Report

Objective

To capture live network traffic using Wireshark, identify and analyze multiple network protocols (ICMP, HTTP/HTTPS, DNS), and design a custom filter to isolate outgoing web traffic.

This helps understand how data travels between layers and how packet inspection reveals protocol behaviour.

Task1- Packet Capture

Task2- Protocol Identification

Task3- Detailed Packet Analysis

Task4- Custom Filter Creation

Most Active Protocols-

ICMP, HTTP, DNS, TCP

Suspicious or Unusual Traffic-

No abnormal or unauthorized connections detected.

All DNS queries corresponded to legitimate visited domains.

No repetitive failed lookups, broadcast storms, or unknown IPs observed.

The network showed normal, stable communication patterns.

Key Insights About Network Communication-

1. Encryption Dominance:

Most modern websites use HTTPS, meaning payload content is encrypted. Only metadata (like IP addresses and port numbers) remains visible.

2. DNS Exposure:

Even when web content is encrypted, DNS queries still reveal the domain names being visited.

3. Layer Interaction:

Each packet moves through multiple layers. Wireshark helps visualize these interactions in real time.

4. Connection Setup:

Every new TCP connection begins with a three-way handshake (SYN → SYN-ACK → ACK), visible in the capture.

5. Traffic Prioritization:

Browsing activity generates far more packets than ping or DNS, showing how web applications dominate bandwidth usage.

Conclusion-

This experiment successfully demonstrated end-to-end packet analysis using Wireshark.

I learned how to:

Capture live traffic from the correct network interface,
Apply display filters to isolate specific protocol types,
Examine packet headers and understand protocol roles, and
Create customized filters to observe targeted traffic patterns.