CS342 ASSIGNMENT-1

NAME-ARTI SAHU

ROLL NO.-200123011

QUS.1-

a)The option required is **-c <ct>** where the **ct** represent **no. of echo** send. Example- **ping -c 10 www.myntra.com**

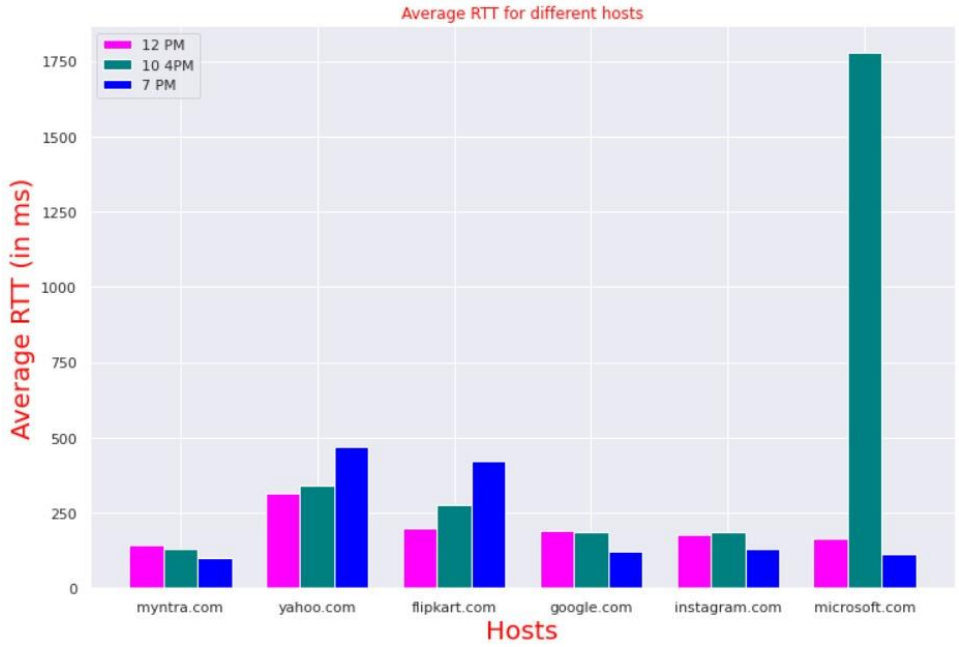
b)The option required to set time interval in sec. is **-i <tm>**,where **tm** represent the interval between 2 successive pin request. Ex.- **ping -i 0.8 www.myntra.com**

c)For sending the echo request continuously with waiting for reply:we have to decrease the time request to 0 but it is possible for sudo user.(The limit for sending packets for normal user is 0.2sec.) the command use is **sudo ping -f myntra.com** (f is use for flooding).

D)The command is to set the **echo_request** packet size is **ping -s <sz>** where the sz represents the size of packet to send.if the packet size set is **32bytes** the total packet size is **40bytes**,because **8 bytes** is the header data for that packet.

QUS.2-A)-The table as follows

HOST	LOCATION	DISTANCE(KM)	Avg. Rtt (at 12PM)in ms	Avg. Rtt (at 4 PM) in ms	Avg. Rtt (at 7 PM) in ms	Overall avg. Rtt in ms
mytnra.com	**Bangaluru**	2950	145	132	102	126
yahoo.com	**Sunnyvale,California**	7529	315	339	468	374
flipkart.com	**Bangaluru**	2950	197	278	420	298
google.com	**California,USA**	12610	190	186	122	166
instagram.com	**Menlo Park ,CA**	12101	176	186	128	163
microsoft.com	**Redmond, Washington**	11133	163	178	112	151



The **RTT** is strongly correlated with the geographical distance between the source and destination. This is due to the fact that larger distance increases propagation delay and the number of hops required. Also there is node processing delay since larger distance means more nodes to pass through. But there are also several other factors too like **network conditions (traffic), internet speed, server capabilities which may also affect the RTT, which is also evident from the data above(myntra and microsoft).**

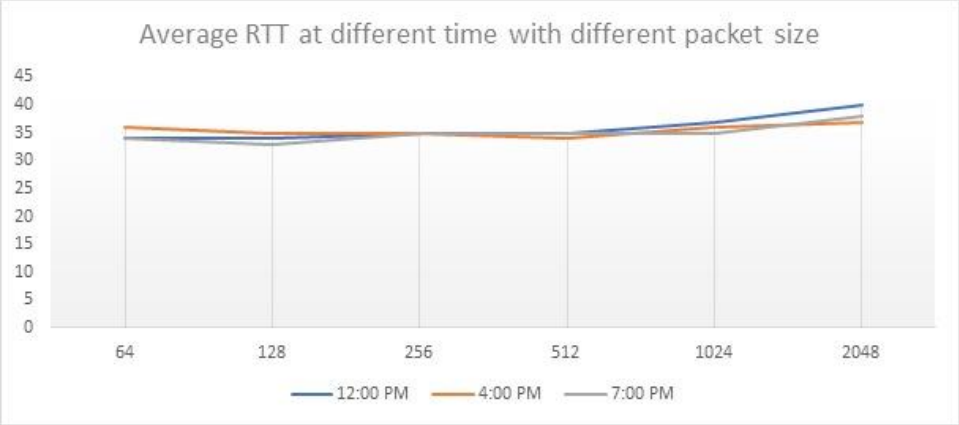
B)Packet loss at different time

HOST	Packet loss at 12PM	Packet loss at 4PM	Packet loss at 7PM
Myntra.com	8%	0%	0%
Yahoo.com	0%	0%	4%
Flipkart.com	0%	0%	0%
Google.com	0%	4%	0%
Instagram.com	4%	0%	0%
Microsoft.com	0%	0%	0%

In the above table at around 12PM there are packet losses. It says that network traffic was higher at 12PM. At rest of the time, the network traffic was smooth since all packets were correctly transmitted without incurring any loss except at 7PM (yahoo.com). So, due to network congestion, some packets got lost. There may be packets collision in the network due to which packets got dropped.

C) Avg. RTT variation in millisecond for different packet size (host google.com)

Time	64Bytes	128 Bytes	256 Bytes	512 Bytes	1024 Bytes	2048 Bytes
12PM	34	34	35	35	37	40
4PM	36	35	35	34	36	37
7PM	34	33	35	35	35	38



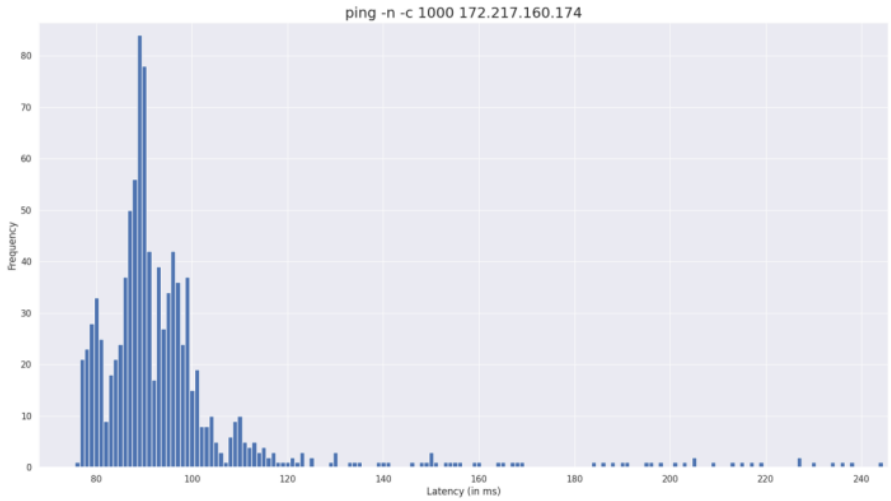
At different hours of the day, we can observe the different values of average RTT because at different hours, intensity of the traffic in network is different i.e., more is the traffic more will be the **RTT value.Size of packet** is **directly proportional to the RTT values.**

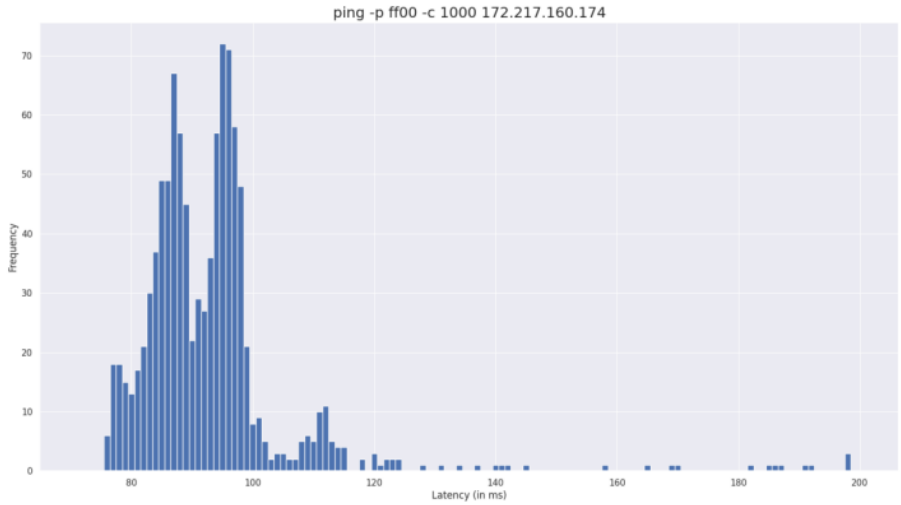
QUS.3-a)(i) 1.2%packet loss (ii) 3.7%packet loss

B)

	ping –n 172.217.160.174	**ping -p ff00 172.217.160.174**
Minimum latency	2.647	2.375
Maximum latency	306.146	292.125
Mean latency	19.534	15.778
Median latency	5.84	4.958

c)





d) When the ping is used with **-p flag**, all the quantities including the packet loss rate higher than that of **-n flag**. Using **-n flag** will cause no attempt to be made to look up symbolic names for host addresses, i.e, no **DNS** resolution takes place. Hence, mean latency is lower in **-n flag** case. Also **-p flag** is generally used for diagnosing data-dependent problems in a network. Here, it will fill out the packets with **ff00 (as specified) - 16 bytes.**

QUS.4-

(I) After running ***ifconfig*** command on my machine I got **2 active network interface eth0 lo**

(a) ***eth0*** is the first Ethernet interface. (Additional Ethernet interfaces would be named **eth1, eth2,** etc.) This type of interface is usually a **NIC** connected to the network by a category 5 cable. ***lo*** is the loopback interface. This is a special network interface that the system uses to communicate with itself. Every interface has some more properties associated with itself like

(b) Is it up or down

(c) Type of packets for which this interface is configured for like broadcast or loopback etc.

(d) The netmask addresses.

(e) Number of **errors, dropped, overruns, carrier, collisions, etc.**

(f) **MTU – The maximum transfer unit** for that interface.

(II)The following options can be used with ifconfig

: i) **-a :** It displays all the interfaces which are currently available, even if they are down.

ii) **-s :** It displays a short list of interfaces. **(like netstate –i)**

. iii) **-v :** It allows the output to be more verbose for some error condition.

iv) **[-] arp** : It enables/disables the use of the **ARP protocol** on this interface

. v) **mtu N** : This parameter sets the **Maximum Transfer Unit (MTU)** of an interface.

(III) ***route*** command can be used to work with the **IP/kernel routing** table. It is used to establish static routes to specific hosts or networks through an interface. some of points from the output of ***route*** command are –

- **a.** it shows the default i.e., the first router, the first hop through which the traffic passes before going to the next hop or final node. For my machine **default gateway is 172.30.16.1**
- b. it also shows the flags i.e., whether the given interface is up or down.
- c. It also shows the type of that interface like **eth0**, etc.
- **d.** It also shows the **destination IP address**

(IV) The following options can be used with route :

i) **- n** : It is used to display the numerical IP addresses.

ii) **- ee** : It will show all parameters from the routing table, generating a long line

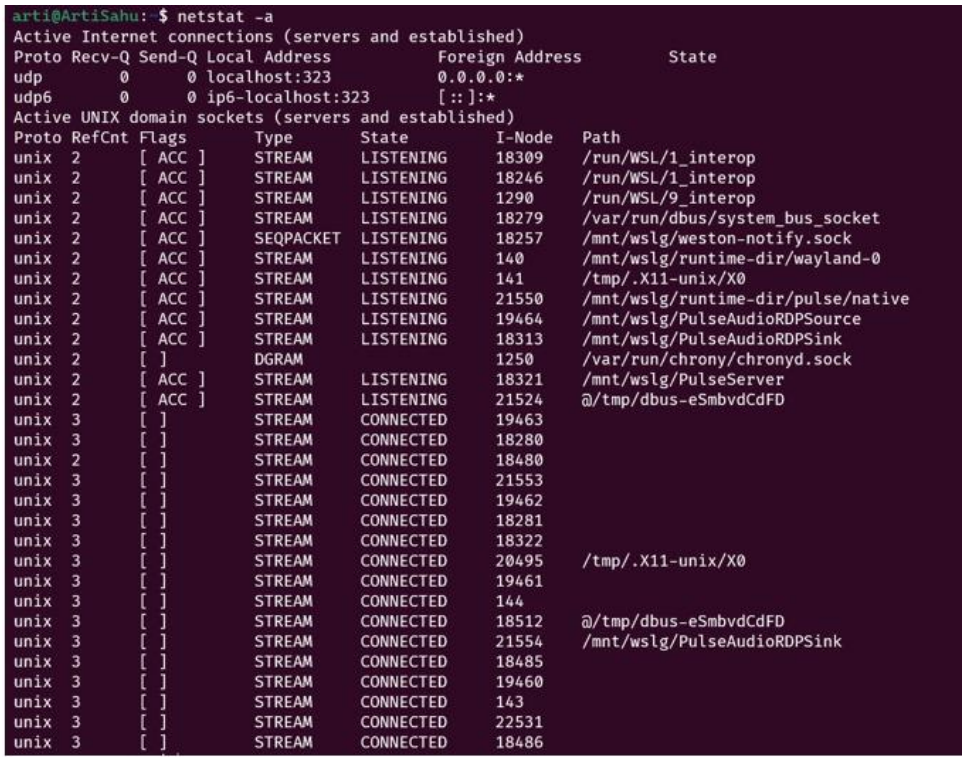
iii) **- e** : It will allow route command to use netstat-format for displaying the routing table.

iv) **- C** : It will list the kernel's routing cache information.

QUS.5-

(a) **netstat (network statistics)** is a command line tool for monitoring network connections both incoming and outgoing as well as viewing **routing tables, interface statistics** etc. It prints **network connections, routing tables, interface statistics, masquerade connections, and multicast memberships.**

(b) -at option is used to show all established TCP connections



```
arti@ArtiSahu:~$ netstat -a
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 localhost:323           0.0.0.0:*
udp6       0      0 ip6-localhost:323       [::]:*
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  2      [ ACC ]     STREAM     LISTENING     18309    /run/WSL/1_interop
unix  2      [ ACC ]     STREAM     LISTENING     18246    /run/WSL/1_interop
unix  2      [ ACC ]     STREAM     LISTENING     1290     /run/WSL/9_interop
unix  2      [ ACC ]     STREAM     LISTENING     18279    /var/run/dbus/system_bus_socket
unix  2      [ ACC ]     SEQPACKET  LISTENING     18257    /mnt/wslg/weston-notify.sock
unix  2      [ ACC ]     STREAM     LISTENING     140      /mnt/wslg/runtime-dir/wayland-0
unix  2      [ ACC ]     STREAM     LISTENING     141      /tmp/.X11-unix/X0
unix  2      [ ACC ]     STREAM     LISTENING     21550    /mnt/wslg/runtime-dir/pulse/native
unix  2      [ ACC ]     STREAM     LISTENING     19464    /mnt/wslg/PulseAudioRDPSource
unix  2      [ ACC ]     STREAM     LISTENING     18313    /mnt/wslg/PulseAudioRDPSink
unix  2      [ ]         DGRAM                    1250     /var/run/chrony/chronyd.sock
unix  2      [ ACC ]     STREAM     LISTENING     18321    /mnt/wslg/PulseServer
unix  2      [ ACC ]     STREAM     LISTENING     21524    @/tmp/dbus-eSmbvdCdFD
unix  3      [ ]         STREAM     CONNECTED     19463
unix  3      [ ]         STREAM     CONNECTED     18280
unix  2      [ ]         STREAM     CONNECTED     18480
unix  3      [ ]         STREAM     CONNECTED     21553
unix  3      [ ]         STREAM     CONNECTED     19462
unix  3      [ ]         STREAM     CONNECTED     18281
unix  3      [ ]         STREAM     CONNECTED     18322
unix  3      [ ]         STREAM     CONNECTED     20495    /tmp/.X11-unix/X0
unix  3      [ ]         STREAM     CONNECTED     19461
unix  3      [ ]         STREAM     CONNECTED     144
unix  3      [ ]         STREAM     CONNECTED     18512    @/tmp/dbus-eSmbvdCdFD
unix  3      [ ]         STREAM     CONNECTED     21554    /mnt/wslg/PulseAudioRDPSink
unix  3      [ ]         STREAM     CONNECTED     18485
unix  3      [ ]         STREAM     CONNECTED     19460
unix  3      [ ]         STREAM     CONNECTED     143
unix  3      [ ]         STREAM     CONNECTED     22531
unix  3      [ ]         STREAM     CONNECTED     18486
```

The description is as follows-

1. ***Proto:*** It Tells about which protocol is used by the given network socket.
2. ***Recv-Q:*** When connection is established then it shows the number of bytes not copied by the user program connected to this socket.
3. ***Send-Q:*** When connection is established this shows the count of bytes not acknowledged by the remote host.
4. ***Local Address****:* This shows the local end of the socket's port number.
5. ***Foreign Address:*** this shows the socket's remote end address and port number .

(c) The output of ***netstat -r*** is same as of ***route*** command. They both shows the routing table.

The explanation of the output is as follows

A. it shows the default i.e., the f**irst router**, the first hop through which the traffic passes before going to the next hop or final node. For my machine, the **default gateway is 192.168.183.152**
B. it also shows the flags i.e., whether the given interface is up or down.
C. It also shows the type of interface like **eth0**, etc.
D. It also shows the destination IP address.

(d) ***netstat -a*** can be used for show all network interfaces. Total **interface in my machine is 6.**

(e) The **statistics of all UDP** connections can be found out by executing **netstat –su.**

(f)

```
arti@ArtiSahu:~$ ifconfig lo
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 1500
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0xfe<compat,link,site,host>
        loop  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

loopback interface is a special virtual interface that is used by the computer to communicate with itself. When a network interface is disconnected, no communication is possible on that interface, even between the computer and itself. In such case, loopback interface comes into the picture. It is also used for **diagnostics purposes, troubleshooting**, and to connect to the servers running on the local machine. The **loopback interface** does not represent actual hardware, but is a logical, virtual interface.

QUS.6-

traceroute tracks the route packets taken from an IP network on their way to a given host. It utilizes the IP protocol's TTL field to find the route a packet takes to reach the host.

(a)The **hop counts** at different time of the day for different hosts are as follows :

Time	myntra.com	yahoo.com	flipkart.com	google.com	instagram.com	microsoft.com
12PM	9	9	19	18	14	11
4PM	9	9	19	18	14	11
7PM	9	9	20	18	14	11

The hops which were common to all the host were **192.168.183.249 (my IP address), 103.206.8.62 (ISP provider)** and **103.206.8.61. 14.143.172.17** was common to **microsoft.com** and **google.com** **203.192.196.30** was common to **yahoo.com** and **myntra.com. 10.248.2.61** was common to **flipkart.com** and **myntra.com. 14.143.59.13** was common to **flipkart.com** and **instagram.com** The hops were common because the packets travelled along the same routes for a part of their journey and so were handled by the same nodes.

(b) It is possible for the route to the hosts to change at different times of the day. It was also evident from the data collected. Due to the **network congestion** and **traffic**, the packets are redirected to the nodes with less traffic to reduce the congestion. Also **destination host** may utilize multiple servers to handle the incoming packets, thereby showing different IP addresses when the command is executed multiple times.

(c) **traceroute** in **Linux systems** use **UDP packets.** Sometimes hosts on the path are configured to block the **ICMP/UDP packets** or they may have a firewall set-up which blocks the packets. As a result, they do not respond. Nevertheless, they send data to the next hops since there are nodes in the results which follows * * *. A hop that outputs * * * means that the router at that hop does not respond to the type of packet we were using for the **traceroute**. Such nodes are configured to prevent **DoS** attacks which are

generated using **UDP/ICMP packets**. Also sometimes due to heavy huge networks traffic, the nodes are disabled for receiving these packets.

(d) Yes, it is possible because ping uses **ICMP echo requests,** while traceroute implementations provide a wide range of protocols including **ICMP echo request, TCP SYN**, and **UDP packets**. ping is straight **ICMP** from point A to point B, that traverses networks via routing rules. traceroute works by targeting the final hop, but limiting the TTL and waiting for a time exceeded message, and then increasing it by one for the next iteration. Therefore, the response it gets is not an **ICMP echo reply** to the **ICMP echo request** from the host along the way, but a time exceeded message from that host - so even though it is using **ICMP**, it is using it in a very different way. Traceroute looks for the **ICMP** Time exceeded packet and not the **ICMP** Reply Packet. Hence, it is possible to discover those hosts using wide variety of protocols available with different implementations of traceroute.

QUS.7-

(a) To display the full **ARP** we can use this command : **arp -a**

Internet Address shows the **IP addresses** of the network connections. Physical Address shows the type of **MAC Address** of the devices (source and destination. There are two types of entries- dynamically and static. A dynamic entry is an **IP to MAC Address** that your computer has learned of itself during recent communication. Whereas static entry on the hand is one that was manually entered.

(b) To add an entry in **ARP table**, we use: **arp -s <IP Address> <MAC Address>** To delete an entry in **ARP table,** we use: **arp -d <IP Address>** 1But since complete deletion of an entry is expensive, the **MAC address** of the entry is changed to instead to invalidate the entry.

(c)**ARP (Address Resolution Protocol)** is for use within a single network only. Computers use it to map **IP addresses to MAC addresses** within a network. **ARP table** helps in discovering link layer address associated with an internet layer address. So, there cannot be an entry from different subnet in **ARP** table of my PC. Yet there is a concept of **ARP proxy** in which a device on a given network answers the ARP queries for an IP address that is not on that network.

(d) After performing the given steps, the IP whose **Ethernet Address** was changed completely failed to respond to the pings, resulting in **100 % packet loss** while the other **IP r**esponded. When ping is sent from one device to another in the network, the **destination IP address** must be resolved to **MAC address** for transmission in data link layer. To achieve this, a broadcast packet is sent out in the network, known as **ARP request**. The **destination machine** with the **IP in the ARP request** then responds with an **ARP reply** that contains the **MAC address** for that **IP**. When 2 devices share the same **MAC address**, it creates confusion in the network. **MAC address** is supposed to be unique since it identifies the hardware in a network. Due to the tampering with the **MAC address (Ethernet address),** the packets were unable to reach that particular host, since its **MAC address** was not available in the **ARP table** and hence, it failed to respond.

QUS.8-

(a)The command to check which PCs in sub-net are up is: **nmap -sP <subnet_address>** In my case, it was : **nmap -sP <192.168.0.1/24>**

(b)The command to detect firewall setting is : **nmap -sA <IP Address>** It will detect whether the packets can pass through the firewall unfiltered **(ACK Scan**). We can also perform SYN scan with

nmap -sS <IP Address>.

(c) This data is as follows:

Time	**9 AM**	**12 PM**	**3 PM**	**6 PM**	**9 PM**	**11 PM**
No. of host online		**3**	**3**	**5**	**15**	**6**	**4**

QUS.9-

(a)**nslookup** followed by the domain name will display the **IP Address of the domain.**

For **example nslookup google.com** **nslookup <domain name>**

(b)This can be done by **Reverse DNS lookup**. Command is **nslookup** followed by **IP Address**. For **example- nslookup 8.8.8.8** **nslookup <IP Address>**

(c) **nslookup -type=soa <DNS Name >** Lookup for an soa record **SOA record (start of authority),** provides the authoritative information about the domain , the e-mail address of the domain, the domain serial number, etc.