Exp no: 7 Analysis of Traffic packets using Wireshark
Date: 29/9/25

Aim:

To analyze and observe network packets exchanged between client and server using Wireshark

Software required:

* A computer with Internet access
* Wireshark Network Analyzer
* Winpcap/ libpcap library

Algorithm:

Step 1: Install Wireshark

1. Download Wireshark from https://wireshark.org/download.html
2. Install the wireshark package along with WinPcap/libpcap capture library.
3. Disable antivirus temporarily if Wireshark cannot capture local traffic.
4. Ensure the system is connected to a wired (Ethernet) or Wi-Fi network.

Step 2: Launch Wireshark

1. Open Wireshark
2. The main interface will display a list of available network interfaces.
3. Identify your active network adapter.

Step 3: Select capture Interface

1. Click on the active adapter
2. Go to Capture → Options to view interface details
3. Click start to begin capturing packets.

Step 4: Generate Network Traffic

1. Open your web browser
2. Visit any website
3. Allow the page to load fully - this will generate HTTP, DNS and TCP packets.

Step 5: Stop Packet Capture

1. Return to Wireshark
2. Click the Red Stop button
3. The captured packets now appear in the Packet list window

Step 6: Apply Display Filters

1. In the Filter Bar, type

   http && (ip.src == <your_IP> || ip.dst == <your_IP>)

2. Click Apply - Wireshark will now show only HTTP packets exchanged with your system.

Step 7: Inspect Captured Packets

1. Select an HTTP GET packet from the list.
2. Observe the three main panes:
   * Frame Details - Basic frame info
   * Ethernet Header - Source & Destination MAC addresses
   * IP Datagram - Source & Destination IPs
   * TCP Segment - Port numbers, sequence, acknowledgments
   * HTTP layer - Request / Response details.

Step 8: Analyze Timing and Response

1. Right-click on a packet → Set Time Reference
2. Check time differences between request & response packets
3. Calculate Round Trip Time (RTT) between the client and server

Step 9: Use additional filters

a. To show only TCP traffic:

   tcp

b. To show only packets from a specific host:

   ip.addr == <specific_IP>

c. To show DNS packets:

   dns

Step 10: Save & Exit

1. Go to File → Save As to save the capture file
2. Close Wireshark

Output

- Captured packets include DNS, TCP, HTTP and ARP
- Observed the complete packet structure:
  - Ethernet Frame → IP Datagram → TCP Segment → HTTP Header
- Verified HTTP GET and HTTP Response message exchange
- Measured packet timing and response delay

Result

Wireshark successfully captured and analyzed real-time packets exchanged between client and server.