CS342: Networks Lab
Assignment 1

Tejas Khairnar
180101081

September 22, 2020

Question 1

(a) ping -c `<no.of echo requests>`

(b) ping -i `<time interval>`

(c) ping -l `<no. of packets>`
# The number of **ECHO_REQUEST** packets that could be sent one after another without waiting for a reply by a normal user is **Atmost 3** and a normal user can send those only after a time interval of **200 ms** using the option -i.

(d) ping -s `<payload/data size of the packets in bytes>`
# If the payload size of the packet is **32 bytes** then the total packet size would be **60 bytes**
(ip header **8bytes** +icmp header **20 bytes** + payload size **32 bytes**).

Question 2

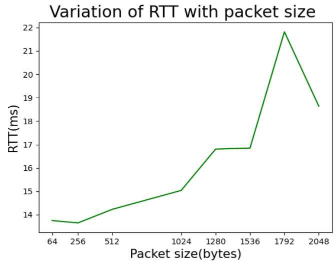
(a)

Sr. no	Host	Avg. RTT 2pm	Avg. RTT 8pm	Avg. RTT 2am	Avg. RTT
1	youtube.com	11.244 ms	33.333 ms	33.690 ms	26.089 ms
2	codeforces.com	177.471 ms	177.248 ms	228.351 ms	194.356 ms
3	amazon.com	242.407 ms	247.706 ms	280.933 ms	257.015 ms
4	sprinklr.com	13.476 ms	13.864 ms	14.750 ms	14.03 ms
5	facebook.com	11.968 ms	11.716 ms	13.730 ms	12.471 ms
6	cricbuzz.com	12.311 ms	13.059 ms	15.207 ms	13.525 ms

RTT and geographical distance from the source: There is a weak relation between the two because of the factors like increased number of hops and increased propagation delay.The packets have to go through more number of nodes and at each nodes there may be a delay i.e **processing delay**. Hence, more the routers, the more is the RTT.The time for propogation of packets increases with the distance i.e. **Propagation delay**. It is a weak relation because there are many other factors on which it depends like network traffic and the server capacities.

(b) NO, in none of the cases the packet loss is greater than 0%. But in general packet loss can occur when there is some network congestion or network traffic.Some packets may collide with other packets and result in packet loss. The ICMP packets have lower priority hence they might take longer time to process in some destination servers queue.Is the server drops all the ICMP packets then we have a 100% packet loss.

(c) Variation of RTT with packet size for sprinklr.com



Packet size(bytes)	64	256	512	1024	1280	1536	1792	2048
RTT(ms)	13.75	13.65	14.23	15.03	16.80	16.84	21.81	18.63

(d) The data shows that the RTT does not vary much till the packet size till 1024 bytes and we see a huge jump when the between 1536 and 1792(RTT almost doubles).The **MTU i.e the Maximum Transmission unit** is by default 1500 Bytes.If the packet size is less than 1500 the packets are padded to make their size 1500 bytes hence RTT is more or less similar for packets with size less than 1500 Bytes.If the packet size if greater than 1500 Bytes then the packet is broken into two new frames of size 1500 Bytes each and hence we observe almost twice the RTT.

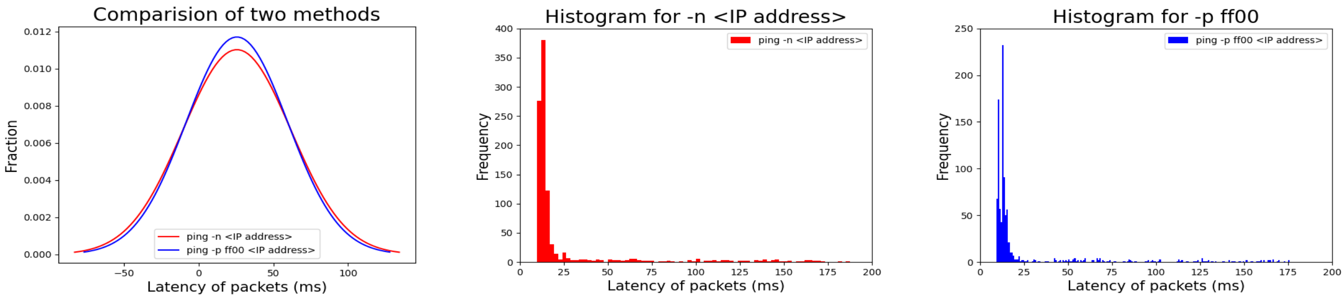
Question 3

(a) Used sprinklr.com as the host. Packet loss rate in the first case i.e **-n** is 0% whereas in other case i.e. **-p ff00** it is 0.1%.

(b)

Command	Min latency	Max latency	Mean latency	Median latency
ping -n `<IPAddress>`	9.78 ms	484 ms	25.604 ms	13.3 ms
ping -p ff00 `<IPAddress>`	9.23 ms	176 ms	25.608 ms	13.2 ms

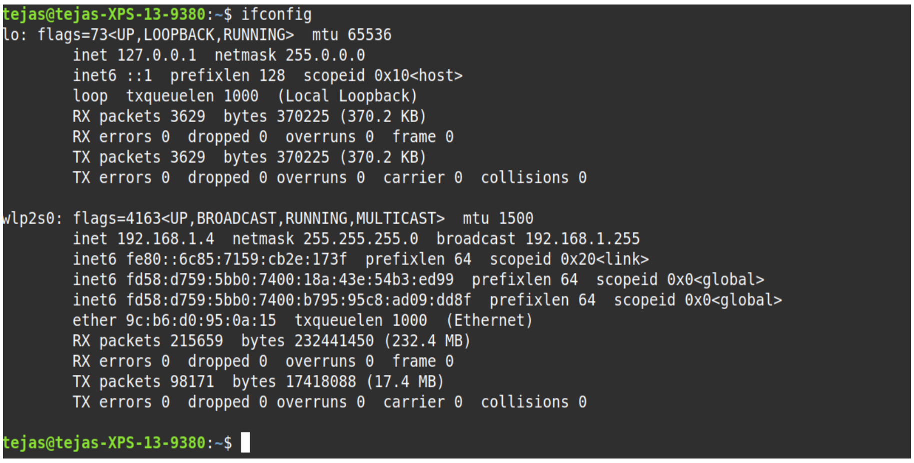
(c)



(d) The two commands are very similar and both represent a normal distribution of values expect that **-p ff00** sends the packet filled with 8 ones and then 8 zeros i.e **11111111100000000** which is used to find out the data dependency issues.This will cause problems with the synchronisation of the clocks because only one transition is present in the padding, from 1 to 0. Hence, the clocks are more likely to go out of synchronisation in case of **-p ff00** and hence we get a higher packet loss rate. While in case of **-n** option no attempt would be made to get the symbolic names for the host address hence mean latency is lower in this case.

Question 4

(a) **ifconfig** : It is a command line for UNIX-like systems tool that allows for diagnosing and configuring network interfaces.



```
tejas@tejas-XPS-13-9380:~$ ifconfig
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 3629  bytes 370225 (370.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3629  bytes 370225 (370.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.4  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::6c85:7159:cb2e:173f  prefixlen 64  scopeid 0x20<link>
        inet6 fd58:d759:5bb0:7400:18a:43e:54b3:ed99  prefixlen 64  scopeid 0x0<global>
        inet6 fd58:d759:5bb0:7400:b795:95c8:ad09:dd8f  prefixlen 64  scopeid 0x0<global>
        ether 9c:b6:d0:95:0a:15  txqueuelen 1000  (Ethernet)
        RX packets 215659  bytes 232441450 (232.4 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 98171  bytes 17418088 (17.4 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

tejas@tejas-XPS-13-9380:~$ 
```

Network interface : It is a software interface to networking hardware.Linux kernel distinguishes between two types of network interfaces i.e physical and virtual.

(i) **Physical network interface** : It represents the actual network hardware such as network interface controller (NIC). In practice found **wlp2s0** interface in the output which represents:
wl for wireless lan (Had it been **en** instead of **wl** then **en** for ethernet)
The following code is often a description of the physical placement of the device in your computer :
p2 is likely PCI bus 2
s0 is likely slot 0

(ii) **Virtual network interface** : It doesnt represent any hardware device and is usually linked to one. There are different kinds of virtual interfaces: Loopback (lo), bridges, VLANs, tunnel interfaces etc.Found **lo i.e Loopback in the output**.

Interface details:

(i) **UP**: Indicates that the kernel modules related to the interface have been loaded and interface is activated.

(ii) **BROADCAST**: Indicates that the interface is configured to handle broadcast packets,which are required for obtaining IP address via DHCP.

(iii) **RUNNING**: Indicates that the interface is ready to accept the data.

(iv) **MULTICAST**: Indicates that the interface supports multicasting.

(v) **broadcast**: broadcast address for the interface.

(vi) **netmask**: network mask for the interface.

(vii) **inet**: IPv4 address assigned to the interface.

(viii) **inet6**: IPv6 address assigned to the interface.

(ix) **scope**: It is the scope of IPv6 address. It can be link-local or global.Link-local address is used in local area network and is not routable. Global address is routable.

(x) **mtu**: It is the maximum transmission unit.

Interface stats:

(i) **RX packets**: The total number of packets received.

(ii) **RX bytes**: The total number of bytes received.

(iii) **RX error**: The total number of packets received with errors.

(iv) **RX dropped**: The total number of number of dropped packets due to unintended VLAN tags or receiving IPv6 frames when interface is not configured for IPv6.

(v) **RX overruns**: The number of packets received that experienced fifo overruns, caused by rate at which a buffer gets full and kernel isnt able to empty it.

(vi) **RX frame**: The number of of misaligned frames, i.e. frames with length not divisible by 8.

(vii) **TX packets**: The total number of packets transmitted.

(viii) **TX bytes**: The total number of bytes transmitted.

(ix) **TX errors**, **TX dropped**, **TX overruns** are similar to RX equivalents.

(x) **TX txqueulen**: The length of transmission queue.

(xi) **TX carriers**: The number of packets that experienced loss of carriers.

(xii) **TX collisions**: The number of transmitted packets that experienced Ethernet collisions.

(b) The options which can be used with ifconfig are :

(i) **-s**: This option is used to display short list instead of deatils.

(ii) **-a**: This option is used to display all the interfaces available even if they are down.

(iii) **up**: This option is used to activate driver for the given interface.

(iv) **down**: This option is used to deactivate driver for the given interface.

(c) The output of route command is organised in the form of a table with following columns:

(i) **Destination**: The destination network or destination host.

(ii) **Gateway**: The gateway address or * if none set.

(iii) **Genmask**: The netmask for the destination net, 255.255.255.255 for a host destination and 0.0.0.0 for the default route.

(iv) **Flags**:

(1) U : route is up
(2) H : target is host
(3) G : Use gateway
(4) R : reinstate route for dynamic routing
(5) D : dynamically installed by daemon or redirect
(6) M : modified from routing daemon or redirect
(7) A : installed by addrconf
(8) C : cache entry
(9) ! : reject route

(v) **Metric**: The distance to the target (usually counted in hops).

(vi) **Ref**: Number of references to this route.

(vii) **Use**: Count of lookups for the route.

(viii) **Iface**: Interface to which packets for this route will be sent.

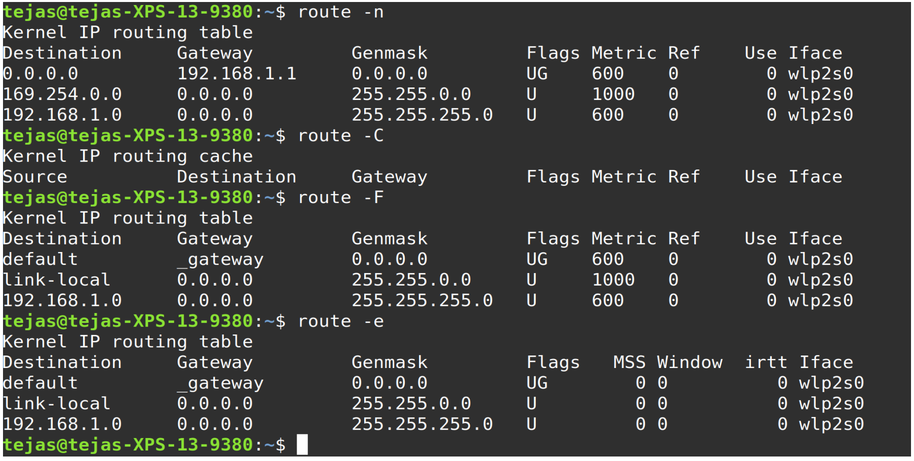
(d) The options which can be used with route command are :

(i) **-n**: This option is used to display routing table in full numeric form and does not resolve names.

(ii) **-C**: This option is used to display routing cache.

(iii) **-F**: This option is used to display Forwarding Information Base.

(iv) **-e**: This option is used to display other/more information.

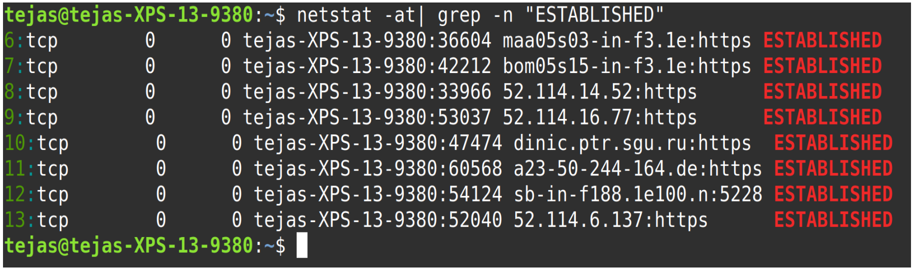


```
tejas@tejas-XPS-13-9380:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.1.1     0.0.0.0         UG    600    0        0 wlp2s0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 wlp2s0
192.168.1.0     0.0.0.0         255.255.255.0   U     600    0        0 wlp2s0
tejas@tejas-XPS-13-9380:~$ route -C
Kernel IP routing cache
Source          Destination     Gateway         Flags Metric Ref    Use Iface
tejas@tejas-XPS-13-9380:~$ route -F
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    600    0        0 wlp2s0
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 wlp2s0
192.168.1.0     0.0.0.0         255.255.255.0   U     600    0        0 wlp2s0
tejas@tejas-XPS-13-9380:~$ route -e
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 wlp2s0
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
tejas@tejas-XPS-13-9380:~$
```

Question 5

(a) **netstat (network statistics)** is used for displaying network connections (both incoming and outgoing), routing tables, interface statistics,multicast memberships, masquerade connections.

(b) netstat -at | grep -n ”ESTABLISHED”



```
tejas@tejas-XPS-13-9380:~$ netstat -at| grep -n "ESTABLISHED"
6:tcp        0      0 tejas-XPS-13-9380:36604 maa05s03-in-f3.1e:https ESTABLISHED
7:tcp        0      0 tejas-XPS-13-9380:42212 bom05s15-in-f3.1e:https ESTABLISHED
8:tcp        0      0 tejas-XPS-13-9380:33966 52.114.14.52:https      ESTABLISHED
9:tcp        0      0 tejas-XPS-13-9380:53037 52.114.16.77:https      ESTABLISHED
10:tcp        0      0 tejas-XPS-13-9380:47474 dinic.ptr.sgu.ru:https  ESTABLISHED
11:tcp        0      0 tejas-XPS-13-9380:60568 a23-50-244-164.de:https ESTABLISHED
12:tcp        0      0 tejas-XPS-13-9380:54124 sb-in-f188.1e100.n:5228 ESTABLISHED
13:tcp        0      0 tejas-XPS-13-9380:52040 52.114.6.137:https      ESTABLISHED
tejas@tejas-XPS-13-9380:~$
```

(c) **netstat -r** shows the kernel routing informations.
The output of the above command is organised in the form of a table with following columns:

(i) **Destination**: The destination network or destination host.

(ii) **Gateway**: The gateway address or * if none set.

(iii) **Genmask**: The netmask for the destination net, 255.255.255.255 for a host destination and 0.0.0.0 for the default route.

(iv) **Flags**:

(1) U : route is up
(2) H : target is host
(3) G : Use gateway
(4) R : reinstate route for dynamic routing
(5) D : dynamically installed by daemon or redirect
(6) M : modified from routing daemon or redirect
(7) A : installed by addrconf
(8) C : cache entry
(9) ! : reject route

(v) **MSS**: Default maximum segment size for TCP connections over this route.

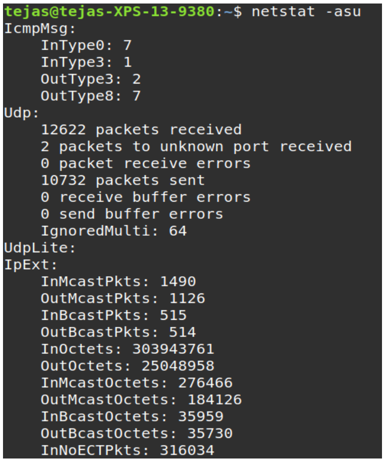
(vi) **Window**: Default window size for TCP connections over this route.

(vii) **irtt**: Initial RTT(Round trip time).The kernel uses this to guess about the best TCP protocol parameters without waiting on (possibly slow) answers.

(viii) **Iface**: Interface to which packets for this route will be sent.

(d) The option **-ai** displays the status of all network interfaces.
The number of interfaces can be found using the command:
echo $[$(netstat -ai| wc -l)-2]
The above command works as all the network interfaces are displayed in the form a table where the first line of the output is the table name and the second line is the names of the columns of the table, Hence the number of interfaces are **number of lines in the output - 2**.

(e) The option **-asu** displays the statistics of all the UDP connections.



```
tejas@tejas-XPS-13-9380:~$ netstat -asu
IcmpMsg:
    InType0: 7
    InType3: 1
    OutType3: 2
    OutType8: 7
Udp:
    12622 packets received
    2 packets to unknown port received
    0 packet receive errors
    10732 packets sent
    0 receive buffer errors
    0 send buffer errors
    IgnoredMulti: 64
UdpLite:
IpExt:
    InMcastPkts: 1490
    OutMcastPkts: 1126
    InBcastPkts: 515
    OutBcastPkts: 514
    InOctets: 303943761
    OutOctets: 25048958
    InMcastOctets: 276466
    OutMcastOctets: 184126
    InBcastOctets: 35959
    OutBcastOctets: 35730
    InNoECTPkts: 316034
```

(f) Loopback interface is a virtual interface used by machine to communicate with itself.It is the very first interface to be activated.Occasionally, you will also see the dummy hostname localhost being used instead of the IP-address.Its primary purpose is **network diagnosis and troubleshooting**.The loopback interface never goes down and provides a **virtual interface** which is not associated with any hardware. When a network interface in a machine is not established, the interface is unable to communicate with the servers in the same machine as well this problem is solved by loopback interface.

For example you can examine all the web documents that are present on your web server and could examine them file by file on the local machine. For IPv4, the loopback interface is assigned all the IPs in the 127.0.0.0/8 address block i.e 127.0.0.1 - 127.255.255.254.

Question 6

(a) Traceroute is a network diagnostic tool used to track in real-time the pathway taken by a packet on an IP network from source to destination, reporting the IP addresses of all the routers it pinged in between. Traceroute also records the time taken for each hop the packet makes during its route to the destination.

(b)

Sr. no	Host	Hops @2pm	Hops @8pm	Hops @2am
1	youtube.com	12	12	12
2	codeforces.com	timed out	timed out	timed out
3	amazon.com	timed out	timed out	timed out
4	sprinklr.com	13	13	13
5	facebook.com	8	8	8
6	cricbuzz.com	16	15	16

There were some obvious common hops i.e my device (192.168.1.1) and the IP address of the destination in every case. Apart form that IP address (122.170.255.255) and IP address (122.185.150.53) were the common hops in all the cases as they were the IP addresses of my wifi service provider.If the routes to the destinations pass through the same internet circles then we have common hops.

(c) The route to the hosts changes at different times of the day in the experiments because of network congestion. The packets are redirected by the nodes to take a route having less traffic. The load balancing is done to reduce the load of congested path.

(d) Sometimes, traceroute might not find a complete path to some host. Some servers/hosts along the path may have not been configured to respond to the ICMP Traffic or may have set up firewalls which block the ICMP Traffic. However, they still send the data to the next hop as there are results that follow. Moreover,when under heavy load many network providers disable ICMP traffic.

(e) It is possible to find the route to certain hosts which fail to respond with ping experiment.Both ping and traceroute use the ICMP Packets but there working is different.

Each IP packet sent on the Internet has a field known as Time-To-Live (TTL). But this field is not explicitly related to the time measured by the number of hops. It is instead, the maximum number of hops that a packet can travel across the Internet before it gets discarded.

Ping is straight ICMP from point A to point B and has a default TTL value between 1 to 255 which decrements by 1 at every router between the source to destination and expects a **ICMP Reply Packet** from the host. Most probably the server is blocking the reply.

On the other hand, In a traceroute, the source re-defines the TTL value every time it gets a response and sends the packet with TTL=TTL+1 until it reaches its destination.When a packet reaches its maximum TTL, the last hop in line will send back an **ICMP TTL Exceeded** packet back to the source.

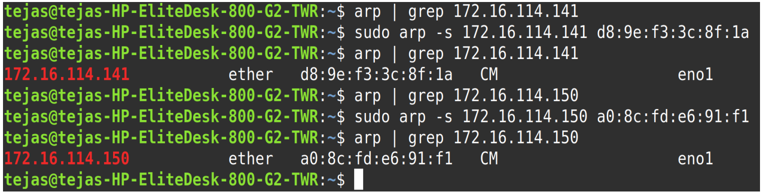
Traceroute looks for the ICMP Time exceeded packet and not the ICMP Reply Packet, and that is why it might be possible.

Question 7

(a) **arp** is used to display the complete ARP table on our machine. The output of the above command is organised in the form of a table with following columns:

- (i) **Hostname**: It is the hostname is the hostname cannot be resolved then you get a ?.
- (ii) **IP address**: It is the IP address of the host.
- (iii) **MAC address**: It is a six part hexadecimal number. In practice also known hardware address or ethernet address.
- (iv) **HWtype**: It is the Hardware type it could be ether i.e ethernet.
- (v) **Flags**:
  - (1) C : Complete Entry
  - (2) M : Permanent Entry
  - (3) P : Published Entry
- (vi) **Iface**: Network interface.

(b) **Delete a entry**: sudo arp -d `<IP address>`
Add a entry: sudo arp -s `<IP address> <MAC address>`



```
tejas@tejas-HP-EliteDesk-800-G2-TWR:~$ arp | grep 172.16.114.141
tejas@tejas-HP-EliteDesk-800-G2-TWR:~$ sudo arp -s 172.16.114.141 d8:9e:f3:3c:8f:1a
tejas@tejas-HP-EliteDesk-800-G2-TWR:~$ arp | grep 172.16.114.141
172.16.114.141           ether   d8:9e:f3:3c:8f:1a   CM                    eno1
tejas@tejas-HP-EliteDesk-800-G2-TWR:~$ arp | grep 172.16.114.150
tejas@tejas-HP-EliteDesk-800-G2-TWR:~$ sudo arp -s 172.16.114.150 a0:8c:fd:e6:91:f1
tejas@tejas-HP-EliteDesk-800-G2-TWR:~$ arp | grep 172.16.114.150
172.16.114.150           ether   a0:8c:fd:e6:91:f1   CM                    eno1
tejas@tejas-HP-EliteDesk-800-G2-TWR:~$ 
```

(c) ARP works only when the devices are in the same IP subnet. When a device with IP address X needs to send a packet to device with IP address Y , it checks with the routing table whether IP address Y belongs to a subnet it can directly reach through its network interfaces, if this is not the case then device with IP address X uses ARP to map IP address Y to a physical ethernet address and then sends Ethernet frame to that address.
When the two IP addresses are on different subnets, the device will follow a completely different logic i.e. it would check its routing table for a route to the destination network and then it will send its packet to the appropriate router in such a case the ARP will be used to find the hardware address of the router, because the destination IP address ,so the packet must be delivered to a router which can take care of it.

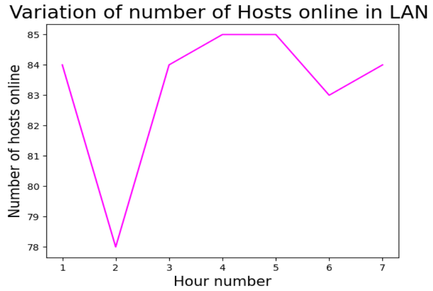
(d) We see a 100% packet loss when we ping the IP address (say A) whose ethernet address was replaced and 0% packet loss with the IP address (say B) whose ethernet address was not changed . This happens because IP address A tries to connect through a port which is already occupied and is unable to establish a connection.This results in IP address A being unreachable form other devices in the subnet hence the ping has 100% packet loss in case of IP address A.

Question 8

(a) nmap -sn `172.16.114.0/24`

(b) nmap -sA `<IP address>`

(c)



Hour number	1	2	3	4	5	6	7
Hosts online	84	78	84	85	85	83	84

We can observe that the number of hosts that are online at a given hour is always between 78 and 85.