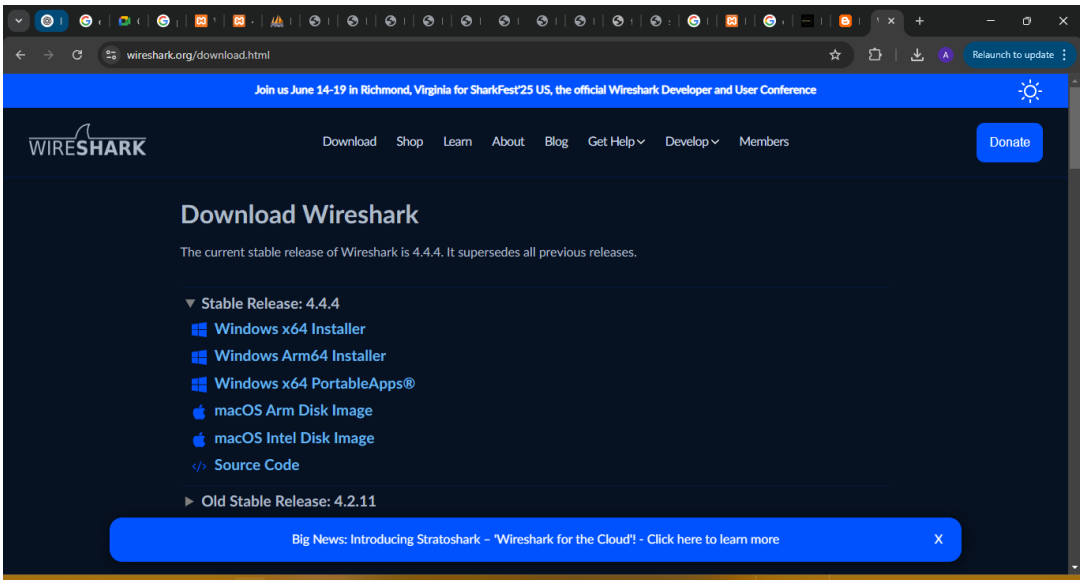
Gacanga Alex Mwangi

C028/401599/2023

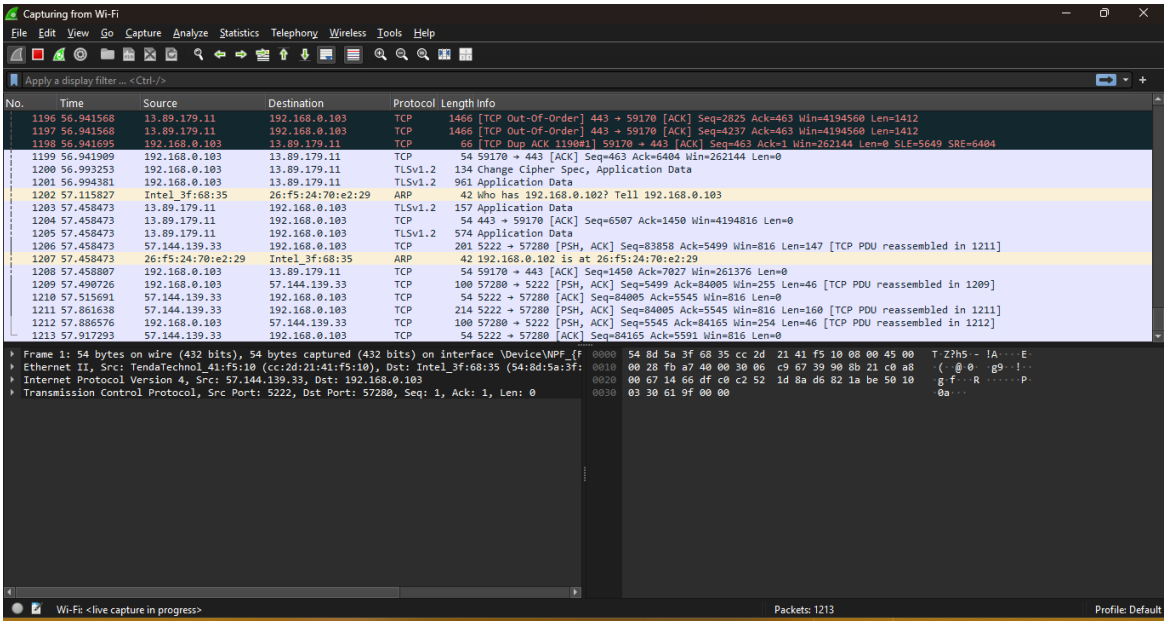
Introduction to Internet Programming assignment two

A) Using Wireshark (or equivalent) to Analyze HTTP Headers

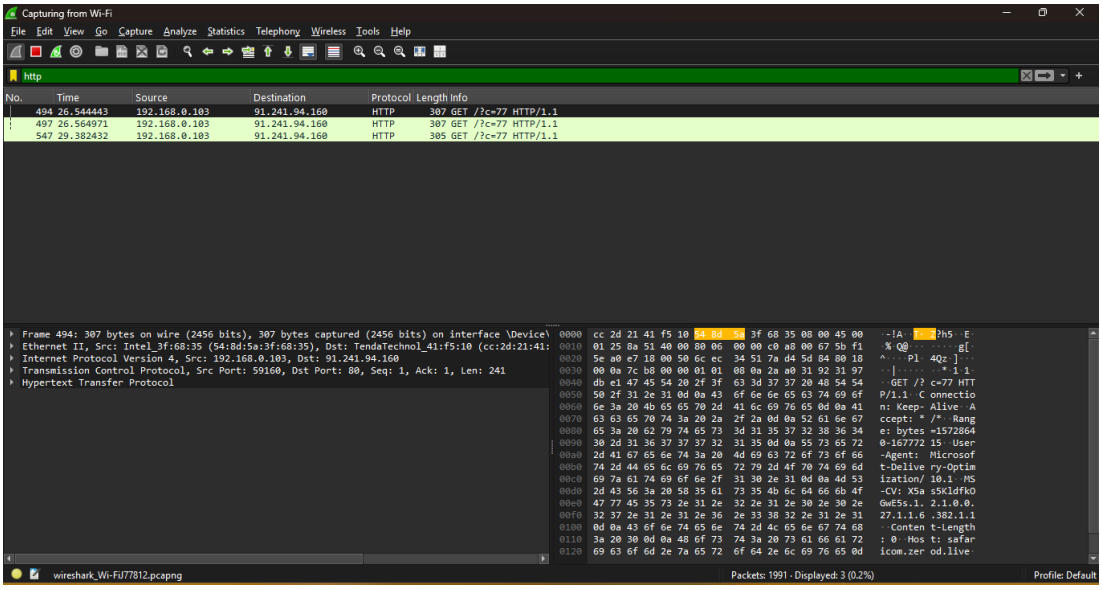
1. **Install Wireshark**: If you don't have Wireshark installed, download and install it from Wireshark's official site.



2. **Capture Network Traffic**:
    - Open Wireshark and select the network interface that you are using (for example, Wi-Fi or Ethernet).
    - In the Wireshark interface, start capturing packets by clicking the **Start capturing packets** button (shaped like a shark fin).

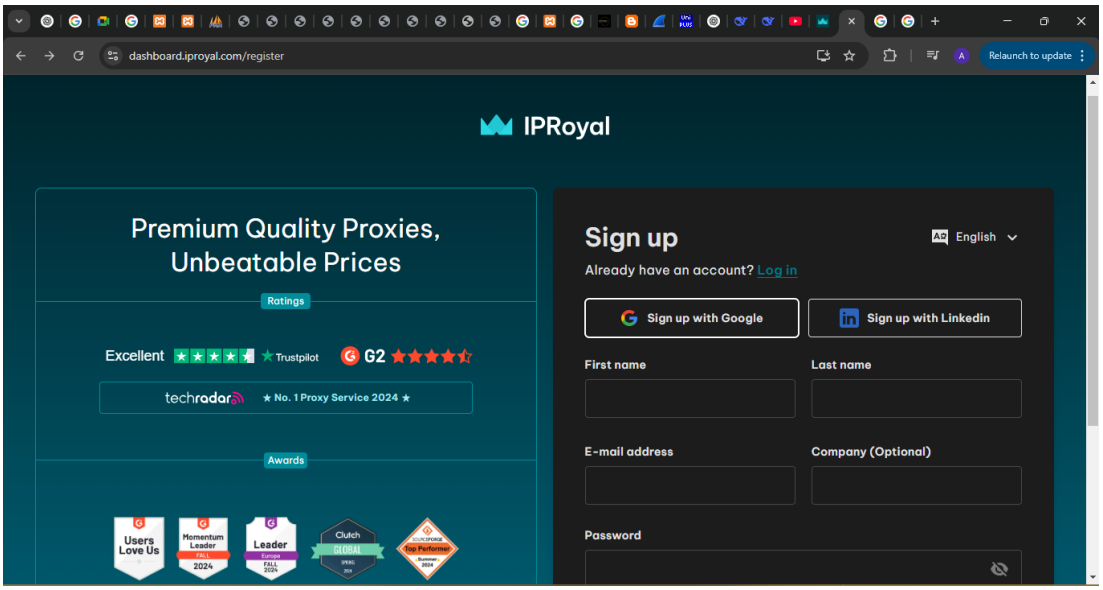


3. **Apply a Display Filter for HTTP**: To focus only on HTTP traffic, use a display filter to capture only HTTP traffic.
    - In the filter bar, enter: http
    - This will show only HTTP traffic (GET, POST, etc.) in the packet capture

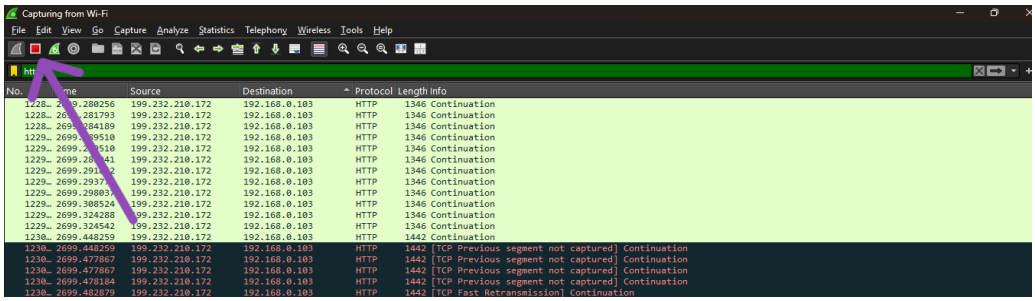


4. **Generate HTTP GET and POST Requests**:
    - **HTTP GET Request**: You can generate a GET request by visiting any website using your browser

- **HTTP POST Request**: You can generate a POST request by submitting a form on a website, for example, a login or registration form. For me I used www.iproyal.com



5. **Stop Capture**: After the requests are made, stop the packet capture by clicking the **Stop capturing packets** button (square red button).

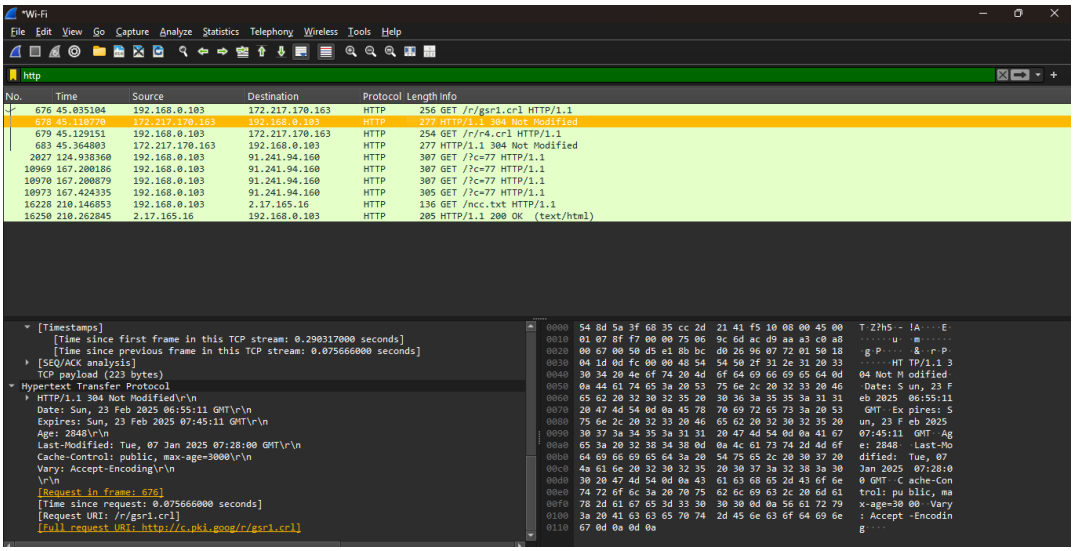


6. **Analyze HTTP Packets**: In the packet capture list, you will see the HTTP requests and responses. Click on a packet to view detailed information.

B) Capturing HTTP GET and POST Requests

Wireshark will show HTTP requests as packets, and you can differentiate between GET and POST requests:

1. **HTTP GET Request**:
    - A GET request retrieves data from the server.

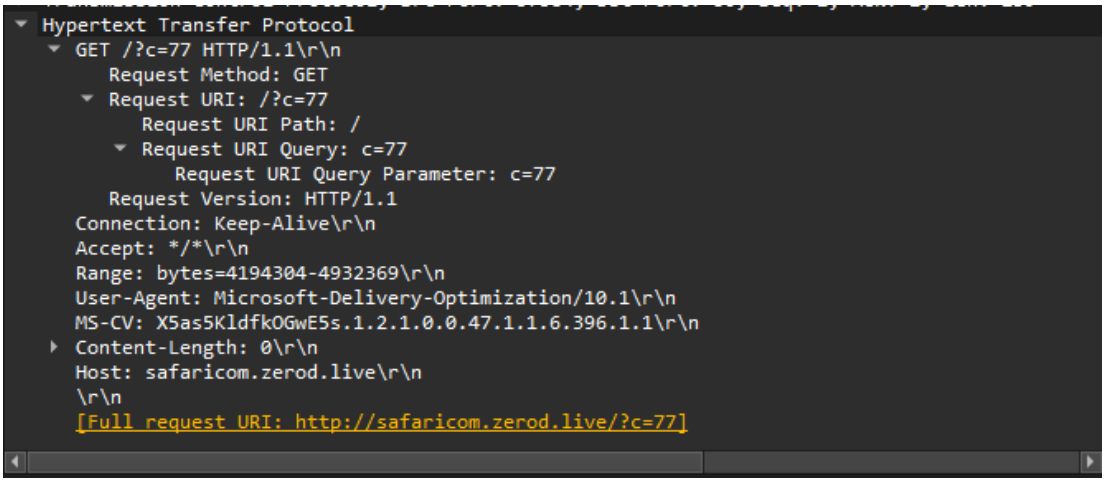


2. **HTTP POST Request**:
    - A POST request sends data to the server (e.g., submitting a form).

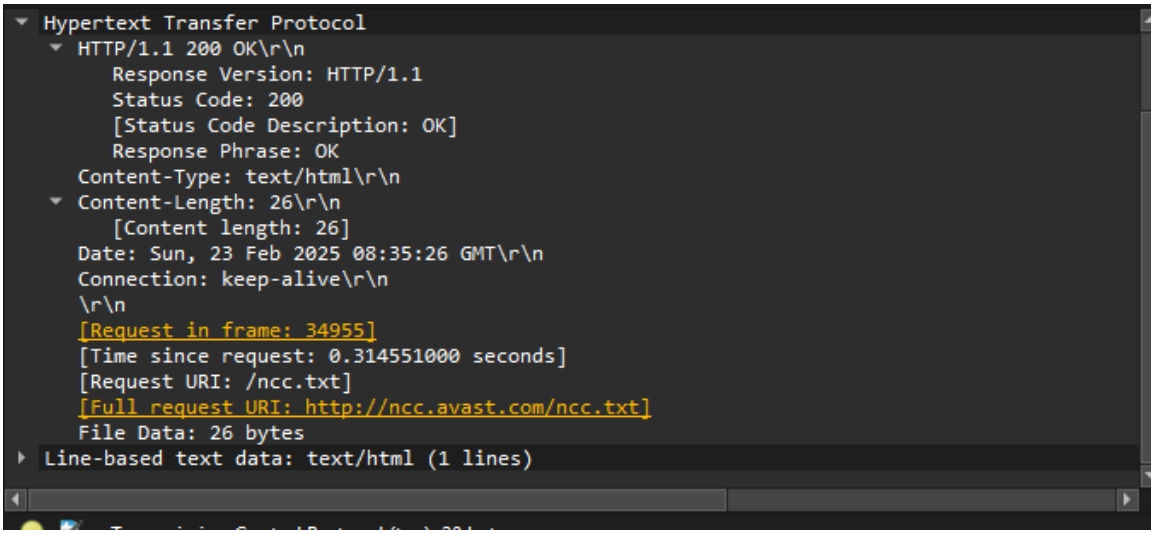
---

c) Identifying Key Information in HTTP Requests and Responses

1. **Request and Response Headers**:
    - **Request Headers**: These are sent by the client (browser or other client) to the server.
        - **Host**: Specifies the domain name.
        - **User-Agent**: Identifies the browser and OS making the request.
        - **Accept**: Indicates the types of content the client is willing to accept (e.g., text/html, application/xml).



- **Response Headers**: These are sent by the server in response to the client's request.
    - **HTTP/1.1 200 OK**: This indicates a successful request.
    - **Content-Type**: Describes the MIME type of the response (e.g., text/html, application/json).
    - **Content-Length**: Specifies the size of the response body.



2. **MIME Type of the Response**:
    - The **MIME type** is found in the **Content-Type** header of the response.

3. **HTTP Status Code and Explanation**:
    - The **HTTP status code** is part of the response header, and it indicates the result of the server's processing of the request.
        - **200**: The request was successful, and the server returned the requested data.

Other common HTTP status codes include:

- **404 Not Found**: The requested resource could not be found on the server.
- **500 Internal Server Error**: The server encountered an error while processing the request.

- **301 Moved Permanently**: The requested resource has been permanently moved to a new URL.