NAME:VISHWAS M SEC:F
SRN:PES2UG20CS390 SUB: CN LAB

Week #1

Study and understand the basic networking tools - Wireshark, Tcpdump, Ping, Traceroute.

Learn and Understand Network Tools

1. Wireshark

- Perform and analyze Ping PDU capture
- Examine HTTP packet capture
- Analyze HTTP packet capture using filter

2. Tcpdump

- Capture packets

3. Ping

- Test the connectivity between 2 systems

4. Traceroute

- Perform traceroute checks

5. Nmap

- Explore an entire network

IMPORTANT INSTRUCTIONS:

- This manual is written for Ubuntu Linux OS only. You can also execute these experiments on VirtualBox or VMWare platform.
- For few tasks, you may need to create 2 VMs for experimental setup.
- Perform **sudo apt-get update** before installing any tool or utility.
- Install any tool or utility using the command **sudo apt-get install name_of_the_tool**
- Take screenshots wherever necessary and upload it to Edmodo as a single PDF file. (Refer general guidelines for submission requirements).
- To define an IP address for your machine (e.g., Section – ‘a’ & Serial number is 1, then your IP address should be 10.0.1.1. Section – ‘h’ & & Serial number is 23, then your IP address should be 10.0.8.23) – applicable only for relevant tasks (which doesn’t requires internet connectivity to execute the tasks).

NAME:VISHWAS M SEC:F
SRN:PES2UG20CS390 SUB: CN LAB

Task 1: Linux Interface Configuration (ifconfig / IP command)

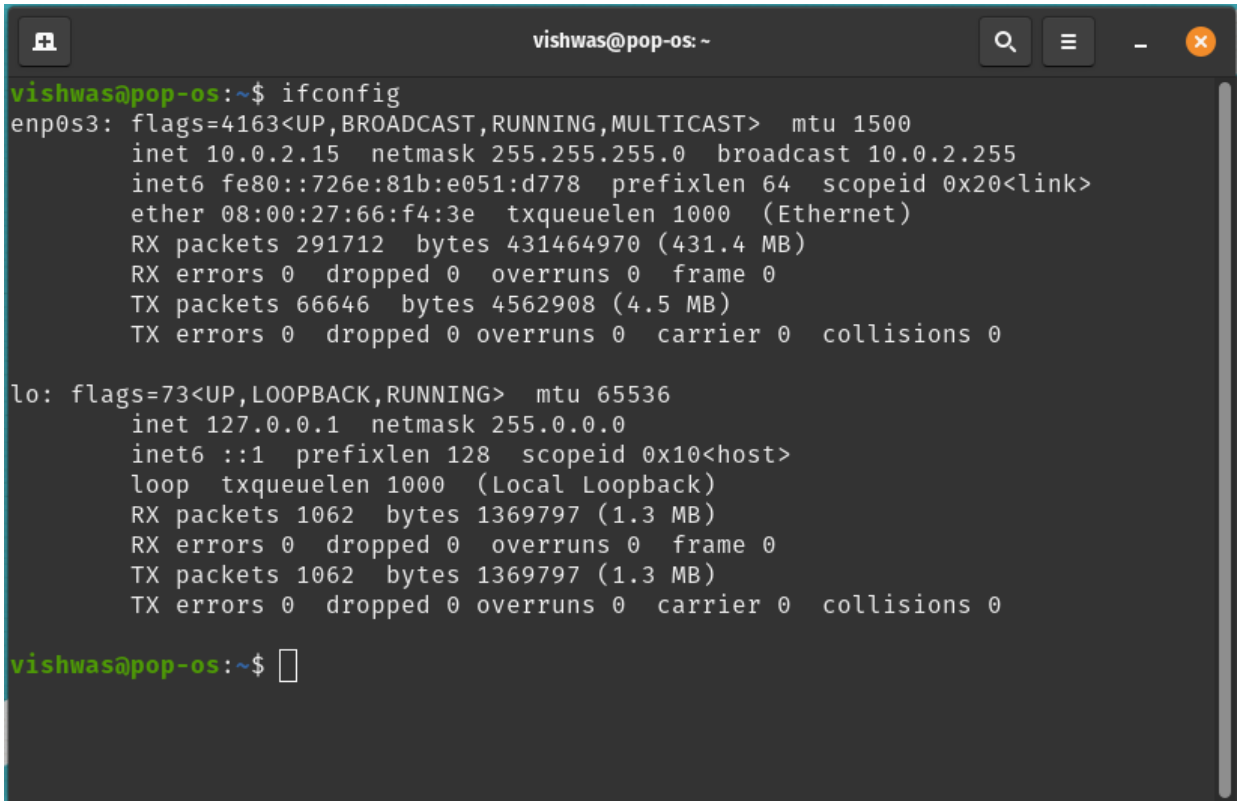
Step 1: To display status of all active network interfaces.

ifconfig (or) **ip addr show**

Analyze and fill the following table:

ip address table:

Interface name	IP address (IPv4 / IPv6)	MAC address	
enp0s3	10.0.2.15	08:00:27:66:f4:3e	
lo	127.0.0.1	-	



The hardware address and the IP address is mentioned,when ifconfig is typed in the terminal.

Step 2: To assign an IP address to an interface, use the following command.

sudo ifconfig interface_name 10.0.your_section.your_sno netmask 255.255.255.0 (or)

sudo ip addr add 10.0.your_section.your_sno /24 dev interface_name

```
vishwas@pop-os: ~
vishwas@pop-os:~$ sudo ifconfig enp0s3 10.0.6.57 netmask 255.255.255.0
vishwas@pop-os:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.6.57  netmask 255.255.255.0  broadcast 10.0.6.255
        inet6 fe80::726e:81b:e051:d778  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:66:f4:3e  txqueuelen 1000  (Ethernet)
        RX packets 319427  bytes 460821369 (460.8 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 81418  bytes 9939154 (9.9 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 7307  bytes 1934469 (1.9 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 7307  bytes 1934469 (1.9 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vishwas@pop-os:~$
```

10.0.6.57 is assigned as the IP address to the interface.

Step 3: To activate / deactivate a network interface, type.

sudo ifconfig interface_name down

sudo ifconfig interface_name up

```
vishwas@pop-os:~$ sudo ifconfig enp0s3 up
```

The configured interface is set to

up and running if it isn't.

Step 4: To show the current neighbor table in kernel, type

ip neigh

```
vishwas@pop-os:~$ ip neigh
192.168.0.1 dev enp0s3  FAILED
vishwas@pop-os:~$ ip neigh
192.168.0.1 dev enp0s3  INCOMPLETE
vishwas@pop-os:~$ ip neigh
192.168.0.1 dev enp0s3  INCOMPLETE
vishwas@pop-os:~$ ip neigh
192.168.0.1 dev enp0s3  INCOMPLETE
vishwas@pop-os:~$ ip neigh
192.168.0.1 dev enp0s3  INCOMPLETE
vishwas@pop-os:~$
```

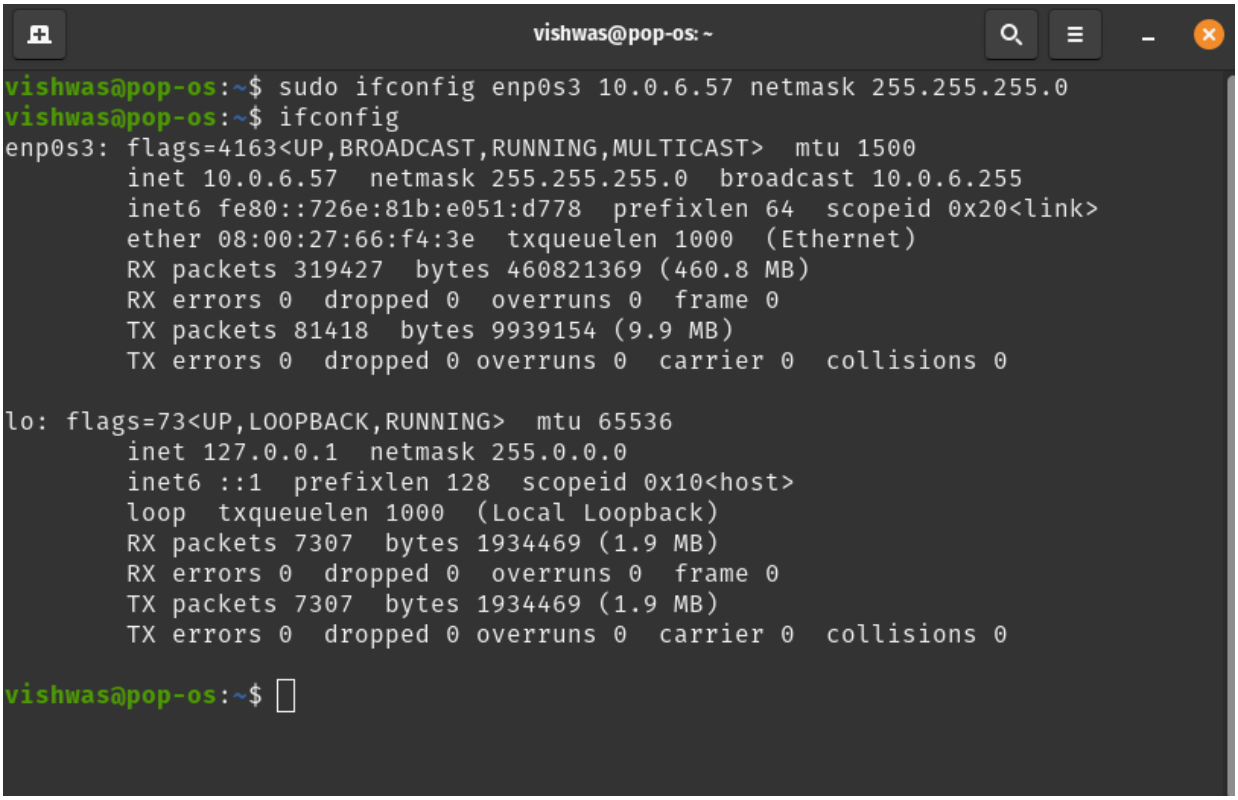
The neighbor table is shown in the output.

NAME:VISHWAS M SEC:F
SRN:PES2UG20CS390 SUB: CN LAB

Task 2: Ping PDU (Packet Data Units or Packets) Capture

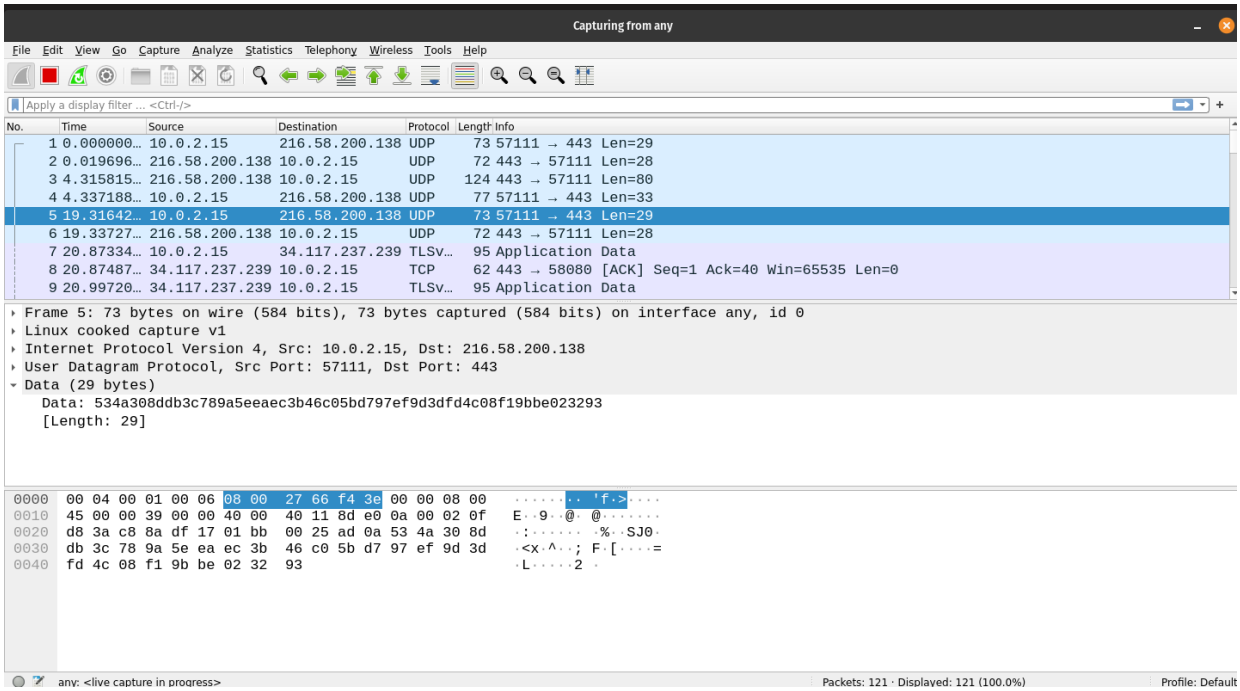
Step 1: Assign an IP address to the system (Host).

Note: IP address of your system should be 10.0.your_section.your_sno.



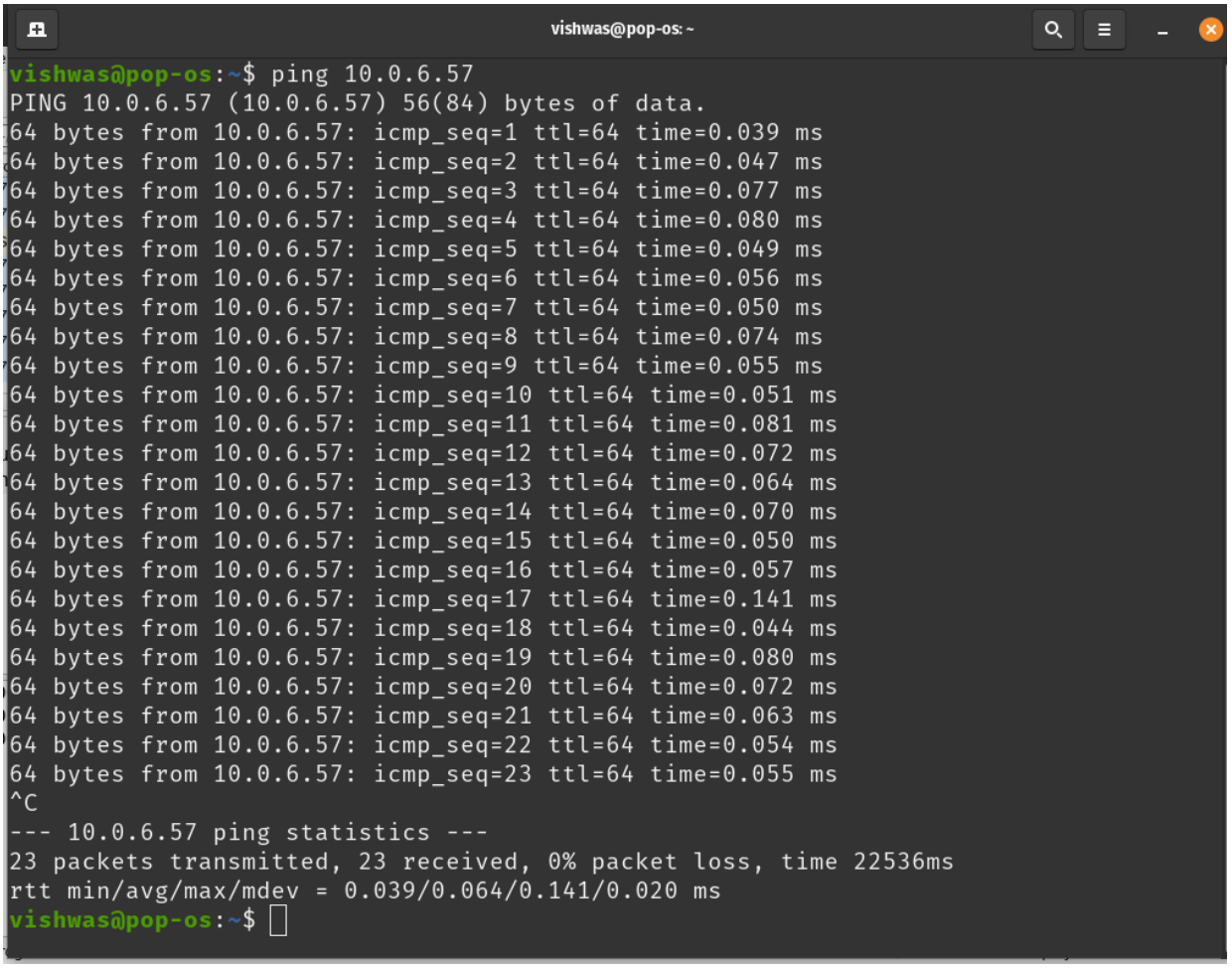
The IP address is set to 10.0.6.57.

Step 2: Launch Wireshark and select ‘any’ interfa



Wireshark on launch and opened into “any”.

Step 3: In terminal, type **ping 10.0.your_section.your_sno**



Observations to be made

Step 4: Analyze the following in Terminal

- TTL
- Protocol used by ping
- Time

Step 5: Analyze the following in Wireshark

On Packet List Pane, select the first echo packet on the list. On Packet Details Pane, click on each of the four "+" to expand the information. Analyze the frames with the first echo request and echo reply and complete the table below.

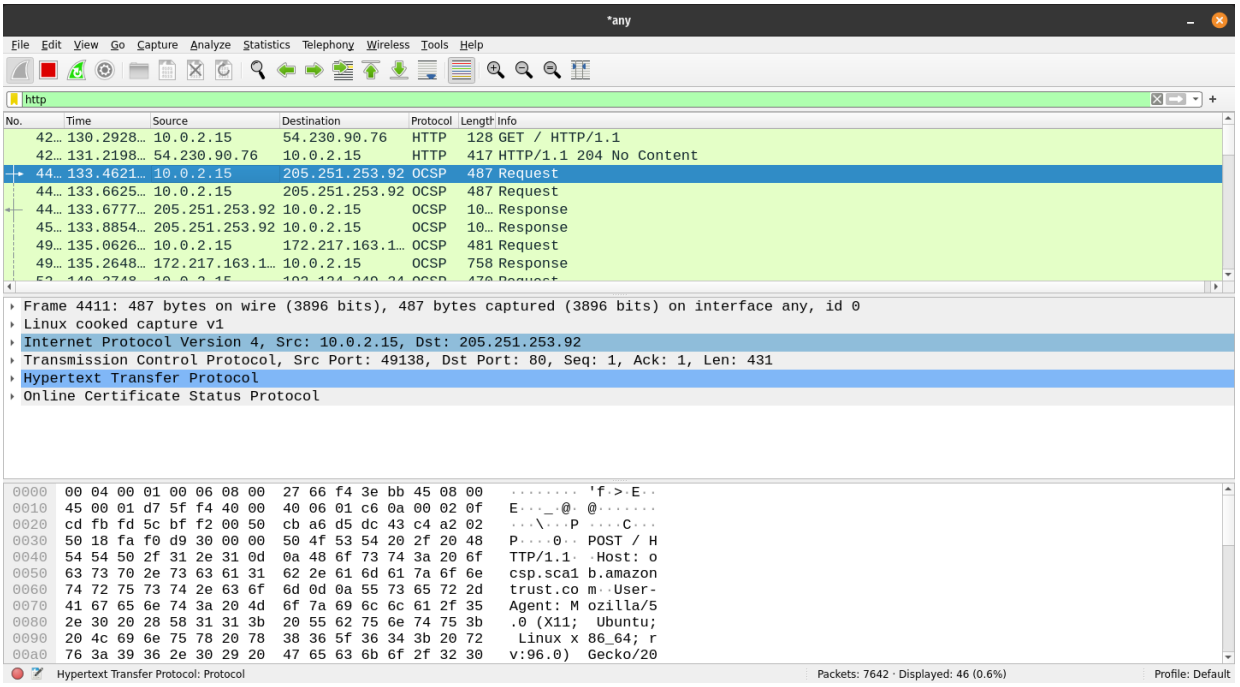
Details	First Echo Request	First Echo Reply
Frame Number	207	208
Source IP address	10.0.6.57	10.0.6.57
Destination IP address	10.0.6.57	10.0.6.57
ICMP Type Value	8	0
ICMP Code Value	0	0
Source Ethernet Address	00:00:00:00:00:00	00:00:00:00:00:00

Destination Ethernet Address	00:00:00:00:00:00	00:00:00:00:00:00
Internet Protocol Version	4	4
Time To Live (TTL) Value	64(reply in 208)	64(request in 207)

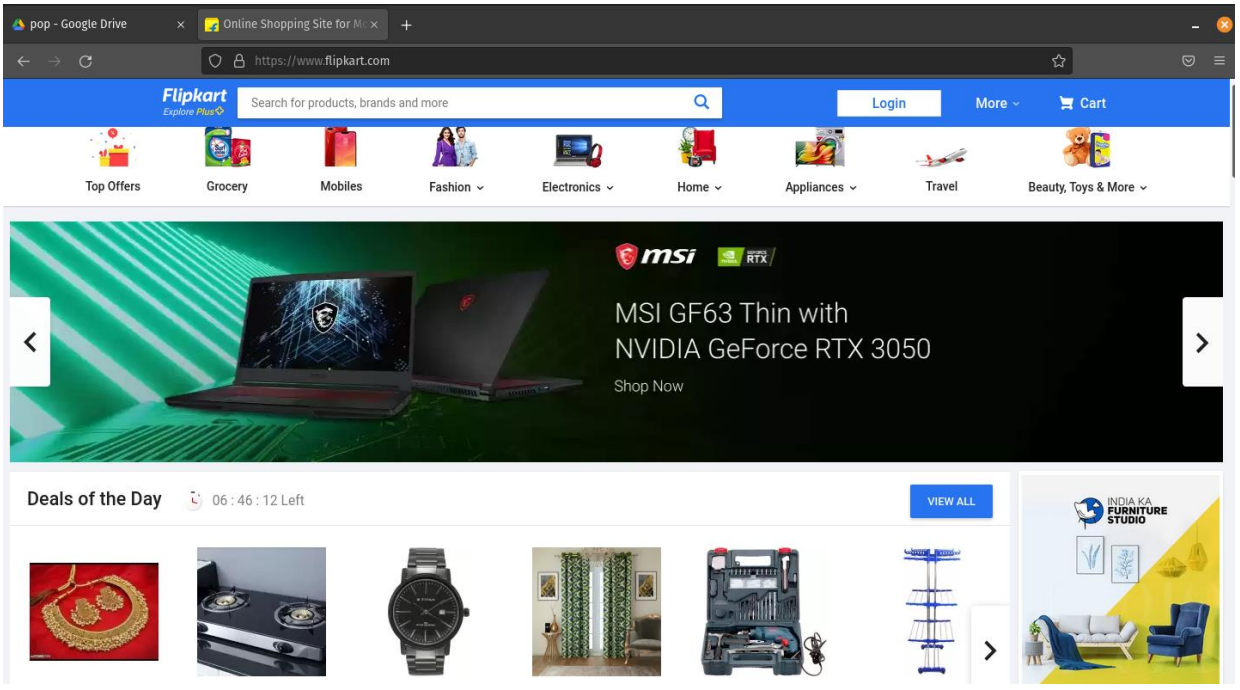
Task 3: HTTP PDU Capture

Using Wireshark's Filter feature

Step 1: Launch Wireshark and select 'any' interface. On the Filter toolbar, type-in 'http' and press enter



Step 2: Open Firefox browser, and browse www.flipkart.com



Observations to be made

Step 3: Analyze the first (interaction of host to the web server) and second frame (response of server to the client). By analyzing the filtered frames, complete the table below:

Details	First Echo Request	First Echo Reply
Frame Number	4411	4470
Source Port	49138	80
Destination Port	80	49138
Source IP address	10.0.2.15	205.251.253.92
Destination IP address	205.251.253.92	10.0.2.15
Source Ethernet Address	08:00:27:66:f4:3e	52:54:00:12:35:02
Destination Ethernet Address	52:54:00:12:35:02	08:00:27:66:f4:3e

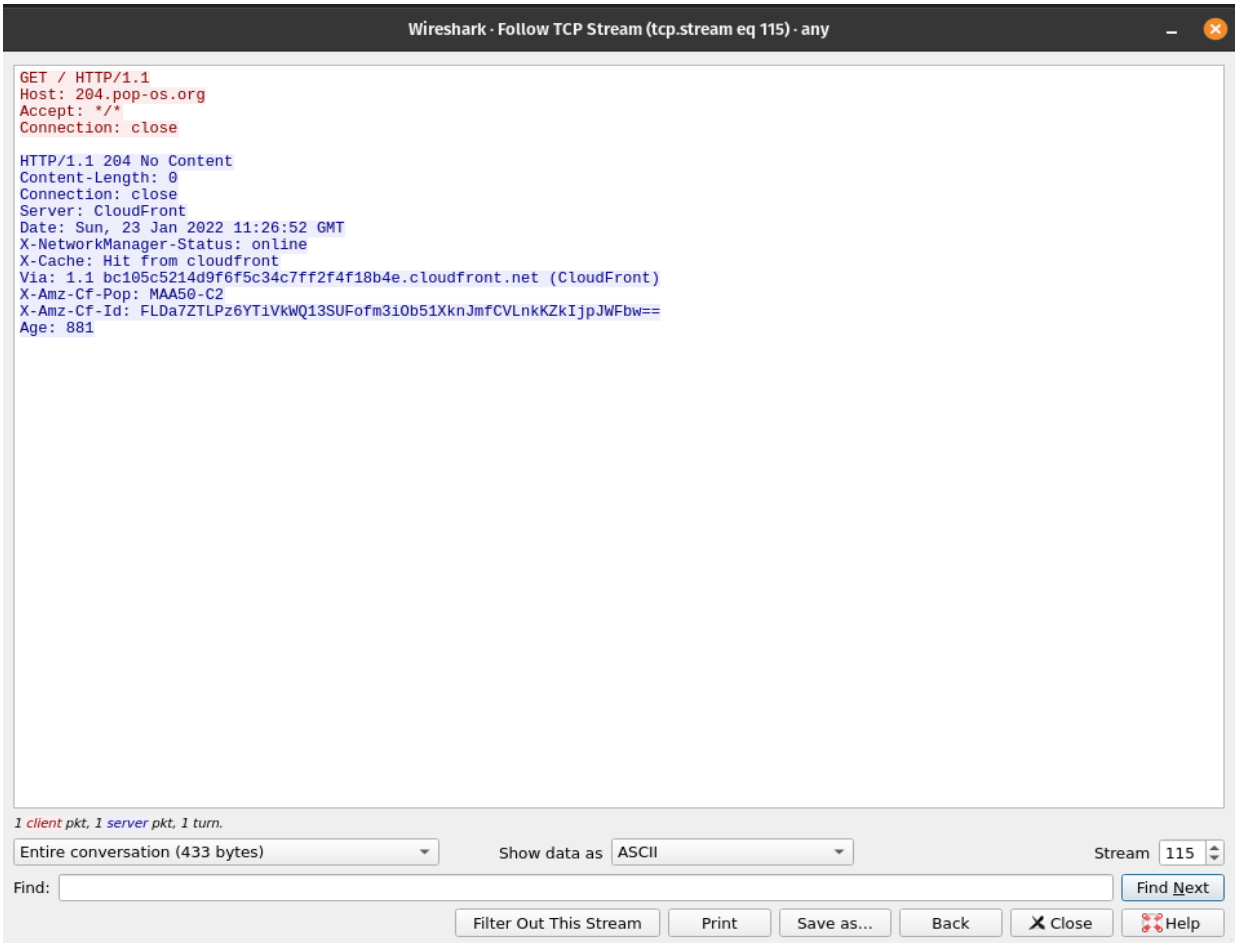
Step 4: Analyze the HTTP request and response and complete the table below.

HTTP Request		HTTP Response	
Get	/HTTP/1.1	Server	ECS (oxr/8323)
Host	Ocsp.sca1b.amazontrust.com	Content-Type	Application/ocsp-request
User-Agent	Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:96.0) Gecko/20100101 Firefox/96.0	Date	Sun,23 Jan 2022 11:41:36 GMT
Accept-Language	En-US,en;q=0.5	Location	<NOT SPECIFIED>
Accept-Encoding	Qzip, deflate	Content-Length	471
Connection	Keep-alive	Connection	Keep-alive

Using Wireshark's Follow TCP Stream

Step 1: Make sure the filter is blank. Right-click any packet inside the Packet List Pane, then select 'Follow TCP Stream'. For demo purpose, a packet containing the HTTP GET request "GET / HTTP / 1.1" can be selected.

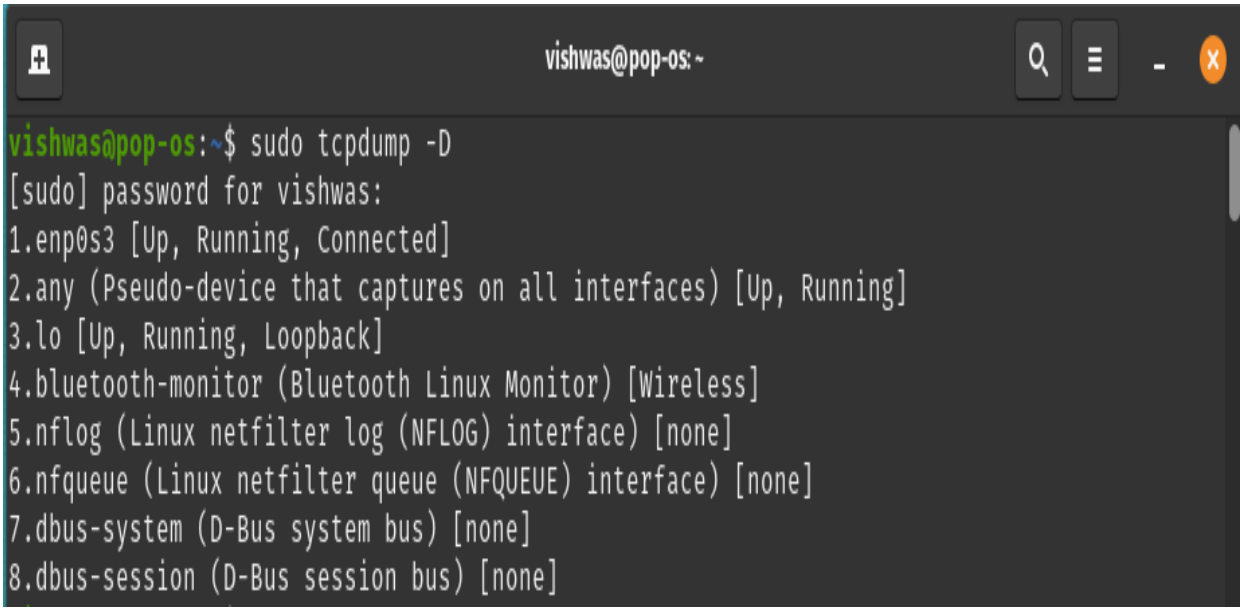
Step 2: Upon following a TCP stream, screenshot the whole window.



Task 4: Capturing packets with tcpdump

Step 1: Use the command **tcpdump -D** to see which interfaces are available for capture.

sudo tcpdump -D

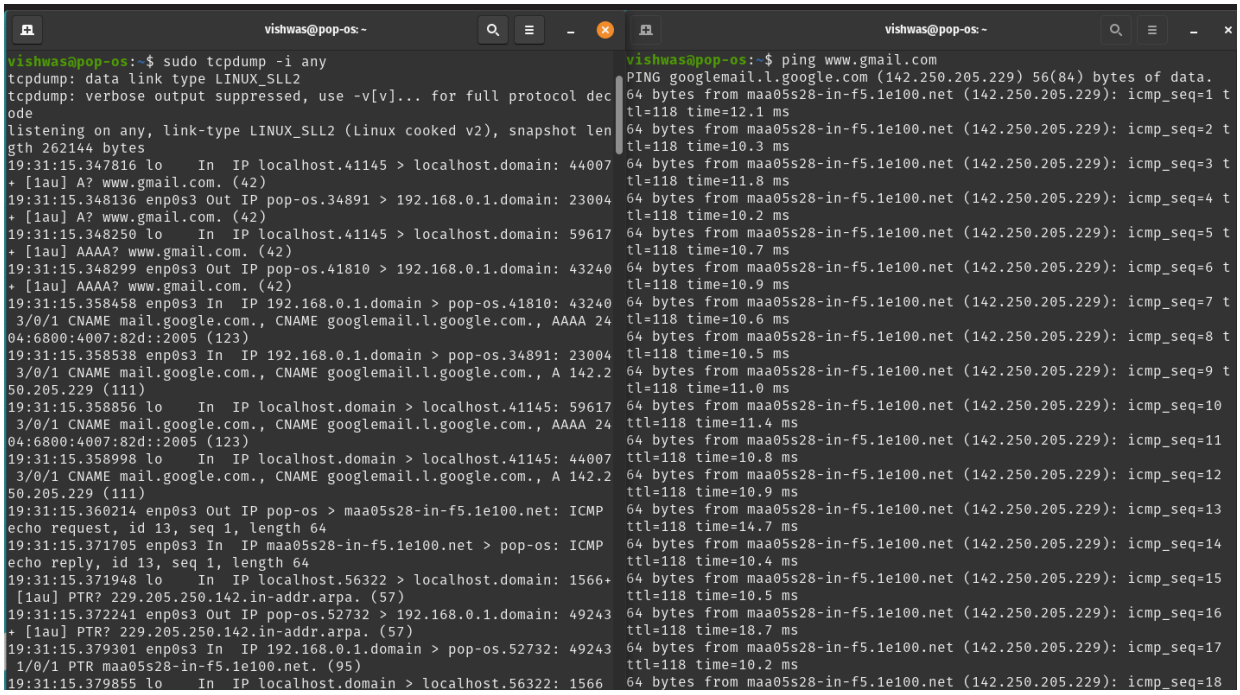


Step 2: Capture all packets in any interface by running this command:

sudo tcpdump -i any

NAME:VISHWAS M
SRN:PES2UG20CS390

SEC:F
SUB: CN LAB



Note: Perform some pinging operation while giving above command. Also type www.google.com in browser.

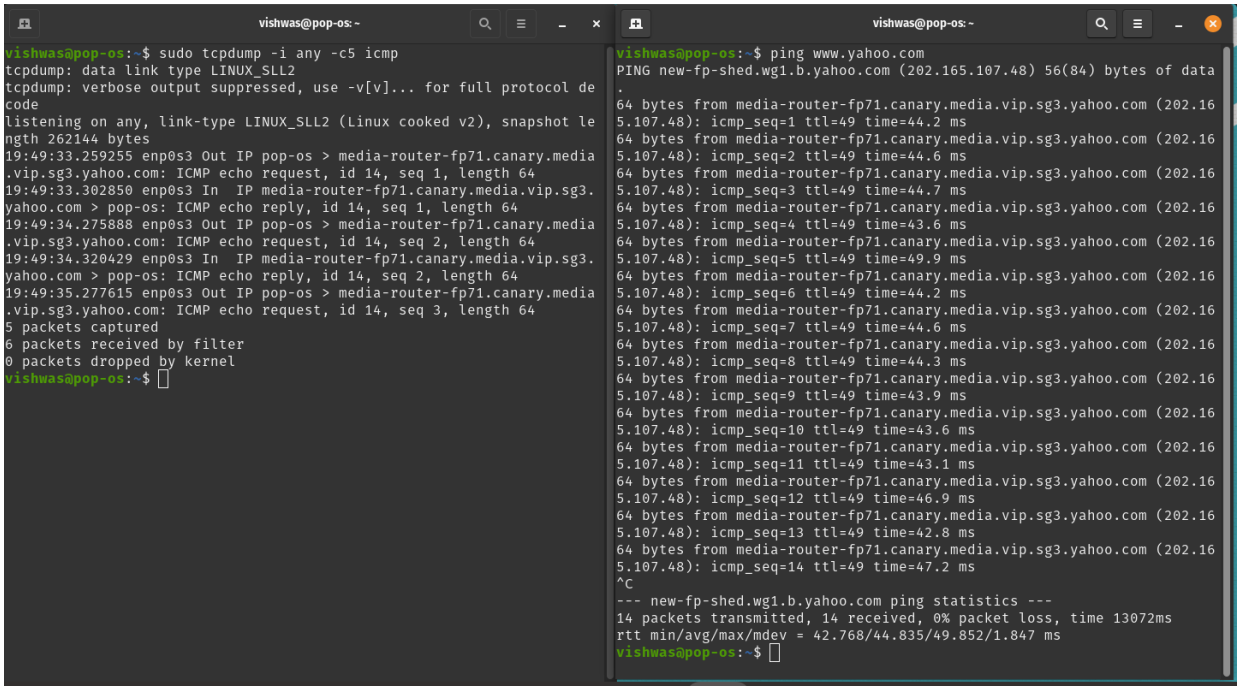
Observation

Step 3: Understand the output format.

The above command is used to capture all the packets from all the interfaces. ICMP, UDP and TCP are the main packets that are visible in the above screenshot. The timestamp followed by the link level headers, then by ARP/RARP packets if any, Then by IPv4 packets if any, followed by TCP packets. The sequence numbers and the length finish defining the outputs.

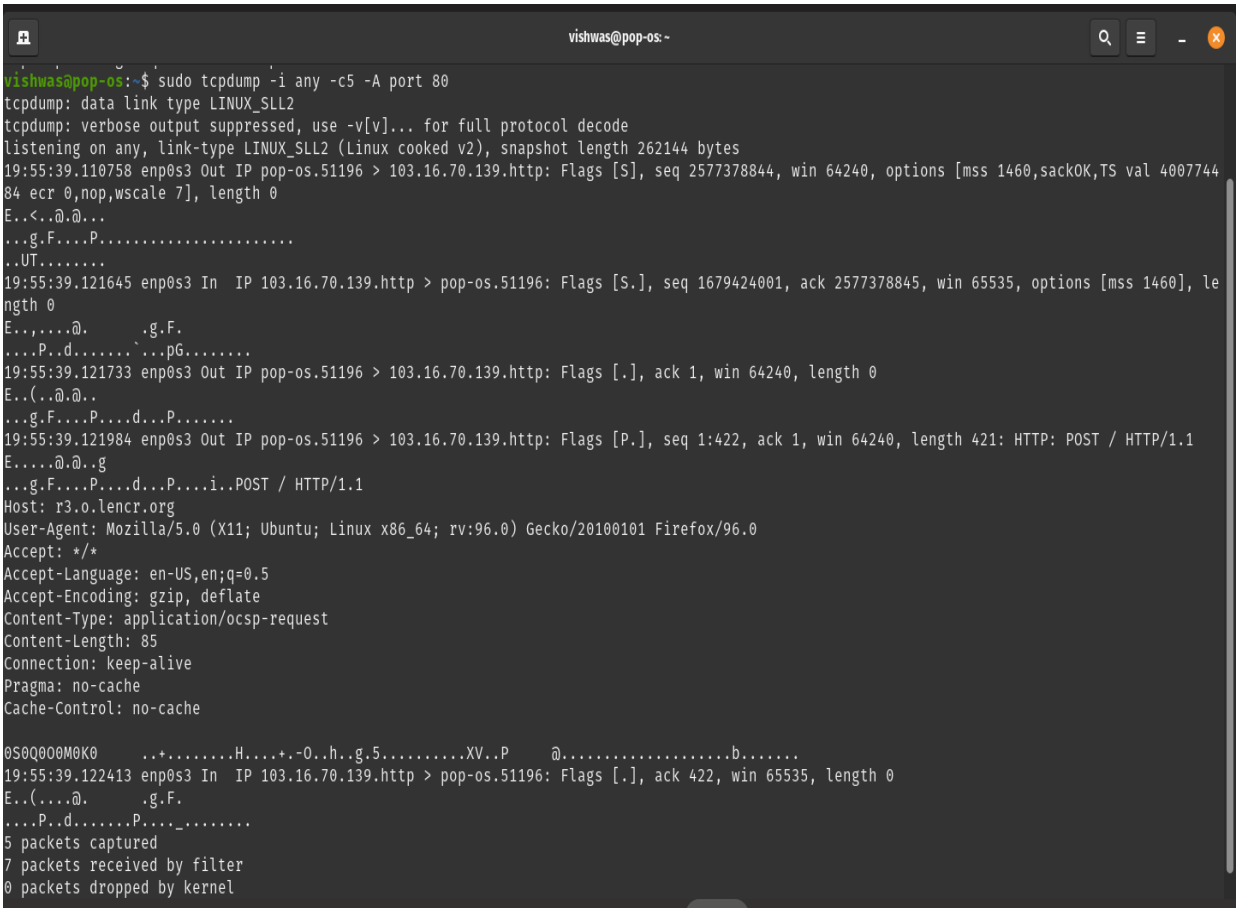
Step 4: To filter packets based on protocol, specifying the protocol in the command line. For example, capture ICMP packets only by using this command:

sudo tcpdump -i any -c5 icmp



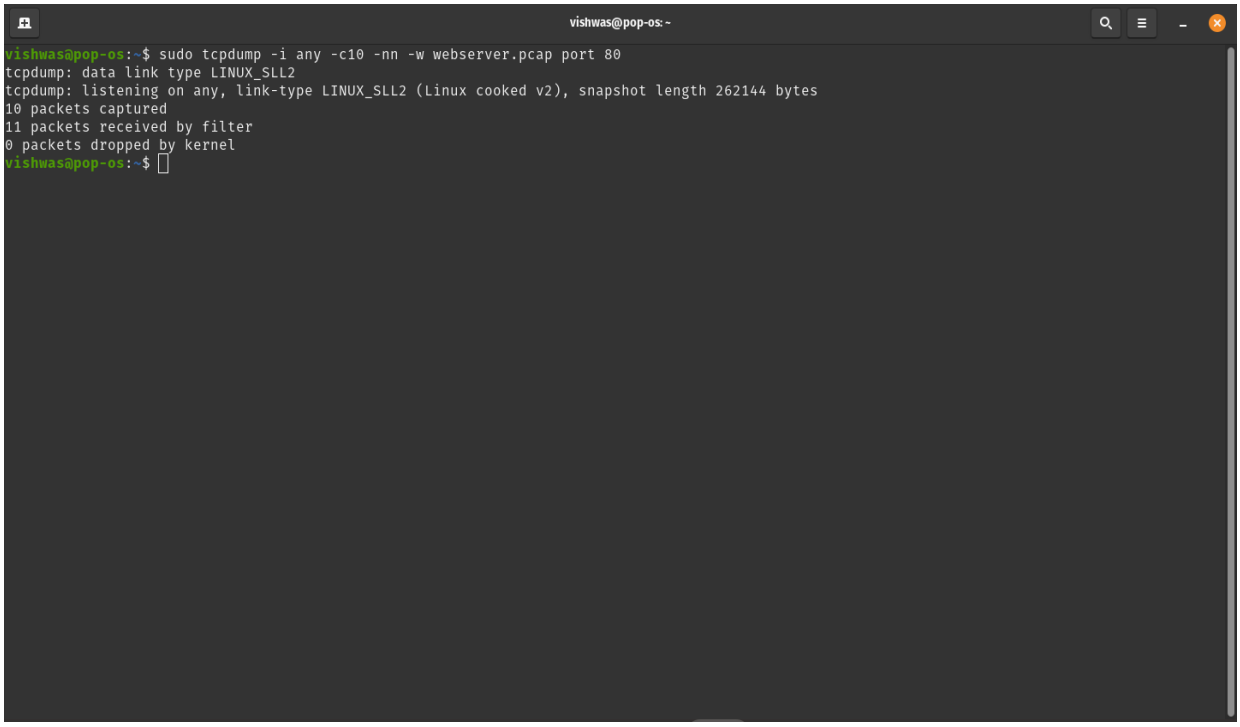
Step 5: Check the packet content. For example, inspect the HTTP content of a web request like this:

sudo tcpdump -i any -c10 -nn -A port 80



Step 6: To save packets to a file instead of displaying them on screen, use the option -w:

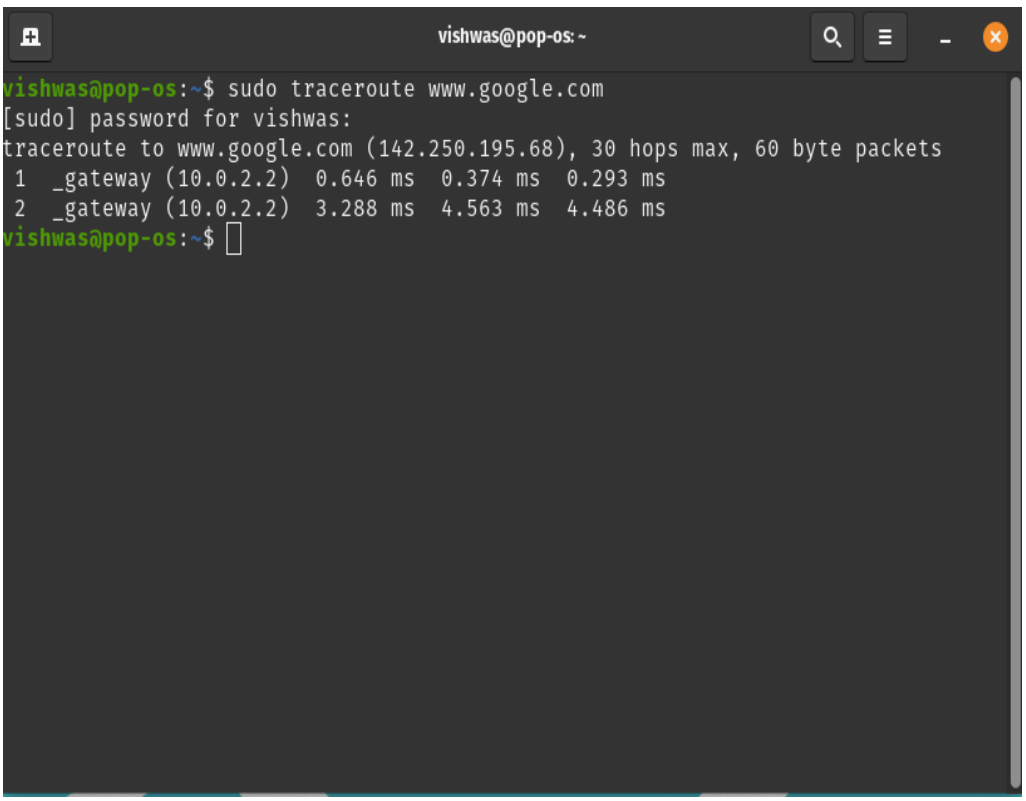
sudo tcpdump -i any -c10 -nn -w webserver.pcap port 80



Task 5: Perform Traceroute checks

Step 1: Run the traceroute using the following command.

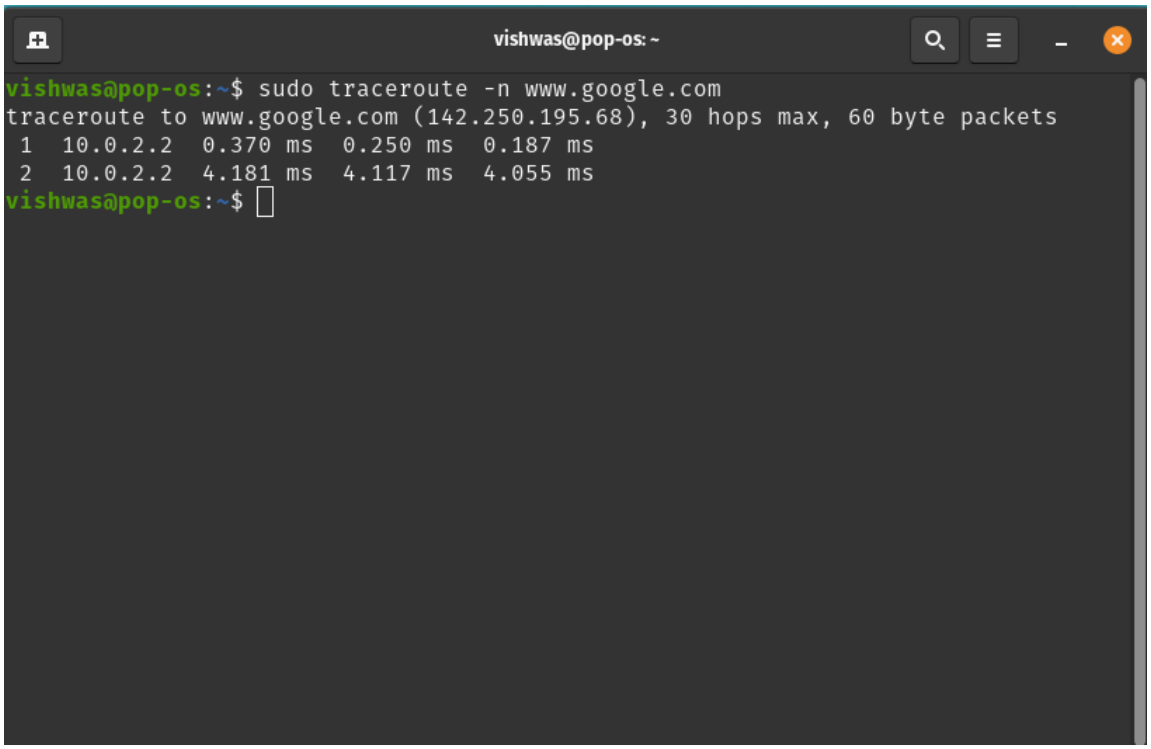
sudo traceroute www.google.com



Step 2: Analyze destination address of google.com and no. of hops

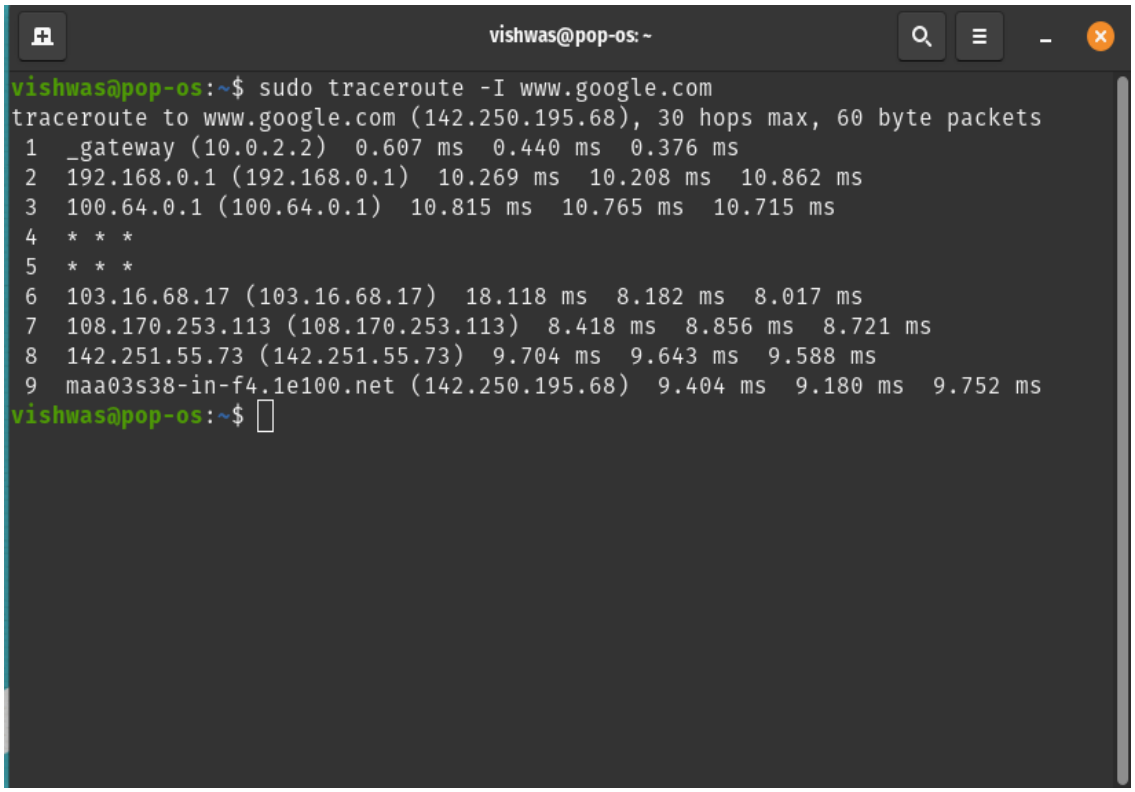
Step 3: To speed up the process, you can disable the mapping of IP addresses with hostnames by using the *-n* option

sudo traceroute -n www.google.com



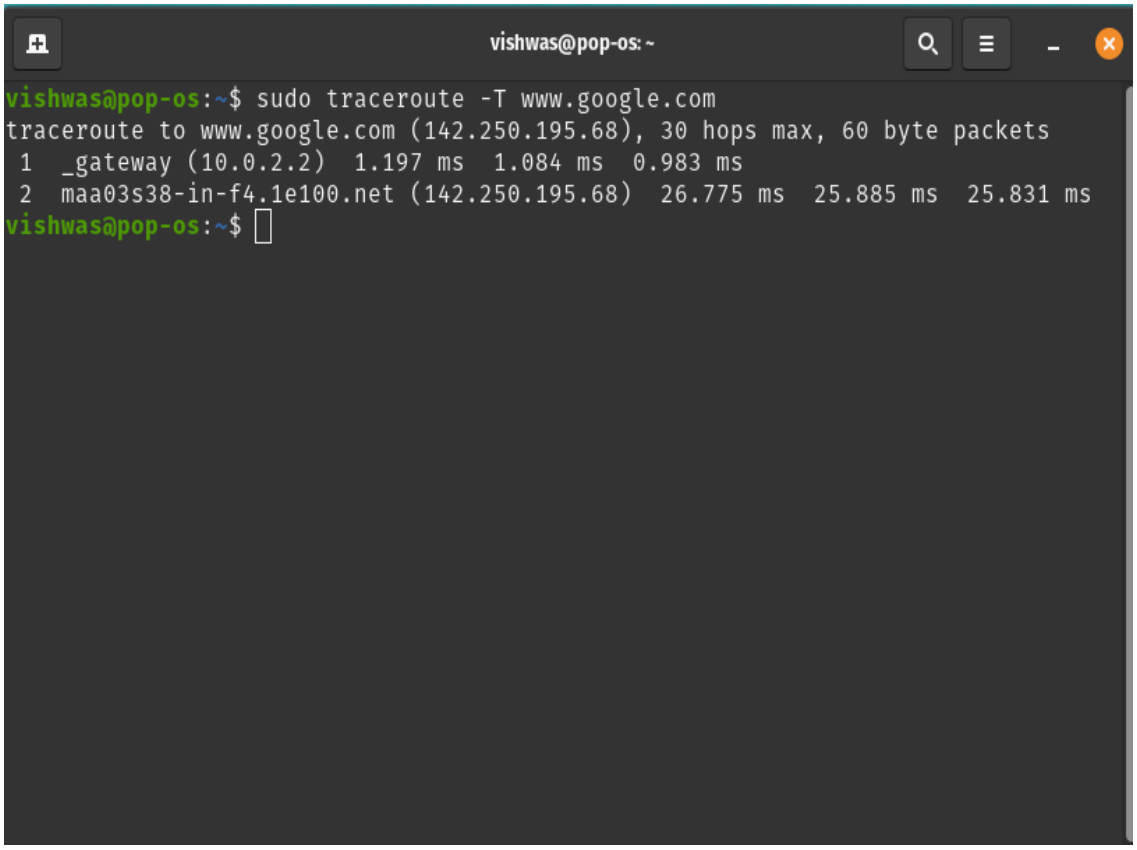
Step 4: The -I option is necessary so that the traceroute uses ICMP.

sudo traceroute -I www.google.com



Step 5: By default, traceroute uses icmp (ping) packets. If you'd rather test a TCP connection to gather data more relevant to web server, you can use the -T flag.

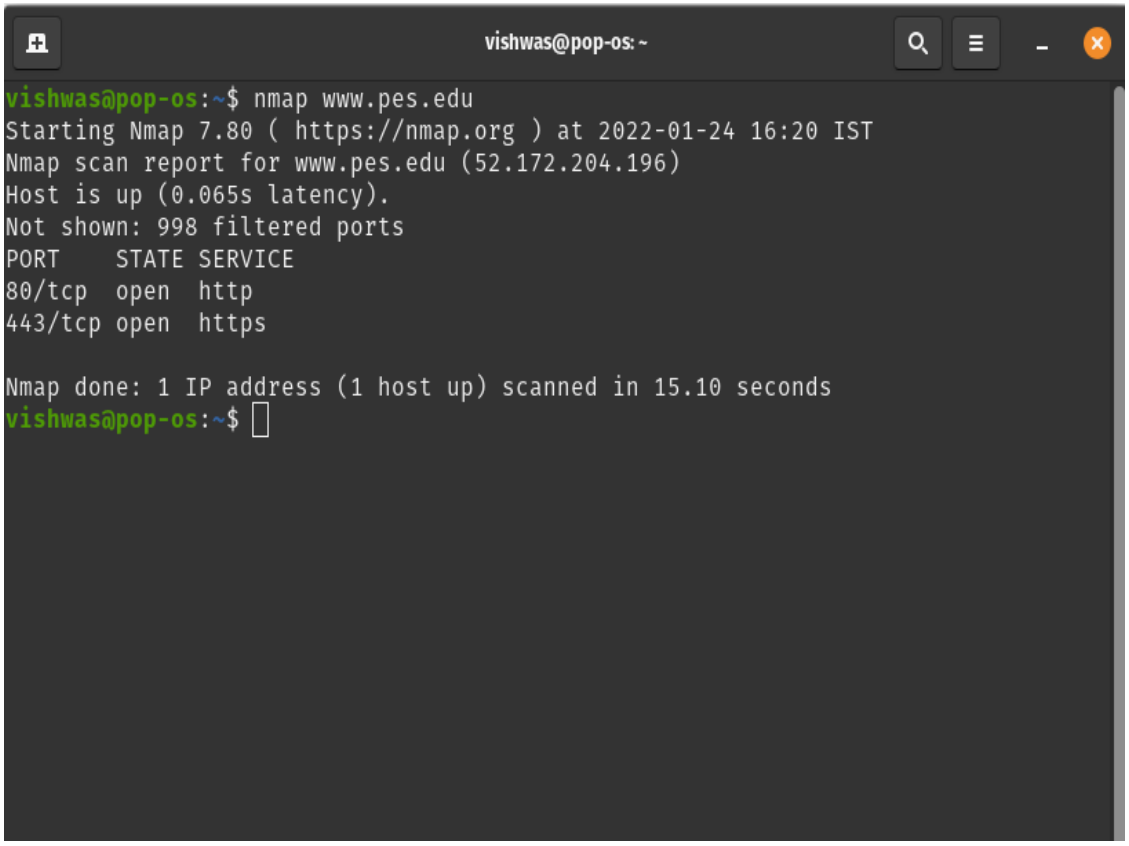
sudo traceroute -T www.google.com



Task 6: Explore an entire network for information (Nmap)

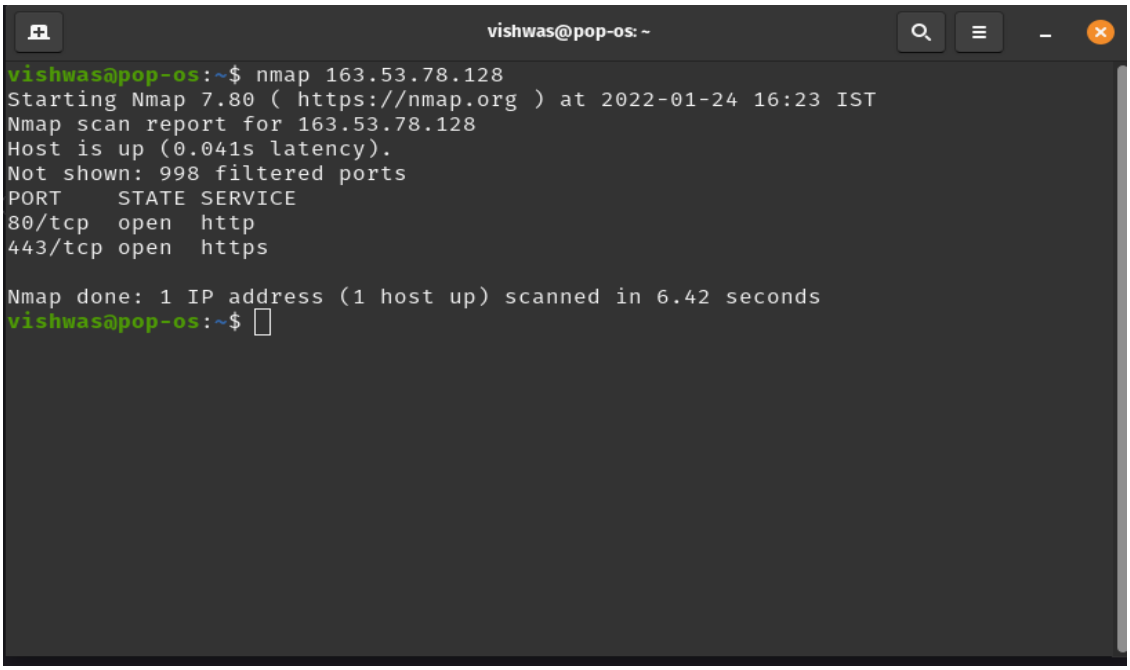
Step 1: You can scan a host using its host name or IP address, for instance.

nmap www.pes.edu



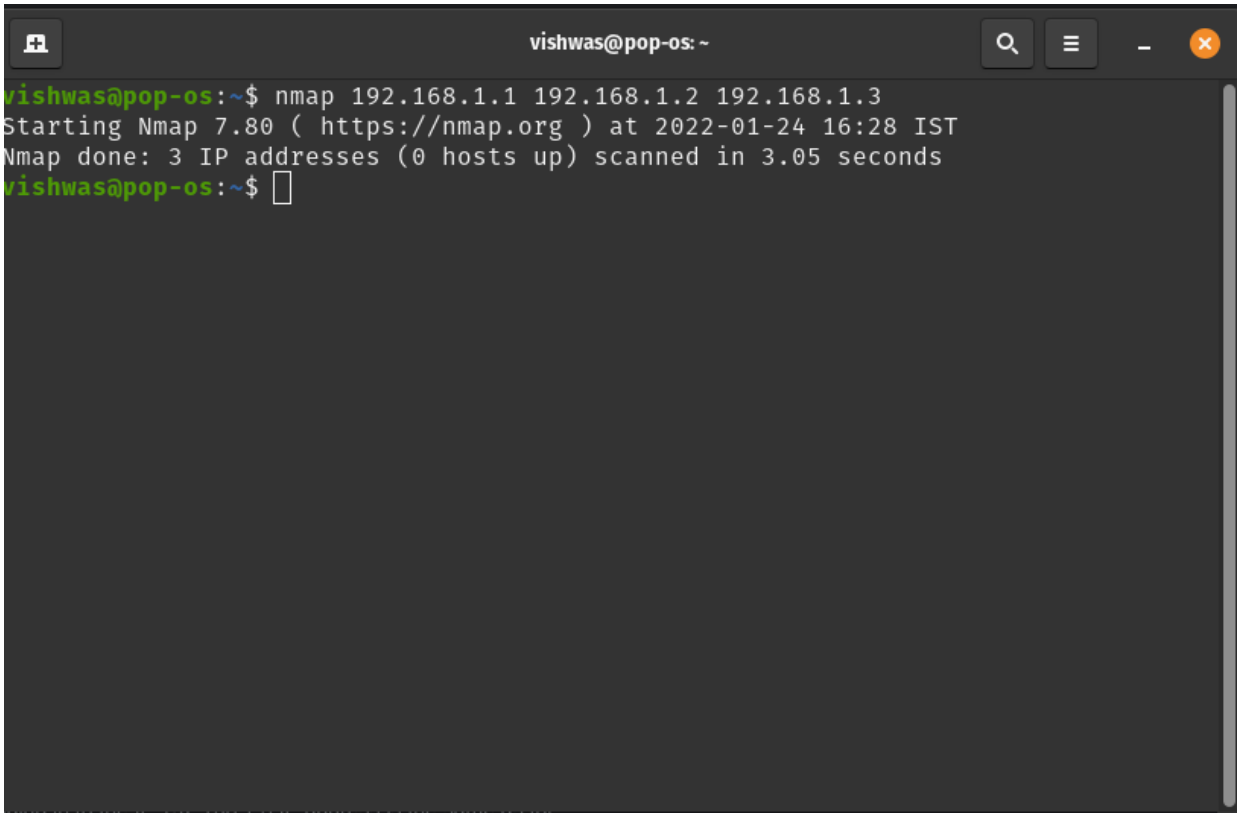
Step 2: Alternatively, use an IP address to scan.

nmap 163.53.78.128



Step 3: Scan multiple IP address or subnet (IPv4)

nmap 192.168.1.1 192.168.1.2 192.168.1.3



NAME:VISHWAS M SEC:F
SRN:PES2UG20CS390 SUB: CN LAB

Questions on above observations:

1) Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server?

   Ans**: 1.1. The version of the server is 1.1 as well.**

2) When was the HTML file that you are retrieving last modified at the server?
   Ans: **Sun,23 Jan 2022 11:41:36 GMT**

3) How to tell ping to exit after a specified number of ECHO_REQUEST packets?

   Ans: **$ ping -c <number of packets> <url>**

4)How will you identify remote host apps and OS?

   Ans: **Simply scan the entire subnet.**

   Eg: $ nmap -sP 10.0.4.*