Date: 18 /09 /2025

Name : Ghorecha Harsh
Roll No: 106

Lab Practical #10:

Study Packet capture and header analysis by Wireshark (HTTP, ICMP, DNS, TCP, UDP etc.)

Practical Assignment #10:

1. **Explain usage of Wireshark tool.**

1. Usage of Wireshark

- Wireshark is a powerful network protocol analyzer widely used for monitoring, diagnosing, and studying network behavior. It allows users to capture real-time network traffic and break down packets to examine protocol details, headers, and payloads. Common protocols analyzed include:
- **HTTP (HyperText Transfer Protocol)**: Facilitates communication between web browsers and servers.
- **ICMP (Internet Control Message Protocol)**: Primarily used for error reporting and diagnostic tools such as ping.
- **DNS (Domain Name System)**: Translates human-readable domain names into numerical IP addresses.
- **TCP (Transmission Control Protocol)**: Provides reliable, connection-based data transfer.
- **UDP (User Datagram Protocol)**: Enables fast, connectionless communication, often used in streaming and online gaming

Date: 18 /09 /2025

2. Here is steps for working with wireshark :

1. Launch **Wireshark** and choose the appropriate network adapter/interface.
2. Begin packet capture by clicking the shark fin button on the toolbar.
3. Apply filters in the filter bar to view specific traffic types:
    - **http** → to see web traffic
    - **icmp** → to capture ping requests/replies
    - **dns** → to observe domain name lookups
    - **tcp** → for reliable, connection-oriented traffic
    - **udp** → for streaming, gaming, or VoIP packets
4. Generate some activity like opening websites or sending pings, then stop the capture.
5. **Analyze packets**:
    - Look into headers and data sections.
    - Identify source and destination IPs.
    - Check sequence numbers, flags, or query names.

Date: 18 /09 /2025

2. Packet capture and header analysis by Wireshark (HTTP, ICMP, DNS, TCP, UDP etc.)

Packet Analysis Example

➤ HTTP Packet

- Source IP: 192.168.1.5
- Destination IP: 172.217.10.78
- Info: GET / HTTP/1.1

Explanation: This shows a request sent from your system to a website's server.

➤ ICMP Packet

- Type: Echo (ping) request
- Sequence number: 1
- Time to live (TTL): 64

Explanation: This shows a diagnostic request to check if another device is reachable.

➤ DNS Packet

- Query: www.google.com
- Response: 172.217.10.78

Explanation: Converts a hostname to an IP address.

➤ TCP Packet

- Flags: SYN, ACK
- Sequence number: 12345

Explanation: Initiates and manages a connection between two devices.

➤ UDP Packet

- Source Port: 12345
- Destination Port: 53

Explanation: Used for quick, connectionless communications like DNS queries.

Date: 18 /09 /2025

- **Packet capture** is the process of intercepting and logging traffic that passes through a network. It involves examining the data packets that are sent and received across a network to diagnose issues, monitor performance, or analyze security threats.
- **Wireshark** is one of the most widely used network protocol analyzers for packet capture and inspection. It allows users to:
- Capture live data from network interfaces.
- Save and open captured packet files.
- Analyze the content of each packet.

2. How Does Wireshark Work?

1. **Network Interface Selection**
   You select the network interface (e.g., Ethernet, Wi-Fi) to capture traffic from.
2. **Capturing Packets**
   Wireshark listens to packets flowing through that interface and records them.
3. **Filtering Packets**
   Filters allow focusing on specific types of packets or traffic patterns.
4. **Header Analysis**
   Each packet is dissected into layers such as Ethernet, IP, TCP/UDP, and Application (HTTP, DNS, etc.).
5. **Exporting or Saving**
   Captured data can be saved and analyzed later.

Network Protocols and Their Header Analysis

A. HTTP (Hypertext Transfer Protocol)

Used for web communication.

Key Fields in HTTP Header:

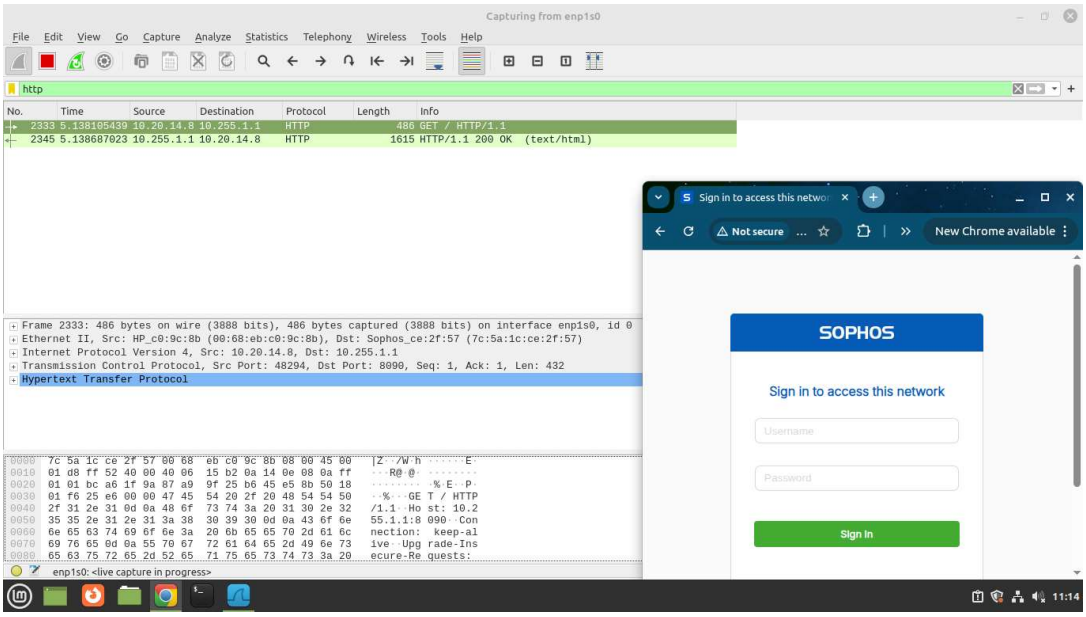
- **Request Line:** Method (GET, POST), URI, version.
- **Host:** Domain name.
- **User-Agent:** Client information.

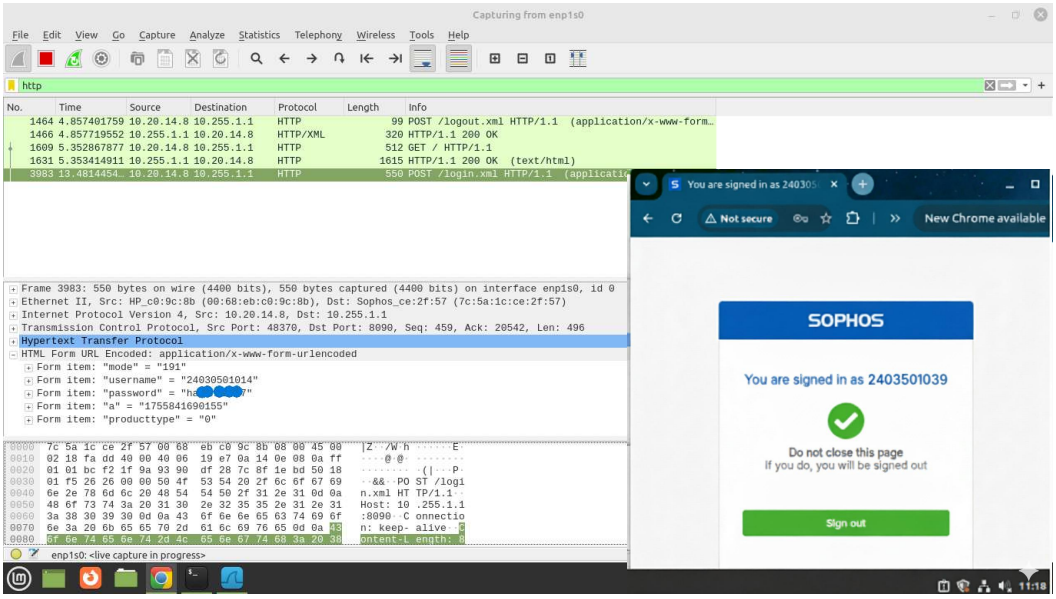
Date: 18 /09 /2025

- **Accept, Content-Type:** Types of supported content.
- **Status Code (in response):** 200 OK, 404 Not Found.

Wireshark Usage:

- Filter: http
- Helps inspect requests, responses, cookies, and headers.
- Good for debugging web applications or finding slow requests.





Date: 18 /09 /2025

B. ICMP (Internet Control Message Protocol)

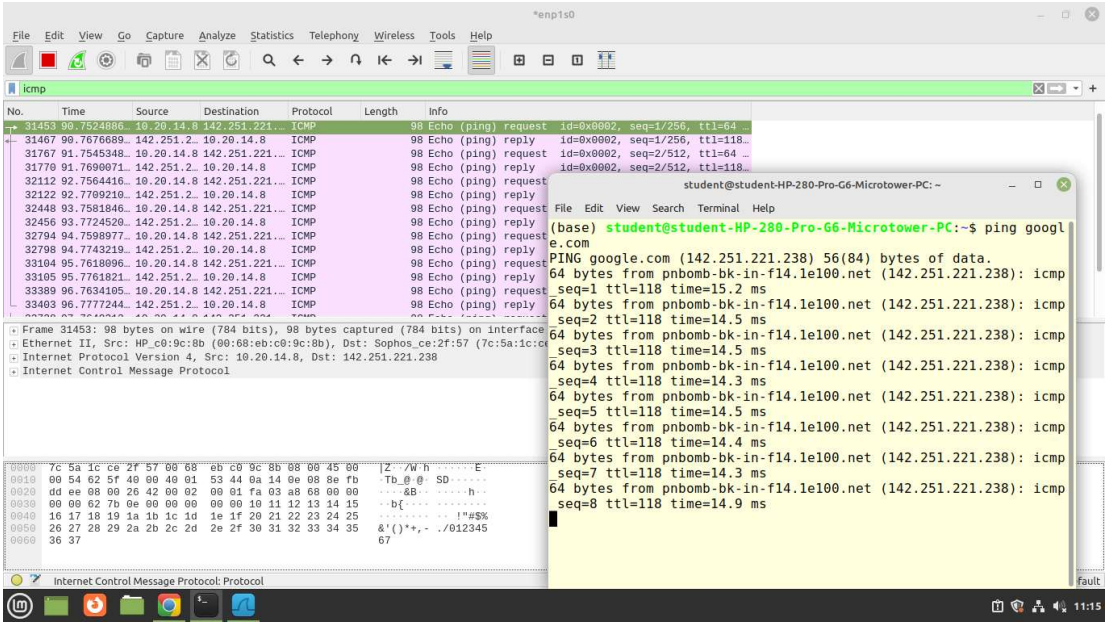
Used for network diagnostics like ping and traceroute.

Key Fields:

- **Type & Code:** Defines message purpose (Echo request/reply).
- **Checksum:** Error-checking value.
- **Identifier & Sequence Number:** Helps track multiple requests.

Wireshark Usage:

- Filter: icmp
- Used to diagnose network reachability and latency.



C. DNS (Domain Name System)

Used to resolve domain names to IP addresses.

Key Fields:

- **Transaction ID:** Matches requests and responses.
- **Flags:** Query/response, authoritative answer.
- **Questions:** Domain name being queried.
- **Answers:** IP address returned.

Date: 18 /09 /2025

Wireshark Usage:

- Filter: dns
- Analyze lookup times, spoofed replies, or failed queries.

D. TCP (Transmission Control Protocol)

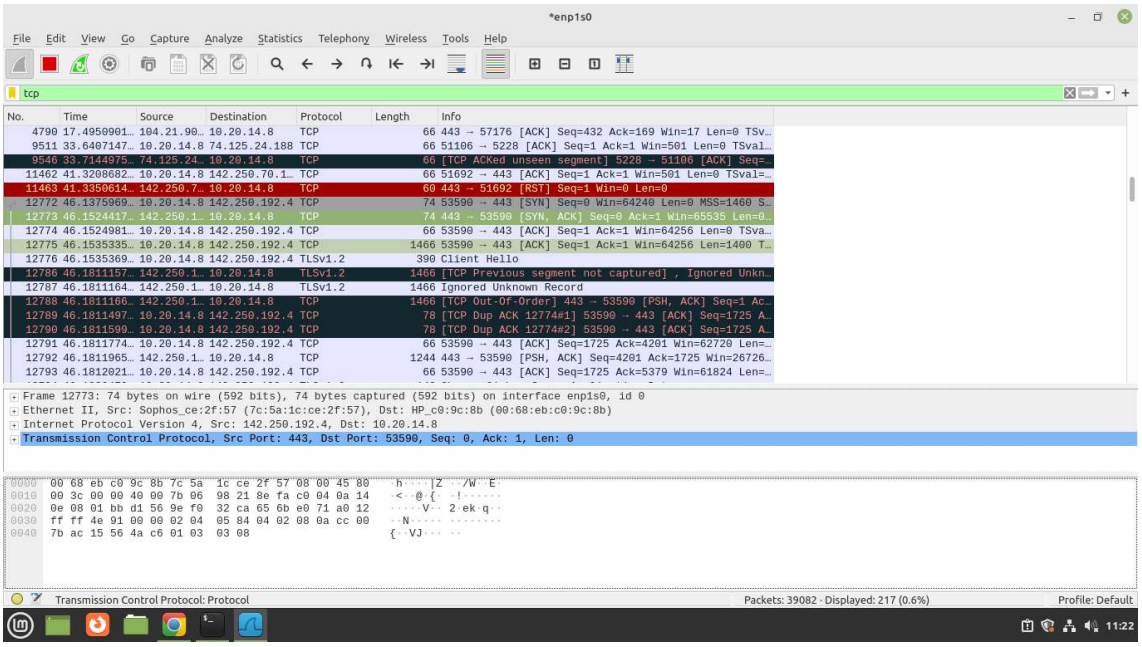
Provides reliable, connection-oriented communication.

Key Fields:

- **Source & Destination Ports:** Identifies service endpoints.
- **Sequence Number & Acknowledgment:** Ensures proper delivery.
- **Flags:** SYN, ACK, FIN, RST (connection setup/teardown).
- **Window Size:** Flow control.
- **Checksum:** Integrity verification.

Wireshark Usage:

- Filter: tcp
- Used to diagnose retransmissions, connection failures, or packet drops.



Date: 18 /09 /2025

E. UDP (User Datagram Protocol)

Used for faster, connectionless communication.

Key Fields:

- **Source & Destination Ports:** Endpoint identifiers.
- **Length:** Size of payload.
- **Checksum:** Integrity check.

Wireshark Usage:

- Filter: udp
- Good for analyzing services like DNS, VoIP, or streaming traffic.