Assignment 1: Getting To Know Network Traffic

COL 334/672, Diwali'25

August 10, 2025
Deadline: August 22, 2025

Goal: The goal of this assignment is to familiarize you with network data collection, traffic analysis, and basic network measurement tools. Appropriate hints have been provided throughout the assignment. If you still have questions, you are encouraged to start a discussion on Piazza.

1 Measurement Tools

The first part of the assignment will involve using `ping` and `traceroute`, two most basic and useful tools in Internet measurement. You can read about these tools from:

- Ping
- Traceroute

1.1 Ping

Ping the following two websites: `google.com` and `craigslist.com` (FYI, this used to be a very popular online classifieds site based in the US). You should ping these websites 10 times and attach screenshots for each case.

A. Explain the protocol(s) being used by the `ping` tool. Where does it sit in the network protocol stack?

B. Compare the average ping latencies for the two websites from the same network. Why might they differ? Also, think and explain why the latency across multiple pings for the same website might differ?

C. Now try to force ping to use IPv6 for both websites. Explain how you did it and whether it worked. Attach relevant screenshots. If IPv6 fails, explain why?

D. Ping allows you to specify the size of packets to send. What is the maximum size of ping packets that you are able to send? Explain.

1.2 Traceroute

Once you've seen how long it takes to get there, let's find out how you get there. Log the server IP addresses for the two websites in above case. Use traceroute to find the path taken by the packets in both cases and attach the screenshot.

A. Mention the number of IP hops in each case. List the autonomous systems you pass through (Hint: Online tools can map IPs to AS numbers.).

B. Did you see "*" in the traceroute output? Explain what it means.

C. Did you observe multiple IP addresses for the same hop count? If yes, explain the reason.

D. Try to geolocate the IP addresses. You can use two different methods: First, try doing the reverse DNS lookup on the IP address and see if you can infer the location from the DNS address. If the reverse DNS lookup fails, use the Maxmind database for IP geolocation. Note the IP geolocation can sometimes be wrong, especially if you are using the Maxmind database. In fact, accurate geolocation of IP addresses is still an active area of research. Now compare the geographical path with the observed RTTs. Do these intuitively make sense? Explain why.

E. Based on the AS hops you observe, can you relate it to the Internet structure discussed in class. More specifically, do you observe a 3-tiered Internet architecture in both cases? What is happening in the case where you don't observe such an architecture?

2 Network Traffic Collection and Analysis

Traffic Capture: Use wireshark to grab all packets on your wired or wireless interface, while visiting the following HTTP website `http://www.httpvshttps.com`. Make sure you explicitly visit the HTTP site (not HTTPS). Manually type the full URL starting with `http://` in your browser's address bar, and double-check in the browser that the connection remains `http://` and does not automatically switch to `https://`. Also, clear your browser and DNS cache before visiting the website. Save the capture file as `http.pcap`.

Answer the following questions:

A. Apply a DNS filter on the packet trace (read more about DNS here. We will formally cover it later in the course.), and check for DNS queries and responses for www.httpvshttps.com. How long did it take for the DNS request–response to complete?

B. Apply an HTTP filter on the packet trace and report the approximate number of HTTP requests generated. What does this tell you about how webpages are structured and how browsers render complex pages with multiple images and files?

C. Use wireshark display filter to filter traffic corresponding to the website. Count the number of TCP connections opened between your browser and the web server. A new TCP connection starts with a 3-way handshake (SYN → SYN-ACK → ACK).

- Is the number of TCP connections the same as the number of HTTP requests?
- Do some content objects get fetched over the same TCP connection?

D. Now try doing a trace for `https://www.httpvshttps.com` and save it as `https.pcap`. Filter for "http". What do you find, is there any HTTP traffic? Browse through the entire trace without any filters, are you able to see the contents of any HTML and Javascript files being transferred? What just happened? Also, do you find any DNS traffic? Explain. Finally, log the number of TCP connections that were opened this time? Are these similar to the `http` case?

Performance analysis: Beyond Wireshark, it is also common to programmatically analyze packet captures. In this part, you will write a Python script to analyze network performance in two scenarios. You may use libraries such as scapy or dpkt to parse the PCAP file.

Your script should first isolate the traffic corresponding to the given webpage. The IP addresses of the client and server endpoints will be provided as command-line arguments. After isolating the relevant traffic, the script should be able to perform the following functions:

- Plot the download throughput (calculated over a 1-second window) in a file named `down_throughput.png` when run as:

```
python traffic_analysis.py --client <client_ip> --server <server_ip> --throughput --down
```

- Plot the upload throughput (calculated over a 1-second window) in a file named `up_throughput.png` when run as:

```
python traffic_analysis.py --client <client_ip> --server <server_ip> --throughput --up
```

- Plot the Round-Trip Time (RTT) for uplink traffic in a file named `rtt.png` when run as:

  ```
  python traffic_analysis.py --client <client_ip> --server <server_ip> --rtt
  ```

  Note: It makes sense to only calculate uplink RTTs since the traffic is collected at the client side.

E. Finally, using this script, obtain the throughput and RTT plots for both the `http` and `https` packet captures. Compare the time taken to download the webpage (this time is displayed on the webpage itself). Explain any differences you observe using the plots.

Submission Instructions

Your submission should contain a single PDF (other formats will not be graded) called `report.pdf`. The report should be written using Latex. Watch these simple videos to learn Latex.

1. For part 1, you should attach the screenshots in the PDF itself. All questions in the first part should be answered in the report.
2. For part 2, you should submit the collected PCAPs. Name it `http.pcap` and `https.pcap`. The answers to the questions should be in the PDF. If you used Wireshark (highly encouraged) for the analysis, mention the filters used. The python script should be named `traffic_analysis.py`. Include the RTT and throughput plots that are generated for both cases in the report itself.

Submit a single zipped file containing all the above files.