

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 receive 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 response 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

Pinging www.ox.ac.uk with 10 packets of size 64 bytes

```
C:\Users\Sarika>ping -n 10 -l 64 www.ox.ac.uk

Pinging www.ox.ac.uk [151.101.66.133] with 64 bytes of data:
Reply from 151.101.66.133: bytes=64 time=8ms TTL=59
Reply from 151.101.66.133: bytes=64 time=7ms TTL=59
Reply from 151.101.66.133: bytes=64 time=8ms TTL=59
Reply from 151.101.66.133: bytes=64 time=7ms TTL=59
Reply from 151.101.66.133: bytes=64 time=7ms TTL=59
Reply from 151.101.66.133: bytes=64 time=6ms TTL=59
Reply from 151.101.66.133: bytes=64 time=13ms TTL=59
Reply from 151.101.66.133: bytes=64 time=21ms TTL=59
Reply from 151.101.66.133: bytes=64 time=6ms TTL=59
Reply from 151.101.66.133: bytes=64 time=6ms TTL=59

Ping statistics for 151.101.66.133:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 6ms, Maximum = 21ms, Average = 8ms
```

Pinging www.ox.ac.uk with 10 packets of size 100 bytes

```
C:\Users\Sarika>ping -n 10 -l 100 www.ox.ac.uk

Pinging www.ox.ac.uk [151.101.66.133] with 100 bytes of data:
Reply from 151.101.66.133: bytes=100 time=6ms TTL=59
Reply from 151.101.66.133: bytes=100 time=9ms TTL=59
Reply from 151.101.66.133: bytes=100 time=9ms TTL=59
Reply from 151.101.66.133: bytes=100 time=7ms TTL=59
Reply from 151.101.66.133: bytes=100 time=6ms TTL=59
Reply from 151.101.66.133: bytes=100 time=7ms TTL=59
Reply from 151.101.66.133: bytes=100 time=7ms TTL=59
Reply from 151.101.66.133: bytes=100 time=7ms TTL=59
Reply from 151.101.66.133: bytes=100 time=50ms TTL=59
Reply from 151.101.66.133: bytes=100 time=7ms TTL=59

Ping statistics for 151.101.66.133:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 6ms, Maximum = 50ms, Average = 11ms
```

Pinging www.ox.ac.uk with 10 packets of size 500 bytes

```
C:\Users\Sarika>ping -n 10 -l 500 www.ox.ac.uk

Pinging www.ox.ac.uk [151.101.66.133] with 500 bytes of data:
Reply from 151.101.66.133: bytes=500 time=10ms TTL=59
Reply from 151.101.66.133: bytes=500 time=10ms TTL=59
Reply from 151.101.66.133: bytes=500 time=9ms TTL=59
Reply from 151.101.66.133: bytes=500 time=8ms TTL=59
Reply from 151.101.66.133: bytes=500 time=63ms TTL=59
Reply from 151.101.66.133: bytes=500 time=7ms TTL=59
Reply from 151.101.66.133: bytes=500 time=11ms TTL=59
Reply from 151.101.66.133: bytes=500 time=7ms TTL=59
Reply from 151.101.66.133: bytes=500 time=8ms TTL=59
Reply from 151.101.66.133: bytes=500 time=7ms TTL=59

Ping statistics for 151.101.66.133:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 63ms, Average = 14ms
```

Pinging www.ox.ac.uk with 10 packets of size 1000 bytes

```
C:\Users\Sarika>ping -n 10 -l 1000 www.ox.ac.uk

Pinging www.ox.ac.uk [151.101.66.133] with 1000 bytes of data:
Reply from 151.101.66.133: bytes=1000 time=6ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=8ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=7ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=10ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=7ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=8ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=9ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=9ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=9ms TTL=59
Reply from 151.101.66.133: bytes=1000 time=14ms TTL=59

Ping statistics for 151.101.66.133:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 14ms, Average = 8ms
```

Pinging www.ox.ac.uk with 10 packets of size 1400 bytes

```
C:\Users\Sarika>ping -n 10 -l 1400 www.ox.ac.uk

Pinging www.ox.ac.uk [151.101.66.133] with 1400 bytes of data:
Reply from 151.101.66.133: bytes=1400 time=10ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=11ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=10ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=7ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=9ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=7ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=11ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=7ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=11ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=9ms TTL=59

Ping statistics for 151.101.66.133:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 11ms, Average = 9ms
```

Pinging spit.ac.in with 10 packets of size 1400 bytes

```
C:\Users\Sarika>ping -n 10 -l 1400 spit.ac.in

Pinging spit.ac.in [43.252.193.19] with 1400 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 43.252.193.19:
    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 (RTT) is the length time it takes for a data packet to be sent to a destination plus the time it takes for an acknowledgment of that packet to be received back at the origin. The RTT between a network and server can be determined by using the ping command..

Network delay is a design and performance characteristic of a [telecommunications network](#). It specifies the [latency](#) for a bit of data to travel across the network from one [communication endpoint](#) to another. It is typically measured in multiples or fractions of a second. 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
- [Queuing delay](#) – time the packet spends in routing queues
- [Transmission delay](#) – time it takes to push the packet's bits onto the link
- [Propagation delay](#) – time for a signal to reach its destination

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

RTT depends on the network infrastructure, the distance between nodes, network conditions, and packet size. Packet size and payload compressibility have a significant impact on RTT for slower links.

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.cornell.edu

```
C:\Users\Sarika>ping -n 10 -l 64 www.cornell.edu

Pinging ucomm-gw1.cornell.media3.us [20.42.25.107] 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 20.42.25.107:
    Packets: Sent = 10, Received = 0, Lost = 10 (100% loss),
```

Pinging www.berkeley.edu

```
C:\Users\Sarika>ping -n 10 -l 64 berkley.edu

Pinging berkley.edu [185.53.177.51] 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 185.53.177.51:
    Packets: Sent = 10, Received = 0, Lost = 10 (100% loss),
```

Pinging www.uchicago.edu

```
C:\Users\Sarika>ping -n 10 -l 64 www.uchicago.edu

Pinging wsee2.elb.uchicago.edu [54.89.29.50] 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 54.89.29.50:
    Packets: Sent = 10, Received = 0, Lost = 10 (100% loss),
```

Pinging www.ox.ac.uk

```
C:\Users\Sarika>ping -n 10 -l 64 www.ox.ac.uk

Pinging www.ox.ac.uk [151.101.194.133] with 64 bytes of data:
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56
Reply from 151.101.194.133: bytes=64 time=30ms TTL=56
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56
Reply from 151.101.194.133: bytes=64 time=30ms TTL=56
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56
Reply from 151.101.194.133: bytes=64 time=29ms TTL=56

Ping statistics for 151.101.194.133:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 29ms, Maximum = 30ms, Average = 29ms

C:\Users\Sarika>
```

Pinging www.ue.edu

```
C:\Users\Sarika>ping -n 10 -l 64 www.ue.edu

Pinging www.ue.edu [87.106.52.243] with 64 bytes of data:
Reply from 87.106.52.243: bytes=64 time=144ms TTL=114
Reply from 87.106.52.243: bytes=64 time=145ms TTL=114
Reply from 87.106.52.243: bytes=64 time=145ms TTL=114
Reply from 87.106.52.243: bytes=64 time=144ms TTL=114
Reply from 87.106.52.243: bytes=64 time=144ms TTL=114
Reply from 87.106.52.243: bytes=64 time=144ms TTL=114
Reply from 87.106.52.243: bytes=64 time=144ms TTL=114
Reply from 87.106.52.243: bytes=64 time=143ms TTL=114
Reply from 87.106.52.243: bytes=64 time=143ms TTL=114
Reply from 87.106.52.243: bytes=64 time=145ms TTL=114

Ping statistics for 87.106.52.243:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 143ms, Maximum = 145ms, Average = 144ms
```

Pinging www.u-tokyo.ac.jp

```
C:\Users\Sarika>ping -n 10 -l 64 www.u-tokyo.ac.jp

Pinging www.u-tokyo.ac.jp [210.152.243.234] 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 210.152.243.234:
    Packets: Sent = 10, Received = 0, Lost = 10 (100% loss),
```

Observation-

Actual round trip time between different hosts can be influenced by:

- **Distance** – The length a signal has to travel correlates with the time taken for a request to reach a server and a response to reach a browser.
- **Transmission medium** – The medium used to route a signal (e.g., copper wire, fiber optic cables) can impact how quickly a request is received by a server and routed back to a user.
- **Number of network hops** – Intermediate routers or servers take time to process a signal, increasing RTT. The more hops a signal has to travel through, the higher the RTT.
- **Traffic levels** – RTT typically increases when a network is congested with high levels of traffic. Conversely, low traffic times can result in decreased RTT.

Server response time – The time taken for a target server to respond to a request depends on its processing capacity, the number of requests being handled and the nature of the request (i.e., how much server-side work is required). A longer server response time increases 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 `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!!!.)


```

C:\Users\Sarika>ipconfig -all

Windows IP Configuration

Host Name . . . . . : DESKTOP-32B0IEB
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No

Wireless LAN adapter Local Area Connection* 9:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter
Physical Address. . . . . : E4-70-B8-8A-E7-ED
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Local Area Connection* 10:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #2
Physical Address. . . . . : E6-70-B8-8A-E7-EC
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . :
Description . . . . . : Intel(R) Dual Band Wireless-AC 8265
Physical Address. . . . . : E4-70-B8-8A-E7-EC
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::c83e:9747:6843:9dd4%15(Preferred)
IPv4 Address. . . . . : 192.168.0.106(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : 11 August 2020 21:45:45
Lease Expires . . . . . : 14 August 2020 15:59:18
Default Gateway . . . . . : 192.168.0.1
DHCP Server . . . . . : 192.168.0.1
DHCPv6 IAID . . . . . : 82079928
DHCPv6 Client DUID. . . . . : 00-01-00-01-22-CC-B2-27-E4-70-B8-8A-E7-EC
DNS Servers . . . . . : 192.168.0.1
                        0.0.0.0
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.)

Command Prompt

C:\Users\Sarika>netstat -n

Active Connections

Proto	Local Address	Foreign Address	State
TCP	127.0.0.1:49675	127.0.0.1:49676	ESTABLISHED
TCP	127.0.0.1:49676	127.0.0.1:49675	ESTABLISHED
TCP	192.168.0.103:64719	74.125.68.188:5228	ESTABLISHED
TCP	192.168.0.103:64739	40.90.189.152:443	ESTABLISHED
TCP	192.168.0.103:64748	74.125.130.132:443	ESTABLISHED
TCP	192.168.0.103:64763	172.217.194.113:443	ESTABLISHED
TCP	192.168.0.103:64764	74.125.24.95:443	ESTABLISHED
TCP	192.168.0.103:64766	40.90.189.152:443	ESTABLISHED
TCP	192.168.0.103:64772	74.125.200.138:443	ESTABLISHED
TCP	192.168.0.103:64774	74.125.200.189:443	ESTABLISHED
TCP	192.168.0.103:64786	52.111.240.8:443	ESTABLISHED
TCP	192.168.0.103:64799	52.109.124.33:443	ESTABLISHED
TCP	192.168.0.103:64801	40.90.189.152:443	ESTABLISHED
TCP	192.168.0.103:64808	5.45.59.35:80	ESTABLISHED
TCP	192.168.0.103:64812	74.125.200.101:443	ESTABLISHED
TCP	192.168.0.103:64813	192.168.0.108:7676	CLOSE_WAIT
TCP	192.168.0.103:64819	52.109.124.5:443	TIME_WAIT
TCP	192.168.0.103:64820	52.109.124.5:443	TIME_WAIT
TCP	192.168.0.103:64822	52.114.88.28:443	ESTABLISHED
TCP	192.168.0.103:64823	5.62.54.63:80	ESTABLISHED
TCP	192.168.0.103:64824	52.43.91.27:443	TIME_WAIT
TCP	192.168.0.103:64825	74.125.24.94:443	ESTABLISHED
TCP	192.168.0.103:64826	52.43.91.27:443	TIME_WAIT
TCP	192.168.0.103:64827	204.79.197.200:443	ESTABLISHED
TCP	192.168.0.103:64828	13.107.18.11:443	ESTABLISHED
TCP	192.168.0.103:64829	52.98.71.210:443	ESTABLISHED
TCP	192.168.0.103:64830	40.90.22.191:443	ESTABLISHED
TCP	192.168.0.103:64833	204.79.197.200:443	ESTABLISHED
TCP	192.168.0.103:64834	204.79.197.200:443	ESTABLISHED
TCP	192.168.0.103:64835	117.18.232.200:443	ESTABLISHED
TCP	192.168.0.103:64836	204.79.197.254:443	ESTABLISHED
TCP	192.168.0.103:64837	23.200.152.73:443	ESTABLISHED
TCP	192.168.0.103:64838	204.79.197.222:443	ESTABLISHED
TCP	192.168.0.103:64839	104.74.21.164:443	ESTABLISHED
TCP	192.168.0.103:64840	104.74.21.164:443	ESTABLISHED
TCP	192.168.0.103:64841	104.74.21.164:443	ESTABLISHED
TCP	192.168.0.103:64842	23.32.20.175:443	ESTABLISHED
TCP	192.168.0.103:64843	23.32.20.175:443	ESTABLISHED
TCP	192.168.0.103:64844	23.32.20.175:443	ESTABLISHED
TCP	192.168.0.103:64845	5.45.58.173:80	FIN_WAIT_2

C:\Users\Sarika>



Type here to search



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 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: `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 $n = 1, 2, 3, \dots$, 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 it 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).

Tracing route to iit.ac.in

tracert_iit - Notepad


File Edit Format View Help

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

1	*	2 ms	1 ms	192.168.0.1
2	5 ms	4 ms	4 ms	100.75.0.1
3	12 ms	8 ms	8 ms	mum-core01.youbroadband.in [203.187.217.163]
4	7 ms	5 ms	5 ms	118.185.45.34
5	8 ms	4 ms	5 ms	182.19.106.103
6	28 ms	27 ms	26 ms	14.142.18.97.static-Mumbai.vsnl.net.in [14.142.18.97]
7	27 ms	27 ms	27 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.

Tracing route to mscs.mu.edu

 tracert_mscs - Notepad

File Edit Format View Help

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

1	1 ms	1 ms	1 ms	192.168.0.1
2	94 ms	6 ms	5 ms	100.75.0.1
3	9 ms	10 ms	8 ms	mum-core01.youbroadband.in [203.187.217.163]
4	8 ms	4 ms	5 ms	118.185.45.34
5	110 ms	108 ms	135 ms	xe-8-3-2.mlu.cw.net [195.89.101.185]
6	198 ms	192 ms	193 ms	ae0-xcr1.mlb.cw.net [195.2.25.98]
7	193 ms	*	195 ms	ae4-pcr1.ptl.cw.net [195.2.9.89]
8	187 ms	189 ms	187 ms	et-7-1-0-xcr1.nyh.cw.net [195.2.24.241]
9	192 ms	192 ms	193 ms	ae3-xcr2.ash.cw.net [195.2.25.41]
10	193 ms	193 ms	195 ms	lag-16.ear1.WashingtonDC12.Level3.net [4.68.39.77]
11	*	*	*	Request timed out.
12	211 ms	212 ms	212 ms	MARQUETTE-U.ear3.Chicago2.Level3.net [4.16.38.70]
13	212 ms	212 ms	212 ms	134.48.10.26
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.

Tracing route to www.cs.grinnell.edu

tracert_grinnell - Notepad

File Edit Format View Help

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

1	2 ms	2 ms	1 ms	192.168.0.1
2	5 ms	4 ms	5 ms	100.75.0.1
3	9 ms	14 ms	23 ms	mum-core01.youbroadband.in [203.187.217.163]
4	5 ms	5 ms	5 ms	118.185.45.34
5	109 ms	109 ms	110 ms	xe-8-3-2.mlu.cw.net [195.89.101.185]
6	129 ms	129 ms	129 ms	mno-b2-link.telia.net [62.115.175.10]
7	221 ms	221 ms	221 ms	prs-bb3-link.telia.net [62.115.116.154]
8	220 ms	221 ms	220 ms	ldn-bb3-link.telia.net [62.115.134.93]
9	*	*	*	Request timed out.
10	210 ms	*	210 ms	chi-b23-link.telia.net [62.115.137.59]
11	219 ms	220 ms	220 ms	omha-b1-link.telia.net [62.115.143.183]
12	219 ms	219 ms	220 ms	aureon-ic-337963-omha-b1.c.telia.net [62.115.46.231]
13	280 ms	277 ms	276 ms	ins-oc4-lo0.omah.netins.net [167.142.66.77]
14	270 ms	270 ms	270 ms	ins-wc1-0-0-1-7.wdmn.netins.net [167.142.67.73]
15	270 ms	270 ms	274 ms	ins-wc2-et-0-0-1-6.wdmn.netins.net [167.142.67.85]
16	283 ms	282 ms	282 ms	167.142.58.40
17	274 ms	273 ms	273 ms	67.224.64.62
18	284 ms	283 ms	284 ms	grinnellcollege1.desm.netins.net [167.142.65.43]
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.

Tracing route to csail.mit.edu

tracert_csail - Notepad

File Edit Format View Help

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

1	1 ms	2 ms	1 ms	192.168.0.1
2	6 ms	5 ms	5 ms	100.75.0.1
3	13 ms	8 ms	8 ms	mum-core01.youbroadband.in [203.187.217.163]
4	5 ms	4 ms	6 ms	118.185.45.34
5	121 ms	120 ms	122 ms	xe-8-3-2.mlu.cw.net [195.89.101.185]
6	120 ms	120 ms	120 ms	ae0-xcr1.mlb.cw.net [195.2.25.98]
7	140 ms	140 ms	141 ms	be1274.rcr21.mil01.atlas.cogentco.com [130.117.14.25]
8	143 ms	143 ms	143 ms	be2194.ccr22.mrs01.atlas.cogentco.com [154.54.61.29]
9	151 ms	151 ms	151 ms	be3093.ccr42.par01.atlas.cogentco.com [130.117.50.165]
10	158 ms	158 ms	158 ms	be12489.ccr42.lon13.atlas.cogentco.com [154.54.57.69]
11	279 ms	280 ms	279 ms	be2101.ccr32.bos01.atlas.cogentco.com [154.54.82.38]
12	283 ms	282 ms	283 ms	38.104.186.186
13	275 ms	275 ms	275 ms	dmz-rtr-1-external-rtr-3.mit.edu [18.0.161.13]
14	282 ms	282 ms	282 ms	dmz-rtr-2-dmz-rtr-1-1.mit.edu [18.0.161.6]
15	280 ms	280 ms	280 ms	mitnet.core-1-ext.csail.mit.edu [18.4.7.65]
16	*	*	*	Request timed out.
17	281 ms	282 ms	281 ms	bdr.core-1.csail.mit.edu [128.30.0.246]
18	275 ms	274 ms	275 ms	inquir-3ld.csail.mit.edu [128.30.2.109]

Trace complete.

Tracing route to cs.stanford.edu

tracert_stanford - Notepad

File Edit Format View Help

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

1	1 ms	1 ms	1 ms	192.168.0.1
2	38 ms	7 ms	4 ms	100.75.0.1
3	9 ms	8 ms	10 ms	mum-core01.youbroadband.in [203.187.217.163]
4	5 ms	4 ms	4 ms	118.185.45.34
5	109 ms	109 ms	110 ms	xe-8-3-2.mlu.cw.net [195.89.101.185]
6	118 ms	118 ms	118 ms	ae6-xcr1.fix.cw.net [195.2.10.245]
7	119 ms	118 ms	118 ms	as6939-gw-xcr1.fix.cw.net [195.2.19.30]
8	137 ms	137 ms	140 ms	100ge8-1.core1.fra1.he.net [184.104.195.13]
9	120 ms	119 ms	119 ms	100ge1-1.core1.par2.he.net [72.52.92.13]
10	196 ms	196 ms	196 ms	100ge10-2.core1.ash1.he.net [184.105.213.173]
11	259 ms	257 ms	257 ms	100ge7-2.core1.pao1.he.net [184.105.222.41]
12	255 ms	257 ms	256 ms	stanford-university.100gigabitethernet5-1.core1.pao1.he.net [184.105.177.238]
13	275 ms	275 ms	282 ms	csee-west-rtr-vl3.SUNet [171.66.255.140]
14	279 ms	281 ms	280 ms	CS.stanford.edu [171.64.64.64]

Trace complete.

Tracing route to cs.manchester.ac.uk

tracert_manchester - 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	2 ms	2 ms	192.168.0.1
2	5 ms	5 ms	5 ms	100.75.0.1
3	34 ms	11 ms	9 ms	mum-core01.youbroadband.in [203.187.217.163]
4	6 ms	6 ms	6 ms	118.185.45.34
5	123 ms	124 ms	122 ms	xe-8-3-2.mlu.cw.net [195.89.101.185]
6	141 ms	142 ms	142 ms	mno-b2-link.telia.net [62.115.175.10]
7	169 ms	169 ms	*	prs-bb4-link.telia.net [62.115.116.168]
8	167 ms	*	*	ldn-bb4-link.telia.net [62.115.114.228]
9	*	261 ms	169 ms	ldn-b2-link.telia.net [62.115.120.239]
10	136 ms	136 ms	136 ms	jisc-ic-345131-ldn-b4.c.telia.net [62.115.175.131]
11	137 ms	137 ms	137 ms	ae24.londhx-sbr1.ja.net [146.97.35.197]
12	138 ms	137 ms	141 ms	ae29.londpg-sbr2.ja.net [146.97.33.2]
13	141 ms	141 ms	141 ms	ae31.erdiss-sbr2.ja.net [146.97.33.22]
14	142 ms	142 ms	142 ms	ae29.manckh-sbr2.ja.net [146.97.33.42]
15	142 ms	142 ms	143 ms	ae23.mancrh-rbr1.ja.net [146.97.38.42]
16	143 ms	*	*	universityofmanchester.ja.net [146.97.169.2]
17	144 ms	145 ms	143 ms	130.88.249.194
18	*	*	*	Request timed out.
19	*	*	*	Request timed out.
20	143 ms	143 ms	143 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\Sarika>tracert math.hws.edu
```

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

1	2 ms	3 ms	2 ms	192.168.0.1
2	45 ms	4 ms	5 ms	100.75.0.1
3	9 ms	13 ms	8 ms	mum-core01.youbroadband.in [203.187.217.163]
4	10 ms	9 ms	10 ms	118.185.45.34
5	112 ms	113 ms	114 ms	xe-8-3-2.mlu.cw.net [195.89.101.185]
6	200 ms	196 ms	196 ms	ae0-xcr1.mlb.cw.net [195.2.25.98]
7	*	*	*	Request timed out.
8	190 ms	189 ms	189 ms	et-7-1-0-xcr1.nyh.cw.net [195.2.24.241]
9	195 ms	195 ms	195 ms	ae3-xcr2.ash.cw.net [195.2.25.41]
10	195 ms	195 ms	196 ms	lag-16.ear1.WashingtonDC12.Level3.net [4.68.39.77]
11	*	200 ms	199 ms	ae-1-3502.ear3.Washington1.Level3.net [4.69.206.77]
12	215 ms	196 ms	210 ms	4.68.72.61
13	276 ms	209 ms	249 ms	roc1-ar5-xe-11-0-0-0.us.twtelecom.net [35.248.1.162]
14	211 ms	211 ms	211 ms	66-195-65-170.static.ctl.one [66.195.65.170]
15	213 ms	211 ms	212 ms	nat.hws.edu [64.89.144.100]
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.
```

Command Prompt

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

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

  1    3 ms    2 ms    2 ms  192.168.0.1
  2    3 ms    5 ms    4 ms  100.75.0.1
  3    9 ms    8 ms    7 ms  mum-core01.youbroadband.in [203.187.217.163]
  4    8 ms    8 ms    7 ms  118.185.45.34
  5   112 ms   112 ms   112 ms xe-8-3-2.mlu.cw.net [195.89.101.185]
  6   195 ms   195 ms   195 ms ae0-xcr1.mlb.cw.net [195.2.25.98]
  7   208 ms   195 ms   196 ms ae4-pcr1.ptl.cw.net [195.2.9.89]
  8   191 ms   189 ms   190 ms et-7-1-0-xcr1.nyh.cw.net [195.2.24.241]
  9   195 ms   195 ms   195 ms ae3-xcr2.ash.cw.net [195.2.25.41]
 10    *      *      195 ms lag-16.ear1.WashingtonDC12.Level3.net [4.68.39.77]
 11    *      *      *      Request timed out.
 12    *      *      *      Request timed out.
 13   209 ms   209 ms   211 ms roc1-ar5-xe-11-0-0-0.us.twtelecom.net [35.248.1.162]
 14   212 ms   214 ms   213 ms 66-195-65-170.static.clt.one [66.195.65.170]
 15   211 ms   211 ms   213 ms nat.hws.edu [64.89.144.100]
 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\Sarika>
```

Observation -

From the above results, we can see that the path followed is the same till the 15th hop, but some requests timed out in one, but got completed in the other. After the 15th hop, all the requests time out in 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.

Tracing route to google.com on Thursday, 13th August, 5pm

Select Command Prompt

```
Packets: Sent = 10, Received = 0, Lost = 10 (100% loss),

C:\Users\Sarika>tracert google.com

Tracing route to google.com [74.125.130.102]
over a maximum of 30 hops:

  1    2 ms    2 ms    2 ms  192.168.0.1
  2    4 ms    4 ms    4 ms  100.75.0.1
  3    9 ms   11 ms    8 ms  mum-core01.youbroadband.in [203.187.217.163]
  4    8 ms    6 ms    5 ms  54-217-187-203.static.youbroadband.in [203.187.217.54]
  5    5 ms    5 ms    5 ms  209.85.175.108
  6    7 ms    8 ms    8 ms  108.170.248.163
  7   67 ms   66 ms    *    108.170.225.145
  8   66 ms   65 ms   65 ms  72.14.236.223
  9   65 ms   65 ms   65 ms  108.170.230.239
 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   64 ms   64 ms   63 ms  sb-in-f102.1e100.net [74.125.130.102]

Trace complete.

C:\Users\Sarika>
```

Tracing route to google.com on Wednesday, 19th August, 1pm

```
Tracing route to google.com [172.217.31.110]
over a maximum of 30 hops:

  1    1 ms    1 ms    1 ms  192.168.0.1
  2    4 ms    4 ms    4 ms  100.75.0.1
  3    8 ms    7 ms    7 ms  mum-core01.youbroadband.in [203.187.217.163]
  4    7 ms    6 ms    5 ms  54-217-187-203.static.youbroadband.in [203.187.217.54]
  5    6 ms    5 ms    5 ms  209.85.175.108
  6    5 ms   10 ms    5 ms  108.170.248.179
  7   64 ms    *    64 ms  108.170.229.13
  8   70 ms   70 ms   71 ms  66.249.94.208
  9   71 ms   72 ms   77 ms  108.170.250.17
 10   71 ms   71 ms   72 ms  108.170.230.101
 11   72 ms   76 ms   72 ms  kul08s08-in-f14.1e100.net [172.217.31.110]

Trace complete.
```

From the above outputs, we can conclude that the path taken to the same host at different times can be different. The number of hops and final IP is also different.

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 ISP is always the 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?

No, there is no correlation between number of nodes that show up in traceroute and the location of the 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?

The further apart two nodes are the more latency there is as latency is dependent on the distance between the two communicating nodes. Theoretically, latency of a packet going on a round trip across the world is 133ms. In actuality, such a round trip takes longer, though latency is decreased when direct connections through network backbones are achieved. When the source and destination are far apart, the hops increase. Due to high number of nodes, the latency adds up and increases the RTT.

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`. *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.

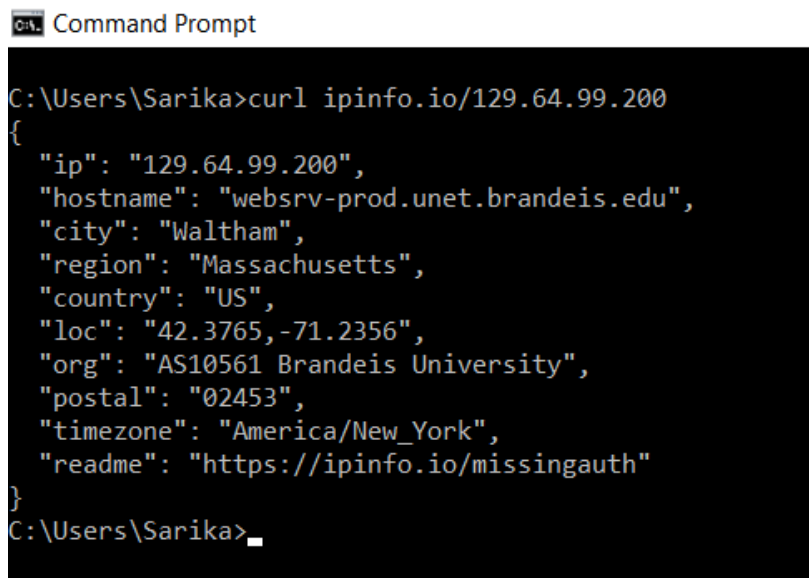
Exercise 5: (Should be short.) Because of NAT, the domain name *spit.ac.in* has a different IP address outside of SPIT than it does on campus. Using information in this lab and working on a home computer, find the outside IP address for *spit.ac.in*. Explain how you did it.

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
```

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

A screenshot of a Windows Command Prompt window. The title bar reads "C:_ Command Prompt". The command prompt shows the user "Sarika" at the "C:\Users\Sarika" directory. The command entered is `curl ipinfo.io/129.64.99.200`. The output is a JSON object containing various details about the IP address 129.64.99.200, including its hostname, city, region, country, location coordinates, organization, postal code, timezone, and a link to a README file.

```
C:\Users\Sarika>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\Sarika>
```

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-

In this experiment, I have learnt about some basic command line networking utilities like ping, tracert and ifconfig. I observed the outputs and studied the factors that affect round trip times and network latency factors.

References-

https://en.wikipedia.org/wiki/Network_delay#:~:text=Processing%20delay%20%E2%80%93%20time%20it%20takes,signal%20to%20reach%20its%20destination

<https://www.imperva.com/learn/performance/round-trip-time-rtt/>

<https://blog.stackpath.com/latency/#:~:text=Propagation%3A%20The%20further%20apart%20two,between%20the%20two%20communicating%20nodes.&text=In%20actuality%2C%20such%20a%20round,through%20network%20backbones%20are%20achieved.>