

# Analyze a Packet With Wireshark

## Skills Acquired:

- Open saved packet capture files
- View high-level packet data
- Use filters to inspect detailed packet data

## Scenario:

In this scenario, you're a security analyst investigating traffic to a website.

You'll analyze a network packet capture file that contains traffic data related to a user connecting to an internet site. The ability to filter network traffic using packet sniffers to gather relevant information is an essential skill as a security analyst.

You must filter the data in order to:

- identify the source and destination IP addresses involved in this web browsing session,
- examine the protocols that are used when the user makes the connection to the website, and
- analyze some of the data packets to identify the type of information sent and received by the systems that connect to each other when the network data is captured.

# Task 1. Explore data with Wireshark

1. Open the packet capture file by double-clicking the file called **sample** on the Windows desktop. Wireshark starts.
2. Double-click the Wireshark title bar next to the **sample.pcap** filename to maximize the Wireshark application window.
3. Scroll down the packet list until a packet is listed where the info column starts with the words 'Echo (ping) request'.
  - **What is the protocol of the first packet in the list where the info column starts with the words 'Echo (ping) request'?**

## ICMP

No.	Time	Source	Destination	Protocol	Length	Info
16	8.642690	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=1/2
17	8.642755	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq=161 Ack=561 Win=10
18	8.643923	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=1/2
19	8.644093	172.21.224.2	169.254.169.254	DNS	86	Standard query 0xb549 PTR 139.1.250.142
20	8.647339	169.254.169.254	172.21.224.2	DNS	120	Standard query response 0xb549 PTR 139.1.250.142
21	8.647514	172.21.224.2	35.235.244.34	SSH	210	Server: Encrypted packet (len=144)
22	8.647587	172.21.224.2	35.235.244.34	SSH	130	Server: Encrypted packet (len=64)
23	8.647668	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq=161 Ack=705 Win=10
24	8.647682	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq=161 Ack=769 Win=10
25	9.644712	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=2/5
26	9.645078	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=2/5
27	9.645214	172.21.224.2	169.254.169.254	DNS	86	Standard query 0x3cdc PTR 139.1.250.142
28	9.645859	169.254.169.254	172.21.224.2	DNS	120	Standard query response 0x3cdc PTR 139.
29	9.646069	172.21.224.2	35.235.244.34	SSH	210	Server: Encrypted packet (len=144)
30	9.646203	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq=161 Ack=913 Win=10
31	10.646040	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831 seq=3/7

## Task 2. Apply a basic Wireshark filter and inspect a packet

In this task, you'll open a packet in Wireshark for more detailed exploration and filter the data to inspect the network layers and protocols contained in the packet.

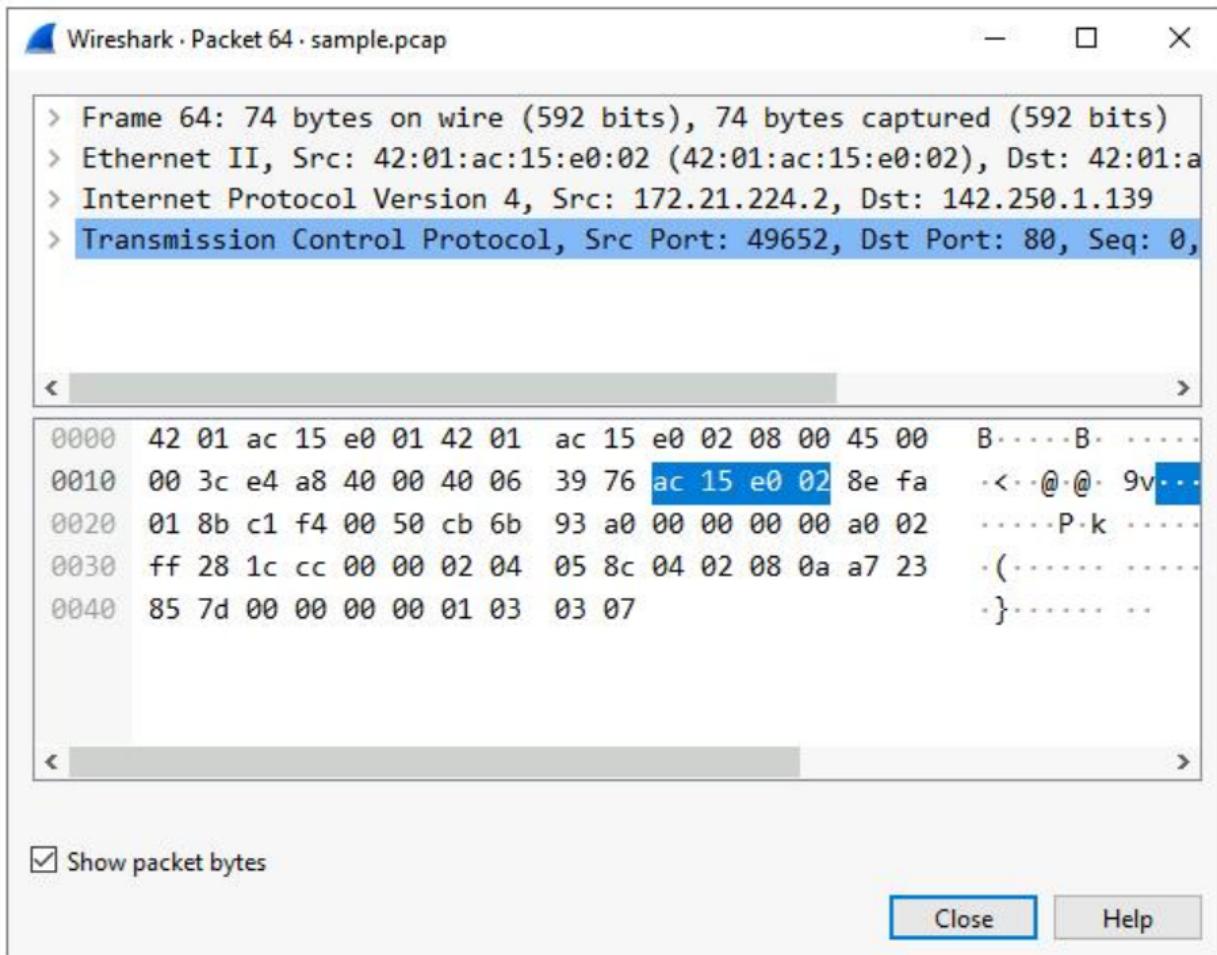
1. Enter the following filter for traffic associated with a specific IP address. Enter this into the **Apply a display filter...** text box immediately above the list of packets:

```
ip.addr == 142.250.1.139
```

2. Press **ENTER** or click the **Apply display filter** icon in the filter text box.

The list of packets displayed is now significantly reduced and contains only packets where either the source or the destination IP address matches the address you entered. Now only two packet colors are used: light pink for ICMP protocol packets and light green for TCP (and HTTP, which is a subset of TCP) packets.

3. Double-click the first packet that lists **TCP** as the protocol.



The upper section of this window contains subtrees where Wireshark will provide you with an analysis of the various parts of the network packet. The lower section of the window contains the raw packet data displayed in hexadecimal and ASCII text. There is also placeholder text for fields where the character data does not apply, as indicated by the dot (".").

4. Double-click the first subtree in the upper section. This starts with the word **Frame**.

This provides you with details about the overall network packet, or frame, including the frame length and the arrival time of the packet. At this level, you're viewing information about the entire packet of data.

```
▼ Frame 64: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
  Encapsulation type: Ethernet (1)
  Arrival Time: Nov 23, 2022 12:38:34.620693000 Greenwich Standard Time
    [Time shift for this packet: 0.000000000 seconds]
  Epoch Time: 1669207114.620693000 seconds
    [Time delta from previous captured frame: 0.000389000 seconds]
    [Time delta from previous displayed frame: 7.386205000 seconds]
    [Time since reference or first frame: 18.032768000 seconds]
  Frame Number: 64
  Frame Length: 74 bytes (592 bits)
  Capture Length: 74 bytes (592 bits)
  [Frame is marked: False]
```

5. Double-click **Frame** again to collapse the subtree and then double-click the **Ethernet II** subtree.

This item contains details about the packet at the Ethernet level, including the source and destination MAC addresses and the type of internal protocol that the Ethernet packet contains.

```
> Frame 64: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
▼ Ethernet II, Src: 42:01:ac:15:e0:02 (42:01:ac:15:e0:02), Dst: 42:01:ac:15:e0:01 (42:01:ac:15:e0:01)
  > Destination: 42:01:ac:15:e0:01 (42:01:ac:15:e0:01)
  > Source: 42:01:ac:15:e0:02 (42:01:ac:15:e0:02)
  Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 172.21.224.2, Dst: 142.250.1.139
> Transmission Control Protocol, Src Port: 49652, Dst Port: 80, Seq: 0, Len: 0
```

6. Double-click **Ethernet II** again to collapse that subtree and then double-click the **Internet Protocol Version 4** subtree.

This provides packet data about the Internet Protocol (IP) data contained in the Ethernet packet. It contains information such as the source and destination IP addresses and the Internal Protocol (for example, TCP or UDP), which is carried inside the IP packet.

The source and destination IP addresses shown here match the source and destination IP addresses in the summary display for this packet in the main Wireshark window.

```
> Frame 64: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
> Ethernet II, Src: 42:01:ac:15:e0:02 (42:01:ac:15:e0:02), Dst: 42:01:ac:15:e0:01 (42:01:ac:15:e0:01)
└ Internet Protocol Version 4, Src: 172.21.224.2, Dst: 142.250.1.139
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
        Total Length: 60
        Identification: 0xe4a8 (58536)
    > 010. .... = Flags: 0x2, Don't fragment
        ...0 0000 0000 0000 = Fragment Offset: 0
        Time to Live: 64
        Protocol: TCP (6)
```

7. Double-click **Internet Protocol Version 4** again to collapse that subtree and then double-click the **Transmission Control Protocol** subtree.

This provides detailed information about the TCP packet, including the source and destination TCP ports, the TCP sequence numbers, and the TCP flags.

The source port and destination port listed here match the source and destination ports in the info column of the summary display for this packet in the list of all of the packets in the main Wireshark window.

- **What is the TCP destination port of this TCP packet?**

```
> Frame 64: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
> Ethernet II, Src: 42:01:ac:15:e0:02 (42:01:ac:15:e0:02), Dst: 42:01:ac:15:e0:01 (42:01:ac:15:e0:01)
> Internet Protocol Version 4, Src: 172.21.224.2, Dst: 142.250.1.139
▼ Transmission Control Protocol, Src Port: 49652, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 49652
    Destination Port: 80
    [Stream index: 4]
    [Conversation completeness: Complete, WITH_DATA (31)]
    [TCP Segment Len: 0]
    Sequence Number: 0      (relative sequence number)
    Sequence Number (raw): 3412824992
    [Next Sequence Number: 1      (relative sequence number)]
```

8. In the **Transmission Control Protocol** subtree, scroll down and double-click

**Flags.**

This provides a detailed view of the TCP flags set in this packet.

```
Acknowledgment Number: 0
Acknowledgment number (raw): 0
1010 .... = Header Length: 40 bytes (10)
▼ Flags: 0x002 (SYN)
    000. .... .... = Reserved: Not set
    ...0 .... .... = Accurate ECN: Not set
    .... 0.... .... = Congestion Window Reduced: Not set
    .... .0... .... = ECN-Echo: Not set
    .... ..0. .... = Urgent: Not set
    .... ...0 .... = Acknowledgment: Not set
    .... .... 0... = Push: Not set
    .... .... .0.. = Reset: Not set
```

9. Click the **X** icon to close the detailed packet inspection window.

10. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.

All the packets have returned to the display.

## Task 3. Use filters to select packets

In this task, you'll use filters to analyze specific network packets based on where the packets came from or where they were sent to. You'll explore how to select packets using either their physical Ethernet Media Access Control (MAC) address or their Internet Protocol (IP) address.

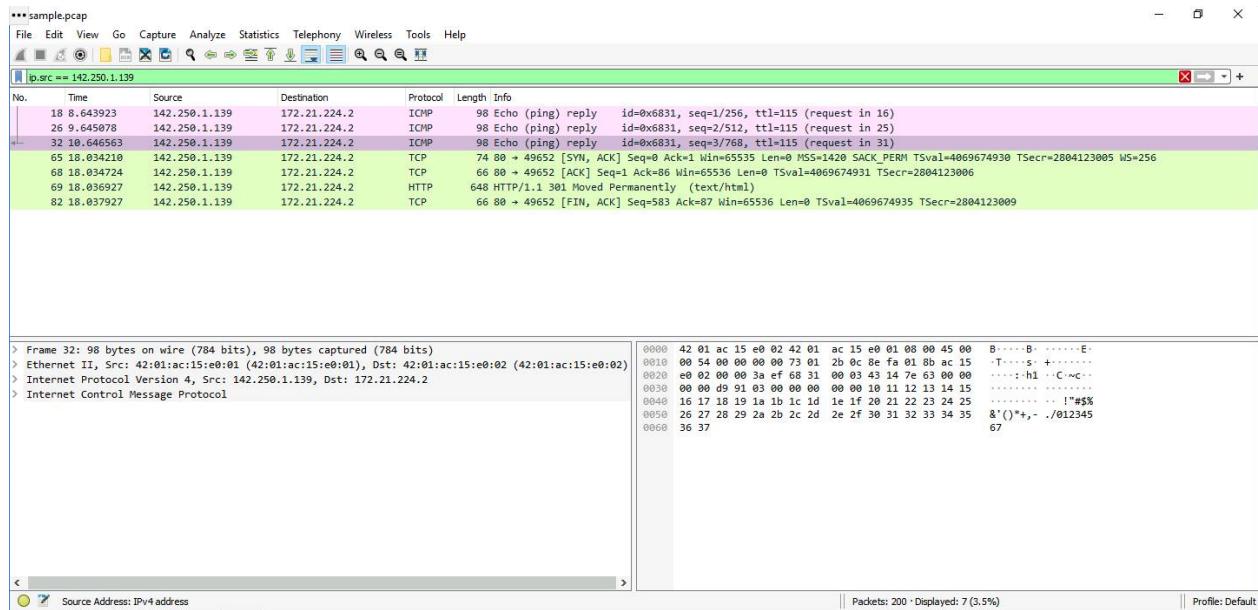
1. Enter the following filter to select traffic for a specific source IP address only.

Enter this into the **Apply a display filter...** text box immediately above the list of packets:

```
ip.src == 142.250.1.139
```

2. Press **ENTER** or click the **Apply display filter** icon in the filter text box.

A filtered list is returned with fewer entries than before. It contains only packets that came from 142.250.1.139.

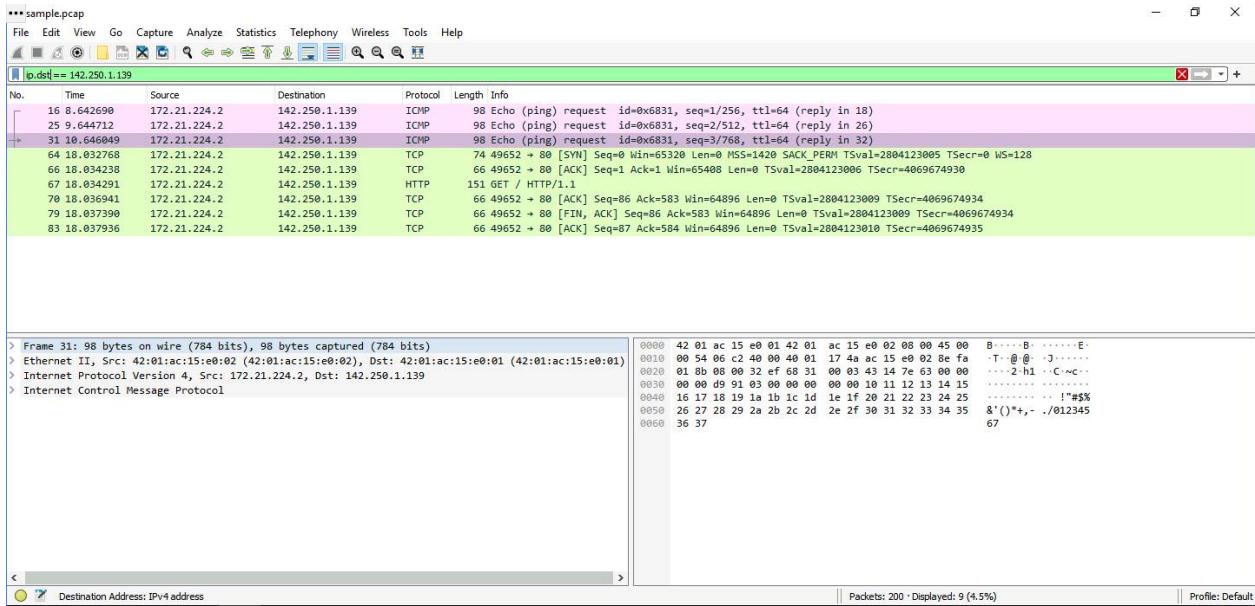


3. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.
4. Enter the following filter to select traffic for a specific destination IP address only:

ip.dst == 142.250.1.139

5. Press **ENTER** or click the **Apply display filter** icon in the filter text box.

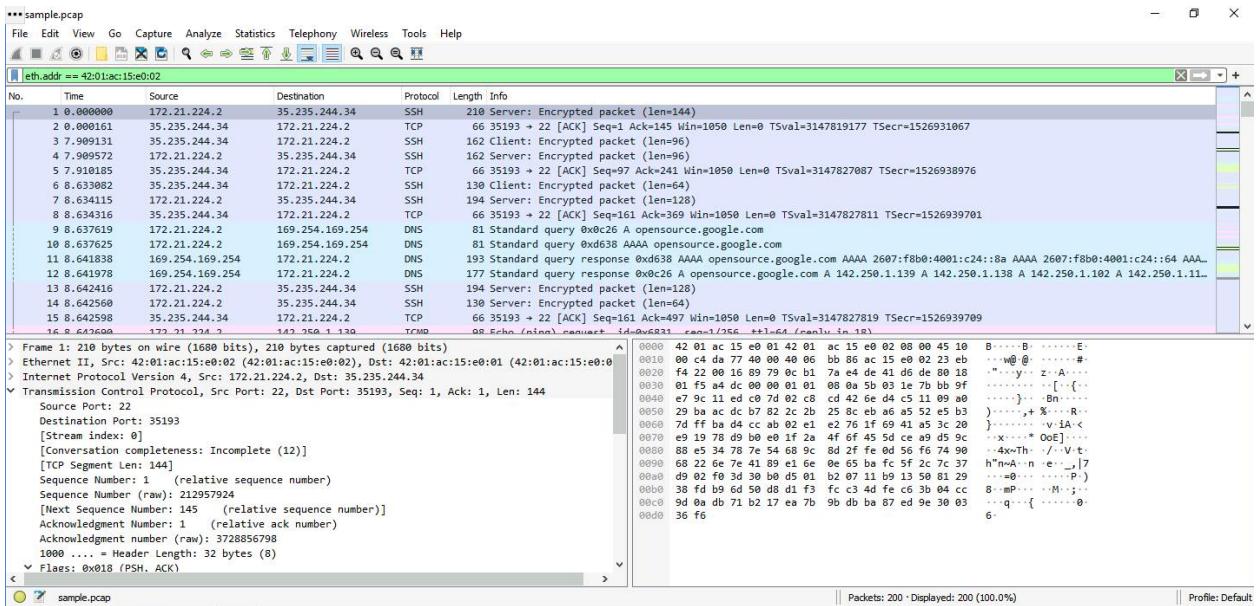
A filtered list is returned that contains only packets that were sent to 142.250.1.139.



6. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.
7. Enter the following filter to select traffic to or from a specific Ethernet MAC address. This filters traffic related to one MAC address, regardless of the other protocols involved:

```
eth.addr == 42:01:ac:15:e0:02
```

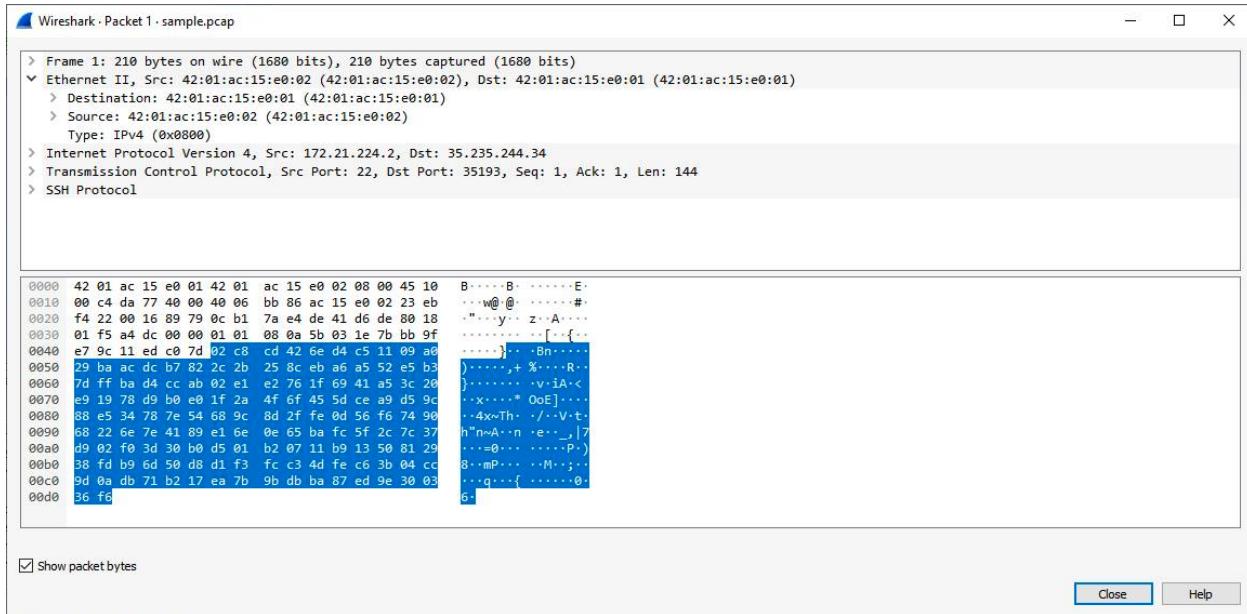
8. Press **ENTER** or click the **Apply display filter** icon in the filter text box.



9. Double-click the first packet in the list. You may need to scroll back to display the first packet in the filtered list.

10. Double-click the **Ethernet II** subtree if it is not already open.

The MAC address you specified in the filter is listed as either the source or destination address in the expanded Ethernet II subtree.



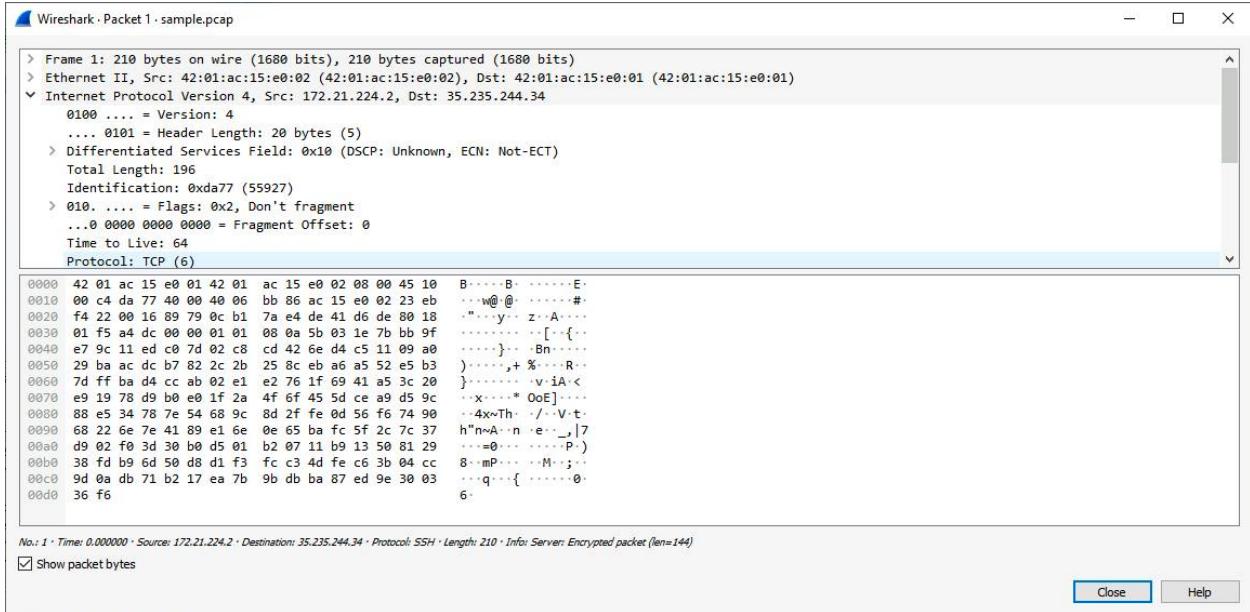
11. Double-click the **Ethernet II** subtree to close it.

12. Double-click the **Internet Protocol Version 4** subtree to expand it and scroll down until the **Time to Live** and **Protocol** fields appear.

The **Protocol** field in the **Internet Protocol Version 4** subtree indicates which IP internal protocol is contained in the packet.

- **What is the protocol contained in the Internet Protocol Version 4 subtree from the first packet related to MAC address 42:01:ac:15:e0:02?**

TCP



12. Click the **X** icon to close the detailed packet inspection window.
13. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the MAC address filter.

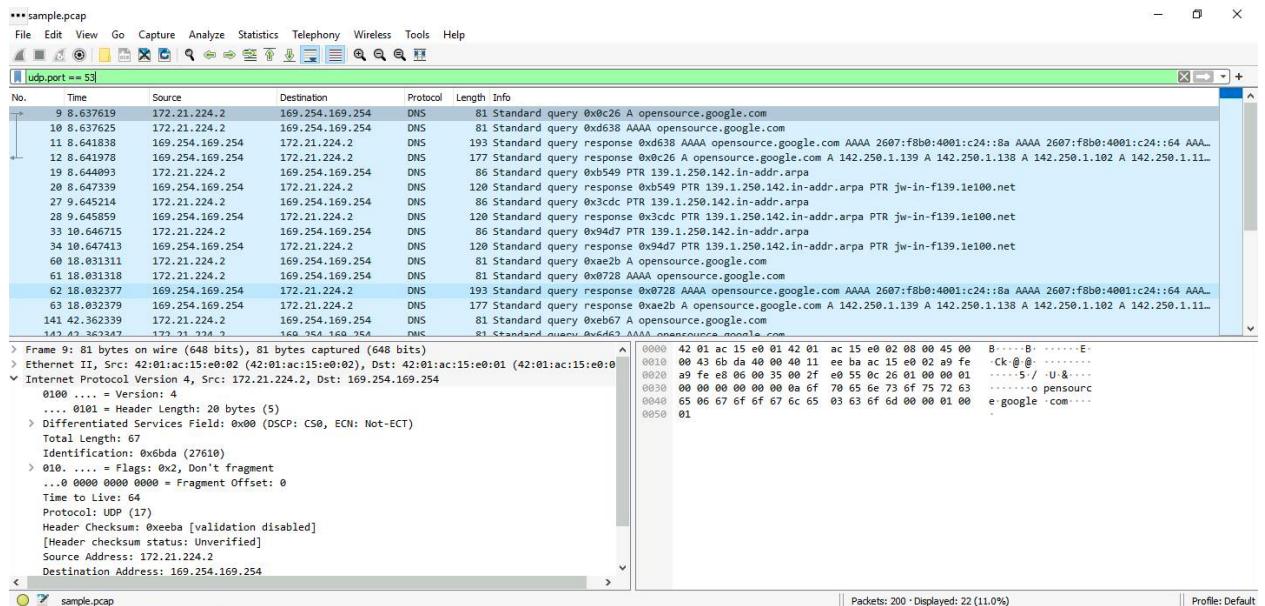
# Task 4. Use filters to explore DNS packets

In this task, you'll use filters to select and examine DNS traffic. Once you've selected sample DNS traffic, you'll drill down into the protocol to examine how the DNS packet data contains both queries (names of internet sites that are being looked up) and answers (IP addresses that are being sent back by a DNS server when a name is successfully resolved).

1. Enter the following filter to select UDP port 53 traffic. DNS traffic uses UDP port 53, so this will list traffic related to DNS queries and responses only. Enter this into the **Apply a display filter...** text box immediately above the list of packets:

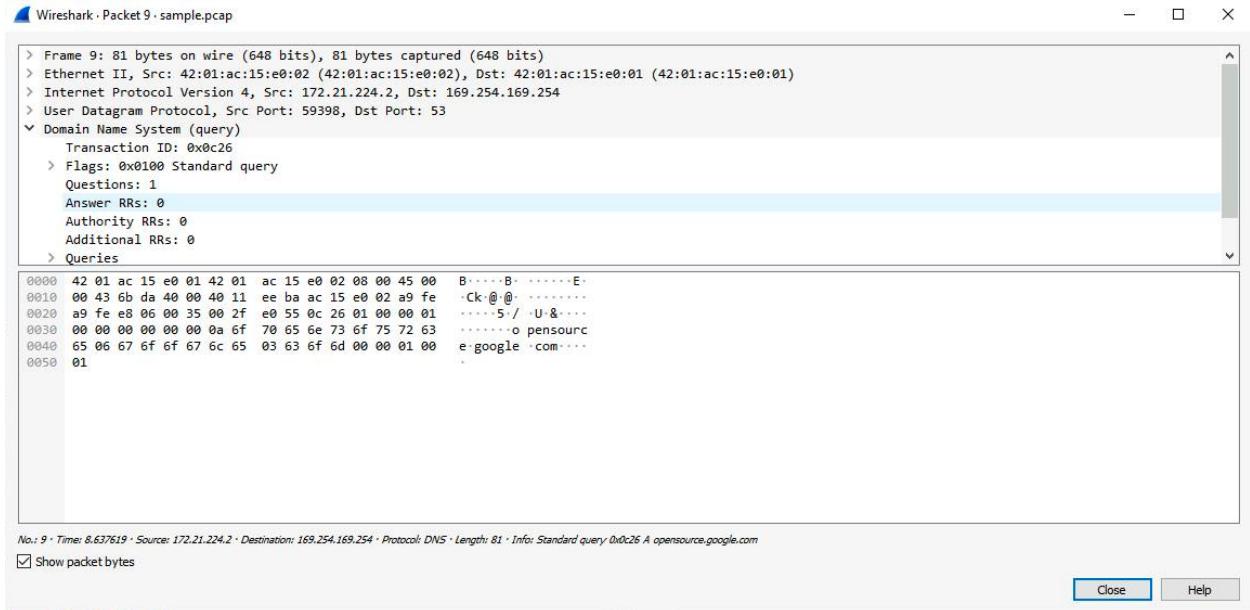
```
udp.port == 53
```

2. Press **ENTER** or click the **Apply display filter** icon in the filter text box.

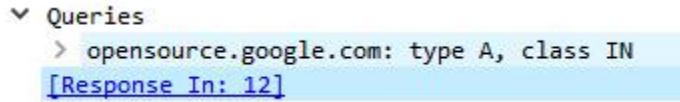


3. Double-click the first packet in the list to open the detailed packet window.

4. Scroll down and double-click the **Domain Name System (query)** subtree to expand it.



5. Scroll down and double-click **Queries**.



You'll notice that the name of the website that was queried is **opensource.google.com**.

6. Click the X icon to close the detailed packet inspection window.
7. Double-click the fourth packet in the list to open the detailed packet window.
8. Scroll down and double-click the **Domain Name System (query)** subtree to expand it.
9. Scroll down and double-click **Answers**, which is in the **Domain Name System (query)** subtree.

The Answers data includes the name that was queried ([opensource.google.com](https://opensource.google.com)) and the addresses that are associated with that name.

- **What is the protocol contained in the Internet Protocol Version 4 subtree from the first packet related to MAC address 42:01:ac:15:e0:02?**

142.250.1.139

10. Click the **X** icon to close the detailed packet inspection window.

11. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the filter.

# Task 5. Use filters to explore TCP packets

In this task, you'll use additional filters to select and examine TCP packets. You'll learn how to search for text that is present in payload data contained inside network packets. This will locate packets based on something such as a name or some other text that is of interest to you.

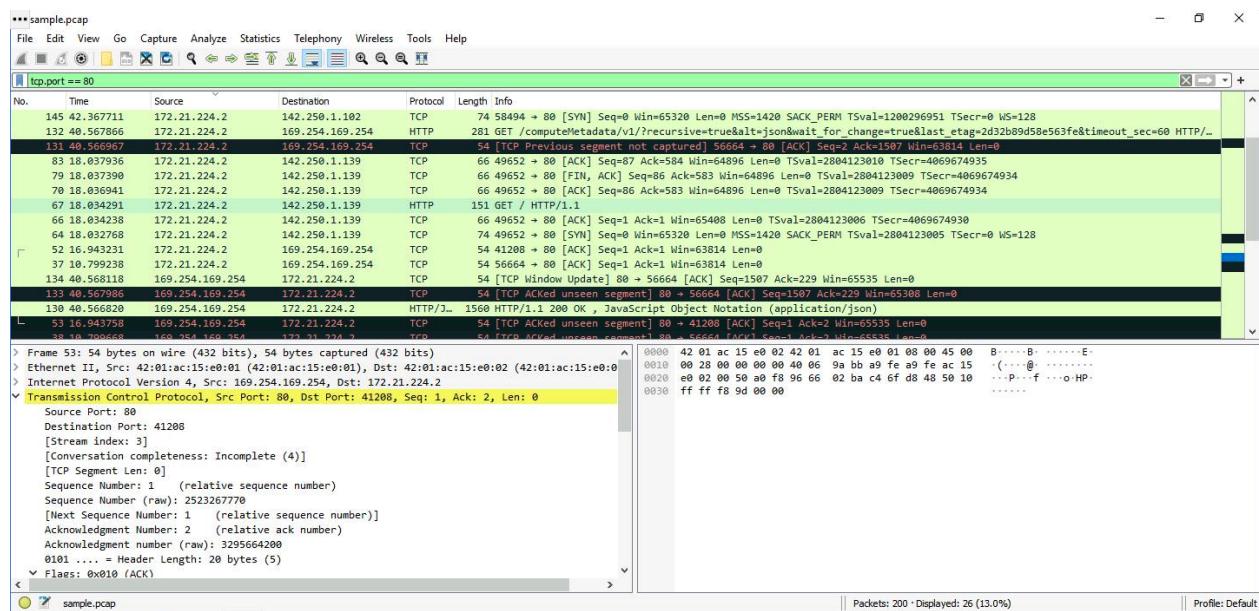
1. Enter the following filter to select TCP port 80 traffic. TCP port 80 is the default port that is associated with web traffic:

```
tcp.port == 80
```

2. Press **ENTER** or click the **Apply display filter** icon in the filter text box.

Quite a few packets were created when the user accessed the web page

<http://opensource.google.com>.



3. Double-click the first packet in the list. The **Destination** IP address of this packet is 169.254.169.254.

- **What is the Time to Live value of the packet as specified in the Internet Protocol Version 4 subtree?**

64

The screenshot shows the Wireshark interface with the title "Wireshark - Packet 53 - sample.pcap". The packet details pane at the top displays the following information for the selected packet:

- 0... .... = Reserved bit: Not set
- .0... .... = Don't fragment: Not set
- ..0.... = More fragments: Not set
- ...0 0000 0000 0000 = Fragment Offset: 0
- Time to Live: 64
- Protocol: TCP (6)
- Header Checksum: 0x9abb [validation disabled]  
[Header checksum status: Unverified]
- Source Address: 169.254.169.254
- Destination Address: 172.21.224.2

The selected packet is highlighted in yellow and shows the following bytes:

0000	42 01 ac 15 e0 02 42 01 ac 15 e0 01 08 00 45 00	B.....B.....E
0010	00 28 00 00 00 00 40 06 9a bb a9 fe a9 fe ac 15	(.....@.....)
0020	e0 02 00 50 a0 f8 96 66 02 ba c4 6f d8 48 50 10	...P...f...o-HP
0030	ff ff f8 9d 00 00 00	.....

- **What is the Frame Length of the packet as specified in the Frame subtree?**

54 bytes

```

Frame 53: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
Encapsulation type: Ethernet (1)
Arrival Time: Nov 23, 2022 12:38:33.531683000 Greenwich Standard Time
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1669207113.531683000 seconds
Frame Number: 53
Frame Length: 54 bytes (432 bits)
Capture Length: 54 bytes (432 bits)
[Frame is marked: False]

0000  42 01 ac 15 e0 02 42 01 ac 15 e0 01 08 00 45 00 B.....B.....E
0010  00 28 00 00 00 00 40 06 9a bb a9 fe a9 fe ac 15 .....@.....
0020  e0 02 00 50 a0 f8 96 66 02 ba c4 6f d8 48 50 10 .....P...f...o.HP
0030  ff ff f8 9d 00 00 .....
```

- What is the Header Length of the packet as specified in the Internet Protocol Version 4 subtree?

20 bytes

```

Frame 132: 281 bytes on wire (2248 bits), 281 bytes captured (2248 bits)
Ethernet II, Src: 42:01:ac:15:e0:02 (42:01:ac:15:e0:02), Dst: 42:01:ac:15:e0:01 (42:01:ac:15:e0:01)
Internet Protocol Version 4, Src: 172.21.224.2, Dst: 169.254.169.254
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
.... Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    0000 00.. = Differentiated Services Codepoint: Default (0)
    .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
Total Length: 267
Identification: 0x89a4 (35236)
010. .... = Flags: 0x2, Don't fragment

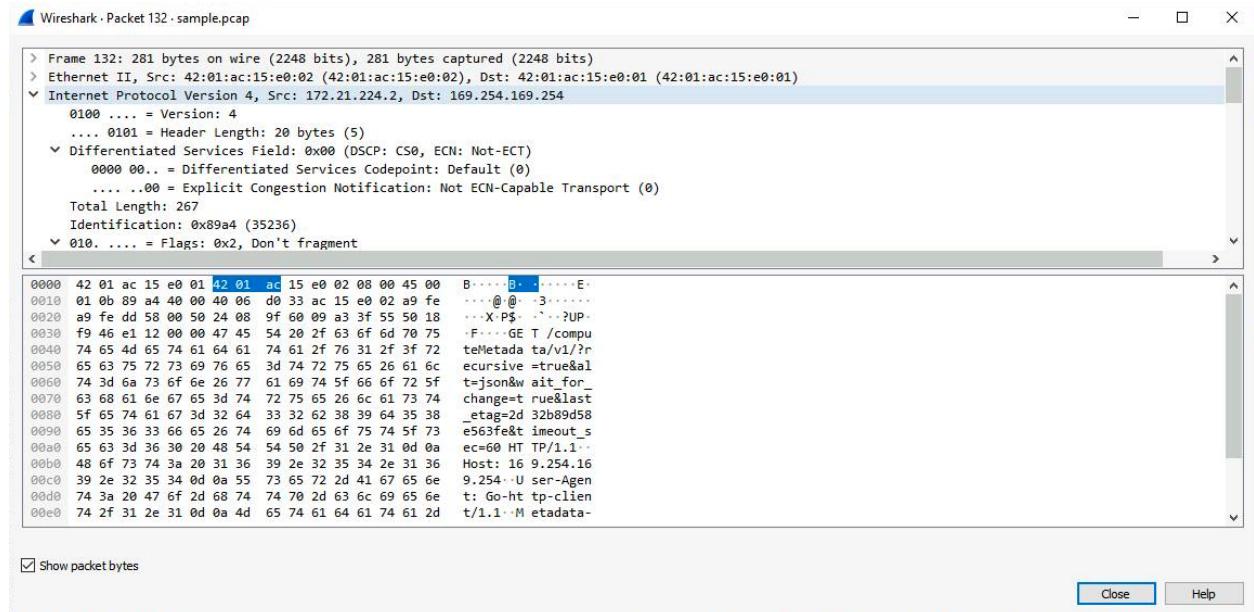
0000  42 01 ac 15 e0 01 42 01 ac 15 e0 02 08 00 45 00 B.....B.....E
0010  01 0b 89 a4 40 00 40 06 d8 33 ac 15 e0 02 a9 fe .....@: 3.....
0020  a9 fe dd 58 00 50 24 08 9f 60 09 a3 3f 55 50 18 .....X P$.....?UP
0030  f9 46 e1 12 00 00 47 45 54 20 2f 63 6f 6d 70 75 .....F.....GE T /compu
0040  74 65 4d 65 74 61 64 61 74 61 2f 76 31 2f 3f 72 teMetadata ta/v1/?r
0050  65 63 75 72 73 69 76 65 3d 74 72 75 65 26 61 6c recursive=true&al
0060  74 3d 6a 73 6f 6e 26 77 61 69 74 5f 66 6f 72 5f t=json&ait_for_
0070  63 68 61 6e 67 65 3d 74 72 75 65 26 6c 61 73 74 change=t rue&last
0080  5f 65 74 61 67 3d 32 64 33 32 62 38 39 64 35 38 _etag=2d 32b89d58
0090  65 35 36 33 66 65 26 74 69 6d 65 6f 75 74 5f 73 e563fe&t imout_s
00a0  65 63 3d 3b 20 48 54 54 50 2f 31 2e 31 0d 0a ec=0@ HT TP/1.1
00b0  48 6f 73 74 3a 20 31 36 39 2e 32 35 34 2e 31 36 Host: 16 9.254.16
00c0  39 2e 32 35 34 0d 0a 55 73 65 72 2d 41 67 65 6e 9.254.1U ser-Agen
00d0  74 3a 20 47 6f 2d 68 74 74 70 2d 63 6c 69 65 6e t: Go-ht tp-clien
00e0  74 2f 31 2e 31 0d 0a 4d 65 74 61 64 61 74 61 2d t/1.1.M etadata-
```

Show packet bytes

Close Help

- What is the Destination Address as specified in the Internet Protocol Version 4 subtree?

169.254.169.254



4. Click the **X** icon to close the detailed packet inspection window.
5. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the filter.
6. Enter the following filter to select TCP packet data that contains specific text data.

tcp contains "curl"

7. Press **ENTER** or click the **Apply display filter** icon in the filter text box.

This filters to packets containing web requests made with the curl command in this sample packet capture file.

