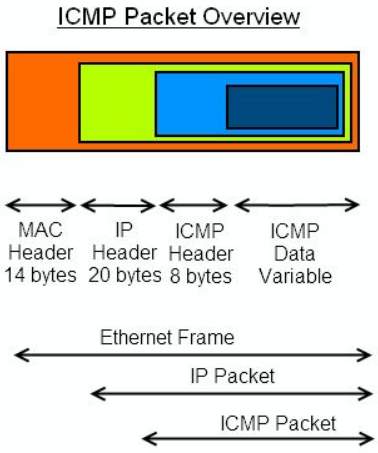
Lavish Gulati
170101082

CS349: Assignment 1 (Learning Network Diagnostic Tools)



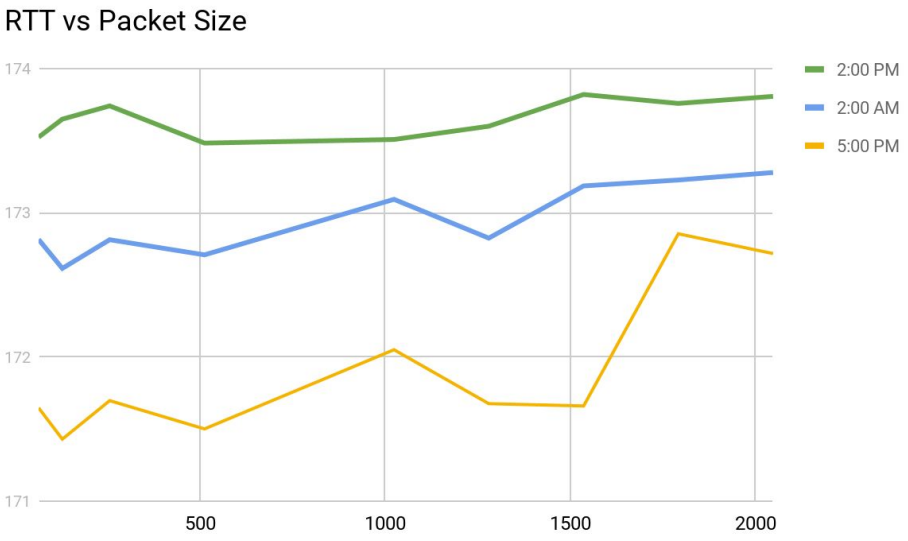
1. Ping command

A. With -c option, the ping command stops after sending the specified number of ECHO_REQUEST packets to the server/host. Example: ping -c 5 intranet.iitg.ernet.in
B. With -i option, we can specify the interval seconds between sending each packet. Example: ping -i 2 intranet.iitg.ernet.in
C. With -l option, ping sends that many packets not waiting for a reply. The limit for sending such packets by normal users is 3. We can also specify the -f option with zero time interval (flood ping), which sends ICMP packets as fast as possible without waiting for replies.
D. With -s option, we can specify the number of data bytes to be sent in each packet. If the value is set to 32 bytes, the total packet size will be 32+28 = 60 bytes (8 bytes ICMP Header + 20 bytes IP Header).

2. Round-Trip Time (RTT)

- The hosts used are flipkart.com, reddit.com, hulu.com, alibaba.com, codeforces.com and rbi.org.in, with observations taken at 2:00 pm, 5:00 pm and 2:00 am. Ping utility was used to take the observations, whose server is located in New Jersey, USA.

Host address	IP Address	Geographical Location	Avg. RTT 1 (ms) @ 2:00 pm	Avg. RTT 2 (ms) @ 2:00 am	Avg. RTT 3 (ms) @ 5:00 pm	Avg. RTT (ms)
flipkart.com	163.53.78.87	Bengaluru, India	245.182 (0%)	233.792 (0%)	250.209 (0%)	243.061
reddit.com	151.101.193.140	California, USA	3.763 (0%)	4.116 (0%)	3.254 (0%)	3.711
hulu.com	88.221.131.142	Massachusetts, USA	42.948 (0%)	43.709 (0%)	42.153 (0%)	42.936
alibaba.com	47.75.85.41	Zhejiang, China	210.603 (0%)	211.447 (0%)	210.637 (0%)	210.895
codeforces.com	81.27.240.126	Sankt-Peterburg, Russia	120.696 (0%)	121.496 (0%)	120.824 (0%)	121.005
rbi.org.in	14.140.169.71	Mumbai, India	217.417 (0%)	217.775 (0%)	216.720 (0%)	217.304



- **Packet Loss:** Packet loss of either 0% or 100% was observed. 100% packet loss occurred in sites such as amazon.com. These sites that have had DoS attacks in the past, have filtered ICMP to prevent ping requests. Because of this when pinging these domain names or their associated IP addresses, "Request timed out" or a "100% packet loss" error was observed.
- **Geographical Distance:** The round-trip time depends on various factors such as nature of transmission medium, LAN traffic, server response time, node count and congestion, and physical distance. As the geographical distance increases, the number of hops required to reach the destination also increases. This leads to an increase in the round-trip time, although it depends on many other factors as well, due to which the correlation is weak. The amount of traffic on the LAN can bottleneck a connection. Also, the greater the number of nodes a connection touches the slower it will be.
- **Packet Size:** The observations were taken using flipkart.com as the host and subnetonline.com as the ping utility. In general, RTT increases very slightly with increasing packet size, as every node in the network has to receive the entire packet or frame

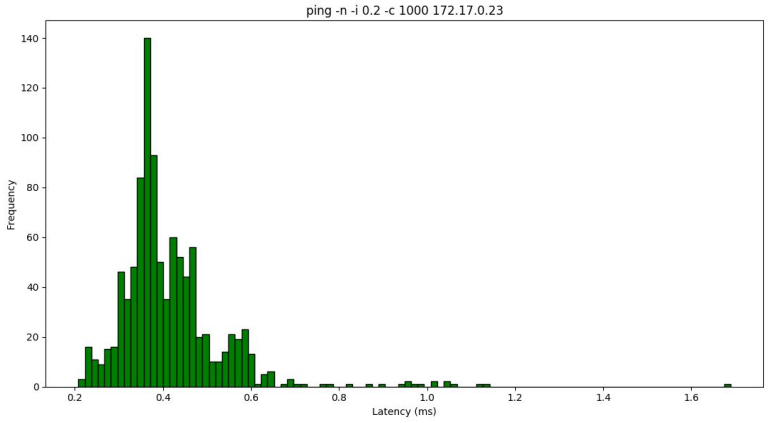
before it can forward it (increasing transmission time). A sudden increase in RTT is observed above 1500 bytes. This is due to the maximum transmission unit (MTU) which fragments the data packets above 1500 bytes into two or more smaller packets.

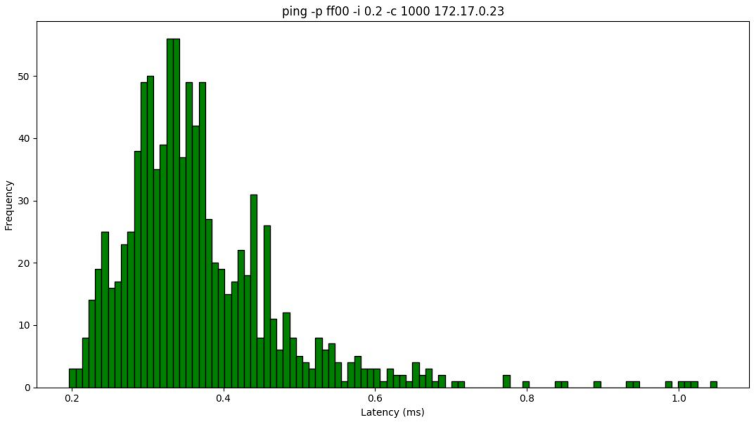
Packet Size (bytes)	64	128	256	512	1024	1280	1536	1792	2048
Avg. RTT (ms) @ 2:00 pm	173.524	173.649	173.741	173.483	173.508	173.600	173.820	173.758	173.807
Avg. RTT (ms) @ 2:00 am	172.815	172.614	172.813	172.708	173.093	172.824	173.186	173.227	173.279
Avg. RTT (ms) @ 5:00 pm	171.647	171.430	171.697	171.501	172.050	171.676	171.660	172.854	172.717

- **Time of the day:** When a server is overwhelmed with requests, especially during a DDoS attack, its ability to respond efficiently can be inhibited, resulting in increased RTT. Hence, during peak hours of the day such as around 2:00 pm, high latency was observed on flipkart.com (being hosted on a server located in India) in comparison to 2:00 am, when most people are asleep.

3. Ping command options

Command	Packets transmitted	Packets received	Min latency (ms)	Max latency (ms)	Avg. latency (ms)	Median latency (ms)
ping -n -i 0.2 -c 1000 172.17.0.23	1000	1000 (0%)	0.208	1.691	0.414	0.352
ping -p ff00 -i 0.2 -c 1000 172.17.0.23	1000	995 (0.5%)	0.197	18.408	0.462	0.389





- '-n' specifies that no attempt will be made to lookup symbolic names for host addresses. Hence, it is faster than normal ping. So, the mean latency is higher in the second case than the mean latency in the first case. '-p' is used to pad bytes to fill out the packets that are sent which is useful for diagnosing data-dependent problems in a network. '-p ff00' will cause the packet to be padded with the pattern 1111 1111 0000 0000. Due to 1 bit transition in the bit pattern, the clocks will go out of synchronization at both sender and receiver end resulting in higher packet loss than the first case.

4. *ifconfig* and *route*

A. ifconfig: ifconfig is a system administration utility for network interface configuration. It has features for configuring, controlling, and querying TCP/IP network interface parameters. Common uses for ifconfig include setting the IP address and netmask of a network interface and disabling or enabling an interface.

- **eno1, lo, virbr0 and wlo1** are the names of the active network interfaces on the system. **eno1:** is the first Ethernet interface. 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. **wlo1:** is the name of the first wireless network interface on the system.
- **inet addr:** indicates the machine IP address **broadcast:** denotes the broadcast address **netmask:** is the network mask. **UP:** This flag indicates that the kernel modules related to the Ethernet interface have been loaded.

- **BROADCAST:** Denotes that the Ethernet device supports broadcasting - a necessary characteristic to obtain an IP address via DHCP. **RUNNING**: The interface is ready to accept data. **MULTICAST**: This indicates that the Ethernet interface supports multicasting. Multicast allows a source to send a packet(s) to multiple machines as long as the machines are watching out for that packet.
- **MTU:** 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. **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.
- **RX Packets, TX Packets:** The total number of packets received and transmitted respectively. In the output shown above, the total errors are 0, no packets are dropped and there are no overruns. **RX Bytes, TX Bytes:** These indicate the total amount of data that has passed through the Ethernet interface either way. As long as there is some network traffic being generated via the Ethernet device, both the RX and TX bytes will go on increasing.

```
lavishgulati@lavishgulati:~$ ifconfig
eno1: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether c8:d3:ff:d4:57:c5  txqueuelen 1000  (Ethernet)
        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

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 2954  bytes 275776 (275.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2954  bytes 275776 (275.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 192.168.122.1  netmask 255.255.255.0  broadcast 192.168.122.255
        ether 52:54:00:bc:db:81  txqueuelen 1000  (Ethernet)
        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

wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.150.37.186  netmask 255.255.248.0  broadcast 10.150.39.255
        inet6 fe80::51ce:6753:498e:6c52  prefixlen 64  scopeid 0x20<link>
        ether 30:e3:7a:a0:ca:17  txqueuelen 1000  (Ethernet)
        RX packets 296099  bytes 110200785 (110.2 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 106938  bytes 40857556 (40.8 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

B. Display Information of All Network Interfaces: The following ifconfig command with -a argument will display information of all active or inactive network interfaces on the server. It displays the results for eth0, lo, sit0 and tun0.

- **View Network Settings of Specific Interface:** Using interface name (eth0) as an argument with "ifconfig" command will display details of the specific network interface.
- **How to Enable a Network Interface:** The "up" or "ifup" flag with interface name (eth0) activates a network interface, if it is not in active state and allowing to send and receive information. For example, "ifconfig eth0 up" or "ifup eth0" will activate the eth0 interface.
- **How to Disable a Network Interface:** The "down" or "ifdown" flag with interface name (eth0) deactivates the specified network interface. For example, "ifconfig eth0 down" or "ifdown eth0" command deactivates the eth0 interface if it is in active state.
- **How to Assign an IP Address to Network Interface:** To assign an IP address to a specific interface, use the following command with an interface name (eth0) and IP address that you want to set. For example, "ifconfig eth0 172.16.25.125" will set the IP address to interface eth0.

C. Route command is used to show/manipulate the IP routing table. It is primarily used to set up static routes to specific hosts or networks via an interface.

- **Destination:** The destination network or destination host. **Gateway:** The gateway address. **Genmask:** The netmask for the destination net; 255.255.255.255 for a host destination and 0.0.0.0 for the default route.
- **Flags:** Possible flags include
  - U (the route is up)
  - H (target is a host)
  - G (use gateway)
  - R (reinstate route for dynamic routing)
  - D (dynamically installed by daemon or redirect)
  - C (cache entry)
- **Metric:** The distance to the target (usually counted in hops). It is not used by recent kernels but may be needed by routing daemons. **Ref:** Number of references to this route. (Not used in the Linux kernel.) **Use:** Count of lookups for the route. Depending on the use of -F and -C this will be either route cache misses (-F) or hits (-C).

```
lavishgulati@lavishgulati:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.19.2.1       0.0.0.0         UG    20100  0        0 eno1
10.19.2.0       0.0.0.0         255.255.254.0   U     100    0        0 eno1
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 eno1
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
lavishgulati@lavishgulati:~$ route -Cn
Kernel IP routing cache
Source          Destination     Gateway         Flags Metric Ref    Use Iface
```

Iface: Interface to which packets for this route will be sent.

```
lavishgulati@lavishgulati:~$ sudo route add -host 193.168.23.132 reject
lavishgulati@lavishgulati:~$ sudo route add -host 193.168.23.132 reject
SIOCADDRT: File exists
```

D.

- **Displaying numerical IP address:** This command displays output in full numerical form. Example: **route -n**

```
lavishgulati@lavishgulati:~$ sudo route add -net 127.0.0.0 netmask 255.0.0.0 metric 1024 dev lo
lavishgulati@lavishgulati:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    100    0        0 eno1
10.19.2.0       0.0.0.0         255.255.254.0   U     100    0        0 eno1
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 eno1
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
193.168.23.131  -               255.255.255.255 !H    0      -        0 -
193.168.23.132  -               255.255.255.255 !H    0      -        0 -
```

- **Adding a new route:** adds a normal loopback entry, using netmask 255.0.0.0 and associated with the "lo" device. Example: **route add -net 127.0.0.0 netmask 255.0.0.0 metric 1024 dev lo**
- **Routing cache information:** Kernel maintains a routing cache table to route the packets faster. Example: **route -Cn**
- **Block access to a single host:** We can block access to a particular host or network by rejecting routes to it. Given below is an example of blocking access to host with the IP address 193.168.23.131. Example: **route add -host 193.168.23.131 reject**

---

5. *netstat*

A. netstat: netstat (network statistics) is a command-line network utility that displays network connections for Transmission Control Protocol, routing tables, and a number of network interface and network protocol statistics. It is used for finding problems in the network and to determine the amount of traffic on the network as a performance measurement.

B. netstat -at | **egrep "ESTABLISHED"** is used to list out the established TCP connections.

```
lavishgulati@lavishgulati:~$ netstat -at
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:27017         0.0.0.0:*               LISTEN
tcp        0      0 localhost:mysql         0.0.0.0:*               LISTEN
tcp        0      0 lavishgulati:domain     0.0.0.0:*               LISTEN
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN
tcp        0      0 localhost:ipp           0.0.0.0:*               LISTEN
tcp        0      0 lavishgulati:50758      180.149.60.171:https    ESTABLISHED
tcp        0      0 lavishgulati:45214      a118-214-44-253.d:https ESTABLISHED
tcp        0      0 lavishgulati:34216      sb-in-f189.1e100.:https ESTABLISHED
tcp        0      0 lavishgulati:40426      maa05s04-in-f3.1e:https ESTABLISHED
tcp        0      0 lavishgulati:32908      ec2-34-238-180-79:https ESTABLISHED
tcp        0      0 lavishgulati:49820      edge-star-shv-02-:https ESTABLISHED
tcp        0      0 lavishgulati:52862      maa03s23-in-f14.1:https ESTABLISHED
tcp        0      0 lavishgulati:54892      maa05s06-in-f3.1e:https ESTABLISHED
tcp        0      0 lavishgulati:35484      ec2-52-10-115-210:https ESTABLISHED
tcp        0      0 lavishgulati:38054      maa05s09-in-f14.1:https ESTABLISHED
tcp        0      0 lavishgulati:49452      180.149.60.168:https    ESTABLISHED
tcp        0      0 lavishgulati:44374      maa03s28-in-f14.1:https ESTABLISHED
tcp        1      0 lavishgulati:57978      maa05s09-in-f16.1e:http CLOSE_WAIT
tcp        0      0 lavishgulati:49862      ec2-3-94-70-57.co:https ESTABLISHED
tcp        0      0 lavishgulati:53222      server-13-33-142-:https ESTABLISHED
tcp        0      0 lavishgulati:52174      74.125.24.188:5228      ESTABLISHED
tcp        0      0 lavishgulati:48146      maa03s31-in-f14.1:https ESTABLISHED
tcp        0      0 lavishgulati:49818      edge-star-shv-02-:https ESTABLISHED
tcp6       0      0 ip6-localhost:ipp       [::]:*                  LISTEN
```

The "Proto" column tells us if the socket listed is TCP or UDP. Those are network protocols. TCP makes reliable connections but slows down dramatically if the network quality is bad. UDP stays fast but may lose a few packets or deliver them in the wrong order.

The "Recv-Q" and "Send-Q" columns tell us how much data is in the queue for that socket, waiting to be read (Recv-Q) or sent (Send-Q). In short: if this is 0, everything's ok, if there are non-zero values anywhere, there may be trouble.

The "Local Address" and "Foreign Address" columns tell to which hosts and ports the listed sockets are connected. The local end is always on the computer on which you're running netstat, and the foreign end is about the other computer (could be somewhere in the local network or somewhere on the internet).

The "State" column tells in which state the listed sockets are. The TCP protocol defines states, including "LISTEN" (wait for some external computer to contact us) and "ESTABLISHED" (ready for communication). The stranger among these is the "CLOSE WAIT" state shown by two sockets. This means that the foreign or remote machine has already closed the connection, but that the local program somehow hasn't followed suit.

```
lavishgulati@lavishgulati:~$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 wlo1
10.150.32.0     0.0.0.0         255.255.248.0   U         0 0          0 wlo1
link-local      0.0.0.0         255.255.0.0     U         0 0          0 wlo1
192.168.122.0   0.0.0.0         255.255.255.0   U         0 0          0 virbr0
```

C. netstat -r displays the kernel routing tables. Most of the columns in the output are the same as in the route command.

- **MSS column** lists the value of the Maximum Segment Size for this line. The MSS is a TCP parameter and is used to split packets when the destination has indicated that it somehow can't handle larger ones.

- **Window column** shows the window size, which indicates how many TCP packets can be sent before at least one of them has to be ACKnowledged.
- **irtt column** stands for Initial round trip time and may be used by the kernel to guess the best TCP parameters without waiting for slow replies.

```
lavishgulati@lavishgulati:~$ netstat -i
Kernel Interface table
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eno1      1500        0      0      0 0             0      0      0      0 BMU
lo       65536     3701      0      0 0          3701      0      0      0 LRU
virbr0    1500        0      0      0 0             0      0      0      0 BMU
wlo1      1500   423559      0      0 0        139913      0      0      0 BMRU
```

```
lavishgulati@lavishgulati:~$ netstat -su
IcmpMsg:
    InType3: 9
    OutType3: 16
Udp:
    86129 packets received
    16 packets to unknown port received
    12 packet receive errors
    7878 packets sent
    12 receive buffer errors
    1 send buffer errors
    IgnoredMulti: 17529
UdpLite:
IpExt:
    InNoRoutes: 3
    InMcastPkts: 26179
    OutMcastPkts: 1118
    InBcastPkts: 17571
    OutBcastPkts: 30
    InOctets: 103512638
    OutOctets: 50776238
    InMcastOctets: 2302760
    OutMcastOctets: 179832
    InBcastOctets: 4128770
    OutBcastOctets: 1630
    InNoECTPkts: 226254
```

D. netstat -i is used to display the status of all network interfaces. The number of interfaces as shown in the figure is 4.

E. netstat -su is used to show the statistics of all UDP connections.

F. Loopback interface 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 can perform the following functions:
Device identification: The loopback interface is used to identify the device. While any interface address can be used to determine if the device is online, the loopback address is the preferred method. **Routing information:** The loopback address is used by protocols such as OSPF to determine protocol-specific properties for the device or network. Further, some commands such as ping mpls require a loopback address to function correctly. **Packet filtering:** Stateless firewall filters can be applied to the loopback address to filter packets originating from, or destined for, the Routing Engine.

---

6. *traceroute*

Time of the day	flipkart.com	reddit.com	hulu.com	alibaba.com	codeforces.com	rbi.org.in
6:00 pm	11	9	15	20	20	10
2:00 am	11	9	16	20	19	10
1:00 pm	11	9	15	19	20	10

- The hosts used are flipkart.com, reddit.com, hulu.com, alibaba.com, codeforces.com and rbi.org.in, with observations taken at 6:00 pm, 2:00 am and 1:00 pm. Uptrends traceroute utility was used to take the observations.

A. The common hops found are the IP addresses: 164.52.192.1, 180.179.194.122 and 180.179.197.121. The first IP address (164.52.192.1) is that of the server itself on which the tests were conducted. The latter two IP addresses must be the local nodes, servers or gateways of the internet service provider (ISP), and hence all requests pass through these addresses.

B. The destination server address for reddit.com is different at times 1:00 pm and 2:00 am. This is due to load balancing by the servers at times of increased network traffic. When there is too much congestion, part of the network traffic is redirected to a different server, thus relieving the former server of the increased congestion.

C. Traceroute may not find a complete path to some hosts. As also explained in the previous exercise, some sites such as google.com, amazon.com and microsoft.com etc. have had DoS attacks in the past, and thus have filtered ICMP requests. Due to this, traceroute is unable to find a complete path to the host and is blocked by some firewall or timed out.

D. Yes, it is possible to find the route to certain hosts which fail to respond with ping experiment. Traceroute is not a protocol itself, it is an application and the protocols used depends on the implementation of the utility. Primarily this is ICMP. This uses UDP packets with an incrementing TTL field to map the hops to the final destination. Due to this difference, some networks may block ICMP by default, so PING will fail but a traceroute from a Linux device may still work.

7. ARP

A. ARP stands for Address Resolution Protocol. This protocol is used by network nodes to match IP addresses to MAC addresses. **arp** command displays and modifies entries in the Address Resolution Protocol (ARP) cache, which contains one or more tables that are used to store IP addresses and their resolved Ethernet or Token Ring physical addresses. The arp command is useful for viewing the ARP cache and resolving address resolution problems.

- **Address** is the IP address of the other workstation to which communication was established.
- **Hardware Type:** Hardware Type field in the Address Resolution Protocol (ARP) Message specifies the type of hardware used for the local network transmitting the Address Resolution Protocol (ARP) message. Ethernet is the common Hardware Type and the value for Ethernet is 1.
- **Sender Hardware Address:** Layer 2 (MAC Address) address of the device sending the message. **Target Hardware Address:** Layer 2 (MAC Address) of the intended receiver. This field is ignored in requests.
- **Flags** indicate if the mac address has been learned, manually set, published (announced by another node than the requested) or is incomplete.

```
lavishgulati@lavishgulati:~$ sudo arp -s 10.19.3.2 00:0c:29:c0:94:bf
lavishgulati@lavishgulati:~$ sudo arp -s 10.19.3.3 00:0c:29:c0:94:bf
lavishgulati@lavishgulati:~$ sudo arp -s 10.19.3.4 00:0c:29:c0:94:bf
lavishgulati@lavishgulati:~$ sudo arp -s 10.19.3.5 00:0c:29:c0:94:bf
lavishgulati@lavishgulati:~$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
10.19.3.185              ether   f8:ca:b8:61:12:39   C                     eno1
.                        ether   94:44:52:95:d7:a0   C                     wlo1
10.19.3.142              ether   94:44:52:95:d7:a1   C                     eno1
_gateway                 ether   ec:44:76:74:60:42   C                     eno1
10.19.3.2                ether   00:0c:29:c0:94:bf   CM                    eno1
10.19.3.1                ether   00:0c:29:c0:94:bf   CM                    eno1
10.19.3.5                ether   00:0c:29:c0:94:bf   CM                    eno1
10.19.3.4                ether   00:0c:29:c0:94:bf   CM                    eno1
10.19.3.3                ether   00:0c:29:c0:94:bf   CM                    eno1
10.19.3.110              ether   0c:80:63:16:c9:52   C                     eno1
```

B. sudo arp -s 10.0.0.2 00:0c:29:c0:94:bf
This commands tells local ARP table that the host with IP address 10.0.0.2 and MAC address 00:0c:29:c0:94:bf has been added permanently.
sudo arp -d 10.0.0.2 deletes a static ARP entry from the local ARP table.

C. By default, entries in the ARP table stay cached for 60 seconds which is stored in the file /proc/sys/net/ipv4/neigh/default/gc_stale_time. A trial and error method to discover the timeout value is to add a temporary entry in the table and keep checking the table after fixed intervals of time. The time after which it is deleted from the cache is the required cache timeout. Alternatively, we can also use binary search for finding the cache timeout. If the entry is still in the table, increase the proposed time and check again. Similarly, if the entry is deleted from the table, decrease the proposed time and check again. The proposed time should converge approximately to the cache timeout.

D. The scenario where two IPs map to the same Ethernet Address occurs when a router or a gateway connects two or more subnet ranges. When communicating with machines on the same subnet range, MAC address is used for directing the packages. In the ARP Table, the IPs of the devices which are connected in the other subnet range have the ethernet address/MAC address as that of the Router or Gateway which connects the two subnet ranges. ARP table is referred to convert these IP addresses to the MAC address and packets are sent to it. The router then uses its routing table and sends the packet further to the correct device.

---

8. Local Network Analysis

The command used for the analysis is **nmap -n -sP 10.19.3.0/21** scanning 2048 IP addresses in the Lohit hostel.
We can easily observe that the number of hosts is low during times of inactivity such as late night and early morning (2:00 - 9:00) and during class hours (9:00 - 11:00 and 14:00 - 17:00). The number of hosts observed was more during the break time (12:00 - 14:00) and after class hours (17:00 - 23:00).