Name: Rudresh Veerkhare
UID: 2018130061
Batch: D

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?

Answers :

1. Yes. Average RTT varies between different hosts. Actual round trip time 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.
   Propagation Delay - time it takes for router to process the packet header, depends on the processing speed of the switch
   Queuing delay – time the packet spends in routing queues depends on the number of packets, size of the packet and bandwidth..
   Transmission delay – time it takes to push the packet's bits onto the link depends on size of the packet and the bandwidth of the network.
   Propagation delay – time for a signal to reach its destination depends on distance and propagation speed.

2. Yes. Average RTT varies with different packet sizes as **Transmission delay** and **Propagation delay** depends upon the size of the packet.

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).

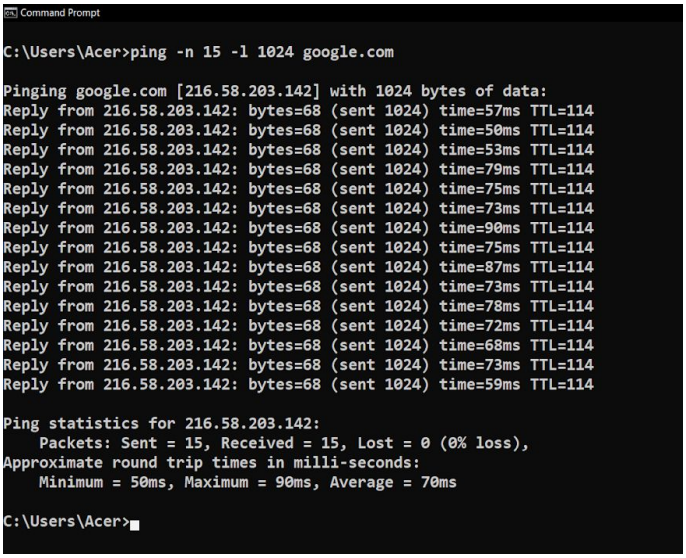
Host : google.com
packet size: 32 bytes
count: 15 packets

```
C:\Users\Acer>ping -n 15 google.com

Pinging google.com [172.217.174.78] with 32 bytes of data:
Reply from 172.217.174.78: bytes=32 time=49ms TTL=114
Reply from 172.217.174.78: bytes=32 time=51ms TTL=114
Reply from 172.217.174.78: bytes=32 time=47ms TTL=114
Reply from 172.217.174.78: bytes=32 time=58ms TTL=114
Reply from 172.217.174.78: bytes=32 time=65ms TTL=114
Reply from 172.217.174.78: bytes=32 time=49ms TTL=114
Reply from 172.217.174.78: bytes=32 time=62ms TTL=114
Reply from 172.217.174.78: bytes=32 time=64ms TTL=114
Reply from 172.217.174.78: bytes=32 time=62ms TTL=114
Reply from 172.217.174.78: bytes=32 time=53ms TTL=114
Reply from 172.217.174.78: bytes=32 time=140ms TTL=114
Reply from 172.217.174.78: bytes=32 time=63ms TTL=114
Reply from 172.217.174.78: bytes=32 time=69ms TTL=114
Reply from 172.217.174.78: bytes=32 time=60ms TTL=114
Reply from 172.217.174.78: bytes=32 time=52ms TTL=114

Ping statistics for 172.217.174.78:
    Packets: Sent = 15, Received = 15, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 47ms, Maximum = 140ms, Average = 62ms
```

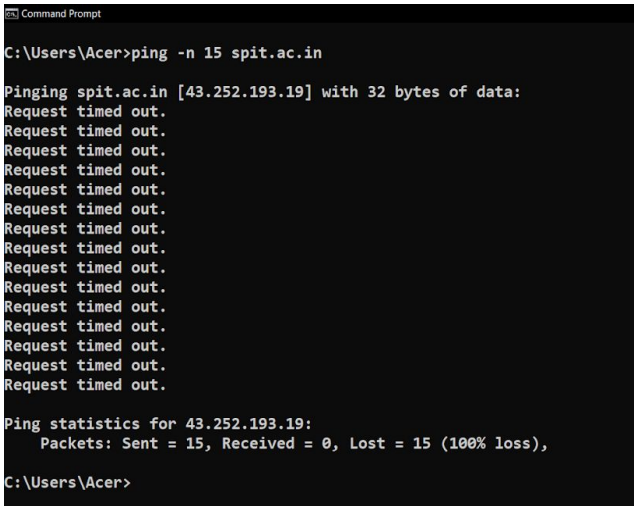
Host : www.uw.edu
packet size: 32 bytes
count: 15 packets

```
C:\Users\Acer>ping -n 15 www.uw.edu

Pinging www.washington.edu [128.95.155.134] with 32 bytes of data:
Reply from 128.95.155.134: bytes=32 time=334ms TTL=44
Reply from 128.95.155.134: bytes=32 time=321ms TTL=44
Reply from 128.95.155.134: bytes=32 time=416ms TTL=44
Reply from 128.95.155.134: bytes=32 time=321ms TTL=44
Reply from 128.95.155.134: bytes=32 time=317ms TTL=44
Reply from 128.95.155.134: bytes=32 time=342ms TTL=44
Reply from 128.95.155.134: bytes=32 time=346ms TTL=44
Reply from 128.95.155.134: bytes=32 time=312ms TTL=44
Reply from 128.95.155.134: bytes=32 time=320ms TTL=44
Reply from 128.95.155.134: bytes=32 time=322ms TTL=44
Reply from 128.95.155.134: bytes=32 time=316ms TTL=44
Reply from 128.95.155.134: bytes=32 time=310ms TTL=44
Reply from 128.95.155.134: bytes=32 time=325ms TTL=44
Reply from 128.95.155.134: bytes=32 time=323ms TTL=44
Reply from 128.95.155.134: bytes=32 time=335ms TTL=44

Ping statistics for 128.95.155.134:
    Packets: Sent = 15, Received = 15, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 310ms, Maximum = 416ms, Average = 330ms

C:\Users\Acer>
```

Host : google.com
packet size: 1024 bytes
count: 15 packets



Host : spit.ac.in
packet size: 32 bytes
count: 15 packets



Observation:

Average RTT depends upon various factors like Distance, Transmission media, Number of network hops, Traffic levels and Server Response Time.Average RTT increases as the distance and the paket size increases. And in the case of spit.ac.in we didn't receive a response

but the website was running, the reason for that is that website admin has disabled the ping command support of the server.

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.

1. math.hws.edu

```
Command Prompt

C:\Users\Acer>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     2 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3    38 ms    47 ms    57 ms  10.71.249.2
  4    40 ms    47 ms    38 ms  172.25.19.133
  5    37 ms    38 ms    49 ms  172.26.46.188
  6    45 ms    49 ms    42 ms  172.17.120.6
  7    42 ms    38 ms    38 ms  172.17.120.65
  8    51 ms    67 ms    88 ms  172.16.92.147
  9   114 ms   110 ms    72 ms  172.16.24.30
 10    47 ms    49 ms    52 ms  172.16.2.48
 11   185 ms   196 ms   177 ms  103.198.140.45
 12   173 ms   187 ms   186 ms  103.198.140.54
 13   181 ms   181 ms   183 ms  103.198.140.45
 14   168 ms   167 ms   168 ms  hu0-4-0-1.agr21.lhr01.atlas.cogentco.com [149.14.196.81]
 15   173 ms   177 ms   166 ms  be3671.ccr51.lhr01.atlas.cogentco.com [130.117.48.137]
 16   161 ms   168 ms   169 ms  be3487.ccr41.lon13.atlas.cogentco.com [154.54.60.5]
 17   164 ms   180 ms   176 ms  be2868.ccr21.lon01.atlas.cogentco.com [154.54.57.154]
 18     *        *        *     Request timed out.
 19   265 ms   179 ms   171 ms  ae-115-3501.edge3.London15.Level3.net [4.69.167.74]
 20   171 ms   168 ms   168 ms  ae-115-3501.edge3.London15.Level3.net [4.69.167.74]
 21   205 ms   175 ms   177 ms  ae4.ar8.lon15.Level3.net [4.68.111.254]
 22   322 ms   296 ms   308 ms  roc1-ar5-xe-11-0-0-0.us.twtelecom.net [35.248.1.162]
 23   297 ms   308 ms   309 ms  66-195-65-170.static.ctl.one [66.195.65.170]
 24   307 ms   310 ms   394 ms  nat.hws.edu [64.89.144.100]
 25     *        *        *     Request timed out.
 26     *        *        *     Request timed out.
 27     *        *        *     Request timed out.
 28     *        *        *     Request timed out.
 29     *        *        *     Request timed out.
```

2. www.hws.edu

```
Command Prompt

C:\Users\Acer>tracert www.hws.edu

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

  1     2 ms     1 ms     2 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3    59 ms    42 ms    45 ms  10.71.249.2
  4    53 ms    36 ms    77 ms  172.25.19.133
  5    52 ms    39 ms    49 ms  172.26.46.184
  6    35 ms    58 ms    67 ms  172.17.120.6
  7    48 ms    59 ms    60 ms  172.17.120.69
  8    49 ms    69 ms    47 ms  172.26.40.5
  9    60 ms    56 ms    58 ms  172.16.24.10
 10    75 ms    81 ms    74 ms  172.16.2.46
 11   226 ms   187 ms   211 ms  103.198.140.45
 12   192 ms   183 ms   199 ms  103.198.140.54
 13   212 ms   196 ms   217 ms  103.198.140.45
 14   223 ms   205 ms   206 ms  hu0-4-0-1.agr21.lhr01.atlas.cogentco.com [149.14.196.81]
 15   223 ms   204 ms   205 ms  be3672.ccr52.lhr01.atlas.cogentco.com [130.117.48.145]
 16   193 ms   202 ms   183 ms  be3488.ccr42.lon13.atlas.cogentco.com [154.54.60.13]
 17   258 ms   203 ms   195 ms  be2871.ccr21.lon01.atlas.cogentco.com [154.54.58.186]
 18   183 ms   191 ms   205 ms  ae-6.edge7.London1.Level3.net [4.68.62.5]
 19   181 ms   188 ms   206 ms  ae-225-3601.edge3.London15.Level3.net [4.69.167.90]
 20   187 ms   191 ms   199 ms  ae-225-3601.edge3.London15.Level3.net [4.69.167.90]
 21   201 ms   205 ms   198 ms  ae4.ar8.lon15.Level3.net [4.68.111.254]
 22   313 ms   323 ms   325 ms  roc1-ar5-xe-11-0-0-0.us.twtelecom.net [35.248.1.162]
 23   326 ms   318 ms   327 ms  66-195-65-170.static.ctl.one [66.195.65.170]
 24   335 ms   316 ms   382 ms  nat.hws.edu [64.89.144.100]
 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.
```

Observation: Both of the traceroute results showed that the final server is the same which is **64.89.144.100**

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.

Host: google.com

1 week earlier:

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

Tracing route to google.com [216.58.203.142]
over a maximum of 30 hops:

  1     2 ms     4 ms     2 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3    47 ms    52 ms    44 ms  10.71.249.10
  4    55 ms    47 ms    38 ms  172.25.19.133
  5    57 ms    49 ms    48 ms  172.26.46.184
  6    52 ms    38 ms    48 ms  172.17.120.6
  7    64 ms    50 ms    45 ms  172.17.120.65
  8    79 ms    57 ms    58 ms  172.26.40.5
  9    65 ms    59 ms    56 ms  172.16.24.8
 10    75 ms    57 ms    64 ms  172.16.2.46
 11    63 ms    61 ms    69 ms  172.16.92.146
 12    70 ms    58 ms    68 ms  49.44.18.38
 13    63 ms    71 ms    59 ms  10.70.14.97
 14    78 ms    73 ms    55 ms  74.125.32.32
 15    63 ms    78 ms    67 ms  209.85.248.57
 16    62 ms    57 ms    59 ms  209.85.251.29
 17    69 ms    89 ms    58 ms  bom05s10-in-f142.1e100.net [216.58.203.142]

Trace complete.
```

now :

```
Command Prompt
C:\Users\Acer>tracert google.com

Tracing route to google.com [216.58.203.14]
over a maximum of 30 hops:

  1     2 ms     5 ms     2 ms  192.168.43.1
  2     *        *        *     Request timed out.
  3    53 ms    38 ms    57 ms  10.71.249.2
  4    46 ms    54 ms    63 ms  172.25.19.133
  5   111 ms    51 ms    61 ms  172.26.46.188
  6    49 ms    40 ms    50 ms  172.17.120.6
  7    63 ms    37 ms    58 ms  172.17.120.65
  8    87 ms    62 ms    48 ms  172.16.92.145
  9    77 ms    58 ms    66 ms  172.16.24.8
 10    64 ms    58 ms    67 ms  172.16.2.46
 11    78 ms    58 ms    68 ms  172.26.40.64
 12    87 ms    50 ms    68 ms  49.44.18.38
 13   757 ms   276 ms   167 ms  10.70.14.97
 14    88 ms    63 ms    64 ms  74.125.32.32
 15   107 ms    75 ms    79 ms  209.85.248.57
 16    81 ms    79 ms    77 ms  172.253.77.23
 17    64 ms    76 ms    79 ms  hkg12s09-in-f14.1e100.net [216.58.203.14]

Trace complete.
```

Observation:

Even if the host is same and destination is also same but route changes after some amount of time.

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. First path is the same.

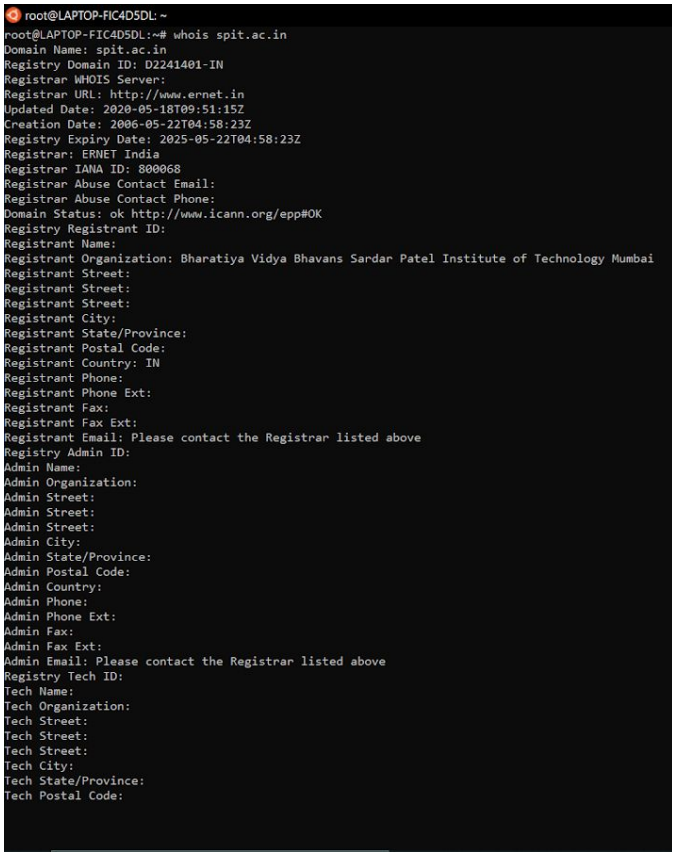
2. Is there a relationship between the number of nodes that show up in the traceroute and the location of the host? If so, what is this relationship?
   No. it's dependent on the physical interface used.

3. Is there a relationship between the number of nodes that show up in the traceroute and latency of the host (from your ping results above)? Does the same relationship hold for all hosts?
   Yes. As the number of nodes increases, delay also increases.

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.

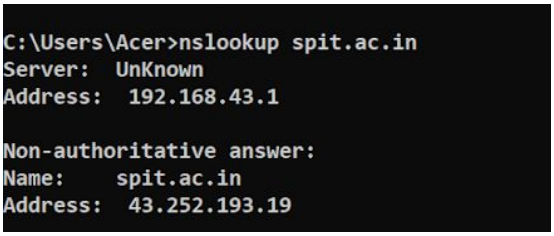
Geolocation — A geolocation service tries to tell, approximately, where a given IP address is located physically. They can't be completely accurate—but they probably get at least the country right most of the time.

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:

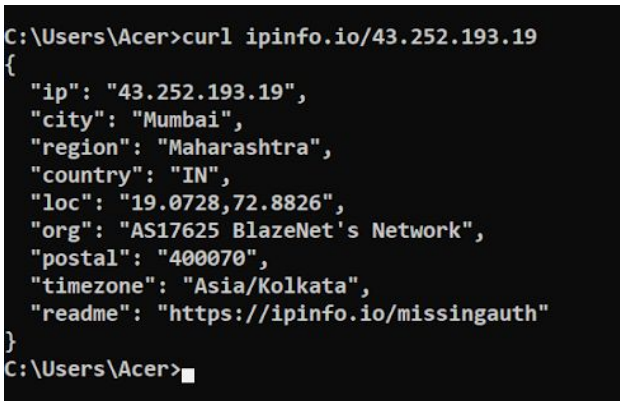
curl ipinfo.io/129.64.99.200

Answer:

1. First we can get an ip address using nslookup command.



2. Then using curl ipinfo.io/<ip-address>



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

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

Conclusion:

Learned methods to analyze the network connections and find insights about host and the intermediate nodes.