# Task 5 – Wireshark Packet Capture Report

## Github link = https://github.com/posurabari/all-internship-tasks/tree/main/task%205

## Objective:

To capture and analyze network traffic using Wireshark in Kali Linux, applying filters for DNS and TCP traffic.

## Tools Used:

- Kali Linux (VirtualBox)  
- Wireshark

## Steps Performed:

1. Launch Wireshark: Opened Wireshark from the Kali Linux application menu and selected the active network interface (e.g., eth0) to start capturing traffic.
2. Start Packet Capture: Clicked on the blue Start Capturing icon in the toolbar.
3. Generate Network Traffic: Opened a web browser and visited several websites to generate live traffic. Used the ping command in the terminal to send packets to external servers (e.g., ping google.com).
4. Apply Filters: In the Wireshark filter bar, typed 'dns' to display only DNS packets. Then typed 'tcp' to display only TCP packets.
5. Observation: DNS Filter showed queries and responses for domain name lookups. TCP Filter showed connection establishment (SYN, SYN-ACK, ACK) and data transfer packets.
6. Stop Capture: Clicked the red Stop button to end packet capture. Saved the capture file for reference.

## Findings:

- DNS packets showed IP resolutions for visited domains.  
- TCP packets displayed full handshakes and data transmissions.  
- This confirmed live network communication analysis was successful.

## Conclusion:

Wireshark is a powerful tool for network traffic monitoring and packet analysis. Filtering DNS and TCP packets helps in focusing on specific communication types, which is useful for troubleshooting and security analysis.

**Screenshot**





