Submitted by : Aranya Aryaman
Roll Number : 170101011

Question 1:

a) ping -c "count-value" is the option required to specify the number of ECHO REQUESTS to be sent using 'ping' command.

b) ping -i "interval-value" is the option required to set the time interval in seconds b/w two ECHO REQUESTS rather than the default one second.

c) ping -l "preload-value" is the option required to send ECHO REQUEST packets to the destination one after another without waiting for a reply. The limit for sending such ECHO REQUEST packets by normal users is 3 (three).

d) ping -s "packet-size-value" is the command to set the ECHO REQUEST packet size (in bytes). The total packet size will be 40 bytes if the packet size is 32 bytes because 8 bytes of ICMP Header Data is added to the packet.

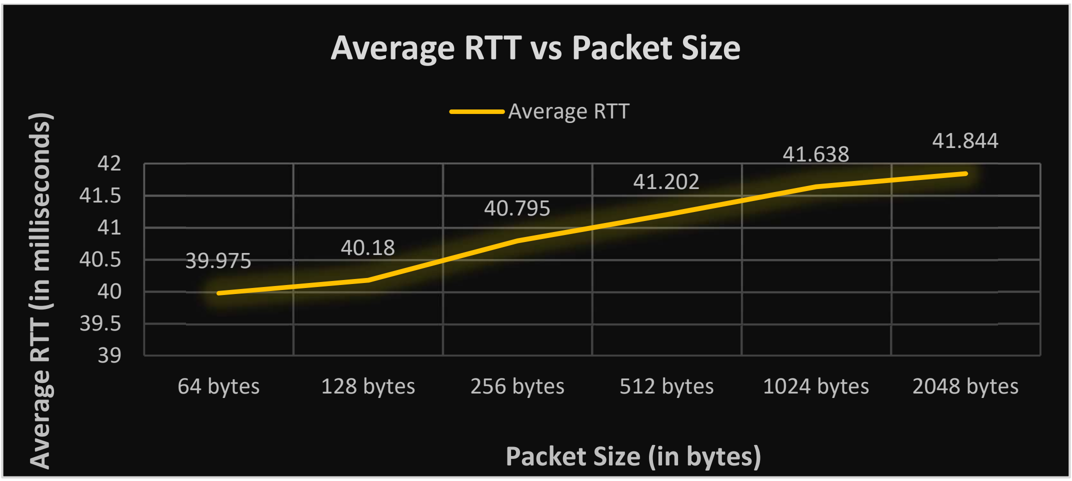
Question 2:

Host Name	Average RTT 10pm (14-01-2020)	Average RTT 8am (15-01-2020)	Average RTT 4pm (15-01-2020)	Average RTT	Distance (in miles)	Percentage Loss
google.com	39.275	39.431	39.608	39.438	8185	0%
facebook.com	21.225	21.405	21.591	21.407	7534	0%
youtube.com	39.481	40.080	40.665	40.075	8185	0%
hotmail.com	No RTT	No RTT	No RTT	N/A	7996	100%
yahoo.com	66.494	66.583	66.739	66.605	7887	0%
web.whatsapp.com	21.977	22.303	25.993	23.424	7534	0%

- Yes, there exists a case where the packet loss percentage is greater than 0%. In-fact there is a 100% loss when we ping hotmail.com using *"spfld.com/ping.html"*. There are many possible reasons. It is possible that there could be restrictions on the source IP address that can be accessed. It is also possible that there is some network congestion or target IP address might not have any network device connected with it. It is also possible that the firewall mayn't allow to ping any particular IP address by simply blocking it.

- There doesn't seem to be a strong inter-relation b/w the geographical location of the hosts and RTT. All the six selected hosts are very close to each another in terms of geographical location whereas the RTT are too scattered. It actually depends on how many routers/switches come in the way between hosts and the transmitter. Since, it is seen that there are higher number of switches/routers if the distance is more, thus it can be said that there is a weak positive correlation b/w distance and RTT.

google.com was picked to repeat the above experiment with different packet sizes. The following graph was obtained with different packet-sizes.



Packet Size (in bytes)	64	128	256	512	1024	2048
Average RTT (in ms)	39.975	40.18	40.795	41.202	41.638	41.844

- It is possible that there is packet loss of 100% for some site while using packet size 2048 bytes if the frame size exceeds the MTU size of the interface (1500 in this case). Using fragmentation, the host can send ECHO REQUESTS for this packet size.
- With increase in packet size, RTT increases as can be seen from the above graph. The increase is almost linear but not perfectly linear.
- There is also an influence of time on RTT measurements due to different time-zones of different countries in each continent and number of users in different countries. For ex- Higher RTT can be observed during daytime in India while lower RTT is observed post sunset.

Question 3:

The chosen IP is 172.17.0.23 *(intranet.iitg.ac.in)*

a) The packet loss rate for:

ping -n 1000 172.17.0.23 is 0.1%

ping -p ff00 172.17.0.23 is 0.3%

b) The following tables can be used to visualize the different kinds of latencies for both the commands:

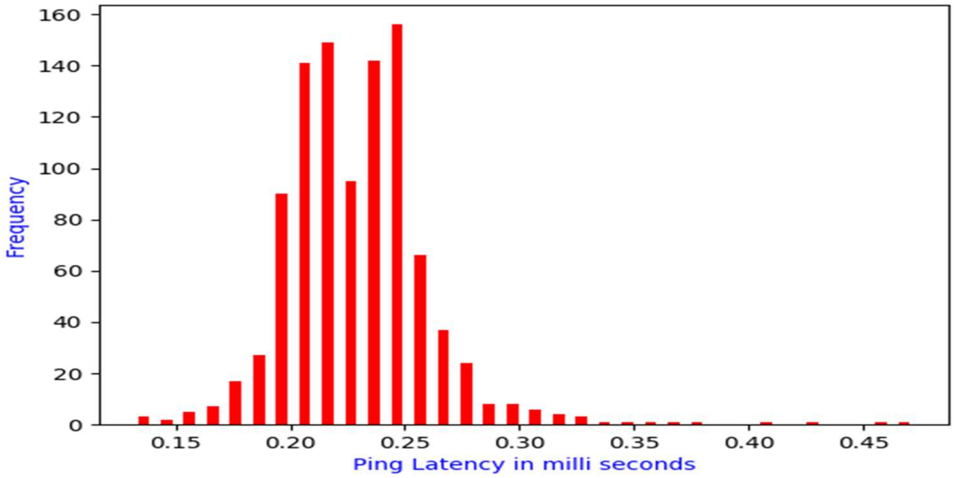
Minimum Latency	Maximum Latency	Mean Latency	Median Latency
0.131	0.473	0.229	0.228

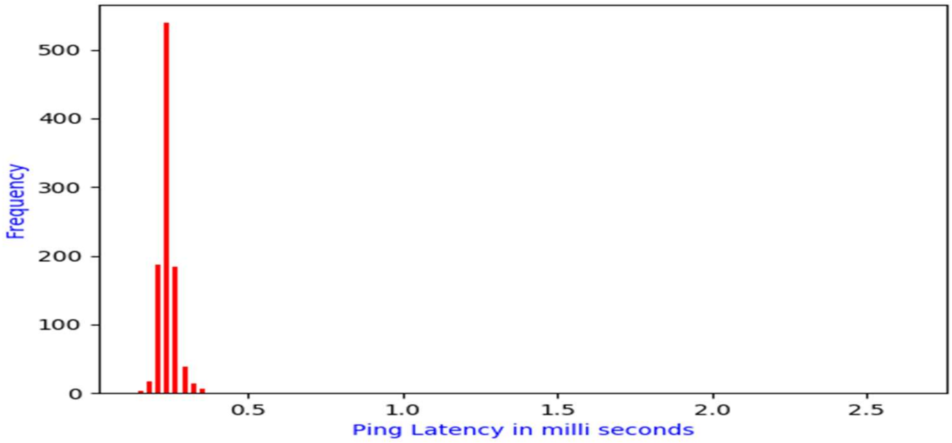
Table for "*ping -n 1000 172.17.0.23*"

Minimum Latency	Maximum Latency	Mean Latency	Median Latency
0.136	2.647	0.246	0.241

Table for "*ping -p ff00 172.17.0.23*"

c) The normal distribution for the two cases is as follows:





d) For *ping -n 1000 172.17.0.23,* the RTT values lied b/w 0.131 and 0.473 as can be visualized from the table in Q2. There was a packet loss of 0.1% i.e. out of the 1000 packets sent, 999 of them were received back successfully.

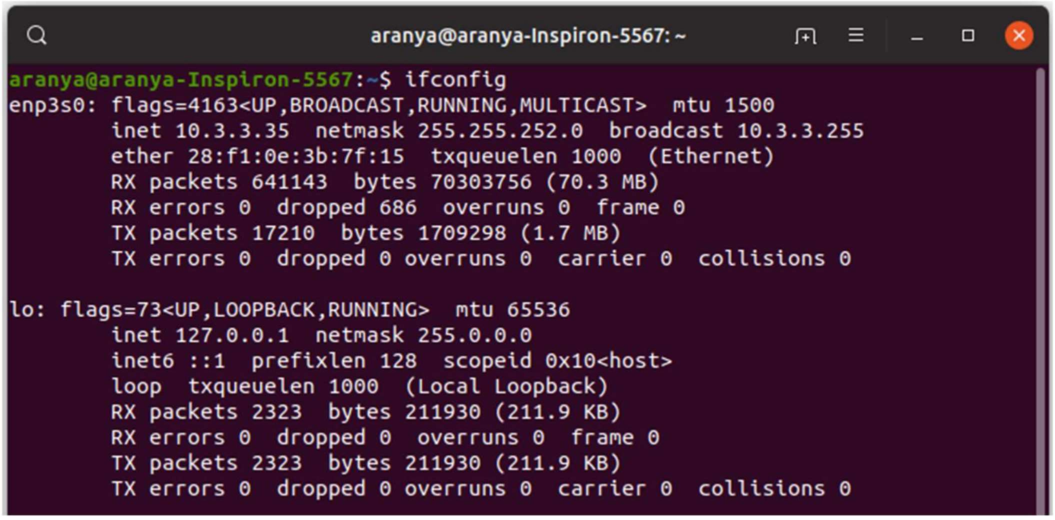
For *ping -p ff00 172.17.0.23*, the RTT values lie b/w 0.136 and 2.647. Since, almost 992 of the received packets had RTT values b/w 0.136 and 0.4, a histogram was used to represent the latencies using this command. There was a packet loss of 0.3% in this case i.e. out of the 1000 packets sent, 997 of them were received back successfully.

The standard deviation for *ping -n* was found to be 0.034 while that for *ping -p* was found to be 0.1. *ping -n* carries the default pattern whereas *ping -p* carries the given i/p pattern.

Question 4:

a) **ifconfig** (interface configuration) command is used to configure the kernel-resident network interfaces. It is used at the boot time to set up the interfaces as necessary. After that, it is usually used when needed during debugging or when you need system tuning. Also, this command is used to assign the IP address and netmask to an interface or to enable or disable a given interface.

ifconfig command shows the above output for my case and it usually shows a similar output for anyone else. The following explanations can be used to understand different things present in the above figure:



```
aranya@aranya-Inspiron-5567:~$ ifconfig
enp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.3.3.35  netmask 255.255.252.0  broadcast 10.3.3.255
        ether 28:f1:0e:3b:7f:15  txqueuelen 1000  (Ethernet)
        RX packets 641143  bytes 70303756 (70.3 MB)
        RX errors 0  dropped 686  overruns 0  frame 0
        TX packets 17210  bytes 1709298 (1.7 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 2323  bytes 211930 (211.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2323  bytes 211930 (211.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

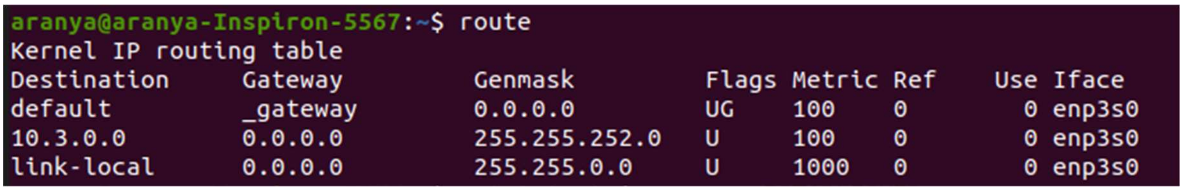
1) enp3s0 is the first Ethernet interface. Subsequent ones would be named as enp3s1, enp3s2, etc.
2) lo is the loopback interface. This is a special network interface, which the system uses to communicate with itself.

UP – This flag indicates that the kernel modules related to the Ethernet interface has been loaded.
BROADCAST – denotes that the Ethernet device supports broadcasting - a necessary characteristic to obtain IP address via DHCP.
RUNNING – The interface is ready to accept data.
MULTICAST – This indicates that the Ethernet interface supports multicasting. Multicasting can be best understood by relating to a radio station. Multiple devices can capture the same signal from the radio station but if and only if they tune to a particular frequency.
mtu 1500 – short form for Maximum Transmission Unit is the size of each packet received by the Ethernet card. The value of MTU for all Ethernet devices by default is set to 1500.
inet – indicates the machine's IP Address.
netmask – is the network mask which we passed using the netmask option
ether 28:f1:0e:3b:7f:15 – This is the hardware address or MAC address which is unique to each Ethernet card which is manufactured. First part is manufacturers' code and second part is the device id.
txqueuelen 1000 – This limits the number of packets in the transmission queue in the interface's device driver.
RX packets, TX packets – This shows the total number of packets received and transmitted respectively.
Errors – Number of damaged packets transmitted/received.
Dropped – Number of dropped packets due to reception error.
Overruns – Number of transmitted/received packets that experienced data overruns.
Frame – Number of transmitted/received packets that experienced frame errors.
Collisions - The value of this field should ideally be 0. If it has a value greater than 0, it could mean that the packets are colliding while traversing your network - a sure sign of network congestion.

b) Some of the options with the *ipconfig* command are:

- -a – To display information for all network interfaces, even if they are down.
- -v – For verbose mode, to display additional information for certain error conditions.
- -s – Display a short list, instead of details.
- up – This option is used to activate the driver for the given interface.
- down – This option is used to deactivate the driver for the given interface.
- -arp – This option is used to enable/disable the use of ARP protocol on an interface.
- -promisc – This option is used to enable/disable the promiscuous mode on an interface.
- mtu N – This user uses this parameter to set the Maximum Transfer Unit (MTU).

c)



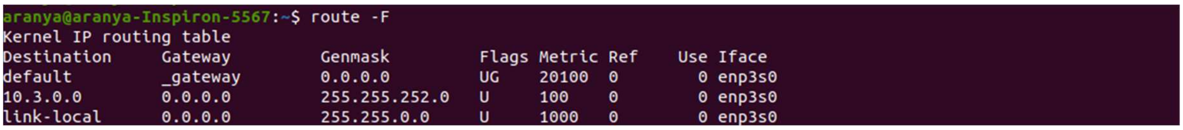
```
aranya@aranya-Inspiron-5567:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    100    0        0 enp3s0
10.3.0.0        0.0.0.0         255.255.252.0   U     100    0        0 enp3s0
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 enp3s0
```

The *route* command is used to view and make changes to the kernel routing table. Running *route* at the command line without any options displays the routing table entries.

This shows us how the system is currently configured. If a packet comes into the system and has a destination in the range 10.3.0.0 through 10.3.0.255, then it is forwarded to the gateway 0.0.0.0 – a special address which represents an invalid or non-existent destination. So, in this case, our system will not route these packets.

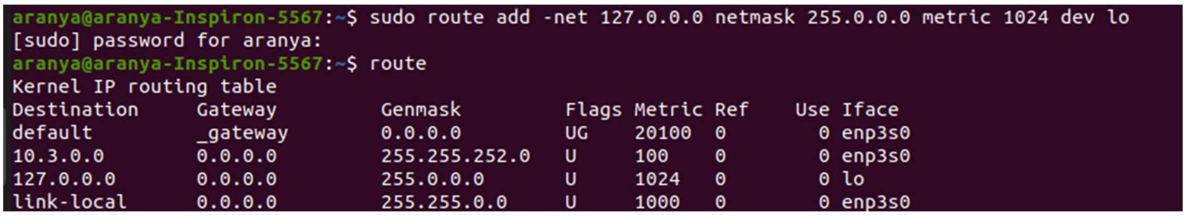
If the packets is not in this IP Address range, it is forwarded to the default gateway which in this case is 10.3.0.254 and that system will determine how to forward the traffic on to the next step towards its destination.

d)



```
aranya@aranya-Inspiron-5567:~$ route -F
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    20100  0        0 enp3s0
10.3.0.0        0.0.0.0         255.255.252.0   U     100    0        0 enp3s0
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 enp3s0
```

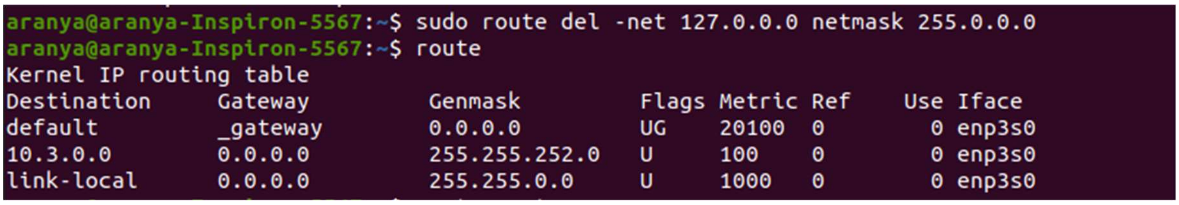
route -F - operate on the kernel's FIB (Forwarding Information Base) routing table. This is the default.



```
aranya@aranya-Inspiron-5567:~$ sudo route add -net 127.0.0.0 netmask 255.0.0.0 metric 1024 dev lo
[sudo] password for aranya:
aranya@aranya-Inspiron-5567:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    20100  0        0 enp3s0
10.3.0.0        0.0.0.0         255.255.252.0   U     100    0        0 enp3s0
127.0.0.0       0.0.0.0         255.0.0.0       U     1024   0        0 lo
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 enp3s0
```

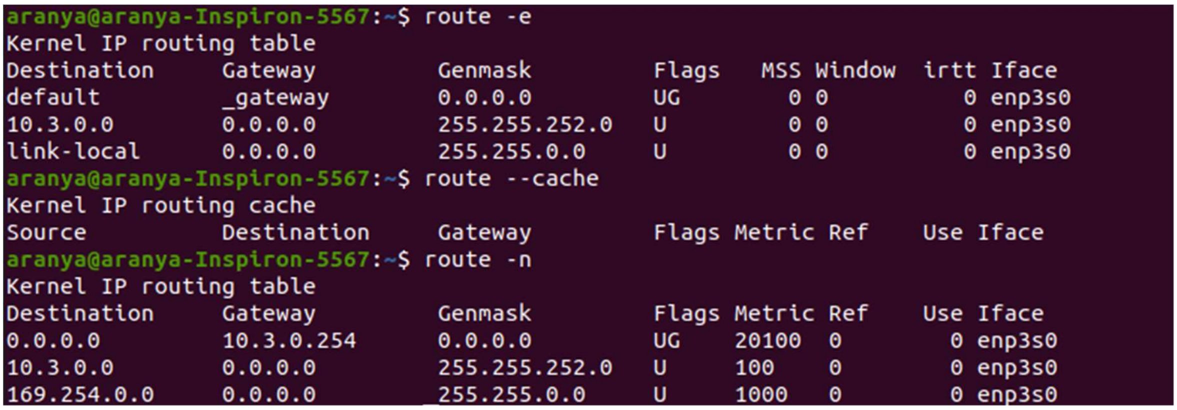
route add – Adds a new route.
netmask – When adding a network route, netmask to be used.
metric – Sets the metric field in the routing table.

dev – Forces the route to be associated with a particular device



```
aranya@aranya-Inspiron-5567:~$ sudo route del -net 127.0.0.0 netmask 255.0.0.0
aranya@aranya-Inspiron-5567:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    20100  0        0 enp3s0
10.3.0.0        0.0.0.0         255.255.252.0   U     100    0        0 enp3s0
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 enp3s0
```

route del – Deletes a route.



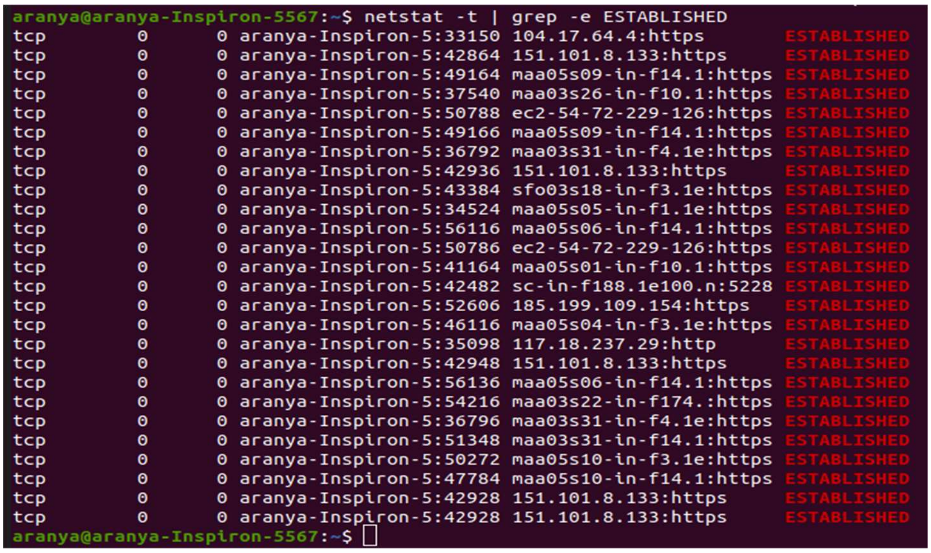
```
aranya@aranya-Inspiron-5567:~$ route -e
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 enp3s0
10.3.0.0        0.0.0.0         255.255.252.0   U         0 0          0 enp3s0
link-local      0.0.0.0         255.255.0.0     U         0 0          0 enp3s0
aranya@aranya-Inspiron-5567:~$ route --cache
Kernel IP routing cache
Source          Destination     Gateway         Flags Metric Ref    Use Iface
aranya@aranya-Inspiron-5567:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.3.0.254      0.0.0.0         UG    20100  0        0 enp3s0
10.3.0.0        0.0.0.0         255.255.252.0   U     100    0        0 enp3s0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 enp3s0
```

route -e – Uses netstat format for displaying the routing table.
route -C / route --cache – Operates on the kernel's routing cache.
route -n – Shows numerical addresses instead of symbolic hostnames.

Question 5:

a) *netstat* provides information and statistics about protocols in use and current TCP/IP network connections. It is used for finding problems in the network, to determine the amount of traffic on the network as a performance measurement and for checking our network configuration and activity.

b)

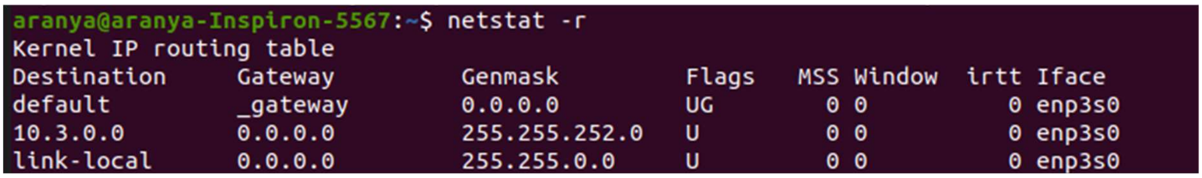


```
aranya@aranya-Inspiron-5567:~$ netstat -t | grep -e ESTABLISHED
tcp        0      0 aranya-Inspiron-5:33150 104.17.64.4:https       ESTABLISHED
tcp        0      0 aranya-Inspiron-5:42864 151.101.8.133:https     ESTABLISHED
tcp        0      0 aranya-Inspiron-5:49164 maa05s09-in-f14.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:37540 maa03s26-in-f10.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:50788 ec2-54-72-229-126:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:49166 maa05s09-in-f14.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:36792 maa03s31-in-f4.1e:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:42936 151.101.8.133:https     ESTABLISHED
tcp        0      0 aranya-Inspiron-5:43384 sfo03s18-in-f3.1e:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:34524 maa05s05-in-f1.1e:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:56116 maa05s06-in-f14.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:50786 ec2-54-72-229-126:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:41164 maa05s01-in-f10.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:42482 sc-in-f188.1e100.n:5228 ESTABLISHED
tcp        0      0 aranya-Inspiron-5:52606 185.199.109.154:https   ESTABLISHED
tcp        0      0 aranya-Inspiron-5:46116 maa05s04-in-f3.1e:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:35098 117.18.237.29:http      ESTABLISHED
tcp        0      0 aranya-Inspiron-5:42948 151.101.8.133:https     ESTABLISHED
tcp        0      0 aranya-Inspiron-5:56136 maa05s06-in-f14.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:54216 maa03s22-in-f174.:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:36796 maa03s31-in-f4.1e:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:51348 maa03s31-in-f14.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:50272 maa05s10-in-f3.1e:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:47784 maa05s10-in-f14.1:https ESTABLISHED
tcp        0      0 aranya-Inspiron-5:42928 151.101.8.133:https     ESTABLISHED
tcp        0      0 aranya-Inspiron-5:42928 151.101.8.133:https     ESTABLISHED
aranya@aranya-Inspiron-5567:~$ 
```

We use the command *netstat -t | grep -e ESTABLISHED* to list all the established TCP connections as can be seen from the above image. The fields are:

- Proto: The name of the protocol used by the socket which is tcp in this case.
- Recv-Q: The count of bytes not copied by the user program connected to this socket.
- Send-Q: The count of bytes yet to be acknowledged by the remote host.
- Local address: Address and port number of the local end of the socket.
- Foreign address: Address and port number of the remote end of the socket.
- State: The state of the socket connected in b/w the Local Address and Foreign Address. These states represent the three-way handshake communication system that TCP uses.

c)



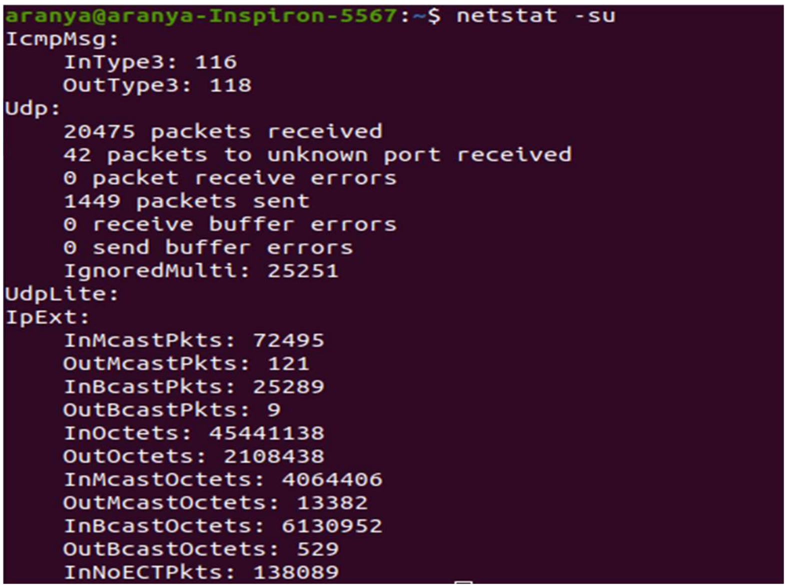
```
aranya@aranya-Inspiron-5567:~$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 enp3s0
10.3.0.0        0.0.0.0         255.255.252.0   U         0 0          0 enp3s0
link-local      0.0.0.0         255.255.0.0     U         0 0          0 enp3s0
```

netstat -r is used to get the kernel routing information. The fields are:

- Destination: The destination network or destination host.
- Gateway: The gateway to which the routing entry points.
- Genmask: The netmask for the destination net; 0.0.0.0 for default route.
- Flags: This signifies route is up to gateway or host.
- MSS: Default maximum segment size for TCP connection over route.
- Window: Default window size over this route.
- irtt: Initial RTT (Round Trip Time).
- Iface: Interface to which packets for this route will be sent.

d) *netstat -i* can be used to display the status of all network interfaces.
netstat -i| *wc -l* can be used to figure out the number of interfaces on PC.

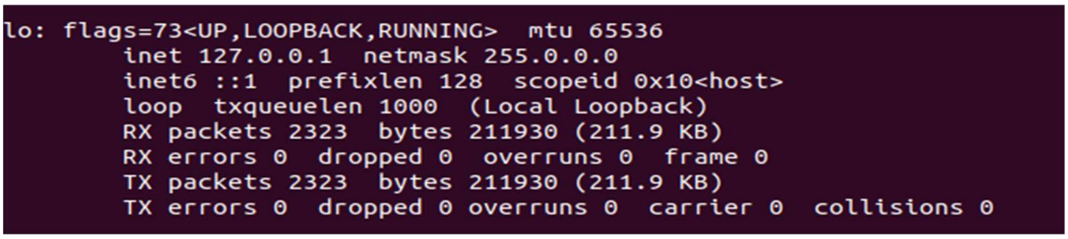
e)



```
aranya@aranya-Inspiron-5567:~$ netstat -su
IcmpMsg:
    InType3: 116
    OutType3: 118
Udp:
    20475 packets received
    42 packets to unknown port received
    0 packet receive errors
    1449 packets sent
    0 receive buffer errors
    0 send buffer errors
    IgnoredMulti: 25251
UdpLite:
IpExt:
    InMcastPkts: 72495
    OutMcastPkts: 121
    InBcastPkts: 25289
    OutBcastPkts: 9
    InOctets: 45441138
    OutOctets: 2108438
    InMcastOctets: 4064406
    OutMcastOctets: 13382
    InBcastOctets: 6130952
    OutBcastOctets: 529
    InNoECTPkts: 138089
```

netstat -su can be used to show the statistics of all the UDP connections.

f) The loopback device is a special, virtual network interface that your computer uses to communicate with itself. It is used mainly for diagnostics and troubleshooting, and to connect to servers running on the local machine. It is the very first interface to be activated. The role of a Loopback Interface comes when a network interface is disconnected--for example, when an Ethernet port is unplugged or Wi-Fi is turned off or not associated with an access point--no communication on that interface is possible, not even communication between your computer and itself.



```
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 2323  bytes 211930 (211.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2323  bytes 211930 (211.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Question 6:

a) The count of hops for traceroute for different domains is as follows:

Domains	Hop Count (18th Jan, 8am)	Hop Count (18th Jan, 1pm)	Hop Count (18th Jan, 7pm)
google.com	7	7	8
facebook.com	8	8	9
youtube.com	7	7	8
hotmail.com	5 (Firewall Reached)	5 (Firewall Reached)	5 (Firewall Reached)
yahoo.com	11	11	11
web.whatsapp.com	8	8	9

The list of common hops for different domains are as follows:

google.com	91.194.90.1, 212.18.7.63, 172, 217.20.238
facebook.com	212.78.183.245, 204.15.22.41, 31.13.84.36
youtube.com	212.18.7.63, 93.104.240.55, 172.217.22.206
hotmail.com	91.194.90.1, 213.248.101.77, 62.115.120.119, 62.115.112.199, 62.115.56.198
yahoo.com	212.78.183.245, 212.36.140.127, 216.115.104.124, 98.138.219.232
web.whatsapp.com	157.240.36.125, 157.240.20.52

b) Yes, route to same host changes at different times of the day. One of the key reasons for the same is that the destination host utilizes multiple Internet servers to handle incoming requests, so it shows different IP addresses. There is fast switching with which after a packet was sent to the next hop, the routing information about how to get to the destination is stored in a fast cache. When the router receives another packet that is directed to the same destination it uses the cache. Therefore, if some router's IP are selected from the routing table, which was used earlier and now is found to be inactive then only another router IP will be selected.

c) Yes, traceroute for *hotmail.com* did not find complete paths to the hosts as it shows trace aborted at the end. Traceroute is unable to find complete paths to some host because Firewall of that host might be blocking our IP, or we need to increase max hops, as packets might not reach to destination within fixed max hops. Other reason may be packet loss between various routers in between the path.

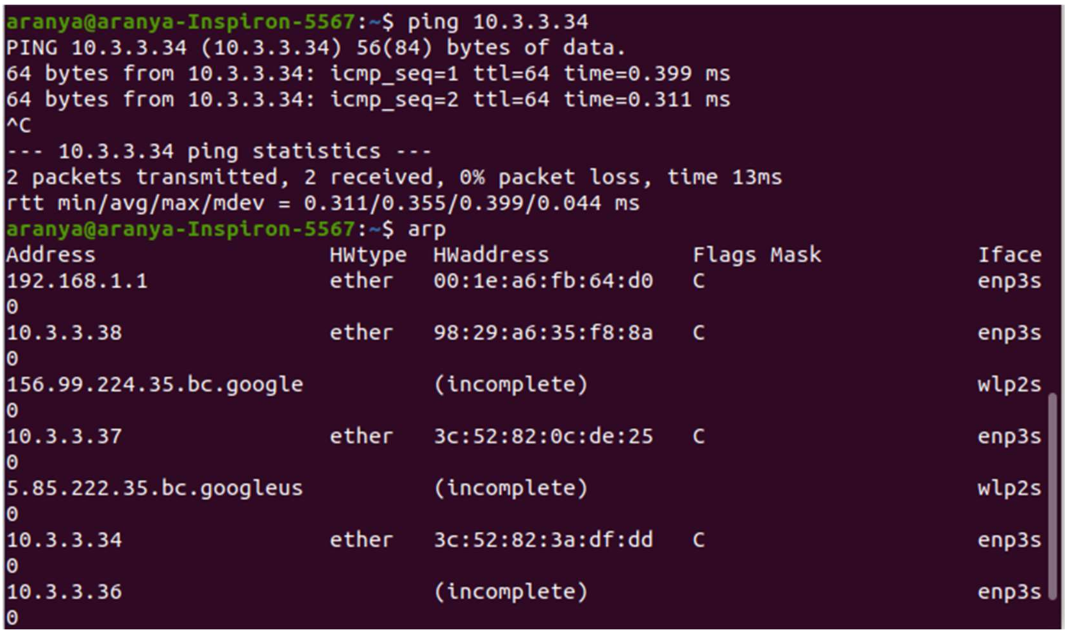
d) Yes, it is possible that tracerouting to certain hosts may be possible even though same host fail to respond to ping experiment. Failing ping is might be because of packet transmission is blocked or packet is discarded, while Traceroute uses an error message from a hop to find the route. Traceroute uses a trick to get the information, which is to manipulate the TTL (Time to Live), so the hop responds with an ICMP error (ICMP TTL exceeded).

Question 7:

a) The command arp -e can be used to show the full ARP Table for any machine. The different fields in the arp table are:

- Address: This column represents the IP address of network connections.
- Hwtype: This represents the hardware type of this machine.
- Hwaddress: This represents the hardware address of the machine of respective rows network connection.
- Flag: Each complete entry in the ARP Cache will be marked with the C flag. Permanent entries are marked with M and published entries have the P flag.
- Mask: This represents Genmask.
- Iface: This represents the network interface of respective rows connection.

b) An entry is manually added in the ARP table using the command *sudo arp -s <IP Add.> <MAC Add.>* and deleted using the command *sudo arp -d <IP Add.>*. However, the entry won't be deleted from the ARP Table from this command. This indeed changes the hardware address to a sign of <incomplete>. Another way to add IP addresses to the ARP table is by pinging to that IP address using the *ping <IP Add.>* command.



```
aranya@aranya-Inspiron-5567:~$ ping 10.3.3.34
PING 10.3.3.34 (10.3.3.34) 56(84) bytes of data.
64 bytes from 10.3.3.34: icmp_seq=1 ttl=64 time=0.399 ms
64 bytes from 10.3.3.34: icmp_seq=2 ttl=64 time=0.311 ms
^C
--- 10.3.3.34 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 13ms
rtt min/avg/max/mdev = 0.311/0.355/0.399/0.044 ms
aranya@aranya-Inspiron-5567:~$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.1              ether   00:1e:a6:fb:64:d0   C                     enp3s
0
10.3.3.38                ether   98:29:a6:35:f8:8a   C                     enp3s
0
156.99.224.35.bc.google          (incomplete)                              wlp2s
0
10.3.3.37                ether   3c:52:82:0c:de:25   C                     enp3s
0
5.85.222.35.bc.googleus          (incomplete)                              wlp2s
0
10.3.3.34                ether   3c:52:82:3a:df:dd   C                     enp3s
0
10.3.3.36                        (incomplete)                              enp3s
0
```

Here just after executing *ping 10.3.3.34,* the IP address was found to be in the ARP table alongside three other IP addresses namely (*10.3.3.37, 10.3.3.38, 10.3.3.34*) which can be seen in the above picture.

c) The command *cat /proc/sys/net/ipv4/neigh/default/gc_stale_time* can be used to find how long entries in the cache of the ARP module of the kernel remain valid and get deleted from the cache.

Trial and Error Method: A linear guess solution can be used to guess the time after which an entry is deleted from the cache of the ARP module of the kernel. It works in a similar fashion to that of Binary Search. We can guess the time of deletion to be say 360 seconds, then make the system clock 360 seconds faster and look at what happens. Now try with 180 seconds if the ARP Cache has been cleared or try with 720 seconds if it has not been cleared. This way we can get an optimum solution by optimized Hit and Trial Method.

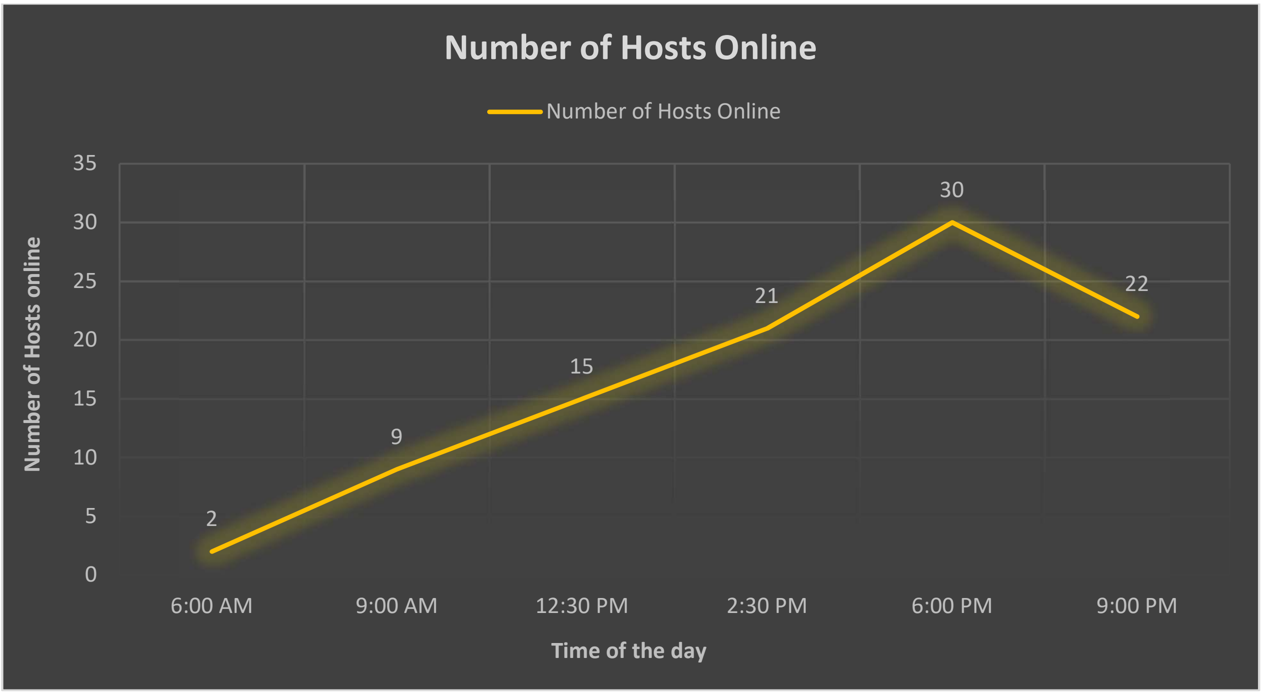
d) If two LANs on the same network have the map to the same Ethernet Address, it will confuse the switches. Any traffic sent to that IP address will be responded by both the LANs which will lead to a lot of confusion for the receiver primarily because both the devices have the same Ethernet Address. Each device will receive some of the frames destined for any one of them, which will be a complete mess.

A subnet is a smaller network created by dividing a larger network into equal parts. There are different hosts on a subnet. Different hosts on the same subnet are basically machines being plugged into the same set of hubs and switches. The machines talk to each other with their MAC Addresses which uniquely identifies each Ethernet/Wi-Fi/any other Network card. Each machine knows the IP address and not the MAC addresses. Let's say a host want to communicate with another host with IP Address "x". The first host then broadcasts a message in the network if some host knows the MAC address of any host with IP "x". Since, MAC Address of one host is unknown to another, only the target host responds back which indicates that the connection is established. One way to establish a connection using *ping x* command from the first machine.

Question 8:

The chosen <subnet range> is *172.16.112.0/26* itself. The following data was acquired after running the command *nmap -n -sP 172.16.112.0/26* at different hours on 17-01-20

Time of the day	6am	9am	12:30pm	2:30pm	6pm	9pm
Number of Hosts	2	9	15	21	30	22


Number of Hosts Online
Number of Hosts Online
Number of Hosts online
6:00 AM
9:00 AM
12:30 PM
2:30 PM
6:00 PM
9:00 PM
Time of the day