CS349 ASSIGNMENT 1
Learning Network Diagnostic Tools

Rashi Singh
170101052

Q1 : PING command :

a) To specify the number of echo requests : ping **-c count** <host_name>

b) To set time interval btw 2 successive pings : ping **-i interval** <host_name>

- * ***min*** *value of interval :* ***0.2 sec*** *for non super-users*
- * ***default*** *value of interval :* ***1 sec*** *for all users*

c) To send packets without waiting for reply : ping **-l count** <host_name>

- * ***max*** *value of count :* ***3*** *for non super-users*

d) To specify packet size : ping **-s size** <host_name>

- * *Additional* ***8 bytes*** *for* ***ICMP header*** *and* ***20 bytes*** *for* ***IP header***
- * *For packet size = 32 bytes,* ***total packet size*** *= 32 + 28 =* ***60 bytes***

Q2 : PING - RTT experiment :

* All readings were taken via http://www.spfld.com/ping.html (server location: New Jersey)

```
PING stackoverflow.com (151.101.65.69) 56(84) bytes of data.

    --- stackoverflow.com ping statistics ---
    30 packets transmitted, 30 received, 0% packet loss, time
    29016ms
    rtt min/avg/max/mdev = 4.754/4.887/5.305/0.111 ms
```

```
--- wikipedia.org ping statistics ---
30 packets transmitted, 0 received, 100% packet loss, time 29001ms
```

PACKET LOSS :

There was no case of packet loss greater than 0% for packet size 64 bytes. However, packet loses might occur due to network congestion. Sometimes, 100% packet loss is also witnessed, as hosts block ICMP packets (low-priority) to reduce traffic.

CORRELATION B/W RTT AND PACKET SIZE :

* From the chart below, the following observations can be made :
  - a) For smaller packet sizes (less than 2048 bytes), RTT is more or less a stable value.
  - b) For packet size of 2048, only 2 out of 6 hosts received any packets. Hence, only imdb, amazon map the RTT corresponding to 2048 byte packet size.
* These observations can be expained from the fact that MTU (Maximum Transmission Unit, i.e, the maximum number of octets that the network interface can handle) has a default value of 1500, Therefore, packets of size greater than 1500 are either rejected or broken into smaller packets resulting into a higher RTT.

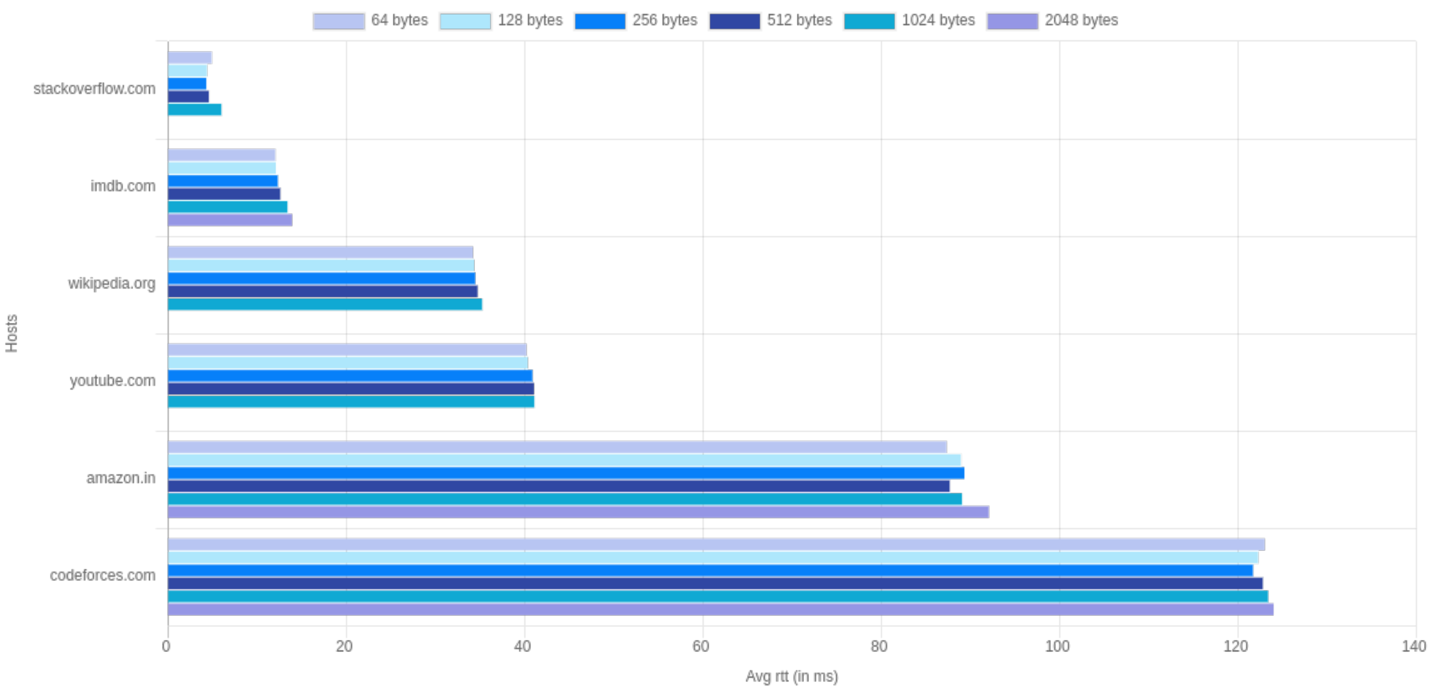


Illustration 1 : CORRELATION BETWEEN RTT AND PACKET SIZE

STRONG CORRELATION B/W RTT AND GEOGRAPHICAL DISTANCE :

- The *ROUND TRIP TIME (RTT)* is shows *"strong, positive linear relation"* with distance bcs :
  - a) Increase in propagation delay – This occurs bcs of increased distance
  - b) Increase in processing and queuing delay – This occurs bcs of more no. of hops
- This can be seen in the table below. RTT increases with distance, especially in case of inter continental pings (Russia, Ireland from USA).
- DEVIATIONS : Significant deviations from linear relation arise due to large-scale routing behaviours, like the noise introduced by the procedure.

HOST_NAME	**IP_ADDRESS**	**LOCATION**	**RTT-3:15PM**	**RTT-7:30PM**	**RTT-10:30PM**	**AVG_RTT**
stackoverflow.com	151.101.65.69	San Francisco, USA	4.887 ms	3.835 ms	4.818 ms	4.513 ms
imdb.com	52.94.225.248	Virginia, USA	12.019 ms	11.739 ms	11.674 ms	11.810 ms
wikipedia.org	208.80.154.224	New York, USA	34.197 ms	34.174 ms	33.510 ms	33.960 ms
youtube.com	64.233.177.91	California, USA	40.194 ms	40.540 ms	40.705 ms	40.479 ms
amazon.in	52.95.116.115	Dublin, Ireland	87.318 ms	86.243 ms	87.009 ms	86.856 ms
codeforces.com	81.27.240.126	Moscow, Russia	123.006 ms	121.192 ms	121.803 ms	122 ms

CORRELATION B/W RTT AND TIME OF THE DAY :

- From the data in the above table, it can be observed that RTT varies with time of the day. This is because higher congestion leads to more ROUND TRIP TIME. Hence it can said that the network traffic is less at 7:30 PM IST as compared to 3:15 PM and 10:30 PM IST.

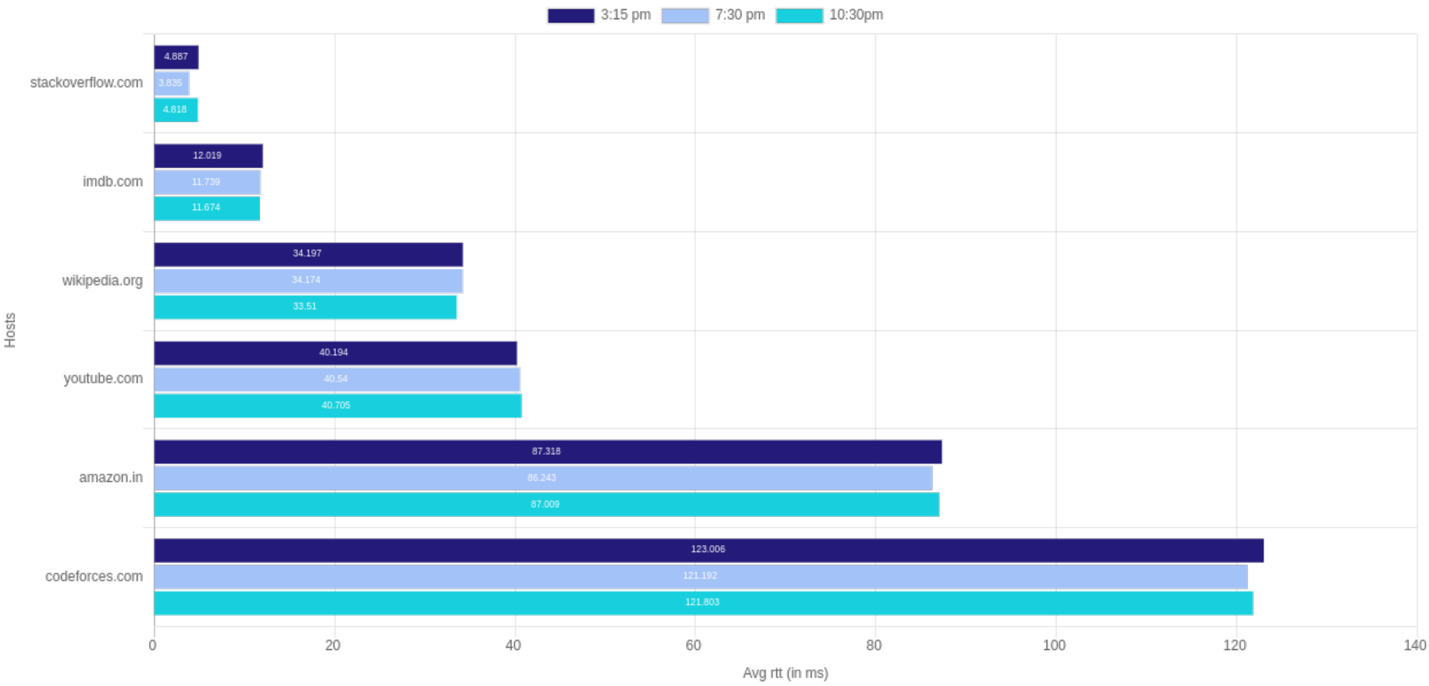


Illustration 2 : CORRELATION BTW RTT ,TIME AND DISTANCE

Q3 : PING -n and PING -p :

Command	Packets Sent	Packets Received	Packet Loss rate	Min Latency	Max Latency	Mean Latency	Median Latency
ping -n -c 1000 172.16.112.12	1,000	1,000	0%	0.246	16	0.595	0.5
ping -p ff00 -c 1000 172.16.112.12	1,000	999	0.1%	0.258	16.1	0.617	0.504

- Both the commands are sending 1000 packets to the same host. Hence, they are very similar, except for two aspects which are as follows :
  - a) The **-n (numeric only)** option is expected to be **faster**, as it displays the raw ip addresses without looking for more human friendly host names (like google.com)
  - b) The **-p ff00 (pattern)** option fills the packet with **16 pad bits.** This is used for dianosing data-

dependent problems, like **synchronisation problem** of clocks due to only one transition from 1 to 0 in 1111111100000000 (ff00).

* Due to these apects, the following observations occur in the mapping of <frequency,latency> :
  a) The mean latency of -n command < the mean latency of -p command ...(1st aspect)
  b) Since synch is data (pattern) dependent, there are higher chances of packet losses, which is observed as shown in the table above. ...(2nd aspect)

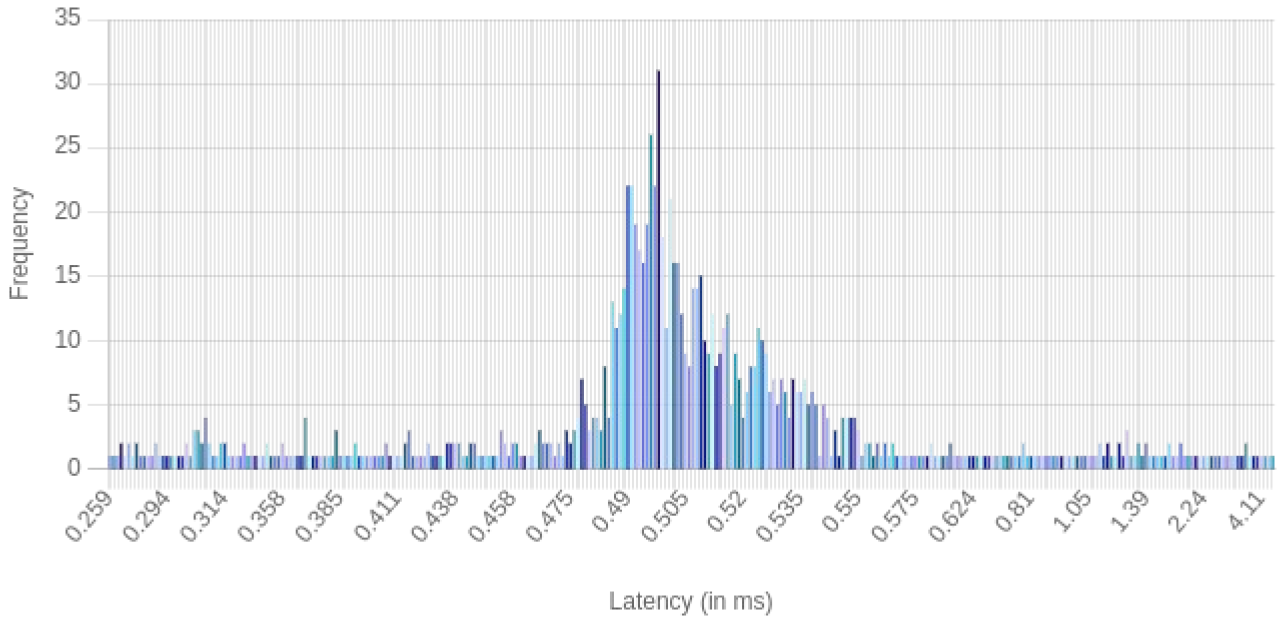


Illustration 3 : Normal Distribution for ping -n c 1000 172.16.112.12

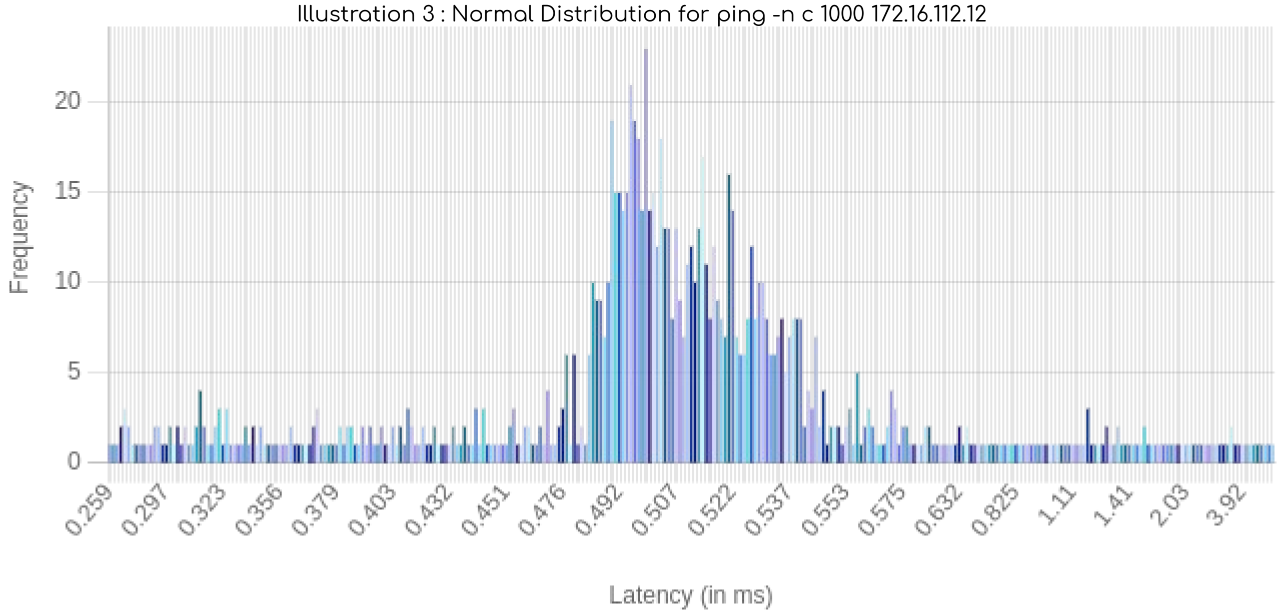


Illustration 4 : Normal Distribution for ping -p ff00 -c 1000 172.16.112.12

Q4 : IFCONFIG and ROUTE command :

```
rashi@rashis-aspire-e5-575g:~$ ifconfig
enp4s0f1: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether a8:1e:84:55:ab:57  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 402  bytes 34644 (34.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 402  bytes 34644 (34.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.42.0.161  netmask 255.255.255.0  broadcast 10.42.0.255
        inet6 fe80::ed82:63a2:ba94:42a3  prefixlen 64  scopeid 0x20<link>
        ether 3c:a0:67:6c:14:2d  txqueuelen 1000  (Ethernet)
        RX packets 12519  bytes 15344697 (15.3 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5747  bytes 1354683 (1.3 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

a) I FCONFIG command :

* **I**nter **F**ace **CONFIG**uration
* Shows/Configures the list of network interfaces that are "UP" (enabled)
* Normal Invocation :
  ifconfig interface (address (parameters))
* It is used while booting to set up interfaces and while debugging and system tuning.

As observed from the image, it displays the <interface-name> followed by its associated flags, address info and the packet statistics. (**RX** : received and **TX** : transmitted)

```
enp4s0f1:
|  |  |
v  |  |
en |  | -----> ethernet
   v  |
   p4 | -----> bus no (4)
      v
      s0f1 --> slot no (0f1)
```

I NTER FACE NAMING :

<type-of-interface, bus no, slot no>

Similarly, wlp3s0 refers to the following :

wl	–	wireless
p3	–	bus no 3
s0	–	slot 3

Interface "**lo**" (as seen in ifconfig result on prev page) however represents **loopback**, which is the computer's ref to itself. It is a virtual network interface to communicate with itself for diagnostics, troubleshooting and to connect to servers running on the local machine.

FLAGS : Represent the current status of the interface, for eg : `<UP, BROADCAST, RUNNING, MULTICAST>`

a) UP : The interface is accessible to the IP layer, hence it has been assigned an address and routing table.
b) BROADCAST : This indicates that the interface supports broadcasting.
c) RUNNING : This indicates that the network driver has been loaded and has initialized the interface.
d) MULTICAST : MULTICAST is like BROADCAST except that instead of automatically including everybody, the only people who receive packets sent to a multicast address are those programmed to listen to it.

ADDRESS INFO. :

a) inet : Same as **IPv4***
b) inet6 : Same as **Ipv6****

```
inet 10.150.37.172  netmask 255.255.248.0  broadcast 10.150.39.255
inet6 fe80::1c9c:a06b:4ec0:504  prefixlen 64  scopeid 0x20<link>
ether 3c:a0:67:6c:14:2d  txqueuelen 1000  (Ethernet)
```

IPv4* : 32 bit numeric IP address*   ***IPv6 : 128 bit aplhanumeric IP address*

c) netmask : Network Mask for the associated IP address
d) broadcast : 32 bit broadcast address
e) MAC addr (H/w addr) : Address assigned to the LAN card. This addr uniquely defines the device
f) scopeID : This defines the scope of the IP address. For eg, in the img above, Ipv6 addr is local.
g) prefixLen : No of bits in the IP address that are to be used as the subnet mask
h) TXqueueLen : Limits the number of packets in the transmission queue in the interface's device driver
i) MTU : Max number of octets the interface is able to handle in one transaction

b) I FCONFIG instructions :

a) Ifconfig **-a**	:	shows all network interfaces including the DOWN ones
b) ifconfig <interface-name> **UP/DOWN**	:	enables/disables the mentioned interface
c) ifconfig <interface-name> **mtu count**	:	changes MTU value for the mentioned interface
d) ifconfig <interface-name> **netmask addr**	:	assigns given netmask addr to the given interface
ifconfig <interface-name> **broadcast addr**	:	assigns given broadcast addr to the given interface
ifconfig <interface-name> **addr**	:	assigns given addr as IP to the given interface
e) ifconfig <interface-name> **promisc**	:	enables promiscuous mode
ifconfig <interface-name> **-promisc**	:	disables promiscuous mode

in **promiscuous** mode, the driver does not check whether the packet is meant/not meant for itself and simply **accepts all packets

c) ROUTE command :

Normal invocation : route (-f) (-p) (command (destination) (mask subnetmask) (gateway) (metric))

```
rashi@rashis-aspire-e5-575g:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    600    0        0 wlp3s0
10.150.32.0     0.0.0.0         255.255.248.0   U     600    0        0 wlp3s0
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 wlp3s0
```

Route is used to show, manipulate kernel's IP routing table
Primary use : To set up static routes to specific hosts/networks via interfa-ce after configuring with ifconfig

IMPORTANT KEYWORDS :

a) Destination	:	The destination network or destination host.
b) Gateway	:	The gateway* address or 0.0.0.0, if none set.
c) Genmask	:	The netmask** for the destination net
d) Metric***	:	The metric option assigns an integer cost metric (distance to target in hops)
e) Ref	:	No of references to the route
f) Use	:	The count of lookups for a route
g) Iface	:	Interface to which packets for this route will be sent
h) Flags	:	**U** (route is up), **H** (target is a host), **G** (use gateway), **C** (cache entry), **!** (reject route), **R** (reinstate route for dynamic routing), **D** (dynamically installed), **M** (modified)

*   *Gateways regulate traffic btw two dissimilar networks, while routers regulate traffic btw similar ones*
**   *255.255.255.255' for a host destination and '0.0.0.0' for the default route.*
***   *Range – [1-9999], used to calculate the fastest, most reliable, and least expensive routes.*

d) ROUTE instructions :

```
rashi@rashis-aspire-e5-575g:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.42.0.1       0.0.0.0         UG    20600  0        0 wlp3s0
10.42.0.0       0.0.0.0         255.255.255.0   U     600    0        0 wlp3s0
```

`route -n`

Show numerical addresses instead of trying to find symbolic host names

```
rashi@rashis-aspire-e5-575g:~$ route -e
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         strix-15        0.0.0.0         UG        0 0          0 wlp3s0
10.42.0.0       0.0.0.0         255.255.255.0   U         0 0          0 wlp3s0
```

`route -e`

Use netstat(8) - format for displaying the routing table.

```
root@rashis-aspire-e5-575g:/home/rashi# route add -host 172.16.112.3 reject
root@rashis-aspire-e5-575g:/home/rashi# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.42.0.1       0.0.0.0         UG    20600  0        0 wlp3s0
10.42.0.0       0.0.0.0         255.255.255.0   U     600    0        0 wlp3s0
172.16.112.3    -               255.255.255.255 !H    0      -        0 -
```

`route add -host <addr> reject`

Rejecting routing to a particular host
**Notice !H in flags*

`route add -net 127.0.0.0 netmask 255.0.0.0 metric 1024 dev lo`

Adds the normal loopback entry, using netmask 255.0.0.0 and associated with the "lo" device

```
root@rashis-aspire-e5-575g:/home/rashi# route add -net 127.0.0.0 netmask 255.0.0.0 metric 1024 dev lo
root@rashis-aspire-e5-575g:/home/rashi# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.42.0.1       0.0.0.0         UG    600    0        0 wlp3s0
10.42.0.0       0.0.0.0         255.255.255.0   U     600    0        0 wlp3s0
127.0.0.0       0.0.0.0         255.0.0.0       U     1024   0        0 lo
```

Q5 : NETSTAT command :

a) USE :

Netstat is a handy cross-platform (Linux, Windows etc.), which is mainly used for trouble-shooting and debugging. It is used to :

1) Display incoming and outgoing network connections
2) Display routing tables
3) Display number of network interfaces
4) Display network protocol statistics

To show all established TCP connections :

`netstat -at | grep "ESTABLISHED"`

```
rashi@rashis-aspire-e5-575g:~$ netstat -at | grep "ESTABLISHED"
tcp        0      0 rashis-aspire-e5-:57958 maa05s06-in-f3.1e:https ESTABLISHED
tcp        0      0 rashis-aspire-e5-:35362 maa03s31-in-f14.1:https ESTABLISHED
tcp        0      0 rashis-aspire-e5-:45662 maa03s29-in-f10.1:https ESTABLISHED
tcp        0      0 rashis-aspire-e5-:57388 edge-star-shv-02-:https ESTABLISHED
tcp        0      0 rashis-aspire-e5-:49014 maa05s04-in-f3.1e:https ESTABLISHED
tcp        0      0 rashis-aspire-e5-:40834 172.217.194.155:https   ESTABLISHED
```

Description of different fields (in order):

Proto : The protocol (tcp, udp, udpl, raw) used by the socket.
Recv-Q : The # of bytes not copied (waiting to be sent) by the user program connected to socket.
Send-Q : The # of bytes not acknowledged (waiting to be read) by the remote host.
Local Address : **<Address, port number>** of the local end of the socket.
If the **--numeric** option is not specified,
the socket address --> canonical hostname
the port number i--> corresponding service name.
Foreign Address : <Address, port number> of the remote end of the socket.
State : This field refers to the state of the socket. It is left **blank** in **UDP/UDPLite** connections.
Some values are :
– **ESTABLISHED** (the socket has an established connection),
TIME_WAIT (The socket is waiting after connection is closed by remote machine to handle packets still in the network),
LISTEN (the socket is listening for incoming connections), etc.

c) netstat -r :

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 wlp3s0
10.150.32.0     0.0.0.0         255.255.248.0   U         0 0          0 wlp3s0
link-local      0.0.0.0         255.255.0.0     U         0 0          0 wlp3s0
```

\`**MSS**\` : Maximum Segment Size(size of largest datagram constructed for transmission),
\`**Window**\` : default window size(maximum amount of data the system will accept in one burst from a remote host),
\`**Irtt**\` indicates Initial Round Trip Time for TCP connections over this route.
Please refer Q4 part d for the remaining fields

d) netstat -i : My machine has 3 interfaces enp4s0f1, lo,wlp3s0. Please refer Interface naming in Q4

```
rashi@rashis-aspire-e5-575g:~$ netstat -i
Kernel Interface table
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
enp4s0f1  1500        0      0      0 0             0      0      0      0 BMU
lo       65536     3676      0      0 0          3676      0      0      0 LRU
wlp3s0    1500  1672981      0      9 0        131134      0      0      0 BMRU
```

e) netstat -su : To show statistics of UDP connections

```
rashi@rashis-aspire-e5-575g:~$ netstat -su
IcmpMsg:
    InType0: 13
    InType3: 49
    InType11: 404
    OutType3: 55
    OutType8: 61
Udp:
    6651 packets received
    30 packets to unknown port received
    0 packet receive errors
    8834 packets sent
    0 receive buffer errors
    0 send buffer errors
    IgnoredMulti: 324197
UdpLite:
IpExt:
    InMcastPkts: 127
    OutMcastPkts: 877
    InBcastPkts: 324492
    OutBcastPkts: 13
    InOctets: 198381975
    OutOctets: 14760039
    InMcastOctets: 13540
    OutMcastOctets: 162472
    InBcastOctets: 34281154
    OutBcastOctets: 795
    InNoECTPkts: 527574
```

f) Loopback interface :

This is the computer's ref to itself. It is a virtual network interface to communicate with itself for diagnostics, troubleshooting and to connect to servers running on the local machine.

The range is 1 – 127.0.0.0/8
Local Host : 127.0.0,1

Q6 : TRACEROUTE command : Prints the route that a packet takes to reach the host.

Useful when you want to know about the route and about all the **hops that a packet takes**.Seeing the traceroute information can help you determine why your connections to a given server might be poor and can help you **identify problems.** It also shows you how systems are connected to each other, letting you see how your ISP connects to the Internet as well as **how the target system is connected.**

a) Hop counts : Below are the tables showing hop counts for each host

HOSTS	7:30am	7:30pm	11:30pm
youtube.com	11	14	13
wikipedia.org	14**	13	14
amazon.in	17**	14**	15**
imdb.com	17**	20**	28**
stackoverflow.com	13**	19**	25**
codeforces.com	12**	15**	15**

***represents hosts that were not reached in 64 hops*

Since only 2 out of 6 hosts from question 2 were reached using trace command, below is a table with some extra hosts who were traced by traceroute

HOSTS	7:30am	7:30pm	11:30pm
iitg.ac.in	2	2	2
google.com	11	11	11
twitter.com	11	10	10

a) Common hops : Just like the images below, the top 7 hops were observed to be common in all the traceroutes, like twitter.com, amazon.in, etc.

```
rashi@rashis-aspire-e5-575g:~$ traceroute youtube.com
traceroute to youtube.com (172.217.166.110), 64 hops max
 1   10.150.32.1  4.337ms  3.192ms  3.929ms
 2   192.168.193.1  1.715ms  1.731ms  1.671ms
 3   14.139.196.17  3.332ms  1.868ms  1.867ms
 4   10.119.254.241  2.315ms  2.170ms  1.979ms
 5   10.177.31.1  39.552ms  38.956ms  38.974ms
 6   10.255.238.205  40.711ms  39.061ms  38.981ms
 7   10.119.73.122  39.736ms  40.578ms  39.620ms
 8   72.14.213.20  43.572ms  43.329ms  43.355ms
 9   74.125.242.145  54.493ms  54.121ms  54.086ms
10   74.125.252.215  53.827ms  53.565ms  53.523ms
11   172.217.166.110  52.642ms  52.539ms  52.470ms
```

```
rashi@rashis-aspire-e5-575g:~$ traceroute google.com
traceroute to google.com (172.217.31.206), 64 hops max
 1   10.150.32.1  5.984ms  1.754ms  1.716ms
 2   192.168.193.1  1.709ms  1.601ms  1.693ms
 3   14.139.196.17  2.033ms  1.893ms  1.730ms
 4   10.119.254.241  19.260ms  2.193ms  2.051ms
 5   10.177.31.21  39.340ms  39.094ms  39.440ms
 6   10.255.238.205  39.863ms  39.596ms  39.490ms
 7   10.119.73.122  39.762ms  39.581ms  39.500ms
 8   72.14.195.128  43.459ms  43.326ms  46.662ms
 9   108.170.253.113  53.036ms  52.762ms  53.058ms
10   74.125.253.17  54.976ms  54.714ms  68.118ms
11   172.217.31.206  53.805ms  53.617ms  53.662ms
```

Here the first common entry is **10.150.32.1**, which is **my device**. The other common hops occur because the packets pass through the same routers (for instance in the example above both servers lie in USA, hence first the packets are routed to reach USA, before diverging to their respective location-based gateways.)

b) Route changes based on time of day : Since the internet commonly follows packet switching, the data is sent in packets which are routed by layer 2, layer 3 routers. This routing is affected by the load balancing techniues to reduce the effect of congestion in the link, due to which packets might go through diff paths to reach the same destination if one path is congested.

```
traceroute to imdb.com (52.94.237.74), 64 hops max
  1   10.42.0.1  1.772ms  1.622ms  1.603ms
  2   10.12.0.254  4.651ms  4.131ms  5.786ms
  3   172.17.0.50  2.403ms  2.339ms  2.518ms
  4   172.17.0.1  2.110ms  2.225ms  3.283ms
  5   192.168.193.1  2.025ms  1.834ms  2.986ms
  6   14.139.196.17  2.032ms  1.832ms  1.946ms
  7   10.119.254.241  2.232ms  2.178ms  2.882ms
  8   10.177.31.1  41.875ms  39.231ms  39.253ms
  9   10.255.238.205  39.438ms  40.658ms  39.034ms
 10   10.119.73.122  40.081ms  39.915ms  41.579ms
 11   *  115.248.104.230  50.882ms  *
 12   80.81.65.97  54.563ms  53.405ms  53.552ms
 13   62.216.135.230  324.193ms  409.357ms  409.332ms
 14   85.95.26.109  409.221ms  254.799ms  251.437ms
 15   85.95.25.5  255.481ms  252.521ms  257.402ms
 16   85.95.26.233  193.872ms  191.163ms  191.377ms
 17   85.95.26.41  404.276ms  409.421ms  254.714ms
```

```
traceroute to imdb.com (52.94.237.74), 64 hops max
  2   10.12.0.254  7.808ms  4.099ms  4.375ms
  3   172.17.0.50  2.449ms  3.130ms  2.429ms
  4   172.17.0.1  5.611ms  1.898ms  3.714ms
  5   192.168.193.1  1.781ms  1.760ms  1.797ms
  6   14.139.196.17  1.914ms  1.903ms  4.471ms
  7   10.119.254.241  2.134ms  1.876ms  1.950ms
  8   10.177.31.1  39.317ms  39.115ms  41.867ms
  9   10.255.238.205  39.461ms  39.232ms  39.164ms
 10   10.119.73.122  39.758ms  43.136ms  41.651ms
 11   125.22.85.29  49.149ms  50.191ms  49.057ms
 12   116.119.49.240  86.751ms  86.907ms  94.537ms
 13   120.29.215.241  80.273ms  80.516ms  80.551ms
 14   180.87.96.22  426.719ms  409.731ms  408.807ms
 15   180.87.12.1  381.163ms  379.311ms  277.131ms
 16   180.87.67.33  149.812ms  146.685ms  147.649ms
 17   120.29.217.67  363.092ms  155.295ms  252.012ms
```

As seen above, the intermediate hops for imdb.com are different for 7:30pm (left) and 11;30pm (right)

c) Traceroute unable to find any route :

```
traceroute to imdb.com (52.94.237.74), 64 hops max
  2   10.12.0.254  7.808ms  4.099ms  4.375ms
  3   172.17.0.50  2.449ms  3.130ms  2.429ms
  4   172.17.0.1  5.611ms  1.898ms  3.714ms ...
 24   52.93.249.165  395.669ms  408.794ms  287.679ms
 25   52.95.62.115  327.258ms  272.473ms  311.444ms
 26   *  *  *
 27   52.93.129.130  409.937ms  288.277ms  325.133ms
 28   54.239.42.182  284.163ms  331.732ms  408.772ms
 29   *  *  *
 30   *  *  *
 31   *  *  *
```

As shown in the image on the left, traceroute may be unable to find a route to some hosts because traceroue follows ICMP protocol which shares data via low priority packets. Due to this, the packets are prone to being dropped by networks in case of heavy congestion. Furthermore, these packets are also prone to being blocked by firewalls. Consecutive rows of * * * (asterisk) symbolize that packets are not acknowledged by the router in the given timelimit.

d) To find route when ping fails : Yes, it is possible to find the route to certain hosts which do not respond to ping. While ping works with simple ICMP, there is another tool known as **Tracert** which 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. Thus, by increasing the TTL based on the last router's response, the packet is saved from being dropped and can reach the host.

Q7 : ARP command : arp (-v) (-i if) (-H type) -a (hostname)

ARP stands for **A**ddress **R**esolution **P**rotocol. The primary function of this protocol is to resolve the IP address of a system to its mac address. ARP command shows the MAC addresses resolved from the IP address (please note : this resolution is not done by ARP command, it simply shows the ARP cache containing the resolved pairs)

a) Show complete ARP table : To show complete table, do not write any hostname after arp

arp : the image shows the complete table in default format of ARP

```
rashi@rashis-aspire-e5-575g:~$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
_gateway                 ether   00:25:b4:d9:f7:c0   C                     wlp3s0
10.150.39.149            ether   8c:85:90:5b:d7:84   C                     wlp3s0
```

arp -a : the image on the right shows the complete table in BSD style output format

```
rashi@rashis-aspire-e5-575g:~$ arp -a
_gateway (10.150.32.1) at 00:25:b4:d9:f7:c0 [ether] on wlp3s0
? (10.150.39.149) at 8c:85:90:5b:d7:84 [ether] on wlp3s0
```

IMPORTANT KEYWORDS :

a) Address	:	This is the IPv4 address of the dest which has been resolved to MAC address
b) HWType	:	This represents the type of hardware, which in the image above is "ethernet"
c) HWAddress	:	This is the MAC address of the dest which has been resolved from the IP address
d) Flags	:	The flags convey the status of the address, check flag types from Q4 ROUTE FLAGS
f) Interface	:	This is the name of the interface, check the convention from Q4 INTERFACE NAMING

b) Modifying ARP table : To modify the ARP table, the user must have super-user privileges

To **add** or **change** an existing entry, follow **arp -s <IP_address> <H/W Address>** ..1)

To **delete** an existing entry, follow **arp -d <IP_address> <H/W Address>**

```
root@rashis-aspire-e5-575g:/home/rashi# arp -s 10.150.39.148 ff:ff:ff:ff:ff:00
root@rashis-aspire-e5-575g:/home/rashi# arp -s 10.150.39.18 ff:ff:ff:ff:ff:00
root@rashis-aspire-e5-575g:/home/rashi# arp -s 10.150.39.16 ff:ff:ff:ff:ff:ff
root@rashis-aspire-e5-575g:/home/rashi# arp -s 10.150.39.1 ff:ff:ff:ff:ff:ff
root@rashis-aspire-e5-575g:/home/rashi# arp
Address                  HWtype  HWaddress           Flags Mask            Iface
_gateway                 ether   00:25:b4:d9:f7:c0   C                     wlp3s0
10.150.39.18             ether   ff:ff:ff:ff:ff:00   CM                    wlp3s0
10.150.39.1              ether   ff:ff:ff:ff:ff:ff   CM                    wlp3s0
10.150.39.149            ether   ff:ff:ff:ff:ff:00   CM                    wlp3s0
10.150.39.148            ether   ff:ff:ff:ff:ff:00   CM                    wlp3s0
10.150.39.16             ether   ff:ff:ff:ff:ff:ff   CM                    wlp3s0
```

arp –s <IP> <H/W Address>

The image on the left shows the four hosts that were added by command 1)

c) ARP table timeout : The exact time when an entry would be removed can not be predicted as it depends on a number of parameter :

gc_stale time : If an entry hasn't been used for these many sec, then it is marked stale (eligible for removal during garbage collection)

gc_interval : This is the time after which the garbage collector runs periodically

It is important to note that permanent entries are not deleted, but normal entries after gc_stale time are make stale (eligible for deletion) and then wait for the next gc_interval to arrive to be deleted in garbage collection process.

ALGO TO PREDICT TIMEOUT :

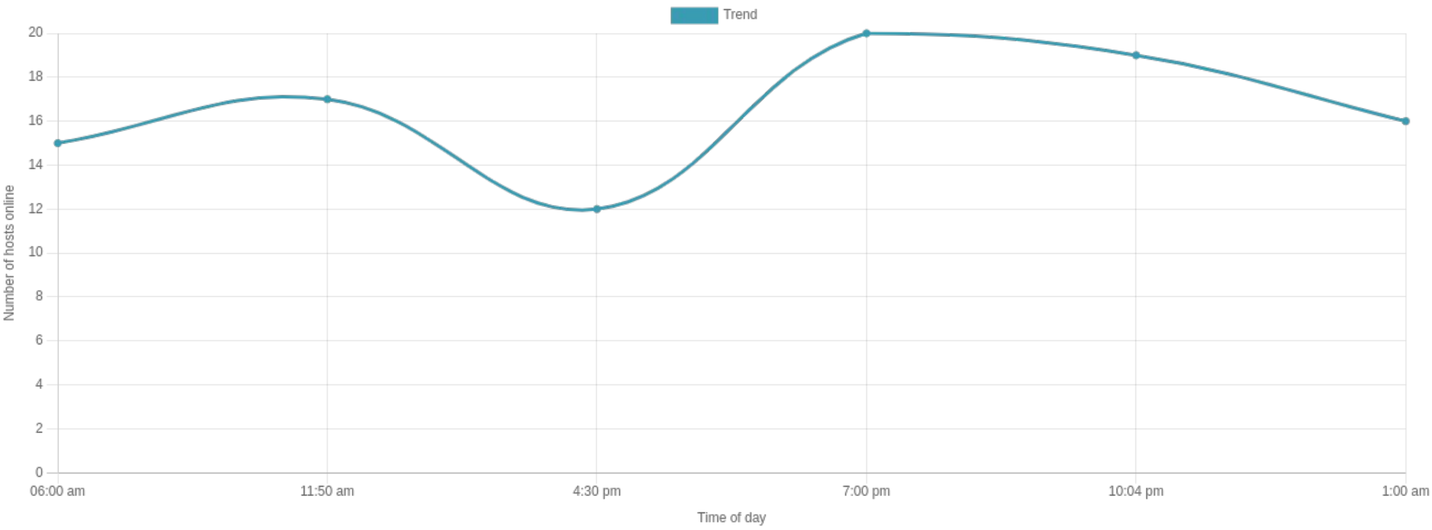
A method similar to poling can be used. For a newly added dummy entry, monitor the time after fixed intervals to see when the entry is deleted A trial and error method to discover the timeout value is to add a temporary entry in the ARP table and keep on checking its presence after fixed intervals of time (say 5 seconds). Smaller the poling time, better would be the approximation.

Alternatively, one can use binary search for finding the cache time, for e.g. – Add a temporary entry in ARP and check after 5000ms. If then entry has been deleted, then add the entry again and check after 2500ms. The more no of iterations, the better will be the approximation.

d) Two IP address mapped to same Ethernet Addr : The scenario where two IP's can map to same Ethernet Address is 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 IP's 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(router/gateway). The router then uses it's routing table and sends the packet further to the correct device.

Q7 : ARP command : nmap -n -sP <subnet>/22

Subnet used : 10.16.0.254



OBSERVATION : As observed from the image above, the number of hosts are very low during the morning and class-hours. The number of hosts online increases from 6:00 am till 11:50 am. Then there is a sharp decrease and reaches a minimum of 12 at 4:30pm (Class hours). After class hours there is an increase and reaches a maximum at around 7:00pm and then gradually decreases again. These observations clearly state when the computers are switched ON or OFF in my LAN.