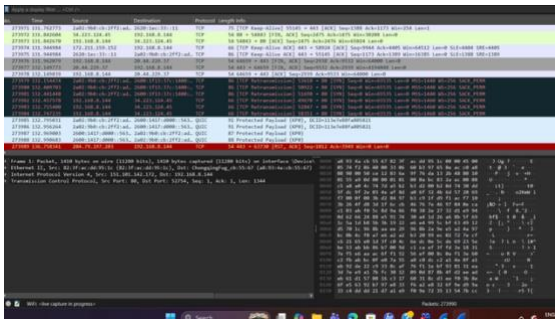
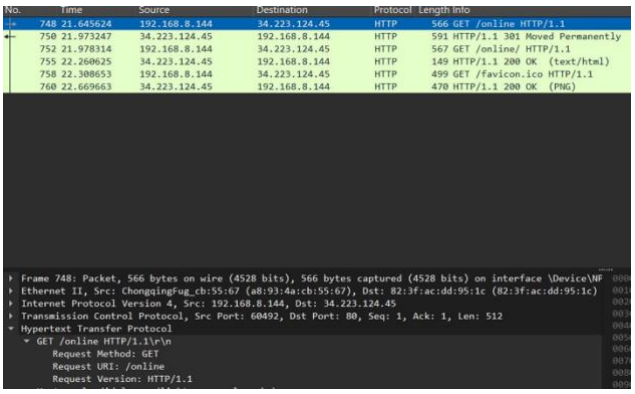
Lap 1

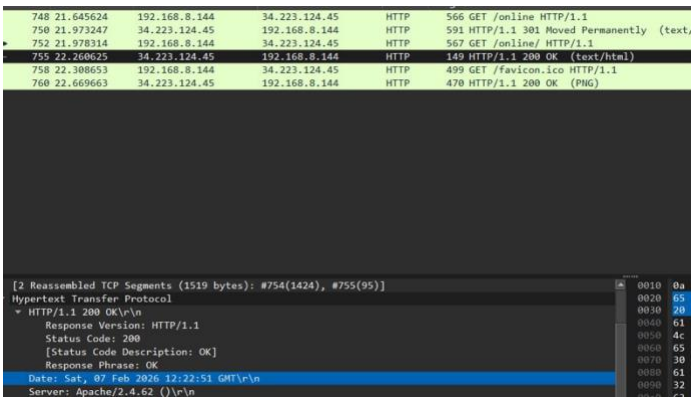
Part 1:

Task 1:



Task 2:





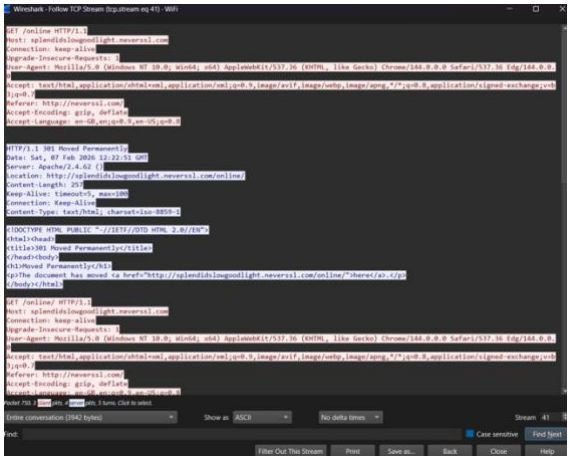
When a user accesses a web page, the web browser sends an HTTP request to the web server. This request typically uses the GET method, which asks the server to retrieve a specific resource identified by a URL. The request includes important header fields such as the Host, User-Agent, and Accept, which describe the client and the type of content it can process.

Upon receiving the request, the server processes it and sends back an HTTP response. The response contains a status code (such as 200 OK indicating success or 301 Moved Permanently indicating redirection), response headers, and optionally the requested content (e.g., HTML). This request–response exchange forms the basis of communication between the client and the server in the HTTP protocol

Part2:

Task 1:

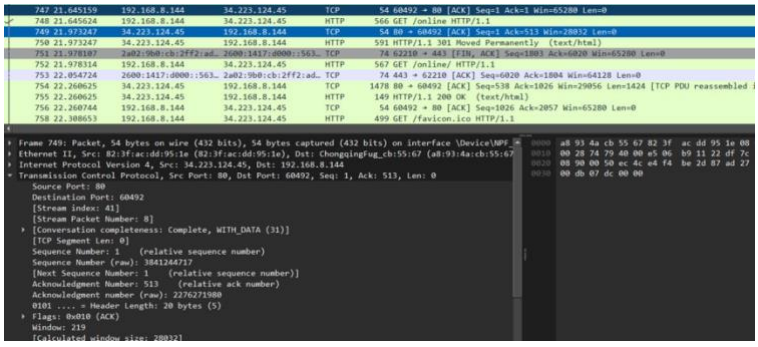
The TCP stream contained unreadable characters because the transmitted data was not in plain text format. This indicates that TCP transfers raw binary data, while application-layer protocols such as HTTP interpret this data into meaningful information

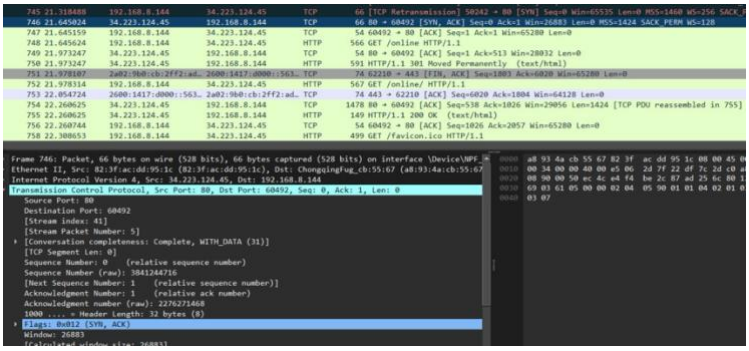


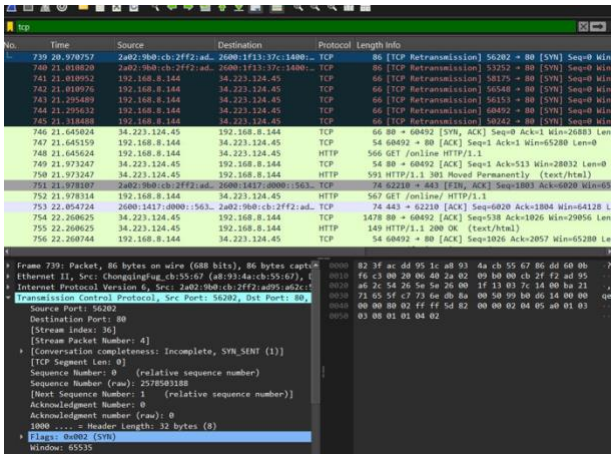
Task 2

1: TCP Three-Way Handshake

The TCP three-way handshake consists of SYN, SYN-ACK, and ACK packets. The sequence and acknowledgment numbers are used to synchronize communication between the client and server.



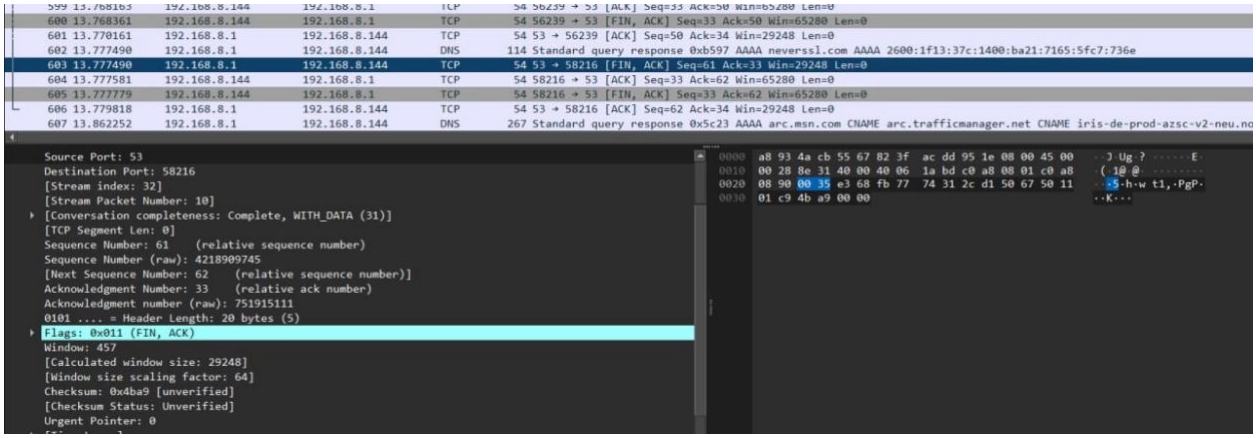




2: Data Transfer: After the TCP connection is established, data transfer begins between the client and the server. The data is transmitted in segments that are acknowledged to ensure reliable delivery. TCP uses sequence and acknowledgment numbers to maintain correct data ordering and detect any lost packets, guaranteeing accurate and complete transmission.

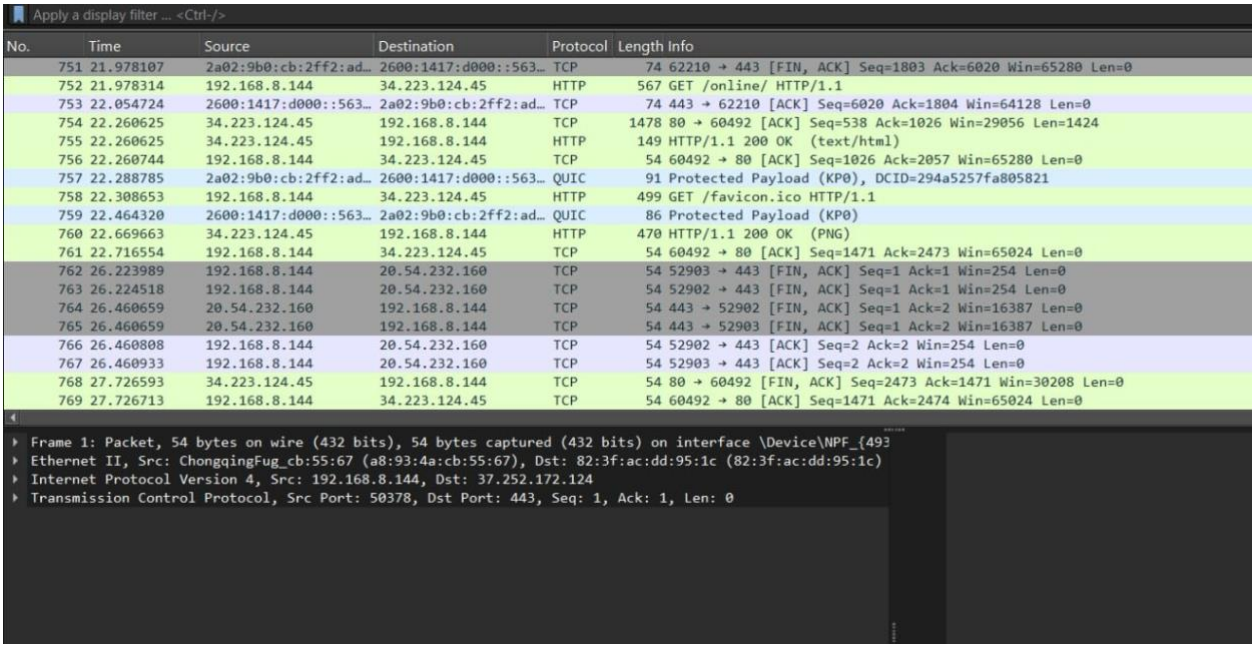
3: TCP Termination

Once data transfer is completed, the TCP connection is terminated in an orderly manner. This process uses FIN and ACK packets exchanged between the client and the server to confirm that both sides have finished sending data, ensuring a graceful and reliable connection closure.



Part 3:

Task 1:



Task 2: UDP is a simple transport-layer protocol that provides minimal communication services. It has a small header and does not include mechanisms for reliability, ordering, or error recovery. This simplicity reduces overhead and allows faster data transmission compared to more complex protocols such as TCP.

UDP is a connectionless protocol, meaning it does not establish a connection or perform a handshake before sending data. Packets are transmitted independently without confirming delivery or readiness of the receiver, which makes UDP suitable for time-sensitive applications where speed is more important than reliability.