**EXPERIMENT NO. 5**

**Introduction to Wireshark and Traffic analysis (packet headers) using Wireshark**

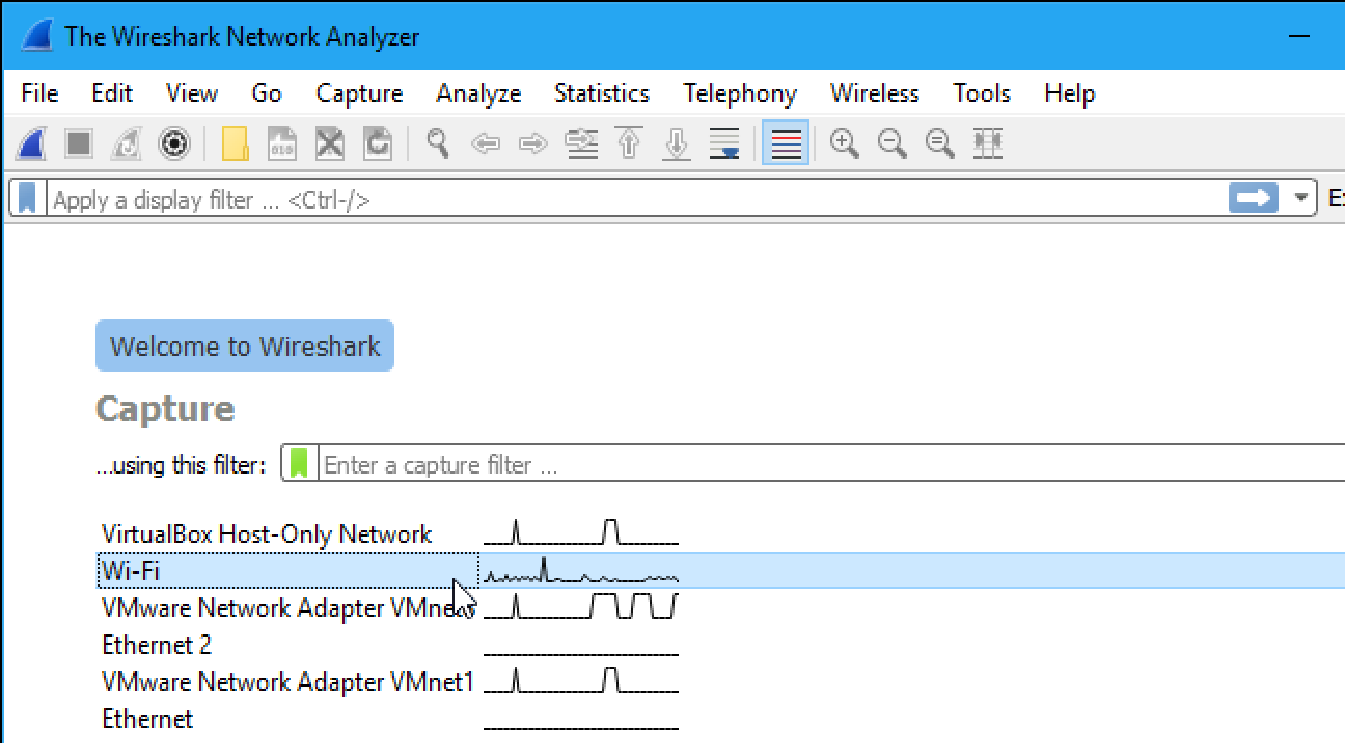
**Aim:** To study Wireshark and its applications in traffic analysis and password sniffing.

**Procedure**:

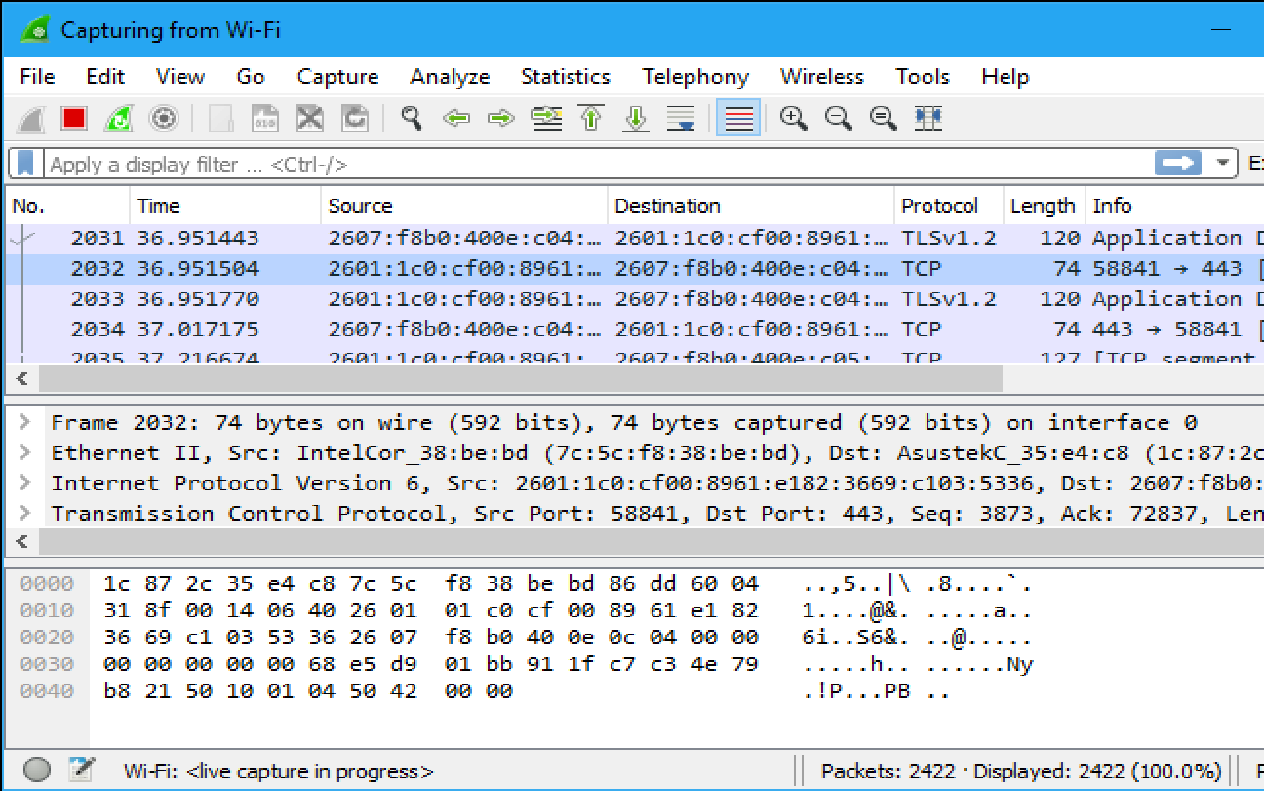
1. Open Wireshark.
2. Study Wireshark GUI and components (display filter, packet listing, packet header and packet content)
3. To begin packet capture, select the Capture pull down menu and select Options. This will cause the “Wireshark: Capture Options”
4. Click Start. Packet capture will now begin. All packets being sent (received) from (by) your computer are now being captured by Wireshark.
5. Once you begin packet capture, a packet-capture-summary window will appear. It will display all information relevant to that packet while Wireshark is running, enter the following URL in a web browser (whichever is installed on your system) <http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html>
6. After your browser has displayed the INTRO-wireshark-file1.html page, stop Wireshark packet-capture by selecting STOP in the Wireshark capture window.
7. Type in “http” (without the quotes, and in lower case – all protocol names are in lower case in Wireshark) into the display filter specification window at the top of the main Wireshark window.
8. Then select Apply (to the right of where you entered “http”). This will cause only HTTP message to be displayed in the packet-listing window.
9. Select the first http message shown in the packet-listing window. This should be the HTTP-GET message which was sent from your computer to the http://gaia.cs.umass.edu HTTP server. When you select the HTTP GET message, the Ethernet frame, IP datagram, TCP segment, and HTTP message header information will be displayed in the packet-header window.
10. By clicking plus and minus boxes to the left side of the packet details window, minimize the amount of Frame, Ethernet, Internet Protocol, and Transmission Control Protocol information displayed. Maximize the amount information displayed about the HTTP protocol).

**Procedure for Capturing Data Packets**

After downloading and installing Wireshark, you can launch it and double-click the name of a network interface under capture to start capturing packets on that interface. For example, if you want to capture traffic on your wireless network, click your wireless interface.

****

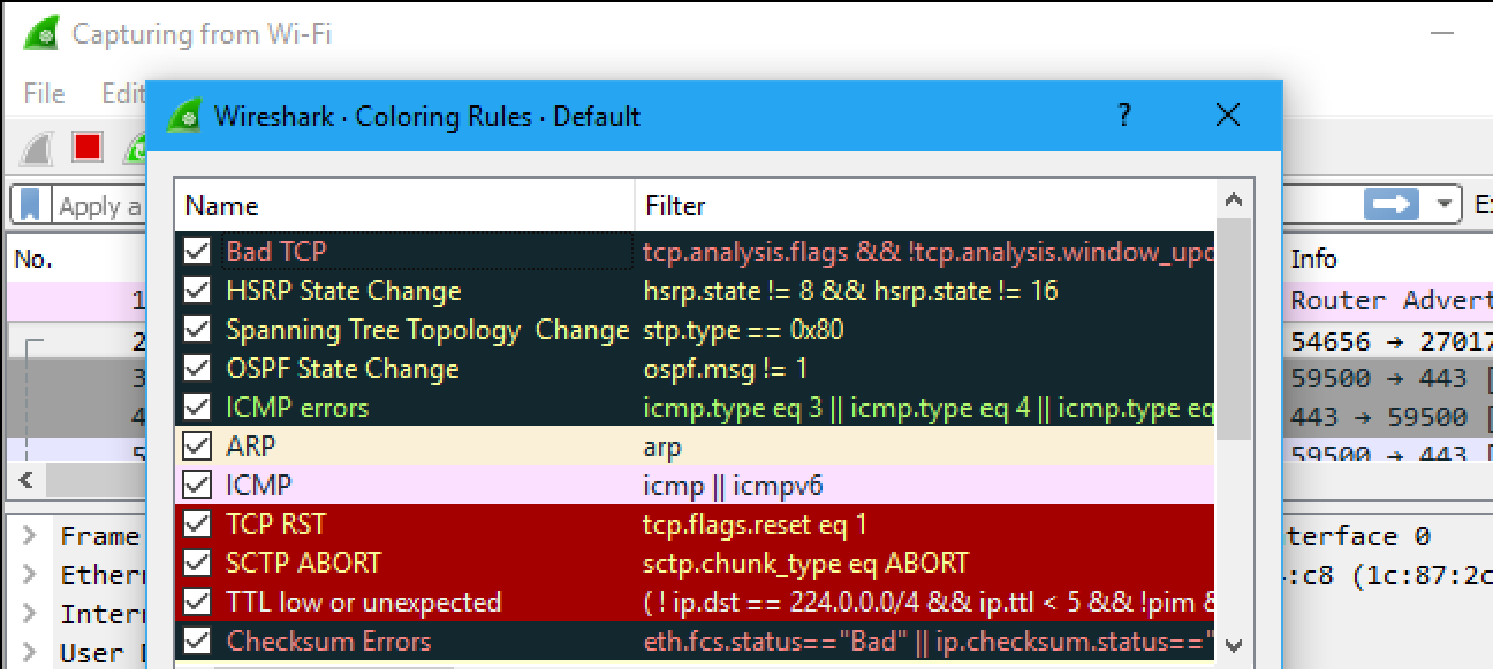
As soon as you click the interface’s name, you’ll see the packets start to appear in real time. Wire-shark captures each packet sent to or from your system.

****

Click the red “Stop” button near the top left corner of the window when you want to stop capturing traffic.

**Color Coding**

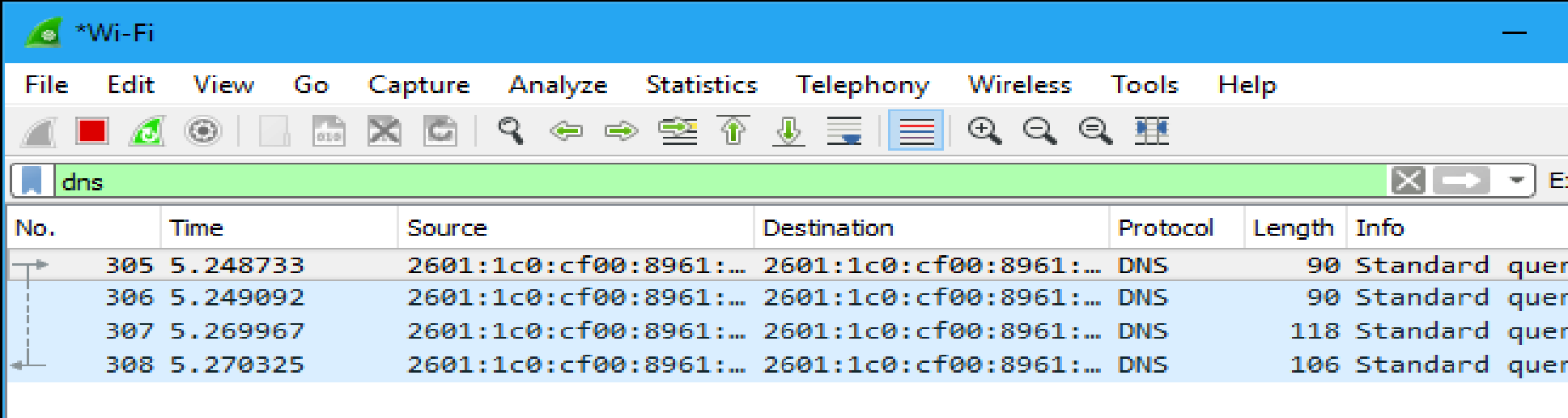
Wireshark uses colors to help you identify the types of traffic at a glance. By default, light purple is TCP traffic, light blue is UDP traffic, and black identifies packets with errors .To view exactly what the colour codes mean, click View > Coloring Rules. You can also customize and modify the colouring rules from here, if you like.

****

**Procedure for Filtering Data Packets**

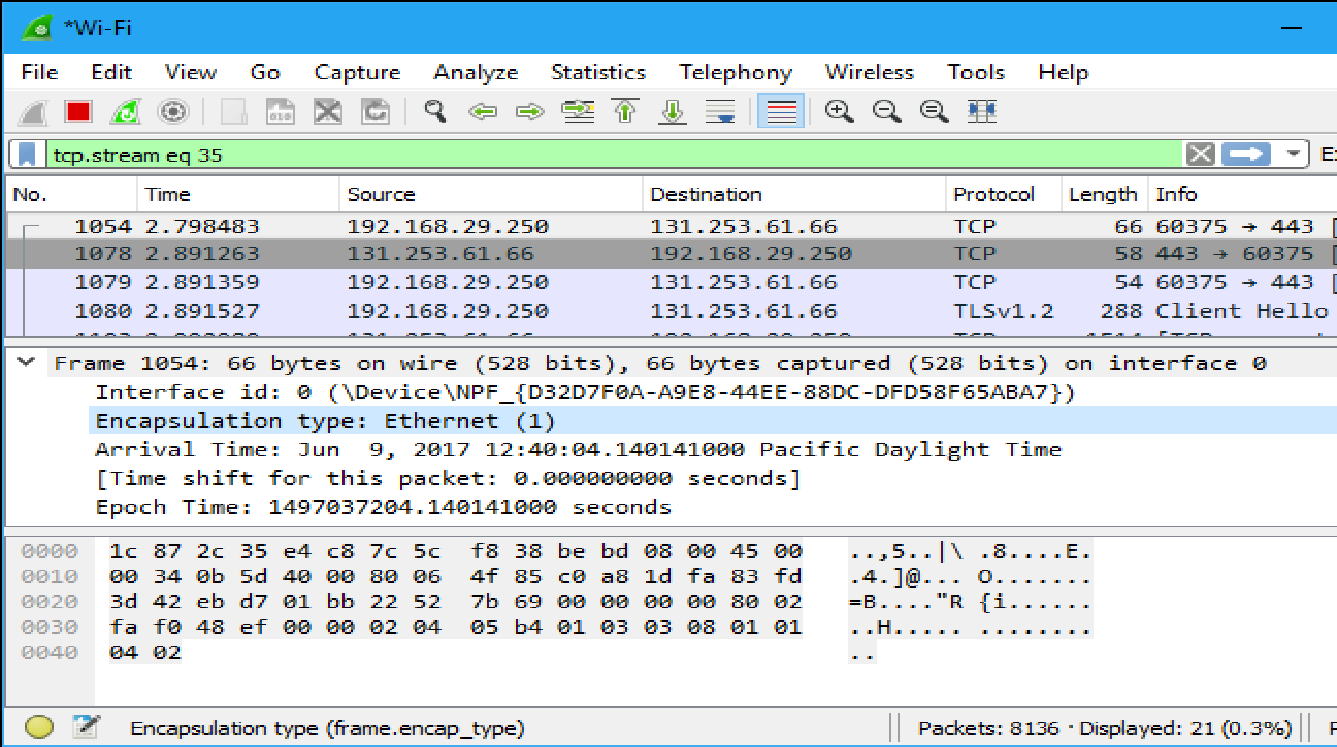
The most basic way to apply a filter is by typing it into the filter box at the top of the window and clicking Apply (or pressing Enter). For example, type “dns” and you will see only DNS packets. When you start typing, Wireshark will help you autocomplete your filter.

You can also click Analyze > Display Filters to choose a filter from among the default filters included in Wireshark. From here, you can add your own custom filters and save them to easily access them in the future.



**Procedure for inspecting Data Packets**

Click a packet to select it and you can dig down to view its details as shown below,

****

**Output:**

Attach Screenshots of following; (a brief explanation of each screenshot is required.)

1. Wireshark Interface window
2. Wireshark packet sniffing window
3. Wireshark window with different filters (arp, http, tcp, icmp, with ip address etc)
4. Wireshark window with Ethernet protocol details.
5. Wireshark window with arp protocol details.
6. Login window of an insecure webpage.
7. Wireshark window with http packet filtering for password sniffing.
8. TCP stream data window showing login id and password details.

**Post Experimental Exercise-**

**Using Wireshark answer the following questions- RIYA INDAP,44**

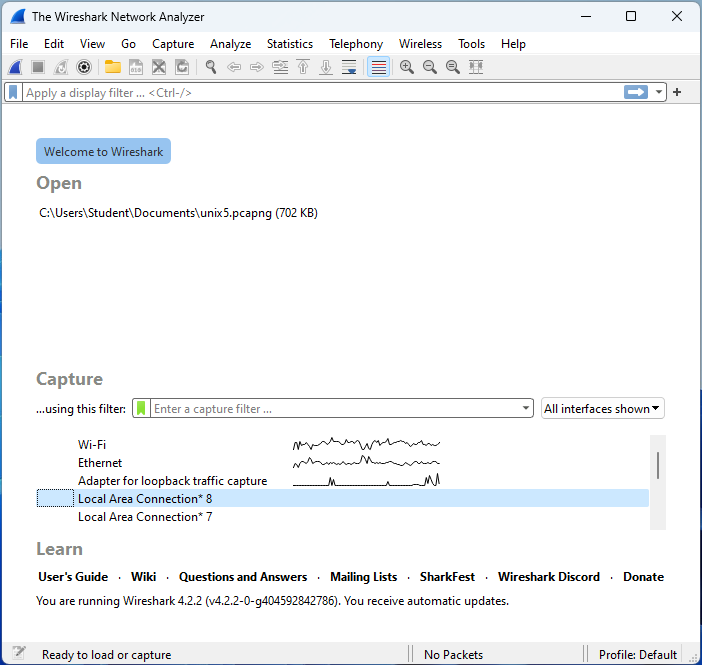
| **For Ethernet Protocol Frame** | |
| --- | --- |
| 1 | Frame length of the packet: 968 bytes (7744 bits) |
| 2 | Frame number: 2808 |
| 3 | Type of frame (Ether-type Number): IPv4 (0x0800) |
| 4 | Can you say that there is padding in the payload? No (padding=use of consecutive zeros,payload=data section) |
| 5 | Destination link-layer address: ASUSTekCOMPU\_16:b8:6b (c8:7f:54:16:b8:6b) |
| 6 | Source link-layer address: Sophos\_c8:66:82 (7c:5a:1c:c8:66:82) |
| 7 | Upper layer protocol: Internet Protocol Version 4 |
| 8 | Is destination link-layer address unicast or broadcast? unicast |

| **For ARP Protocol Frame** | |
| --- | --- |
| 1 | Hardware type: Ethernet (1) |
| 2 | Protocol type: IPv4 (0x0800) |
| 3 | Hardware size: 6 |
| 4 | Protocol size: 4 |
| 5 | Source hardware address: Sophos\_c8:66:82 (7c:5a:1c:c8:66:82) |
| 6 | Source IP address? 192.168.7.254 |
| 7 | Destination hardware address: 66:2a:ac:27:85:d3 (66:2a:ac:27:85:d3) |
| 8 | Destination IP address: 192.168.2.46 |
| 9 | Number of bytes of padding in a frame: 18bytes |
| 10 | Is it arp request or reply packet? reply packet |
| 11 | Identify two differences between arp request and reply packets.   1. arp reply packets are always unicast and arp request packets are always broadcast. 2. arp reply packets has opcode field set to (2) and arp request packets has opcode field set to (1). |

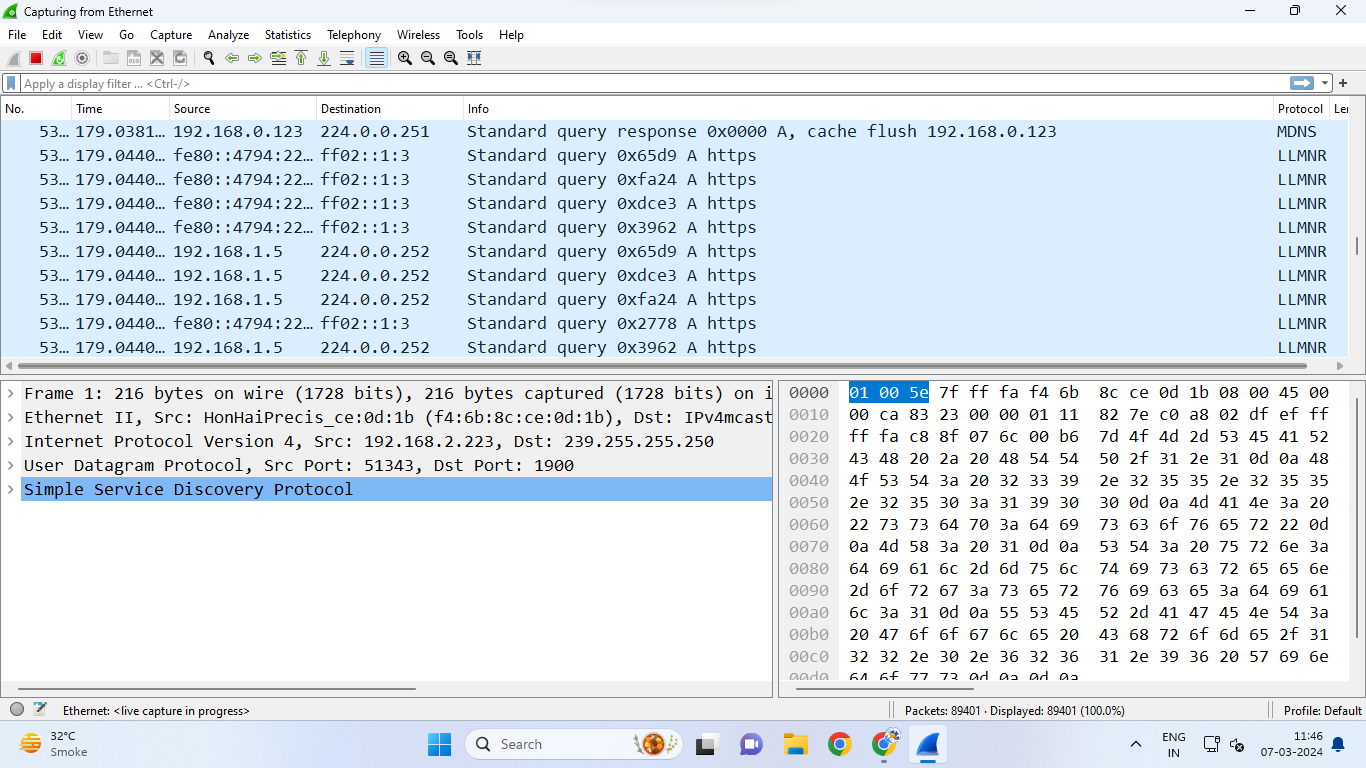
**Conclusion:** *(To be hand written on journal sheet)*

RIYA INDAP,44

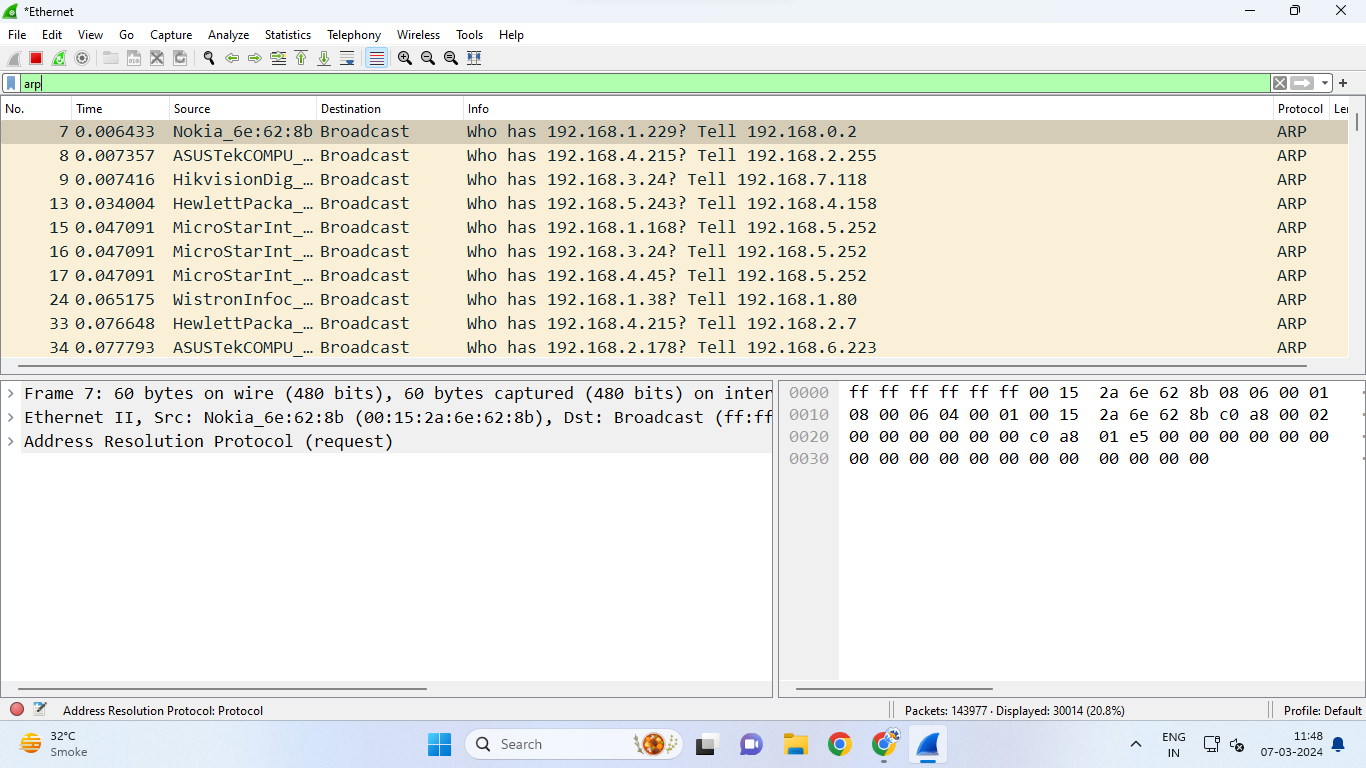
* Wireshark Interface window

*AFTER INSTALLING WIRESHARK WE OPENED IT AND UNDER CAPTURE SECTION OPENED ETHERNET INTERFACE*

* Wireshark packet sniffing window

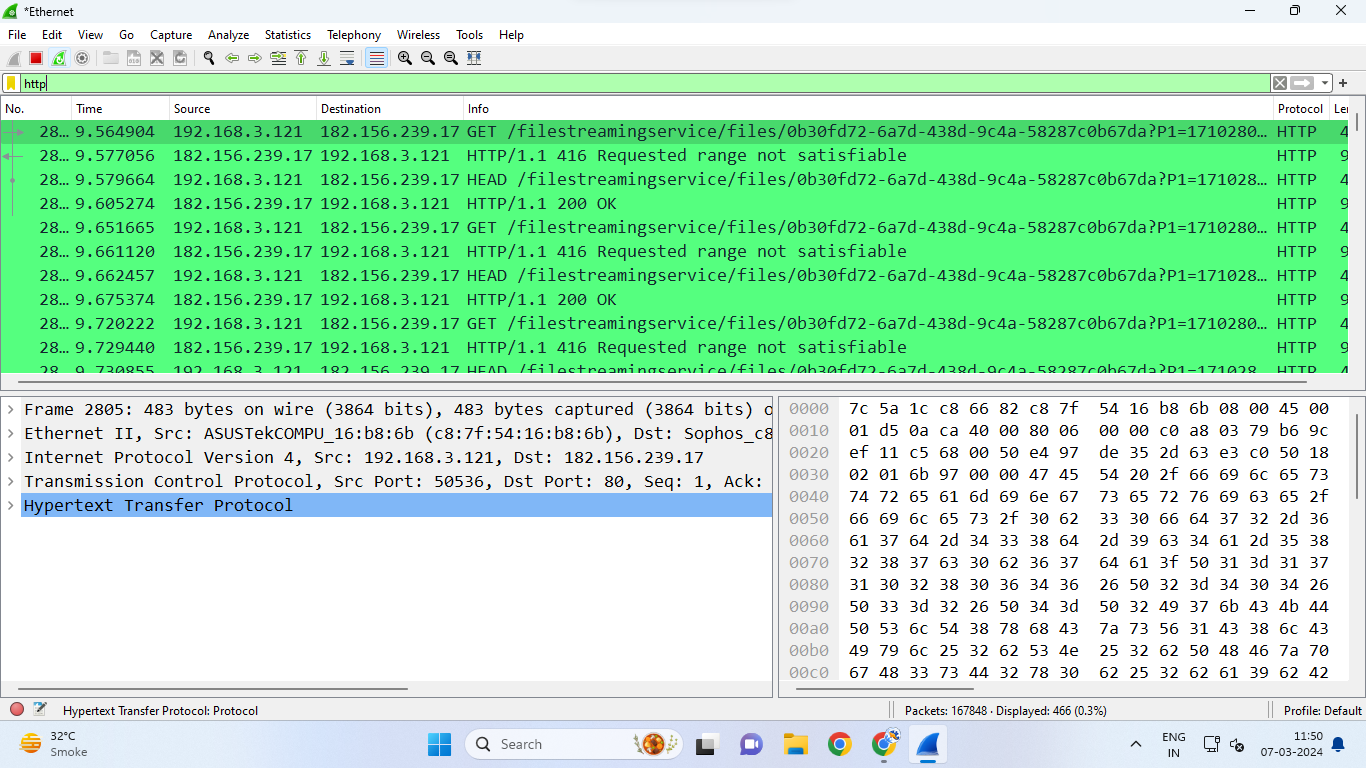
**

* Wireshark window with different filters (arp, http, tcp, icmp, with ip address etc)

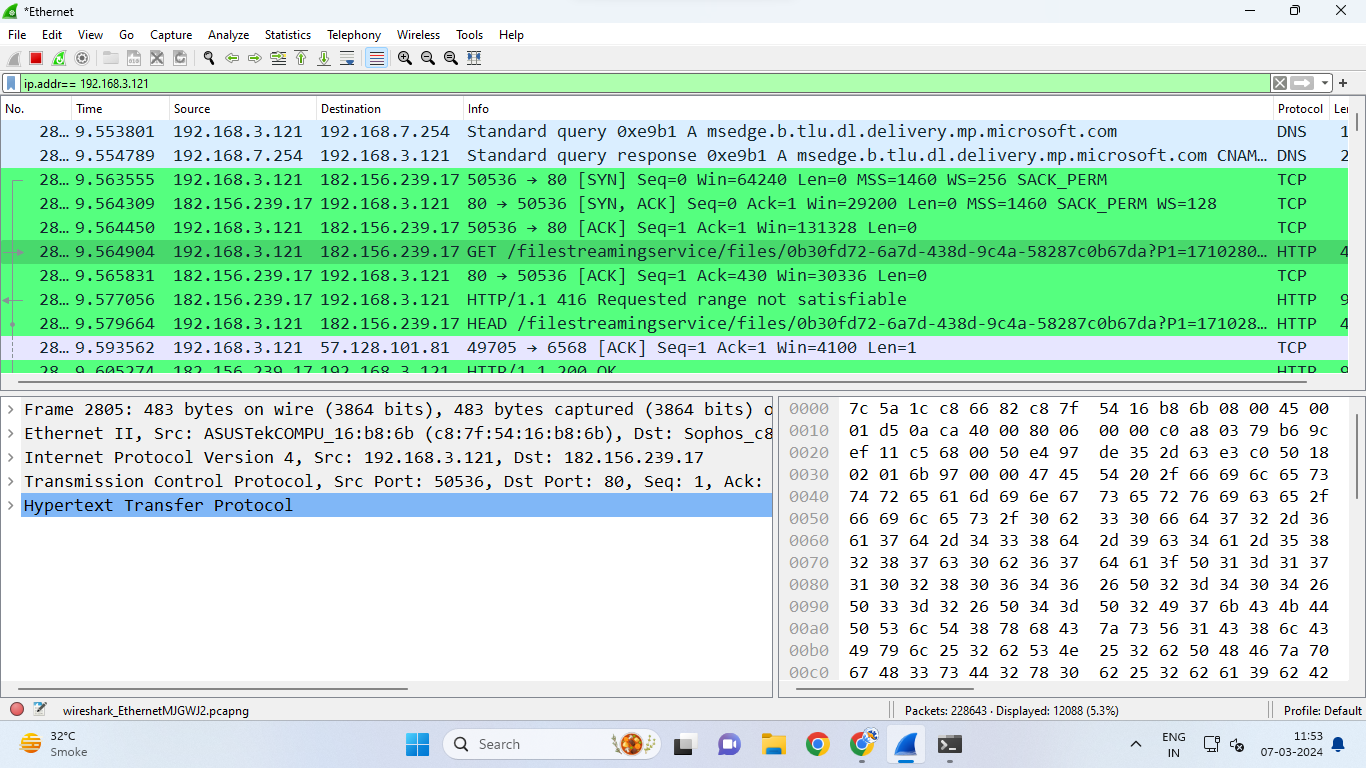
**

*in the top section there is apply display filter option using which the the packets are filtered based on the*

*filter applied.here we have applied ‘arp’ filter due to which the packets with ‘arp’ protocol are filtered.*

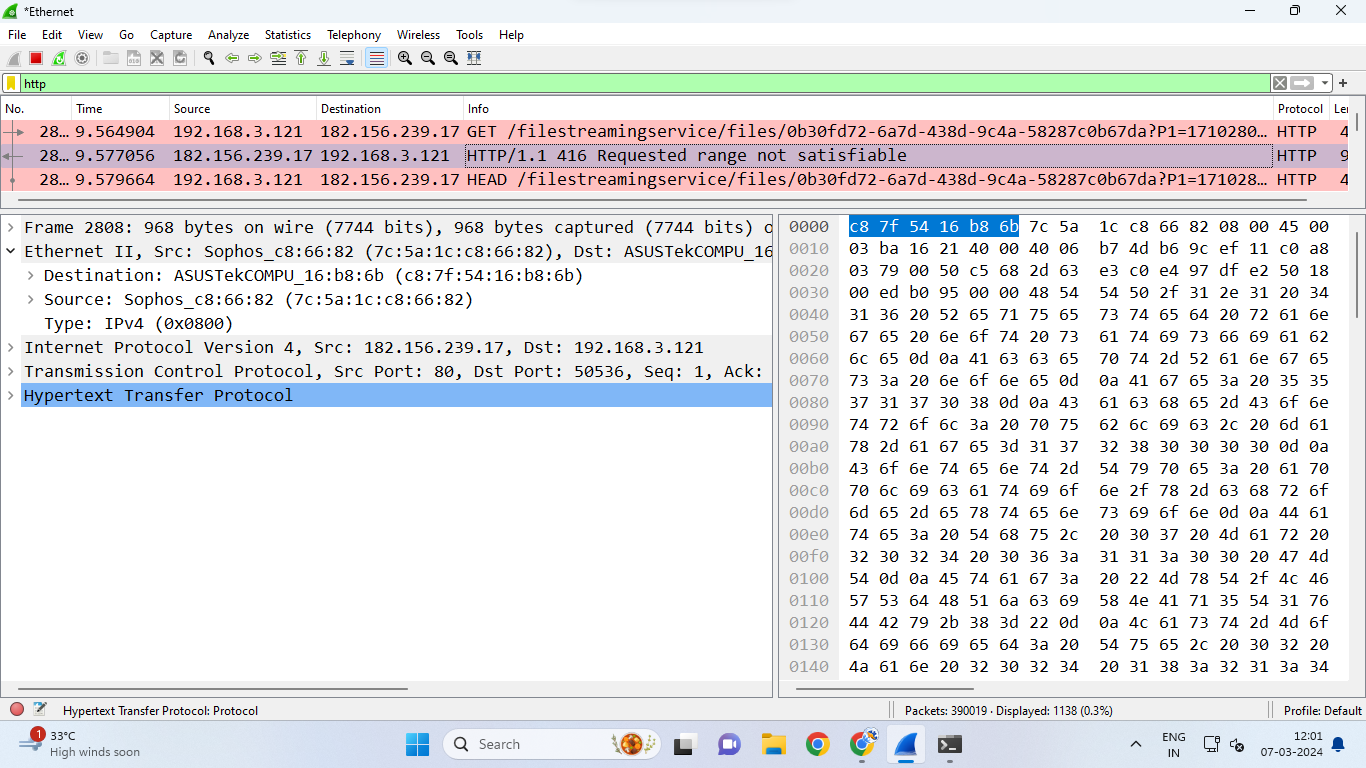
**

*here we have applied ‘http’ filter due to which packets with ‘http’ protocols are filtered.*

**

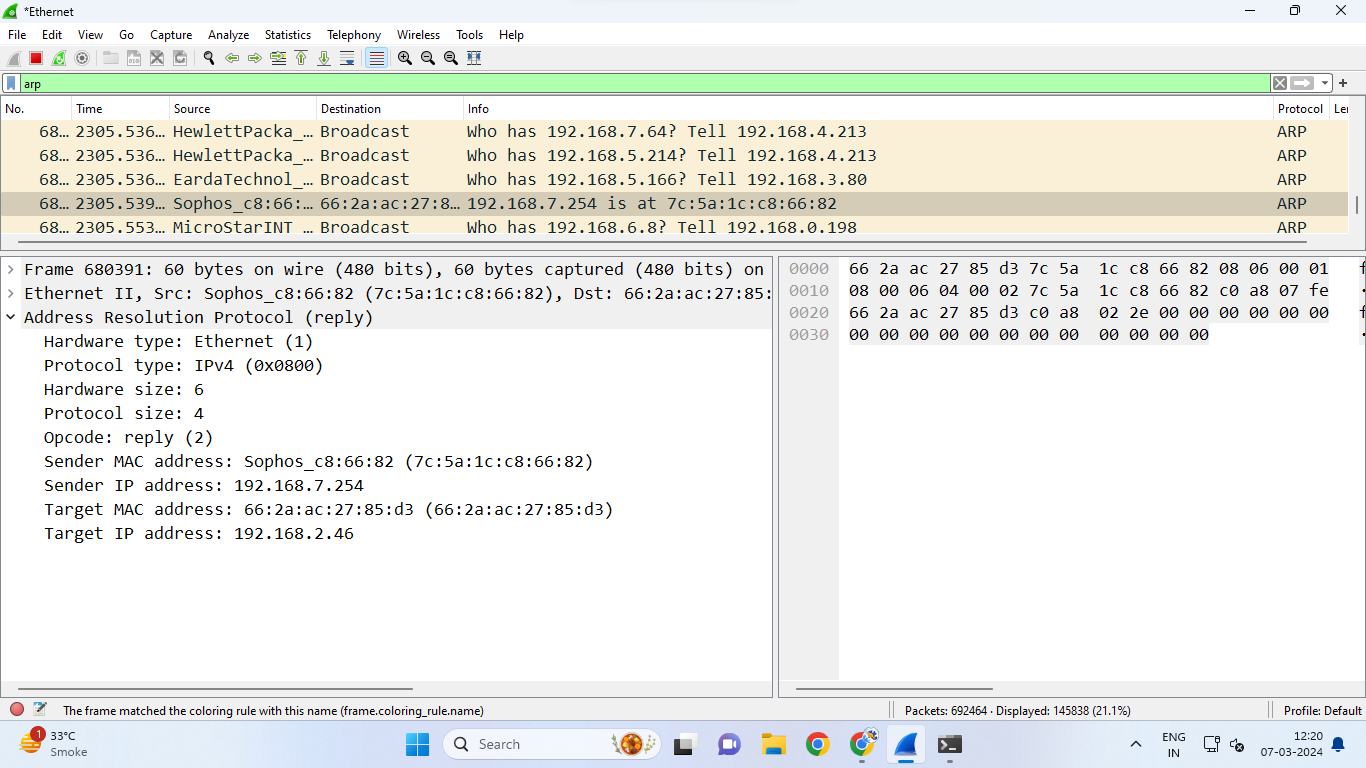
*here we have applied some ‘ip add of our ethernet’ as filter so the packets related to this ip address are filtered.*

* Wireshark window with Ethernet protocol details.



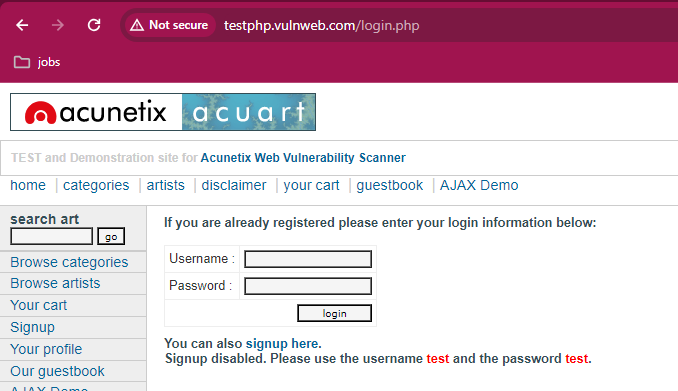
*We now selected a packet and saw it’s color scheme was changed .On selecting a packet its details were revealed in packet details pane.Using this one can explore information about Ethernet frames, including source and destination MAC addresses, frame types (such as IPv4, ARP, or IPv6), VLAN tags, frame lengths, and Ethernet protocol-specific details, facilitating in-depth analysis and troubleshooting of network communication.*

* Wireshark window with arp protocol details.



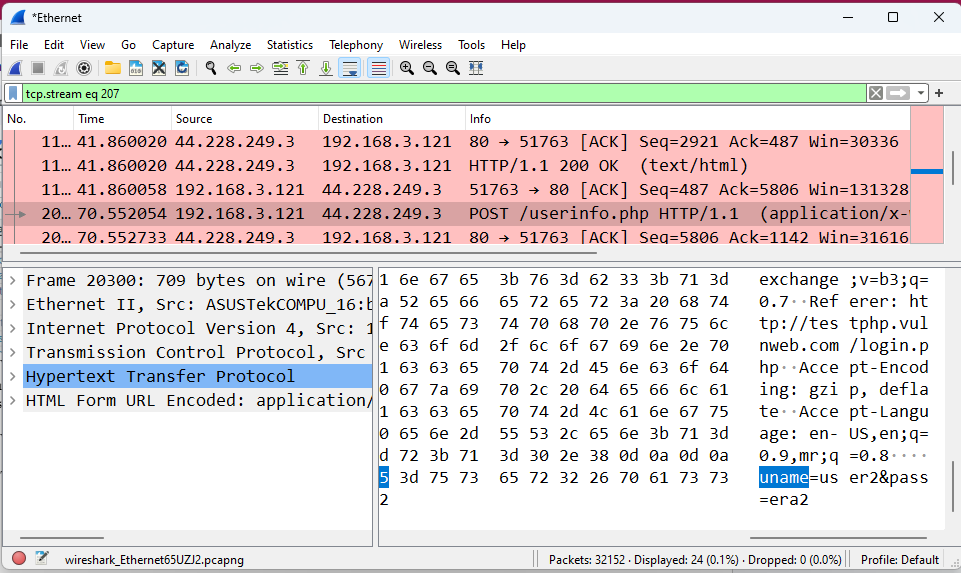
*The we applied a filter ‘arp’ and selected a packet and checked for it’s information in packet details pane.By this method one can get essential information such as ARP request and reply messages, sender and target MAC addresses, as well as IP addresses involved, aiding in network troubleshooting, MAC address resolution, and detecting potential ARP spoofing or cache poisoning attacks.*

* Login window of an insecure webpage.



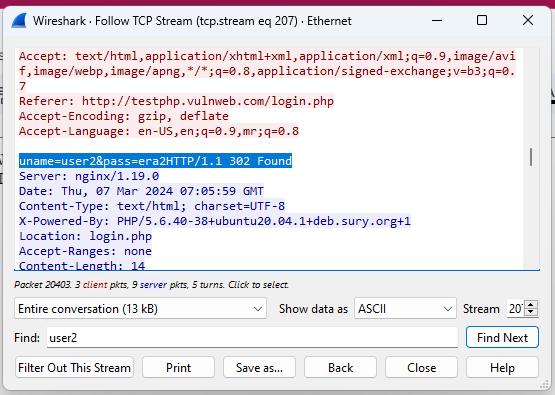
*We searched a insecure login page and login into it entering username and password.then we started wireshark then searched,stopped the packets using stop capturing packet buttons, applied http filter,then searched for ‘post’ in info section.*

* Wireshark window with http packet filtering for password sniffing.



*After finding ‘post’ in info section ,we selected that packet and checked in its packet byte section where the username and password that we entered on that insecure login page were clearly visible without any encryption.*

* TCP stream data window showing login id and password details.



*On right clicking on that same packet,browsing through ‘follow’ in it selected ‘tcp stream’.In tcp stream frame the whole data entered on that insecure login page was seen clearly without any kind off encryption.*