

# CS331 – Computer Networks

## Assignment 1

Jaidev Sanjay Khalane – 22110103

Sriram Srinivasan – 22110258

Group 13

### **4.pcap**

## Contents

Part 1 .....	2
Approach.....	2
Question 1 .....	2
Question 2 .....	3
Question 3 .....	3
Question 4 .....	3
Supplementary Screenshots for Results .....	5
Part 2 .....	6
Approach.....	6
Question 1 .....	6
Question 2 .....	6
Question 3 .....	7
Question 4 .....	7
Supplementary Screenshot for Results .....	7
Part 3 .....	7
Question 1 .....	8
Question 2 .....	10
Part (a).....	11
Part (b) .....	13
Part (c).....	17
References.....	19
Supplementary Documents and Data.....	20

# Part 1

## Part 1: Metrics and Plots (40 pts)

From the chosen X.pcap file, extract and generate the following metrics for the data as captured by your program when you perform the pcap replay using tools like tcpreplay:

1. Find the total amount of data transferred (in bytes), the total number of packets transferred, and the minimum, maximum, and average packet sizes. Also, show the distribution of packet sizes (e.g., by plotting a histogram of packet sizes).
2. Find unique source-destination pairs (source IP:port and destination IP:port) in the captured data.
3. Display a dictionary where the key is the IP address and the value is the total flows for that IP address as the source. Similarly display a dictionary where the key is the IP address and the value is the total flows for that IP address as the destination. Find out which source-destination (source IP:port and destination IP:port) have transferred the most data.
4. List the top speed in terms of 'pps' and 'mbps' that your program is able to capture the content without any loss of data when i) running both tcpreplay and your program on the same machine (VM), and ii) when running on different machines: Two student group should run the program on two different machines eg. tcpreplay on physical-machine of student1 and sniffer program physical-machine of student2. Single students should run between two VMs.

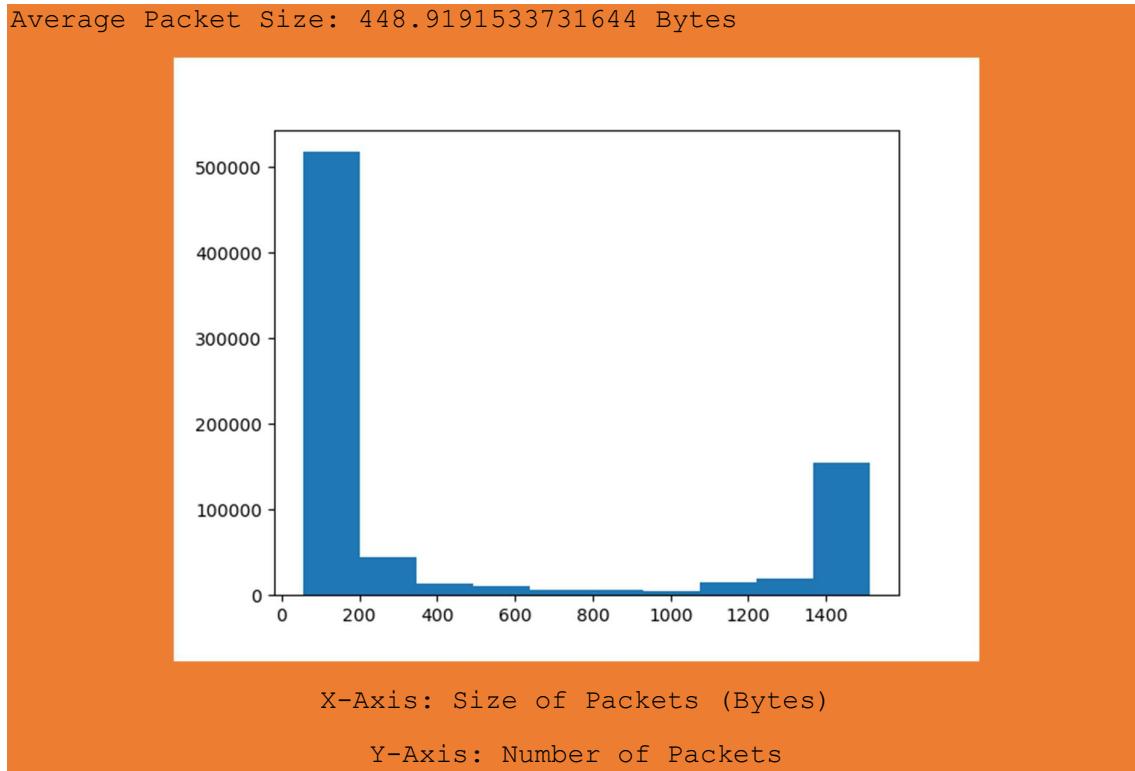
## Approach

To approach this problem in the right way, I was required to know the details of the protocols used. So, I decided to go from the basics. I first created a raw socket in Python using the sockets module. Since the raw socket gives us raw packets, that is, in our case, the ethernet packet, I disintegrated the packet using the reference [1]. So, I sliced the header of the frame using the Struct module in Python following the format stated in [1]. Then, I analysed the different types of protocols in the next layer, which was the network layer. The Ethertype number from this header was used to find the corresponding protocol used in the network layer. The detailed results are given in the supplementary information section. Using this Ethertype number, I referred to [2] in order to find the corresponding protocols. The major protocols used in this layer were IPv4 and IPv6. I then used [3] to understand the structure of the IPv4 header. It was then disintegrated similarly to find the types of protocols used in the Transport layer. The exact count of the findings is given in the supplementary section. The major types of protocols found were TCP and UDP, with minor traces of other protocols. The headers and payload of TCP and UDP were also disintegrated using [4] and [5] references. A similar approach was also used for ICMP [6] and IPv6 [7].

Once this was done, global lists, sets, and dictionaries were initialised and used throughout the program to capture various required parameters for Part 1. The program also has the feature of automatically ending sniffing after 5 seconds of inactivity, that is, 5 seconds of not receiving any packets automatically. I implemented this using the 'signal' library. On the whole, I used libraries 'socket' [8] in Python for capturing the data, 'struct' [9] for disintegrating the headers, 'signal' [10] for automatically ending the program after 5 seconds of inactivity and 'matplotlib' for plotting the histogram as well as 'time' [11] for finding the total time.

## Question 1

```
Total Packets: 792179
Total Amount of Data Transferred: 355624326 Bytes
Minimum Packet Size: 54 Bytes
Maximum Packet Size: 1514 Bytes
```



## Question 2

Since the result is too large to be displayed here, I have given the results for this Question in the file Problem1\_results2.txt.

## Question 3

Source Flows: {'166.131.131.6': 1, '80.239.144.76': 193, ...}

Destination Flows: {'80.239.144.76': 222, '81.131.131.6': 208, ...}

Since the result is too large to be displayed here, I have given the results for this Question in the file Problem1\_results3.txt.

The flow considered here is the number of packets.

Maximum Data Transfer:

Source IP: 172.16.133.95      Source Port: 49358

Destination IP: 157.56.240.102      Destination Port: 443

## Question 4

### Results for the same VM (Ubuntu 24.04):

Time taken: 197.3298

pps: 4014.491574724454

mBps: 1.8021821589490032

mbps: 14.4174572715920256

## Results for the different VMs (Ubuntu 24.04 and Ubuntu 20.04):

Time taken: 204.6458

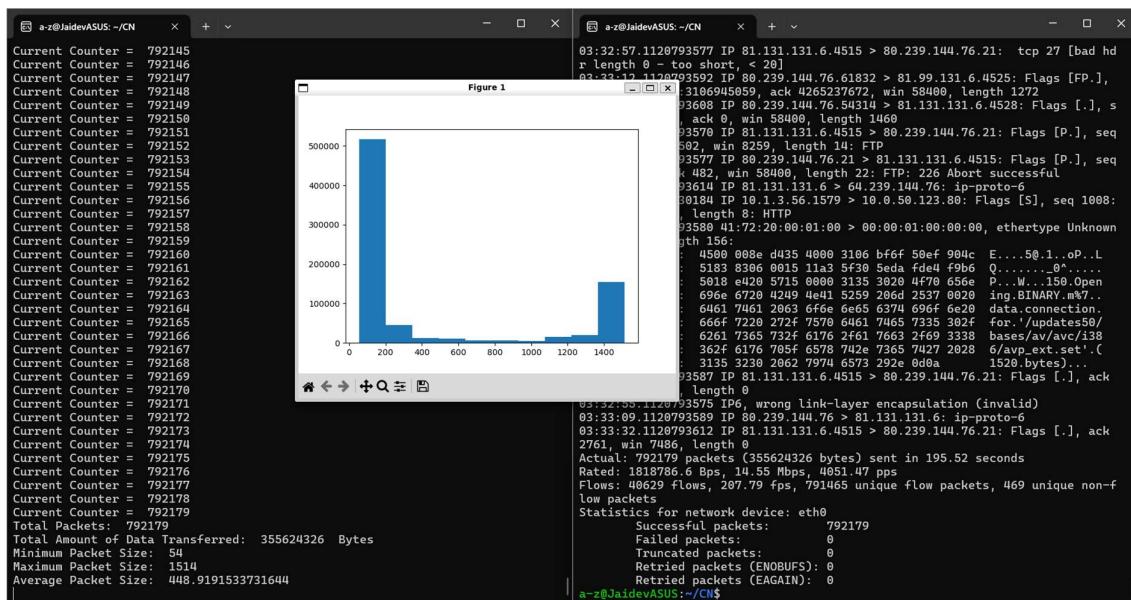
pps: 3870.9761

mBps: 1.7377

mbps: 13.9016

## Supplementary Screenshots for Results

The left terminal window shows the command `a-z@JaidevASUS:~/CN$ sudo python3 Problem1.py` being run. The right terminal window shows the command `a-z@JaidevASUS:~/CN$ sudo tcpreplay -i eth0 -v -t 4.pcap` being run.



```

a-z@JaidevASUS: ~$ /usr/bin/termux-pcap-dump -r /sdcard/termux-pcap.pcap
a-z@JaidevASUS: ~$ 

[Output of termux-pcap-dump]
[Output of Wireshark]

```

## Part 2

### 4.pcap

Hiding a Message in TCP Packet Payload

Q1. Can you extract the hidden message from the packet payload?

Hint : Filter packet with source port 1579 search keyword CS331.

Q2. How many packets contain the hidden message?

Q3. What protocol is used to transmit the packet containing the hidden message?

Q4. What is the checksum of the TCP segment containing the hidden message?

### Approach

While analysing the sniffed packets in the previous problem, instead of storing the parameters like the Source flows, etc., I instead checked for the condition that the source port should be 1579 as given in the hint, and the TCP payload should contain the keyword ‘CS331’. I solved this question assuming the original encoding to be of utf-8 type.

### Question 1

Yes, I can extract the hidden message using the hint from the packet payload.

The hidden message is:

Welcome to Computer Networks CS331

### Question 2

A total of 11 packets contain the hidden message.

## Question 3

Since this message was first obtained from the raw socket with an ethernet header, the first protocol (at the datalink layer) was Ethernet II. Then, at the Network layer, we had the IPv4 Protocol, and at the Transport layer, the protocol was TCP.

In summary,

Layer	Protocol
Transport	TCP
Network	IPv4
Datalink	Ethernet II

## Question 4

The checksum values are as given below:

547, 755, 908, 703, 958, 599, 651, 858, 443, 807, 495

## Supplementary Screenshot for Results

The screenshot shows two terminal windows side-by-side. Both windows are titled 'a-z@JaidevASUS: ~/CN' and are running on the command line. The left window displays a series of network packets captured by Wireshark, showing details like source and destination MAC addresses, IP addresses, port numbers, and payload data. The right window shows the results of a 'tcpdump -C 1' command, which lists the total number of hidden messages found (792179), the total number of hidden protocols (11), and the total number of hidden checksums (11). It also provides statistics for the network device eth0, including successful, failed, truncated, retried, and retransmitted packets.

```
a-z@JaidevASUS: ~/CN$ 
Current Counter = 792155
Current Counter = 792156
Current Counter = 792157
Current Counter = 792158
Current Counter = 792159
Current Counter = 792160
Current Counter = 792161
Current Counter = 792162
Current Counter = 792163
Current Counter = 792164

Hidden Message Found TCP:
|--->Data: b'Welcome to Computer Networks CS331'
|--->Checksum: 495
|--->Protocol: 6
Current Counter = 792165
Current Counter = 792166
Current Counter = 792167
Current Counter = 792168
Current Counter = 792169
Current Counter = 792170
Current Counter = 792171
Current Counter = 792172
Current Counter = 792173
Current Counter = 792174
Current Counter = 792175
Current Counter = 792176
Current Counter = 792177
Current Counter = 792178
Current Counter = 792179
Finished capturing packets...
Hidden Messages: [b'Welcome to Computer Networks CS331', b'Welcome to Computer Networks CS331']
Hidden Protocols: [6, 6, 6, 6, 6, 6, 6, 6, 6, 6]
Checksums: [547, 755, 908, 703, 958, 599, 651, 858, 443, 807, 495]
Total Number of Hidden Messages: 11
a-z@JaidevASUS:~/CN$ |
```

```
a-z@JaidevASUS: ~/CN$ 
r length 0 - too short, < 20]
03:32:12.1120793592 IP 88.239.144.76.61832 > 81.99.131.6.4525: Flags [FP.], seq 3106943787:3106945059, ack 4265237672, win 58400, length 1272
03:32:28.1120793608 IP 88.239.144.76.54314 > 81.131.131.6.4528: Flags [.], s
eq 35041:36501, ack 0, win 58400, length 1468
03:32:50.1120793570 IP 81.131.131.6.4515 > 80.239.144.76.21: Flags [P.], seq
258:272, ack 502, win 8259, length 14: FTP
03:32:57.1120793577 IP 88.239.144.76.21 > 81.131.131.6.4515: Flags [P.], seq
1140:1162, ack 482, win 58400, length 22: FTP: 226 Abort successful
03:33:34.1120793614 IP 81.131.131.6 > 64.239.144.76: ip-proto-6
11:03:04.1737630184 IP 18.1.3.56.1579 > 10.0.50.123.80: Flags [S], seq 1008:
1016, win 8192, length 8: HTTP
03:33:00.1120793580 41:72:20:00:01:00 > 00:00:01:00:00:00, ethertype Unknown
(0x8800), length 156:
0x0000: 4500 008a d435 4000 3106 bf6f 50ef 904c E...56.1..oP..
0x0010: 5183 8306 0015 11a3 5f30 5eda fde4 f9b6 Q.....^0^.....
0x0020: 5018 e420 5715 0000 3135 3028 4f70 656e P...W...158.Open
0x0030: 696e 6720 4249 4e41 5259 206d 2537 0020 ing.BINARY.m%7..
0x0040: 6461 7461 2063 6f6e 6e65 6374 696f 6e20 data.connection.
0x0050: 666f 7220 272f 7570 6461 7465 7335 302f for.'/updatecs%8/
0x0060: 6261 7361 732f 6170 2f61 7663 2f69 3338 bases/av/avc/i38
0x0070: 362f 6170 705f 6570 742e 7365 7427 2028 /avp_ext.set.(.
0x0080: 3135 3230 2062 7974 6573 292 0d0a. 1520.bytes)...
03:33:07.1120793587 IP 81.131.131.6.4515 > 88.239.144.76.21: Flags [.], ack
2698, win 8192, length 0
03:32:55.1120793575 IP6 wrong link-layer encapsulation (invalid)
03:33:09.1120793589 IP 88.239.144.76 > 81.131.131.6: ip-proto-6
03:33:32.1120793612 IP 81.131.131.6.4515 > 88.239.144.76.21: Flags [.], ack
776, win 708, length 0
Actual: 792179 packets (35562432 bytes) sent in 204.36 seconds
Rated: 1700171.6 Eps, 13.92 Mbps, 3876.35 pps
Flows: 40629 flows, 198.80 fps, 791465 unique flow packets, 469 unique non-flow packets
Statistics for network device eth0:
Successful packets: 792179
Failed packets: 0
Truncated packets: 0
Retried packets (ENOBUFS): 0
Retried packets (EAGAIN): 0
a-z@JaidevASUS:~/CN$ |
```

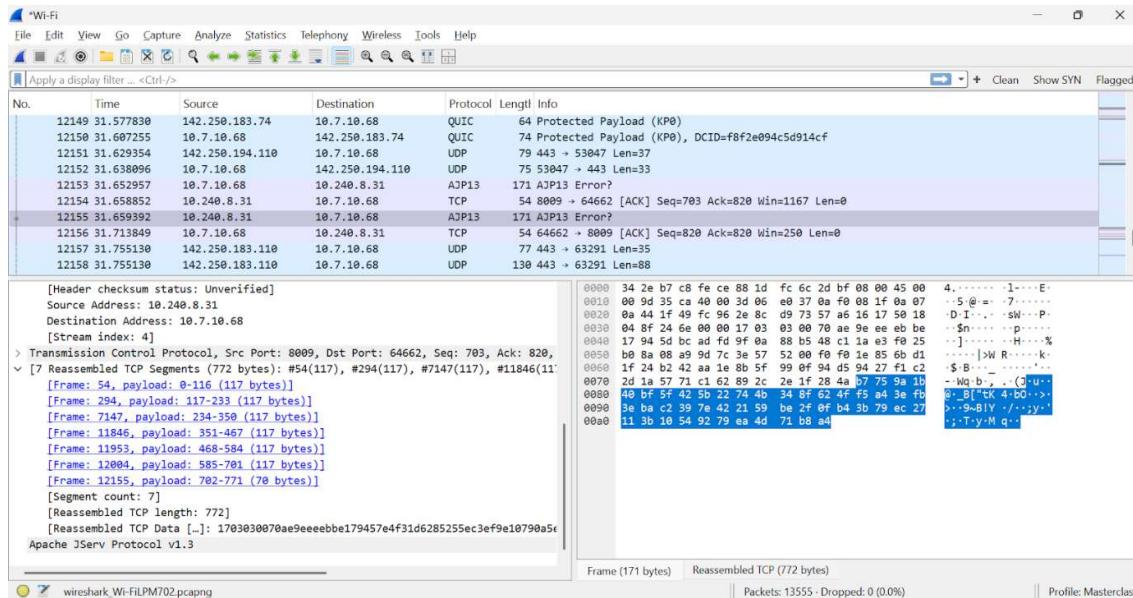
## Part 3

### Part 3: Capture the packets (20 points)

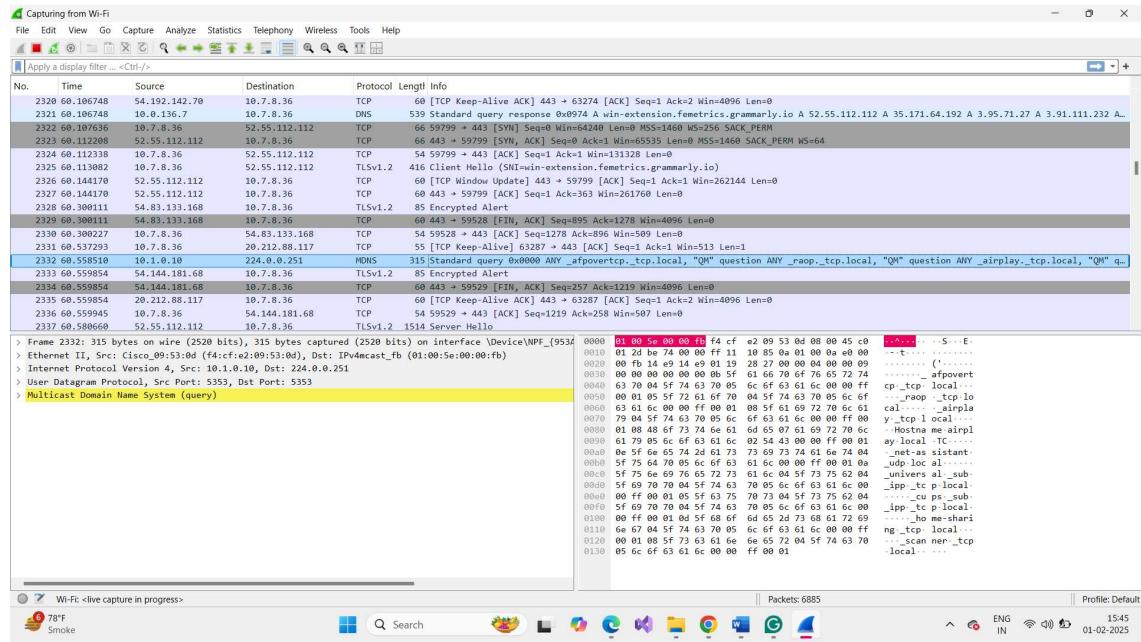
1. Run the Wireshark tool and capture the trace of the network packets on your host device. We expect you would be connected to the Internet and perform regular network activities.
  - a. List at-least 5 different application layer protocols that we have not discussed so far in the classroom and describe in 1-2 sentences the operation/usage of protocol and its layer of operation and indicate the associated RFC number if any.
  
2. Analyze the following details by visiting the following websites in your favourite browser.
  - i) canarabank.in
  - ii) github.com
  - iii) netflix.com
  - a. Identify 'request line' with the version of the application layer protocol and the IP address. Also, identify whether the connection(s) is/are persistent or not.
  - b. For any one of the websites, list any three header field names and corresponding values in the request and response message. Any three HTTP error codes obtained while loading one of the pages with a brief description.
  - c. Capture the Performance metrics that your browser records when a page is loaded and also report the list the cookies used and the associated flags in the request and response headers. Please report the browser name and screenshot of the performance metrics reported for any one of the page loads.

## Question 1

The first protocol we saw was AJP.

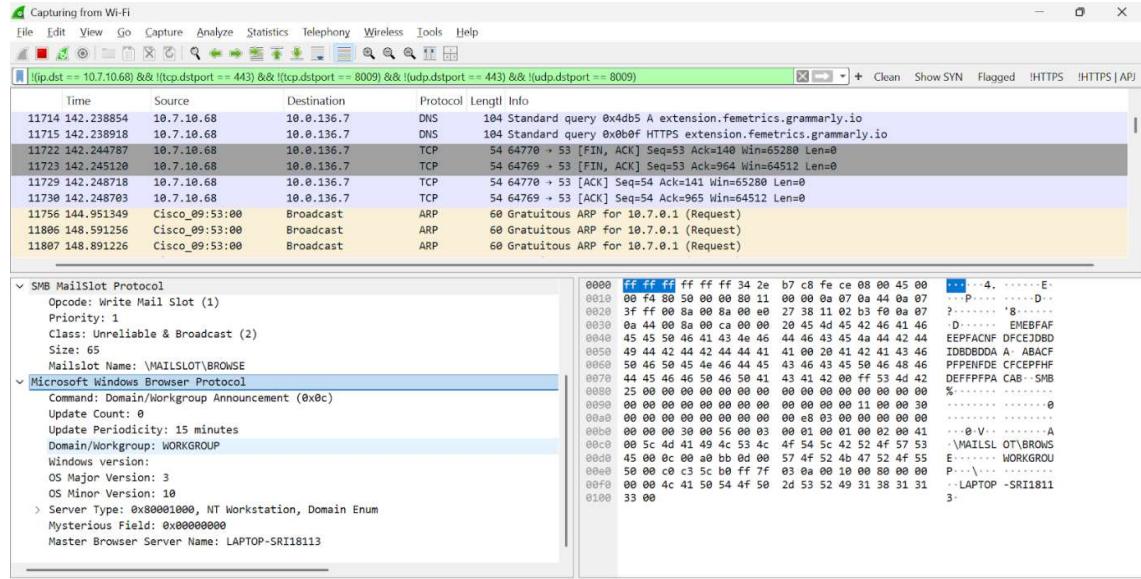


*"The Apache JServ Protocol (AJP) is a binary protocol that can proxy inbound requests from a web server to an application server behind the web server. It is used at Application layer" [15]*



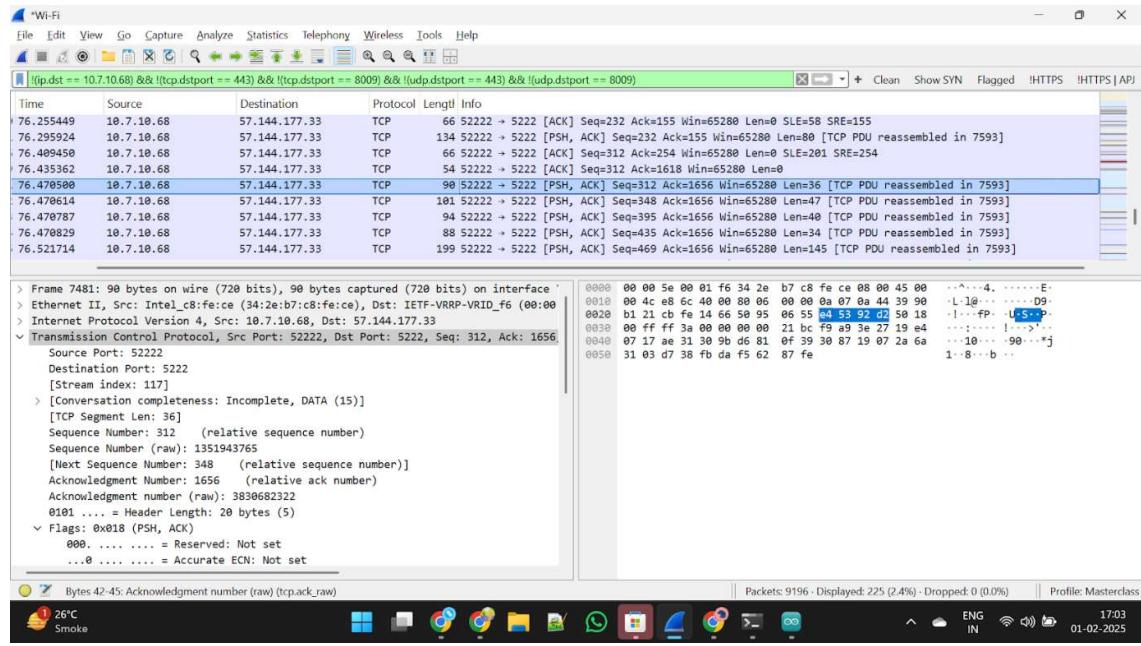
The next protocol was MDNS.

**“Multicast DNS (mDNS) is a computer networking protocol that resolves hostnames to IP addresses within small networks that do not include a local name server. It works at the Application layer, similar to DNS. RFC 6763.” [16]**

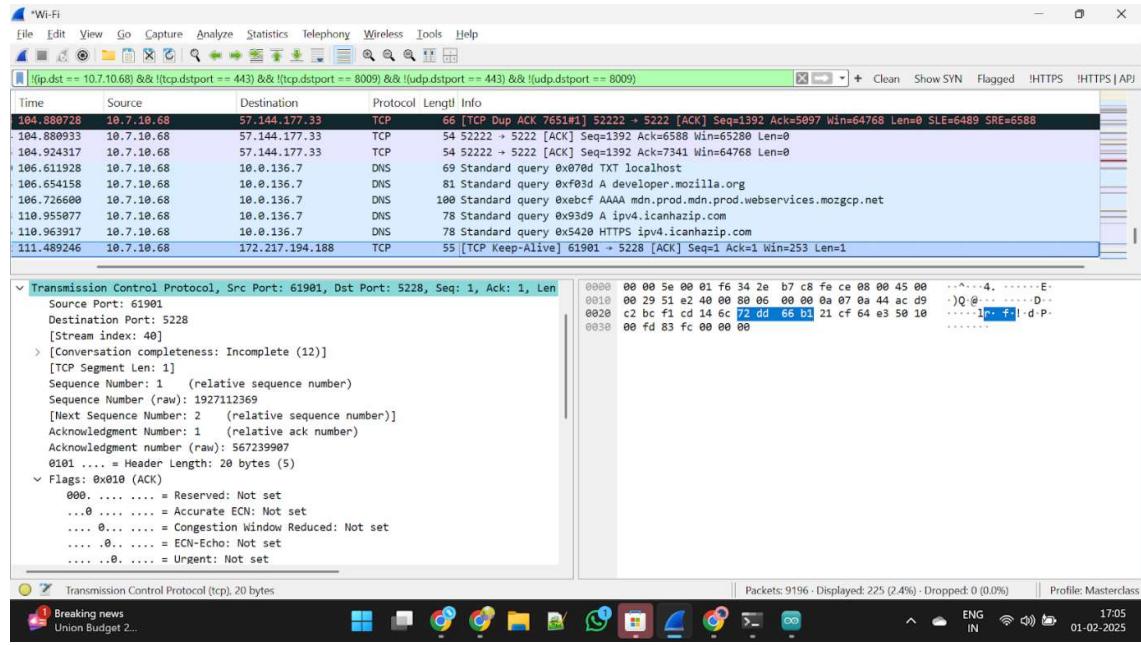


**“The Server Message Block (SMB) Protocol is a network file sharing protocol, and as implemented in Microsoft Windows, it is known as Microsoft SMB Protocol. It works at Application Layer.“ [17]**

**“Microsoft Windows Browser Protocol is also an application layer protocol, and it is used in browser applications on Microsoft Windows Devices (using Microsoft Browser)“ [18]**



**"XMPP, Extensible Messaging and Presence Protocol, is an open communication protocol designed for instant messaging (IM), presence information, and contact list maintenance. It operates on Port 5222. It was defined in RFC 6120."** [19]

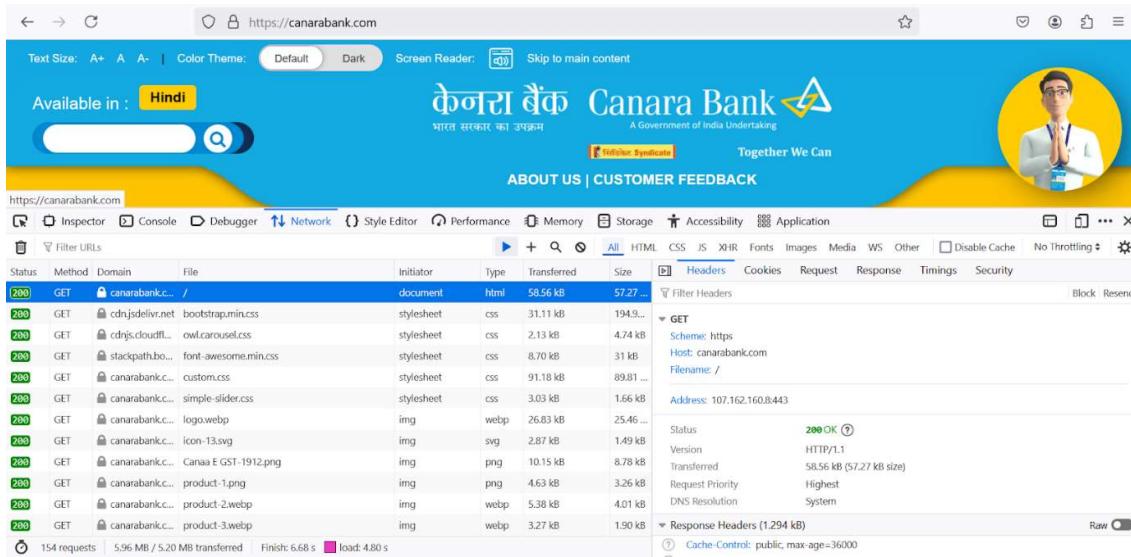


**"HP Virtual Rooms Protocol uses Port 5228. It works on the application layer and is used for facilitating online team meetings."** [20]

## Question 2

canarabank.in did not exist. So, we used canarabank.com.

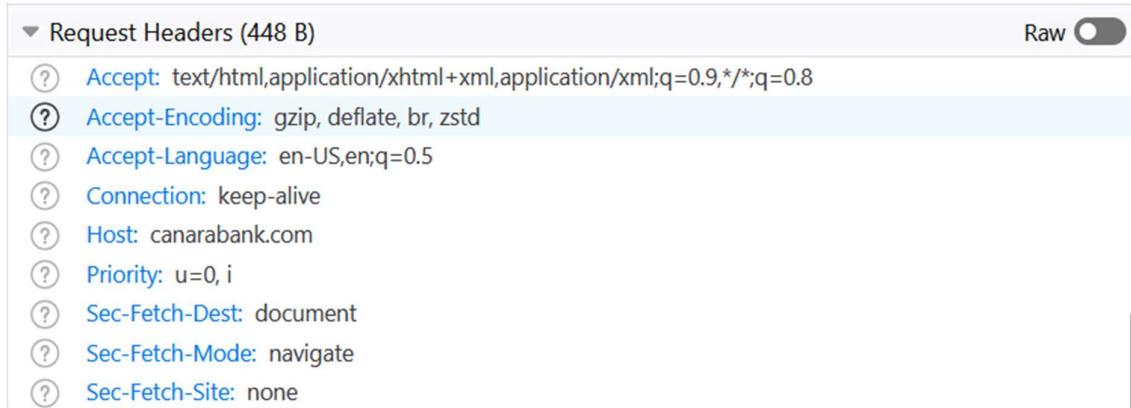
## Part (a)



The screenshot shows the Canara Bank homepage with the URL <https://canarabank.com>. The Network tab of the developer tools is open, displaying a list of requests made by the browser. The table has columns for Status, Method, Domain, File, Initiator, Type, Transferred, Size, Headers, Cookies, Request, Response, Timings, and Security. Most requests are 200 OK status, GET method, and document type. One request is a stylesheet from cdnjsdelivr.net. The response details for one of the CSS files show the scheme as https, host as canarabank.com, and filename as /.

Status	Method	Domain	File	Initiator	Type	Transferred	Size	Headers	Cookies	Request	Response	Timings	Security
200	GET	canarabank.c...	/	document	HTML	58.56 kB	57.27 ...	Filter Headers					
200	GET	cdnjsdelivr.net	bootstrap.min.css		stylesheet	31.11 kB	194.9...	GET					
200	GET	cdnjs.cloudflare...	owl.carousel.css		stylesheet	2.13 kB	4.74 kB						
200	GET	stackpath.bootstrapcdn...	font-awesome.min.css		stylesheet	8.70 kB	31 kB						
200	GET	canarabank.c...	custom.css		stylesheet	91.18 kB	89.81 ...						
200	GET	canarabank.c...	simple-slider.css		stylesheet	3.03 kB	1.66 kB						
200	GET	canarabank.c...	logo.webp		img	26.83 kB	25.46 ...						
200	GET	canarabank.c...	icon-13.svg		img	2.87 kB	1.49 kB						
200	GET	canarabank.c...	Canaa E GST-1912.png		img	10.15 kB	8.78 kB						
200	GET	canarabank.c...	product-1.png		img	4.63 kB	3.26 kB						
200	GET	canarabank.c...	product-2.webp		img	5.38 kB	4.01 kB						
200	GET	canarabank.c...	product-3.webp		img	3.27 kB	1.90 kB	Response Headers (1,294 kB)					

From the above snippet, it's clear that the application layer protocol is HTTP (Version 1.1) and the IP address of the server is 107.162.160.8 running on port 443 (corresponding to Secure HTTP).



The screenshot shows the Request Headers section of the developer tools. It lists various headers sent by the client to the server. The 'Raw' toggle switch is off. The headers include Accept, Accept-Encoding, Accept-Language, Connection, Host, Priority, Sec-Fetch-Dest, Sec-Fetch-Mode, and Sec-Fetch-Site. The Connection header is set to 'keep-alive'.

Header	Value
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding	gzip, deflate, br, zstd
Accept-Language	en-US,en;q=0.5
Connection	keep-alive
Host	canarabank.com
Priority	u=0, i
Sec-Fetch-Dest	document
Sec-Fetch-Mode	navigate
Sec-Fetch-Site	none

The connection is persistent as the value of the *connection parameter* is 'keep-alive'.

The screenshot shows the Network tab of a browser developer tools interface. It displays a list of network requests made by the Netflix application. Key details from the table include:

- Status:** Most requests are 200 OK.
- Method:** GET or POST.
- Domain:** logs.netflix.com or www.netflix.com.
- File:** Various file paths such as /in/, /n/1, /n/1/1, /n/1/1/1, etc.
- Initiator:** fetch.
- Type:** document, stylesheet, script, etc.
- Size:** File sizes range from 0 B to 88.80 kB.
- Details:** Headers, Cookies, Request, Response, Timings, and Security tabs are visible at the top of the tool.

From the above snippet, it's clear that the application layer protocol is HTTP (Version 2) and the IP address of the server is 18.200.8.190 running on port 443 (corresponding to Secure HTTP).

The screenshot shows the Headers tab of a browser developer tools interface, specifically for a single request. The Request Headers section contains the following entries:

- Accept:** text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8
- Accept-Encoding:** gzip, deflate, br, zstd
- Accept-Language:** en-US,en;q=0.5
- Connection:** keep-alive
- Cookie:** nfvidid=BQFmAAEBeADa8h\_GAMbLrKViGbdb6otA69zgAeB3gFCGYZrDAKVNALmpDkJAGnItpCGe\_6KnP4PhTkuaTUhZ3F95pKgSG8DlmV5YFmnwfj3krzO66NyTg%3D%3D; NetflixId=v%3D3%26ct%3DBaiHIOvcAxLAAbZ6VECW6bcRr80O0Hxnk5R7f4dokLW6wxlrH2OTluO64WTYYu5ml02UHc7e90D

The connection is persistent as the value of the *connection parameter* is '*keep-alive*'.

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	github.com	/	document	html	51.79 kB	279.6...
200	GET	github.githubbas...	light-7aa84bb7e11e.css	stylesheet	css	cached	7.90 kB
200	GET	github.githubbas...	dark-f65db3e8d171.css	stylesheet	css	cached	7.99 kB
200	GET	github.githubbas...	primer-primitives-d9abecd14f1e.css	stylesheet	css	cached	2.61 kB
200	GET	github.githubbas...	primer-93adeda0ee8a1.css	stylesheet	css	cached	39.27 ...
200	GET	github.githubbas...	global-d5794a45d443.css	stylesheet	css	cached	38.16 ...
200	GET	github.githubbas...	github-8049ff990d299.css	stylesheet	css	cached	21.35 ...
200	GET	github.githubbas...	site-0fc4fbfc895.css	stylesheet	css	cached	9.30 kB
200	GET	github.githubbas...	landing-pages-41e641406dd3.css	stylesheet	css	cached	68.11 ...
200	GET	github.githubbas...	home-ea3957bd9fb.css	stylesheet	css	cached	7.29 kB
200	GET	github.githubbas...	primer-react-1275b2aabc5faaff7be57.module.css	stylesheet	css	cached	21.80 ...
GET		github.github...	global-banner-disable-f988792b49f.js	script	js	206 kB (raced)	398 B
GET		github.github...	mona-sans-d1bf285e9b9b.woff2	font	woff2	85.04 kB (raced)	84.39 ...
200	GET	github.github...	wp-runtime-0344c588f5c.js	script	js	15.03 kB	53.22 ...
200	GET	github.github...	vendors-node_modules_@dibbird_popover-polyfill	script	js	cached	0 B

92 requests | 3.86 MB / 875.21 kB transferred | Finish: 2.78 s DOMContentLoaded: 1.22 s load: 3.07 s

From the above snippet, it's clear that the application layer protocol is HTTP (Version 2) and the IP address of the server is 20.207.73.82 running on port 443 (corresponding to Secure HTTP).

Request Headers (1.040 kB)	
Accept:	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding:	gzip, deflate, br, zstd
Accept-Language:	en-US,en;q=0.5
Connection:	keep-alive
Cookie:	_gh_sess=5M%2BGr2McyOvys1f4%2B95Tm6hYQs0gwFWjjzeKgkivsPMf7UeoDA30T%2FfshUyz6dwIFVyHUbH9wYArVzVKAi%2FVAV5qXXemurH%2F1%2F88xXMzZymN5DjdrQluVyt%2FZbqncuZj1ygTnuVgZnTVWNUhRw4cNG2AgxuzDJANUiBqB%2BmmFiTJCzYPJ%2B2n9LQH%2FAQ%2Fe1mYaZ%2

The connection is persistent as the value of the *connection parameter is 'keep-alive'*.

## Part (b)

The below analysis was performed on the *github.com* website.

▼ Response Headers (4.717 kB)	
	Raw <input checked="" type="checkbox"/>
cache-control:	max-age=0, private, must-revalidate
content-encoding:	gzip
content-security-policy:	default-src 'none'; base-uri 'self'; child-src <a href="https://github.com/assets-cdn/worker/">github.com/assets-cdn/worker/</a> <a href="https://github.com/webpack/">github.com/webpack/</a> <a href="https://github.com/assets/">github.com/assets/</a> <a href="https://gist.github.com/assets-cdn/worker/">gist.github.com/assets-cdn/worker/</a> ; connect-src 'self' uploads.githubusercontent.com <a href="https://www.githubstatus.com">www.githubstatus.com</a> collector.github.com raw.githubusercontent.com api.github.com git hub-cloud.s3.amazonaws.com github-production-repository-file-5c1aeb.s3.amazonaws.com github-production-upload-manifest-file-7fdce7.s3.amazonaws.com github-production-user-asset-6210df.s3.amazonaws.com *.rel.tunnels.api.visualstudio.com ...igin.githubusercontent.com *.githubusercontent.com; manifest-src 'self'; media-src <a href="https://github.com/user-images.githubusercontent.com/secured-user-images.githubusercontent.com/">githubusercontent.com/user-images.githubusercontent.com/secured-user-images.githubusercontent.com/</a> private-user-images.githubusercontent.com github-production-user-asset-6210df.s3.amazonaws.com gist.github.com; script-src github.githubassets.com; style-src 'unsafe-inline' <a href="https://github.githubassets.com/">github.githubassets.com/</a> ; upgrade-insecure-requests; worker-src <a href="https://github.com/assets-cdn/worker/">github.com/assets-cdn/worker/</a> <a href="https://github.com/webpack/">github.com/webpack/</a> <a href="https://github.com/assets/">github.com/assets/</a> <a href="https://gist.github.com/assets-cdn/worker/">gist.github.com/assets-cdn/worker/</a>
content-type:	text/html; charset=utf-8
date:	Sat, 01 Feb 2025 03:32:25 GMT
etag:	W/"50928216aee8b41aa04a85a8e4d70a0c"
referrer-policy:	origin-when-cross-origin, strict-origin-when-cross-origin
server:	GitHub.com
set-cookie:	_gh_sess=DWS%2BoOnjwo8FUc6wkizgiKGGAFGF1YXGUHZsjQyWPj0rlmhOFuV%2FxltmNCsJklQrulal2kZ%2FIV0JgfYcY8U9o2lLz1CXG5Up2U6ihYRvWRCTTt8DMUce2mzOGxVJAUyW8vk6Y9gZXPJwVZgEofYYcy5PSOHftUFtWq%2FwU2RexUTiv0K2NWMGkeUbGD954B2VKB7ZsMYKCITCt6fs1j3NNIMh2grkNjagUyyri7g%2BfSVIGDS1DTMvZ9rHmLKZ2d%2Fq3PQaPPpM6tsp5UblxXczfNqtB3BEIWxjTFCLRRViQuRL3yxAlPnNgRU8iF7o%2Bxugd2hT2xKEnYm3k6lYmvJlbVdJSfUr3sEGoOvmEimbbr6ufejAl5heCmO1xtzPw8hg%2BmXviKtGzOtKyUIYYqDxf9au%2F9vafp8HB557wHzK%2BaWd2ZJRr4EmfRI3U

#### Response Headers:

- cache-control: max-age=0, private, must-revalidate
- content-encoding: gzip
- date: Sat, 01 Feb 2025 03:32:25 GMT

▶ Response Headers (4.717 kB) Raw

▼ Request Headers (2.178 kB) Raw

- ② **Accept:** text/html, application/xhtml+xml, application/json
- ② **Accept-Encoding:** gzip, deflate, br, zstd
- ② **Accept-Language:** en-US,en;q=0.5
- ② **Connection:** keep-alive
- ② **Cookie:** \_gh\_sess=ns0fEGIWwj8bqmUerhMXhpPSRZamBsrFiy8dLiDhJeKZBECzCgL8rIAYucoEN%2FgjcO3Rms5YfgQr%2FTiLjGyF9cQVWoRLEgc994IMuEz4E%2FqUQUTKbQeICRw1B%2BzrdVv5y6vyudloYat6jJOBiWeXUbHj%2FdNpU8rze5faEKeL8cuHq8vJs54X0ziRbKBuGix0TrNsCi7ZXDwTcgLW2ZTTdbRwV3f42XnlcdXbBXm84BSSCv6KfwcuPVmQq%2FuwQHxYaMLWPts6SvlOJOFD6aH0Tgr3ALAf6jkd%2BaSUewSvMjvr2dV4fAM1P0nfEN%2B0qV5Aq%2FrpgqrnfpI34vVQ7r20YOCmaAExvqNVV4niDQErvc%2FIBmu n9SPx4JsL8KrrZQd3Tgx7begPQj%2F6JqRcGljexPxlw1Hmb3bNH4dP5g8wo9O6ylgf%2FdD8eq96G9W oHAPD0AZhoTGoBeVeoMnhvbtH0r%2B6f%...olor\_mode%22%3A%22light%22%2C%22light\_them e%22%3A%7B%22name%22%3A%22light%22%2C%22color\_mode%22%3A%22light%22%7D%2C%22 2dark\_theme%22%3A%7B%22name%22%3A%22dark\_dimmed%22%2C%22color\_mode%22%3A%22 dark%22%7D%7D; dotcom\_user=srirams04; GHCC=Required:1-Analytics:1-SocialMedia:1-Advertisin g:1; MicrosoftApplicationsTelemetryDeviceId=0a5345d8-fde4-4372-94af-e756312ae4e1; ai\_session= ClCtKvOOYXllm6N538T0aZ|1738380564896|1738380564896; MSFPC=GUID=6a4f694a6921401f8d7b7 3b1ee6d165d&HASH=6a4f&LV=202502&V=4&LU=1738380547414
- ② **Host:** github.com
- ② **If-None-Match:** W/"0553a38cf48e641c65fa2a261308daa"
- ② **Priority:** u=0
- ② **Referer:** <https://github.com/srirams04>
- ② **Sec-Fetch-Dest:** empty
- ② **Sec-Fetch-Mode:** cors

Request Headers:

- **Accept-language:** en-US,en;q=0.5
- **Accept-encoding:** gzip, deflate, br, zstd
- **Priority:** u=0

## Errors:

The screenshot shows a browser window with multiple tabs open. The active tab is for a 404 error page at <https://github.com/srirassrin04>. The page features a large '404' in the center with a cartoon cat silhouette. Below the error message, there's a list of files and their sizes. At the bottom of the page, there's a note about exceeding a secondary rate limit. The browser's developer tools are open, specifically the Network tab, which lists all the resources loaded by the page. One entry in the list is highlighted with a red border, corresponding to the '404' error message. The Headers section of the developer tools shows standard HTTP headers like 'Content-Type', 'Content-Length', and 'Content-Security-Policy'. The 'Content-Security-Policy' header includes directives such as 'default-src none', 'base-uri self', 'child-src github.com/assets-cdn/worker/github.com/webpack/github.com/assets/gist.github.com/assets-cdn/worker2', and 'connect-src self'. Other headers listed include 'Content-Encoding: gzip' and 'Content-Security-Policy: default-src "none"; base-uri "self"; child-src github.com/assets-cdn/worker/github.com/webpack/github.com/assets/gist.github.com/assets-cdn/worker2; connect-src "self" uploads.githubusercontent.com www.githubstatus.com collector.github.com raw.githubusercontent.com api.github.com git'. The developer tools also show other requests for CSS and JS files.

*“The HTTP 404 Not Found client error response status code indicates that the server cannot find the requested resource. Links that lead to a 404 page are often called broken or dead links and can be subject to link rot. A 404 status code only indicates that the resource is missing without indicating if this is temporary or permanent. If a resource is permanently removed, servers should send the 410 Gone status instead.” [12]*

The screenshot shows a browser window with multiple tabs open. The active tab is for a 429 error page at <https://github.com/srirams04>. The page has a simple message 'Whoa there!' and a note below it stating 'You have exceeded a secondary rate limit. Please wait a few minutes before you try again; in some cases this may take up to an hour.' The browser's developer tools are open, specifically the Network tab, which lists all the resources loaded by the page. One entry in the list is highlighted with a red border, corresponding to the '429' error message. The Headers section of the developer tools shows standard HTTP headers like 'Content-Type', 'Content-Length', and 'Content-Security-Policy'. The 'Content-Security-Policy' header includes directives such as 'default-src none', 'base-uri self', 'child-src github.com/assets-cdn/worker/github.com/webpack/github.com/assets/gist.github.com/assets-cdn/worker2', and 'connect-src self'. Other headers listed include 'Content-Encoding: gzip' and 'Content-Security-Policy: default-src "none"; base-uri "self"; child-src github.com/assets-cdn/worker/github.com/webpack/github.com/assets/gist.github.com/assets-cdn/worker2; connect-src "self" uploads.githubusercontent.com www.githubstatus.com collector.github.com raw.githubusercontent.com api.github.com git'. The developer tools also show other requests for CSS and JS files.

*“The HTTP 429 Too Many Requests client error response status code indicates the client has sent too many requests in a given amount of time. This mechanism of asking the client to slow down the rate of requests is commonly called “rate limiting”. A Retry-After header may be included to this response to indicate how long a client should wait before making the request again.” [13]*

The screenshot shows the Mozilla Firefox developer tools Network tab. A POST request to `/repositories/pocket-chess` resulted in a 422 error. The error message is: "The repository pocket-chess already exists on this account." Below the error, it says: "Template repositories let users generate new repositories with the same directory structure and files. Learn more about template repositories."

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	POST	collector.github...	collect	vendors-node...	plain	NS_BINDING_A...	0 B
200	GET	github.com	favicon.png	FaviconLoaders...	png	cached	33.27 kB
200	GET	github.com	favicon.svg	FaviconLoaders...	svg	cached	959 B
200	POST	api.github.com	stats	ui.packages fail...	plain	3.09 kB	0 B
200	POST	api.github.com	stats	ui.packages fail...	plain	2.56 kB	0 B
200	POST	collector.github...	collect	vendors-node...	plain	NS_BINDING_A...	0 B
200	POST	github.com	check-name?current_name=optimal-move-finder	vendors-node...	html	4.68 kB	20 B
200	GET	github.github...	octocat-spinner-lightmode-3acfd133ead5.svg	img	svg	4.11 kB	3.46 kB
422	POST	github.com	check-name?current_name=optimal-move-finders	vendors-node...	fragment	4.63 kB	77 B
200	GET	github.github...	alert-fill-12-fc42d910fc15.svg	app.assets.mod...	svg	940 B	287 B

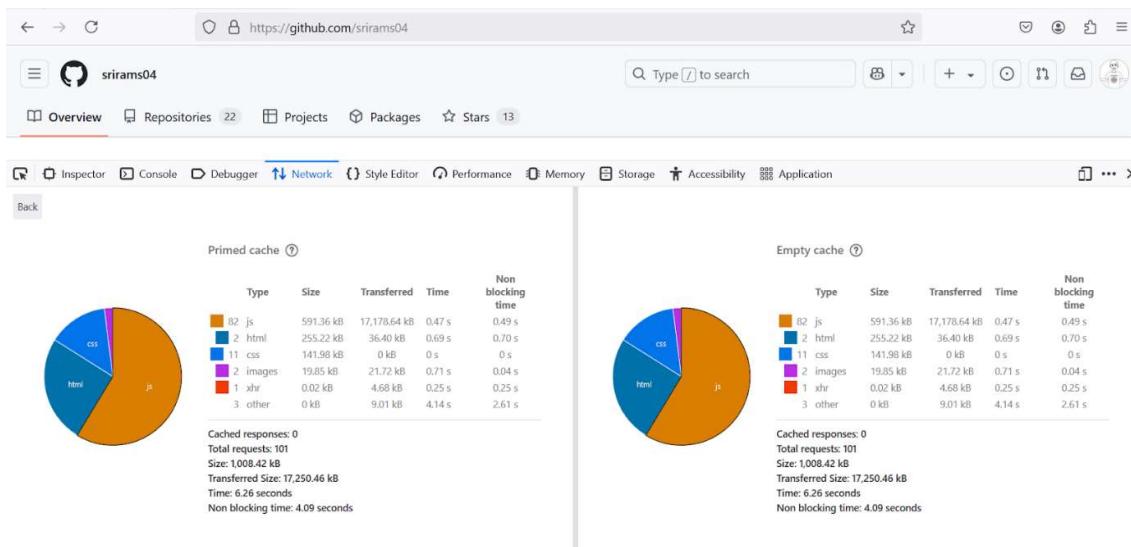
Address: 20.207.73.82:443

"The **HTTP 422 Unprocessable Content** client error response status code indicates that the server understood the content type of the request content, and the syntax of the request content was correct, but it was unable to process the contained instructions. Clients that receive a 422 response should expect that repeating the request without modification will fail with the same error." [14]

## Part (c)

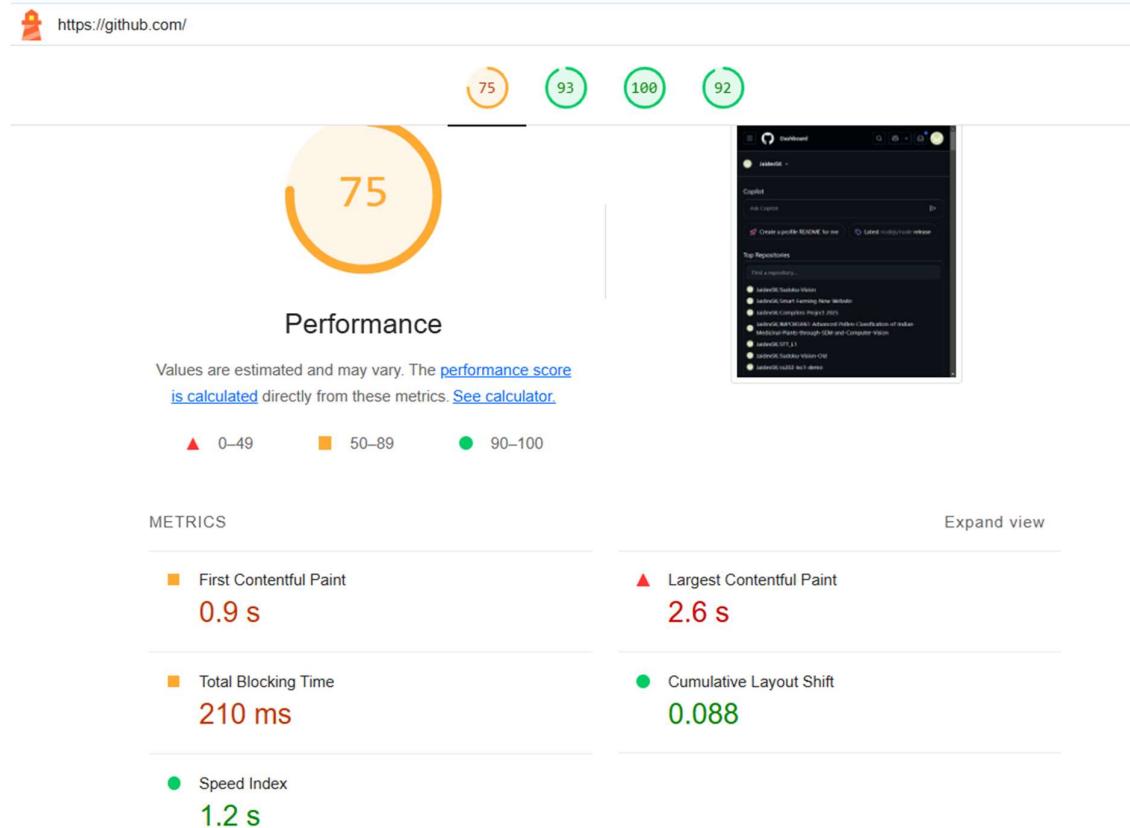
Browser Name: Mozilla Firefox

Screenshots of Performance Metrics:



Browser Name: Chrome

Screenshots of Performance Metrics:



## Cookies:

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
_Host_us...	vM0ZYXswf4wTrR...	github.com	/	Sat, 15 Feb 2025 03:...	77	true	true	Strict	Sat, 01 Feb 2025 03:...
_device_id	a3b000d86a2c28a...	github.com	/	Sun, 01 Feb 2026 03:...	42	true	true	Lax	Sat, 01 Feb 2025 03:...
_gh_ses...	6zbg932UyJWz2BHP...	github.com	/	Session	662	true	true	Lax	Sat, 01 Feb 2025 12:...
_octo	GHI.1.772484801.1...	github.com	/	Sat, 31 Jan 2026 17:...	31	false	true	Lax	Sat, 01 Feb 2025 03:...
_ai_session	CICKWOOVXlmfNS...	github.com	/	Sat, 01 Feb 2025 03:...	60	false	true	None	Sat, 01 Feb 2025 03:...
_color_m...	%7B%22color_mod...	github.com	/	Session	222	false	true	Lax	Sat, 01 Feb 2025 03:...
_cpu_bucket	lg	github.com	/	Session	12	false	true	Lax	Sat, 01 Feb 2025 03:...
_disabled_...	copilot_free_global	github.com	/	Mon, 03 Mar 2025 0...	47	false	true	Lax	Sat, 01 Feb 2025 03:...
_dotcom_...	srirams04	github.com	/	Sun, 01 Feb 2026 03:...	20	true	true	Lax	Sat, 01 Feb 2025 03:...
GHCC	Required:1-Analytic...	github.com	/	Thu, 31 Jul 2025 03:...	54	false	true	Lax	Sat, 01 Feb 2025 03:...
logged_in	yes	github.com	/	Sun, 01 Feb 2026 03:...	12	true	true	Lax	Sat, 01 Feb 2025 03:...
Microsoft...	0a5345d8-fde4-437...	github.com	/	Sun, 01 Feb 2026 03:...	74	false	true	None	Sat, 01 Feb 2025 03:...
MSFPC	GUID::6a4f694a692...	github.com	/	Sun, 01 Feb 2026 03:...	83	false	true	None	Sat, 01 Feb 2025 03:...
preferred_...	light	github.com	/	Session	25	false	true	Lax	Sat, 01 Feb 2025 03:...
saved_us...	75443405%3AvM0...	github.com	/	Fri, 02 May 2025 03:...	78	true	true	Lax	Sat, 01 Feb 2025 03:...
tz	Asia%2FKolkata	github.com	/	Session	16	false	true	Lax	Sat, 01 Feb 2025 03:...
tz	Asia%2FKolkata	github.com	/	Session	16	true	true	Lax	Sat, 01 Feb 2025 03:...
user_sessi...	vM0ZYXswf4wTrR...	github.com	/	Sat, 15 Feb 2025 03:...	60	true	true	Lax	Sat, 01 Feb 2025 03:...

All the header-value pairs for the selected cookie are visible on the right-hand side of the above image. The image below displays the flags associated with the request and response headers on reloading github.com.

## Response:

② **strict-transport-security:** max-age=31536000; includeSubdomains; preload  
② **vary:** X-PJAX, X-PJAX-Container, Turbo-Visit, Turbo-Frame  
② **vary:** Accept-Encoding, Accept, X-Requested-With  
② **x-content-type-options:** nosniff  
X-Firefox-Spdy: h2  
② **x-frame-options:** deny  
**x-github-request-id:** 8914:0A6F:5F35AF:7CB2FC:679D9919  
② **x-xss-protection:** 0

#### Request:

② **Host:** github.com  
② **Priority:** u=4  
② **Referer:** <https://github.com/srirams04>  
② **Sec-Fetch-Dest:** empty  
② **Sec-Fetch-Mode:** cors  
② **Sec-Fetch-Site:** same-origin  
② **TE:** trailers  
② **User-Agent:** Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:134.0) Gecko/20100101 Firefox/134.0  
**X-Requested-With:** XMLHttpRequest

## References

- [1] [https://en.wikipedia.org/wiki/Ethernet\\_frame](https://en.wikipedia.org/wiki/Ethernet_frame)
- [2] <https://en.wikipedia.org/wiki/EtherType>
- [3] <https://en.wikipedia.org/wiki/IPv4>
- [4] [https://en.wikipedia.org/wiki/Transmission\\_Control\\_Protocol](https://en.wikipedia.org/wiki/Transmission_Control_Protocol)
- [5] [https://en.wikipedia.org/wiki/User\\_Datagram\\_Protocol](https://en.wikipedia.org/wiki/User_Datagram_Protocol)
- [6] [https://en.wikipedia.org/wiki/Internet\\_Control\\_Message\\_Protocol](https://en.wikipedia.org/wiki/Internet_Control_Message_Protocol)
- [7] <https://en.wikipedia.org/wiki/IPv6>
- [8] <https://docs.python.org/3/library/socket.html>
- [9] <https://docs.python.org/3/library/struct.html>
- [10] <https://docs.python.org/3/library/signal.html>
- [11] <https://docs.python.org/3/library/time.html>
- [12] <https://developer.mozilla.org/en-US/docs/Web/HTTP>Status/404>
- [13] <https://developer.mozilla.org/en-US/docs/Web/HTTP>Status/429>
- [14] <https://developer.mozilla.org/en-US/docs/Web/HTTP>Status/422>
- [15] [https://en.wikipedia.org/wiki/Apache\\_JServ\\_Protocol](https://en.wikipedia.org/wiki/Apache_JServ_Protocol)
- [16] [https://en.wikipedia.org/wiki/Multicast\\_DNS](https://en.wikipedia.org/wiki/Multicast_DNS)

- [17] <https://learn.microsoft.com/en-us/windows/win32/fileio/microsoft-smb-protocol-and-cifs-protocol-overview>
- [18] [https://learn.microsoft.com/en-us/openspecs/windows\\_protocols/ms-brws/3cfbad92-09b3-4abc-808f-c6f6347d5677](https://learn.microsoft.com/en-us/openspecs/windows_protocols/ms-brws/3cfbad92-09b3-4abc-808f-c6f6347d5677)
- [19] <https://en.wikipedia.org/wiki/XMPP>
- [20] [https://www.hp.com/hpinfo/newsroom/press\\_kits/2009/domorewithless/HPVirtualRooms.PDF](https://www.hp.com/hpinfo/newsroom/press_kits/2009/domorewithless/HPVirtualRooms.PDF)
- [21] [https://en.wikipedia.org/wiki/List\\_of\\_TCP\\_and\\_UDP\\_port\\_numbers](https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers)

## Supplementary Documents and Data

I have also provided the supplementary code, which displays my stepwise approach to analyse the problem. The data obtained in each step is also provided as comments in the remaining part of the program files.