Date: 22/08/2025

Lab Practical #09:

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

Practical Assignment #09:

1. **Explain usage of Wireshark tool.**
2. **Packet capture and header analysis by Wireshark (HTTP, TCP, UDP etc.)**

Wireshark is a powerful network protocol analyser used for capturing and analysing the data packets transmitted over a network. It allows users to see what's happening on their network at a microscopic level, making it a valuable tool for network troubleshooting, security analysis, and software development.

Key Features and Uses:

- **Packet Capture:** Wireshark captures data packets that are transferred over a network. It can capture traffic on different interfaces like Ethernet, Wi-Fi, and more.
- **Protocol Analysis:** Wireshark supports deep inspection of hundreds of protocols, including TCP, UDP, HTTP, DNS, and more. It can dissect the protocol layers and display them in an understandable format.
- **Real-Time Analysis:** Wireshark can analyse network traffic in real-time or from saved capture files.
- **Filtering:** Wireshark provides powerful filtering capabilities to isolate specific traffic. You can use display filters to view only the packets that match certain criteria.
- **Packet Colouring:** Different packets are color-coded based on protocol type or other rules, making it easier to identify specific types of traffic at a glance.
- **Expert Information:** Wireshark includes an "Expert Info" feature that highlights potential problems in the network traffic, such as retransmissions, out-of-order packets, or other anomalies.

Common Uses of Wireshark:

- **Network Troubleshooting:** Identify issues with network performance or connectivity by analysing the captured traffic.
- **Security Analysis:** Detect potential security threats, such as suspicious traffic patterns or unauthorized data transmissions.

Date: 22/08/2025

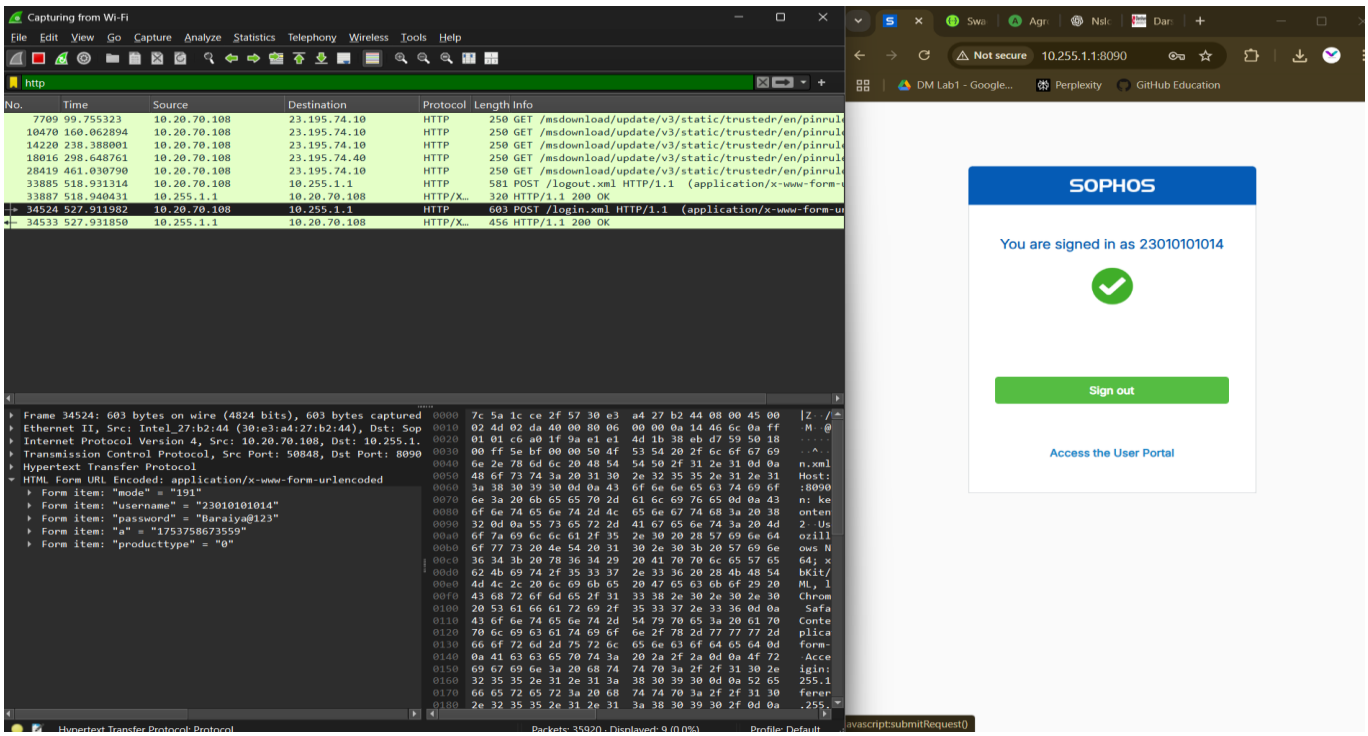
- **Learning Tool:** Wireshark is a great educational tool for understanding how protocols work and how data travels across a network.
- **Development and Testing:** Network and software developers use Wireshark to ensure that their applications are communicating correctly over the network.

Steps for Packet Capture and Analysis:

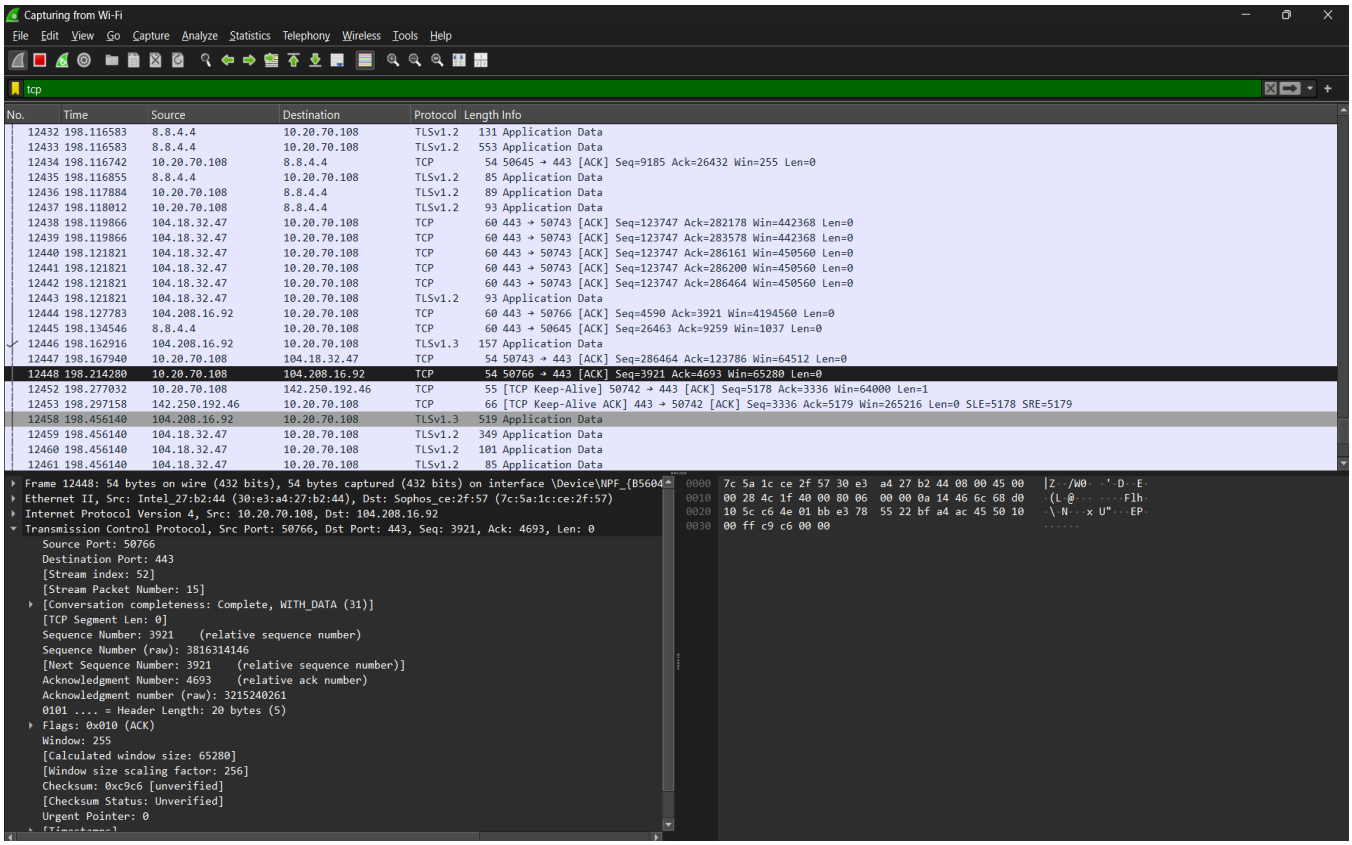
1. **Open Wireshark:**
   - Launch Wireshark on your computer.
   - Select the network interface you want to capture traffic on (e.g., Ethernet, Wi-Fi).
2. **Start Capturing Packets:**
   - Click on the "Start Capturing Packets" button (the shark fin icon) to begin capturing network traffic.
3. **Generate Traffic:**
   - To capture specific types of traffic, you might want to perform actions like browsing a website (for HTTP), sending a ping (for ICMP), or using a specific application.
4. **Stop Capturing:**
   - After sufficient traffic has been captured, click on the "Stop" button (the red square icon) to stop capturing.
5. **Filtering Packets:**
   - Use display filters to narrow down the captured packets to specific protocols. For example:

Date: 22/08/2025

- **HTTP Traffic:** Filter with http

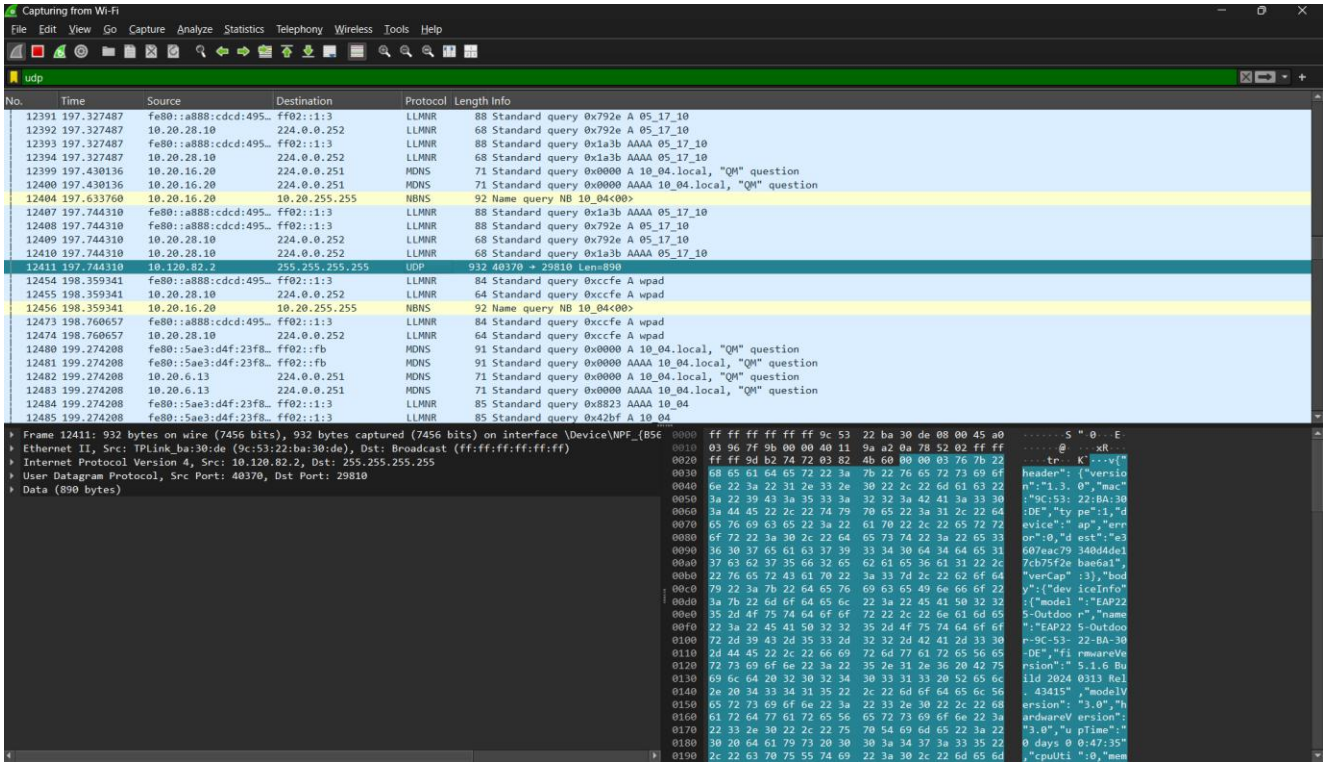


- **TCP Traffic:** Filter with TCP

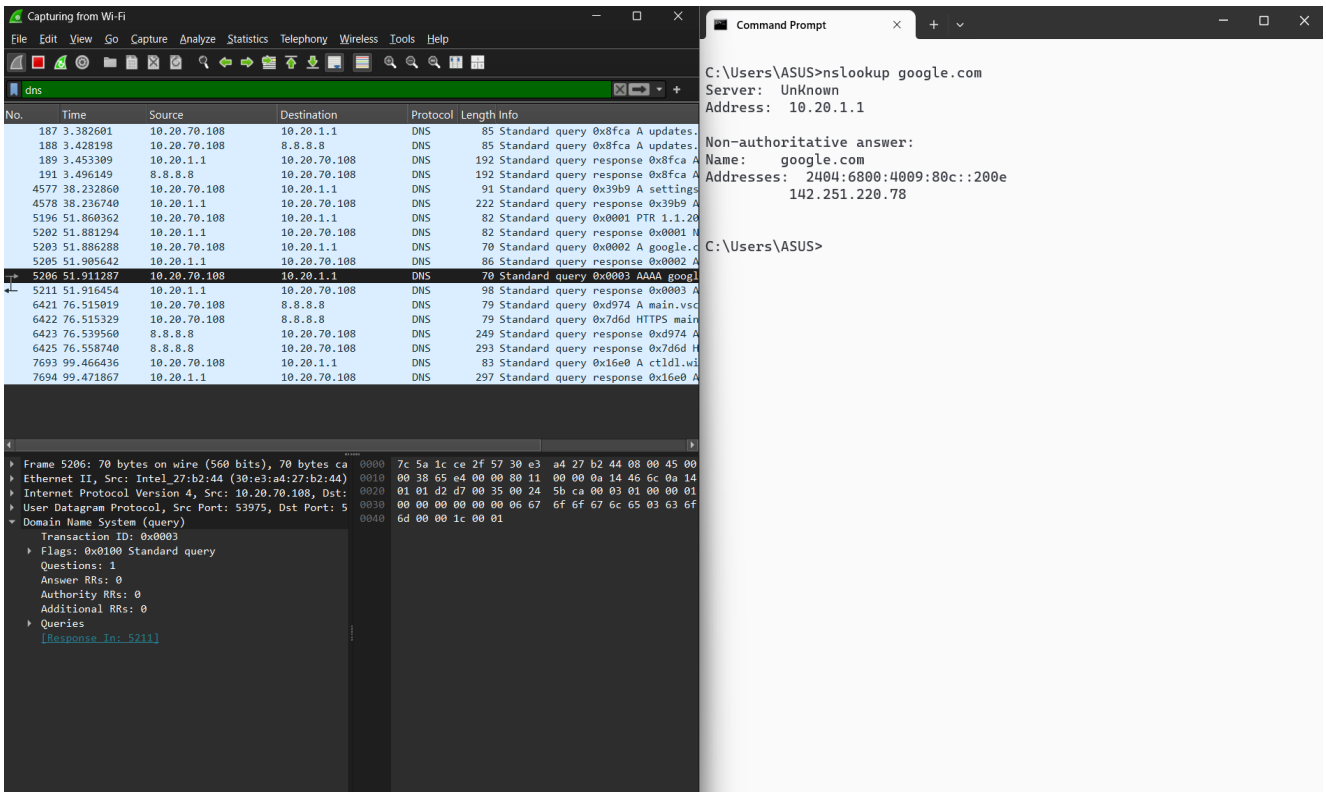


Date: 22/08/2025

- **UDP Traffic:** Filter with UDP

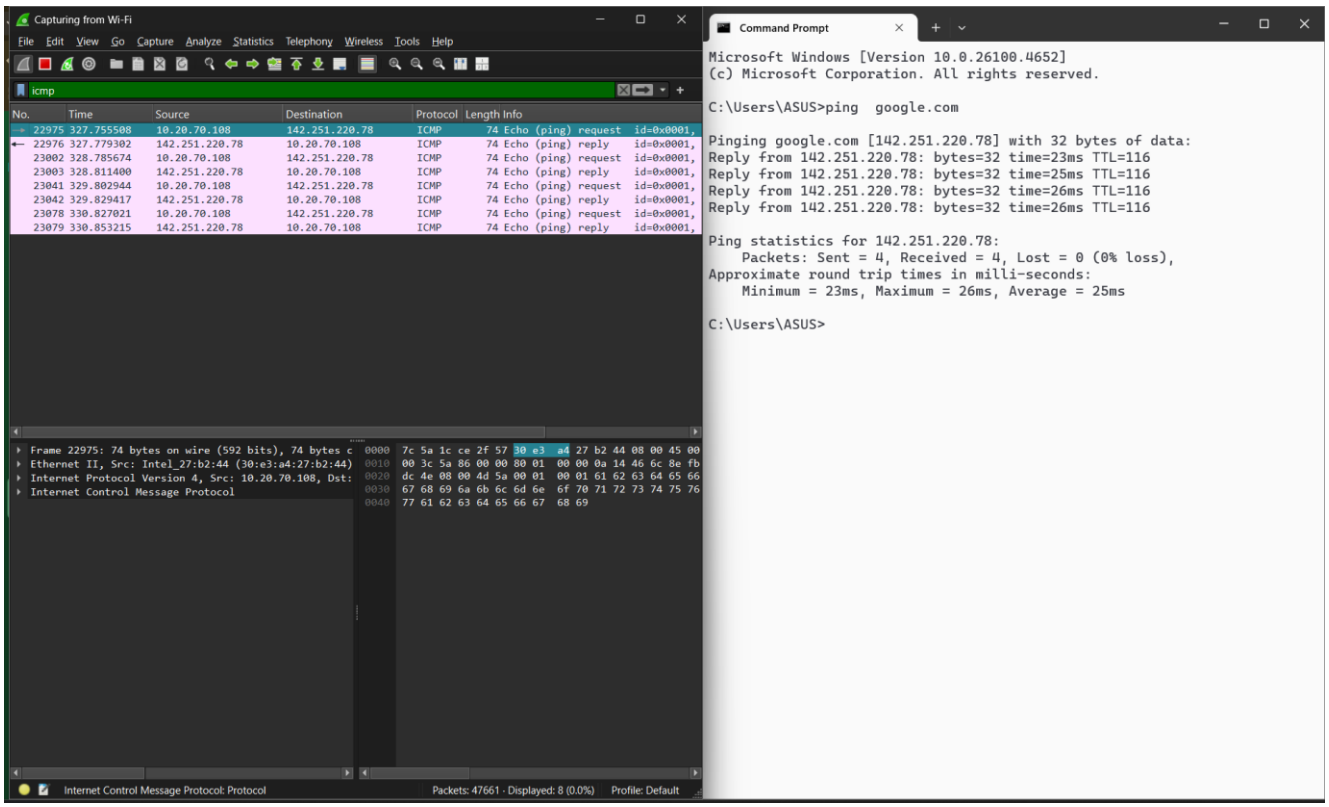


- **DNS Traffic:** Filter with DNS



Date: 22/08/2025

- **ICMP Traffic:** Filter with DNS



6. **Analyse Packet Headers:**
   - Click on a packet in the capture window to view its detailed information.
   - **TCP Header:** Analyse Source Port, Destination Port, Sequence Number, Acknowledgment Number, Flags, Window Size, etc.
   - **UDP Header:** Examine Source Port, Destination Port, Length, and Checksum.
   - **HTTP Header:** Review HTTP requests and responses, including methods (GET, POST), status codes, headers like Host, User-Agent, and more.
7. **Interpretation:**
   - Wireshark presents the data in three panes: a list of packets, detailed information for the selected packet, and a hexadecimal representation of the packet. Use these panes to drill down into the specific details of each packet.