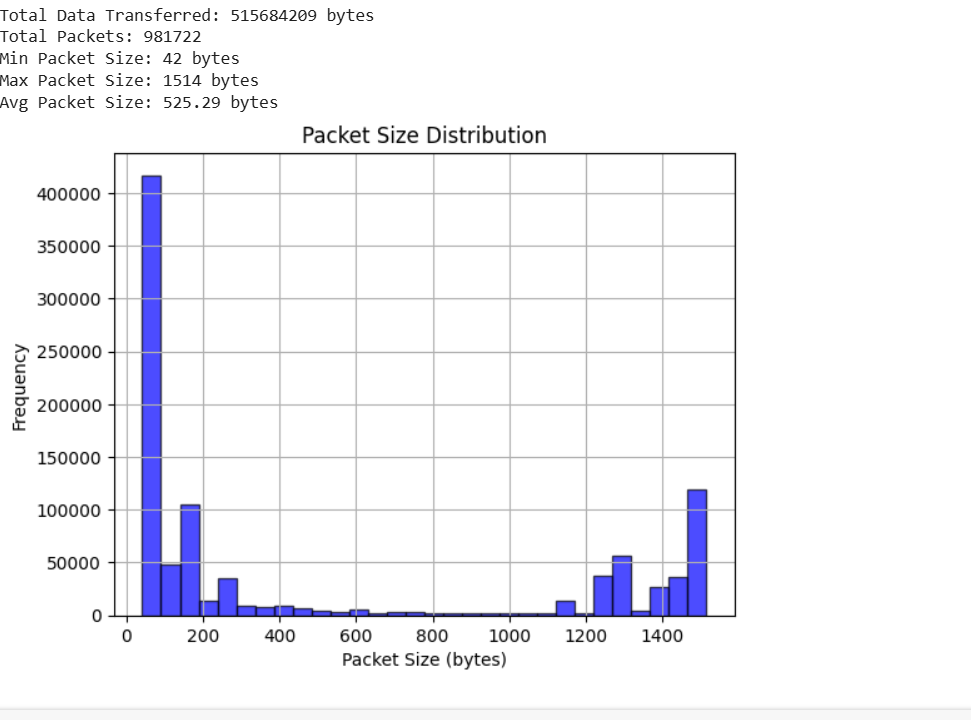
Report

Part 1

Q1



Q2

unique source-destination pairs (source IP:port and destination IP:port) in the captured data are in part1 (ques2).txt.

Q3

Dictionary is in Source IP Flow Count.txt file.

Part 3

* 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.

Ans

1.

**AJP13 (Apache JServ Protocol version 1.3)** is an application layer protocol.

**Overview of AJP13:**

* **Operation**: AJP13 is a binary protocol that allows web servers (such as Apache HTTP Server) to communicate efficiently with application servers (such as Apache Tomcat). It is mainly used for forwarding HTTP requests from a web server to a servlet container.
* **Layer**: Application Layer (Layer 7 of the OSI model)
* **Key Features**:
  + Faster than HTTP due to its binary format.
  + Supports request forwarding, load balancing, and session management.
  + Uses persistent connections to improve efficiency.

**However, there is no official RFC for AJP13,** as it is primarily used within Apache-based environments.

**2. TELNET**

Telnet stands for the teletype network. It helps in terminal emulation. It allows Telnet clients to access the resources of the Telnet server. It is used for managing files on the Internet. It is used for the initial setup of devices like switches. The telnet command is a command that uses the Telnet protocol to communicate with a remote device or system. The port number of the telnet is 23. The rfc is 854.

3.

STUN (Session Traversal Utilities for NAT) is considered an application layer protocol because it operates at the highest level of the TCP/IP model, enabling applications to discover their public IP address and port behind a NAT, which is crucial for establishing peer-to-peer connections in applications like video calls and online gaming. The RFC that defines the STUN protocol is RFC 5389.

4.

mDNS stands for Multicast DNS, and the RFC that defines it is RFC 6762. This protocol allows devices on a local network to discover each other and resolve hostnames to IP addresses without the need for a central DNS server.

5.

DHCP (Dynamic Host Configuration Protocol) is considered an application layer protocol, and its corresponding RFC is 2131.  DHCP operates on a client-server model where devices (clients) request network configuration information from a designated DHCP server.

**iii) netflix.com**

**(a) Request Line and Connection Persistence**

* **Request Line: GET /in/ HTTP/1.1**
  + The request uses the HTTP/1.1 protocol.
  + The request is sent to [www.netflix.com](http://www.netflix.com), which resolves to the IP address 3.251.50.149.
* **Connection Persistence:**
  + The response does not explicitly include Connection: close, which means persistent connections are enabled by default in HTTP/1.1.
  + If the keep-alive directive is present, it would confirm that multiple requests can be handled over the same TCP connection**.**

**(b) Header Fields and HTTP Error Codes**

**Three Header Field Names and Values**

1. **From Request Headers:**
   * User-Agent: "Mozilla/5.0 (Windows NT 10.0; Win64; x64) Apple WebKit/537.36 (KHTML, like Gecko) Chrome/132.0.0.0 Safari/537.36"
   * Refer-Policy: "strict-origin-when-cross-origin"
   * Accept-Encoding: "grip, deflate, brr, zest"
2. **From Response Headers:**
   * Content-Type: "text/html; charset=utf-8"
   * Strict-Transport-Security: "max-age=31536000; include Subdomains"
   * X-Frame-Options: "DENY"

**Three Common HTTP Error Codes**

1. 404 Not Found – Occurs when a requested resource (such as a page or an image) does not exist on the server.
2. 403 Forbidden – The server refuses to serve the request due to access restrictions.
3. 500 Internal Server Error – A generic error when something unexpected happens on the server.

**(c) Performance Metrics, Cookies, and Browser Details**

**Cookies Used in Request & Response**

* **Request Cookies:**

nfvdid=BQFmAAEBEGwZBxdoKrCH2nk3dRrhL7xAHkr8XZfrPqS1PNluHG6wv-Dysv-BZmYOrEuwhpeDRXzVATRDBAt2ONIMTPxnA4Xhg7WBoU-juDu9q-4KmRfD8g%3D%3D;

SecureNetflixId=v%3D3%26mac%3DAQEAEQABABTgElixymWqtidfdVEii\_QRhx8etXG17pI.%26dt%3D1738347612139;

NetflixId=v%3D3%26ct%3DBgjHlOvcAxLAAZviay9iZBBRrHBk9jZJ2G-Btkw09A0n5Dsr7OXLXO0xzxjG3Hrkmu\_3KgXr1Kzw4EnlTmqBI8D\_glY-\_Em\_dw8HLz4NOtNQhiOMAwduxKvny73UN-azlwOY3xwevq0PLaGrXfrqqs22gBetiRrR6NM9wfmVio3udM7cc-JGGsmZmHhYASF4eWsVOwxdfxM\_wOEznYeB1axnERi0rg\_yMaKwQ4dhUTAO5RBKllfpBxvc\_hW-pRWDpjyVKKYw2m7ApxgGIg4KDDNt4DPVowYKCSQFxg.

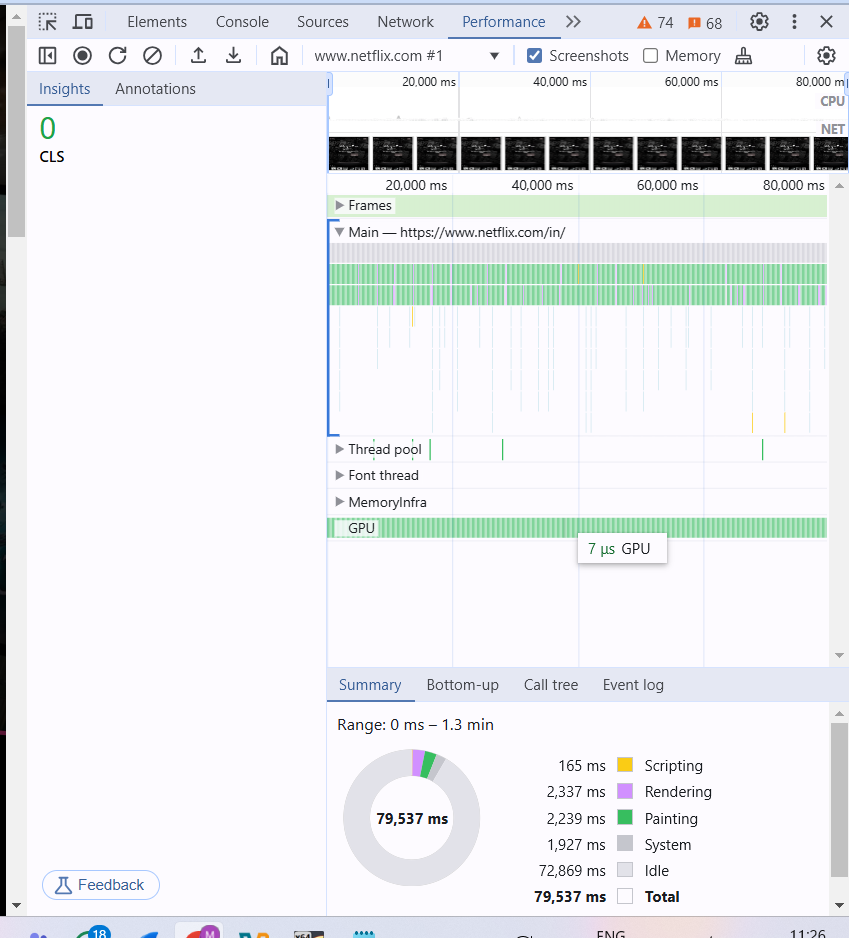
* **Response Cookies (with Flags):**

netflix-sans-normal-3-loaded=true; Max-Age=7776000; Domain=.netflix.com; Path=/

netflix-sans-bold-3-loaded=true; Max-Age=7776000; Domain=.netflix.com; Path=/flows=206c54ba-0646-4ab0-9ef1-d83a1b66e5b6; Max-Age=10800; Domain=.netflix.com;

Path=/

* + Max-Age sets how long the cookie is valid.
  + Secure ensures the cookie is transmitted only over HTTPS.
* **Browser Name:** Google Chrome
* **Version:** 132.0.0.0

**.**