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

Network latency, specifically round trip time (RTT), can be measured using ping, which sends ICMP (Internet Control Message Protocol) 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:

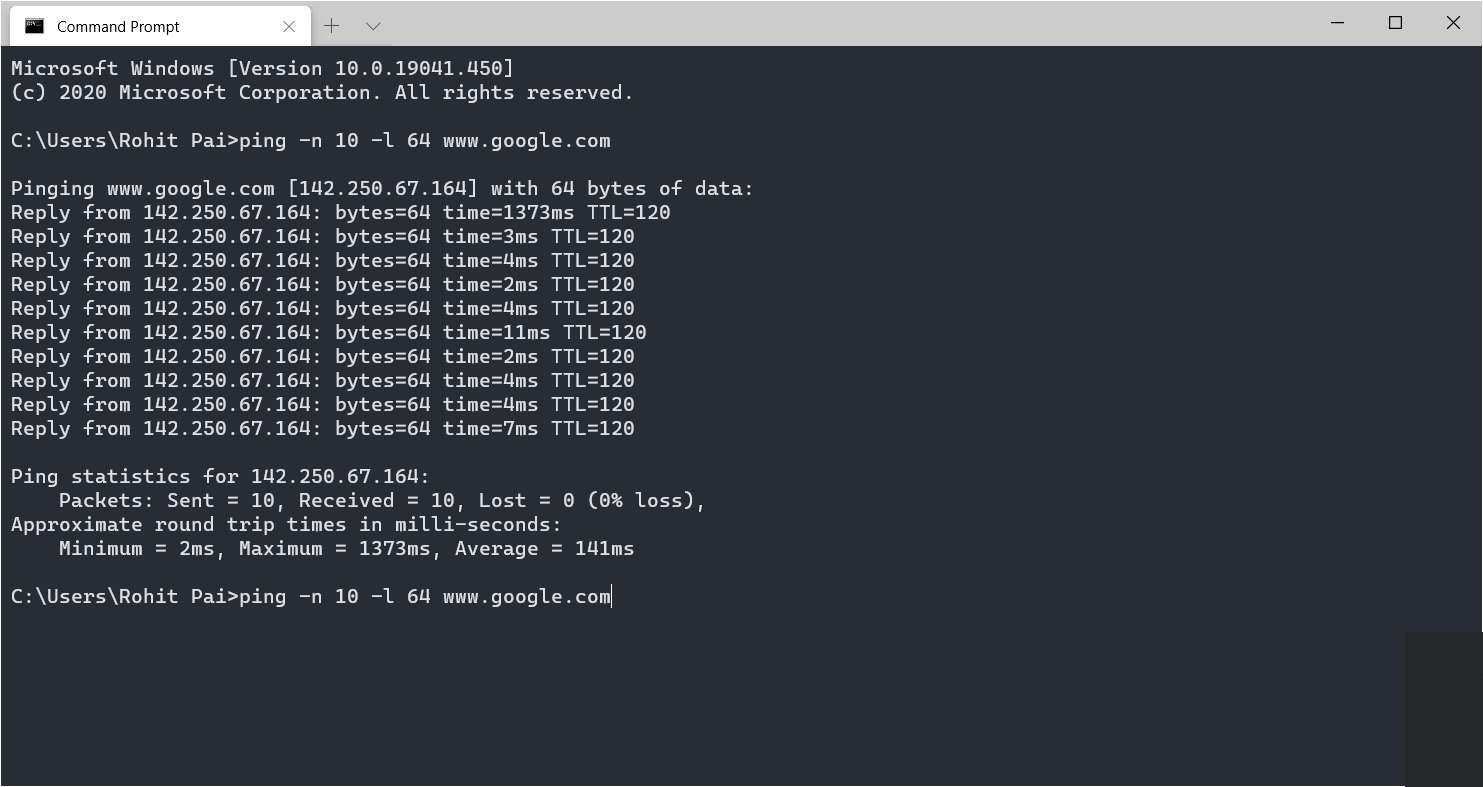
ping -c 10 google.com > ping\_c10\_s64\_google.log

**Experiments with Ping**

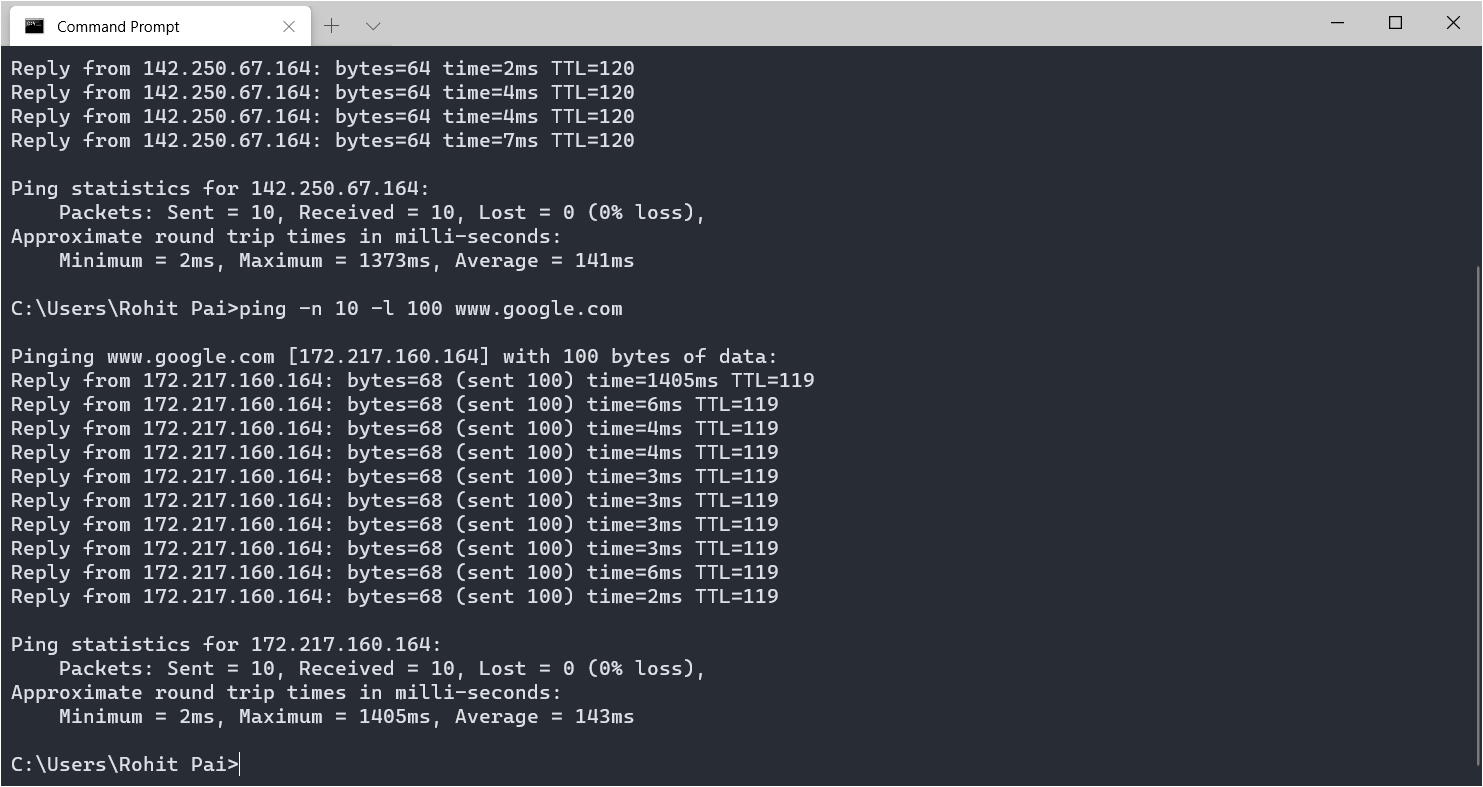
1. Ping the any hosts 10 times (i.e., packet count is 10) with a packet size of 64 bytes, 100 bytes, 500 bytes, 1000 bytes, 1400 bytes

Results –

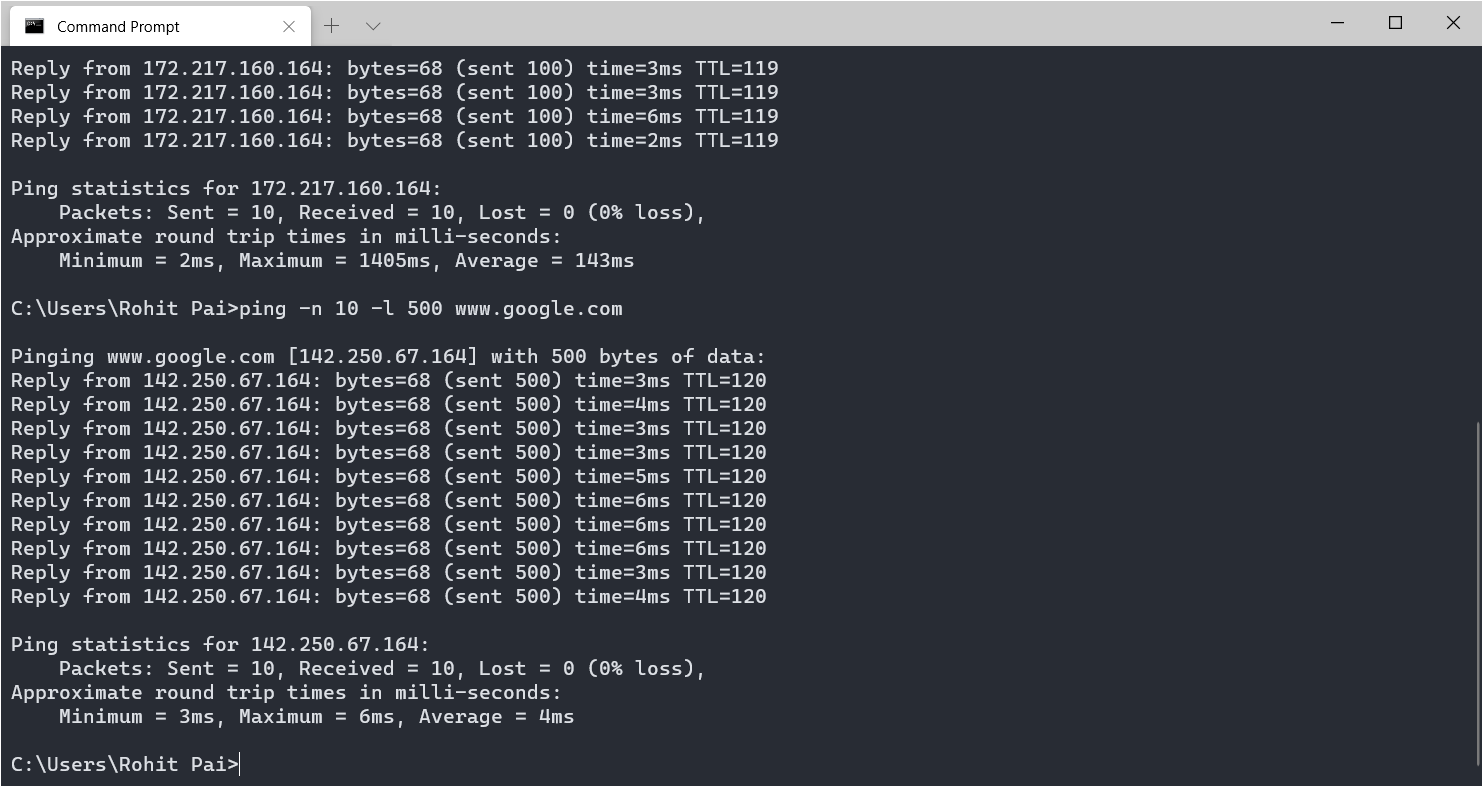
Pinging www.google.com 10 times with a packet size of 64 bytes



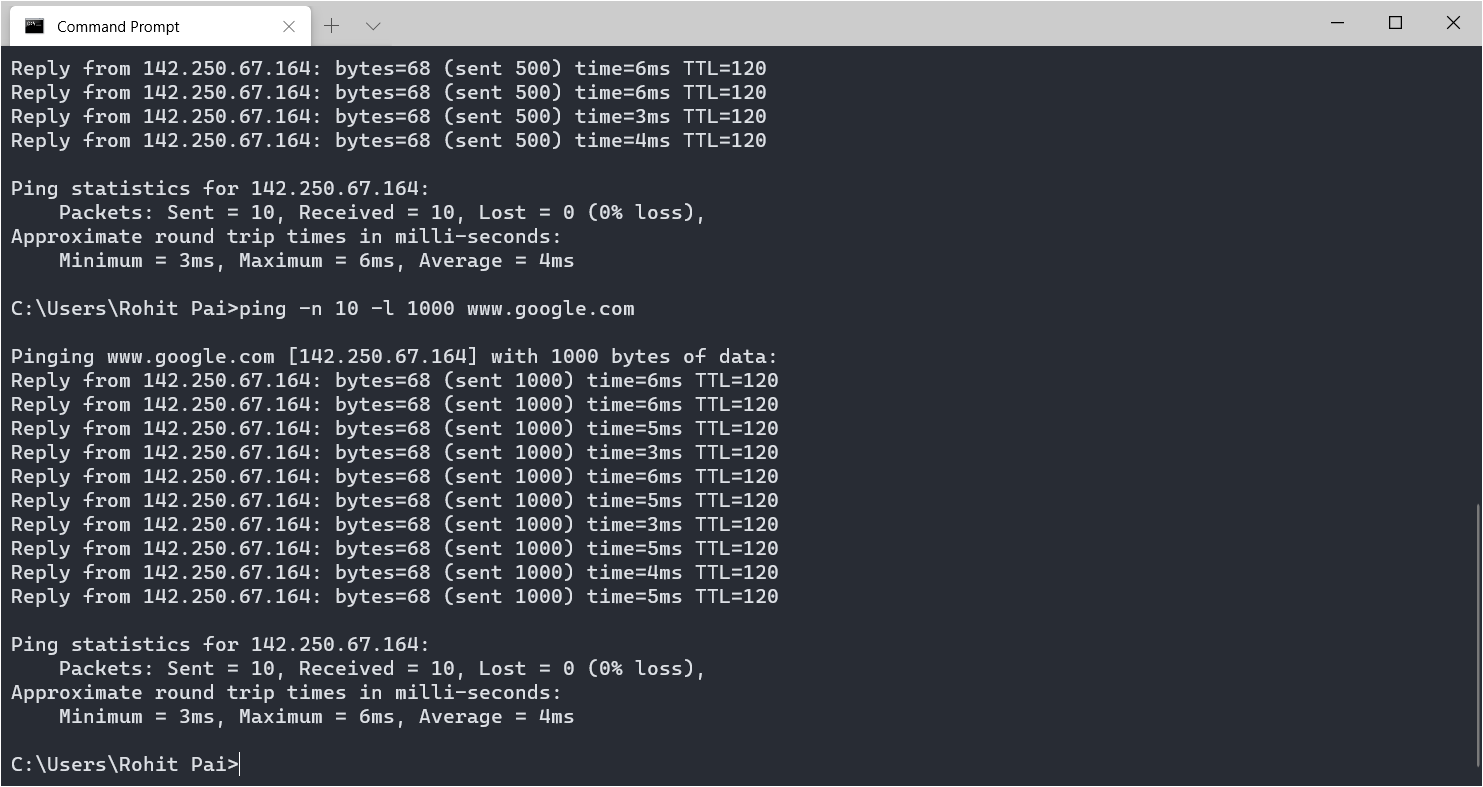
Pinging www.google.com 10 times with a packet size of 100 bytes



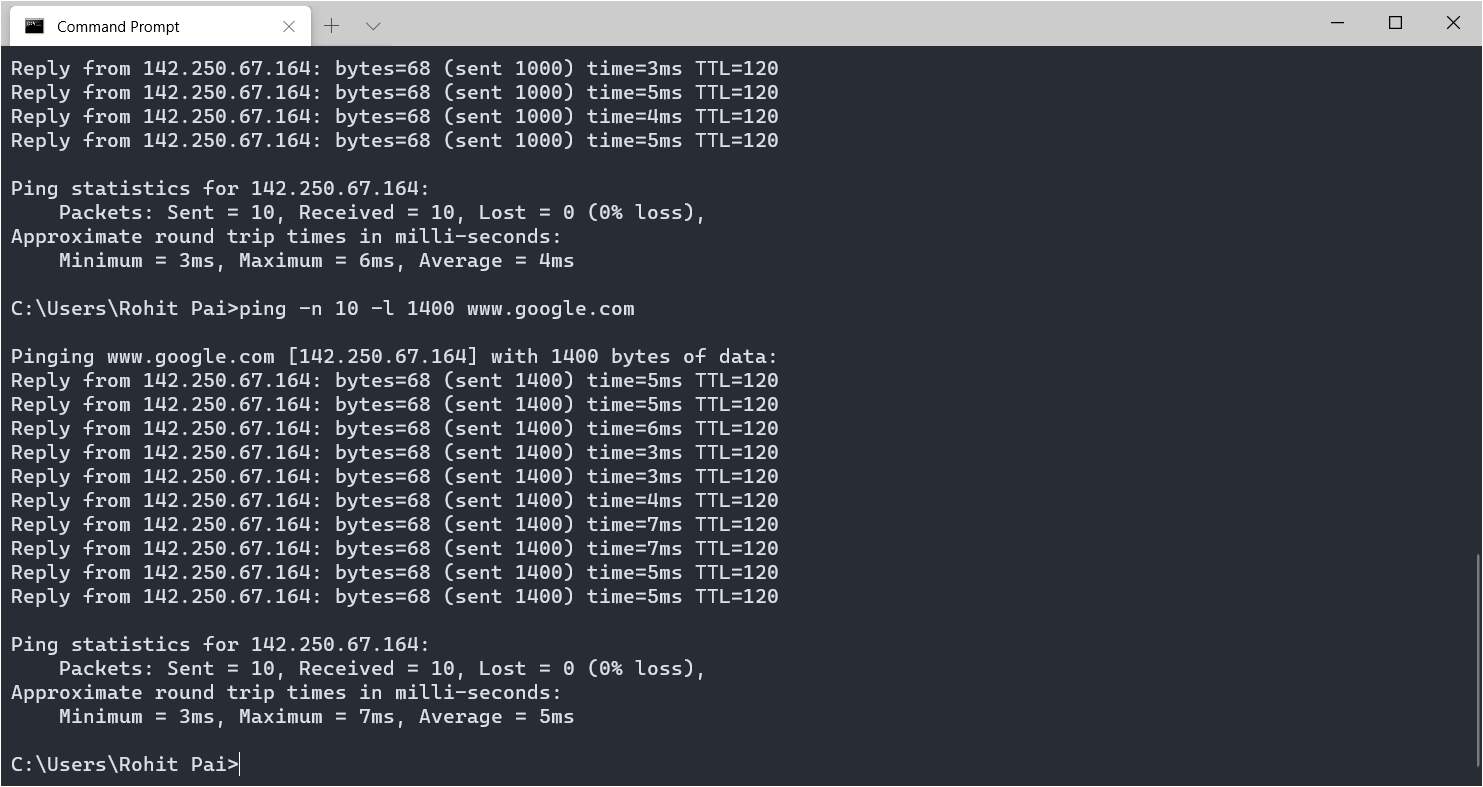
Ping www.google.com 10 times with a packet size of 500 bytes



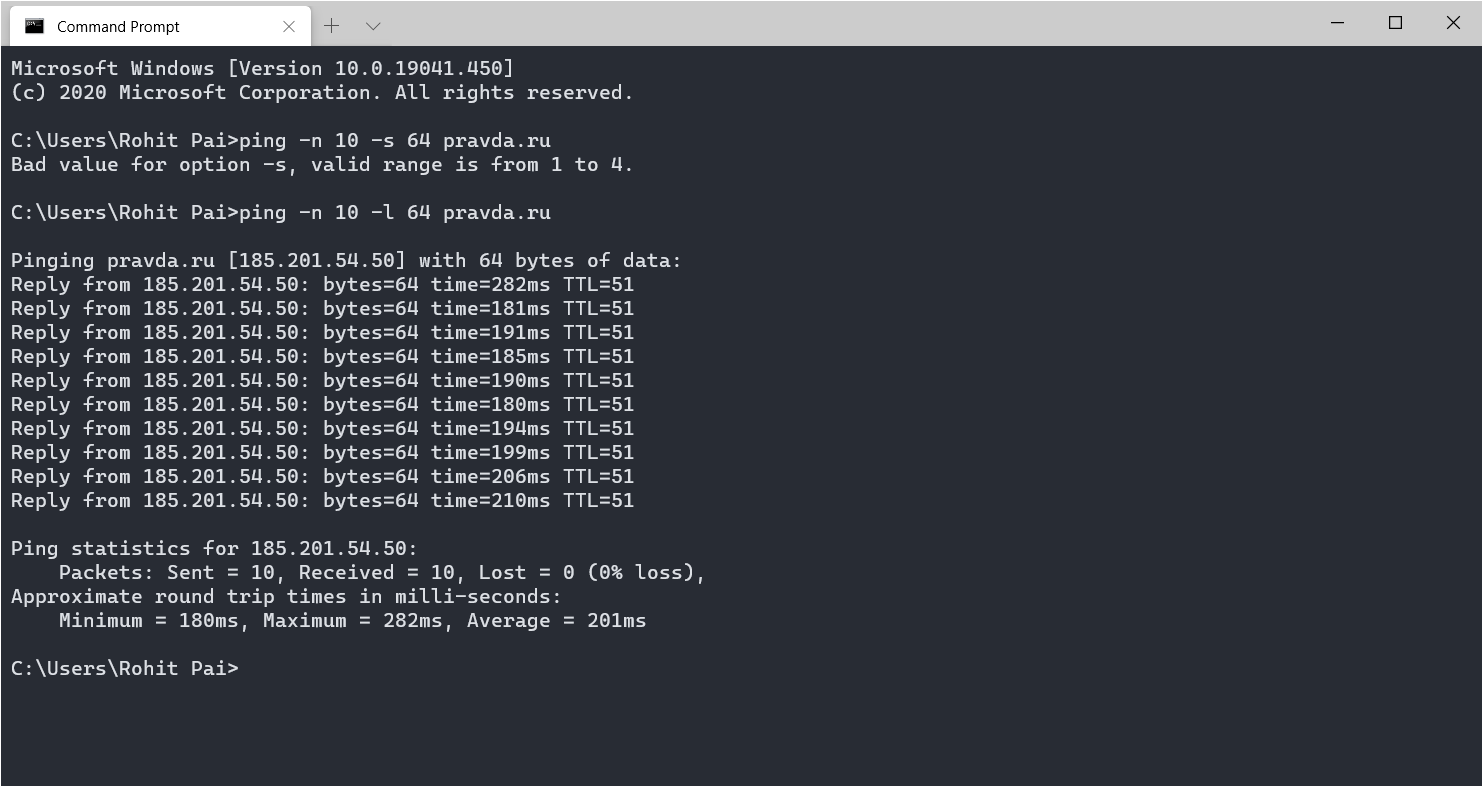
Pinging www,google.com 10 times with a packet size of 1000 bytes



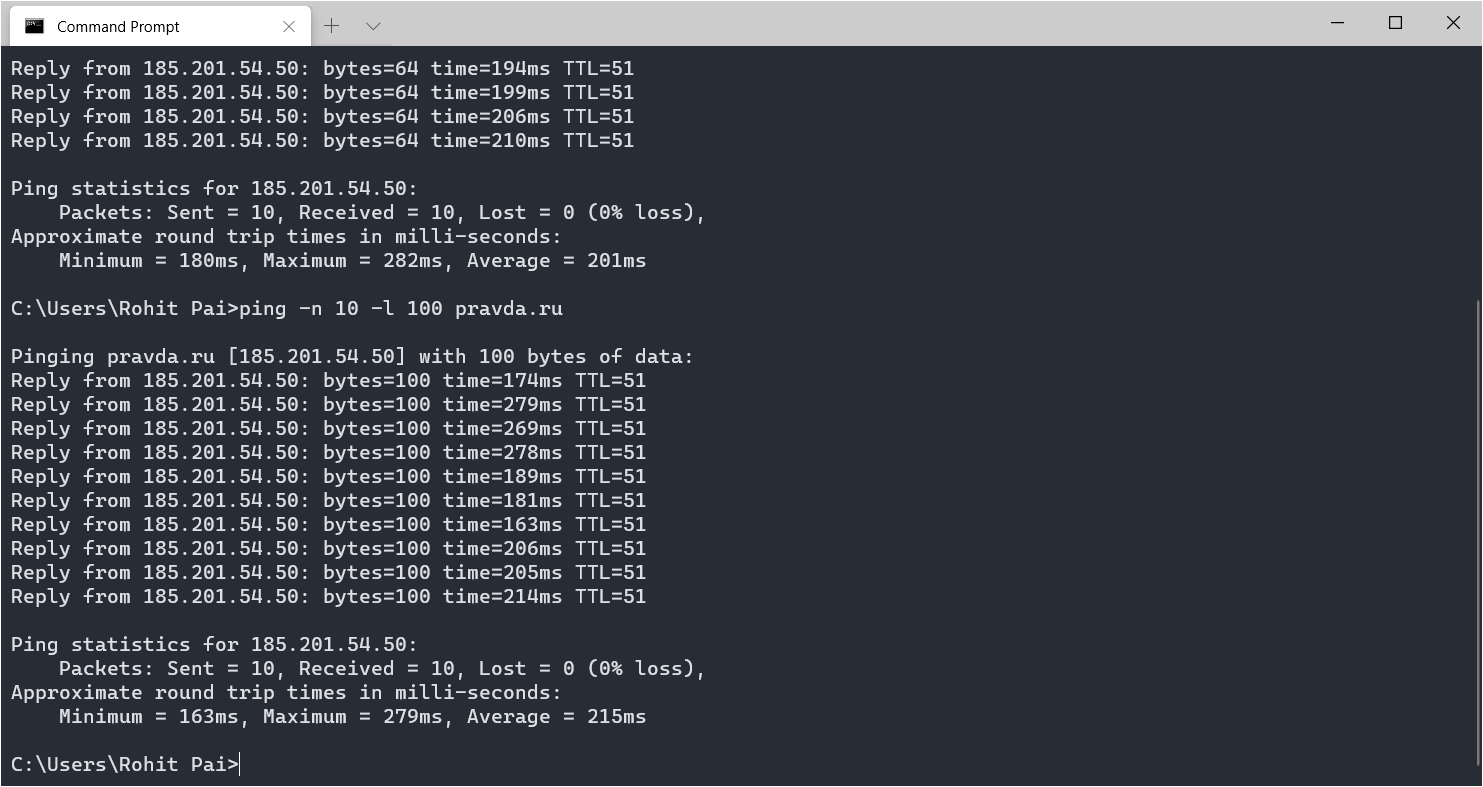
Pinging www.google.com 10 times with a packet size of 1400 bytes



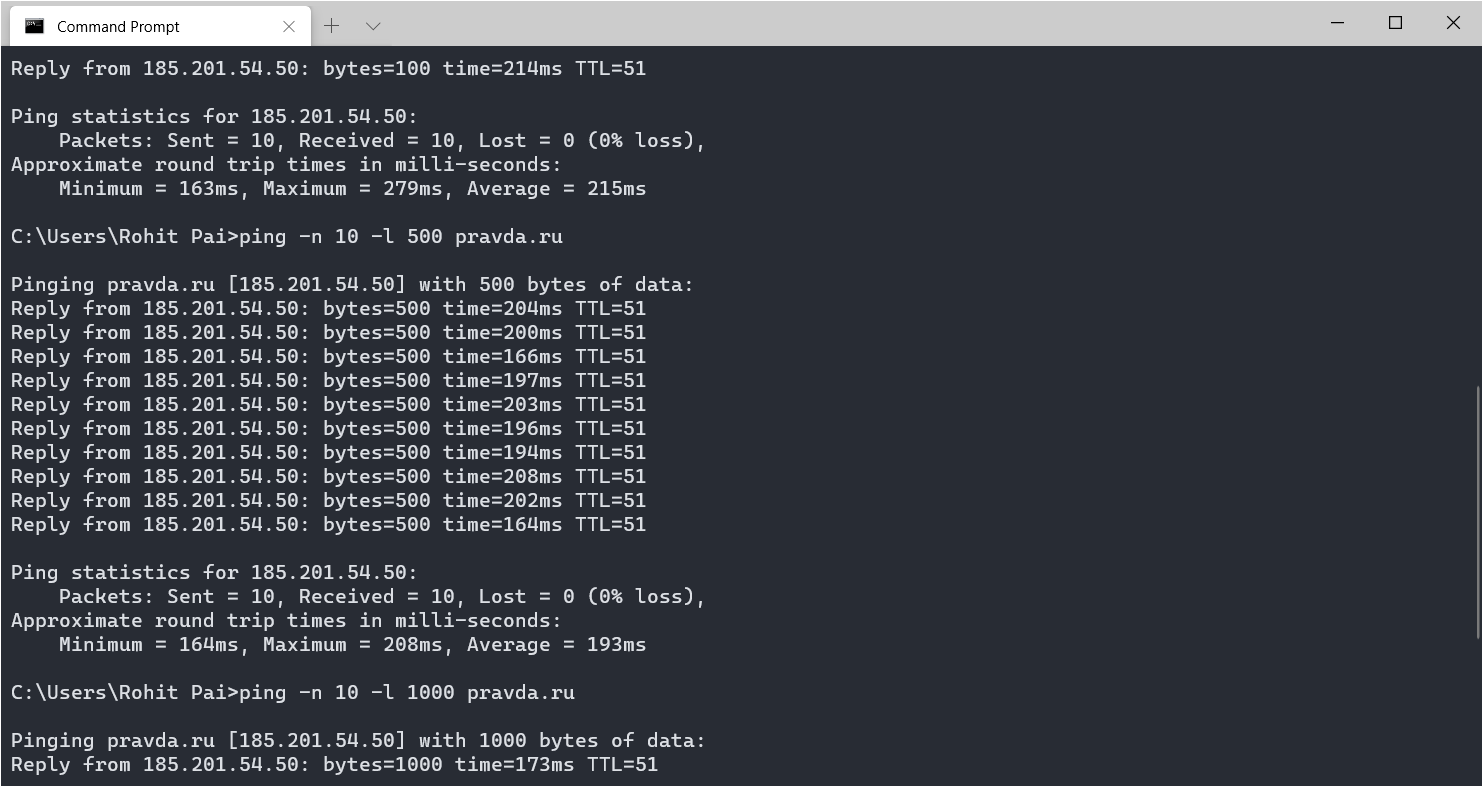
Pinging pravda.ru 10 times with a packet size of 64 bytes



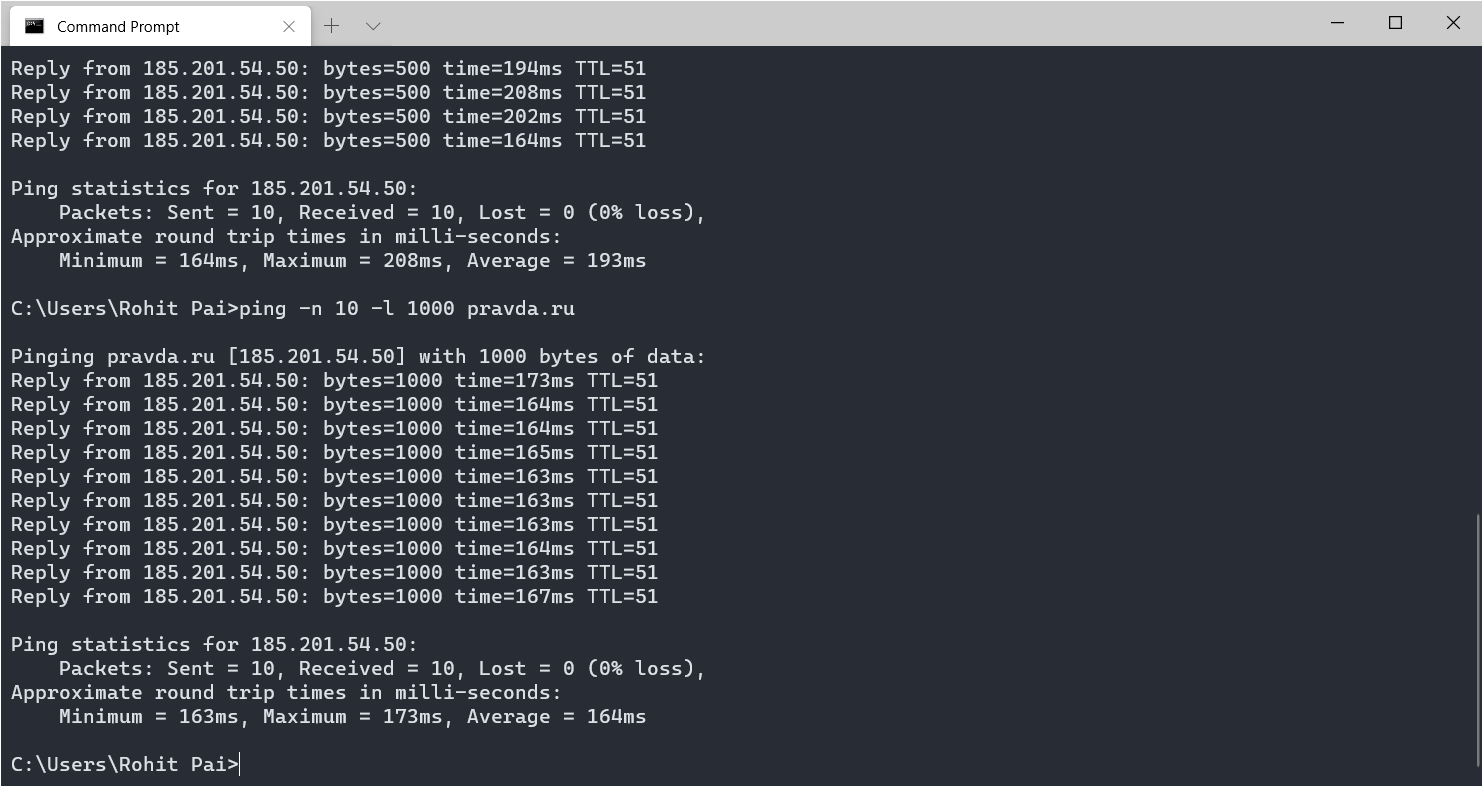
Pinging pravda.ru 10 times with a packet size of 100 bytes



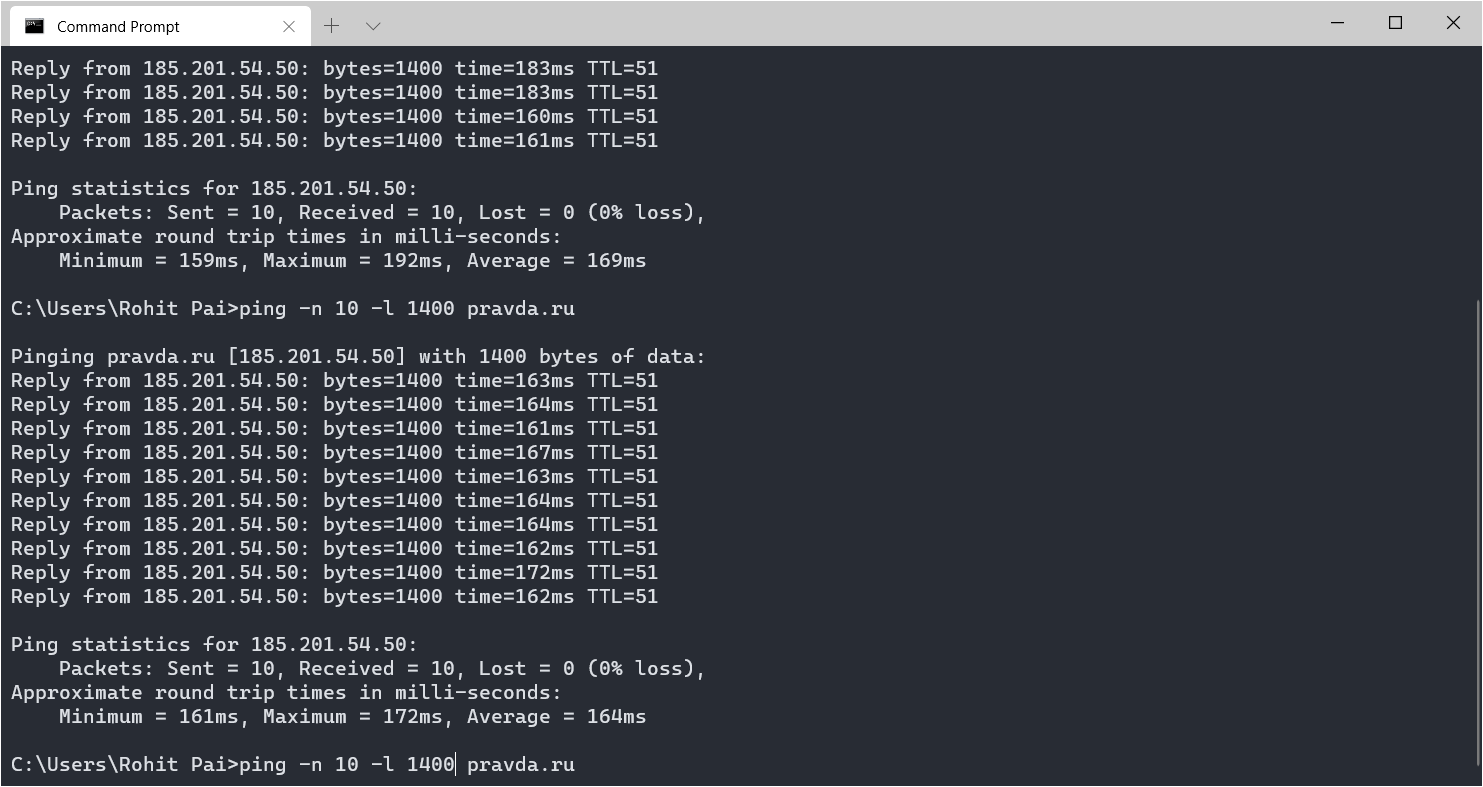
Pinging pravda.ru 10 times with a packet size of 500 bytes



Pinging pravda.ru 10 times with a packet size of 1000 bytes



Pinging pravda.ru 10 times with a packet size of 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?

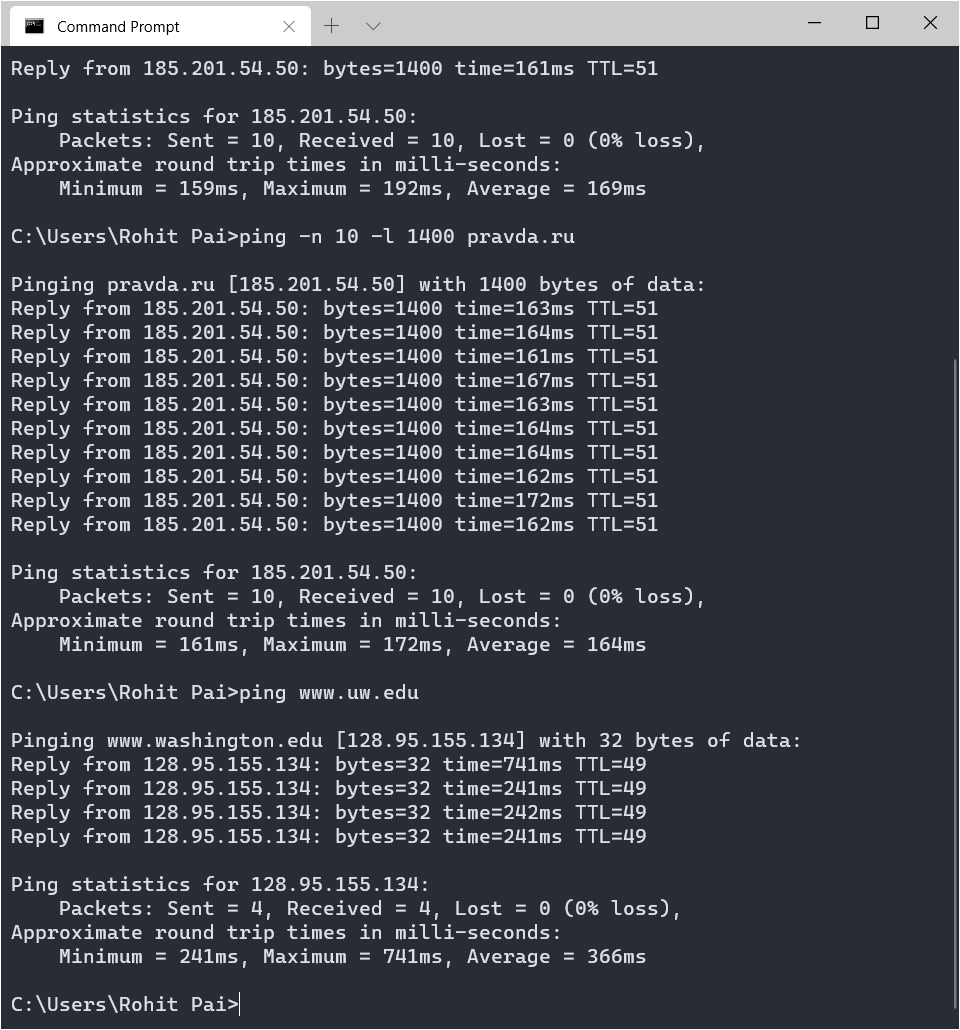
Ans. Yes, the average RTT varies between different hosts as there are many factors such as physical distance between the nodes, various delays such as processing delay, queuing delay that affect the transmission time. If the physical distance is more, the packet will have to go through more nodes and the time taken (RTT) will increase accordingly. Queuing delay occurs when a gateway receives multiple packets from different sources heading towards the same destination. Since typically only one packet can be transmitted at a time, some of the packets must queue for transmission, incurring additional delay. A processing delay is incurred while a gateway determines what to do with a newly received packet.

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

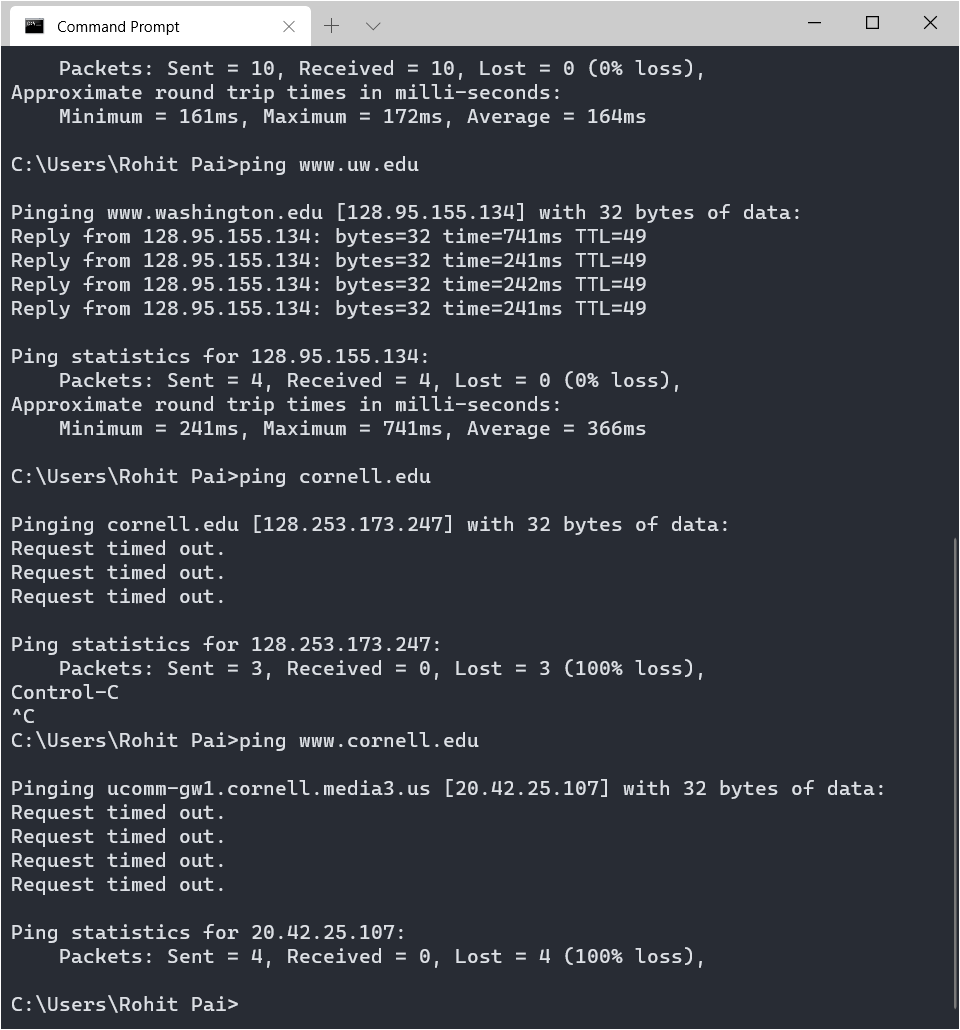
Ans. Yes, the average RTT usually varies with different packet sizes. If a packet size is bigger, it would hog the bandwidth of the channel during transmission and there are chances that some nodes may drop this packet. If the packet size is less, this problem would not occur during the ping command. But in my case, I assume that the bandwidth of my network is sufficient, hence the average RTT remains similar for all packet sizes.

**Exercise 1**: Experiment with ping to find the round trip times to a variety of destinations. Write up any interesting observations, including in particular how the round trip time compares to the physical distance. Here are few places from who to get replies: www.uw.edu, www.cornell.edu, berkeley.edu, www.uchicago.edu, www.ox.ac.uk (England), www.u-tokyo.ac.jp (Japan).

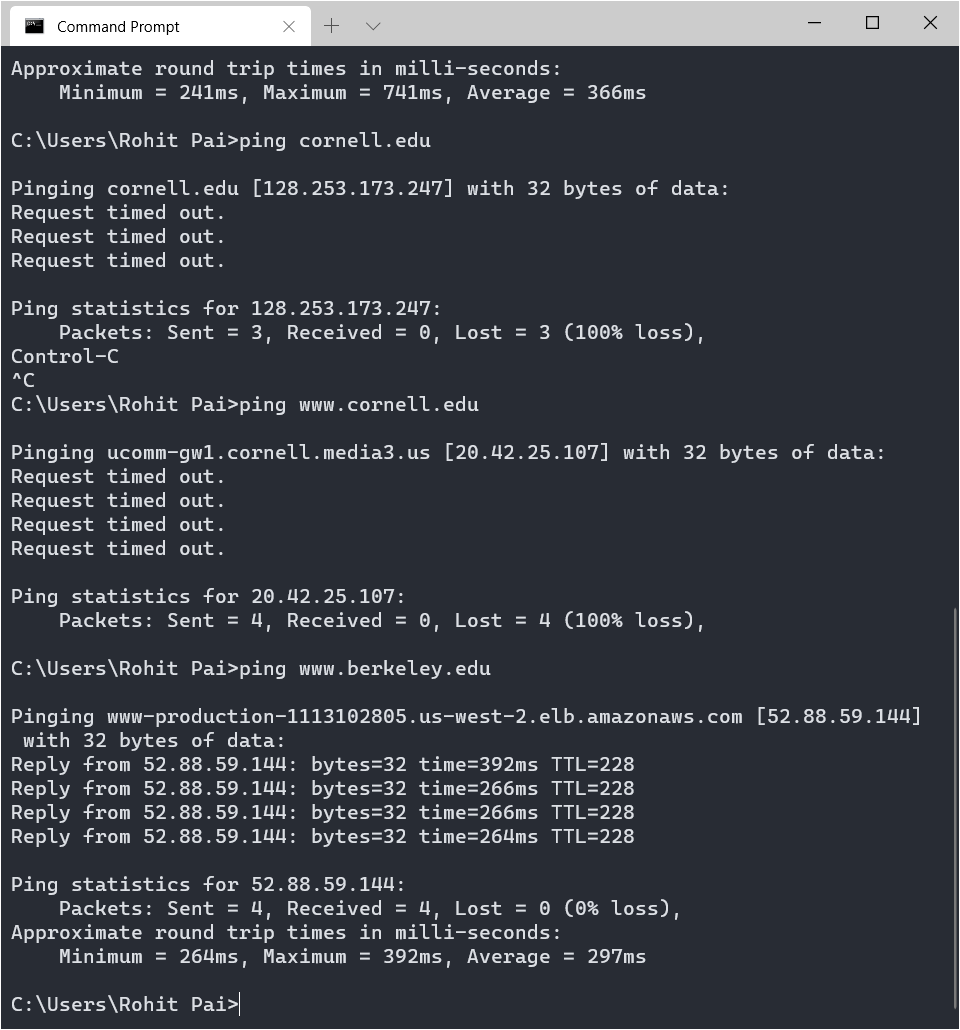
Pinging [www.uw.edu](http://www.uw.edu) (Washingtion, USA)



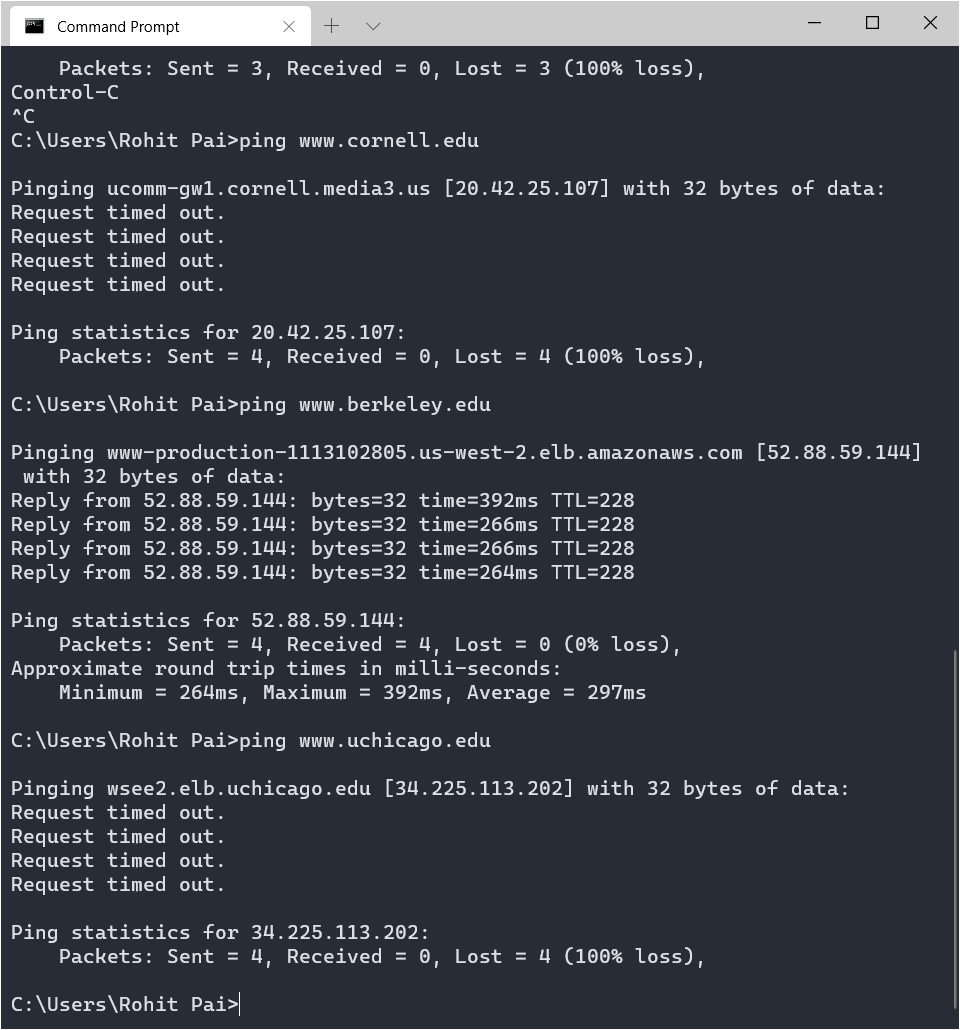
Pinging [www.cornell.edu](http://www.cornell.edu) (New York, USA)



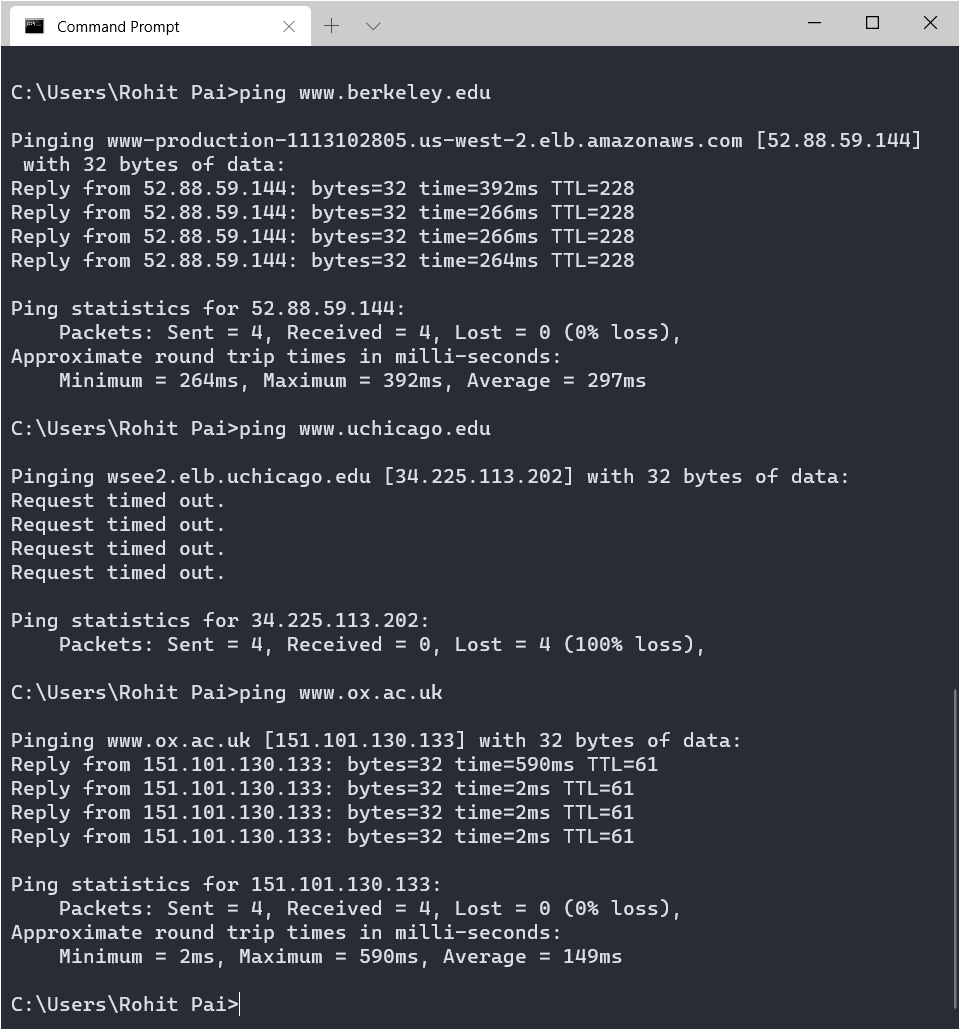
Pinging [www.berkeley.edu](http://www.berkeley.edu) (California, USA)



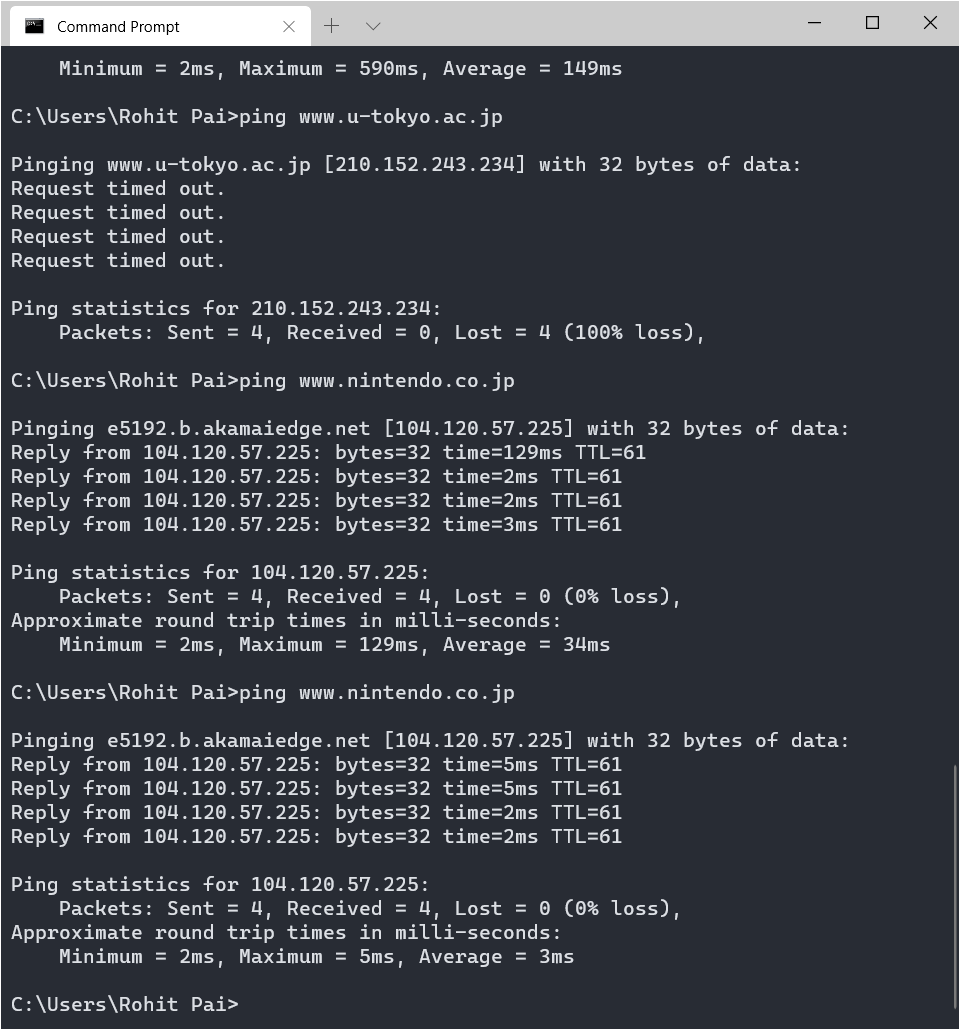
Pinging [www.uchicago.edu](http://www.uchicago.edu) (Illinois, USA)



Pinging www.ox.ac.uk (Oxford, England)



Pinging [www.nintendo.co.jp](http://www.nintendo.co.jp) (Japan)



Observations

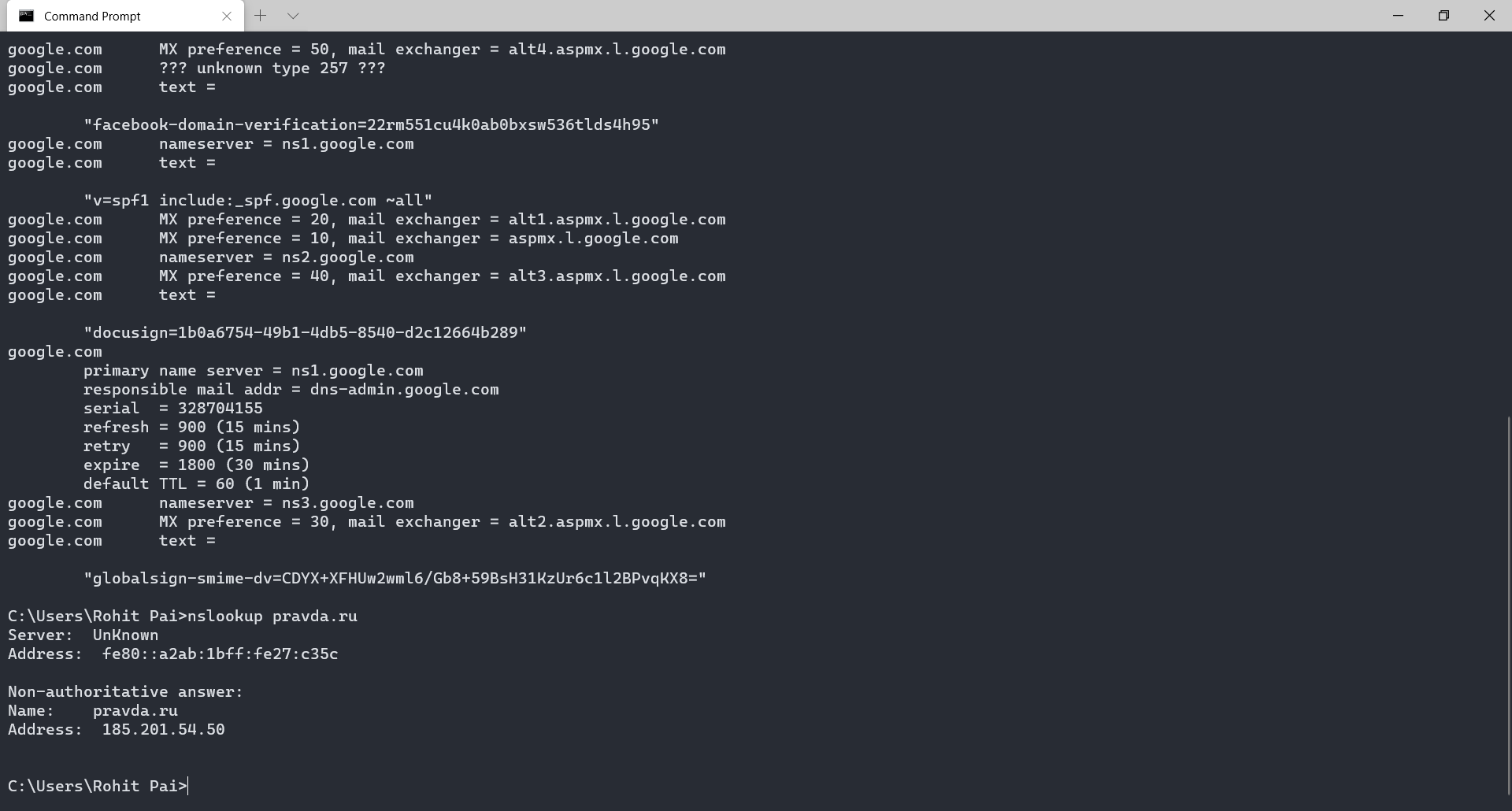
I observed that some of the sites that were mentioned in the exercise did not have their server open to ICMP requests and was the reason for Request Timed Out.

I also observed that RTT indeed depends on the physical distance between my device and the server. Pinging [www.nintendo.co.jp](http://www.nintendo.co.jp) in Japan (Average = 3 ms) took the least time as it is closest to Mumbai as compared to cities in USA (Average of 2 sites = 331.5 ms) and England (Average = 149 ms).

I also observed that the first packet that sent in while pinging a site for the first time has a very large RTT compared to the others.

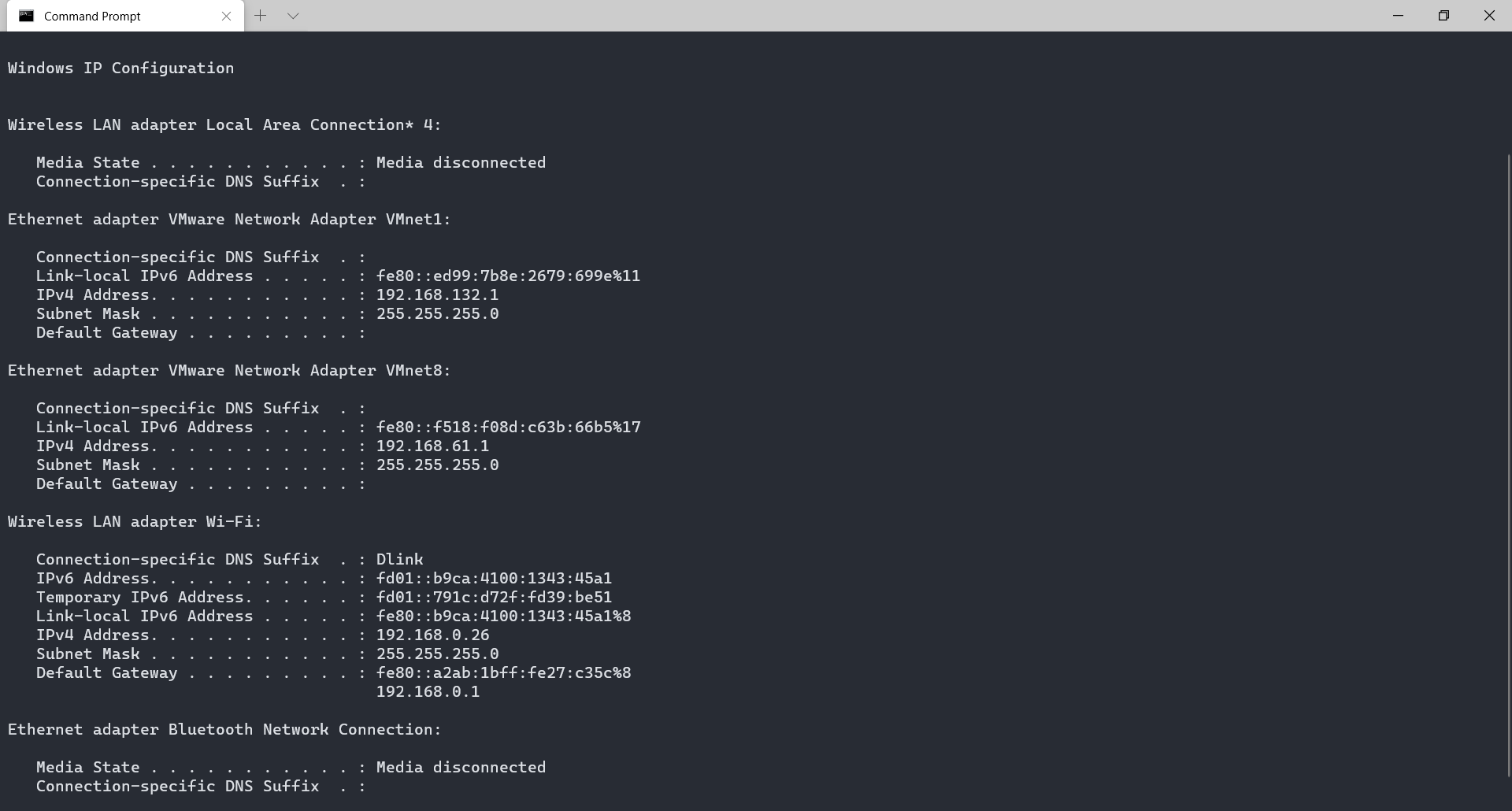
**nslookup** — The command nslookup <host> will do a DNS query to find and report the IP address (oraddresses) for a domain name or the domain name corresponding to an IP address. To do this, it contacts a "DNS server." Default DNS servers are part of a computer's network configuration. (For a static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

Using nslookup to find IP Address of pravda.ru



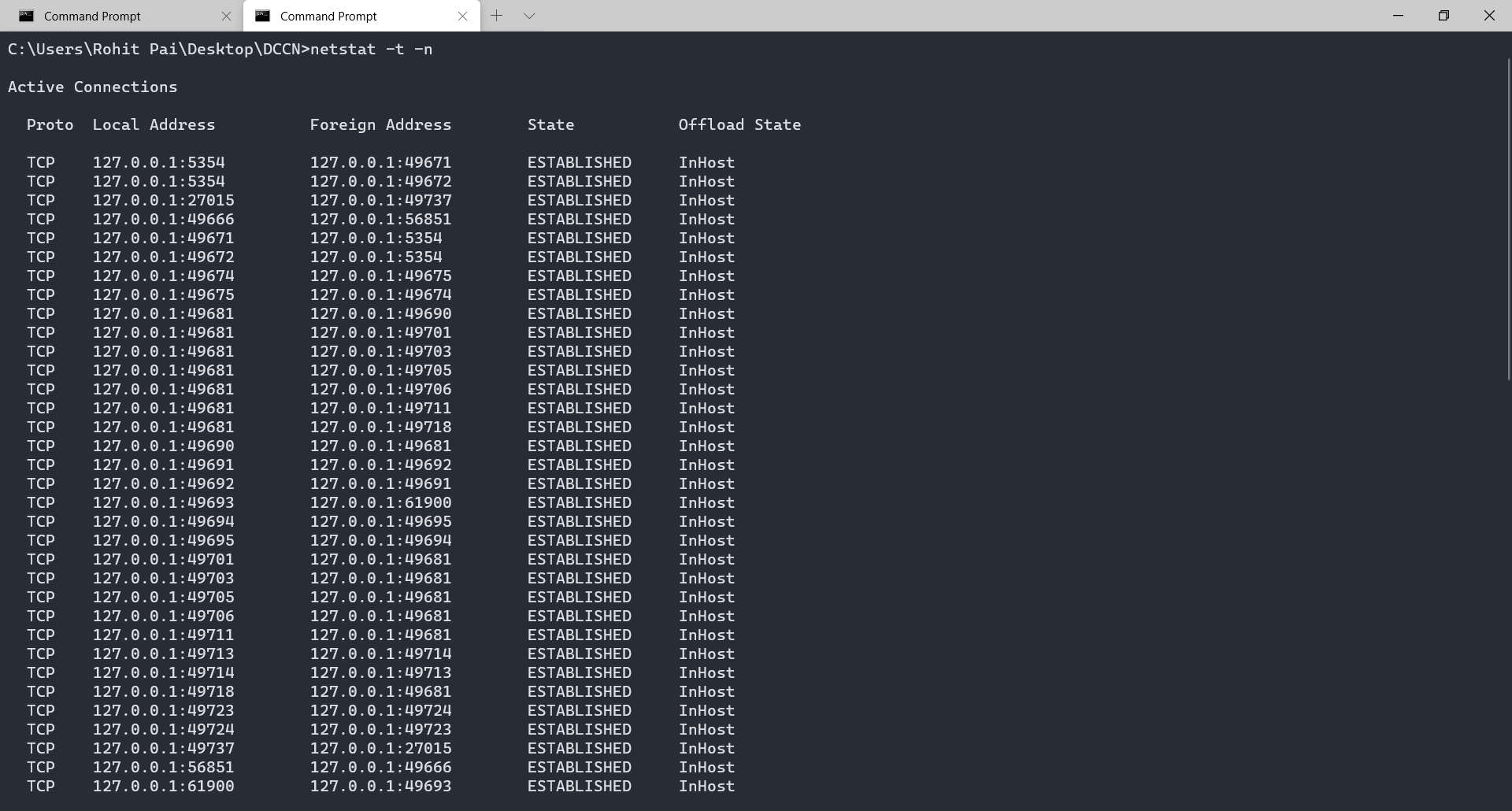
**ifconfig** — You used ifconfig in the previous lab. When used with no parameters, ifconfig reports someinformation about the computer's network interfaces. This usually includes lo which stands for localhost; it can be used for communication between programs running on the same computer. Linux often 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!!!.)

Output (This is the output for ipconfig. The output for ipconfig /all is stored in a log file) –



**netstat** — The netstat command gives information about network connections. I often use netstat -t -nwhich 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.)

Output (This is a sample of the output, the entire output is store in a log file) -



**telnet** — Telnet is an old program for remote login. It's not used so much for that any more, since it hasno 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: telnet <host> <port>. For example, to connect to the web server on www.spit.ac.in: telnet spit.ac.in 80

**traceroute** — Traceroute is discussed in man utility. The command traceroute <host> will show routers encountered by packets on their way from your computer to a specified <host>. For each n = 1, 2, 3,..., traceroute sends a packet with "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**

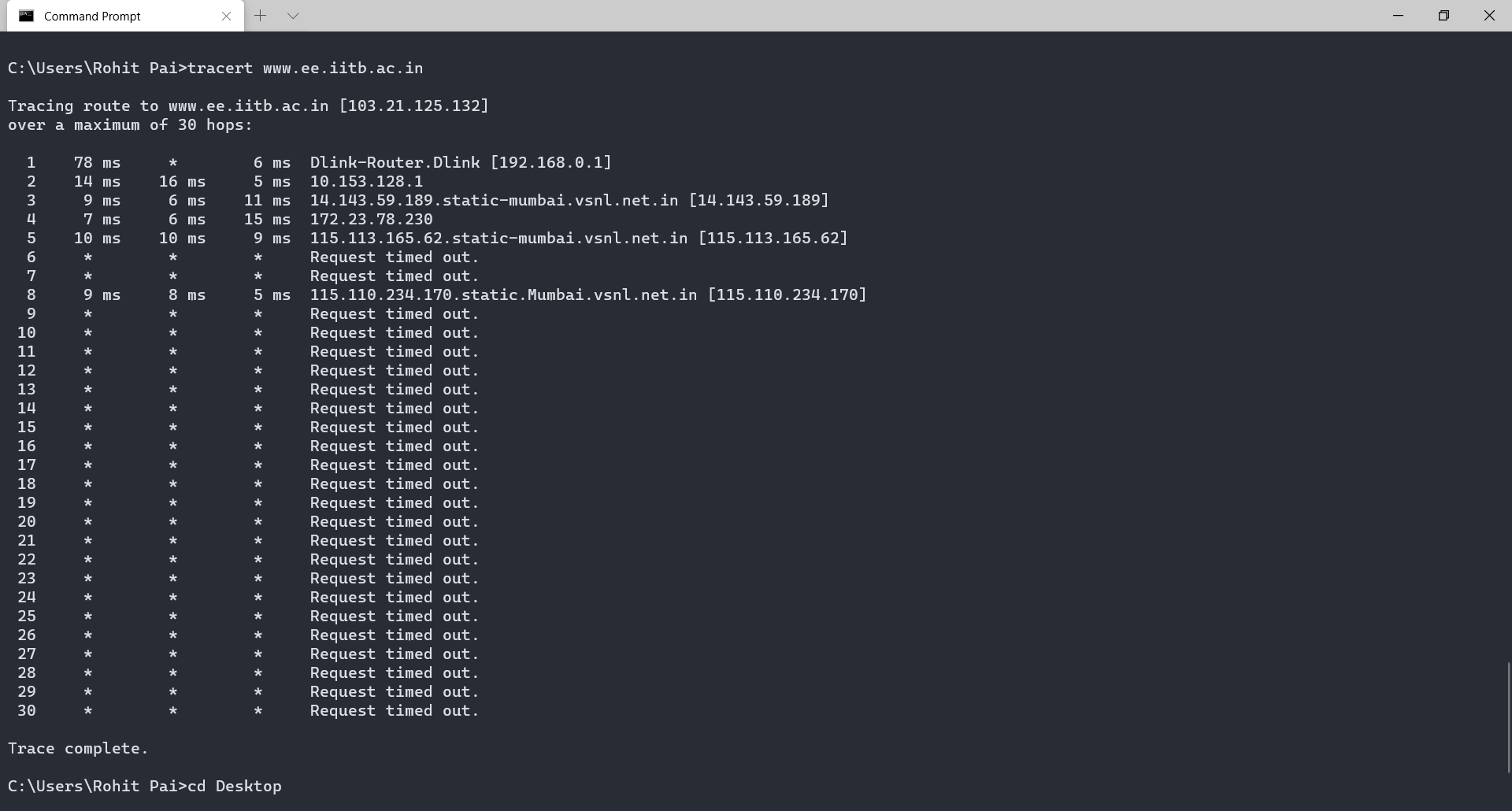
From **your machine** traceroute to the following hosts:

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

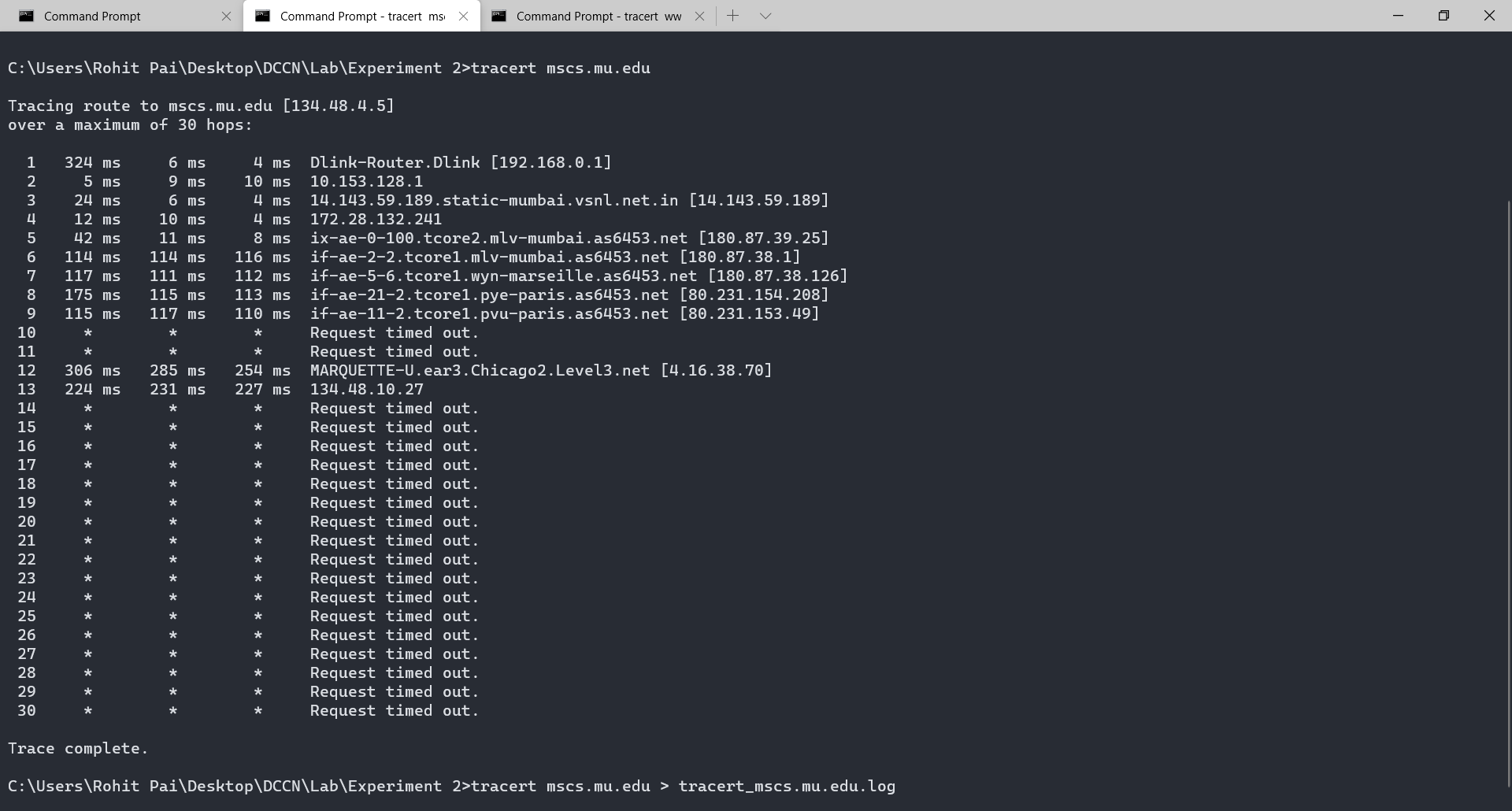
Store the output of each traceroute command in a separate file named

traceroute\_HOSTNAME.log, replacing HOSTNAME with the hostname for end-host you pinged (e.g., traceroute\_ee.iitb.ac.in.log).

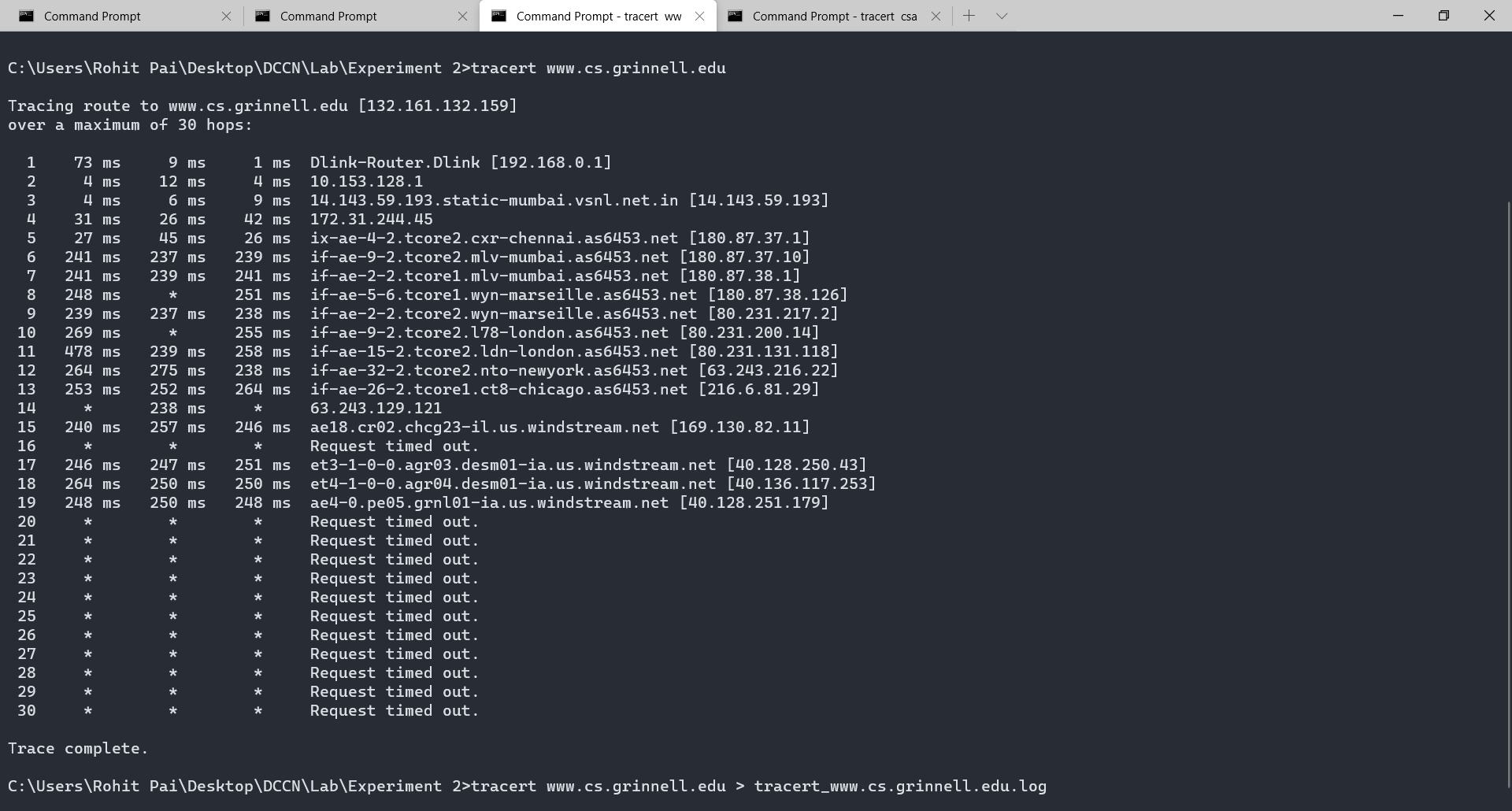
Tracing route to ee.iitb.ac.in



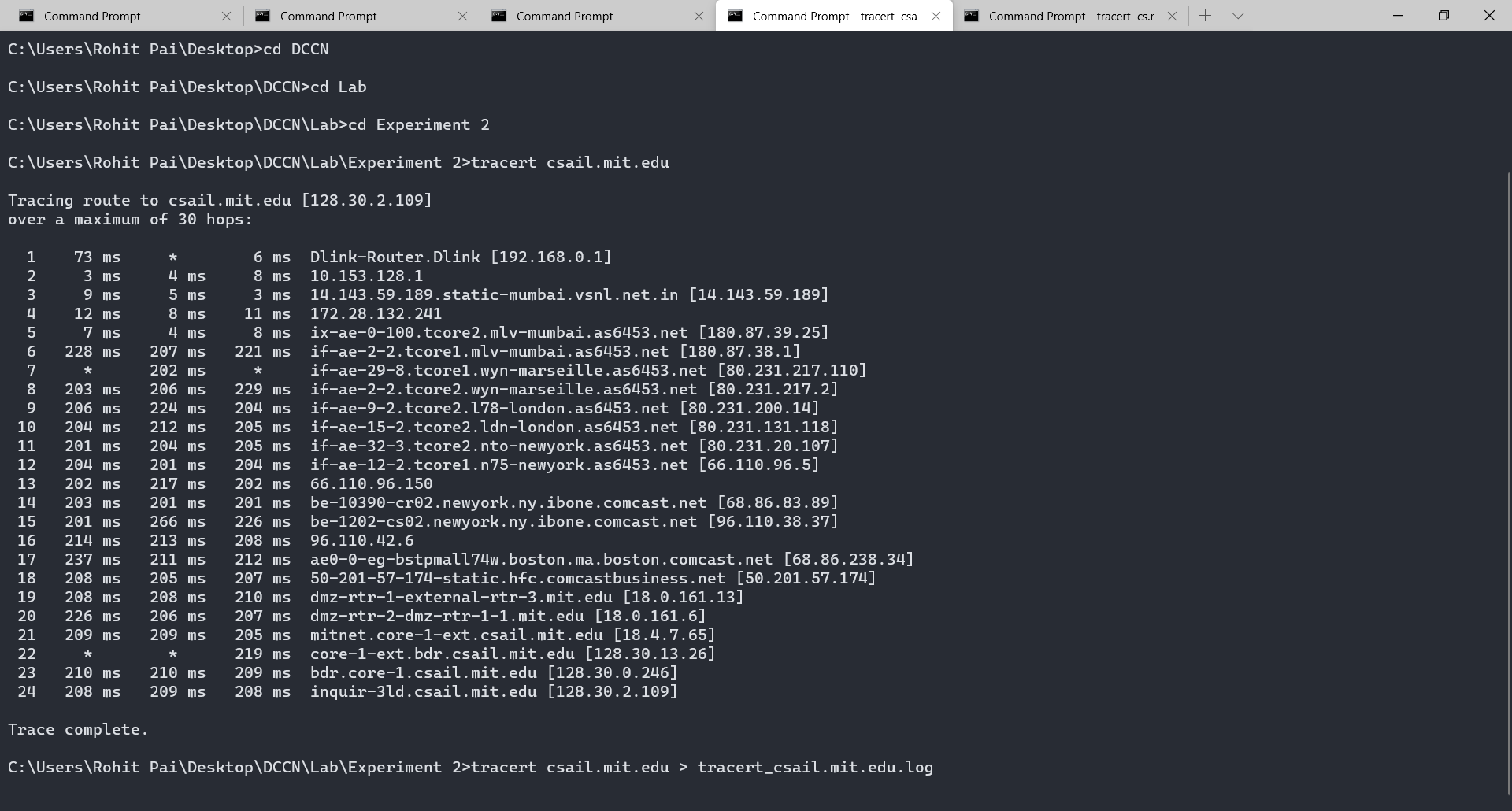
Tracing route to mscs.mu.edu



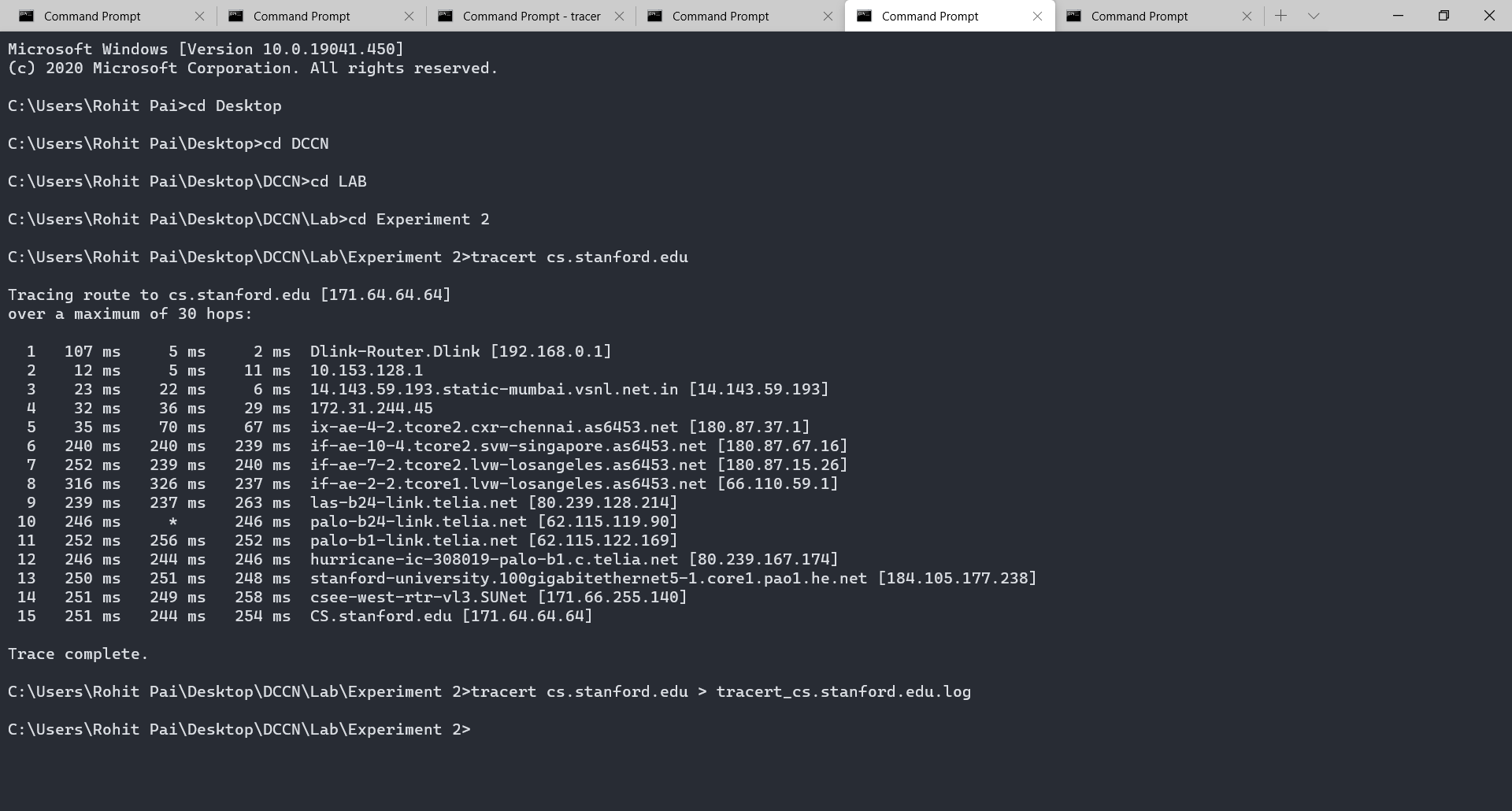
Tracing route to [www.cs.grinnell.edu](http://www.cs.grinnell.edu)



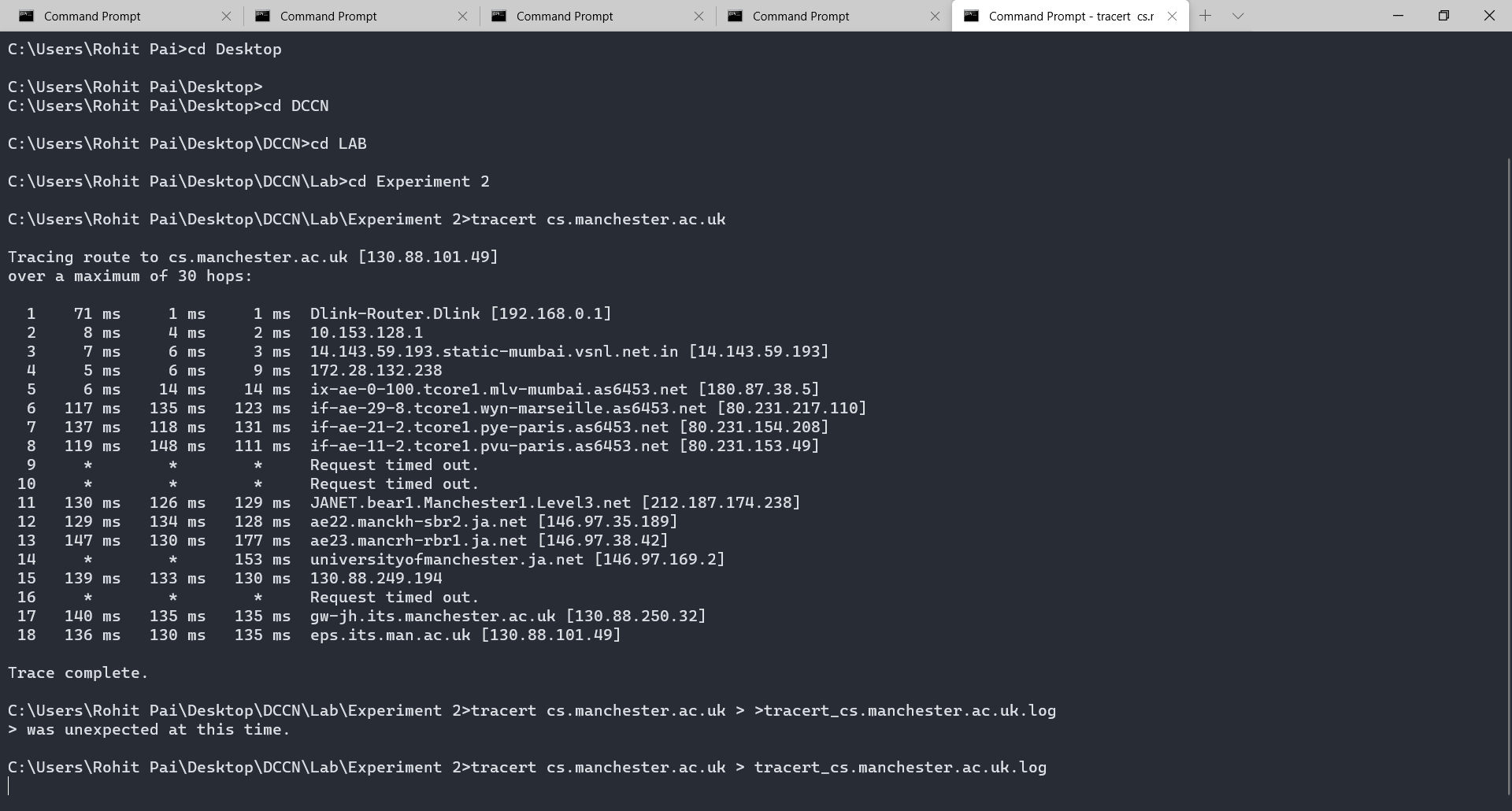
Tracing route to csail.mit.edu



Tracing route to cs.stanford.edu

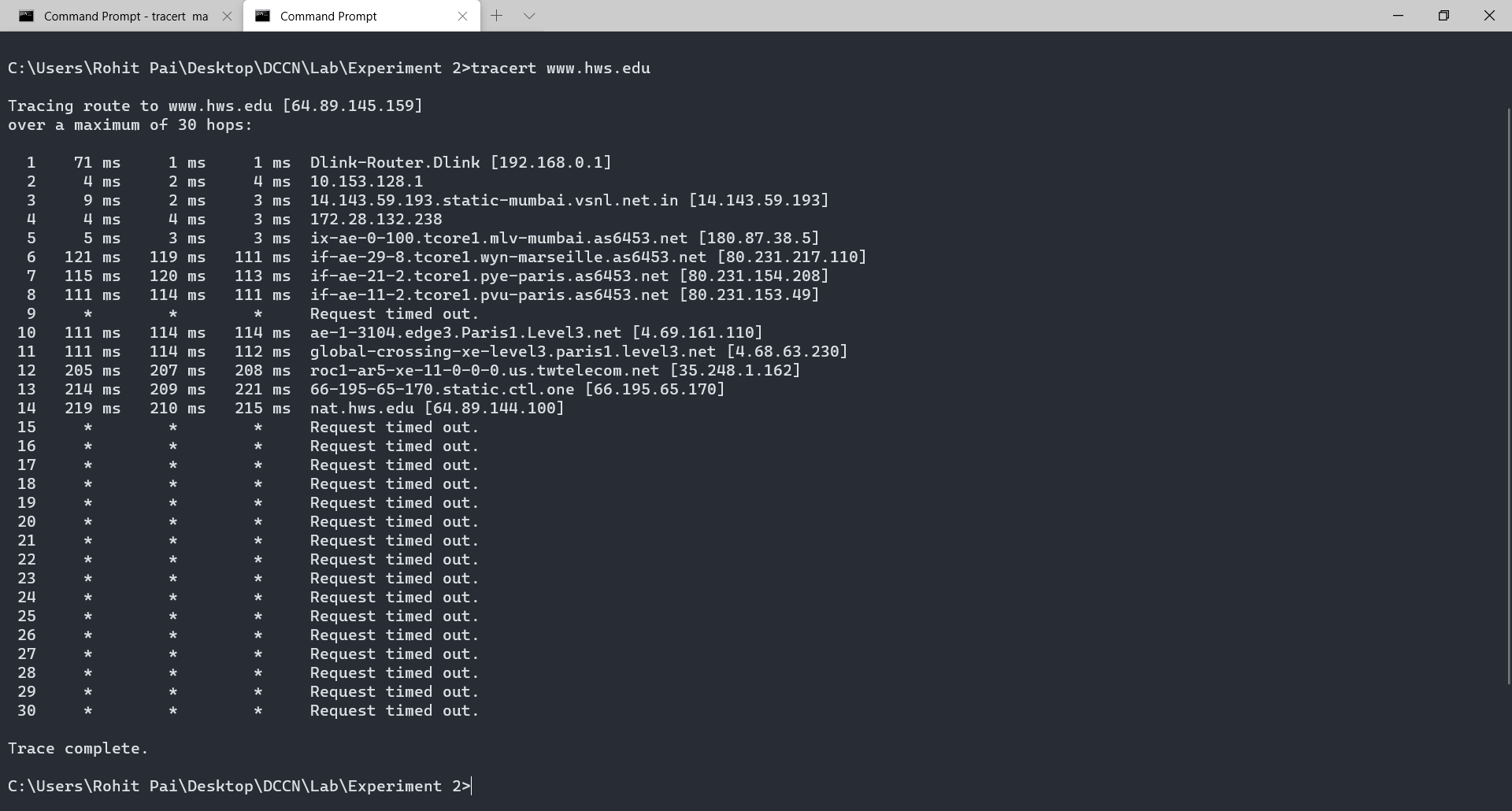


Tracing route to cs.manchester.ac.uk

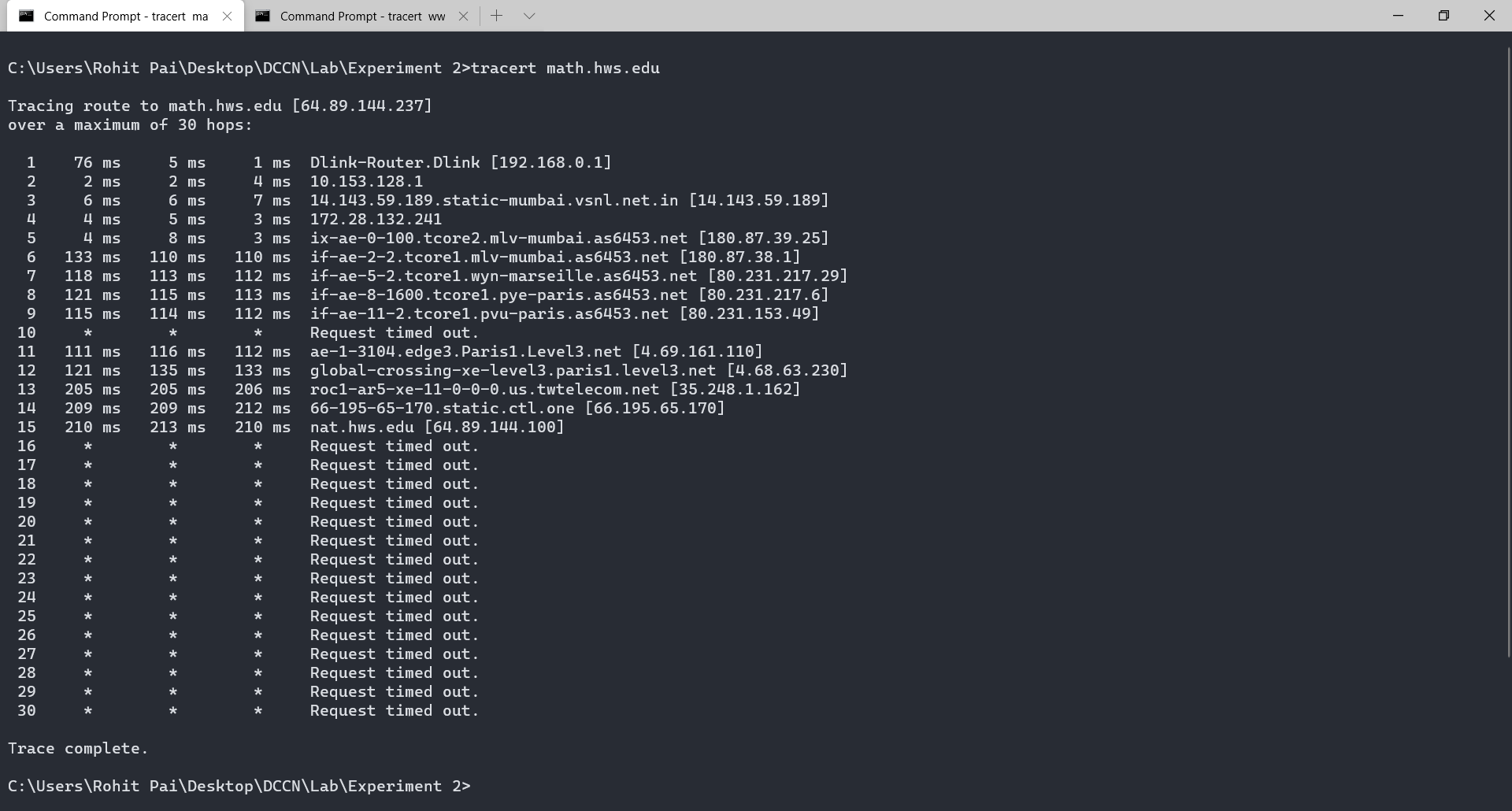


**Exercise 2:** (Very short.) Use traceroute to trace the route from your computer to math.hws.edu and towww.hws.edu. Explain the difference in the results.

Tracing route to [www.hws.edu](http://www.hws.edu)



Tracing route to math.hws.edu



Observations

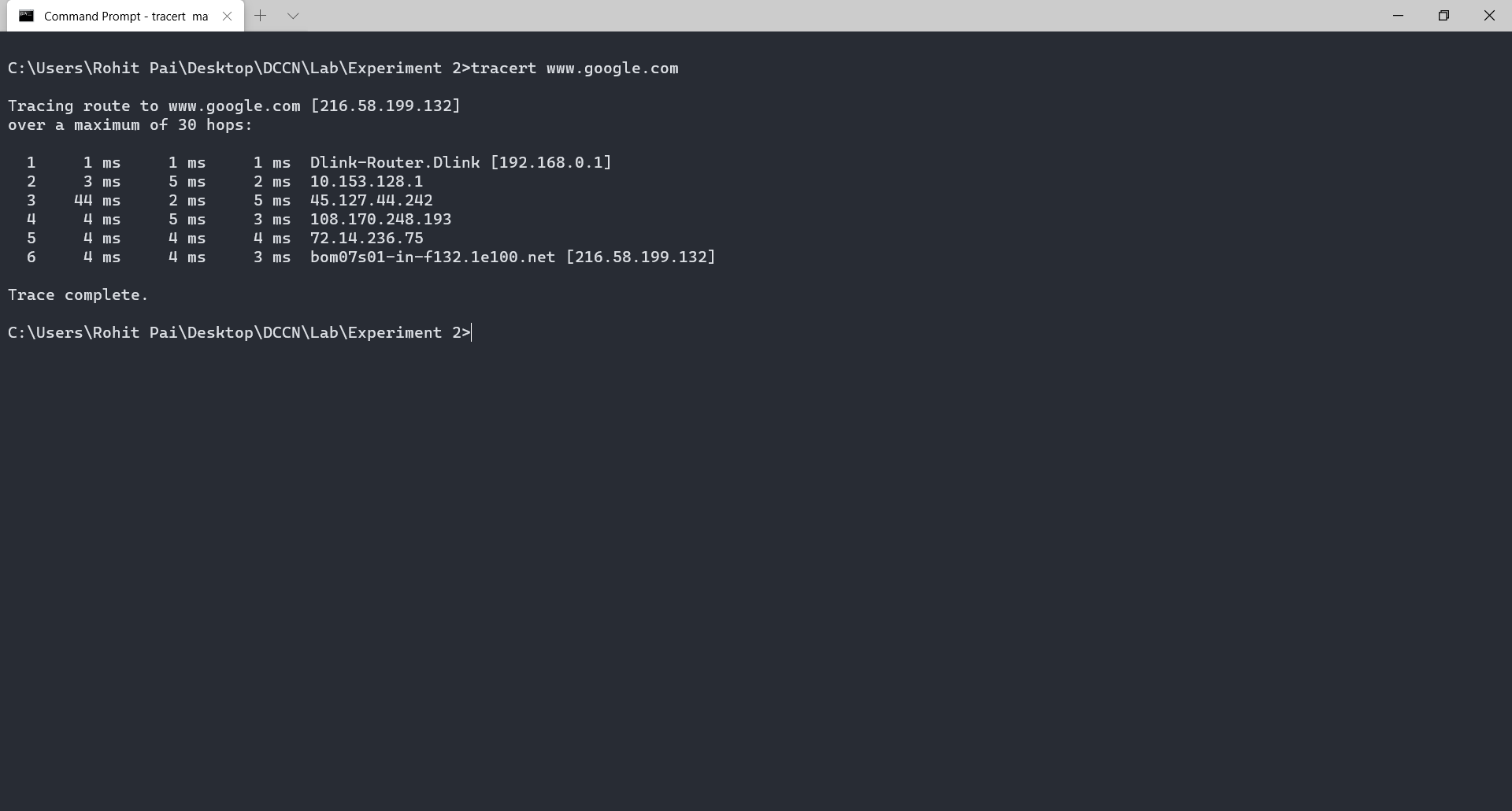
The difference is the two results is that the IP Address of the 2 destinations is slightly different.

[www.hws.edu](http://www.hws.edu) translates to 64.89.145.159 while math.hws.edu translates to 64.89.144.237.

I can also see that the request gets timed out after the node nat.hws.edu on both the sites meaning that the node after that has blocked ICMP requests.

**Exercise 3:** Two packets sent from the same source to the same destination do not necessarily follow thesame path through the net. Experiment with some sources that are fairly far away. Can you find cases where packets sent to the same destination follow different paths? How likely does it seem to be? What about when the packets are sent at very different times? Save some of the outputs from traceroute. (You can copy them from the Terminal window by highlighting and right-clicking, then paste into a text editor.) Come back sometime next week, try the same destinations again, and compare the results with the results from today. Report your observations.

Tracing route to [www.google.com](http://www.google.com) at 13:57 on 28th August, 2020



**Questions About Paths**

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

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

Yes, the first two nodes are always common to all the routes I have traced.

1. Is there a relationship between the number of nodes that show up in the traceroute and the location of the host? If so, what is this relationship?

No, there is no relation between the number of nodes and the location of the nodes. We might assume that the number of nodes increase with the physical distance but it is not guaranteed.

1. 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?

No, there isn’t any defined relationship between number of nodes and latency.

I can see that the latency increases with an increase in the number of nodes but we cannot guarantee that x nodes would give y latency.

**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*).

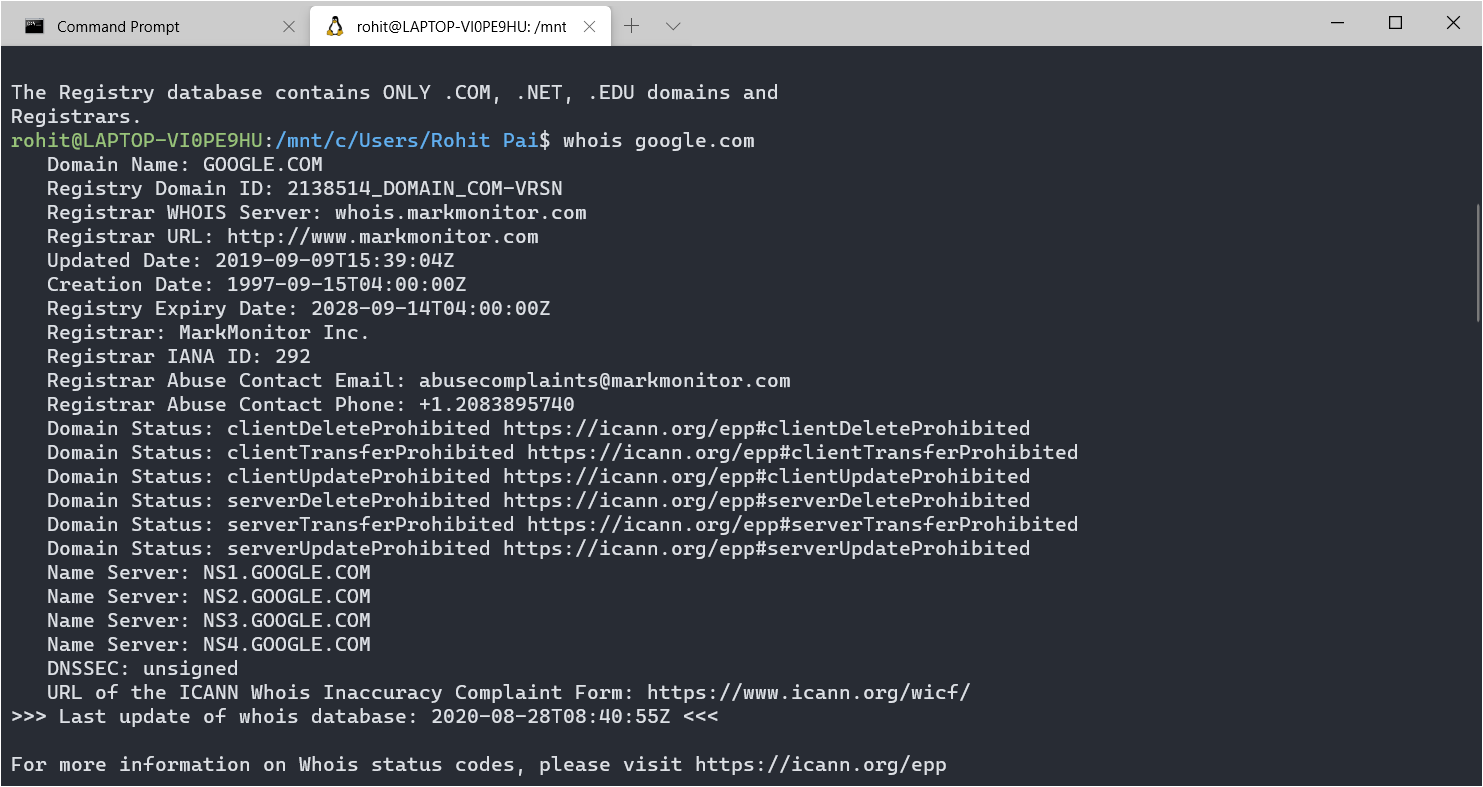
Note: Whois is not an inbuilt command in Windows. I installed the Sysinternals package by Mark Russinovich which has Whois implemented for windows.

Whois on windows (stored in a file called whois\_spit.ac.in.log)

Whois on Ubuntu (stored in a file called whois\_ubuntu\_spit.ac.in.log)

**Exercise 4:** (Short.) Use*whois*to investigate a well-known web site such as google.com oramazon.com, and write a couple of sentences about what you find out.

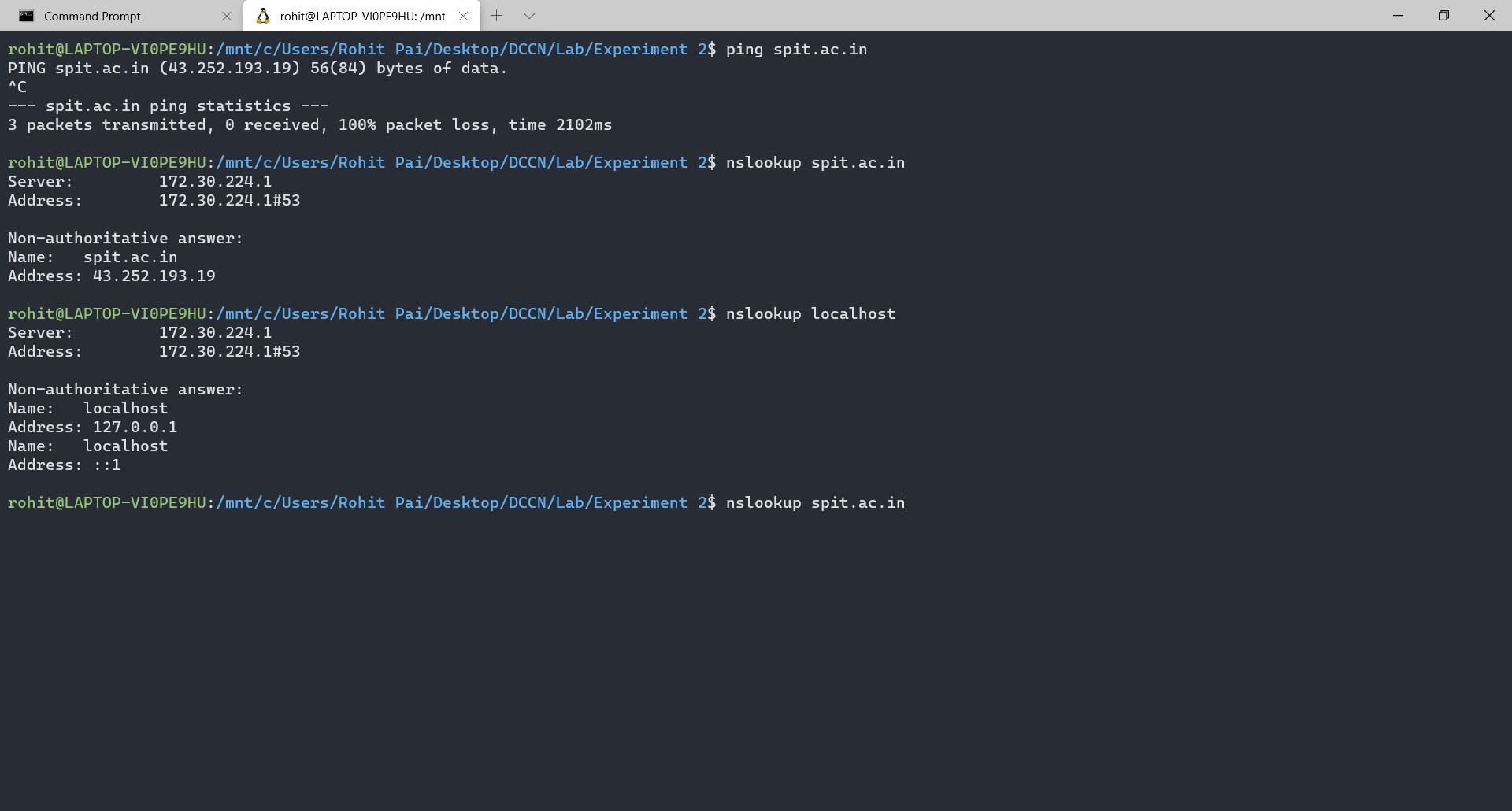
Redacted output (Full output in whois\_ubuntu\_google.com.log)



Running the whois command on google.com gives us information on the registry domain id of Google and the name servers of google.com. We also get details on the registrant organization, admin organization and tech organization for the domain. This includes the name, state and country of each organization.

**Exercise 5:** (Should be short.) Because of NAT, the domain name*spit.ac.in*has a different IP addressoutside 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.

I used nslookup to find out the external IP address of spit.ac.in



The IP Address of spit.ac.in is 43.252.193.19

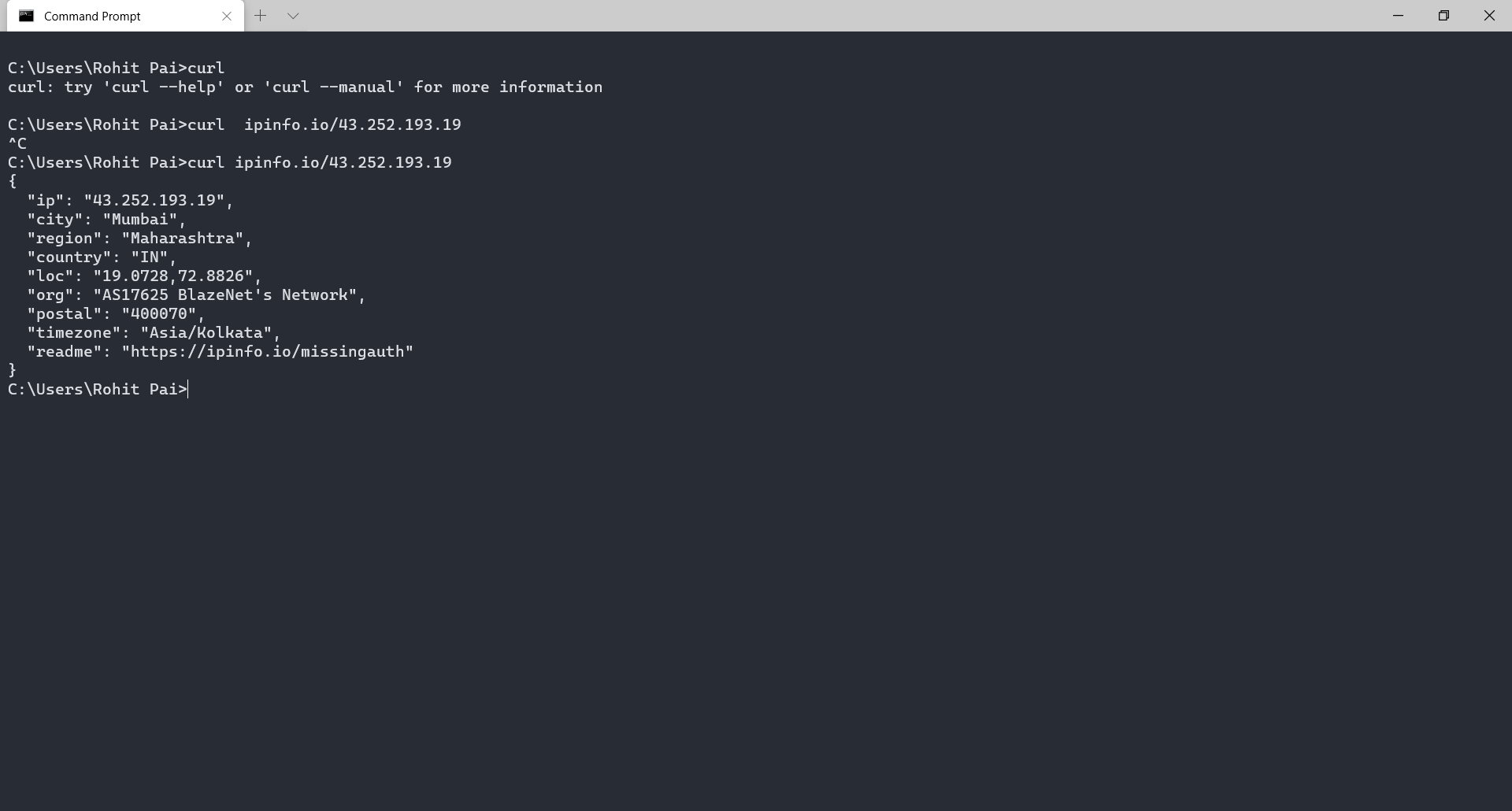
Geolocation — A geolocation service tries to tell, approximately, where a given IP address is located physically. They can't be completely accurate—but they probably get at least the country right most of the time.

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.)

Using geolocation program to find the location of spit.ac.in’s server



**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.