# Wireshark Packet Analysis – Task 5

## Objective

The objective of this task is to capture live network traffic using Wireshark and identify at least three different network protocols within the captured data.

## Tools Used

• Wireshark (Packet capture & analysis)

• Kali Linux (Operating system)

## Actions Performed

1. Launched Wireshark on Kali Linux and selected the active network interface for packet capture.  
2. Opened a web browser and visited the website elevateapp.com to generate DNS, HTTP, and HTTPS traffic.  
3. Opened a terminal and used the ping command to test connectivity to a remote server.  
4. Allowed the capture to run for approximately one minute to gather enough traffic.  
5. Stopped the capture and applied protocol filters such as dns, http, and icmp in Wireshark.  
6. Analyzed the captured packets to identify the protocols in use.  
7. Saved the capture as network\_capture\_task5.pcap for submission.

## Protocols Observed

1. DNS – The system queried elevateapp.com to resolve its domain name into IP addresses.  
 Example: Standard query 'A elevateapp.com'.  
2. HTTP/HTTPS – Web traffic generated while accessing elevateapp.com.  
 HTTP traffic was visible in plain text; HTTPS traffic was encrypted.  
3. ICMP – Ping packets sent to test connectivity.  
 Example: Echo Request from 192.168.220.8 to 104.26.7.47, and Echo Reply received.

## Findings

• DNS queries were transmitted over UDP to the configured DNS server.  
• HTTP requests and responses were visible in clear text, showing the structure of the data being exchanged.  
• HTTPS traffic was encrypted, meaning the actual content could not be inspected, but metadata such as source/destination IPs and ports were visible.  
• ICMP packets confirmed that the local system (192.168.220.8) was able to communicate with the remote server (104.26.7.47).  
• Most of the traffic observed was TCP-based, except for DNS queries (UDP) and ICMP messages.

## Conclusion

This task provided practical experience in capturing and analyzing live network traffic with Wireshark. By observing multiple protocols such as DNS, HTTP/HTTPS, and ICMP, a deeper understanding of how data is transmitted and structured over the network was achieved. This analysis also highlights the importance of encryption in securing web communications.

## Files Included

• network\_capture\_task5.pcap – Raw packet capture file.  
• README.md – Summary of the task, steps, and findings.  
• Wireshark\_Report.docx – Detailed explained report (this document).