Name: Khushi Jashnani
Batch – B
UID – 2018130017

Experiment - 2

AIM: To study the 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

```
C:\Users\Khushi>ping -n 10 -l 64 www.google.com

Pinging www.google.com [142.250.67.164] with 64 bytes of data:
Reply from 142.250.67.164: bytes=64 time=5ms TTL=118
Reply from 142.250.67.164: bytes=64 time=4ms TTL=118
Reply from 142.250.67.164: bytes=64 time=5ms TTL=118
Reply from 142.250.67.164: bytes=64 time=7ms TTL=118
Reply from 142.250.67.164: bytes=64 time=6ms TTL=118
Reply from 142.250.67.164: bytes=64 time=5ms TTL=118
Reply from 142.250.67.164: bytes=64 time=4ms TTL=118
Reply from 142.250.67.164: bytes=64 time=4ms TTL=118
Reply from 142.250.67.164: bytes=64 time=4ms TTL=118
Reply from 142.250.67.164: bytes=64 time=5ms TTL=118

Ping statistics for 142.250.67.164:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 7ms, Average = 4ms
```

```
C:\Users\Khushi>ping -n 10 -l 100 www.google.com

Pinging www.google.com [142.250.67.164] with 100 bytes of data:
Reply from 142.250.67.164: bytes=68 (sent 100) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=6ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 100) time=4ms TTL=118

Ping statistics for 142.250.67.164:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 6ms, Average = 4ms
```

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

Pinging www.google.com [142.250.67.164] with 500 bytes of data:
Reply from 142.250.67.164: bytes=68 (sent 500) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=6ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 500) time=5ms TTL=118

Ping statistics for 142.250.67.164:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 6ms, Average = 4ms
```

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

Pinging www.google.com [142.250.67.164] with 1000 bytes of data:
Reply from 142.250.67.164: bytes=68 (sent 1000) time=11ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=6ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=23ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1000) time=5ms TTL=118

Ping statistics for 142.250.67.164:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 23ms, Average = 7ms
```

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

Pinging www.google.com [142.250.67.164] with 1400 bytes of data:
Reply from 142.250.67.164: bytes=68 (sent 1400) time=6ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=6ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=39ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=8ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=5ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=4ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=6ms TTL=118
Reply from 142.250.67.164: bytes=68 (sent 1400) time=5ms TTL=118

Ping statistics for 142.250.67.164:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 39ms, Average = 8ms
```

Since we are receiving the same packet size of 68, the RTT is almost the same.

```
C:\Users\Khushi>ping -n 10 -l 64 www.stanford.edu

Pinging 89wyd637cdel.wpeproxy.com [104.18.167.96] with 64 bytes of data:
Reply from 104.18.167.96: bytes=64 time=5ms TTL=58
Reply from 104.18.167.96: bytes=64 time=4ms TTL=58
Reply from 104.18.167.96: bytes=64 time=5ms TTL=58
Reply from 104.18.167.96: bytes=64 time=5ms TTL=58
Reply from 104.18.167.96: bytes=64 time=6ms TTL=58
Reply from 104.18.167.96: bytes=64 time=5ms TTL=58
Reply from 104.18.167.96: bytes=64 time=5ms TTL=58
Reply from 104.18.167.96: bytes=64 time=6ms TTL=58
Reply from 104.18.167.96: bytes=64 time=7ms TTL=58
Reply from 104.18.167.96: bytes=64 time=8ms TTL=58

Ping statistics for 104.18.167.96:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 8ms, Average = 5ms
```

```
C:\Users\Khushi>ping -n 10 -l 100 www.stanford.edu

Pinging 89wyd637cdel.wpeproxy.com [104.18.167.96] with 100 bytes of data:
Reply from 104.18.167.96: bytes=100 time=7ms TTL=58
Reply from 104.18.167.96: bytes=100 time=5ms TTL=58
Reply from 104.18.167.96: bytes=100 time=6ms TTL=58
Reply from 104.18.167.96: bytes=100 time=8ms TTL=58
Reply from 104.18.167.96: bytes=100 time=5ms TTL=58
Reply from 104.18.167.96: bytes=100 time=5ms TTL=58
Reply from 104.18.167.96: bytes=100 time=5ms TTL=58
Reply from 104.18.167.96: bytes=100 time=5ms TTL=58
Reply from 104.18.167.96: bytes=100 time=6ms TTL=58
Reply from 104.18.167.96: bytes=100 time=5ms TTL=58

Ping statistics for 104.18.167.96:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 8ms, Average = 5ms
```

```
C:\Users\Khushi>ping -n 10 -l 500 www.stanford.edu

Pinging 89wyd637cdel.wpeproxy.com [104.18.167.96] with 500 bytes of data:
Reply from 104.18.167.96: bytes=500 time=6ms TTL=58
Reply from 104.18.167.96: bytes=500 time=7ms TTL=58
Reply from 104.18.167.96: bytes=500 time=9ms TTL=58
Reply from 104.18.167.96: bytes=500 time=5ms TTL=58
Reply from 104.18.167.96: bytes=500 time=5ms TTL=58
Reply from 104.18.167.96: bytes=500 time=6ms TTL=58
Reply from 104.18.167.96: bytes=500 time=6ms TTL=58
Reply from 104.18.167.96: bytes=500 time=6ms TTL=58
Reply from 104.18.167.96: bytes=500 time=9ms TTL=58
Reply from 104.18.167.96: bytes=500 time=5ms TTL=58

Ping statistics for 104.18.167.96:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 9ms, Average = 6ms
```

```
C:\Users\Khushi>ping -n 10 -l 1000 www.stanford.edu

Pinging 89wyd637cdel.wpeproxy.com [104.18.167.96] with 1000 bytes of data:
Reply from 104.18.167.96: bytes=1000 time=8ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=6ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=6ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=6ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=6ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=7ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=6ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=9ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=6ms TTL=58
Reply from 104.18.167.96: bytes=1000 time=6ms TTL=58

Ping statistics for 104.18.167.96:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 9ms, Average = 6ms
```

```
C:\Users\Khushi>ping -n 10 -l 1400 www.stanford.edu

Pinging 89wyd637cdel.wpeproxy.com [104.18.167.96] with 1400 bytes of data:
Reply from 104.18.167.96: bytes=1400 time=8ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=6ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=9ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=9ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=9ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=7ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=7ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=8ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=7ms TTL=58
Reply from 104.18.167.96: bytes=1400 time=6ms TTL=58

Ping statistics for 104.18.167.96:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 9ms, Average = 7ms
```

Here since the packets received are in increasing order, the average RTT is also in the increasing order.

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?

Answer –

Average RTT can vary between different hosts due to Processing delay**,** queuing delay, Transmission delay , and Propagation delay.

- **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.

Thus the different average RTT values of google.com and gmail.com can be because of the above mentioned factors.

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

Answer –

Yes, the average RTT increases with packet size as queuing, transmission delay increases as they 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).

Answer –

```
C:\Users\Khushi>ping www.uw.edu

Pinging www.washington.edu [128.95.155.135] with 32 bytes of data:
Reply from 128.95.155.135: bytes=32 time=258ms TTL=47
Reply from 128.95.155.135: bytes=32 time=259ms TTL=47
Reply from 128.95.155.135: bytes=32 time=257ms TTL=47
Reply from 128.95.155.135: bytes=32 time=257ms TTL=47

Ping statistics for 128.95.155.135:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 257ms, Maximum = 259ms, Average = 257ms

C:\Users\Khushi>ping www.cornell.edu

Pinging ucomm-gw1.cornell.media3.us [20.42.25.107] with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 20.42.25.107:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\Users\Khushi>ping www.uchicago.edu

Pinging wsee2.elb.uchicago.edu [3.224.151.213] with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 3.224.151.213:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

```
C:\Users\Khushi>ping www.berkeley.edu

Pinging www-production-1113102805.us-west-2.elb.amazonaws.com [35.160.53.243] with 32 bytes of data:
Reply from 35.160.53.243: bytes=32 time=273ms TTL=224
Reply from 35.160.53.243: bytes=32 time=275ms TTL=224
Reply from 35.160.53.243: bytes=32 time=270ms TTL=224
Reply from 35.160.53.243: bytes=32 time=269ms TTL=224

Ping statistics for 35.160.53.243:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 269ms, Maximum = 275ms, Average = 271ms

C:\Users\Khushi>ping www.ox.ac.uk

Pinging www.ox.ac.uk [151.101.66.133] with 32 bytes of data:
Reply from 151.101.66.133: bytes=32 time=33ms TTL=57
Reply from 151.101.66.133: bytes=32 time=41ms TTL=57
Reply from 151.101.66.133: bytes=32 time=30ms TTL=57
Reply from 151.101.66.133: bytes=32 time=56ms TTL=57

Ping statistics for 151.101.66.133:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 30ms, Maximum = 56ms, Average = 40ms

C:\Users\Khushi>ping www.u-tokyo.ac.jp

Pinging www.u-tokyo.ac.jp [210.152.243.234] with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 210.152.243.234:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Factors that influences RTT: Ref - [1]

There are certain factors that can bring huge changes in the value of RTT. These are enlisted below:

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.

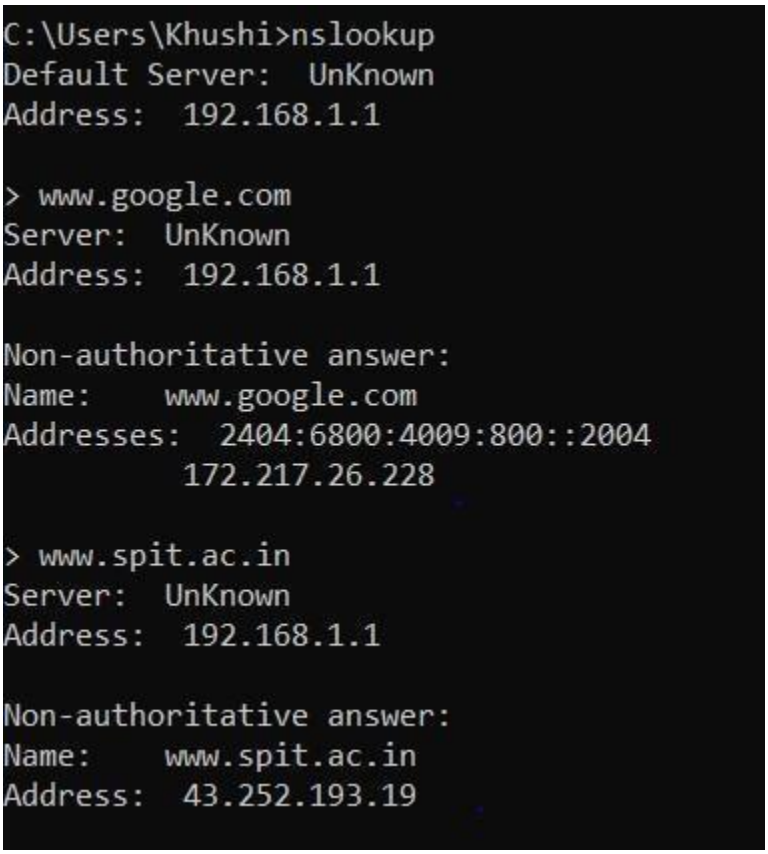
Server response time – the amount of time it takes a server to process and respond to a request is a potential bottleneck in network latency. When a server is overwhelmed with requests, such as during a DDoS attack, its ability to respond efficiently can be inhibited, resulting in increased RTT.

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.

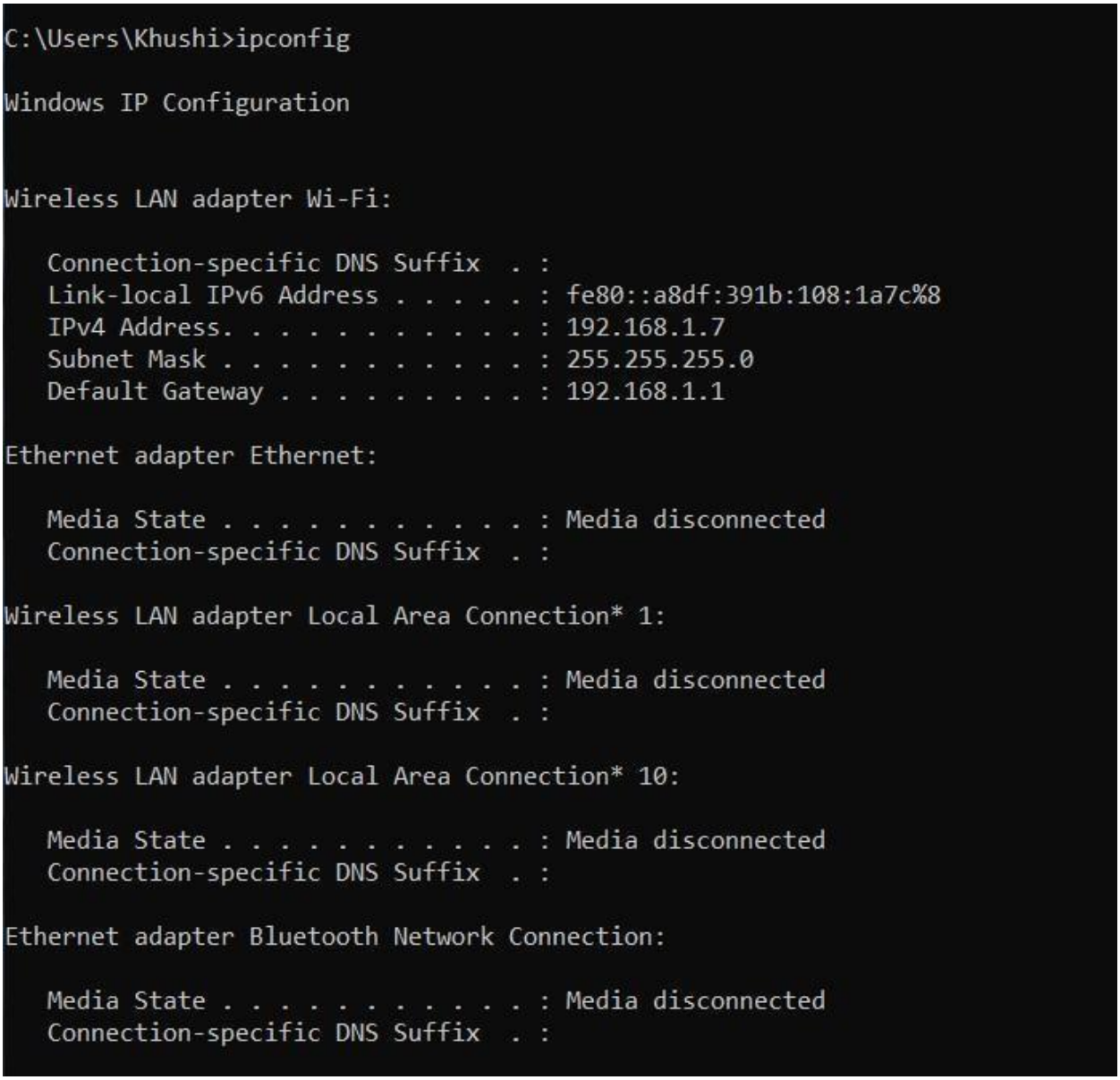
Physical distance – although a connection optimized by a CDN can often reduce the number of hops required to reach a destination, there is no way of getting around the limitation imposed by the speed of light; the distance between a start and end point is a limiting factor in network connectivity that 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.

Thus the round trip times varies due to these factors.

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 nslookup 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!!!.)



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.) Ref – [2]

Displays active TCP connections, ports on which the computer is listening, Ethernet statistics, the IP routing table, IPv4 statistics (for the IP, ICMP, TCP, and UDP

protocols), and IPv6 statistics (for the IPv6, ICMPv6, TCP over IPv6, and UDP over IPv6 protocols).

C:\Users\Khushi>netstat -n

Active Connections

Proto	Local Address	Foreign Address	State
TCP	127.0.0.1:49674	127.0.0.1:49675	ESTABLISHED
TCP	127.0.0.1:49675	127.0.0.1:49674	ESTABLISHED
TCP	127.0.0.1:49681	127.0.0.1:49682	ESTABLISHED
TCP	127.0.0.1:49682	127.0.0.1:49681	ESTABLISHED
TCP	127.0.0.1:49683	127.0.0.1:61900	ESTABLISHED
TCP	127.0.0.1:49684	127.0.0.1:49685	ESTABLISHED
TCP	127.0.0.1:49685	127.0.0.1:49684	ESTABLISHED
TCP	127.0.0.1:49686	127.0.0.1:49687	ESTABLISHED
TCP	127.0.0.1:49687	127.0.0.1:49686	ESTABLISHED
TCP	127.0.0.1:49688	127.0.0.1:61900	ESTABLISHED
TCP	127.0.0.1:49689	127.0.0.1:49690	ESTABLISHED
TCP	127.0.0.1:49690	127.0.0.1:49689	ESTABLISHED
TCP	127.0.0.1:49693	127.0.0.1:49830	ESTABLISHED
TCP	127.0.0.1:49693	127.0.0.1:49884	ESTABLISHED
TCP	127.0.0.1:49693	127.0.0.1:50303	ESTABLISHED
TCP	127.0.0.1:49694	127.0.0.1:49695	ESTABLISHED
TCP	127.0.0.1:49695	127.0.0.1:49694	ESTABLISHED
TCP	127.0.0.1:49707	127.0.0.1:49958	ESTABLISHED
TCP	127.0.0.1:49707	127.0.0.1:50326	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49726	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49736	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49740	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49741	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49742	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49744	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49757	ESTABLISHED
TCP	127.0.0.1:49721	127.0.0.1:49770	ESTABLISHED
TCP	127.0.0.1:49726	127.0.0.1:49721	ESTABLISHED
TCP	127.0.0.1:49727	127.0.0.1:49728	ESTABLISHED
TCP	127.0.0.1:49728	127.0.0.1:49727	ESTABLISHED
TCP	127.0.0.1:49729	127.0.0.1:61900	ESTABLISHED
TCP	127.0.0.1:49730	127.0.0.1:49731	ESTABLISHED

TCP 127.0.0.1:49731 127.0.0.1:49730 ESTABLISHED TCP 127.0.0.1:49736 127.0.0.1:49721 ESTABLISHED TCP 127.0.0.1:49740 127.0.0.1:49721 ESTABLISHED

TCP 127.0.0.1:49741 127.0.0.1:49721 ESTABLISHED
TCP 127.0.0.1:49742 127.0.0.1:49721 ESTABLISHED
TCP 127.0.0.1:49744 127.0.0.1:49721 ESTABLISHED
TCP 127.0.0.1:49748 127.0.0.1:49749 ESTABLISHED
TCP 127.0.0.1:49749 127.0.0.1:49748 ESTABLISHED
TCP 127.0.0.1:49750 127.0.0.1:61900 ESTABLISHED
TCP 127.0.0.1:49751 127.0.0.1:49752 ESTABLISHED
TCP 127.0.0.1:49752 127.0.0.1:49751 ESTABLISHED
TCP 127.0.0.1:49753 127.0.0.1:49754 ESTABLISHED
TCP 127.0.0.1:49754 127.0.0.1:49753 ESTABLISHED
TCP 127.0.0.1:49757 127.0.0.1:49721 ESTABLISHED
TCP 127.0.0.1:49758 127.0.0.1:49759 ESTABLISHED
TCP 127.0.0.1:49759 127.0.0.1:49758 ESTABLISHED
TCP 127.0.0.1:49770 127.0.0.1:49721 ESTABLISHED
TCP 127.0.0.1:49773 127.0.0.1:49774 ESTABLISHED
TCP 127.0.0.1:49774 127.0.0.1:49773 ESTABLISHED
TCP 127.0.0.1:49828 127.0.0.1:49829 ESTABLISHED
TCP 127.0.0.1:49829 127.0.0.1:49828 ESTABLISHED
TCP 127.0.0.1:49830 127.0.0.1:49693 ESTABLISHED
TCP 127.0.0.1:49831 127.0.0.1:49832 ESTABLISHED
TCP 127.0.0.1:49832 127.0.0.1:49831 ESTABLISHED
TCP 127.0.0.1:49882 127.0.0.1:49883 ESTABLISHED
TCP 127.0.0.1:49883 127.0.0.1:49882 ESTABLISHED
TCP 127.0.0.1:49884 127.0.0.1:49693 ESTABLISHED
TCP 127.0.0.1:49885 127.0.0.1:49886 ESTABLISHED
TCP 127.0.0.1:49886 127.0.0.1:49885 ESTABLISHED
TCP 127.0.0.1:49958 127.0.0.1:49707 ESTABLISHED
TCP 127.0.0.1:50260 127.0.0.1:50261 ESTABLISHED
TCP 127.0.0.1:50261 127.0.0.1:50260 ESTABLISHED
TCP 127.0.0.1:50262 127.0.0.1:61900 ESTABLISHED
TCP 127.0.0.1:50264 127.0.0.1:50265 ESTABLISHED
TCP 127.0.0.1:50265 127.0.0.1:50264 ESTABLISHED
TCP 127.0.0.1:50301 127.0.0.1:50302 ESTABLISHED
TCP 127.0.0.1:50302 127.0.0.1:50301 ESTABLISHED TCP 127.0.0.1:50303 127.0.0.1:49693 ESTABLISHED

TCP 127.0.0.1:50304 127.0.0.1:50305 ESTABLISHED
TCP 127.0.0.1:50305 127.0.0.1:50304 ESTABLISHED
TCP 127.0.0.1:50326 127.0.0.1:49707 ESTABLISHED

```
  TCP    127.0.0.1:50580        127.0.0.1:51879
ESTABLISHED     TCP    127.0.0.1:51879
127.0.0.1:50580        ESTABLISHED
  TCP    127.0.0.1:51939        127.0.0.1:51940        ESTABLISHED
  TCP    127.0.0.1:51940        127.0.0.1:51939        ESTABLISHED
  TCP    127.0.0.1:61900        127.0.0.1:49683        ESTABLISHED
  TCP    127.0.0.1:61900        127.0.0.1:49688        ESTABLISHED
  TCP    127.0.0.1:61900        127.0.0.1:49729        ESTABLISHED
  TCP    127.0.0.1:61900        127.0.0.1:49750        ESTABLISHED
  TCP    127.0.0.1:61900        127.0.0.1:50262        ESTABLISHED
  TCP    192.168.1.6:50038      13.227.165.79:443      CLOSE_WAIT
  TCP    192.168.1.6:50808      23.212.240.10:443      CLOSE_WAIT
  TCP    192.168.1.6:51835      74.125.68.188:443      ESTABLISHED
  TCP    192.168.1.6:51838      52.139.250.253:443     ESTABLISHED
  TCP    192.168.1.6:51858      52.194.117.234:443     ESTABLISHED
  TCP    192.168.1.6:51865      74.125.24.189:443      ESTABLISHED
  TCP    192.168.1.6:51871      172.67.132.251:443     ESTABLISHED
  TCP    192.168.1.6:52192      3.229.221.109:443      ESTABLISHED
  TCP    192.168.1.6:52214      117.18.232.200:443     CLOSE_WAIT
  TCP    192.168.1.6:52215      23.50.252.69:443       CLOSE_WAIT
  TCP    192.168.1.6:52323      40.90.189.152:443      ESTABLISHED
  TCP    192.168.1.6:52407      172.217.26.238:443     ESTABLISHED
  TCP    192.168.1.6:52479      216.58.203.42:443      ESTABLISHED
  TCP    192.168.1.6:52489      111.221.29.254:443     ESTABLISHED
  TCP    192.168.1.6:52490      104.28.4.80:443        TIME_WAIT
  TCP    192.168.1.6:52491      104.28.4.80:443        TIME_WAIT
  TCP    192.168.1.6:52493      23.50.244.164:443      ESTABLISHED
  TCP    192.168.1.6:52494      52.34.70.172:443       TIME_WAIT
  TCP    192.168.1.6:52495      104.28.4.80:443        TIME_WAIT
  TCP    192.168.1.6:52496      142.250.67.238:443     ESTABLISHED
  TCP    192.168.1.6:52497      216.58.196.74:443      ESTABLISHED
  TCP    192.168.1.6:52498      52.34.70.172:443       TIME_WAIT
  TCP    192.168.1.6:52499      104.28.4.80:443        TIME_WAIT
  TCP    [::1]:49708            [::1]:49709            ESTABLISHED
  TCP    [::1]:49709            [::1]:49708            ESTABLISHED
  TCP    [::1]:49714            [::1]:49715            ESTABLISHED
  TCP    [::1]:49715            [::1]:49714            ESTABLISHED
  TCP    [::1]:49717            [::1]:49718            ESTABLISHED
  TCP    [::1]:49718            [::1]:49717            ESTABLISHED
  TCP    [::1]:49719            [::1]:49720            ESTABLISHED
  TCP    [::1]:49720            [::1]:49719            ESTABLISHED
```

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 telnet to connect: telnet <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 "timeto-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>
```

Hop #	RTT 1	RTT 2	RTT 3	Name/IP Address
10	81 ms	74 ms	74 ms	205.134.225.38

Hop Number – This is the first column and is simply the number of the hop along the route. In this case, it is the tenth hop.

RTT Columns – The next three columns display the round trip time (RTT) for your packet to reach that point and return to your computer. This is listed in milliseconds.
There are three columns because the traceroute sends three separate signal packets. This is to display consistency, or a lack thereof, in the route.

Domain/IP column – The last column has the IP address of the router. If it is available, the domain name will also be listed.

You can specify either a hostname (e.g., cs.iitb.ac.in) or an IP address (e.g., 128.105.2.6). Ref – [3]

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\Khushi>tracert www.iitb.ac.in

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

  1     2 ms     *        1 ms  192.168.1.1
  2     2 ms     3 ms     3 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4     5 ms     7 ms     5 ms  103.42.160.13
  5     5 ms     4 ms     4 ms  182.79.146.180
  6     7 ms     8 ms     5 ms  115.110.234.141.static.Mumbai.vsnl.net.in [115.110.234.141]
  7     6 ms     5 ms     5 ms  115.110.234.170.static.Mumbai.vsnl.net.in [115.110.234.170]
  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.
```

```
C:\Users\Khushi>tracert mscs.mu.edu

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

  1     1 ms     1 ms     1 ms  192.168.1.1
  2     3 ms     2 ms     2 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4     8 ms     7 ms    18 ms  103.42.160.13
  5   201 ms   204 ms   193 ms  182.79.222.233
  6   776 ms   285 ms   218 ms  core1.nyc4.he.net [198.32.118.57]
  7     *        *      241 ms  100ge9-1.core2.chi1.he.net [184.105.223.161]
  8     *        *        *     Request timed out.
  9   311 ms   246 ms   369 ms  r-222wwash-isp-ae6-3926.wiscnet.net [140.189.8.126]
 10  1452 ms   243 ms   245 ms  r-milwaukeeci-809-isp-ae3-0.wiscnet.net [140.189.8.230]
 11   242 ms   245 ms   242 ms  MarquetteUniv.site.wiscnet.net [216.56.1.202]
 12   243 ms   243 ms   243 ms  134.48.10.27
 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.
```

```
C:\Users\Khushi>tracert www.cs.grinnell.edu

Tracing route to www.cs.grinnell.edu [132.161.132.159]
over a maximum of 30 hops:

  1     2 ms     1 ms     1 ms  192.168.1.1
  2     2 ms     2 ms     3 ms  45.117.0.82
  3     *        4 ms     *     249-1-226-103.intechonline.net [103.226.1.249]
  4     4 ms     4 ms     4 ms  103.42.160.13
  5   205 ms   204 ms   206 ms  116.119.52.163
  6   210 ms   210 ms   210 ms  core1.nyc4.he.net [198.32.118.57]
  7     *      252 ms     *     100ge2-1.core2.chi1.he.net [184.104.193.173]
  8   260 ms   250 ms   250 ms  100ge14-2.core1.msp1.he.net [184.105.223.178]
  9     *      253 ms   253 ms  aureon-network-services-inc.e0-26.switch1.msp1.he.net [216.66.77.218]
 10   249 ms   248 ms     *     peer-as5056.br02.msp1.tfbnw.net [157.240.76.37]
 11  1802 ms   257 ms   256 ms  167.142.58.40
 12  1048 ms   256 ms   256 ms  67.224.64.62
 13   255 ms   303 ms   267 ms  grinnellcollege1.desm.netins.net [167.142.65.43]
 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.
```

```
C:\Users\Khushi>tracert csail.mit.edu

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

  1     2 ms     1 ms     1 ms  192.168.1.1
  2     3 ms     2 ms     5 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4   360 ms    28 ms     4 ms  103.42.160.13
  5   266 ms   232 ms   249 ms  182.79.243.31
  6   228 ms   228 ms   229 ms  xe-5-1-0.edge1.LosAngeles6.Level3.net [4.26.0.89]
  7     *        *        *     Request timed out.
  8   267 ms   266 ms   266 ms  MASSACHUSET.bear1.Boston1.Level3.net [4.53.48.98]
  9   271 ms   267 ms   267 ms  dmz-rtr-1-external-rtr-1.mit.edu [18.0.161.17]
 10   277 ms   277 ms   277 ms  dmz-rtr-2-dmz-rtr-1-2.mit.edu [18.0.162.6]
 11   264 ms   265 ms   263 ms  mitnet.core-1-ext.csail.mit.edu [18.4.7.65]
 12     *        *        *     Request timed out.
 13   874 ms   264 ms   337 ms  bdr.core-1.csail.mit.edu [128.30.0.246]
 14  1411 ms   277 ms   277 ms  inquir-3ld.csail.mit.edu [128.30.2.109]

Trace complete.
```

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

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

  1    12 ms     2 ms     4 ms  192.168.1.1
  2     3 ms     2 ms     4 ms  45.117.0.82
  3     *        5 ms     3 ms  249-1-226-103.intechonline.net [103.226.1.249]
  4     5 ms     4 ms    44 ms  103.42.160.13
  5   213 ms   212 ms  1142 ms  aes-static-150.36.144.59.airtel.in [59.144.36.150]
  6   222 ms   686 ms   314 ms  core1.nyc4.he.net [198.32.118.57]
  7   256 ms   271 ms     *     100ge8-1.core1.sjc2.he.net [184.105.81.218]
  8   302 ms   251 ms  1334 ms  100ge1-1.core1.pao1.he.net [72.52.92.158]
  9   263 ms   263 ms   268 ms  stanford-university.100gigabitethernet5-1.core1.pao1.he.net [184.105.177.238]
 10   257 ms   257 ms   257 ms  csee-west-rtr-vl3.SUNet [171.66.255.140]
 11   257 ms   264 ms   257 ms  CS.stanford.edu [171.64.64.64]

Trace complete.
```

```
C:\Users\Khushi>tracert cs.manchester.ac.uk

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

  1     1 ms     1 ms     1 ms  192.168.1.1
  2     5 ms     2 ms     2 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4     4 ms     5 ms     4 ms  103.42.160.13
  5   252 ms   255 ms   252 ms  182.79.154.0
  6     *      923 ms   241 ms  ldn-b4-link.telia.net [62.115.162.232]
  7   249 ms   249 ms   248 ms  jisc-ic-345131-ldn-b4.c.telia.net [62.115.175.131]
  8   240 ms   240 ms   239 ms  ae24.londhx-sbr1.ja.net [146.97.35.197]
  9   261 ms   255 ms   255 ms  ae29.londpg-sbr2.ja.net [146.97.33.2]
 10   254 ms   253 ms   252 ms  ae31.erdiss-sbr2.ja.net [146.97.33.22]
 11   256 ms   256 ms   259 ms  ae29.manckh-sbr2.ja.net [146.97.33.42]
 12   249 ms   250 ms   250 ms  ae23.mancrh-rbr1.ja.net [146.97.38.42]
 13     *        *      256 ms  universityofmanchester.ja.net [146.97.169.2]
 14   247 ms   246 ms   249 ms  130.88.249.194
 15     *        *        *     Request timed out.
 16   251 ms  1191 ms   250 ms  gw-jh.its.manchester.ac.uk [130.88.250.32]
 17   290 ms  1199 ms   258 ms  eps.its.man.ac.uk [130.88.101.49]

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\Khushi>tracert math.hws.edu

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

  1     1 ms     3 ms     1 ms  192.168.1.1
  2     3 ms     2 ms     2 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4     8 ms     7 ms     4 ms  103.42.160.13
  5   239 ms   231 ms   233 ms  182.79.245.81
  6   220 ms   221 ms   221 ms  ae58.edge1.LosAngeles6.Level3.net [4.26.0.17]
  7     *        *        *     Request timed out.
  8     *        *        *     Request timed out.
  9   262 ms   261 ms   263 ms  roc1-ar5-xe-0-0-0-0.us.twtelecom.net [35.248.1.158]
 10   263 ms   263 ms   263 ms  66-195-65-170.static.ctl.one [66.195.65.170]
 11   262 ms   262 ms   265 ms  64.89.144.100
 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.
```

```
C:\Users\Khushi>tracert www.hws.edu

Tracing route to www.hws.edu [64.89.145.159]
over a maximum of 30 hops:

  1     3 ms     1 ms     1 ms  192.168.1.1
  2     3 ms     5 ms     2 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4     6 ms     6 ms     4 ms  103.42.160.13
  5   227 ms   225 ms   226 ms  182.79.247.32
  6   226 ms   225 ms   228 ms  xe-5-1-0.edge1.LosAngeles6.Level3.net [4.26.0.89]
  7   226 ms   224 ms   225 ms  ae-1-51.ear3.LosAngeles1.Level3.net [4.69.206.225]
  8     *        *        *     Request timed out.
  9   264 ms   263 ms   264 ms  roc1-ar5-xe-0-0-0-0.us.twtelecom.net [35.248.1.158]
 10   277 ms   269 ms   266 ms  66-195-65-170.static.ctl.one [66.195.65.170]
 11   263 ms   264 ms   263 ms  64.89.144.100
 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.
```

From the above results , we can see that
since the two domains are from the same university the initial part of the route is same.

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.

Observation on 18th August

```
C:\Users\Khushi>tracert cs.manchester.ac.uk

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

  1     1 ms     1 ms     1 ms  192.168.1.1
  2     5 ms     2 ms     2 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4     4 ms     5 ms     4 ms  103.42.160.13
  5   252 ms   255 ms   252 ms  182.79.154.0
  6     *      923 ms   241 ms  ldn-b4-link.telia.net [62.115.162.232]
  7   249 ms   249 ms   248 ms  jisc-ic-345131-ldn-b4.c.telia.net [62.115.175.131]
  8   240 ms   240 ms   239 ms  ae24.londhx-sbr1.ja.net [146.97.35.197]
  9   261 ms   255 ms   255 ms  ae29.londpg-sbr2.ja.net [146.97.33.2]
 10   254 ms   253 ms   252 ms  ae31.erdiss-sbr2.ja.net [146.97.33.22]
 11   256 ms   256 ms   259 ms  ae29.manckh-sbr2.ja.net [146.97.33.42]
 12   249 ms   250 ms   250 ms  ae23.mancrh-rbr1.ja.net [146.97.38.42]
 13     *        *      256 ms  universityofmanchester.ja.net [146.97.169.2]
 14   247 ms   246 ms   249 ms  130.88.249.194
 15     *        *        *     Request timed out.
 16   251 ms  1191 ms   250 ms  gw-jh.its.manchester.ac.uk [130.88.250.32]
 17   290 ms  1199 ms   258 ms  eps.its.man.ac.uk [130.88.101.49]

Trace complete.
```

Observation on 25th August

```
C:\Users\Khushi>tracert cs.manchester.ac.uk

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

  1     2 ms     *        1 ms  192.168.1.1
  2     3 ms     2 ms     2 ms  45.117.0.82
  3     *        *        *     Request timed out.
  4     6 ms     3 ms     3 ms  103.42.160.13
  5   245 ms   267 ms   243 ms  182.79.154.0
  6     *        *        *     Request timed out.
  7   240 ms   240 ms   249 ms  jisc-ic-345131-ldn-b4.c.telia.net [62.115.175.131]
  8   239 ms   238 ms   240 ms  ae24.londhx-sbr1.ja.net [146.97.35.197]
  9   239 ms   238 ms   238 ms  ae29.londpg-sbr2.ja.net [146.97.33.2]
 10   244 ms   243 ms   244 ms  ae31.erdiss-sbr2.ja.net [146.97.33.22]
 11   247 ms   246 ms   249 ms  ae29.manckh-sbr2.ja.net [146.97.33.42]
 12   249 ms   251 ms   250 ms  ae23.mancrh-rbr1.ja.net [146.97.38.42]
 13     *        *      249 ms  universityofmanchester.ja.net [146.97.169.2]
 14   247 ms   246 ms   249 ms  130.88.249.194
 15     *        *        *     Request timed out.
 16   248 ms   248 ms   983 ms  gw-jh.its.manchester.ac.uk [130.88.250.32]
 17   253 ms   257 ms   247 ms  eps.its.man.ac.uk [130.88.101.49]

Trace complete.
```

Through this we get to know that in spite of the source and destination being the same it is not necessary that the path of the route or the intermediate nodes and their respective RTTs will also be the same.

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?

   Answer - Yes, the tracerouting follows a particular path from the user's IP address through the IP addresses of the ISP and then the path depends on which access point is ready to respond and which access points or routers have firewalls configured for blocking the requests and accordingly, the destination can be reached through different paths at different times.

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?

   Answer – Yes, the number of nodes depends on the distance between the source and destination and intermediate interfaces.

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?

   Answer - There is a direct relationship between the number of nodes and the latency of the host. It also depends on the packet size. The amount of latency is largely dependent on how far the visitor is from the server location and how many nodes the signal has to travel through.

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\Khushi>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"
}
```

(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 –

I learnt that the main difference between Ping and Traceroute is that Ping is a quick and easy utility to tell if the specified server is live and reachable whereas Traceroute finds the exact route taken to reach the server and time taken by each step (hop).

References –

1) https://www.cloudflare.com/learning/cdn/glossary/round-trip-time-rtt/
2) https://docs.microsoft.com/enus/windowsserver/administration/windowscommands/netstat
3) https://www.inmotionhosting.com/support/website/ssh/read-traceroute/