Project 2: Wireshark Packet Capture and Analysis

Created by Luis Umana

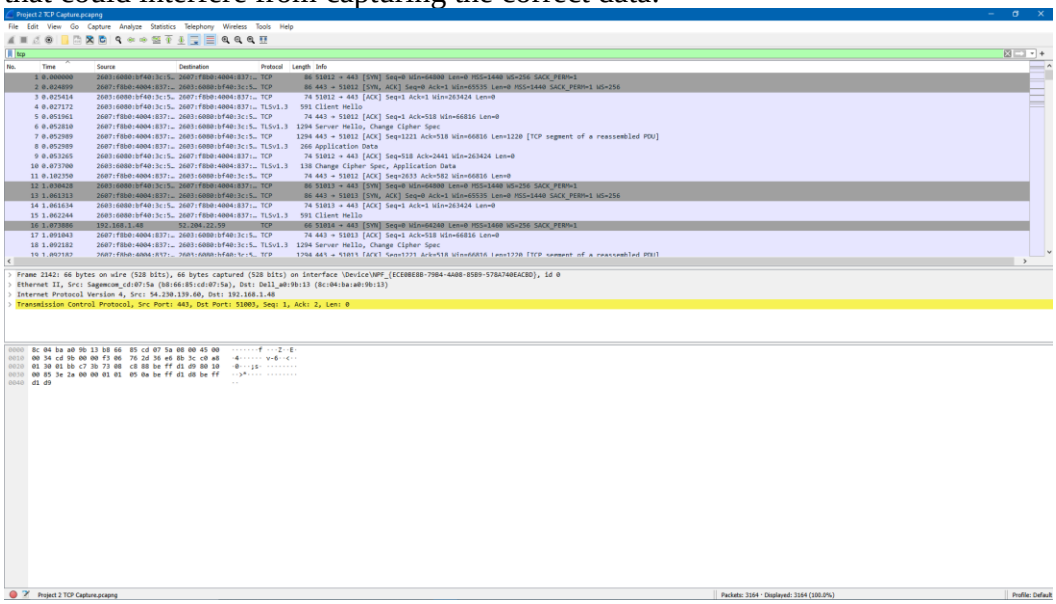
1.1. Data Capture

Location/Device

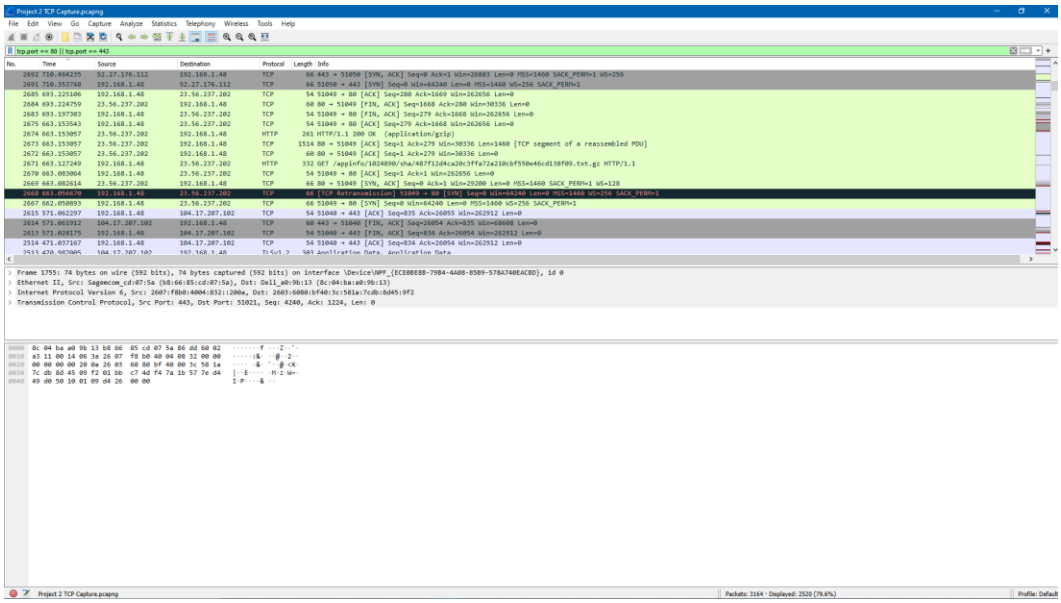
The project was performed at my home with a Dell G7 Laptop. I used an ethernet cable instead of Wi-Fi, in order to isolate the TCP traffic of YouTube on WireShark.

Wireshark filters for capture and display

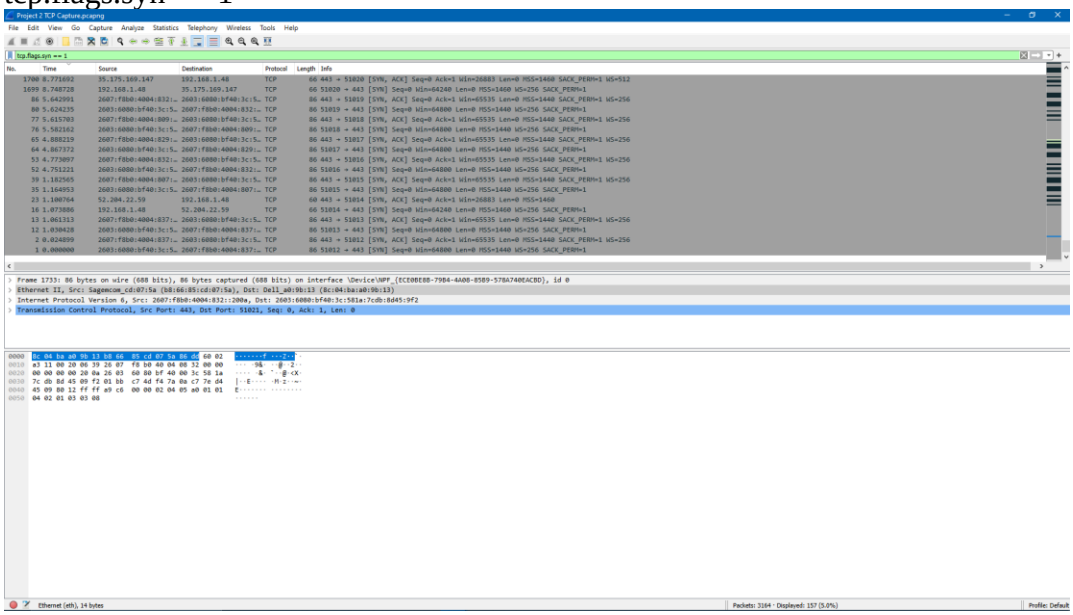
Capture

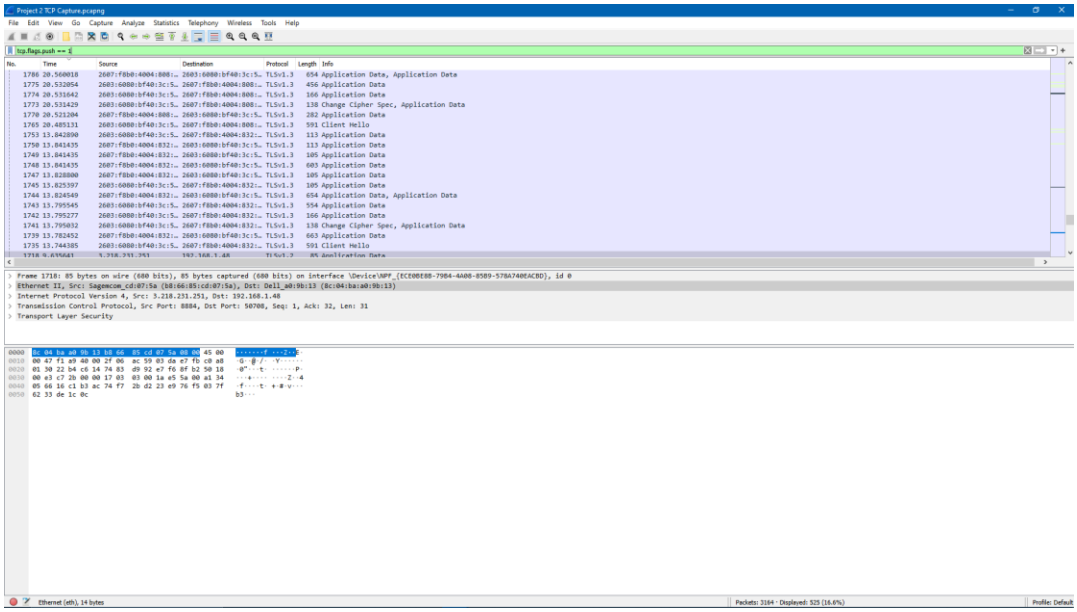
1. TCP for Capture filter
    - I decided to use a general TCP capture because, the project required us to capture all TCP and HTTP traffic; HTTP is part of the TCP family, so they can both be captured with the same filter.
    - Since I was using an ethernet connection, rather than using the Wi-Fi, I could easily isolate the traffic coming from only my laptop. By closing all applications and websites while only having YouTube open, I can reduce the amount of traffic that could interfere from capturing the correct data.
    - 
      (Screenshot of all TCP traffic)

2. HTTP Traffic
    - The traffic of YouTube can be displayed by using two filters:
    - tcp.port == 80 || tcp.port == 443
    - As stated in the description of the project, port 80 can separate the HTTP packets from the TCP packets and port 443 can capture some HTTP packets that are secure.

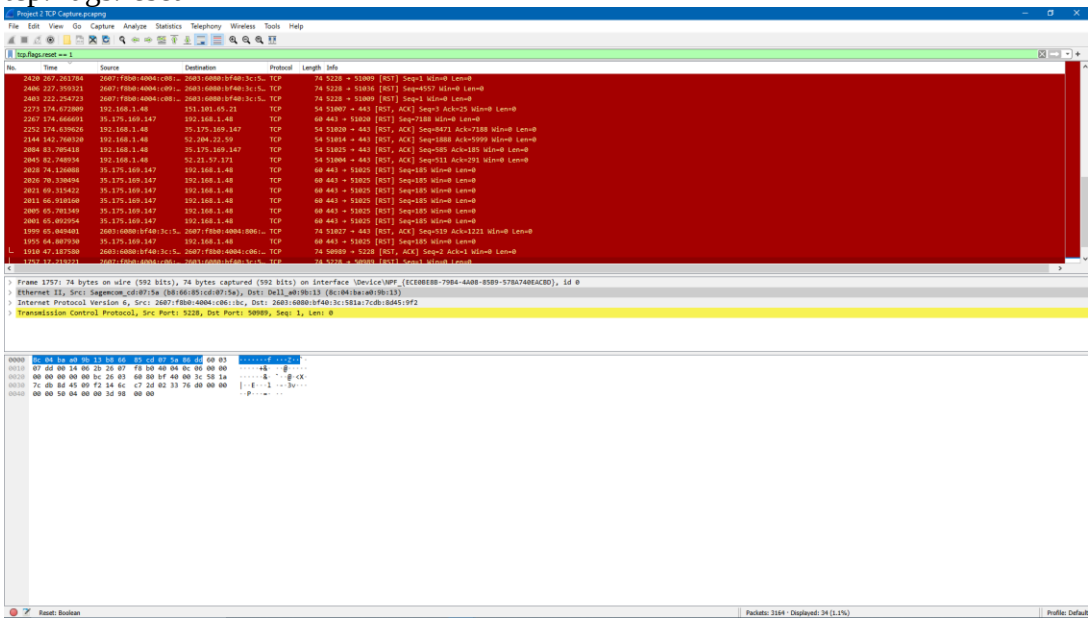
- 
  (Screenshot of all HTTP traffic to/from youtube)
- Note: If tcp.port 80 is only used then it will only show 16 packets while tcp.port 443 shows 2504 packets.

Display

1. SYN, PSH, and RST filters for all TCP packets
   - tcp.flags.syn == 1
   - 
   - tcp.flags.push == 1

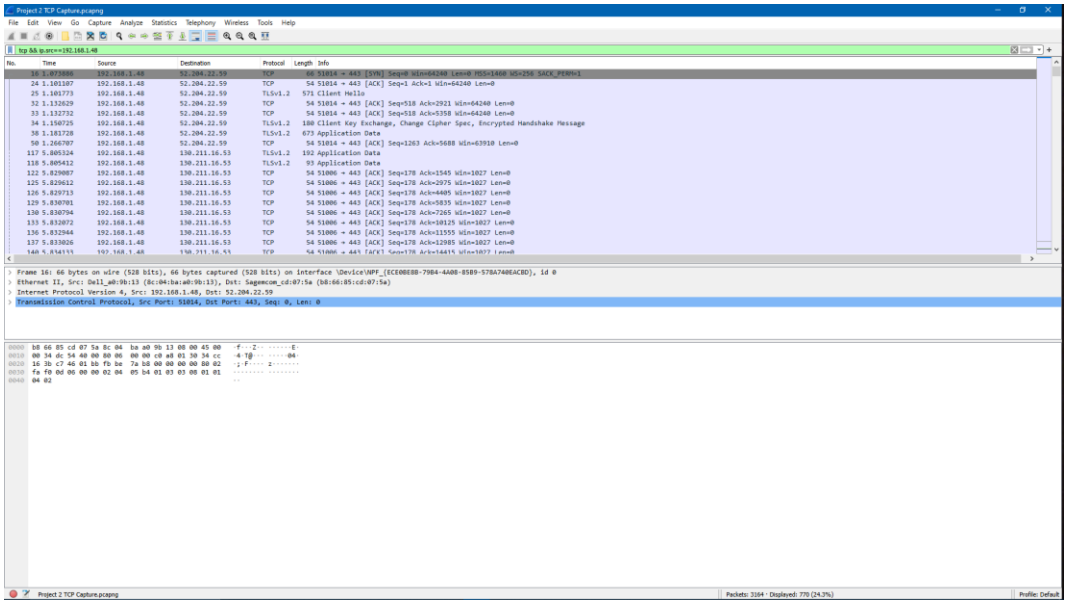
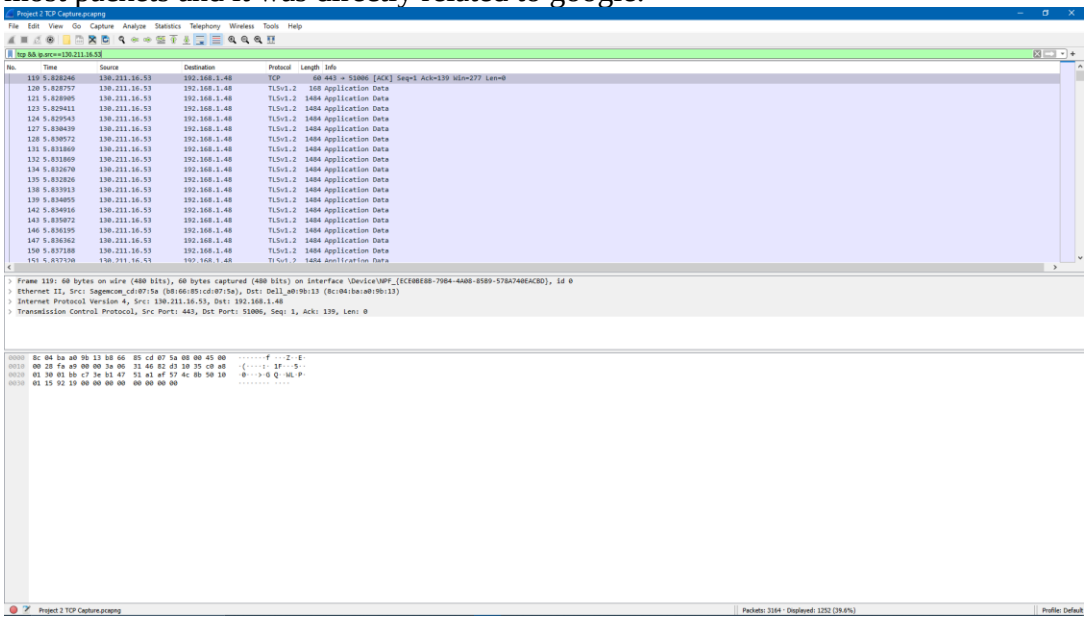


- tcp.flags.reset == 1



- In this case, the filters are pretty much self-explanatory since the filters displays only the flags, and each filter will give either SYN, PSH, or RST.

2. Packets sent by your computer vs Received from YouTube.
    - tcp && ip.src==192.168.1.48
    - This is the filter that contains only TCP traffic sent from my computer

- 
- tcp && ip.src==130.211.15.53
- This is the filter used that contains only TCP Traffic received from YouTube.
- If you examine the TCP traffic, you will notice how there are multiple different IP going on as the video plays. When searching for the information on the ips, most of these videos either relate to the video, software such as cloudflare, and other ips related to ads, such as amazon. I decided to choose this ip because, it had the most packets and it was directly related to google.
- 
- The TCP sent from my computer consisted of 24.3% of the total TCP, while the traffic received from YouTube consisted of 39.6% of the total TCP.

YouTube Video used for this project:

- **Video Title:** “Teaching a Robot Dog to Pee Beer”

- **YouTuber:** “Michael Reeves”
- **Link:** https://www.youtube.com/watch?v=tqsy9Wtr1qE

1.2. Data Analytics

TCP packets received from/Sent to YouTube

Components	# of Packets	Percent of total
Received	**1252**	**39.6%**
Sent	**770**	**24.3%**
HTTP	**2504**	**79.1%**
HTTPS	**16**	**0.5%**

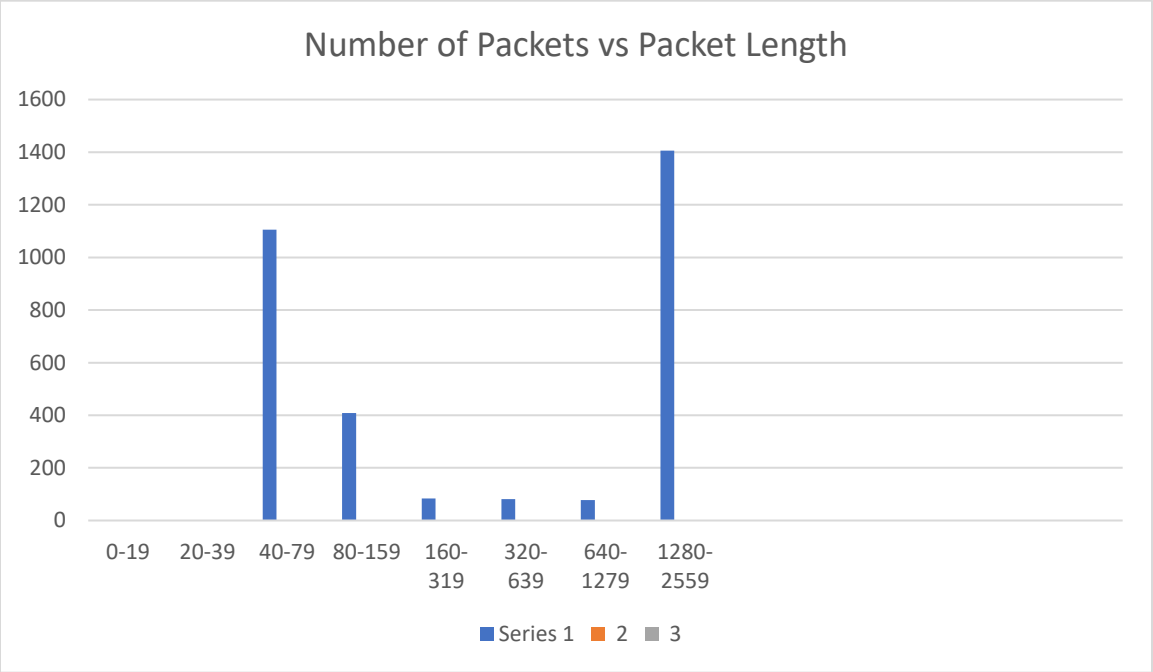
TCP Flag packets

Components	# of Packets	Percent of total
Syn	**157**	**21.9%**
Push	**525**	**73.1%**
Reset	**35**	**5%**
Total	**717**	**100%**

From what I have noticed looking at both the SYN and PSH flags, a majority of both flags were sent from my computer while very little is actually received from YouTube.

1.3. Additional Requirements

YouTube Packet Lengths



References

https://ipinfo.io/AS15169/130.211.0.0/16-130.211.14.0/23
(This was used to show ip address I choose is from google)