Computer Networks Assignment 1

Souparno Ghose (2022506)

August 30, 2024

Question 1

(a) Learn to use the `ifconfig` command, and figure out the IP address of your network interface. Put a screenshot.

(b) Go to the webpage `https://www.whatismyip.com` and find out what IP is shown for your machine. Are they identical or different? Why?

Solution

(a) The IP Address using `ifconfig` is 172.24.113.215.

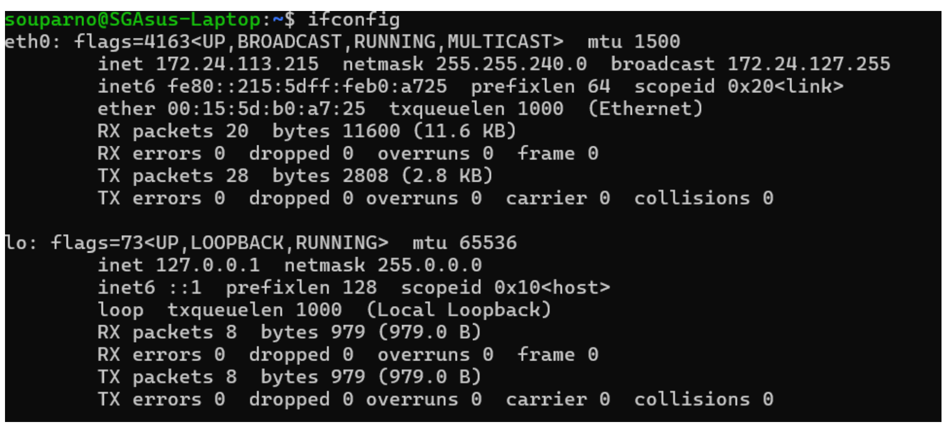


Figure 1: Screenshot of the `ifconfig` command output

(b) The IP address shown on the website is different from the IP address shown using `ifconfig`. The IP Address through `https://www.whatismyip.com` is 103.25.231.126.

The IP address seen on my local machine, like `172.24.113.215`, is a private IP used within the local network. However, when checking a site like `https://www.whatismyip.com`, it shows a different public IP assigned by the ISP, which is visible to the outside world. This difference occurs because the router uses Network Address Translation (NAT) to translate between private and public IP addresses.

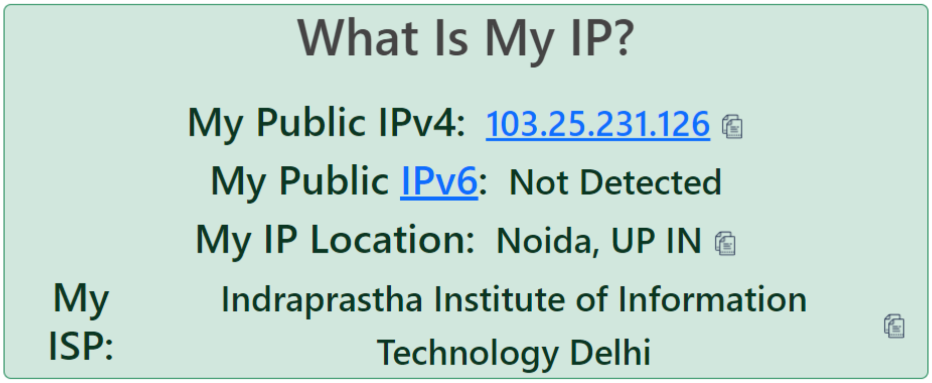


Figure 2: Screenshot of the IP address using `https://www.whatismyip.com`

Question 2

Change the IP address of your network interface using the command line. Put a screenshot that shows the change. Revert to the original IP address.

Solution

The command for changing the IP address is `sudo ifconfig eth0 192.168.1.50`.
To revert back to the original IP address, use `sudo ifconfig eth0 172.24.113.215`.

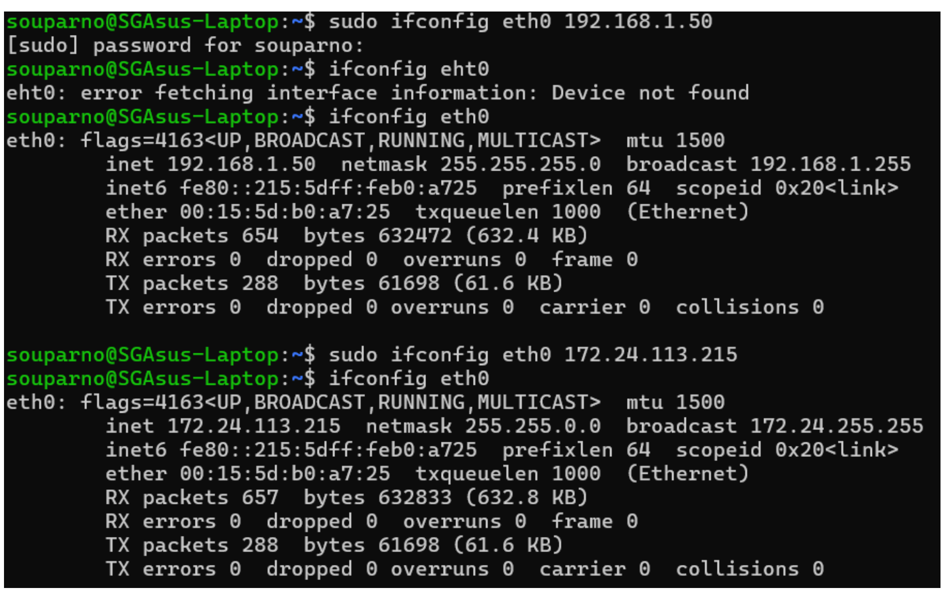


Figure 3: Screenshot of the command to change and reverting the IP address

Question 3

(a) Use `netcat` to set up a TCP client/server connection between your VM and host machine. If you are not using a VM, you can set up the connection with localhost. Put a screenshot.

(b) Determine the state of this TCP connection(s) at the client node. Put a screenshot.

Solution

(a) To establish a TCP client-server connection using `netcat`, I used the following commands: `nc -lv 3737` to set up a server listening on port `3737` and `nc 127.0.0.1 3737` to connect as a client to the server on the same port. The server command (`nc -lv 3737`) initializes the server in listening mode with verbose output, while the client command (`nc 127.0.0.1 3737`) connects to the server running on `localhost`. This setup allows for testing the TCP connection between the client and server on the same machine.

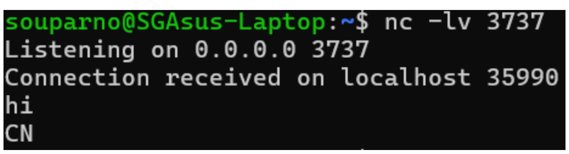


Figure 4: Server setup using netcat

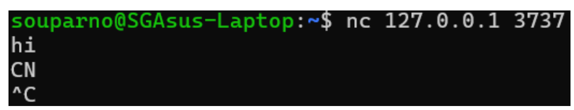


Figure 5: Client connection to server using netcat

(b) To determine the state of the TCP connection on the client side, first establish a connection using `netcat` with the server by running `nc -lv 3737` on the server and `nc 127.0.0.1 3737` on the client. After ensuring the connection was active, open a new terminal on the client machine and executed the command `netstat -tn`. This command listed all TCP connections with numerical addresses, showing the local and foreign addresses, as well as the state of each connection. The output confirmed the status of the connection by displaying the `ESTABLISHED` state for the relevant port.

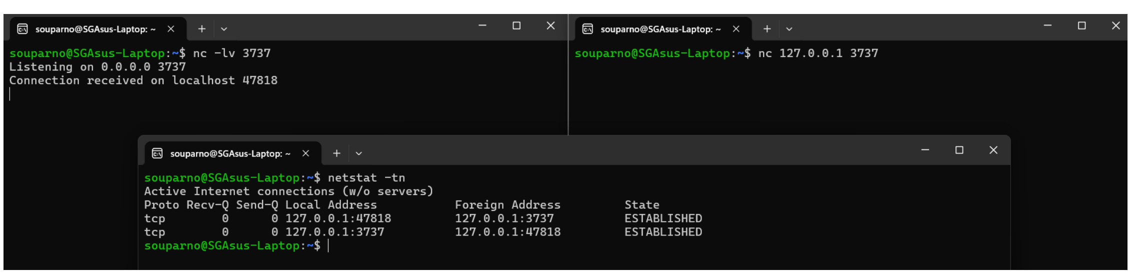


Figure 6: Output of `netstat -tn` showing ESTABLISHED connection

Question 4

(a) Get an authoritative result for `google.in` using `nslookup`. Put a screenshot. Explain how you did it.

(b) Find out the time to live for any website on the local DNS. Put a screenshot. Explain in words (with unit) after how much time this entry would expire from the local DNS server.

Solution

(a) The `nslookup` command I used queried DNS records for the domain `google.in` to obtain authoritative information. By setting the query type to `soa` (Start of Authority), the command requested the SOA record, which provides details about the domain's primary DNS server and its administrative settings. The output showed the SOA record information, including the name server and administrative contact. This command helped retrieve the authoritative DNS data for the domain, demonstrating how to obtain and interpret essential DNS records.

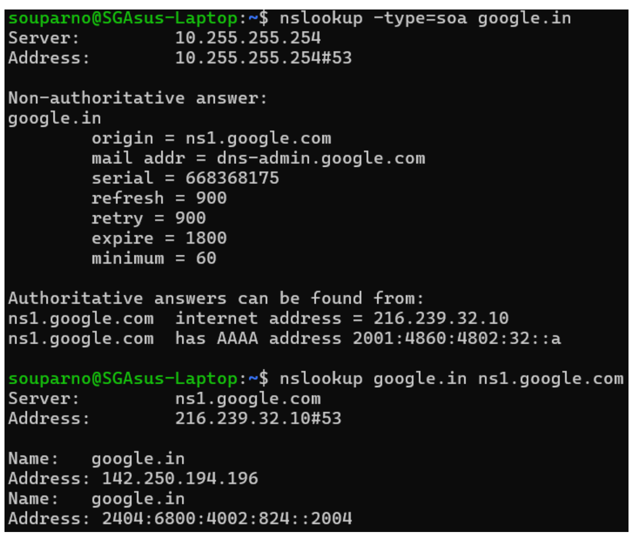


Figure 7: SOA record output from `nslookup` command

(b) The `dig` command is used to query DNS records and provides detailed information, including the Time to Live (TTL) value, which indicates how long a DNS entry is cached by the local DNS server. By running `dig <domain>`, such as `dig google.in`, the command returns the TTL value in the ANSWER SECTION of the output. This TTL value, expressed in seconds, specifies the duration for which the DNS record will be considered valid before needing to be refreshed. For example, a TTL of 241 seconds means the DNS entry will expire after 4 minutes and 1 second, prompting a new query to retrieve updated information. This helps determine the caching period of DNS records effectively.

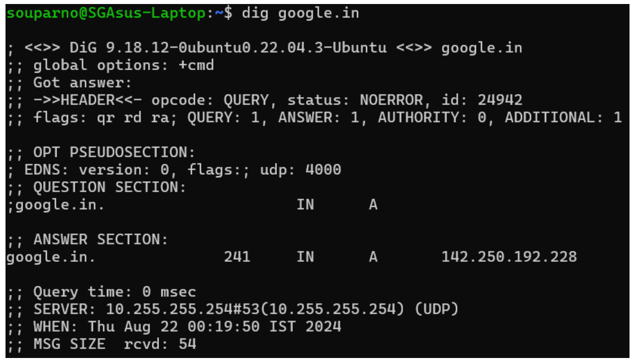


Figure 8: TTL value from `dig` command for `google.in`

Question 5

(a) Run the command, `traceroute google.in`. How many intermediate hosts do you see? What are the IP addresses? Compute the average latency to each intermediate host. Put a screenshot.

(b) Send 50 ping messages to `google.in`, Determine the average latency. Put a screenshot.

(c) Add up the ping latency of all the intermediate hosts obtained in (a) and compare with (b). Are they matching, explain?

(d) Take the maximum ping latency amongst the intermediate hosts (in (a)) and compare it with (b). Are they matching, explain?

(e) You may see multiple entries for a single hop while using the traceroute command. What do these entries mean?

(f) Send 50 ping messages to `stanford.edu`, Determine the average latency. Put a screenshot.

(g) Run the command, `traceroute stanford.edu`. Compare the number of hops between `google.in` and `stanford.edu` (between the `traceroute` result of `google.in` and `stanford.edu`).

(h) Can you explain the reason for the latency difference between `google.in` and `stanford.edu` (see (b) & (f))?

Solution

(a) Based on the `traceroute google.in` command output:

1. **Number of Intermediate Hosts:**

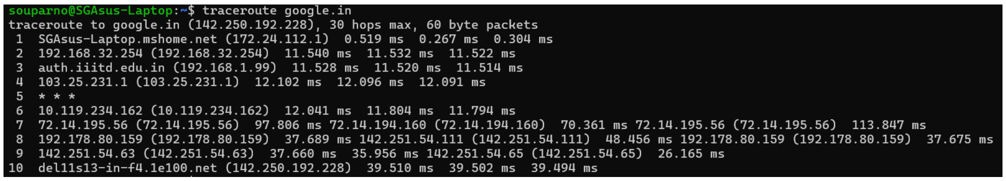
   - If we exclude the first host (which is our own laptop), there are 7 intermediate hosts (ignoring the ones that show `* *`).

- If we include the first host, there are 8 intermediate hosts (ignoring the ones that show `* *`).

2. **IP Addresses of Intermediate Hosts:**
   - **Excluding the first host (7 intermediate hosts):**
     - 192.168.32.254
     - 192.168.1.99
     - 103.25.231.1
     - 10.119.234.162
     - 72.14.195.56 / 72.14.194.160
     - 192.178.80.159 / 142.251.54.111
     - 142.251.54.63 / 142.251.54.65
   - **Including the first host (8 intermediate hosts):**
     - 172.24.112.1
     - 192.168.32.254
     - 192.168.1.99
     - 103.25.231.1
     - 10.119.234.162
     - 72.14.195.56 / 72.14.194.160
     - 192.178.80.159 / 142.251.54.111
     - 142.251.54.63 / 142.251.54.65

3. **Average Latency to Each Intermediate Host:**
   - **Excluding the first host:**
     - **192.168.32.254:** $(11.540 + 11.532 + 11.522)/3 = 11.531\,\mathrm{ms}$
     - **192.168.1.99:** $(11.528 + 11.520 + 11.514)/3 = 11.520\,\mathrm{ms}$
     - **103.25.231.1:** $(12.102 + 12.096 + 12.091)/3 = 12.096\,\mathrm{ms}$
     - **10.119.234.162:** $(12.041 + 11.804 + 11.794)/3 = 11.880\,\mathrm{ms}$
     - **72.14.195.56 / 72.14.194.160:** $(97.806 + 70.361 + 113.847)/3 = 94.671\,\mathrm{ms}$
     - **192.178.80.159 / 142.251.54.111:** $(37.689+48.456+37.675)/3 = 41.273\,\mathrm{ms}$
     - **142.251.54.63 / 142.251.54.65:** $(37.660 + 35.956 + 26.165)/3 = 33.927\,\mathrm{ms}$
   - **Including the first host:**
     - **172.24.112.1:** $(0.519 + 0.267 + 0.304)/3 = 0.363\,\mathrm{ms}$
     - **192.168.32.254:** $(11.540 + 11.532 + 11.522)/3 = 11.531\,\mathrm{ms}$
     - **192.168.1.99:** $(11.528 + 11.520 + 11.514)/3 = 11.520\,\mathrm{ms}$
     - **103.25.231.1:** $(12.102 + 12.096 + 12.091)/3 = 12.096\,\mathrm{ms}$
     - **10.119.234.162:** $(12.041 + 11.804 + 11.794)/3 = 11.880\,\mathrm{ms}$

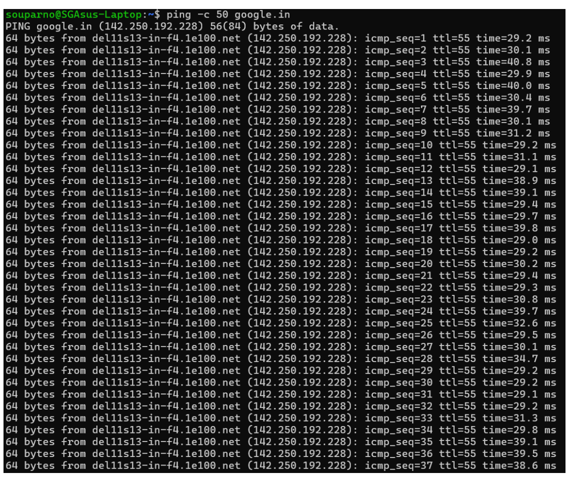
- **72.14.195.56 / 72.14.194.160:** $(97.806 + 70.361 + 113.847)/3 = 94.671\,\text{ms}$
- **192.178.80.159 / 142.251.54.111:** $(37.689{+}48.456{+}37.675)/3 = 41.273\,\text{ms}$
- **142.251.54.63 / 142.251.54.65:** $(37.660 + 35.956 + 26.165)/3 = 33.927\,\text{ms}$



```
souparno@SGAsus-Laptop:~$ traceroute google.in
traceroute to google.in (142.250.192.228), 30 hops max, 60 byte packets
 1  SGAsus-Laptop.mshome.net (172.24.112.1)  0.519 ms  0.267 ms  0.304 ms
 2  192.168.32.254 (192.168.32.254)  11.540 ms  11.532 ms  11.522 ms
 3  auth.iiitd.edu.in (192.168.1.99)  11.528 ms  11.520 ms  11.514 ms
 4  103.25.231.1 (103.25.231.1)  12.102 ms  12.096 ms  12.091 ms
 5  * * *
 6  10.119.234.162 (10.119.234.162)  12.041 ms  11.804 ms  11.794 ms
 7  72.14.195.56 (72.14.195.56)  97.806 ms 72.14.194.160 (72.14.194.160)  70.361 ms 72.14.195.56 (72.14.195.56)  113.847 ms
 8  192.178.80.159 (192.178.80.159)  37.689 ms 142.251.54.111 (142.251.54.111)  48.456 ms 192.178.80.159 (192.178.80.159)  37.675 ms
 9  142.251.54.63 (142.251.54.63)  37.660 ms  35.956 ms 142.251.54.65 (142.251.54.65)  26.165 ms
10  del11s13-in-f4.1e100.net (142.250.192.228)  39.510 ms  39.502 ms  39.494 ms
```

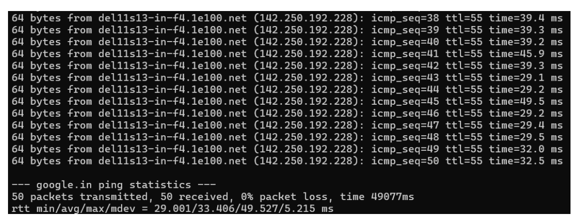
Figure 9: Traceroute command output visualization

(b) Used the `ping` command to send 50 packets to `google.in` and observed the results. The average latency, as reported by the command, is 33.406 milliseconds. This value is calculated from the round-trip times of all the packets sent, giving an indication of the typical time it takes for a packet to travel to the destination and back.



```
souparno@SGAsus-Laptop:~$ ping -c 50 google.in
PING google.in (142.250.192.228) 56(84) bytes of data.
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=1 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=2 ttl=55 time=30.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=3 ttl=55 time=40.8 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=4 ttl=55 time=29.9 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=5 ttl=55 time=40.0 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=6 ttl=55 time=30.4 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=7 ttl=55 time=39.7 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=8 ttl=55 time=30.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=9 ttl=55 time=31.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=10 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=11 ttl=55 time=31.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=12 ttl=55 time=29.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=13 ttl=55 time=38.9 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=14 ttl=55 time=39.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=15 ttl=55 time=29.4 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=16 ttl=55 time=29.7 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=17 ttl=55 time=39.8 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=18 ttl=55 time=29.0 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=19 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=20 ttl=55 time=30.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=21 ttl=55 time=29.4 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=22 ttl=55 time=29.3 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=23 ttl=55 time=30.8 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=24 ttl=55 time=39.7 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=25 ttl=55 time=32.6 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=26 ttl=55 time=29.5 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=27 ttl=55 time=30.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=28 ttl=55 time=34.7 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=29 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=30 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=31 ttl=55 time=29.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=32 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=33 ttl=55 time=31.3 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=34 ttl=55 time=29.8 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=35 ttl=55 time=39.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=36 ttl=55 time=39.5 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=37 ttl=55 time=38.6 ms
```

Figure 10: Ping Results 1



```
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=38 ttl=55 time=39.4 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=39 ttl=55 time=39.3 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=40 ttl=55 time=39.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=41 ttl=55 time=45.9 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=42 ttl=55 time=39.3 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=43 ttl=55 time=29.1 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=44 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=45 ttl=55 time=49.5 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=46 ttl=55 time=29.2 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=47 ttl=55 time=29.4 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=48 ttl=55 time=29.5 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=49 ttl=55 time=32.0 ms
64 bytes from del11s13-in-f4.1e100.net (142.250.192.228): icmp_seq=50 ttl=55 time=32.5 ms

--- google.in ping statistics ---
50 packets transmitted, 50 received, 0% packet loss, time 49077ms
rtt min/avg/max/mdev = 29.001/33.406/49.527/5.215 ms
```

Figure 11: Ping Results 2

(c) **Sum of Intermediate Host Latencies (from traceroute)(Excluding the first host):**

$$11.531 + 11.520 + 12.096 + 11.880 + 94.671 + 41.273 + 33.927 = 216.898\,\text{ms}$$

Average Ping Latency:

$$33.406\,\text{ms}$$

Reason:

The sum of latencies for intermediate hosts obtained from traceroute is 216.898 ms, which is significantly higher than the average ping latency of 33.406 ms. This difference occurs because traceroute accumulates delays from each intermediate hop, whereas ping measures the round-trip time directly to the final destination, resulting in a lower, more immediate latency.

(d) **Maximum Ping Latency Among Intermediate Hosts:**

From the `traceroute` data, the maximum latency among the intermediate hosts is 113.847 ms (for IP `72.14.195.56`).

Maximum Ping Latency:

From the `ping` command data, the maximum ping latency is 49.527 ms.

Reason:

The maximum latency observed for an intermediate host (113.847 ms) is significantly higher than the maximum ping latency (49.527 ms). This discrepancy occurs because `traceroute` measures the time taken to reach each hop along the route, which can include network congestion or routing delays at specific points. The maximum ping latency, however, reflects the highest round-trip time directly to the final destination and is generally lower because it doesn't account for the individual delays at each intermediate hop.

(e) Multiple entries for a single hop in a `traceroute` command output indicate that the command sends multiple probes (in this case three) to each hop along the route. Each probe measures the time it takes for a packet to reach that hop and return. The multiple entries show the response times for each of these probes. These times can vary due to factors like network congestion, routing changes, or packet processing delays at the intermediate routers. Seeing multiple entries allows us to get a more accurate and comprehensive understanding of the latency and reliability of each hop.

(f) The average latency obtained by pinging `stanford.edu` 50 times came out to be 286.919 ms.

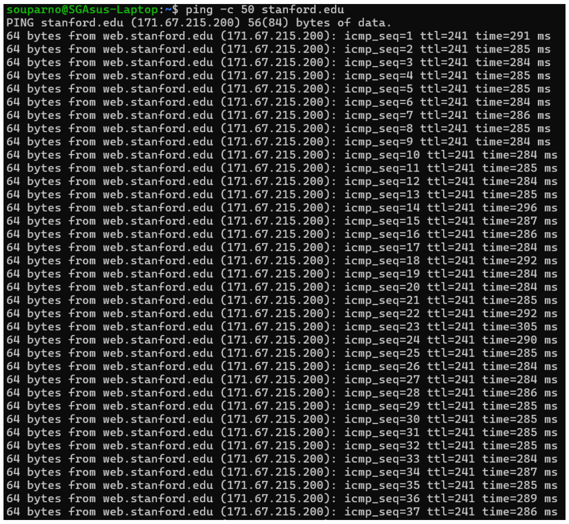


Figure 12: Ping Results 1

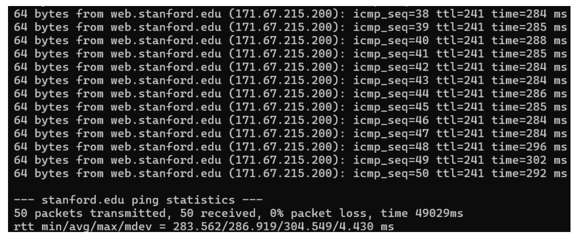


Figure 13: Ping Results 2

(g) When we compare the number of hops between `google.in` and `stanford.edu` using the `traceroute` results (ignoring the * * * hops), we get the following:

- **Traceroute to `stanford.edu`**: Number of hops: 10 (ignoring the * * * hops from 9 to 24)
- **Traceroute to `google.in`**: Number of hops: 9

Reason: The `traceroute` to `stanford.edu` involves more hops (10) compared to `google.in` (9). This suggests that the route to `stanford.edu` traverses through more network nodes, which could contribute to higher latency compared to `google.in`. This is consistent with the earlier explanation that accessing Google's global network is typically faster due to its optimized infrastructure.

If we included the * * * hops in the count, the number of hops for `stanford.edu` would be significantly higher, indicating many potential points of delay or unreachable nodes along the route.

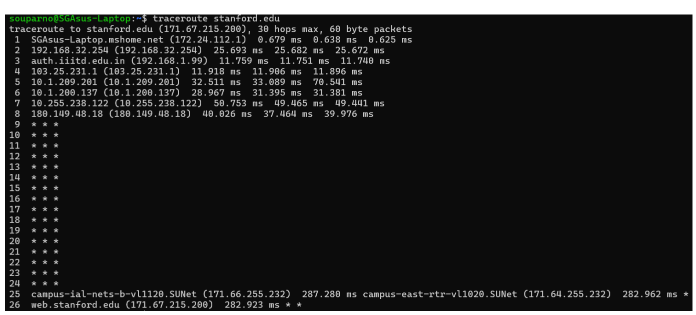


Figure 14: Traceroute Results to stanford.edu

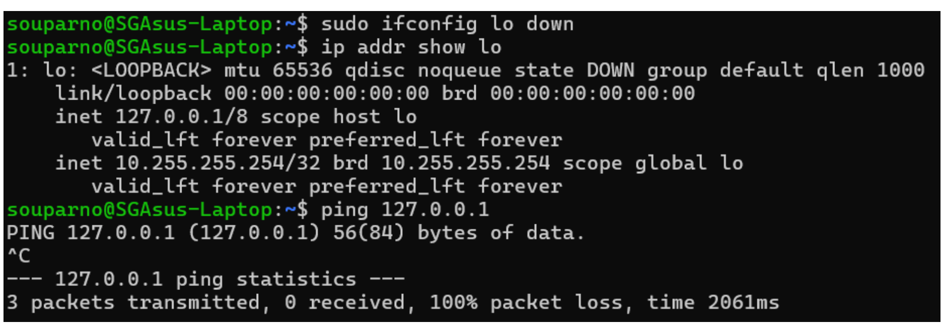
(h) The latency difference between `google.in` and `stanford.edu` can be attributed to several factors. The average latency for `google.in` is 33.406 ms, benefiting from Google's highly optimized global network, which reduces latency through advanced infrastructure and closer geographical proximity. In contrast, `stanford.edu` has a higher average latency of 286.919 ms, due to a more complex routing path involving more network hops and potentially less optimized infrastructure. Additionally, network load and routing complexity further contribute to the increased latency for `stanford.edu`.

Question 6

Make your ping command fail for `127.0.0.1` (with 100% packet loss). Explain how you do it. Put a screenshot that it failed.

Solution

To make `ping` fail for `127.0.0.1` with 100% packet loss, use the command `sudo ifconfig lo down` to disable the loopback interface, effectively blocking all network traffic to `127.0.0.1`. verified this by running `ip addr show lo` to confirm the interface was down and executed `ping 127.0.0.1`, which showed 100% packet loss. After testing, restore the interface using `sudo ifconfig lo up` and verified it was operational again with `ip addr show lo`. This sequence of commands ensures that all local communication via `127.0.0.1` is interrupted while the loopback interface is disabled.



```
souparno@SGAsus-Laptop:~$ sudo ifconfig lo down
souparno@SGAsus-Laptop:~$ ip addr show lo
1: lo: <LOOPBACK> mtu 65536 qdisc noqueue state DOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet 10.255.255.254/32 brd 10.255.255.254 scope global lo
       valid_lft forever preferred_lft forever
souparno@SGAsus-Laptop:~$ ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
^C
--- 127.0.0.1 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2061ms
```

Figure 15: Output showing 100% packet loss and interface status