Experiment no.: 08

Title: Filter packets according to protocol using any packet filtering tool.
Course Outcome: Build systems that are more secure against attacks.
Theory:

Packet filtering is a fundamental concept in **computer networking and network security**. It involves monitoring and controlling network data packets based on predefined rules or protocols such as **TCP, UDP, ICMP, or HTTP.** Each packet contains header information, including source and destination addresses, ports, and the communication protocol used.

When filtering packets according to protocol, a **firewall or network analyzer** inspects each packet and allows or blocks it depending on the specified protocol type. For example, a rule may allow only **TCP packets** and block **UDP or ICMP packets.** This process helps in managing network traffic, enhancing security, and preventing unauthorized access or attacks.

Tool used: Wireshark

Wireshark is a popular open-source **network protocol analyzer** used to capture and examine data packets traveling across a network. It helps in understanding how network protocols work, identifying network problems, and ensuring secure communication.

By using **packet filtering** in Wireshark, users can view packets of a **specific protocol** (like TCP, UDP, ICMP, or HTTP), which makes analysis easier and more focused.

Concept

Every data packet transmitted over a network contains header information, which includes the **protocol type** used for communication.
Wireshark allows users to apply **display filters** to show only packets matching a specific protocol or condition.

For example:

- tcp → Displays only TCP packets
- udp → Displays only UDP packets
- icmp → Displays only ping (Internet Control Message Protocol) packets
- http → Displays only web traffic packets

This helps in analyzing how specific protocols behave and interact within the network.

Applications

1. **Network Troubleshooting** – Helps identify network issues like delay or packet loss.
2. **Security Monitoring** – Detects suspicious or unauthorized network activities.
3. **Performance Analysis** – Analyzes traffic patterns and data flow efficiency.
4. **Educational Use** – Demonstrates how protocols like TCP, UDP, and HTTP work.

Advantages

1. **Real-Time Monitoring** – Captures live network traffic instantly.
2. **Detailed Packet Information** – Displays in-depth protocol-level data.
3. **Protocol-Specific Filtering** – Simplifies analysis using filter expressions.
4. **Free and Open Source** – Available for all major operating systems.

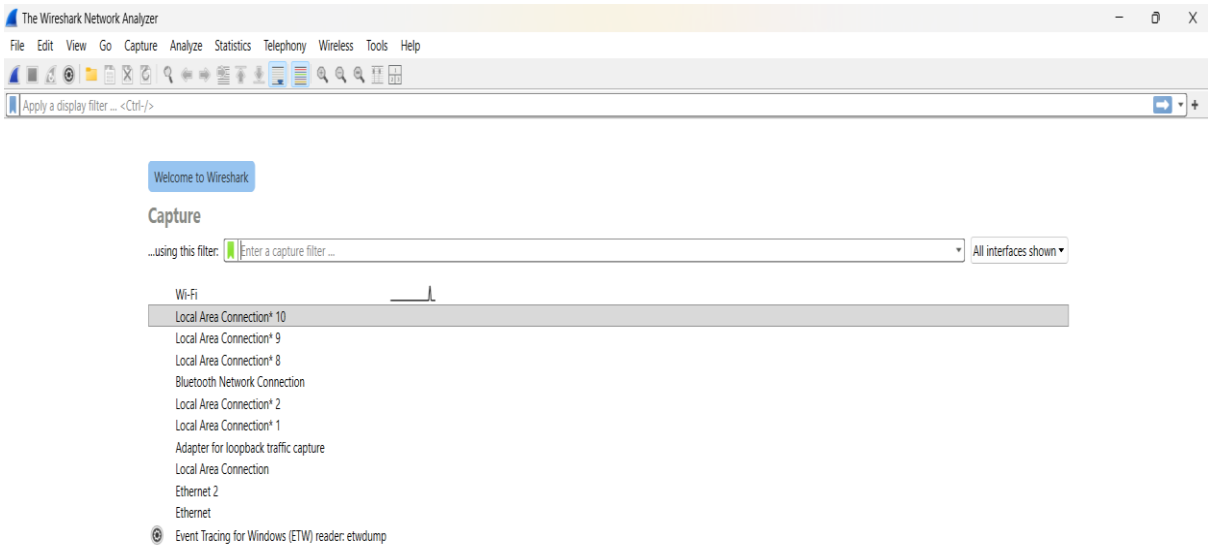
Limitations

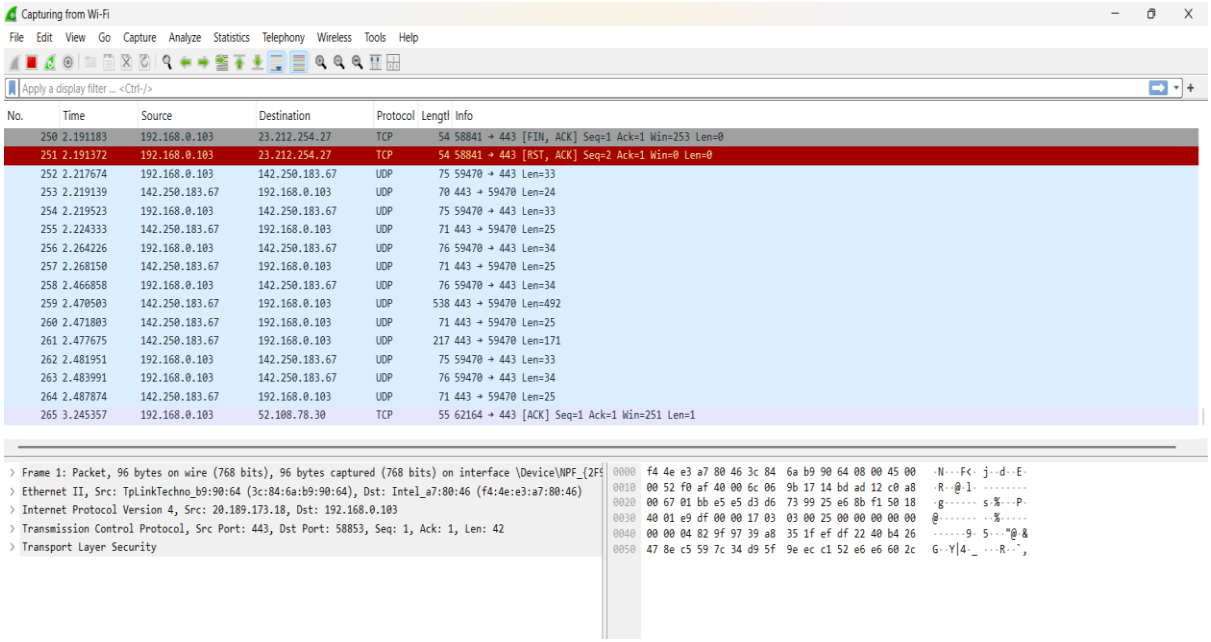
1. **Complex for Beginners** – Requires understanding of network protocols.
2. **High Data Volume** – Capturing large traffic can slow down the system.
3. **Limited Access on Encrypted Traffic** – Cannot read encrypted HTTPS data.
4. **Administrative Privileges Required** – Needs elevated permissions to capture packets.

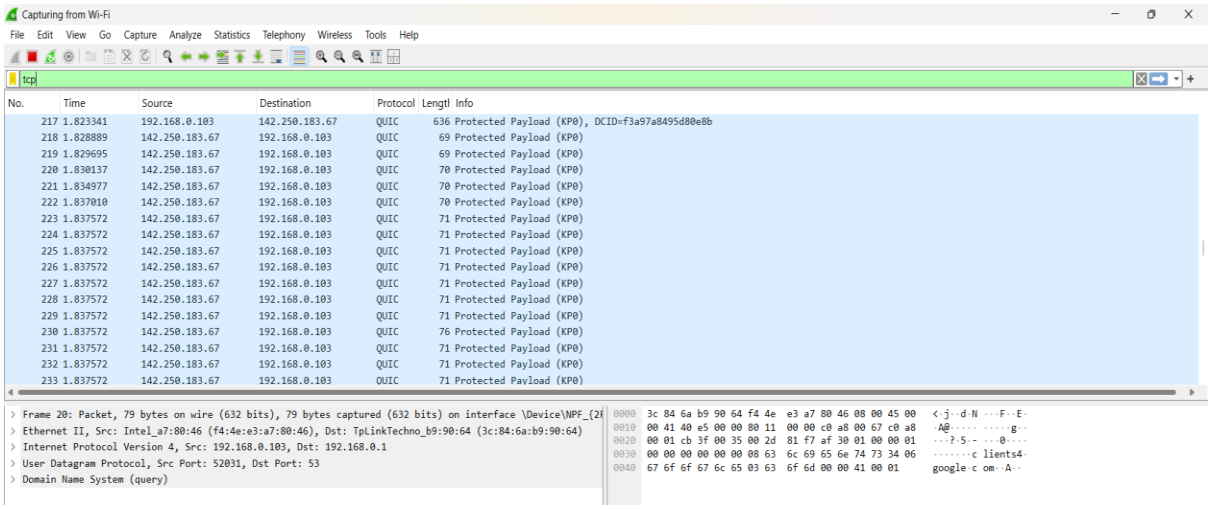
Steps:

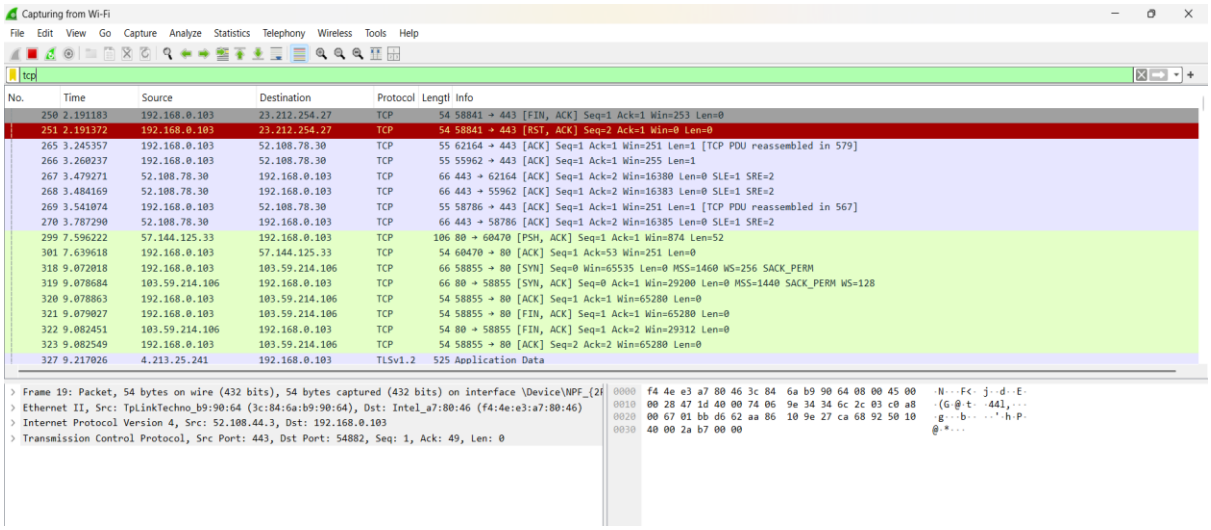
1. **Install Wireshark** – Download and install Wireshark along with WinPcap/Npcap.
2. **Open Wireshark** – Launch the application to access the main interface.
3. **Select Network Interface** – Choose the network (Wi-Fi or Ethernet) to capture packets.
4. **Start Packet Capture** – Begin capturing live network traffic.
5. **Apply Protocol Filter** – Enter the desired protocol (e.g., TCP, UDP, ICMP) in the filter bar.

6. **Analyze the Packets** – View details like source, destination, ports, and data of filtered packets.
7. **Stop the Capture** – End the packet capture after collecting sufficient data.
8. **Save or Export Results** – Save the captured or filtered packets for future analysis.









Conclusion:

I have successfully learned packet filtering using Wireshark.