Name : Jinal Shah
UID: 2019230070
Batch : D, T.E. Comps
Subject : DCCN

EXPERIMENT NO. 2

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>
```

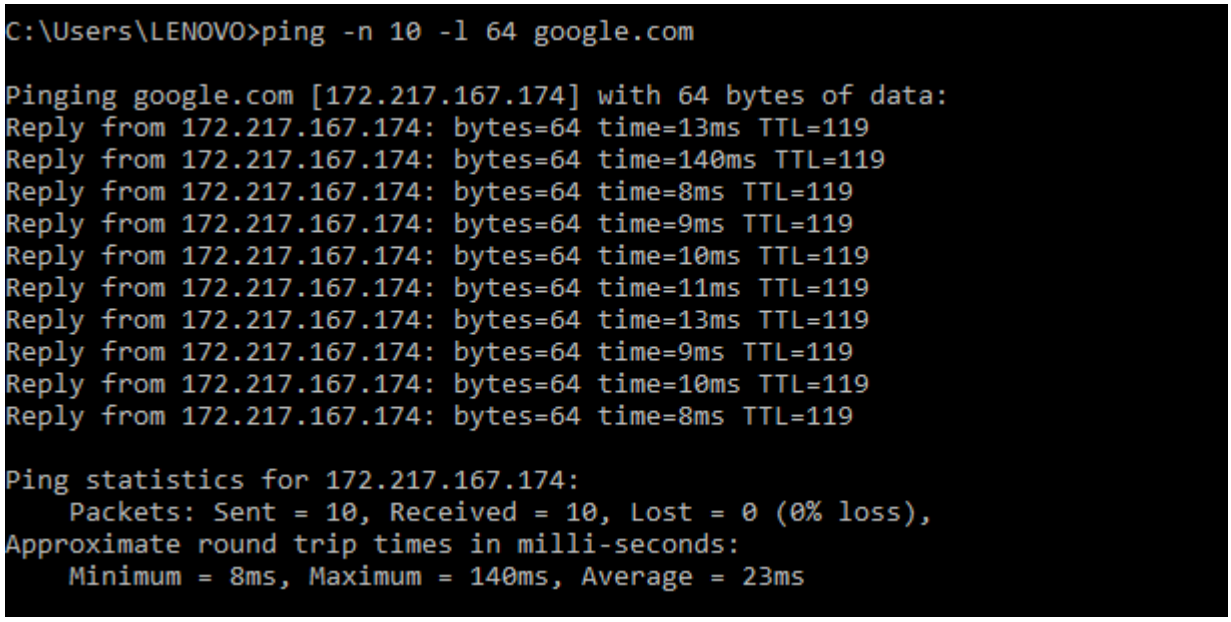
The default number of ICMP packets to send is either infinite (in Linux and Mac OS) or 4 (in Windows). The default packet size is either 64 bytes (in Linux) or 32 bytes (in Windows). You can specify either a hostname (e.g., `spit.ac.in`) or an IP address.

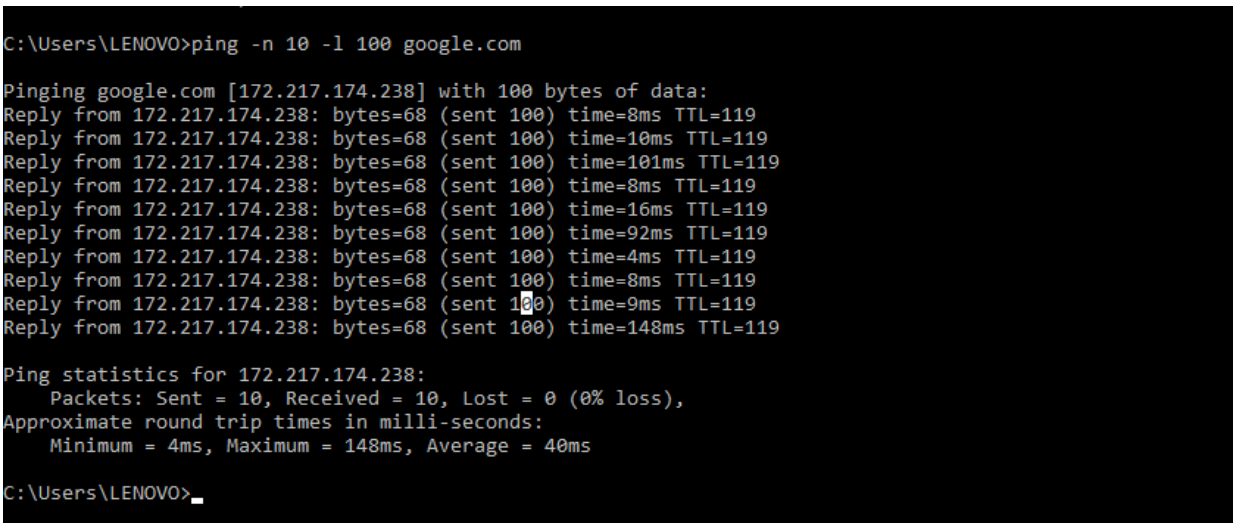
To save the output from `ping` to a file, include a greater than symbol and a file name at the end of the command. For example:

```
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





```
C:\Users\LENOVO>ping -n 10 -l 500 google.com

Pinging google.com [172.217.167.174] with 500 bytes of data:
Reply from 172.217.167.174: bytes=68 (sent 500) time=11ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=14ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=1863ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=77ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=16ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=99ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=7ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=13ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=10ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 500) time=11ms TTL=119

Ping statistics for 172.217.167.174:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 1863ms, Average = 212ms
```

```
C:\Users\LENOVO>ping -n 10 -l 1000 google.com

Pinging google.com [172.217.167.174] with 1000 bytes of data:
Reply from 172.217.167.174: bytes=68 (sent 1000) time=5ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=4ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=12ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=6ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=14ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=9ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=5ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=6ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=8ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1000) time=10ms TTL=119

Ping statistics for 172.217.167.174:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 14ms, Average = 7ms
```

```
C:\Users\LENOVO>ping -n 10 -l 1400 google.com

Pinging google.com [172.217.167.174] with 1400 bytes of data:
Reply from 172.217.167.174: bytes=68 (sent 1400) time=10ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=9ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=11ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=9ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=10ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=31ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=7ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=4ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=10ms TTL=119
Reply from 172.217.167.174: bytes=68 (sent 1400) time=11ms TTL=119

Ping statistics for 172.217.167.174:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 31ms, Average = 11ms
```

```
C:\Users\LENOVO>ping -n 10 -l 64 srmd.org

Pinging srmd.org [40.114.74.94] with 64 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 40.114.74.94:
    Packets: Sent = 10, Received = 0, Lost = 10 (100% loss),
```

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?

   Round-trip time represents the amount of time it takes data to go roundtrip to another location. 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.

   Network latency is closely related, but different than RTT. Latency is the time it takes for a packet to go from the sending endpoint to the receiving endpoint. Many factors may affect the latency of a service.

- **Processing Delay** is the time associated with the system analysing a packet header and determining where the packet must be sent. This depends heavily on the entries in the routing table, the execution of data structures in the system, and the hardware implementation.
- **Queueing Delay** is the time between a packet being queued and it being sent. This varies depending on the amount of traffic, the type of traffic, and what router queue algorithms are implemented. Different algorithms may adjust delays for system preference, or require the same delay for all traffic.

- **Transmission Delay** is the time needed to push a packet's data bits into the wire. This changes based on the size of the packet and the bandwidth. This does not depend on the distance of the wire, as it is solely the time to push a packet's bits into the wire, not to travel down the wire to the receiving endpoint.
- **Propagation Delay** is the time associated with the first bit of the packet traveling from the sending endpoint to the receiving endpoint. This is often referred to as a delay by distance, and as such is influenced by the distance the bit must travel and the propagation speed.

  These pieces of delay come together to make up the total delay in a network. This delay is extended by more variable levels of delay due to network congestion along with IP network delays which can range from a few ms to several hundred ms. Hence, from the above conclusions we can say that the average RTT does vary between different hosts.

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

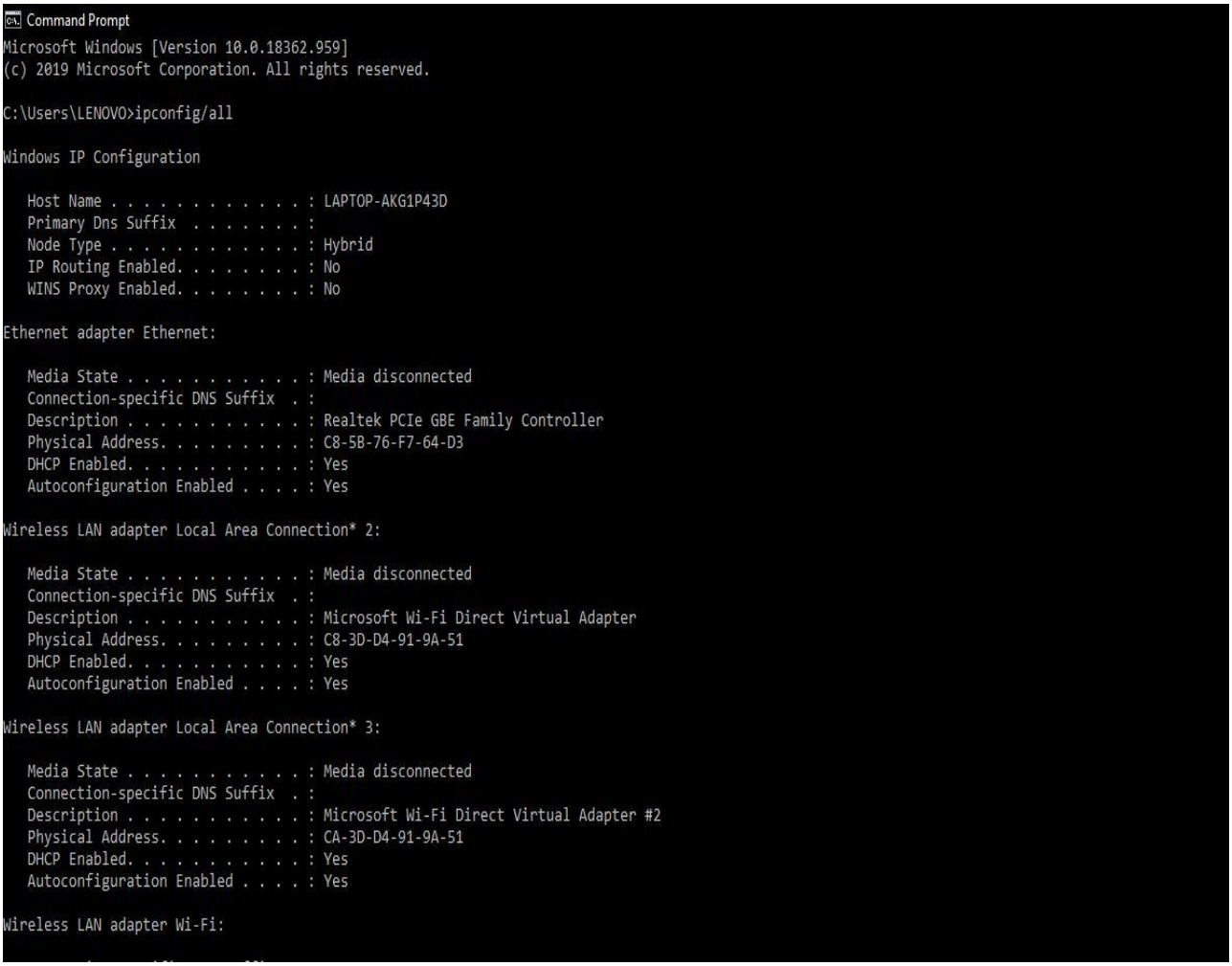
   The average RTT does vary with difference in packet sizes. This is because Queueing delay and transmission delay, both are dependent on the packet sizes, thus adding to the RTT.

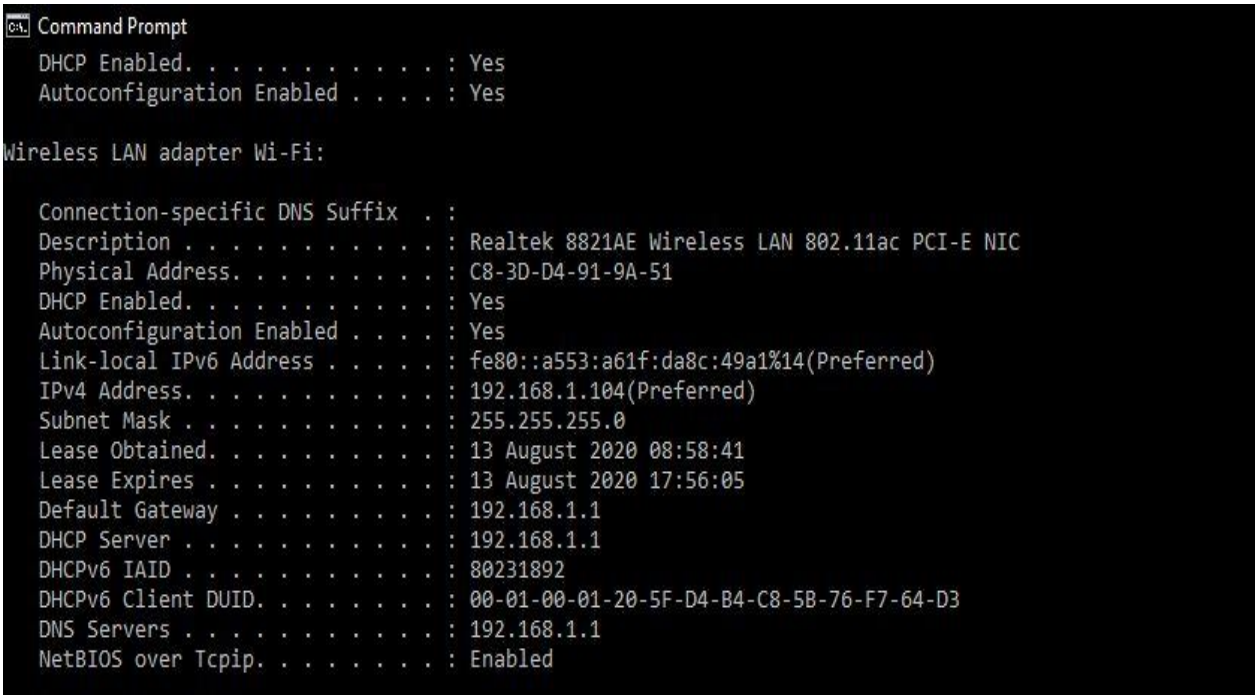
Exercise 1: Experiment with ping to find the round trip times to a variety of destinations. Write up any interesting observations, including in particular how the round trip time compares to the physical distance. Here are few places from who to get replies: www.uw.edu, www.cornell.edu, berkeley.edu,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

```
C:\Users\LENOVO>ping -n 7 -l 32 www.google.com

Pinging www.google.com [142.250.67.164] with 32 bytes of data:
Reply from 142.250.67.164: bytes=32 time=5ms TTL=119
Reply from 142.250.67.164: bytes=32 time=7ms TTL=119
Reply from 142.250.67.164: bytes=32 time=4ms TTL=119
Reply from 142.250.67.164: bytes=32 time=218ms TTL=119
Reply from 142.250.67.164: bytes=32 time=177ms TTL=119
Reply from 142.250.67.164: bytes=32 time=8ms TTL=119
Reply from 142.250.67.164: bytes=32 time=7ms TTL=119

Ping statistics for 142.250.67.164:
    Packets: Sent = 7, Received = 7, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 218ms, Average = 60ms
```

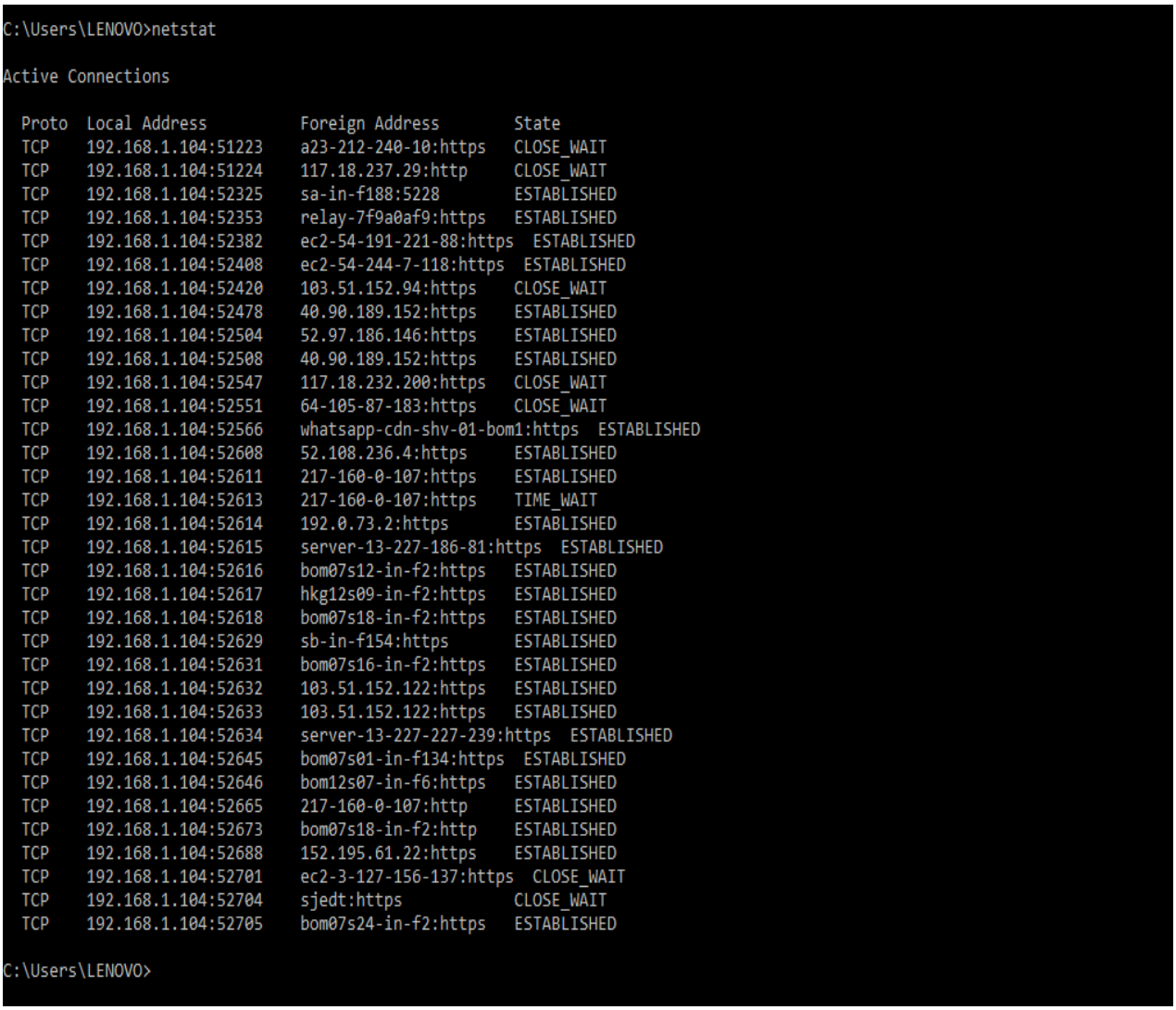
nslookup — The command nslookup <host> will do a DNS query to find and report the IP address (or addresses) for a domain name or the domain name corresponding to an IP address. To do this, it contacts a "DNS server." Default DNS servers are part of a computer's network configuration. (For a static IP 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>

ifconfig — 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!!!.)


Command Prompt
Microsoft Windows [Version 10.0.18362.959]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\LENOVO>ipconfig/all
Windows IP Configuration
Host Name . . . . . : LAPTOP-AKG1P43D
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
Ethernet adapter Ethernet:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Realtek PCIe GBE Family Controller
Physical Address. . . . . : C8-5B-76-F7-64-D3
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Wireless LAN adapter Local Area Connection* 2:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter
Physical Address. . . . . : C8-3D-D4-91-9A-51
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Wireless LAN adapter Local Area Connection* 3:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #2
Physical Address. . . . . : CA-3D-D4-91-9A-51
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Wireless LAN adapter Wi-Fi:


Command Prompt
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Wireless LAN adapter Wi-Fi:
Connection-specific DNS Suffix . :
Description . . . . . : Realtek 8821AE Wireless LAN 802.11ac PCI-E NIC
Physical Address. . . . . : C8-3D-D4-91-9A-51
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::a553:a61f:da8c:49a1%14(Preferred)
IPv4 Address. . . . . : 192.168.1.104(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : 13 August 2020 08:58:41
Lease Expires . . . . . : 13 August 2020 17:56:05
Default Gateway . . . . . : 192.168.1.1
DHCP Server . . . . . : 192.168.1.1
DHCPv6 IAID . . . . . : 80231892
DHCPv6 Client DUID. . . . . : 00-01-00-01-20-5F-D4-B4-C8-5B-76-F7-64-D3
DNS Servers . . . . . : 192.168.1.1
NetBIOS over Tcpip. . . . . : Enabled

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:\Users\LENOVO>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    192.168.1.104:51223    a23-212-240-10:https   CLOSE_WAIT
  TCP    192.168.1.104:51224    117.18.237.29:http     CLOSE_WAIT
  TCP    192.168.1.104:52325    sa-in-f188:5228        ESTABLISHED
  TCP    192.168.1.104:52353    relay-7f9a0af9:https   ESTABLISHED
  TCP    192.168.1.104:52382    ec2-54-191-221-88:https  ESTABLISHED
  TCP    192.168.1.104:52408    ec2-54-244-7-118:https  ESTABLISHED
  TCP    192.168.1.104:52420    103.51.152.94:https    CLOSE_WAIT
  TCP    192.168.1.104:52478    40.90.189.152:https    ESTABLISHED
  TCP    192.168.1.104:52504    52.97.186.146:https    ESTABLISHED
  TCP    192.168.1.104:52508    40.90.189.152:https    ESTABLISHED
  TCP    192.168.1.104:52547    117.18.232.200:https   CLOSE_WAIT
  TCP    192.168.1.104:52551    64-105-87-183:https    CLOSE_WAIT
  TCP    192.168.1.104:52566    whatsapp-cdn-shv-01-bom1:https  ESTABLISHED
  TCP    192.168.1.104:52608    52.108.236.4:https     ESTABLISHED
  TCP    192.168.1.104:52611    217-160-0-107:https    ESTABLISHED
  TCP    192.168.1.104:52613    217-160-0-107:https    TIME_WAIT
  TCP    192.168.1.104:52614    192.0.73.2:https       ESTABLISHED
  TCP    192.168.1.104:52615    server-13-227-186-81:https  ESTABLISHED
  TCP    192.168.1.104:52616    bom07s12-in-f2:https   ESTABLISHED
  TCP    192.168.1.104:52617    hkg12s09-in-f2:https   ESTABLISHED
  TCP    192.168.1.104:52618    bom07s18-in-f2:https   ESTABLISHED
  TCP    192.168.1.104:52629    sb-in-f154:https       ESTABLISHED
  TCP    192.168.1.104:52631    bom07s16-in-f2:https   ESTABLISHED
  TCP    192.168.1.104:52632    103.51.152.122:https   ESTABLISHED
  TCP    192.168.1.104:52633    103.51.152.122:https   ESTABLISHED
  TCP    192.168.1.104:52634    server-13-227-227-239:https  ESTABLISHED
  TCP    192.168.1.104:52645    bom07s01-in-f134:https  ESTABLISHED
  TCP    192.168.1.104:52646    bom12s07-in-f6:https   ESTABLISHED
  TCP    192.168.1.104:52665    217-160-0-107:http     ESTABLISHED
  TCP    192.168.1.104:52673    bom07s18-in-f2:http    ESTABLISHED
  TCP    192.168.1.104:52688    152.195.61.22:https    ESTABLISHED
  TCP    192.168.1.104:52701    ec2-3-127-156-137:https  CLOSE_WAIT
  TCP    192.168.1.104:52704    sjedt:https            CLOSE_WAIT
  TCP    192.168.1.104:52705    bom07s24-in-f2:https   ESTABLISHED

C:\Users\LENOVO>
```

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

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>
```

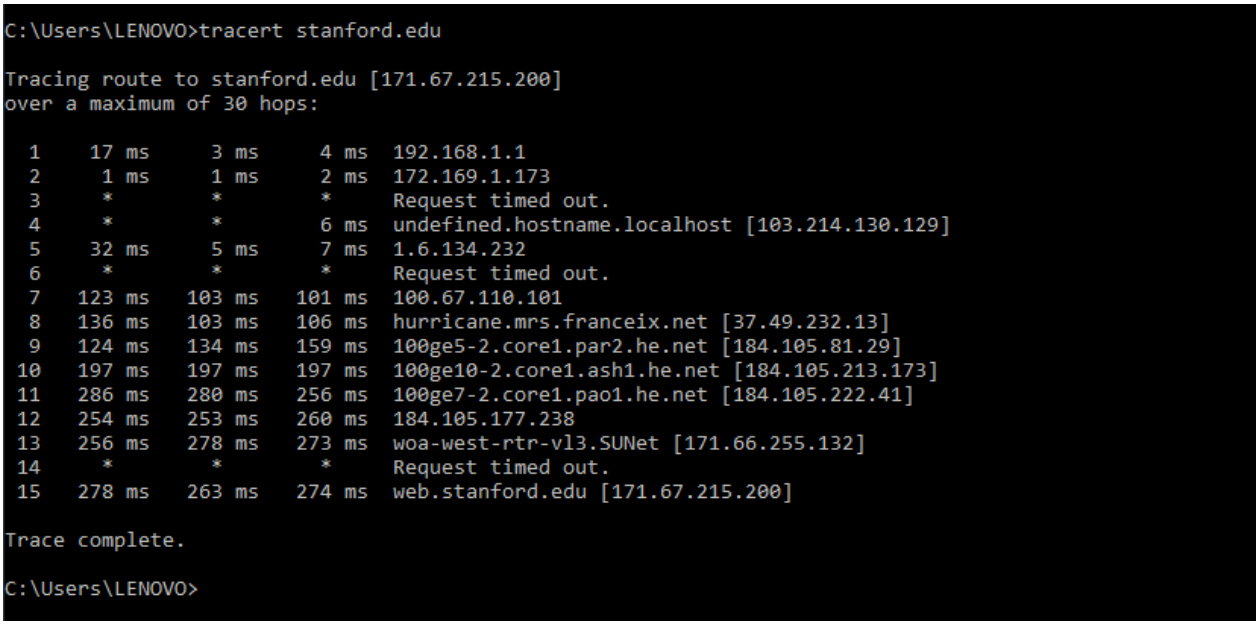
You can specify either a hostname (e.g., `cs.iitb.ac.in`) or an IP address (e.g., `128.105.2.6`).

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`).



```
C:\Users\LENOVO>tracert stanford.edu

Tracing route to stanford.edu [171.67.215.200]
over a maximum of 30 hops:

  1    17 ms     3 ms     4 ms  192.168.1.1
  2     1 ms     1 ms     2 ms  172.169.1.173
  3     *        *        *     Request timed out.
  4     *        *        6 ms  undefined.hostname.localhost [103.214.130.129]
  5    32 ms     5 ms     7 ms  1.6.134.232
  6     *        *        *     Request timed out.
  7   123 ms   103 ms   101 ms  100.67.110.101
  8   136 ms   103 ms   106 ms  hurricane.mrs.franceix.net [37.49.232.13]
  9   124 ms   134 ms   159 ms  100ge5-2.core1.par2.he.net [184.105.81.29]
 10   197 ms   197 ms   197 ms  100ge10-2.core1.ash1.he.net [184.105.213.173]
 11   286 ms   280 ms   256 ms  100ge7-2.core1.pao1.he.net [184.105.222.41]
 12   254 ms   253 ms   260 ms  184.105.177.238
 13   256 ms   278 ms   273 ms  woa-west-rtr-vl3.SUNet [171.66.255.132]
 14     *        *        *     Request timed out.
 15   278 ms   263 ms   274 ms  web.stanford.edu [171.67.215.200]

Trace complete.

C:\Users\LENOVO>
```

```
C:\Users\LENOVO>tracert cs.stanford.edu

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

  1     1 ms     1 ms     2 ms  192.168.1.1
  2     *        *        *     Request timed out.
  3     *        *        *     Request timed out.
  4     *        *        *     Request timed out.
  5   119 ms    32 ms   138 ms  38-97-87-183.mysipl.com [183.87.97.38]
  6    21 ms     3 ms     4 ms  172.23.78.237
  7    29 ms    30 ms    27 ms  172.31.244.45
  8    78 ms    33 ms    55 ms  ix-ae-4-2.tcore2.cxr-chennai.as6453.net [180.87.37.1]
  9   261 ms   260 ms   264 ms  if-ae-10-4.tcore2.svw-singapore.as6453.net [180.87.67.16]
 10   239 ms   245 ms   239 ms  if-ae-7-2.tcore2.lvw-losangeles.as6453.net [180.87.15.26]
 11   267 ms   241 ms   239 ms  if-ae-2-2.tcore1.lvw-losangeles.as6453.net [66.110.59.1]
 12   252 ms   278 ms   265 ms  las-b24-link.telia.net [80.239.128.214]
 13     *      298 ms   339 ms  palo-b24-link.telia.net [62.115.119.90]
 14     *      261 ms   275 ms  palo-b1-link.telia.net [62.115.122.169]
 15   263 ms   257 ms   258 ms  hurricane-ic-308019-palo-b1.c.telia.net [80.239.167.174]
 16   257 ms   323 ms   309 ms  stanford-university.100gigabitethernet5-1.core1.pao1.he.net [184.105.177.238]
 17   270 ms   377 ms   258 ms  csee-west-rtr-vl3.SUNet [171.66.255.140]
 18   320 ms   271 ms   254 ms  CS.stanford.edu [171.64.64.64]

Trace complete.

C:\Users\LENOVO>
```

```
C:\Users\LENOVO>tracert spit.ac.in

Tracing route to spit.ac.in [43.252.193.19]
over a maximum of 30 hops:

  1    22 ms     6 ms     3 ms  192.168.1.1
  2     5 ms     4 ms     5 ms  172.169.1.173
  3     *        *        *     Request timed out.
  4     5 ms     6 ms     8 ms  103.27.170.50
  5    10 ms     9 ms     9 ms  27.109.1.150
  6  3411 ms     8 ms     8 ms  103.205.124.82
  7    10 ms    12 ms     9 ms  43.252.192.230
  8     *        *        *     Request timed out.
  9     *        *        *     Request timed out.
 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.
```

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.

```
C:\Users\LENOVO>tracert google.com

Tracing route to google.com [172.217.160.206]
over a maximum of 30 hops:

  1    65 ms     2 ms     1 ms  192.168.1.1
  2     3 ms     2 ms    60 ms  172.169.1.173
  3     *        *        *     Request timed out.
  4   107 ms     5 ms     8 ms  72.14.219.48
  5     9 ms     5 ms     9 ms  209.85.247.207
  6    11 ms    63 ms    10 ms  216.239.47.149
  7     4 ms     4 ms     4 ms  bom07s16-in-f14.1e100.net [172.217.160.206]

Trace complete.

C:\Users\LENOVO>tracert www.google.com

Tracing route to www.google.com [216.58.203.4]
over a maximum of 30 hops:

  1     2 ms     2 ms     1 ms  192.168.1.1
  2    15 ms     3 ms     2 ms  172.169.1.173
  3     *        *        *     Request timed out.
  4    14 ms    12 ms     5 ms  72.14.219.48
  5     9 ms    11 ms    11 ms  209.85.245.197
  6    10 ms     7 ms     5 ms  172.253.77.21
  7    10 ms     5 ms     8 ms  bom12s04-in-f4.1e100.net [216.58.203.4]

Trace complete.
```

The only difference spotted in the tracert of both the websites is a slight time difference.

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.

tracert google.com executed on 13th Aug

```
C:\Users\LENOVO>tracert google.com

Tracing route to google.com [172.217.160.206]
over a maximum of 30 hops:

  1    65 ms     2 ms     1 ms  192.168.1.1
  2     3 ms     2 ms    60 ms  172.169.1.173
  3     *        *        *     Request timed out.
  4   107 ms     5 ms     8 ms  72.14.219.48
  5     9 ms     5 ms     9 ms  209.85.247.207
  6    11 ms    63 ms    10 ms  216.239.47.149
  7     4 ms     4 ms     4 ms  bom07s16-in-f14.1e100.net [172.217.160.206]

Trace complete.
```

tracert google.com executed on 19th Aug

```
C:\Users\LENOVO>tracert google.com

Tracing route to google.com [172.217.174.238]
over a maximum of 30 hops:

  1     4 ms     3 ms     1 ms  192.168.1.1
  2    66 ms     3 ms     2 ms  172.169.1.173
  3     *        *        *     Request timed out.
  4     6 ms     5 ms     5 ms  72.14.219.48
  5     7 ms     9 ms     7 ms  209.85.245.197
  6     4 ms     9 ms     6 ms  142.250.60.135
  7     6 ms    10 ms    10 ms  bom12s03-in-f14.1e100.net [172.217.174.238]
```

The only difference spotted while performing during different duration is a slight time difference in reaching the final destination.

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?

   Yes, the path to my Internet Service Provider (ISP) is same.

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?

   There is a proportional relation between the number of nodes and location of host.

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?

   Yes, there is a proportional relationship between the number of nodes and the latency of hosts.

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.

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.

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

```
C:\Users\LENOVO>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"
}
C:\Users\LENOVO>
```

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

Exercise 6: Find a few IP addresses that are connected to the web server on spit.ac.in right now, and determine where those IP addresses are located. (I'm expecting that there will be several; if not, try again in a few minutes or sometime later.) Find one that is far from Geneva, NY. Explain how you did it.

Conclusion :

Thus, in this experiment, I learnt about different networking tools and commands.

References :

1) https://www.cloudflare.com/learning/cdn/glossary/round-trip-time-rtt/
2) https://www.callstats.io/blog/2018/03/07/difference-between-jitter-and-latency-webrtc
3) https://www.callstats.io/blog/what-is-round-trip-time-and-how-does-it-relate-to-network-latency