Computer Networks Lab -1

Name : P K Navin Shrinivas

Section : D

SRN : PES2UG20CS237

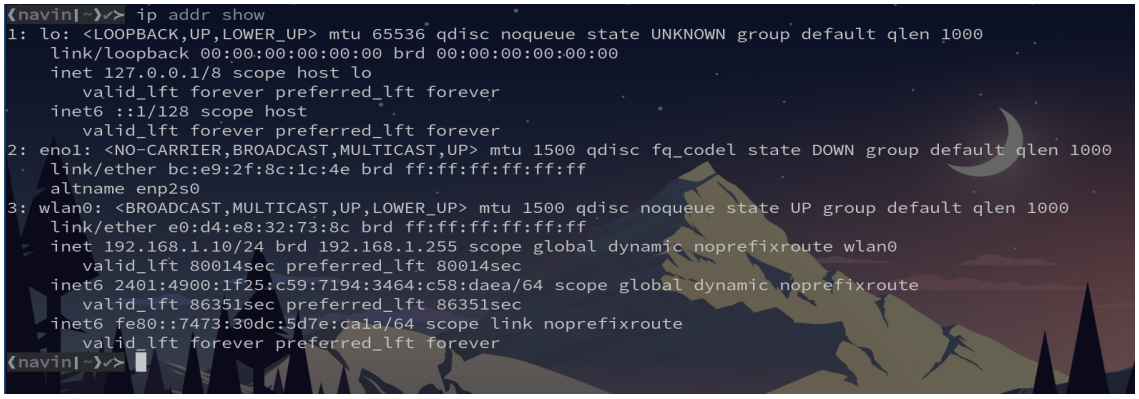
Week : 1

Note : All written answers are in bold.

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

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

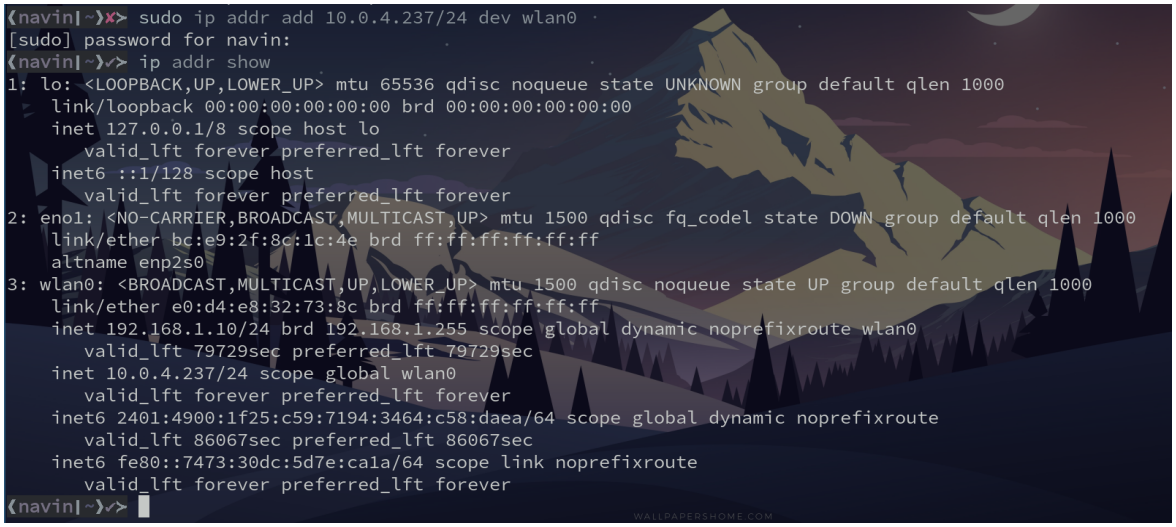
Analyze and fill the following table:



ip-address table:

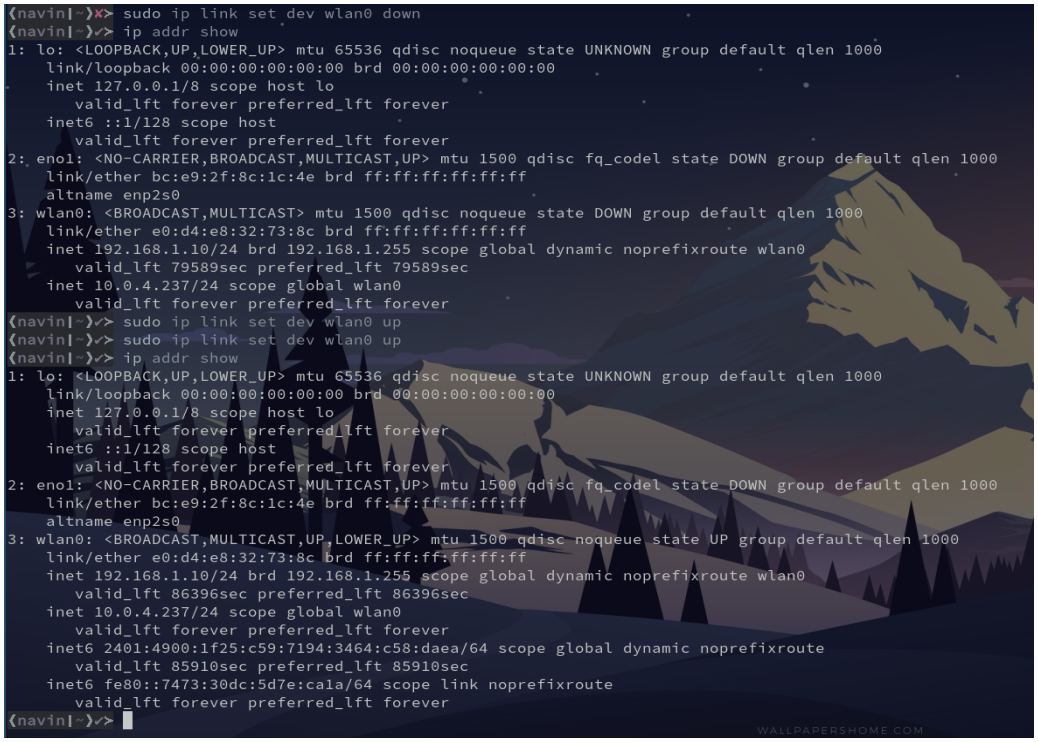
Interface name	**IP address (IPv4 / IPv6)**	**MAC address**	
Lo	**127.0.0.1**	**00:00:00:00:00:00**	**Loop back device**
Eno1 [enp2s0]	**-**	**bc:e9:2f:8c:1c:4e**	**Ethernet, disconnected**
Wlan0	**192.168.1.10**	**e0:d4:e8:32:73:8c**	**Wireless LAN**

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



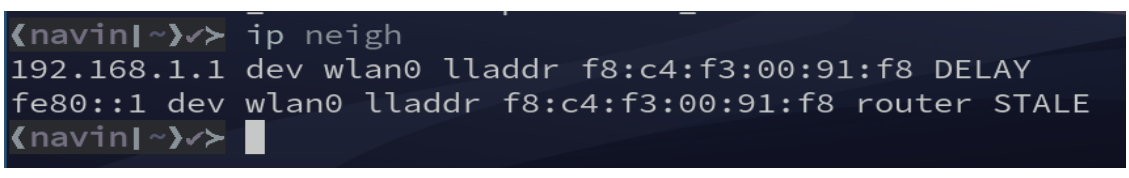
Step 3: To activate / deactivate a network interface, type.

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).

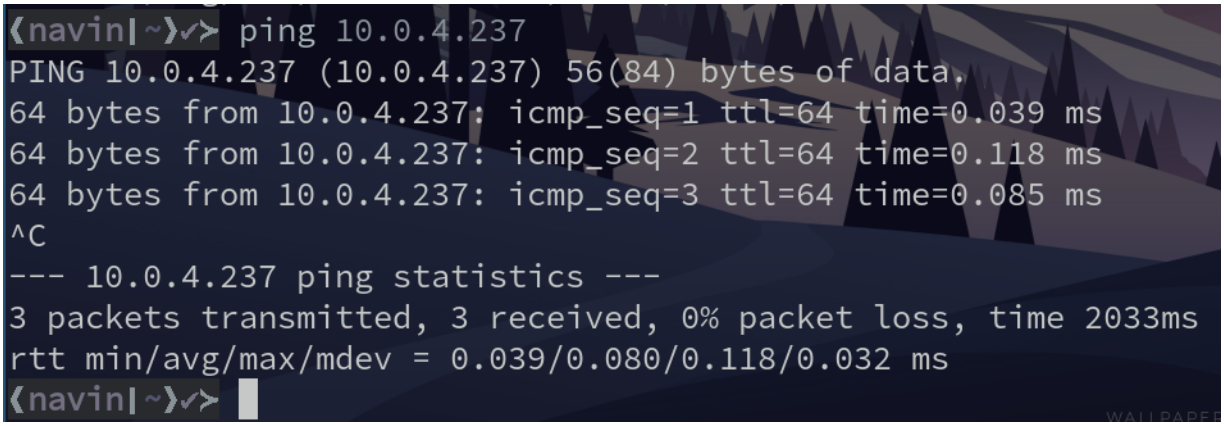
Note: IP address of your system should be 10.0.your_section.your_sno.

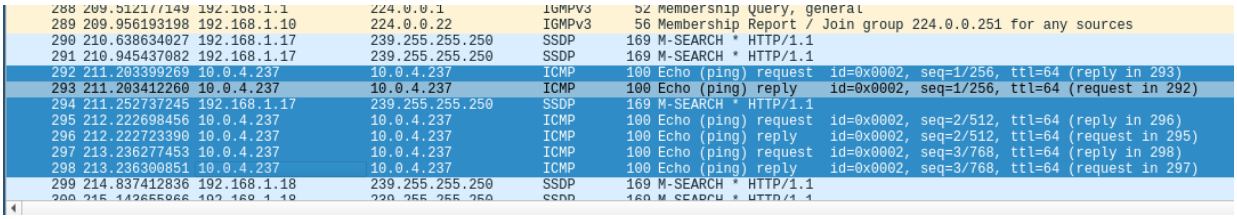
Done in previous tasks, hence pinging the same machine causes a loopback, this can be seen with indiffrentiating ip add in the table.Can also be seen with the mac address of device used.

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 : **~2000ms , observed from ping command**

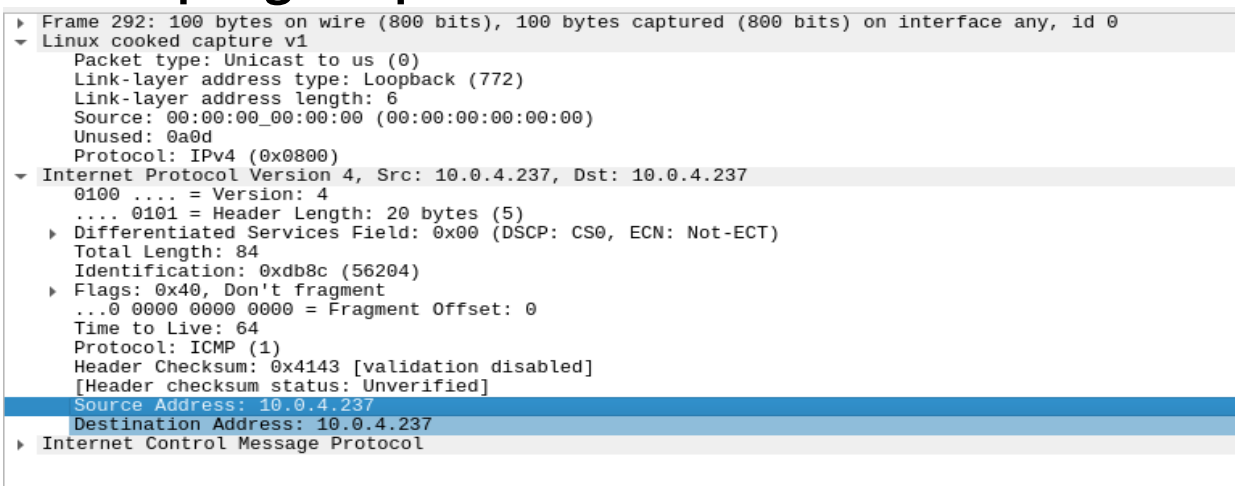
Step 5: Analyze the following in Wireshark

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

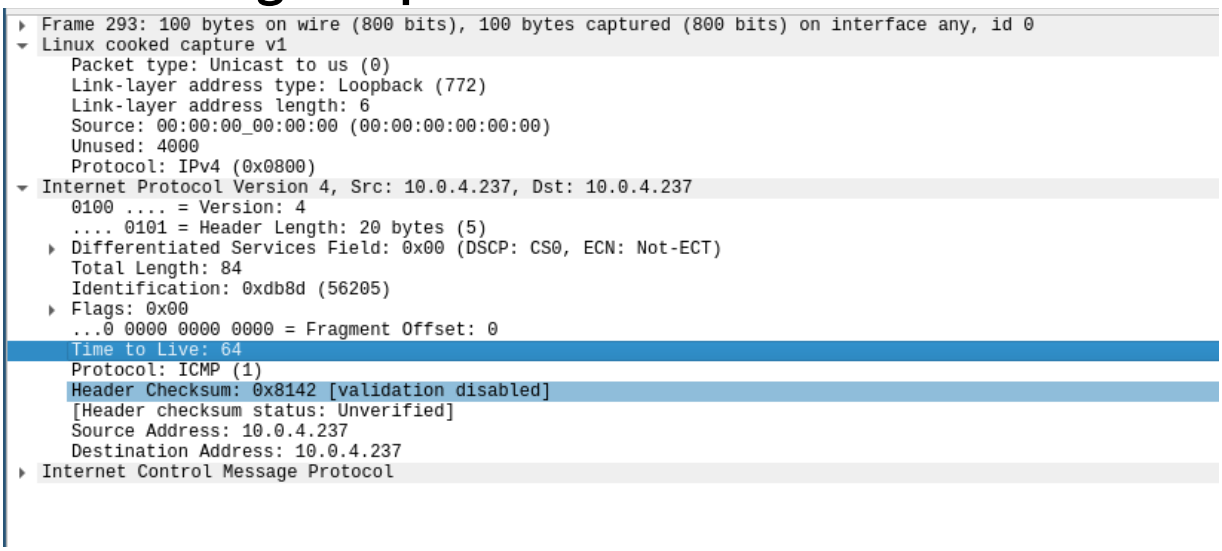
and echo reply and complete the table below.

Details	First Echo Request	First Echo Reply
Frame Number	292	293
Source IP address	10.0.4.237	10.0.4.237
Destination IP address	10.0.4.237	10.0.4.237
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

First ping request :



First Ping response :



Task 3: HTTP PDU Capture

Using Wireshark's

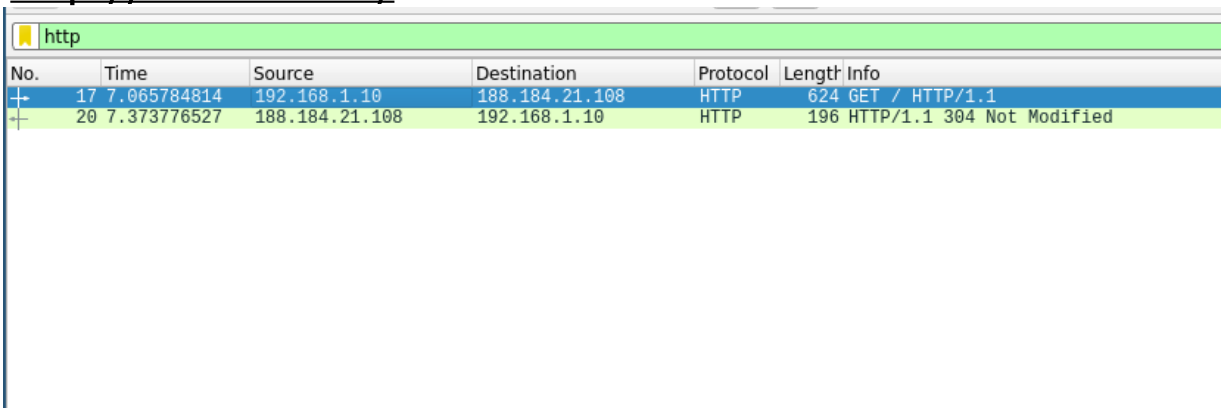
Filter feature

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

www.flipkart.com does not do http requests and hence wireshark can not analyze them, thus I would be doing my analysis using :

http://info.cern.ch/

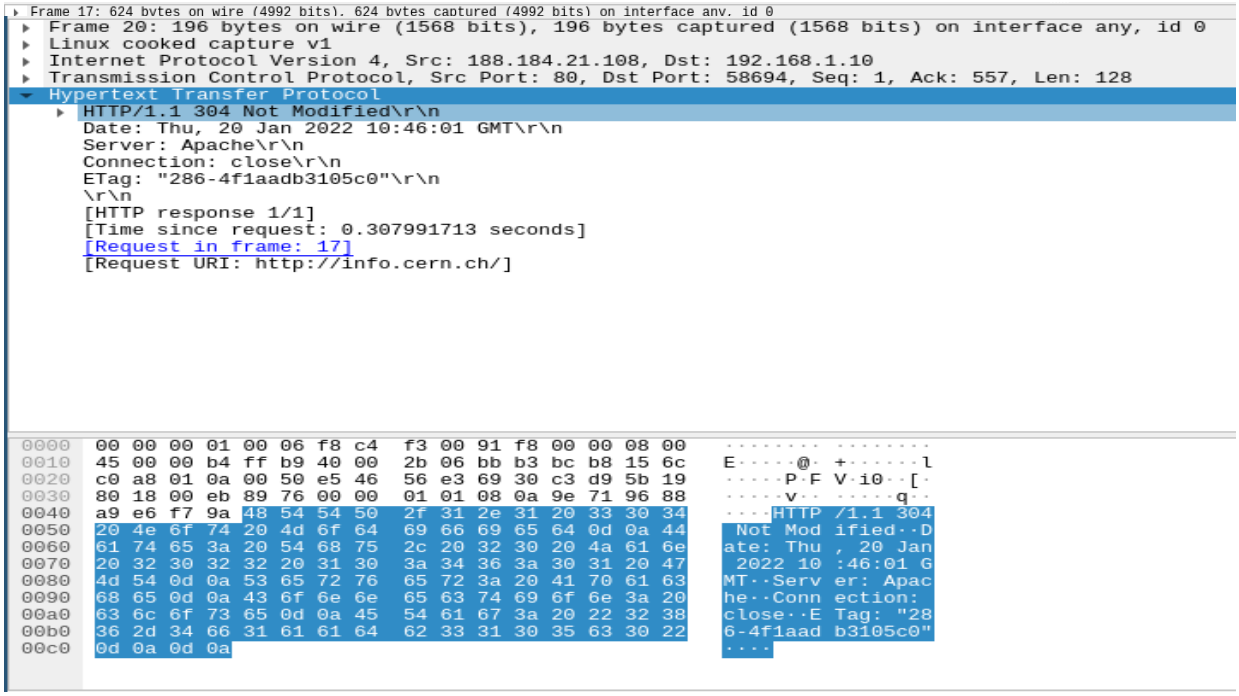


Observations to be made

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	17	20
Source Port	58694	80
Destination Port	80	58964
Source IP address	192.168.1.10	188.184.21.108
Destination IP address	188.184.21.108	192.148.1.10
Source Ethernet Address	e0:d4:e8:32:73:8c	f8:c4:f3:00:91:f8
Destination Ethernet Address	f8:c4:f3:00:91:f8	e0:d4:e8:32:73:8c

HTTP First request :



HTTP First response :

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

HTTP Request		HTTP Response	
Get	GET / HTTP/1.1\r\n	Server	Apache
Host	info.cern.ch	Content-Type	text/plain
User-Agent	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36	Date	Thu, 20 Jan 2022 12:09:16 GMT
Accept-Language	en-GB,en-US;q=0.9,en;q=0.8	Location	
Accept-Encoding	gzip, deflate	Content-Length	25
Connection	keep-alive	Connection	Close

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.

Entire conversation :

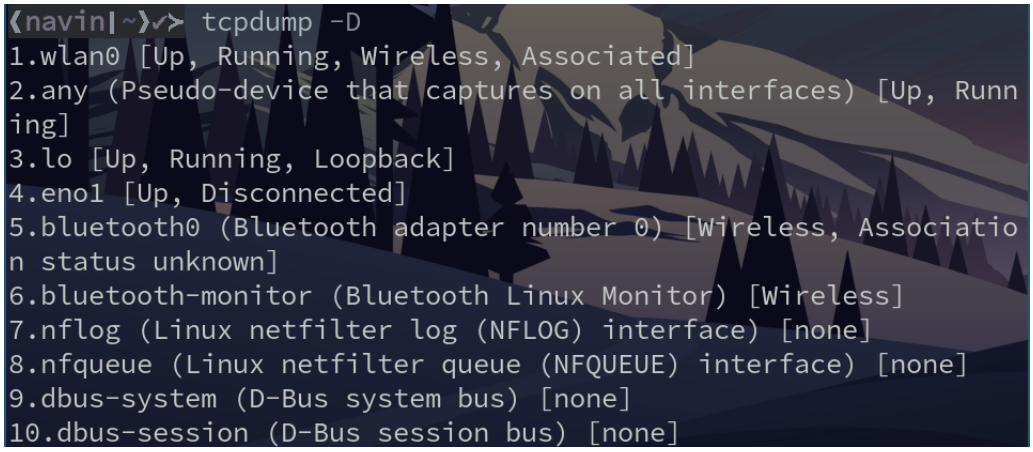
```
GET / HTTP/1.1
Host: info.cern.ch
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-GPC: 1
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
If-None-Match: "286-4f1aadb3105c0"
If-Modified-Since: Wed, 05 Feb 2014 16:00:31 GMT

HTTP/1.1 304 Not Modified
Date: Thu, 20 Jan 2022 10:46:01 GMT
Server: Apache
Connection: close
ETag: "286-4f1aadb3105c0"
```

Sadly this connection did not yield any TCP connections to follow.

Task 4: Capturing packets with tcpdump

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

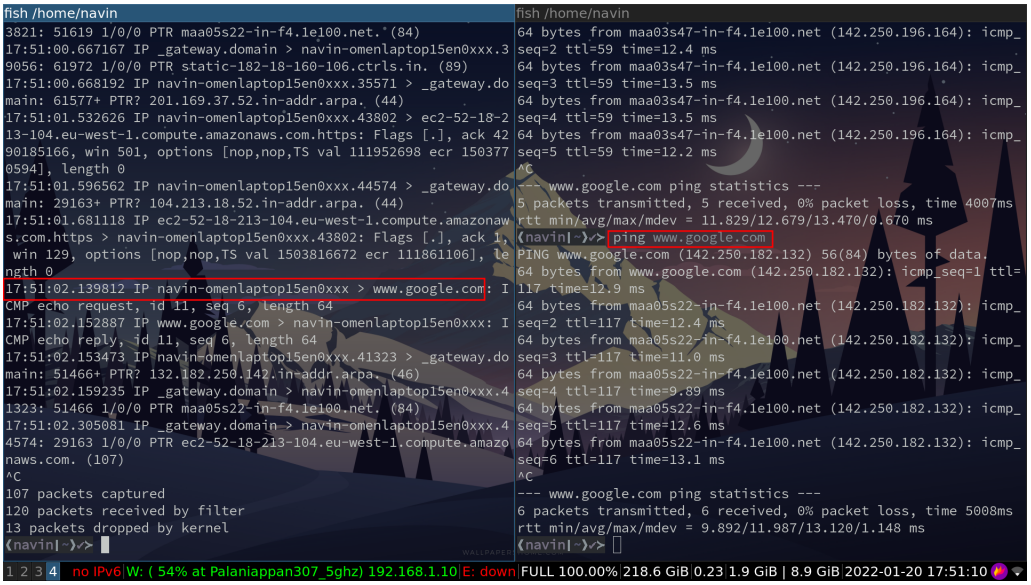
```
sudo tcpdump -D
```



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.



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:

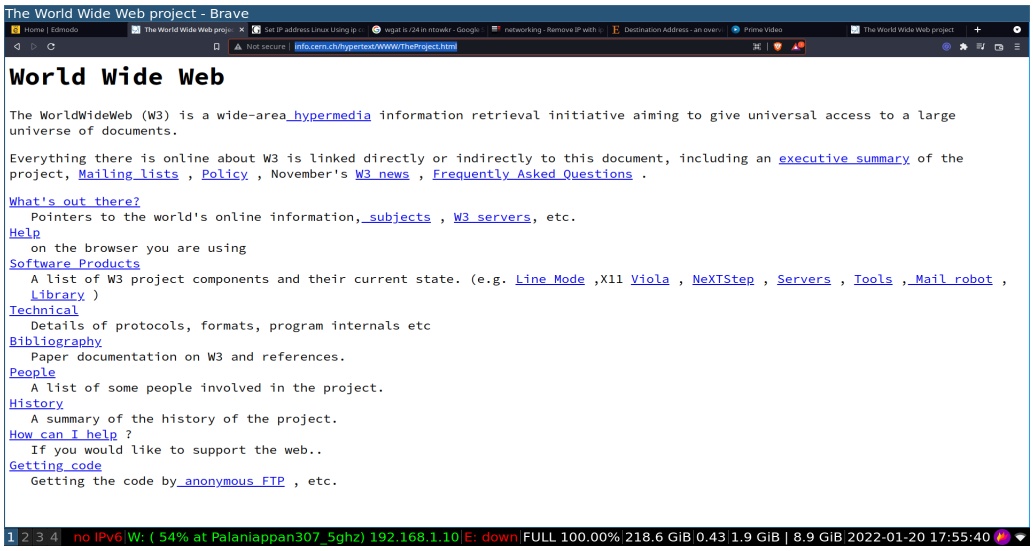
```
sudo tcpdump -i any -c10 -nn -A port 80
```

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

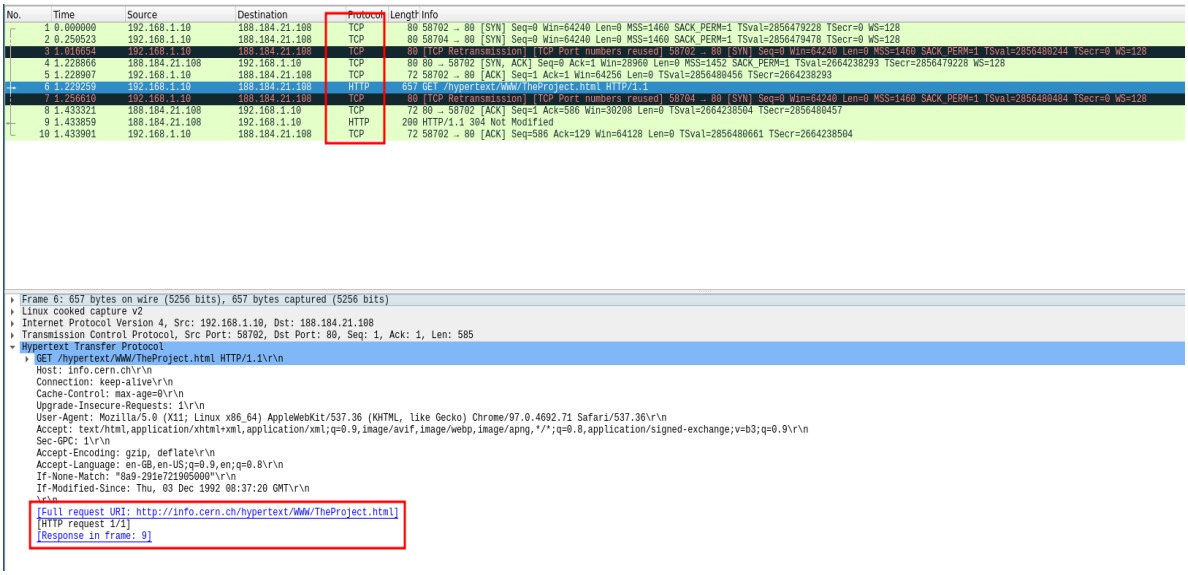
```
sudo tcpdump -i any -c10 -nn -w webserver.pcap port 80
```

PING TO IN BROWSER : http://info.cern.ch/hypertext/WWW/TheProject.html

A ping command was not used as filter command was for HTTP and not ICMP



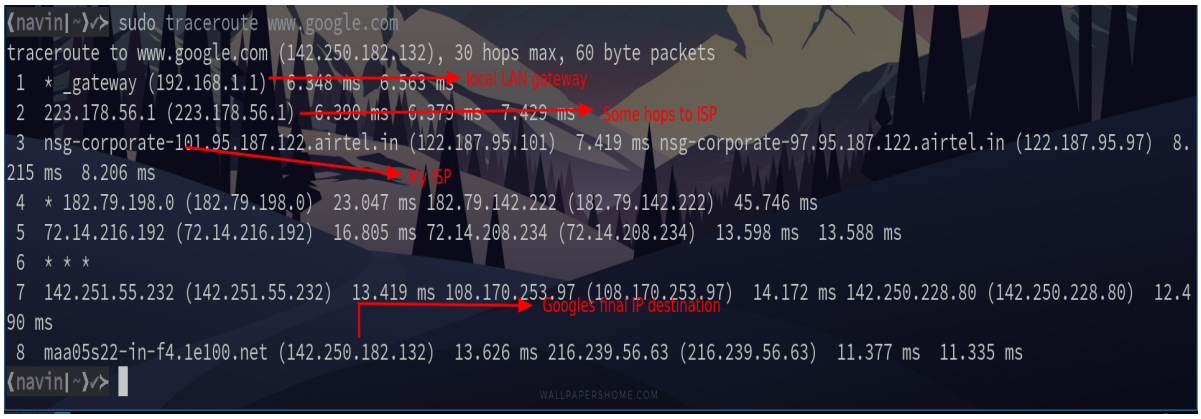
Opening the above file in wireshark :



Task 5: Perform Traceroute checks

Step 1: Run the traceroute using the following command.

sudo traceroute www.google.com




```
(navin|~)↵> sudo traceroute www.google.com
traceroute to www.google.com (142.250.182.132), 30 hops max, 60 byte packets
 1  * _gateway (192.168.1.1)  6.348 ms  6.563 ms
 2  223.178.56.1 (223.178.56.1)  6.390 ms  6.379 ms  7.429 ms
 3  nsg-corporate-101.95.187.122.airtel.in (122.187.95.101)  7.419 ms nsg-corporate-97.95.187.122.airtel.in (122.187.95.97)  8.
215 ms  8.206 ms
 4  * 182.79.198.0 (182.79.198.0)  23.047 ms 182.79.142.222 (182.79.142.222)  45.746 ms
 5  72.14.216.192 (72.14.216.192)  16.805 ms 72.14.208.234 (72.14.208.234)  13.598 ms  13.588 ms
 6  * * *
 7  142.251.55.232 (142.251.55.232)  13.419 ms 108.170.253.97 (108.170.253.97)  14.172 ms 142.250.228.80 (142.250.228.80)  12.4
90 ms
 8  maa05s22-in-f4.1e100.net (142.250.182.132)  13.626 ms 216.239.56.63 (216.239.56.63)  11.377 ms  11.335 ms
(navin|~)↵>
```

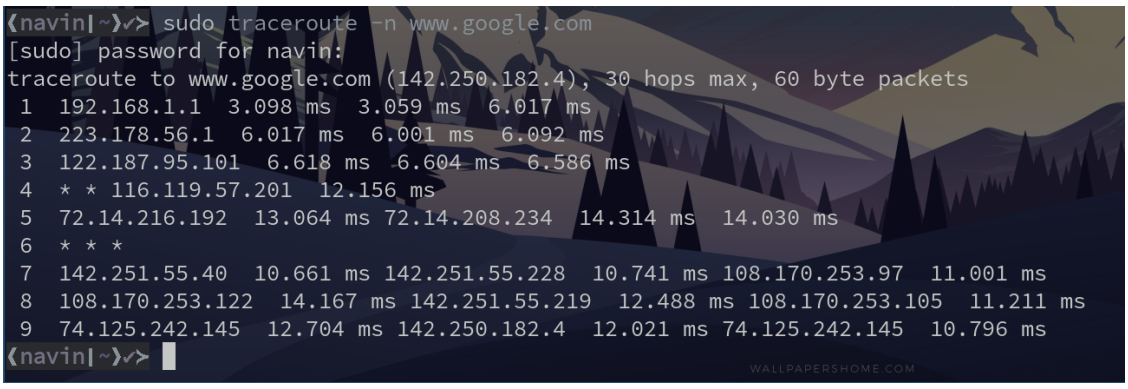
Step 2: Analyze destination address of google.com and no. of hops

Destination : 142.250.182.132

Number of hops : 8

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

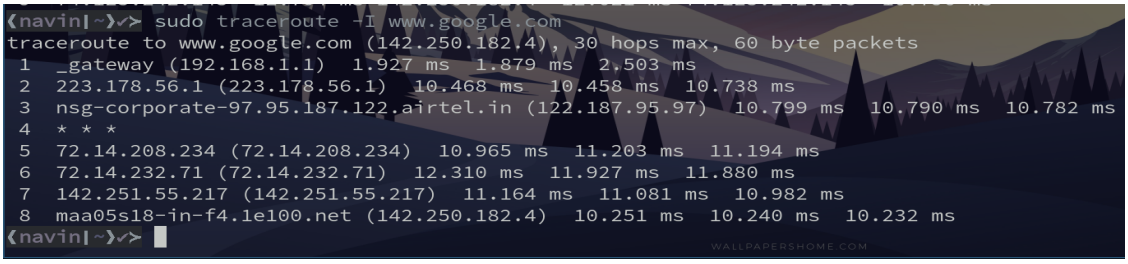
sudo traceroute -n www.google.com



```
(navin|~)↵> sudo traceroute -n www.google.com
[sudo] password for navin:
traceroute to www.google.com (142.250.182.4), 30 hops max, 60 byte packets
 1  192.168.1.1  3.098 ms  3.059 ms  6.017 ms
 2  223.178.56.1  6.017 ms  6.001 ms  6.092 ms
 3  122.187.95.101  6.618 ms  6.604 ms  6.586 ms
 4  * * 116.119.57.201  12.156 ms
 5  72.14.216.192  13.064 ms 72.14.208.234  14.314 ms  14.030 ms
 6  * * *
 7  142.251.55.40  10.661 ms 142.251.55.228  10.741 ms 108.170.253.97  11.001 ms
 8  108.170.253.122  14.167 ms 142.251.55.219  12.488 ms 108.170.253.105  11.211 ms
 9  74.125.242.145  12.704 ms 142.250.182.4  12.021 ms 74.125.242.145  10.796 ms
(navin|~)↵>
```

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

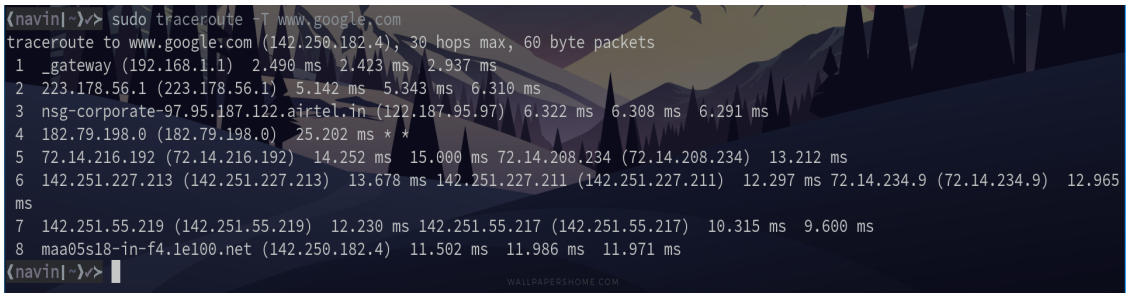
sudo traceroute -I www.google.com



```
(navin|~)↵> sudo traceroute -I www.google.com
traceroute to www.google.com (142.250.182.4), 30 hops max, 60 byte packets
 1  _gateway (192.168.1.1)  1.927 ms  1.879 ms  2.503 ms
 2  223.178.56.1 (223.178.56.1)  10.468 ms  10.458 ms  10.738 ms
 3  nsg-corporate-97.95.187.122.airtel.in (122.187.95.97)  10.799 ms  10.790 ms  10.782 ms
 4  * * *
 5  72.14.208.234 (72.14.208.234)  10.965 ms  11.203 ms  11.194 ms
 6  72.14.232.71 (72.14.232.71)  12.310 ms  11.927 ms  11.880 ms
 7  142.251.55.217 (142.251.55.217)  11.164 ms  11.081 ms  10.982 ms
 8  maa05s18-in-f4.1e100.net (142.250.182.4)  10.251 ms  10.240 ms  10.232 ms
(navin|~)↵>
```

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.

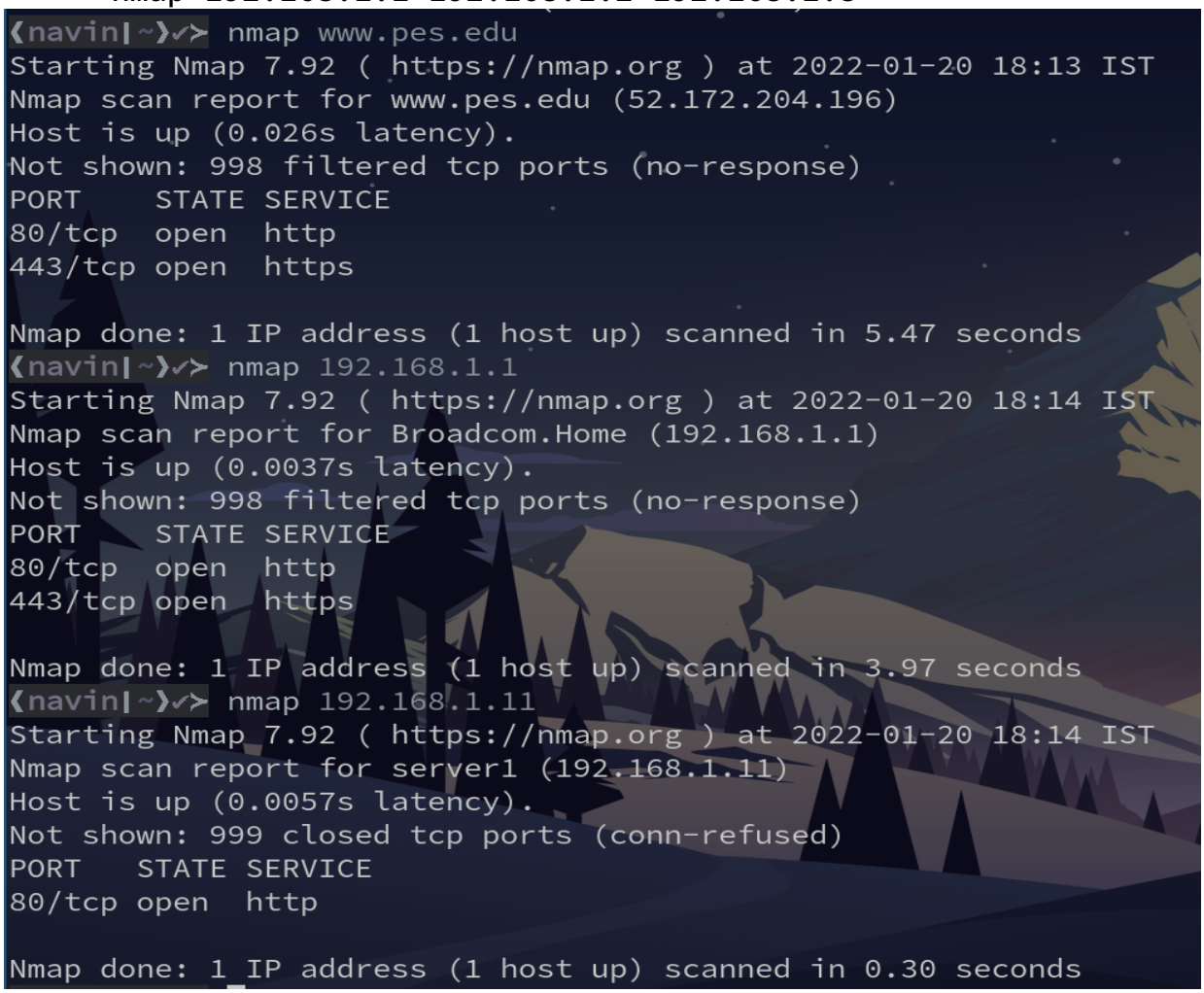
nmap www.pes.edu

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

nmap 163.53.78.128

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

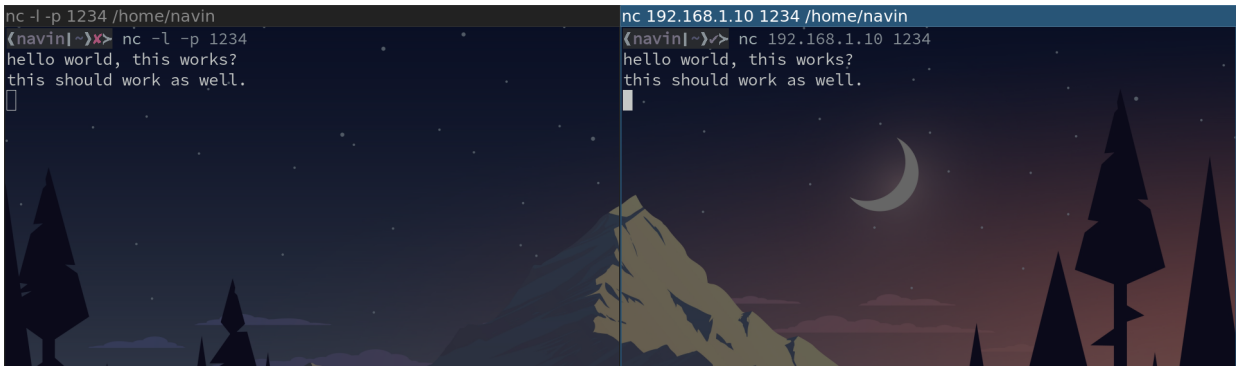
nmap 192.168.1.1 192.168.1.2 192.168.1.3



TASK 7 A) NETCAT AS CHAT TOOL

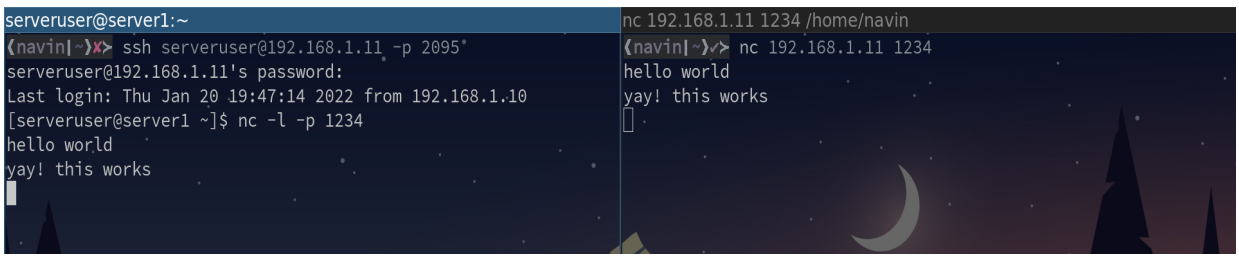
a) Intra system communication (Using 2 terminals in the same system)

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



b) Inter system communication Setup a simple switched network of 2 PCs with one acting as Web server. Assign IP addresses for both PCs. Set the capture option as described above.

Step 1: Open terminal on Server machine (Machine 1).
Step 2: Type nc -l any_portnum
Step 3: Open terminal on the Client machine (Machine 2)
Step 4: Type nc



INFO :
The two machines are :
192.168.1.10 : local machine
192.168.1.11 : my server withing my LAN, which I SSH'ed into.

TASK 7 B): USE NETCAT TO TRANSFER FILES The netcat utility can also be used to transfer files.

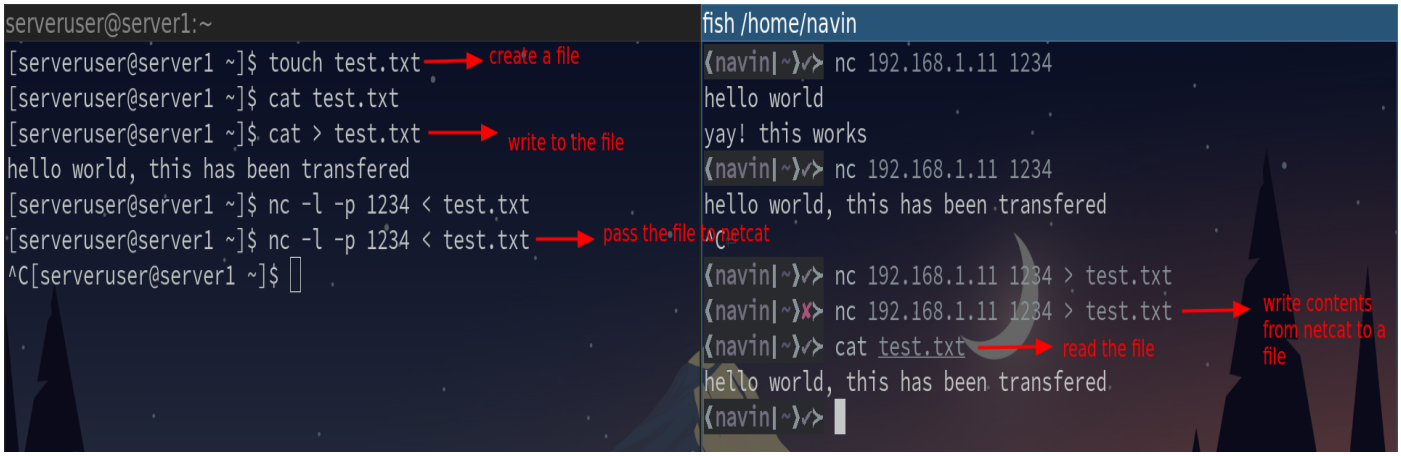
Step 1: At the server side, create an empty file named ‘test.txt’ sudo nc -l 555 > test.txt

Note: 2 students can combine for the following tasks (switch and cables can be taken from Lab technicians)

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

Step 3: Run the client as: sudo nc 10.0.2.8 555 < test.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



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

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

   304 Not Modified

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

   ping ip -v number_of_pings

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

   The nmap -O -v ip_addr

   The above command will show remote hosts and apps