Harshwardhan Pardeshi
2018130034
Batch C

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

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

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

## Some Basic command line Networking utilities

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

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

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

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

The syntax in Windows is:

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

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

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

```
ping -c 10 google.com > ping_c10_s64_google.log
```

**EXPERIMENTS WITH PING**

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

```
C:\Users\jaswa>ping -n 10 -l 64 www.uw.edu

Pinging www.washington.edu [128.95.155.135] with 64 bytes of data:
Reply from 128.95.155.135: bytes=64 time=331ms TTL=47
Reply from 128.95.155.135: bytes=64 time=245ms TTL=47
Reply from 128.95.155.135: bytes=64 time=254ms TTL=47
Reply from 128.95.155.135: bytes=64 time=272ms TTL=47
Reply from 128.95.155.135: bytes=64 time=285ms TTL=47
Reply from 128.95.155.135: bytes=64 time=304ms TTL=47
Reply from 128.95.155.135: bytes=64 time=268ms TTL=47
Reply from 128.95.155.135: bytes=64 time=244ms TTL=47
Reply from 128.95.155.135: bytes=64 time=244ms TTL=47
Reply from 128.95.155.135: bytes=64 time=244ms TTL=47

Ping statistics for 128.95.155.135:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 244ms, Maximum = 331ms, Average = 269ms
```

```
C:\Users\jaswa>ping -n 10 -l 100 www.uw.edu

Pinging www.washington.edu [128.95.155.197] with 100 bytes of data:
Reply from 128.95.155.197: bytes=100 time=275ms TTL=47
Reply from 128.95.155.197: bytes=100 time=240ms TTL=47
Reply from 128.95.155.197: bytes=100 time=301ms TTL=47
Reply from 128.95.155.197: bytes=100 time=241ms TTL=47
Reply from 128.95.155.197: bytes=100 time=322ms TTL=47
Reply from 128.95.155.197: bytes=100 time=329ms TTL=47
Reply from 128.95.155.197: bytes=100 time=347ms TTL=47
Reply from 128.95.155.197: bytes=100 time=257ms TTL=47
Reply from 128.95.155.197: bytes=100 time=240ms TTL=47
Reply from 128.95.155.197: bytes=100 time=271ms TTL=47

Ping statistics for 128.95.155.197:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 240ms, Maximum = 347ms, Average = 282ms
```

```
C:\Users\jaswa>ping -n 10 -l 500 www.uw.edu

Pinging www.washington.edu [128.95.155.134] with 500 bytes of data:
Reply from 128.95.155.134: bytes=500 time=268ms TTL=47
Reply from 128.95.155.134: bytes=500 time=276ms TTL=47
Reply from 128.95.155.134: bytes=500 time=243ms TTL=47
Reply from 128.95.155.134: bytes=500 time=247ms TTL=47
Reply from 128.95.155.134: bytes=500 time=242ms TTL=47
Reply from 128.95.155.134: bytes=500 time=240ms TTL=47
Reply from 128.95.155.134: bytes=500 time=240ms TTL=47
Reply from 128.95.155.134: bytes=500 time=241ms TTL=47
Reply from 128.95.155.134: bytes=500 time=346ms TTL=47
Reply from 128.95.155.134: bytes=500 time=241ms TTL=47

Ping statistics for 128.95.155.134:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 240ms, Maximum = 346ms, Average = 258ms
```

```
C:\Users\jaswa>ping -n 10 -l 1000 www.uw.edu

Pinging www.washington.edu [128.95.155.134] with 1000 bytes of data:
Reply from 128.95.155.134: bytes=1000 time=296ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=303ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=314ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=267ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=326ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=242ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=241ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=256ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=265ms TTL=47
Reply from 128.95.155.134: bytes=1000 time=273ms TTL=47

Ping statistics for 128.95.155.134:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 241ms, Maximum = 326ms, Average = 278ms
```

```
C:\Users\jaswa>ping -n 10 -l 1400 www.uw.edu

Pinging www.washington.edu [128.95.155.198] with 1400 bytes of data:
Reply from 128.95.155.198: bytes=1400 time=292ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=310ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=246ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=309ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=246ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=245ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=352ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=354ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=248ms TTL=47
Reply from 128.95.155.198: bytes=1400 time=267ms TTL=47

Ping statistics for 128.95.155.198:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 245ms, Maximum = 354ms, Average = 286ms
```

**QUESTIONS ABOUT LATENCY**

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

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

Ans. Yes, the average RTT varies between different hosts.

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

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

Ans. Yes the average RTT varies with different packet sizes. The differences are likely caused by transmit delay

**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).

>> Infrastructure components, network traffic, and physical distance along the path between a source and a destination are all potential factors that can affect 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.

```
C:\Users\jaswa>ping -n 10 -l 1400 berkeley.edu

Pinging berkeley.edu [35.163.72.93] with 1400 bytes of data:
Reply from 35.163.72.93: bytes=1400 time=299ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=307ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=313ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=309ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=272ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=329ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=339ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=272ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=272ms TTL=33
Reply from 35.163.72.93: bytes=1400 time=272ms TTL=33

Ping statistics for 35.163.72.93:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 272ms, Maximum = 339ms, Average = 298ms
```

```
C:\Users\jaswa>ping -n 10 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),
Control-C
^C
```

```
C:\Users\jaswa>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=7ms 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=5ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=5ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=5ms TTL=59
Request timed out.
Reply from 151.101.66.133: bytes=1400 time=9ms TTL=59
Reply from 151.101.66.133: bytes=1400 time=5ms TTL=59

Ping statistics for 151.101.66.133:
    Packets: Sent = 10, Received = 9, Lost = 1 (10% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 11ms, Average = 6ms
```

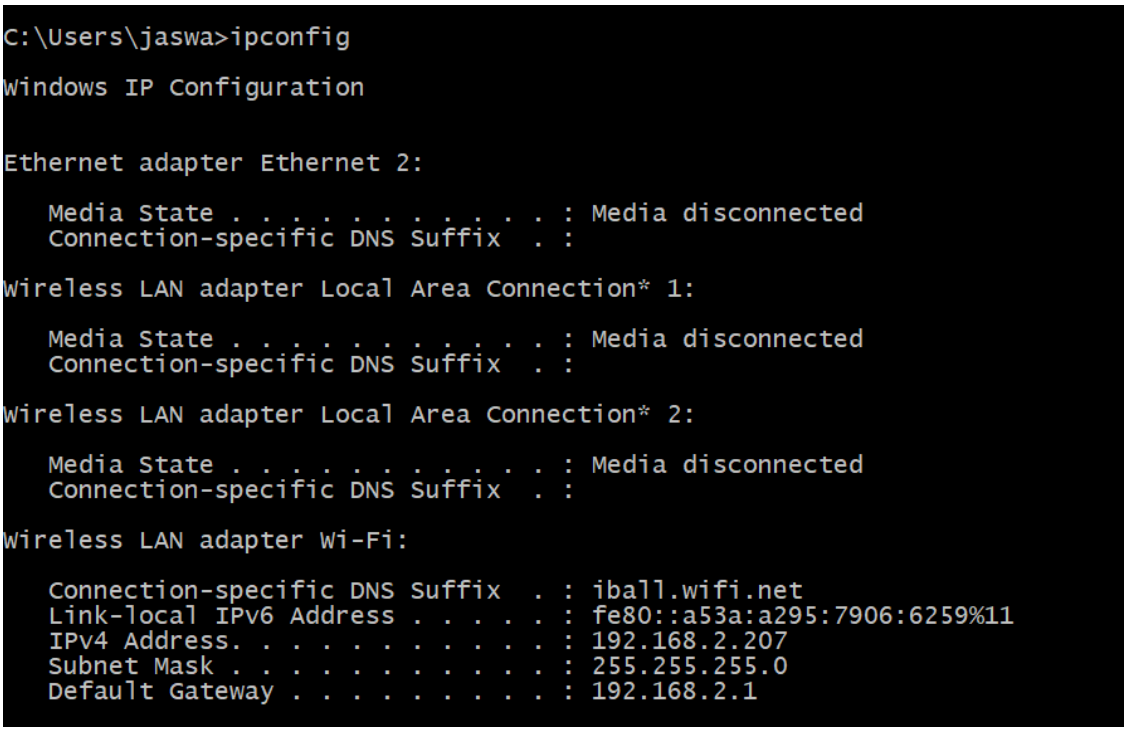
```
C:\Users\jaswa>ping -n 10 -l 1400 www.uchicago.edu

Pinging wsee2.elb.uchicago.edu [54.89.29.50] 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.

Ping statistics for 54.89.29.50:
    Packets: Sent = 7, Received = 0, Lost = 7 (100% loss),
Control-C
^C
```

```
C:\Users\jaswa>ping -n 10 -l 1400 www.u-tokyo.ac.jp

Pinging www.u-tokyo.ac.jp [210.152.243.234] 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.

Ping statistics for 210.152.243.234:
    Packets: Sent = 9, Received = 0, Lost = 9 (100% loss),
Control-C
^C
```

**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>

```
C:\Users\jaswa>nslookup amazon.com
Server:  UnKnown
Address:  192.168.2.1

Non-authoritative answer:
Name:    amazon.com
Addresses:  205.251.242.103
          176.32.103.205
          176.32.98.166
```

**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!!!.)




**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

**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\jaswa>netstat -t -n

Active Connections

  Proto  Local Address          Foreign Address        State           Offload State
  TCP    127.0.0.1:49677        127.0.0.1:49678        ESTABLISHED     InHost
  TCP    127.0.0.1:49678        127.0.0.1:49677        ESTABLISHED     InHost
  TCP    127.0.0.1:51255        127.0.0.1:58583        ESTABLISHED     InHost
  TCP    127.0.0.1:58583        127.0.0.1:51255        ESTABLISHED     InHost
  TCP    192.168.2.207:58538    5.62.54.42:80          ESTABLISHED     InHost
  TCP    192.168.2.207:58585    74.125.24.188:5228     ESTABLISHED     InHost
  TCP    192.168.2.207:58708    52.139.250.253:443     ESTABLISHED     InHost
  TCP    192.168.2.207:58713    52.194.132.91:443      ESTABLISHED     InHost
  TCP    192.168.2.207:59179    104.110.244.10:443     CLOSE_WAIT      InHost
  TCP    192.168.2.207:59180    93.184.220.29:80       CLOSE_WAIT      InHost
  TCP    192.168.2.207:60475    54.147.43.175:443      ESTABLISHED     InHost
  TCP    192.168.2.207:61040    5.45.59.35:80          ESTABLISHED     InHost
  TCP    192.168.2.207:61322    3.7.171.148:443        ESTABLISHED     InHost
  TCP    192.168.2.207:61791    52.5.194.233:443       ESTABLISHED     InHost
  TCP    192.168.2.207:61821    40.67.251.132:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62245    150.136.156.92:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62247    35.244.159.8:443       ESTABLISHED     InHost
  TCP    192.168.2.207:62248    74.118.186.210:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62251    88.221.41.31:443       ESTABLISHED     InHost
  TCP    192.168.2.207:62256    95.217.58.90:443       ESTABLISHED     InHost
  TCP    192.168.2.207:62259    52.41.11.190:443       ESTABLISHED     InHost
  TCP    192.168.2.207:62267    64.233.177.120:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62276    216.58.206.34:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62283    216.58.208.142:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62298    2.20.36.48:443         ESTABLISHED     InHost
  TCP    192.168.2.207:62300    51.138.106.75:443      TIME_WAIT       InHost
  TCP    192.168.2.207:62305    150.136.156.92:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62315    106.10.218.42:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62316    172.217.160.196:443    ESTABLISHED     InHost
  TCP    192.168.2.207:62319    216.58.209.54:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62324    172.217.167.174:443    ESTABLISHED     InHost
  TCP    192.168.2.207:62325    216.58.206.34:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62327    172.217.167.162:443    ESTABLISHED     InHost
  TCP    192.168.2.207:62329    216.58.198.46:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62331    216.58.198.46:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62335    54.197.49.8:443        ESTABLISHED     InHost
  TCP    192.168.2.207:62338    37.252.161.190:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62340    158.69.124.122:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62344    216.58.208.138:443     ESTABLISHED     InHost
  TCP    192.168.2.207:62345    216.58.208.138:443     TIME_WAIT       InHost
  TCP    192.168.2.207:62346    216.58.205.67:443      ESTABLISHED     InHost
  TCP    192.168.2.207:62347    216.58.205.67:443      TIME_WAIT       InHost
  TCP    192.168.2.207:62348    77.234.45.211:80       TIME_WAIT       InHost
```

**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:

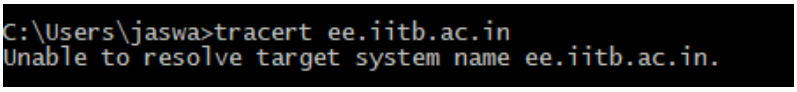
```
tracert <hostname>
```

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

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

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

  1    13 ms     6 ms     5 ms  192.168.2.1
  2     3 ms     3 ms     4 ms  103.145.18.242
  3     2 ms     2 ms     5 ms  103.145.18.241
  4     6 ms     6 ms     4 ms  static-189.122.143.114.in-addr.arpa [114.143.122.189]
  5     5 ms     4 ms     4 ms  static-10.79.156.182-tataidc.co.in [182.156.79.10]
  6     5 ms     4 ms     6 ms  10.117.137.146
  7     4 ms     4 ms     5 ms  14.141.63.225.static-Mumbai.vsnl.net.in [14.141.63.225]
  8     *        *        *     Request timed out.
  9     4 ms     5 ms     5 ms  ix-ae-0-100.tcore1.mlv-mumbai.as6453.net [180.87.38.5]
 10   114 ms   116 ms   113 ms  if-ae-5-6.tcore1.wyn-marseille.as6453.net [180.87.38.126]
 11   115 ms   113 ms   112 ms  if-ae-21-2.tcore1.pye-paris.as6453.net [80.231.154.208]
 12   112 ms   118 ms   112 ms  if-ae-11-2.tcore1.pvu-paris.as6453.net [80.231.153.49]
 13     *        *        *     Request timed out.
 14   219 ms   219 ms   220 ms  ae-2-3603.ear3.Chicago2.Level3.net [4.69.159.186]
 15   222 ms   220 ms   220 ms  MARQUETTE-U.ear3.Chicago2.Level3.net [4.16.38.70]
 16   225 ms   225 ms   225 ms  134.48.10.26
 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.
```

3. www.cs.grinnell.edu

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

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

  1     3 ms     2 ms     5 ms  192.168.2.1
  2     3 ms     2 ms     2 ms  103.145.18.242
  3     2 ms     2 ms     2 ms  103.145.18.241
  4     4 ms     3 ms     4 ms  static-189.122.143.114.in-addr.arpa [114.143.122.189]
  5     4 ms    42 ms    43 ms  static-10.79.156.182-tataidc.co.in [182.156.79.10]
  6     4 ms     4 ms     5 ms  10.117.137.146
  7     5 ms     7 ms     6 ms  14.141.63.225.static-Mumbai.vsnl.net.in [14.141.63.225]
  8     *        *        *     Request timed out.
  9     *        *        *     Request timed out.
 10    49 ms    26 ms    27 ms  ix-ae-4-2.tcore2.cxr-chennai.as6453.net [180.87.37.1]
 11   246 ms   246 ms   243 ms  if-ae-9-2.tcore2.mlv-mumbai.as6453.net [180.87.37.10]
 12   244 ms   242 ms   242 ms  if-ae-2-2.tcore1.mlv-mumbai.as6453.net [180.87.38.1]
 13   245 ms     *      245 ms  if-ae-29-8.tcore1.wyn-marseille.as6453.net [80.231.217.110]
 14   242 ms     *      241 ms  if-ae-2-2.tcore2.wyn-marseille.as6453.net [80.231.217.2]
 15   243 ms     *      244 ms  if-ae-9-2.tcore2.l78-london.as6453.net [80.231.200.14]
 16   244 ms   244 ms   245 ms  if-ae-15-2.tcore2.ldn-london.as6453.net [80.231.131.118]
 17   246 ms   245 ms   246 ms  if-ae-32-2.tcore2.nto-newyork.as6453.net [63.243.216.22]
 18   244 ms   243 ms   242 ms  if-ae-26-2.tcore1.ct8-chicago.as6453.net [216.6.81.29]
 19     *        *      244 ms  63.243.129.121
 20     *        *      309 ms  gi0-0-0-3.agr02.mtld01-fl.us.windstream.net [169.130.82.82]
 21   255 ms   252 ms   252 ms  et3-1-0-0.agr03.desm01-ia.us.windstream.net [40.128.250.43]
 22   255 ms   255 ms   256 ms  ae4-0.pe04.grnl01-ia.us.windstream.net [40.128.248.35]
 23   254 ms   255 ms   254 ms  ae7-0.pe05.grnl01-ia.us.windstream.net [40.138.127.29]
 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.
```

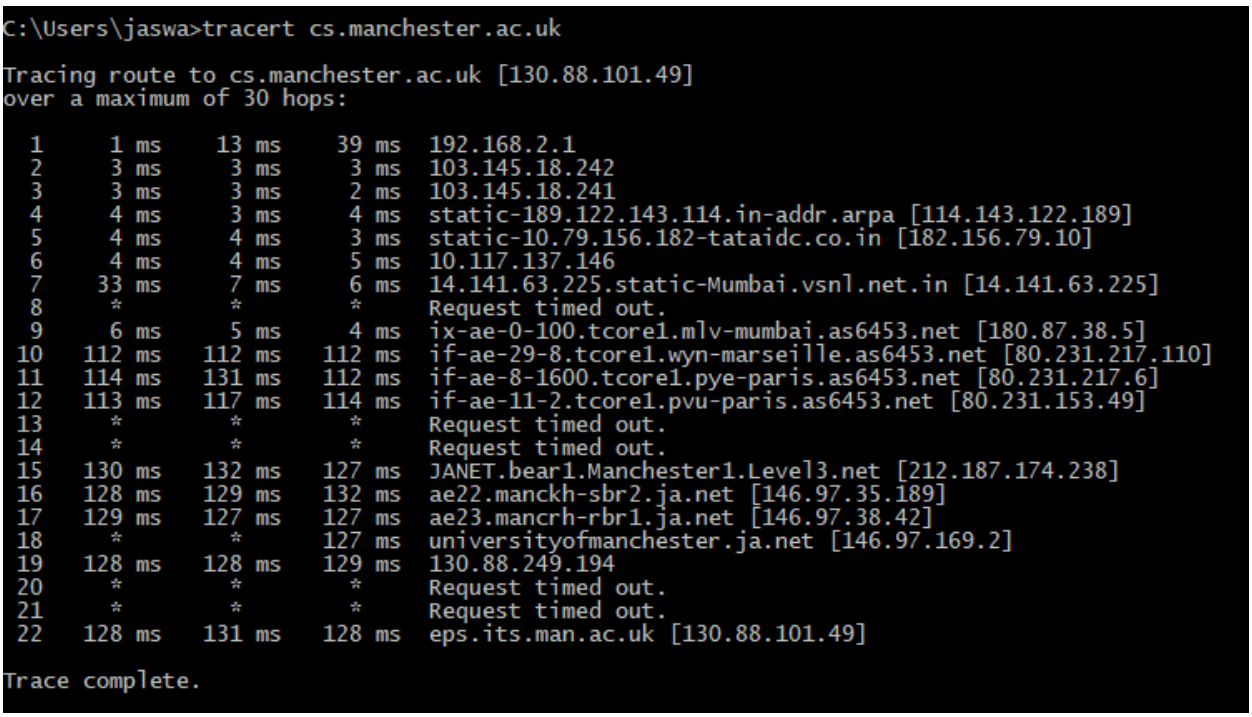
4. csail.mit.edu

```
C:\Users\jaswa>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  192.168.2.1
  2     3 ms     2 ms     2 ms  103.145.18.242
  3     3 ms     3 ms     3 ms  103.145.18.241
  4     4 ms     3 ms     4 ms  static-189.122.143.114.in-addr.arpa [114.143.122.189]
  5     6 ms     6 ms     4 ms  static-10.79.156.182-tataidc.co.in [182.156.79.10]
  6     4 ms     4 ms     4 ms  10.117.137.146
  7     7 ms     5 ms     5 ms  14.141.63.225.static-Mumbai.vsnl.net.in [14.141.63.225]
  8     *        *        *     Request timed out.
  9     6 ms     4 ms     4 ms  ix-ae-0-100.tcore1.mlv-mumbai.as6453.net [180.87.38.5]
 10   228 ms   200 ms   200 ms  if-ae-5-2.tcore1.wyn-marseille.as6453.net [80.231.217.29]
 11   200 ms   202 ms   200 ms  if-ae-2-2.tcore2.wyn-marseille.as6453.net [80.231.217.2]
 12   200 ms   203 ms   202 ms  if-ae-9-2.tcore2.l78-london.as6453.net [80.231.200.14]
 13   201 ms   202 ms   201 ms  if-ae-15-2.tcore2.ldn-london.as6453.net [80.231.131.118]
 14   206 ms   203 ms   204 ms  if-ae-32-2.tcore2.nto-newyork.as6453.net [63.243.216.22]
 15   201 ms   201 ms   202 ms  if-ae-12-2.tcore1.n75-newyork.as6453.net [66.110.96.5]
 16   201 ms   202 ms   200 ms  66.110.96.146
 17   203 ms   204 ms   202 ms  be-10390-cr02.newyork.ny.ibone.comcast.net [68.86.83.89]
 18   201 ms   202 ms   202 ms  be-1402-cs04.newyork.ny.ibone.comcast.net [96.110.38.45]
 19   208 ms   206 ms   206 ms  96.110.42.14
 20   208 ms   213 ms   224 ms  ae0-0-eg-bstpmall74w.boston.ma.boston.comcast.net [68.86.238.34]
 21   207 ms   207 ms   208 ms  50-201-57-174-static.hfc.comcastbusiness.net [50.201.57.174]
 22   207 ms   207 ms   209 ms  dmz-rtr-1-external-rtr-3.mit.edu [18.0.161.13]
 23   207 ms   208 ms   211 ms  dmz-rtr-2-dmz-rtr-1-1.mit.edu [18.0.161.6]
 24   209 ms   207 ms   209 ms  mitnet.core-1-ext.csail.mit.edu [18.4.7.65]
 25     *        *        *     Request timed out.
 26   211 ms   208 ms   210 ms  bdr.core-1.csail.mit.edu [128.30.0.246]
 27   214 ms   210 ms   210 ms  inquir-3ld.csail.mit.edu [128.30.2.109]

Trace complete.
```

5. cs.stanford.edu

```
C:\Users\jaswa>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     1 ms  192.168.2.1
  2     4 ms     3 ms     2 ms  103.145.18.242
  3     2 ms     2 ms     3 ms  103.145.18.241
  4     4 ms     5 ms     5 ms  static-189.122.143.114.in-addr.arpa [114.143.122.189]
  5     3 ms     4 ms     4 ms  static-10.79.156.182-tataidc.co.in [182.156.79.10]
  6     4 ms     6 ms     5 ms  10.117.137.146
  7     5 ms     5 ms     5 ms  14.141.63.225.static-Mumbai.vsnl.net.in [14.141.63.225]
  8     *        *        *     Request timed out.
  9     *        *        *     Request timed out.
 10    21 ms    22 ms    21 ms  ix-ae-4-2.tcore2.cxr-chennai.as6453.net [180.87.37.1]
 11   259 ms   235 ms   234 ms  if-ae-10-4.tcore2.svw-singapore.as6453.net [180.87.67.16]
 12   245 ms   240 ms   240 ms  if-ae-7-2.tcore2.lvw-losangeles.as6453.net [180.87.15.26]
 13   244 ms   243 ms   247 ms  if-ae-2-2.tcore1.lvw-losangeles.as6453.net [66.110.59.1]
 14   244 ms   245 ms   240 ms  las-b24-link.telia.net [80.239.128.214]
 15   383 ms   254 ms   416 ms  palo-b24-link.telia.net [62.115.119.90]
 16   319 ms   319 ms   303 ms  palo-b1-link.telia.net [62.115.122.169]
 17   359 ms   260 ms   261 ms  hurricane-ic-308019-palo-b1.c.telia.net [80.239.167.174]
 18   248 ms   248 ms   248 ms  stanford-university.100gigabitethernet5-1.core1.pao1.he.net [184.105.177.238]
 19   253 ms   252 ms   252 ms  csee-west-rtr-vl3.SUNet [171.66.255.140]
 20   253 ms   254 ms   254 ms  CS.stanford.edu [171.64.64.64]

Trace complete.
```

6. cs.manchester.ac.uk



```
C:\Users\jaswa>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    13 ms    39 ms  192.168.2.1
  2     3 ms     3 ms     3 ms  103.145.18.242
  3     3 ms     3 ms     2 ms  103.145.18.241
  4     4 ms     3 ms     4 ms  static-189.122.143.114.in-addr.arpa [114.143.122.189]
  5     4 ms     4 ms     3 ms  static-10.79.156.182-tataidc.co.in [182.156.79.10]
  6     4 ms     4 ms     5 ms  10.117.137.146
  7    33 ms     7 ms     6 ms  14.141.63.225.static-Mumbai.vsnl.net.in [14.141.63.225]
  8     *        *        *     Request timed out.
  9     6 ms     5 ms     4 ms  ix-ae-0-100.tcore1.mlv-mumbai.as6453.net [180.87.38.5]
 10   112 ms   112 ms   112 ms  if-ae-29-8.tcore1.wyn-marseille.as6453.net [80.231.217.110]
 11   114 ms   131 ms   112 ms  if-ae-8-1600.tcore1.pye-paris.as6453.net [80.231.217.6]
 12   113 ms   117 ms   114 ms  if-ae-11-2.tcore1.pvu-paris.as6453.net [80.231.153.49]
 13     *        *        *     Request timed out.
 14     *        *        *     Request timed out.
 15   130 ms   132 ms   127 ms  JANET.bear1.Manchester1.Level3.net [212.187.174.238]
 16   128 ms   129 ms   132 ms  ae22.manckh-sbr2.ja.net [146.97.35.189]
 17   129 ms   127 ms   127 ms  ae23.mancrh-rbr1.ja.net [146.97.38.42]
 18     *        *      127 ms  universityofmanchester.ja.net [146.97.169.2]
 19   128 ms   128 ms   129 ms  130.88.249.194
 20     *        *        *     Request timed out.
 21     *        *        *     Request timed out.
 22   128 ms   131 ms   128 ms  eps.its.man.ac.uk [130.88.101.49]

Trace complete.
```

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`).

**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\jaswa>tracert math.hws.edu

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

  1     1 ms     1 ms     1 ms  192.168.2.1
  2     3 ms     3 ms     3 ms  103.145.18.242
  3     3 ms     2 ms     2 ms  103.145.18.241
  4     5 ms     5 ms     5 ms  static-189.122.143.114.in-addr.arpa [114.143.122.189]
  5    25 ms     3 ms     8 ms  static-10.79.156.182-tataidc.co.in [182.156.79.10]
  6     6 ms     6 ms    19 ms  10.117.137.146
  7    18 ms    17 ms     4 ms  14.141.63.225.static-Mumbai.vsnl.net.in [14.141.63.225]
  8     *        *        *     Request timed out.
  9    25 ms     5 ms     5 ms  ix-ae-0-100.tcore1.mlv-mumbai.as6453.net [180.87.38.5]
 10     *        *      113 ms  if-ae-5-2.tcore1.wyn-marseille.as6453.net [80.231.217.29]
 11   120 ms   147 ms   118 ms  if-ae-21-2.tcore1.pye-paris.as6453.net [80.231.154.208]
 12   113 ms   135 ms   113 ms  if-ae-11-2.tcore1.pvu-paris.as6453.net [80.231.153.49]
 13     *        *      115 ms  80.231.153.66
 14   113 ms   115 ms   112 ms  ae-2-3204.edge3.Paris1.Level3.net [4.69.161.114]
 15   121 ms   114 ms   112 ms  global-crossing-xe-level3.paris1.level3.net [4.68.63.230]
 16   207 ms   223 ms   207 ms  roc1-ar5-xe-11-0-0-0.us.twtelecom.net [35.248.1.162]
 17   209 ms   208 ms   217 ms  66-195-65-170.static.ctl.one [66.195.65.170]
 18   208 ms   209 ms   212 ms  nat.hws.edu [64.89.144.100]
 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\jaswa>tracert www.hws.edu

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

  1     1 ms     2 ms     1 ms  192.168.2.1
  2     8 ms     2 ms     2 ms  103.145.18.242
  3     3 ms     3 ms     2 ms  103.145.18.241
  4     5 ms     4 ms     4 ms  static-189.122.143.114.in-addr.arpa [114.143.122.189]
  5     5 ms     4 ms     4 ms  static-10.79.156.182-tataidc.co.in [182.156.79.10]
  6     7 ms     5 ms     4 ms  10.117.137.146
  7     6 ms     4 ms     4 ms  14.141.63.225.static-Mumbai.vsnl.net.in [14.141.63.225]
  8     *        *        *     Request timed out.
  9     4 ms     7 ms     5 ms  ix-ae-0-100.tcore1.mlv-mumbai.as6453.net [180.87.38.5]
 10     *        *        *     Request timed out.
 11   117 ms   113 ms   124 ms  if-ae-8-1600.tcore1.pye-paris.as6453.net [80.231.217.6]
 12   112 ms   116 ms   136 ms  if-ae-11-2.tcore1.pvu-paris.as6453.net [80.231.153.49]
 13     *        *        *     Request timed out.
 14   114 ms   114 ms   113 ms  ae-2-3204.edge3.Paris1.Level3.net [4.69.161.114]
 15   115 ms   113 ms   114 ms  global-crossing-xe-level3.paris1.level3.net [4.68.63.230]
 16   206 ms   208 ms   206 ms  roc1-ar5-xe-11-0-0-0.us.twtelecom.net [35.248.1.162]
 17   209 ms   208 ms   211 ms  66-195-65-170.static.ctl.one [66.195.65.170]
 18   209 ms   209 ms   209 ms  nat.hws.edu [64.89.144.100]
 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.
```

As per observations we can see if we ignore the request timed outs there is a difference on line 11.
We can see that the first row shows that the process of route tracing has started as the last column shows the Default Gateway of the user. The next few rows in both the cases are similar as the route is being traced starting from the ISP (Internet service provider) of the user. The next few rows, after which the tracing reaches the common IP address of 66.195.65.170 and then math.hws.edu [64.89.144.100], clearly show that the route is completely different after crossing the ISP for both the cases

**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.

```
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     1 ms  192.168.2.1
  2     8 ms     4 ms    91 ms  103.145.18.242
  3     8 ms     4 ms     4 ms  103.145.18.241
  4     *        *        *     Request timed out.
  5   132 ms    41 ms   155 ms  nsg-static-173.107.75.182-airtel.com [182.75.107.173]
  6   224 ms   205 ms   302 ms  116.119.52.163
  7   275 ms   303 ms   201 ms  core1.nyc4.he.net [198.32.118.57]
  8   256 ms   256 ms   255 ms  100ge8-1.core1.sjc2.he.net [184.105.81.218]
  9   255 ms   254 ms   255 ms  10ge4-5.core1.pao1.he.net [72.52.92.69]
 10   256 ms   255 ms   257 ms  184.105.177.238
 11   253 ms   368 ms   303 ms  csee-west-rtr-vl12.SUNet [171.66.0.238]
 12   253 ms   252 ms   253 ms  CS.stanford.edu [171.64.64.64]

Trace complete.
```

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

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

  1     3 ms     2 ms     1 ms  192.168.2.1
  2     8 ms     4 ms     4 ms  103.145.18.242
  3     8 ms     4 ms     3 ms  103.145.18.241
  4     8 ms     4 ms     *     103.6.184.9
  5    47 ms   143 ms    41 ms  nsg-static-173.107.75.182-airtel.com [182.75.107.173]
  6   209 ms   208 ms   287 ms  116.119.52.163
  7   256 ms   302 ms   303 ms  core1.nyc4.he.net [198.32.118.57]
  8   293 ms   256 ms   259 ms  100ge8-1.core1.sjc2.he.net [184.105.81.218]
  9   348 ms   303 ms   256 ms  10ge4-5.core1.pao1.he.net [72.52.92.69]
 10   256 ms   257 ms   267 ms  184.105.177.238
 11     *      293 ms   288 ms  csee-west-rtr-vl12.SUNet [171.66.0.238]
 12   269 ms   305 ms   302 ms  CS.stanford.edu [171.64.64.64]

Trace complete.
```

**QUESTIONS ABOUT PATHS**

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

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

➔ Yes some part of the path is common for all hosts that I tracerouted.

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?

➔ Yes, the number of nodes(number of hops subtract 1) is directly proportional to the distance between the source and destination.

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?

➔ 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.

➔ The *whois* command gives information about the domain name, the Registry Domain ID and some other details such as the details of the Registrar and the Registrant.

```
C:\Users\Lenovo\Downloads\WhoIs>whois -v google.com

Whois v1.21 - Domain information lookup
Copyright (C) 2005-2019 Mark Russinovich
Sysinternals - www.sysinternals.com

Connecting to COM.whois-servers.net...
Server COM.whois-servers.net returned the following for GOOGLE.COM

   Domain Name: GOOGLE.COM
   Registry Domain ID: 2138514_DOMAIN_COM-VRSN
   Registrar WHOIS Server: whois.markmonitor.com
   Registrar URL: http://www.markmonitor.com
   Updated Date: 2019-09-09T15:39:04Z
   Creation Date: 1997-09-15T04:00:00Z
   Registry Expiry Date: 2028-09-14T04:00:00Z
   Registrar: MarkMonitor Inc.
   Registrar IANA ID: 292
   Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
   Registrar Abuse Contact Phone: +1.2083895740
   Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
   Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
   Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
   Domain Status: serverDeleteProhibited https://icann.org/epp#serverDeleteProhibited
   Domain Status: serverTransferProhibited https://icann.org/epp#serverTransferProhibited
   Domain Status: serverUpdateProhibited https://icann.org/epp#serverUpdateProhibited
   Name Server: NS1.GOOGLE.COM
   Name Server: NS2.GOOGLE.COM
   Name Server: NS3.GOOGLE.COM
   Name Server: NS4.GOOGLE.COM
   DNSSEC: unsigned
   URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2020-10-09T07:37:37Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

NOTICE: The expiration date displayed in this record is the date the
registrar's sponsorship of the domain name registration in the registry is
currently set to expire. This date does not necessarily reflect the expiration
date of the domain name registrant's agreement with the sponsoring
registrar.  Users may consult the sponsoring registrar's Whois database to
view the registrar's reported date of expiration for this registration.


TERMS OF USE: You are not authorized to access or query our Whois
database through the use of electronic processes that are high-volume and
automated except as reasonably necessary to register domain names or
modify existing registrations; the Data in VeriSign Global Registry
Services' ("VeriSign") Whois database is provided by VeriSign for
information purposes only, and to assist persons in obtaining information
about or related to a domain name registration record. VeriSign does not
guarantee its accuracy. By submitting a Whois query, you agree to abide
by the following terms of use: You agree that you may use this Data only
for lawful purposes and that under no circumstances will you use this Data
to: (1) allow, enable, or otherwise support the transmission of mass
unsolicited, commercial advertising or solicitations via e-mail, telephone,
or facsimile; or (2) enable high volume, automated, electronic processes
that apply to VeriSign (or its computer systems). The compilation,
repackaging, dissemination or other use of this Data is expressly
prohibited without the prior written consent of VeriSign. You agree not to
use electronic processes that are automated and high-volume to access or
query the Whois database except as reasonably necessary to register
domain names or modify existing registrations. VeriSign reserves the right
to restrict your access to the Whois database in its sole discretion to ensure
operational stability.  VeriSign may restrict or terminate your access to the
Whois database for failure to abide by these terms of use. VeriSign
reserves the right to modify these terms at any time.

The Registry database contains ONLY .COM, .NET, .EDU domains and
Registrars.
```

```
The Registry database contains ONLY .COM, .NET, .EDU domains and
Registrars.


Connecting to whois.markmonitor.com...
Server whois.markmonitor.com returned the following for GOOGLE.COM

Domain Name: google.com
Registry Domain ID: 2138514_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.markmonitor.com
Registrar URL: http://www.markmonitor.com
Updated Date: 2019-09-09T08:39:04-0700
Creation Date: 1997-09-15T00:00:00-0700
Registrar Registration Expiration Date: 2028-09-13T00:00:00-0700
Registrar: MarkMonitor, Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
Registrar Abuse Contact Phone: +1.2083895770
Domain Status: clientUpdateProhibited (https://www.icann.org/epp#clientUpdateProhibited)
Domain Status: clientTransferProhibited (https://www.icann.org/epp#clientTransferProhibited)
Domain Status: clientDeleteProhibited (https://www.icann.org/epp#clientDeleteProhibited)
Domain Status: serverUpdateProhibited (https://www.icann.org/epp#serverUpdateProhibited)
Domain Status: serverTransferProhibited (https://www.icann.org/epp#serverTransferProhibited)
Domain Status: serverDeleteProhibited (https://www.icann.org/epp#serverDeleteProhibited)
Registrant Organization: Google LLC
Registrant State/Province: CA
Registrant Country: US
Registrant Email: Select Request Email Form at https://domains.markmonitor.com/whois/google.com
Admin Organization: Google LLC
Admin State/Province: CA
Admin Country: US
Admin Email: Select Request Email Form at https://domains.markmonitor.com/whois/google.com
Tech Organization: Google LLC
Tech State/Province: CA
Tech Country: US
Tech Email: Select Request Email Form at https://domains.markmonitor.com/whois/google.com
Name Server: ns1.google.com
Name Server: ns2.google.com
Name Server: ns4.google.com
Name Server: ns3.google.com
DNSSEC: unsigned
URL of the ICANN WHOIS Data Problem Reporting System: http://wdprs.internic.net/
>>> Last update of WHOIS database: 2020-10-09T00:28:15-0700 <<<

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

```
If you wish to contact this domainΓÇÖs Registrant, Administrative, or Technical
contact, and such email address is not visible above, you may do so via our web
form, pursuant to ICANNΓÇÖs Temporary Specification. To verify that you are not a
robot, please enter your email address to receive a link to a page that
facilitates email communication with the relevant contact(s).

Web-based WHOIS:
  https://domains.markmonitor.com/whois

If you have a legitimate interest in viewing the non-public WHOIS details, send
your request and the reasons for your request to whoisrequest@markmonitor.com
and specify the domain name in the subject line. We will review that request and
may ask for supporting documentation and explanation.

The data in MarkMonitorΓÇÖs WHOIS database is provided for information purposes,
and to assist persons in obtaining information about or related to a domain
nameΓÇÖs registration record. While MarkMonitor believes the data to be accurate,
the data is provided "as is" with no guarantee or warranties regarding its
accuracy.

By submitting a WHOIS query, you agree that you will use this data only for
lawful purposes and that, under no circumstances will you use this data to:
  (1) allow, enable, or otherwise support the transmission by email, telephone,
or facsimile of mass, unsolicited, commercial advertising, or spam; or
  (2) enable high volume, automated, or electronic processes that send queries,
data, or email to MarkMonitor (or its systems) or the domain name contacts (or
its systems).

MarkMonitor reserves the right to modify these terms at any time.

By submitting this query, you agree to abide by this policy.

MarkMonitor Domain Management(TM)
Protecting companies and consumers in a digital world.

Visit MarkMonitor at https://www.markmonitor.com
Contact us at +1.8007459229
In Europe, at +44.02032062220
--
```

**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.)

```
C:\Users\Lenovo\Downloads\WhoIs>curl  ipinfo.io/103.145.19.104
{
  "ip": "103.145.19.104",
  "city": "Mumbai",
  "region": "Maharashtra",
  "country": "IN",
  "loc": "19.0931,72.9049",
  "org": "AS139498 Speedostar Telco Private Limited",
  "postal": "400071",
  "timezone": "Asia/Kolkata",
  "readme": "https://ipinfo.io/missingauth"
}
```

## Conclusion:

1. Understood and implemented some basic command line network utilities.
2. Learnt about Network Latency, RTT and the factors impacting RTT.