

Practical – 2

Aim : To experiment on Packet capture tool: Wireshark.

About Wireshark:

What is Wireshark?

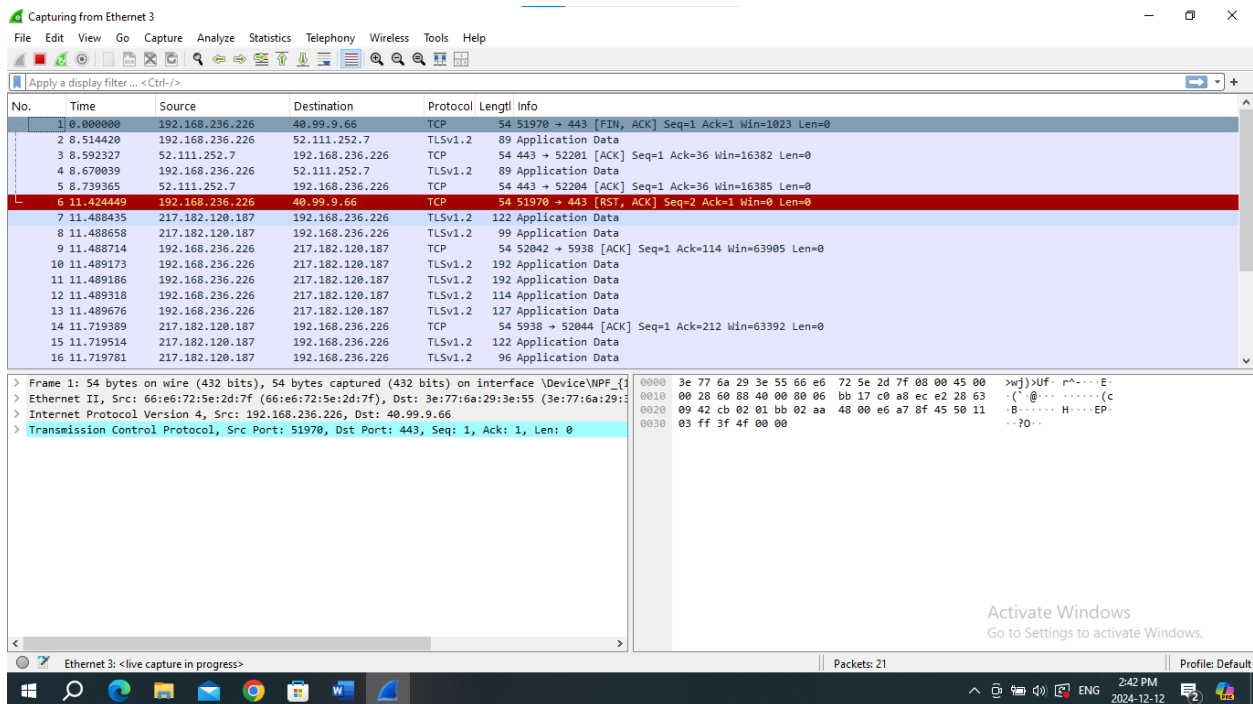
Wireshark is a free and open-source network packet analyzer used to monitor and analyze network traffic in real time. It captures data packets traveling across a network and displays them in a detailed and readable format, helping users troubleshoot network issues, monitor performance, and detect security vulnerabilities. Wireshark supports multiple protocols, making it useful for analyzing different types of networks, including Ethernet, Wi-Fi, and TCP/IP. It provides filtering, visualization, and export features for deeper packet inspection. Widely used by network administrators, security experts, and developers, Wireshark is a powerful tool for understanding network behavior.

Uses of Wireshark:

1. **Network Troubleshooting:** Helps identify and resolve network issues by analyzing packet-level details.
2. **Performance Monitoring:** Monitors network performance, bandwidth usage, and traffic patterns.
3. **Protocol Analysis:** Examines how different protocols (e.g., TCP, UDP, HTTP) operate and interact on the network.
4. **Security Analysis:** Detects suspicious activities, potential attacks, or vulnerabilities in network traffic.
5. **Packet Filtering:** Filters specific packets based on criteria like IP address, port, or protocol for focused analysis.
6. **Learning and Development:** Used for educational purposes to understand networking concepts and protocols.

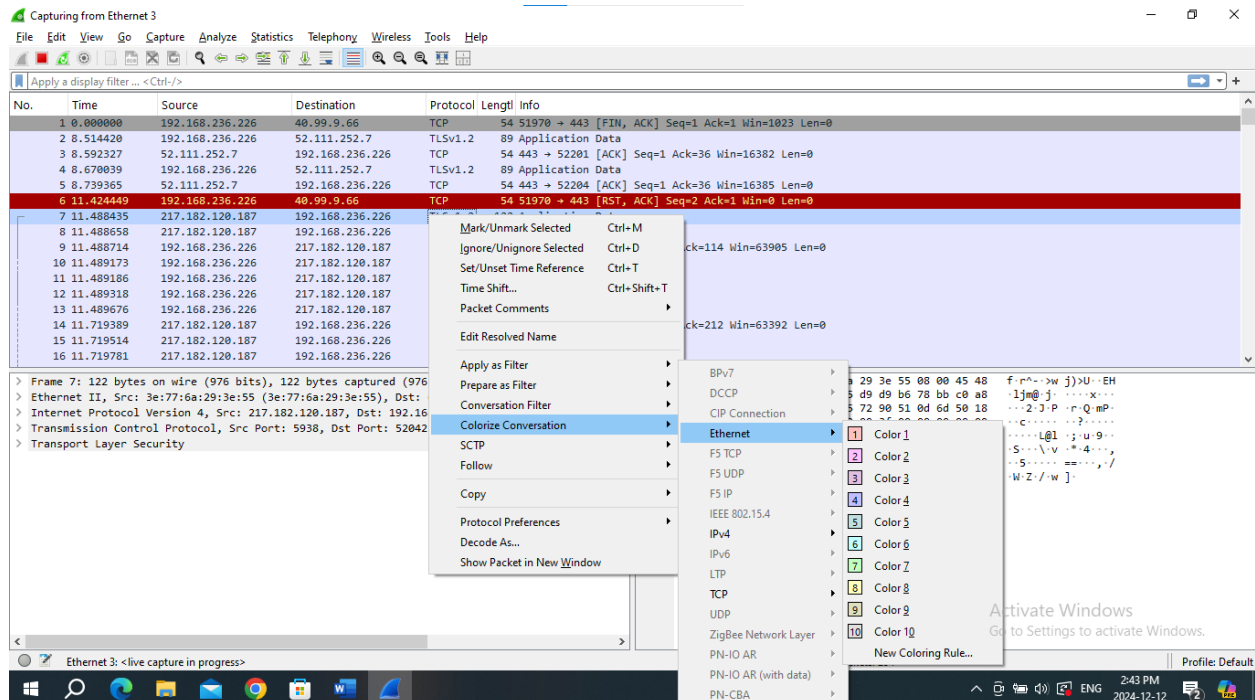
Functionality of Wireshark:

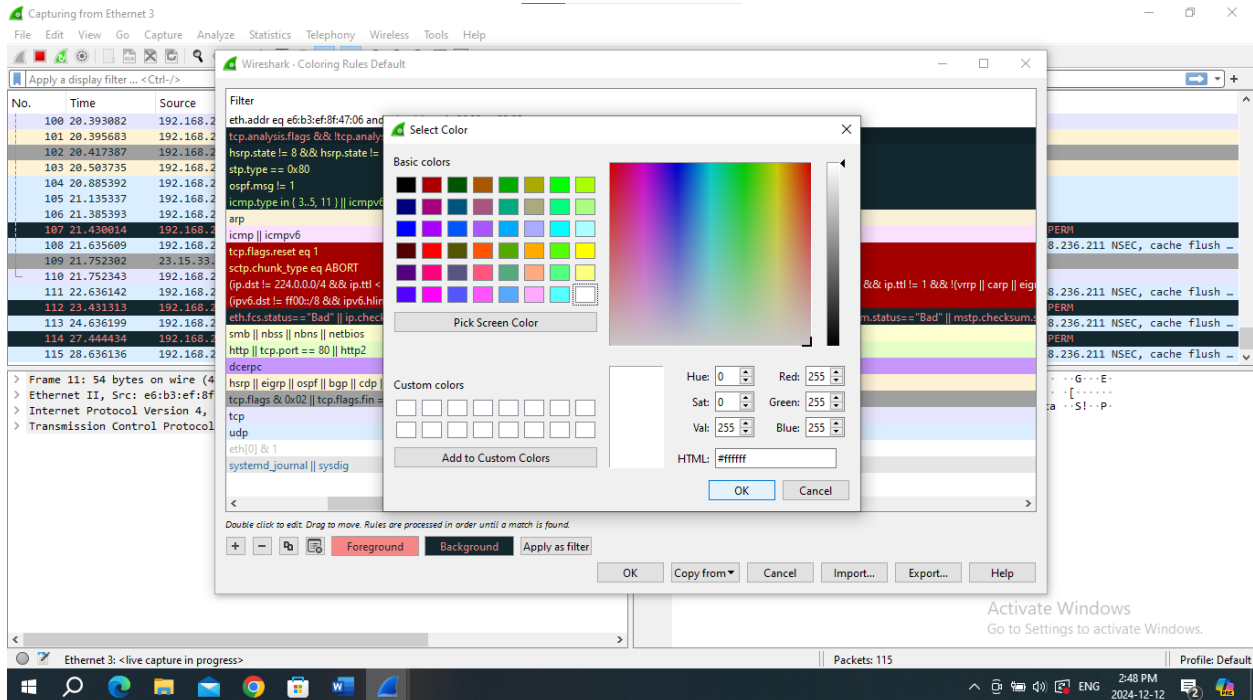
1. **Packet Capture:** Captures live network traffic and displays detailed packet information.
2. **Packet Filtering:** Filters captured data based on protocols, IP addresses, ports, or keywords for focused analysis.
3. **Protocol Analysis:** Decodes and analyzes a wide range of network protocols like TCP, UDP, HTTP, and more.
4. **Traffic Inspection:** Allows in-depth inspection of packet data to detect errors, delays, or anomalies.
5. **Data Visualization:** Provides graphical representations such as flow charts and statistics to analyze traffic trends.
6. **Export and Save:** Saves captured data in various formats for future analysis or sharing with others.



Color Coding in Wireshark:

1. **Default Coloring Rules:** Wireshark uses color coding to highlight different types of traffic for easier analysis.
2. **Light Purple:** Represents TCP traffic.
3. **Light Blue:** Denotes UDP traffic.
4. **Black:** Highlights packets with errors or issues, such as checksum errors.
5. **Green:** Indicates successful and established TCP connections.
6. **Custom Rules:** Users can create custom coloring rules based on filters to identify specific packets quickly.

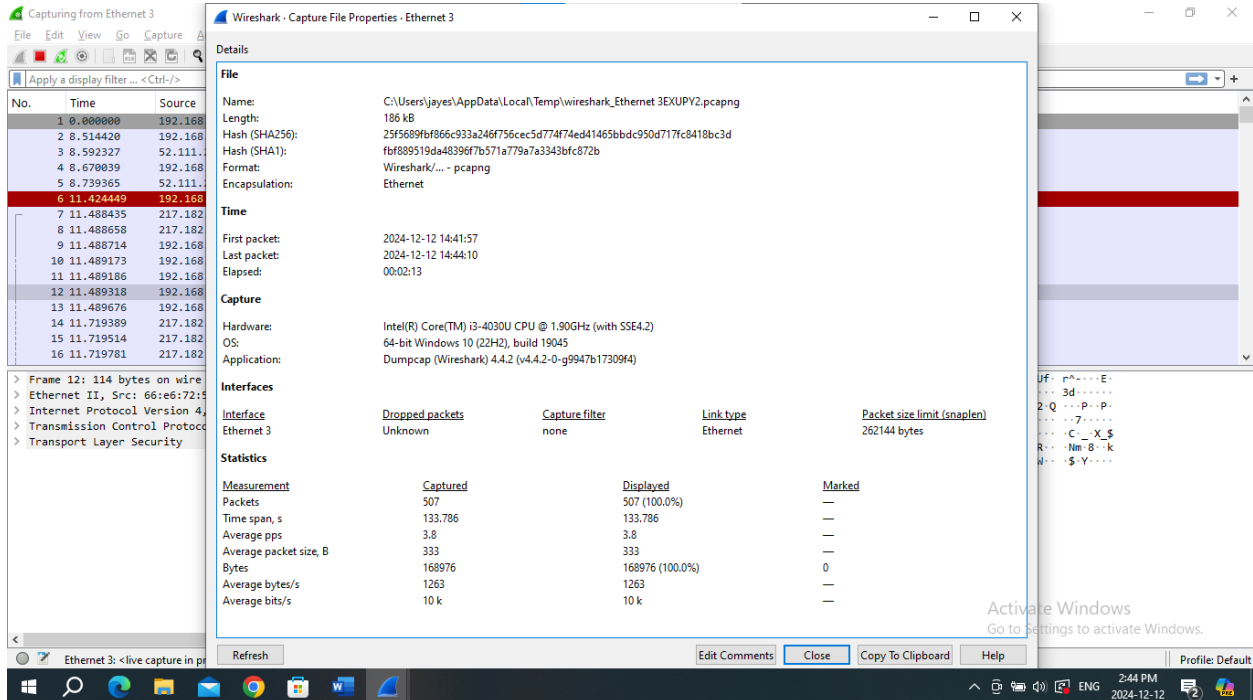




Features of Wireshark:

1. **Deep Packet Inspection:** Allows detailed analysis of individual packets, including headers and payloads.
2. **Real-Time Traffic Capture:** Captures network traffic live from various network interfaces (Ethernet, Wi-Fi, etc.).
3. **Protocol Support:** Supports a wide range of protocols (over 1000), including HTTP, DNS, TCP/IP, and more.
4. **Advanced Filtering:** Provides powerful display filters to focus on specific packets based on conditions like IP address, protocol, or port.
5. **Packet Reassembly:** Reassembles fragmented packets and stream data for a complete analysis of communication sessions.
6. **Statistics and Graphs:** Generates statistics and graphs like protocol hierarchy, packet length distribution, and throughput analysis.
7. **Export Options:** Saves captured data in various formats (PCAP, CSV, etc.) for further analysis or sharing.

8. **Cross-Platform:** Available on multiple operating systems, including Windows, macOS, and linux.



Conclusion:

In this practical, we explored the functionality and uses of Wireshark, a powerful network packet capture and analysis tool. We learned how Wireshark captures live network traffic, decodes various protocols, and provides deep insights into network behavior. The tool helps in monitoring and troubleshooting network issues by analyzing individual packets, detecting performance bottlenecks, and identifying security vulnerabilities.

We observed the color-coding system used in Wireshark to easily differentiate between types of traffic, such as TCP, UDP, and errors, enhancing the overall user experience. The ability to filter captured packets based on IP addresses, protocols, or ports allowed us to focus on specific network interactions and conduct more efficient analysis.

This practical has provided valuable hands-on experience with Wireshark, demonstrating its importance as an essential tool for network analysis and security monitoring.