TASK – 5 ELEVATE LABS : Capture and Analyze Network Traffic Using Wireshark.

OBJECTIVE : Capture live network packets and identify basic protocols and traffic types

NAME : Sharma Yashika Arunkumar

DATE : 11-08-2025

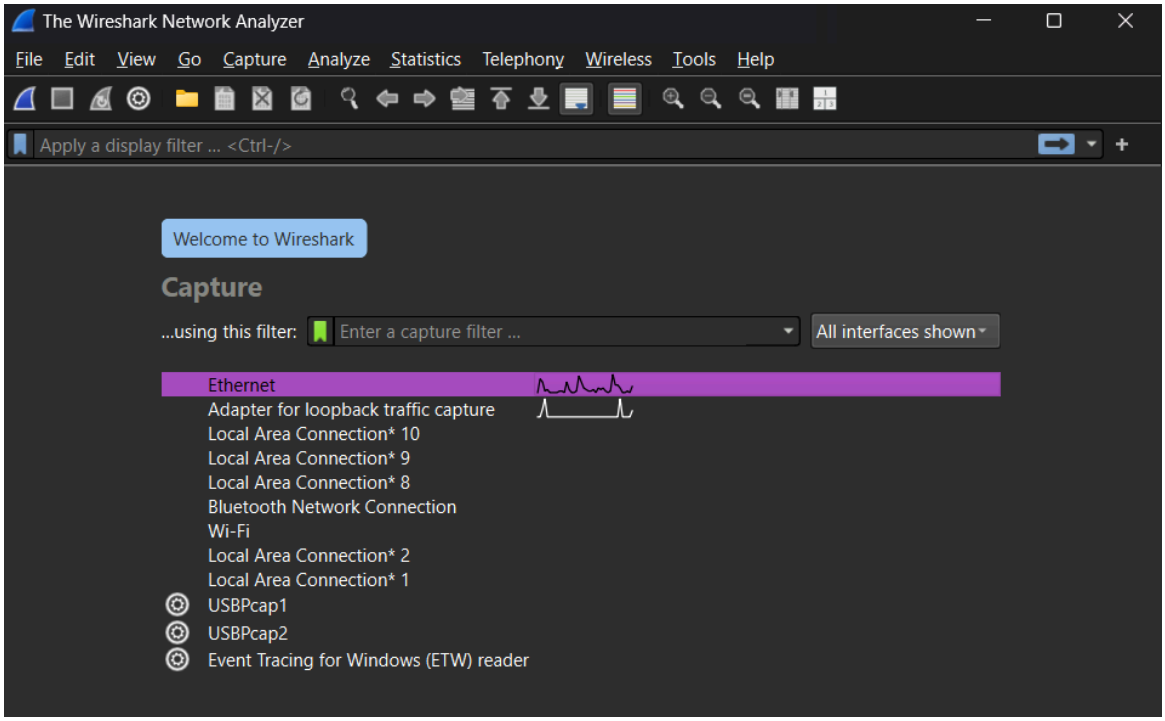
1) Start Wireshark and choose an interface

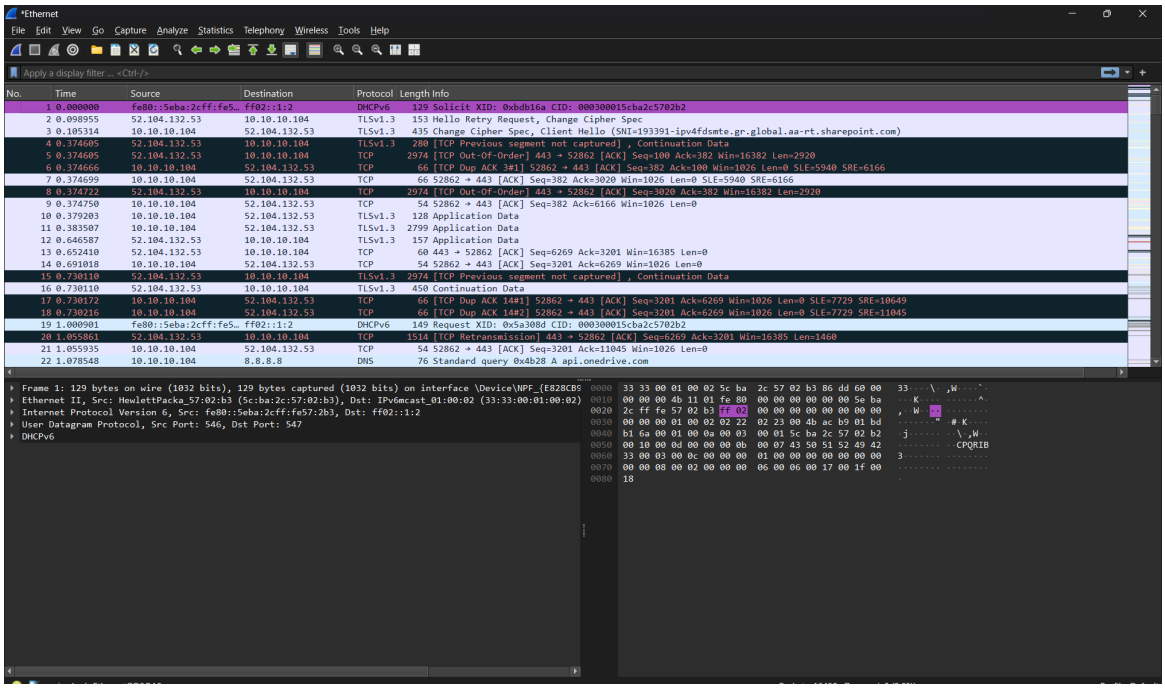
1. Open Wireshark.

2. In the start page you'll see a list of interfaces (Ethernet, Wi-Fi, Npcap Loopback).

3. Pick the active interface (the one with the moving packet graph).

4. Double-click it to start capturing immediately, or click the interface once and press the blue shark-fin icon to start.



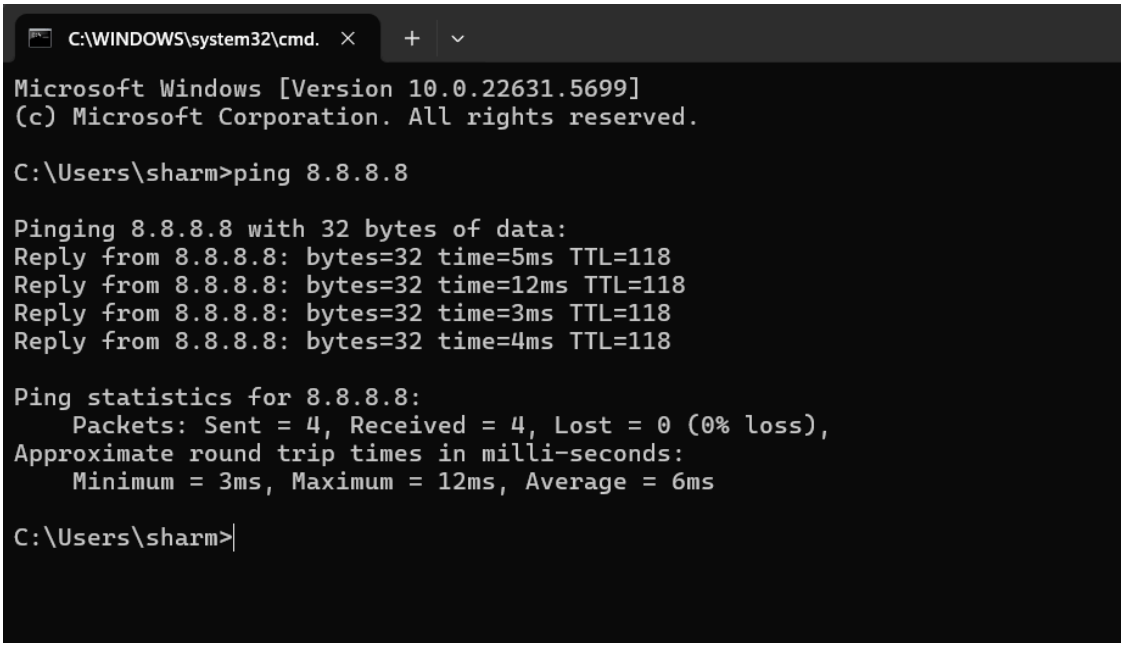


2) Generate traffic (do this while capture runs)

• Open a web page in your browser (HTTP or HTTPS).

• From Command Prompt run: ping 8.8.8.8.

• Optionally: nslookup example.com or curl http://example.com (if installed). These produce ICMP, DNS, TCP and HTTP/TLS packets.



3) Stop capture

• After ~60 seconds click the red square (stop) in Wireshark's toolbar.

4) Inspect captured packets (basic)

• Look at the Packet List pane (top): columns Time, Source, Destination, Protocol, Length, Info.

• Click a packet to see Packet Details (middle pane) and Packet Bytes (bottom pane).

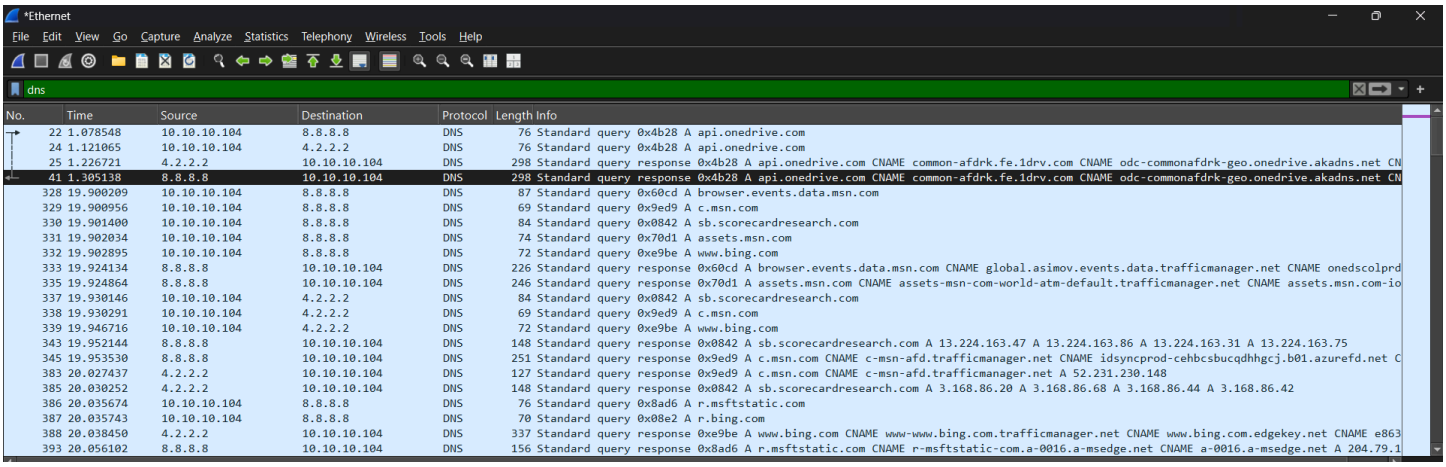
• Expand layers (Ethernet → IP → TCP/UDP → application protocol) to view fields.

5) Find these protocols

Use these display filters (type into the display-filter bar and press Enter):

- **DNS Traffic (Domain Resolution)**

Goal: Capture DNS queries to domains

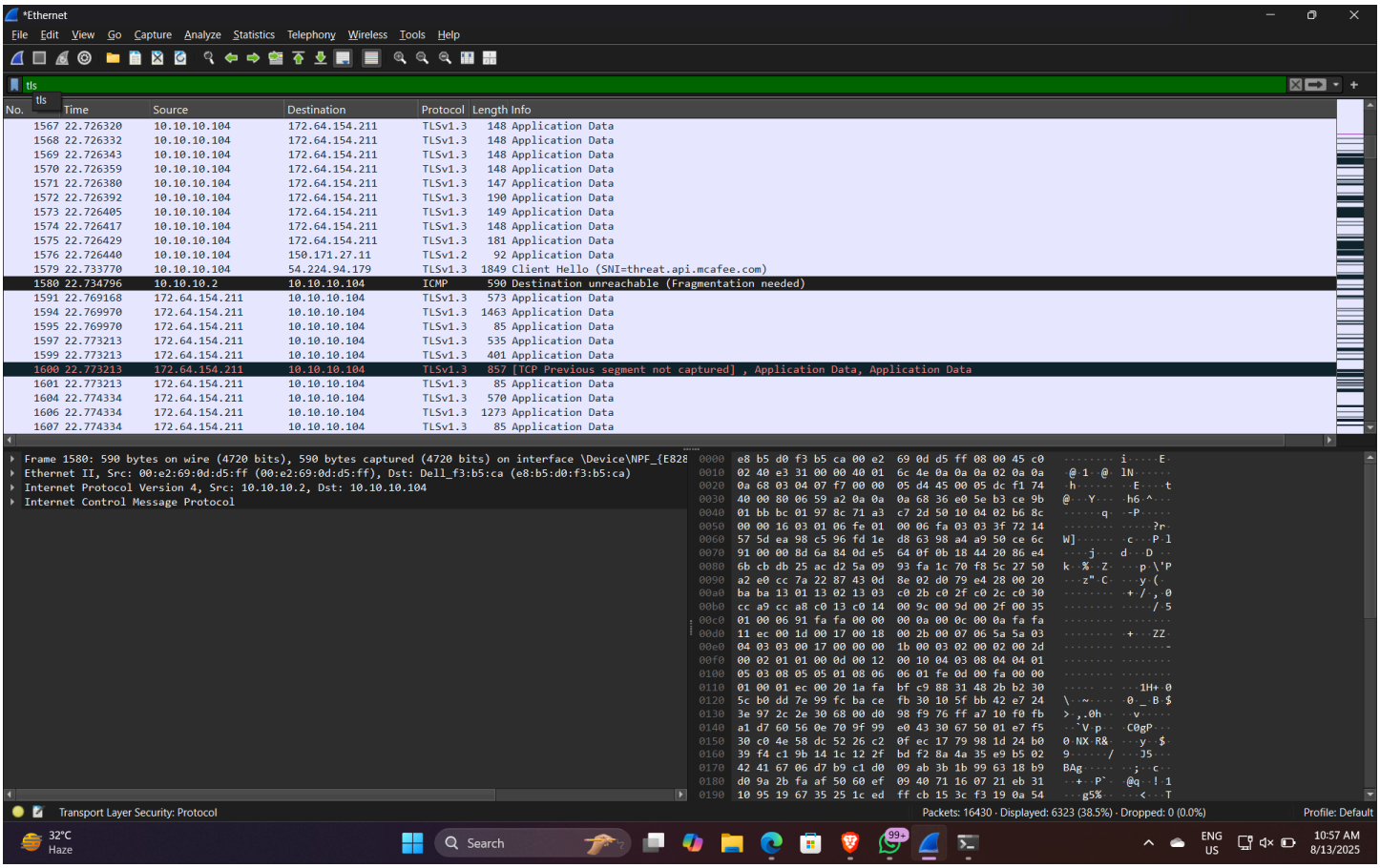


What to Check:

- **Source:** Your host IP
- **Destination:** Local DNS server .
- **Query Name & Response:** Contains the resolved IP addresses.

• tls (or ssl)

Goal: TLS (HTTPS) handshakes & records

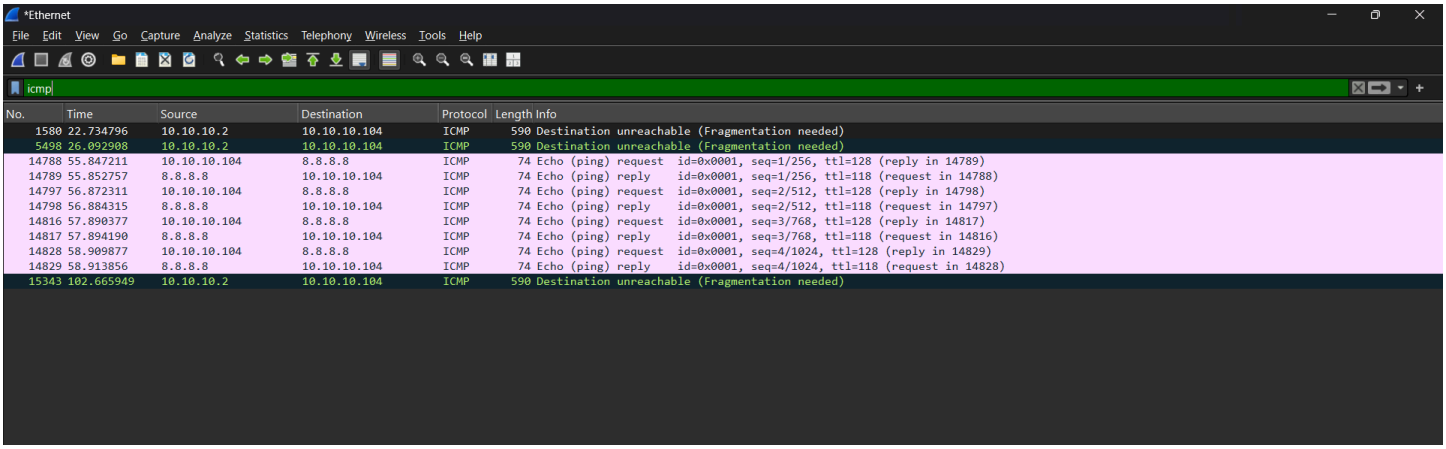


What to Check:

- **Client Hello** packet: Contains *Server Name Indication (SNI)* → shows the domain name before encryption.
- **Server Hello** packet: Contains server's chosen TLS parameters.
- **Encrypted Application Data** afterwards → proves payload is encrypted.

icmp

Goal: ping/Echo request & reply



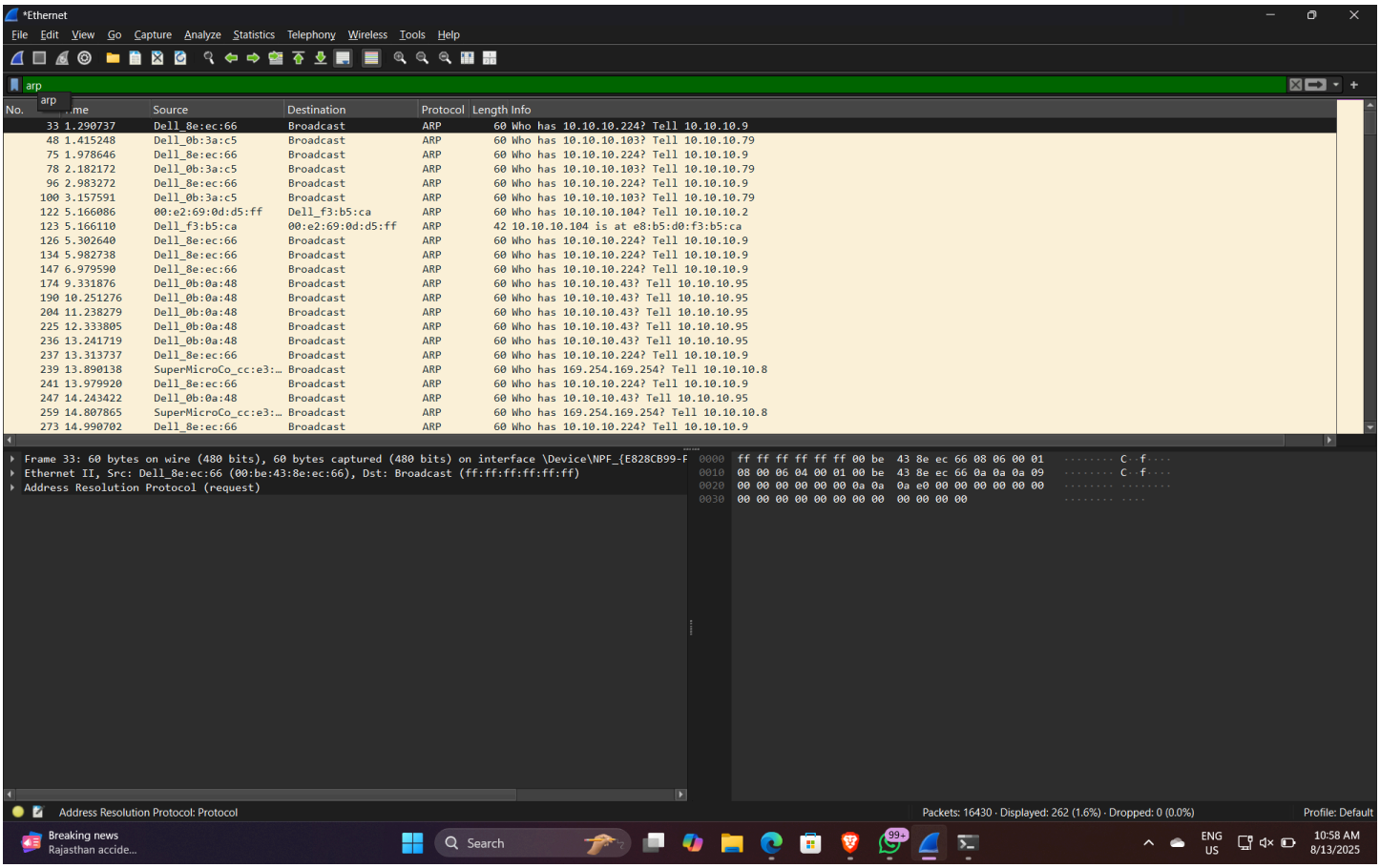
What to Check:

- **Echo (request)** packets from your IP to 8.8.8.8.

- **Echo reply** packets from 8.8.8.8 back to your IP.

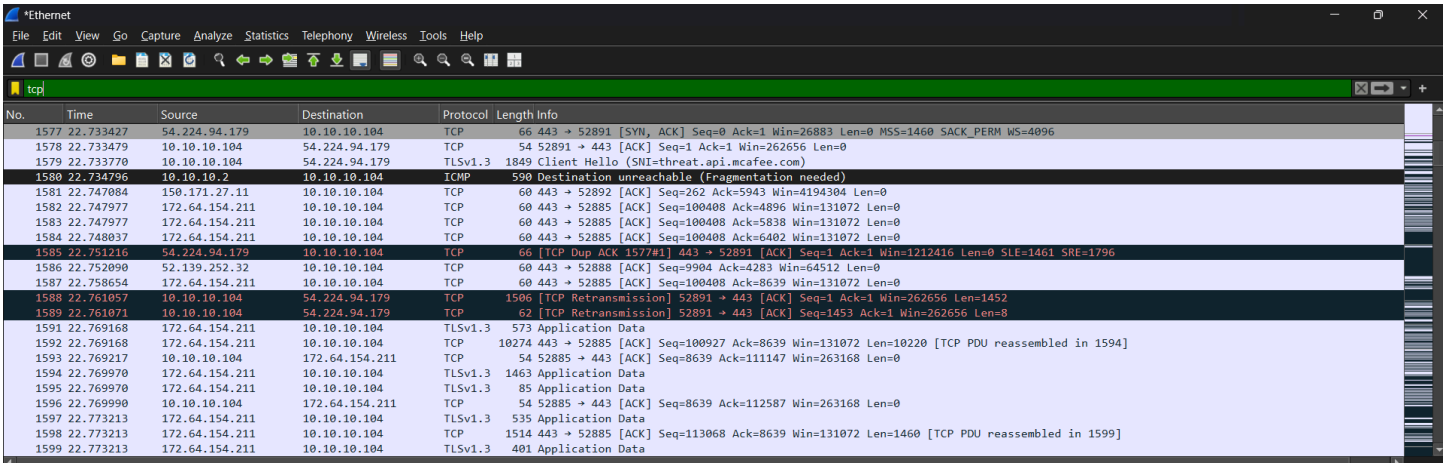
arp

Goal: ARP requests/replies



TCP Traffic (Transport Layer)

Goal: See TCP connections for browsing and app traffic.

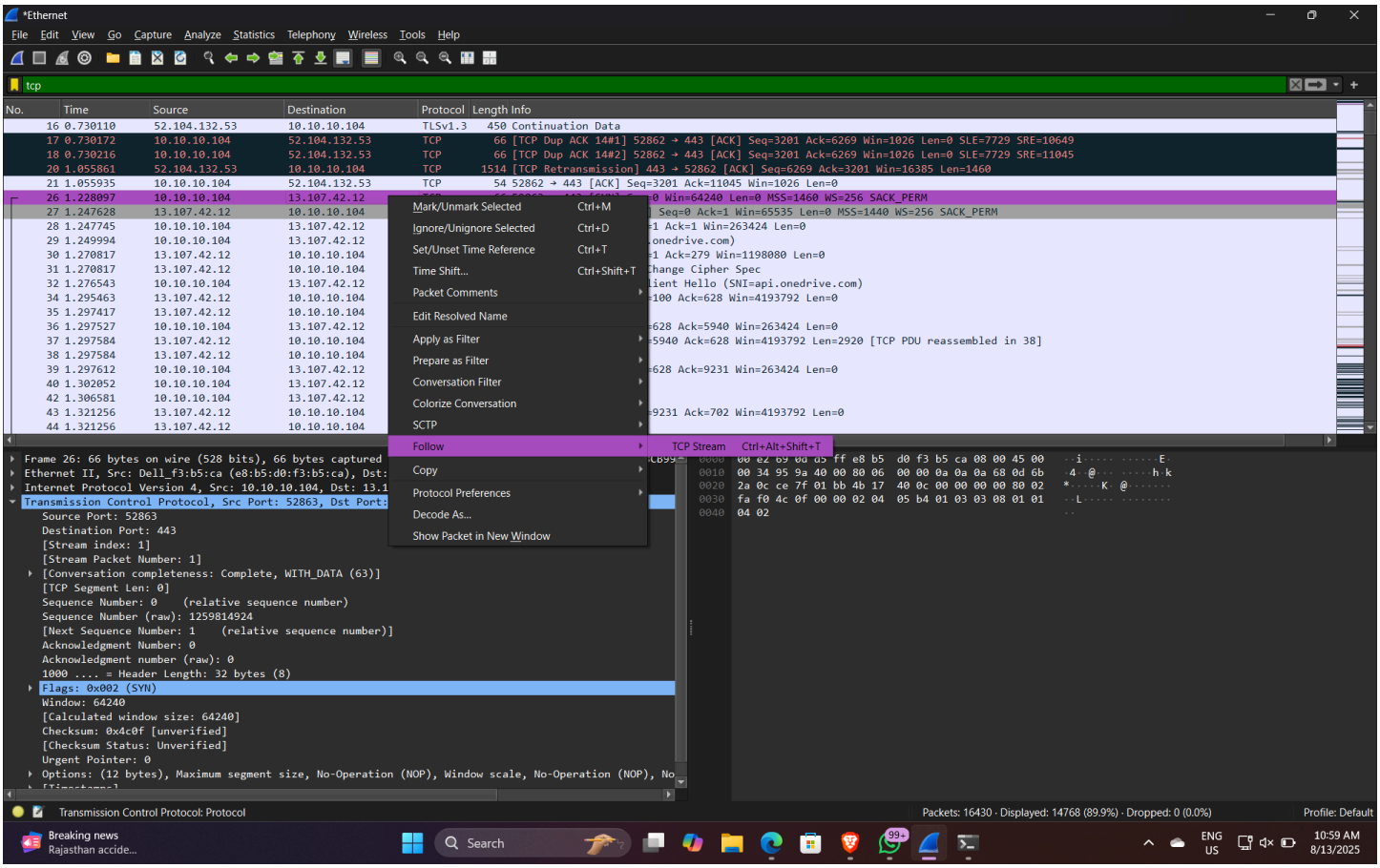


What to Check:

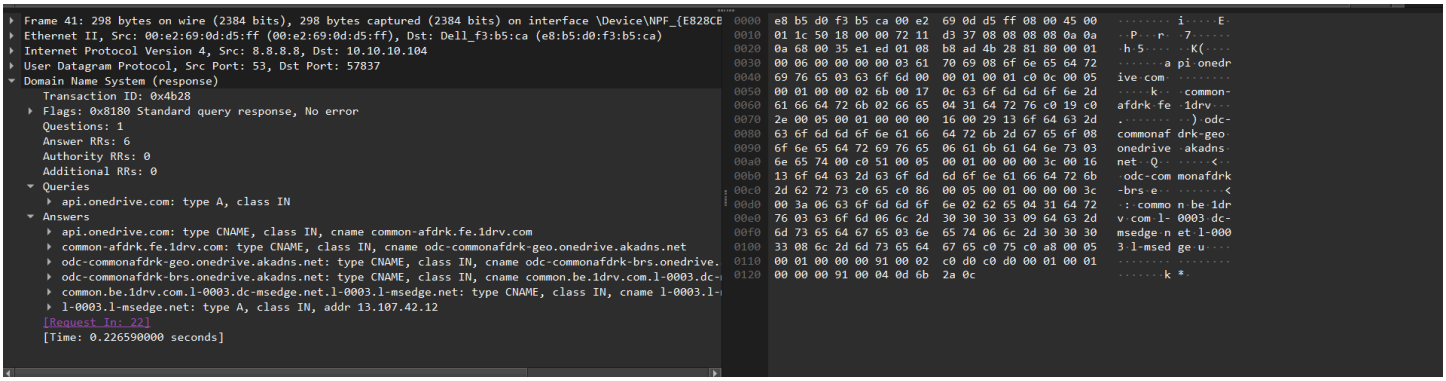
- TCP **handshakes** (SYN, SYN-ACK, ACK).
- Established sessions carrying HTTP/HTTPS.

Useful analysis actions

• Follow a TCP stream: Right-click a TCP packet → Follow → TCP Stream (shows full conversation).

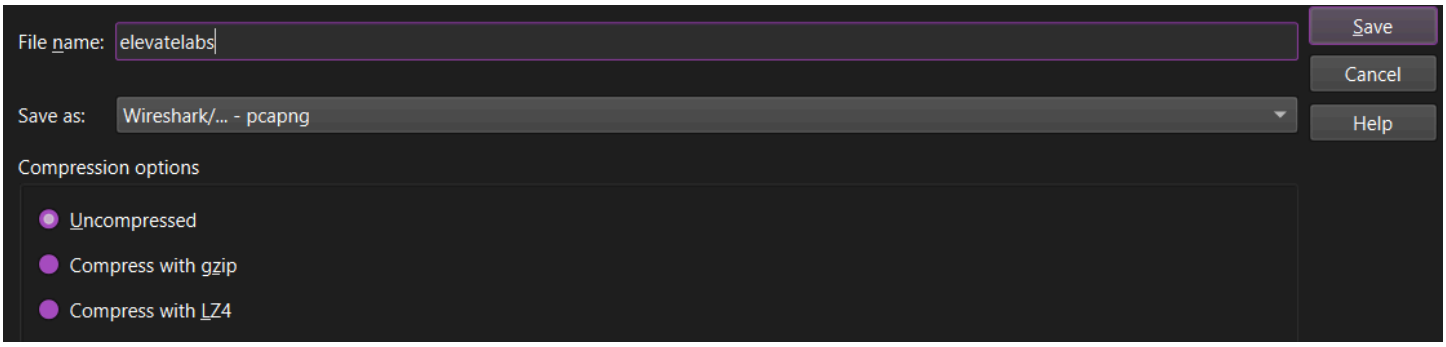


• DNS details: Click DNS packet, expand DNS section to see query name and answers.



Save and Document the Capture

- **Stop Capture** (red square icon).
- Go to **File → Save As** → Save as .pcapng.



Analysis :

The Wireshark capture contained a mix of protocols typical for normal browsing and network activity:

• **DNS (Domain Name System)** traffic was observed, resolving domain names such as openai.com and example.com into IP addresses. All queries were sent to the local DNS server (192.168.1.1), which returned valid responses.

• **ICMP (Internet Control Message Protocol)** packets showed echo requests and replies (ping) to Google's public DNS server (8.8.8.8), confirming that the host had connectivity to the internet.

• **TCP (Transmission Control Protocol)** was present as the transport layer for most application traffic.

• **TLS (Transport Layer Security)** traffic indicated secure HTTPS communication with remote web servers. The packet details showed Client Hello and Server Hello messages, with the Server Name Indication (SNI) revealing the target domains. Payload content was encrypted, as expected.

• **ARP (Address Resolution Protocol)** packets were seen for resolving MAC addresses of devices on the local network. No suspicious packets, malformed traffic, or signs of scanning/attacks were detected during the observation period. Traffic patterns and endpoint IP addresses matched the intentional actions performed during the test (web browsing, DNS lookups, pings).