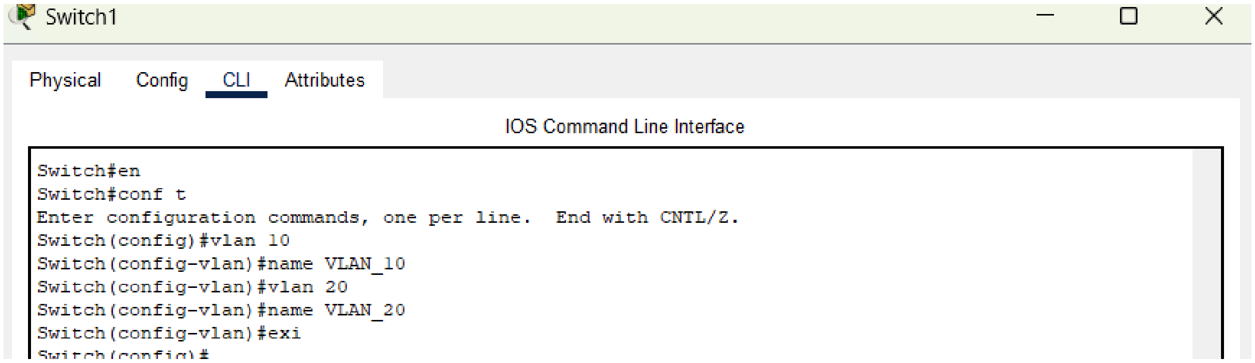


```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#sw
Switch(config-if)#switchport m
Switch(config-if)#switchport mode t
Switch(config-if)#switchport mode trunk

Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

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

Then in each switch create the respective vlans and assign them their names :

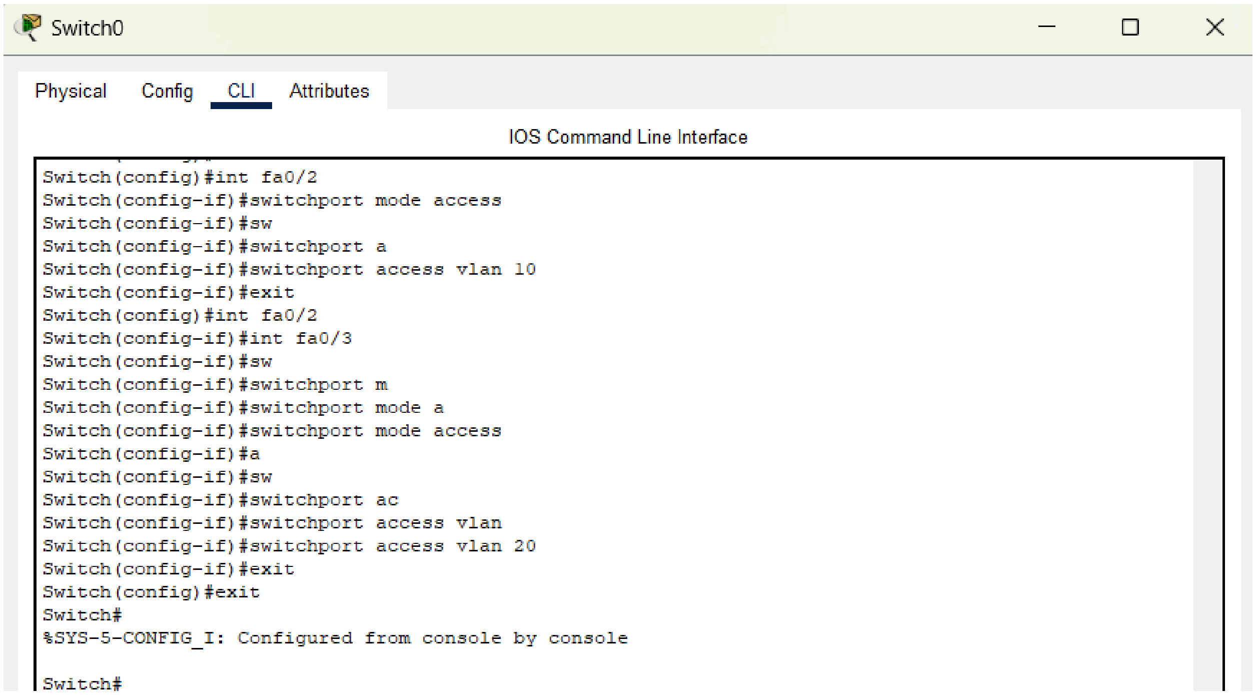


```
Switch#en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name VLAN_10
Switch(config-vlan)#vlan 20
Switch(config-vlan)#name VLAN_20
Switch(config-vlan)#exit
Switch(config)#
```

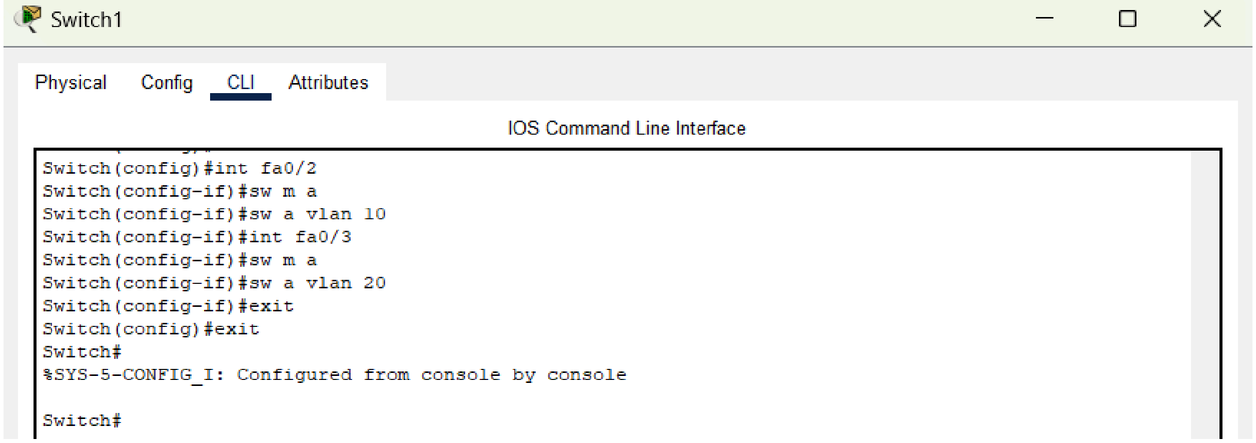


```
Switch#en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name VLAN_10
Switch(config-vlan)#vlan 20
Switch(config-vlan)#name VLAN_20
Switch(config-vlan)#exi
Switch(config)#
```

Now for each switch, configure the respective interfaces with which vlan it belongs to with access port as given below :

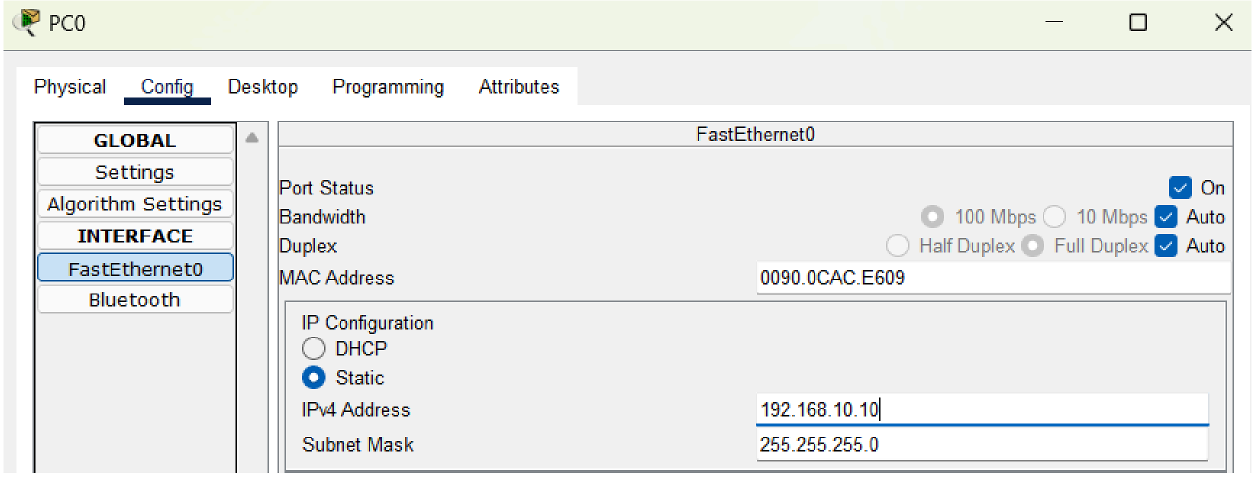


```
Switch(config)#int fa0/2
Switch(config-if)#switchport mode access
Switch(config-if)#sw
Switch(config-if)#switchport a
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
Switch(config)#int fa0/2
Switch(config-if)#int fa0/3
Switch(config-if)#sw
Switch(config-if)#switchport m
Switch(config-if)#switchport mode a
Switch(config-if)#switchport mode access
Switch(config-if)#a
Switch(config-if)#sw
Switch(config-if)#switchport ac
Switch(config-if)#switchport access vlan
Switch(config-if)#switchport access vlan 20
Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#
```



```
Switch(config)#int fa0/2
Switch(config-if)#sw m a
Switch(config-if)#sw a vlan 10
Switch(config-if)#int fa0/3
Switch(config-if)#sw m a
Switch(config-if)#sw a vlan 20
Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#
```

Configure IP address as follows:



FastEthernet0

Port Status ☑ On

Bandwidth ○ 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ○ Full Duplex ☑ Auto

MAC Address 0090.0CAC.E609

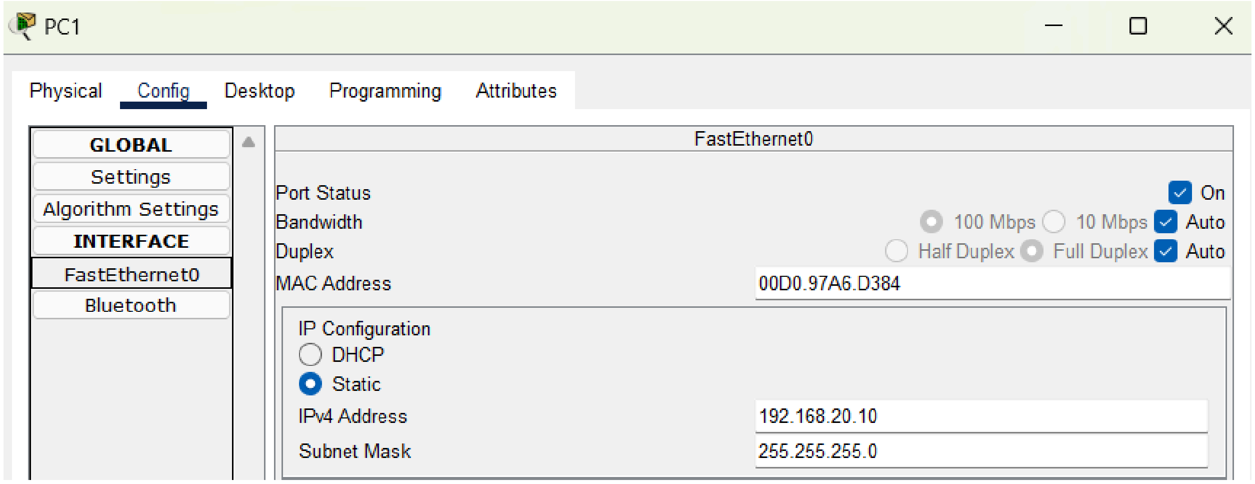
IP Configuration

○ DHCP

● Static

IPv4 Address 192.168.10.10

Subnet Mask 255.255.255.0



FastEthernet0

Port Status ☑ On

Bandwidth ○ 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ○ Full Duplex ☑ Auto

MAC Address 00D0.97A6.D384

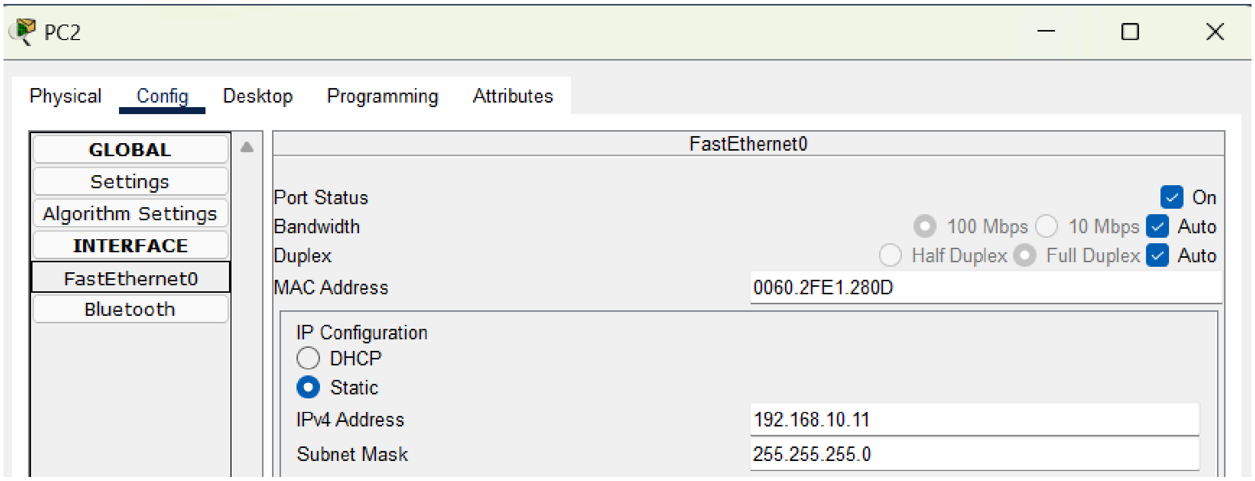
IP Configuration

○ DHCP

● Static

IPv4 Address 192.168.20.10

Subnet Mask 255.255.255.0



FastEthernet0

Port Status ☑ On

Bandwidth ○ 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ○ Full Duplex ☑ Auto

MAC Address 0060.2FE1.280D

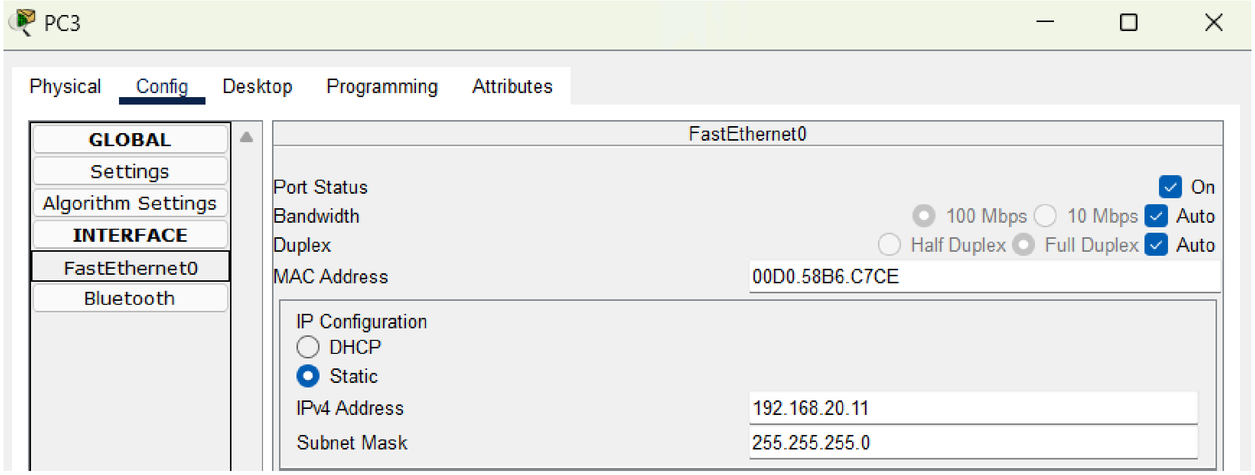
IP Configuration

○ DHCP

● Static

IPv4 Address 192.168.10.11

Subnet Mask 255.255.255.0



FastEthernet0

Port Status ☑ On

Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

MAC Address 00D0.58B6.C7CE

IP Configuration

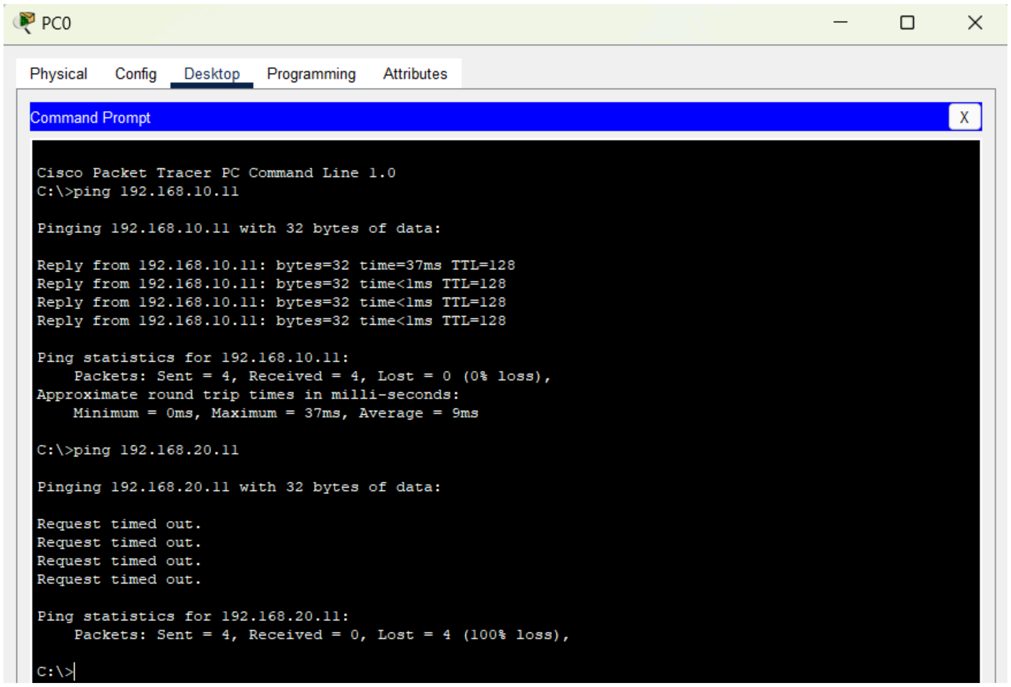
○ DHCP

● Static

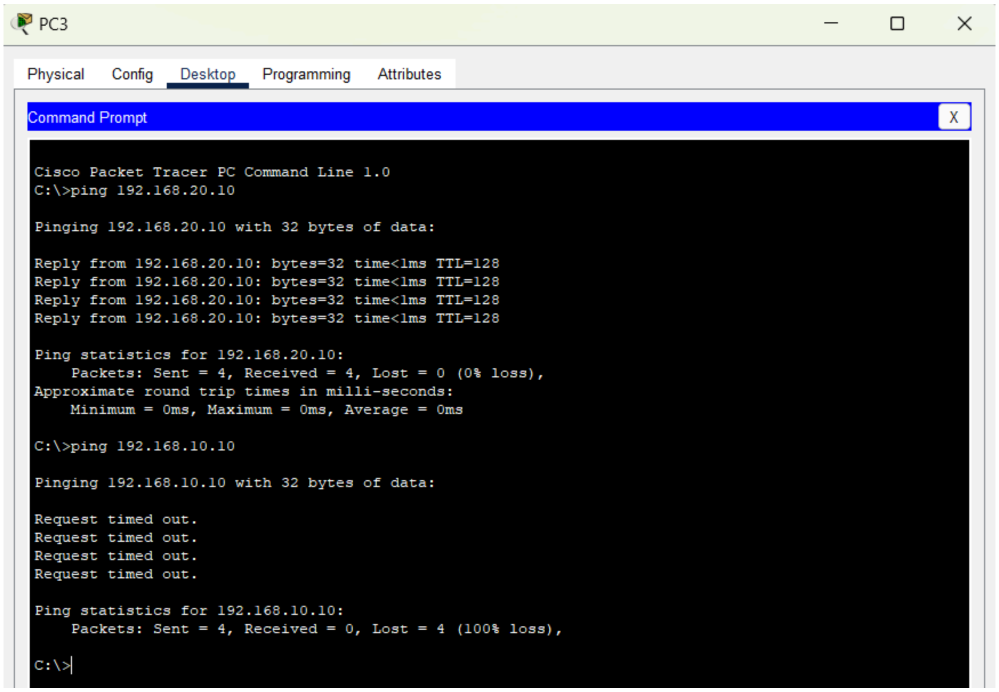
IPv4 Address 192.168.20.11

Subnet Mask 255.255.255.0

Try pinging from each PC, we will see that it allows PCs of same VLAN to ping but not of different VLANs.



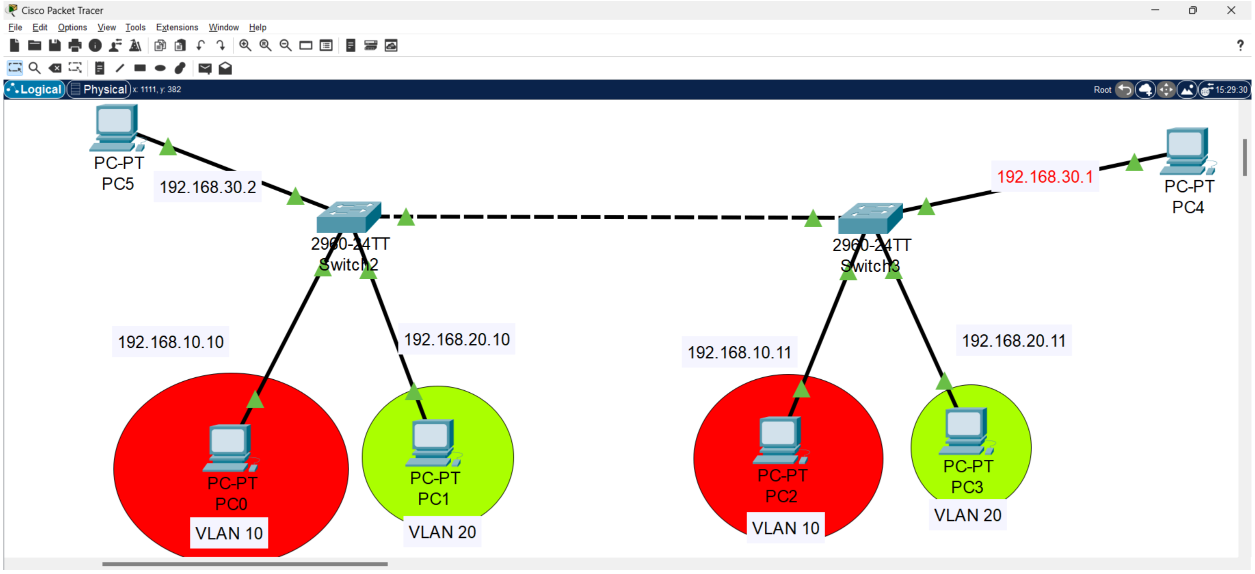
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.11

Pinging 192.168.10.11 with 32 bytes of data:

Reply from 192.168.10.11: bytes=32 time=37ms TTL=128
Reply from 192.168.10.11: bytes=32 time<1ms TTL=128
Reply from 192.168.10.11: bytes=32 time<1ms TTL=128
Reply from 192.168.10.11: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.10.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 37ms, Average = 9ms

C:\>ping 192.168.20.11

Pinging 192.168.20.11 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.20.11:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```



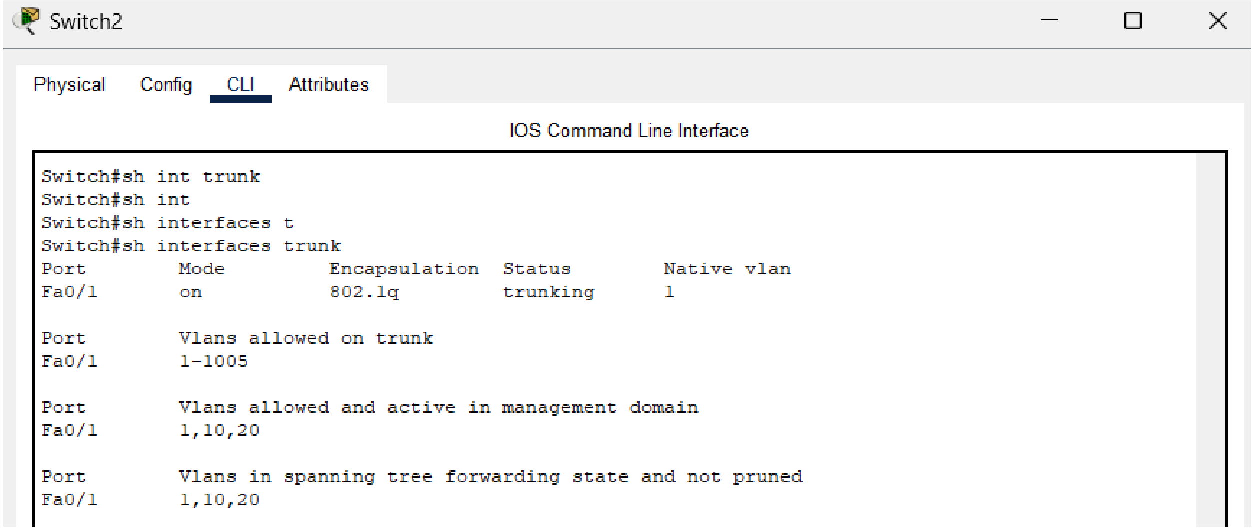
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.20.10

Pinging 192.168.20.10 with 32 bytes of data:

Reply from 192.168.20.10: bytes=32 time<1ms TTL=128
Reply from 192.168.20.10: bytes=32 time<1ms TTL=128
Reply from 192.168.20.10: bytes=32 time<1ms TTL=128
Reply from 192.168.20.10: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.20.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.10.10

Pinging 192.168.10.10 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.10.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

6. Change the native VLAN on a trunk port. Test for VLAN mismatches and troubleshooting.

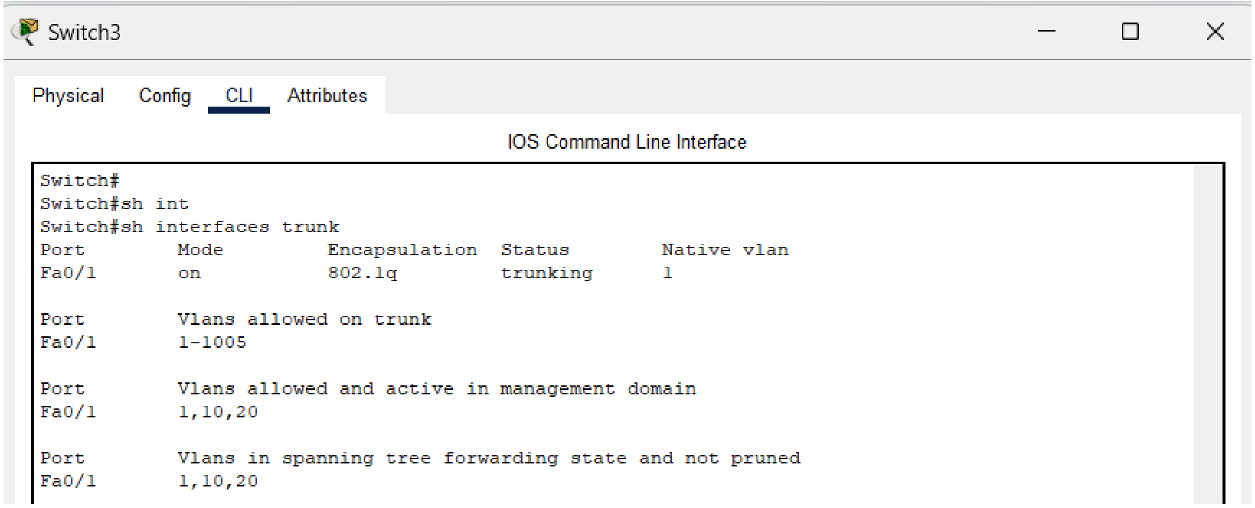
Configure the Network :



Lets see both the switches that both have native VLAN as 1 which is default :

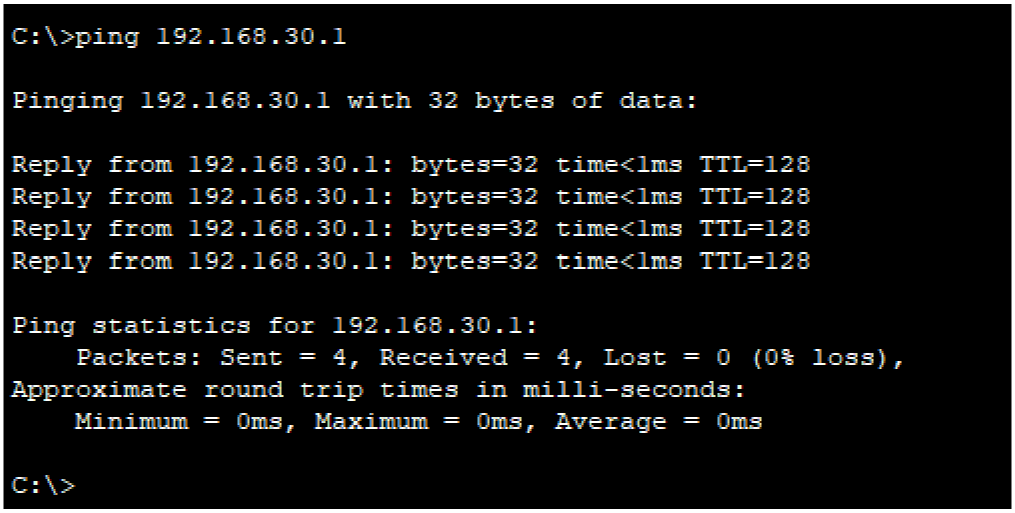


```
Switch#sh int trunk
Switch#sh int
Switch#sh interfaces t
Switch#sh interfaces trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/1       on           802.1q         trunking      1

Port        Vlans allowed on trunk
Fa0/1       1-1005

Port        Vlans allowed and active in management domain
Fa0/1       1,10,20

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/1       1,10,20
```



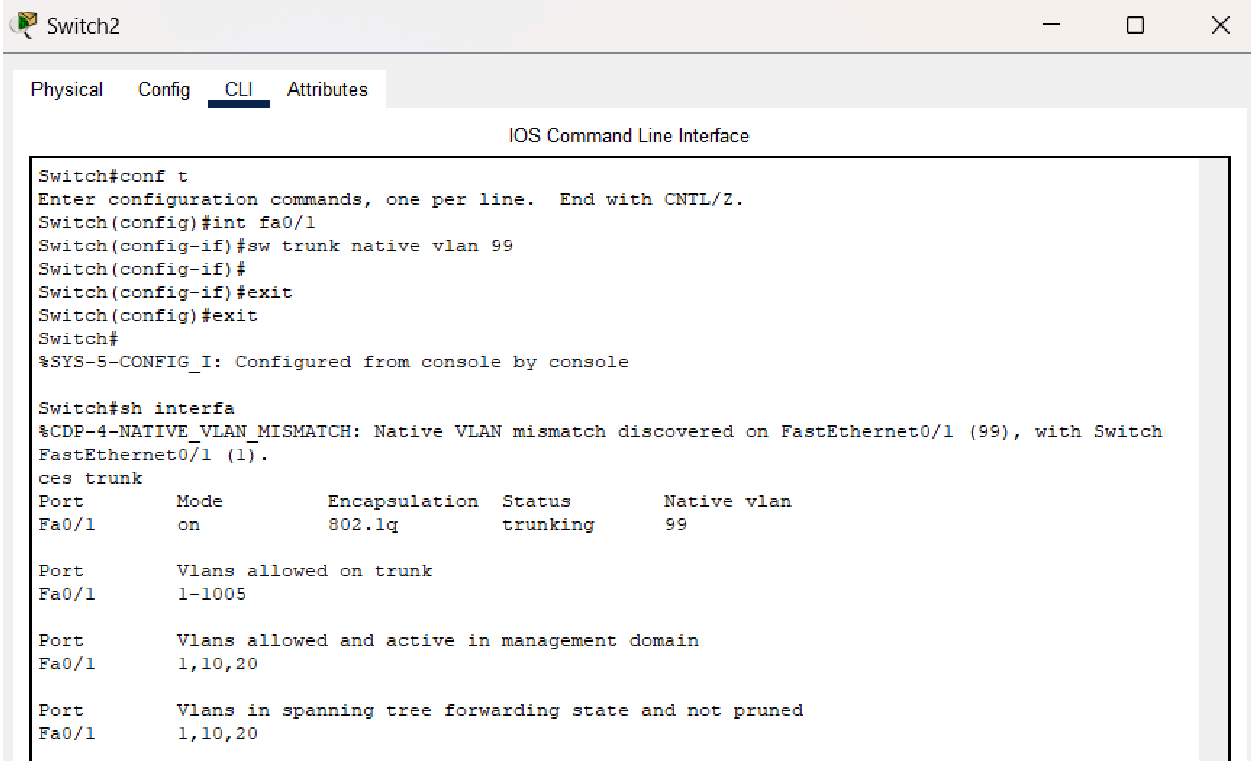
```
Switch#
Switch#sh int
Switch#sh interfaces trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/1       on           802.1q         trunking      1

Port        Vlans allowed on trunk
Fa0/1       1-1005

Port        Vlans allowed and active in management domain
Fa0/1       1,10,20

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/1       1,10,20
```

And the ping works perfectly for PCs which belongs to the default VLAN.



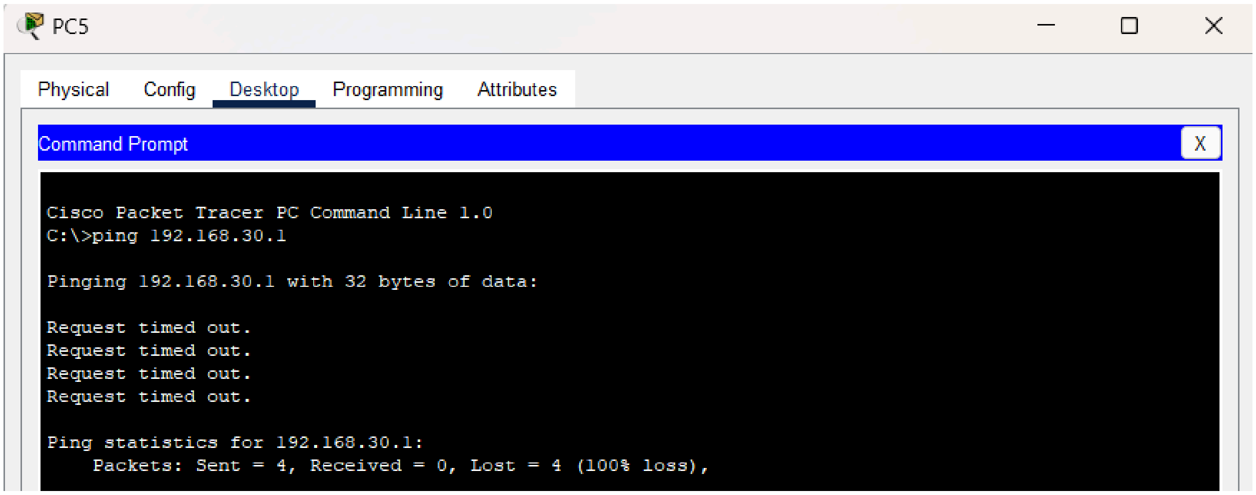
```
C:\>ping 192.168.30.1

Pinging 192.168.30.1 with 32 bytes of data:

Reply from 192.168.30.1: bytes=32 time<1ms TTL=128
Reply from 192.168.30.1: bytes=32 time<1ms TTL=128
Reply from 192.168.30.1: bytes=32 time<1ms TTL=128
Reply from 192.168.30.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.30.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

To misconfigure it, make one of the switch native vlan to 99.



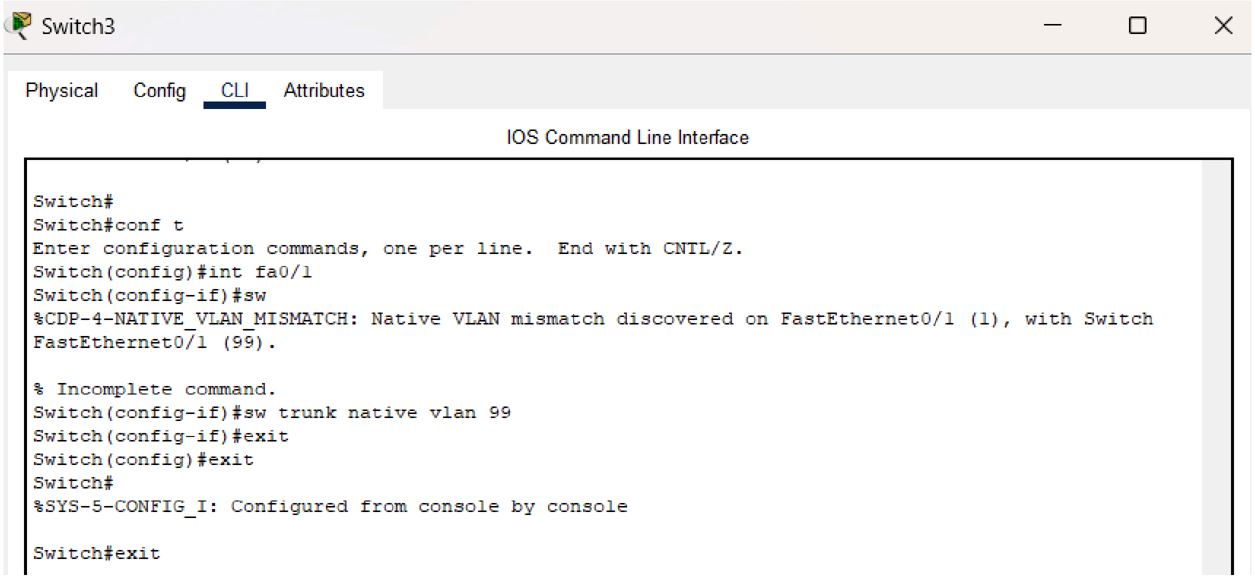
```
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#sw trunk native vlan 99
Switch(config-if)#
Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#sh interfa
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on FastEthernet0/1 (99), with Switch
FastEthernet0/1 (1).
ces trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/1       on           802.1q         trunking      99

Port        Vlans allowed on trunk
Fa0/1       1-1005

Port        Vlans allowed and active in management domain
Fa0/1       1,10,20

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/1       1,10,20
```

Now, the PCs won’t be able to connect because of the Native VLAN mismatch :



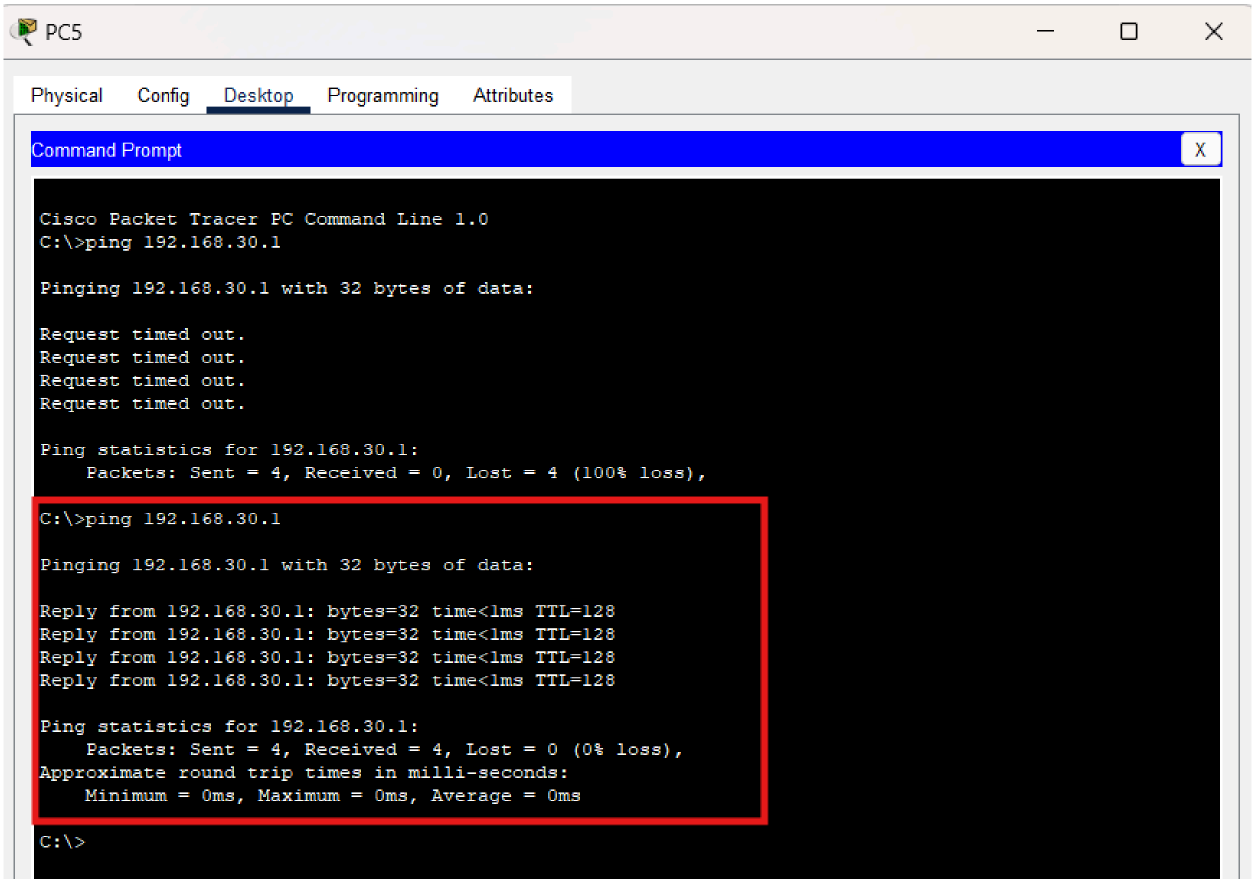
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.30.1

Pinging 192.168.30.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.30.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

To troubleshoot it, make the other switch VLAN also 99, and now the ping will work :



```
Switch#
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#sw
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on FastEthernet0/1 (1), with Switch
FastEthernet0/1 (99).

% Incomplete command.
Switch(config-if)#sw trunk native vlan 99
Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#exit
```

As the native VLANs matches, the ping again works perfectly.

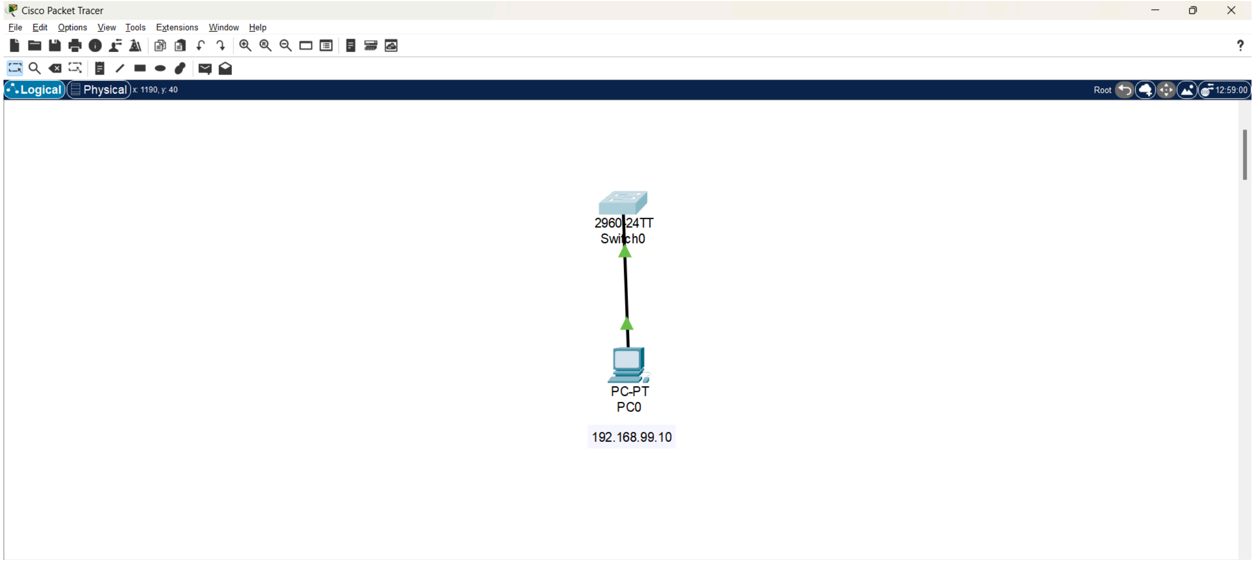


```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.30.1

Pinging 192.168.30.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.30.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 192.168.30.1

Pinging 192.168.30.1 with 32 bytes of data:

Reply from 192.168.30.1: bytes=32 time<1ms TTL=128
Reply from 192.168.30.1: bytes=32 time<1ms TTL=128
Reply from 192.168.30.1: bytes=32 time<1ms TTL=128
Reply from 192.168.30.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.30.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

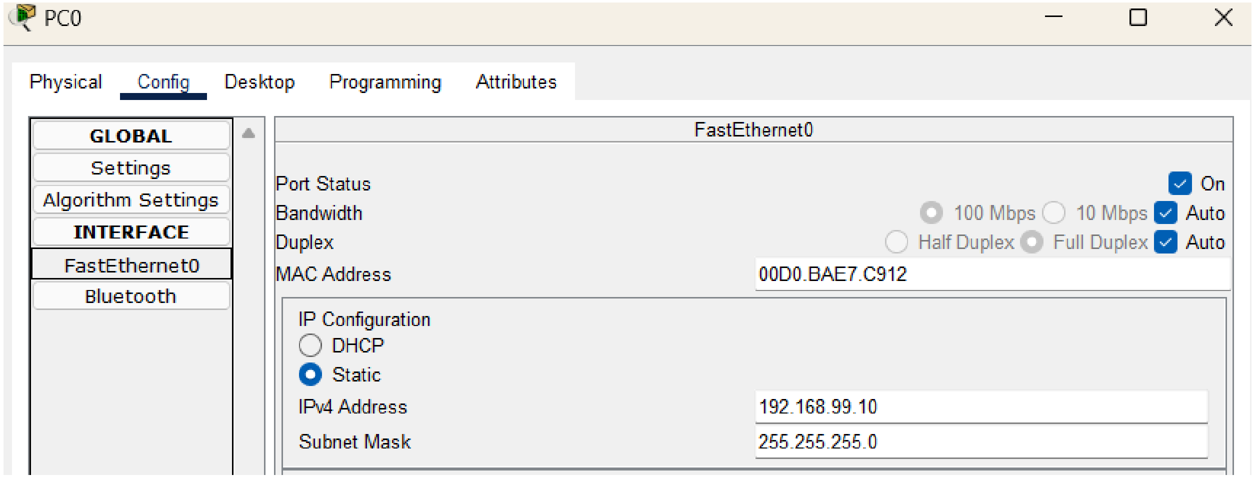
7. Configure a management VLAN and assign an IP address for remote access.

   Test SSH or Telnet access to the switch.\

Create this network in Packet Tracer :



Configure the IP address of PC :



FastEthernet0

Port Status ☑ On

Bandwidth ○ 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ○ Full Duplex ☑ Auto

MAC Address 00D0.BAE7.C912

IP Configuration

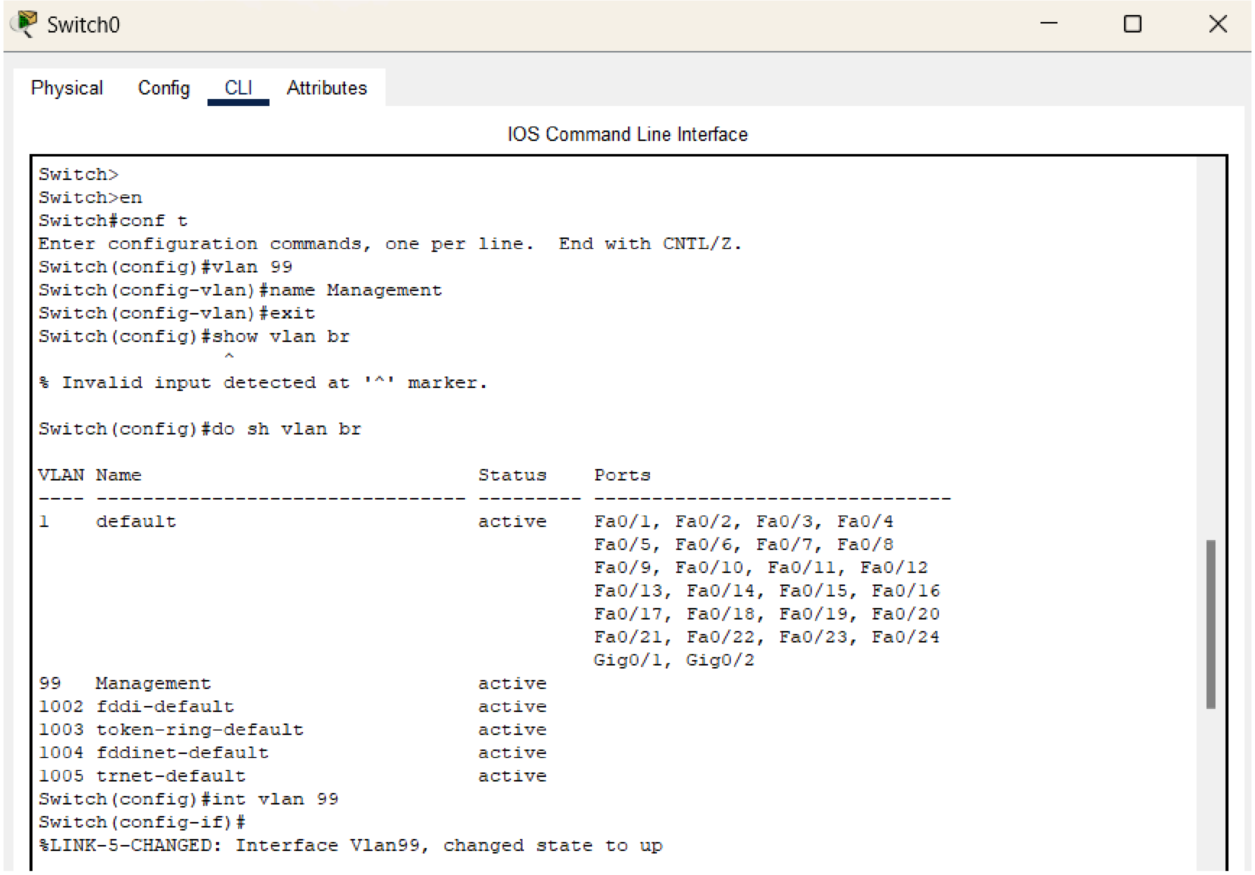
○ DHCP

● Static

IPv4 Address 192.168.99.10

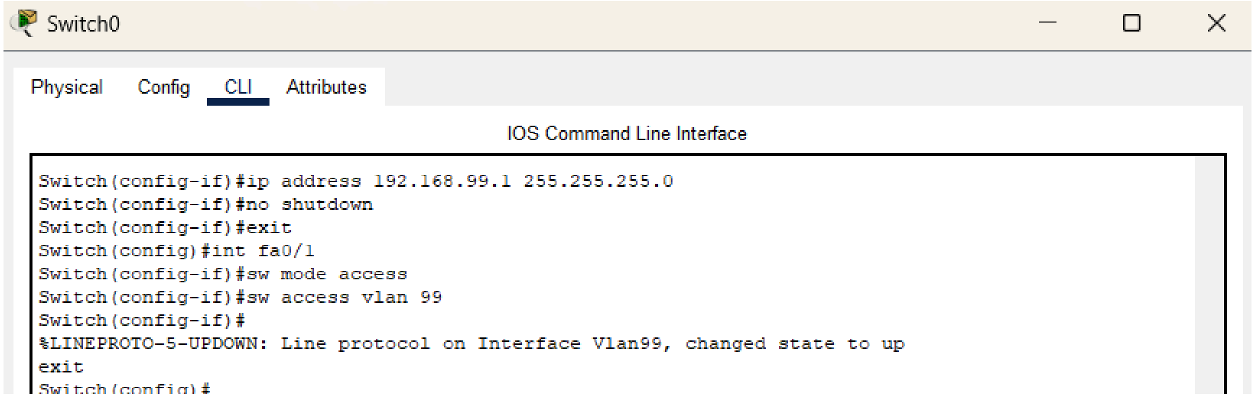
Subnet Mask 255.255.255.0

Create a VLAN 99 in the switch :



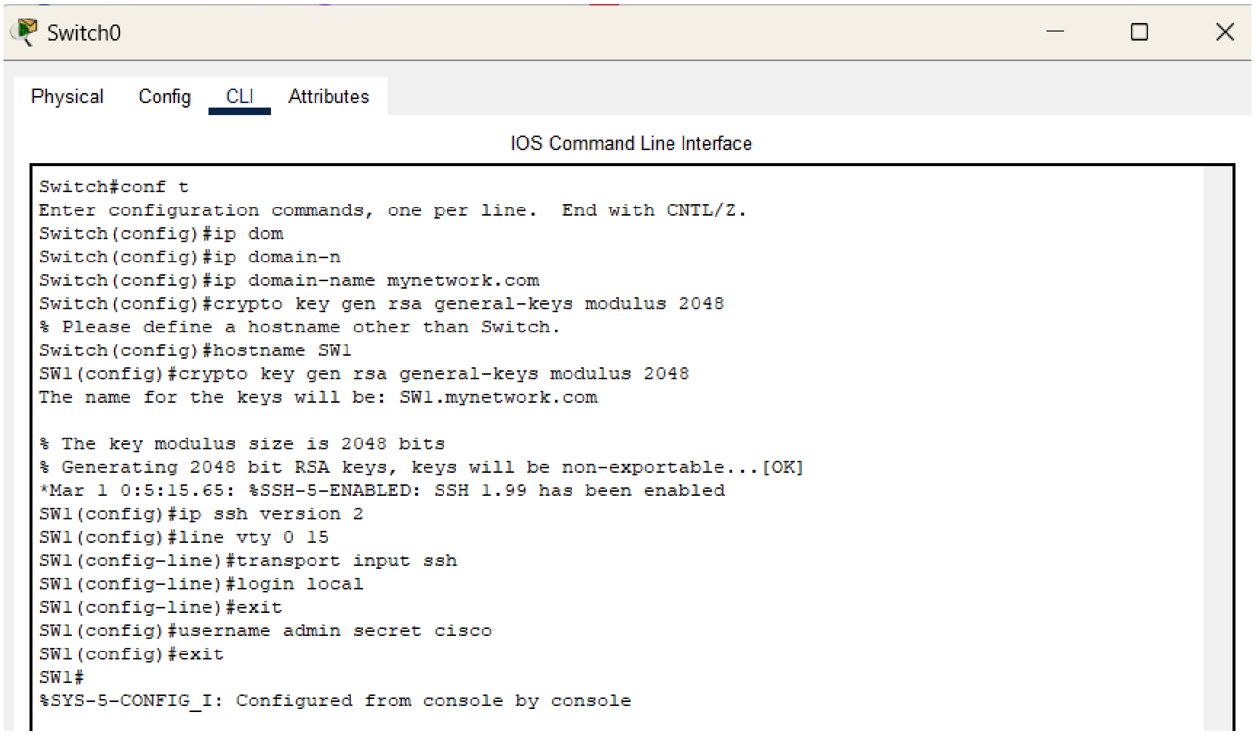
```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 99
Switch(config-vlan)#name Management
Switch(config-vlan)#exit
Switch(config)#show vlan br
                ^
% Invalid input detected at '^' marker.

Switch(config)#do sh vlan br

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
                                                Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                                Gig0/1, Gig0/2
99   Management                       active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch(config)#int vlan 99
Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan99, changed state to up
```

Assign IP address for the interface of the VLAN in switch :

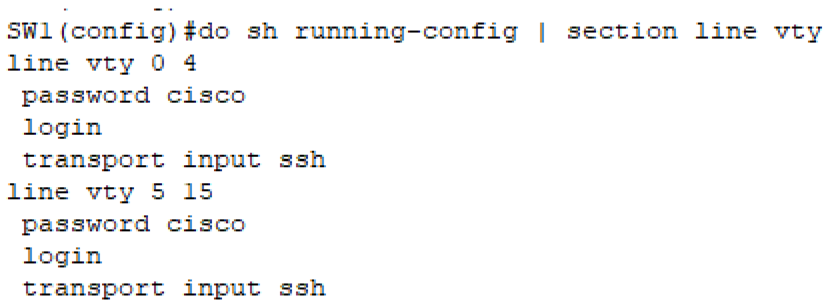


```
Switch(config-if)#ip address 192.168.99.1 255.255.255.0
Switch(config-if)#no shutdown
Switch(config-if)#exit
Switch(config)#int fa0/1
Switch(config-if)#sw mode access
Switch(config-if)#sw access vlan 99
Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state to up
exit
Switch(config)#
```

Do as given in the below screenshot to configure the SSH in switch :

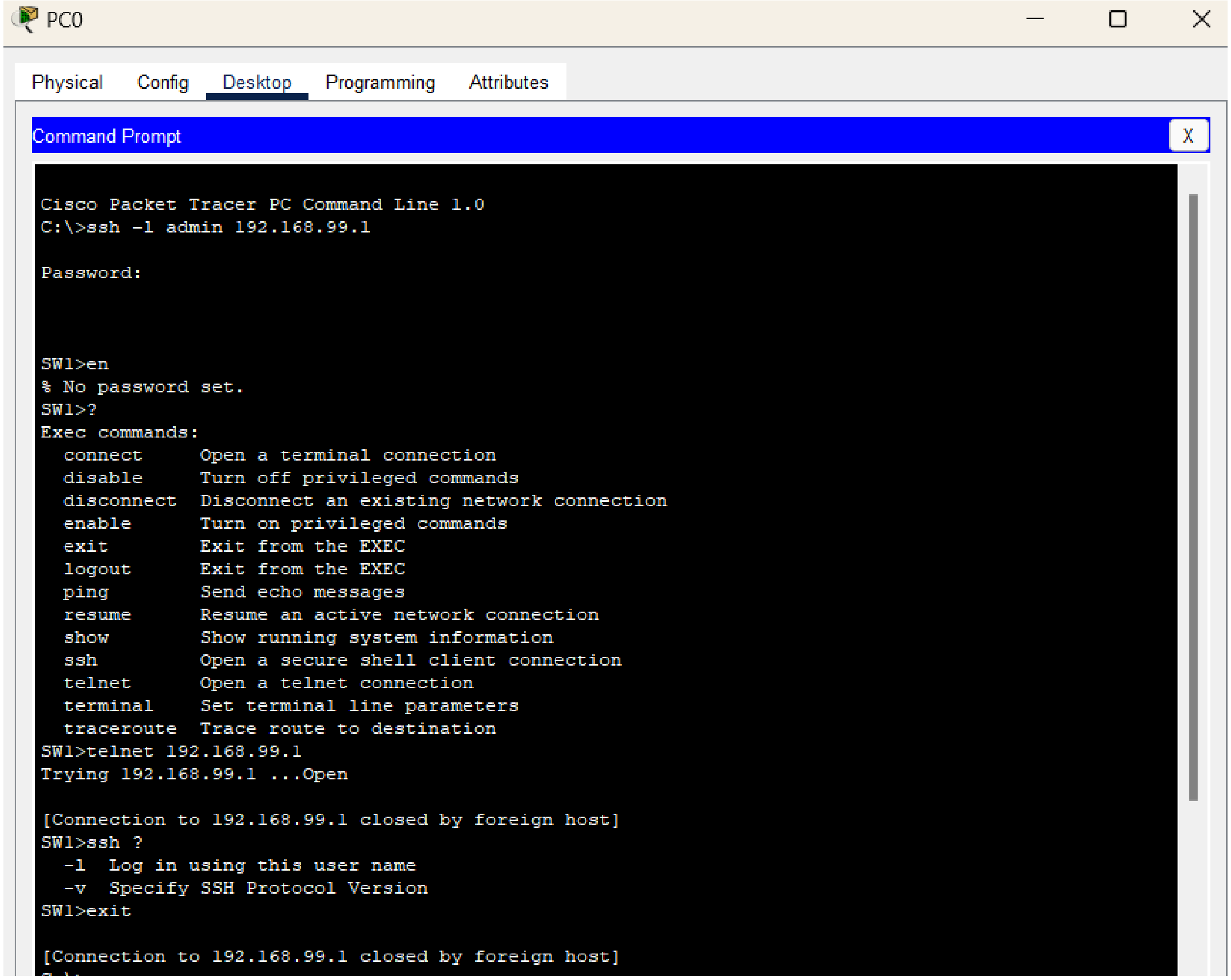


```
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#ip dom
Switch(config)#ip domain-n
Switch(config)#ip domain-name mynetwork.com
Switch(config)#crypto key gen rsa general-keys modulus 2048
% Please define a hostname other than Switch.
Switch(config)#hostname SW1
SW1(config)#crypto key gen rsa general-keys modulus 2048
The name for the keys will be: SW1.mynetwork.com

% The key modulus size is 2048 bits
% Generating 2048 bit RSA keys, keys will be non-exportable...[OK]
*Mar 1 0:5:15.65: %SSH-5-ENABLED: SSH 1.99 has been enabled
SW1(config)#ip ssh version 2
SW1(config)#line vty 0 15
SW1(config-line)#transport input ssh
SW1(config-line)#login local
SW1(config-line)#exit
SW1(config)#username admin secret cisco
SW1(config)#exit
SW1#
%SYS-5-CONFIG_I: Configured from console by console
```



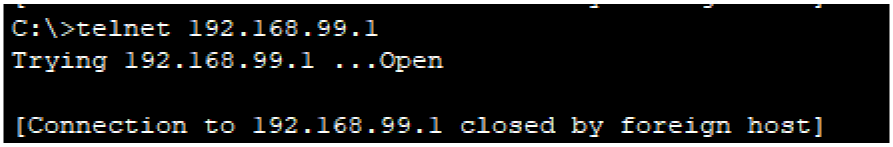
```
SW1(config)#do sh running-config | section line vty
line vty 0 4
 password cisco
 login
 transport input ssh
line vty 5 15
 password cisco
 login
 transport input ssh
```

Now in PC we will be able to connect to switch using SSH :



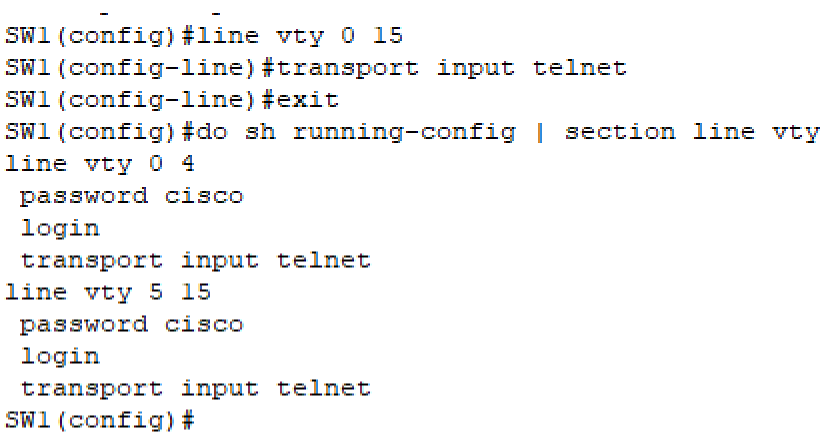
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ssh -l admin 192.168.99.1

Password:



SW1>en
% No password set.
SW1>?
Exec commands:
  connect     Open a terminal connection
  disable     Turn off privileged commands
  disconnect  Disconnect an existing network connection
  enable      Turn on privileged commands
  exit        Exit from the EXEC
  logout      Exit from the EXEC
  ping        Send echo messages
  resume      Resume an active network connection
  show        Show running system information
  ssh         Open a secure shell client connection
  telnet      Open a telnet connection
  terminal    Set terminal line parameters
  traceroute  Trace route to destination
SW1>telnet 192.168.99.1
Trying 192.168.99.1 ...Open

[Connection to 192.168.99.1 closed by foreign host]
SW1>ssh ?
  -l  Log in using this user name
  -v  Specify SSH Protocol Version
SW1>exit

[Connection to 192.168.99.1 closed by foreign host]
```

We will not be able to connect to telnet and we have to configure it :



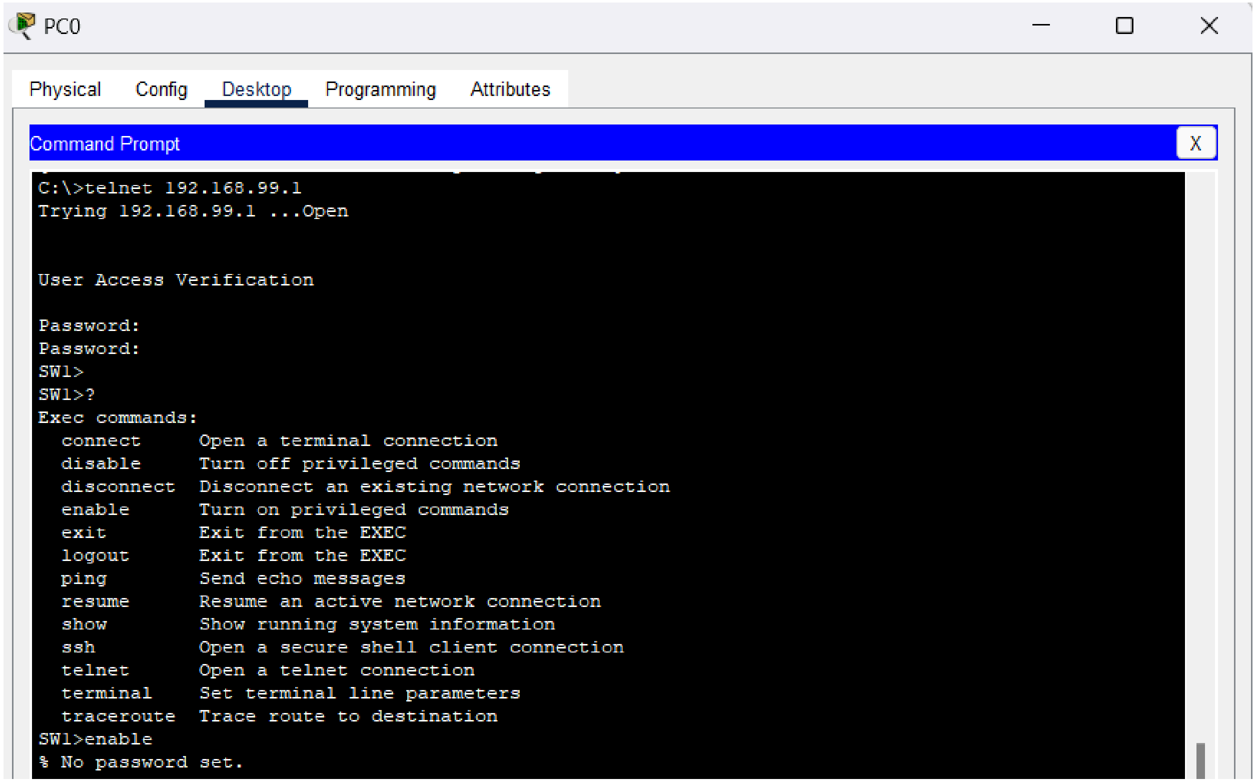
```
C:\>telnet 192.168.99.1
Trying 192.168.99.1 ...Open

[Connection to 192.168.99.1 closed by foreign host]
```

To Configure telnet, just change the transport input from SSH to telnet and save.



```
SW1(config)#line vty 0 15
SW1(config-line)#transport input telnet
SW1(config-line)#exit
SW1(config)#do sh running-config | section line vty
line vty 0 4
 password cisco
 login
 transport input telnet
line vty 5 15
 password cisco
 login
 transport input telnet
SW1(config)#
```

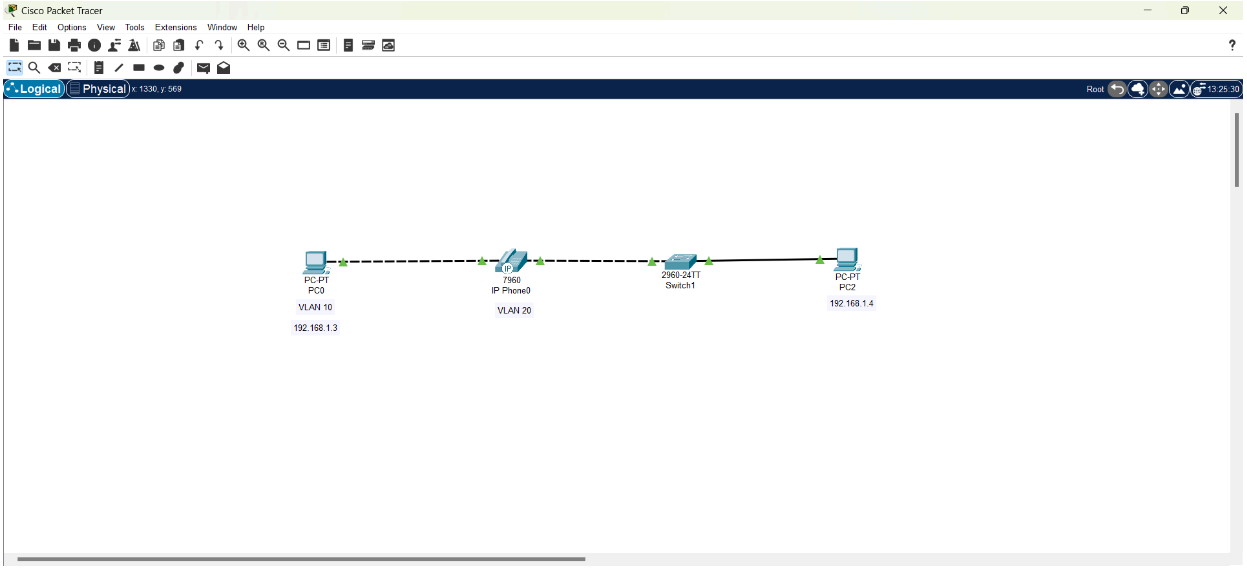
Now you will able to access Switch using telnet :



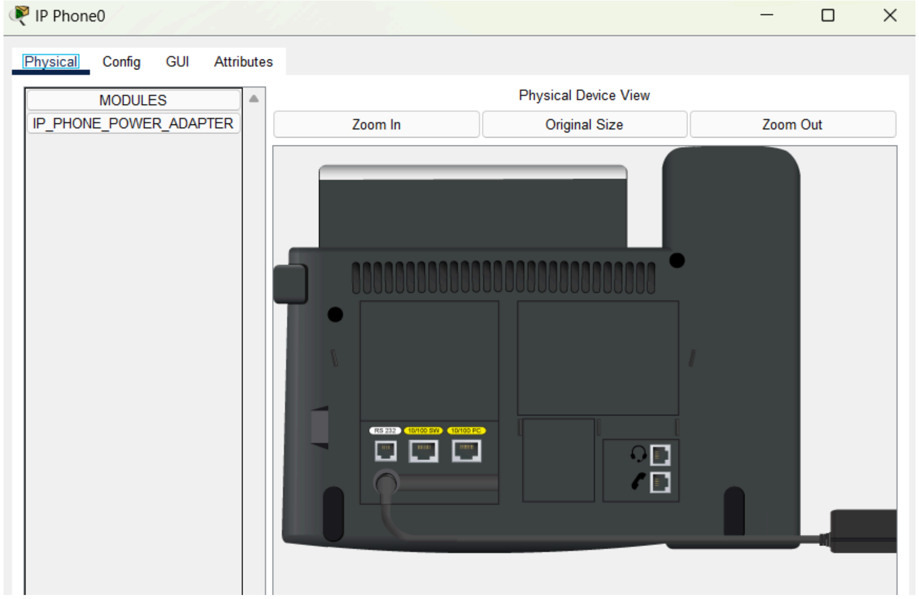
```
C:\>telnet 192.168.99.1
Trying 192.168.99.1 ...Open


User Access Verification

Password:
Password:
SW1>
SW1>?
Exec commands:
  connect     Open a terminal connection
  disable     Turn off privileged commands
  disconnect  Disconnect an existing network connection
  enable      Turn on privileged commands
  exit        Exit from the EXEC
  logout      Exit from the EXEC
  ping        Send echo messages
  resume      Resume an active network connection
  show        Show running system information
  ssh         Open a secure shell client connection
  telnet      Open a telnet connection
  terminal    Set terminal line parameters
  traceroute  Trace route to destination
SW1>enable
% No password set.
```

8. You have a Cisco switch and a VoIP phone that needs to be placed in a voice VLAN (VLAN 20). The data for the PC should remain in a separate VLAN (VLAN 10). Configure the switch port to support both voice and data traffic.

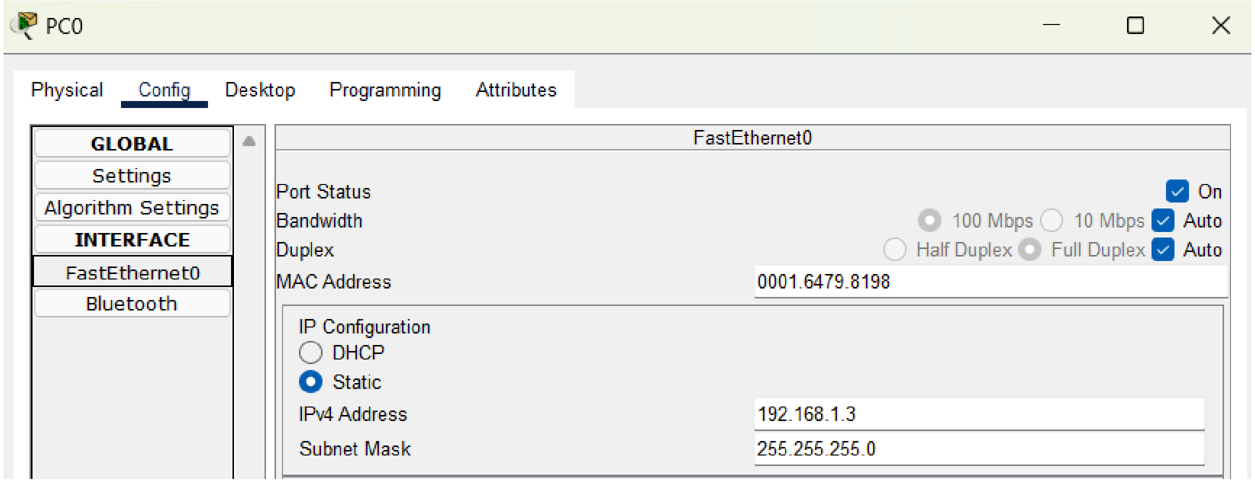
Create this network in the Cisco Packet Tracer :



Make sure to connect the Adapter module to power up the VoIP Phone :



Configure the PCs with their IP address and subnet mask and also the default gateway :



FastEthernet0

Port Status ☑ On

Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

MAC Address 0001.6479.8198

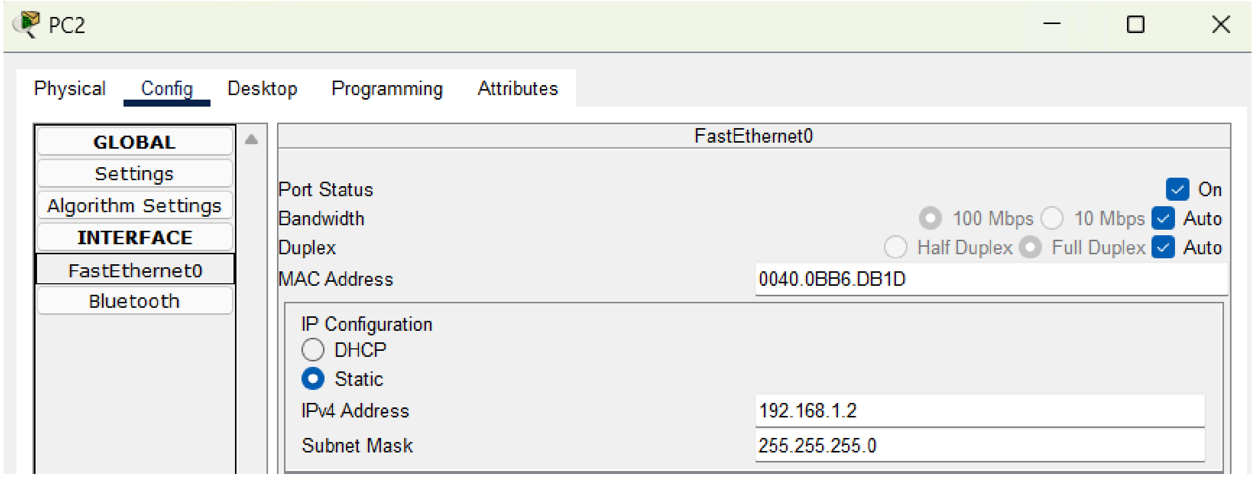
IP Configuration

○ DHCP

● Static

IPv4 Address 192.168.1.3

Subnet Mask 255.255.255.0



FastEthernet0

Port Status ☑ On

Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

MAC Address 0040.0BB6.DB1D

IP Configuration

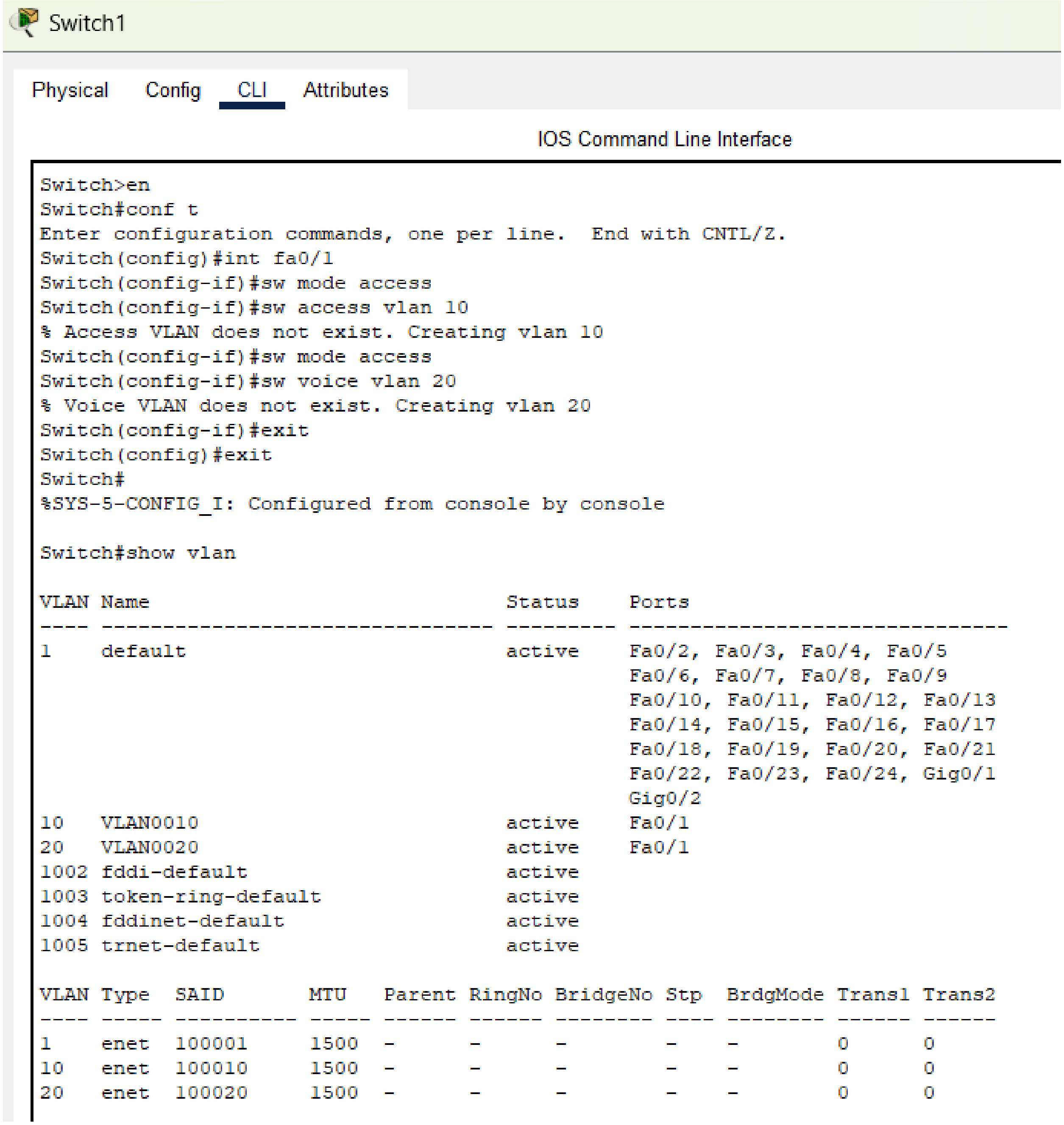
○ DHCP

● Static

IPv4 Address 192.168.1.2

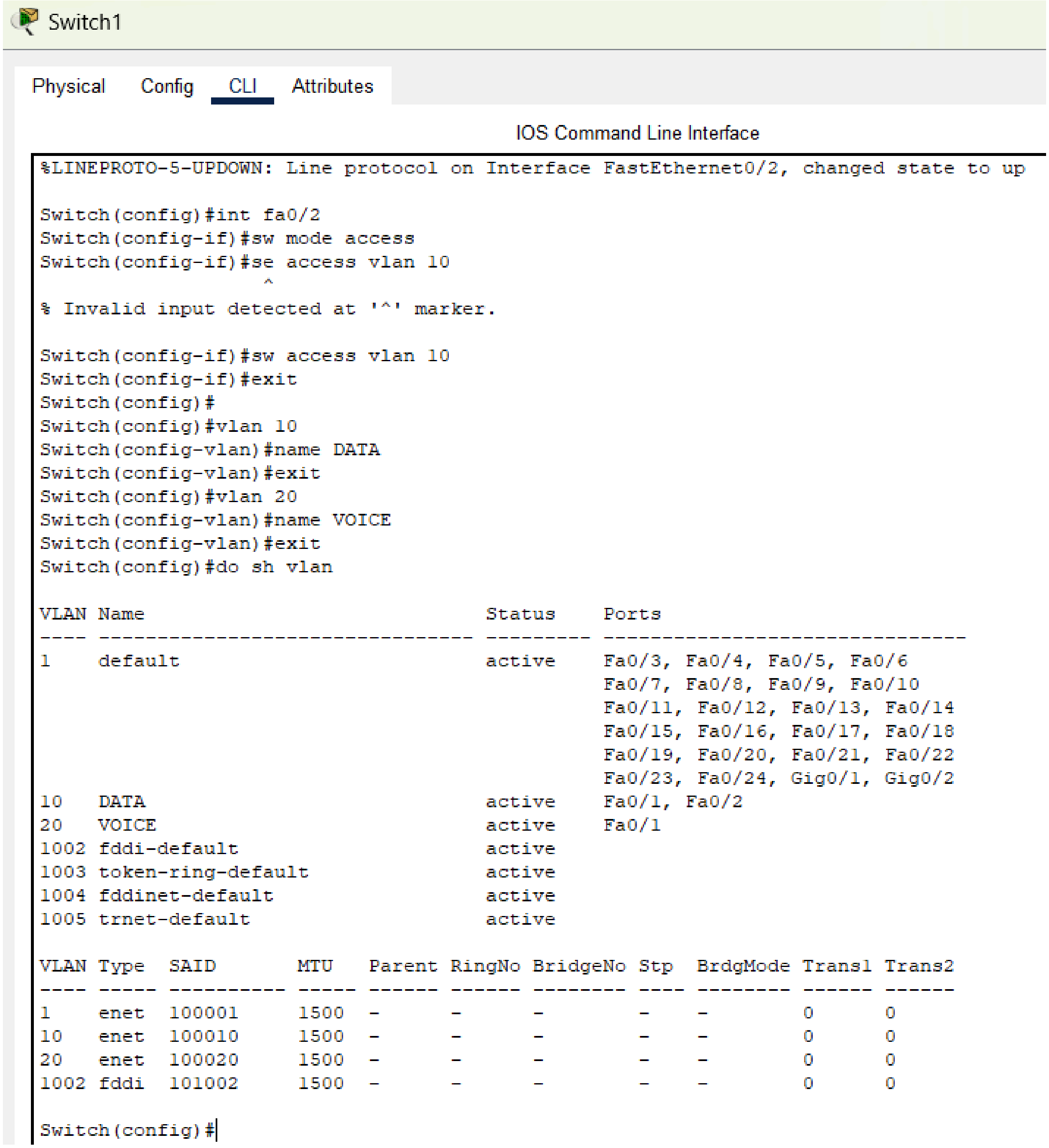
Subnet Mask 255.255.255.0

Configure the switch with access mode and create the vlans respectively :



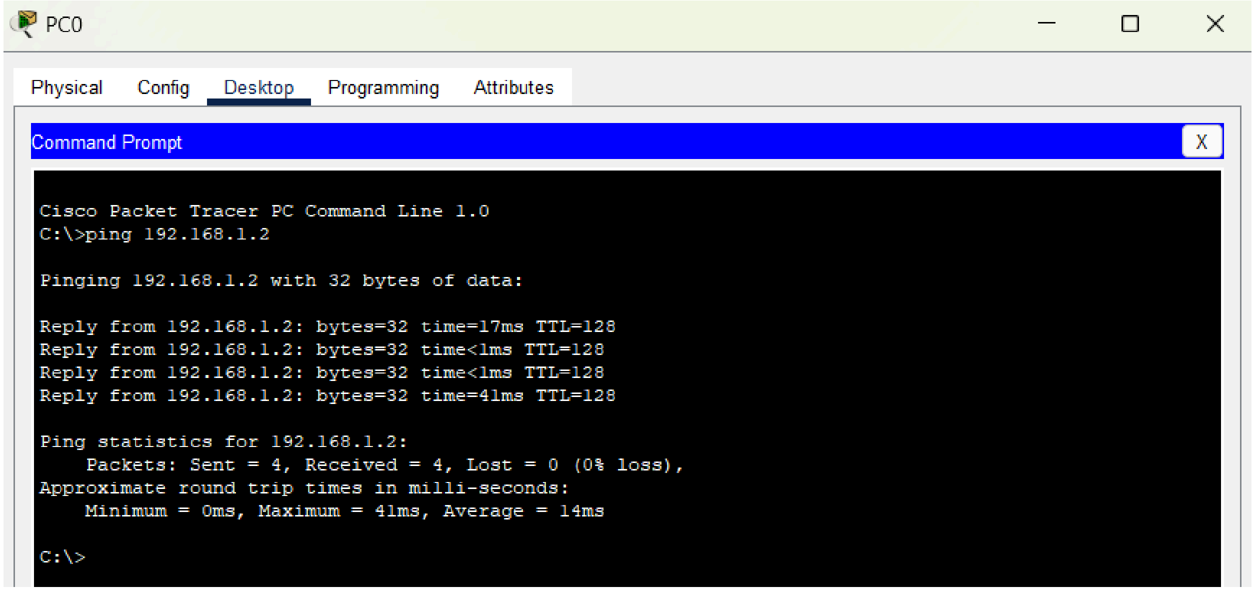
```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#sw mode access
Switch(config-if)#sw access vlan 10
% Access VLAN does not exist. Creating vlan 10
Switch(config-if)#sw mode access
Switch(config-if)#sw voice vlan 20
% Voice VLAN does not exist. Creating vlan 20
Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#show vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    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, Fa0/21
                                                Fa0/22, Fa0/23, Fa0/24, Gig0/1
                                                Gig0/2
10   VLAN0010                         active    Fa0/1
20   VLAN0020                         active    Fa0/1
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
20   enet  100020     1500  -      -      -        -    -        0      0
```

Rename them and add the interfaces accordingly to the VLANs :



```
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up

Switch(config)#int fa0/2
Switch(config-if)#sw mode access
Switch(config-if)#se access vlan 10
                   ^
% Invalid input detected at '^' marker.

Switch(config-if)#sw access vlan 10
Switch(config-if)#exit
Switch(config)#
Switch(config)#vlan 10
Switch(config-vlan)#name DATA
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#name VOICE
Switch(config-vlan)#exit
Switch(config)#do sh vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    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, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24, Gig0/1, Gig0/2
10   DATA                             active    Fa0/1, Fa0/2
20   VOICE                            active    Fa0/1
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
20   enet  100020     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0

Switch(config)#
```

Lets try pinging from PC0 to PC1 and we see that it is working :

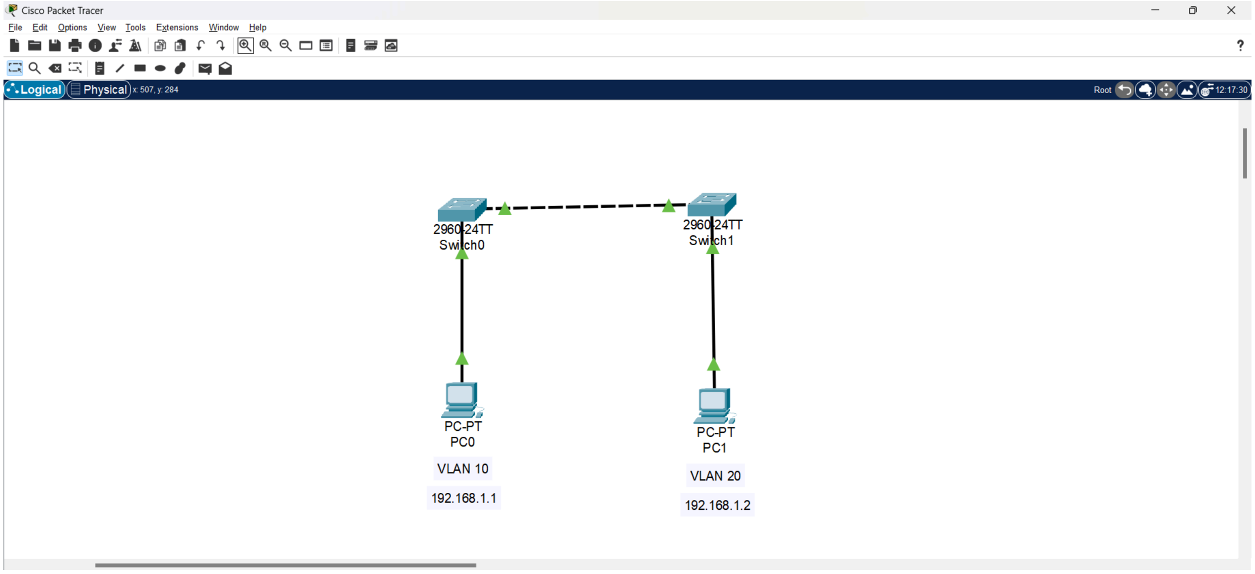


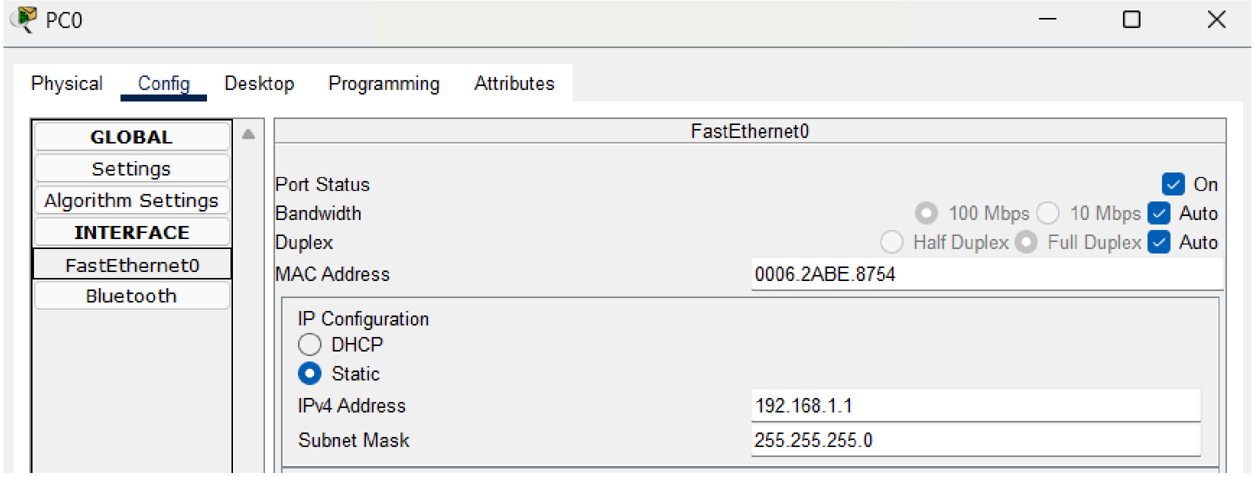
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=17ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time=41ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 41ms, Average = 14ms

C:\>
```

Since there is isn’t any configuration that can be done in IP phone for IP address, we are not going to test it and assume that the traffic of Voice is not interfered by the traffic of data.

9. You configured VLANs 10 and 20 on your switch and assigned ports to each VLAN. However, devices in VLAN 10 cannot communicate with devices in VLAN 20. Troubleshoot the issue.

Configure the network and assign the IP address and subnet mask respectively :





FastEthernet0

Port Status ☑ On

Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

MAC Address 0006.2ABE.8754

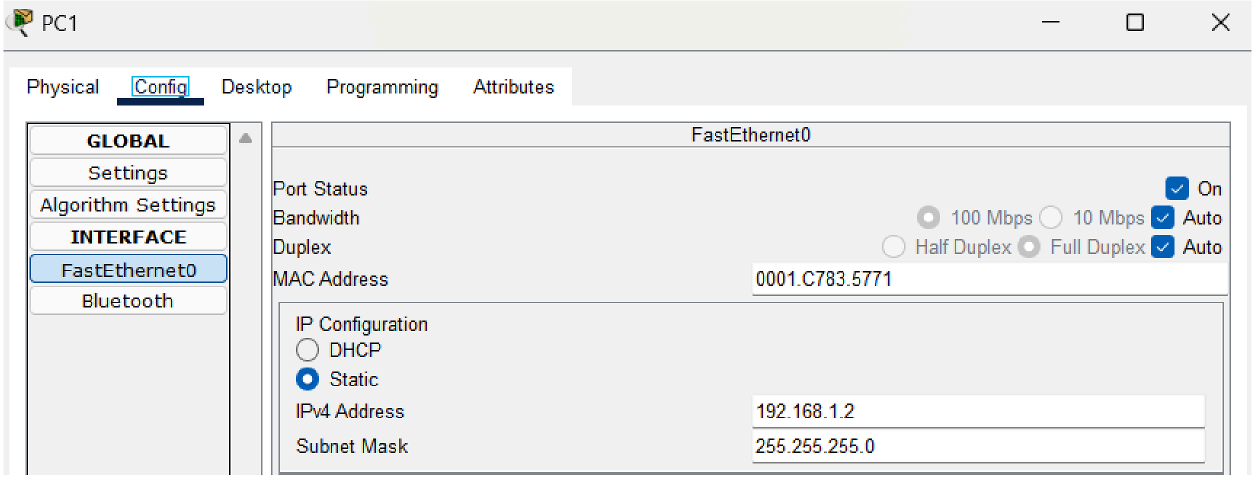
IP Configuration

○ DHCP

● Static

IPv4 Address 192.168.1.1

Subnet Mask 255.255.255.0



FastEthernet0

Port Status ☑ On

Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

MAC Address 0001.C783.5771

IP Configuration

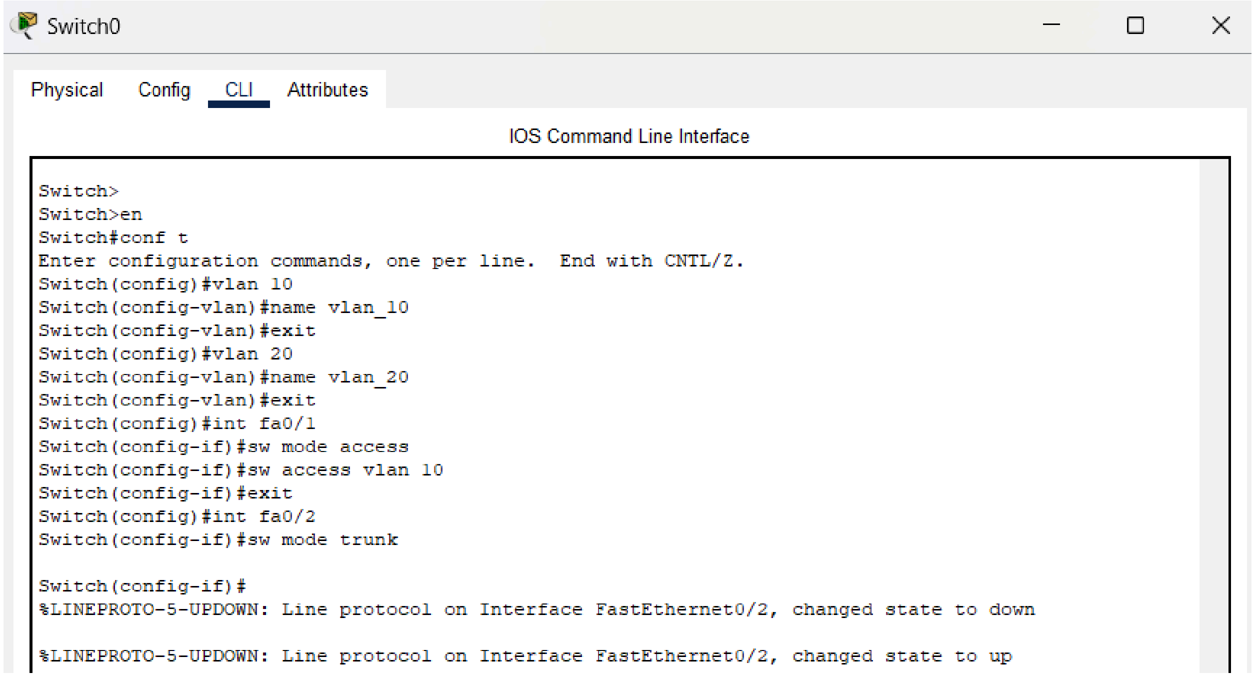
○ DHCP

● Static

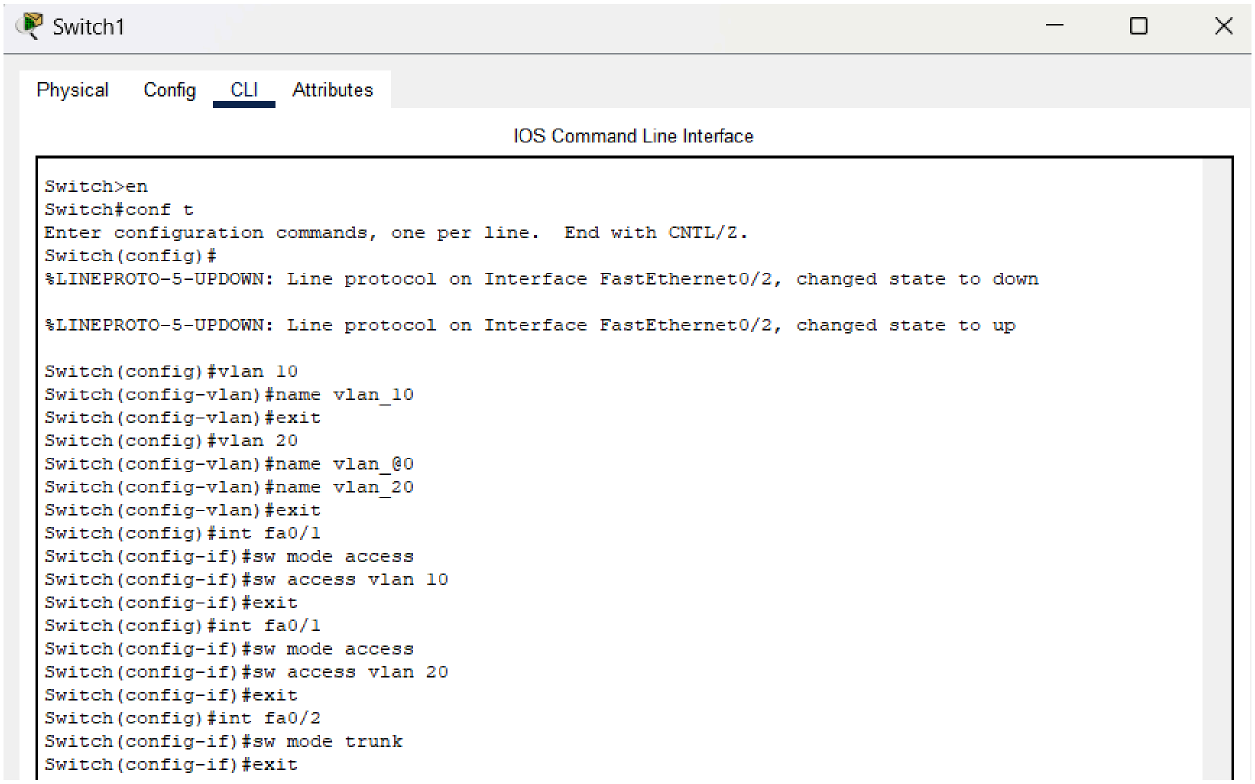
IPv4 Address 192.168.1.2

Subnet Mask 255.255.255.0

In each of the switches, create the vlans, name them and add the interfaces respectively with access mode or trunk mode depending on connection with end host or switch.

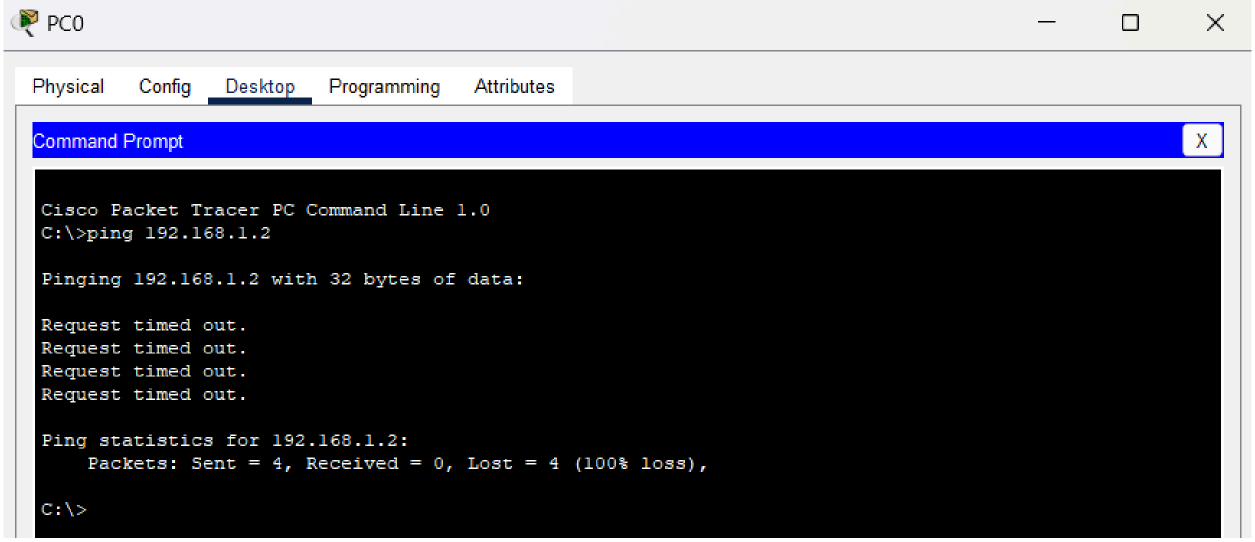


```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name vlan_10
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#name vlan_20
Switch(config-vlan)#exit
Switch(config)#int fa0/1
Switch(config-if)#sw mode access
Switch(config-if)#sw access vlan 10
Switch(config-if)#exit
Switch(config)#int fa0/2
Switch(config-if)#sw mode trunk

Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up
```



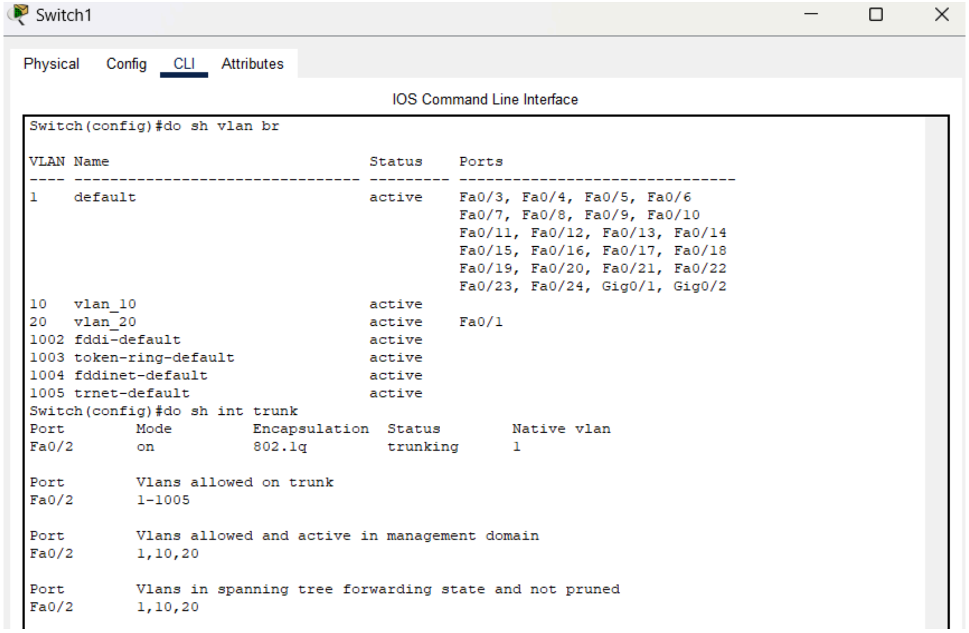
```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up

Switch(config)#vlan 10
Switch(config-vlan)#name vlan_10
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#name vlan_@0
Switch(config-vlan)#name vlan_20
Switch(config-vlan)#exit
Switch(config)#int fa0/1
Switch(config-if)#sw mode access
Switch(config-if)#sw access vlan 10
Switch(config-if)#exit
Switch(config)#int fa0/1
Switch(config-if)#sw mode access
Switch(config-if)#sw access vlan 20
Switch(config-if)#exit
Switch(config)#int fa0/2
Switch(config-if)#sw mode trunk
Switch(config-if)#exit
```

On trying to ping from PC 0 to PC 1 we find that it is not able ping.

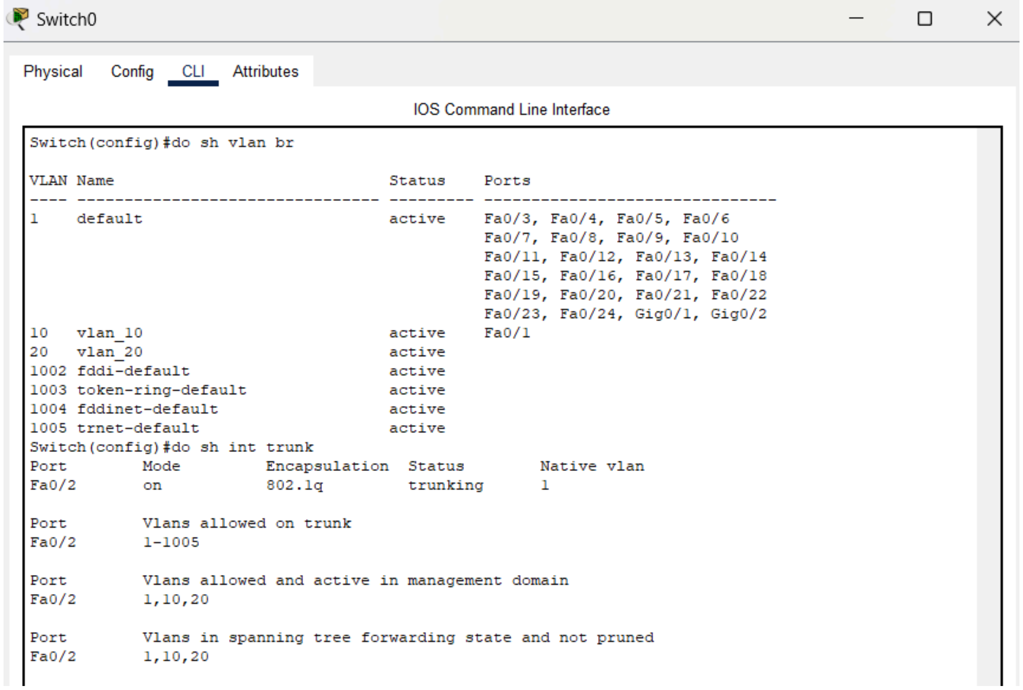


```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

On checking both the vlans and trunk interfaces, we see that both the PC are in different vlans and there is not inter-vlan routing. So we have to change them to have to use the same vlan.

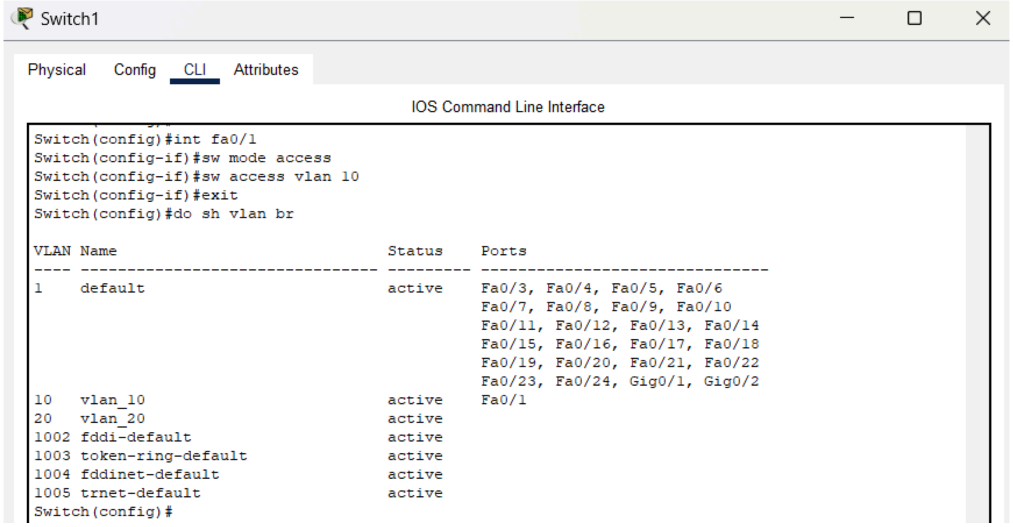


```
Switch(config)#do sh vlan br

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    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, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24, Gig0/1, Gig0/2
10   vlan_10                          active
20   vlan_20                          active    Fa0/1
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch(config)#do sh int trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/2       on           802.1q         trunking      1

Port        Vlans allowed on trunk
Fa0/2       1-1005

Port        Vlans allowed and active in management domain
Fa0/2       1,10,20

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/2       1,10,20
```



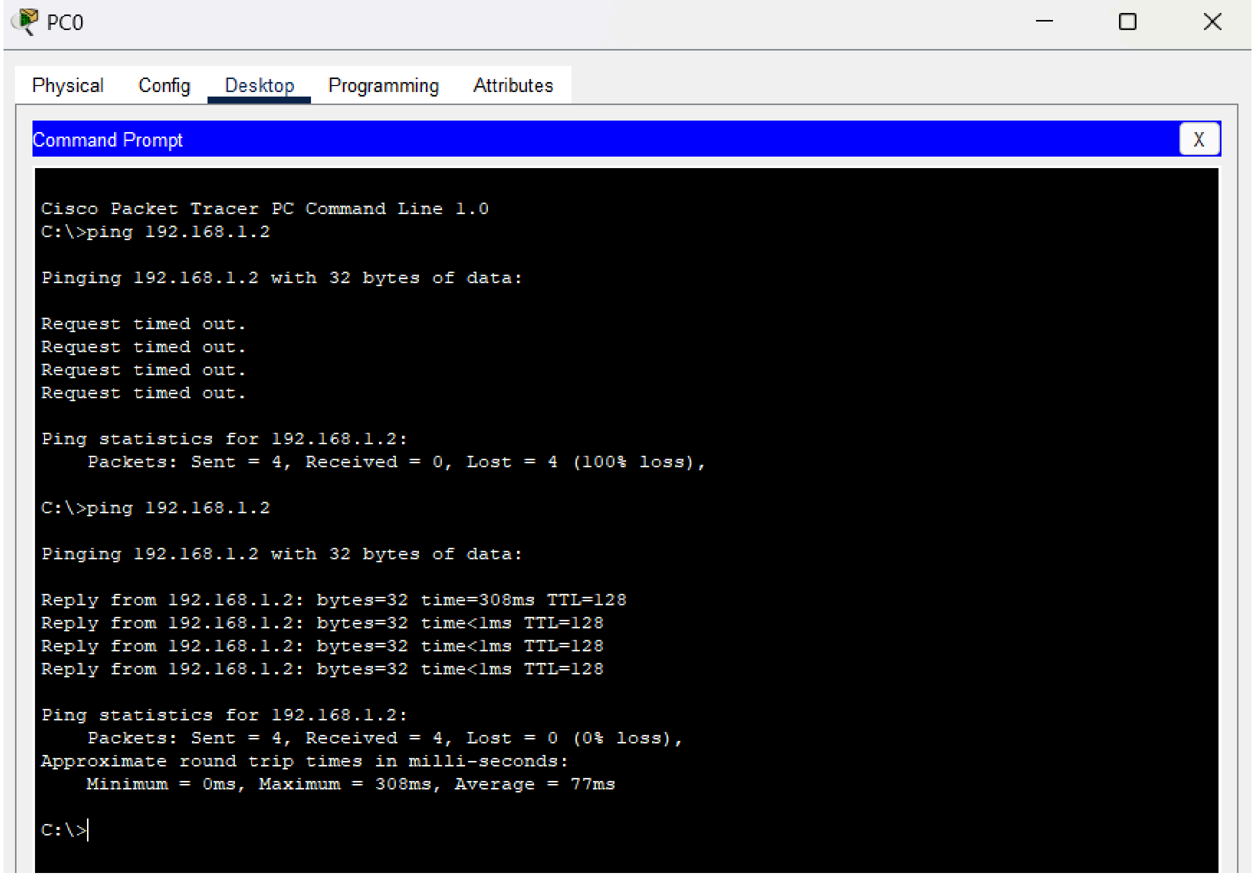
```
Switch(config)#do sh vlan br

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    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, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24, Gig0/1, Gig0/2
10   vlan_10                          active    Fa0/1
20   vlan_20                          active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch(config)#do sh int trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/2       on           802.1q         trunking      1

Port        Vlans allowed on trunk
Fa0/2       1-1005

Port        Vlans allowed and active in management domain
Fa0/2       1,10,20

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/2       1,10,20
```

Changing the Switch Fa0/1 to use vlan 10 :



```
Switch(config)#int fa0/1
Switch(config-if)#sw mode access
Switch(config-if)#sw access vlan 10
Switch(config-if)#exit
Switch(config)#do sh vlan br

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    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, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24, Gig0/1, Gig0/2
10   vlan_10                          active    Fa0/1
20   vlan_20                          active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
Switch(config)#
```

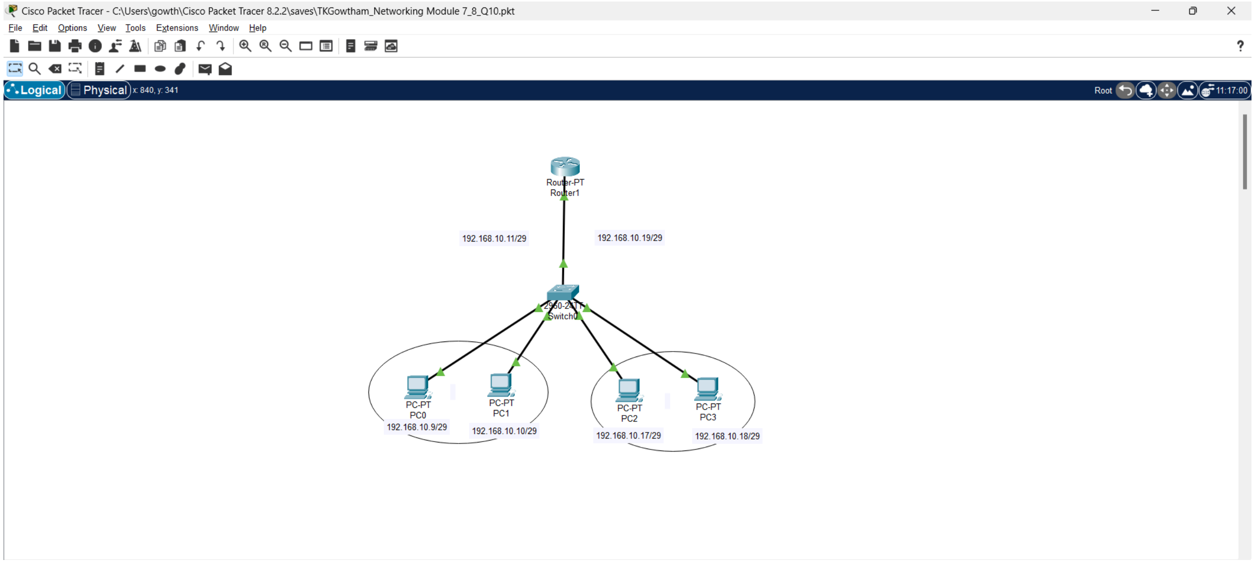
Now, on checking the ping, it is able to successfully ping.

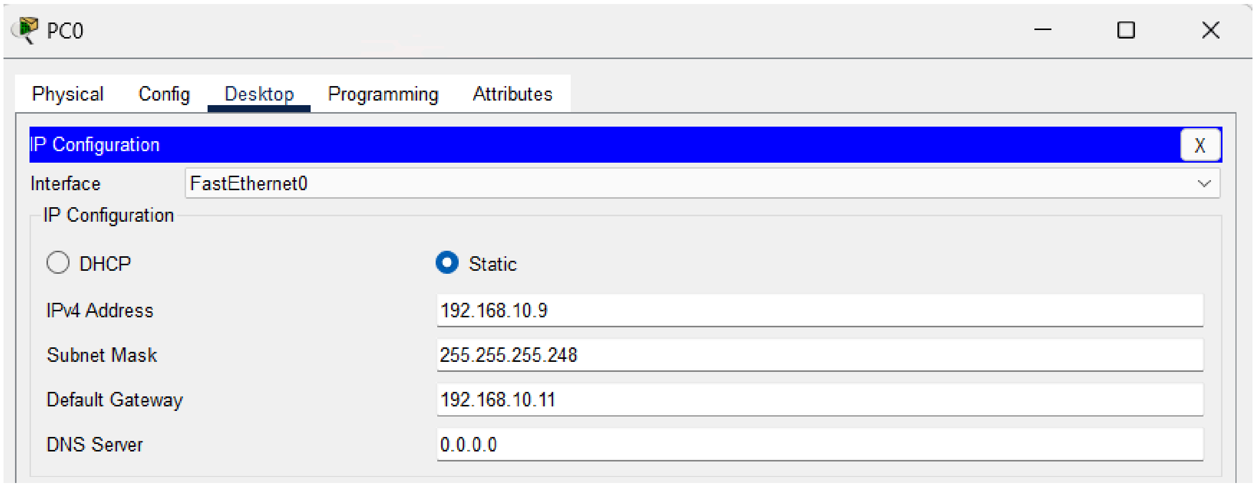


```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=308ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 308ms, Average = 77ms

C:\>
```

10. Try Inter VLAN routing with Router.

Create this network in cisco packet tracer and assign the IP address, subnet mask and the default gateway to respective PCs accordingly.





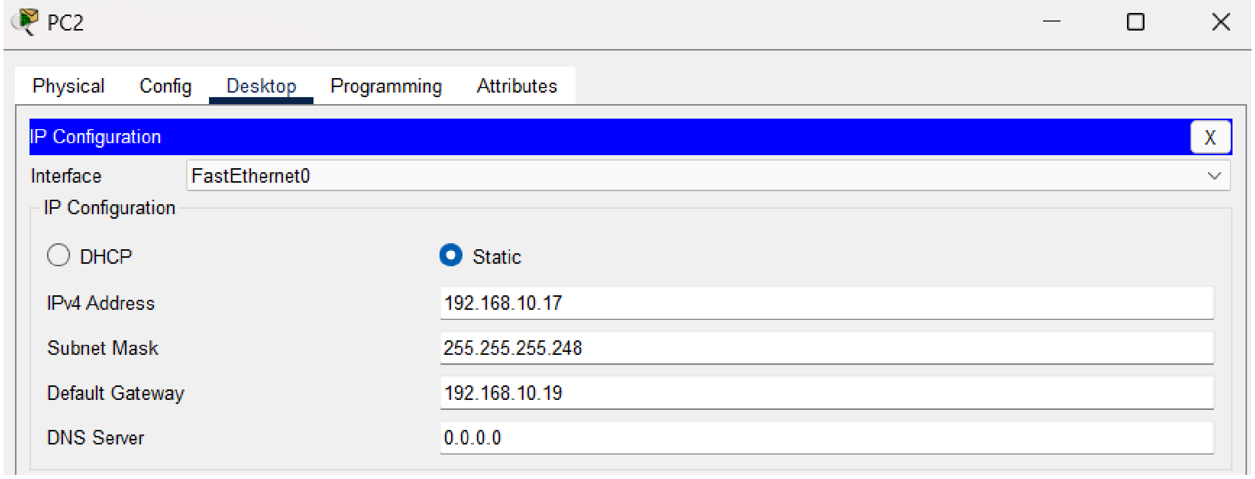
IP Configuration

Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address	192.168.10.9
Subnet Mask	255.255.255.248
Default Gateway	192.168.10.11
DNS Server	0.0.0.0



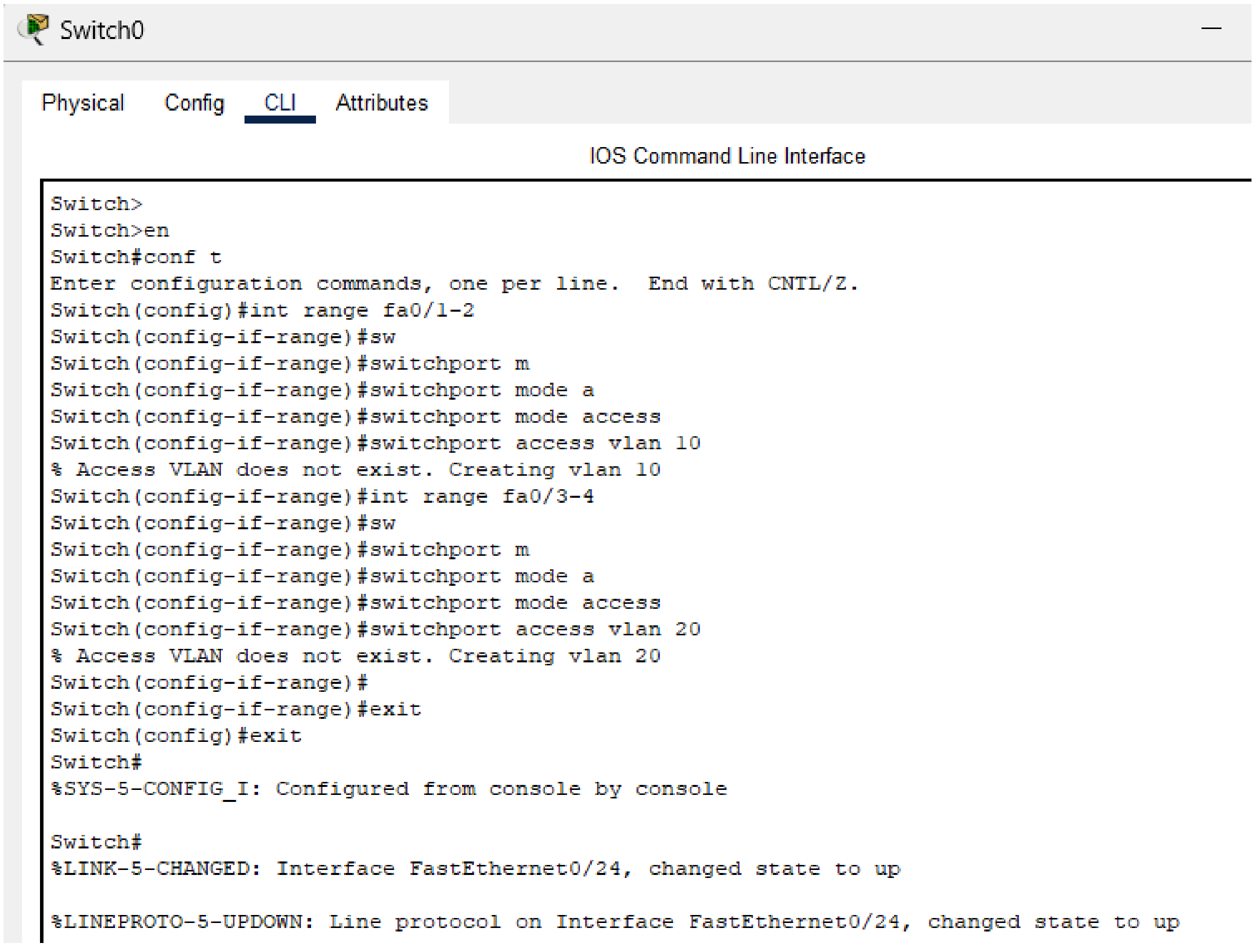
Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

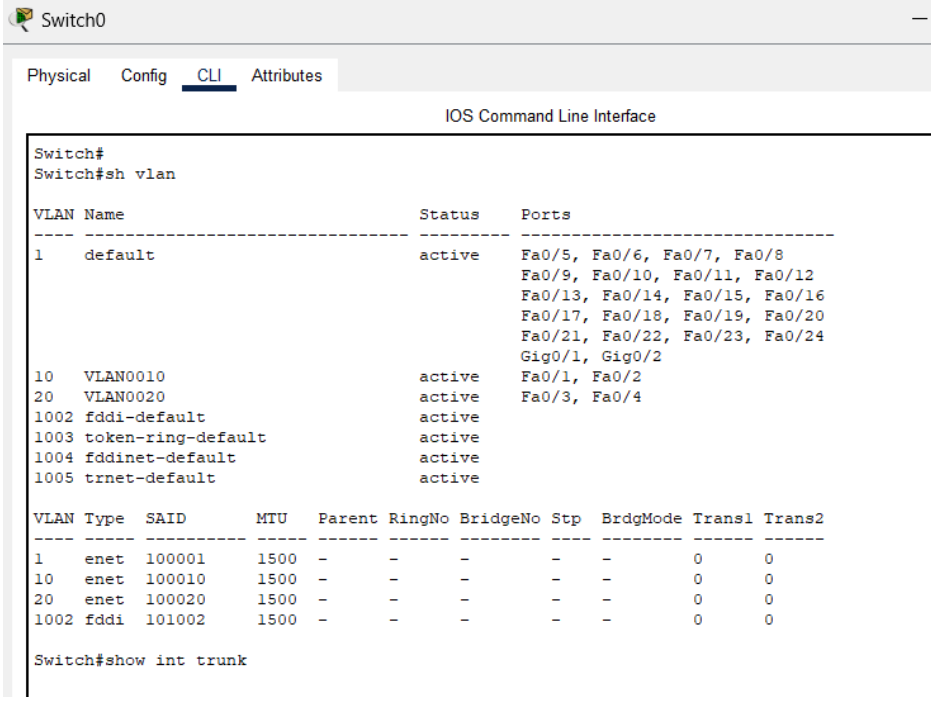
IPv4 Address	192.168.10.17
Subnet Mask	255.255.255.248
Default Gateway	192.168.10.19
DNS Server	0.0.0.0

In the switch, create the vlan, and assign the them range of interfaces each of them contains in Access mode.



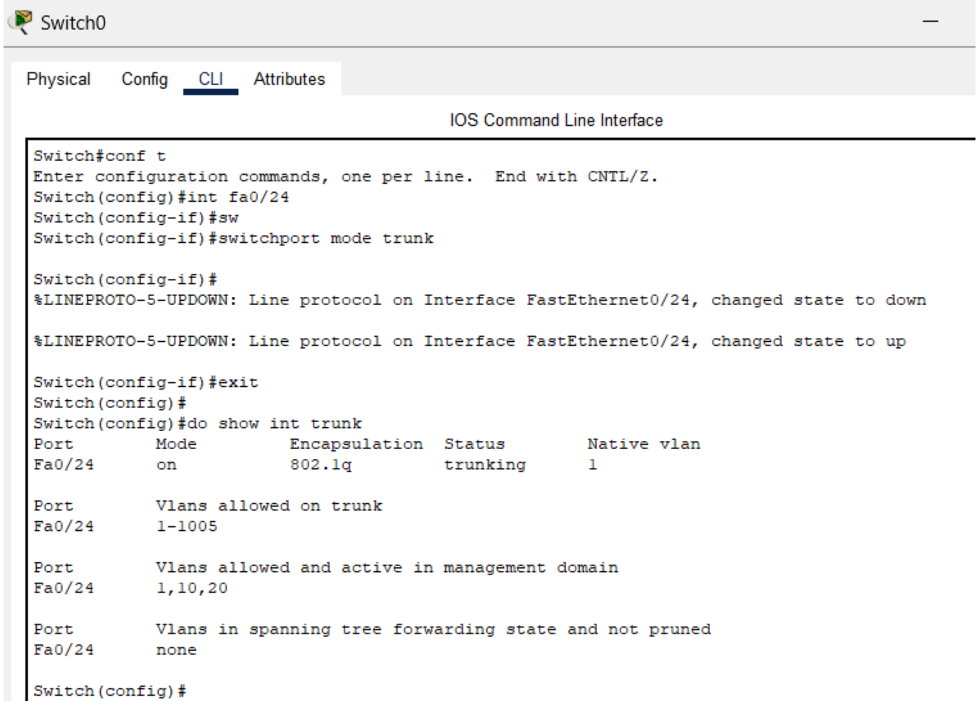
```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int range fa0/1-2
Switch(config-if-range)#sw
Switch(config-if-range)#switchport m
Switch(config-if-range)#switchport mode a
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
% Access VLAN does not exist. Creating vlan 10
Switch(config-if-range)#int range fa0/3-4
Switch(config-if-range)#sw
Switch(config-if-range)#switchport m
Switch(config-if-range)#switchport mode a
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 20
% Access VLAN does not exist. Creating vlan 20
Switch(config-if-range)#
Switch(config-if-range)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#
%LINK-5-CHANGED: Interface FastEthernet0/24, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24, changed state to up
```

Here we see the VLAN configuration :



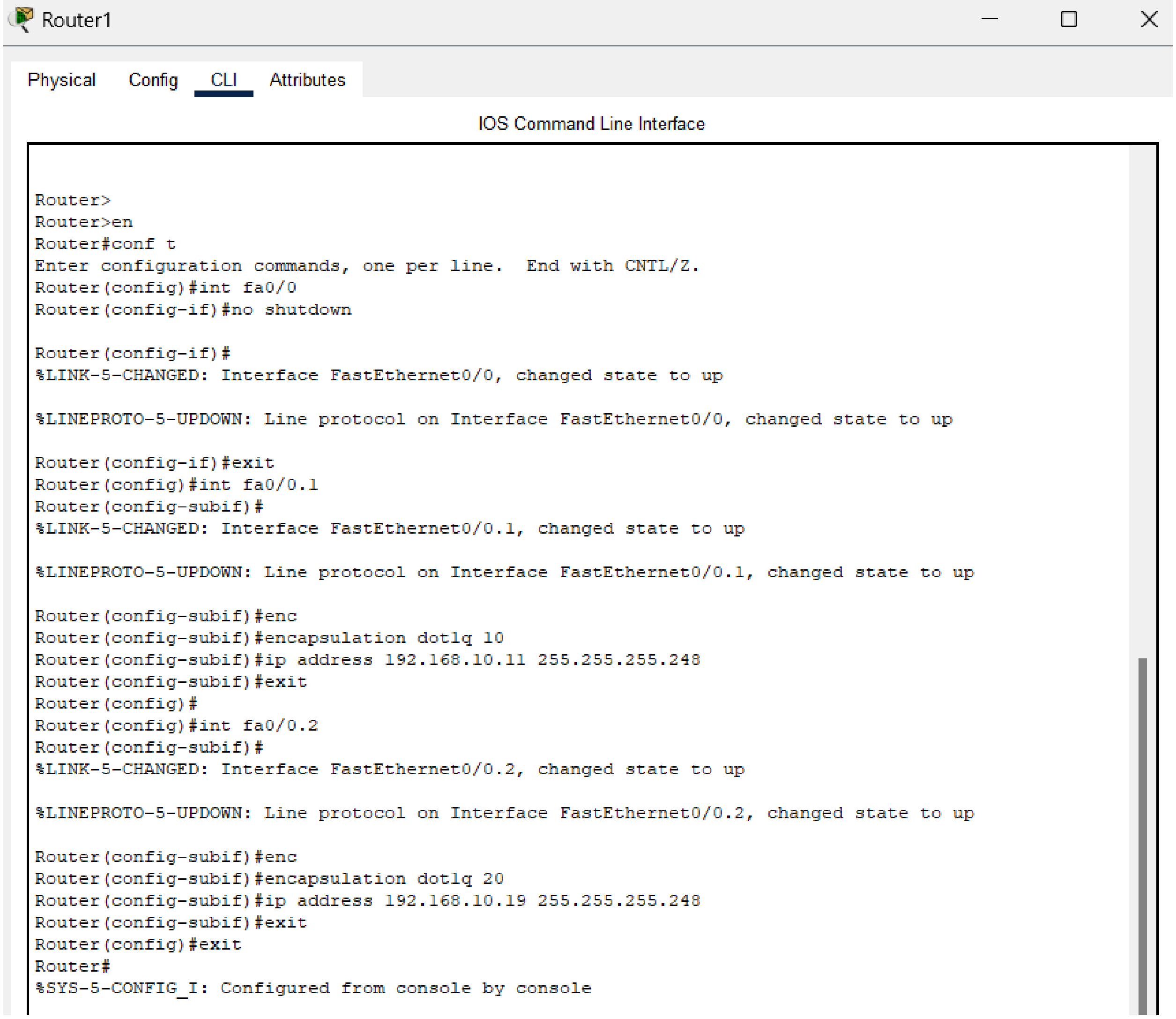
```
Switch#
Switch#sh vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    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
                                                Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                                Gig0/1, Gig0/2
10   VLAN0010                         active    Fa0/1, Fa0/2
20   VLAN0020                         active    Fa0/3, Fa0/4
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
20   enet  100020     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0

Switch#show int trunk
```

Also, create Router access with trunk mode for it communicate :



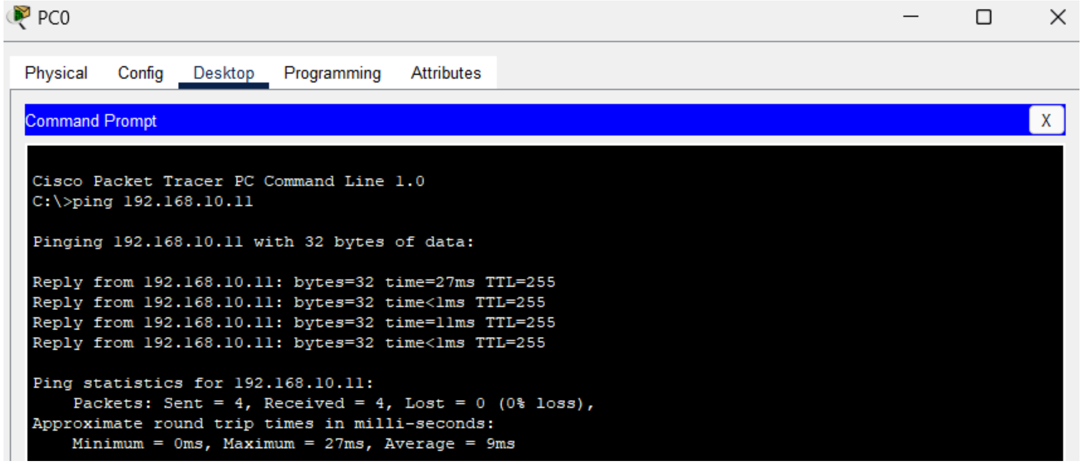
```
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/24
Switch(config-if)#sw
Switch(config-if)#switchport mode trunk

Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/24, changed state to up

Switch(config-if)#exit
Switch(config)#
Switch(config)#do show int trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/24      on           802.1q         trunking      1

Port        Vlans allowed on trunk
Fa0/24      1-1005

Port        Vlans allowed and active in management domain
Fa0/24      1,10,20

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/24      none

Switch(config)#
```

Lets use Router on a Stick method, to divide one single interface into two different logical interfaces and assign them ip addresses so the respective VLANs traffic can go through and be routed through it :

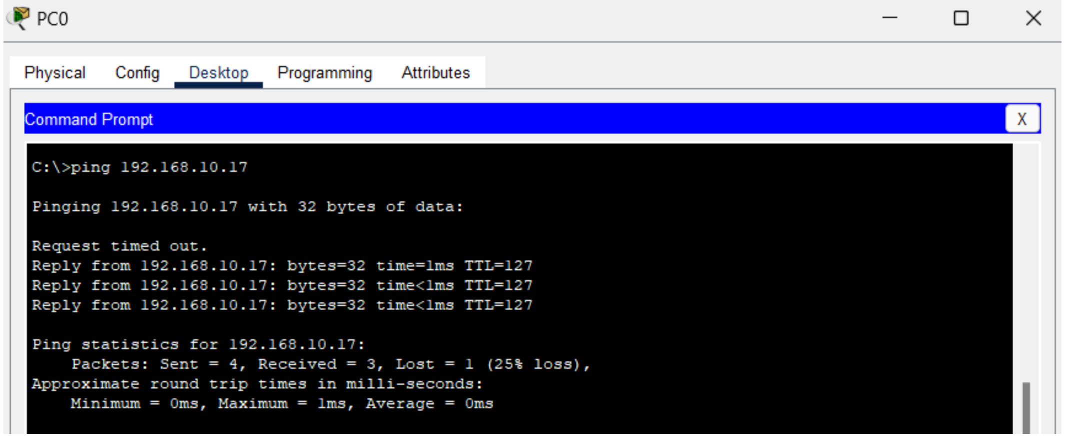


```
Router>
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

Router(config-if)#exit
Router(config)#int fa0/0.1
Router(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/0.1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.1, changed state to up

Router(config-subif)#enc
Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#ip address 192.168.10.11 255.255.255.248
Router(config-subif)#exit
Router(config)#
Router(config)#int fa0/0.2
Router(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/0.2, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.2, changed state to up

Router(config-subif)#enc
Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip address 192.168.10.19 255.255.255.248
Router(config-subif)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

On pinging the PCs of the other VLAN we are able to see it successful :

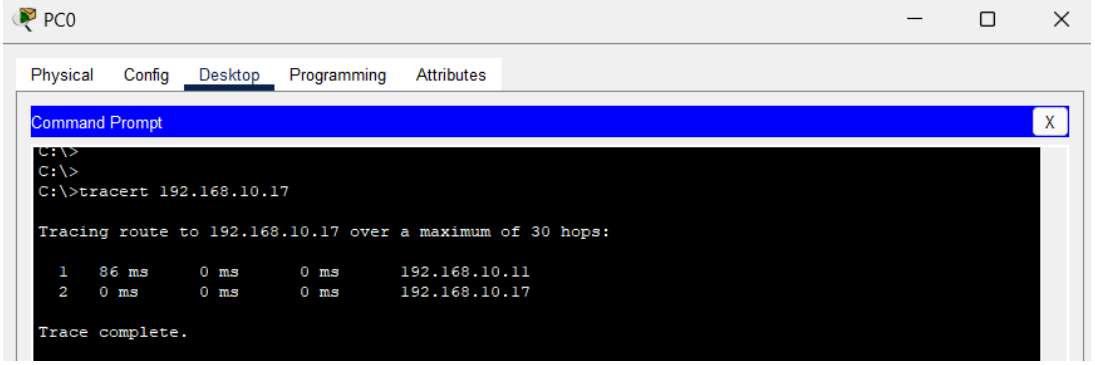


```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.11

Pinging 192.168.10.11 with 32 bytes of data:

Reply from 192.168.10.11: bytes=32 time=27ms TTL=255
Reply from 192.168.10.11: bytes=32 time<1ms TTL=255
Reply from 192.168.10.11: bytes=32 time=11ms TTL=255
Reply from 192.168.10.11: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.10.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 27ms, Average = 9ms
```



```
C:\>ping 192.168.10.17

Pinging 192.168.10.17 with 32 bytes of data:

Request timed out.
Reply from 192.168.10.17: bytes=32 time=1ms TTL=127
Reply from 192.168.10.17: bytes=32 time<1ms TTL=127
Reply from 192.168.10.17: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.10.17:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

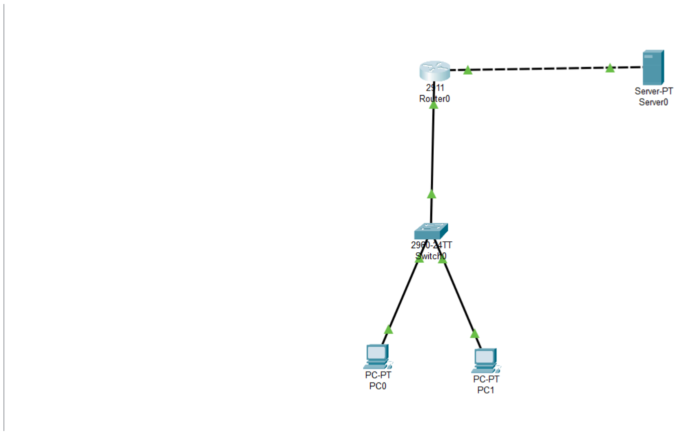
We also see the traffic is going through the one sub interface of the router and coming out of the other sub interface of the router :



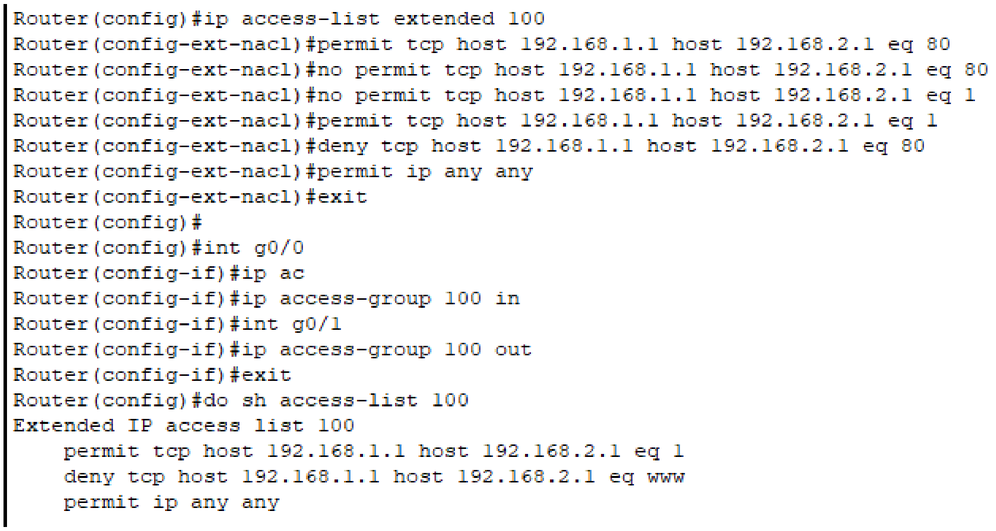
```
C:\>
C:\>
C:\>tracert 192.168.10.17

Tracing route to 192.168.10.17 over a maximum of 30 hops:

  1   86 ms     0 ms      0 ms      192.168.10.11
  2   0 ms      0 ms      0 ms      192.168.10.17

Trace complete.
```

11. Implement ACLs to restrict traffic based on source and destination ports. Test rules by simulating legitimate and unauthorized traffic.

Create the network like this in Packet Tracer (More detailed explanation is also given in the Q12)

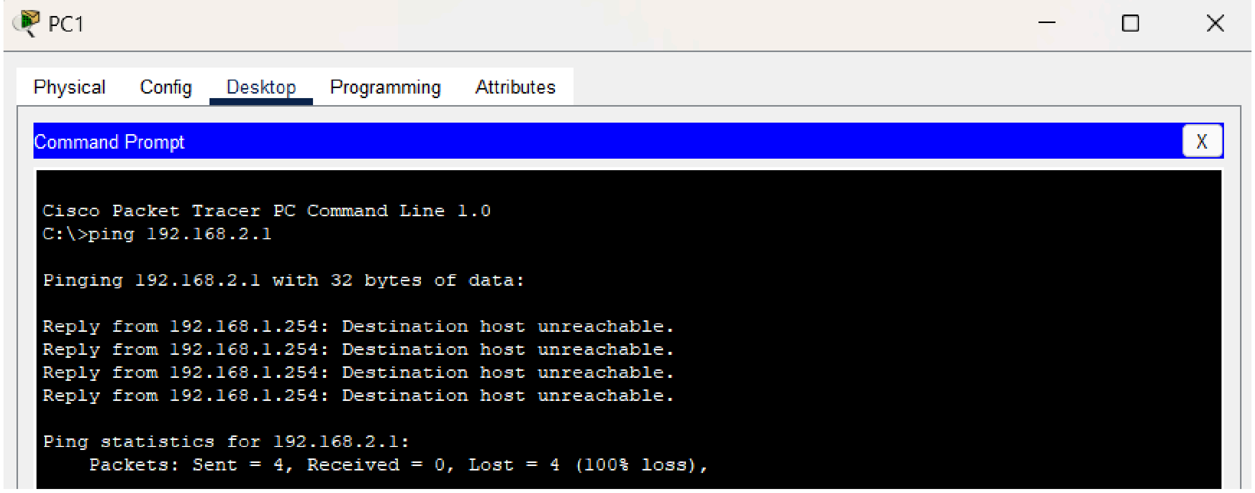


Configure the router to have extended ACL and use the ports in each rule to create the rules and attach them to the interface accordingly :

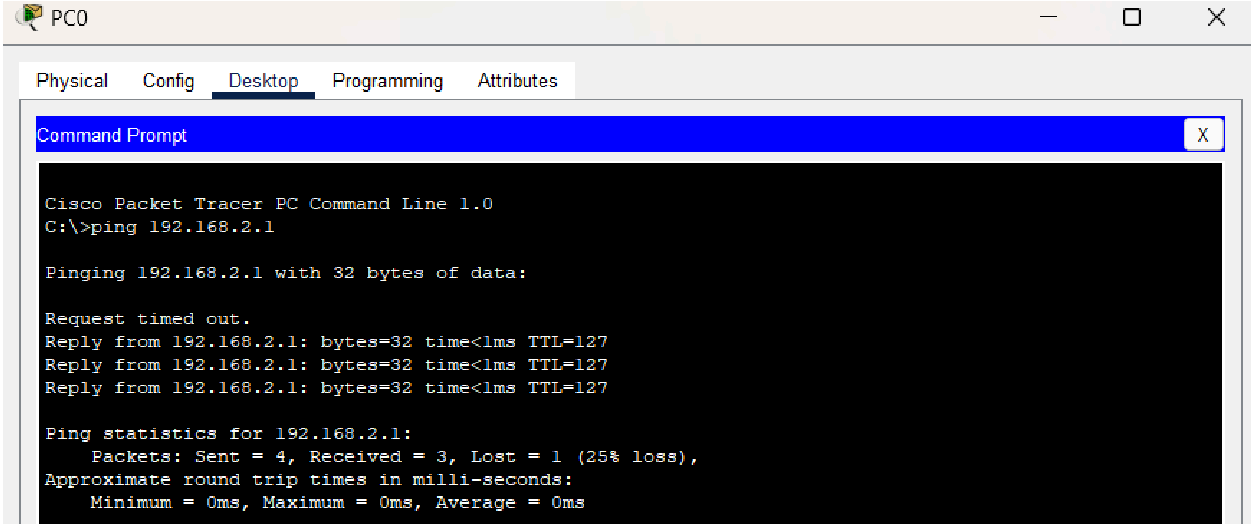


```
Router(config)#ip access-list extended 100
Router(config-ext-nacl)#permit tcp host 192.168.1.1 host 192.168.2.1 eq 80
Router(config-ext-nacl)#no permit tcp host 192.168.1.1 host 192.168.2.1 eq 80
Router(config-ext-nacl)#no permit tcp host 192.168.1.1 host 192.168.2.1 eq 1
Router(config-ext-nacl)#permit tcp host 192.168.1.1 host 192.168.2.1 eq 1
Router(config-ext-nacl)#deny tcp host 192.168.1.1 host 192.168.2.1 eq 80
Router(config-ext-nacl)#permit ip any any
Router(config-ext-nacl)#exit
Router(config)#
Router(config)#int g0/0
Router(config-if)#ip ac
Router(config-if)#ip access-group 100 in
Router(config-if)#int g0/1
Router(config-if)#ip access-group 100 out
Router(config-if)#exit
Router(config)#do sh access-list 100
Extended IP access list 100
    permit tcp host 192.168.1.1 host 192.168.2.1 eq 1
    deny tcp host 192.168.1.1 host 192.168.2.1 eq www
    permit ip any any
```

In the ACL Port rules we have made such that PC1 won’t be able to access the server while PC0 will be able to access the server and we see it here below :



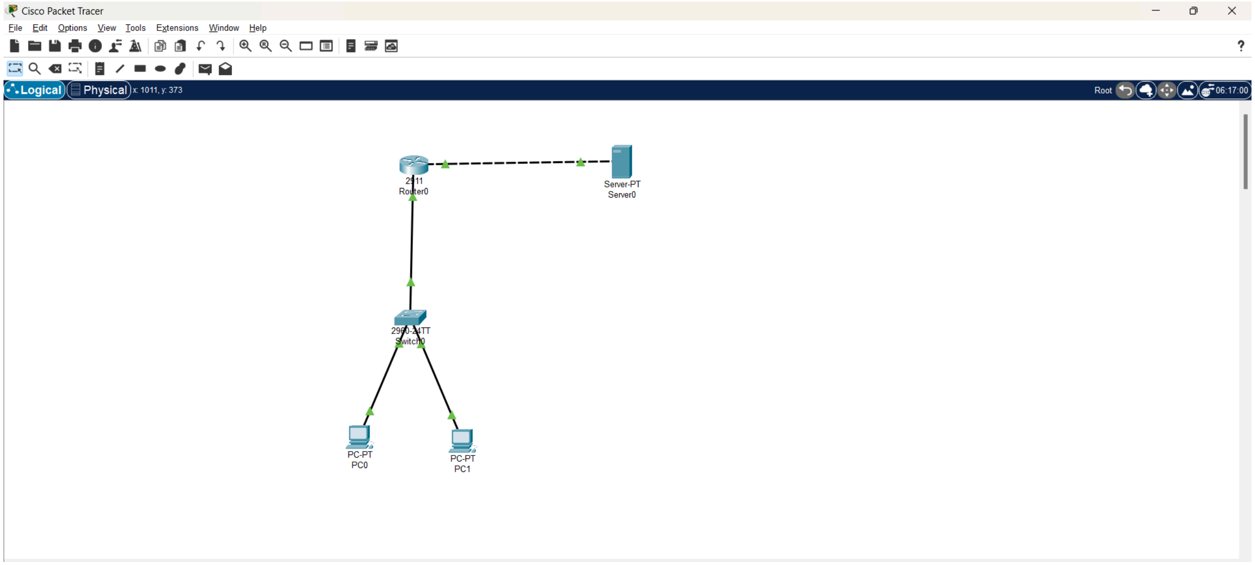
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

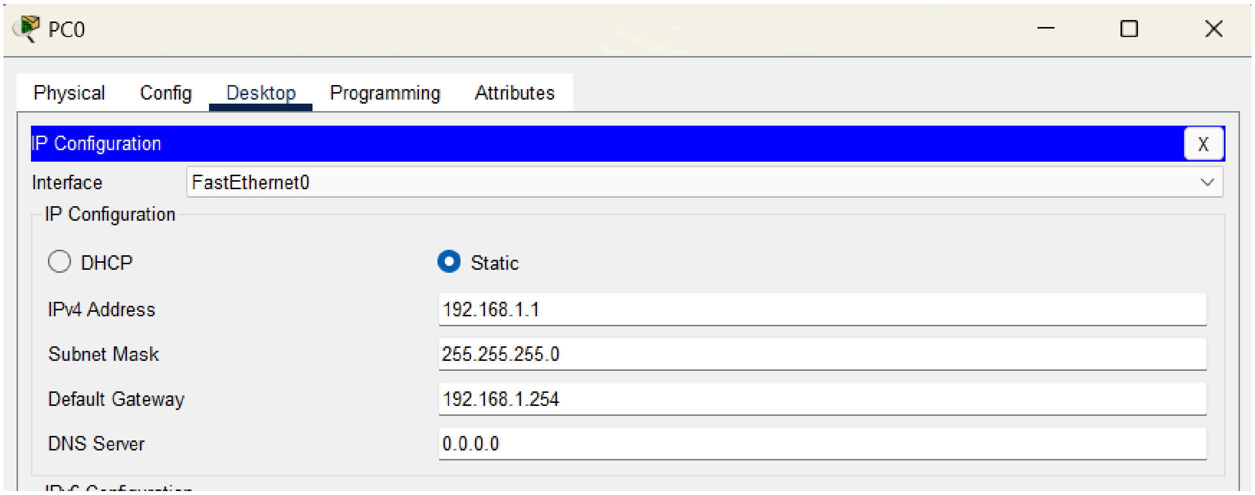


```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Request timed out.
Reply from 192.168.2.1: bytes=32 time<1ms TTL=127
Reply from 192.168.2.1: bytes=32 time<1ms TTL=127
Reply from 192.168.2.1: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

12. Configure a standard Access Control List (ACL) on a router to permit traffic from a specific IP range. Test connectivity to verify the ACL is working as intended.

Configure the Network accordingly and also assign IP address, subnet mask, and default gateway to the PCs and Server.





IP Configuration

Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

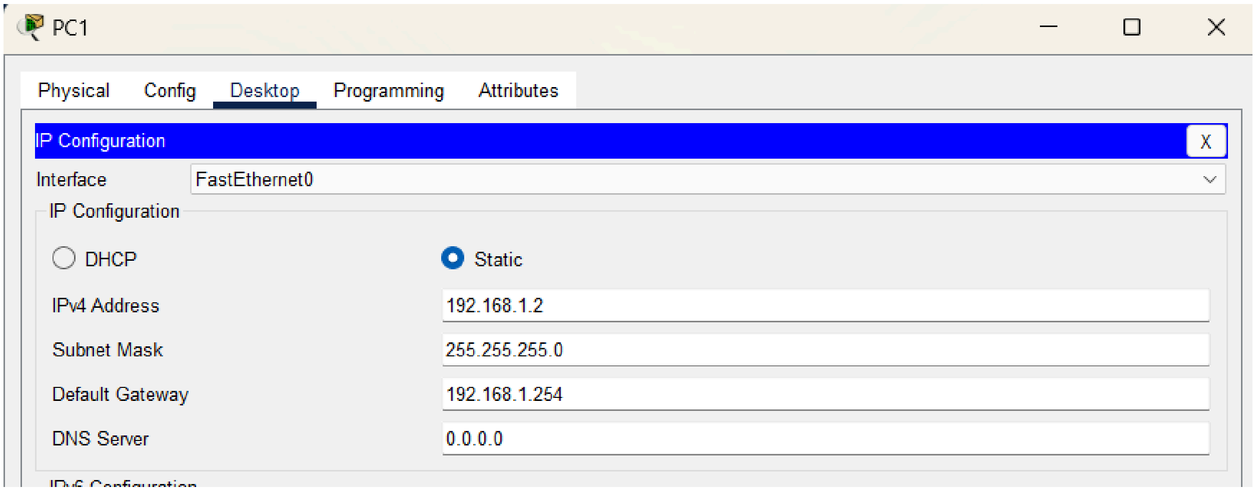
IPv4 Address: 192.168.1.1

Subnet Mask: 255.255.255.0

Default Gateway: 192.168.1.254

DNS Server: 0.0.0.0

IPv6 Configuration



IP Configuration

Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

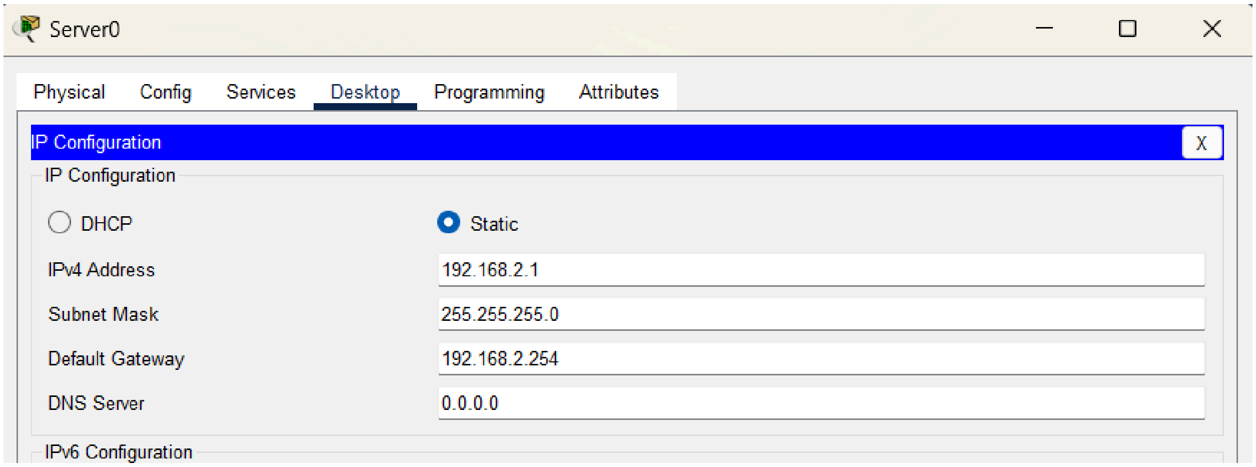
IPv4 Address: 192.168.1.2

Subnet Mask: 255.255.255.0

Default Gateway: 192.168.1.254

DNS Server: 0.0.0.0

IPv6 Configuration



IP Configuration

IP Configuration

○ DHCP ● Static

IPv4 Address: 192.168.2.1

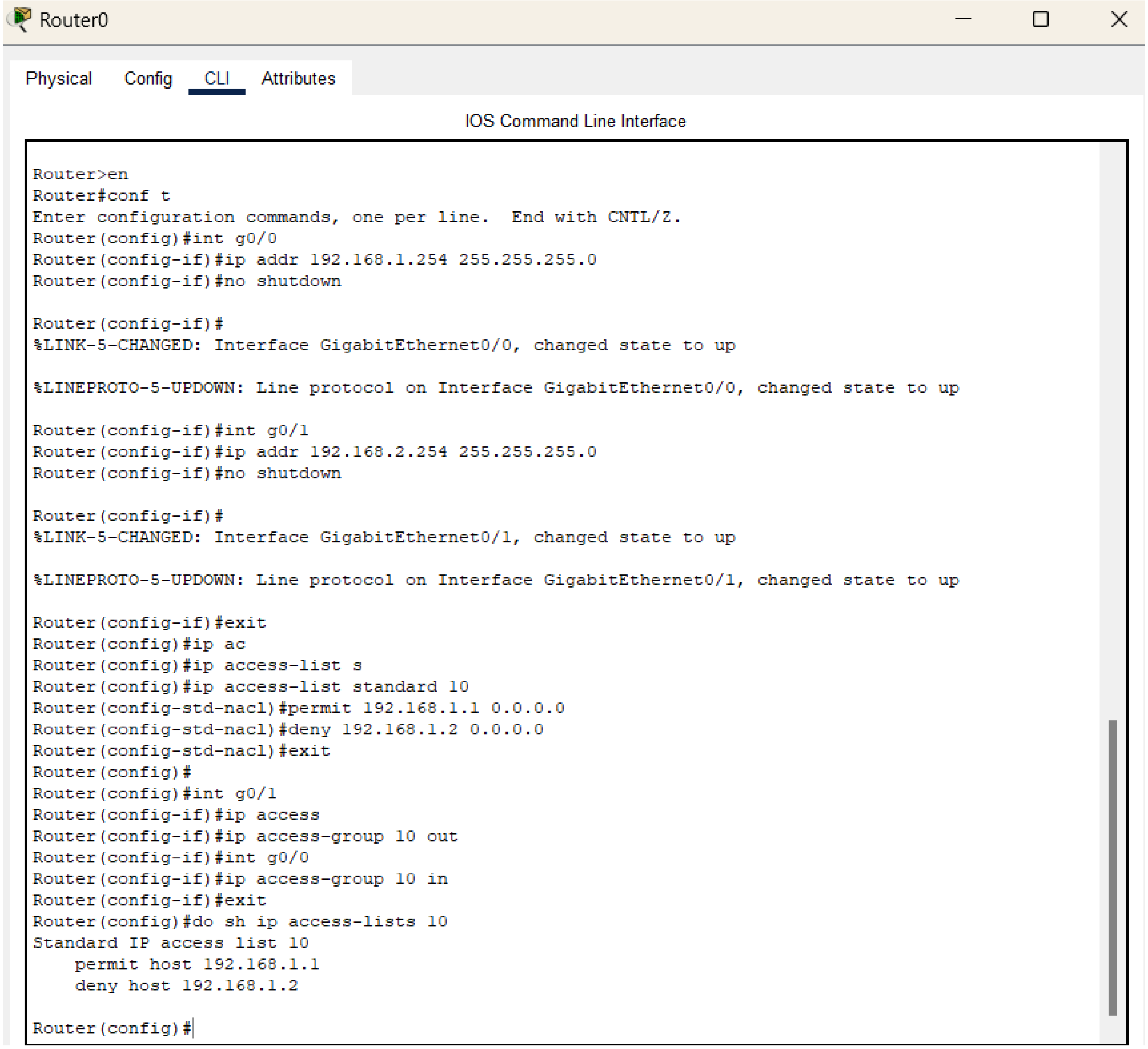
Subnet Mask: 255.255.255.0

Default Gateway: 192.168.2.254

DNS Server: 0.0.0.0

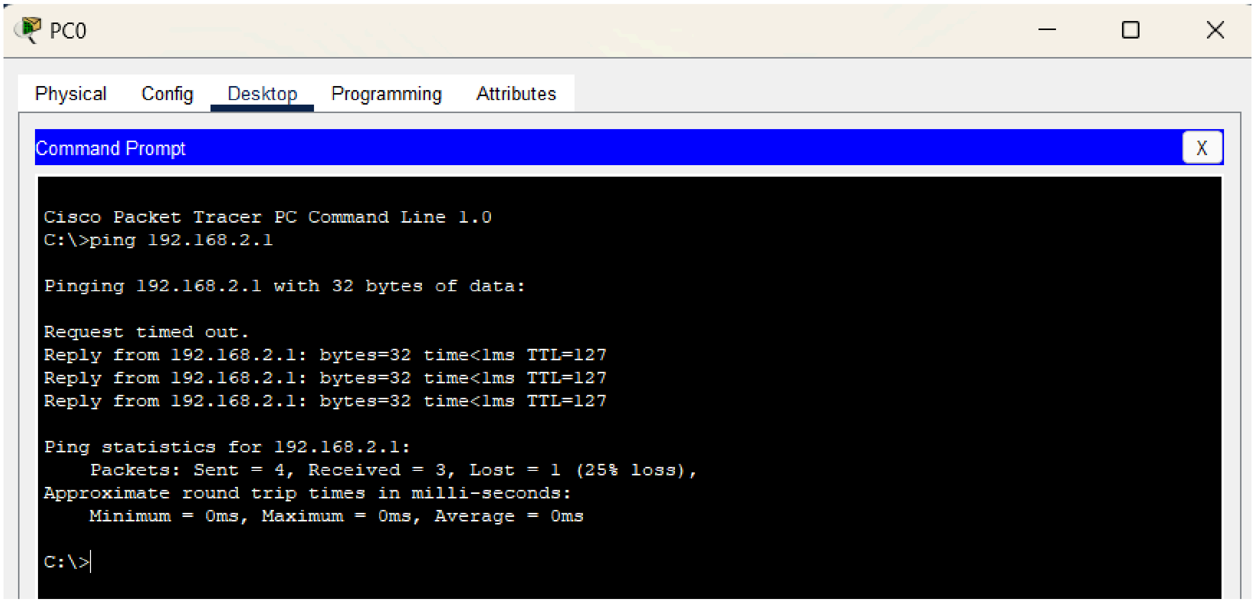
IPv6 Configuration

In the router, first assign the IP address of the interfaces, followed by the creation of access list standard and add permit and deny rules accordingly on which range of IPs you want to allow and not allow. Add the access list as inbound and outbound to the router interface accordingly :

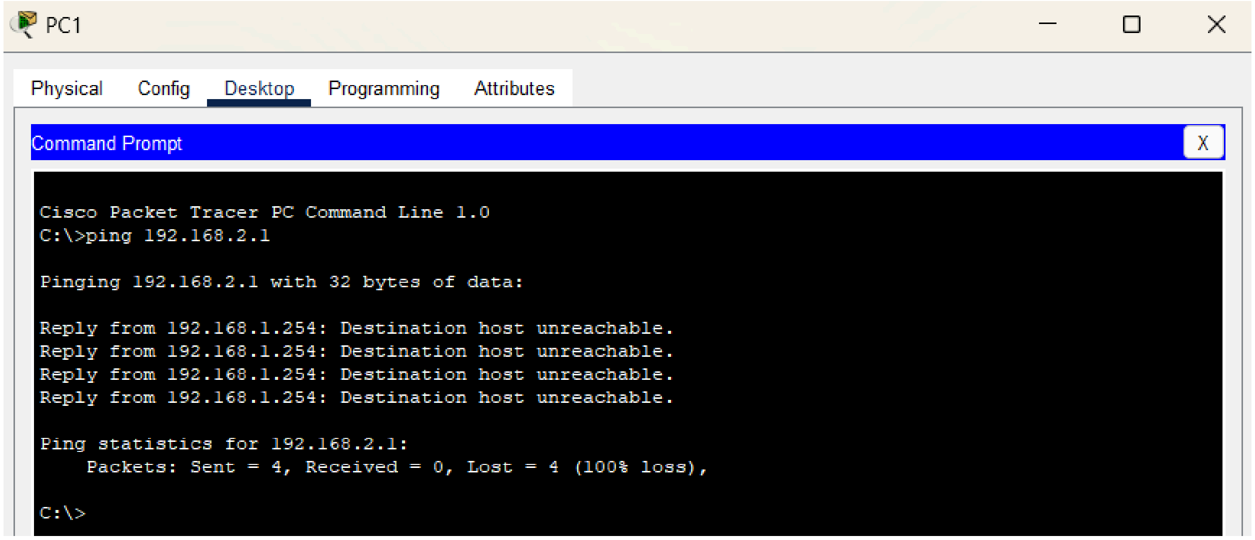


```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int g0/0
Router(config-if)#ip addr 192.168.1.254 255.255.255.0
Router(config-if)#no shutdown

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

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

Router(config-if)#int g0/1
Router(config-if)#ip addr 192.168.2.254 255.255.255.0
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)#ip ac
Router(config)#ip access-list s
Router(config)#ip access-list standard 10
Router(config-std-nacl)#permit 192.168.1.1 0.0.0.0
Router(config-std-nacl)#deny 192.168.1.2 0.0.0.0
Router(config-std-nacl)#exit
Router(config)#
Router(config)#int g0/1
Router(config-if)#ip access
Router(config-if)#ip access-group 10 out
Router(config-if)#int g0/0
Router(config-if)#ip access-group 10 in
Router(config-if)#exit
Router(config)#do sh ip access-lists 10
Standard IP access list 10
    permit host 192.168.1.1
    deny host 192.168.1.2

Router(config)#
```

Now on checking from the respective PCs, PC0 should able to ping to server while PC1 won’t be able to ping to the server.



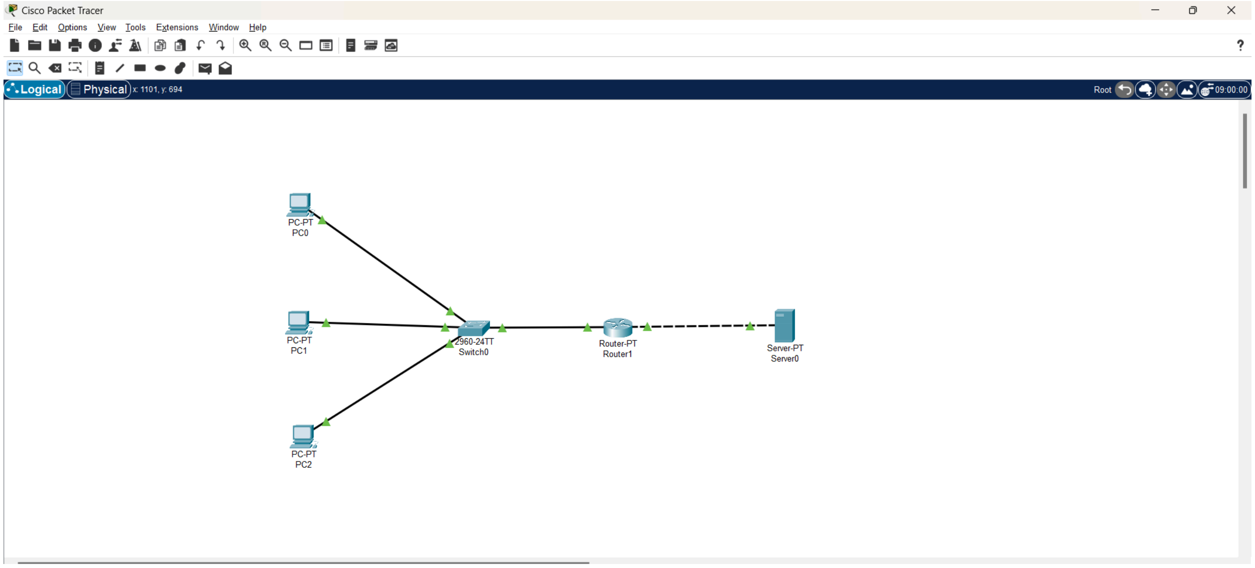
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Request timed out.
Reply from 192.168.2.1: bytes=32 time<1ms TTL=127
Reply from 192.168.2.1: bytes=32 time<1ms TTL=127
Reply from 192.168.2.1: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

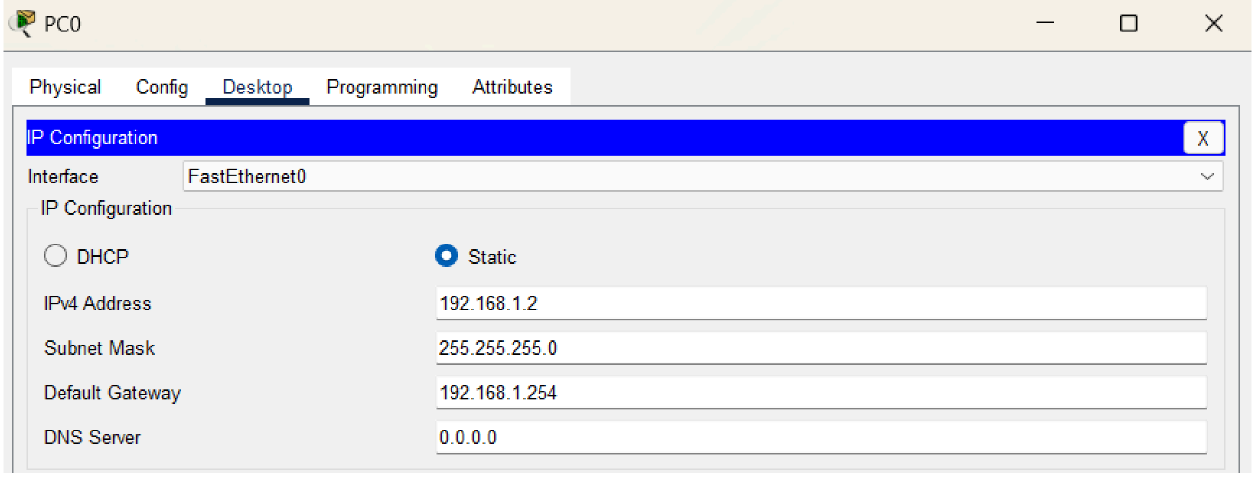


```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

13. Create an extended ACL to block specific applications, such as HTTP or FTP traffic. Test the ACL rules by attempting to access blocked services.

Creating this network in the Packet Tracer, and assign the IP address, subnet mask, default gateway accordingly to PCs and Server.





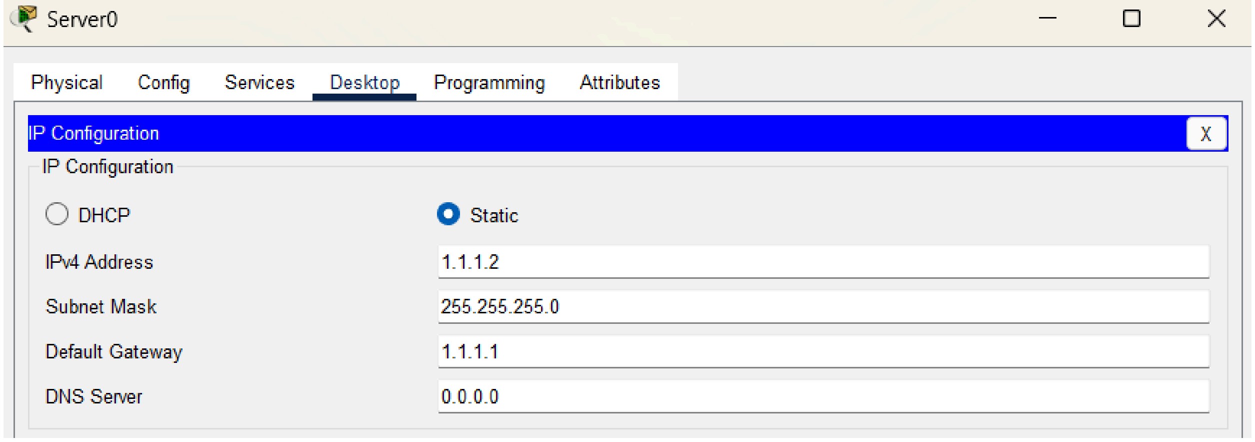
IP Configuration

Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address	192.168.1.2
Subnet Mask	255.255.255.0
Default Gateway	192.168.1.254
DNS Server	0.0.0.0



IP Configuration

IP Configuration

○ DHCP ● Static

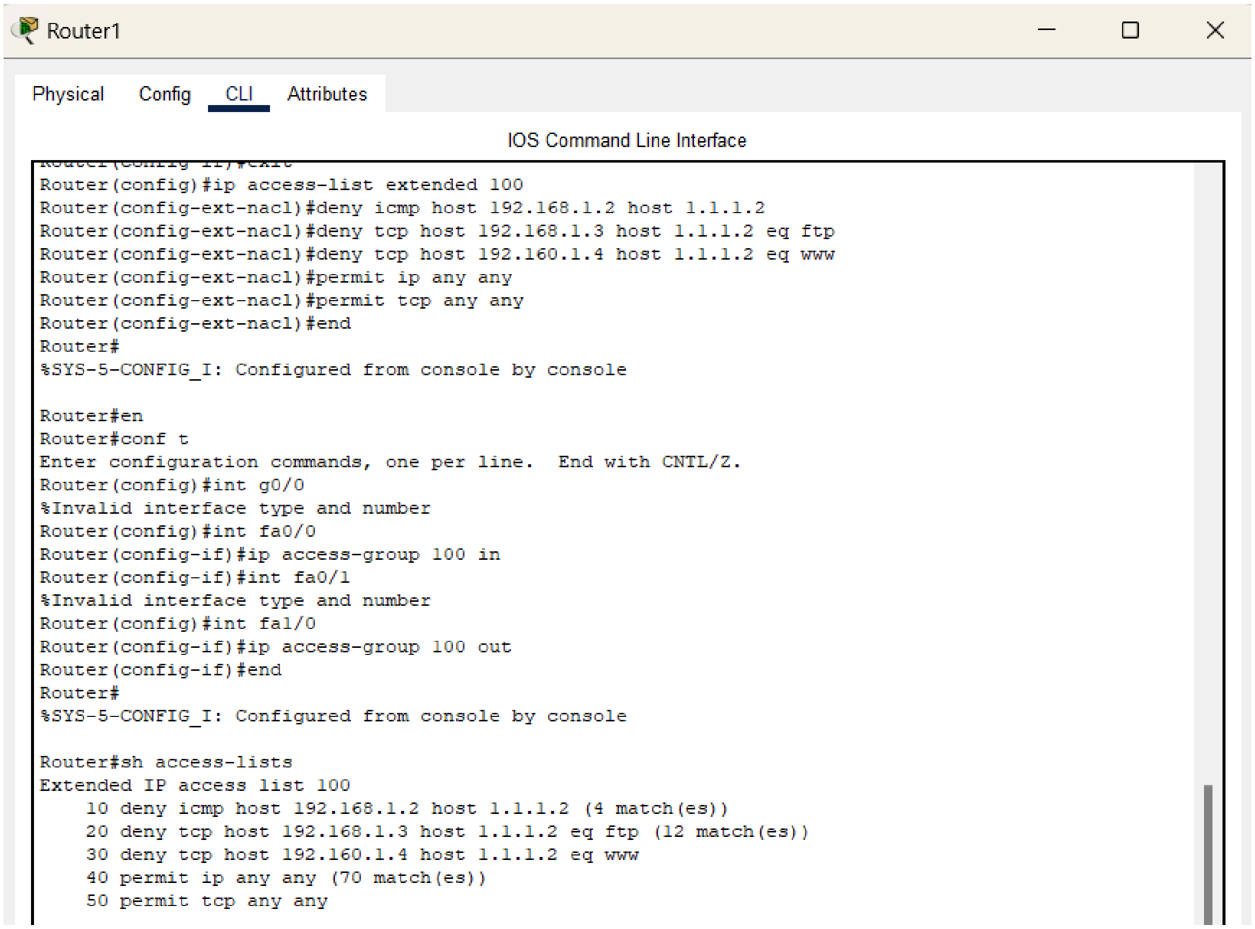
IPv4 Address 1.1.1.2

Subnet Mask 255.255.255.0

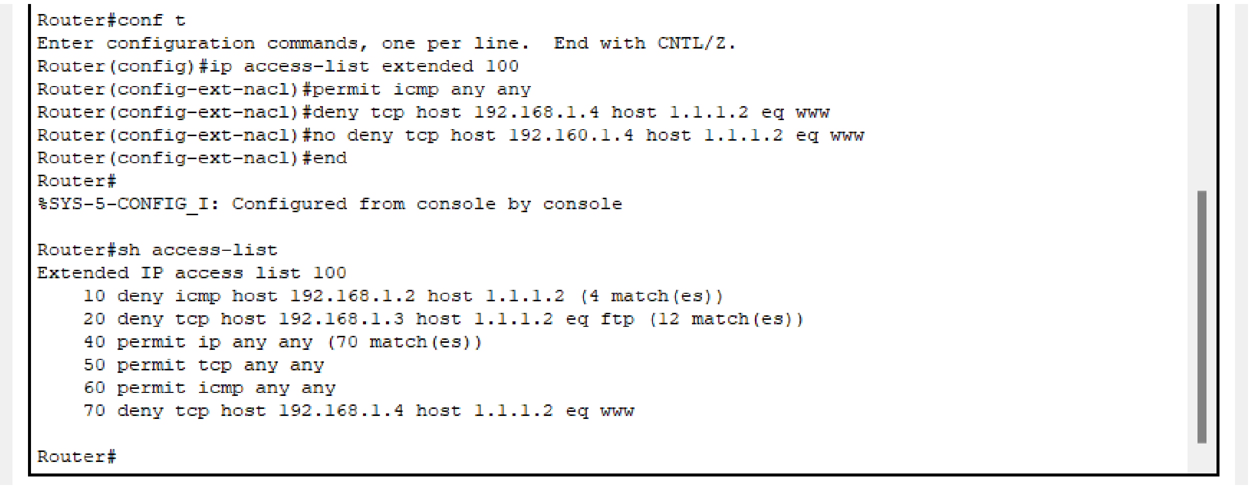
Default Gateway 1.1.1.1

DNS Server 0.0.0.0

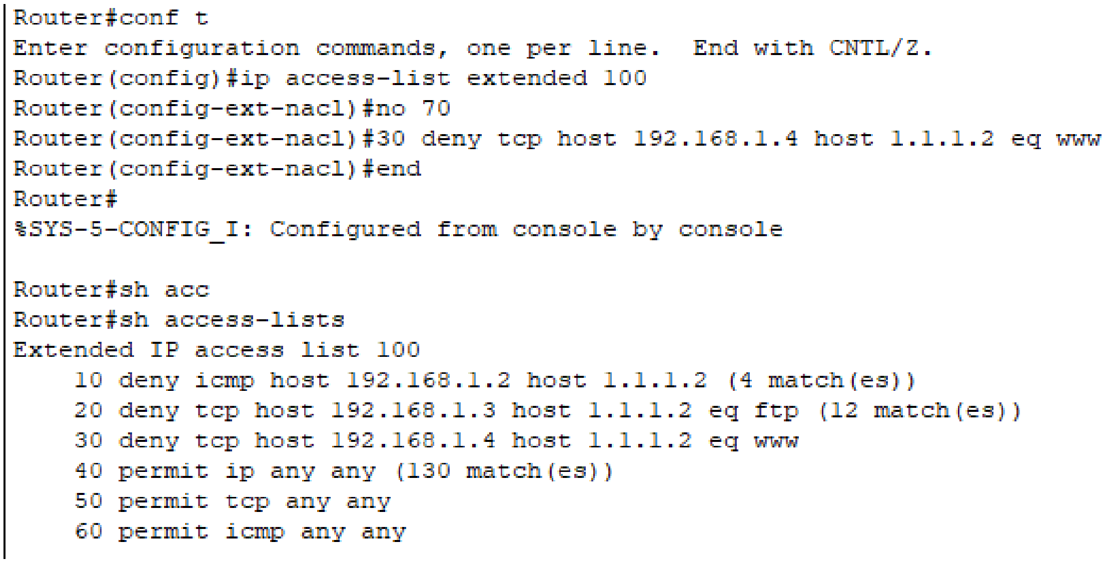
On the router, create the extended access list and add the permit and deny rules for the protocols accordingly and add them to the routers inbound and outbound interfaces respectively :



```
Router(config-if)#exit
Router(config)#ip access-list extended 100
Router(config-ext-nacl)#deny icmp host 192.168.1.2 host 1.1.1.2
Router(config-ext-nacl)#deny tcp host 192.168.1.3 host 1.1.1.2 eq ftp
Router(config-ext-nacl)#deny tcp host 192.160.1.4 host 1.1.1.2 eq www
Router(config-ext-nacl)#permit ip any any
Router(config-ext-nacl)#permit tcp any any
Router(config-ext-nacl)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int g0/0
%Invalid interface type and number
Router(config)#int fa0/0
Router(config-if)#ip access-group 100 in
Router(config-if)#int fa0/1
%Invalid interface type and number
Router(config)#int fa1/0
Router(config-if)#ip access-group 100 out
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#sh access-lists
Extended IP access list 100
    10 deny icmp host 192.168.1.2 host 1.1.1.2 (4 match(es))
    20 deny tcp host 192.168.1.3 host 1.1.1.2 eq ftp (12 match(es))
    30 deny tcp host 192.160.1.4 host 1.1.1.2 eq www
    40 permit ip any any (70 match(es))
    50 permit tcp any any
```



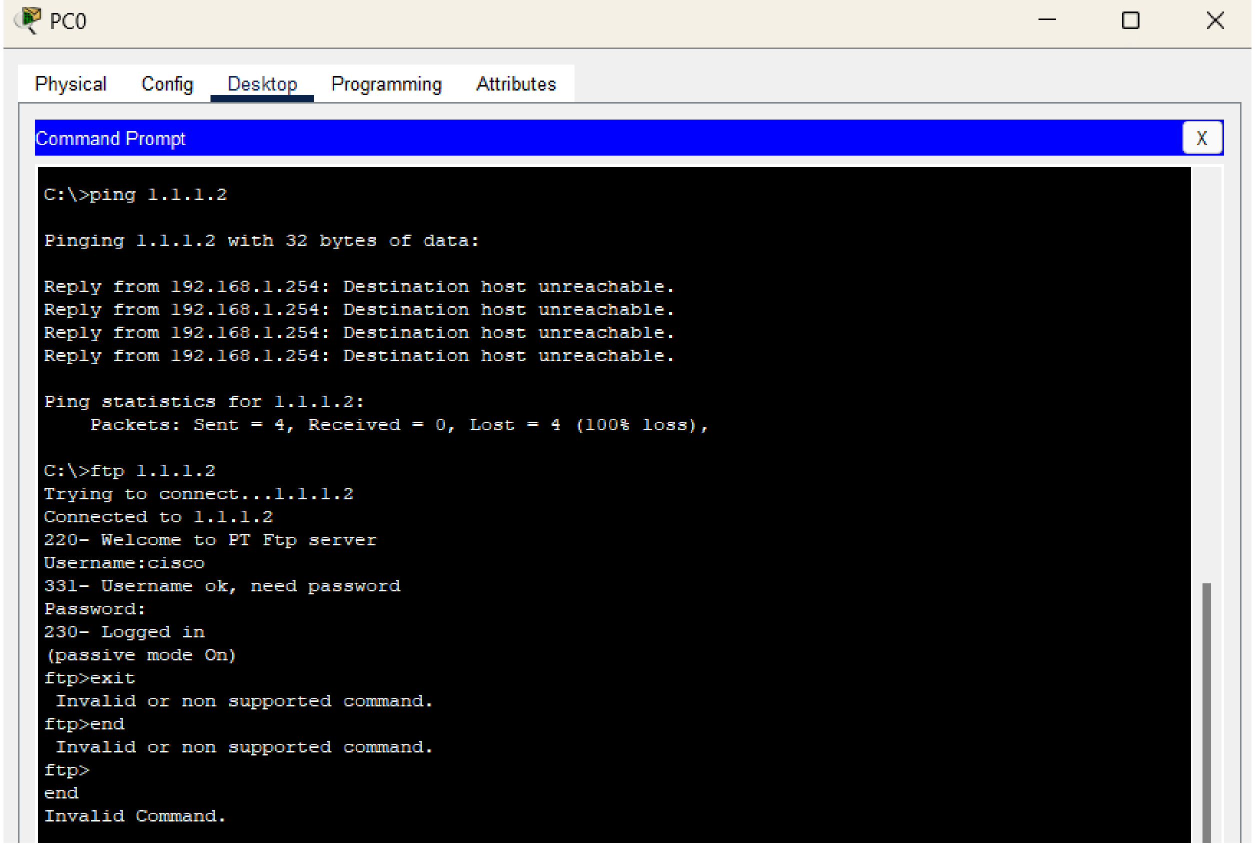
```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip access-list extended 100
Router(config-ext-nacl)#permit icmp any any
Router(config-ext-nacl)#deny tcp host 192.168.1.4 host 1.1.1.2 eq www
Router(config-ext-nacl)#no deny tcp host 192.160.1.4 host 1.1.1.2 eq www
Router(config-ext-nacl)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#sh access-list
Extended IP access list 100
    10 deny icmp host 192.168.1.2 host 1.1.1.2 (4 match(es))
    20 deny tcp host 192.168.1.3 host 1.1.1.2 eq ftp (12 match(es))
    40 permit ip any any (70 match(es))
    50 permit tcp any any
    60 permit icmp any any
    70 deny tcp host 192.168.1.4 host 1.1.1.2 eq www

Router#
```



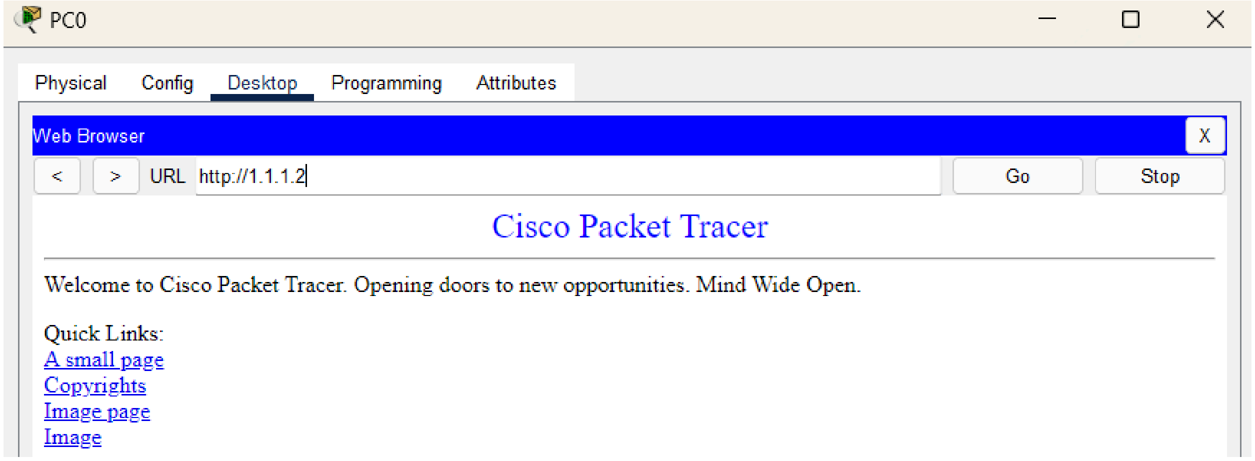
```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip access-list extended 100
Router(config-ext-nacl)#no 70
Router(config-ext-nacl)#30 deny tcp host 192.168.1.4 host 1.1.1.2 eq www
Router(config-ext-nacl)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#sh acc
Router#sh access-lists
Extended IP access list 100
    10 deny icmp host 192.168.1.2 host 1.1.1.2 (4 match(es))
    20 deny tcp host 192.168.1.3 host 1.1.1.2 eq ftp (12 match(es))
    30 deny tcp host 192.168.1.4 host 1.1.1.2 eq www
    40 permit ip any any (130 match(es))
    50 permit tcp any any
    60 permit icmp any any
```

Make sure to have the deny rules which are more specific in the first else all the traffic will go to the permit rules and deny rules will be ignored. The number is the priority order of the access list.

Now on checking the respective PCs, we will find that PC0 won’t be able to ping, PC1 won’t be able to access FTP and PC2 won’t be able to view the Website (HTTP).



```
C:\>ping 1.1.1.2

Pinging 1.1.1.2 with 32 bytes of data:

Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.
Reply from 192.168.1.254: Destination host unreachable.

Ping statistics for 1.1.1.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ftp 1.1.1.2
Trying to connect...1.1.1.2
Connected to 1.1.1.2
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>exit
 Invalid or non supported command.
ftp>end
 Invalid or non supported command.
ftp>
end
Invalid Command.
```



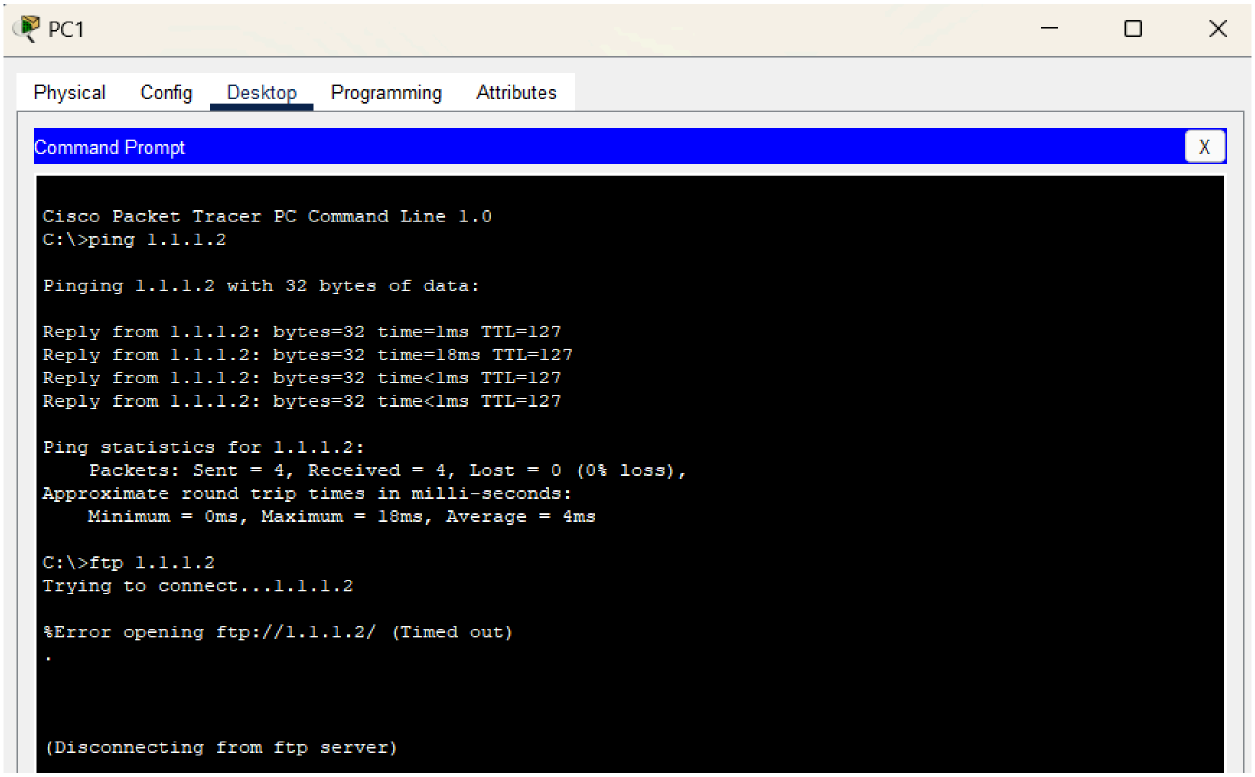
URL http://1.1.1.2

Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

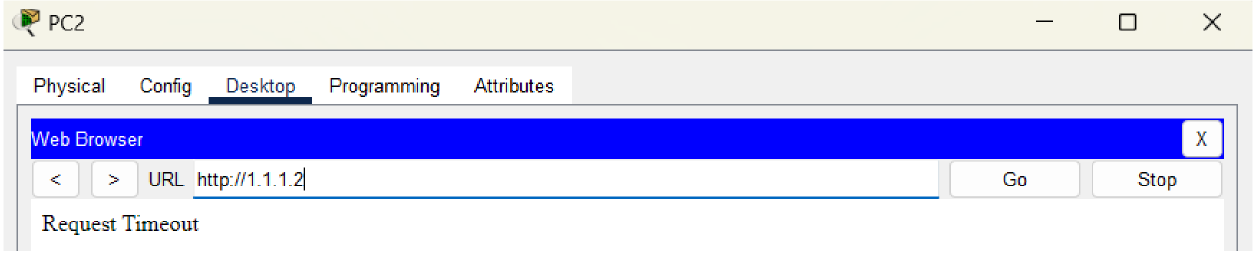
Quick Links:
A small page
Copyrights
Image page
Image

PC1 not able to use FTP :



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 1.1.1.2

Pinging 1.1.1.2 with 32 bytes of data:

Reply from 1.1.1.2: bytes=32 time=1ms TTL=127
Reply from 1.1.1.2: bytes=32 time=18ms TTL=127
Reply from 1.1.1.2: bytes=32 time<1ms TTL=127
Reply from 1.1.1.2: bytes=32 time<1ms TTL=127

Ping statistics for 1.1.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 18ms, Average = 4ms

C:\>ftp 1.1.1.2
Trying to connect...1.1.1.2

%Error opening ftp://1.1.1.2/ (Timed out)
.



(Disconnecting from ftp server)
```

PC2 not able to view the website (HTTP):



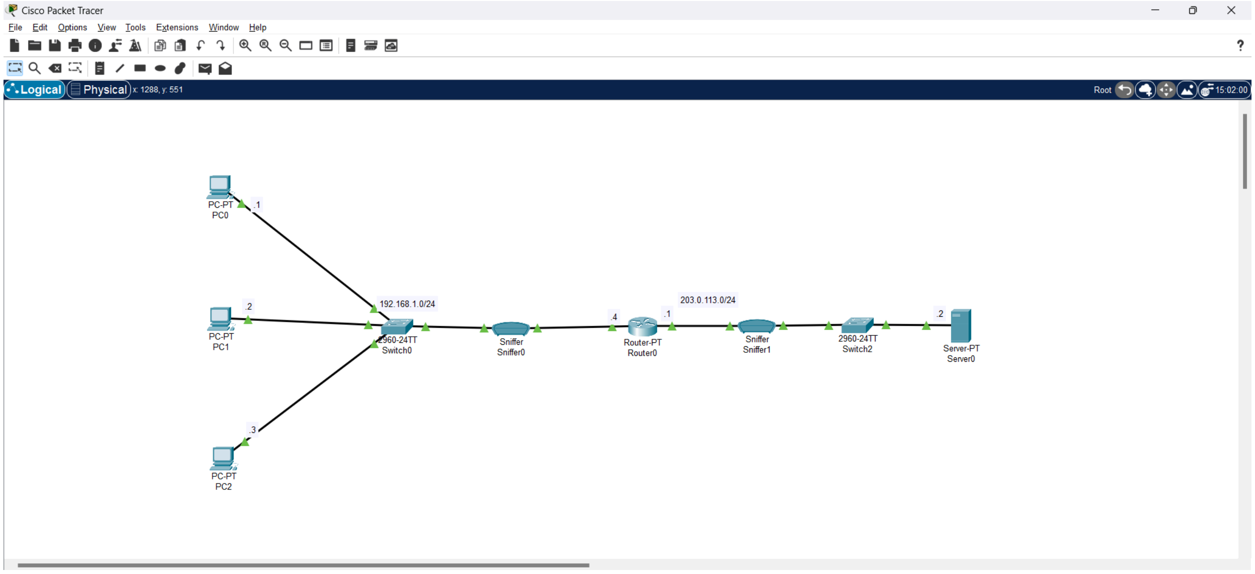
URL http://1.1.1.2

Request Timeout

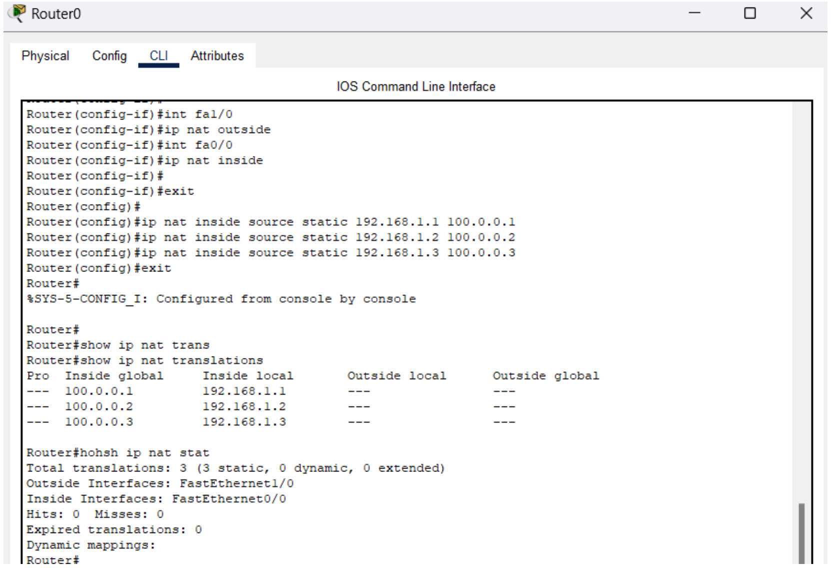
14. Try Static NAT, Dynamic NAT and PAT to translate IPs.

Static NAT :

Configure the network topology in Cisco Packet Tracer :

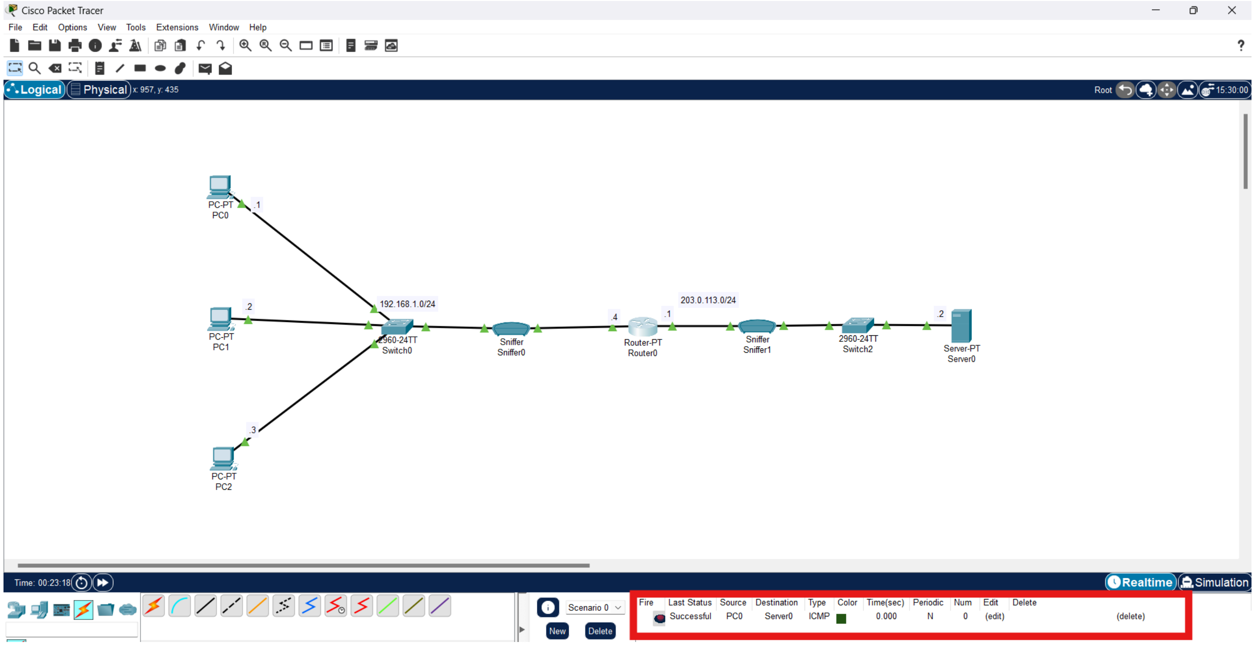


Configure the PC side (FastEthernet0/0) as Inside of NAT and Server Side (FastEthernet1/0) as Outside of NAT, followed by which add the static NAT source IP for each PC IP which will be converted to its NAT IP, then you will be able to see the NAT translations :



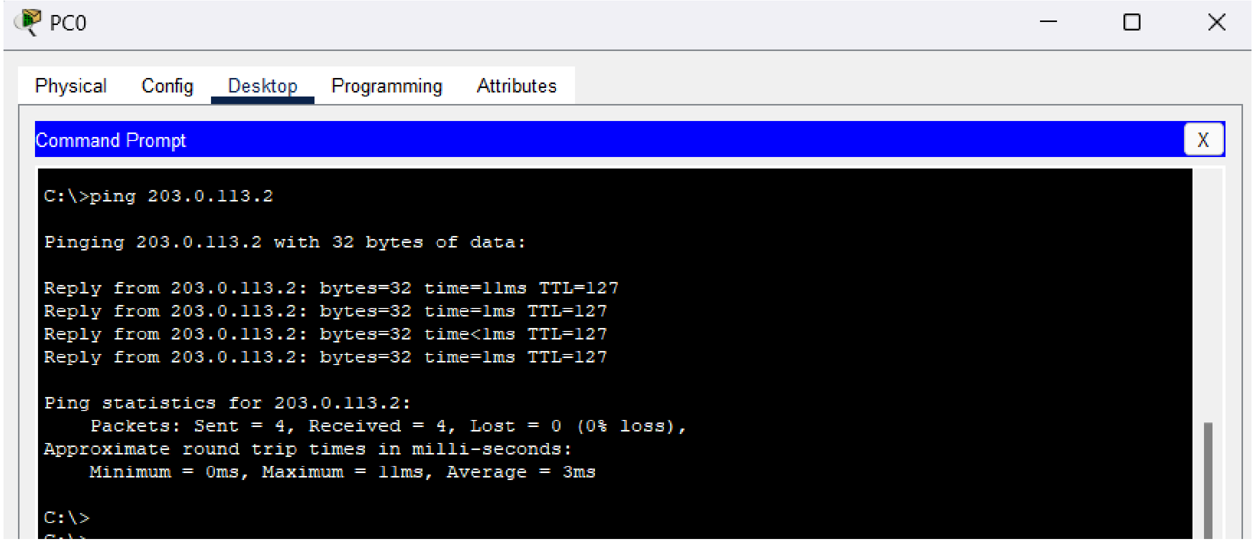
```
Router(config-if)#int fa1/0
Router(config-if)#ip nat outside
Router(config-if)#int fa0/0
Router(config-if)#ip nat inside
Router(config-if)#
Router(config-if)#exit
Router(config)#
Router(config)#ip nat inside source static 192.168.1.1 100.0.0.1
Router(config)#ip nat inside source static 192.168.1.2 100.0.0.2
Router(config)#ip nat inside source static 192.168.1.3 100.0.0.3
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
Router#show ip nat trans
Router#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
---  100.0.0.1         192.168.1.1        ---                ---
---  100.0.0.2         192.168.1.2        ---                ---
---  100.0.0.3         192.168.1.3        ---                ---

Router#hohsh ip nat stat
Total translations: 3 (3 static, 0 dynamic, 0 extended)
Outside Interfaces: FastEthernet1/0
Inside Interfaces: FastEthernet0/0
Hits: 0  Misses: 0
Expired translations: 0
Dynamic mappings:
Router#
```

Let's check using PDU from PC1 to server and its successfully working :



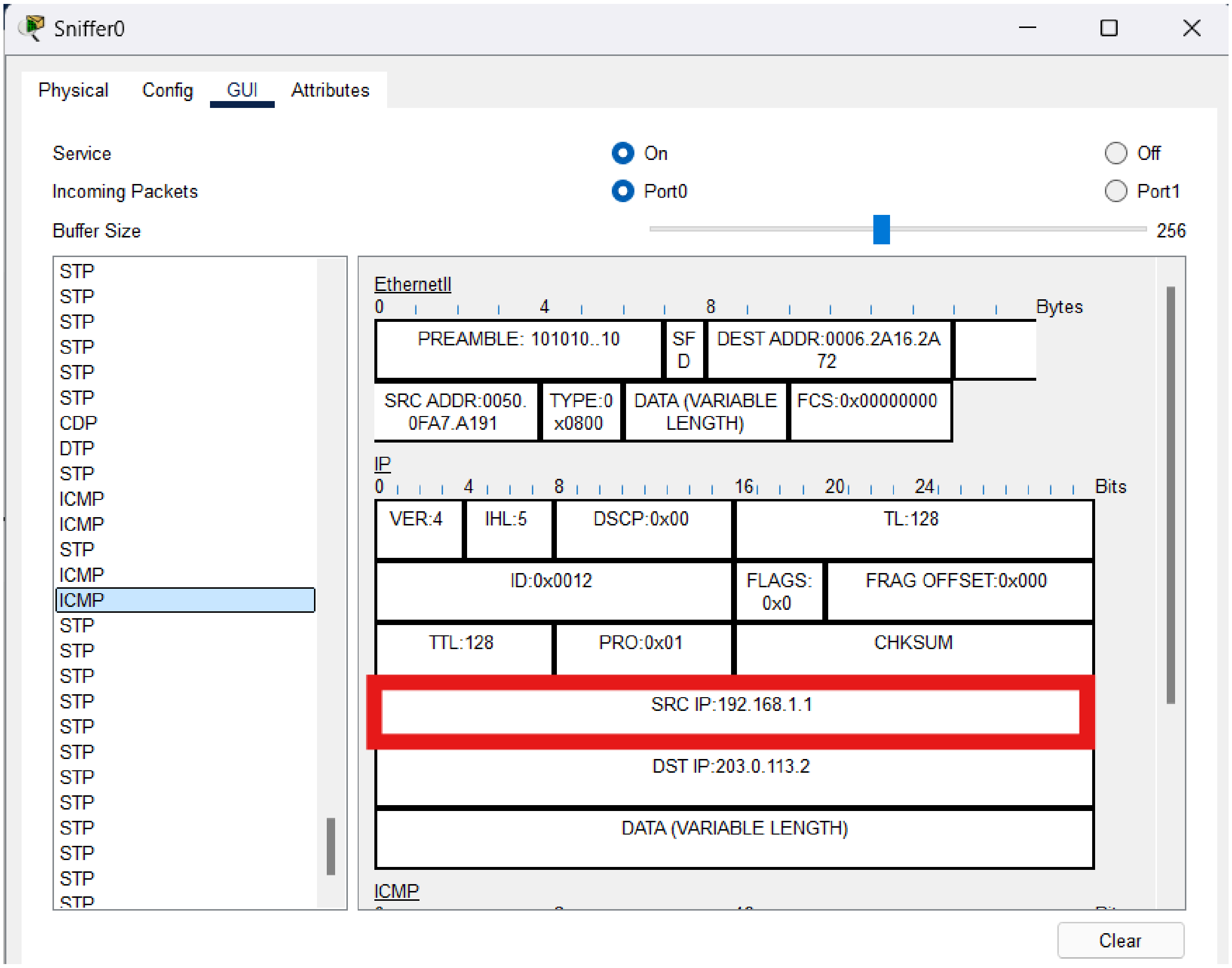
Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
●	Successful	PC0	Server0	ICMP	🟩	0.000	N	0	(edit)	(delete)

Pinging from PC0 to server also we are able to see the successful connection with Server :



```
C:\>ping 203.0.113.2

Pinging 203.0.113.2 with 32 bytes of data:

Reply from 203.0.113.2: bytes=32 time=11ms TTL=127
Reply from 203.0.113.2: bytes=32 time=1ms TTL=127
Reply from 203.0.113.2: bytes=32 time<1ms TTL=127
Reply from 203.0.113.2: bytes=32 time=1ms TTL=127

Ping statistics for 203.0.113.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 3ms

C:\>
C:\>
```

Since we don’t have Wireshark in Cisco Packet Tracer, as it is a simulation tool by itself, we can use the Sniffer device to view the Packets that are sent through the network and we see the ICMP packet in the NAT Inside, that the source IP is same as the PC’s IP :



Service ● On ○ Off

Incoming Packets ● Port0 ○ Port1

Buffer Size 256

```
STP
STP
STP
STP
STP
STP
CDP
DTP
STP
ICMP
ICMP
STP
ICMP
ICMP
STP
STP
STP
STP
STP
STP
STP
STP
STP
STP
STP
STP
```

EthernetII

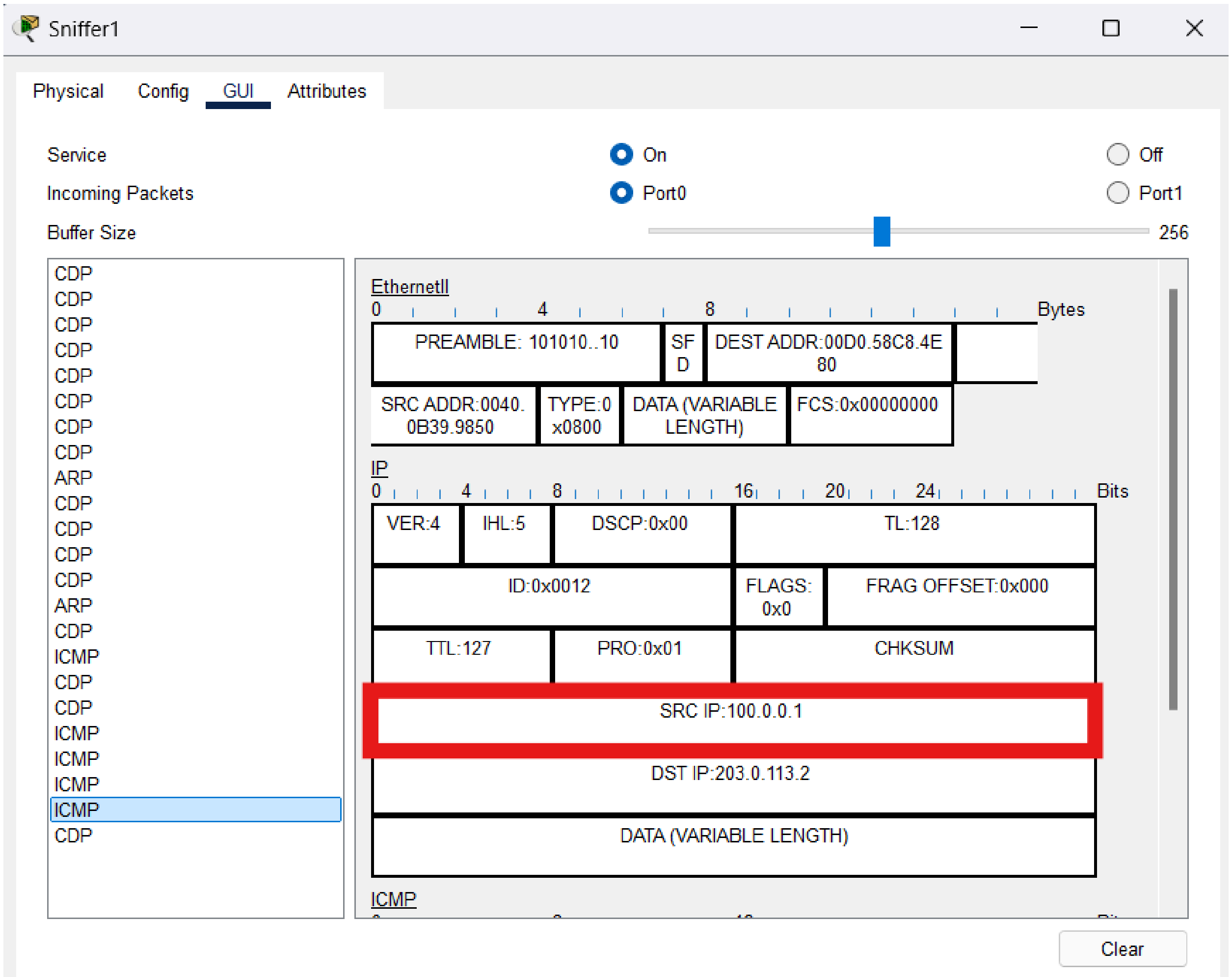
PREAMBLE: 101010..10	SFD	DEST ADDR:0006.2A16.2A72		
SRC ADDR:0050.0FA7.A191	TYPE:0x0800	DATA (VARIABLE LENGTH)	FCS:0x00000000	

IP

VER:4	IHL:5	DSCP:0x00	TL:128
ID:0x0012		FLAGS: 0x0	FRAG OFFSET:0x000
TTL:128	PRO:0x01	CHKSUM	
SRC IP:192.168.1.1			
DST IP:203.0.113.2			
DATA (VARIABLE LENGTH)			

ICMP

Similarly, in the NAT outside, we are able to see that the Source IP has successfully changed by the NAT in the router as configured by the static NAT using the CLI, from the PC’s IP to 100.0.0.1 which was configured. Hence the NAT translation is successful :



Service ● On ○ Off

Incoming Packets ● Port0 ○ Port1

Buffer Size 256

CDP
CDP
CDP
CDP
CDP
CDP
CDP
CDP
ARP
CDP
CDP
CDP
CDP
ARP
CDP
ICMP
CDP
CDP
ICMP
ICMP
ICMP
ICMP
CDP

EthernetII

0 4 8 Bytes

PREAMBLE: 101010..10 | SFD | DEST ADDR:00D0.58C8.4E80

SRC ADDR:0040.0B39.9850 | TYPE:0x0800 | DATA (VARIABLE LENGTH) | FCS:0x00000000

IP

0 4 8 16 20 24 Bits

VER:4 | IHL:5 | DSCP:0x00 | TL:128

ID:0x0012 | FLAGS: 0x0 | FRAG OFFSET:0x000

TTL:127 | PRO:0x01 | CHKSUM

SRC IP:100.0.0.1

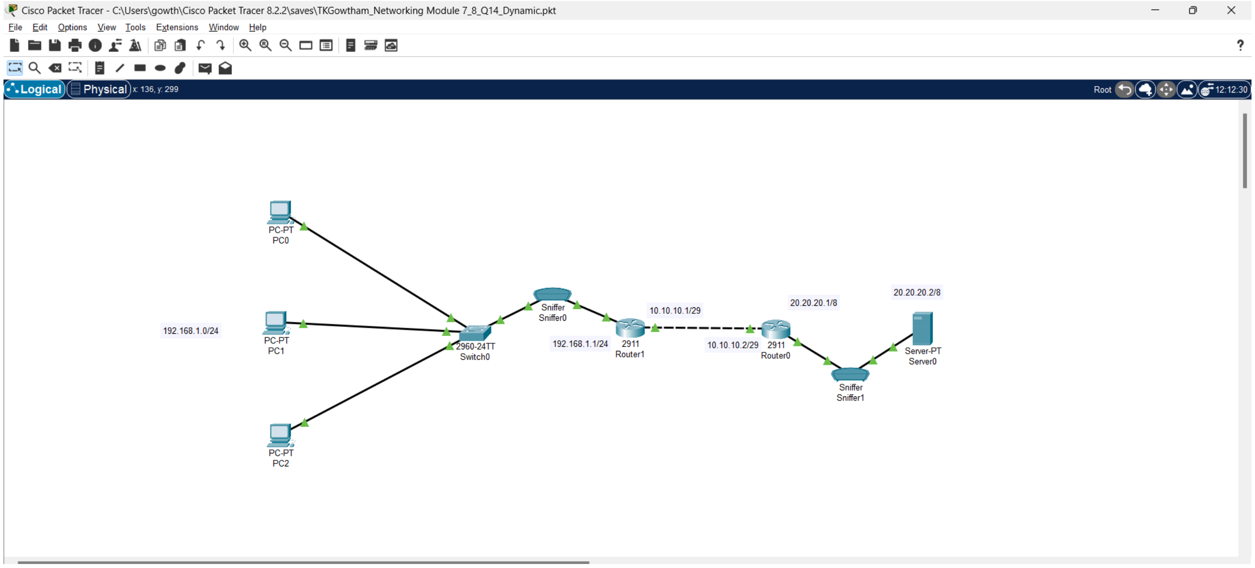
DST IP:203.0.113.2

DATA (VARIABLE LENGTH)

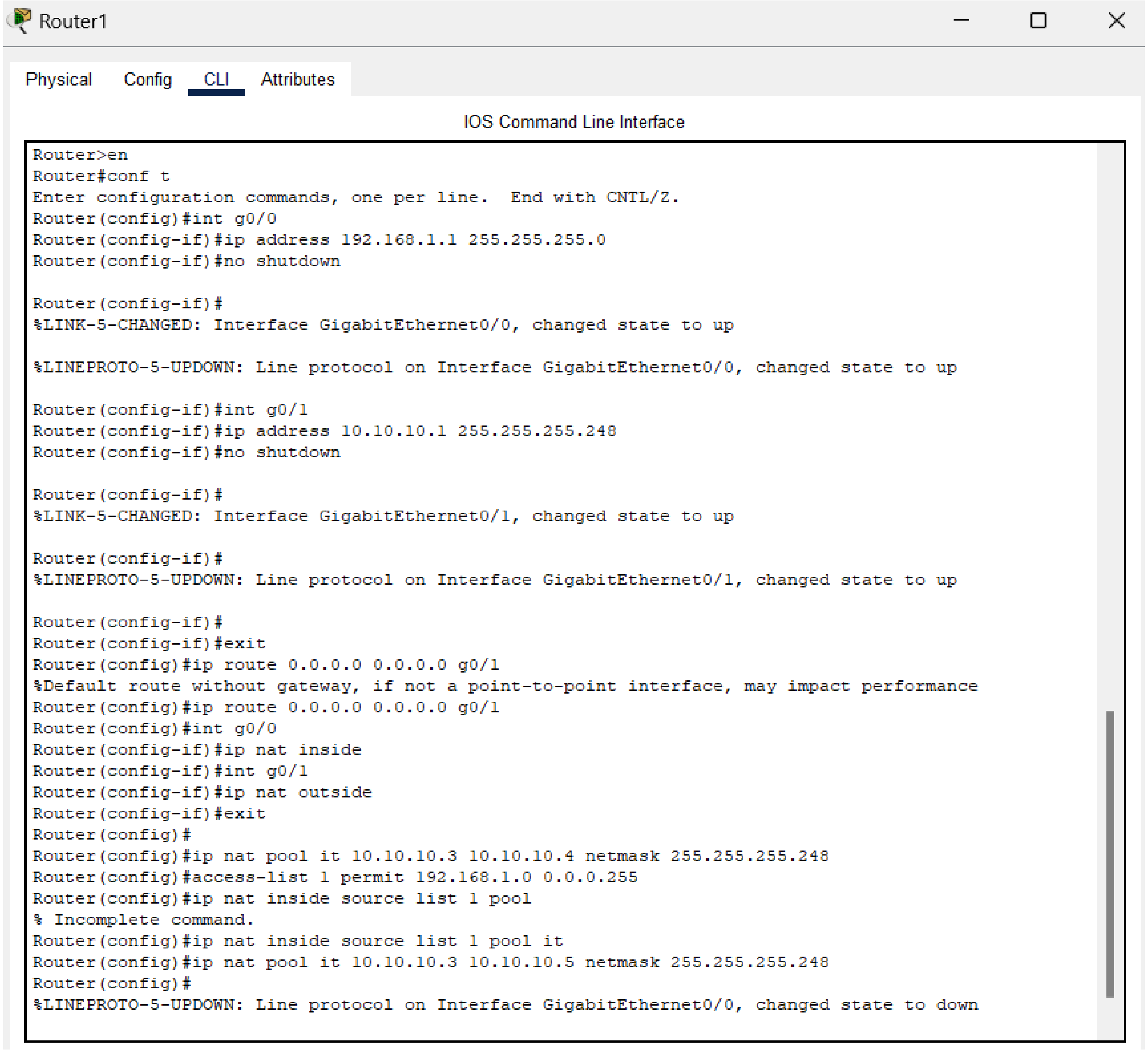
ICMP

Dynamic NAT :

Creating this network in the cisco packet tracer accordingly :

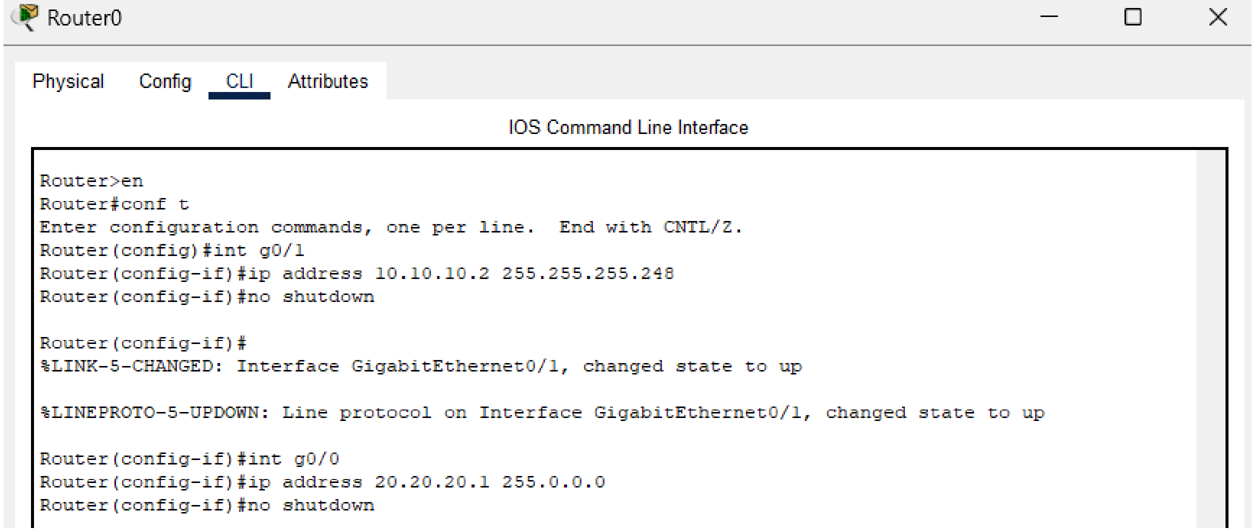


In the router, add the NAT inside and outside accordingly, then create an access list of pool of IP address. And assign the pool of IP address to the NAT source which will be dynamically chosen based on the need from the pool whichever IP is free.



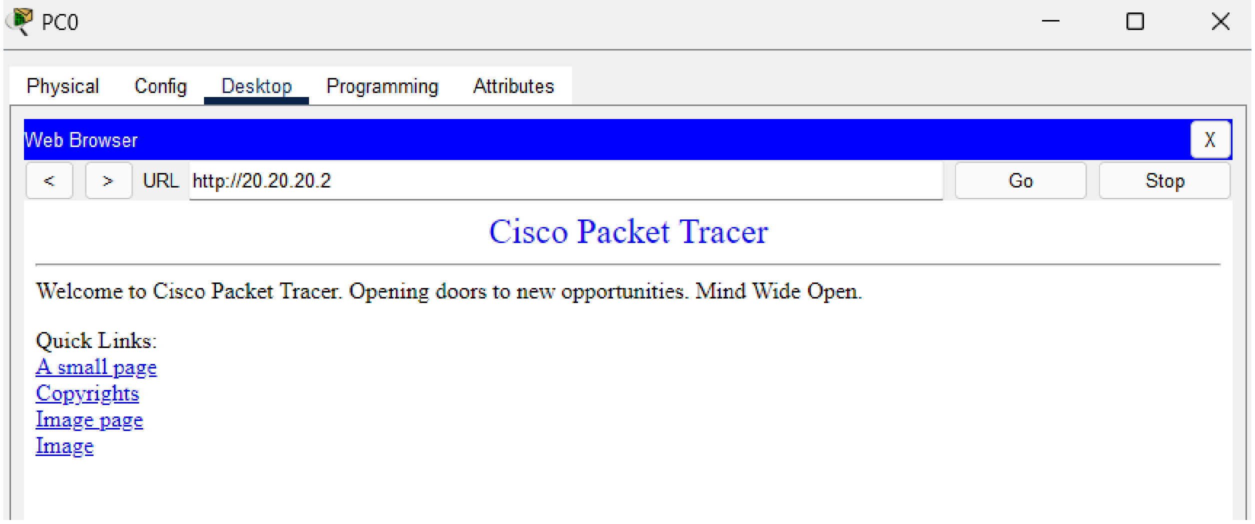
```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int g0/0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown

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

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

Router(config-if)#int g0/1
Router(config-if)#ip address 10.10.10.1 255.255.255.248
Router(config-if)#no shutdown

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

Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

Router(config-if)#
Router(config-if)#exit
Router(config)#ip route 0.0.0.0 0.0.0.0 g0/1
%Default route without gateway, if not a point-to-point interface, may impact performance
Router(config)#ip route 0.0.0.0 0.0.0.0 g0/1
Router(config)#int g0/0
Router(config-if)#ip nat inside
Router(config-if)#int g0/1
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#
Router(config)#ip nat pool it 10.10.10.3 10.10.10.4 netmask 255.255.255.248
Router(config)#access-list 1 permit 192.168.1.0 0.0.0.255
Router(config)#ip nat inside source list 1 pool
% Incomplete command.
Router(config)#ip nat inside source list 1 pool it
Router(config)#ip nat pool it 10.10.10.3 10.10.10.5 netmask 255.255.255.248
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to down
```

Also configure the other router with the interface IP address :



```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int g0/1
Router(config-if)#ip address 10.10.10.2 255.255.255.248
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)#int g0/0
Router(config-if)#ip address 20.20.20.1 255.0.0.0
Router(config-if)#no shutdown
```

Now try pinging or viewing the IP address in any PC :



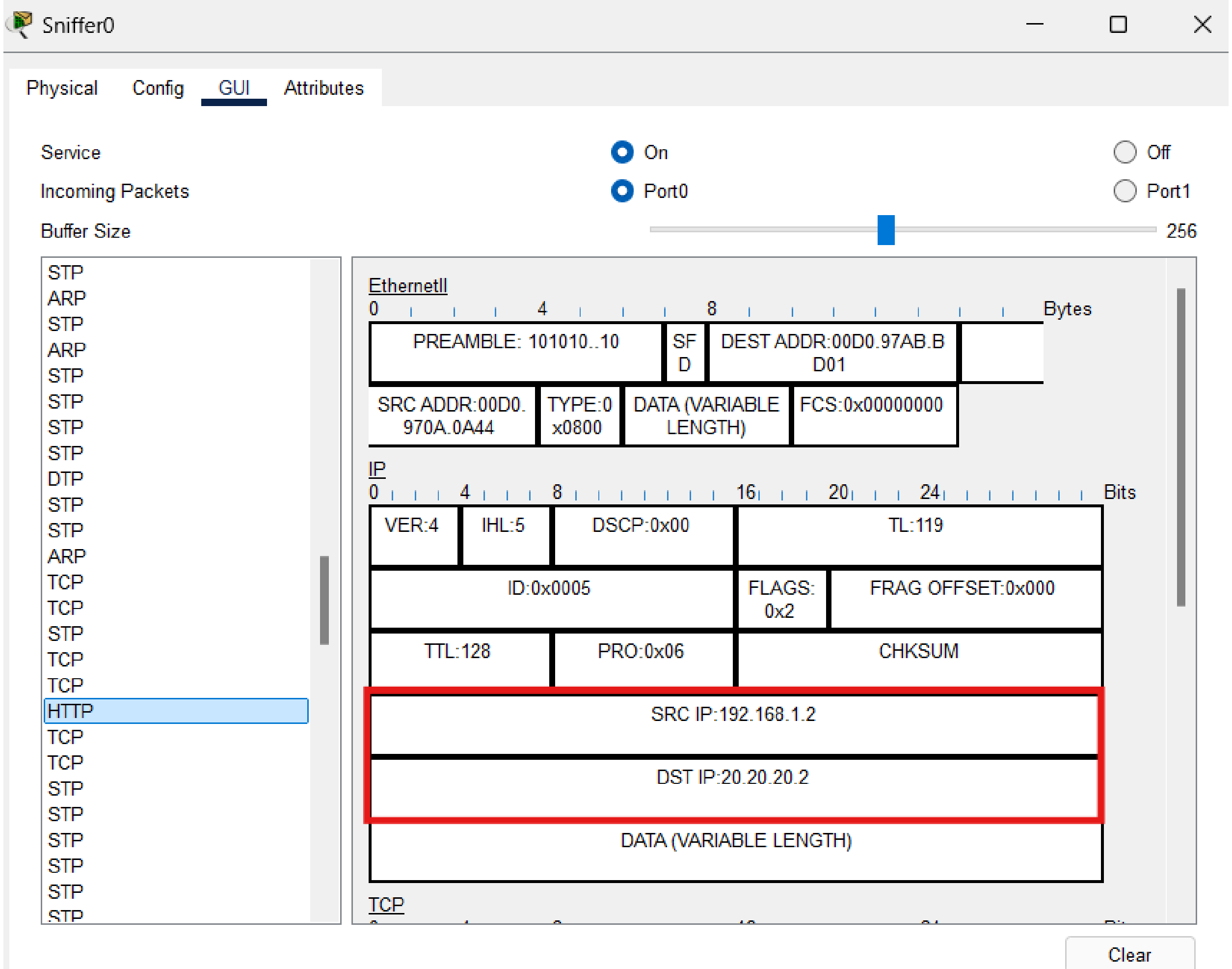
URL http://20.20.20.2

Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

Quick Links:
A small page
Copyrights
Image page
Image

In the NAT inside of the router sniffer, we are able to see that Source IP is still Private IP address.



Service ● On ○ Off

Incoming Packets ● Port0 ○ Port1

Buffer Size 256

STP
ARP
STP
ARP
STP
STP
STP
STP
DTP
STP
STP
ARP
TCP
TCP
STP
TCP
TCP
HTTP
TCP
TCP
STP
STP
STP
STP
STP
STP

EthernetII

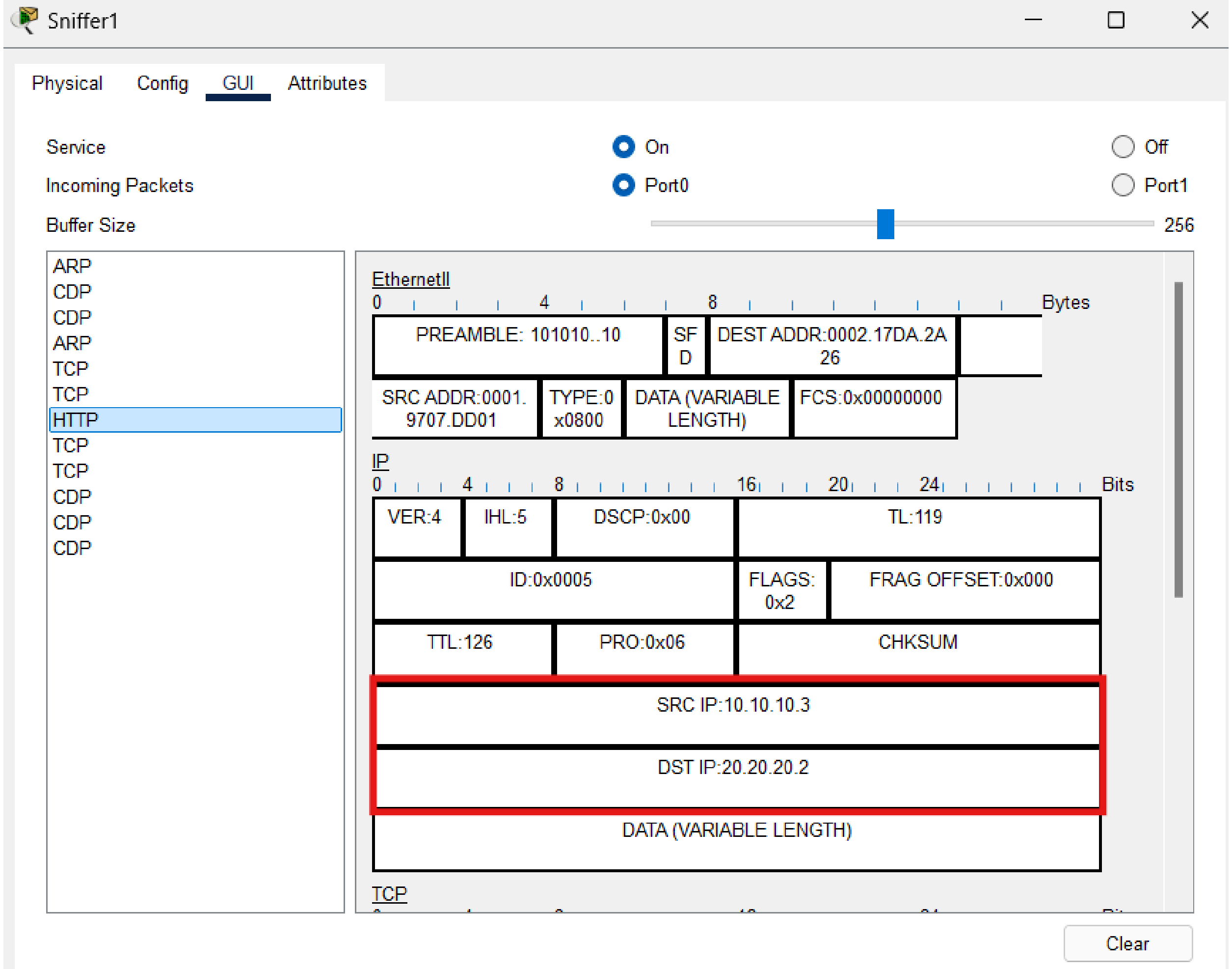
0 4 8 Bytes			
PREAMBLE: 101010..10	SFD	DEST ADDR:00D0.97AB.BD01	
SRC ADDR:00D0.970A.0A44	TYPE:0x0800	DATA (VARIABLE LENGTH)	FCS:0x00000000

IP

0 4 8 16 20 24 Bits			
VER:4	IHL:5	DSCP:0x00	TL:119
ID:0x0005		FLAGS: 0x2	FRAG OFFSET:0x000
TTL:128	PRO:0x06	CHKSUM	
SRC IP:192.168.1.2			
DST IP:20.20.20.2			
DATA (VARIABLE LENGTH)			

TCP

In the NAT Outside, we are able to see the Private IP address has been changed to Public IP address by dynamic NAT as the IP is chosen automatically from the pool of IP address.



Service ● On ○ Off

Incoming Packets ● Port0 ○ Port1

Buffer Size 256

ARP
CDP
CDP
ARP
TCP
TCP
HTTP
TCP
TCP
CDP
CDP
CDP

EthernetII

PREAMBLE: 101010..10 | SFD | DEST ADDR:0002.17DA.2A26

SRC ADDR:0001.9707.DD01 | TYPE:0x0800 | DATA (VARIABLE LENGTH) | FCS:0x00000000

IP

VER:4 | IHL:5 | DSCP:0x00 | TL:119

ID:0x0005 | FLAGS: 0x2 | FRAG OFFSET:0x000

TTL:126 | PRO:0x06 | CHKSUM

SRC IP:10.10.10.3

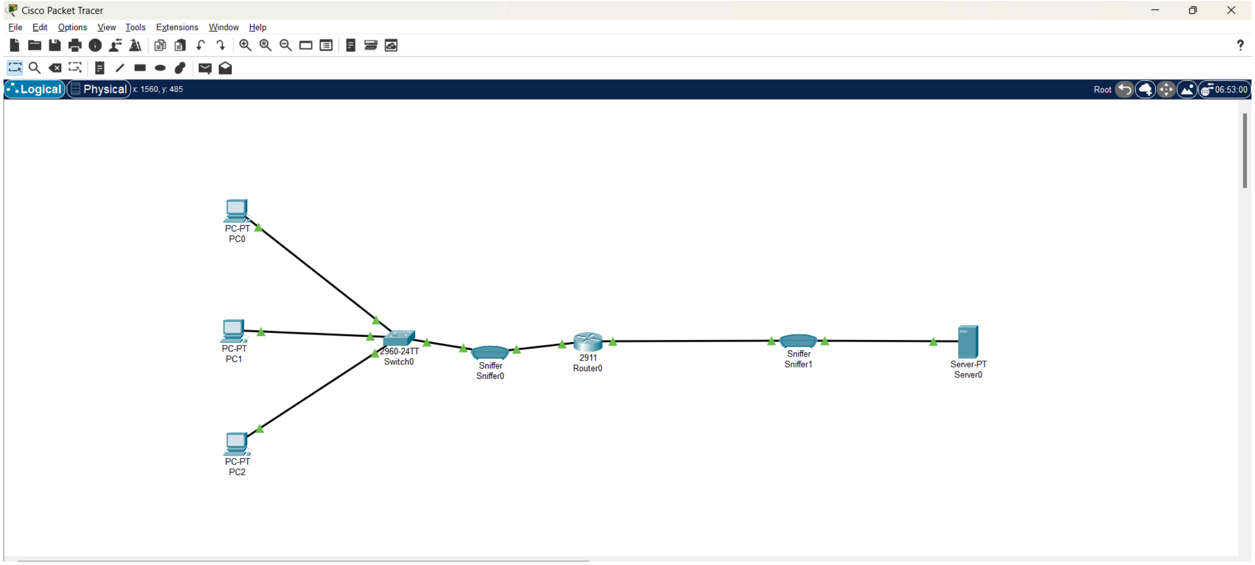
DST IP:20.20.20.2

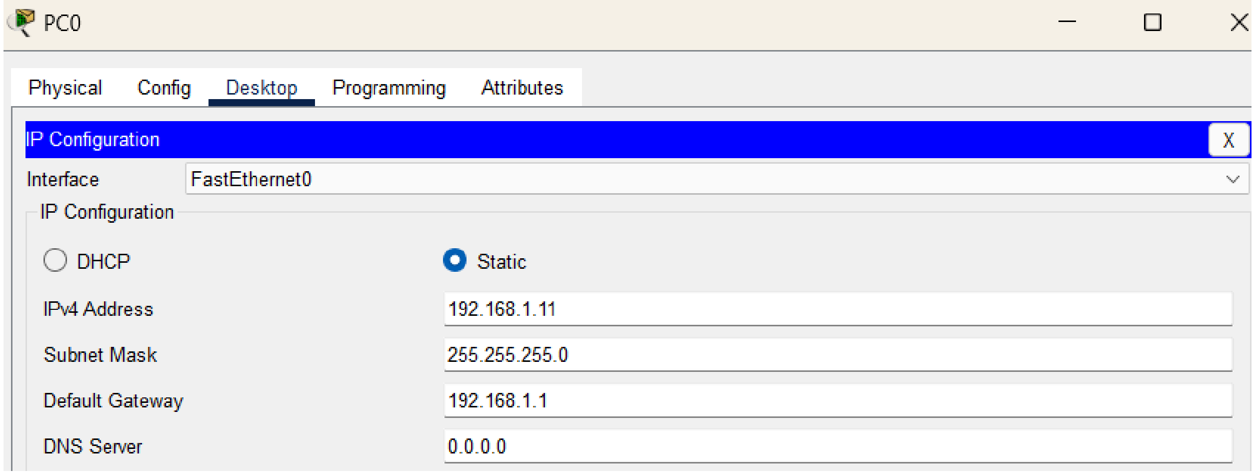
DATA (VARIABLE LENGTH)

TCP

PAT (Port Address Translation) :

Create this network and also assign the IP address and other details for the end devices accordingly :





IP Configuration

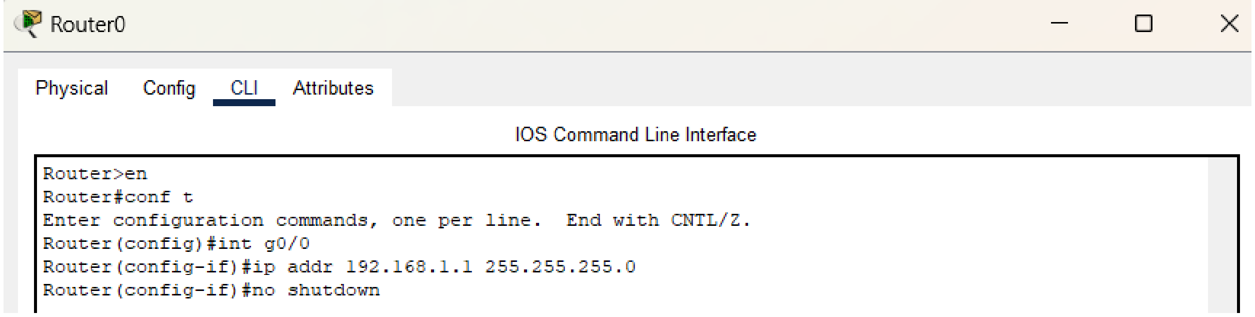
Interface FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address	192.168.1.11
Subnet Mask	255.255.255.0
Default Gateway	192.168.1.1
DNS Server	0.0.0.0

In the router add the IP address to the Interfaces :

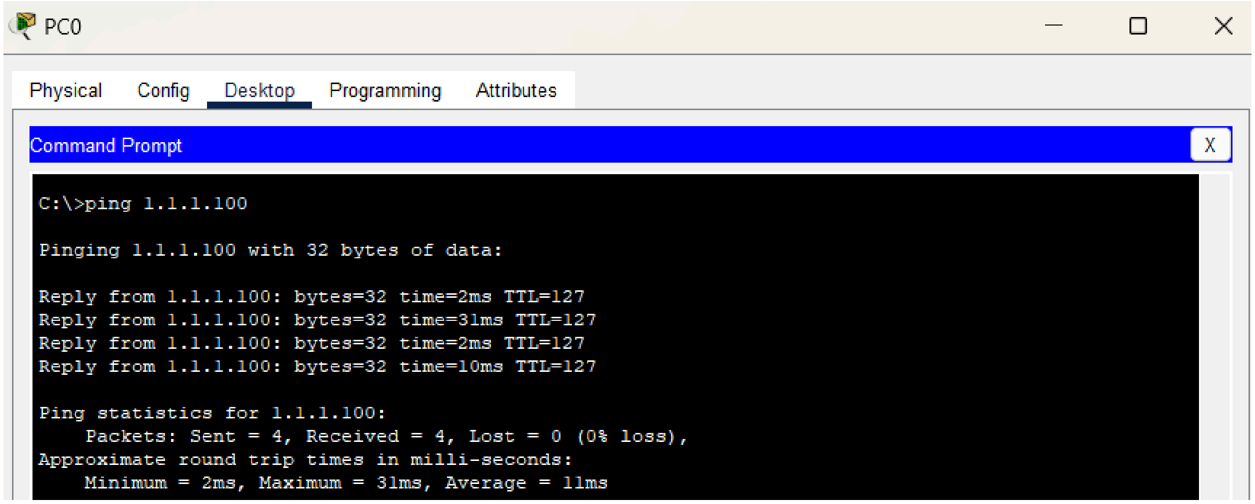


```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int g0/0
Router(config-if)#ip addr 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config)#int g0/1
Router(config-if)#ip addr 1.1.1.1 255.255.255.0
Router(config-if)#exit
```

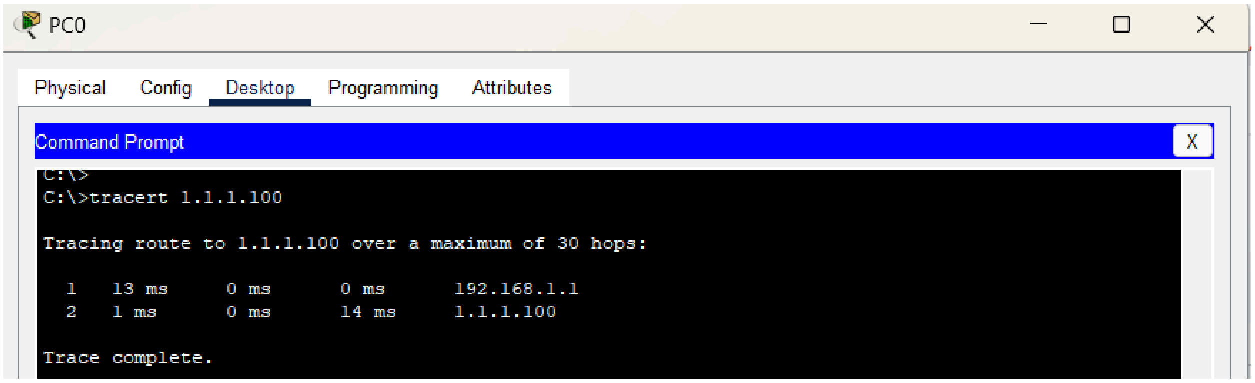
Assign NAT inside and Outside and then followed by an access list of port of IP address with overload.

```
Router(config-if)#int g0/0
Router(config-if)#ip nat inside
Router(config-if)#int g0/1
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#access-list 1 permit 192.168.1.0 0.0.0.255
Router(config)#ip nat inside source list 1 interface g0/1
Router(config)#ip nat inside source list 1 interface g0/1 overload
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to down

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

Now try to ping to server from any PC :

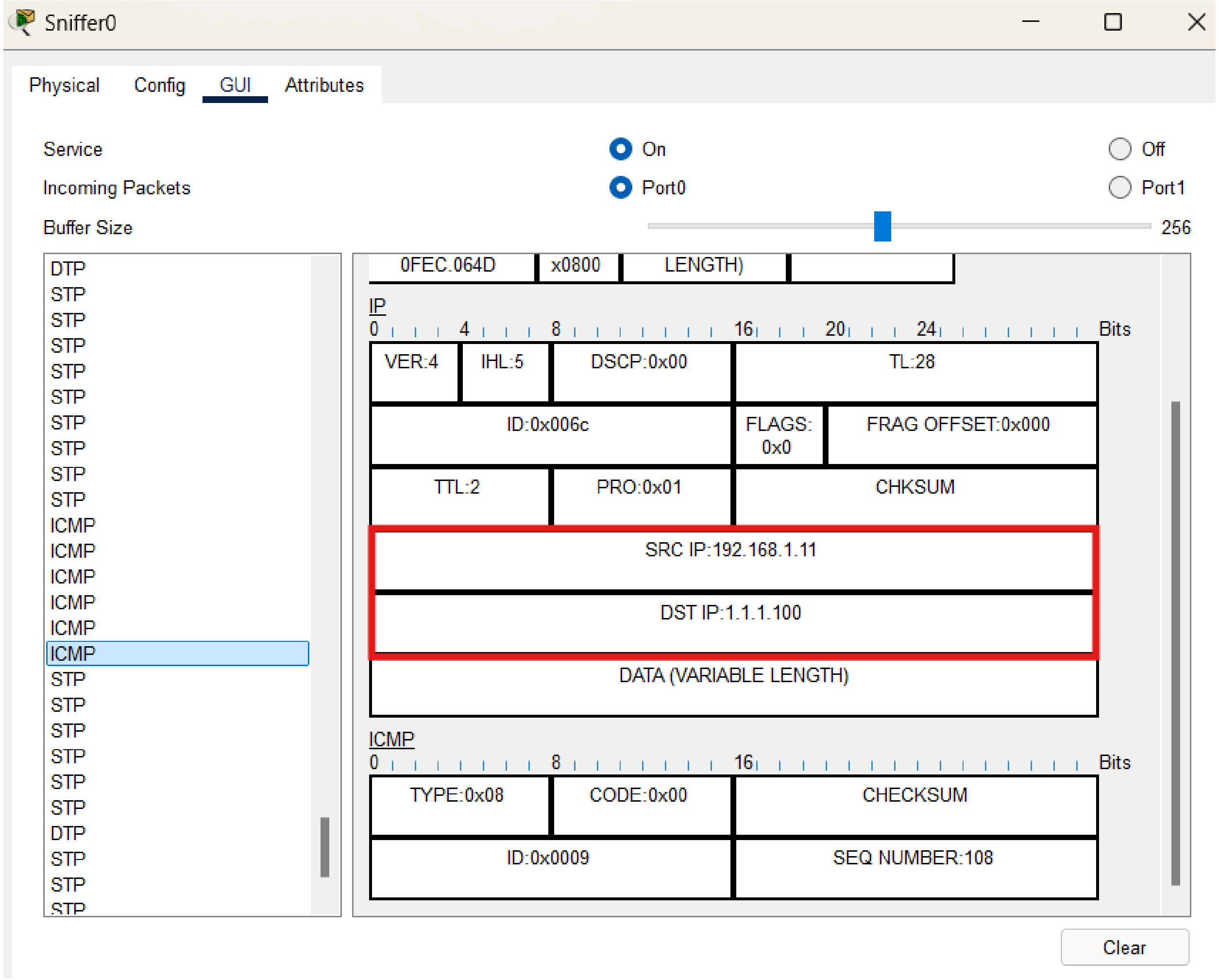


```
C:\>ping 1.1.1.100

Pinging 1.1.1.100 with 32 bytes of data:

Reply from 1.1.1.100: bytes=32 time=2ms TTL=127
Reply from 1.1.1.100: bytes=32 time=31ms TTL=127
Reply from 1.1.1.100: bytes=32 time=2ms TTL=127
Reply from 1.1.1.100: bytes=32 time=10ms TTL=127

Ping statistics for 1.1.1.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 31ms, Average = 11ms
```



```
C:\>
C:\>tracert 1.1.1.100

Tracing route to 1.1.1.100 over a maximum of 30 hops:

  1   13 ms     0 ms      0 ms      192.168.1.1
  2   1 ms      0 ms      14 ms     1.1.1.100

Trace complete.
```

We see that Private IP is successfully changed to public IP :



Service ● On ○ Off

Incoming Packets ● Port0 ○ Port1

Buffer Size 256

DTP
STP
STP
STP
STP
STP
STP
STP
STP
STP
ICMP
ICMP
ICMP
ICMP
ICMP
ICMP
STP
STP
STP
STP
STP
STP
DTP
STP
STP
STP

0FEC.064D	x0800	LENGTH)	

IP

0 4 8 16 20 24 Bits

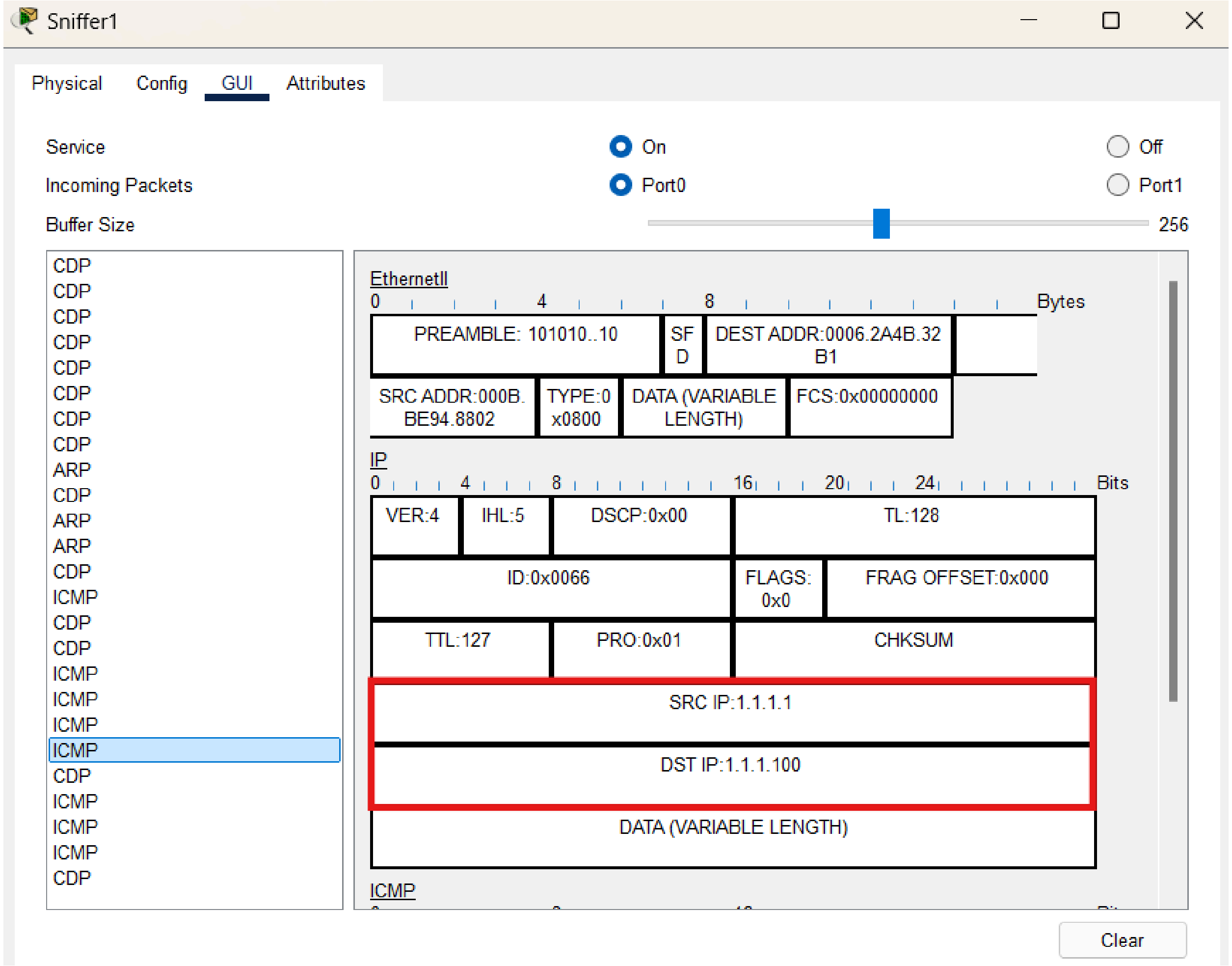
VER:4	IHL:5	DSCP:0x00	TL:28	
ID:0x006c			FLAGS: 0x0	FRAG OFFSET:0x000
TTL:2		PRO:0x01	CHKSUM	
SRC IP:192.168.1.11				
DST IP:1.1.1.100				
DATA (VARIABLE LENGTH)				

ICMP

0 8 16 Bits

TYPE:0x08	CODE:0x00	CHECKSUM
ID:0x0009		SEQ NUMBER:108

Changed to public IP of the router :



Sniffer1

Service ● On ○ Off

Incoming Packets ● Port0 ○ Port1

Buffer Size 256

CDP
CDP
CDP
CDP
CDP
CDP
CDP
CDP
ARP
CDP
ARP
ARP
CDP
ICMP
CDP
CDP
ICMP
ICMP
ICMP
ICMP
CDP
ICMP
ICMP
ICMP
CDP

EthernetII

0 4 8 Bytes

PREAMBLE: 101010..10 | SFD | DEST ADDR:0006.2A4B.32B1

SRC ADDR:000B.BE94.8802 | TYPE:0x0800 | DATA (VARIABLE LENGTH) | FCS:0x00000000

IP

0 4 8 16 20 24 Bits

VER:4 | IHL:5 | DSCP:0x00 | TL:128

ID:0x0066 | FLAGS: 0x0 | FRAG OFFSET:0x000

TTL:127 | PRO:0x01 | CHKSUM

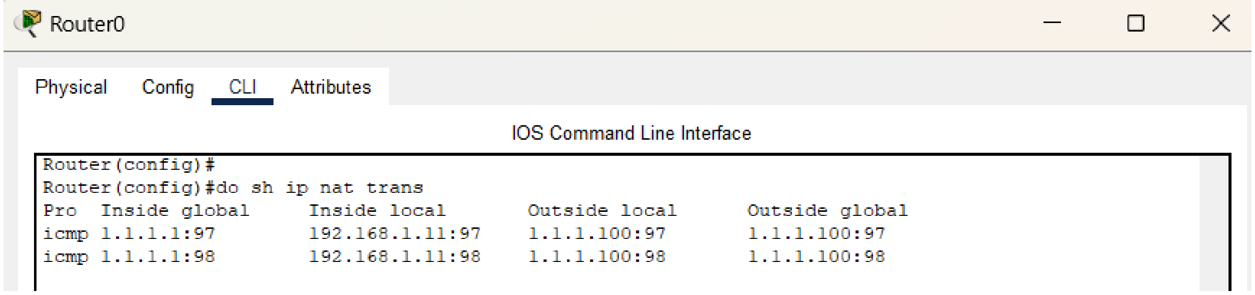
SRC IP:1.1.1.1

DST IP:1.1.1.100

DATA (VARIABLE LENGTH)

ICMP

We can see in the router NAT translations the Ports have been changed too.



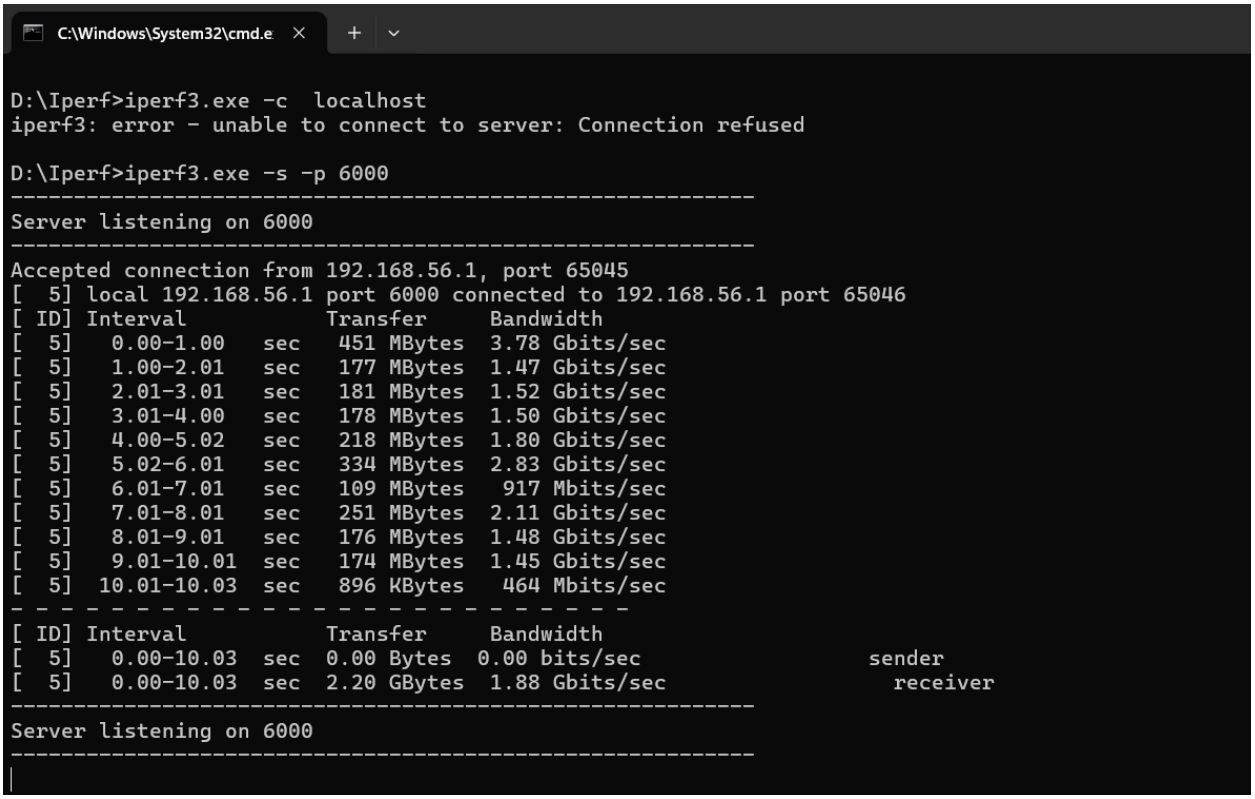
Router0

IOS Command Line Interface

```
Router(config)#
Router(config)#do sh ip nat trans
Pro  Inside global     Inside local       Outside local      Outside global
icmp 1.1.1.1:97        192.168.1.11:97    1.1.1.100:97       1.1.1.100:97
icmp 1.1.1.1:98        192.168.1.11:98    1.1.1.100:98       1.1.1.100:98
```

15. Download iperf in laptop/phone and make sure they are in same network. Try different iperf commands with tcp, udp, bidirectional, reverse, multicast, parallel options and analyze the bandwidth and rate of transmission, delay, jitter etc.

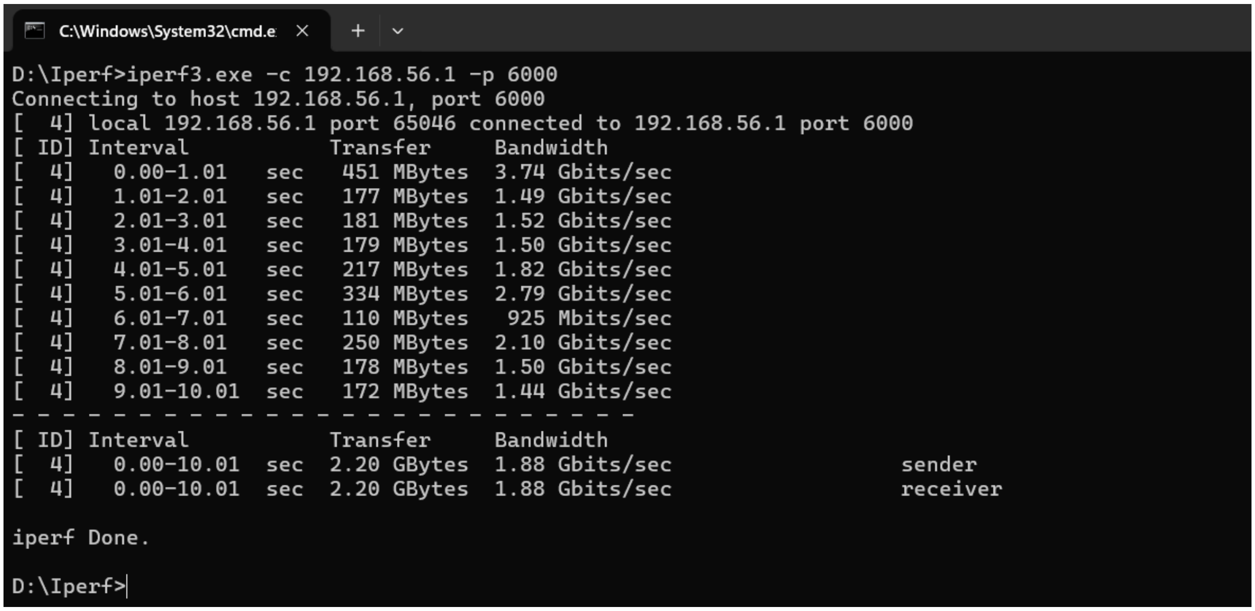
Using the same PC as Server and Router to examine the network. Here is the Server side :



```
D:\Iperf>iperf3.exe -c  localhost
iperf3: error - unable to connect to server: Connection refused

D:\Iperf>iperf3.exe -s -p 6000
-----------------------------------------------------------
Server listening on 6000
-----------------------------------------------------------
Accepted connection from 192.168.56.1, port 65045
[  5] local 192.168.56.1 port 6000 connected to 192.168.56.1 port 65046
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-1.00   sec   451 MBytes  3.78 Gbits/sec
[  5]   1.00-2.01   sec   177 MBytes  1.47 Gbits/sec
[  5]   2.01-3.01   sec   181 MBytes  1.52 Gbits/sec
[  5]   3.01-4.00   sec   178 MBytes  1.50 Gbits/sec
[  5]   4.00-5.02   sec   218 MBytes  1.80 Gbits/sec
[  5]   5.02-6.01   sec   334 MBytes  2.83 Gbits/sec
[  5]   6.01-7.01   sec   109 MBytes   917 Mbits/sec
[  5]   7.01-8.01   sec   251 MBytes  2.11 Gbits/sec
[  5]   8.01-9.01   sec   176 MBytes  1.48 Gbits/sec
[  5]   9.01-10.01  sec   174 MBytes  1.45 Gbits/sec
[  5]  10.01-10.03  sec   896 KBytes   464 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-10.03  sec  0.00 Bytes  0.00 bits/sec                  sender
[  5]   0.00-10.03  sec  2.20 GBytes  1.88 Gbits/sec                  receiver
-----------------------------------------------------------
Server listening on 6000
-----------------------------------------------------------
```

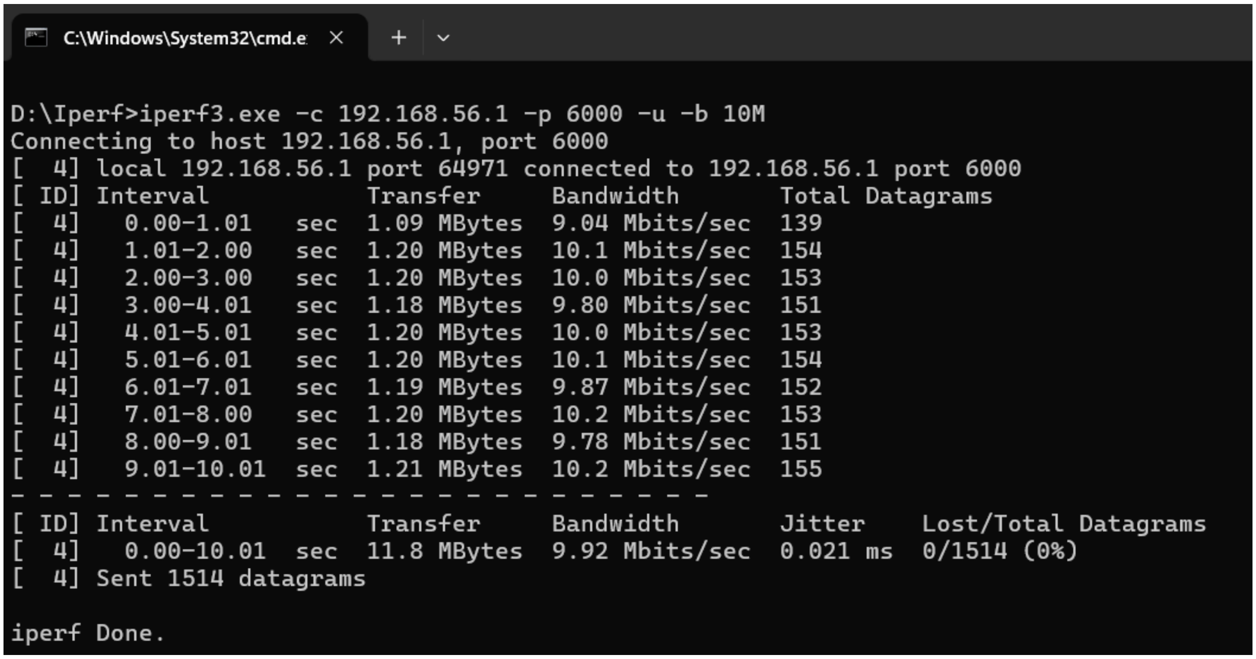
In the client side we will check with different parameters as given below :

TCP :



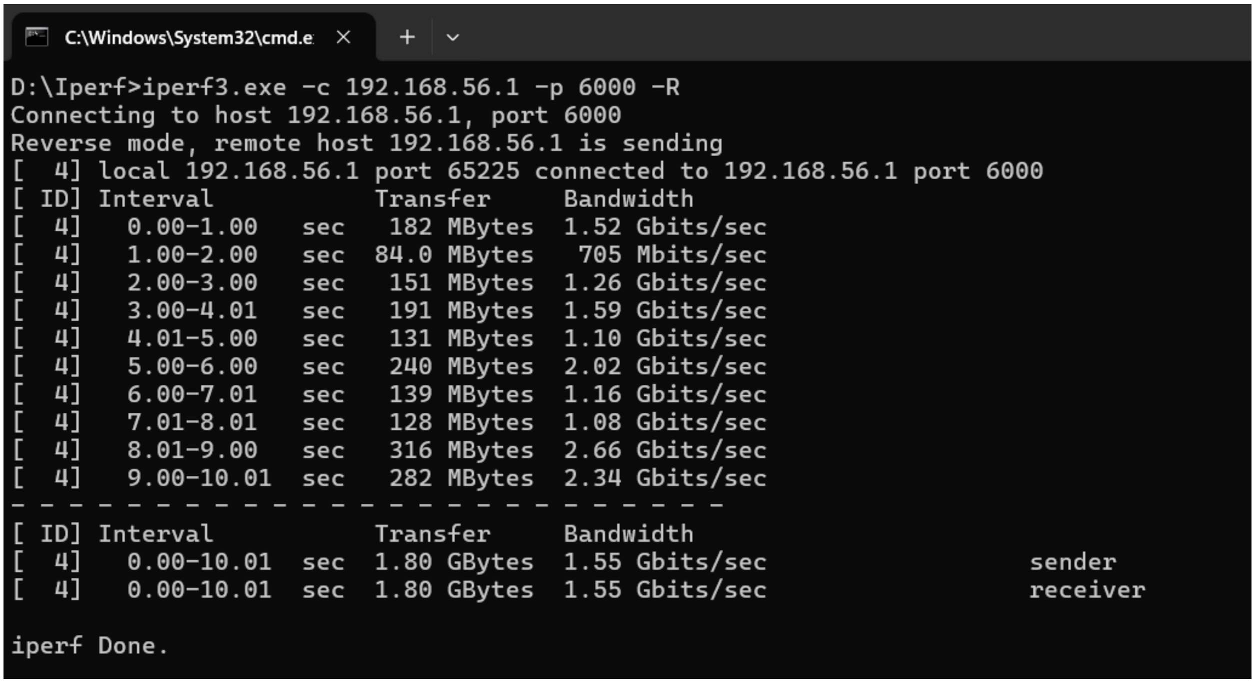
```
D:\Iperf>iperf3.exe -c 192.168.56.1 -p 6000
Connecting to host 192.168.56.1, port 6000
[  4] local 192.168.56.1 port 65046 connected to 192.168.56.1 port 6000
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-1.01   sec   451 MBytes  3.74 Gbits/sec
[  4]   1.01-2.01   sec   177 MBytes  1.49 Gbits/sec
[  4]   2.01-3.01   sec   181 MBytes  1.52 Gbits/sec
[  4]   3.01-4.01   sec   179 MBytes  1.50 Gbits/sec
[  4]   4.01-5.01   sec   217 MBytes  1.82 Gbits/sec
[  4]   5.01-6.01   sec   334 MBytes  2.79 Gbits/sec
[  4]   6.01-7.01   sec   110 MBytes   925 Mbits/sec
[  4]   7.01-8.01   sec   250 MBytes  2.10 Gbits/sec
[  4]   8.01-9.01   sec   178 MBytes  1.50 Gbits/sec
[  4]   9.01-10.01  sec   172 MBytes  1.44 Gbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-10.01  sec  2.20 GBytes  1.88 Gbits/sec                  sender
[  4]   0.00-10.01  sec  2.20 GBytes  1.88 Gbits/sec                  receiver

iperf Done.

D:\Iperf>
```

UDP :



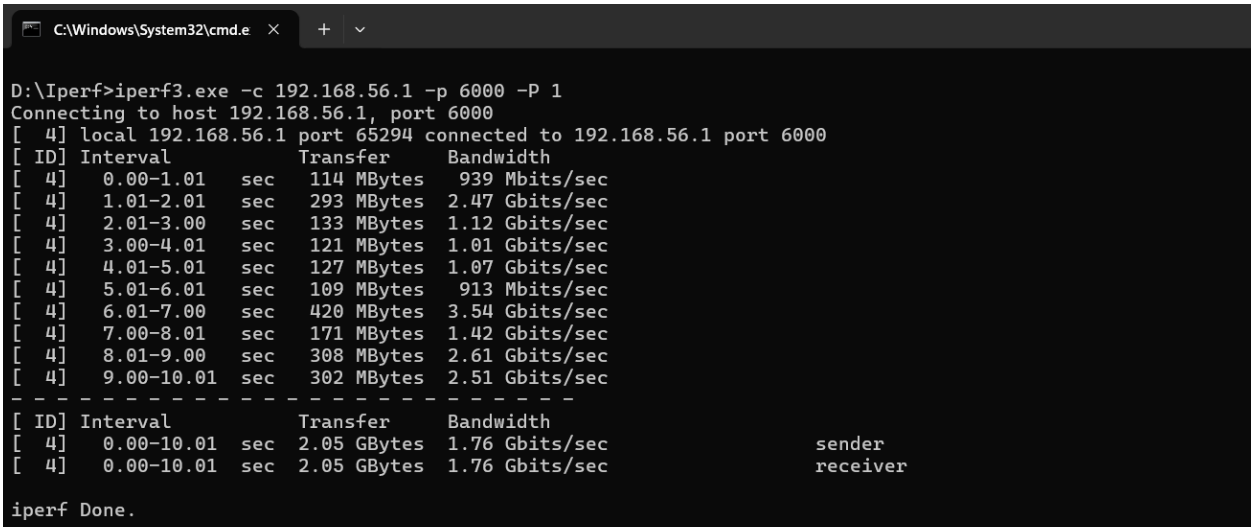
```
D:\Iperf>iperf3.exe -c 192.168.56.1 -p 6000 -u -b 10M
Connecting to host 192.168.56.1, port 6000
[  4] local 192.168.56.1 port 64971 connected to 192.168.56.1 port 6000
[ ID] Interval           Transfer     Bandwidth       Total Datagrams
[  4]   0.00-1.01   sec  1.09 MBytes  9.04 Mbits/sec  139
[  4]   1.01-2.00   sec  1.20 MBytes  10.1 Mbits/sec  154
[  4]   2.00-3.00   sec  1.20 MBytes  10.0 Mbits/sec  153
[  4]   3.00-4.01   sec  1.18 MBytes  9.80 Mbits/sec  151
[  4]   4.01-5.01   sec  1.20 MBytes  10.0 Mbits/sec  153
[  4]   5.01-6.01   sec  1.20 MBytes  10.1 Mbits/sec  154
[  4]   6.01-7.01   sec  1.19 MBytes  9.87 Mbits/sec  152
[  4]   7.01-8.00   sec  1.20 MBytes  10.2 Mbits/sec  153
[  4]   8.00-9.01   sec  1.18 MBytes  9.78 Mbits/sec  151
[  4]   9.01-10.01  sec  1.21 MBytes  10.2 Mbits/sec  155
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  4]   0.00-10.01  sec  11.8 MBytes  9.92 Mbits/sec  0.021 ms  0/1514 (0%)
[  4] Sent 1514 datagrams

iperf Done.
```

Reverse :



```
D:\Iperf>iperf3.exe -c 192.168.56.1 -p 6000 -R
Connecting to host 192.168.56.1, port 6000
Reverse mode, remote host 192.168.56.1 is sending
[  4] local 192.168.56.1 port 65225 connected to 192.168.56.1 port 6000
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-1.00   sec   182 MBytes  1.52 Gbits/sec
[  4]   1.00-2.00   sec  84.0 MBytes   705 Mbits/sec
[  4]   2.00-3.00   sec   151 MBytes  1.26 Gbits/sec
[  4]   3.00-4.01   sec   191 MBytes  1.59 Gbits/sec
[  4]   4.01-5.00   sec   131 MBytes  1.10 Gbits/sec
[  4]   5.00-6.00   sec   240 MBytes  2.02 Gbits/sec
[  4]   6.00-7.01   sec   139 MBytes  1.16 Gbits/sec
[  4]   7.01-8.01   sec   128 MBytes  1.08 Gbits/sec
[  4]   8.01-9.00   sec   316 MBytes  2.66 Gbits/sec
[  4]   9.00-10.01  sec   282 MBytes  2.34 Gbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-10.01  sec  1.80 GBytes  1.55 Gbits/sec                  sender
[  4]   0.00-10.01  sec  1.80 GBytes  1.55 Gbits/sec                  receiver

iperf Done.
```

Parallel Streams :



```
D:\Iperf>iperf3.exe -c 192.168.56.1 -p 6000 -P 1
Connecting to host 192.168.56.1, port 6000
[  4] local 192.168.56.1 port 65294 connected to 192.168.56.1 port 6000
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-1.01   sec   114 MBytes   939 Mbits/sec
[  4]   1.01-2.01   sec   293 MBytes  2.47 Gbits/sec
[  4]   2.01-3.00   sec   133 MBytes  1.12 Gbits/sec
[  4]   3.00-4.01   sec   121 MBytes  1.01 Gbits/sec
[  4]   4.01-5.01   sec   127 MBytes  1.07 Gbits/sec
[  4]   5.01-6.01   sec   109 MBytes   913 Mbits/sec
[  4]   6.01-7.00   sec   420 MBytes  3.54 Gbits/sec
[  4]   7.00-8.01   sec   171 MBytes  1.42 Gbits/sec
[  4]   8.01-9.00   sec   308 MBytes  2.61 Gbits/sec
[  4]   9.00-10.01  sec   302 MBytes  2.51 Gbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-10.01  sec  2.05 GBytes  1.76 Gbits/sec                  sender
[  4]   0.00-10.01  sec  2.05 GBytes  1.76 Gbits/sec                  receiver

iperf Done.
```

Hence, we are able to successfully use iperf3 and see various parameters of the network with Server-Client usage in the same PC.