CS 558: Computer Systems Lab

Assignment – 1:
Network Diagnostic Commands & Socket Programming

Shantanu Shrivastav	234101063
Shubham Mourya	234101064
Ayush Agarwal	234101060
Pradoom Varma	234101036

Q1. The Internet Ping command bounces a small packet(s) to test network communications,
and then shows how long this packet(s) took to make the round trip. The Internet Ping program works much like a sonar echo-location, sending a small packet of information containing an ICMP ECHO_REQUEST to a specified computer, which then sends an ECHO_REPLY packet in return. Explore more about the ping command and answer the
following questions (Unix or GNU/Linux version only):
a) What is the option required to specify the number of echo requests to send with ping command?
b) What is the option required to set time interval (in seconds), rather than the default one second interval, between two successive ping ECHO_REQUESTs?
c) What is the command to send ECHO_REQUEST packets to the destination one after another without waiting for a reply? What is the limit for sending such ECHO_REQUEST packets by normal users (not super user)?
d) What is the command to set the ECHO_REQUEST packet size (in bytes)? If the PacketSize is set to 32 bytes, what will be the total packet size?

a. Unix-like systems (including Linux and macOS), use the -c option followed by the number of packets to specify the count. Here's an example:

```
ping -c 4 google.com
```

b. In Unix-like systems (including Linux and macOS) use the -i option followed by the time interval in seconds. Here's an example with a 2-second interval:

```
ping -i 2 google.com
```

c. To send ECHO_REQUEST packets to the destination one after another without waiting for a reply, you can use the ping command with the -f option. The -f option is often used for flood ping, where packets are sent as quickly as possible.

   Here's an example:

   ```
   ping -f google.com
   ```

   The limit for sending such ECHO_REQUEST packets by normal users is often controlled by the network or system administrator and is configured through various means such as rate limiting. In many systems, regular users may be limited to avoid network congestion or potential misuse.

d. To set the ECHO_REQUEST packet size in bytes with the ping command, you can use the -s option followed by the desired packet size. Here's an example:

   ```
   ping -s 32 google.com
   ```

   This command sets the packet size to 32 bytes for the ECHO_REQUEST.

The total packet size, however, will likely be larger than the specified payload size (32 bytes). The ICMP header itself adds overhead to the packet. The ICMP header is typically 8 bytes, and the IP header adds an additional 20 bytes (assuming an IPv4 header). Therefore, the total packet size can be calculated as follows:

Total Packet Size = Payload Size + ICMP Header Size + IP Header Size

For example, with a payload size of 32 bytes:

Total Packet Size = 32 bytes + 8 bytes + 20 bytes = 60 bytes

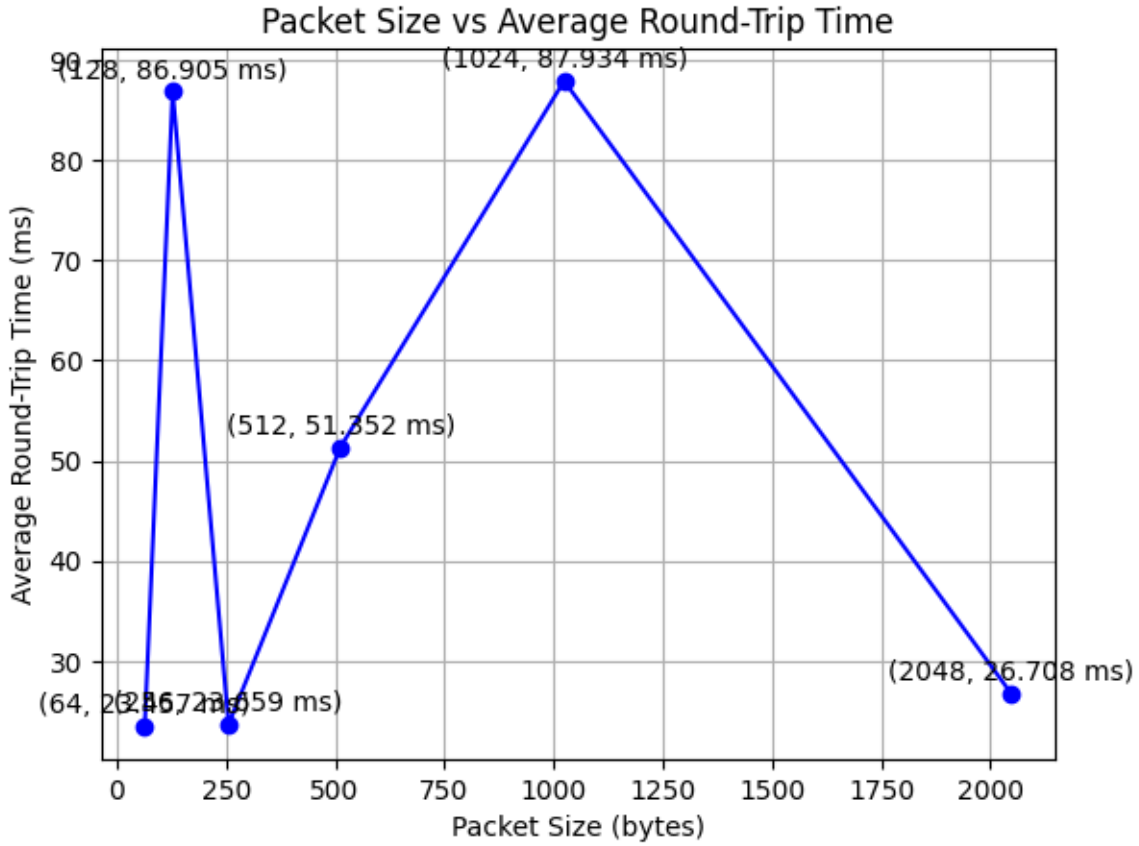
Q2. Select six hosts of your choice in the Internet (mention the list in your report) and experiment with pinging each host 25 times at three different hours of the day. Check if there exist cases, which show packet loss greater than 0% and provide reasoning. Find out average RTT for each host and explain whether measured RTTs are strongly or weakly correlated with the geographical distance of the hosts. Pick one of the above used hosts and repeat the experiment with different packet sizes ranging from 64 bytes to 2048 bytes. Plot the average RTT, and explain how change in packet size and time of the day impact RTT. You can use the following online tools for this experiment:

i) http://www.spfld.com/ping.html
ii) https://www.subnetonline.com/pages/network-tools/online-ping-ipv4.php

Server name	Ping time	AVG RTT (ms)	% packet loss
google.com	10:00 am	12.03	0.0%
	12:00 pm	12.34	0.0%
	5:00 pm	11.68	0.0%
youtube.com	10:00 am	10.9	0.0%
	12:00 pm	11.86	0.0%
	5:00 pm	12.48	0.0%
facebook.com	10:00 am	11.338	0.0%
	12:00 pm	11.787	0.0%
	5:00 pm	11.937	0.0%
yahoo.com	10:00 am	22.245	0.0%
	12:00 pm	30.051	0.0%
	5:00 pm	24.729	0.0%
cricbuzz.com	10:00 am	257.607	0.0%
	12:00 pm	247.507	0.0%
	5:00 pm	248.609	0.0%
unacademy.com	10:00 am	12.418	0.0%
	12:00 pm	12.916	0.0%
	5:00 pm	15.387	0.0%

RTT is strongly correlated with the geographical distance. For host cricbuzz.com RTT jump was observed when location was changed.

Selecting Yahoo.com as the host and sending packet for different sizes we get



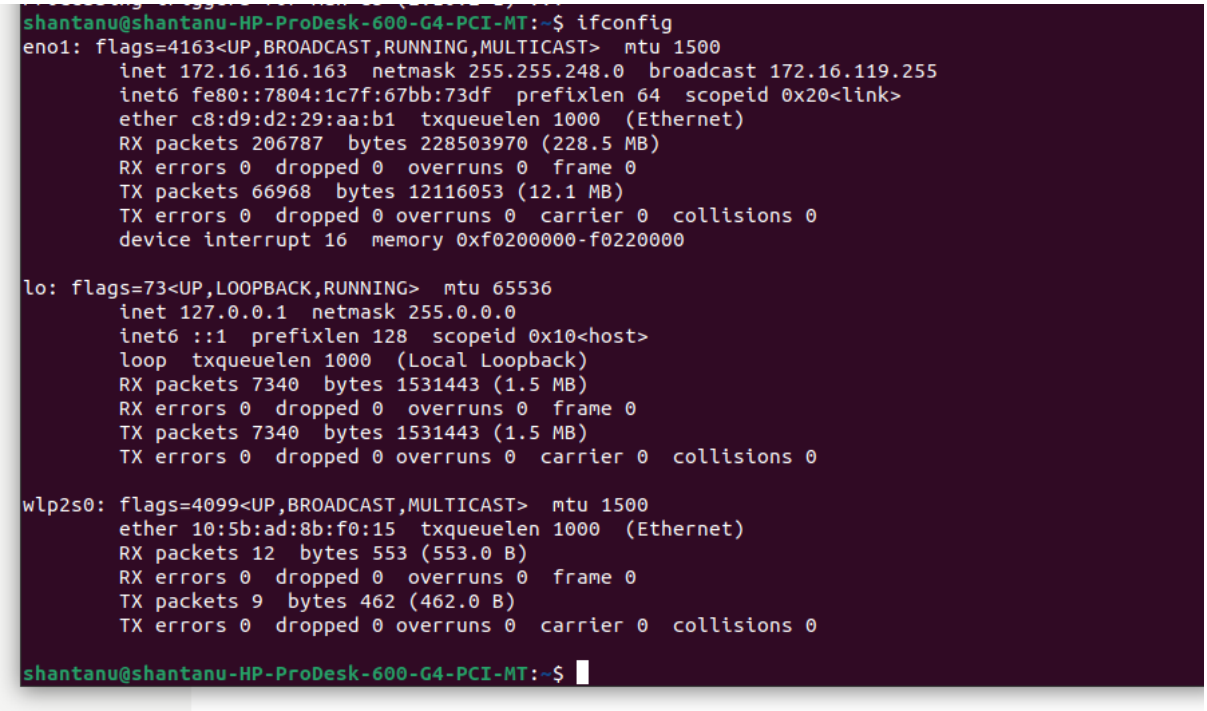
Q3. With regard to ifconfig and route commands, answer the following questions:
a) Run ifconfig command and describe its output (identify and explain as much of what is printed on the screen as you can).
b) What options can be provided with the ifconfig command? Mention and explain at least four options.
c) Explain the output of route command.
d) Mention and explain at least four options of the route command. Execute the route command with these four options and show the output.

a. The ifconfig command displays information about network interfaces on your system. Here's a general overview of what you might see in its output:

Interface Name (eth0, wlan0, etc.): This is the name of the network interface.

1. Link Encapsulation: It indicates the type of link-layer encapsulation, such as Ethernet.
2. Hardware/MAC Address: This is the unique identifier for the network interface.
3. IP Address: Displays the IP address assigned to the interface.
4. Broadcast Address: Shows the broadcast address for the network.
5. Netmask: Specifies the network mask associated with the interface.

6. MTU (Maximum Transmission Unit): It represents the maximum size of a data packet that can be transmitted over the network.
7. RX (Receive) and TX (Transmit) Statistics: Displays the number of packets received and transmitted, as well as the number of errors.
8. Flags: Various flags may be present, indicating the status of the interface (UP, DOWN, PROMISC, etc.).



b.

1. **Displaying Interface Statistics:**

   Option: -a or no specific option (to display all interfaces)

   Using the -a option or simply running ifconfig without any specific interface name will display information about all active network interfaces, including their IP addresses, MAC addresses, and various statistics such as packets transmitted and received.

2. **Assign a Broadcast to Network Interface**

   Using the “broadcast” argument with an interface name will set the broadcast address for the given interface. For example, the “ifconfig eth0 broadcast 172.16.25.63” command sets the broadcast address to an interface eth0.

   ```
   ifconfig eth0 broadcast 172.16.25.63
   ```

3. **Netmask**

   Specifies the network mask associated with the interface. The netmask defines the range of IP addresses that are considered part of the local network. For example:

```
ifconfig eth0 netmask 255.255.255.0
```

4. **Maximum Transmission Unit (mtu)**

   Specifies the Maximum Transmission Unit, which is the maximum size of a data packet that can be transmitted over the network. For example:

```
ifconfig eth0 mtu 1500
```

c.

The route command allows you to make manual entries into the network routing tables.
The route command distinguishes between routes to hosts and routes to networks by interpreting the network address of the Destination variable, which can be specified either by symbolic name or numeric address. The route command resolves all symbolic names into addresses, using either the /etc/hosts file or the network name server.

```
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    100    0        0 eno1
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 eno1
172.16.112.0    0.0.0.0         255.255.248.0   U     100    0        0 eno1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$
```

1. **Destination**: This column represents the destination network or host. It indicates where the packet is trying to go.
2. **Gateway**: The IP address of the next-hop router or gateway. If the destination is on the same network, the gateway is often set to "*".
3. **Genmask**: The netmask associated with the destination network. It defines which portion of the IP address is the network part.
4. **Flags**: Various flags indicate the status of the route. Common flags include:
   - U (Up): The route is up.
   - G (Gateway): A route through a gateway.
   - H (Host): A route to a specific host.
   - D (Dynamic): A dynamically learned route.
   - UG (Up and Gateway): A route that is both up and uses a gateway.
5. **Metric**: The metric represents the cost of the route. Lower metrics are preferred. It's used when there are multiple routes to the same destination.
6. **Ref**: The reference count of the route, indicating how many active references exist to this route.
7. **Use**: The number of packets that have been routed through this entry.
8. **Iface**: The network interface through which the packets are routed.

d.

1. **Display the Routing Table:**

   Option: -n

   Displays the numeric IP addresses in the output instead of resolving them to hostnames. For example:

route -n

```
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         172.16.112.1    0.0.0.0         UG    100    0        0 eno1
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 eno1
172.16.112.0    0.0.0.0         255.255.248.0   U     100    0        0 eno1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$
```

2. **Verbose mode:**

   Option: -v

   Verbose mode displays detailed routing table information, including that for inactive routes.

route -v

```
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ route -v
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    20100  0        0 eno1
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 eno1
172.16.112.0    0.0.0.0         255.255.248.0   U     100    0        0 eno1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$
```

3. **FIB**

   Option: -F

   Displays Forward Information Base(FIB).

route -F

```
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ route -F
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    100    0        0 eno1
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 eno1
172.16.112.0    0.0.0.0         255.255.248.0   U     100    0        0 eno1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$
```

4. **More Information**

Option: -e

Display more information compared to route

route -e

```
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ route -e
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 eno1
link-local      0.0.0.0         255.255.0.0     U         0 0          0 eno1
172.16.112.0    0.0.0.0         255.255.248.0   U         0 0          0 eno1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$
```

Q4. Answer the following questions related to netstat command.
a) What is the command netstat used for?
b) What parameters for netstat should you use to show all the established TCP connections? Include a screenshot of this list for your computer and explain all the fields of the table in the output.
c) What does "netstat –r" show? Explain all the fields of the output.
d) What option of netstat can be used to display the status of all network interfaces? By using netstat, figure out the number of interfaces on your computer.
e) What option of netstat can be used to show the statistics of all UDP connections? Run the command for this purpose on your computer and show the output.
f) Show and explain the function of loopback interface.

a.
The netstat command generates displays that show network status and protocol statistics. You can display the status of TCP and UDP endpoints in table format, routing table information, and interface information.

netstat displays various types of network data depending on the command line option selected. These displays are the most useful for system administration. The syntax for this form is:

```
netstat [-m] [-n] [-s] [-i | -r] [-f address_family]
```

b.

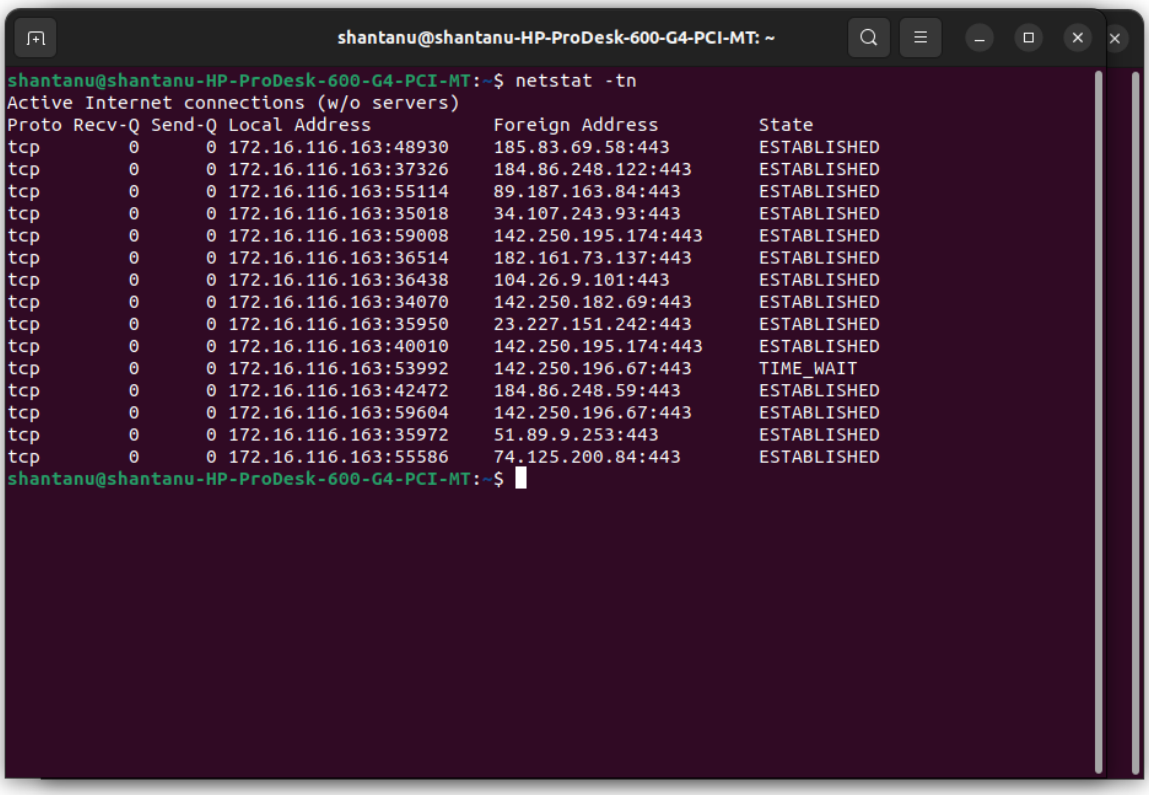
To show all the established TCP connections using the netstat command, you can use the -t option to filter the results for TCP connections, and the -n option to display numerical addresses instead of resolving them to hostnames.

```
netstat -tn
```

Options used:

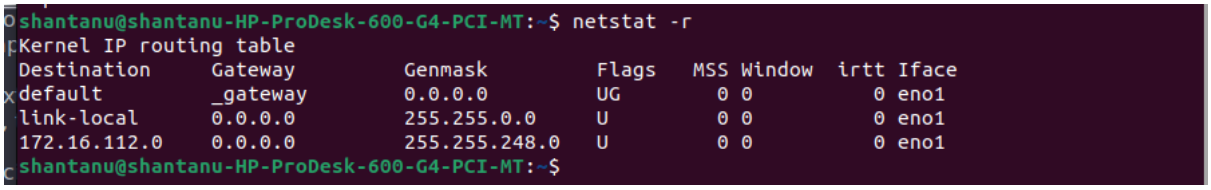
-t: Filters the output to display only TCP connections.

-n: Displays numerical addresses (IP addresses and port numbers) instead of resolving them to hostnames



c.

The netstat -r command is used to display the kernel routing table on Unix-like operating systems. The output shows information related to routing, indicating how network packets should be forwarded.



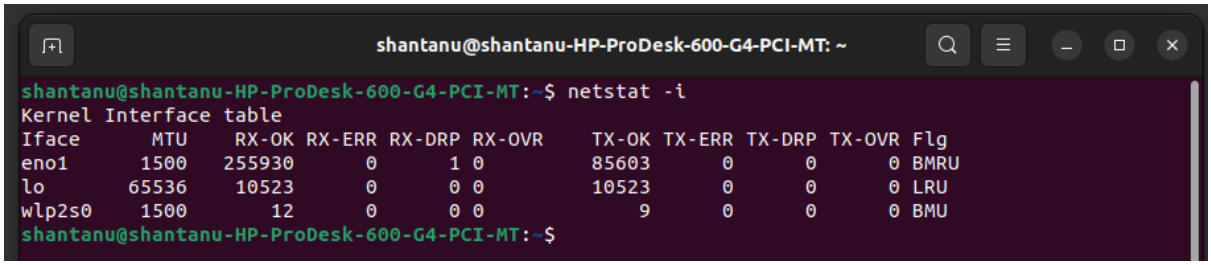
1. **Destination**: The destination network or host. It indicates where the packet is trying to go. For example, default represents the default route.
2. **Gateway**: The IP address of the next-hop router or gateway. If the destination is on the same network, the gateway is often set to 0.0.0.0.
3. **Genmask**: The network mask associated with the destination. It defines the range of IP addresses that are considered part of the destination network.
4. **Flags**: Various flags indicate the status of the route:
   U (Up): The route is up.
   G (Gateway): A route through a gateway.
   H (Host): A route to a specific host.

5. **MSS (Maximum Segment Size):** The maximum amount of data that can be sent in a single TCP segment. It is often used in TCP connection establishment.
6. **Window**: The size of the TCP receive window. It represents the amount of data that can be sent by the sender before an acknowledgment must be received.
7. **irtt (Initial Round Trip Time):** The initial estimated round-trip time. It is used by some routing algorithms.
8. **Iface (Interface):** The network interface through which the packets are routed.

d.

To display the status of all network interfaces using netstat, you can use the -i option. This option is used to show information about the network interfaces, including their status, packets transmitted and received, errors, and other statistics.

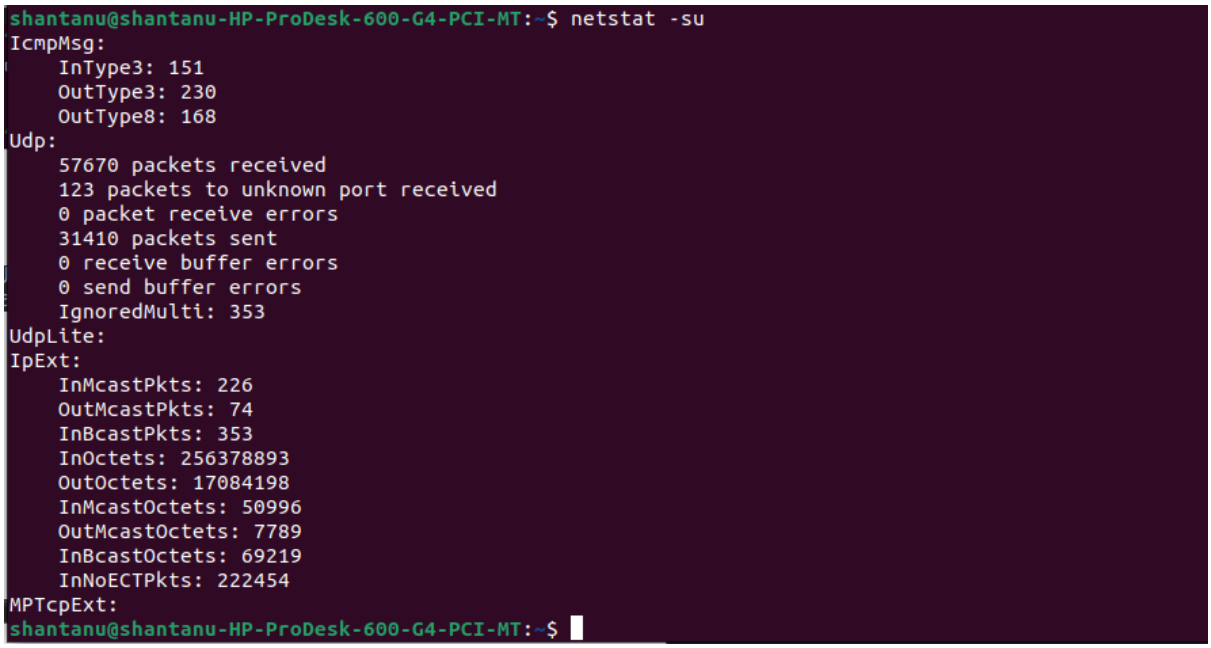
netstat -i



3 interfaces are connected.

e.
To show the statistics of all UDP connections using netstat, you can use the -su option. This option displays a summary of statistics for UDP (User Datagram Protocol) connections.

netstat -su



f.
The loopback interface is a special network interface in computer networking that allows a device to communicate with itself. It is often represented by the IP address 127.0.0.1 in IPv4, and "::1" in IPv6. The loopback interface is commonly referred to as "localhost."

- Communication with Itself
- Testing Network Services
- Address Representation
- Isolation from External Networks

Q5. What is a traceroute tool used for? Perform a traceroute experiment (with same hosts used in Q2) at three different hours of the day, and then answer the questions below. Use any one of the following online tools for this experiment:

- **http://ping.eu;**
- **http://www.cogentco.com/en/network/looking-glass;**
- **http://network-tools.com;**

a) List out the hop counts for each host in each time slot. Determine the common hops between two routes if they exist.
b) Check and explain the reason if route to same host changes at different times of the day.
c) Inspect the cases when traceroute does not find complete paths to some hosts and provide reasoning.
d) Is it possible to find the route to certain hosts which fail to respond with pingBroadbandSearch.net experiment? Give reasoning.

a.
A traceroute provides a map of how data on the internet travels from its source to its destination. When you connect with a website, the data you get must travel across multiple devices and networks along the way, particularly routers.

	Number of hops for each time slots		
	13:00	**17:00**	**18:30**
Google	15	15	15
Facebook	15	15	15
Youtube	15	15	15
Yahoo	15	14	15
Unacademy	15	15	15
Cricbuzz	10	10	11

Hosts	Common Hops
Google	100.100.36.104, 108.170.246.49
Facebook	204.15.23.143, 100.65.57.144, 157.240.81.236
Youtube	240.0.236.6, 100.66.13.124
Yahoo	99.83.114.249, 66.196.67.103
Unacademy	240.64.185.163
Cricbuzz	242.2.212.197

b.
For Yahoo there were 15 hops at 13:00 but 14 hops were observed at 17:00

Reason for change in hops can be

1. **Network Congestion**: Network congestion occurs when the amount of data being transmitted over a network surpasses its capacity to handle that data efficiently. It leads to a slowdown in data transfer speeds, increased latency, and potential packet loss. Congestion can happen at various points within a network, including routers, switches, and links between them.
2. **Maintenance**: Network maintenance activities, such as updates, patches, or repairs, may occur during specific time windows. During these periods, certain routes may be temporarily unavailable, causing routers to select alternative paths until the maintenance is completed.
3. **Load Balancing**: Implement load balancing techniques to distribute traffic across multiple paths to prevent congestion on a single route. Load balancing algorithms may adjust the distribution of traffic based on the current network load, causing the route to change dynamically.
4. **Dynamic Routing Protocols**: Changes in network conditions, such as link failures or fluctuations in the quality of links, can trigger the routers to recalculate the best path to reach a destination. These recalculations may lead to different routes being chosen at different times of the day.

c.

Reasons why traceroute did not complete the path to some hosts:

1. **Network Topology Changes:** Dynamic routing protocols, changes in network topology, or temporary network outages can also impact the completeness of traceroute results. Routes may change dynamically, leading to variations in traceroute output over time.

2. **Packet Filtering:** Some routers or network devices may be configured to filter out specific types of traffic, including ICMP packets used by traceroute. This can result in gaps in the traceroute output.

3. **Destination Host Configuration:** The behaviour of the destination host can influence traceroute results. If the host is configured to deprioritize or drop certain types of traffic, including ICMP, it may not respond to traceroute requests, causing gaps in the path information.

d.
It is possible to attempt to find the route to certain hosts even if they fail to respond to ping requests. Some hosts or routers may be configured to block ICMP traffic, including ping requests. In such cases, you can use alternative methods to determine the network path to the destination.

Even traceroute command can be used. Unlike ping, traceroute uses ICMP Time-to-Live (TTL) expiration messages to identify the routers along the path. This can help you identify the network hops to the destination, even if the destination host does not respond to ICMP Echo Requests.

MTR is a combination of ping and traceroute. It continuously sends packets to the destination and provides a real-time display of both latency and the route taken. MTR can be helpful in identifying where along the path the packets might be dropping.

Q6. Answer the following questions with regard to network addresses.
a) How do you show the full ARP table for your machine? Explain each column of the ARP table.
b) Check and explain what happens if you try and use the arp command to add or delete
an entry to the ARP table. Find out how to add, delete or change entries in the ARP table. Use this mechanism to add at least four new hosts to the ARP table and include a printout.
c) What are the parameters that determine how long the entries in the cache of the ARP module of the kernel remain valid and when they get deleted from the cache? Describe a trial-and-error method to discover the timeout value for the ARP cache entries.
d) What will happen if two IP addresses map to the same Ethernet address? Be specific on how all hosts on the subnet operate.

a.

Using arp -n command we can show full ARP table

```
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
172.16.115.224           ether   48:0f:cf:52:ce:8a   C                     eno1
172.16.115.207           ether   00:d8:61:f4:6c:a1   C                     eno1
172.16.115.214           ether   00:d8:61:f4:6b:d3   C                     eno1
172.16.115.209           ether   00:d8:61:f4:6b:72   C                     eno1
172.16.112.100           ether   00:17:c8:24:b3:64   C                     eno1
172.16.117.22            ether   44:94:fc:6e:d3:80   C                     eno1
172.16.117.181           ether   c0:18:03:55:9d:20   C                     eno1
172.16.116.9             ether   c8:d9:d2:29:b6:a6   C                     eno1
172.16.113.168           ether   3c:52:82:24:c1:f6   C                     eno1
172.16.112.1             ether   f8:0b:cb:cb:49:e4   C                     eno1
172.16.117.57            ether   00:11:32:c9:b4:bf   C                     eno1
172.16.117.162           ether   38:ca:84:32:6c:8d   C                     eno1
172.16.115.200           ether   00:d8:61:f4:4a:e1   C                     eno1
172.16.116.226           ether   6c:02:e0:68:5d:e5   C                     eno1
172.16.117.20            ether   40:a8:f0:b9:b5:7c   C                     eno1
172.16.112.101           ether   34:9f:7b:5a:01:3f   C                     eno1
172.16.115.220           ether   00:d8:61:f4:6b:c2   C                     eno1
172.16.117.85            ether   8c:dc:d4:5e:5f:ba   C                     eno1
172.16.117.103           ether   b0:b9:8a:45:bd:94   C                     eno1
172.16.112.136           ether   4c:cc:6a:36:a2:f5   C                     eno1
172.16.117.46            ether   80:e8:2c:d8:96:a9   C                     eno1
172.16.117.36            ether   6c:c2:17:28:f7:32   C                     eno1
172.16.115.194           ether   00:d8:61:f5:27:99   C                     eno1
172.16.113.177           ether   5c:60:ba:92:06:87   C                     eno1
172.16.117.91            ether   c8:cb:b8:62:87:25   C                     eno1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$
```

Column types

1. **Address**: This column displays the IP address of the device in the ARP table.
2. **HWtype**: Indicates the hardware type, typically Ethernet for most systems.
3. **HWaddress**: This column shows the MAC (Media Access Control) address, which is the physical address of the device.
4. **Flags**: Represents the state of the ARP entry. Common flags include C (complete) for a valid entry and I (incomplete) for an entry that is still being resolved.
5. **Mask**: Displays the network mask associated with the IP address.
6. **Device**: Specifies the network interface for which the ARP entry is valid.

b.

Adding an Entry:

If you try to add an entry using the arp command, you may encounter an error or a message indicating that the operation is not supported. This is because ARP tables are typically dynamically populated based on network activity, and the ARP protocol manages this process. Manually adding entries might not be a supported operation through the standard arp command.

However, some systems may allow you to add static ARP entries using specific options or additional commands. Manually adding static ARP entries is not a common practice and should be done with caution.

To add a static ARP

```
sudo arp -s <ip_address> <mac_address>
```

Deleting an Entry:

If you attempt to delete an entry using the arp command, you might encounter an error if the entry is essential for network connectivity or if it is a dynamic entry that will be re-added automatically as devices communicate.

Deleting static ARP entries using the appropriate command is generally supported, and the specified entry will be removed from the ARP table. Deleting dynamic entries might not have a significant impact as they will be recreated when needed.

To delete a static ARP entry:

```
sudo arp -d <ip_address>
```

```
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ sudo arp -s 172.16.115.207 00:d8:61:f4:6c:a1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ sudo arp -s 172.16.115.224 00:11:22:33:44:55
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ sudo arp -s 172.16.117.162 00:17:c8:24:b3:64
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ sudo arp -s 172.16.117.36 6c:c2:17:28:f7:32
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
172.16.115.224           ether   00:11:22:33:44:55   CM                    eno1
172.16.115.207           ether   00:d8:61:f4:6c:a1   CM                    eno1
172.16.115.214           ether   00:d8:61:f4:6b:d3   C                     eno1
172.16.115.209           ether   00:d8:61:f4:6b:72   C                     eno1
172.16.112.100           ether   00:17:c8:24:b3:64   C                     eno1
172.16.117.22            ether   44:94:fc:6e:d3:80   C                     eno1
172.16.117.181           ether   c0:18:03:55:9d:20   C                     eno1
172.16.116.9             ether   c8:d9:d2:29:b6:a6   C                     eno1
172.16.113.168           ether   3c:52:82:24:c1:f6   C                     eno1
172.16.112.1             ether   f8:0b:cb:cb:49:e4   C                     eno1
172.16.117.57            ether   00:11:32:c9:b4:bf   C                     eno1
172.16.117.162           ether   00:17:c8:24:b3:64   CM                    eno1
172.16.115.200           ether   00:d8:61:f4:4a:e1   C                     eno1
172.16.116.226           ether   6c:02:e0:68:5d:e5   C                     eno1
172.16.117.20            ether   40:a8:f0:b9:b5:7c   C                     eno1
172.16.112.101           ether   34:9f:7b:5a:01:3f   C                     eno1
172.16.115.220           ether   00:d8:61:f4:6b:c2   C                     eno1
172.16.117.85            ether   8c:dc:d4:5e:5f:ba   C                     eno1
172.16.117.103           ether   b0:b9:8a:45:bd:94   C                     eno1
172.16.112.136           ether   4c:cc:6a:36:a2:f5   C                     eno1
172.16.117.46            ether   80:e8:2c:d8:96:a9   C                     eno1
172.16.117.36            ether   6c:c2:17:28:f7:32   CM                    eno1
172.16.115.194           ether   00:d8:61:f5:27:99   C                     eno1
172.16.113.177           ether   5c:60:ba:92:06:87   C                     eno1
172.16.117.91            ether   c8:cb:b8:62:87:25   C                     eno1
shantanu@shantanu-HP-ProDesk-600-G4-PCI-MT:~$
```

c.

The duration for which ARP (Address Resolution Protocol) entries remain valid in the cache is determined by two primary parameters: the ARP timeout and the ARP aging timer.

1. **ARP Timeout:**

   ARP timeout is the time duration during which an ARP entry is considered valid.

   Default Value: Common default values are around 60 seconds or less.

2. **ARP Aging Timer:**

The ARP aging timer is a mechanism used to periodically check and remove stale ARP entries from the cache.

Default Value: The ARP aging timer default value is often set to a multiple of the ARP timeout. For example, if the ARP timeout is 60 seconds, the ARP aging timer might be set to run every 300 seconds (5 minutes).

Entries Get Deleted:

Dynamic Entries: Dynamic ARP entries, which are learned through network communication, have a timeout associated with them. When this timeout period elapses without any communication with the corresponding IP address, the entry becomes stale.

Static Entries: Static ARP entries, manually added by the user, typically do not expire based on a timeout. They remain in the ARP cache until manually deleted by the user or until the system is rebooted.

When the ARP aging timer runs, it checks the ARP cache for entries that have exceeded their timeout period. Stale entries are then removed, freeing up space in the ARP cache.

Sep-by-step trial-and-error method:

1. **Identify Current ARP Timeout:**
    - Check the documentation for your operating system to find the default ARP timeout value. This will give you a starting point.
2. **Observe Network Behaviour:**
    - Monitor the network and note the frequency of device movements or changes. Observe how quickly devices obtain new IP addresses or change their MAC addresses.
3. **Adjust Timeout Value:**
    - Based on your observations, make an educated guess about an appropriate ARP timeout value. Consider factors such as network stability, device mobility, and the frequency of IP address changes.
4. **Implement the New Timeout:**
    - Change the ARP timeout value in the system settings or using specific commands. The process varies depending on the operating system.
5. **Monitor ARP Cache Entries:**
    - Keep a close eye on the ARP cache entries over time. Use commands like `arp -n` on Linux to view the ARP table.
6. **Evaluate Network Performance:**
    - Monitor network performance and check for any disruptions or delays caused by ARP cache timeouts. If devices experience issues resolving IP addresses, the timeout value might be too short.
7. **Adjust as Needed:**

- If issues arise, consider increasing the ARP timeout value to reduce the frequency of ARP cache entries expiring. If entries are not getting cleared promptly, you may need to decrease the timeout value.

8. **Repeat the Process:**

    - Continue adjusting the timeout value based on network behaviour and performance. Iterate through steps 4 to 7 until you find a balance that meets the requirements of your specific network environment.

d.

If two IP addresses map to the same Ethernet address (MAC address) within the same subnet, it creates what is known as an "IP address conflict" or "IP address duplication." This situation can lead to several issues and disruptions in network communication. Here's how all hosts on the subnet may be affected:

1. ARP Cache Confusion:
2. Unpredictable Network Behaviour:
3. Address Resolution Ambiguity:
4. Data Corruption and Security Risks:
5. Impact on Routing and Switching:
6. Duplicated IP Address Warnings:
7. Potential Disruptions in DHCP Environments:

Q7. Local network analysis: Query your LAN using the nmap command to discover which hosts are online. Use a command such as: *nmap –n –sP <Subnet Range> (e.g., 172.16.112.0/26)*
You can choose a different LAN subnet address as well (make sure you report the same in your report explicitly).
Now run the command repeatedly at different times of the day, and find the number of hosts online. Do it for at least 6 times with sufficient time gap. Plot a graph against time to see if there are any hourly trends for when computers are switched ON or OFF in your LAN.

LAN Subnet used for experiment is 172.16.112.0/26