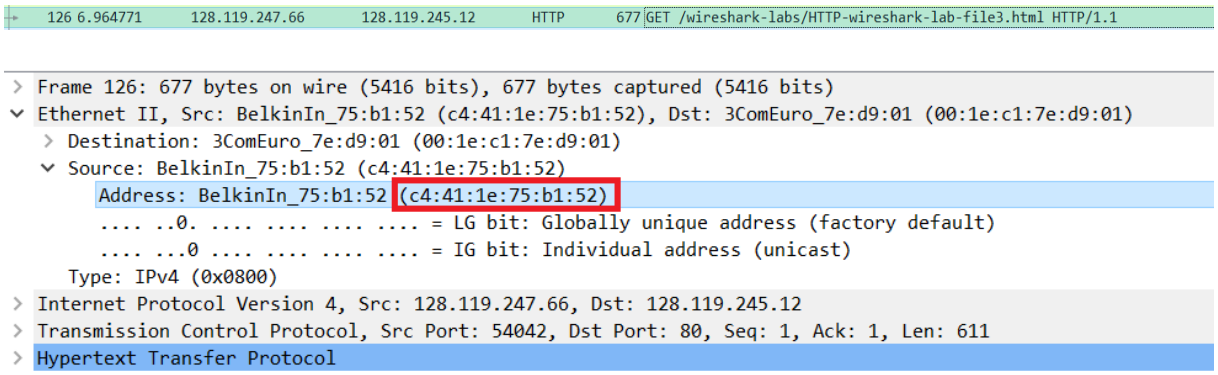
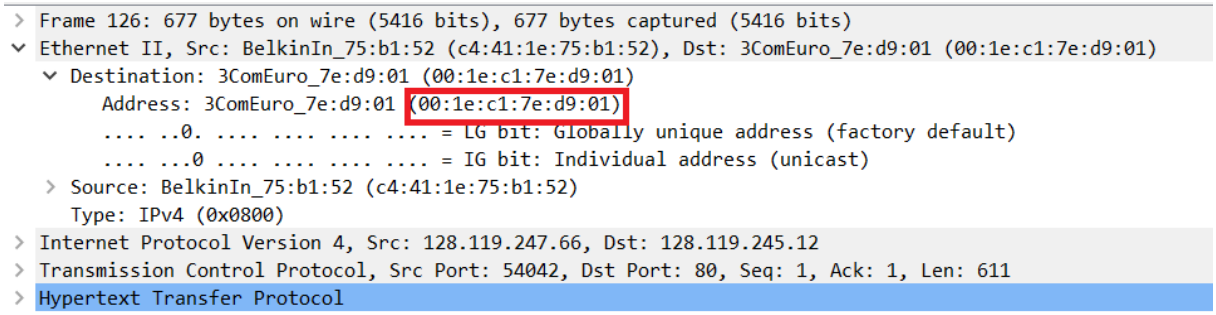
Anıl Eren Göçer – WSA7

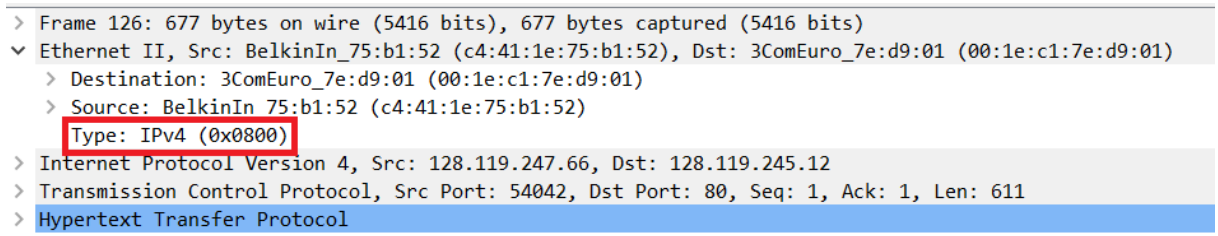
1) Ethernet address of the sender = c4:41:1e:75:b1:52



2) Destination address in the Ethernet frame = 00:1e:c1:7e:d9:01
No, this is not the Ethernet adress of gaia.cs.umass.edu . This is the Ethernet address of the gateway router in the subnet of the sender. That is the router which enables packages from the sender to get off the subnet.

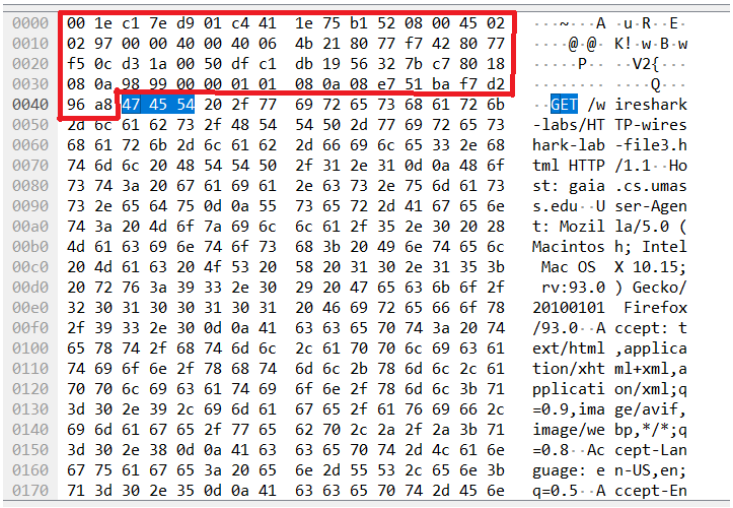


3) Hexadecimal value of the type field is 0x0800 and it corresponds to upper layer protocol IPv4.

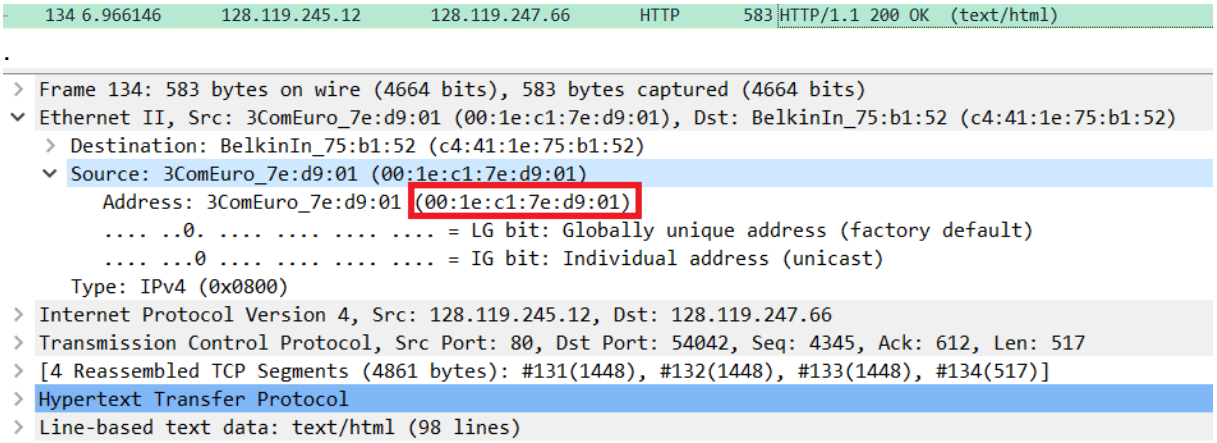


4) There are 66 bytes until 'G' in "GET" :
   - 14 bytes coming from "Destination", "Source" and "Type" in Ethernet II.
   - 20 bytes coming from IPv4 header fields.
   - 32 bytes coming from TCP header fields.

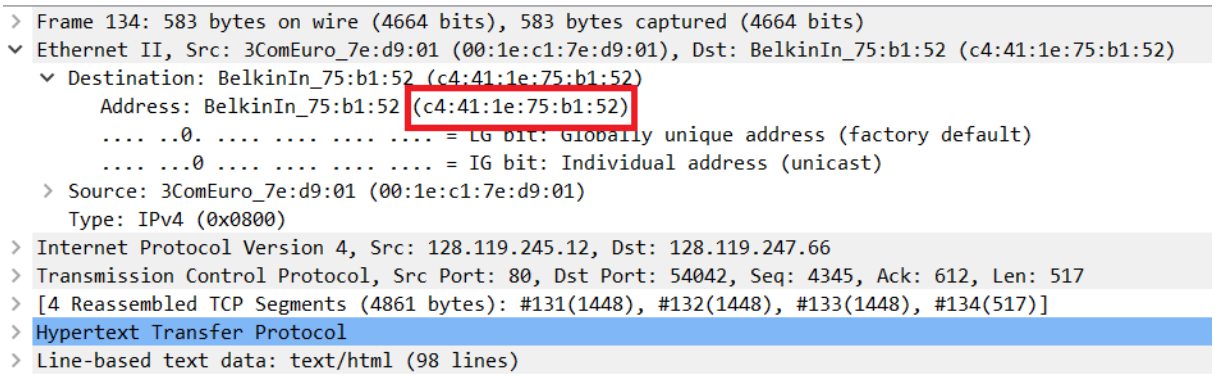
   Therefore, 66 bytes in total.



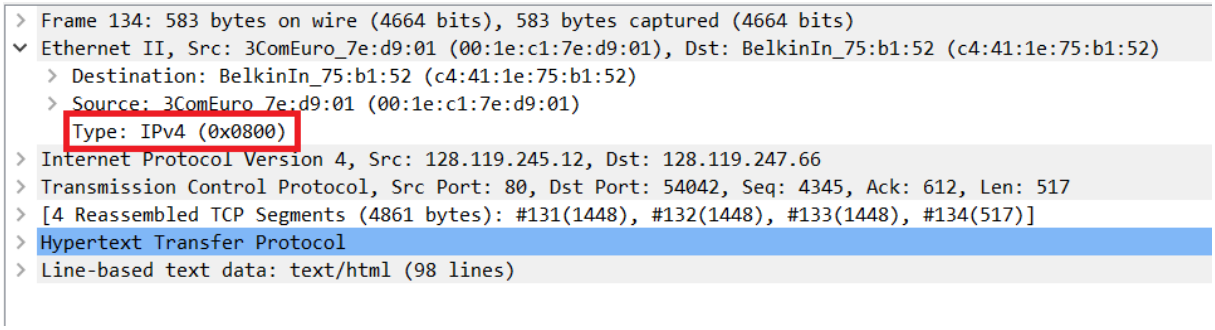
5) Ethernet source address = 00:1e:c1:7e:d9:01
   No, this is not the address of the sending computer, or of gaia.cs.umass.edu . Similar to question 2, this is th Ethernet address of the gateway router in the subnet of client which sends HTTP request and gets HTTP response. That is the router which enables to get off the subnet



6) Destination address in the Ethernet frame = c4:41:1e:75:b1:52
Yes, this is the Ethernet address of the sender.



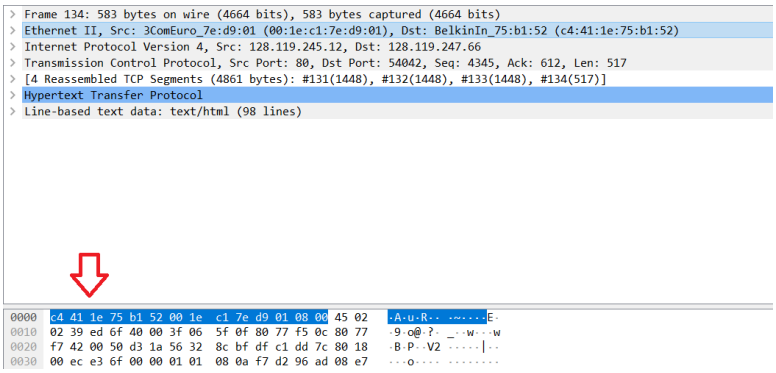
7) Hexadecimal value of the type field is 0x8000 and it corresponds to upper layer protocol IPv4.

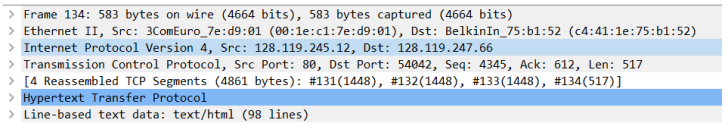


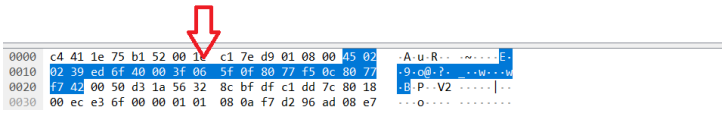
8) There are 79 bytes until 'O' in "OK"

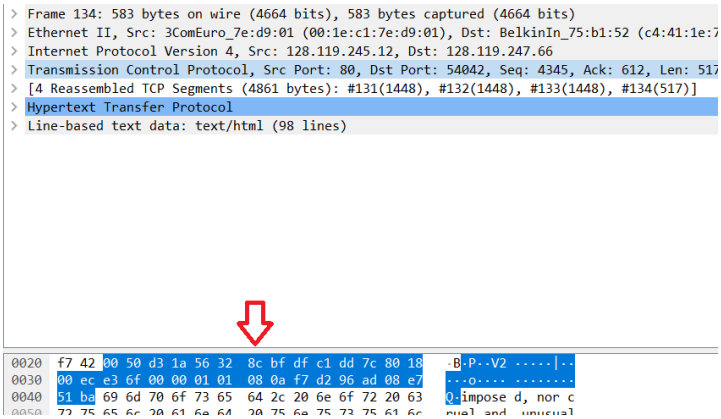
- 14 bytes coming from Ethernet II header fields
- 20 bytes coming from IPv4 header fields
- 32 bytes coming from TCP header fields
- 13 bytes coming from HTTP until 'O' in "OK"

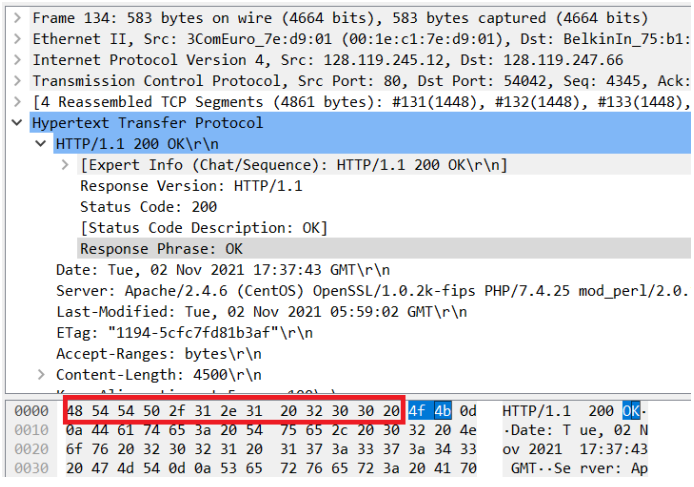
Therefore, 79 bytes in total.



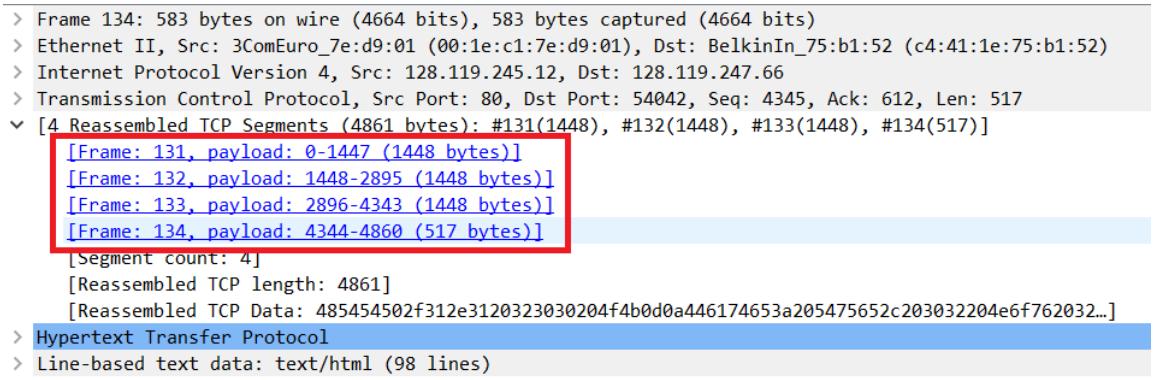

Frame 134: 583 bytes on wire (4664 bits), 583 bytes captured (4664 bits)
Ethernet II, Src: 3ComEuro_7e:d9:01 (00:1e:c1:7e:d9:01), Dst: BelkinIn_75:b1:52 (c4:41:1e:75:b1:52)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 128.119.247.66
Transmission Control Protocol, Src Port: 80, Dst Port: 54042, Seq: 4345, Ack: 612, Len: 517
[4 Reassembled TCP Segments (4861 bytes): #131(1448), #132(1448), #133(1448), #134(517)]
Hypertext Transfer Protocol
Line-based text data: text/html (98 lines)


0000 c4 41 1e 75 b1 52 00 1e c1 7e d9 01 08 00 45 02
0010 02 39 ed 6f 40 00 3f 06 5f 0f 80 77 f5 0c 80 77
0020 f7 42 00 50 d3 1a 56 32 8c bf df c1 dd 7c 80 18
0030 00 ec e3 6f 00 00 01 01 08 0a f7 d2 96 ad 08 e7

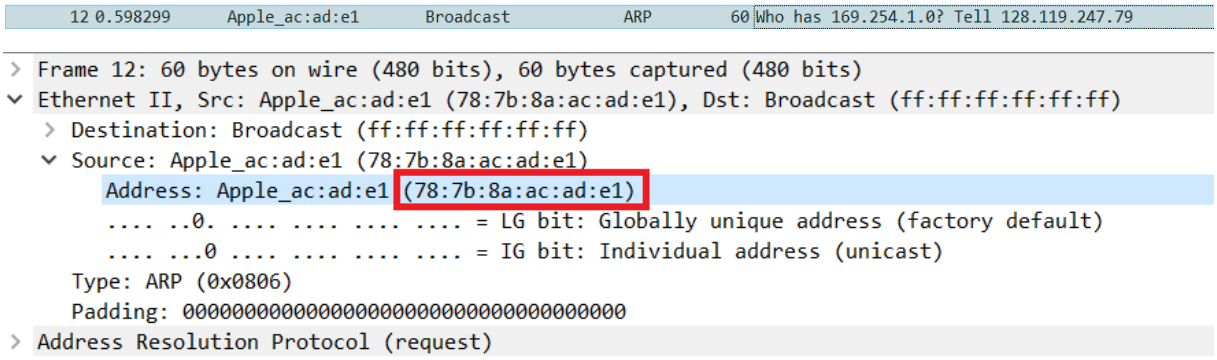

Frame 134: 583 bytes on wire (4664 bits), 583 bytes captured (4664 bits)
Ethernet II, Src: 3ComEuro_7e:d9:01 (00:1e:c1:7e:d9:01), Dst: BelkinIn_75:b1:52 (c4:41:1e:7
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 128.119.247.66
Transmission Control Protocol, Src Port: 80, Dst Port: 54042, Seq: 4345, Ack: 612, Len: 517
[4 Reassembled TCP Segments (4861 bytes): #131(1448), #132(1448), #133(1448), #134(517)]
Hypertext Transfer Protocol
Line-based text data: text/html (98 lines)
0020 f7 42 00 50 d3 1a 56 32 8c bf df c1 dd 7c 80 18
0030 00 ec e3 6f 00 00 01 01 08 0a f7 d2 96 ad 08 e7
0040 51 ba 69 6d 70 6f 73 65 64 2c 20 6e 6f 72 20 63


Frame 134: 583 bytes on wire (4664 bits), 583 bytes captured (4664 bits)
Ethernet II, Src: 3ComEuro_7e:d9:01 (00:1e:c1:7e:d9:01), Dst: BelkinIn_75:b1:
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 128.119.247.66
Transmission Control Protocol, Src Port: 80, Dst Port: 54042, Seq: 4345, Ack:
[4 Reassembled TCP Segments (4861 bytes): #131(1448), #132(1448), #133(1448),
Hypertext Transfer Protocol
HTTP/1.1 200 OK\r\n
[Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
Response Version: HTTP/1.1
Status Code: 200
[Status Code Description: OK]
Response Phrase: OK
Date: Tue, 02 Nov 2021 17:37:43 GMT\r\n
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.25 mod_perl/2.0.
Last-Modified: Tue, 02 Nov 2021 05:59:02 GMT\r\n
ETag: "1194-5cfc7fd81b3af"\r\n
Accept-Ranges: bytes\r\n
Content-Length: 4500\r\n
0000 48 54 54 50 2f 31 2e 31 20 32 30 30 20 4f 4b 0d
0010 0a 44 61 74 65 3a 20 54 75 65 2c 20 30 32 20 4e
0020 6f 76 20 32 30 32 31 20 31 37 3a 33 37 3a 34 33
0030 20 47 4d 54 0d 0a 53 65 72 76 65 72 3a 20 41 70

9) 4 reassembled TCP segments correspond to 4 Ethernet frames.

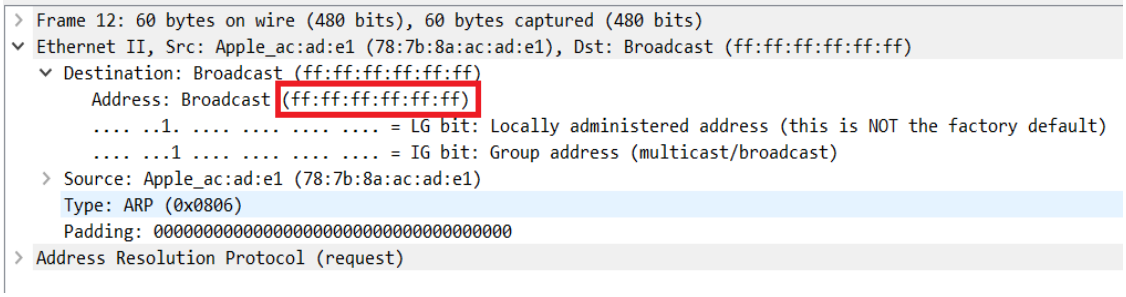


10) Source address = 78:7b:8a:ac:ad:e1

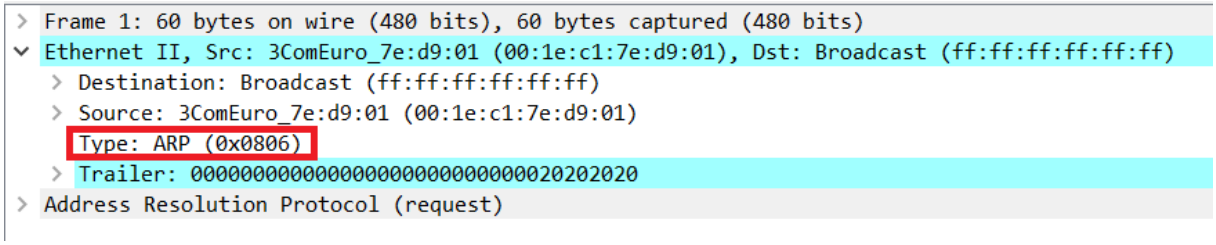


11) Destination address = ff:ff:ff:ff:ff:ff

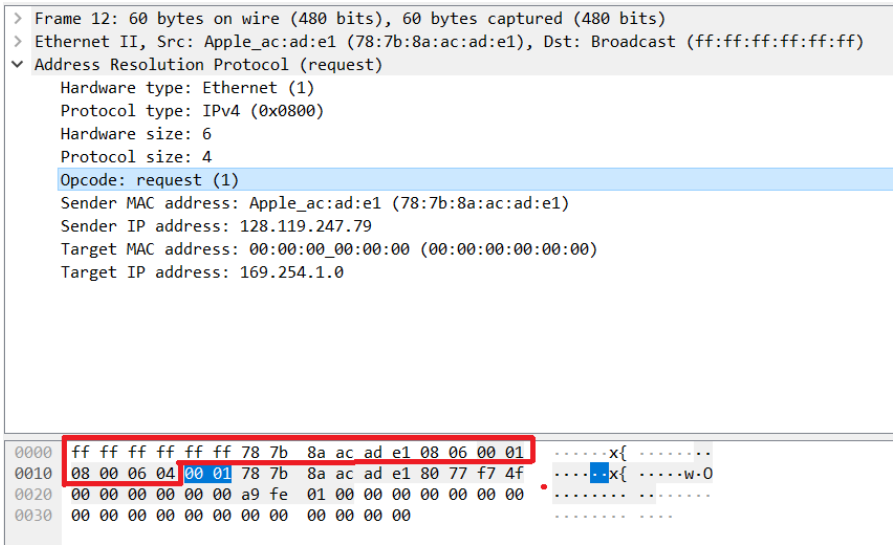
This address is broadcast address, indicating that the ARP request is sent to all devices on the local network to discover the MAC address associated with a particular IP address. The device that owns the requested IP address will respond with its MAC address.



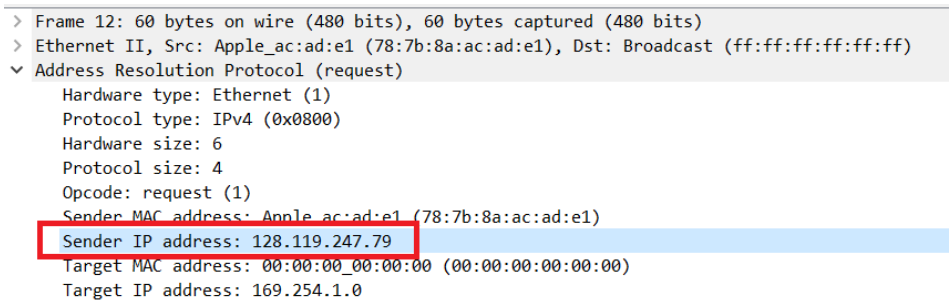
12) The hexadecimal value for the two-byte Ethernet frame type field is 0x0806 and it corresponds to upper layer protocol ARP.



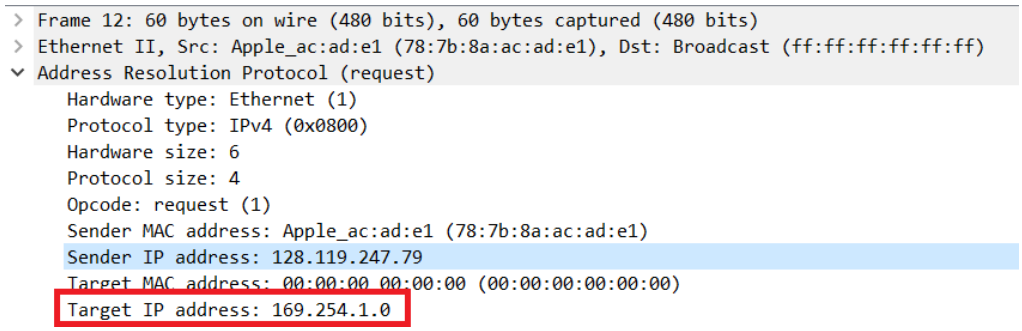
13) There are 20 bytes from the very beginning of the Ethernet frame until the ARP opcode field.



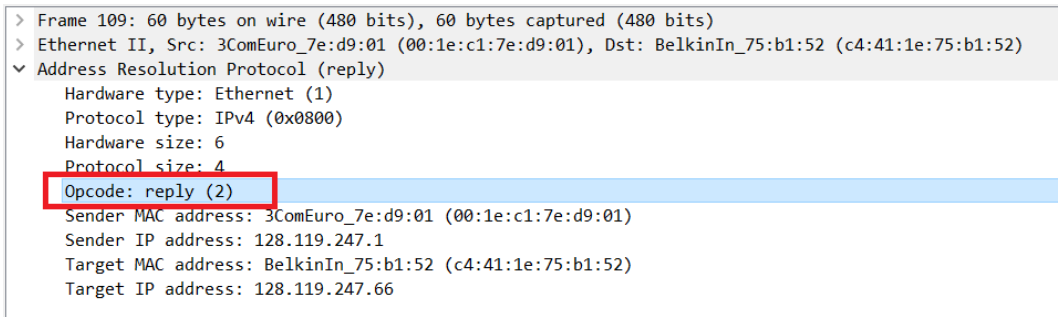
14) Yes, it contains.
IP address of the sender: 128.119.247.79



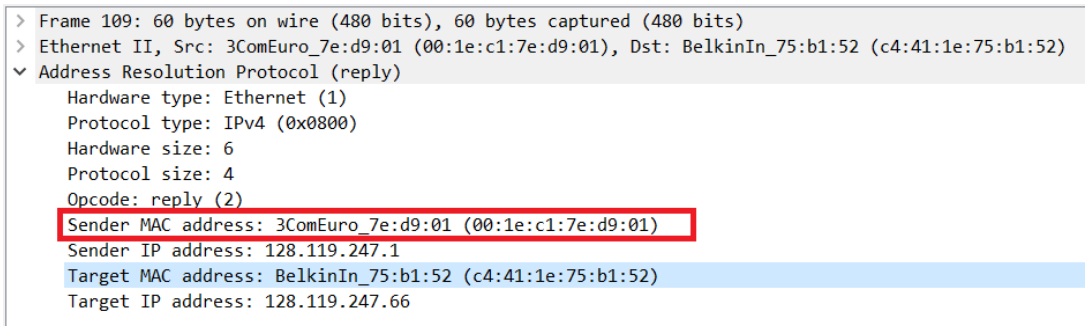
15) It is the target IP address and its value is 169.254.1.0



16) Value of the opcode field is 2.



17) It is 00:1e:c1:7e:d9:01



18) The absence of ARP replies in this trace is due to the fact that we are not analyzing the network from the perspective of the machine that initiated the ARP request. In the ARP protocol, when a device sends out an ARP request, it broadcasts the request to all devices on the local network. However, the ARP reply is unicast, meaning it is sent directly back to the specific Ethernet address of the original requester. Since we are not situated at the machine that sent the initial ARP request, we do not capture the unicast ARP replies in this trace.