Computer Networks(CS 331)
Assignment 01: Raw Sockets

Submitted to:
Prof. Sameer Kulkarni

Submitted by:
Group 1,

Mitansh Patel	24120033	mitansh.patel@iitgn.ac.in
Gaurav Kumar	24120027	gaurav.k@iitgn.ac.in

Please find the code and pertaining files at on Github link:
Link or the URL: https://github.com/Mitansh-Patel-24120033/CN_Assignment_1

Note:
The assignment is done in the programming language of ‘Python’.
For sniffing purposes, ‘scapy’ tool is utilized on the assigned ‘1.pcap’ file.
‘tcpreplay’ tool is run(at topspeed) on Booted Linux OS(Fedora 41) and VM(VirtualBox, Ubuntu 24LTS).

Part 1: Metrics and Plots:
From the chosen 1.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).

Total Data Transferred: 364571772 bytes
Total Packets(with IP layer): 805559
Min Packet Size: 54 bytes
Max Packet Size: 1514 bytes
Avg Packet Size: 452.57 bytes
Unique Source-Destination Pairs: 41903

2. Find unique source-destination pairs (source IP:port and destination IP:port) in the captured data.

Number of Unique Source-Destination Pairs: 41927

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.

```
Source-Destination Pair with Most Data: ('172.16.133.95:49358', '157.56.240.102:443') (17342229 bytes)
```

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:

On same machine:

Sniffer Program(Student 1):

```
Capture Duration: 40.225 seconds
Packets Per Second (PPS): 20026.312
Megabits Per Second (Mbps): 72.506
```

tcpreplay(Student 1):

```
Test start: 1970-01-01 07:54:54.291343885 ...
Test complete: 1970-01-01 07:55:19.151184691
Actual: 806013 packets (364643322 bytes) sent in 24.85 seconds
Rated: 14667967.3 Bps, 117.34 Mbps, 32422.29 pps
Flows: 41723 flows, 1678.32 fps, 805314 unique flow packets, 454 unique non-flow packets
Statistics for network device: enp0s20f0u4
        Successful packets:       806013
        Failed packets:           0
        Truncated packets:        0
        Retried packets (ENOBUFS): 0
        Retried packets (EAGAIN):  0
```

On different machines:

Sniffer Program(Student 2):

```
Capture Duration: 55.358 seconds
Packets Per Second (PPS): 14551.353
Megabits Per Second (Mbps): 52.685
```

tcpreplay(student 1):

```
Test start: 1970-01-01 07:54:54.291343885 ...
Test complete: 1970-01-01 07:55:19.151184691
Actual: 806013 packets (364643322 bytes) sent in 24.85 seconds
Rated: 14667967.3 Bps, 117.34 Mbps, 32422.29 pps
Flows: 41723 flows, 1678.32 fps, 805314 unique flow packets, 454 unique non-flow packets
Statistics for network device: enp0s20f0u4
        Successful packets:      806013
        Failed packets:          0
        Truncated packets:       0
        Retried packets (ENOBUFS): 0
        Retried packets (EAGAIN):  0
```

Part 2: Catch Me If You Can (40 points)

For the designated X.pcap file, extend the program to sniff and answer the specific questions:

Q1) An email was sent from localhost. What was the subject of the email?

Hint: Analyze SMTP packets and look for the "Subject" field in the email headers

>> Request extension for assignment 1 of CS433

Q2) What is the recipient's email address for the email sent from localhost?

>> sameersir@iitgn.ac.in

Q3) What is the IP address resolved for the domain routerswitches.com?

Hint: Inspect DNS query and response packets.

>> 93.184.216.34

Q4) Which DNS server was used to resolve the domain routerswitches.com?

>> 8.8.8.8

Part 3: Capture the packets

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.

SIP (Session Initiation Protocol):

- **Usage**: SIP is used for initiating, maintaining, and terminating real-time communications, such as voice and video calls.
- **RFC**: RFC 3261

NTP (Network Time Protocol):

- **Usage**: NTP is used to synchronize the clocks of computers over a network to ensure consistent and accurate time across systems.
- **RFC**: RFC 5905

SNMP (Simple Network Management Protocol):

- **Usage**: SNMP is used for managing and monitoring network devices like routers and switches. It helps administrators gather information about network health.
- **RFC**: RFC 1157

XMPP (Extensible Messaging and Presence Protocol):

- **Usage**: XMPP is used for real-time messaging, presence information, and contact list management, typically for instant messaging and VoIP.
- **RFC**: RFC 6120

RDP (Remote Desktop Protocol):

- **Usage**: RDP is used for remote access to computers, allowing users to interact with the desktop of a remote system as if they were physically present.
- **RFC**: While there isn’t a specific RFC for RDP, it is standardized by Microsoft.

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.

i) canarabank.in
Request Line: `GET / HTTP/1.1`
Application Layer Protocol Version: HTTP/1.1
IP Address: 107.162.160.8:443
Connection: Non-persistent (indicated by `Connection: close` in the response headers)

ii) github.com

Request Line: `GET / HTTP/1.1`
Application Layer Protocol Version: HTTP/1.1
IP Address: 20.207.73.82:443
Connection: Persistent (indicated by `Connection: keep-alive` in the request headers)

iii) netflix.com

Request Line: `GET / HTTP/1.1`
Application Layer Protocol Version: HTTP/1.1
IP Address: 54.73.148.110:443
Connection: Non-persistent (indicated by `Connection: close` in the response headers)

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.

Request Headers:

1. **User-Agent**:
   - **Value**: `Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/132.0.0.0 Mobile Safari/537.36 Edg/132.0.0.0`
   - **Description**: Provides information about the client (browser) making the request.

2. **Accept**:
    - **Value**: `text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7`
    - **Description**: Specifies the media types that the client can understand.
3. **Host**:
    - **Value**: `github.com`
    - **Description**: Specifies the domain name of the server and (optionally) the TCP port number on which the server is listening.

Response Headers:

1. **Content-Type**:
    - **Value**: `text/html; charset=utf-8`
    - **Description**: Indicates the media type of the resource and the character encoding.
2. **Cache-Control**:
    - **Value**: `max-age=0, private, must-revalidate`
    - **Description**: Directives for caching mechanisms in both requests and responses.
3. **Server**:
    - **Value**: `GitHub.com`
    - **Description**: Contains information about the software used by the origin server to handle the request.

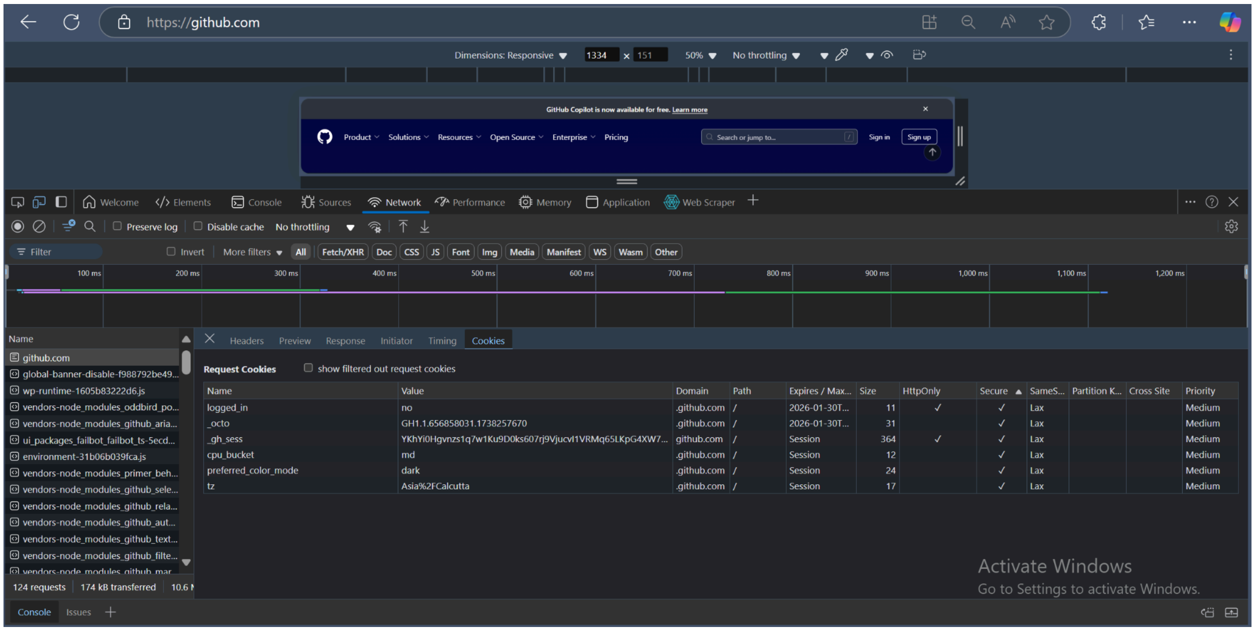
HTTP Error Codes:

1. **404 Not Found**:
    - **Description**: The requested resource could not be found on the server.
2. **500 Internal Server Error**:
    - **Description**: The server encountered an unexpected condition that prevented it from fulfilling the request.
3. **403 Forbidden**:
    - **Description**: The server understood the request but refuses to authorize it.

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

Browser used: Google Chrome

Associated flags with request and response headers:



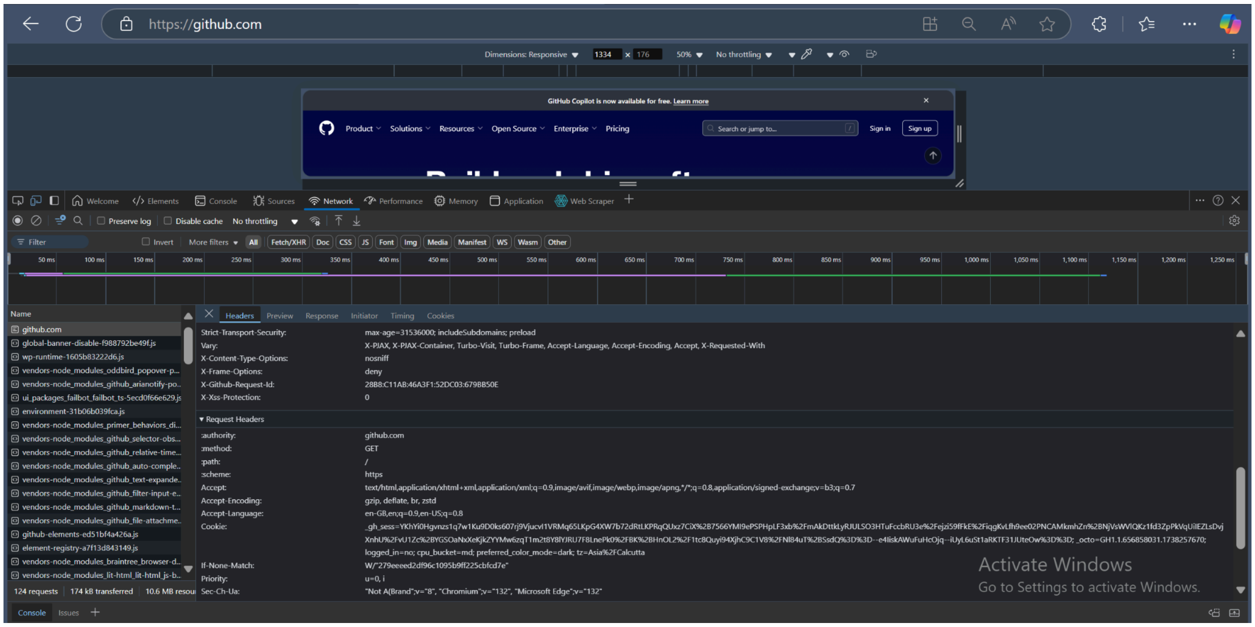
https://github.com

Request Cookies □ show filtered out request cookies

Name	Value	Domain	Path	Expires / Max...	Size	HttpOnly	Secure ▲	SameS...	Partition K...	Cross Site	Priority
logged_in	no	.github.com	/	2026-01-30T...	11	✓	✓	Lax			Medium
_octo	GH1.1.656858031.1738257670	.github.com	/	2026-01-30T...	31		✓	Lax			Medium
_gh_sess	YKhYi0Hgvnzs1q7w1Ku9D0ks607rj9Vjucvl1VRMq65LKpG4XW7...	github.com	/	Session	364	✓	✓	Lax			Medium
cpu_bucket	md	.github.com	/	Session	12		✓	Lax			Medium
preferred_color_mode	dark	.github.com	/	Session	24		✓	Lax			Medium
tz	Asia%2FCalcutta	.github.com	/	Session	17		✓	Lax			Medium

124 requests | 174 kB transferred

List of cookies:

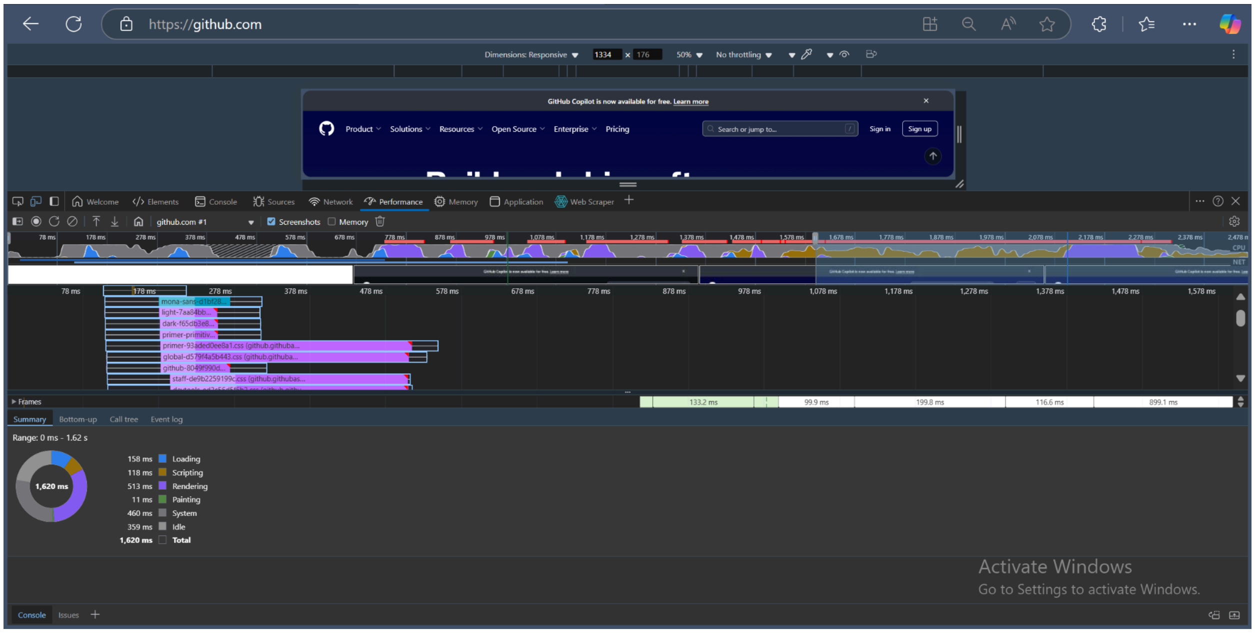


https://github.com

```
Strict-Transport-Security:   max-age=31536000; includeSubdomains; preload
Vary:                        X-PJAX, X-PJAX-Container, Turbo-Visit, Turbo-Frame, Accept-Language, Accept-Encoding, Accept, X-Requested-With
X-Content-Type-Options:      nosniff
X-Frame-Options:             deny
X-Github-Request-Id:         28B8:C11AB:46A3F1:52DC03:679BB50E
X-Xss-Protection:            0
▾ Request Headers
:authority:                  github.com
:method:                     GET
:path:                       /
:scheme:                     https
Accept:                      text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding:             gzip, deflate, br, zstd
Accept-Language:             en-GB,en;q=0.9,en-US;q=0.8
Cookie:                      _gh_sess=YKhYi0Hgvnzs1q7w1Ku9D0ks607rj9Vjucvl1VRMq65LKpG4XW7b72dRtLKPRqQUxz7CiX%2B7566YMI9ePSPHpLF3xb%2FmAkDttkLyRJULSO3HTuFccbRU3e%2Fejzi59fFkE%2FiqgKvLfh9ee02PNCAMkmhZn%2BNjVsWVlQKz1fd3ZpPkVqUiIEZLsDvj
                             XnhU%2FvU1Zc%2BYGSOaNxXeKjkZYYMw6zqT1m2t8Y8lYJRU7F8LnePk0%2FBK%2BHnOL2%2F1tc8Quyi94XjhC9C1V8%2FNl84uT%2BSsdQ%3D%3D--e4liskAWuFuHcOjq--iUyL6uSt1aRKTF31JUteOw%3D%3D; _octo=GH1.1.656858031.1738257670;
                             logged_in=no; cpu_bucket=md; preferred_color_mode=dark; tz=Asia%2FCalcutta
If-None-Match:               W/"279eeeed2df96c1095b9ff225cbfcd7e"
Priority:                    u=0, i
Sec-Ch-Ua:                   "Not A(Brand";v="8", "Chromium";v="132", "Microsoft Edge";v="132"
```

124 requests | 174 kB transferred | 10.6 MB resou

Performance metrics:



https://github.com

Dimensions: Responsive 1334 × 176 50% No throttling

github.com #1 ☑ Screenshots □ Memory

Summary | Bottom-up | Call tree | Event log

Range: 0 ms - 1.62 s

158 ms	🟦	Loading
118 ms	🟫	Scripting
513 ms	🟪	Rendering
11 ms	🟩	Painting
460 ms	⬜	System
359 ms	⬜	Idle
1,620 ms	□	**Total**