Name: Jainam Jain (Roll No : 19) UID: 2018130016 (Batch B)

CEL 51, DCCN, Monsoon 2020
Lab 2: Basic Network Utilities

This lab introduces some basic network monitoring/analysis tools. There are a few exercises along the way. You should write up answers to the ***ping*** and ***traceroute*** exercises and turn them in next lab. (You should try out each tool, whether it is needed for an exercise or not!).

Prerequisite: Basic understanding of command line utilities of Linux Operating system.

Some Basic command line Networking utilities

Start with a few of the most basic command line tools. These commands are available on Unix, including Linux (and the first two, at least, are also for Windows). Some parameters or options might differ on different operating systems. Remember that you can use man <command> to get information about a command and its options.

ping — The command ping <host> sends a series of packets and expects to receieve a response to each packet. When a return packet is received, ping reports the round trip time (the time between sending the packet and receiving the response). Some routers and firewalls block ping requests, so you might get no reponse at all. Ping can be used to check whether a computer is up and running, to measure network delay time, and to check for dropped packets indicating network congestion. Note that <host> can be either a domain name or an IP address. By default, ping will send a packet every second indefinitely; stop it with Control-C

Network latency, specifically round trip time (RTT), can be measured using ping, which sends ICMP packets. The syntax for the command in Linux or Mac OS is:

```
ping [-c <count>] [-s <packetsize>] <hostname>
```

The syntax in Windows is:

```
ping [-n <count>] [-l <packetsize>] <hostname>
```

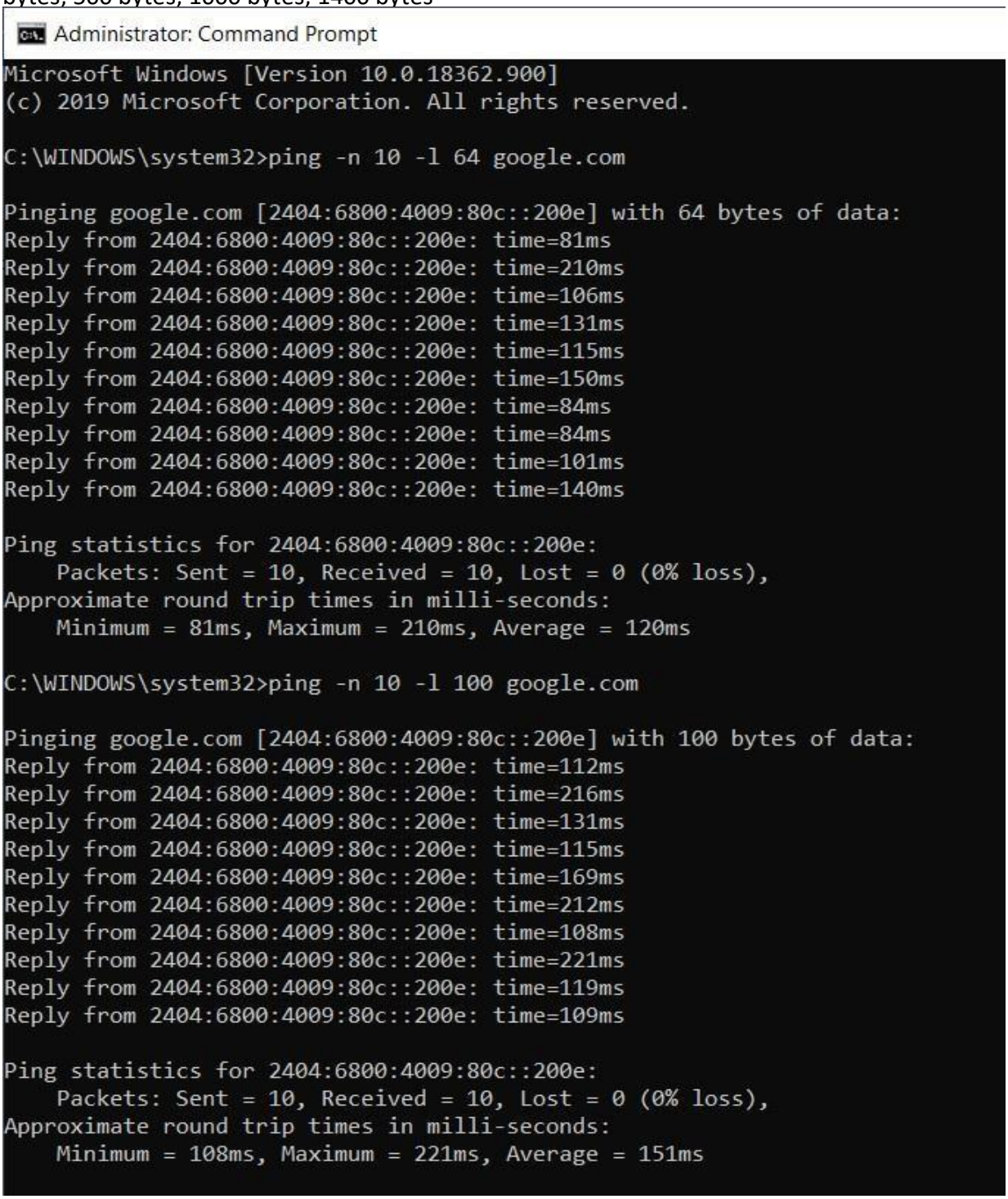
The default number of ICMP packets to send is either infinite (in Linux and Mac OS) or 4 (in Windows). The default packet size is either 64 bytes (in Linux) or 32 bytes (in Windows). You can specify either a hostname (e.g., spit.ac.in) or an IP address.

To save the output from ping to a file, include a greater than symbol and a file name at the end of the command. For example:

ping -c 10 google.com > ping_c10_s64_google.log

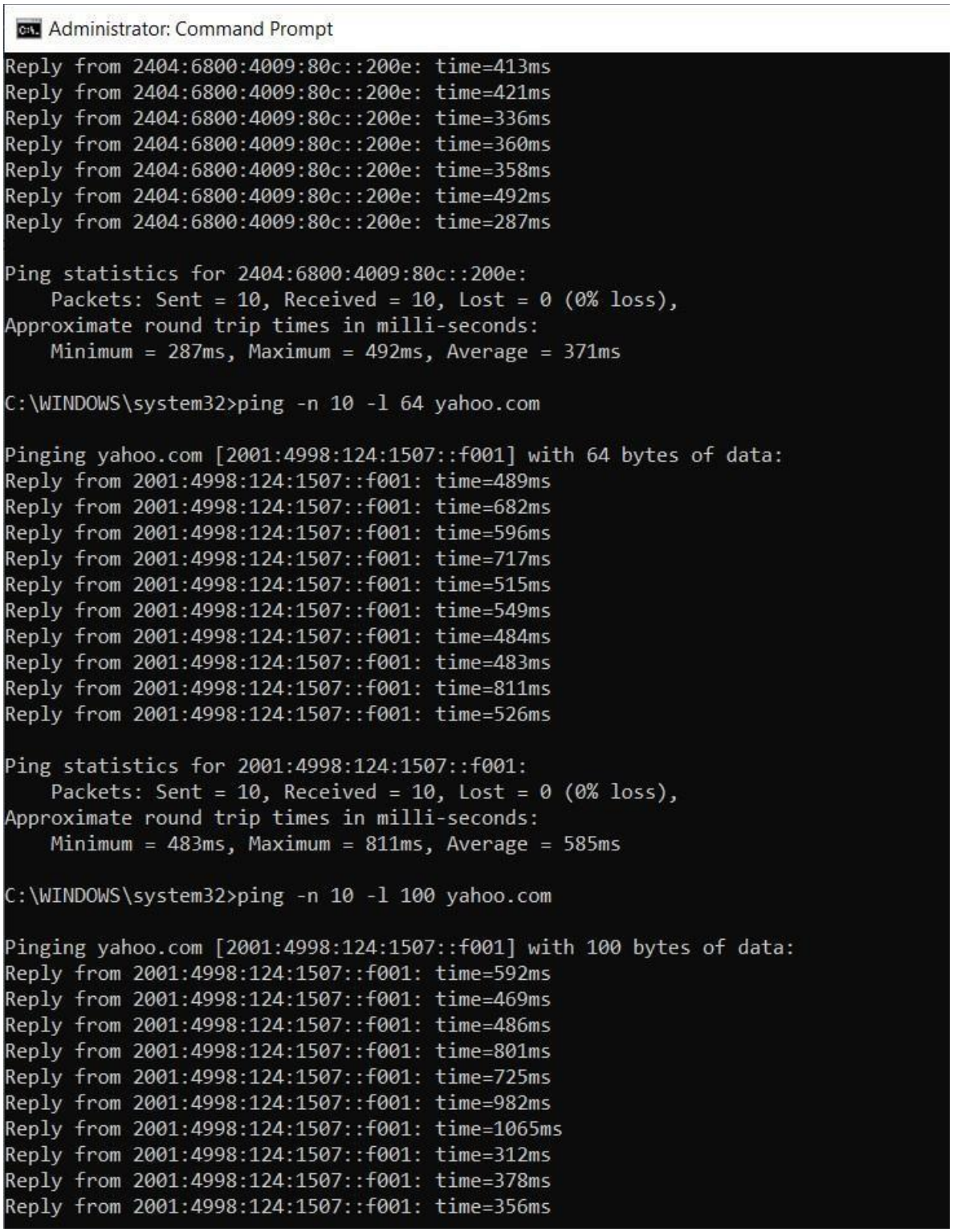
EXPERIMENTS WITH PING

1. Ping the any hosts 10 times (i.e., packet count is 10) with a packet size of 64 bytes, 100 bytes, 500 bytes, 1000 bytes, 1400 bytes



Administrator: Command Prompt

```
C:\WINDOWS\system32>ping -n 10 -l 500 google.com

Pinging google.com [2404:6800:4009:80c::200e] with 500 bytes of data:
Reply from 2404:6800:4009:80c::200e: time=248ms
Reply from 2404:6800:4009:80c::200e: time=230ms
Reply from 2404:6800:4009:80c::200e: time=305ms
Reply from 2404:6800:4009:80c::200e: time=304ms
Reply from 2404:6800:4009:80c::200e: time=220ms
Reply from 2404:6800:4009:80c::200e: time=258ms
Reply from 2404:6800:4009:80c::200e: time=256ms
Reply from 2404:6800:4009:80c::200e: time=277ms
Reply from 2404:6800:4009:80c::200e: time=213ms
Reply from 2404:6800:4009:80c::200e: time=210ms

Ping statistics for 2404:6800:4009:80c::200e:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 210ms, Maximum = 305ms, Average = 252ms

C:\WINDOWS\system32>ping -n 10 -l 1000 google.com

Pinging google.com [2404:6800:4009:80c::200e] with 1000 bytes of data:
Reply from 2404:6800:4009:80c::200e: time=617ms
Reply from 2404:6800:4009:80c::200e: time=342ms
Reply from 2404:6800:4009:80c::200e: time=328ms
Reply from 2404:6800:4009:80c::200e: time=220ms
Reply from 2404:6800:4009:80c::200e: time=395ms
Reply from 2404:6800:4009:80c::200e: time=310ms
Reply from 2404:6800:4009:80c::200e: time=203ms
Reply from 2404:6800:4009:80c::200e: time=345ms
Reply from 2404:6800:4009:80c::200e: time=568ms
Reply from 2404:6800:4009:80c::200e: time=278ms

Ping statistics for 2404:6800:4009:80c::200e:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 203ms, Maximum = 617ms, Average = 360ms

C:\WINDOWS\system32>ping -n 10 -l 1400 google.com

Pinging google.com [2404:6800:4009:80c::200e] with 1400 bytes of data:
Reply from 2404:6800:4009:80c::200e: time=296ms
Reply from 2404:6800:4009:80c::200e: time=367ms
Reply from 2404:6800:4009:80c::200e: time=386ms
Reply from 2404:6800:4009:80c::200e: time=413ms
Reply from 2404:6800:4009:80c::200e: time=421ms
Reply from 2404:6800:4009:80c::200e: time=336ms
Reply from 2404:6800:4009:80c::200e: time=360ms
Reply from 2404:6800:4009:80c::200e: time=358ms
Reply from 2404:6800:4009:80c::200e: time=492ms
Reply from 2404:6800:4009:80c::200e: time=287ms

Ping statistics for 2404:6800:4009:80c::200e:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 287ms, Maximum = 492ms, Average = 371ms
```



QUESTIONS ABOUT LATENCY

Now look at the results you gathered and answer the following questions about latency. Store your answers in a file named ping.txt.

1. Does the average RTT vary between different hosts? What aspects of latency (transmit, propagation, and queueing delay) might impact this and why?

Ans: **Round-trip time (RTT)** is the duration in milliseconds (ms) it takes for a network request to go from a starting point to a destination and back again to the starting point. RTT is an important metric in determining the health of a connection on a local network or the larger Internet, and is commonly utilized by network administrators to diagnose the speed and reliability of network connections.

Delay may differ slightly, depending on the location of the specific pair of communicating endpoints. Engineers usually report both the maximum and average delay, and they divide the delay into several parts:

- **Processing delay** – time it takes a router to process the packet header, depends on the processing speed of the switch
- **Queuing delay** – time the packet spends in routing queues depends on the number of packets, size of the packet and bandwidth
- **Transmission delay** – time it takes to push the packet's bits onto the link depends on size of the packet and the bandwidth of the network.
- **Propagation delay** – time for a signal to reach its destination depends on distance and propagation speed.

A certain minimum level of delay is experienced by signals due to the time it takes to transmit a packet serially through a link. This delay is extended by more variable levels of delay due to network congestion. IP network delays can range from a few milliseconds to several hundred milliseconds.

So yes, Average RTT does vary between different hosts due to queuing delay as we can see in above example the average RTT was calculated for google.com and yahoo.com differs. This can mostly be due to propagation Delay as it depends on distance and due to Queuing delay as the packet may be in queue

2. Does the average RTT vary with different packet sizes? What aspects of latency (transmit, propagation, and queueing delay) might impact this and why?

   Yes, the average RTT increases with packet size as Queuing delay and Transmission delay increases as they both to rely on size of packets eventually increasing the average RTT

Exercise 1: Experiment with ping to find the round trip times to a variety of destinations. Write up any interesting observations, including in particular how the round trip time compares to the physical distance. Here are few places from who to get replies: www.uw.edu, www.cornell.edu, berkeley.edu, www.uchicago.edu, www.ox.ac.uk (England), www.u-tokyo.ac.jp (Japan).

```
Pinging www.washington.edu [128.95.155.197] with 100 bytes of data:
Reply from 128.95.155.197: bytes=100 time=270ms TTL=46
Reply from 128.95.155.197: bytes=100 time=256ms TTL=46
Reply from 128.95.155.197: bytes=100 time=259ms TTL=46
Reply from 128.95.155.197: bytes=100 time=260ms TTL=46
Reply from 128.95.155.197: bytes=100 time=261ms TTL=46
Reply from 128.95.155.197: bytes=100 time=257ms TTL=46
Reply from 128.95.155.197: bytes=100 time=255ms TTL=46
Reply from 128.95.155.197: bytes=100 time=258ms TTL=46
Reply from 128.95.155.197: bytes=100 time=264ms TTL=46
Reply from 128.95.155.197: bytes=100 time=253ms TTL=46

Ping statistics for 128.95.155.197:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 253ms, Maximum = 270ms, Average = 259ms
```

```
Pinging berkeley.edu [35.163.72.93] with 100 bytes of data:
Reply from 35.163.72.93: bytes=100 time=267ms TTL=32
Reply from 35.163.72.93: bytes=100 time=268ms TTL=32
Reply from 35.163.72.93: bytes=100 time=270ms TTL=32
Reply from 35.163.72.93: bytes=100 time=278ms TTL=32
Reply from 35.163.72.93: bytes=100 time=272ms TTL=32
Reply from 35.163.72.93: bytes=100 time=265ms TTL=32
Reply from 35.163.72.93: bytes=100 time=279ms TTL=32
Reply from 35.163.72.93: bytes=100 time=265ms TTL=32
Reply from 35.163.72.93: bytes=100 time=264ms TTL=32
Reply from 35.163.72.93: bytes=100 time=271ms TTL=32

Ping statistics for 35.163.72.93:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 264ms, Maximum = 279ms, Average = 269ms
```

List of factors affecting RTT:

The nature of the transmission medium - the way in which connections are made affects how fast the connection moves; connections made over optical fiber will behave

differently than connections made over copper. Likewise, a connection made over a wireless frequency will behave differently than that of a satellite communication.

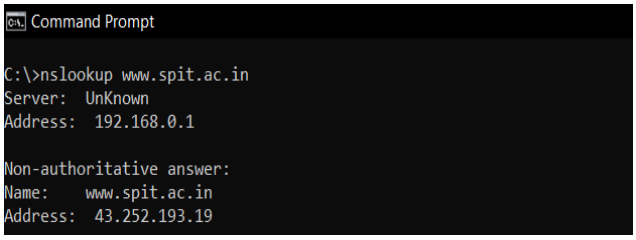
Local area network (LAN) traffic - the amount of traffic on the local area network can bottleneck a connection before it ever reaches the larger Internet. For example, if many users are using streaming video service simultaneously, round-trip time may be inhibited even though the external network has excess capacity and is functioning normally.

Server response time – the amount of time it takes a server to process and respond to a request is a potential bottleneck in network latency. When a server is overwhelmed with requests, such as during a DDoS attack, its ability to respond efficiently can be inhibited, resulting in increased RTT.

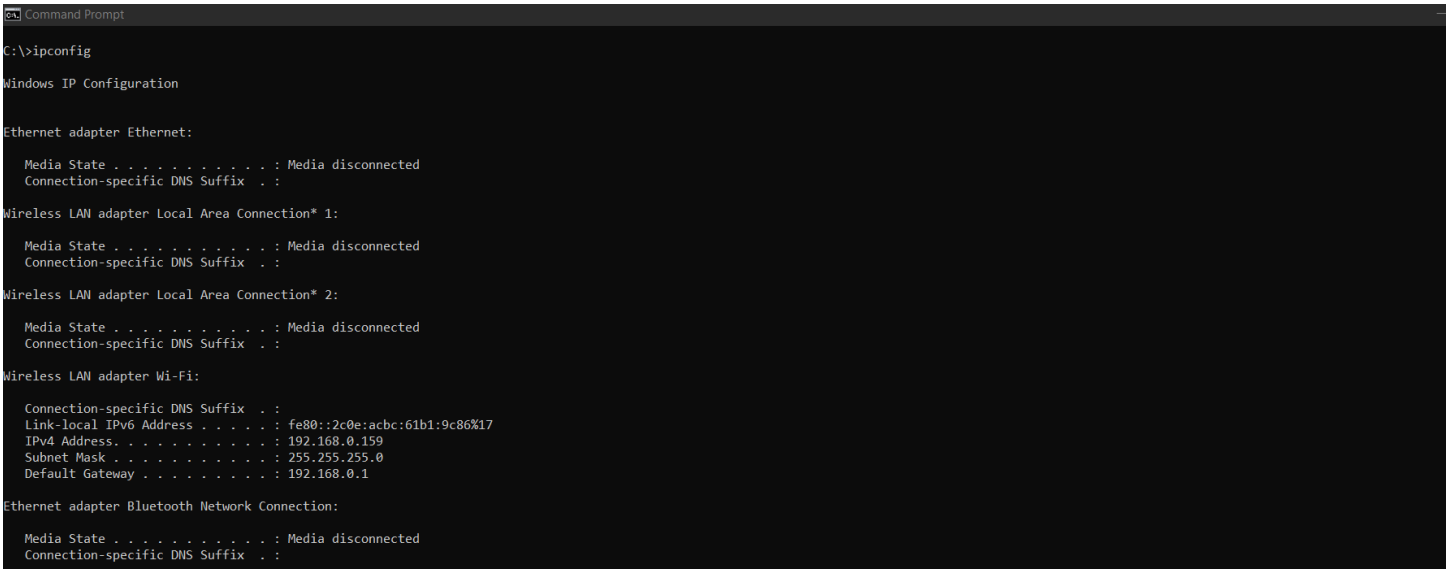
Node count and congestion – depending on the path that a connection takes across the Internet, it may be routed or “hop” through a different number of intermediate nodes. Generally speaking, the greater the number of nodes a connection touches the slower it will be. A node may also experience network congestion from other network traffic, which will slow down the connection and increase RTT.

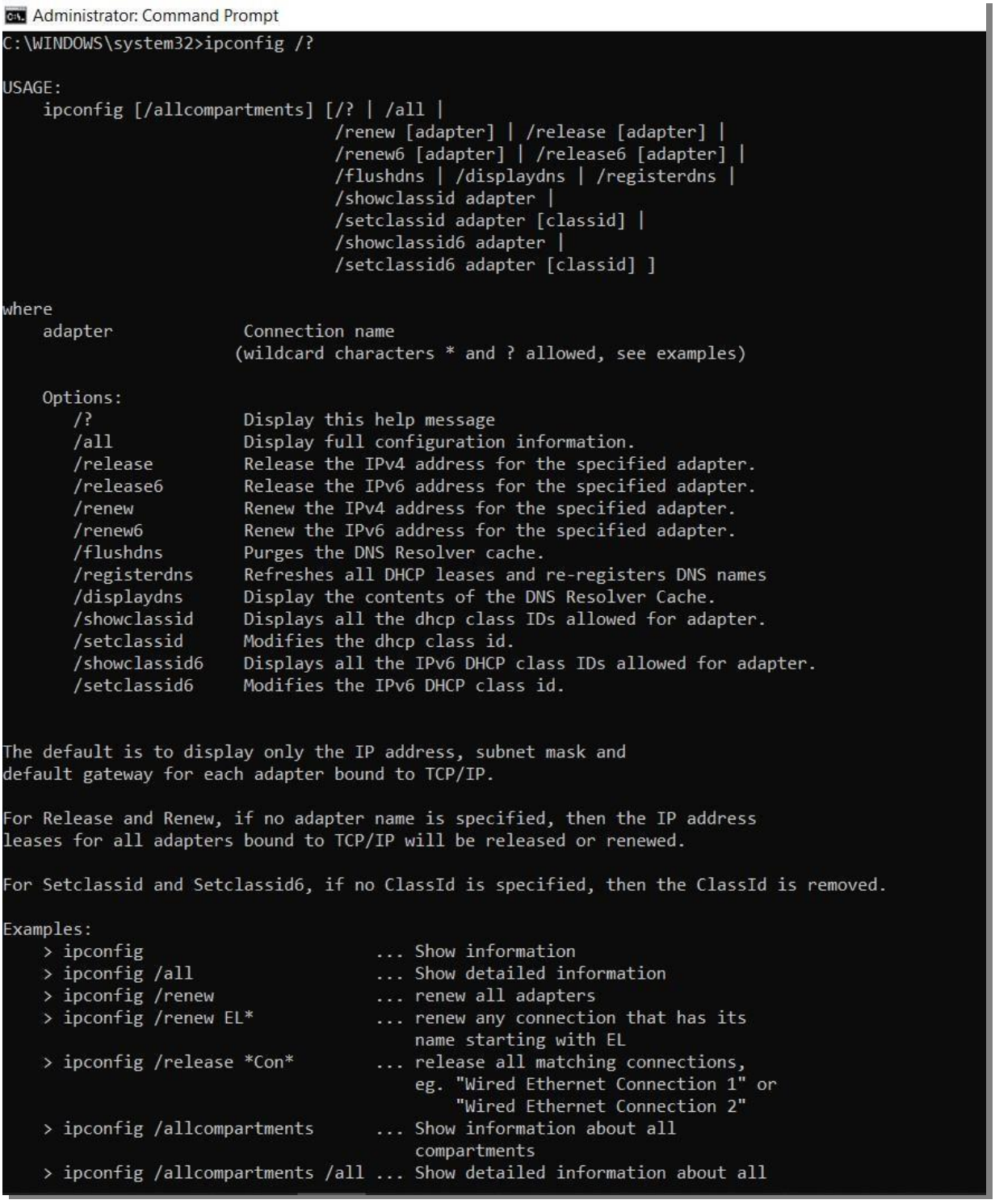
Physical distance – although a connection optimized by a CDN can often reduce the number of hops required to reach a destination, there is no way of getting around the limitation imposed by the speed of light; the distance between a start and end point is a limiting factor in network connectivity that can only be reduced by moving content closer to the requesting users. To overcome this obstacle, a CDN will cache content closer to the requesting users, thereby reducing RTT.

nslookup — The command nslookup <host> will do a DNS query to find and report the IP address (or addresses) for a domain name or the domain name corresponding to an IP address. To do this, it contacts a "DNS server." Default DNS servers are part of a computer's network configuration. (For a static IP address in Linux, they are configured in the file /etc/network/interfaces that you encountered in the last lab.) You can specify a different DNS server to be used by nslokup by adding the server name or IP address to the command:
nslookup <host> <server>



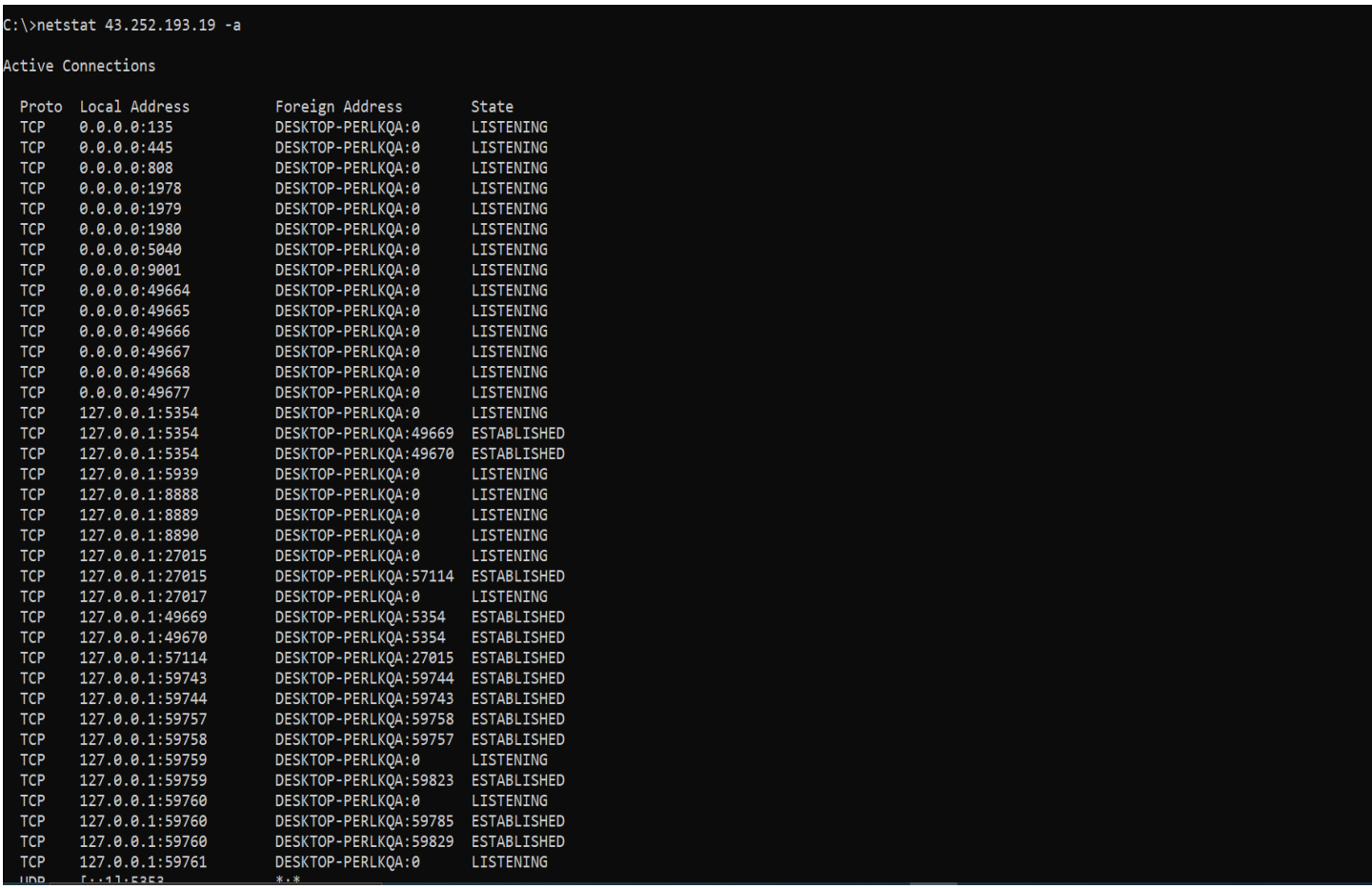
ipconfig — You used ifconfig in the previous lab. When used with no parameters, ifconfig reports some information about the computer's network interfaces. This usually includes lo which stands for localhost; it can be used for communication between programs running on the same computer. Linux often has an interface named eth0, which is the first ethernet card. The information is different on Mac OS and Linux, but includes the IP or "inet" address and ethernet or "hardware" address for an ethernet card. On Linux, you get the number of packets received (RX) and sent (TX), as well as the number of bytes transmitted and received. (A better place to monitor network bytes on our Linux computers is in the GUI program System Monitor, if it is installed!!!.)



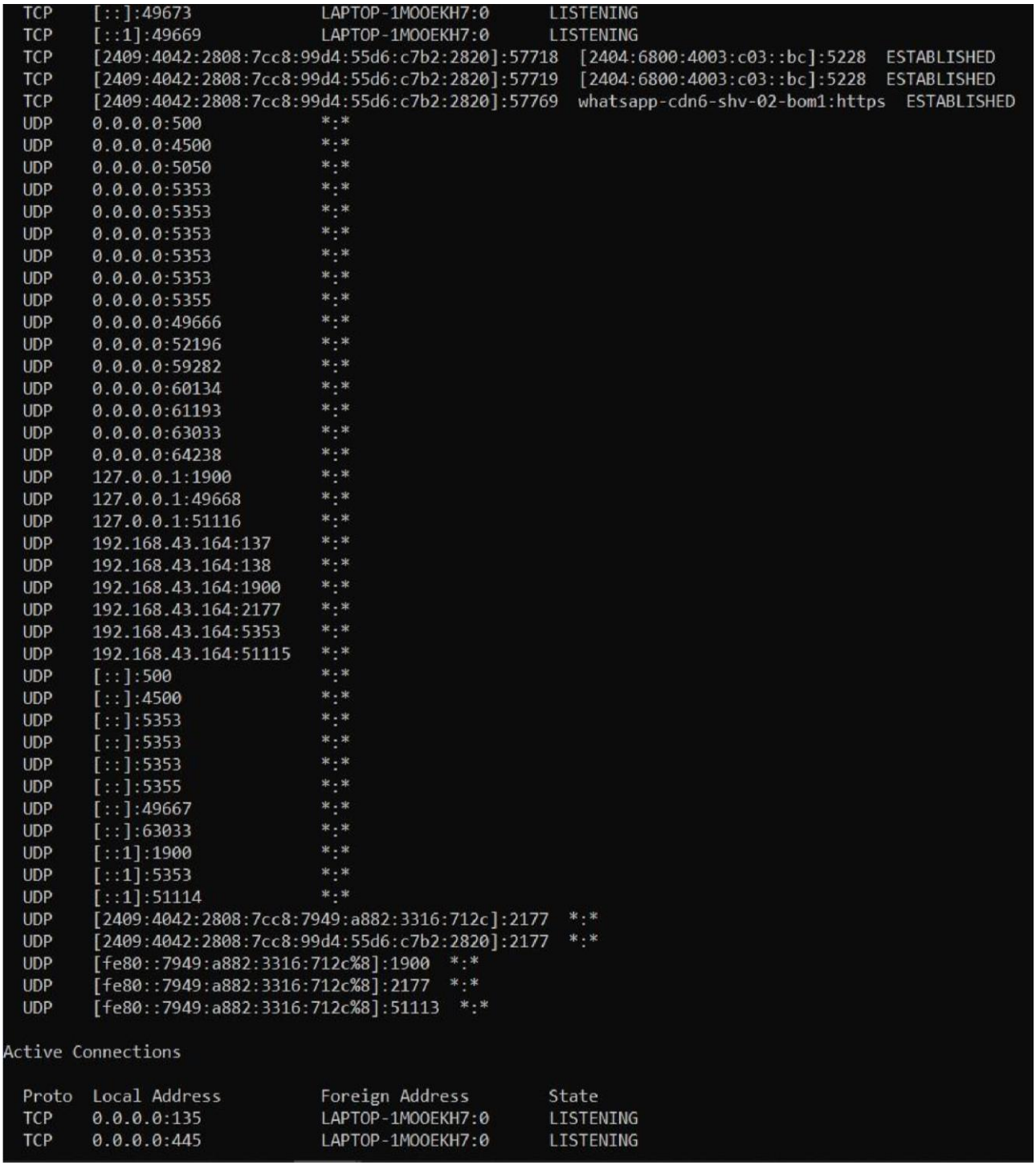


netstat — The netstat command gives information about network connections. I often use netstat -t -n which lists currently open TCP connections (that's the "-t" option) by IP address rather than domain name (that's the "-n" option). Add the option "-l" (lower case ell) to list listening sockets, that is sockets that have been opened by server programs to wait for connection requests from

clients: netstat -t -n -l. (On Mac, use netstat -p tcp to list tcp connections, and add "-a" to include listening sockets in the list.



```
C:\>netstat 43.252.193.19 -a

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:445            DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:808            DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:1978           DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:1979           DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:1980           DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:5040           DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:9001           DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:49664          DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:49665          DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:49666          DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:49667          DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:49668          DESKTOP-PERLKQA:0      LISTENING
  TCP    0.0.0.0:49677          DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:5354         DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:5354         DESKTOP-PERLKQA:49669  ESTABLISHED
  TCP    127.0.0.1:5354         DESKTOP-PERLKQA:49670  ESTABLISHED
  TCP    127.0.0.1:5939         DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:8888         DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:8889         DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:8890         DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:27015        DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:27015        DESKTOP-PERLKQA:57114  ESTABLISHED
  TCP    127.0.0.1:27017        DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:49669        DESKTOP-PERLKQA:5354   ESTABLISHED
  TCP    127.0.0.1:49670        DESKTOP-PERLKQA:5354   ESTABLISHED
  TCP    127.0.0.1:57114        DESKTOP-PERLKQA:27015  ESTABLISHED
  TCP    127.0.0.1:59743        DESKTOP-PERLKQA:59744  ESTABLISHED
  TCP    127.0.0.1:59744        DESKTOP-PERLKQA:59743  ESTABLISHED
  TCP    127.0.0.1:59757        DESKTOP-PERLKQA:59758  ESTABLISHED
  TCP    127.0.0.1:59758        DESKTOP-PERLKQA:59757  ESTABLISHED
  TCP    127.0.0.1:59759        DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:59759        DESKTOP-PERLKQA:59823  ESTABLISHED
  TCP    127.0.0.1:59760        DESKTOP-PERLKQA:0      LISTENING
  TCP    127.0.0.1:59760        DESKTOP-PERLKQA:59785  ESTABLISHED
  TCP    127.0.0.1:59760        DESKTOP-PERLKQA:59829  ESTABLISHED
  TCP    127.0.0.1:59761        DESKTOP-PERLKQA:0      LISTENING
  UDP    [::1]:5353             *:*
```



```
  TCP    [::]:49673             LAPTOP-1MOOEKH7:0      LISTENING
  TCP    [::1]:49669            LAPTOP-1MOOEKH7:0      LISTENING
  TCP    [2409:4042:2808:7cc8:99d4:55d6:c7b2:2820]:57718  [2404:6800:4003:c03::bc]:5228  ESTABLISHED
  TCP    [2409:4042:2808:7cc8:99d4:55d6:c7b2:2820]:57719  [2404:6800:4003:c03::bc]:5228  ESTABLISHED
  TCP    [2409:4042:2808:7cc8:99d4:55d6:c7b2:2820]:57769  whatsapp-cdn6-shv-02-bom1:https  ESTABLISHED
  UDP    0.0.0.0:500            *:*
  UDP    0.0.0.0:4500           *:*
  UDP    0.0.0.0:5050           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5355           *:*
  UDP    0.0.0.0:49666          *:*
  UDP    0.0.0.0:52196          *:*
  UDP    0.0.0.0:59282          *:*
  UDP    0.0.0.0:60134          *:*
  UDP    0.0.0.0:61193          *:*
  UDP    0.0.0.0:63033          *:*
  UDP    0.0.0.0:64238          *:*
  UDP    127.0.0.1:1900         *:*
  UDP    127.0.0.1:49668        *:*
  UDP    127.0.0.1:51116        *:*
  UDP    192.168.43.164:137     *:*
  UDP    192.168.43.164:138     *:*
  UDP    192.168.43.164:1900    *:*
  UDP    192.168.43.164:2177    *:*
  UDP    192.168.43.164:5353    *:*
  UDP    192.168.43.164:51115   *:*
  UDP    [::]:500               *:*
  UDP    [::]:4500              *:*
  UDP    [::]:5353              *:*
  UDP    [::]:5353              *:*
  UDP    [::]:5353              *:*
  UDP    [::]:5355              *:*
  UDP    [::]:49667             *:*
  UDP    [::]:63033             *:*
  UDP    [::1]:1900             *:*
  UDP    [::1]:5353             *:*
  UDP    [::1]:51114            *:*
  UDP    [2409:4042:2808:7cc8:7949:a882:3316:712c]:2177  *:*
  UDP    [2409:4042:2808:7cc8:99d4:55d6:c7b2:2820]:2177  *:*
  UDP    [fe80::7949:a882:3316:712c%8]:1900  *:*
  UDP    [fe80::7949:a882:3316:712c%8]:2177  *:*
  UDP    [fe80::7949:a882:3316:712c%8]:51113  *:*

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            LAPTOP-1MOOEKH7:0      LISTENING
  TCP    0.0.0.0:445            LAPTOP-1MOOEKH7:0      LISTENING
```

telnet — Telnet is an old program for remote login. It's not used so much for that any more, since it has no security features. But basically, all it does is open a connection to a server and allow server and client to send lines of plain text to each other. It can be used to check that it's possible to connect to a server and, if the server communicates in plain text, even to interact with the server by hand. Since the Web uses a plain text protocol, you can use telnet to connect to a web client and play the part of the web browser. I will suggest that you to do this with your own web server when you write it, but you might want to try it now. When you use telnet in this way, you need to specify both the

host and the port number to which you want to connect: telent <host> <port>. For example, to connect to the web server on www.spit.ac.in:
telnet spit.ac.in 80

A blank command prompt screen appears showing that the connection is established.

traceroute — Traceroute is discussed in man utility. The command traceroute <host> will show routers encountered by packets on their way from your computer to a specified <host>. For each n = 1, 2, 3,..., traceroute sends a packet with "time-to-live" (ttl) equal to n. Every time a router forwards a packet, it decreases the ttl of the packet by one. If the ttl drops to zero, the router discards the packet and sends an error message back to the sender of the packet. (Again, as with ping, the packets might be blocked or might not even be sent, so that the error messages will never be received.) The sender gets the identity of the router from the source of the error message. Traceroute will send packets until n reaches some set upper bound or until a packet actually gets through to the destination. It actually does this three times for each n. In this way, it identifies routers that are one step, two steps, three steps, ... away from the source computer. A packet for which no response is received is indicated in the output as a *.

Traceroute is installed on the computers. If was not installed in your virtual server last week, but you can install it with the command sudo apt-get install traceroute

The path taken through a network, can be measured using traceroute. The syntax for the command in Linux is:

```
traceroute <hostname>
```

The syntax in Windows is:

```
tracert <hostname>
```

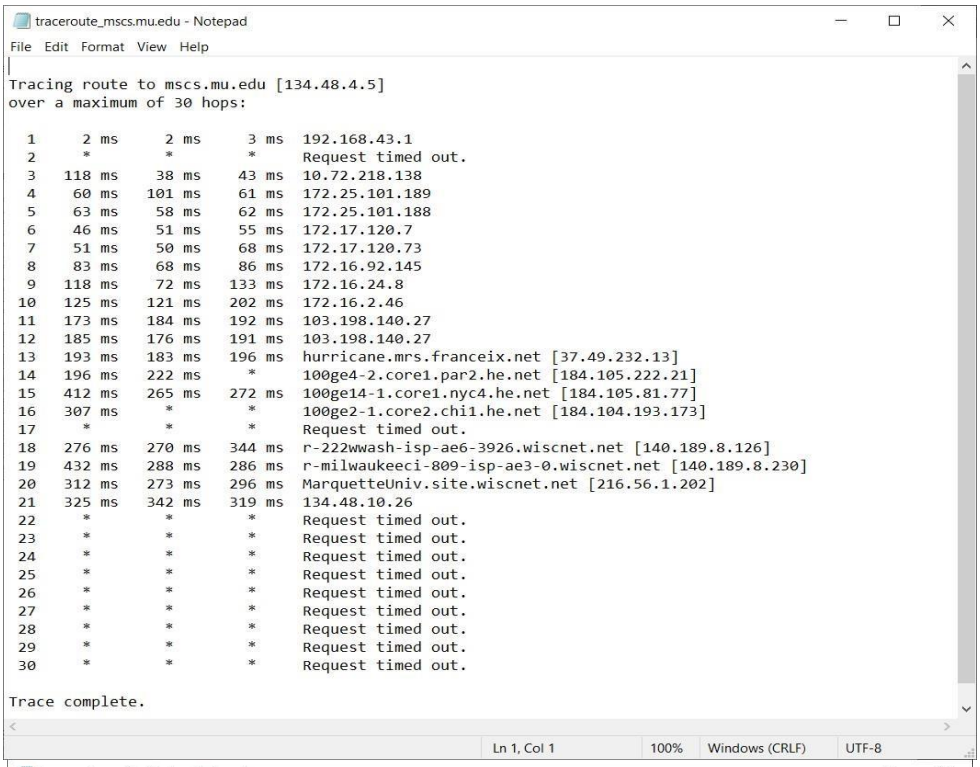
You can specify either a hostname (e.g., cs.iitb.ac.in) or an IP address (e.g., 128.105.2.6).

1.2.1 EXPERIMENTS WITH TRACEROUTE

From **your machine** traceroute to the following hosts:

1. ee.iitb.ac.in
2. mscs.mu.edu
3. www.cs.grinnell.edu
4. csail.mit.edu
5. cs.stanford.edu
6. cs.manchester.ac.uk

Store the output of each traceroute command in a separate file named traceroute_HOSTNAME.log, replacing HOSTNAME with the hostname for end-host you pinged (e.g., traceroute_ee.iitb.ac.in.log).

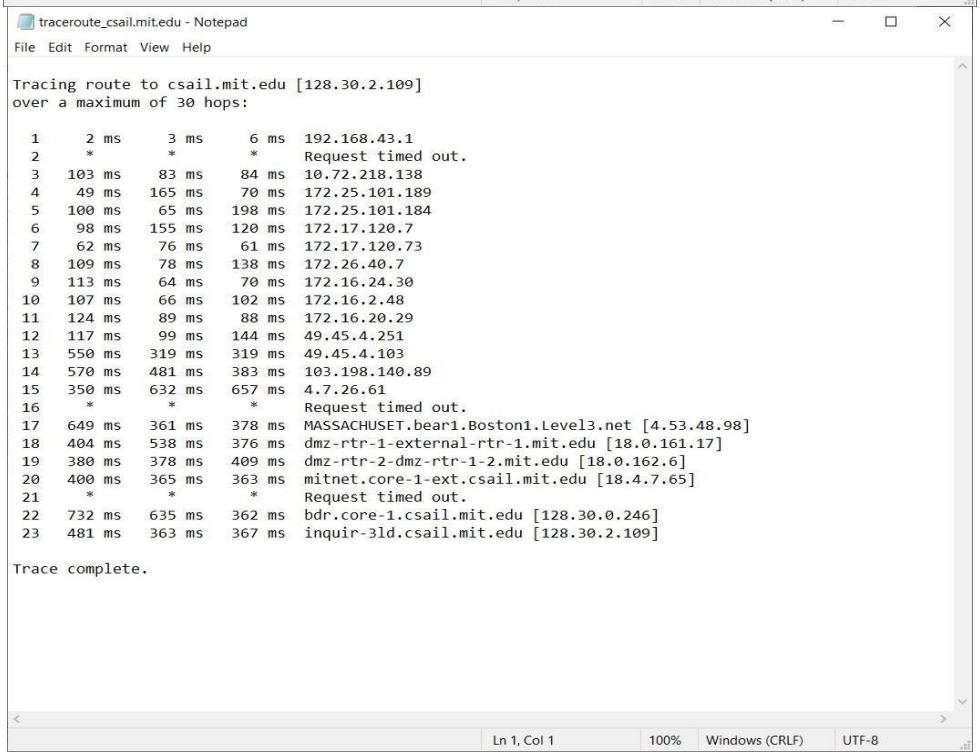


traceroute_mscs.mu.edu - Notepad

File Edit Format View Help

```
Tracing route to mscs.mu.edu [134.48.4.5]
over a maximum of 30 hops:

  1     2 ms     2 ms     3 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3   118 ms    38 ms    43 ms  10.72.218.138
  4    60 ms   101 ms    61 ms  172.25.101.189
  5    63 ms    58 ms    62 ms  172.25.101.188
  6    46 ms    51 ms    55 ms  172.17.120.7
  7    51 ms    50 ms    68 ms  172.17.120.73
  8    83 ms    68 ms    86 ms  172.16.92.145
  9   118 ms    72 ms   133 ms  172.16.24.8
 10   125 ms   121 ms   202 ms  172.16.2.46
 11   173 ms   184 ms   192 ms  103.198.140.27
 12   185 ms   176 ms   191 ms  103.198.140.27
 13   193 ms   183 ms   196 ms  hurricane.mrs.franceix.net [37.49.232.13]
 14   196 ms   222 ms     *     100ge4-2.core1.par2.he.net [184.105.222.21]
 15   412 ms   265 ms   272 ms  100ge14-1.core1.nyc4.he.net [184.105.81.77]
 16   307 ms     *        *     100ge2-1.core2.chi1.he.net [184.104.193.173]
 17     *        *        *     Request timed out.
 18   276 ms   270 ms   344 ms  r-222wwash-isp-ae6-3926.wiscnet.net [140.189.8.126]
 19   432 ms   288 ms   286 ms  r-milwaukeeci-809-isp-ae3-0.wiscnet.net [140.189.8.230]
 20   312 ms   273 ms   296 ms  MarquetteUniv.site.wiscnet.net [216.56.1.202]
 21   325 ms   342 ms   319 ms  134.48.10.26
 22     *        *        *     Request timed out.
 23     *        *        *     Request timed out.
 24     *        *        *     Request timed out.
 25     *        *        *     Request timed out.
 26     *        *        *     Request timed out.
 27     *        *        *     Request timed out.
 28     *        *        *     Request timed out.
 29     *        *        *     Request timed out.
 30     *        *        *     Request timed out.

Trace complete.
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8



traceroute_csail.mit.edu - Notepad

File Edit Format View Help

```
Tracing route to csail.mit.edu [128.30.2.109]
over a maximum of 30 hops:

  1     2 ms     3 ms     6 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3   103 ms    83 ms    84 ms  10.72.218.138
  4    49 ms   165 ms    70 ms  172.25.101.189
  5   100 ms    65 ms   198 ms  172.25.101.184
  6    98 ms   155 ms   120 ms  172.17.120.7
  7    62 ms    76 ms    61 ms  172.17.120.73
  8   109 ms    78 ms   138 ms  172.26.40.7
  9   113 ms    64 ms    70 ms  172.16.24.30
 10   107 ms    66 ms   102 ms  172.16.2.48
 11   124 ms    89 ms    88 ms  172.16.20.29
 12   117 ms    99 ms   144 ms  49.45.4.251
 13   550 ms   319 ms   319 ms  49.45.4.103
 14   570 ms   481 ms   383 ms  103.198.140.89
 15   350 ms   632 ms   657 ms  4.7.26.61
 16     *        *        *     Request timed out.
 17   649 ms   361 ms   378 ms  MASSACHUSET.bear1.Boston1.Level3.net [4.53.48.98]
 18   404 ms   538 ms   376 ms  dmz-rtr-1-external-rtr-1.mit.edu [18.0.161.17]
 19   380 ms   378 ms   409 ms  dmz-rtr-2-dmz-rtr-1-2.mit.edu [18.0.162.6]
 20   400 ms   365 ms   363 ms  mitnet.core-1-ext.csail.mit.edu [18.4.7.65]
 21     *        *        *     Request timed out.
 22   732 ms   635 ms   362 ms  bdr.core-1.csail.mit.edu [128.30.0.246]
 23   481 ms   363 ms   367 ms  inquir-3ld.csail.mit.edu [128.30.2.109]

Trace complete.
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

traceroute_cs.stanford.edu - Notepad

File Edit Format View Help

```
Tracing route to cs.stanford.edu [171.64.64.64]
over a maximum of 30 hops:

  1     3 ms     3 ms     2 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3    47 ms    58 ms    52 ms  10.72.216.10
  4    56 ms    60 ms   104 ms  172.25.101.189
  5    55 ms    53 ms    73 ms  172.25.101.184
  6   102 ms    74 ms    49 ms  172.17.120.7
  7    53 ms    44 ms    63 ms  172.17.120.77
  8    74 ms    72 ms    63 ms  172.16.92.145
  9    88 ms   266 ms    67 ms  172.16.24.10
 10    65 ms    86 ms    78 ms  172.16.2.46
 11   174 ms   168 ms   161 ms  103.198.140.56
 12   182 ms   169 ms   180 ms  103.198.140.56
 13   178 ms   168 ms   165 ms  hurricane.mrs.franceix.net [37.49.232.13]
 14   209 ms   181 ms   185 ms  100ge4-2.core1.par2.he.net [184.105.222.21]
 15   260 ms   305 ms   289 ms  100ge10-2.core1.ash1.he.net [184.105.213.173]
 16   329 ms   307 ms   309 ms  100ge7-2.core1.pao1.he.net [184.105.222.41]
 17   300 ms   334 ms   293 ms  stanford-university.100gigabitethernet5-1.core1.pao1.he.net [184.105.17
 18   313 ms   295 ms   517 ms  csee-west-rtr-vl3.SUNet [171.66.255.140]
 19   309 ms   304 ms   316 ms  CS.stanford.edu [171.64.64.64]

Trace complete.
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

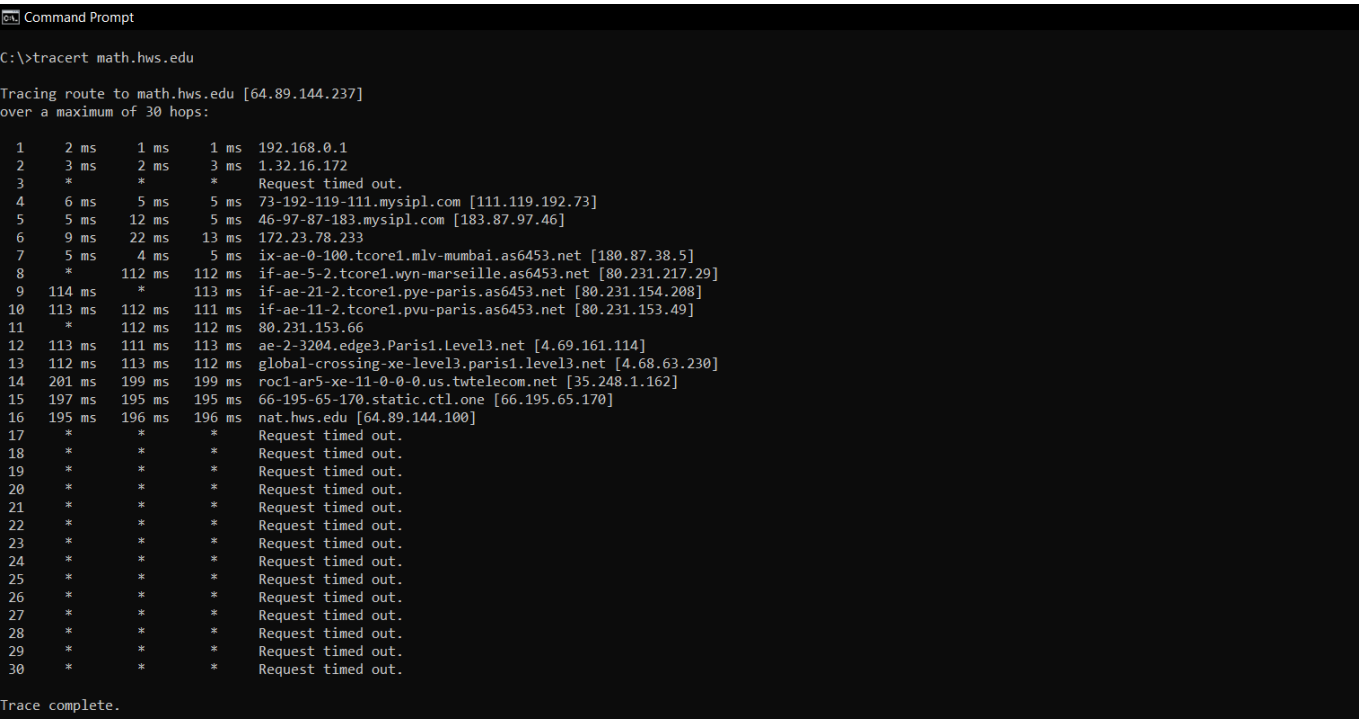
traceroute_cs.manchester.ac.uk - Notepad

File Edit Format View Help

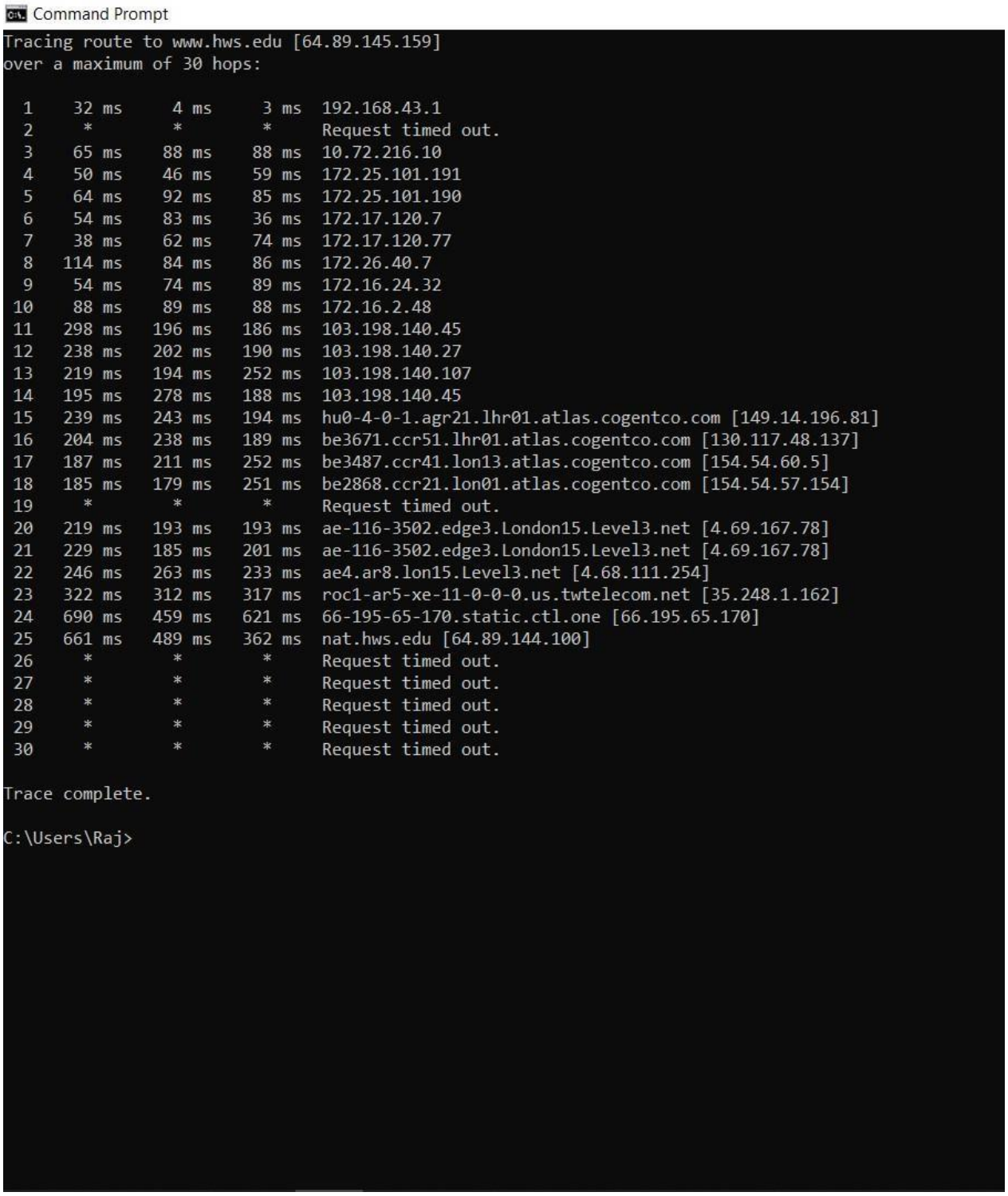
```
Tracing route to cs.manchester.ac.uk [130.88.101.49]
over a maximum of 30 hops:

  1     2 ms     3 ms     3 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3   108 ms   437 ms   100 ms  10.72.218.138
  4    74 ms    79 ms    84 ms  172.25.101.189
  5   103 ms   104 ms   678 ms  172.25.101.184
  6   106 ms    84 ms    85 ms  172.17.120.7
  7   163 ms    80 ms    83 ms  172.17.120.77
  8   635 ms   373 ms   636 ms  172.26.40.5
  9    65 ms    70 ms    76 ms  172.16.24.10
 10    95 ms    91 ms    82 ms  172.16.2.46
 11   228 ms   217 ms   207 ms  103.198.140.45
 12   206 ms   255 ms   588 ms  103.198.140.56
 13   201 ms   208 ms   210 ms  103.198.140.107
 14   239 ms   194 ms   224 ms  103.198.140.45
 15   232 ms   208 ms   251 ms  hu0-4-0-1.agr21.lhr01.atlas.cogentco.com [149.14.196.81]
 16   221 ms   206 ms   216 ms  be3672.ccr52.lhr01.atlas.cogentco.com [130.117.48.145]
 17   212 ms   226 ms   214 ms  be3488.ccr42.lon13.atlas.cogentco.com [154.54.60.13]
 18   212 ms   196 ms   222 ms  be2871.ccr21.lon01.atlas.cogentco.com [154.54.58.186]
 19   228 ms   286 ms   230 ms  ldn-b1-link.telia.net [62.115.9.28]
 20   197 ms   209 ms   254 ms  ldn-bb3-link.telia.net [62.115.120.74]
 21   363 ms   211 ms     *     ldn-b2-link.telia.net [62.115.122.189]
 22   199 ms   269 ms   203 ms  jisc-ic-345131-ldn-b4.c.telia.net [62.115.175.131]
 23   210 ms   197 ms   226 ms  ae24.londhx-sbr1.ja.net [146.97.35.197]
 24   224 ms   195 ms   392 ms  ae29.londpg-sbr2.ja.net [146.97.33.2]
 25   224 ms   324 ms   251 ms  ae31.erdiss-sbr2.ja.net [146.97.33.22]
 26   182 ms   230 ms   221 ms  ae29.manckh-sbr2.ja.net [146.97.33.42]
 27   263 ms   252 ms   429 ms  ae23.mancrh-rbr1.ja.net [146.97.38.42]
 28     *        *      205 ms  universityofmanchester.ja.net [146.97.169.2]
 29   343 ms   208 ms   248 ms  130.88.249.194
 30     *        *        *     Request timed out.

Trace complete.
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

Exercise 2: (Very short.) Use traceroute to trace the route from your computer to math.hws.edu and to www.hws.edu. Explain the difference in the results.

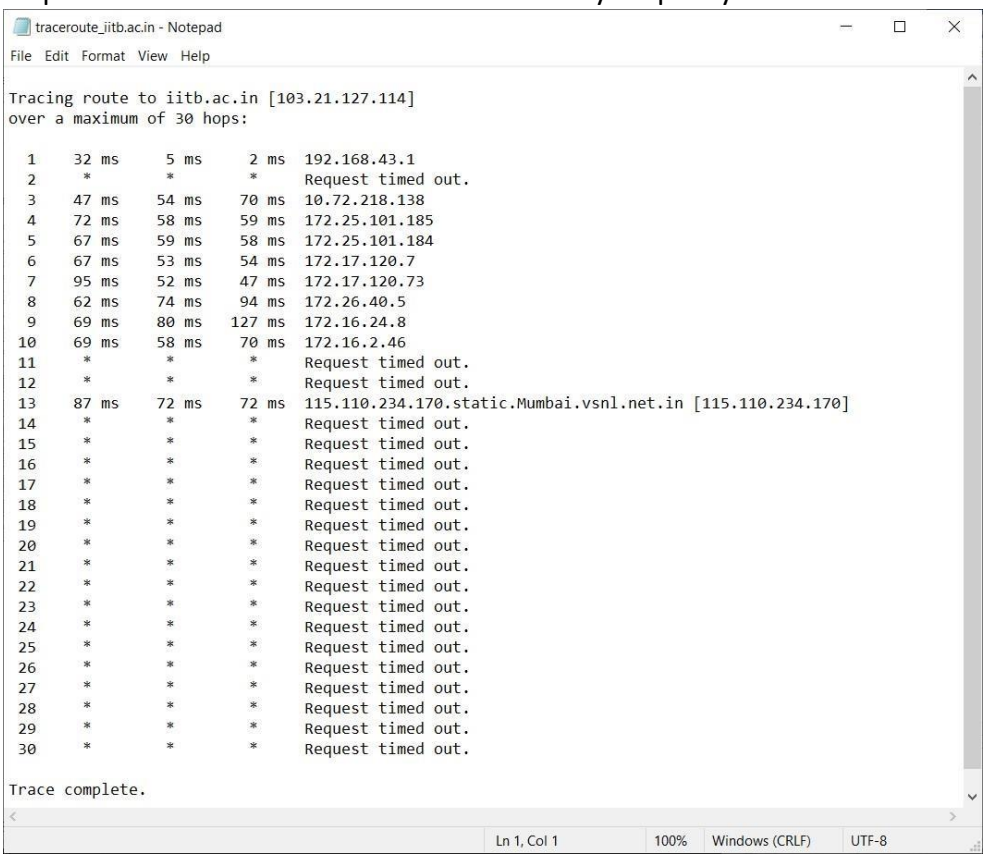


```
Command Prompt

C:\>tracert math.hws.edu

Tracing route to math.hws.edu [64.89.144.237]
over a maximum of 30 hops:

  1     2 ms     1 ms     1 ms  192.168.0.1
  2     3 ms     2 ms     3 ms  1.32.16.172
  3     *        *        *     Request timed out.
  4     6 ms     5 ms     5 ms  73-192-119-111.mysipl.com [111.119.192.73]
  5     5 ms    12 ms     5 ms  46-97-87-183.mysipl.com [183.87.97.46]
  6     9 ms    22 ms    13 ms  172.23.78.233
  7     5 ms     4 ms     5 ms  ix-ae-0-100.tcore1.mlv-mumbai.as6453.net [180.87.38.5]
  8     *      112 ms   112 ms  if-ae-5-2.tcore1.wyn-marseille.as6453.net [80.231.217.29]
  9   114 ms     *      113 ms  if-ae-21-2.tcore1.pye-paris.as6453.net [80.231.154.208]
 10   113 ms   112 ms   111 ms  if-ae-11-2.tcore1.pvu-paris.as6453.net [80.231.153.49]
 11     *      112 ms   112 ms  80.231.153.66
 12   113 ms   111 ms   113 ms  ae-2-3204.edge3.Paris1.Level3.net [4.69.161.114]
 13   112 ms   113 ms   112 ms  global-crossing-xe-level3.paris1.level3.net [4.68.63.230]
 14   201 ms   199 ms   199 ms  roc1-ar5-xe-11-0-0-0.us.twtelecom.net [35.248.1.162]
 15   197 ms   195 ms   195 ms  66-195-65-170.static.ctl.one [66.195.65.170]
 16   195 ms   196 ms   196 ms  nat.hws.edu [64.89.144.100]
 17     *        *        *     Request timed out.
 18     *        *        *     Request timed out.
 19     *        *        *     Request timed out.
 20     *        *        *     Request timed out.
 21     *        *        *     Request timed out.
 22     *        *        *     Request timed out.
 23     *        *        *     Request timed out.
 24     *        *        *     Request timed out.
 25     *        *        *     Request timed out.
 26     *        *        *     Request timed out.
 27     *        *        *     Request timed out.
 28     *        *        *     Request timed out.
 29     *        *        *     Request timed out.
 30     *        *        *     Request timed out.

Trace complete.
```



Even if both the addresses are hosted on the same servers the response time that is the round trip time is different for both.

Exercise 3: Two packets sent from the same source to the same destination do not necessarily follow the same path through the net. Experiment with some sources that are fairly far away. Can you find cases where packets sent to the same destination follow different paths? How likely does it seem to be? What about when the packets are sent at very different times? Save some of the outputs from traceroute. (You can copy them from the Terminal window by highlighting and right-clicking, then paste into a text editor.) Come back sometime next week, try the same destinations again, and compare the results with the results from today. Report your observations.


traceroute_iitb.ac.in - Notepad

File Edit Format View Help

```
Tracing route to iitb.ac.in [103.21.127.114]
over a maximum of 30 hops:

  1    32 ms     5 ms     2 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3    47 ms    54 ms    70 ms  10.72.218.138
  4    72 ms    58 ms    59 ms  172.25.101.185
  5    67 ms    59 ms    58 ms  172.25.101.184
  6    67 ms    53 ms    54 ms  172.17.120.7
  7    95 ms    52 ms    47 ms  172.17.120.73
  8    62 ms    74 ms    94 ms  172.26.40.5
  9    69 ms    80 ms   127 ms  172.16.24.8
 10    69 ms    58 ms    70 ms  172.16.2.46
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
 13    87 ms    72 ms    72 ms  115.110.234.170.static.Mumbai.vsnl.net.in [115.110.234.170]
 14     *        *        *     Request timed out.
 15     *        *        *     Request timed out.
 16     *        *        *     Request timed out.
 17     *        *        *     Request timed out.
 18     *        *        *     Request timed out.
 19     *        *        *     Request timed out.
 20     *        *        *     Request timed out.
 21     *        *        *     Request timed out.
 22     *        *        *     Request timed out.
 23     *        *        *     Request timed out.
 24     *        *        *     Request timed out.
 25     *        *        *     Request timed out.
 26     *        *        *     Request timed out.
 27     *        *        *     Request timed out.
 28     *        *        *     Request timed out.
 29     *        *        *     Request timed out.
 30     *        *        *     Request timed out.

Trace complete.
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

```
C:\Users\Raj>tracert iitb.ac.in

Tracing route to iitb.ac.in [103.21.127.114]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  192.168.0.1
  2     1 ms     1 ms     1 ms  103.135.6.146
  3     *        *        *     Request timed out.
  4     *        *       17 ms  175.100.177.221
  5    10 ms    27 ms     6 ms  172.16.2.101
  6    10 ms     4 ms    17 ms  121.241.42.57.static-mumbai.vsnl.net.in [121.241.43.57]
  7     5 ms    17 ms     6 ms  172.23.78.237
  8     8 ms     8 ms     4 ms  172.23.78.234
  9   102 ms     9 ms   147 ms  115.110.234.170.static.Mumbai.vsnl.net.in [115.110.234.170]
 10     *        *        *     Request timed out.
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
 13     *        *        *     Request timed out.
 14     *        *        *     Request timed out.
 15     *        *        *     Request timed out.
 16     *        *        *     Request timed out.
 17     *        *        *     Request timed out.
 18     *        *        *     Request timed out.
 19     *        *        *     Request timed out.
 20     *        *        *     Request timed out.
 21     *        *        *     Request timed out.
 22     *        *        *     Request timed out.
 23     *        *        *     Request timed out.
 24     *        *        *     Request timed out.
 25     *        *        *     Request timed out.
 26     *        *        *     Request timed out.
 27     *        *        *     Request timed out.
 28     *        *        *     Request timed out.
 29     *        *        *     Request timed out.
 30     *        *        *     Request timed out.

Trace complete.
```

QUESTIONS ABOUT PATHS

Now look at the results you gathered and answer the following questions about the paths taken by your packets. Store your answers in a file named traceroute.txt.

1. Is any part of the path common for all hosts you tracerouted?

Ans : Yes, the path to my ISP is always the same, and then the path depends on which access point is ready to respond.

2. Is there a relationship between the number of nodes that show up in the traceroute and the location of the host? If so, what is this relationship?

Ans : A hop is limited only to a specific distance and also depends largely on the bandwidth and the traffic present on the network. If the distance between the location of the user and that of the destination url is more, then more hops will be required in order to reach the destination as more number of access points will be used for routing.

3. Is there a relationship between the number of nodes that show up in the traceroute and latency of the host (from your ping results above)? Does the same relationship hold for all hosts?

Ans : Yes there is a direct relationship between the number of nodes and the latency of the host. If the latency of the host causes the traceroute request to get timed out even after the conventional three tries, then it keeps on sending the data packets until the host responds or upto a certain maximum hops. The same relationship may not hold for each host as it really depends on the time which the host takes to respond. If the host responds in the first request itself, the tracerouting stops with a success message.

Whois — The *whois* command can give detailed information about domain names and IP addresses. If it is not installed on the computers then install it with command sudo apt-get install whois in. *Whois* can tell you what organization owns or is responsible for the name or address and where to contact them. It often includes a list of domain name servers for the organization.

When using *whois* to look up a domain name, use the simple two-part network name, not an individual computer name (for example, *whois spit.ac.in*).

Exercise 4: (Short.) Use *whois* to investigate a well-known web site such as google.com or amazon.com, and write a couple of sentences about what you find out.

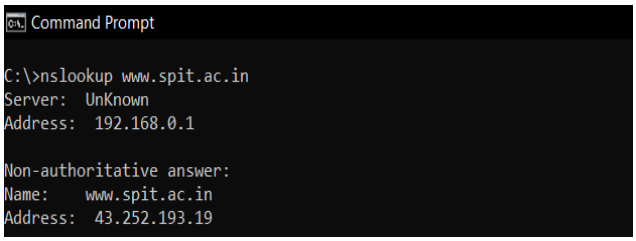
As shown in the below image, the whois command gives information about the domain name, the Registry Domain ID and some other details such as the details of the Registrar and the Registrant. For example, in case of amazon.com (domain name), the Registrant Organization is Amazon Technologies, Inc., the Registrant State/Province is NV and the Registrant Country is the United States. It also provides the domain expiry date.

Administrator: Command Prompt

```
Domain Name: amazon.com
Registry Domain ID: 281209_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.markmonitor.com
Registrar URL: http://www.markmonitor.com
Updated Date: 2019-08-26T12:19:56-0700
Creation Date: 1994-10-31T21:00:00-0800
Registrar Registration Expiration Date: 2024-10-30T00:00:00-0700
Registrar: MarkMonitor, Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
Registrar Abuse Contact Phone: +1.2083895770
Domain Status: clientUpdateProhibited (https://www.icann.org/epp#clientUpdateProhibited)
Domain Status: clientTransferProhibited (https://www.icann.org/epp#clientTransferProhibited)
Domain Status: clientDeleteProhibited (https://www.icann.org/epp#clientDeleteProhibited)
Domain Status: serverUpdateProhibited (https://www.icann.org/epp#serverUpdateProhibited)
Domain Status: serverTransferProhibited (https://www.icann.org/epp#serverTransferProhibited)
Domain Status: serverDeleteProhibited (https://www.icann.org/epp#serverDeleteProhibited)
Registry Registrant ID:
Registrant Name: Hostmaster, Amazon Legal Dept.
Registrant Organization: Amazon Technologies, Inc.
Registrant Street: P.O. Box 8102
Registrant City: Reno
Registrant State/Province: NV
Registrant Postal Code: 89507
Registrant Country: US
Registrant Phone: +1.2062664064
Registrant Phone Ext:
Registrant Fax: +1.2062667010
Registrant Fax Ext:
Registrant Email: hostmaster@amazon.com
Registry Admin ID:
Admin Name: Hostmaster, Amazon Legal Dept.
Admin Organization: Amazon Technologies, Inc.
Admin Street: P.O. Box 8102
Admin City: Reno
Admin State/Province: NV
Admin Postal Code: 89507
Admin Country: US
Admin Phone: +1.2062664064
Admin Phone Ext:
Admin Fax: +1.2062667010
Admin Fax Ext:
Admin Email: hostmaster@amazon.com
Registry Tech ID:
Tech Name: Hostmaster, Amazon Legal Dept.
Tech Organization: Amazon Technologies, Inc.
Tech Street: P.O. Box 8102
Tech City: Reno
Tech State/Province: NV
Tech Postal Code: 89507
Tech Country: US
```

```
Admin Phone: +1.2062664064
Admin Phone Ext:
Admin Fax: +1.2062667010
Admin Fax Ext:
Admin Email: hostmaster@amazon.com
Registry Tech ID:
Tech Name: Hostmaster, Amazon Legal Dept.
Tech Organization: Amazon Technologies, Inc.
Tech Street: P.O. Box 8102
Tech City: Reno
Tech State/Province: NV
Tech Postal Code: 89507
Tech Country: US
Tech Phone: +1.2062664064
Tech Phone Ext:
Tech Fax: +1.2062667010
Tech Fax Ext:
Tech Email: hostmaster@amazon.com
Name Server: ns2.p31.dynect.net
Name Server: pdns6.ultradns.co.uk
Name Server: ns1.p31.dynect.net
Name Server: ns3.p31.dynect.net
Name Server: ns4.p31.dynect.net
Name Server: pdns1.ultradns.net
DNSSEC: unsigned
URL of the ICANN WHOIS Data Problem Reporting System: http://wdprs.internic.net/
>>> Last update of WHOIS database: 2020-09-04T02:50:58-0700 <<<

For more information on WHOIS status codes, please visit:
  https://www.icann.org/resources/pages/epp-status-codes

If you wish to contact this domainΓÇÖs Registrant, Administrative, or Technical
contact, and such email address is not visible above, you may do so via our web
form, pursuant to ICANNΓÇÖs Temporary Specification. To verify that you are not a
```

Exercise 5: (Should be short.) Because of NAT, the domain name *spit.ac.in* has a different IP address outside of SPIT than it does on campus. Using information in this lab and working on a home computer, find the outside IP address for spit.ac.in. Explain how you did it. **Ans : nslookup** command could be used to find out the ip address of spit.ac.in



Geolocation — A geolocation service tries to tell, approximately, where a given IP address is located physically. They can't be completely accurate—but they probably get at least the country right most of the time.

This geolocation program is not installed on our computers, but you can access one on the command line using the *curl* command, which can send HTTP requests and display the response. The following command uses *curl* to contact a public web service that will look up an IP address for you: curl ipinfo.io/<IP-address>. For a specific example:curl ipinfo.io/129.64.99.200

(As you can see, you get back more than just the location.)

```
C:\WINDOWS\system32>curl ipinfo.io/129.64.99.200
{
  "ip": "129.64.99.200",
  "hostname": "websrv-prod.unet.brandeis.edu",
  "city": "Waltham",
  "region": "Massachusetts",
  "country": "US",
  "loc": "42.3765,-71.2356",
  "org": "AS10561 Brandeis University",
  "postal": "02453",
  "timezone": "America/New_York",
  "readme": "https://ipinfo.io/missingauth"
}
```

Conclusion : Basic network utilities which help in various functions were studied about in detail and their demonstration was done using the various exercises given to us to perform , helping us with the networking and communication using simple command line arguments.