**Experiment-10**

**Aim:** Study of Wireshark and understand its functionality

**Prerequisite:** Nil

**Outcome:** To impart knowledge of Application Layer Protocol

**Theory:**

* Wireshark stands as a robust network protocol analyzer, empowering users to capture, analyze, and scrutinize data packets circulating across a computer network. This tool offers an intricate view of the traffic within a network, aiding in the comprehension, troubleshooting, identification of security flaws, and enhancement of network performance. Supporting a wide array of network protocols, Wireshark serves as a versatile utility suitable for novices and seasoned network professionals alike.
* Its functionality involves intercepting and logging data packets in transit through a network interface. Once recorded, Wireshark organizes this information in an easy-to-navigate graphical interface, enabling users to filter and dissect packets based on multiple criteria such as source, destination, protocol type, and more. This level of detail allows users to analyze individual packet content, facilitating problem diagnosis, anomaly detection, and a comprehensive understanding of network behavior. Wireshark remains an indispensable tool for network administrators, security professionals, and developers engaged in network-related applications.

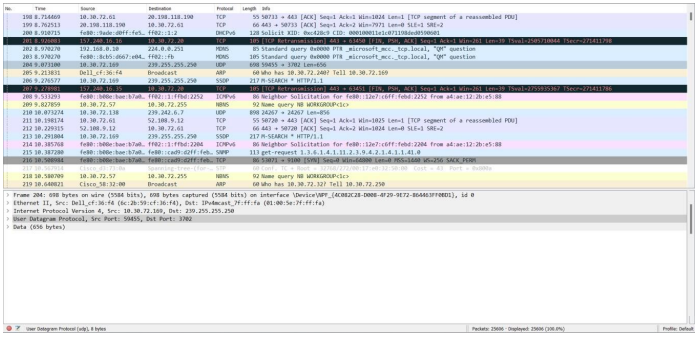
**Procedure:**

1. Install the Wireshark and integrate it with network simulator.
2. Analyze the traffic using Wireshark.

**Steps:**

1. Install the Wireshark and look at the different network interfaces on your computer.
2. Identify and select the network interface that you want to monitor. This could be a wireless connection, Ethernet, or any other network interface your computer uses to send and receive data. Click the “Start Capturing Packets” button (the shark fin icon) next to the chosen interface to begin capturing live traffic.
3. Allow Wireshark to capture traffic for a sufficient period to gather the data you need for analysis. This might be a few seconds, minutes, or more, depending on your requirements. Click the red “Stop” button to halt the packet capture when you feel you have enough data.
4. Use the “Display Filter” bar to enter specific protocols, IPs, or criteria to isolate specific types of traffic. For example, typing “http” will show only HTTP traffic. Click on a packet to view detailed information. The middle pane shows the packet details, while the bottom pane shows the raw data.
5. Use the color coding as a guide to quickly identify the types of traffic - each color represents a different protocol or traffic type. Right-click a packet and select “Follow” -> “TCP Stream” (or UDP, etc.) to see the entire conversation between endpoints.
6. Examine the packet details pane for more information on each layer of network protocols within a selected packet. If needed, you can save your packet capture by going to “File” -> “Save As” to analyze it later or use it as documentation.
7. Explore the “Statistics” menu for graphical views, endpoint summaries, protocol hierarchy, and other advanced analyses. You can export specific packets or conversations by selecting them and then choosing the appropriate option under the “File” menu.

**Output:**



**Observation & Learning**

* In Wireshark presents a comprehensive array of functionalities designed to capture and analyze network traffic originating from various sources. Its real-time analysis capabilities empower users to swiftly identify and resolve network issues, ultimately streamlining the troubleshooting process. One of Wireshark's key strengths lies in its broad support for numerous protocols, enabling a thorough examination of network traffic across various communication standards.
* Moreover, Wireshark excels in packet decoding, transforming captured data packets into a human-readable format, offering valuable insights into the underlying protocols and their respective fields. This feature is essential for obtaining a clear understanding of the complexities within network communication. Additionally, the tool supports packet filtering, enabling users to refine their analysis by applying specific criteria. This functionality significantly enhances the precision of investigations by focusing on relevant traffic. Overall, Wireshark serves as a robust and indispensable tool for network professionals aiming to gain comprehensive visibility into their network traffic.

**Conclusion**

Wireshark is a crucial tool valuable for network administrators, security experts, and developers. Its primary function involves capturing, analyzing, and examining network traffic, delivering a multitude of advantages. This encompasses swift troubleshooting via real-time analysis and packet filtering. Wireshark's extensive protocol support provides a deep understanding of network behaviors. Moreover, it serves as a valuable resource for monitoring network performance and investigating security incidents. Its adaptability through scripting and dissectors further amplifies its usefulness for specialized tasks, while its availability across multiple platforms ensures accessibility for users operating various systems. In summary, Wireshark is an indispensable asset for those aiming to comprehensively comprehend and efficiently manage network traffic.