Arka Haldi
Date: 17/08/2020
2018130014

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

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

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).

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

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

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.)

telnet — Telnet is an old program for remote login. It's not used so much for that any more, since it has no security features. But basically, all it does is open a connection to a server and allow server and client to send lines of plain text to each other. It can be used to check that it's possible to connect to a server and, if the server communicates in plain text, even to interact with the server by hand. Since the Web uses a plain text protocol, you can use telnet to connect to a web client and play the part of the web browser. I will suggest that you to do this with your own web server when you write it, but you might want to try it now. When you use telnet in this way, you need to specify both the host and the port number to which you want to connect: telent <host> <port>. For example, to connect to the web server on www.spit.ac.in: telnet spit.ac.in 80

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

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

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

```
traceroute <hostname>
```

The syntax in Windows is:

```
tracert <hostname>
```

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

1.2.1 Experiments with Traceroute

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

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

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

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.

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.

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?
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?
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?

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
```

(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.

Ex 1:

Washington University Webpage (www.uw.edu)
*Example PING output for Washington University

```
PING www.washington.edu (128.95.155.135) 64(92) bytes of data.
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=1 ttl=49 time=306 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=2 ttl=49 time=328 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=3 ttl=49 time=249 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=4 ttl=49 time=272 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=5 ttl=49 time=294 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=6 ttl=49 time=316 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=7 ttl=49 time=339 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=8 ttl=49 time=260 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=9 ttl=49 time=282 ms
72 bytes from www2.cac.washington.edu (128.95.155.135): icmp_seq=10 ttl=49 time=305 ms

--- www.washington.edu ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9012ms
rtt min/avg/max/mdev = 248.644/295.067/339.142/27.918 ms
```

```
PING www.washington.edu (128.95.155.134) 100(128) bytes of data.
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=1 ttl=49 time=306 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=2 ttl=49 time=331 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=3 ttl=49 time=355 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=4 ttl=49 time=276 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=5 ttl=49 time=301 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=6 ttl=49 time=325 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=7 ttl=49 time=348 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=8 ttl=49 time=269 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=9 ttl=49 time=293 ms
108 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=10 ttl=49 time=317
ms

--- www.washington.edu ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9001ms
rtt min/avg/max/mdev = 268.826/312.047/354.955/27.011 ms

PING www.washington.edu (128.95.155.134) 500(528) bytes of data.
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=1 ttl=49 time=320 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=2 ttl=49 time=342 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=3 ttl=49 time=263 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=4 ttl=49 time=285 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=5 ttl=49 time=308 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=6 ttl=49 time=331 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=7 ttl=49 time=242 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=8 ttl=49 time=274 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=9 ttl=49 time=240 ms
508 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=10 ttl=49 time=320
ms

--- www.washington.edu ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9011ms
rtt min/avg/max/mdev = 239.539/292.376/341.950/35.049 ms

PING www.washington.edu (128.95.155.134) 1000(1028) bytes of data.
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=1 ttl=49 time=241
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=2 ttl=49 time=240
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=3 ttl=49 time=308
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=4 ttl=49 time=240
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=5 ttl=49 time=241
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=6 ttl=49 time=263
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=7 ttl=49 time=297
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=8 ttl=49 time=313
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=9 ttl=49 time=452
ms
1008 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=10 ttl=49 time=267
ms

--- www.washington.edu ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9006ms
```

```
rtt min/avg/max/mdev = 239.932/286.231/451.854/61.718 ms

PING www.washington.edu (128.95.155.134) 1400(1428) bytes of data.
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=1 ttl=49 time=408
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=2 ttl=49 time=325
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=3 ttl=49 time=348
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=4 ttl=49 time=374
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=5 ttl=49 time=292
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=6 ttl=49 time=315
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=7 ttl=49 time=338
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=8 ttl=49 time=258
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=9 ttl=49 time=280
ms
1408 bytes from www1.cac.washington.edu (128.95.155.134): icmp_seq=10 ttl=49 time=303
ms

--- www.washington.edu ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9010ms
rtt min/avg/max/mdev = 257.516/324.086/408.491/42.766 ms
```

*Washington University physical distance : **18,666.6** km (roughly)

- Round trip time for **64** bytes
  min = 248.644 ms
  avg = **295.067** ms
  max = 339.142 ms
- Round trip time for **100** bytes
  min = 268.826 ms
  avg = **312.047** ms
  max = 354.955 ms
- Round trip time for **500** bytes
  min = 239.539 ms
  avg = **292.376** ms
  max = 341.950 ms
- Round trip time for **1000** bytes
  min = 239.932 ms
  avg = **286.231** ms
  max = 451.854 ms
- Round trip time for **4000** bytes
  min = 257.516 ms
  avg = **324.086** ms
  max = 408.491 ms

Similar results are seen when pinging other websites. They generally have the **same** average round trip time irrespective of **packet size**. There is **no clear trend.**

Berkeley University, California (berkeley.edu)

*Berkeley website uses AWS server in West US (www-production-1113102805.us-west-2.elb.amazonaws.com).
But it is still a similar distance away from India i.e **18,666.6**km(roughly)

- Average round trip time for 64 byte packet: **285.658** ms
- Average round trip time for 100 byte packet: **293.094** ms
- Average round trip time for 500 byte packet: **314.845** ms
- Average round trip time for 1000 byte packet: **309.332** ms
- Average round trip time for 4000 byte packet: **289.739** ms

Oxford University, Australia(www.ox.ac.uk)

*This webpage is hosted in Australia as opposed to USA. We observed similar Round trip time in both above cases, but in this case we see a drastic **reduction** in Round trip time as compared to the former websites.
Oxford University distance: **7000**km

- Average round trip time for 64 byte packet: **40.506** ms
- Average round trip time for 100 byte packet: **51.796** ms
- Average round trip time for 500 byte packet: **40.751** ms
- Average round trip time for 1000 byte packet: **49.616** ms
- Average round trip time for 4000 byte packet: **53.816** ms

Hence, we conclude that the Round Trip Time varies with the physical distance that it has to travel, and the size of packets sent doesn't influence the Round Trip Time. The other webpages were pinged but didn't give any response/were blocked via firewall.

Ex 2:

traceroute math.hws.edu

```
traceroute to math.hws.edu (64.89.144.237), 30 hops max, 60 byte packets
 1  _gateway (192.168.43.1)  8.073 ms  7.636 ms  9.969 ms
 2  192.168.0.1 (192.168.0.1)  13.749 ms  16.701 ms  19.020 ms
 3  nsg-static-122.167.76.182-airtel.com (182.76.167.122)  24.354 ms  41.534 ms
50.127 ms
 4  10.237.0.205 (10.237.0.205)  52.492 ms * *
 5  182.74.195.161 (182.74.195.161)  43.511 ms  45.478 ms  47.496 ms
 6  182.79.255.9 (182.79.255.9)  289.265 ms 182.79.239.78 (182.79.239.78)  236.587 ms
182.79.211.194 (182.79.211.194)  231.148 ms
 7  ae58.edge1.LosAngeles6.Level3.net (4.26.0.17)  227.885 ms
xe-9-1-0.edge1.LosAngeles6.Level3.net (4.26.0.61)  228.929 ms
ae58.edge1.LosAngeles6.Level3.net (4.26.0.17)  229.827 ms
 8  * * *
 9  * * *
```

```
10  roc1-ar5-xe-0-0-0-0.us.twtelecom.net (35.248.1.158)  288.757 ms  290.979 ms
360.215 ms
11  66-195-65-170.static.ctl.one (66.195.65.170)  360.140 ms  408.000 ms  407.923 ms
12  nat.hws.edu (64.89.144.100)  407.737 ms  407.771 ms *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
```

traceroute www.hws.edu

```
traceroute to www.hws.edu (64.89.145.159), 30 hops max, 60 byte packets
 1  _gateway (192.168.43.1)  3.805 ms  4.394 ms  4.360 ms
 2  192.168.0.1 (192.168.0.1)  6.808 ms  8.412 ms  9.375 ms
 3  nsg-static-122.167.76.182-airtel.com (182.76.167.122)  9.332 ms  9.284 ms  9.235
ms
 4  * * *
 5  182.74.195.161 (182.74.195.161)  19.480 ms  19.466 ms  19.753 ms
 6  182.79.239.78 (182.79.239.78)  238.537 ms 182.79.211.194 (182.79.211.194)  226.767
ms 182.79.201.106 (182.79.201.106)  229.856 ms
 7  ae58.edge1.LosAngeles6.Level3.net (4.26.0.17)  229.735 ms
xe-9-1-0.edge1.LosAngeles6.Level3.net (4.26.0.61)  229.763 ms
ae58.edge1.LosAngeles6.Level3.net (4.26.0.17)  235.125 ms
 8  ae-2-52.ear3.LosAngeles1.Level3.net (4.69.207.49)  222.579 ms  225.831 ms  224.634
ms
 9  * * *
10  roc1-ar5-xe-0-0-0-0.us.twtelecom.net (35.248.1.158)  294.884 ms * *
11  * 66-195-65-170.static.ctl.one (66.195.65.170)  409.801 ms  409.664 ms
12  nat.hws.edu (64.89.144.100)  409.463 ms  409.477 ms  409.433 ms
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
```

- Both pages are different, but fetched from the same server, as indicated by the same IP address.

- The first website route goes to a bogon server, whereas second site route doesn’t

```
{
  "ip": "10.237.0.205",
  "bogon": true
}
```

- The IP address of one server in the middle changes, but the location and organisation is the same for both “BHARTI Airtel Ltd.”. So, it’s probably routed to different IP’s in the same Network Service Provider’s Server.

```
math.hws.edu
{
  "ip": "182.79.255.9",
  "city": "Gurgaon",
  "region": "Haryana",
  "country": "IN",
  "loc": "28.4601,77.0263",
  "org": "AS9498 BHARTI Airtel Ltd.",
  "postal": "122004",
  "timezone": "Asia/Kolkata",
  "readme": "https://ipinfo.io/missingauth"
}

www.hws.edu
{
  "ip": "182.79.239.78",
  "city": "Gurgaon",
  "region": "Haryana",
  "country": "IN",
  "loc": "28.4601,77.0263",
  "org": "AS9498 BHARTI Airtel Ltd.",
  "postal": "122004",
  "timezone": "Asia/Kolkata",
  "readme": "https://ipinfo.io/missingauth"
}
```

- The IP address of one server in the middle belonging to L.A. changes, but the location and organisation is the same “LosAngeles1.Level3.net” . So, it’s probably routed to one more IP’s in the same Network Service Provider’s Server.

Ex 3:

- As part of this experiment, we tracerouted google multiple times, **afternoon** and **evening**, and after a gap of **4 days**, starting Tuesday.

- We observed that the path taken is slightly different, but the number of nodes is the same i.e 8. And the path taken is not dependent on time of the day.
- We also observe that it connects to Google LLC Bombay, so we tried with the institutes pinged above to get some traceroute results from IP's far away.

*Tuesday 18/08/20 Afternoon

```
 traceroute to www.google.com (216.58.199.132), 30 hops max, 60 byte packets
 1  _gateway (192.168.0.1)  5.518 ms  5.455 ms  5.406 ms
 2  nsg-static-122.167.76.182-airtel.com (182.76.167.122)  5.358 ms  5.306 ms  5.256
ms
 3  10.237.0.205 (10.237.0.205)  13.415 ms * *
 4  124.40.245.250 (124.40.245.250)  15.177 ms  15.325 ms  15.280 ms
 5  209.85.149.210 (209.85.149.210)  32.838 ms  32.860 ms  32.814 ms
 6  108.170.248.193 (108.170.248.193)  33.062 ms 108.170.248.209 (108.170.248.209)
32.373 ms  33.030 ms
 7  72.14.239.235 (72.14.239.235)  30.523 ms  30.862 ms  30.773 ms
 8  bom07s01-in-f4.1e100.net (216.58.199.132)  31.243 ms  31.137 ms  31.072 ms
```

*Tuesday 18/08/20 Evening

```
traceroute to www.google.com (172.217.174.228), 30 hops max, 60 byte packets
 1  _gateway (192.168.43.1)  2.678 ms  7.264 ms  7.554 ms
 2  192.168.0.1 (192.168.0.1)  9.967 ms  11.747 ms  12.299 ms
 3  nsg-static-122.167.76.182-airtel.com (182.76.167.122)  13.211 ms  13.578 ms
13.974 ms
 4  * * *
 5  124.40.245.250 (124.40.245.250)  22.923 ms  23.276 ms  23.676 ms
 6  209.85.149.210 (209.85.149.210)  44.491 ms  34.201 ms  33.858 ms
 7  108.170.248.161 (108.170.248.161)  32.676 ms 108.170.248.177 (108.170.248.177)
34.814 ms 108.170.248.161 (108.170.248.161)  33.171 ms
 8  142.250.60.135 (142.250.60.135)  36.740 ms  37.483 ms  38.417 ms
 9  bom12s03-in-f4.1e100.net (172.217.174.228)  33.705 ms  36.148 ms  38.870 ms
```

*Saturday 22/08/20 Afternoon

```
Tracing route to www.google.com [142.250.67.164]
over a maximum of 30 hops:

  1     1 ms     1 ms     2 ms  192.168.0.1
  2     5 ms     8 ms     5 ms  nsg-static-122.167.76.182-airtel.com [182.76.167.122]
  3    17 ms    15 ms    13 ms  10.237.0.205
  4    12 ms    12 ms    12 ms  124.40.245.250
  5    30 ms    29 ms    29 ms  209.85.149.210
  6    29 ms    29 ms    30 ms  108.170.248.177
  7    29 ms    29 ms    29 ms  142.250.227.75
  8    29 ms    33 ms    31 ms  bom12s07-in-f4.1e100.net [142.250.67.164]
```

*Saturday 22/08/20 Evening

```
Tracing route to www.google.com [142.250.67.228]
over a maximum of 30 hops:

  1     2 ms     1 ms     2 ms  192.168.0.1
  2     2 ms     2 ms     2 ms  nsg-static-122.167.76.182-airtel.com [182.76.167.122]
  3    13 ms    12 ms    12 ms  10.237.0.205
  4    13 ms    13 ms    14 ms  124.40.245.250
  5    32 ms    30 ms    29 ms  209.85.149.210
  6    30 ms    30 ms    32 ms  108.170.248.209
  7    31 ms    29 ms    29 ms  216.239.58.19
  8    32 ms    29 ms    29 ms  bom07s24-in-f4.1e100.net [142.250.67.228]
```

- **Washington University** (`www.uw.edu`) and **Oxford University** (www.ox.ac.uk) were also tested on Saturday(22/08/2020) and the next Monday(24/08/2020), since they are both farther from India.

Saturday 22/08/2020

```
(Washington University)
traceroute to www.uw.edu (128.95.155.135), 30 hops max, 60 byte packets
 1  _gateway (192.168.0.1)  2.029 ms  2.658 ms  2.628 ms
 2  nsg-static-122.167.76.182-airtel.com (182.76.167.122)  2.899 ms  2.859 ms  2.825
ms
 3  * * *
 4  182.74.195.161 (182.74.195.161)  14.713 ms  15.859 ms  15.740 ms
 5  182.79.224.145 (182.79.224.145)  55.716 ms 116.119.49.240 (116.119.49.240)  55.069
ms 182.79.146.196 (182.79.146.196)  56.498 ms
 6  63-218-107-193.static.pccwglobal.net (63.218.107.193)  51.900 ms * *
 7  TenGE0-3-0-1.br03.sea01.pccwbtn.net (63.223.47.49)  343.882 ms
TenGE0-3-0-5.br03.sea01.pccwbtn.net (63.223.47.65)  343.854 ms
TenGE0-3-0-1.br03.sea01.pccwbtn.net (63.223.47.49)  343.809 ms
 8  TenGE0-3-0-5.br03.sea01.pccwbtn.net (63.223.47.65)  343.730 ms
TenGE0-3-0-1.br03.sea01.pccwbtn.net (63.223.47.49)  343.723 ms
TenGE0-3-0-5.br03.sea01.pccwbtn.net (63.223.47.65)  343.660 ms
 9  pnwgp-six1.pnw-gigapop.net (206.81.80.84)  344.062 ms  343.195 ms  343.971 ms
10  ae20--4010.icar-sttl1-3.infra.pnw-gigapop.net (209.124.188.134)  343.929 ms
343.852 ms  230.462 ms
11  et-7-0-0--4010.uwcr-atg-1.infra.washington.edu (209.124.188.135)  305.938 ms
305.878 ms  358.652 ms

(Oxford University)
Tracing route to www.washington.edu [128.95.155.198]
over a maximum of 30 hops:

  1     2 ms     1 ms     1 ms  192.168.0.1
  2     2 ms     2 ms     3 ms  nsg-static-122.167.76.182-airtel.com [182.76.167.122]
  3    13 ms    13 ms    13 ms  10.237.0.205
  4    13 ms    13 ms    14 ms  182.74.195.161
  5    53 ms    54 ms    53 ms  182.79.146.194
  6     *       52 ms    53 ms  63-218-107-193.static.pccwglobal.net [63.218.107.193]
  7   250 ms   262 ms   249 ms  TenGE0-3-0-5.br03.sea01.pccwbtn.net [63.223.47.65]
  8   248 ms   248 ms   248 ms  TenGE0-3-0-5.br03.sea01.pccwbtn.net [63.223.47.65]
  9   258 ms   256 ms   256 ms  pnwgp-six1.pnw-gigapop.net [206.81.80.84]
 10   256 ms   256 ms   257 ms  et-7-0-0--4000.uwcr-ads-1.infra.washington.edu
[209.124.188.133]
 11     *        *        *     Request timed out.
 12   252 ms   256 ms   252 ms  ae3--836.uwar-uwtc-1.infra.washington.edu
[128.95.155.195]
 13   251 ms   250 ms   250 ms  www4.cac.washington.edu [128.95.155.198]
```

Monday 22/08/2020

```
(Washington University)
traceroute to www.uw.edu (128.95.155.197), 30 hops max, 60 byte packets

 1  _gateway (192.168.43.1)  3.160 ms  4.096 ms  4.463 ms
 2  192.168.0.1 (192.168.0.1)  9.087 ms  19.424 ms  20.376 ms
 3  nsg-static-122.167.76.182-airtel.com (182.76.167.122)  29.895 ms * *
 4  * * *
 5  182.74.195.161 (182.74.195.161)  42.495 ms * *
 6  182.79.224.145 (182.79.224.145)  74.737 ms 182.79.134.149 (182.79.134.149)  66.956
ms 182.79.237.18 (182.79.237.18)  77.549 ms
 7  63-218-107-193.static.pccwglobal.net (63.218.107.193)  74.903 ms *  74.883 ms
 8  TenGE0-3-0-1.br03.sea01.pccwbtn.net (63.223.47.49)  245.481 ms  245.468 ms
TenGE0-3-0-5.br03.sea01.pccwbtn.net (63.223.47.65)  247.237 ms
 9  TenGE0-3-0-5.br03.sea01.pccwbtn.net (63.223.47.65)  245.413 ms  245.389 ms
247.165 ms
10  pnwgp-six1.pnw-gigapop.net (206.81.80.84)  257.620 ms  257.633 ms  304.345 ms
11  ae20--4010.icar-sttl1-3.infra.pnw-gigapop.net (209.124.188.134)  240.915 ms
246.252 ms  252.564 ms
12  et-7-0-0--4010.uwcr-atg-1.infra.washington.edu (209.124.188.135)  253.704 ms
257.081 ms  250.064 ms

(Oxford University)
Tracing route to www.washington.edu [128.95.155.135]
over a maximum of 30 hops:

  1     2 ms     1 ms     4 ms  192.168.0.1
  2     6 ms     2 ms     2 ms  nsg-static-122.167.76.182-airtel.com [182.76.167.122]
  3    14 ms    12 ms    13 ms  10.237.0.205
  4    14 ms    19 ms    23 ms  182.74.195.161
  5    51 ms    53 ms    51 ms  182.79.146.194
  6    50 ms    49 ms     *     63-218-107-193.static.pccwglobal.net [63.218.107.193]
  7   242 ms   240 ms   240 ms  TenGE0-3-0-5.br03.sea01.pccwbtn.net [63.223.47.65]
  8   246 ms   242 ms   240 ms  TenGE0-3-0-5.br03.sea01.pccwbtn.net [63.223.47.65]
  9   247 ms   247 ms   251 ms  pnwgp-six1.pnw-gigapop.net [206.81.80.84]
 10   249 ms   247 ms   246 ms  et-7-0-0--4000.uwcr-ads-1.infra.washington.edu
[209.124.188.133]
 11     *        *        *     Request timed out.
 12   250 ms   246 ms   247 ms  ae4--583.uwar-ads-1.infra.washington.edu
[128.95.155.131]
 13   248 ms   249 ms   247 ms  www2.cac.washington.edu [128.95.155.135]
```

- We observe here that even if the servers are remote, the traceroute shows that it follows the similar path, even if it's taken on different days of the week and encounters same/similar nodes.
- The ping time/latency shows no clear pattern with the number of nodes visited.

Ex 4:

- **Whois**

- We get information about the registrar WHOIS server (whois.markmonitor.com), it's complaint cell details
- We also get Domain Name Registry expiry date(2028-09-14T04:00:00Z),list of domain status and domain names, and so on.

```
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: 2020-08-17T17:09:11Z <<<

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.
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: ns3.google.com
Name Server: ns4.google.com
DNSSEC: unsigned
URL of the ICANN WHOIS Data Problem Reporting System: http://wdprs.internic.net/
>>> Last update of WHOIS database: 2020-08-17T10:03:20-0700 <<<

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

If you wish to contact this domain’s Registrant, Administrative, or Technical
contact, and such email address is not visible above, you may do so via our web
form, pursuant to ICANN’s 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 MarkMonitor’s WHOIS database is provided for information purposes,
and to assist persons in obtaining information about or related to a domain
name’s 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
```

Ex 5:

- traceroute spit.ac.in

```
nslookup spit.ac.in
Server:         127.0.0.53
Address:  127.0.0.53#53

Non-authoritative answer:
Name:     spit.ac.in
Address: 43.252.193.19
```

- curl ipinfo.io/43.252.193.19

```
{
  "ip": "43.252.193.19",
  "city": "Mumbai",
  "region": "Maharashtra",
  "country": "IN",
  "loc": "19.0728,72.8826",
  "org": "AS17625 BlazeNet's Network",
  "postal": "400070",
  "timezone": "Asia/Kolkata",
  "readme": "https://ipinfo.io/missingauth"
}
```

Ex 6:

```
netstat -t -n
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 192.168.43.218:55782    69.173.159.48:443       ESTABLISHED
tcp        0      0 192.168.43.218:51466    216.58.199.163:443      ESTABLISHED
tcp        0      0 192.168.43.218:41468    74.125.200.125:443      ESTABLISHED
tcp        0      1 192.168.43.218:43030    103.231.98.193:443      FIN_WAIT1
tcp        0      0 192.168.43.218:36780    216.58.199.174:443      ESTABLISHED
tcp        0      0 192.168.43.218:60854    13.107.6.171:443        ESTABLISHED
tcp        0      0 192.168.43.218:38406    52.11.231.199:443       ESTABLISHED
tcp        0      0 192.168.43.218:57350    216.58.203.35:443       ESTABLISHED
tcp        0      0 192.168.43.218:39502    74.125.24.188:5228      ESTABLISHED
tcp        0      0 192.168.43.218:39722    34.213.232.243:443      ESTABLISHED
tcp        0      0 192.168.43.218:44642    142.250.67.228:443      ESTABLISHED
tcp        0      0 192.168.43.218:44616    209.58.162.201:443      ESTABLISHED
tcp        0      0 192.168.43.218:43122    172.217.166.35:443      ESTABLISHED
tcp        0      0 192.168.43.218:53900    74.125.200.189:443      ESTABLISHED
tcp        0      0 192.168.43.218:35090    35.190.63.234:443       ESTABLISHED
tcp        0      0 192.168.43.218:34036    52.32.142.97:443        ESTABLISHED
tcp        1      1 192.168.43.218:33732    35.213.34.3:443         LAST_ACK
tcp        0      0 192.168.43.218:56616    184.25.161.247:443      ESTABLISHED
tcp        0      0 192.168.43.218:43234    107.178.247.57:443      ESTABLISHED
tcp        0      0 192.168.43.218:44962    54.244.7.118:443        ESTABLISHED
▶ curl ipinfo.io/69.173.159.48
{
  "ip": "69.173.159.48",
  "city": "Central",
  "region": "Central and Western",
  "country": "HK",
  "loc": "22.2830,114.1585",
  "org": "AS26667 The Rubicon Project, Inc.",
  "timezone": "Asia/Hong_Kong",
  "readme": "https://ipinfo.io/missingauth"
}
▶ curl ipinfo.io/216.58.199.163
{
  "ip": "216.58.199.163",
  "hostname": "bom05s08-in-f163.1e100.net",
  "city": "Mumbai",
  "region": "Maharashtra",
  "country": "IN",
  "loc": "19.0728,72.8826",
  "org": "AS15169 Google LLC",
  "postal": "400070",
  "timezone": "Asia/Kolkata",
  "readme": "https://ipinfo.io/missingauth"
}
▶ curl ipinfo.io/103.231.98.193
{
  "ip": "103.231.98.193",
  "city": "Singapore",
  "region": "Singapore",
  "country": "SG",
  "loc": "1.2897,103.8501",
  "org": "AS62713 PubMatic, Inc.",
  "postal": "048508",
  "timezone": "Asia/Singapore",
  "readme": "https://ipinfo.io/missingauth"
}
▶ curl ipinfo.io/52.32.142.97
{
  "ip": "52.32.142.97",
  "hostname": "ec2-52-32-142-97.us-west-2.compute.amazonaws.com",
  "city": "Portland",
  "region": "Oregon",
  "country": "US",
  "loc": "45.5235,-122.6762",
  "org": "AS16509 Amazon.com, Inc.",
  "postal": "97258",
  "timezone": "America/Los_Angeles",
  "readme": "https://ipinfo.io/missingauth"
}
```

Conclusion:

- We used the different network utilities to perform the experiment.
- We used ping and found that there is no relation of packet size with latency, and that the latency varies with the physical location of the IP address.
- We experimented with traceroute to find that the latency doesn't correlate with the number of nodes visited.
- We also used whois and ipinfo.io to find more information about DNS/geolocation respectively.