BPA-KOM Lab 1: Address Resolution Protocol (ARP)

Name	Dave Galea
VUT ID	284844
Lab Number	1
Date	October 2025

---

1. Objective 1

Task Assignment:
The aim of this task was to examine the ARP table on the local computer.

Solution:
To view the ARP table, the command `arp -a` was used. Upon execution, this was the result:

```
C:\Users\Student>arp -a

Interface: 192.168.99.1 --- 0x4
  Internet Address      Physical Address      Type
  192.168.99.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static

Interface: 192.168.31.1 --- 0x6
  Internet Address      Physical Address      Type
  192.168.31.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 192.168.229.1 --- 0x8
  Internet Address      Physical Address      Type
  192.168.229.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 192.168.37.123 --- 0xe
  Internet Address      Physical Address      Type
  192.168.37.1          b8-69-f4-9f-9e-09     dynamic
  192.168.37.119        10-7b-44-15-bb-2f     dynamic
  192.168.37.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 192.168.88.254 --- 0x12
  Internet Address      Physical Address      Type
  192.168.88.1          78-9a-18-1d-a7-ee     dynamic
  192.168.88.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Figure 1: ARP table of the local computer.

Ethernet Adapter Public, identified through execution of the ipconfig command, is identified by IP 192.168.37.123 (the fourth interface). The presence of the dynamic record in the table is indicative that the computer has communicated with the router.

The static record with *ff-ff-ff-ff-ff-ff* is used as a broadcast MAC address; it is used to send data to all devices on a local network.

---

2. Objective 2

Task Assignment:
The aim of this task was to capture and analyze the ARP communication between two computers in Wireshark.

Solution:
First, `ipconfig /all` was executed to give detailed information about the network configuration of the local devices.

```
C:\Users\Student>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : PC-SC5037-23
   Primary Dns Suffix  . . . . . . . : utko.feec.vutbr.cz
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : utko.feec.vutbr.cz
                                       feec.vutbr.cz

Ethernet adapter Public:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) Ethernet Connection (2) I219-V
   Physical Address. . . . . . . . . : 10-7B-44-15-C0-AF
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::4d90:7ab8:ca08:3f59%14(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.37.123(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : středa 15. října 2025 10:02:46
   Lease Expires . . . . . . . . . . : středa 15. října 2025 10:37:41
   Default Gateway . . . . . . . . . : 192.168.37.1
   DHCP Server . . . . . . . . . . . : 192.168.37.1
   DHCPv6 IAID . . . . . . . . . . . : 84966212
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-30-59-E7-64-10-7B-44-15-C0-AF
   DNS Servers . . . . . . . . . . . : 8.8.8.8
                                       147.229.71.10
                                       208.67.222.222
   NetBIOS over Tcpip. . . . . . . . : Enabled

Ethernet adapter LOCAL-Mikrotik:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Realtek PCIe GbE Family Controller #2
   Physical Address. . . . . . . . . : 00-E0-72-54-45-BD
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::ccec:fc75:e12f:6aea%18(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.88.254(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : středa 15. října 2025 10:02:44
   Lease Expires . . . . . . . . . . : středa 15. října 2025 10:47:37
   Default Gateway . . . . . . . . . : 192.168.88.1
   DHCP Server . . . . . . . . . . . : 192.168.88.1
   DHCPv6 IAID . . . . . . . . . . . : 302047346
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-30-59-E7-64-10-7B-44-15-C0-AF
   DNS Servers . . . . . . . . . . . : 192.168.88.1
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Figure 2: Output of ipconfig /all (first part).

```
Ethernet adapter Ethernet 4:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : VirtualBox Host-Only Ethernet Adapter
   Physical Address. . . . . . . . . : 0A-00-27-00-00-04
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::117b:1ebc:a592:12e2%4(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.99.1(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
   DHCPv6 IAID . . . . . . . . . . . : 487194663
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-30-59-E7-64-10-7B-44-15-C0-AF
   NetBIOS over Tcpip. . . . . . . . : Enabled

Ethernet adapter VMware Network Adapter VMnet1:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : VMware Virtual Ethernet Adapter for VMnet1
   Physical Address. . . . . . . . . : 00-50-56-C0-00-01
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::a9e6:aede:e50:7f11%8(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.229.1(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : středa 15. října 2025 10:02:30
   Lease Expires . . . . . . . . . . : středa 15. října 2025 10:47:30
   Default Gateway . . . . . . . . . :
   DHCP Server . . . . . . . . . . . : 192.168.229.254
   DHCPv6 IAID . . . . . . . . . . . : 553668694
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-30-59-E7-64-10-7B-44-15-C0-AF
   NetBIOS over Tcpip. . . . . . . . : Enabled

Ethernet adapter VMware Network Adapter VMnet8:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : VMware Virtual Ethernet Adapter for VMnet8
   Physical Address. . . . . . . . . : 00-50-56-C0-00-08
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::9f1f:a44e:1c55:304e%6(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.31.1(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : středa 15. října 2025 10:02:33
   Lease Expires . . . . . . . . . . : středa 15. října 2025 10:47:30
   Default Gateway . . . . . . . . . :
   DHCP Server . . . . . . . . . . . : 192.168.31.254
   DHCPv6 IAID . . . . . . . . . . . : 570445910
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-30-59-E7-64-10-7B-44-15-C0-AF
   Primary WINS Server . . . . . . . : 192.168.31.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Figure 3: Output of ipconfig /all (continued).

The Ethernet interface of interest had the following:
IPv4 Address: `192.168.37.123`
Physical Address: `10-7B-44-15-C0-AF`

Then, in Wireshark, the arp filter was applied.

arp

No.	Time	Source	Destination	Protocol	Length	Info
3655	199.799073	ASUSTekCOMPU_15:bc:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.113
3673	201.936476	ASUSTekCOMPU_15:bb:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.120
3726	205.970788	ASUSTekCOMPU_15:bd:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.100
3727	206.958229	ASUSTekCOMPU_15:ba:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.121
3763	210.636757	ASUSTekCOMPU_15:c0:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.108
3768	210.902529	ASUSTekCOMPU_15:bb:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.104
3784	212.535282	ASUSTekCOMPU_15:ba:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.110
3798	214.013666	ASUSTekCOMPU_d3:ca:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.103
3822	218.784876	ASUSTekCOMPU_15:bf:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.101
3838	219.814257	ASUSTekCOMPU_15:bc:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.113
3839	220.621027	ASUSTekCOMPU_d3:ca:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.105
3855	222.002196	ASUSTekCOMPU_15:bb:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.114

Figure 4: Wireshark output after applying the arp protocol filter.

Then, one of two people generated an ARP request to obtain the MAC address of the

other by using ping.

```
C:\Windows\System32>ping 192.168.37.124

Pinging 192.168.37.124 with 32 bytes of data:
Reply from 192.168.37.124: bytes=32 time=1ms TTL=128
Reply from 192.168.37.124: bytes=32 time=2ms TTL=128
Reply from 192.168.37.124: bytes=32 time=2ms TTL=128
Reply from 192.168.37.124: bytes=32 time=2ms TTL=128

Ping statistics for 192.168.37.124:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 2ms, Average = 1ms
```

Figure 5: Ping generated from the command line.

The result in Wireshark:

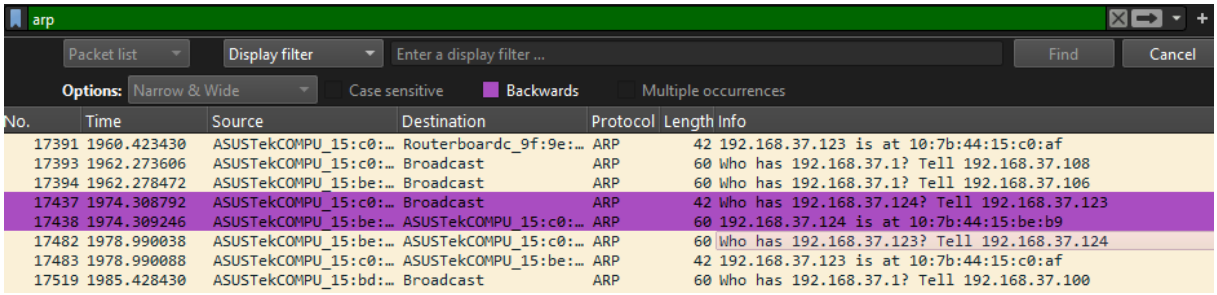


Figure 6: The result of the ping displayed in Wireshark.

In the next section, both the ARP request packet and ARP response packet were examined.

ARP request

Upon selecting the request packet details pane and expanding Ethernet II and Address Resolution Protocol:

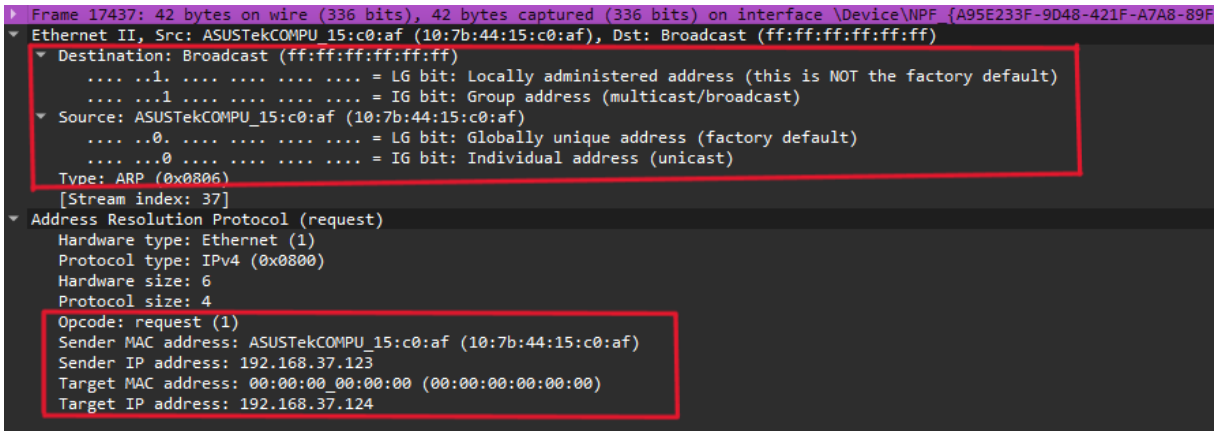


Figure 7: Request packet details.

One can note that the Target MAC address, the address of who the data is intended for, is `00:00:00:00:00:00`; this is because it is still unknown; it will be known after the ARP response. The Destination MAC address, the address of where the frame is physically meant to go, on the other hand, is `ff:ff:ff:ff:ff:ff`; because it is being broadcast to all devices on the LAN.

ARP Response

Upon selecting the response packet details pane and expanding Ethernet II and Address Resolution Protocol:

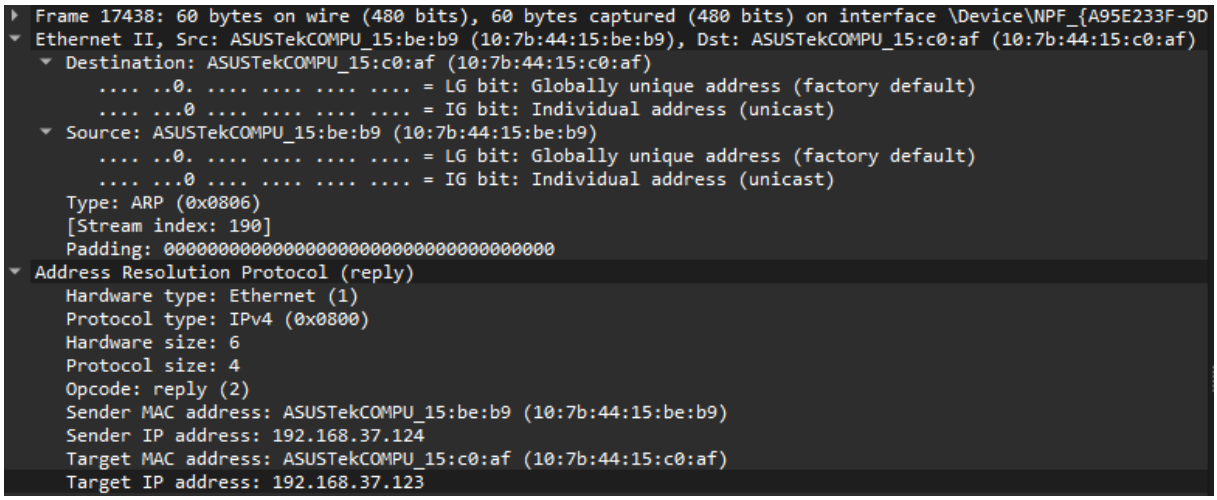


Figure 8: Response packet details.

It can be noted that now, both the Target and Destination MAC addresses are the same, because the packet no longer needs to be flooded, and the packet can be sent directly.

3. Objective 3

Task Assignment:
The aim of this task was to add the IP and MAC address of a colleague to the ARP table using the static method, and then examine the communication in Wireshark.

Solution:
The contents of the ARP table were re-displayed.

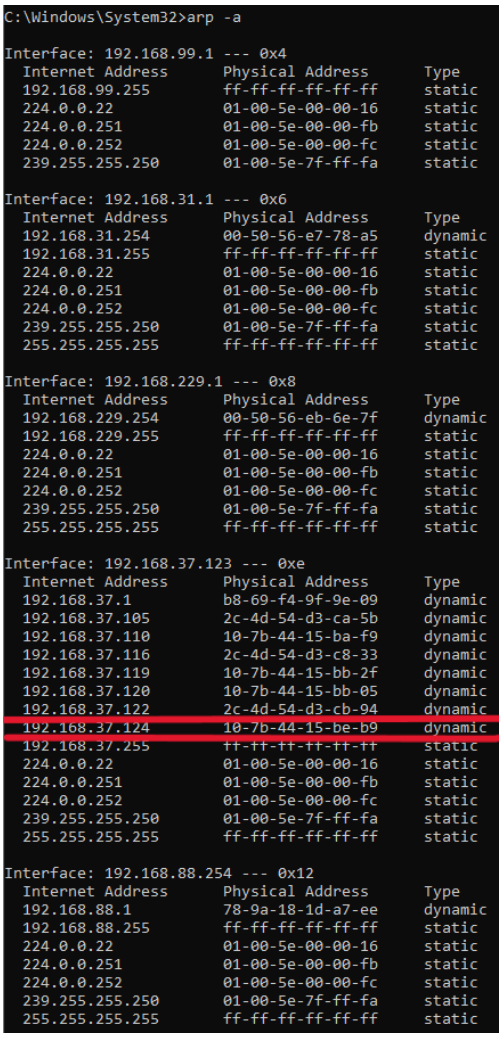


Figure 9: Output of the redisplayed ARP table after previous actions.

The expected record; that of my colleague:

- IP: `192.168.37.124`
- MAC: `10-7B-44-15-BE-B9`

was present in the table as a **dynamic** record.

Then, after removing the record using `arp -d <IP>`, the ARP table printed as follows:

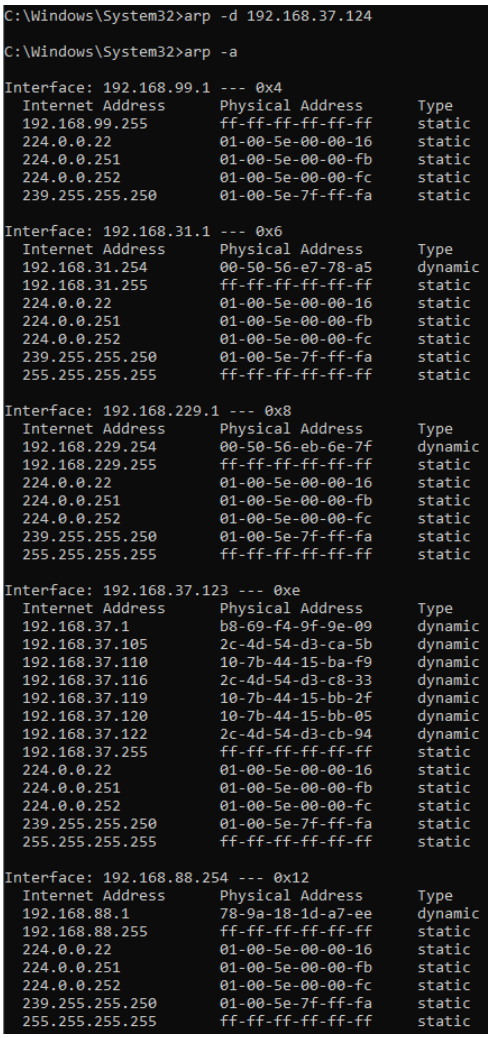


Figure 10: The ARP table after deleting the record.

It can be noted that my colleague's record is no longer present.

Then record was re-added manually. This was done via `arp -s <IP> <MAC>`, with both IP and MAC being of my colleague. The result is a static record.

The `ping` command was then used to test the availability of my colleague.

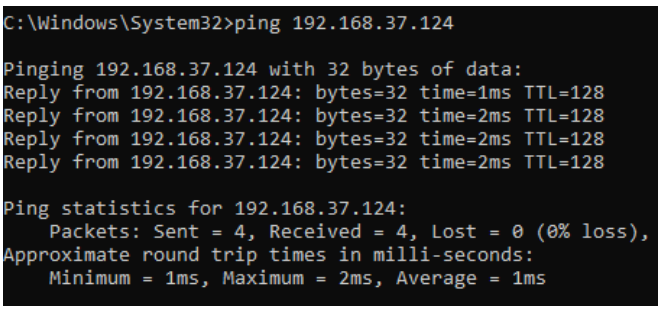
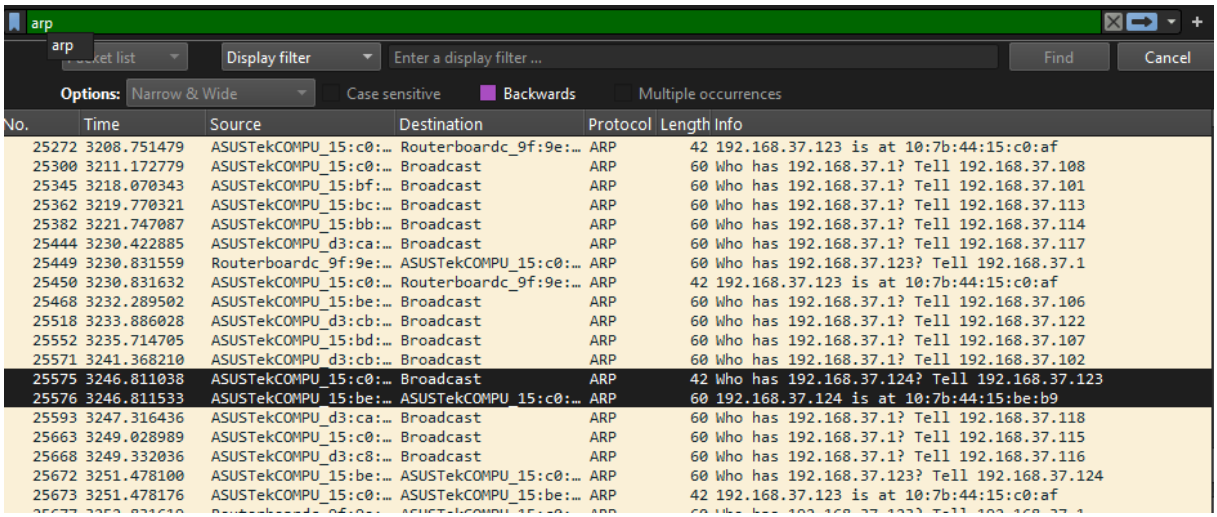


Figure 11: The ping command executed in the command line.

After pinging, Wireshark was tested to see if any new ARP packets were captured. In theory, no new ARP packets should be captured, instead, the computer uses the pre-configured static mapping and bypasses the ARP process. However, in our case, new packets were still captured as shown below.



No.	Time	Source	Destination	Protocol	Length	Info
25272	3208.751479	ASUSTekCOMPU_15:c0:…	Routerboardc_9f:9e:…	ARP	42	192.168.37.123 is at 10:7b:44:15:c0:af
25300	3211.172779	ASUSTekCOMPU_15:c0:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.108
25345	3218.070343	ASUSTekCOMPU_15:bf:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.101
25362	3219.770321	ASUSTekCOMPU_15:bc:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.113
25382	3221.747087	ASUSTekCOMPU_15:bb:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.114
25444	3230.422885	ASUSTekCOMPU_d3:ca:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.117
25449	3230.831559	Routerboardc_9f:9e:…	ASUSTekCOMPU_15:c0:…	ARP	60	Who has 192.168.37.123? Tell 192.168.37.1
25450	3230.831632	ASUSTekCOMPU_15:c0:…	Routerboardc_9f:9e:…	ARP	42	192.168.37.123 is at 10:7b:44:15:c0:af
25468	3232.289502	ASUSTekCOMPU_15:be:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.106
25518	3233.886028	ASUSTekCOMPU_d3:cb:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.122
25552	3235.714705	ASUSTekCOMPU_15:bd:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.107
25571	3241.368210	ASUSTekCOMPU_d3:cb:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.102
25575	3246.811038	ASUSTekCOMPU_15:c0:…	Broadcast	ARP	42	Who has 192.168.37.124? Tell 192.168.37.123
25576	3246.811533	ASUSTekCOMPU_15:be:…	ASUSTekCOMPU_15:c0:…	ARP	60	192.168.37.124 is at 10:7b:44:15:be:b9
25593	3247.316436	ASUSTekCOMPU_d3:ca:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.118
25663	3249.028989	ASUSTekCOMPU_15:c0:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.115
25668	3249.332036	ASUSTekCOMPU_d3:c8:…	Broadcast	ARP	60	Who has 192.168.37.1? Tell 192.168.37.116
25672	3251.478100	ASUSTekCOMPU_15:be:…	ASUSTekCOMPU_15:c0:…	ARP	60	Who has 192.168.37.123? Tell 192.168.37.124
25673	3251.478176	ASUSTekCOMPU_15:c0:…	ASUSTekCOMPU_15:be:…	ARP	42	192.168.37.123 is at 10:7b:44:15:c0:af

Figure 12: Output in Wireshark after pinging.

4. Objective 4

Task Assignment:
The aim of this task was to display the graph of captured packets in Wireshark.

Solution:
In Wireshark, captured packets can be displayed in a graph. After setting the filter to arp, setting the interval to 1 second, and measuring bytes, the following graph was produced:

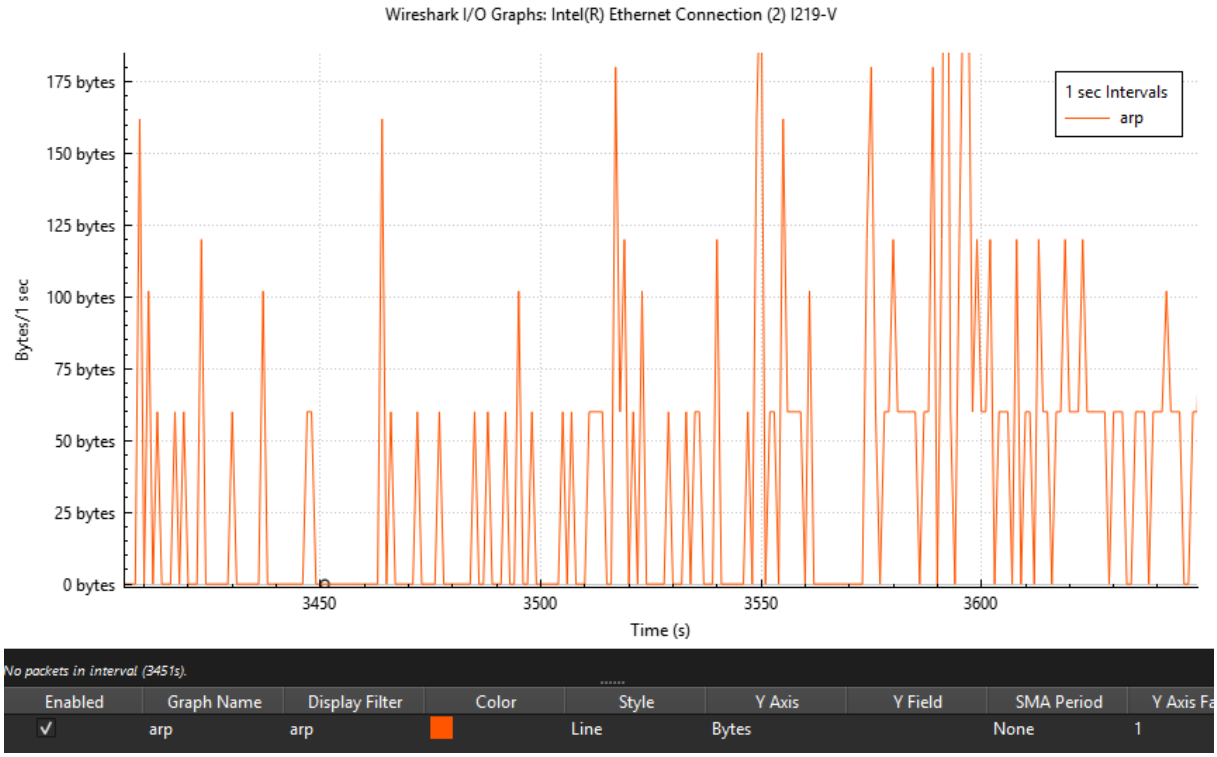


Figure 13: Graph of Bytes captured over time.

It can be noted that many peaks are at 102 bytes; these represent the ARP request. The peaks are 102 bytes because as seen in wireshark; the length of the request packet is 60 bytes, and the length of the response packet is 42 bytes, making a total of 102 bytes in an ARP communication.

The graph can also be altered to display packets instead of bytes:

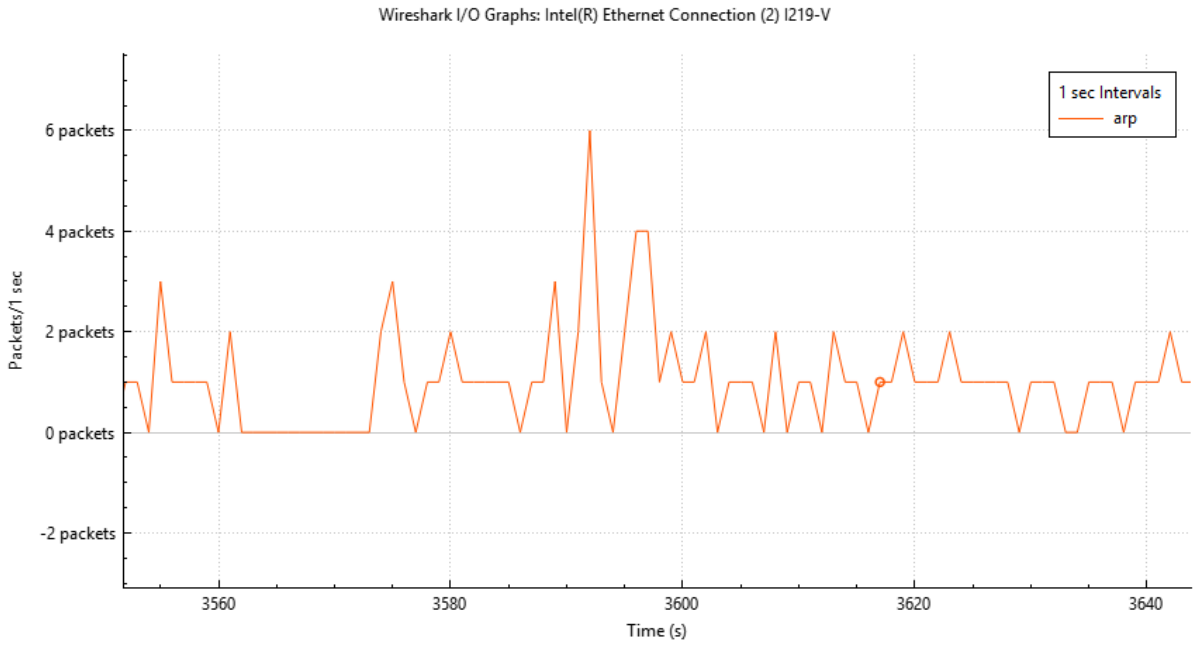


Figure 14: Graph of packets captured over time.

As confirmed by the bytes and the packets graphs, 2 packets are transferred per ARP communication.

---

5. Objective 5

Task Assignment:
The aim of this task was to create a topology in Cisco Packet Tracer.

Solution:
To create the topology, the switch 2960 was connected via Copper Straight-Through cable to 4 PCs. The Fast Ethernet ports were used for connection.

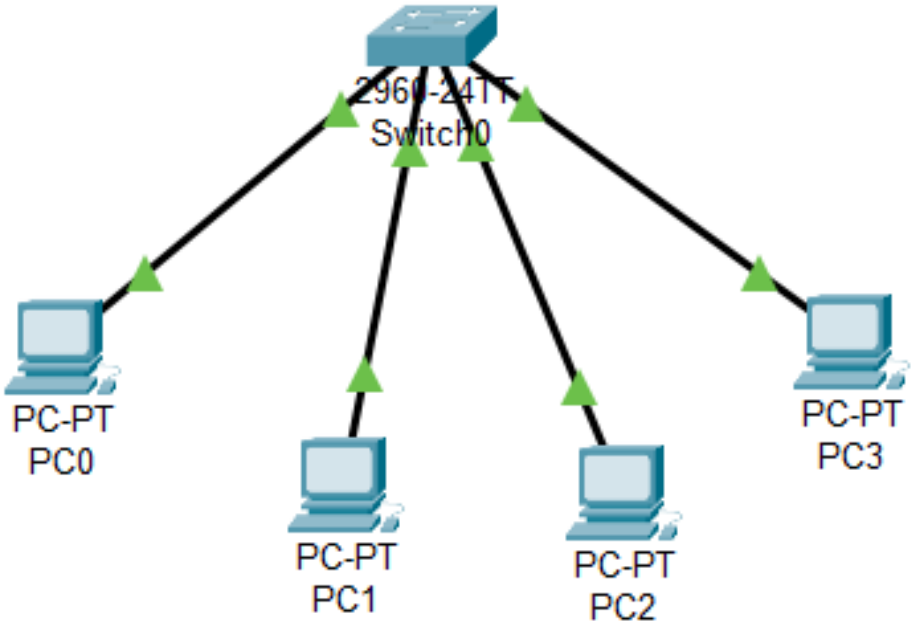


Figure 15: Topology created in Cisco Packet Tracer.

---

6. Objective 6

Task Assignment:
The aim of this task was to generate and examine the ARP communication in a Packet Tracer Simulation. Additionally, the switch's MAC table was explored.

Solution:
After issuing the `ipconfig /all` command on each PC through the command prompt, the following table could be generated:

Device	IP Address	MAC Address
PC0	192.168.1.1	0060.3ECE.6DDC
PC1	192.168.1.2	00E0.F9A5.E89D
PC2	192.168.1.3	0002.17C3.D4E3
PC3	192.168.1.4	000C.85AD.D769

Table 1: Device IP and MAC Address Table

Then, Simulation Mode was enabled, and filters were applied to display ARP. Before generating an ARP communication between PC0 and PC2, their respective ARP tables were checked to verify that they are empty.

From PC0, a ping was generated via `192.168.1.3`. This triggered a new event in the Event List, as shown below.

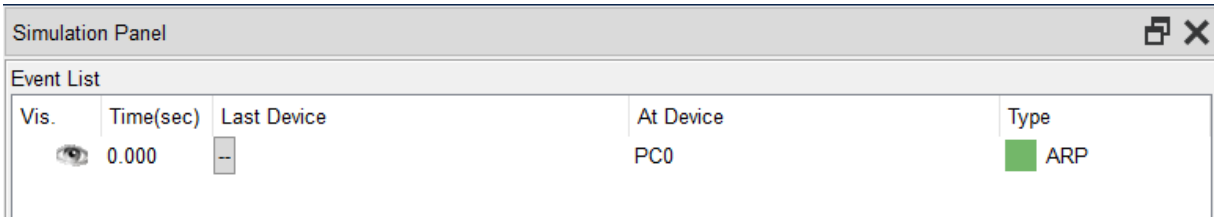


Figure 16: A new event displayed in the Event List.

Upon clicking this event, the OSI model is displayed, and it can be noted that only the lowest 2 layers contain data.

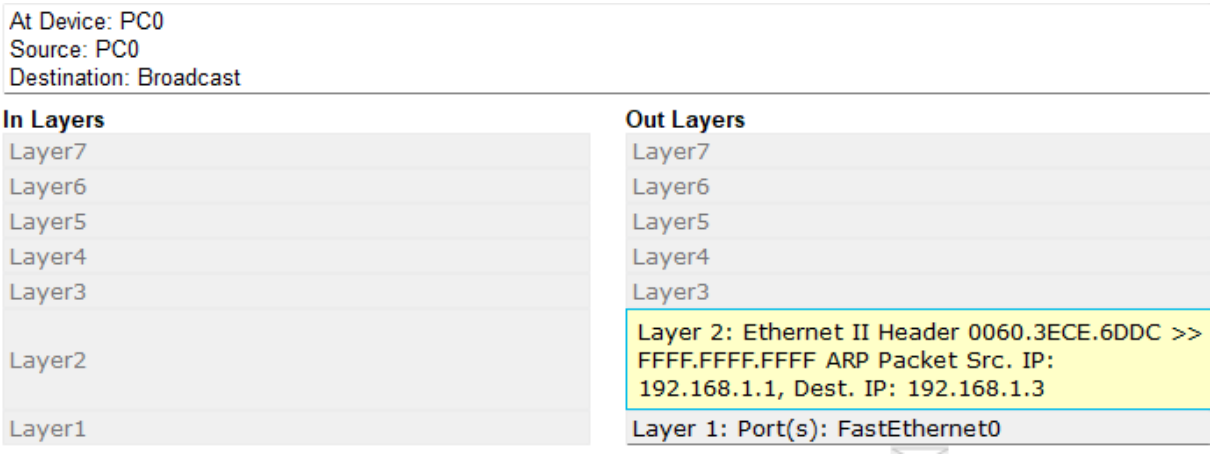


Figure 17: The OSI model displayed upon clicking the event.

Additionally, Outbound PDU details can be generated, as shown below.

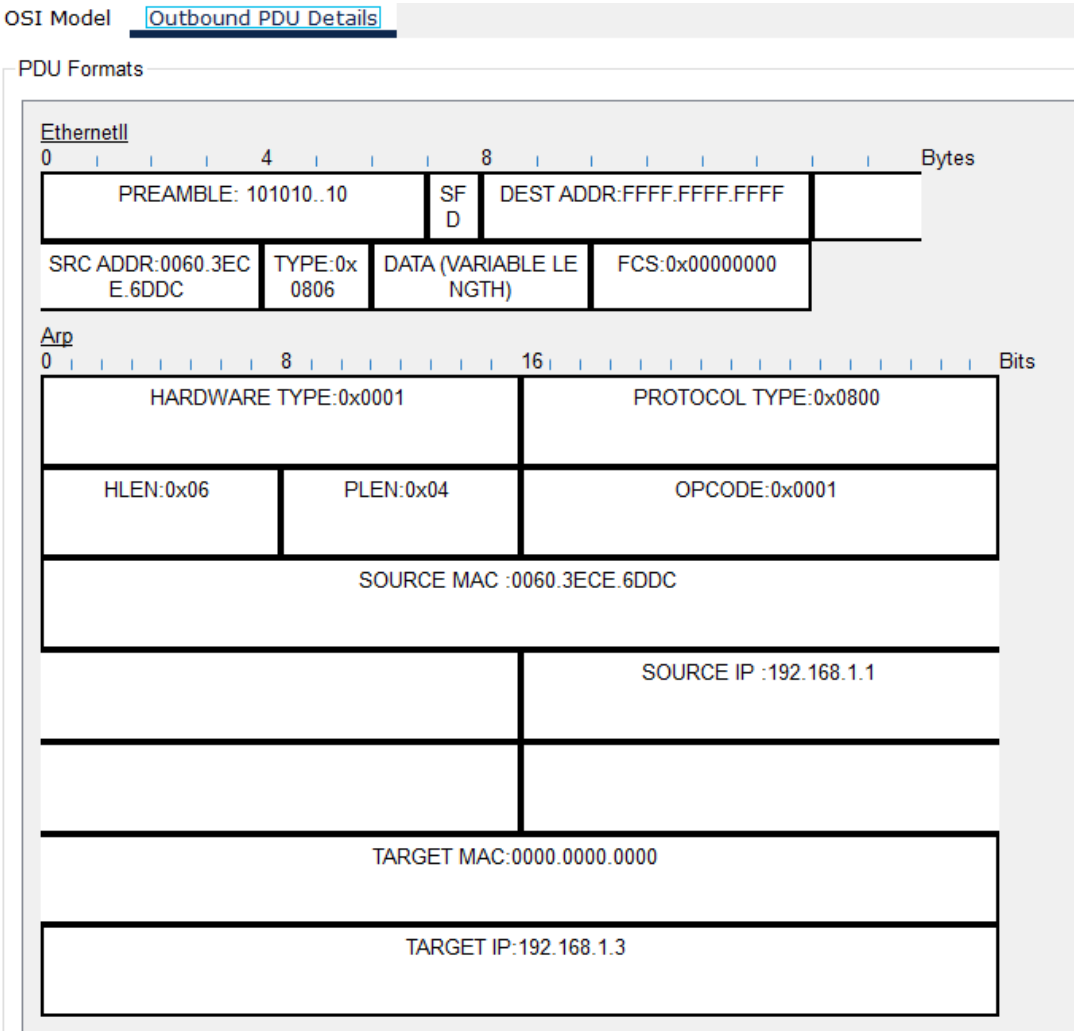


Figure 18: The Outbound PDU details.

Important Details such as the source and target IP and MAC addresses, the destination MAC addresses, and the Opcode value can all be verified through this output.

The MAC address table of the switch can be observed through the CLI, using the command `show mac-address-table` while in **user mode**. In the lab it is expected that all PCs are already shown here; however, in my case, even after multiple attempts, the table showed empty, and only filled after the following steps.

```
Switch#show mac-address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
```

Figure 19: MAC address table observed through the CLI of the switch.

After Clicking Capture then forward, the frame arrives at the switch and the MAC table prints as follows:

```
Switch#show mac-address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----

   1    0060.3ece.6ddc    DYNAMIC     Fa0/1
```

Figure 20: MAC address table after the frame arrives at the switch.

This is then repeated, and the frame is flooded out all ports except the inbound port. After clicking again, PC2 sends the packet to the switch. PC2's ARP table, at this stage, is as follows:

```
C:\>arp -a
  Internet Address      Physical Address      Type
  192.168.1.1           0060.3ece.6ddc        dynamic
```

Figure 21: ARP table of PC2.

The MAC table now has the following entries:

```
Switch#show mac-address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----

   1    0002.17c3.d4e3    DYNAMIC     Fa0/3
   1    0060.3ece.6ddc    DYNAMIC     Fa0/1
```

Figure 22: MAC address table after PC2 sends the packet to the switch.

And the Inbound PDU details are generated as follows:

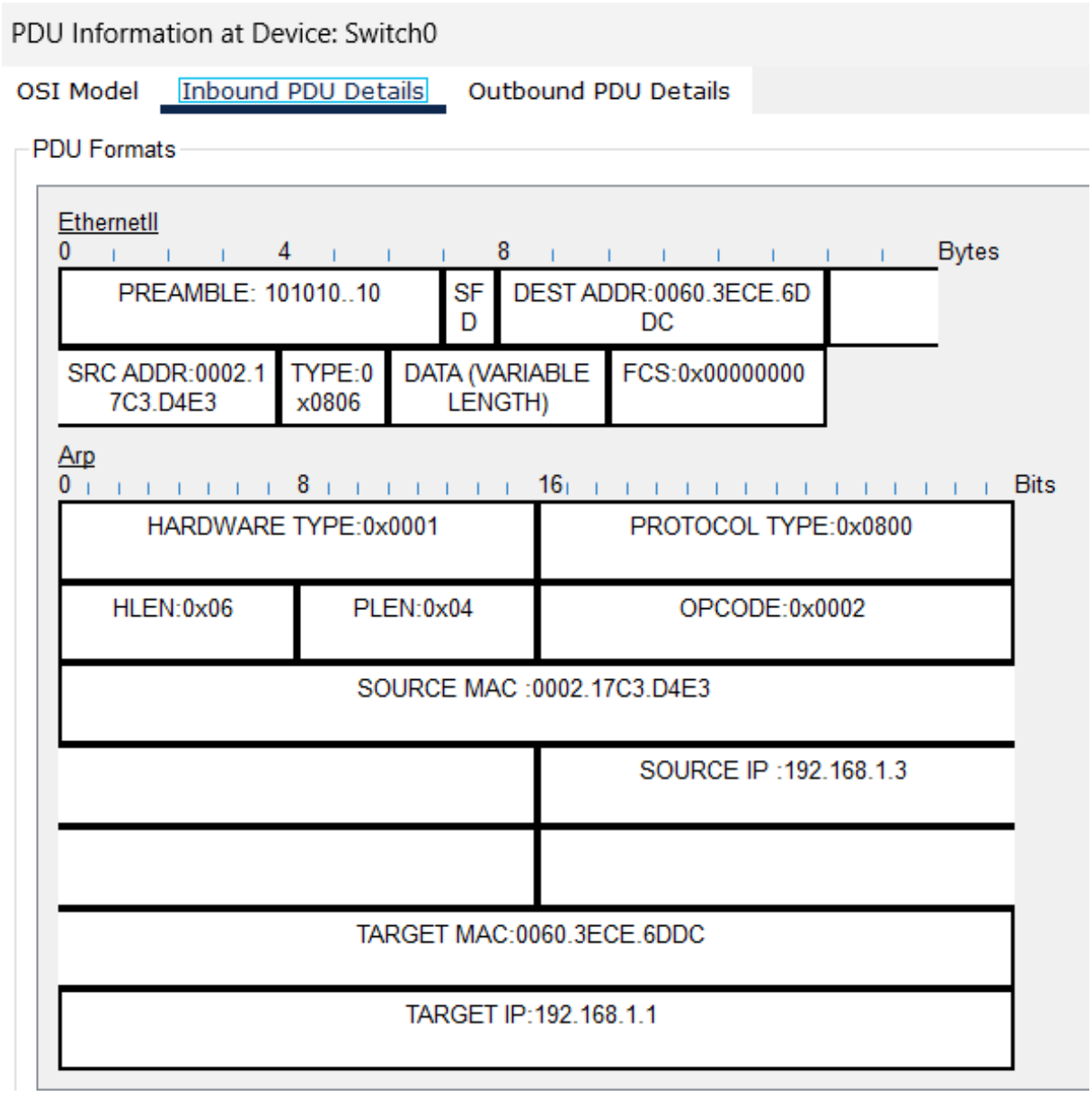


Figure 23: Inbound PDU details.

Once again, important details such as the source and target IP and MAC addresses, the destination MAC addresses, and the Opcode value can all be verified through this output.// For a final time, Capture then forward is pressed, and the switch sends the packet to PC0. The packet is not flooded, as the switch has the MAC addresses of the respective PCs, so flooding is unnecessary.

In the Command Prompt of PC0, the ICMP replies were successfully received, and the ARP table at PC0 is printed as follows:

```
C:\> arp -a
  Internet Address      Physical Address      Type
  192.168.1.3           0002.17c3.d4e3        dynamic
```

Figure 24: ARP table at PC0 after the ARP communication is complete.

It can be noted that record in the ARP table at PC0 contains the address of PC2, meaning the ARP communication was successful

---

7. Final Questions

Question 1: What is the destination MAC address for the ARP request?
The destination MAC address is the broadcast address `ff:ff:ff:ff:ff:ff`. This is because the ARP request is sent to all devices on the Local Area Network (LAN) since the requester does not yet know the MAC address of the target.

Question 2: What is the Opcode (Operation) value for the ARP request and response?
For the ARP request, the Opcode is 1, while for the ARP response, the Opcode is 2.

Question 3: What values does the ARP table contain?
The ARP table contains information about the values:

1. IP Address of the device
2. MAC Address of the device's network interface
3. Type of entry (static or dynamic)

Question 4: What is the size of ARP packet?
In this Lab, the request packet was 60 bytes, while the response packet was 42 bytes.

Question 5: What is the difference between static and dynamic records in ARP table?
Static records are manually configured, and while are rarely used, are useful for communicating with devices whose address does not vary over long periods; this is because static records have no timeout.
Dynamic records, on the other hand are automatically configured via ARP communication, and expire after a timeout of 2 minutes if not used, or a maximum of 10 minutes of continually used.

Question 6: At which OSI layers does the ARP operate?
ARP operates at the Data Link Layer and the Network Layer; it makes use of MAC address (a Data Link Layer address), and IP address (a Network Layer address)

Question 7: What is the difference between ARP and MAC table?
An ARP table contains mapping of IP addresses and MAC addresses of devices on the same local network, while a MAC table contains mapping of different device MAC addresses to a switch port.

Question 8: What does the switch do with a packet whose destination address is not contained in the MAC table?
The packet is flooded to all the ports except the port from which the packet came from. After flooding; the correct device responds, and it's MAC address is added to the MAC table for future reference.
