CENG312

Assignment#3

IP, ICMP, NAT, DHCP

Name Surname: Sude Nur Çevik

ID: 270201041

Department: Computer Engineering

Lecture: Computer Networks - Spring 2023

Instructor: Prof. Cüneyt Fehmi Bazlamaçcı

IP and ICMP Section

1.
   a. The packets that trigger the "Fragmentation Needed" ICMP message are packet 2, packet 9, packet 11, and packet 18*(see fig1.1)*. These packets share the following characteristics that necessitate fragmentation:
      i. **Protocol:** The protocol field in the IP header of these packets is set to ICMP (1) *(see fig 1.2)*. A fragmentation problem is being indicated via ICMP, a network-layer protocol used for diagnostic and control reasons.
      ii. **Type and Code:** The ICMP message type is 3, indicating a "Destination Unreachable" message, and the code is 4, specifically indicating "Fragmentation Needed." *(see fig 1.3)*. The packet is too big to be sent without fragmentation, according to this type and code combination.
      iii. **MTU of Next Hop:** The "MTU of next hop" ICMP message field in each of these packets has the value 1476. The MTU (Maximum Transmission Unit) designates the largest packet size that may be transferred over a network without being fragmented. *(see fig1.4).* This field's existence indicates that fragmentation is necessary since the packet's size exceeds the next hop network's MTU.
      iv. **Total Length:** These packets' IP headers have a "Total Length" field that reveals the complete size of the packet, including the header and contents. These packets are 56 bytes long overall, which is longer than the next hop network's MTU. *(see fig 1.5).*

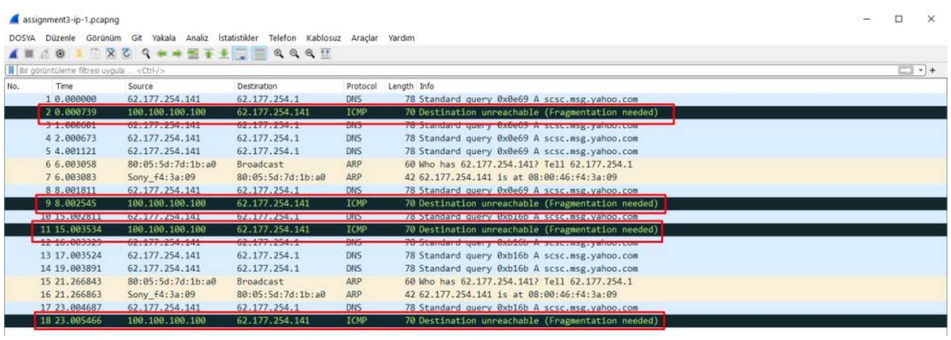

Fig1.1 "Fragmentation Needed" packets

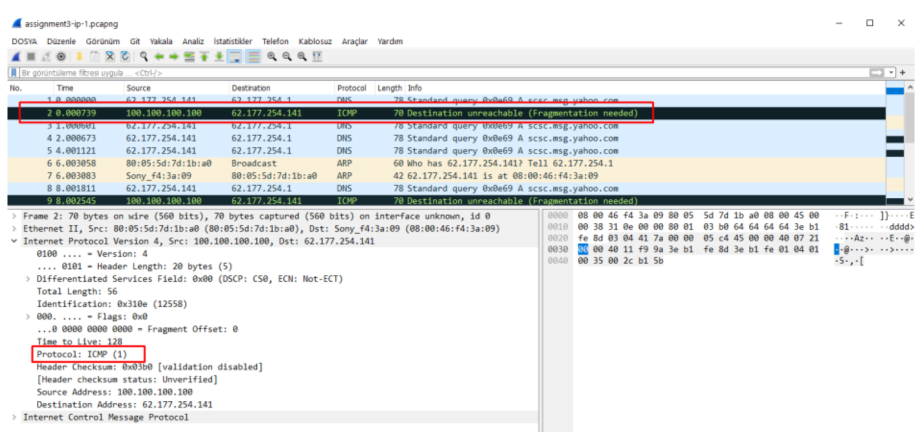

Fig1.2 Protocol "ICMP (1)" Example

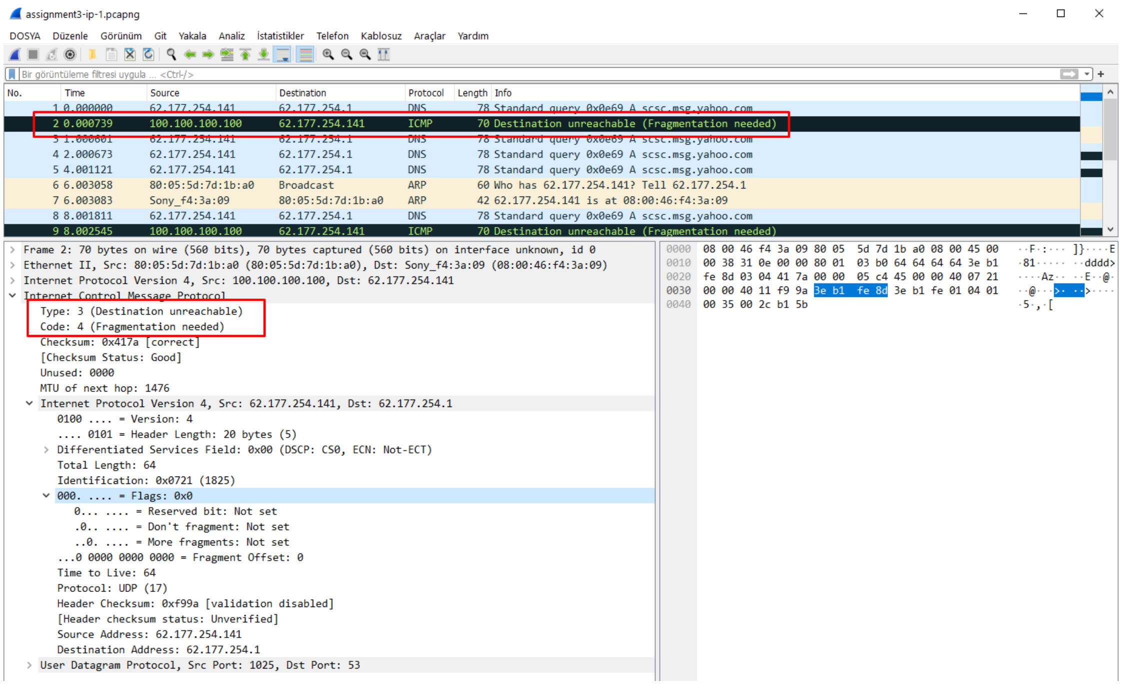

Fig1.3 Type and Code Example

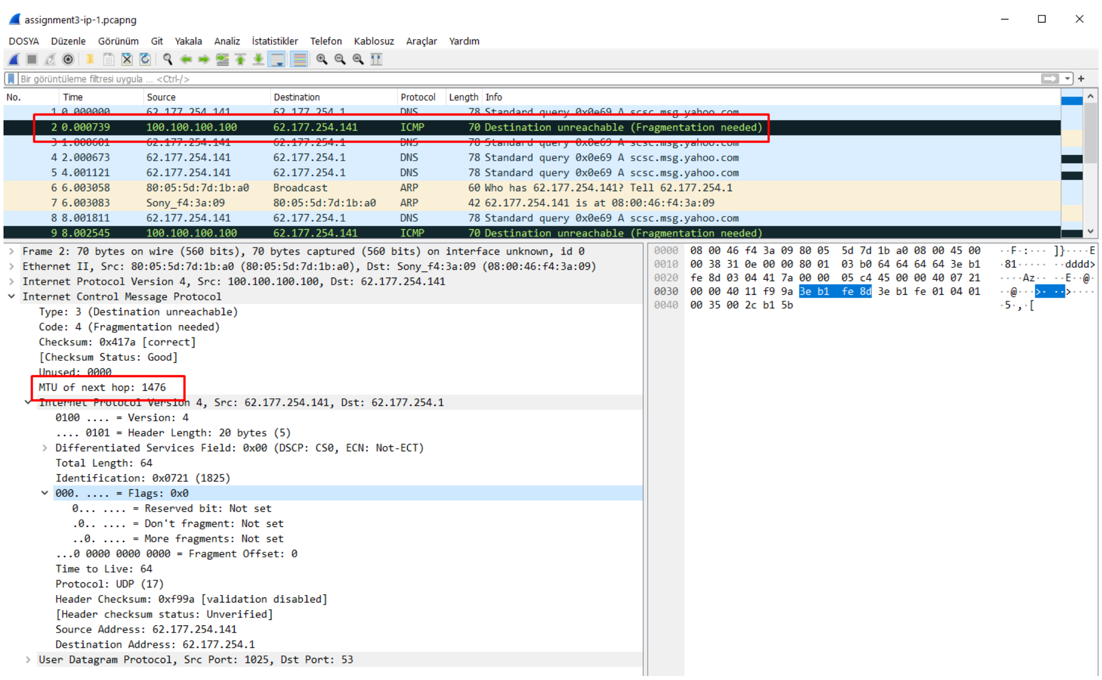

Fig1.4 MTU of Next Hop Example

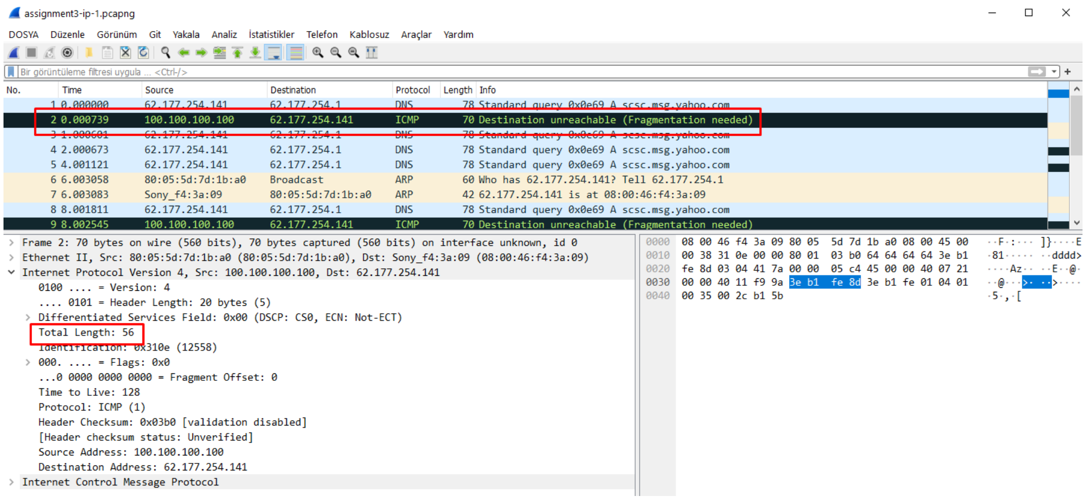

Fig1.5 Total Length Example

b. When endpoints receive a packet with the "Fragmentation Needed" flag, it indicates that the packet is too large to be transmitted over the network without fragmentation. To process these packets and ensure successful delivery, the following steps can be taken:

   i. Upon receiving the "Fragmentation Needed" ICMP message, the endpoint analyzes the message and extracts relevant information, such as the original packet causing the fragmentation and the recommended MTU size.

   ii. The endpoint then checks its local MTU configuration to determine the maximum size of packets it can transmit without fragmentation. This value is typically based on the network interface or the network path to the destination.

   iii. If the local MTU is smaller than the recommended MTU in the ICMP message, the endpoint needs to fragment the original packet into smaller fragments that fit within its local MTU. This process involves breaking the packet into smaller pieces and adjusting the necessary IP header fields.

   iv. The endpoint creates multiple fragments, each with a smaller size that fits within the local MTU. The IP header of each fragment is modified to indicate its position within the original packet, using the "Fragment Offset" and "More Fragments" fields.

   v. The endpoint then transmits each fragment individually, ensuring that they are sent in the correct order. The receiving endpoint or intermediate routers will use the information in the IP headers to reassemble the fragments into the original packet.

   vi. Upon receiving the fragments, the receiving endpoint or intermediate routers reassemble them based on the information in the IP headers. This involves combining the fragments in the correct order and adjusting the IP header fields of the reassembled packet.

vii. Once the reassembly is complete, the receiving endpoint can process the packet as a complete unit. This may involve further processing at higher network layers, such as forwarding the packet or passing it to the relevant application or service.

Yes there are 2 fragmentation process on fig 1.6. There is fragmentation needed messages and MTU configuration. There is no fragmentation, no reassembling but those two are exist.

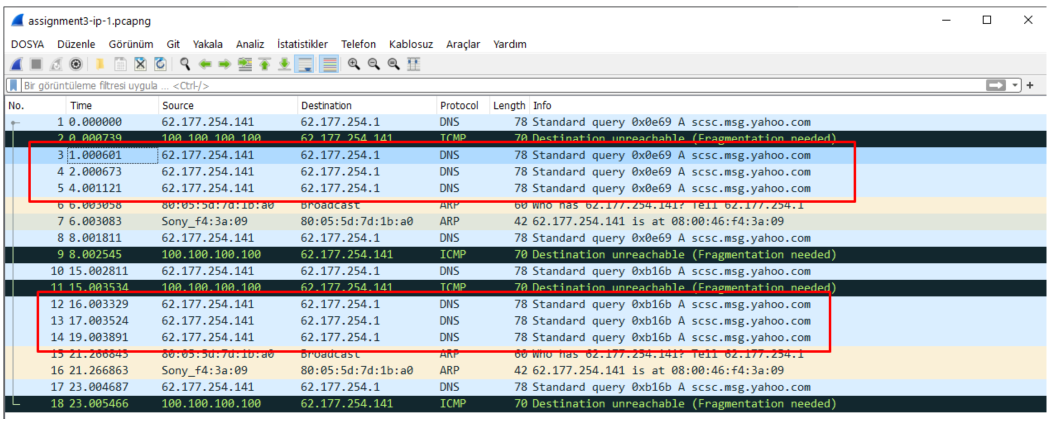

Fig1.6 Fragmented Packets

2.

a. We can determine fragmented packets by looking at the more fragment flag whether it is set and recognizing the fragment offset. The fragment offset is a 13-bit field in the IP header that identifies the position of a fragment relative to the beginning of the original unfragmented datagram. It indicates the number of data bytes preceding or ahead of the fragment. To see the details of fragmented packet, look at the fig2.1.

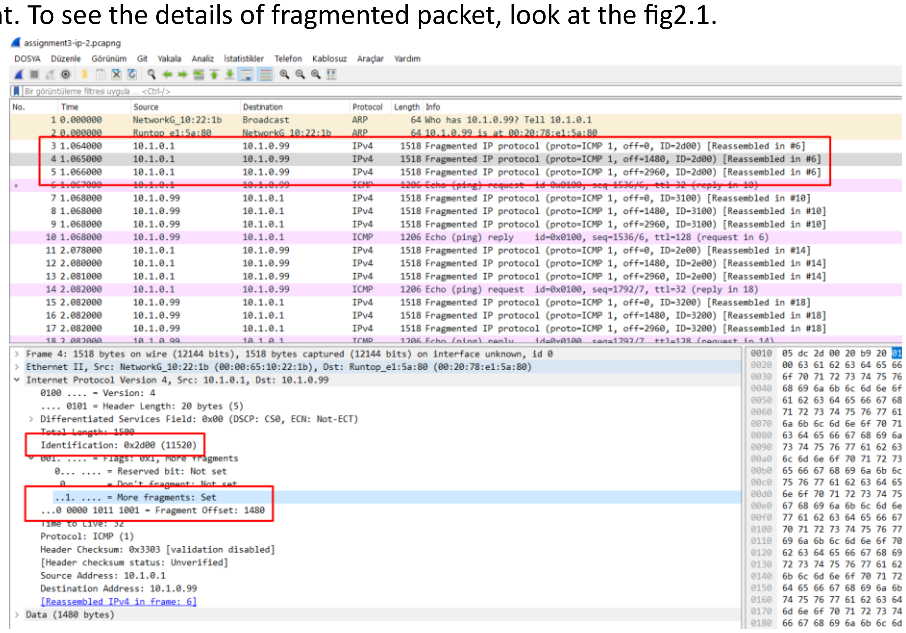

Fig2.1 Fragmented Packet

b. IP fragmentation is the process of breaking up a large IP packet into smaller packets that can be transmitted over the network. This is necessary because different networks have different maximum transmission unit (MTU) sizes, which is the largest size of packet that can be transmitted over the network without being fragmented. If a packet is too large to be transmitted over the network without being fragmented, it will be broken up into smaller packets that can be transmitted over the network and then reassembled at the destination.

c. There are four fragmented packet for all purple colored packets(see fig2.2) .To identify and reassemble fragmented IP packets in Wireshark, you can use the “Follow TCP Stream” feature. This feature will automatically reassemble all the fragments of a TCP stream into a single packet. You can also use the “IP Fragmentation” filter to display only fragmented packets.

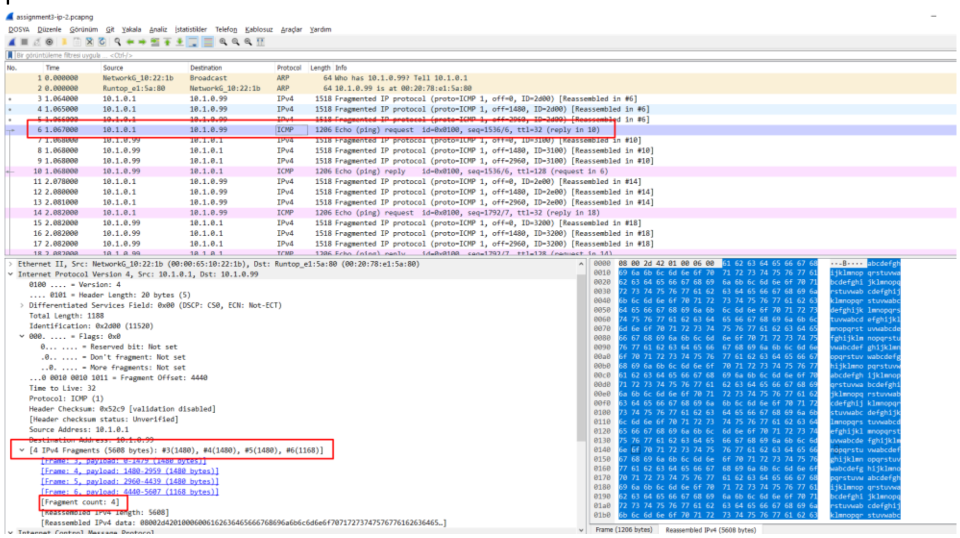

Fig2.2 Fragmented Packet

d. Total Size = (Last Fragment Offset + Last Fragment Length),
Total size= 4400 + 1168 = 5608 (see fig2.3)

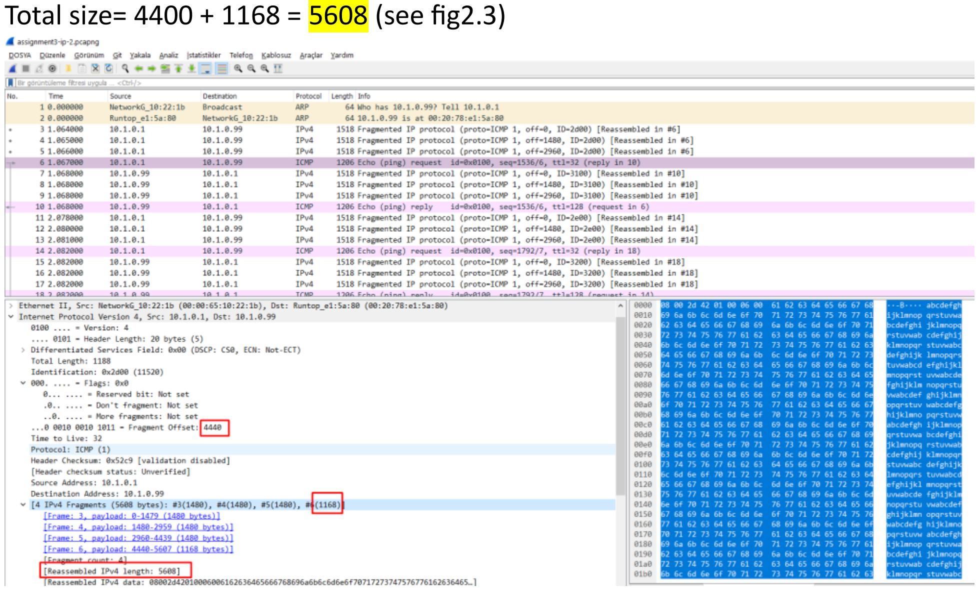

Fig2.3 Total Size of IP Packet

e. The time to live field is not changed (see fig2.5 & fig2.6 & fig2.7). In this case, the TTL value remains the same (32) for all fragments. The TTL field in the IP header is initially set by the source host and is decremented by one by each router that forwards the packet. The purpose of the TTL field is to limit the lifetime of IP packets and prevent them from indefinitely circulating in the network. If the TTL value were to change in the fragments, it could indicate that the fragments took different paths in the network, encountering routers with different default TTL settings or routers that modified the TTL value. However, in this case, the TTL value remains constant across all fragments, suggesting that they likely followed the same path and encountered routers that did not modify the TTL field.

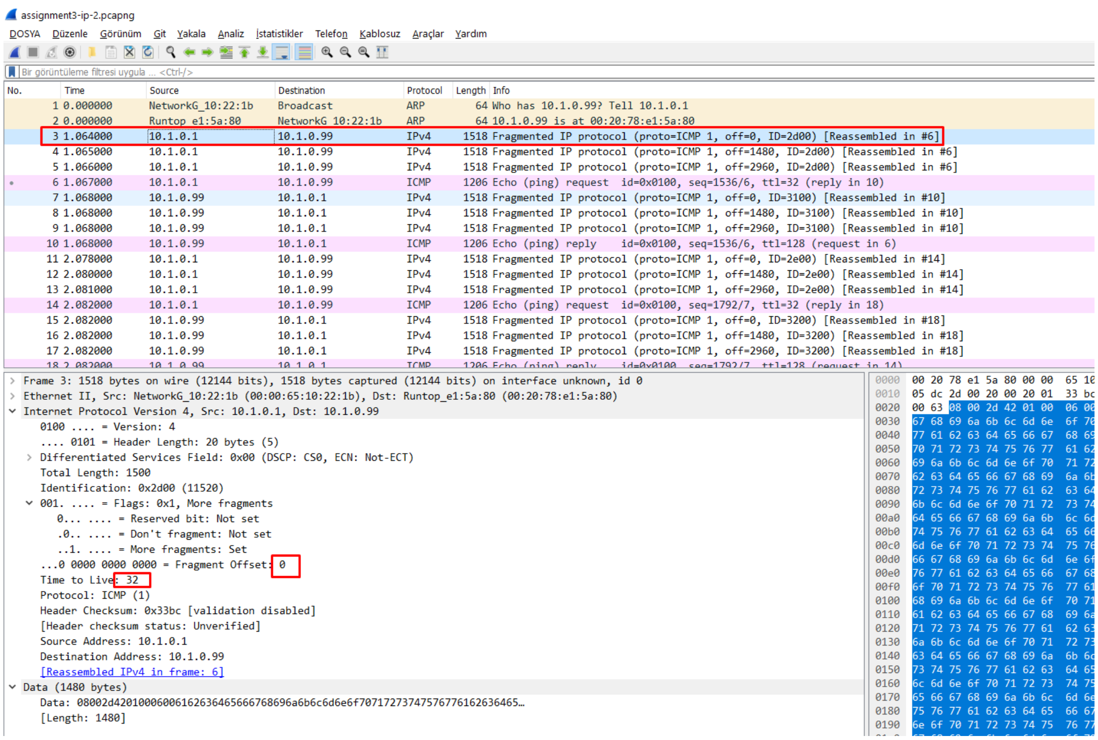

Fig2.5 First Fragmented Packet

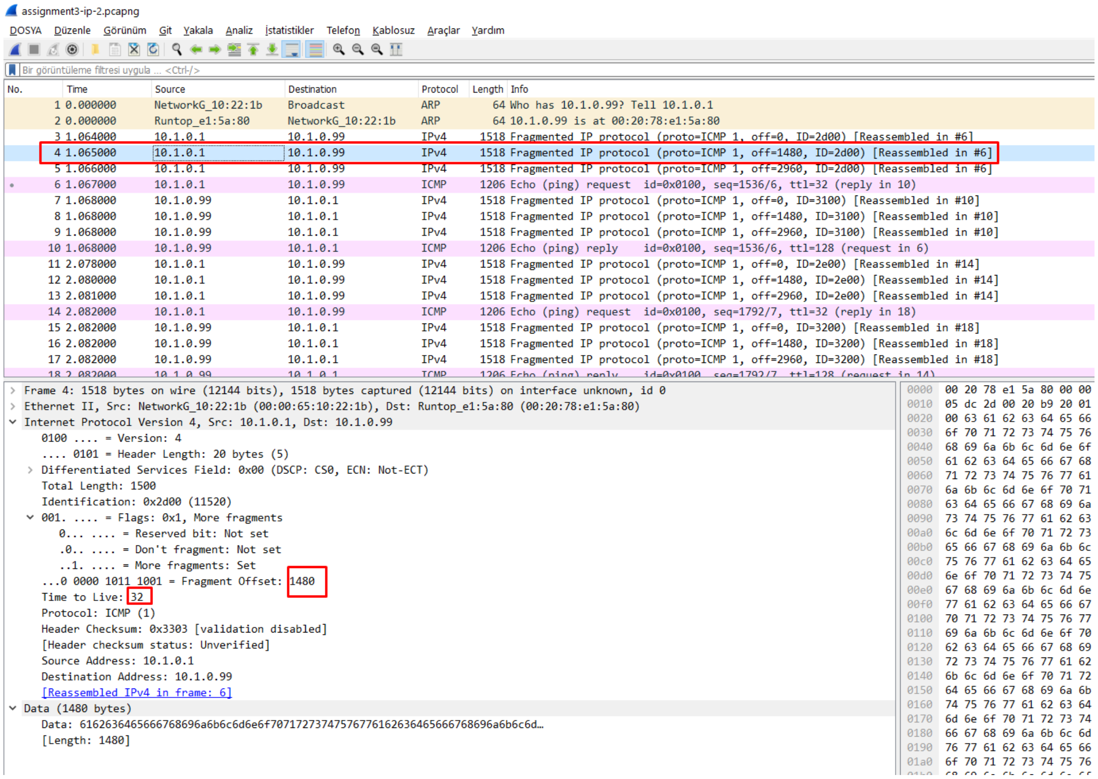

Fig2.6 Second Fragmented Packet

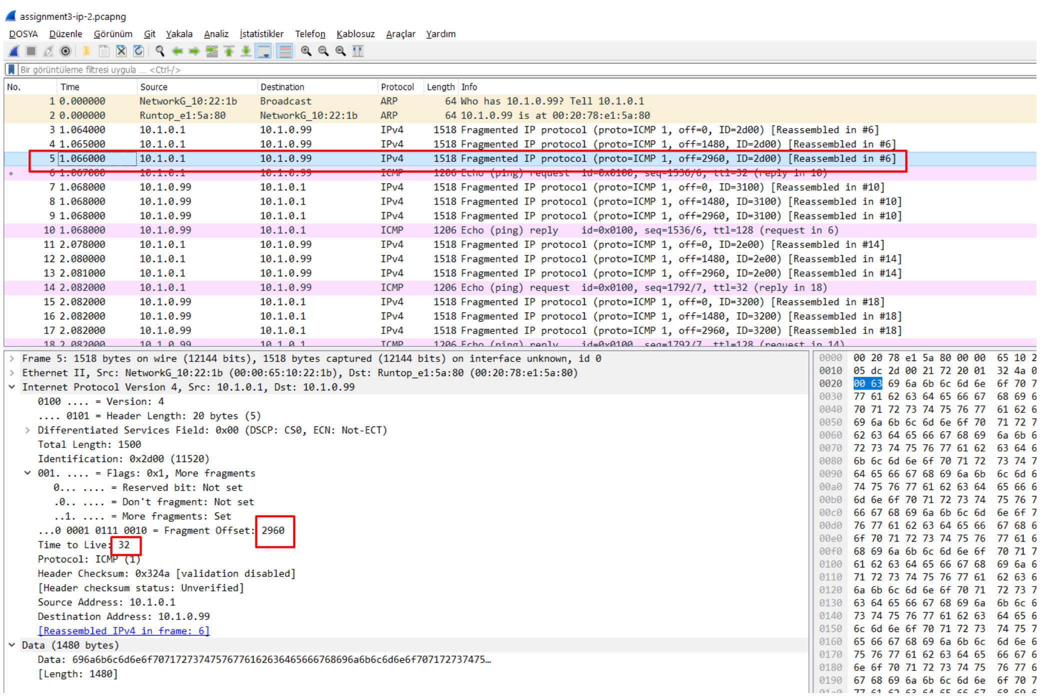

Fig2.7 Third Fragmented Packet

3.

It appears that there is no direct IPv6 connectivity between the two networks. To send IP datagrams between these networks, a technique called IPv6 over IPv4 tunneling is used. In this scenario, IPv6 packets are encapsulated within IPv4 packets to traverse an IPv4 network and reach the destination network.

To identify and understand this action within Wireshark, you can look for the following steps or indicators:

Source and Destination IP Addresses: Check the source and destination IP addresses in the captured packets. If you see IPv6 addresses that start with "2002:" and are followed by an IPv4 address, it indicates the presence of IPv6 over IPv4 tunneling. For example, in the provided packet capture, you can see addresses like "2002:1806:addc::1806:addc" and "2607:f0d0:2001:e:1::120," where the IPv6 address is encapsulated within the IPv4 address.(fig 3.1)

Protocol Field: In the Ethernet II header, the protocol field should indicate "IPv4" (0x0800) if the encapsulation is being used. This indicates that the Ethernet frame is carrying IPv4 packets with encapsulated IPv6 packets. (fig 3.2)

TTL Values: Compare the TTL values of the IPv4 and IPv6 packets. If the TTL values of the encapsulating IPv4 packets are lower than the TTL values of the encapsulated IPv6 packets, it suggests that the encapsulation is taking place, as the IPv4 TTL is decremented by each router it passes through.

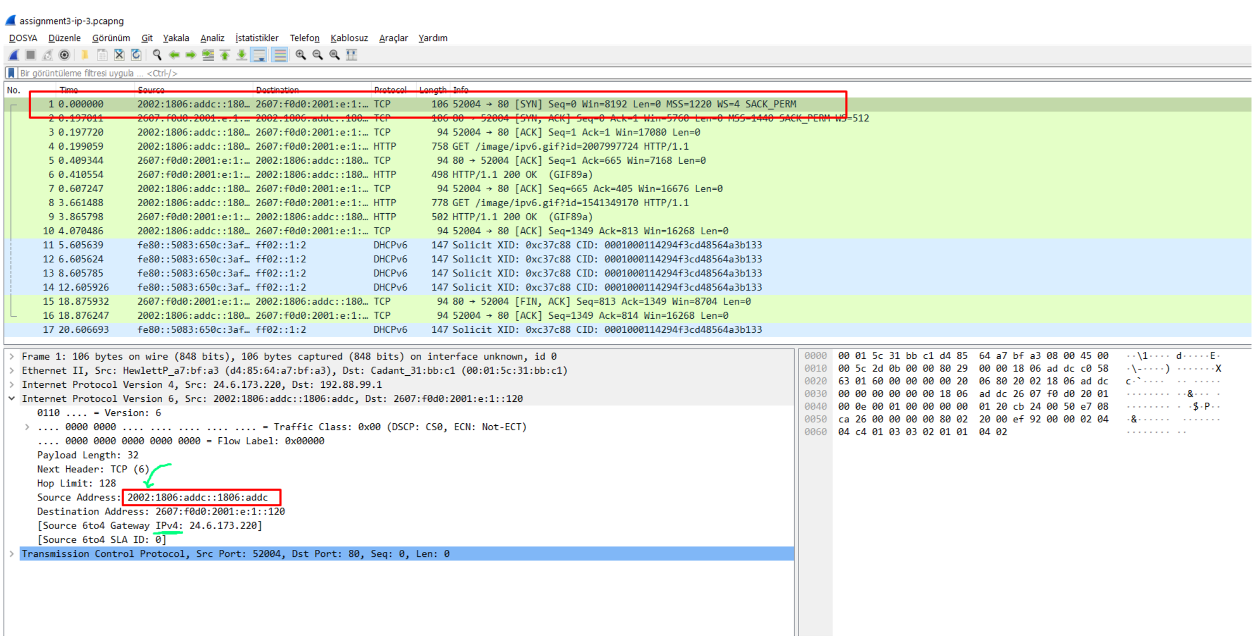

Fig3.1 IPv6 Address is Encapsulated Within the IPv4 Address

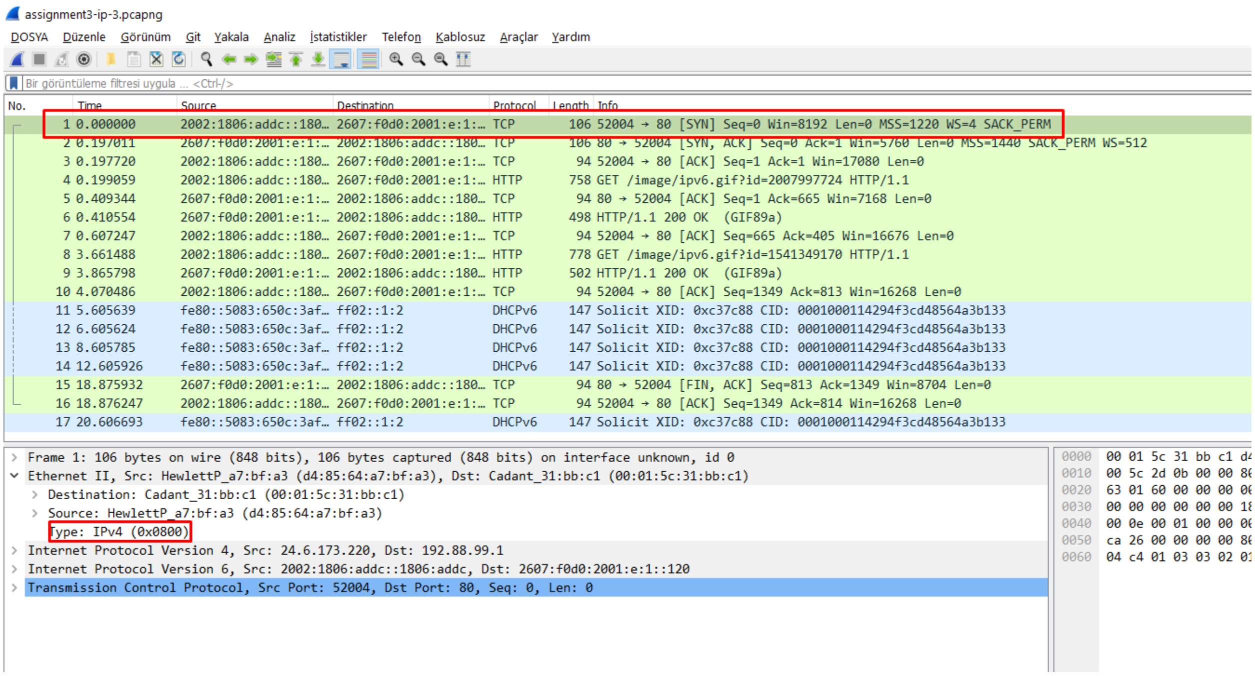

Fig3.2 "IPv4" (0x0800)

4.

a. These are the ICMP types and codes below (*see fig4.1*):

i. ICMP Type: 8 (Echo (ping) request)
ICMP Code: 0 Explanation: This ICMP type and code combination indicates an echo request message. It is commonly known as a "ping" request. In this case, a host is requesting a response from another host to check if it is reachable and measure the round-trip time.

ii. ICMP Type: 0 (Echo (ping) reply)
ICMP Code: 0 Explanation: This ICMP type and code combination indicates an echo reply message. It is the response to an echo request (ping) message. It confirms the reachability of a host and provides the round-trip time.

iii. ICMP Type: 3 (Destination Unreachable)
ICMP Code: 3 (Port Unreachable) Explanation: This ICMP type and code combination indicates a destination unreachable message due to a specific port being unreachable. It is used to inform the sender that the destination host is reachable, but the port requested is not open or available on that host.

iv. ICMP Type: 11 (Time Exceeded)
ICMP Code: 0 (Time to Live (TTL) exceeded in transit) Explanation: This ICMP type and code combination indicates a time exceeded message. It is sent by a router when the TTL field in an IP packet reaches zero. The purpose is to prevent packets from circulating indefinitely in the network, and it helps identify routing problems.

b. Yes, they are the same, however there are some differences. The intermediate network devices identified in the traceroute results and Wireshark analysis may not always be the same. Traceroute might miss some devices that Wireshark can detect, especially if ICMP messages are blocked or filtered along the network path. Additionally, Wireshark can provide more granular information about the network traffic, including devices that might not participate in the ICMP Time Exceeded message exchange.

c. Three packets are sent *(see fig 4.2)*

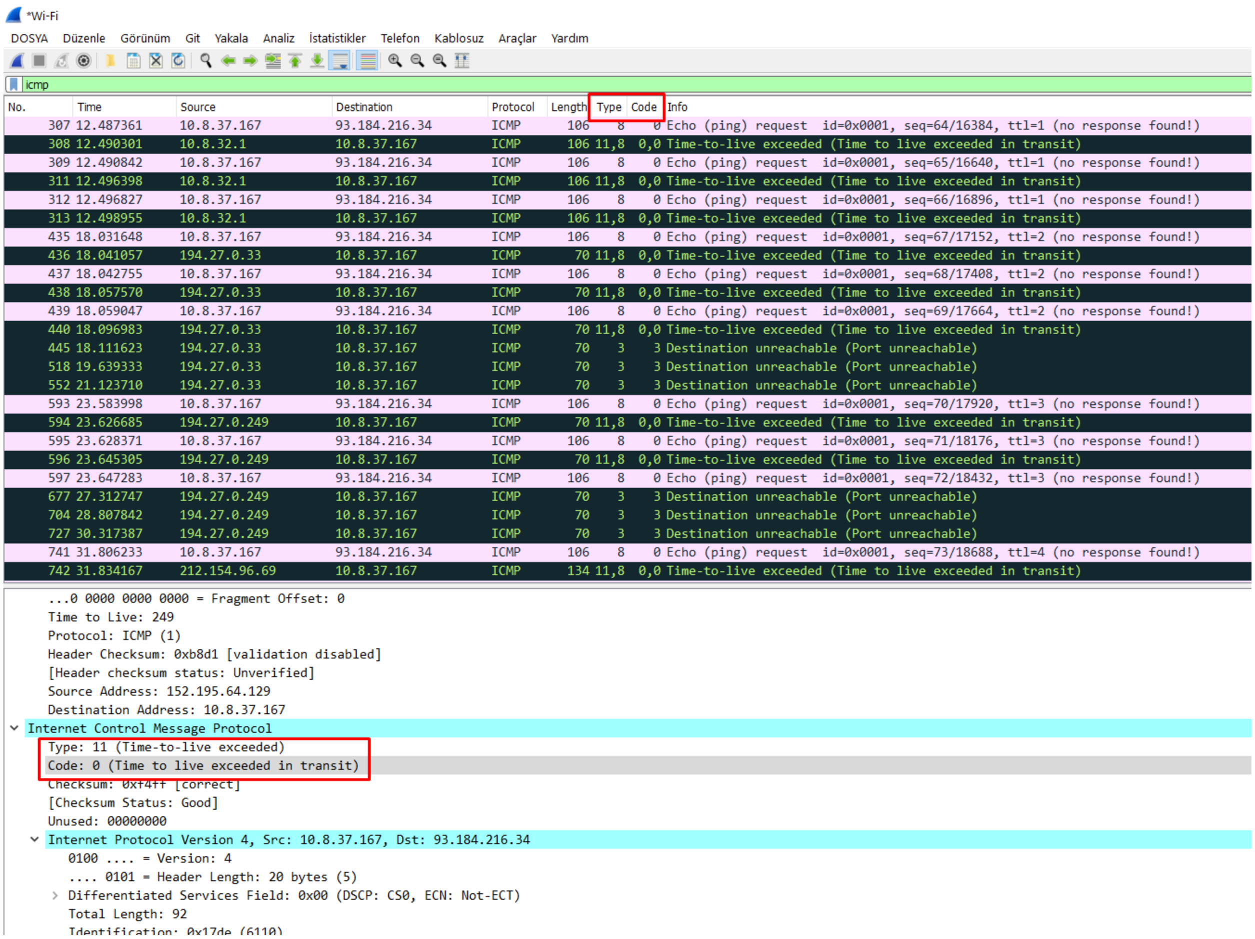

Fig4.1 ICMP Type & Code Column

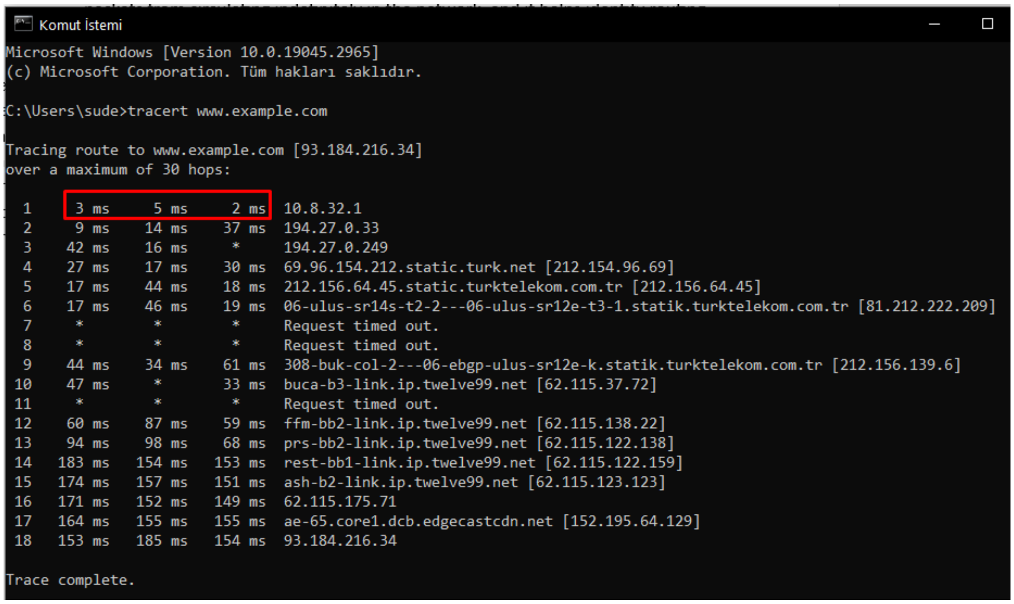

Fig4.2 Traceroute

NAT Section:

1. Network Address Translation (NAT) is a networking protocol that enables multiple devices on a private network to share a single public IP address when accessing the internet. NAT is typically implemented on routers, firewalls, or other network devices that act as a gateway between a private network and the public internet. NAT is a process in which one or more local IP addresses are translated into one or more Global IP address and vice versa in order to provide Internet access to the local hosts. NAT is often implemented at the WAN edge router to enable internet access in core, campus, branch, and colocation sites.

2. Packets are created as seen in *fig5.1*.

   a. If these packets were collected from a real environment, they could be collected at various points within the network infrastructure. Here are some possible locations where these packets might be captured:

      Endpoints: The LAN packet capture could be collected from an endpoint device within a local network, such as a desktop computer, laptop, or server. Similarly, the WAN packet capture could be collected from an endpoint device in a different network, such as a server or gateway.

Network Taps: Packet captures can be obtained by using network taps, which are physical devices connected to a network link to capture and monitor network traffic. Network taps can be placed at various points in the network infrastructure, such as between routers, switches, or firewall devices.

Network Monitoring Systems: Organizations often deploy network monitoring systems or network analyzers to capture and analyze network traffic. These systems can be set up to capture packets at specific network segments or interfaces, providing visibility into the network traffic passing through those points.

Network Security Appliances: Network security appliances, such as intrusion detection systems (IDS) or intrusion prevention systems (IPS), often capture packets for security analysis and threat detection. These appliances are typically placed at strategic points in the network to monitor and analyze traffic.

Internet Service Provider (ISP): Packet captures could also be collected by an Internet Service Provider (ISP) as part of their network monitoring and troubleshooting activities. ISPs may capture packets at their network gateways or points of presence (PoPs) to monitor and ensure the quality and reliability of their services.

It's important to note that the specific location where these packets are collected would depend on the network infrastructure and the purpose of the packet capture. Different organizations and network administrators may choose different points for capturing network traffic based on their requirements and objectives, such as troubleshooting, performance monitoring, or security analysis.

b. In the LAN packet capture (see fig5.2), we see communication between a device with the IP address 10.17.2.243 (source) and a server with the IP address 8.8.8.8 (destination). The communication is using both TCP and UDP protocols. Similarly, in the WAN packet capture (see fig5.3), we see communication between a device with the public IP address 102.37.24.187 (source) and the same server with the IP address 8.8.8.8 (destination).

The presence of a public IP address (102.37.24.187) in the WAN packet capture suggests that NAT is involved. When the LAN device communicates with the server on the WAN, the NAT protocol translates the private IP address (10.17.2.243) to a public IP address (102.37.24.187) before the packets are sent over the internet. This translation allows the LAN device to communicate with the server using a public IP address that can be routed on the WAN.

The LAN packet capture shows the communication initiated by the LAN device with a source port of 5000 and a destination port of 80. The WAN packet capture shows the translated packets with the source IP address of 102.37.24.187 and the same source and destination ports.

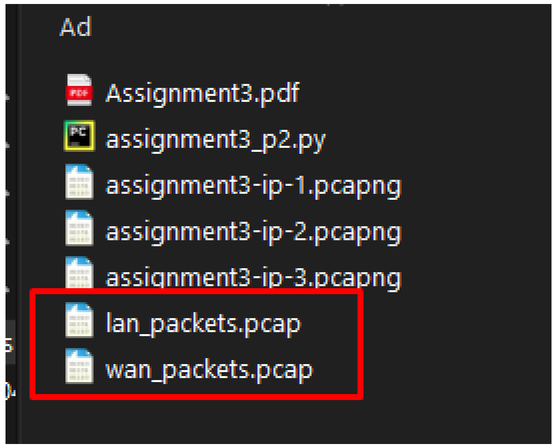

Fig5.1 Packets Created

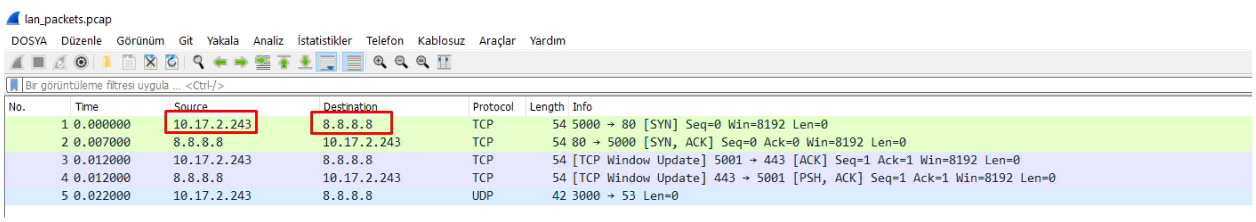

Fig5.2 Packets Created

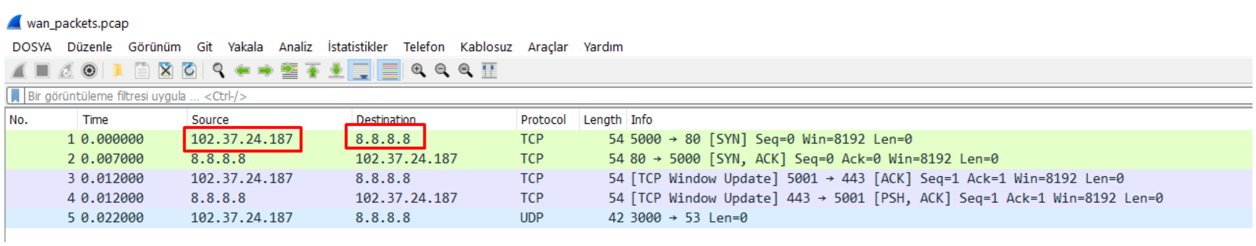

Fig5.3 Packets Created

DHCP Section:

i. The "ipconfig /release" and "ipconfig /renew" commands are related to the Dynamic Host Configuration Protocol (DHCP). DHCP is a network protocol used to automatically assign IP addresses and network configuration settings to devices on a network.

"ipconfig /release": This command is used to release the currently assigned IP address lease obtained through DHCP. When executed, it sends a request to the DHCP server to release the IP address associated with the network interface of the device. By releasing the IP address, the device informs the DHCP server that it no longer requires the IP address lease.

"ipconfig /renew": This command is used to request a new IP address lease from the DHCP server. When executed, it sends a DHCP request to the server, asking for a new IP address assignment. The DHCP server responds by providing the device with a new IP address lease, along with other network configuration settings such as subnet mask, default gateway, and DNS servers.

ii. One packet is created for ipconfig/release command (see fig6.2)
Four packet is created for first ipconfig/renew command (see fig6.3)
Two packet is created for first ipconfig/renew command (see fig6.4)

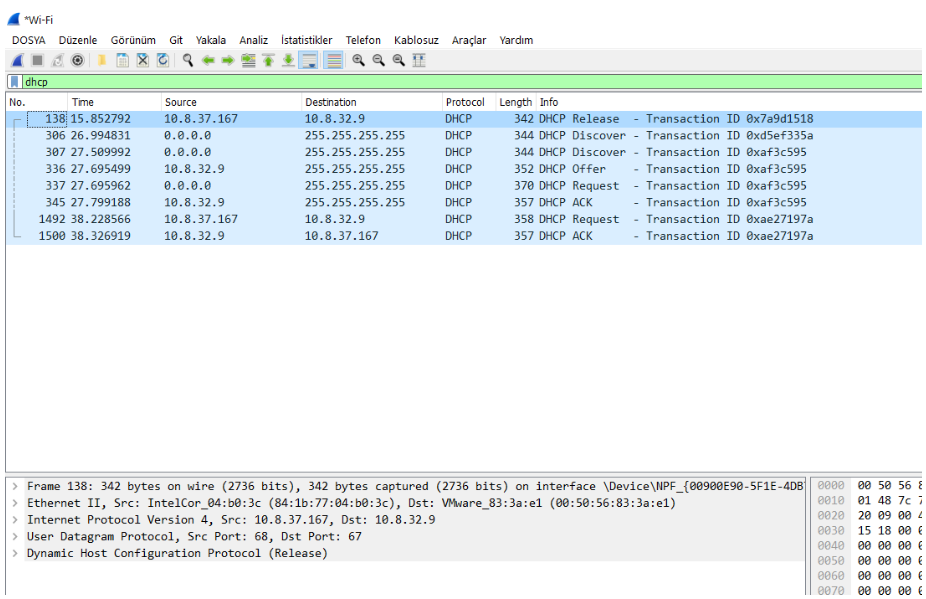

Fig6.1 Packets with DHCP Protocol

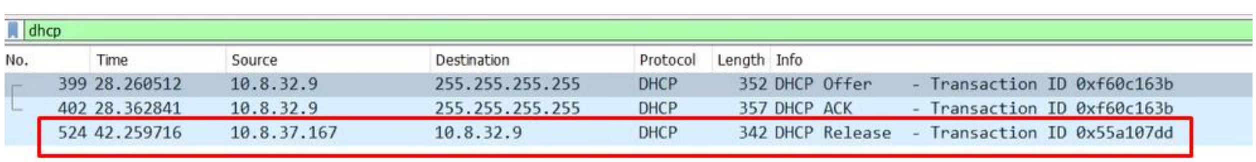

Fig6.2 One Packet Created for "ipconfig/release" Command

dhcp

No.	Time	Source	Destination	Protocol	Length	Info
399	28.260512	10.8.32.9	255.255.255.255	DHCP	352	DHCP Offer - Transaction ID 0xf60c163b
402	28.362841	10.8.32.9	255.255.255.255	DHCP	357	DHCP ACK - Transaction ID 0xf60c163b
524	42.259716	10.8.37.167	10.8.32.9	DHCP	342	DHCP Release - Transaction ID 0x55a107dd
593	47.308360	0.0.0.0	255.255.255.255	DHCP	344	DHCP Discover - Transaction ID 0x54556882
594	47.409280	10.8.32.9	255.255.255.255	DHCP	352	DHCP Offer - Transaction ID 0x54556882
595	47.409799	0.0.0.0	255.255.255.255	DHCP	370	DHCP Request - Transaction ID 0x54556882
601	47.512337	10.8.32.9	255.255.255.255	DHCP	357	DHCP ACK - Transaction ID 0x54556882

Fig6.3 Four Packets Created for first “ipconfig/renew” Command

dhcp

No.	Time	Source	Destination	Protocol	Length	Info
399	28.260512	10.8.32.9	255.255.255.255	DHCP	352	DHCP Offer - Transaction ID 0xf60c163b
402	28.362841	10.8.32.9	255.255.255.255	DHCP	357	DHCP ACK - Transaction ID 0xf60c163b
524	42.259716	10.8.37.167	10.8.32.9	DHCP	342	DHCP Release - Transaction ID 0x55a107dd
593	47.308360	0.0.0.0	255.255.255.255	DHCP	344	DHCP Discover - Transaction ID 0x54556882
594	47.409280	10.8.32.9	255.255.255.255	DHCP	352	DHCP Offer - Transaction ID 0x54556882
595	47.409799	0.0.0.0	255.255.255.255	DHCP	370	DHCP Request - Transaction ID 0x54556882
601	47.512337	10.8.32.9	255.255.255.255	DHCP	357	DHCP ACK - Transaction ID 0x54556882
1083	53.143840	10.8.37.167	10.8.32.9	DHCP	358	DHCP Request - Transaction ID 0x1133b75f
1086	53.150326	10.8.32.9	10.8.37.167	DHCP	357	DHCP ACK - Transaction ID 0x1133b75f

Fig6.4 Two Packets Created for second “ipconfig/renew” Command

iii. First "ipconfig /renew":

Packet 593: DHCP Discover - This packet is sent by the device to discover available DHCP servers on the network.
Packet 594: DHCP Offer - This packet is sent by the DHCP server in response to the DHCP Discover. It offers an IP address lease to the device.
Packet 595: DHCP Request - This packet is sent by the device to request the offered IP address lease from the DHCP server.
Packet 601: DHCP ACK - This packet is sent by the DHCP server in response to the DHCP Request. It acknowledges the request and provides the device with the IP address lease.

Second "ipconfig /renew":

Packet 1083: DHCP Request - This packet is sent by the device to request a new IP address lease from the DHCP server.
Packet 1086: DHCP ACK - This packet is sent by the DHCP server in response to the DHCP Request. It acknowledges the request and provides the device with the new IP address lease.
From the packet captures, we can see that the first "ipconfig /renew" command goes through the complete DHCP process, starting with a DHCP Discover, receiving a DHCP Offer, sending a DHCP Request, and receiving a DHCP ACK. This indicates that the device had no previous IP address lease or it expired, and it is obtaining a new IP address from the DHCP server.

On the other hand, the second "ipconfig /renew" command only involves a DHCP Request and a DHCP ACK. This suggests that the device already had an existing IP address lease, and it is simply requesting a renewal of the same IP address from the DHCP server.

Therefore, the differences in the captured packets reflect the different stages of the DHCP process and the specific actions taken by the device depending on whether it needs to obtain a new IP address or renew an existing IP address lease.

iv. DHCP packets are observed as the result of the first "ipconfig /renew" command:

DHCP Discover (Packet 593): This packet is sent by the device with the source IP address of 0.0.0.0 and the broadcast destination address (255.255.255.255). The DHCP Discover packet is used by the device to request network configuration information from DHCP servers on the network. The Transaction ID (0x54556882) helps identify this specific DHCP transaction.

DHCP Offer (Packet 594): This packet is sent by a DHCP server in response to the DHCP Discover. It has the DHCP server's IP address (10.8.32.9) as the source and a broadcast destination address. The DHCP Offer packet provides an IP address lease offer to the device, along with other network configuration parameters. The Transaction ID (0x54556882) matches the one in the DHCP Discover packet.

DHCP Request (Packet 595): This packet is sent by the device to accept the offered IP address lease from the DHCP server. It has the source IP address of 0.0.0.0 and the broadcast destination address. The DHCP Request packet indicates the device's acceptance of the DHCP server's offer. The Transaction ID (0x54556882) matches the one in the DHCP Discover packet.

DHCP ACK (Packet 601): This packet is sent by the DHCP server in response to the DHCP Request. It acknowledges the device's acceptance of the IP address lease. The DHCP ACK packet provides the device with the confirmed IP address lease and additional network configuration details. The Transaction ID (0x54556882) matches the one in the DHCP Discover packet.

Therefore, the captured packets for the first "ipconfig /renew" command include the DHCP Discover, DHCP Offer, DHCP Request, and DHCP ACK packets. These DHCP packet types are fundamental to the DHCP process, allowing devices to obtain IP addresses and network settings dynamically from DHCP servers.

v. Yes, by inspecting the option fields in the captured DHCP packets as the result of the first "ipconfig /renew" command, there are common options present in all packets (see fig6.5 & fig6.6 & fig6.7 & fig6.7). These common options are as follows:

DHCP Message Type (Option 53): This option specifies the type of DHCP message being transmitted. In all captured packets, this option is present and indicates the message type for each DHCP packet (Discover, Offer, Request, ACK).

End Option (Option 255): The End option is a standard DHCP option and is mandatory to signify the completion of the options section within the DHCP packet. It ensures that the DHCP client or server parsing the packet knows where the options field ends and where other fields or sections of the packet begin.

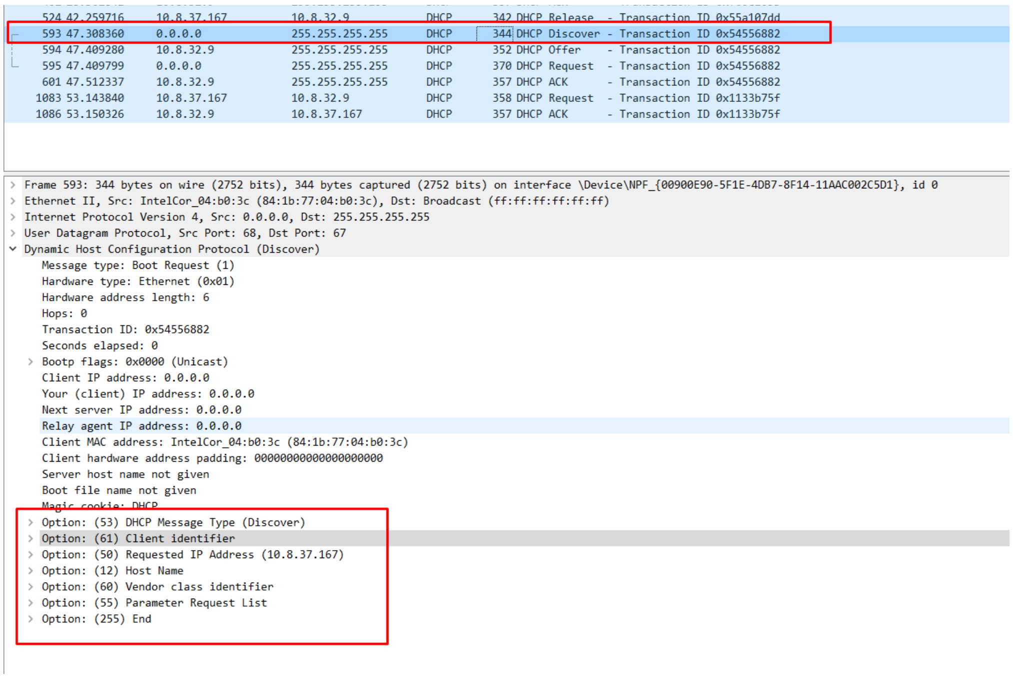

Fig6.5 Options of the first packet of the first ipconfig/release command

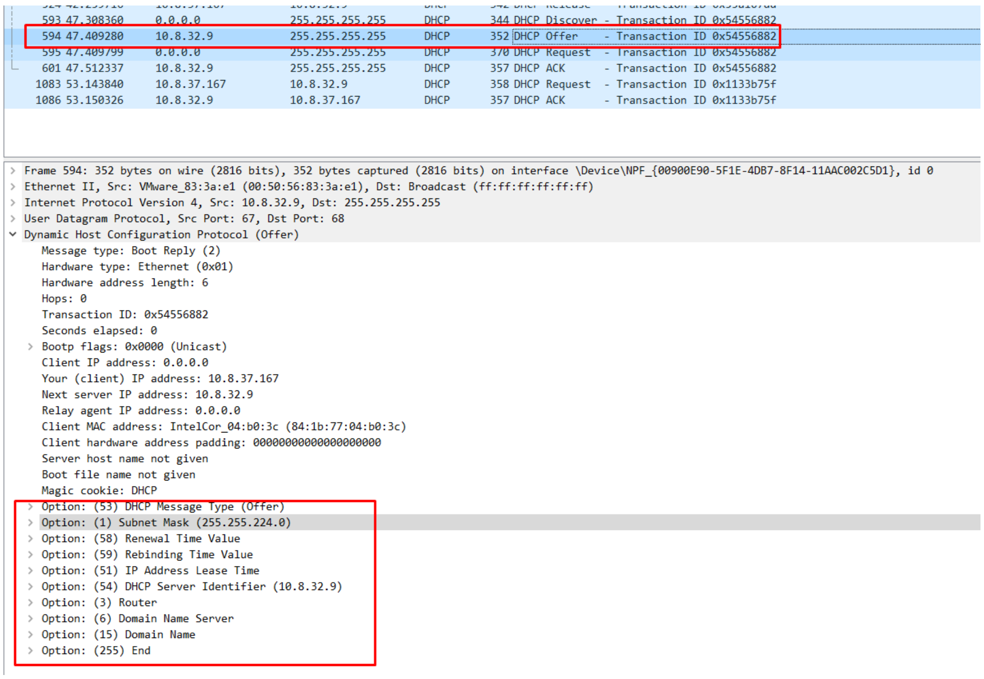

Fig6.6 Options of the second packet of the first ipconfig/release command

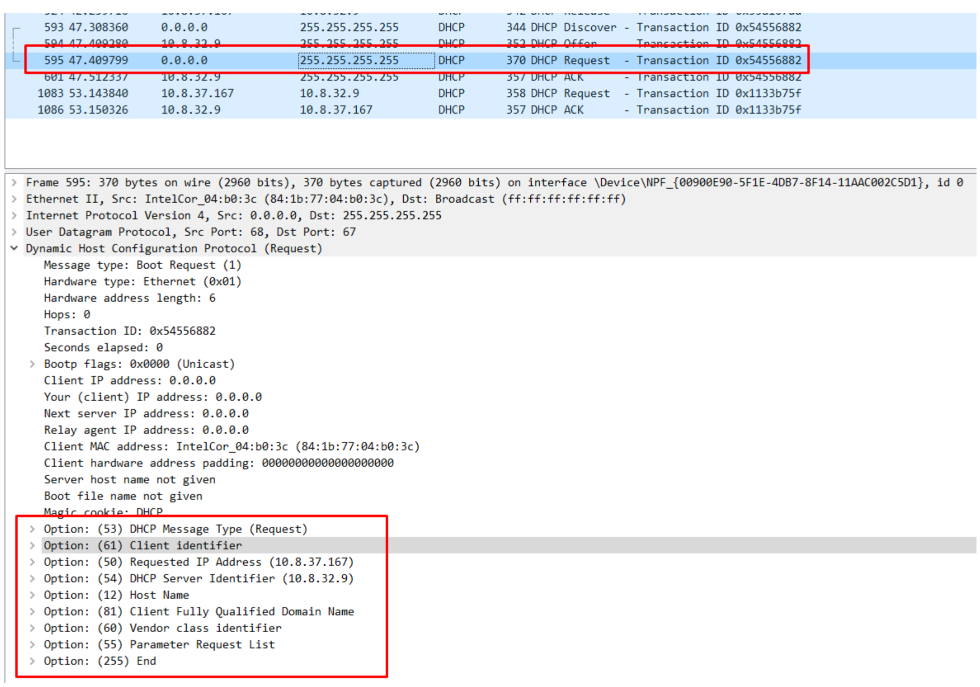

Fig6.7 Options of the third packet of the first ipconfig/release command

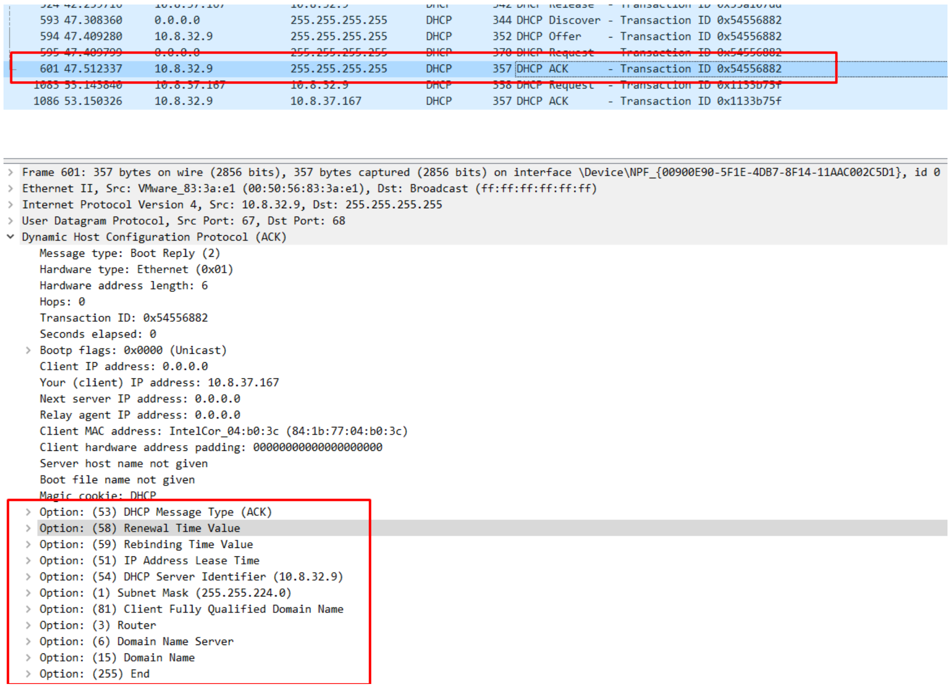

Fig6.8 Options of the fourth packet of the first ipconfig/release command