CS 315: Computer Networks Lab
Spring 2024-25, IIT Dharwad
Assignment-11
Wireshark Lab: Ethernet and ARP
April 2, 2025

Chidurala Tejaswini
(220010012 / CS22BT012)

Part 0: Paste a screenshot of your system IP address, using ipconfig (on Windows) or ifconfig (on Mac and Linux), and fill out this Google form to submit the details of your system. The same system must be used to attempt all exercises of this lab.

```
user@sysad-HP-Elite-Tower-600-G9-Desktop-PC:~$ ifconfig
eno1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.240.118.97  netmask 255.255.248.0  broadcast 10.240.119.255
        inet6 fe80::1d6b:1bfb:2bd6:ef0d  prefixlen 64  scopeid 0x20<link>
        ether e0:73:e7:0a:99:9a  txqueuelen 1000  (Ethernet)
        RX packets 391173  bytes 254098205 (254.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 113219  bytes 14485817 (14.4 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  memory 0x80900000-80920000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 9065  bytes 931682 (931.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 9065  bytes 931682 (931.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

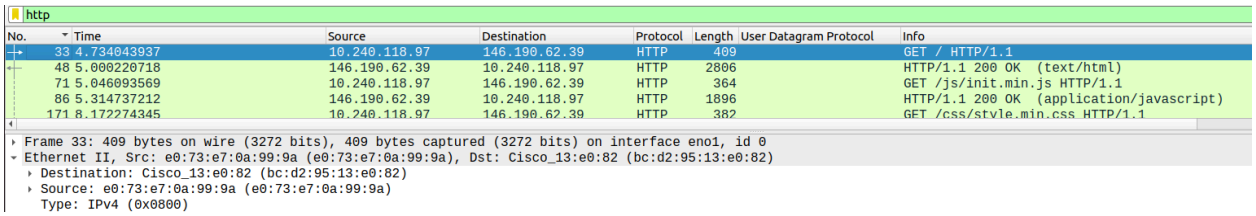
Part-1: Capturing and analyzing Ethernet frames

Let's begin by capturing a set of Ethernet frames to study. To do this, of course, you'll need access to a wired Ethernet connection for your system.

Do the following:

1. First, make sure your browser's cache of previously downloaded documents is empty.
2. Start up Wireshark and enter the following URL into your browser: `http://httpforever.com/`
3. Stop Wireshark packet capture.

Answer the following questions based on the Ethernet frame carrying the first HTTP GET request to the requested webpage:



1. **What is the 48-bit Ethernet address of your computer?**

   The 48-bit Ethernet address of my computer is source MAC address in the Ethernet Frame i.e `e0:73:e7:0a:99:9a`

2. **What is the 48-bit destination address in the Ethernet frame? Is this the Ethernet address of `httpforever.com`? What device has this as its Ethernet address?**
   - The destination MAC address is: `bc:d2:95:13:e0:82`
   - This is **not** the Ethernet address of `httpforever.com` because MAC addresses are only relevant within the local network (LAN).
   - This MAC address likely belongs to the **default gateway (router)**, which forwards packets to external networks.

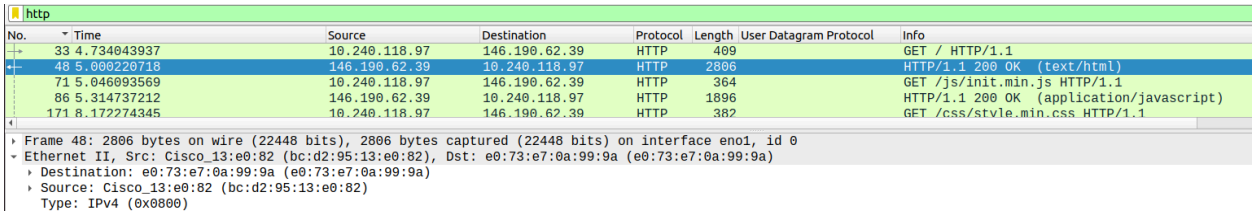
3. **What is the hexadecimal value for the two-byte Frame type field in the Ethernet frame carrying the HTTP GET request? Which network-layer protocol does this correspond to?**
   - The hexadecimal value for the two-byte Frame Type field is: `0x0800`
   - This corresponds to network layer protocol **IPv4 (Internet Protocol version 4).**

4. **What is the total size (in bytes) of the Ethernet frame encapsulating the HTTP GET request in Wireshark?**

   The total size of the Ethernet frame encapsulating the HTTP GET request is **409 bytes**(since the frame size is **409 bytes** in wireshark).

5. **Is the Ethernet frame carrying the first HTTP GET request transmitted as a unicast, multicast, or broadcast frame? How can this be determined from the destination MAC address?**
   - **Unicast**. The destination MAC address is **bc:d2:95:13:e0:82**, which is a specific, unique hardware address. This means the frame is being sent directly to a single device (likely a router or gateway).
   - Broadcast addresses have all bits set to 1 (**FF:FF:FF:FF:FF:FF**), and multicast addresses typically start with **01:00:5E** for IPv4. Since the destination address doesn't match either of these formats, it confirms that the frame is **unicast**.

Answer the following questions based on the Ethernet frame carrying the first HTTP response from the requested webpage:



6. **What is the value of the Ethernet source address? Is this the address of your computer, or `httpforever.com`? What device has this as its Ethernet address?**
    - The **source MAC address** is `bc:d2:95:13:e0:82`, as seen in the **Ethernet II section** of the frame.
    - This is **not** the MAC address of `httpforever.com` because MAC addresses are only relevant within a local network.
    - Instead, this MAC address belongs to the **default gateway (router)** that forwards packets between your computer and external servers like `httpforever.com`.

7. **What is the destination address in the Ethernet frame? Is this the Ethernet address of your computer?**
    - The **destination MAC address** is `e0:73:e7:0a:99:9a`.
    - Yes, this is the **MAC address of your computer** because it matches the previously identified address of our computer (previously identified in the GET request frame).

8. **How many bytes from the very start of the Ethernet frame does the ASCII "O" in "OK" appear? After how many bytes in the HTTP does the "O" in "OK" appear?**

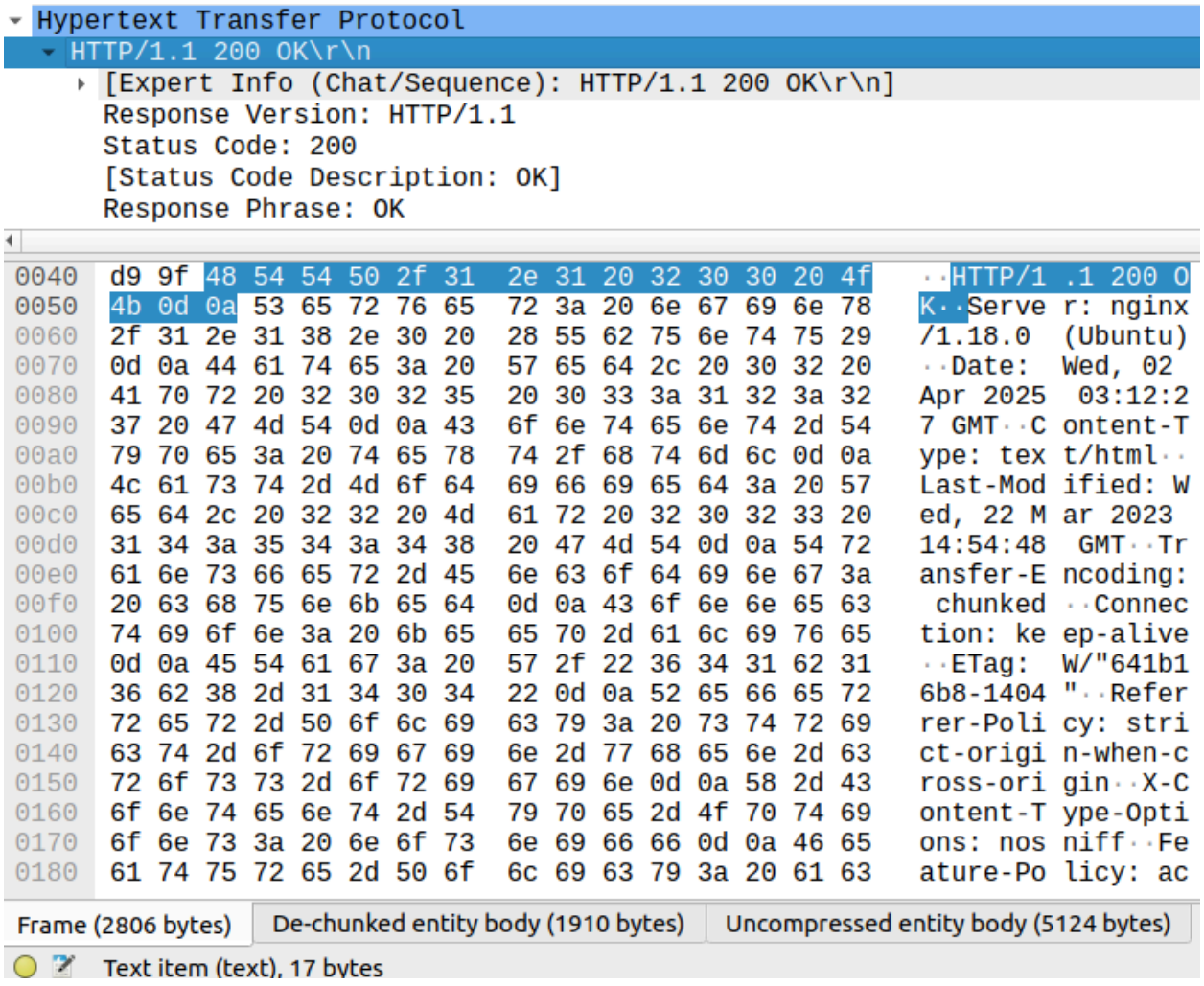
    The Ethernet frame consists of:

    - **Ethernet header**: 14 bytes
    - **IPv4 header**: 20 bytes
    - **TCP header**: 32 bytes
    - HTTP response headers:A typical HTTP response starts with a status line like this:**HTTP/1.1 200 OK** .Each character in the response is 1 byte in size. The breakdown is:

Part	Content	Byte Count
HTTP version	"HTTP/1.1"	9

Status Code	“200”	4
Status Message	“OK”	2

- The **ASCII "O" in "OK"** appears **after 13 bytes(9+4=13)** in the HTTP message from the start of the HTTP response.
- The ASCII "O" in "OK" appears **after 79 bytes(14+20+32+13=79)** from the start of the Ethernet frame.



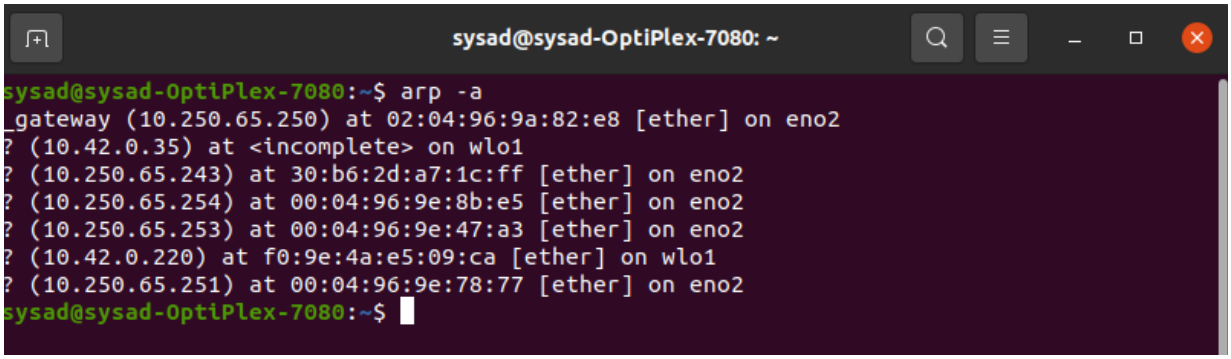
Part-2: The Address Resolution Protocol

In this section, we’ll observe the ARP protocol in action.

Recall that the ARP protocol typically maintains a cache of IP-to-Ethernet address translation pairs on your computer. The *arp* command (in both DOS, MacOS and Linux) is used to view and manipulate the contents of this cache. Since the `arp` command and the ARP protocol have the

same name, it's understandably easy to confuse them. But keep in mind that they are different - the `arp` command is used to view and manipulate the ARP cache contents, while the ARP protocol defines the format and meaning of the messages sent and received, and defines the actions taken on ARP message transmission and receipt.

Let's take a look at the contents of the ARP cache on your computer. In DOS, MacOS, and Linux, the "`arp -a`" command will display the contents of the ARP cache on your computer. So at the terminal, type "`arp -a`". The results of entering this command are shown in the Figure below.



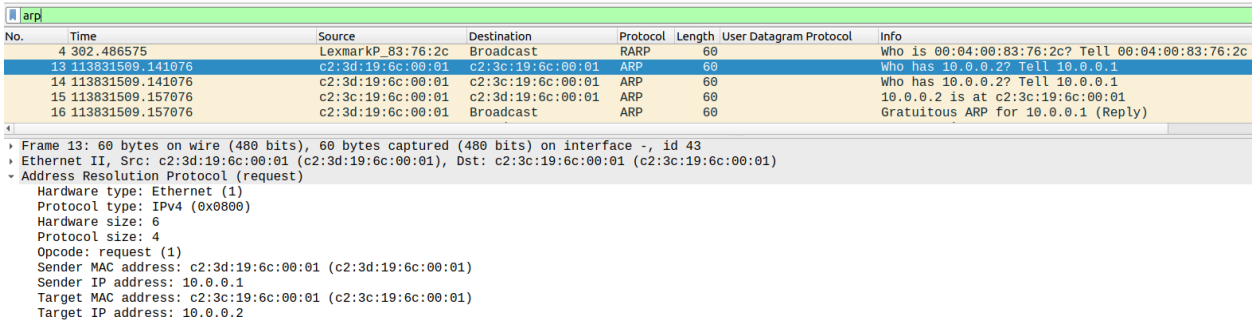
In order to observe your computer sending and receiving ARP messages, we'll need to clear the ARP cache, since otherwise your computer is likely to find a needed IP-Ethernet address translation pair in its cache and consequently not need to send out an ARP message.

Note: To delete the ARP cache on a Linux machine use the following command:

- `arp -a`: this lists all all the IP_Address.
- `sudo arp -d IP_Address`: You need to do this for all IPs in the above list.

Observing ARP in action

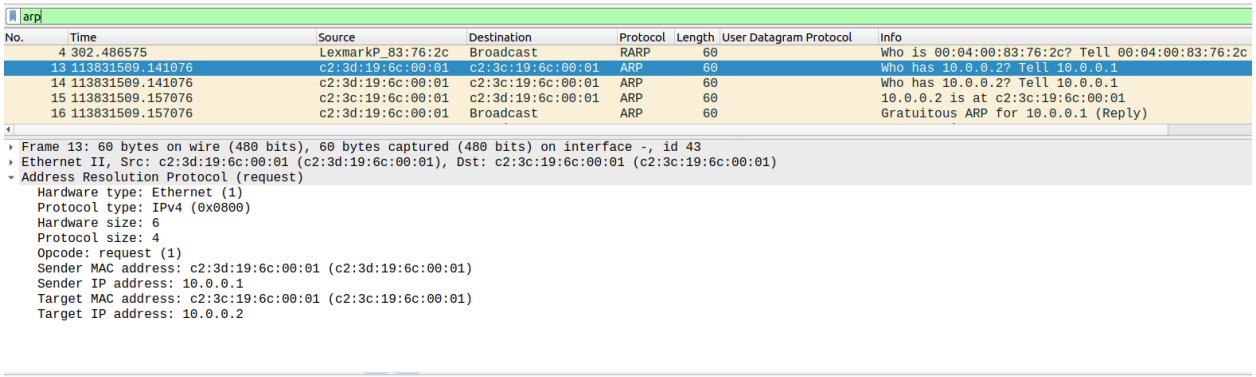
Answer the following questions based on the ARP_Trace.pcapng which was captured on an interface with IP Address 10.0.0.1:

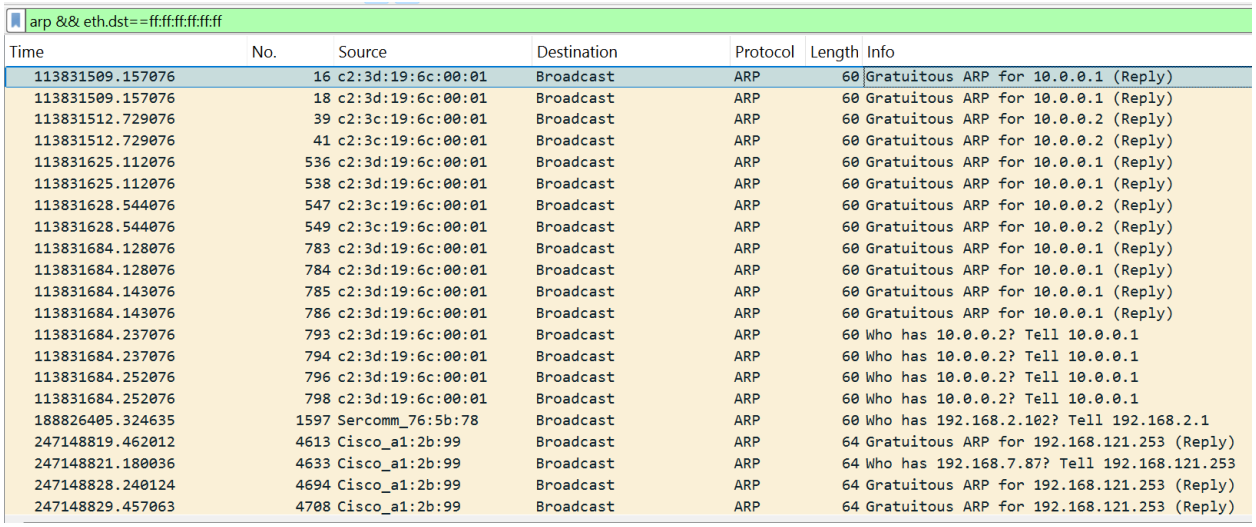


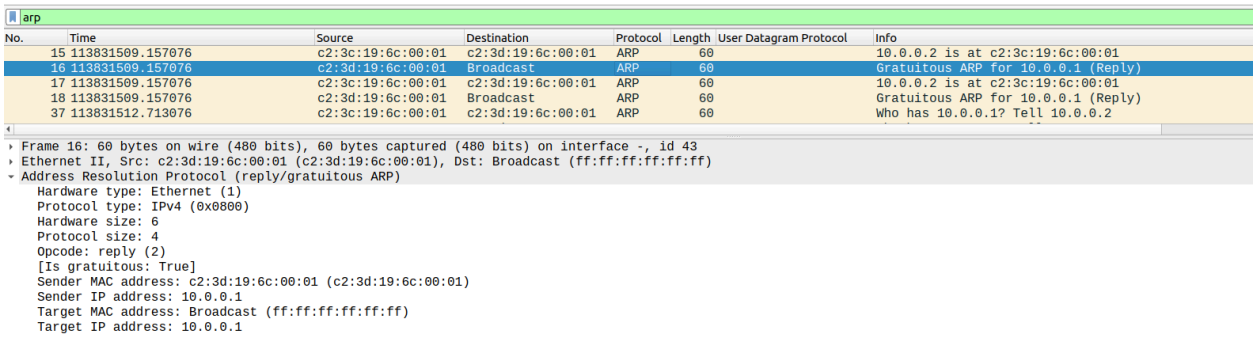
1. **State the sender's MAC address.**
    - The sender's MAC address in the ARP request (Frame 13) is c2:3d:19:6c:00:01.
    - This MAC address belongs to the system with IP 10.0.0.1.

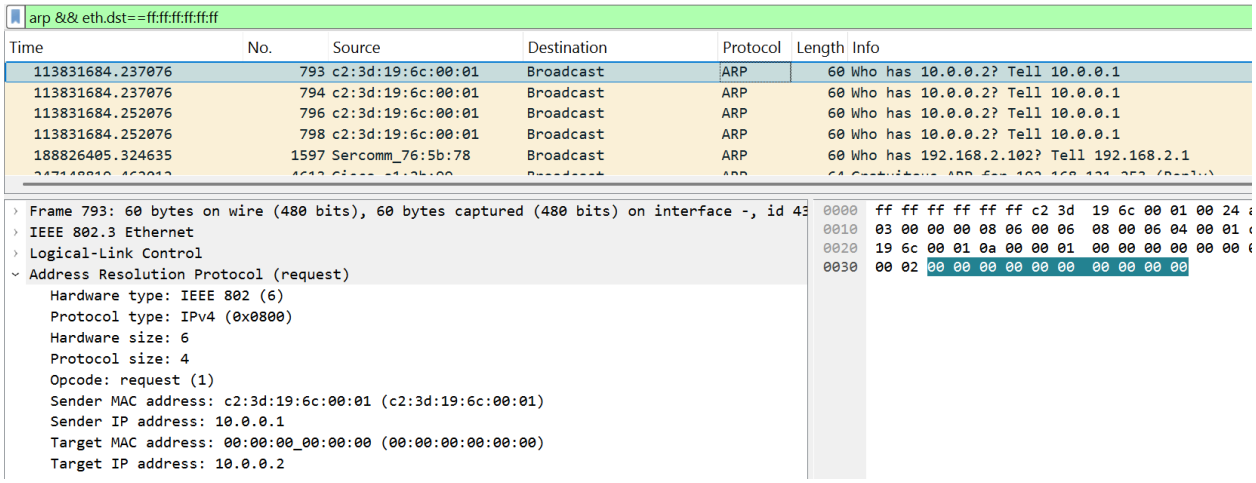
2. **Which target is the sender trying to connect to? Mention its IP and MAC addresses.**
    - **Target IP Address:** `10.0.0.2`
    - **Target MAC Address:** `c2:3c:19:6c:00:01`
    - **Explanation:** The sender at `10.0.0.1` is looking for the MAC address associated with `10.0.0.2`. Once the target responds, its MAC address is identified as `c2:3c:19:6c:00:01` (seen in Frame 15).

3. **At what point does broadcasting occur in the ARP trace? Explain the reason for broadcasting.**









Broadcasting in the ARP trace occurs mainly in the following two scenarios:

1. ARP Requests

- **When it occurs:**
  When a device wants to find the MAC address corresponding to a specific IP address (e.g., "Who has 10.0.0.2? Tell 10.0.0.1").
- **Example Frames:** Frame 13, 14, and 793 in the trace.
- **Mechanism:**
  The sender (e.g., 10.0.0.1) doesn't know the MAC address of the target IP (10.0.0.2), so it sends an ARP request to the broadcast MAC address: **ff:ff:ff:ff:ff:ff**.
- **Reason for Broadcasting:**
  Since the sender doesn't know which device on the network owns the target IP, it broadcasts the request so **all devices on the local network** receive the message and the correct device can respond.

2. Gratuitous ARP

- **When it occurs:**
  When a device sends an unsolicited ARP reply to announce its own IP-MAC mapping (e.g., "10.0.0.1 is at c2:3d:19:6c:00:01").

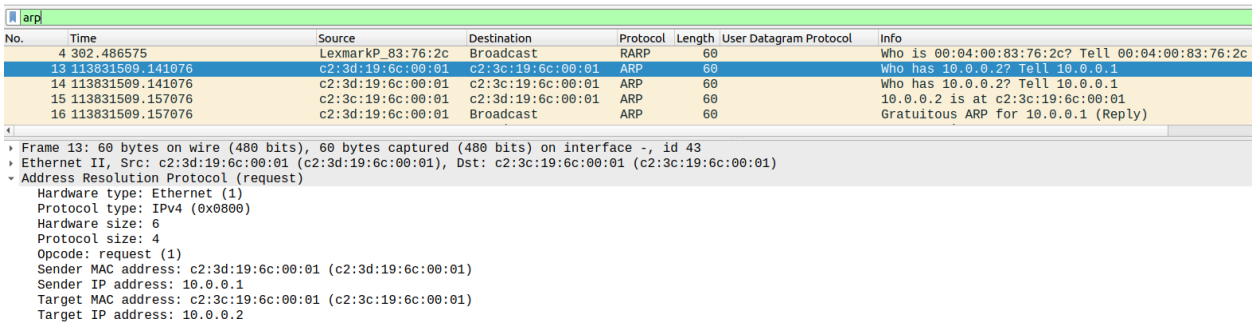
- **Mechanism:**
  The device sends a broadcast packet with the destination MAC as **ff:ff:ff:ff:ff:ff** even though no ARP request was made.
- **Reason for Broadcasting:**
  Gratuitous ARP is used to:
  - Announce or update its IP-MAC mapping across the network
  - Detect IP address conflicts
  - Inform switches of the device's presence for proper packet forwarding
  - Update other devices' ARP caches

Broadcasting in ARP occurs in:

- **ARP Requests:** When the sender doesn't know the MAC address of the target IP.
- **Gratuitous ARP:** When the sender announces its own IP-MAC mapping.

These broadcasts ensure that **all relevant devices on the local network** receive the necessary information to either respond or update their ARP tables.

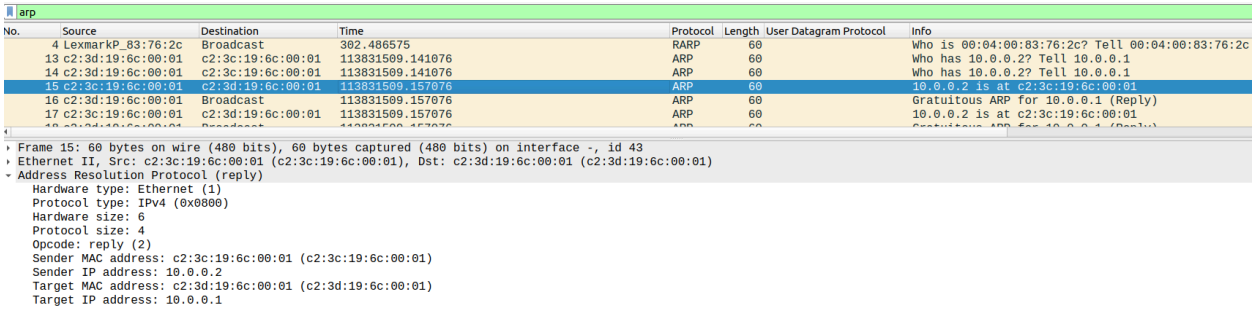
4. **List out all field values in the ARP request from the sender to the target.**



From **Frame 13 (ARP Request)**:

- **Hardware type**: Ethernet (1)
- **Protocol type**: IPv4 (0x0800)
- **Hardware size**: 6
- **Protocol size**: 4
- **Opcode**: Request (1)
- **Sender MAC address**: `c2:3d:19:6c:00:01`
- **Sender IP address**: `10.0.0.1`
- **Target MAC address**: `c2:3c:19:6c:00:01`
- **Target IP address**: `10.0.0.2`

5. **List out all field values in the ARP reply from the target to the sender.**



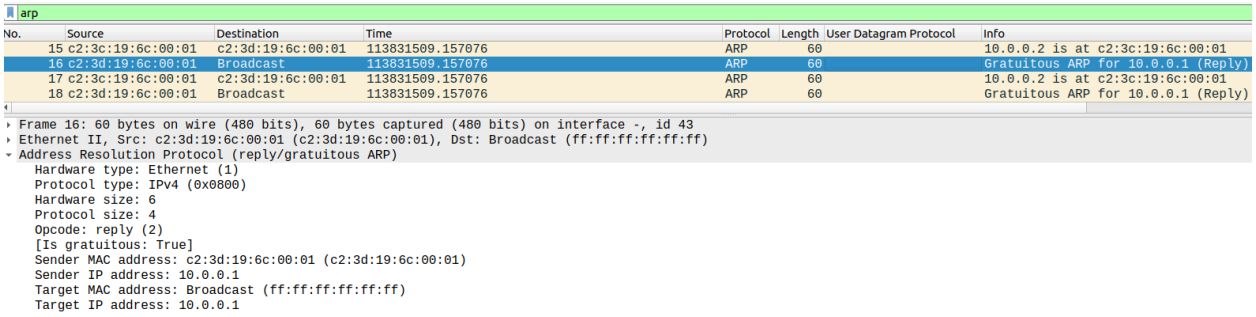
From **Frame 15 (ARP Reply)**:

- **Hardware type**: Ethernet (1)
- **Protocol type**: IPv4 (0x0800)
- **Hardware size**: 6
- **Protocol size**: 4
- **Opcode**: Reply (2)
- **Sender MAC address**: `c2:3c:19:6c:00:01`
- **Sender IP address**: `10.0.0.2`
- **Target MAC address**: `c2:3d:19:6c:00:01`
- **Target IP address**: `10.0.0.1`

6. **What are the differences between the ARP request and ARP reply field values?**
   - **ARP request:**The sender asks for a MAC address.
   - **ARP reply:**The target provides its MAC address.

Field	**ARP request**	**ARP reply**
Opcode	**Request(1)**	**Reply(1)**
Sender MAC address	MAC of the requesting host: **c2:3d:19:6c:00:01**	MAC of the target host: **c2:3c:19:6c:00:01**
Sender IP address	IP of the requesting host: **10.0.0.1**	IP of the target host: **10.0.0.2**
Target MAC address	MAC of the target host(since MAC Address of the target is known before already as ARP cache is not cleared ,Otherwise need to be broadcasted). **c2:3c:19:6c:00:01**	MAC Address of the sender(request): **c2:3d:19:6c:00:01**
Target IP address	IP of the target host: **10.0.0.2**	IP Address of the sender(request):**10.0.0.1**

7. **Explain the presence of a Gratuitous ARP packet in the trace. What is its purpose?**



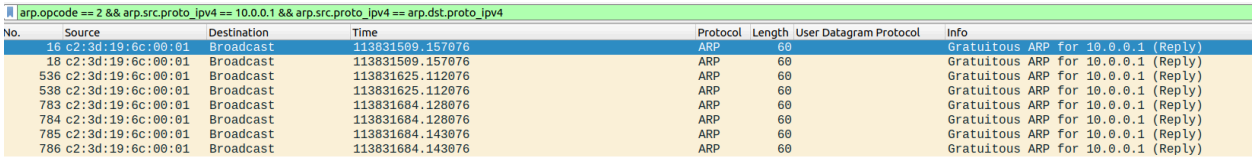
- The **Gratuitous ARP (GARP)** is seen in **Frame 16 ("Gratuitous ARP for 10.0.0.1 (Reply)")**.They are broadcasted and typically contain the sender's own IP as both sender and target.
- **Purpose**:
    - To **announce or update** a device's IP-to-MAC mapping to the entire network.
    - It helps to detect **IP conflicts**, ensuring no two devices use the same IP.
    - It allows devices like switches to update their ARP caches.

8. **What is the significance of the IP and MAC addresses in the Gratuitous ARP packet?**
    - **Sender IP and Target IP** are the **same** (`10.0.0.1`), meaning the sender is announcing its own address.
    - **Sender MAC address** is `c2:3d:19:6c:00:01`, and the **Target MAC address** is `ff:ff:ff:ff:ff:ff` (broadcast), meaning it informs all devices about its presence.

    Significance:

        - It tells all network devices that `10.0.0.1` is at `c2:3d:19:6c:00:01`.
        - Ensures that **no other device** on the network is using `10.0.0.1`.

9. **How many Gratuitous ARP packets are present in the trace corresponding to the sender's IP Address? Provide the packet number(s).**



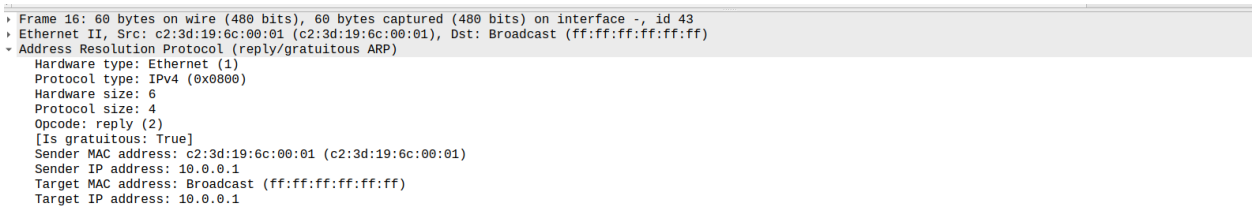
Filter type :

arp.opcode == 2 && arp.src.proto_ipv4 == 10.0.0.1 && arp.src.proto_ipv4 == arp.dst.proto_ipv4

There are **8 Gratuitous ARP packets** for `10.0.0.1`:

- **Frame 16**: "Gratuitous ARP for 10.0.0.1 (Reply)"
- **Frame 18**: "Gratuitous ARP for 10.0.0.1 (Reply)"
- **Frame 536**: "Gratuitous ARP for 10.0.0.1 (Reply)"
- **Frame 538**: "Gratuitous ARP for 10.0.0.1 (Reply)"
- **Frame 783**: "Gratuitous ARP for 10.0.0.1 (Reply)"
- **Frame 784**: "Gratuitous ARP for 10.0.0.1 (Reply)"
- **Frame 785**: "Gratuitous ARP for 10.0.0.1 (Reply)"
- **Frame 786:** "Gratuitous ARP for 10.0.0.1 (Reply)"

10. What is the sender and target MAC address in the Gratuitous ARP packet?



In the Gratuitous ARP packet (Frame 16):

- **Sender MAC address**: `c2:3d:19:6c:00:01`
- **Target MAC address**: `ff:ff:ff:ff:ff:ff` (Broadcast)

Submission Details

- Write your answers in a single doc/tex file, and submit its PDF named after your IIT Dharwad roll number, which contains all answers (with screenshots, if necessary).