**Task 5 : Capture and Analyze Network Traffic Using Wireshark.**

1.Install Wireshark.

2.Start capturing on your active network interface.

3.Browse a website or ping a server to generate traffic.

4.Stop capture after a minute.

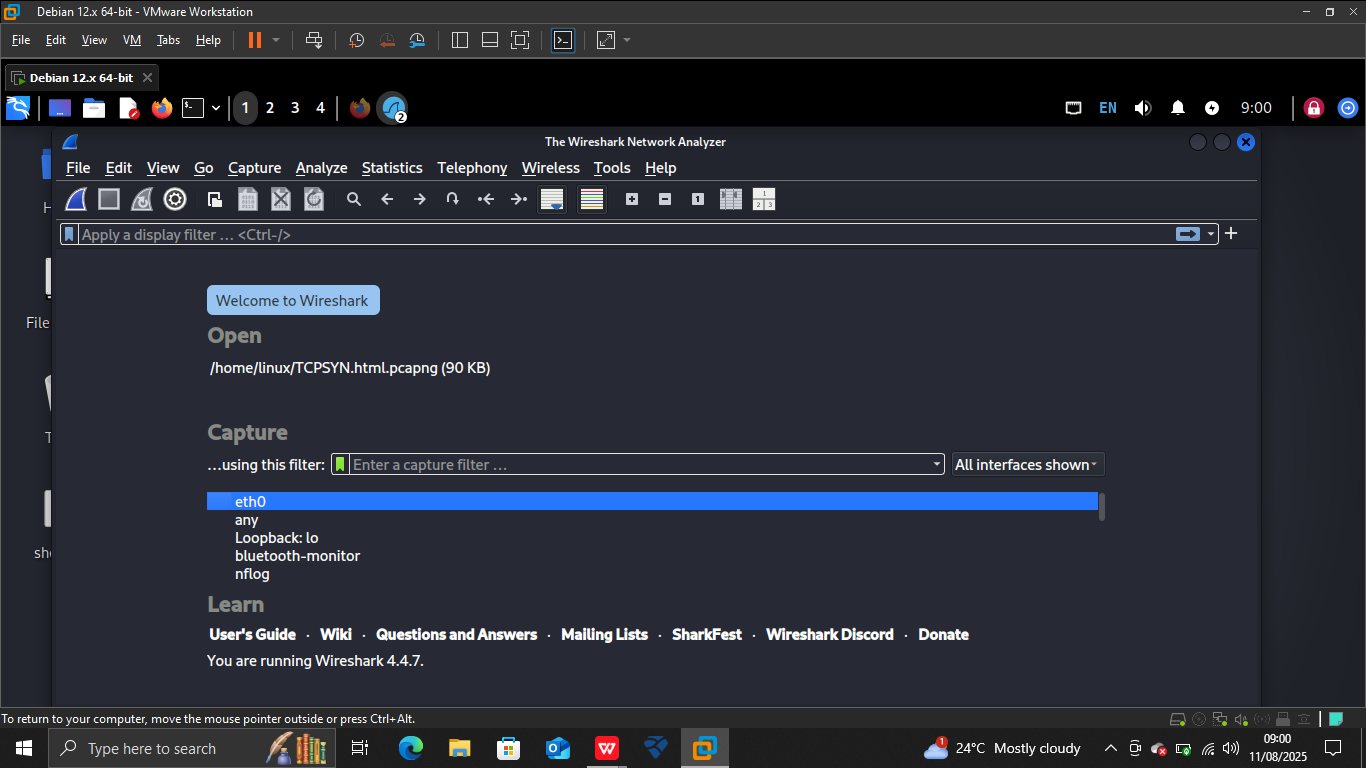
5.Filter captured packets by protocol (e.g., HTTP, DNS, TCP).

6.Identify at least 3 different protocols in the capture.

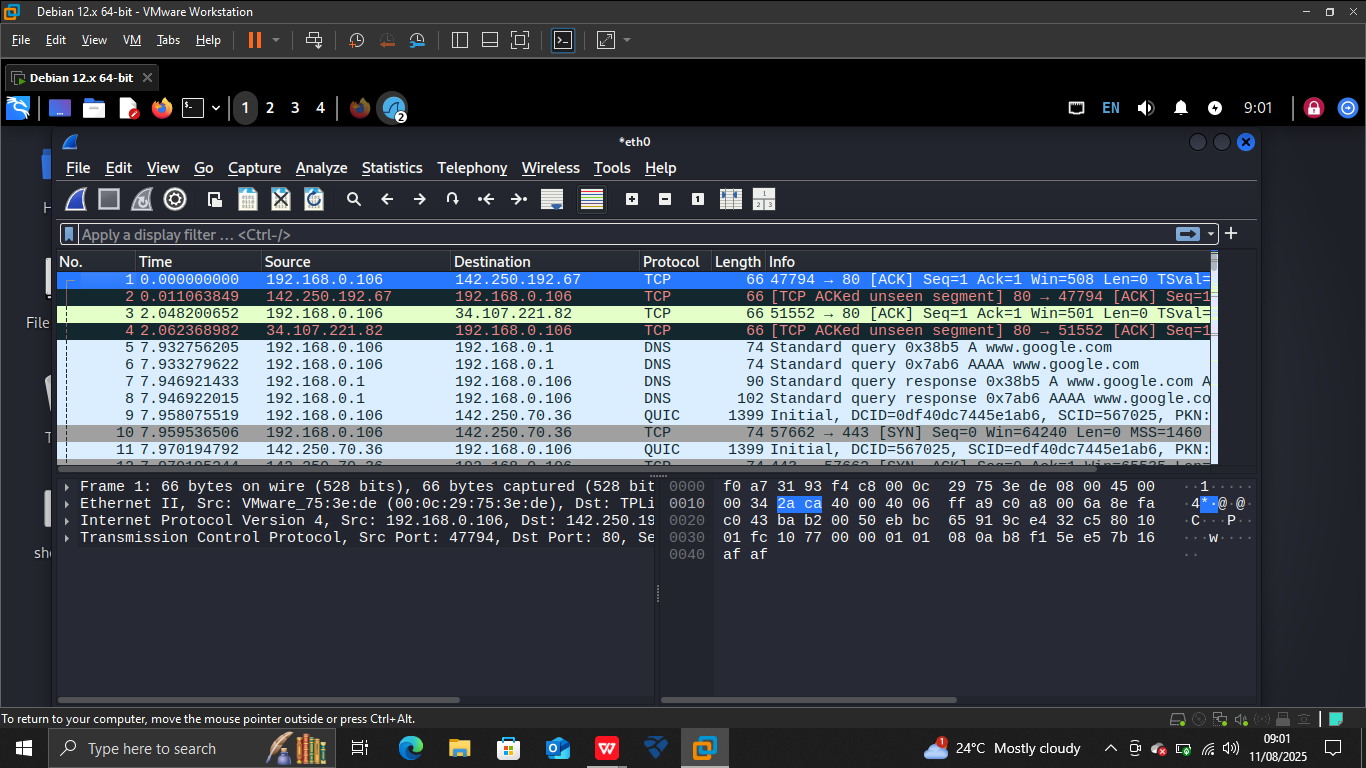
7.Export the capture as a .pcap file.

8.Summarize your findings and packet details.

**1.Install Wireshark.**

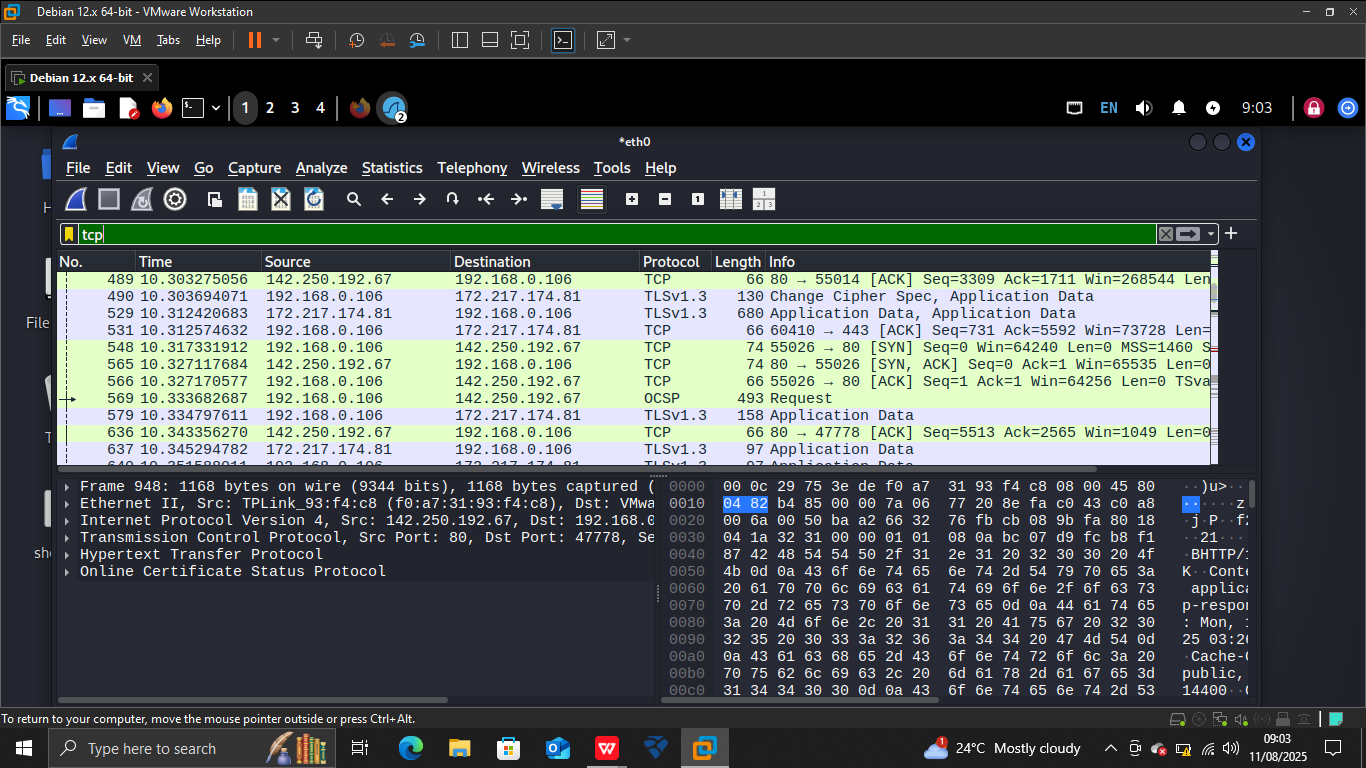


**2.Start capturing on your active network interface.**

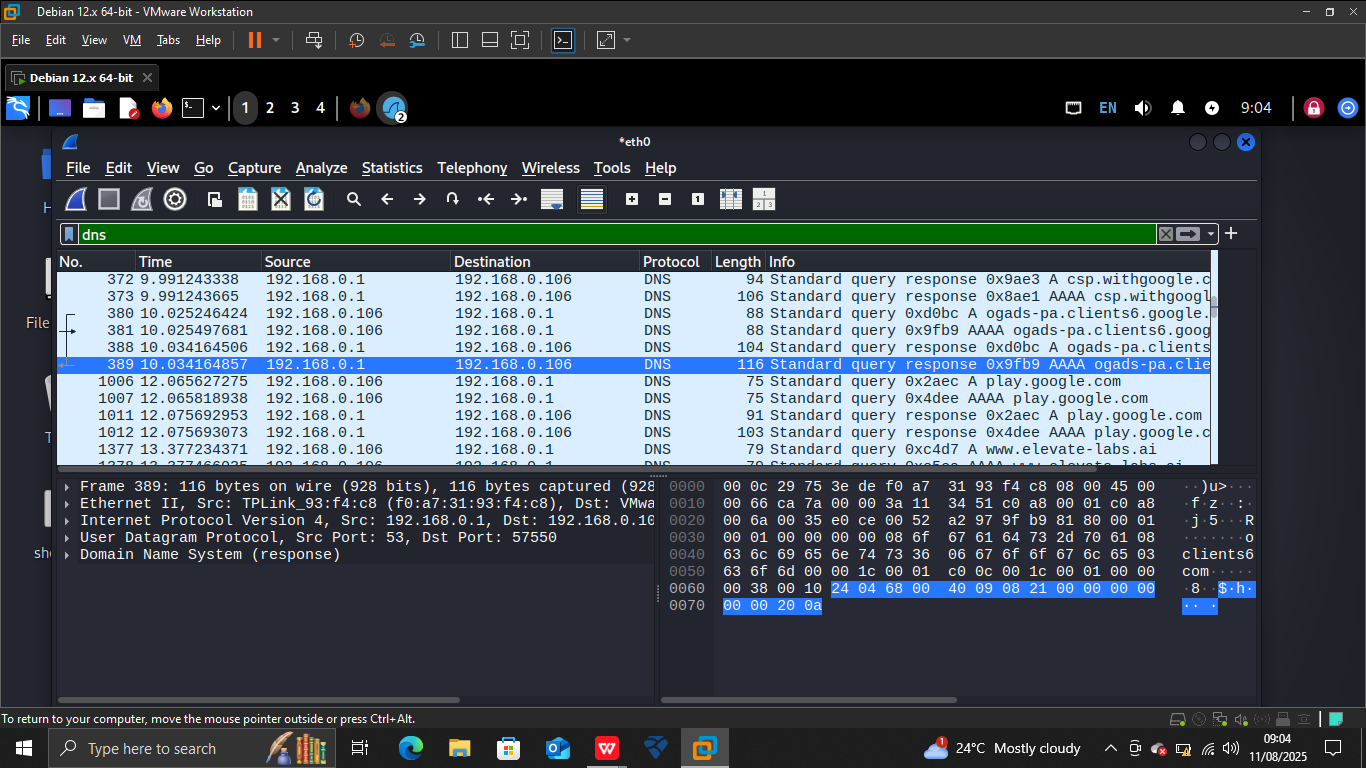


1. **Filter captured packets by protocol (e.g., HTTP, DNS, TCP).**

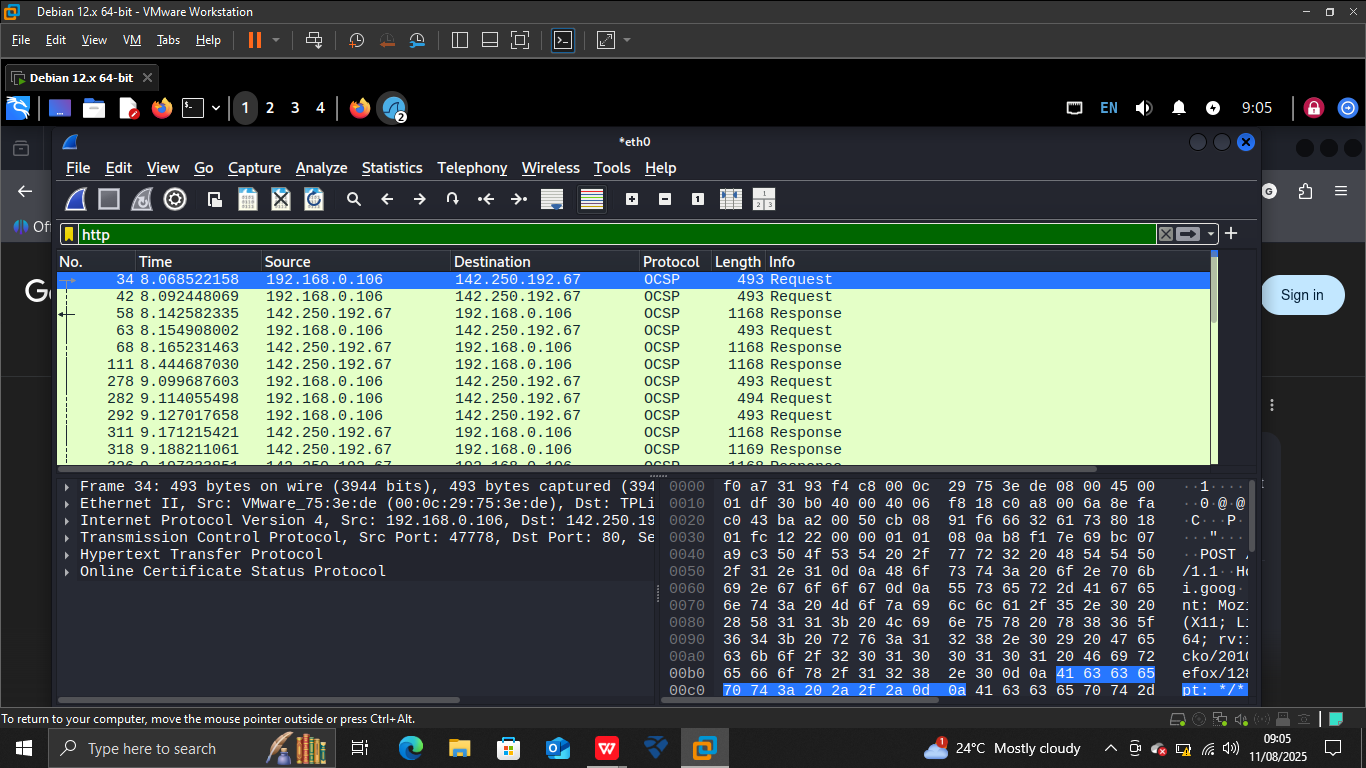
**TCP**

****

**DNS**

****

**HTTP**

****

**6.Identify at least 3 different protocols in the capture.**

**TCP**

**DNS**

**HTTP**

**8.Summarize your findings and packet details**

**Summary of Findings**

Protocols Detected: TCP, DNS, HTTP.

**Traffic Activity:**

TCP: Used for establishing and managing connections between devices. Observed multiple TCP handshake and acknowledgment packets during the session.

DNS: Recorded DNS query and response packets, translating domain names into IP addresses.

HTTP: Detected GET requests and server responses when browsing websites.

**Packet Details**

**TCP Packets**

Source & Destination IPs: Showed connections between the local system and multiple external servers.

Flags: SYN, ACK, and FIN flags observed, indicating session initiation, acknowledgment, and termination.

Port Usage: Commonly port 80 (HTTP) and ephemeral client-side ports.

**DNS Packets**

Queries: Domain names requested during browsing.

Responses: Returned corresponding IP addresses.

Ports: UDP port 53 primarily used.

**HTTP Packets**

Methods: Mainly GET requests for web resources.

Responses: Contained HTML, CSS, and sometimes image data.

Ports: Communication over TCP port 80.