Computer Network Lab – Week 1

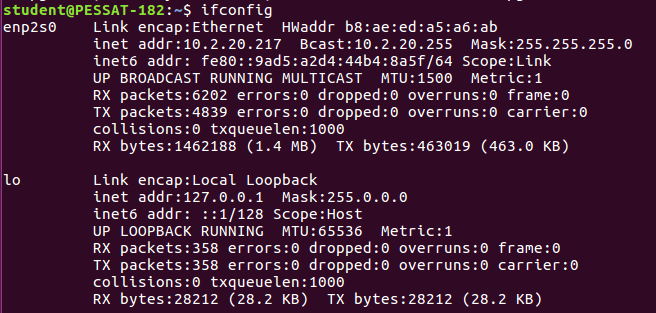
PES1UG20CS806

Divyanshu Sharma

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

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

**Command Used:** ifconfig (**Displaying all the active network interface)**



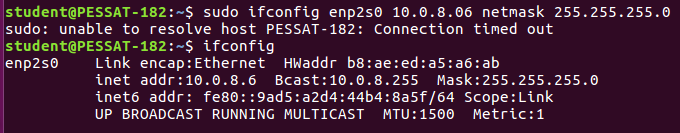
Analyse and fill the following table:

|  |  |  |
| --- | --- | --- |
| **Interface Name** | **IPv4/IPv6** | **MAC Address** |
| Enp2s0 | 10.2.20.217 / fe80::9ad5:a2d4:44b4:8a5f | b8:ae:ed:a5:a6:ab |
| Lo | 127.0.0.1 / ::1 | 00:00:00:00:00:00 |

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

**Command Used:** sudo ifconfig interface\_name 10.0.your\_section.your\_sno netmask 255.255.255.0

**:** sudo ifconfig enp2s0 10.0.8.06 netmask 255.255.255.0



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

**Command Used:** sudo ifconfig interface\_name down

sudo ifconfig enp2s0 down **(Deactivating enp2s0)**

G:\IMAGES\BANGALORE\BANGALORE\PESU UNIVERCITY BTECH\PESU\2ND YEAR\4th SEM\Computer Netwok Lab\task1c.png

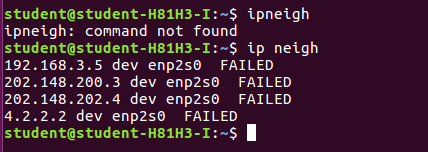
**Command Used:** sudo ifconfig interface\_name up

sudo ifconfig enp2s0 up **(Activating enp2s0)**



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

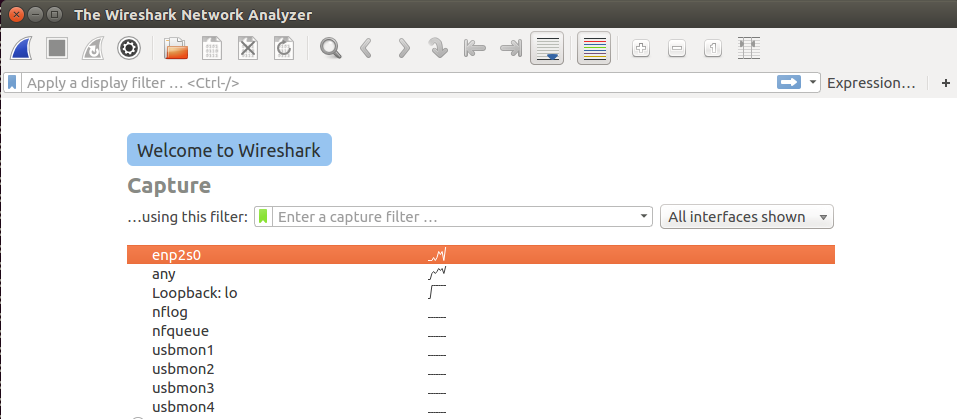
**Command Used:** ip neigh



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

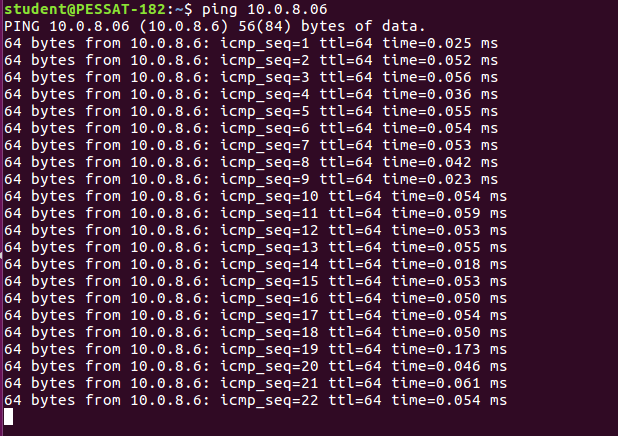
**Step 1:** Assign an IP address to the system (Host). Note: IP address of your system should be 10.0.your\_section.your\_sno.

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



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

**Command Used** – ping 10.0.8.06



**Observations to be made**

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

• TTL

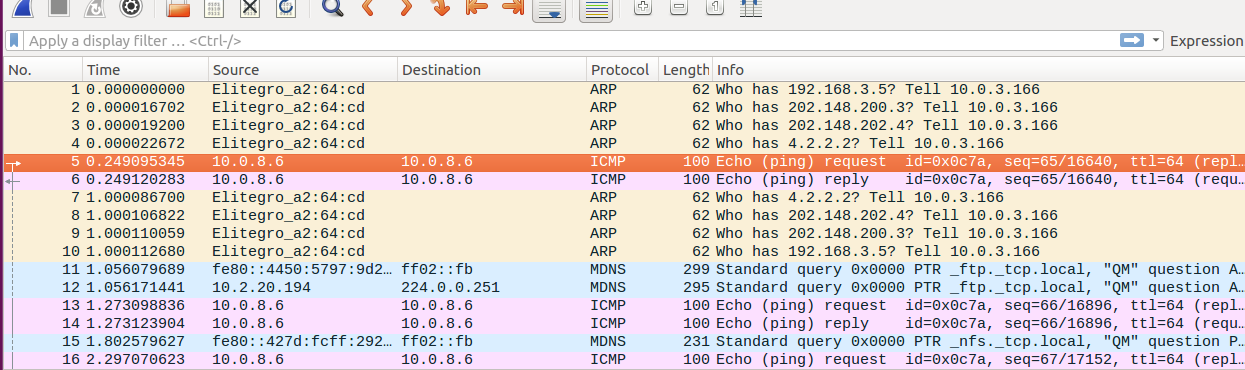
• Protocol used by ping

• Time

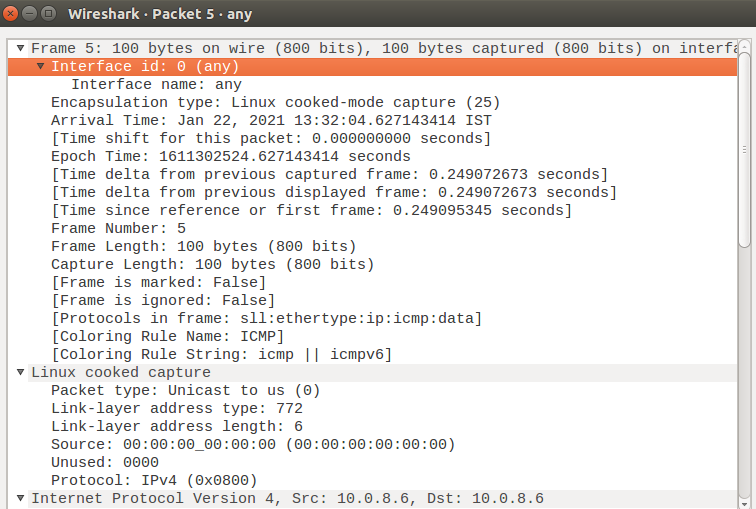
|  |  |
| --- | --- |
| **TTL** | 64 |
| **Protocol used by ping** | ICMP |
| **Time** | Order of 10-2 ms |

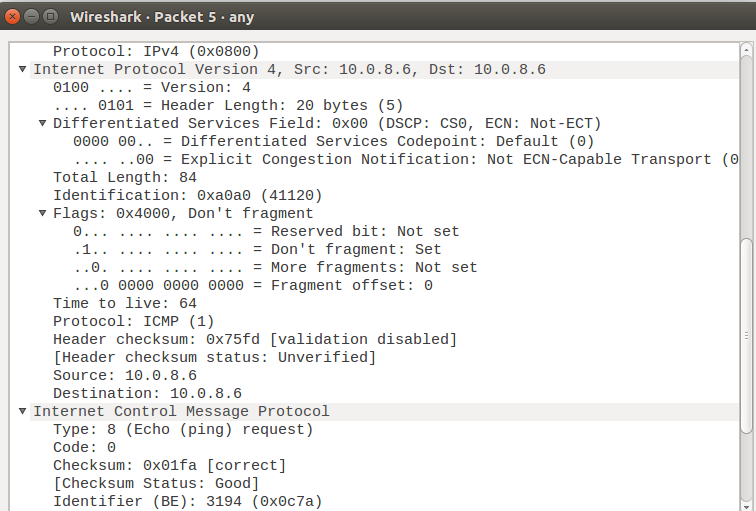
**Step 5:** Analyze the following in Wireshark On Packet List Pane, select the first echo packet on the list. On Packet Details Pane, click on each of the four “+” to expand the information. Analyze the frames with the first echo request and echo reply and complete the table below.

**Showing Request and Response Packet**

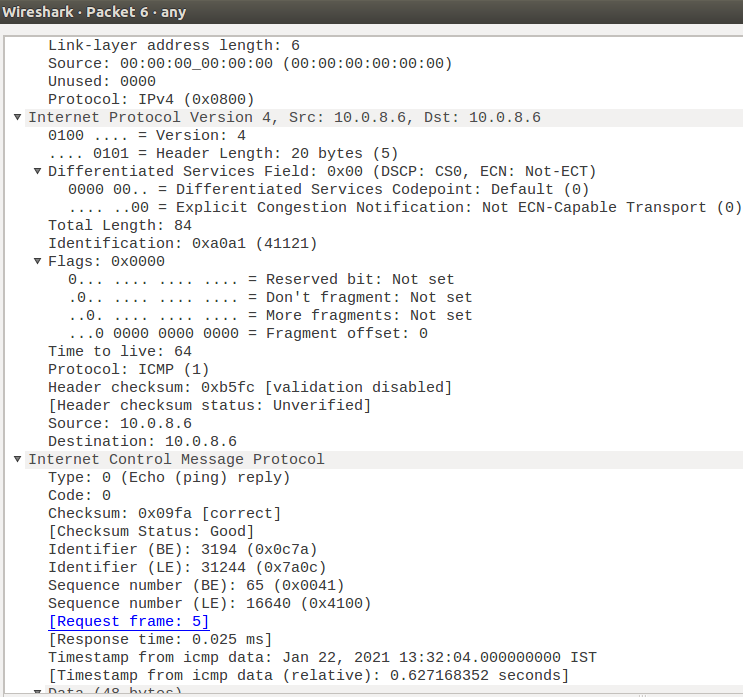


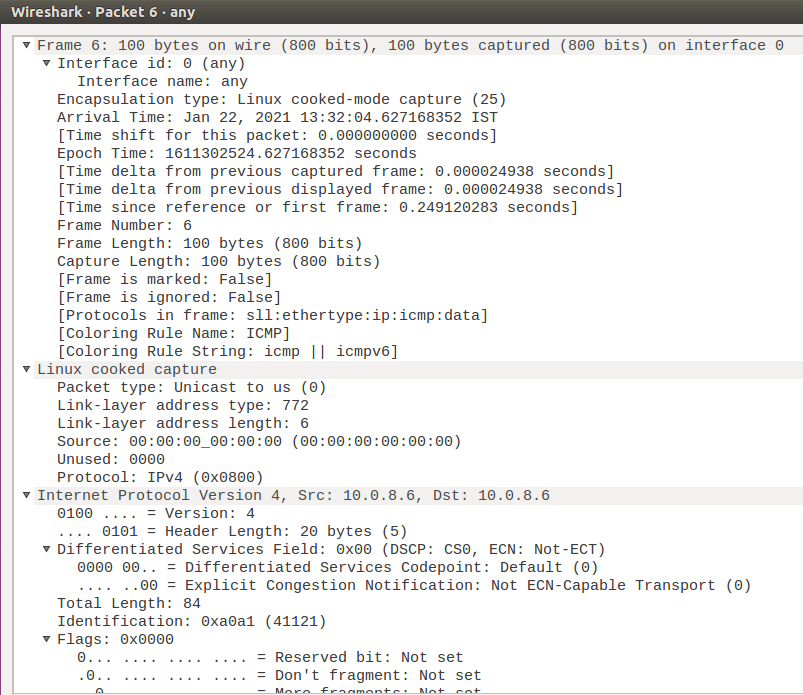
**Request Packet**





**Response Packet**





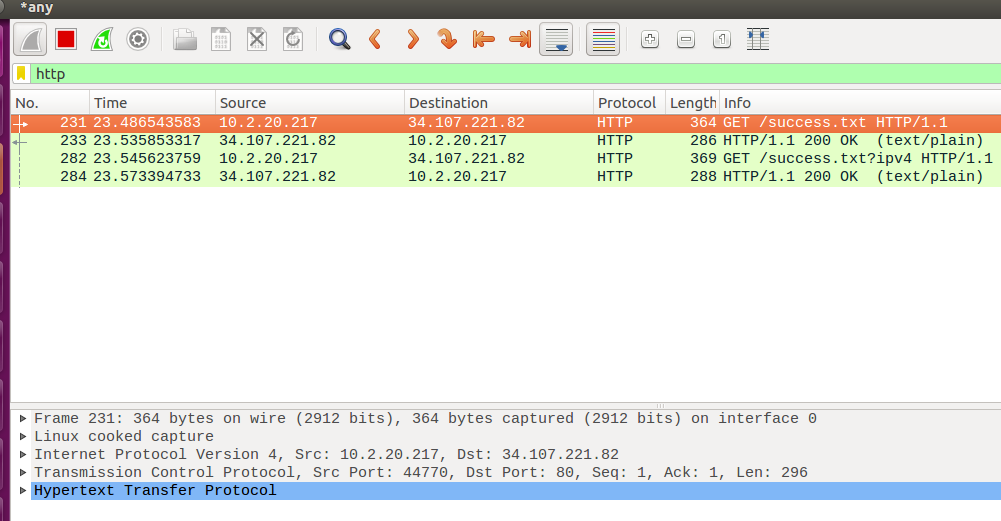
|  |  |  |
| --- | --- | --- |
| **Details** | **First Echo Request** | **First Echo Reply** |
| **Frame Number** | 5 | 6 |
| **Source IP address** | 10.0.8.6 | 10.0.8.6 |
| **Destination IP address** | 10.0.8.6 | 10.0.8.6 |
| **ICMP Type Value** | 8 | 0 |
| **ICMP Code Value** | 0 | 0 |
| **Source Ethernet 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** | IPv4 | IPv4 |
| **Time to Live (TTL) Value** | 64 | 64 |

**Task 3: HTTP PDU Capture**

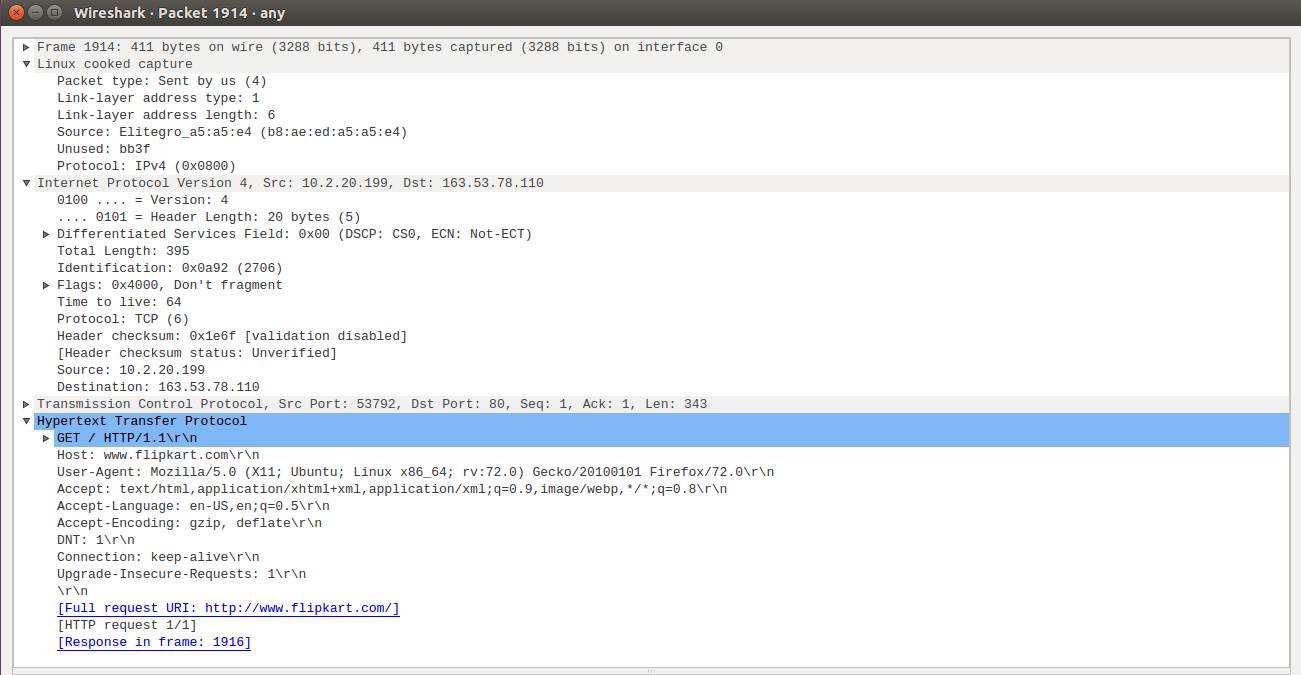
Using Wireshark’s Filter feature

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

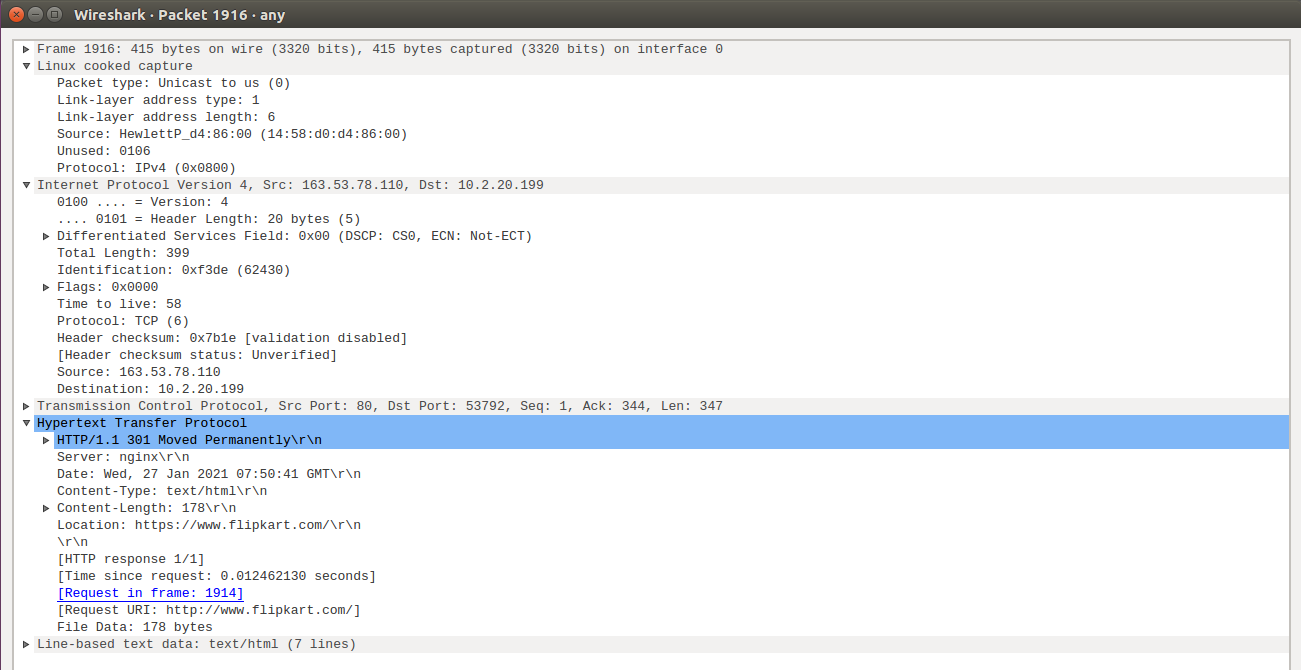
**Step 2:** Open Firefox browser, and browse [www.flipkart.com](http://www.flipkart.com)



**Request Packet**



**Response Packet**



**Observations to be made**

**Step 3**

|  |  |  |
| --- | --- | --- |
| **Details** | **First Echo Request** | **First Echo Reply** |
| **Frame Number** | 1914 | 1916 |
| **Source Port** | 53792 | 80 |
| **Destination Port** | 80 | 53792 |
| **Source IP address** | 10.2.20.199 | 163.53.78.110 |
| **Destination IP address** | 163.53.78.110 | 10.2.20.199 |
| **Source Ethernet Address** | b8:ae:ed:a5:a5:e4 | 14:58:d0:d4:86:00 |
| **Destination Ethernet Address** | 14:58:d0:d4:86:00 | b8:ae:ed:a5:a5:e4 |

(Connection Details)

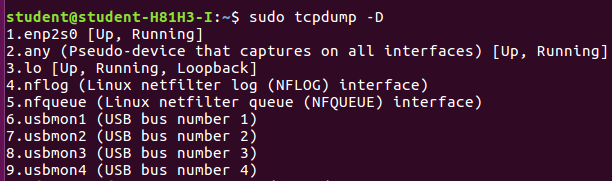
**Step 4:** Analyze the HTTP request and response and complete the table below

|  |  |  |  |
| --- | --- | --- | --- |
| **HTTP Request** | | **HTTP Response** | |
| **Get** | GET / HTTP1.1\r\n | **Server** | nginx |
| **Host** | www.flipkart.com | **Content-Type** | Text/html |
| **User-Agent** | Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:72.0) Gecko/20100101 Firefox/72.0\r\n | **Date** | Wed, 27 Jan 2021 07:50:41 GMT |
| **Accept-Language** | En-US, en;q=0.5\r\n | **Location** | https://www.flipkart.com/ |
| **Accept-Encoding** | Gzip, deflate\r\n | **Content-Length** | 178\r\n |
| **Connection** | Keep-alive | **Connection** | Keep-alive |

**Task 4: Capturing packets with tcpdump**

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

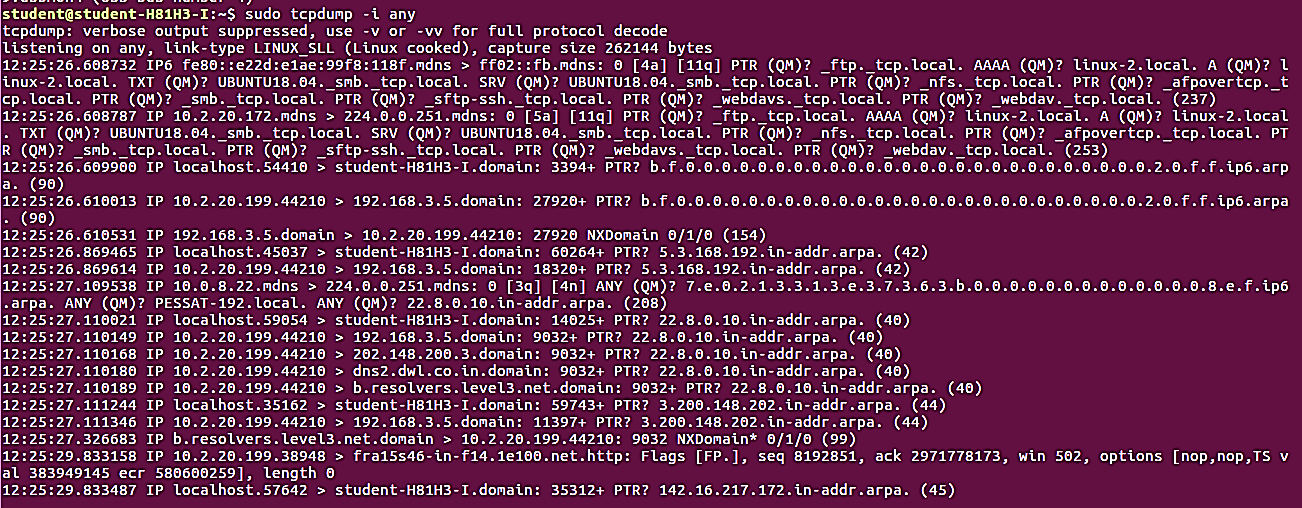
**Command Used -** sudo tcpdump -D



(Viewing Interfaces available for Capture)

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

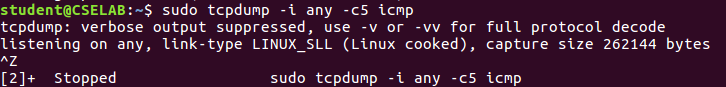
**Command Used -** sudo tcpdump -i any



(Capturing all Packets in any Interface)

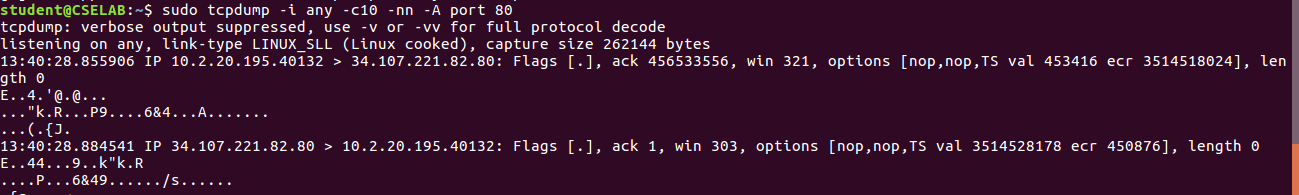
**Step 4**: To filter packets based on protocol, specifying the protocol in the command line. For example, capture ICMP packets only by using this command:

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



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

**Command Used -** sudo tcpdump -i any -c10 -nn -A port 80

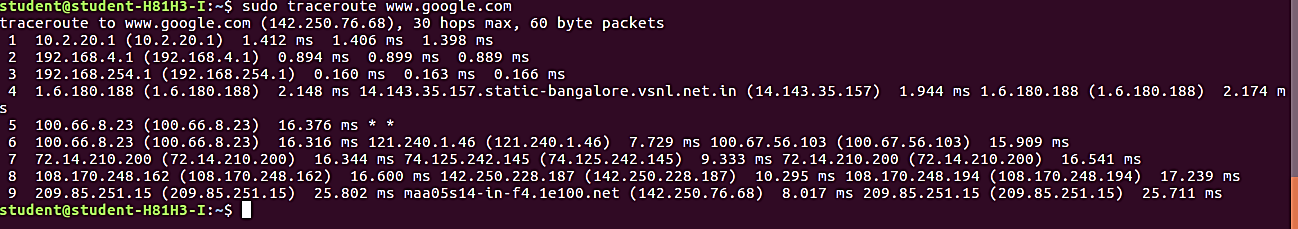




**Task 5: Perform Traceroute checks**

**Step 1:** Run the traceroute using the following command.

**Command Used -** sudo traceroute www.google.com

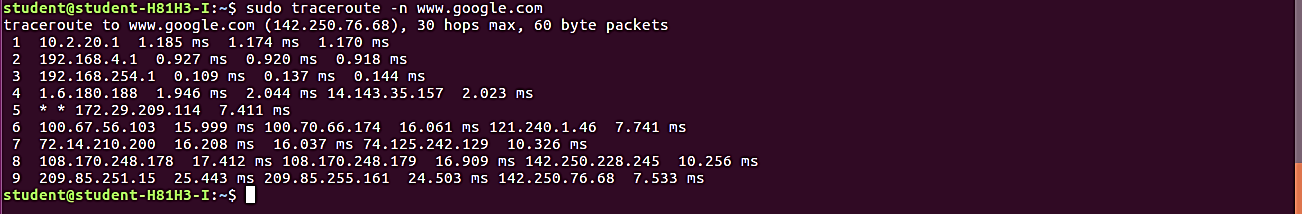


**Step 2:** Analyze destination address of google.com and no. of hops

The destination address is **142.250.76.68** and there were **30 hops**.

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

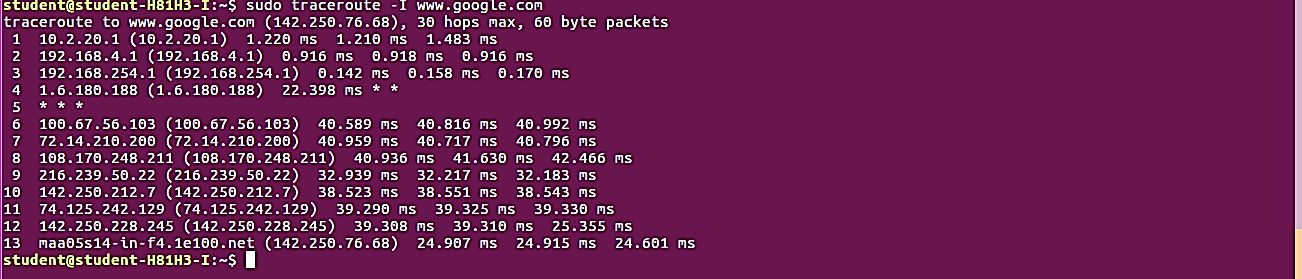
**Command used -** sudo traceroute -n [www.google.com](http://www.google.com)



(Disabling mapping of IP addresses with hostnames)

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

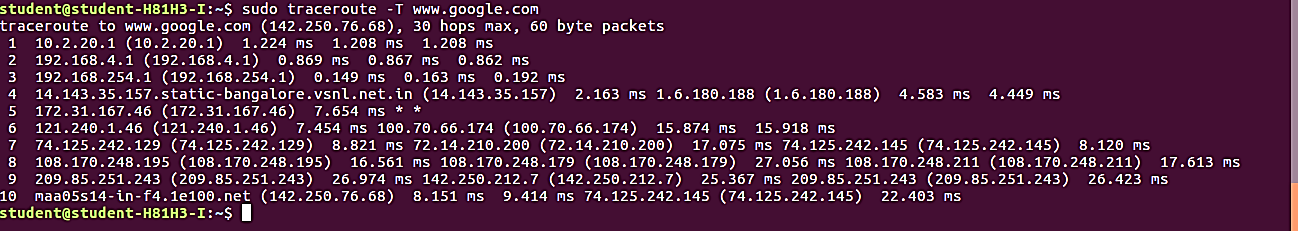
**Command Used -** sudo traceroute -I [www.google.com](http://www.google.com)



(traceroute with ICMP Protocol)

**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.

**Command Used -** sudo traceroute -T [www.google.com](http://www.google.com)

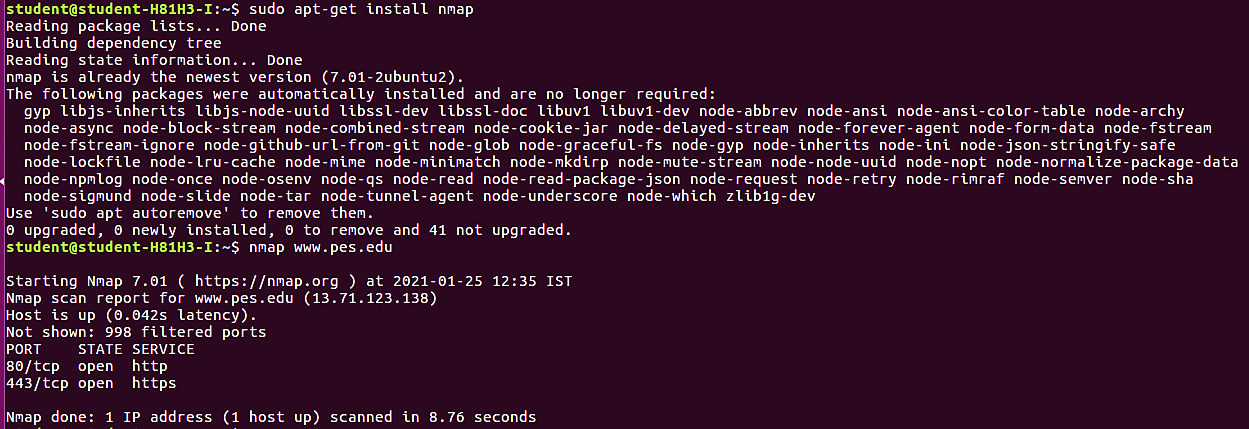


(Testing TCP Connection with traceroute)

**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.

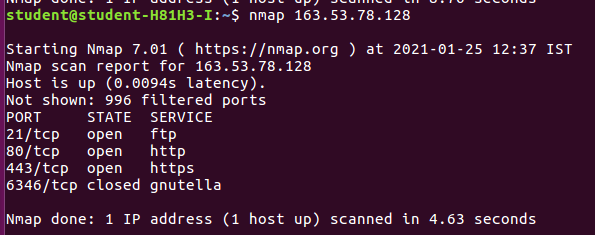
**Command Used -** nmap [www.pes.edu](http://www.pes.edu)



(Scanning Host with Hostname)

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

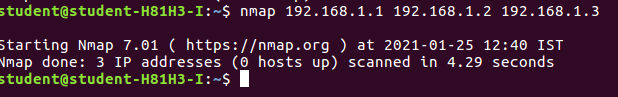
**Command Used -** nmap 163.53.78.128



(Scanning Host with IP Address)

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

**Command Used -** 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)

**Command Used :** nc –l 8090

Note: It will goto listening mode

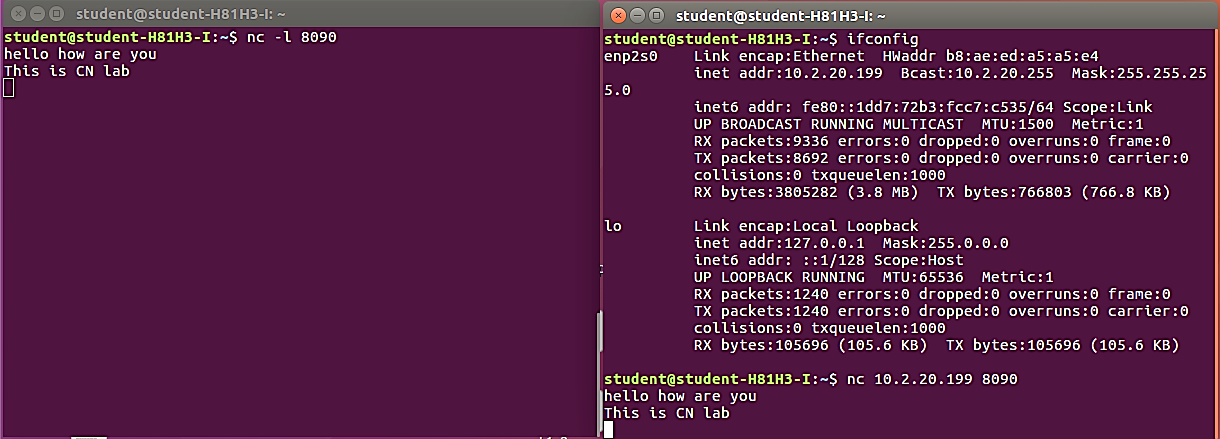
**Step 3:** Open another terminal and this will act as a client.

**Step 4:** Type nc <your-system-ip-address> portnum

**Command Used :** nc 10.2.20.199 8090

Note: portnum should be common in both the terminals (for eg., nc 10.0.2.8 1234)

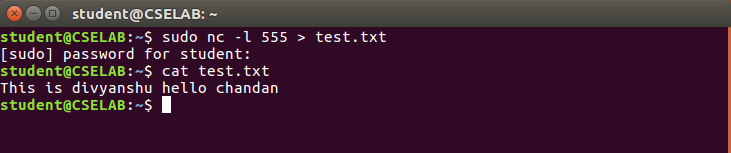
**Step 5:** Type anything in client will appear in server



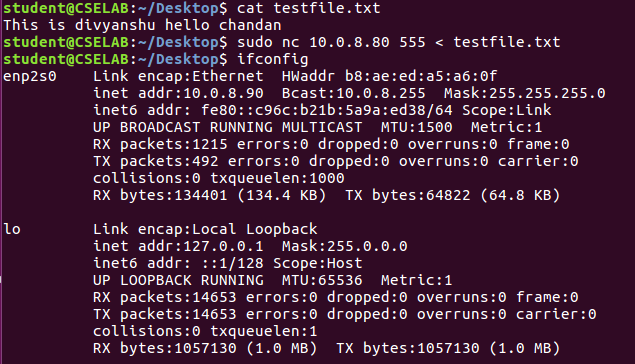
**Task 7 b): Use Netcat to Transfer Files The netcat utility can also be used to transfer files.**

**Step 1:** At the server side, create an empty file named ‘test.txt’ sudo nc -l 555 > test.txt

**Step 2:** At the client side, we have a file ‘testfile.txt’. Add some contents to it.



**Step 3:** Run the client as: sudo nc 10.0.2.8 555 < testfile.txt here, 10.0.2.8 is the IP address of server and 555 is the port number.



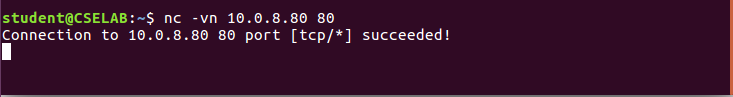
**Step 4:** At server side, verify the file transfer using the command cat test.txt



**Task 7 c): Other Commands**

**1)** To test if a particular TCP port of a remote host is open.

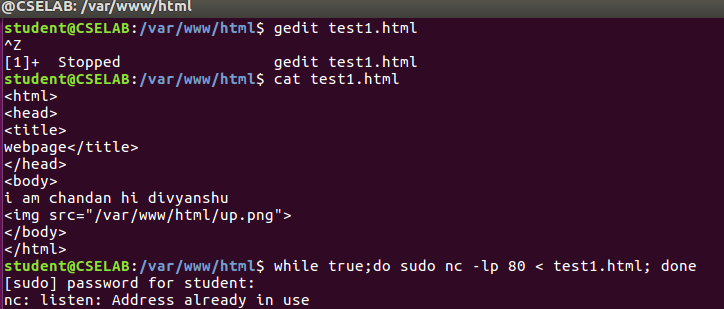
**Command Used :-** nc -vn 10.0.2.8 555

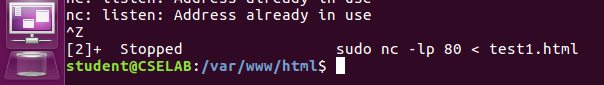


**2)** Run a web server with a static web page.

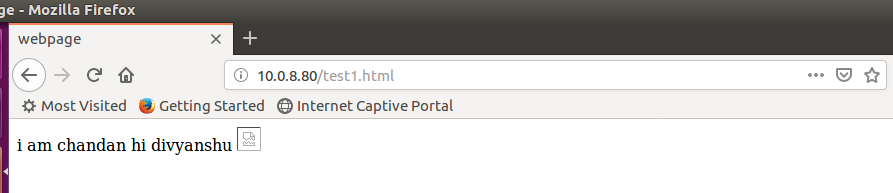
**Step 1:** Run the command below on local host (e.g. 10.0.2.8) to start a web server that serves test.html on port 80.

**Command Used:-** **while true; do sudo nc -lp 80 < test1.html; done**





**Step 2:** Now open **http://10.0.8.80/test1.html** from another host to access it.



**Step 3:** Observe the details on the terminal

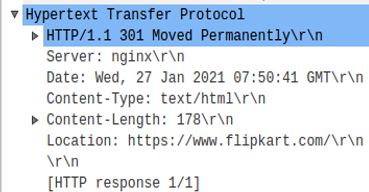
**Questions**

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

**Answer –** The Firefox browser used is running HTTP v1.1, and this can be seen in the request header which contains the method (GET) followed by the HTTP version. Similarly, the HTTP version of the web server is v1.1 and can be seen in the header of the HTTP response sent back to the browser.



**Request**



**Response**

1. **How to tell ping to exit after a specified number of ECHO\_REQUEST packets?**

**Answer –** Ping continues to send ICMP packages until it receives an interrupt signal. To specify the number of ECHO\_REQUEST packages after which ping will exit, we can use the -c option followed by the number of packages. ping -c 10 www.pes.edu