Experiment - 7

Date: 24/09/2025

ANALYSIS OF TRAFFIC PACKETS USING WIRESHARK

Aim:

To analyze network traffic packets using Wireshark and understand the structure, contents, and flow of different protocol layers in a network communication.

Introduction:

The basic tool for observing the messages exchanged between executing protocol entities is called a packet sniffer. Packet sniffers are a fundamental tool for monitoring and analyzing network traffic. They capture messages being sent or received by a computer and display detailed information about the various protocol fields. Wireshark is one such packet analyzer that uses a packet capture library to display these details. It is a free and widely used network protocol analyzer that allows users to observe packets at different layers of the network protocol stack, including Ethernet, IP, TCP, UDP, and HTTP.

Procedure:

1. Step 1: Get Wireshark
   • Download and install Wireshark from wireshark.org/download.html.
   • Ensure libpcap or WinPcap packet capture library is installed.
   • Disable antivirus software if it blocks packet capture.
   • Connect to an Ethernet or WiFi network (in monitor mode).
2. Step 2: Run Wireshark
   • Launch the Wireshark application.
   • The graphical user interface (GUI) will appear showing menus, toolbars, and empty packet windows.
3. Step 3: Explore Wireshark Interface
   • Command menus allow actions like opening, saving, and capturing packets.
   • Packet listing window shows each captured packet with timestamp, source, destination, and protocol.
   • Packet details window shows protocol-level breakdown (Ethernet, IP, TCP/UDP, HTTP).
   • Packet contents window shows raw data in ASCII and hexadecimal.
   • The filter field allows focusing on specific protocols (e.g., 'http').
4. Step 4: Perform a Test Run
   • Open a web browser and clear cache.
   • Start Wireshark capture from the correct network interface.
   • Visit a webpage (e.g., http://www.ece.cmu.edu/~ini740/Lab0/lab0.html).
   • Stop capture once the page loads.

- Apply filters such as 'http' or specific IPs to view HTTP communication.
- Expand/collapse layers in packet details to observe headers and payloads.

5. Step 5: Analyze and Interpret
   - Observe TCP handshake, HTTP GET/POST requests, and responses.
   - Learn to use color coding and time references for better analysis.
   - Experiment with display and capture filters for efficiency.
   - Save the capture file for documentation or later analysis.
6. Step 6: Exit Wireshark
   - Stop capture and close the application after saving captured data.

Output:

The captured packet data in Wireshark displays various layers of network communication. Users can observe Ethernet frames, IP datagrams, TCP segments, and HTTP messages exchanged between client and server. The HTTP GET request and corresponding response can be analyzed to understand network protocol interactions.

Result:

Successfully analyzed network traffic packets using Wireshark. Understood the structure of network packets and learned how to capture, filter, and analyze different protocol layers.