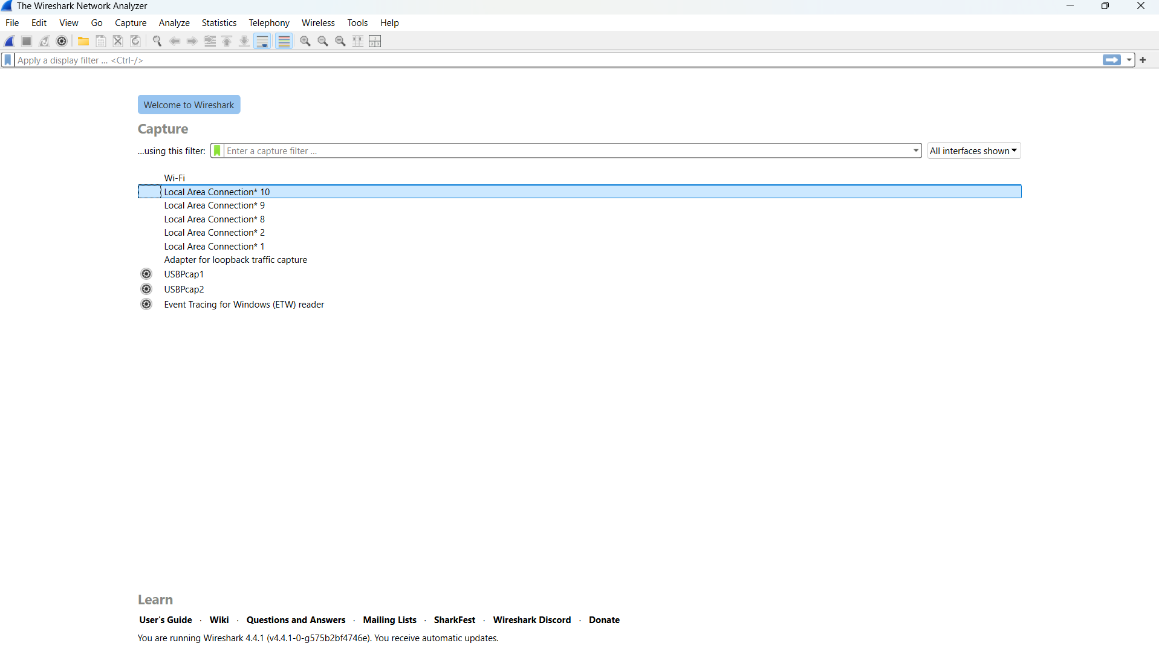
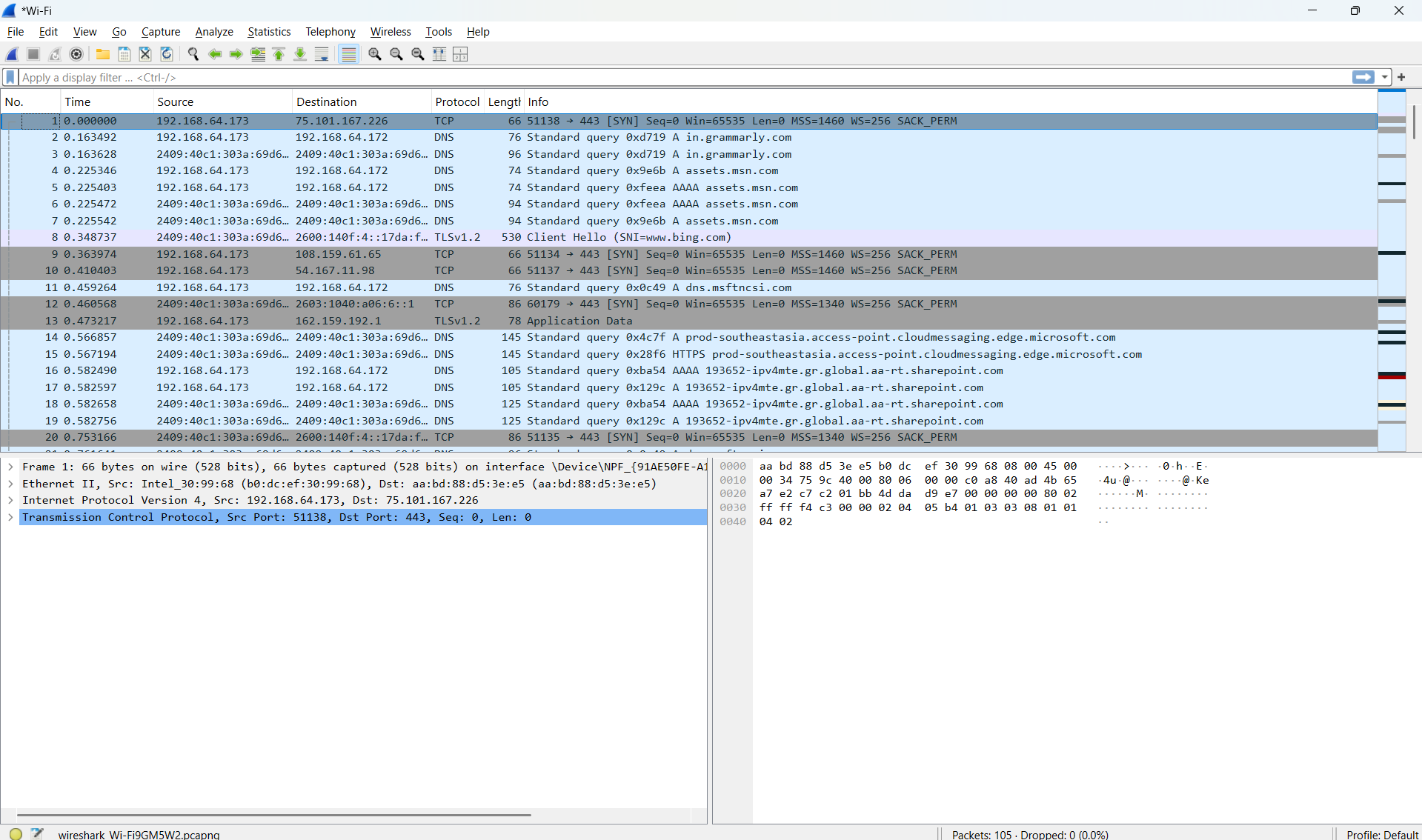
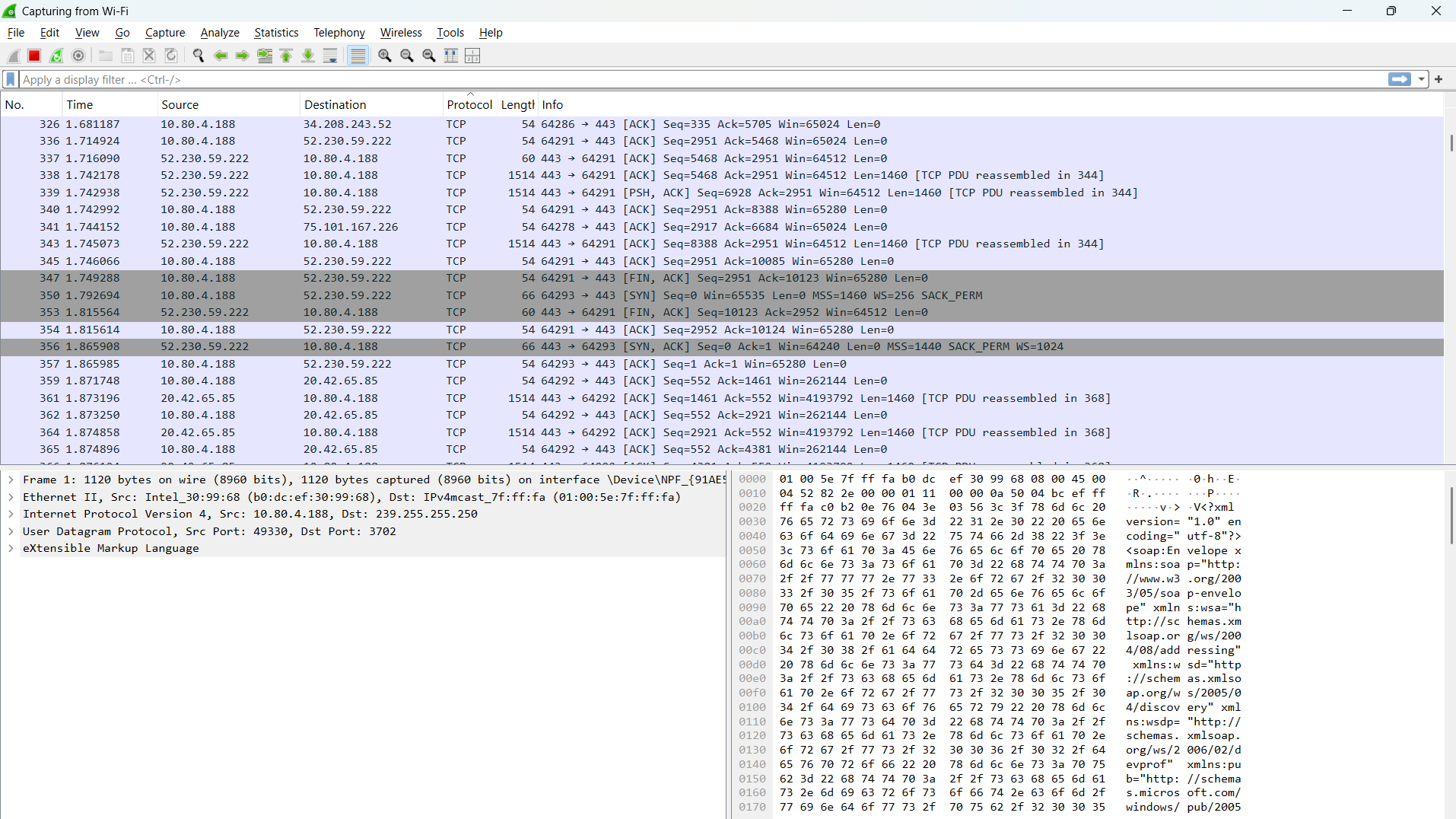
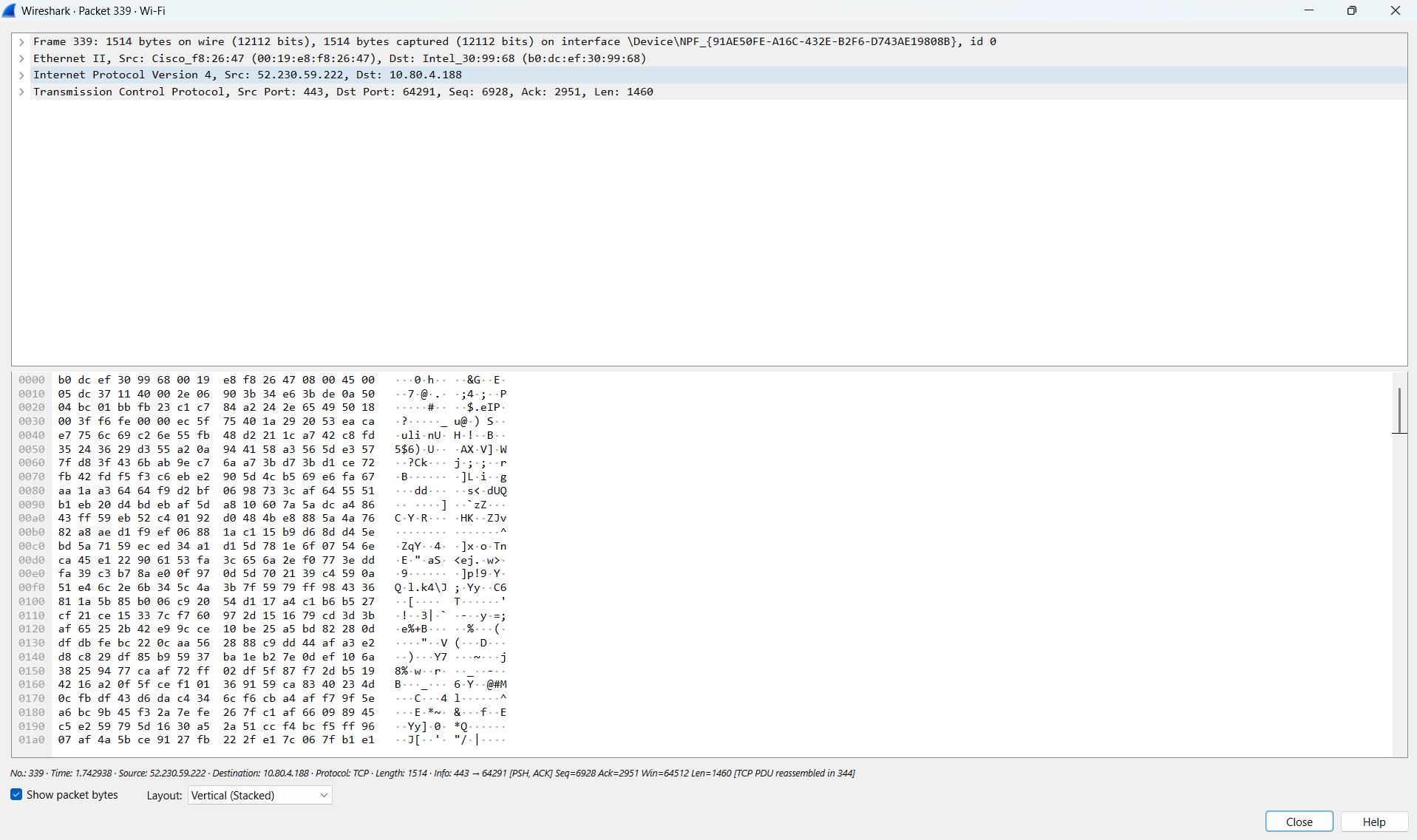
**Aim**: Monitor the live/real time network and analyze the concepts of various networking protocols like ARP, RARP, DHCP, HTTP, etc.

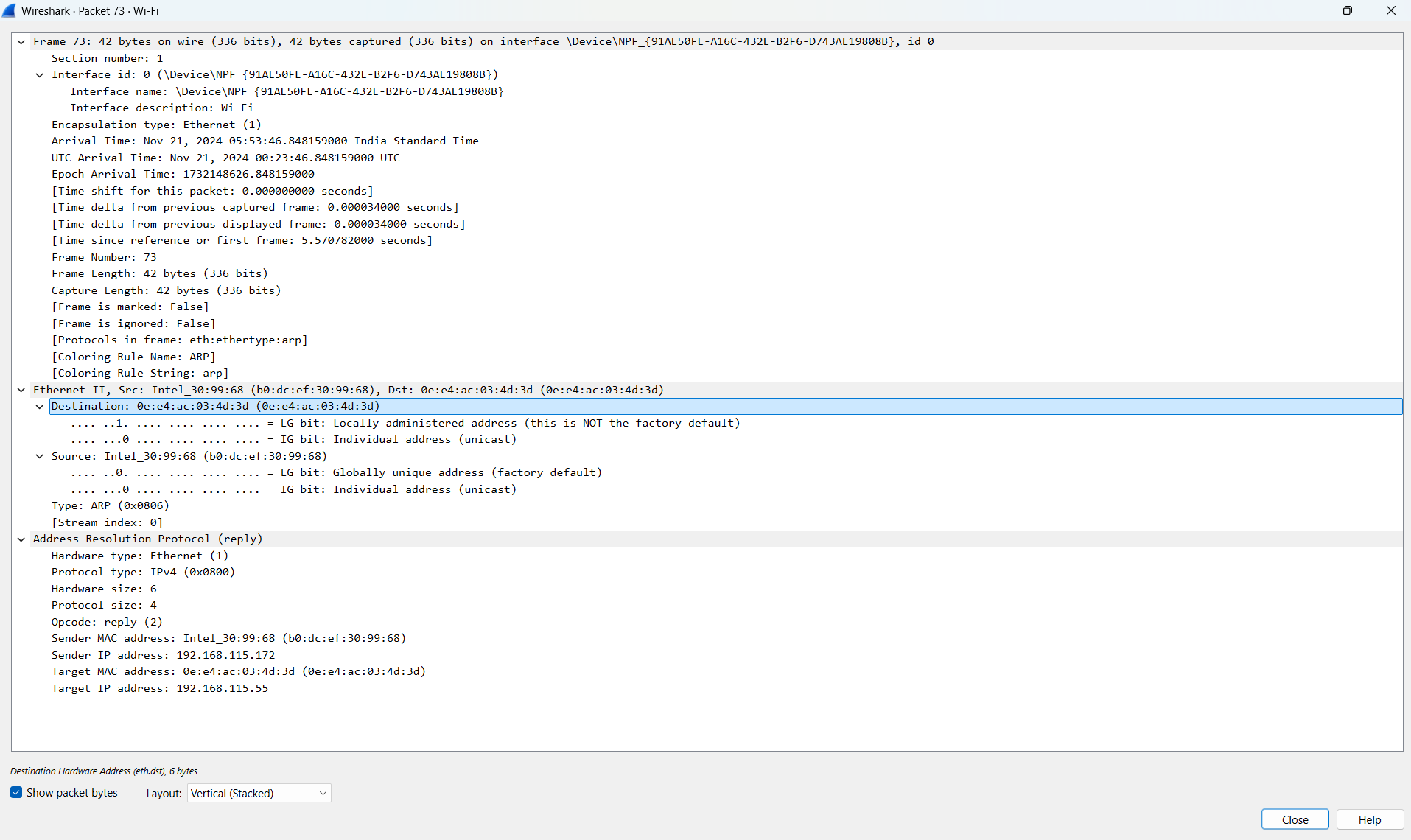
**Step – 1:-** Open Wireshark

**Step – 2 :-**  Select the Network from which you want to communicate

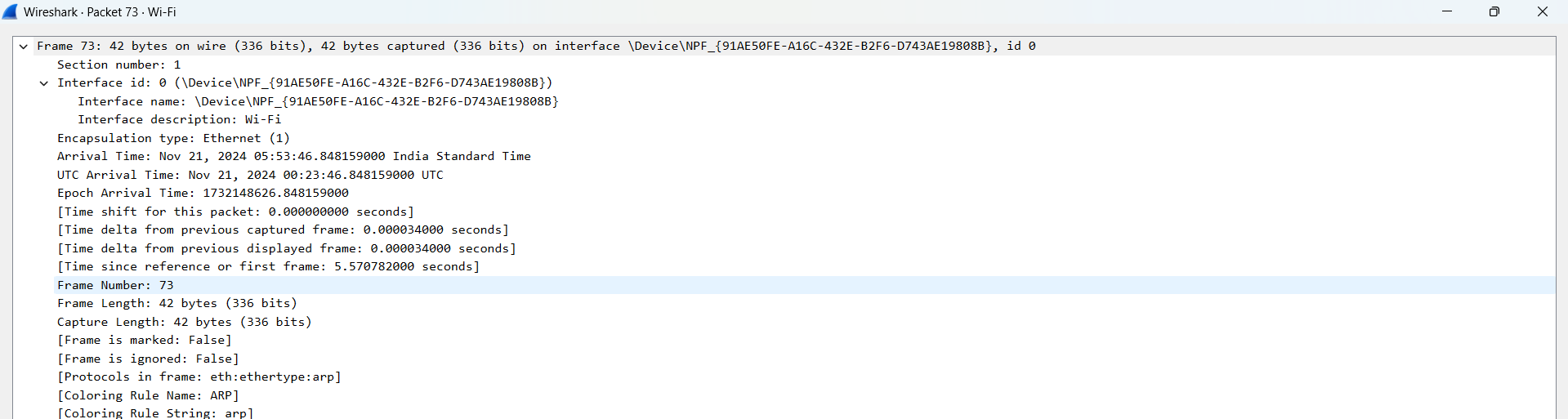


**Step – 3 :-**  Now when we press Protocol button it will sort the packet based on protocol used.

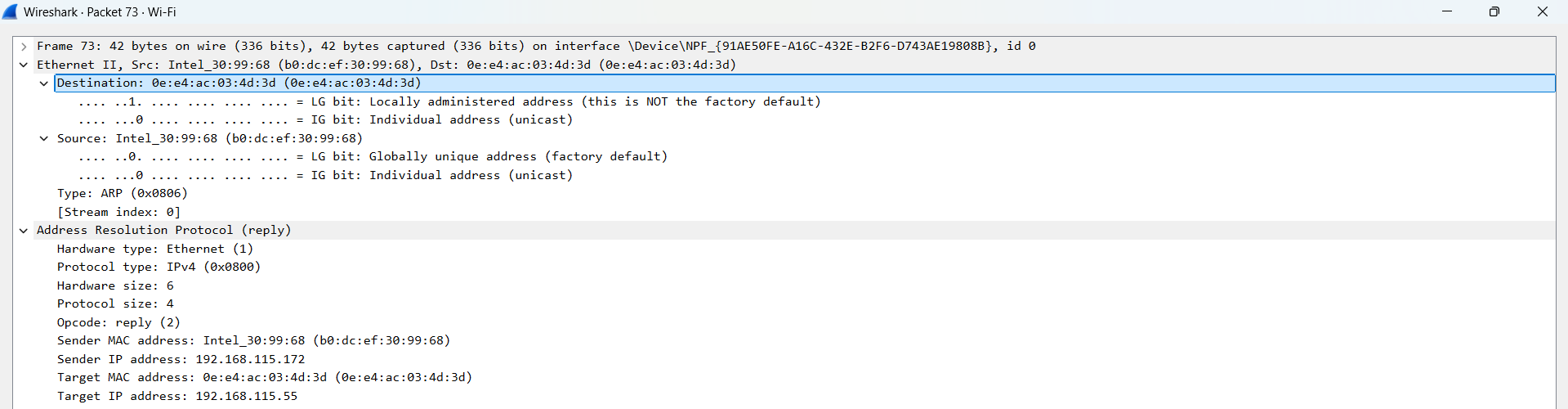
**Step – 4 :-**  Now when we press one of the packet it will open the packet and show every detsils.

**Step – 5 :-**  Now we will analyze one ARP Packet

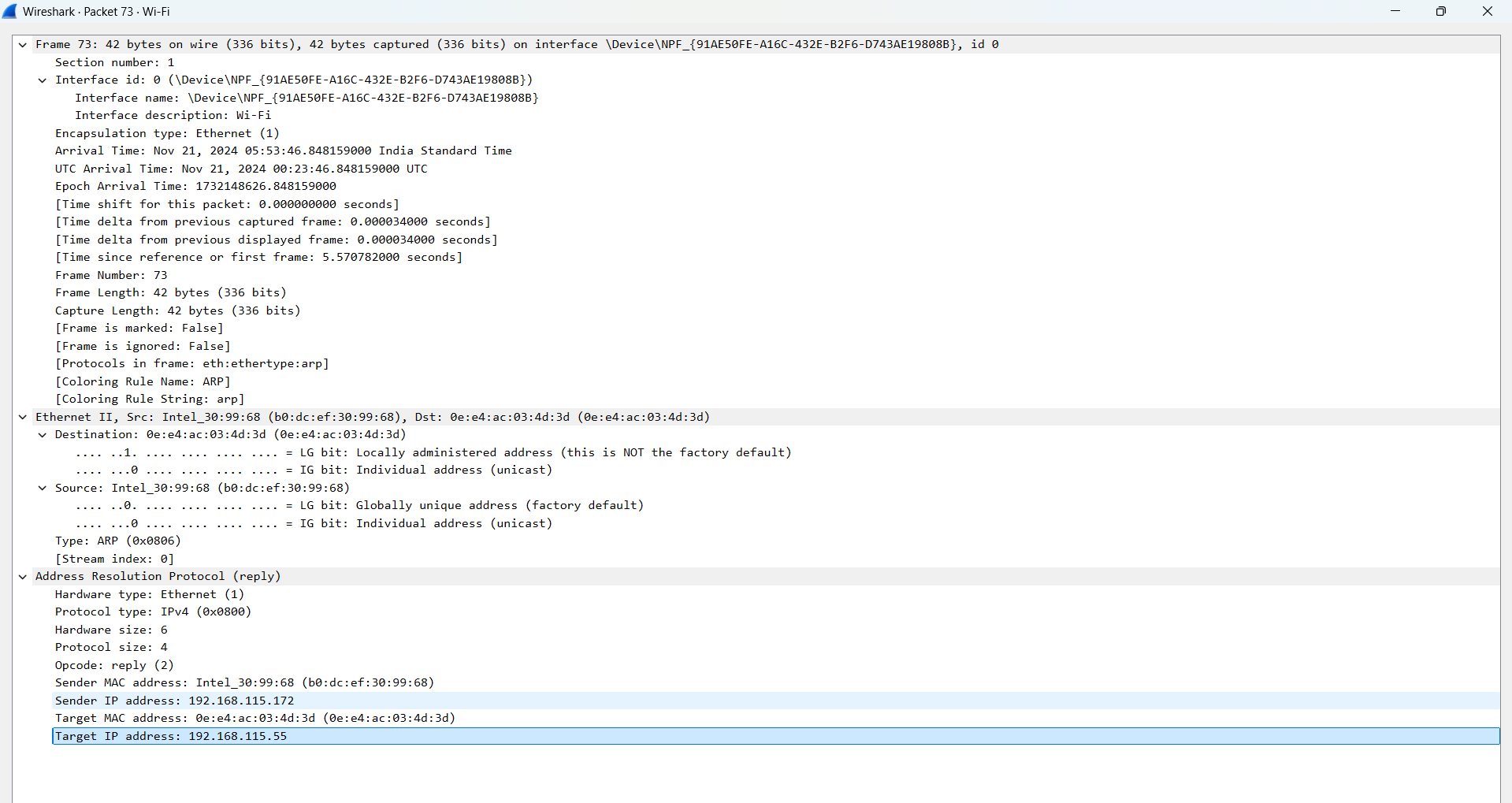
**Step – 7 :-**  Analysis of ARP Packet



* It is the timing details and frame length and frame no.

**Step – 8:-**  It is showing the source and destination IP Address:-

**Step – 9:-**  It is showing the TCP related details stored in the packets: like header section src and destination port no flags , checksum , length , timestamps.

****

 **Destination MAC Address**:

* 0e:e4:ac:03:4d:3d
* This is the MAC address of the device receiving the ARP reply.
* LG Bit indicates it's a **locally administered address**, not factory default.

 **Source MAC Address**:

* b0:dc:ef:30:99:68
* This is the MAC address of the device sending the ARP reply.
* IG Bit indicates it's an **individually assigned address**, meaning it's a unique address for this device.

 **Type**:

* 0x0806 (ARP)
* Indicates this is an ARP packet.

 **Hardware Type**:

* Ethernet (1)
* Specifies the hardware type for the network, in this case, Ethernet.

 **Protocol Type**:

* IPv4 (0x0800)
* Indicates the protocol type used, which is IPv4.

 **Hardware Size**:

* 6
* Represents the size (in bytes) of the hardware address (MAC address).

 **Protocol Size**:

* 4
* Represents the size (in bytes) of the protocol address (IPv4 address).

 **Opcode**:

* reply (2)
* Indicates that this is an ARP reply packet.

 **Sender MAC Address**:

* b0:dc:ef:30:99:68
* The MAC address of the sender (the device replying to the ARP request).

 **Sender IP Address**:

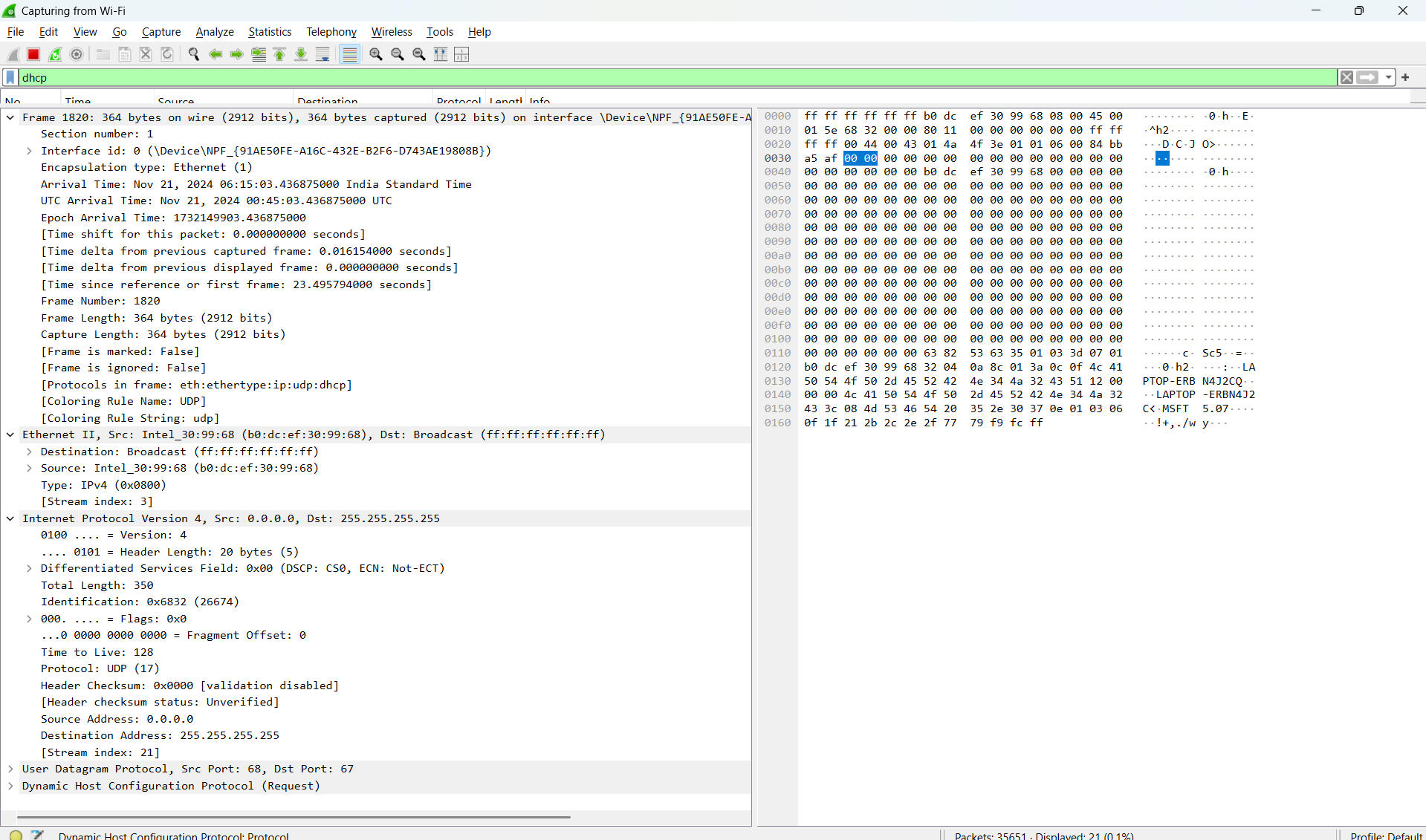
* 192.168.115.172
* The IP address associated with the sender's MAC address.

 **Target MAC Address**:

* 0e:e4:ac:03:4d:3d
* The MAC address of the device that sent the ARP request (target device).

 **Target IP Address**:

* 192.168.115.55
* The IP address of the target device that the ARP request was intended for.

**Step - 11:-**  now we will analyze the DHCP Packet.

**DHCP Packet Details**

1. Destination MAC Address:
   * Intel\_30:99:68 (b0:dc:ef:30:99:68)
   * The MAC address of the destination device (the DHCP client).
2. Source MAC Address:
   * HewlettPacka\_6e:03:5c (00:0b:86:6e:03:5c)
   * The MAC address of the source device (the DHCP server).
3. Type:
   * IPv4 (0x0800)
   * Indicates the packet is an IPv4 protocol.

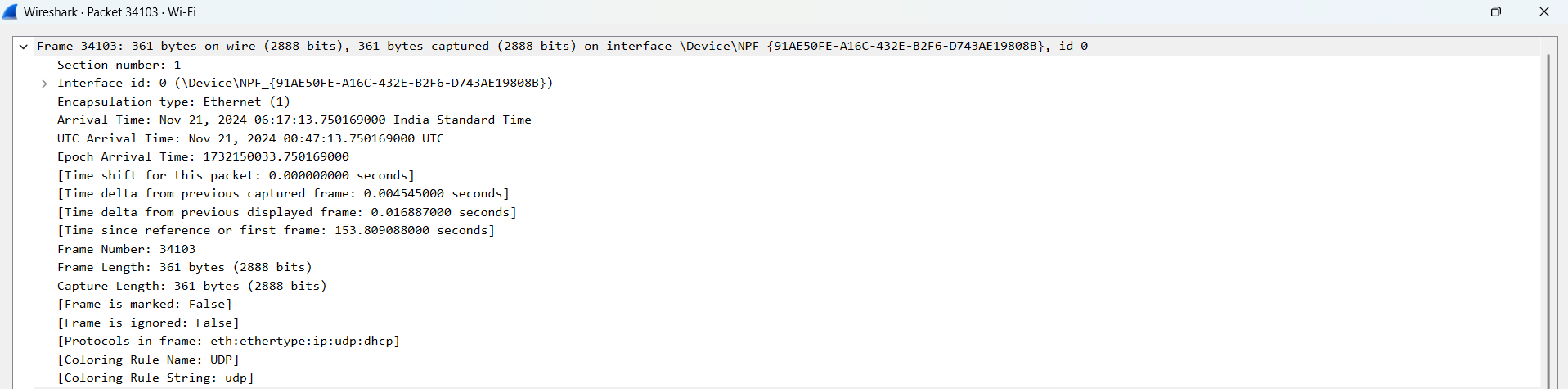
Internet Protocol Version 4 (IPv4)

1. Version:
   * 4
   * The version of the Internet Protocol, here IPv4.
2. Header Length:
   * 20 bytes
   * Length of the IPv4 header.
3. Differentiated Services Field:
   * 0x00
   * DSCP (Differentiated Services Code Point) and ECN (Explicit Congestion Notification) values.
4. Total Length:
   * 347
   * Total length of the IP packet, including headers and payload.
5. Identification:
   * 0x0000
   * Used for packet reassembly; all fragments of the same packet have the same identification.
6. Flags:
   * 0x0
   * Indicates fragmentation; here, no fragmentation is used.
7. Time to Live (TTL):
   * 128
   * Maximum hops the packet can take before being discarded.
8. Protocol:
   * UDP (17)
   * The transport protocol used.
9. Header Checksum:
   * 0x233d
   * Verifies the integrity of the IP header.
10. Source Address:
    * 10.140.0.4
    * The IP address of the DHCP server.
11. Destination Address:
    * 10.140.1.58
    * The IP address of the DHCP client.

User Datagram Protocol (UDP)

1. Source Port:
   * 67
   * The port used by the DHCP server.
2. Destination Port:
   * 68
   * The port used by the DHCP client.
3. Length:
   * 327
   * Length of the UDP header and payload.
4. Checksum:
   * Verifies the integrity of the UDP header and payload. Value not explicitly shown here.

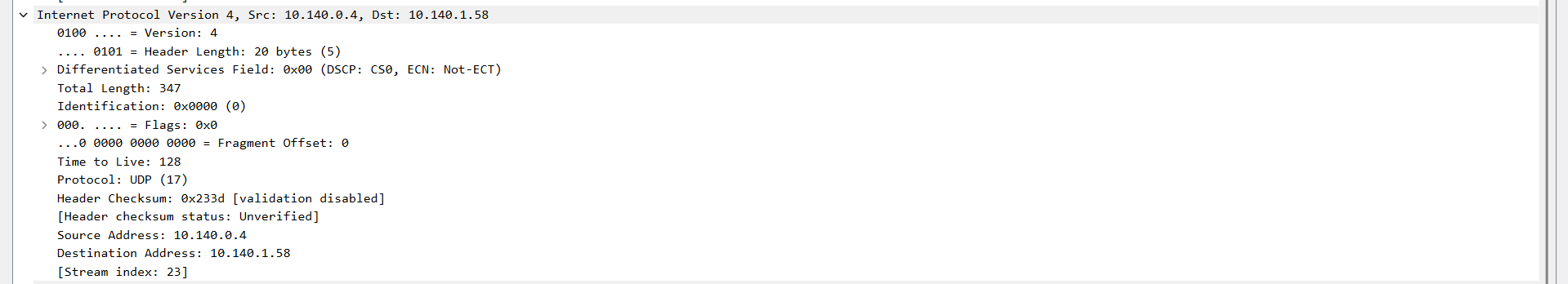
**Step - 12:-**  It us showing the timing related details of DHCP Packet.



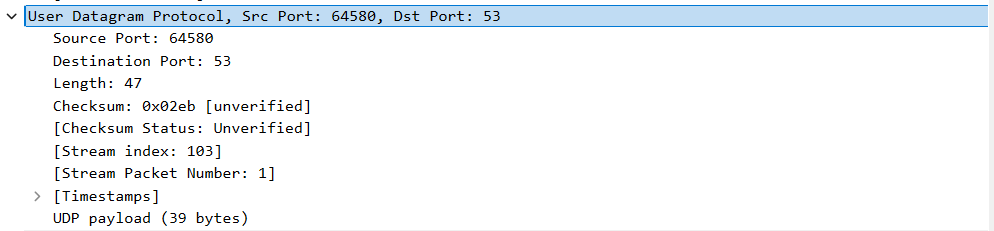
**Step - 13:-**  It is showing the ip related details of DHCP Packet.

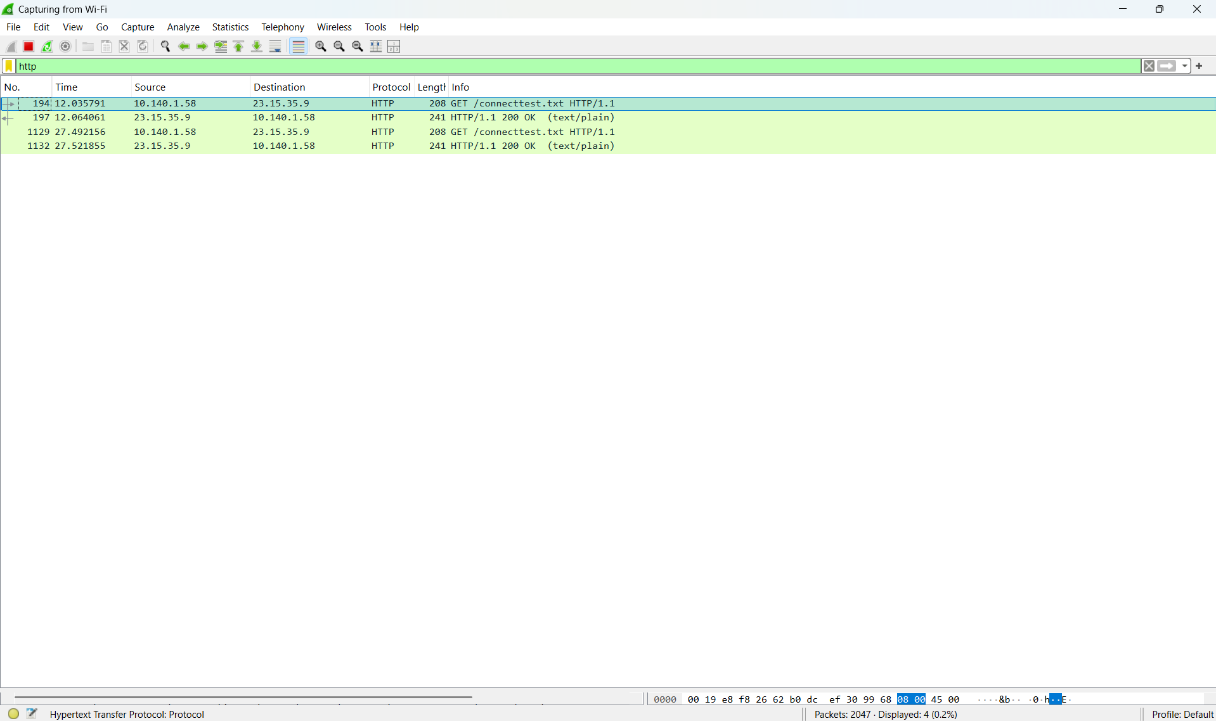
****

**Step - 14:-**  It is showing the details about the flags of DHCP Packet.

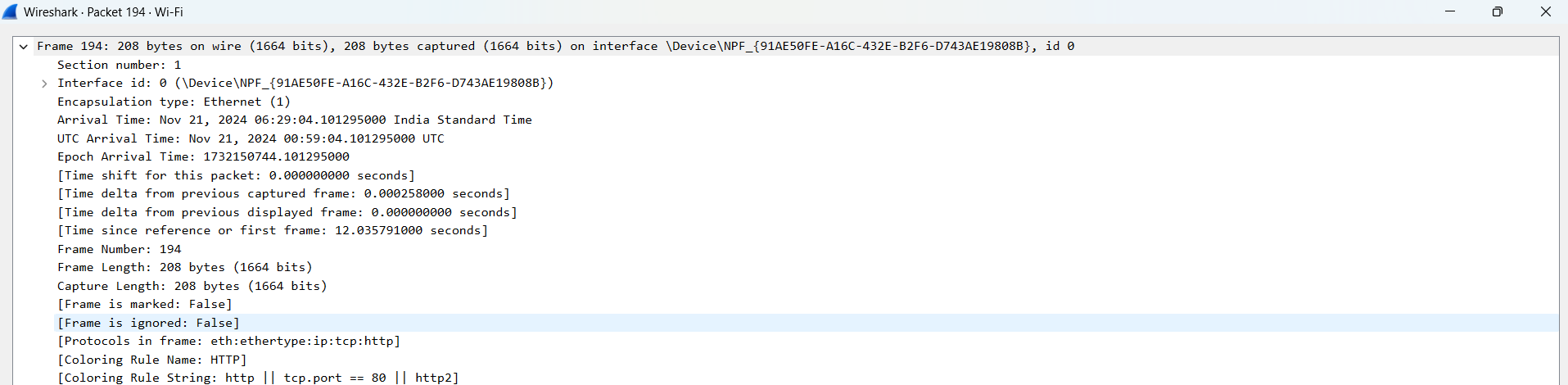
****

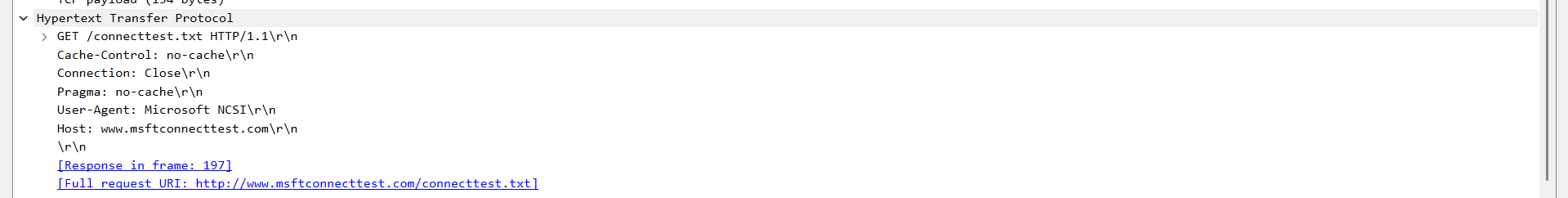
**Step - 15:-**  It is showing the details about the header of DHCP Packet.

****

**Step - 16:-**  Now we will analyze the HTTP Protocol.

**Step - 17:-**  These are the timing related details of http packet



**Step - 18:-**  these are the fields of http packets :-

* HTTP/1.1
* Version: This shows that the request and response are operating with the Hypertext Transfer Protocol version 1.1. This version brought along several enhancements of HTTP/1.0, such as persistent connection and pipelining.
* Cache-Control: no-cache
* Purposes: This header tells both the client and intermediate caches not to cache the response. It is applied on dynamic content when it is changing frequently or sensitive information that should not be cached.
* Connection: close
* Purpose: This header tells the client and server to close the connection after this request/response cycle. That is to say that further requests will need a new connection to be opened.
* Pragma: no-cache
* Purpose: Although this header is considered deprecated in HTTP/1.1, some clients may still make use of this. This has exactly the same effect as Cache-Control: no-cache.
* User-Agent: Microsoft MSI
* Purpose: This header indicates what client software is making the request. In this example, it's a part of Microsoft Installer (MSI).
* Host: www.msftconnecttest.com [invalid URL deleted]
* [middle]
* Purpose: This header names the domain name of the server that the client wishes to connect to.
* Response in frame: 197
* [middle]
* Purpose: This means that the response to the query is framed in frame number 197 of the captured packet.

**Conclusion:-**

By Performing this experiment I analyzed the packes of ARP , DHCP and HTTP.I seen how ARP translates IP addresses to MAC addresses, enabling devices to communicate within a local network .

I analyzed how device acquires IP address and other network configurations dynamically.I analzed Discover, Offer, Request, and Acknowledgement packets, and at the end Ianalyzed http request packet

understood how client-server communication operates in retrieving or sending data over the web.