CS 349 - Computer Networks Lab - Assignment 1

- Rohit Pant (160101049)

Ans 1.

a) ping -c [NUMBER_OF_PACKETS] website_or_ip
b) ping -i [TIME_INTERVAL] website_or_ip
c) sudo ping -l [NUMBER_OF_PACKETS] website_or_ip, Limit for sending such ECHO_REQUEST packets of **3 packets for non-superusers**
d) ping -s [PACKET_SIZE] website_or_ip, Total packet size = 92 bytes

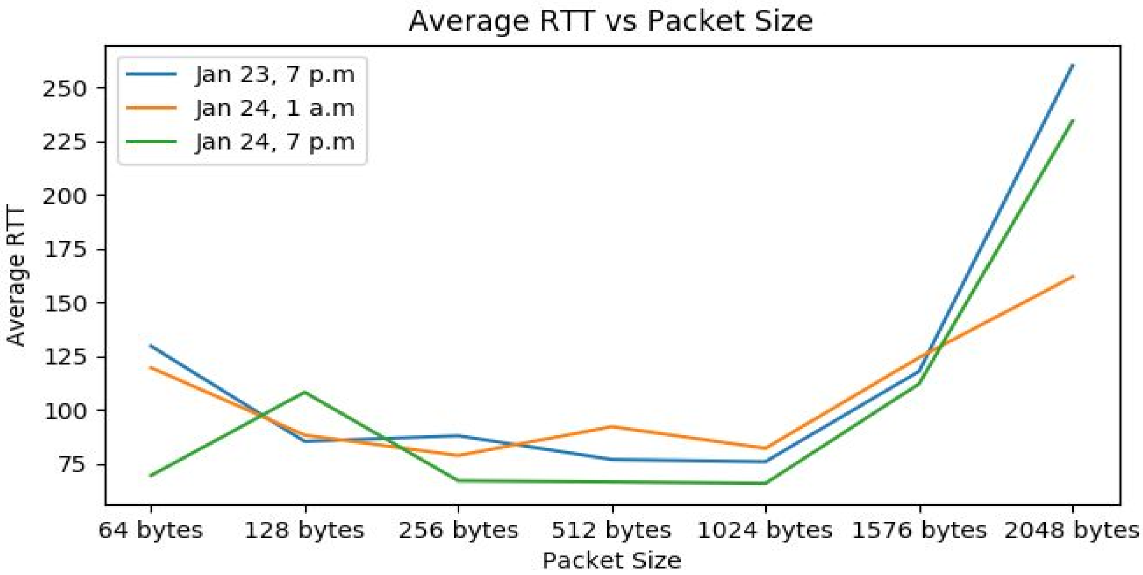
Ans 2. Readings were taken on 7 p.m, 1 a.m and 1 p.m. PC was connected to DigitalOcean VPN (Bangalore) while taking the readings.

Domain name	IP Address	Geolocation	Avg. RTT1	Avg. RTT2	Avg. RTT3	Total Avg. RTT
google.com	216.58.196.174	California, US	110.246	101.602	79.927	97.258
facebook.com	31.13.79.35	Dublin, Ireland	128.655	87.703	81.428	98.860
flipkart.com	163.53.78.128	India	82.189	80.221	68.014	80.141
mega.nz	89.44.169.135	Luxembourg	270.918	227.924	214.527	237.790
youtube.com	216.58.200.142	California, US	105.653	183.739	65.014	118.135

Flipkart.com was pinged using packet sizes varying from 64 bytes to 2048 bytes on the aforementioned times.

	64	128	256	512	1024	1576	2048
Avg. RTT1	129.774	85.471	88.078	77.073	76.071	118.018	260.072
Avg. RTT2	119.723	88.410	78.294	92.336	82.249	124.430	161.965
Avg. RTT3	69.610	108.324	67.205	66.659	65.953	112.215	234.320

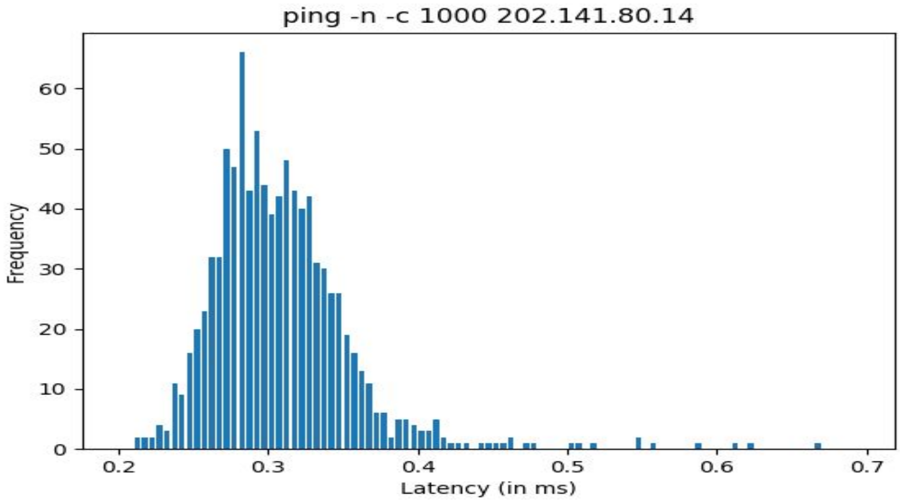
Packet loss :- Packet loss was found to be 0% in all cases thus showcasing a stable network connection and good load balancing on the server side. In general packet loss may occur due to heavy traffic leading to packet collisions.
RTT vs Distance :- Based on the readings we can conclude that RTT is weakly correlated with distance. With increasing distance usually more hops are required leading to greater latency due to processing delays at each hop. But this is not the only factor as many companies may have highly efficient and robust servers, thus minimizing this delay. So, the correlation is weak.
RTT vs Packet SIze :- RTT was found to be constant for sizes less than 1024 bytes. After that it was found to increase quite rapidly. This happens due to the fact the Maximum Transmission Unit is set to 1500 by default. Thus if packet size increases beyond 1500, it is split into multiple frames of maximum size 1500. This causes the RTT to increase.



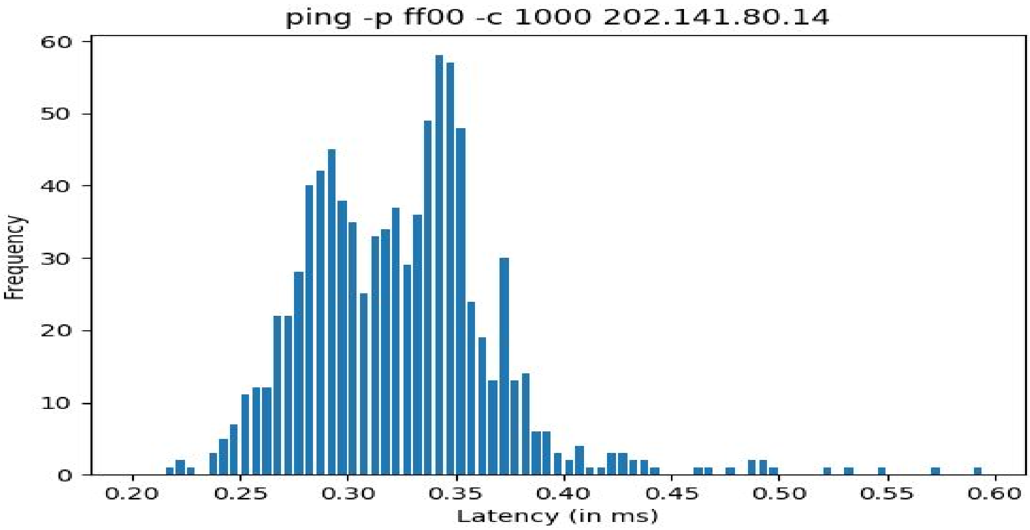
RTT vs Time of day :- A nearly constant trend was seen to be followed wherein the RTT was found to be lowest at around 1 p.m. The RTT was also less at around 1 a.m whereas it was found to be highest around 7 p.m. This corroborates the fact that website traffic is at its peak in the evening as opposed to the afternoon and midnight.

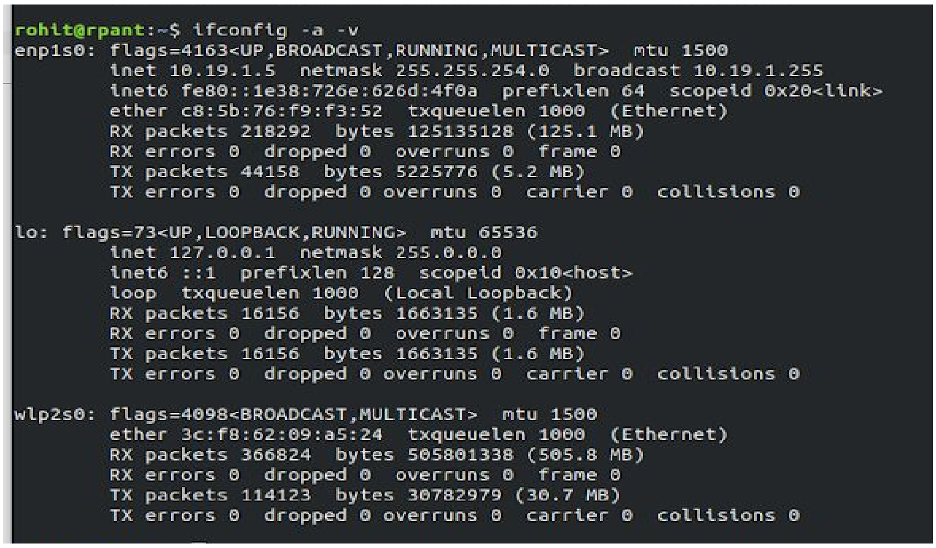
Ans 3.

Command	Packets Sent	Packets Received	Packet Loss Rate	Min. Latency	Max. Latency	Mean Latency	Median Latency
ping -n 202.141.80.14	1000	958	4.2%	0.213	2.002	0.316	0.305
ping -p ff00 202.141.80.14	1000	901	9.9%	0.219	2.599	0.332	0.3325



By the histograms we can see that the latency times have an approximately log normal distribution. With the '-n' option no attempt is made to lookup symbolic names for host addresses thus leading to a slightly better mean latency. In the second case "1111111100000000" will be sent along with the packet. As this string contains long patterns consisting of no transitions from 0 to 1 or vice versa the clock is more likely to go out of sync, thus leading to a higher packet loss rate.



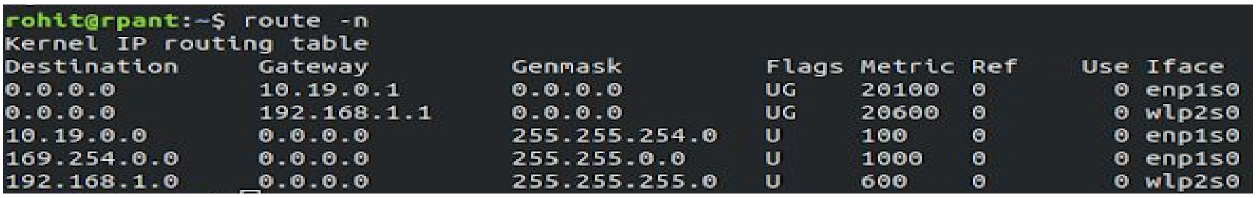


```
rohit@rpant:~$ ifconfig -a -v
enp1s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.19.1.5  netmask 255.255.254.0  broadcast 10.19.1.255
        inet6 fe80::1e38:726e:626d:4f0a  prefixlen 64  scopeid 0x20<link>
        ether c8:5b:76:f9:f3:52  txqueuelen 1000  (Ethernet)
        RX packets 218292  bytes 125135128 (125.1 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 44158  bytes 5225776 (5.2 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 16156  bytes 1663135 (1.6 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16156  bytes 1663135 (1.6 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp2s0: flags=4098<BROADCAST,MULTICAST>  mtu 1500
        ether 3c:f8:62:09:a5:24  txqueuelen 1000  (Ethernet)
        RX packets 366824  bytes 505801338 (505.8 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 114123  bytes 30782979 (30.7 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Ans 4. ifconfig stands for "interface configuration". It is used to view and change the configuration of the network interfaces on your system.

On running the command, I found that there were 3 active network interfaces on my PC - ‘enp1s0’ wired ethernet interface, ‘lo’ loopback interface and ‘wlp2s0’ wireless ethernet interface. The **UP** and **RUNNING** flags indicate that the interface is active. The **BROADCAST** and **MULTICAST** show that these features are enabled. **MTU** (Maximum Transmission Unit) signifies the largest possible packet size. ‘**netmask**’ and ‘**broadcast**’ give the Netmask and ‘Broadcast Address’ of the interface. Broadcast address is the address used to broadcast to the network connected to the interface. ‘**inet**’ and ‘**inet6**’ give the machine’s IPv4 and IPv6 addresses associated with the interface respectively. ‘prefixlen’ specifies the number of bits in the IP address that are to be used as the subnet mask. ‘Netmask’ and ‘broadcast’ The ‘**ether**’ gives the MAC Address. ‘**txqueuelen**’ (1000) gives the length of the transmit queue of the interface. **RX and TX packets** gives the number of packets received and transmitted respectively. The number of dropped, overrun, collided packets are also displayed.

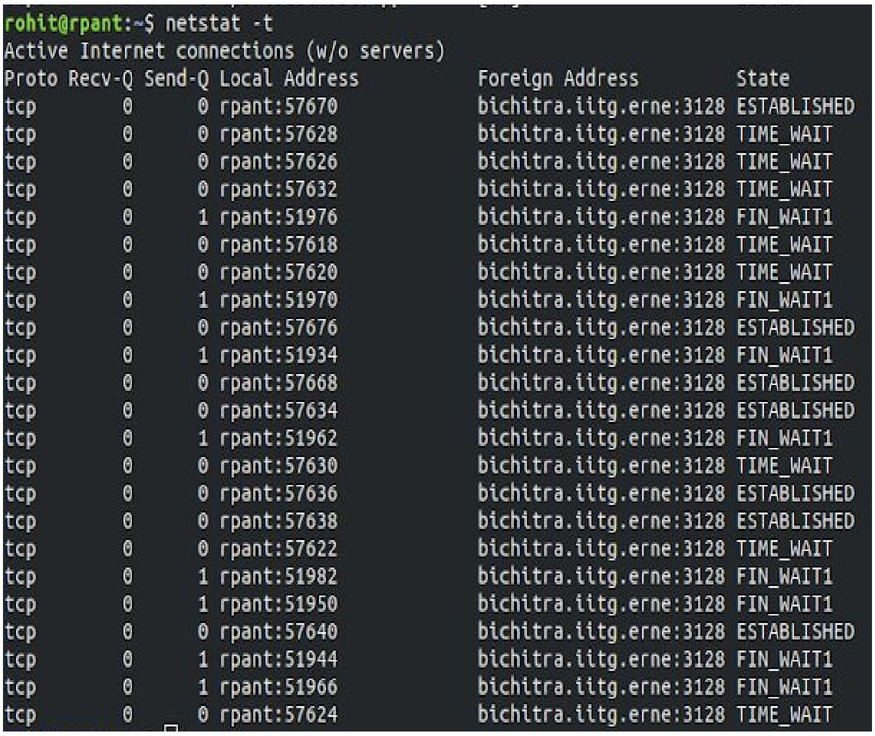
Route command is used to show/manipulate the IP routing table. It is primarily used to setup static routes to specific host or networks via an interface. ‘**Destination**’ column gives the destination network. ‘**Gateway**’ and the ‘**Genmask**’ columns give the gateway and netmask to be used for the destination network. The gateway



```
rohit@rpant:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.19.0.1       0.0.0.0         UG    20100  0        0 enp1s0
0.0.0.0         192.168.1.1     0.0.0.0         UG    20600  0        0 wlp2s0
10.19.0.0       0.0.0.0         255.255.254.0   U     100    0        0 enp1s0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 enp1s0
192.168.1.0     0.0.0.0         255.255.255.0   U     600    0        0 wlp2s0
```

address is set to '*' by default if none is specified. The **U flag** indicates that the given route is ‘up’. The **G flag** signifies that the gateway defined in the ‘Gateway’ column be used. **Metric** is a relative measure of distance between my device and the destination (usually measured in hops). ‘**Ref**’ gives the number of references to a route. The ‘**Iface**’ column shows the network interfaces on the device. ‘enp1s0’ and ‘wlp2s0’ are the wired and wireless Ethernet interfaces respectively. The route command supports several options. The **-n option** shows numerical addresses instead of trying to determine symbolic host names. The **-v option** displays a verbose operation. The del and add options are used to delete and add a route respectively. The **-C option** is used to operate on the kernel's routing cache.

Ans 5. **Netstat** command displays various network related information such as network connections, routing tables, interface statistics etc. It is a command line utility that tells us about all the tcp/udp/unix socket connections on our system. It provides list of all connections that are currently established or are in waiting state.



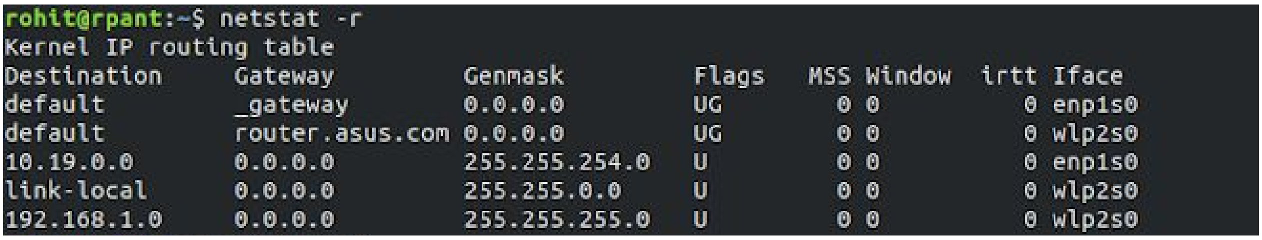
```
rohit@rpant:~$ netstat -t
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 rpant:57670             bichitra.iitg.erne:3128 ESTABLISHED
tcp        0      0 rpant:57628             bichitra.iitg.erne:3128 TIME_WAIT
tcp        0      0 rpant:57626             bichitra.iitg.erne:3128 TIME_WAIT
tcp        0      0 rpant:57632             bichitra.iitg.erne:3128 TIME_WAIT
tcp        0      1 rpant:51976             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      0 rpant:57618             bichitra.iitg.erne:3128 TIME_WAIT
tcp        0      0 rpant:57620             bichitra.iitg.erne:3128 TIME_WAIT
tcp        0      1 rpant:51970             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      0 rpant:57676             bichitra.iitg.erne:3128 ESTABLISHED
tcp        0      1 rpant:51934             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      0 rpant:57668             bichitra.iitg.erne:3128 ESTABLISHED
tcp        0      0 rpant:57634             bichitra.iitg.erne:3128 ESTABLISHED
tcp        0      1 rpant:51962             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      0 rpant:57630             bichitra.iitg.erne:3128 TIME_WAIT
tcp        0      0 rpant:57636             bichitra.iitg.erne:3128 ESTABLISHED
tcp        0      0 rpant:57638             bichitra.iitg.erne:3128 ESTABLISHED
tcp        0      0 rpant:57622             bichitra.iitg.erne:3128 TIME_WAIT
tcp        0      1 rpant:51982             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      1 rpant:51950             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      0 rpant:57640             bichitra.iitg.erne:3128 ESTABLISHED
tcp        0      1 rpant:51944             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      1 rpant:51966             bichitra.iitg.erne:3128 FIN_WAIT1
tcp        0      0 rpant:57624             bichitra.iitg.erne:3128 TIME_WAIT
```

The ‘**netstat -t**’ is used to display all TCP connections.

The ‘**Proto**’ column tells about the protocol used by the socket. ‘**Recv-Q**’ is the count of bytes not copied by the user program connected to this socket. ‘**Send-Q**’ is the count of bytes not acknowledged by the remote host. ‘**Local Address**’ and ‘**Foreign Address**’ give the address and port number of the local end and remote end of the socket respectively. ‘**State**’ gives the status of the socket.

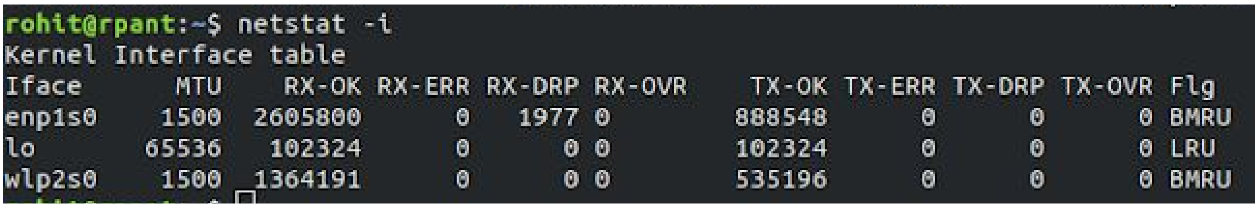
‘**netstat -r**’ is used to display the kernel routing tables. The output is nearly identical to that of the ‘route’ command. The ‘**Destination**’ gives the address to the destination network or host. Before sending a packet, this table is looked up to see if the destination address matches any entry. Then the packet is forwarded to the location specified in the corresponding entry in the ‘**Gateway**’ column. The ‘**Genmask**’ gives the netmask to be used for the destination network or host. **Flags** are used to indicate various states, for e.g - **U** (route is up), **H** (target is a host), **G** (use specified gateway), **R** (reinstate route for dynamic routing), **D** (dynamically installed by daemon or redirect), **M** (modified from routing daemon or redirect). The ‘**MSS**’ and ‘**WIndow**’ give the default maximum segment size and the default window size for TCP connections over this route. ‘**irtt**’ gives the Initial Round Trip Time. ‘**Iface**’ gives the network interface to which packets for this route will be sent.

‘**netstat -i**’ is used to display the network interface status. The ‘Iface’ column shows the 3 active interfaces on my device, namely - ‘enp1s0’ (Wired Ethernet), ‘wlp2s0’ (Wireless Ethernet) and ‘lo’ (Loopback). The ‘**MTU**’ column



```
rohit@rpant:~$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 enp1s0
default         router.asus.com 0.0.0.0         UG        0 0          0 wlp2s0
10.19.0.0       0.0.0.0         255.255.254.0   U         0 0          0 enp1s0
link-local      0.0.0.0         255.255.0.0     U         0 0          0 wlp2s0
192.168.1.0     0.0.0.0         255.255.255.0   U         0 0          0 wlp2s0
```

gives the Maximum Transmission Unit size of an interface. The **RX** and **TX OK, ERR, DROP and OVR** columns give the packets obtained error-free, damaged, dropped and lost because of overrun respectively for received or transmitted packets. The ‘**Flg**’ column gives various characteristics of the interface, for ex - **B** (Broadcast address sent), **L** (device is a loopback device), **R** (interface active and running), **P** (connection is point to point).



```
rohit@rpant:~$ netstat -i
Kernel Interface table
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
enp1s0    1500  2605800      0   1977 0        888548      0      0      0 BMRU
lo       65536   102324      0      0 0        102324      0      0      0 LRU
wlp2s0    1500  1364191      0      0 0        535196      0      0      0 BMRU
```

The **loopback interface** is a virtual interface. The only purpose of the loopback interface is to return the packets sent to it, i.e. whatever you send to it is received on the interface. It is used mainly for diagnostics and troubleshooting, and to connect to servers running on the local machine. It is extensively used in development phase of web projects. For example, if you run a web server, you have all your web documents and could examine them file by file. The loopback interface can allow processes to talk to each other over the "network" that doesn't actually exist.It is assigned a standard IP address on every machine: 127.0.0.1 for IPv4 and ::1 for IPv6.

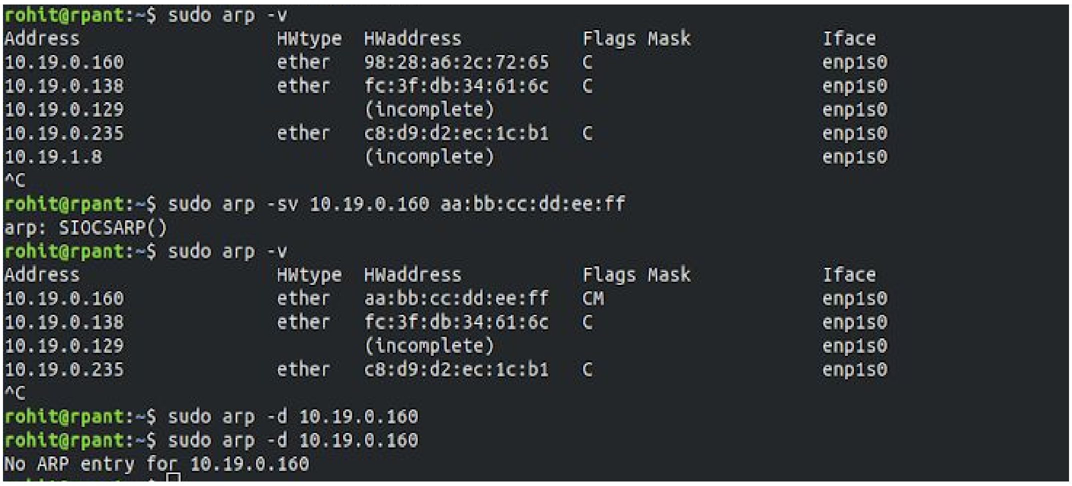
Ans 6. Readings were taken on 7 p.m, 1 a.m and 1 p.m. PC was connected to DigitalOcean VPN (Bangalore) while taking the readings.

	google.com	facebook.com	flipkart.com (*tcp probes)	mega.nz	youtube.com
Hop Count 1	12	12	12	14	12
Hop Count 2	12	12	12	14	12
Hop Count 3	12	12	12	14	10

a) A common hop between all traceroutes was 10.8.0.1 (gateway) and 142.93.208.253 or 138.197.249.22 (VPN service provider’s IP). Several common hops were found between ‘google.com’ and ‘youtube.com’ with their paths only diverging for the last 2-3 hops.
b) Most companies have servers with efficient load balancing so that no server is overloaded. Thus the workload is distributed, leading to different paths at different times of the day due to varying traffic.
c) Cases may arise where a server’s firewall may block ICMP packets to reduce network traffic. This situation was faced by me while tracerouting ‘flipkart.com’ where servers after 6-7 hops failed to respond. The above readings were obtained by using TCP probes instead of ICMP probes.
d) Yes, it is possible to do so. Ping expects an ICMP reply packet from the host. Traceroute on the other hand uses ICMP Time Exceeded packets and the concept of TTL. Each time a packet is sent the TTL value for a packet is increased by 1. Then the router decrements the TTL generating an ICMP error (Time Exceeded) if it falls to zero without receiving a reply and discards the packet. Thereafter it continues to send the next packet. Thus even in situations where ICMP reply is not generated, ICMP error is generated. So it is possible to find the path to a host even if it fails the ping experiment.

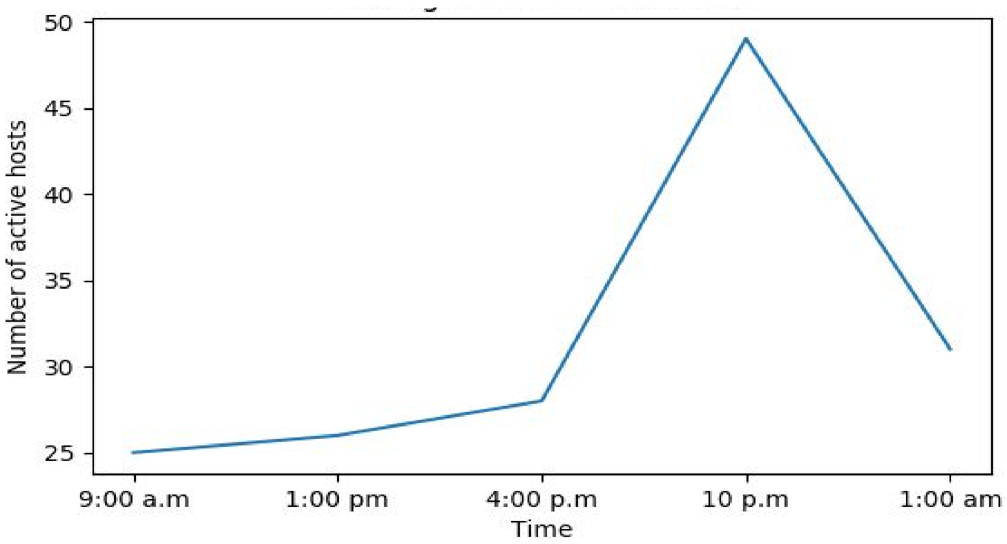
Ans 7. The arp command manipulates or displays the kernel's IPv4 network neighbour cache. It can add entries to the table, delete one, or display the current content. ARP stands for Address Resolution Protocol. **arp -a or arp -v** can be used to display the ARP table as can be seen in the photo below (the arp table has been truncated as

several entries were displayed). The ‘**Address**’ column displays the IP address while the ‘**HWaddress**’ column displays the MAC address. The ‘**Iface**’ column shows the network interface (‘enp1s0’ represents wired ethernet). The arp table caches the IP address and the MAC address to quickly find or connect to network neighbours. The ‘**sudo arp -s <IP Address> <MAC Address>**’ command can be used to manually set the MAC address for an IP. The ‘sudo arp -d <IP Address>’ command can be used to delete an entry from the table. The entries stay cached in the table for 60 seconds. The direct way to find this is to run the command -
‘**cat /proc/sys/net/ipv4/neigh/default/gc_stale_time**’. The hit and trial method is to manually make an entry into the table and check for its disappearance in time intervals using the binary search approach (eg. try 1s first, if the entry has not disappeared, repeat the process with waiting time 1*2=2s).
This case may arise when we want to relay a packet to another subnet which is connected to your subnet via a router. In this case all the devices belonging to the other subnet are given the MAC address of the router in the ARP table. Packets are sent to the router which then redirects them to their destination using its routing table.



```
rohit@rpant:~$ sudo arp -v
Address                  HWtype  HWaddress           Flags Mask            Iface
10.19.0.160              ether   98:28:a6:2c:72:65   C                     enp1s0
10.19.0.138              ether   fc:3f:db:34:61:6c   C                     enp1s0
10.19.0.129                      (incomplete)                              enp1s0
10.19.0.235              ether   c8:d9:d2:ec:1c:b1   C                     enp1s0
10.19.1.8                        (incomplete)                              enp1s0
^C
rohit@rpant:~$ sudo arp -sv 10.19.0.160 aa:bb:cc:dd:ee:ff
arp: SIOCSARP()
rohit@rpant:~$ sudo arp -v
Address                  HWtype  HWaddress           Flags Mask            Iface
10.19.0.160              ether   aa:bb:cc:dd:ee:ff   CM                    enp1s0
10.19.0.138              ether   fc:3f:db:34:61:6c   C                     enp1s0
10.19.0.129                      (incomplete)                              enp1s0
10.19.0.235              ether   c8:d9:d2:ec:1c:b1   C                     enp1s0
^C
rohit@rpant:~$ sudo arp -d 10.19.0.160
rohit@rpant:~$ sudo arp -d 10.19.0.160
No ARP entry for 10.19.0.160
```

Ans 8.



The ‘**nmap -n -sP 10.19.1.5/22**’ command was used to find the active users in Lohit Hostel.
By looking at the graph we can infer that number of active users are typically low early in the morning. Thereafter the count remains nearly constant till around 5 p.m. Then (as should be expected) the number of active users increase rapidly and reach a peak at around 10:00 pm, only to fall down after midnight.