**COMPUTER NETWORKS LAB**

**WEEK 1**

**NAME: VISHWA MEHUL MEHTA**

**SRN: PES2UG20CS389**

**SECTION: F**

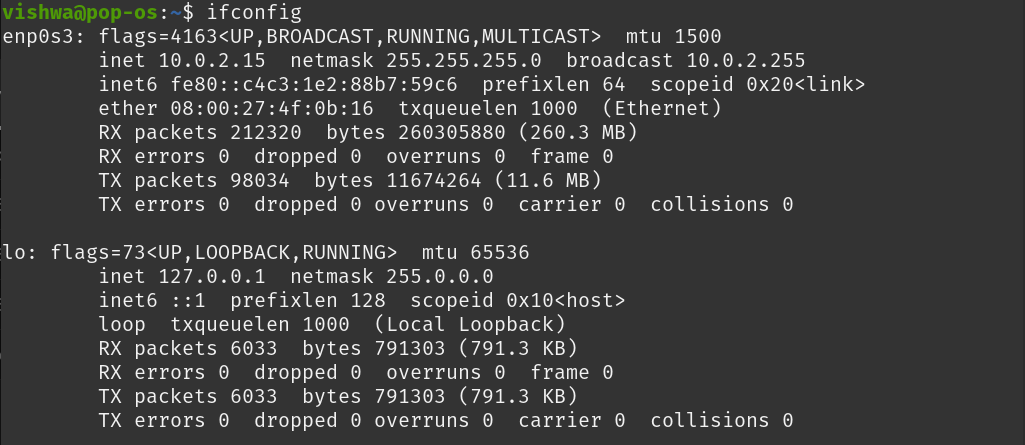
**DATE: 21/01/2022**

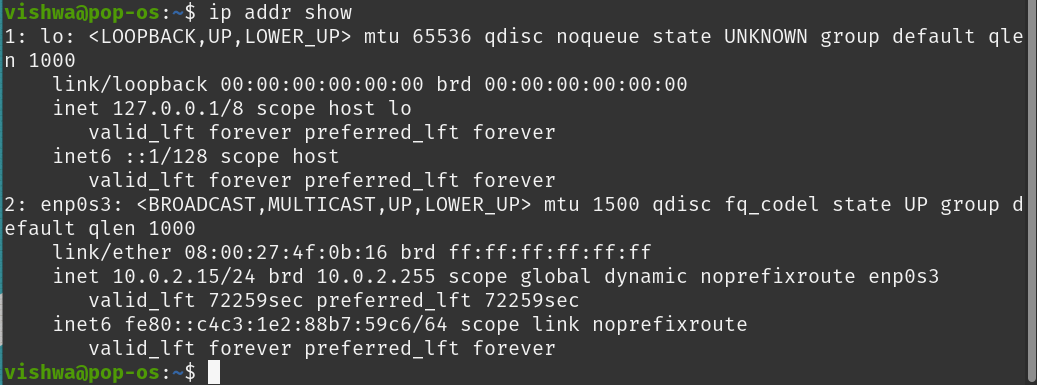
**Study and understand the basic networking tools - Wireshark, Tcpdump, Ping, Traceroute and Netcat.**

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

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

**ifconfig** (or) **ip addr show**

****

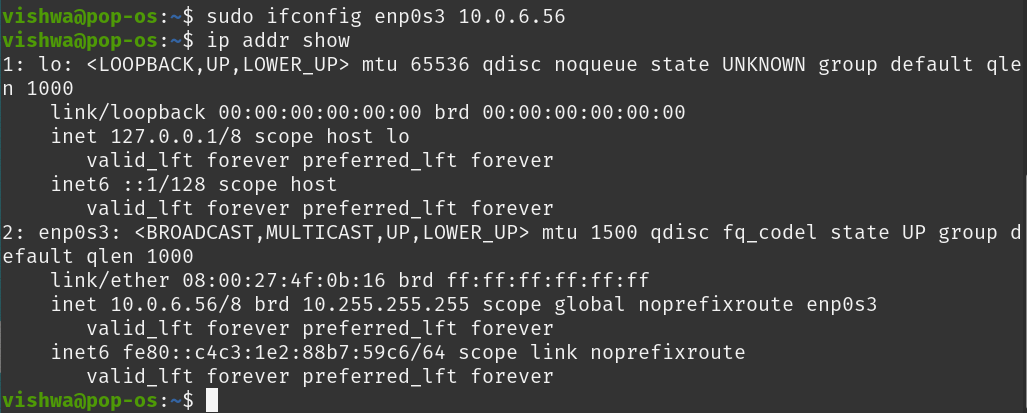
****

Analyze and fill the following table:

**ip address table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Interface name** | **IP address**  **(IPv4 / IPv6)** | **MAC address** |  |
| **lo** | **127.0.0.1** | **00:00:00:00:00:00** | **Loop back device** |
| **enp0s3** | **10.0.2.15** | **08:00:27:4f:0b:16** | **Ethernet** |
| **WLAN** | **-** | **-** | **-** |

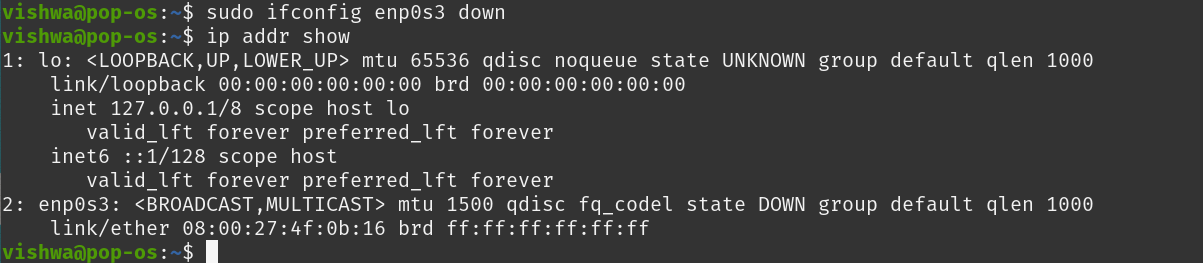
**Step 2:** To assign an IP address to an interface, use the following command. **sudo ifconfig interface\_name 10.0.your\_section.your\_sno netmask 255.255.255.0** (or) **sudo ip addr add 10.0.your\_section.your\_sno /24 dev interface\_name**

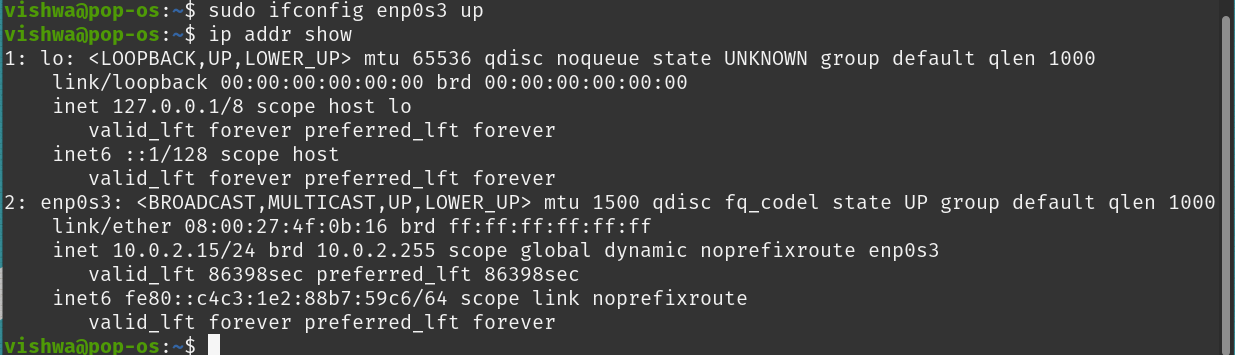
****

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

**sudo ifconfig interface\_name down**

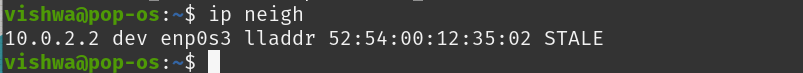
**sudo ifconfig interface\_name up**

****

****

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

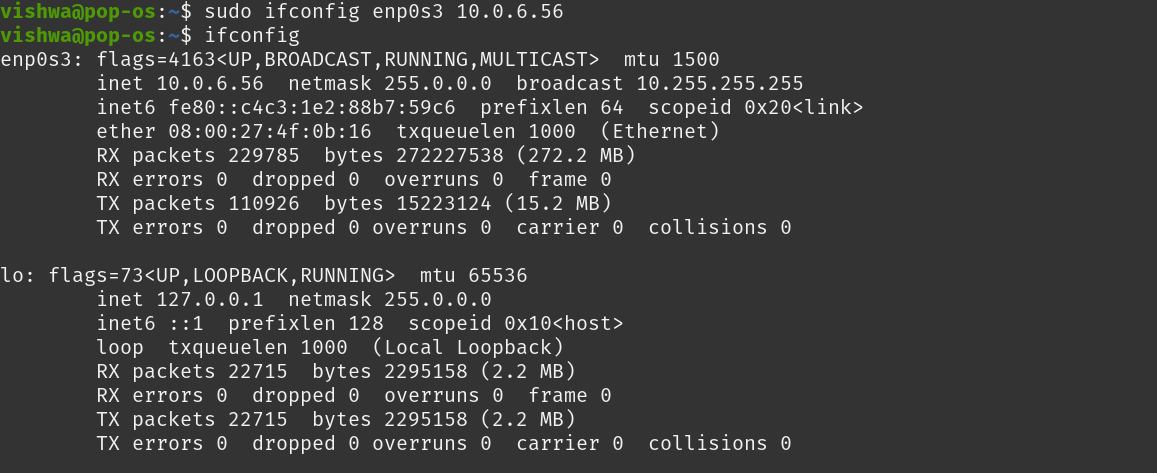
**ip neigh**

****

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

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

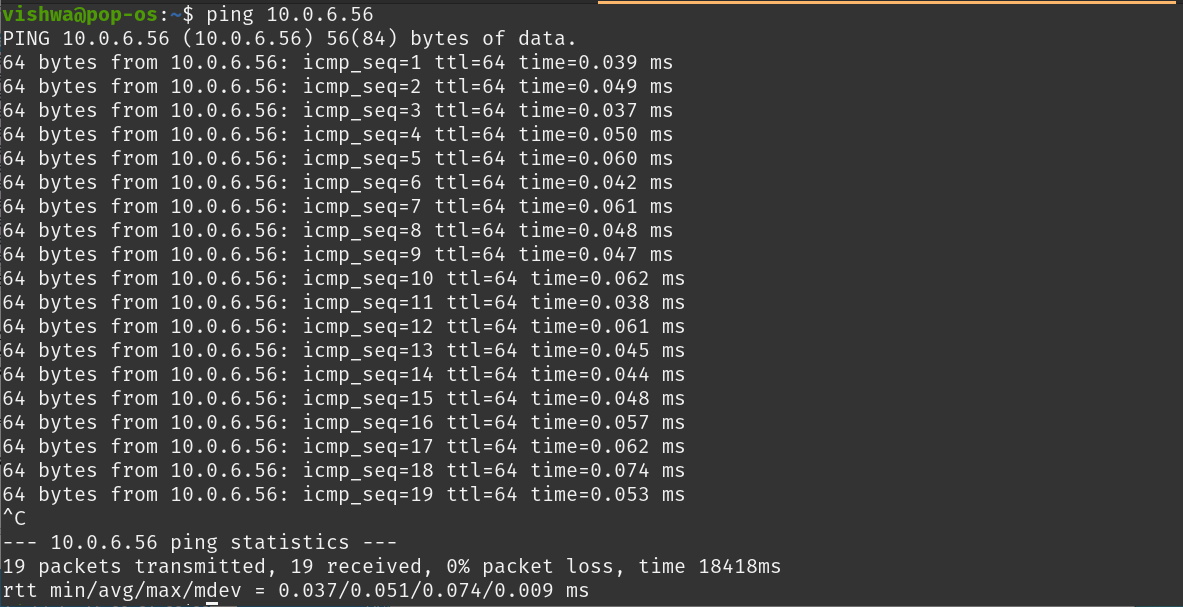
Note: IP address of your system should be 10.0.your\_section.your\_sno.

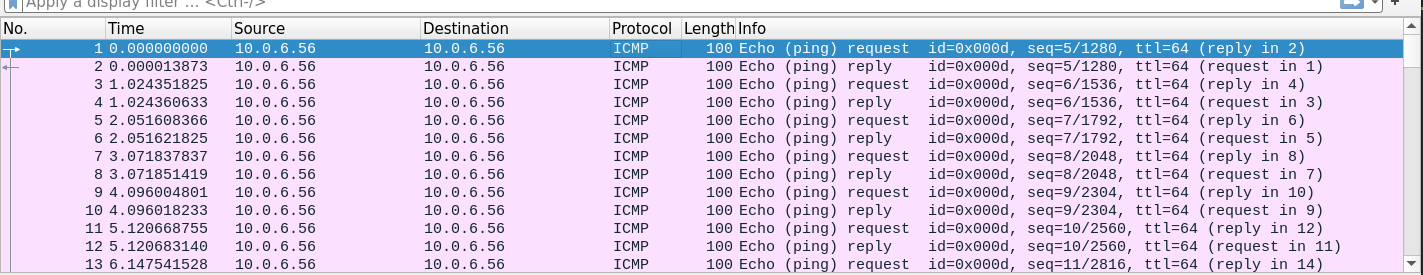


**Step 2:** Launch Wireshark and select ‘any’ interface



**Step 3:** In terminal, type **ping 10.0.your\_section.your\_sno**

****

****

**Observations to be made**

**Step 4:** Analyze the following in Terminal

• TTL : **64**

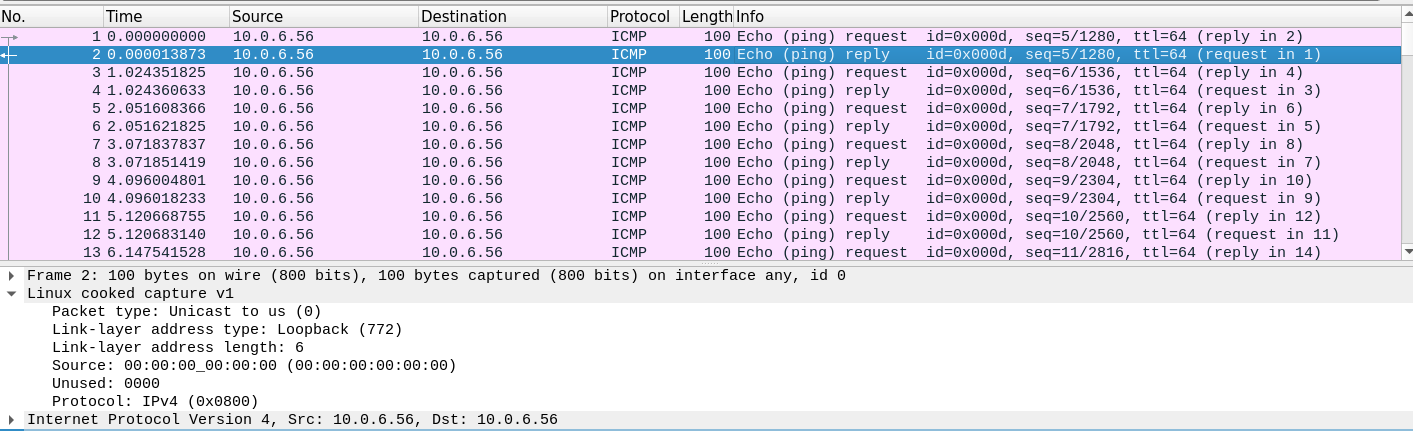
• Protocol used by ping : **ICMP (Internet Control Message Protocol)**

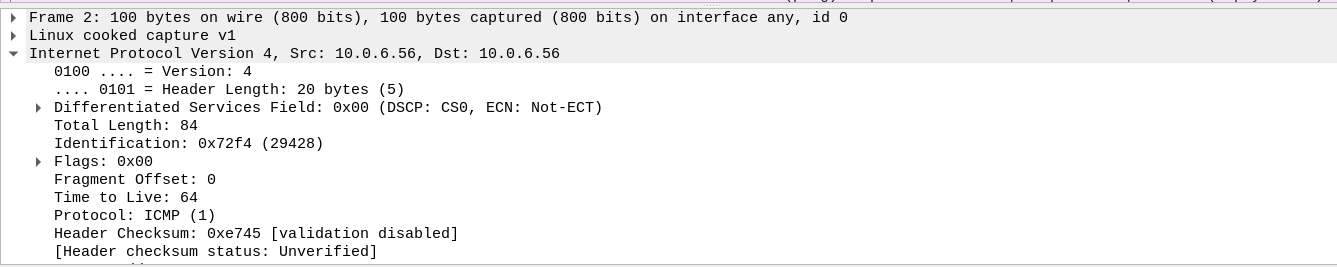
• Time : **18418ms**

**Step 5:** Analyze the following in Wireshark

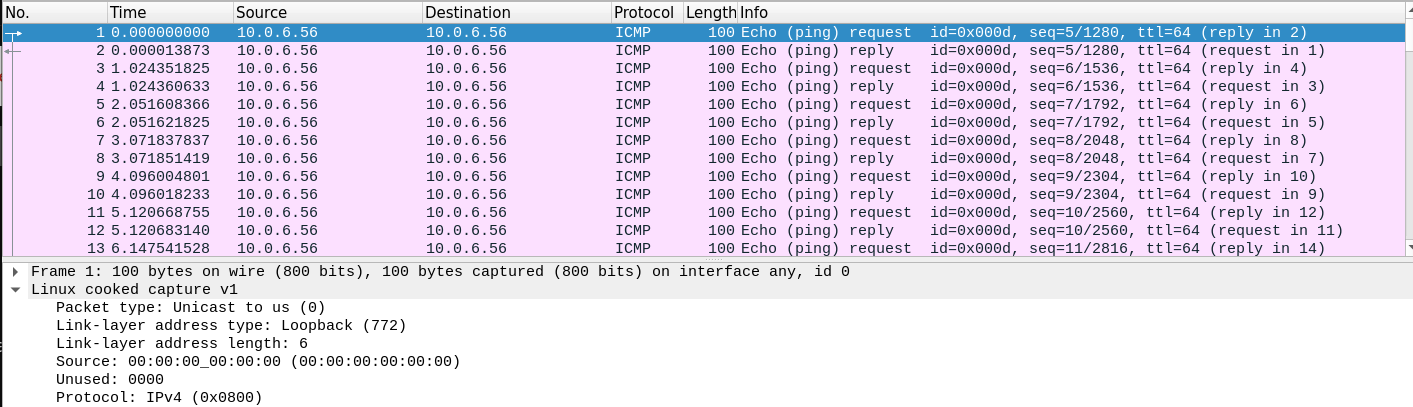
On Packet List Pane, select the first echo packet on the list. On Packet Details Pane, click on each of the four “+” to expand the information. Analyze the frames with the first echo request and echo reply and complete the table below.

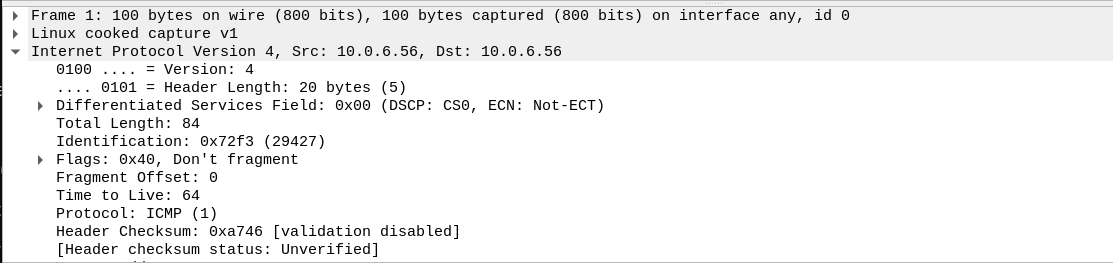
**First echo request:**

****

****

**First echo reply:**

****

****

|  |  |  |
| --- | --- | --- |
| **Details** | **First Echo Request** | **First Echo Reply** |
| Frame Number | **1** | **2** |
| Source IP address | **10.0.6.56** | **10.0.6.56** |
| Destination IP address | **10.0.6.56** | **10.0.6.56** |
| 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** | **64** |

**Task 3: HTTP PDU Capture**

**Using Wireshark’s Filter feature**

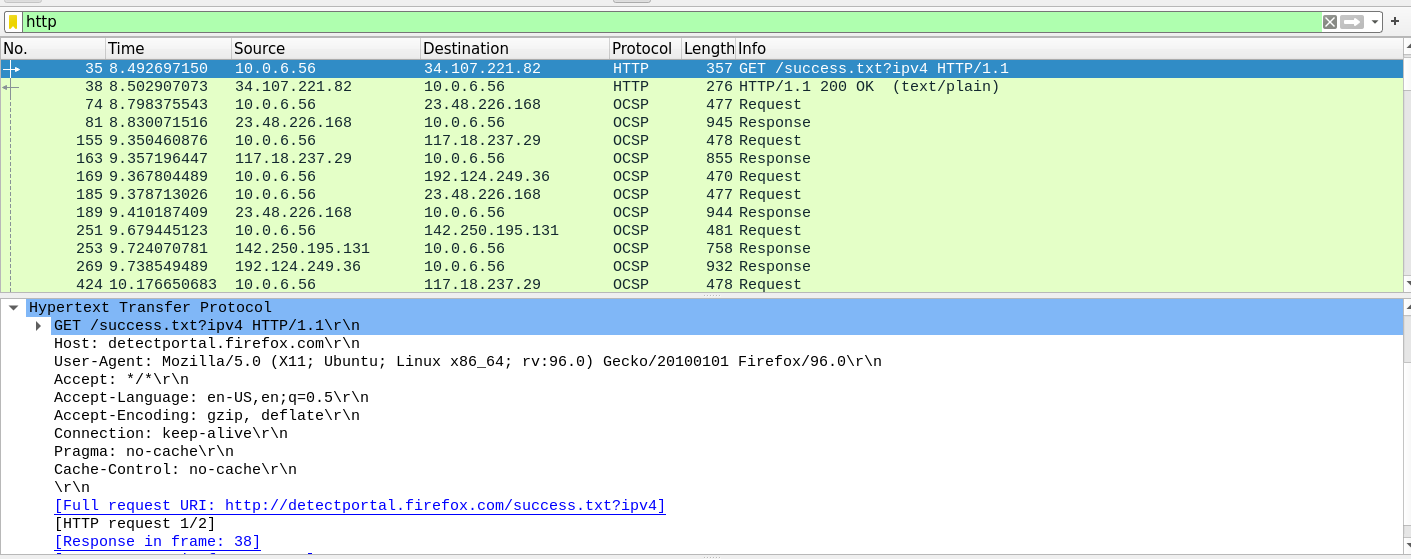
**Step 1:** Launch Wireshark and select ‘any’ interface. On the Filter toolbar, type-in ‘http’ and press enter

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

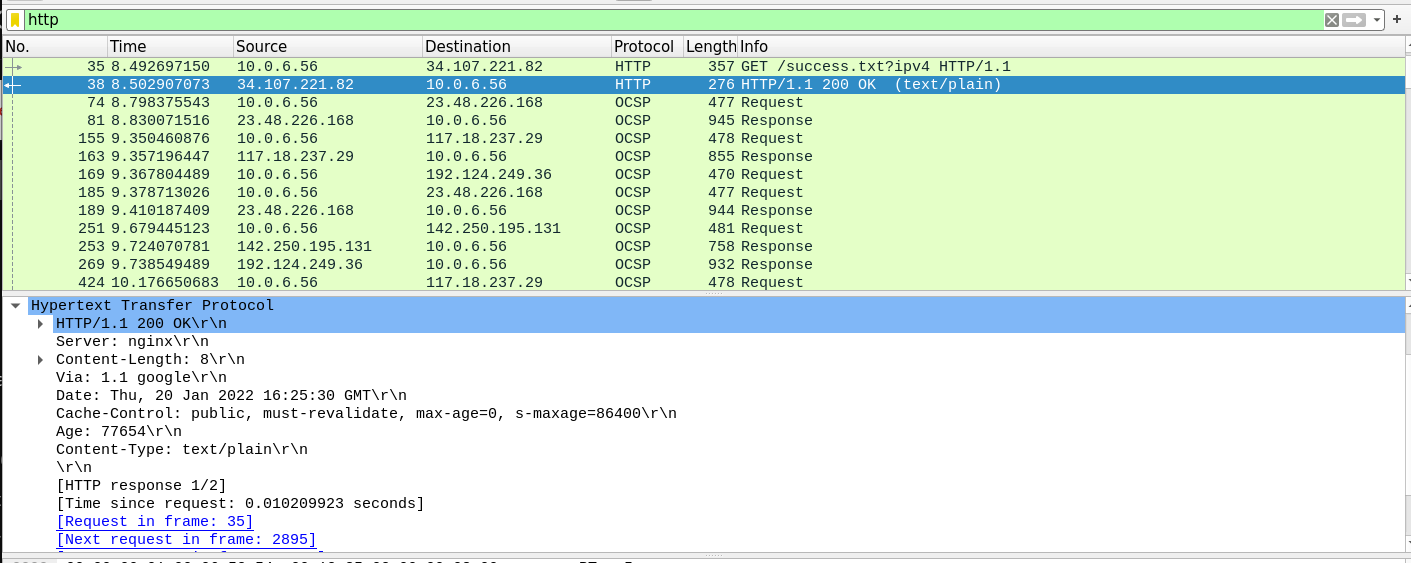
**Observations to be made**

**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:

**HTTP First Request:**



**HTTP First Response:**



|  |  |  |
| --- | --- | --- |
| **Details** | **First Echo Request** | **First Echo Reply** |
| Frame Number | **35** | **38** |
| Source Port | **38282** | **80** |
| Destination Port | **80** | **38282** |
| Source IP address | **10.0.6.56** | **34.107.221.82** |
| Destination IP address | **34.107.221.82** | **10.0.6.56** |
| Source Ethernet Address | **08:00:27:4f:0b:16** | **52:54:00:12:35:02** |
| Destination Ethernet Address | **52:54:00:12:35:02** | **08:00:27:4f:0b:16** |

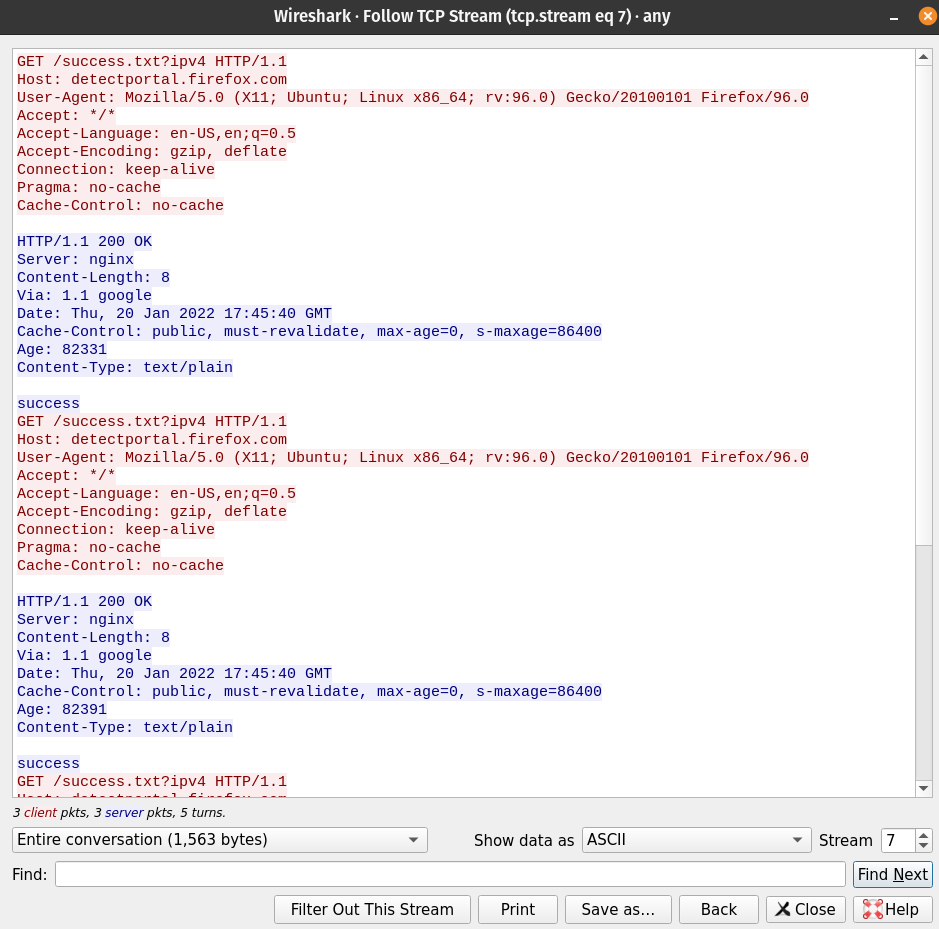
**Step 4:** Analyze the HTTP request and response and complete the table below.

|  |  |  |  |
| --- | --- | --- | --- |
| **HTTP Request** |  | **HTTP Response** |  |
| Get | **GET /Success.txt?ipv4 HTTP/1.1** | Server | **nginx\r\n** |
| Host | **detectportal.firefox\r\n** | Content-Type | **text/plain\r\n** |
| User-Agent | **Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:96.0) Gecko/20100101 Firefox/96.0\r\n** | Date | **Thu, 20 Jan 2022 16:25:30 GMT\r\n** |
| Accept-Language | **en-US,en;q=0.5\r\n** | Location | **-** |
| Accept-Encoding | **gzip, deflate\r\n** | Content-Length | **8\r\n** |
| Connection | **keep-alive\r\n** | Connection | **close\r\n** |

**Using Wireshark’s Follow TCP Stream**

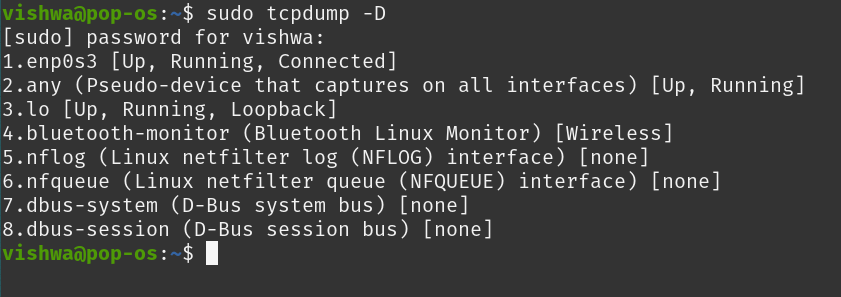
**Step 1:** Make sure the filter is blank. Right-click any packet inside the Packet List Pane, then select ‘Follow TCP Stream’. For demo purpose, a packet containing the HTTP GET request “GET / HTTP / 1.1” can be selected.

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



**Task 4: Capturing packets with tcpdump**

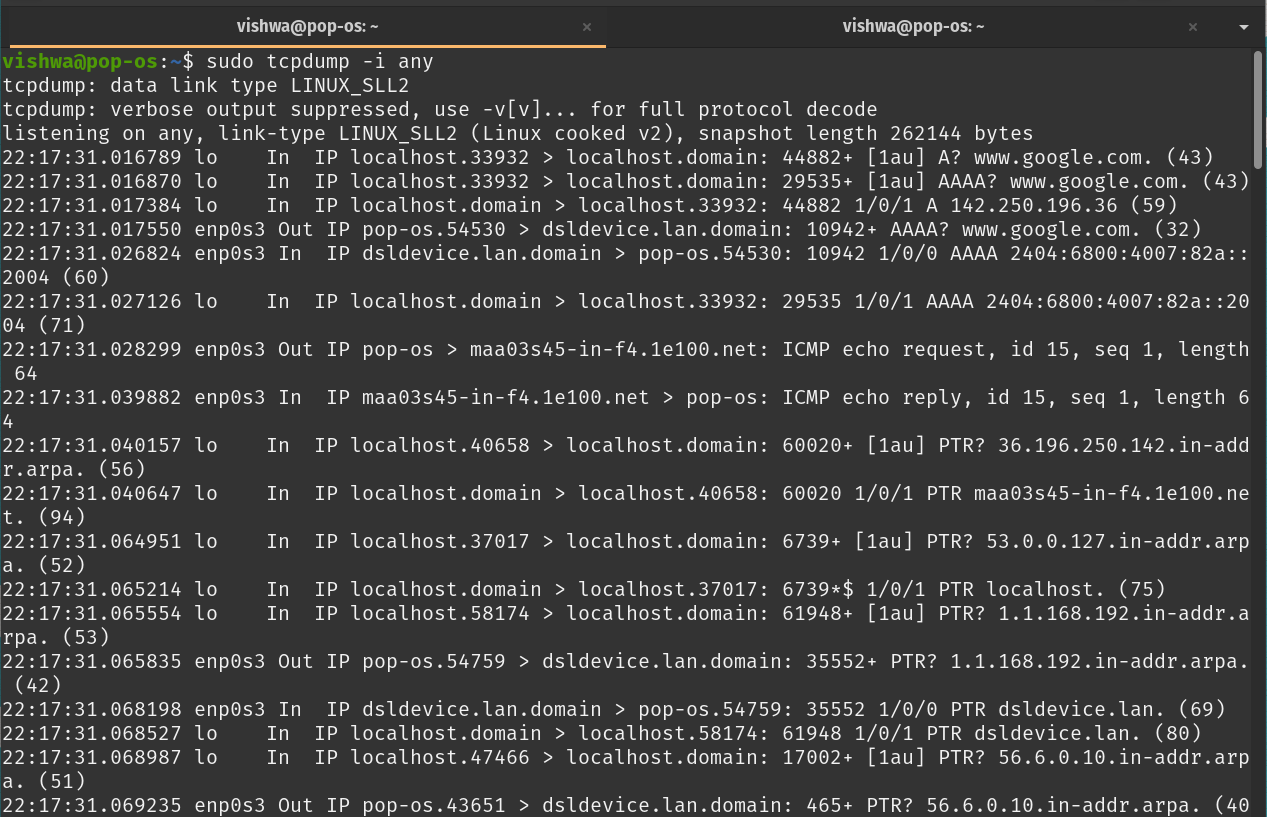
**Step 1:** Use the command **tcpdump -D** to see which interfaces are available for capture. **sudo tcpdump -D**

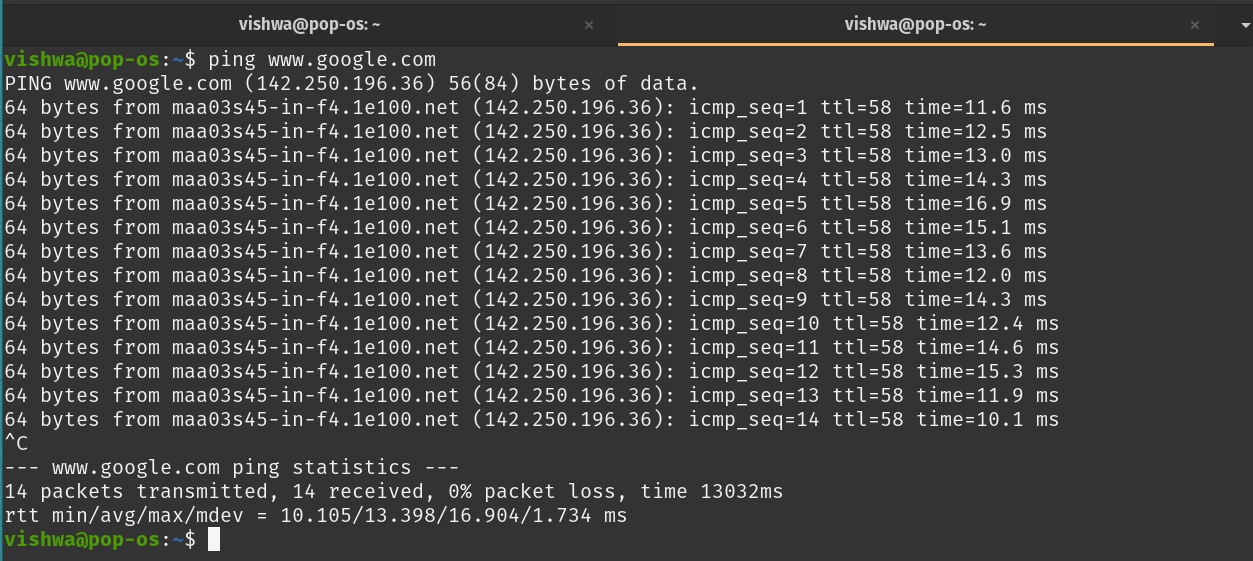
****

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

**sudo tcpdump -i any**

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



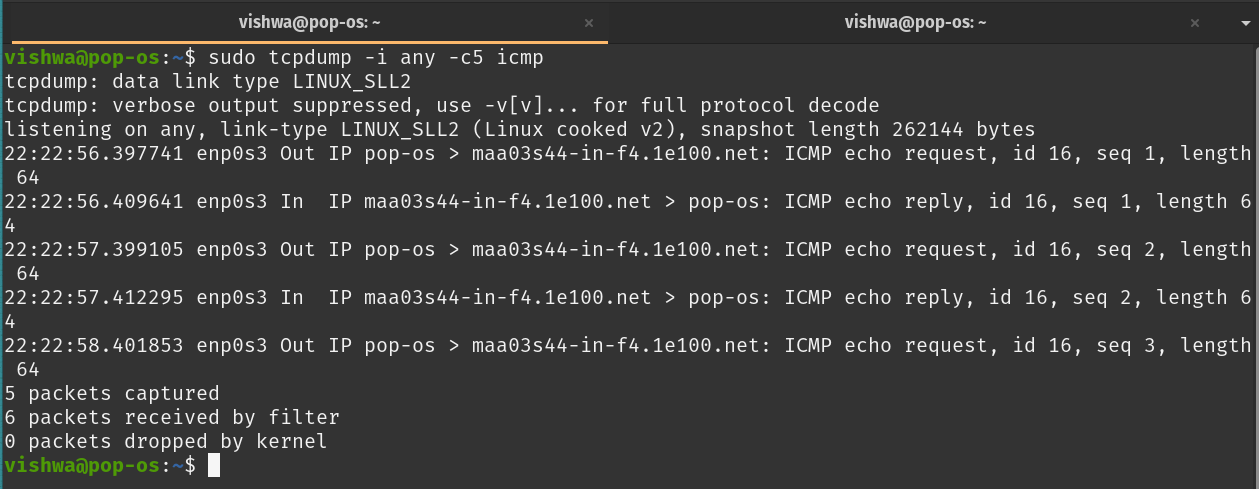


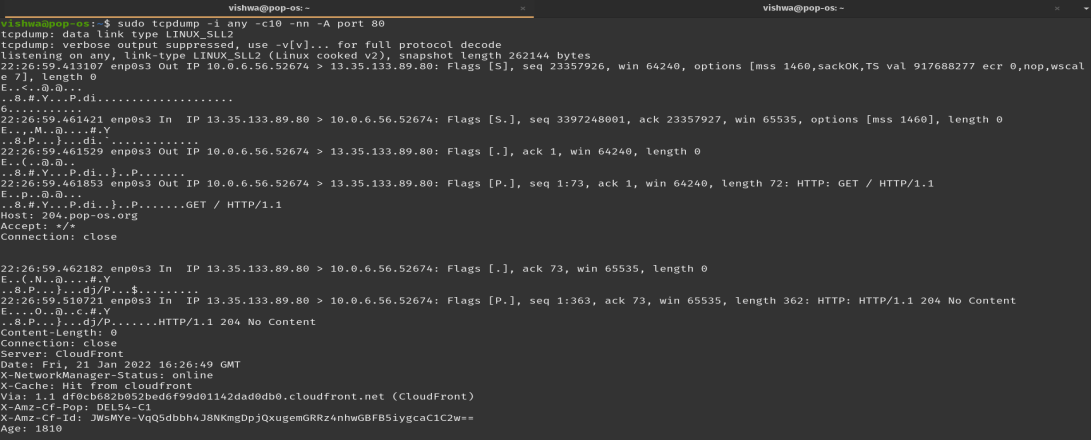
**Observation**

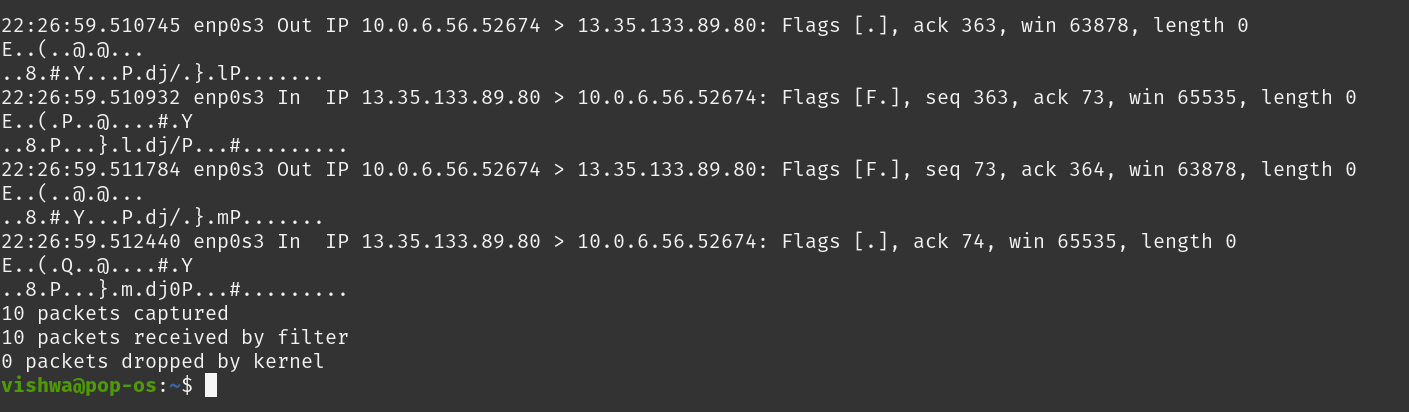
**Step 3:** Understand the output format.

**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:

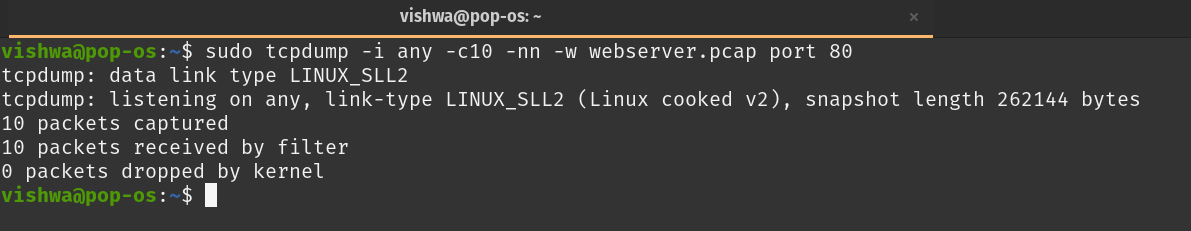
**sudo tcpdump -i any -c5 icmp**

****

**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 **

****

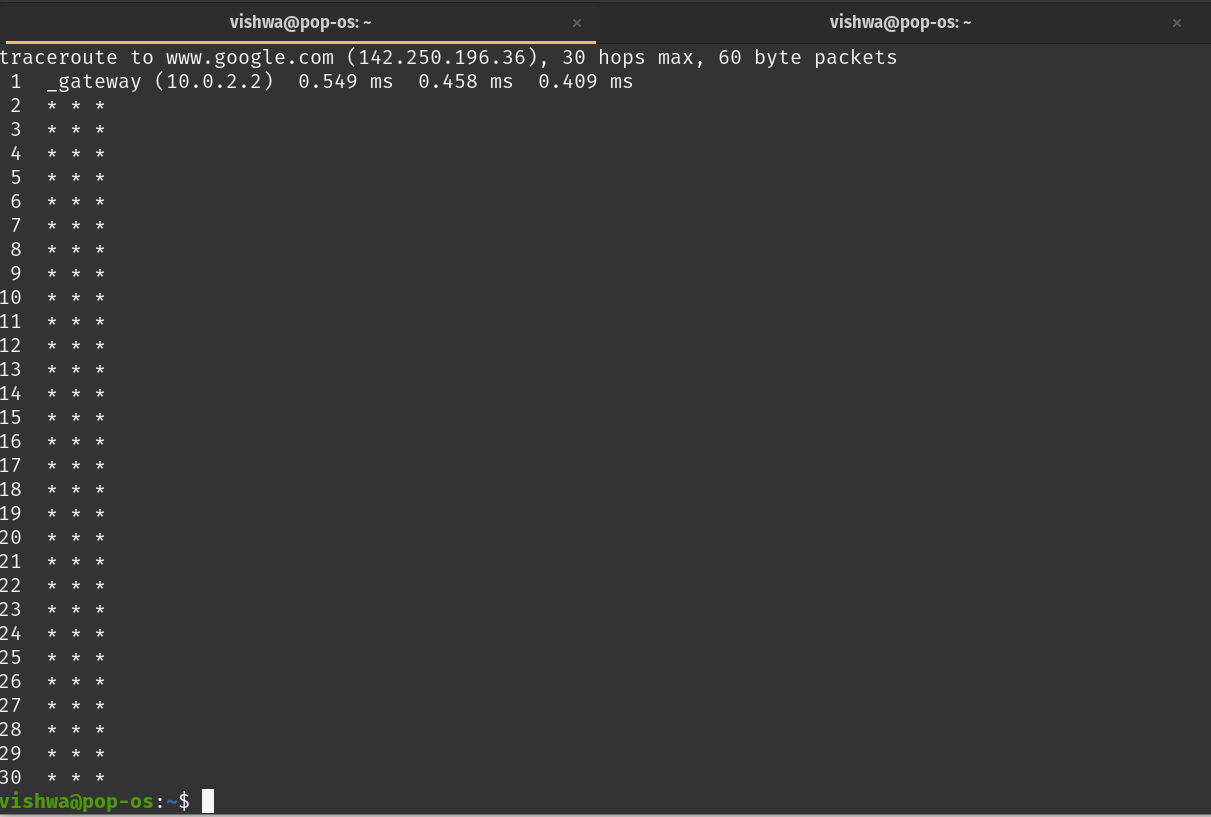
**Step 6:** To save packets to a file instead of displaying them on screen, use the option -w: **sudo tcpdump -i any -c10 -nn -w webserver.pcap port 80**

****

**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

**Destination: 142.250.196.36 No. of hops: 30**

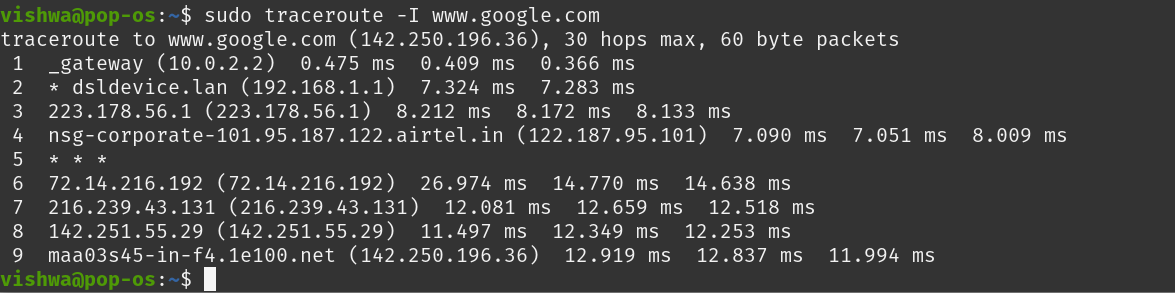
**Step 3:** To speed up the process, you can disable the mapping of IP addresses with hostnames by using the *-n* option

**sudo traceroute -n www.google.com**

****

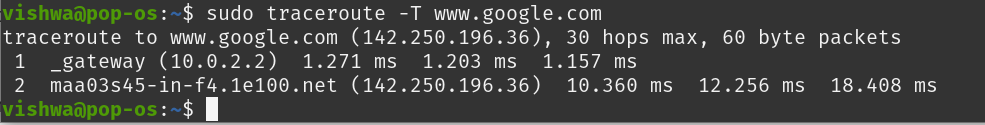
**Step 4:** The -I option is necessary so that the traceroute uses ICMP.

**sudo traceroute -I www.google.com**

****

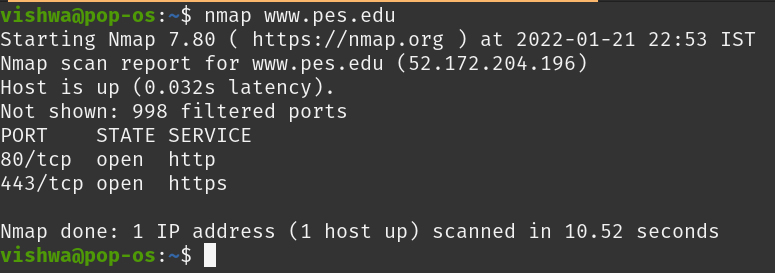
**Step 5:** By default, traceroute uses icmp (ping) packets. If you’d rather test a TCP connection to gather data more relevant to web server, you can use the -T flag.

**sudo traceroute -T www.google.com**

****

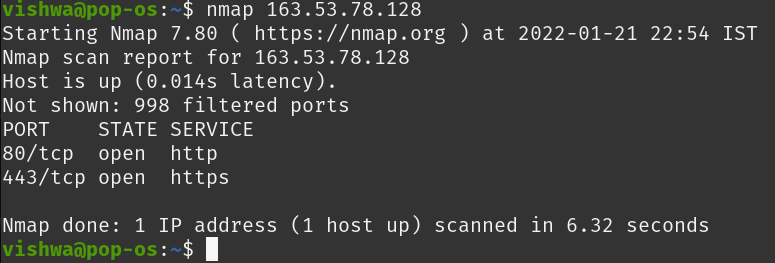
**Task 6: Explore an entire network for information (Nmap)**

**Step 1:** You can scan a host using its host name or IP address, for instance. **nmap** [**www.pes.edu**](http://www.pes.edu)

****

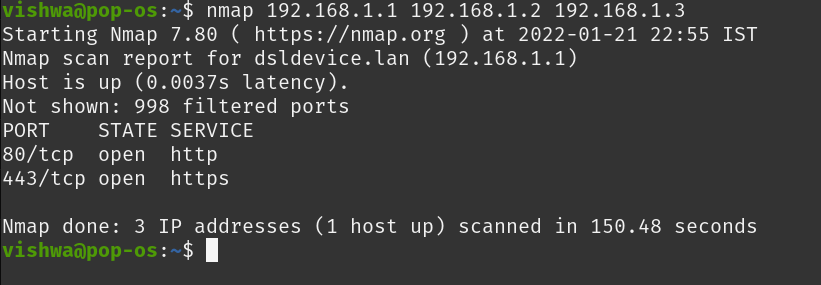
**Step 2:** Alternatively, use an IP address to scan.

**nmap 163.53.78.128**

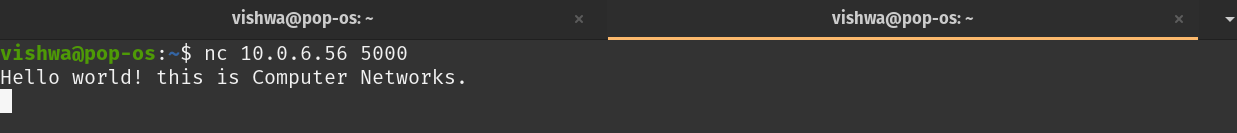
****

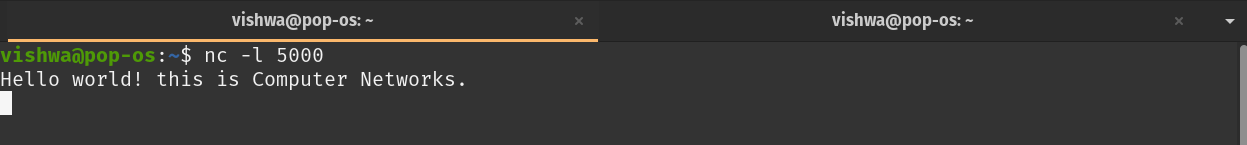
**Step 3:** Scan multiple IP address or subnet (IPv4)

**nmap 192.168.1.1 192.168.1.2 192.168.1.3**

****

**TASK 7 A): NETCAT AS CHAT TOOL  
a) Intra system communication (Using 2 terminals in the same system)  
Step 1:** Open a terminal (Ctrl+Alt+T). This will act as a Server. **Step 2:** Type nc -l any\_portnum (For eg., nc -l 1234) **Note:** It will goto listening mode **Step 3:** Open another terminal and this will act as a client. **Step 4:** Type nc <your-system-ip-address> portnum **Note:** portnum should be common in both the terminals (for eg., nc 10.0.2.8 1234) **Step 5:** Type anything in client will appear in server





**Questions on above observations:**

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

**HTTP 1.1 for browser and server.**

2) When was the HTML file that you are retrieving last modified at the server?

**Thu, 20 Jan 2022 16:25:30 GMT\r\n**

3) How to tell ping to exit after a specified number of ECHO\_REQUEST packets?

**ping ip –v no\_of\_pings**

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

**nmap –O –v ip\_addr**