Analyze your first packet with Wireshark

Summary:

The "Analyze Your First Packet with Wireshark" project provided foundational experience in network traffic analysis, offering insights into packet structure and data transmission within networks. Through this activity, I gained hands-on practice using Wireshark to capture, inspect, and interpret network packets—a critical skill for identifying and mitigating potential security threats. This project marks the beginning of my journey into network analysis, and I plan to expand my expertise further by pursuing additional classes and certifications to solidify my proficiency with this essential cybersecurity tool.

Activity Review:

As a security analyst, analyzing network traffic is a crucial skill for understanding the type of data being sent to and from systems within the networks you'll work with. This activity involved using Wireshark to inspect packet data and applying filters to efficiently sort through packet information. It emphasized practical techniques for examining network traffic to uncover meaningful insights for threat detection and resolution.

Scenario:

In this scenario, you are a security analyst investigating website traffic. Your task is to analyze a network packet capture file containing traffic data related to a user's connection to an internet site. Filtering network traffic using packet sniffers like Wireshark to gather relevant information is an essential skill for a security analyst.

Your goals for this scenario include:

- Identifying the source and destination IP addresses involved in the web browsing session.
- Examining the protocols used when the user connects to the website.
- Analyzing data packets to identify the type of information sent and received during the network session.

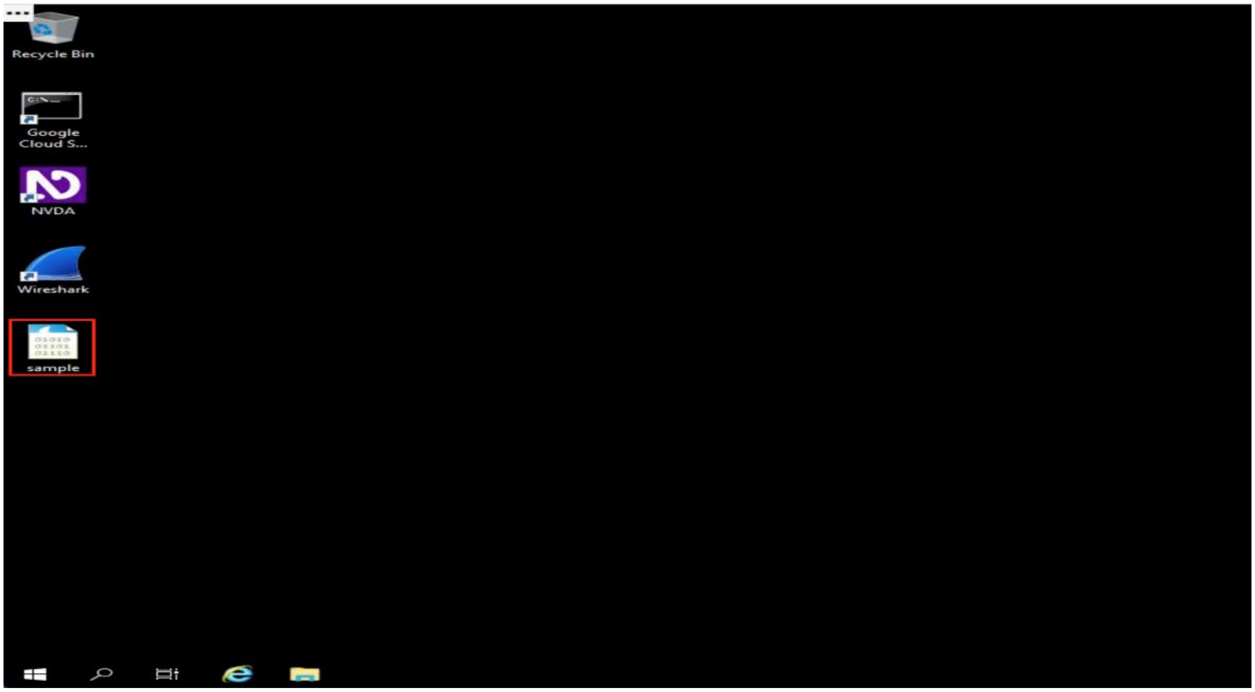
Steps to Achieve This:

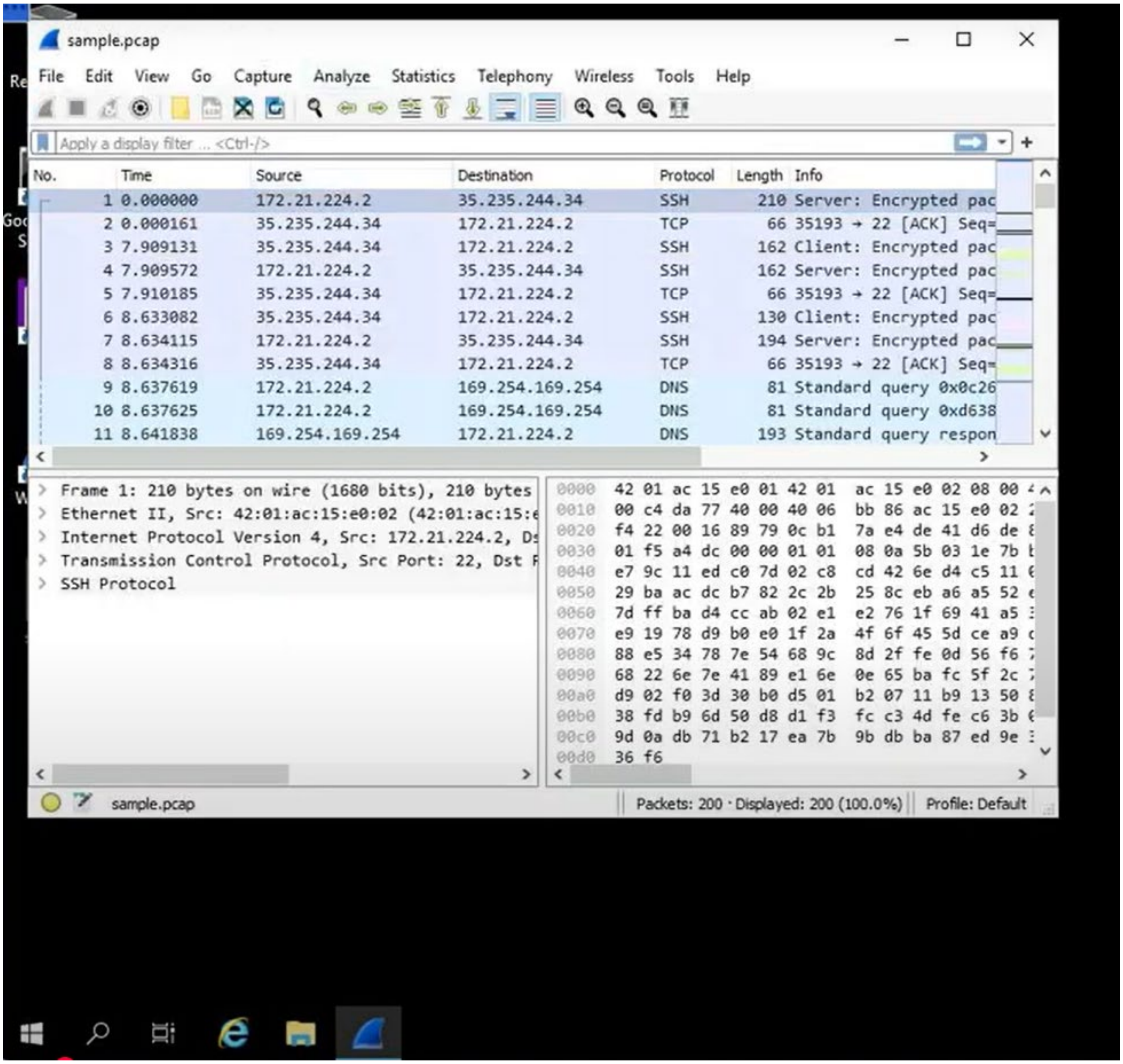
1. Open the packet capture file and explore the Wireshark graphical user interface.
2. Open a detailed view of a single packet to examine the various protocols and data layers inside a network packet.
3. Apply filters to select and inspect packets based on specific criteria.
4. Filter and inspect UDP DNS traffic to examine protocol data.
5. Apply filters to TCP packet data to search for specific payload text data.

Task 1. Explore data with Wireshark

In this task, you must open a network packet capture file that contains data captured from a system that made web requests to a site. You need to open this data with Wireshark to get an overview of how the data is presented in the application.

1. To open the packet capture file, double-click the **sample** file on the Windows desktop. This will start Wireshark.


Recycle Bin
Google Cloud S...
NVDA
Wireshark
sample


sample.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time      Source           Destination      Protocol  Length Info
   1 0.000000  172.21.224.2     35.235.244.34    SSH          210 Server: Encrypted pac
   2 0.000161  35.235.244.34    172.21.224.2     TCP           66 35193 → 22 [ACK] Seq=
   3 7.909131  35.235.244.34    172.21.224.2     SSH          162 Client: Encrypted pac
   4 7.909572  172.21.224.2     35.235.244.34    SSH          162 Server: Encrypted pac
   5 7.910185  35.235.244.34    172.21.224.2     TCP           66 35193 → 22 [ACK] Seq=
   6 8.633082  35.235.244.34    172.21.224.2     SSH          130 Client: Encrypted pac
   7 8.634115  172.21.224.2     35.235.244.34    SSH          194 Server: Encrypted pac
   8 8.634316  35.235.244.34    172.21.224.2     TCP           66 35193 → 22 [ACK] Seq=
   9 8.637619  172.21.224.2     169.254.169.254  DNS           81 Standard query 0x0c26
  10 8.637625  172.21.224.2     169.254.169.254  DNS           81 Standard query 0xd638
  11 8.641838  169.254.169.254  172.21.224.2     DNS          193 Standard query respon
```

```
> Frame 1: 210 bytes on wire (1680 bits), 210 bytes
> Ethernet II, Src: 42:01:ac:15:e0:02 (42:01:ac:15:e
> Internet Protocol Version 4, Src: 172.21.224.2, Ds
> Transmission Control Protocol, Src Port: 22, Dst P
> SSH Protocol
```

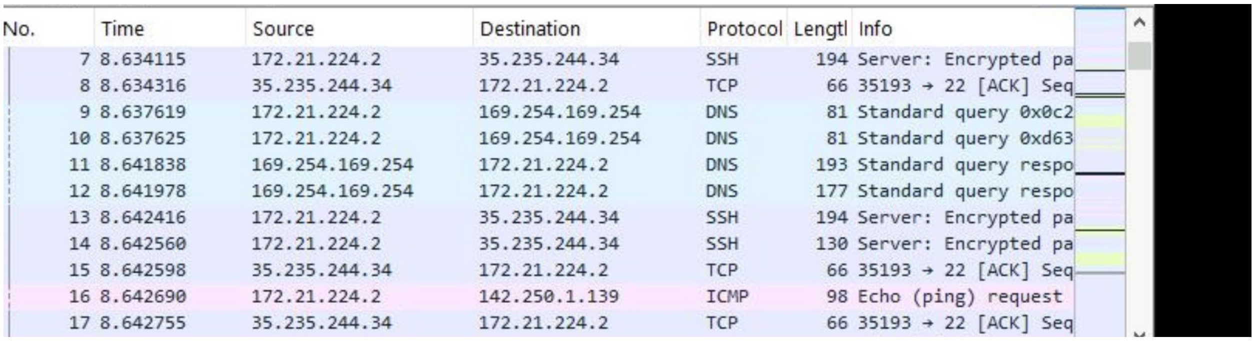
```
0000  42 01 ac 15 e0 01 42 01  ac 15 e0 02 08 00
0010  00 c4 da 77 40 00 40 06  bb 86 ac 15 e0 02
0020  f4 22 00 16 89 79 0c b1  7a e4 de 41 d6 de
0030  01 f5 a4 dc 00 00 01 01  08 0a 5b 03 1e 7b
0040  e7 9c 11 ed c0 7d 02 c8  cd 42 6e d4 c5 11
0050  29 ba ac dc b7 82 2c 2b  25 8c eb a6 a5 52
0060  7d ff ba d4 cc ab 02 e1  e2 76 1f 69 41 a5
0070  e9 19 78 d9 b0 e0 1f 2a  4f 6f 45 5d ce a9
0080  88 e5 34 78 7e 54 68 9c  8d 2f fe 0d 56 f6
0090  68 22 6e 7e 41 89 e1 6e  0e 65 ba fc 5f 2c
00a0  d9 02 f0 3d 30 b0 d5 01  b2 07 11 b9 13 50
00b0  38 fd b9 6d 50 d8 d1 f3  fc c3 4d fe c6 3b
00c0  9d 0a db 71 b2 17 ea 7b  9b db ba 87 ed 9e
00d0  36 f6
```

sample.pcap | Packets: 200 · Displayed: 200 (100.0%) | Profile: Default

The packet capture file icon is a Wireshark shark's fin swimming above three rows of binary digits. The file has a **.pcap** extension that is hidden by default in Windows Explorer and on the desktop view.

2. Double-click the Wireshark title bar next to the **sample.pcap** filename to maximize the Wireshark application window.

   A lot of network packet traffic is listed, which is why you'll apply filters to find the information needed in an upcoming step.



No.	Time	Source	Destination	Protocol	Lengtl	Info
7	8.634115	172.21.224.2	35.235.244.34	SSH	194	Server: Encrypted pa
8	8.634316	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq
9	8.637619	172.21.224.2	169.254.169.254	DNS	81	Standard query 0x0c2
10	8.637625	172.21.224.2	169.254.169.254	DNS	81	Standard query 0xd63
11	8.641838	169.254.169.254	172.21.224.2	DNS	193	Standard query respo
12	8.641978	169.254.169.254	172.21.224.2	DNS	177	Standard query respo
13	8.642416	172.21.224.2	35.235.244.34	SSH	194	Server: Encrypted pa
14	8.642560	172.21.224.2	35.235.244.34	SSH	130	Server: Encrypted pa
15	8.642598	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq
16	8.642690	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request
17	8.642755	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq

For now, here is an overview of the key property columns listed for each packet:

- **No.** : The index number of the packet in this packet capture file
- **Time**: The timestamp of the packet
- **Source**: The source IP address
- **Destination**: The destination IP address
- **Protocol**: The protocol contained in the packet
- **Length**: The total length of the packet
- **Info**: Some information about the data in the packet (the payload) as interpreted by Wireshark

Not all the data packets are the same color. Coloring rules are used to provide high-level visual cues to help you quickly classify the different types of data. Since network packet capture files can contain large amounts of data, you can use coloring rules to identify the data that is relevant to you quickly. The example packet lists a group of light blue packets that all contain DNS traffic, followed by green packets that contain a mixture of TCP and HTTP protocol traffic.

3. Scroll down the packet list until you find a packet whose 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'?

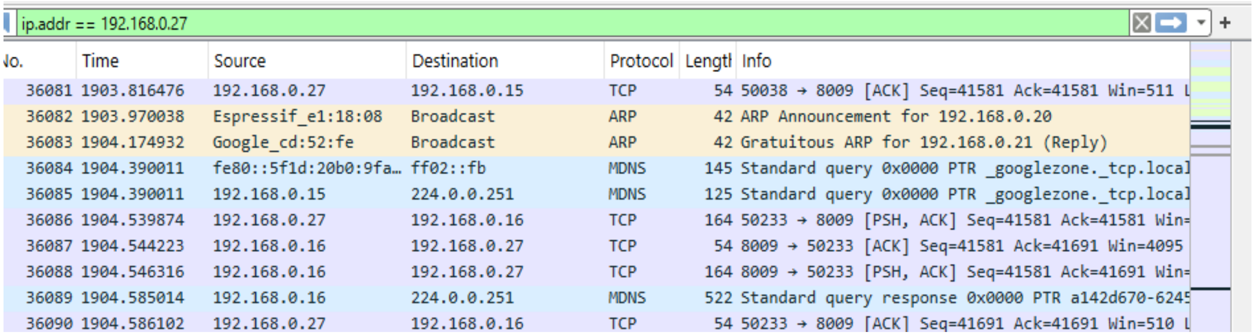
- ☐TCP
- ☐HTTP
- ☐SSH
- ☑ICMP

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.

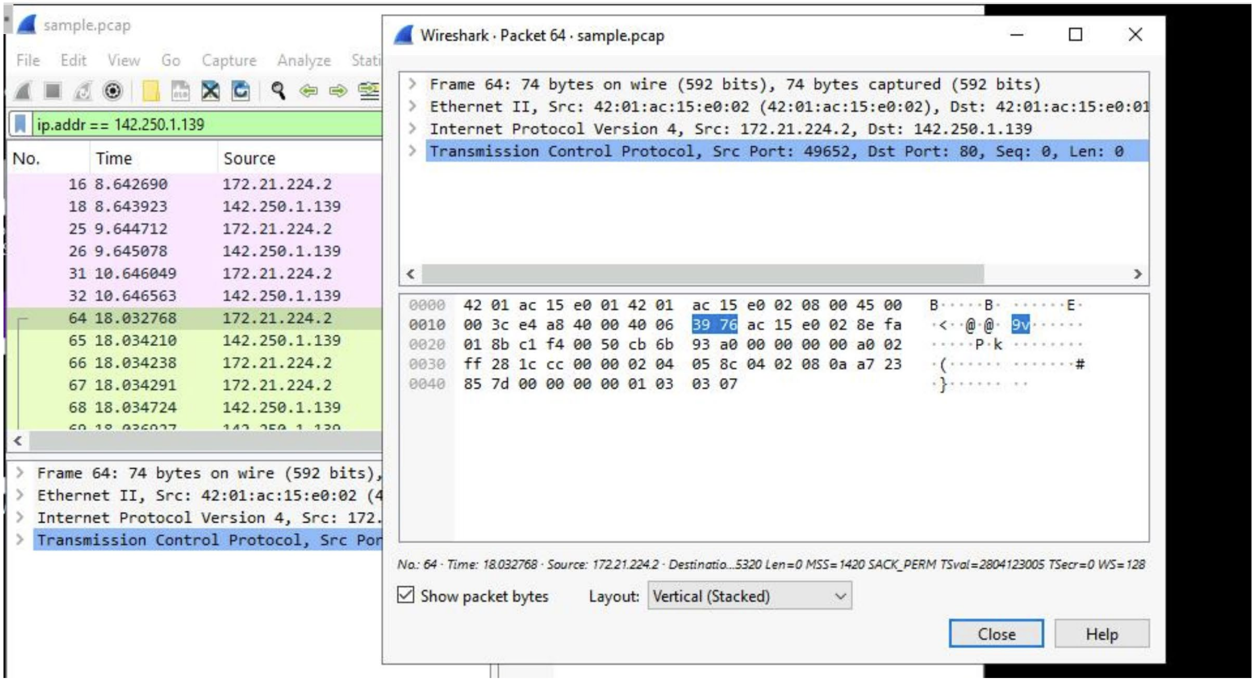
1. Enter the following filter for traffic associated with a specific IP address. Enter this into the **Apply a display filter...** text box immediately above the list of packets:

ip.addr == 192.168.0.27



```
ip.addr == 192.168.0.27
No.   Time        Source                 Destination   Protocol Lengtl Info
36081 1903.816476 192.168.0.27           192.168.0.15  TCP        54 50038 → 8009 [ACK] Seq=41581 Ack=41581 Win=511 L
36082 1903.970038 Espressif_e1:18:08     Broadcast     ARP        42 ARP Announcement for 192.168.0.20
36083 1904.174932 Google_cd:52:fe        Broadcast     ARP        42 Gratuitous ARP for 192.168.0.21 (Reply)
36084 1904.390011 fe80::5f1d:20b0:9fa…   ff02::fb      MDNS      145 Standard query 0x0000 PTR _googlezone._tcp.local
36085 1904.390011 192.168.0.15           224.0.0.251   MDNS      125 Standard query 0x0000 PTR _googlezone._tcp.local
36086 1904.539874 192.168.0.27           192.168.0.16  TCP       164 50233 → 8009 [PSH, ACK] Seq=41581 Ack=41581 Win=
36087 1904.544223 192.168.0.16           192.168.0.27  TCP        54 8009 → 50233 [ACK] Seq=41581 Ack=41691 Win=4095
36088 1904.546316 192.168.0.16           192.168.0.27  TCP       164 8009 → 50233 [PSH, ACK] Seq=41581 Ack=41691 Win=
36089 1904.585014 192.168.0.16           224.0.0.251   MDNS      522 Standard query response 0x0000 PTR a142d670-6245
36090 1904.586102 192.168.0.27           192.168.0.16  TCP        54 50233 → 8009 [ACK] Seq=41691 Ack=41691 Win=510 L
```

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



```
ip.addr == 142.250.1.139
No.   Time       Source
   16 8.642690   172.21.224.2
   18 8.643923   142.250.1.139
   25 9.644712   172.21.224.2
   26 9.645078   142.250.1.139
   31 10.646049  172.21.224.2
   32 10.646563  142.250.1.139
   64 18.032768  172.21.224.2
   65 18.034210  142.250.1.139
   66 18.034238  172.21.224.2
   67 18.034291  172.21.224.2
   68 18.034724  142.250.1.139

> Frame 64: 74 bytes on wire (592 bits),
> Ethernet II, Src: 42:01:ac:15:e0:02 (4
> Internet Protocol Version 4, Src: 172.
> Transmission Control Protocol, Src Por
```

Wireshark · Packet 64 · sample.pcap

```
> 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
> 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

0000  42 01 ac 15 e0 01 42 01  ac 15 e0 02 08 00 45 00   B·····B· ······E·
0010  00 3c e4 a8 40 00 40 06  39 76 ac 15 e0 02 8e fa   ·<··@·@· 9v······
0020  01 8b c1 f4 00 50 cb 6b  93 a0 00 00 00 00 a0 02   ·····P·k ········
0030  ff 28 1c cc 00 00 02 04  05 8c 04 02 08 0a a7 23   ·(······ ·······#
0040  85 7d 00 00 00 00 01 03  03 07                     ·}······ ··

No.: 64 · Time: 18.032768 · Source: 172.21.224.2 · Destinatio...5320 Len=0 MSS=1420 SACK_PERM TSval=2804123005 TSecr=0 WS=128
```

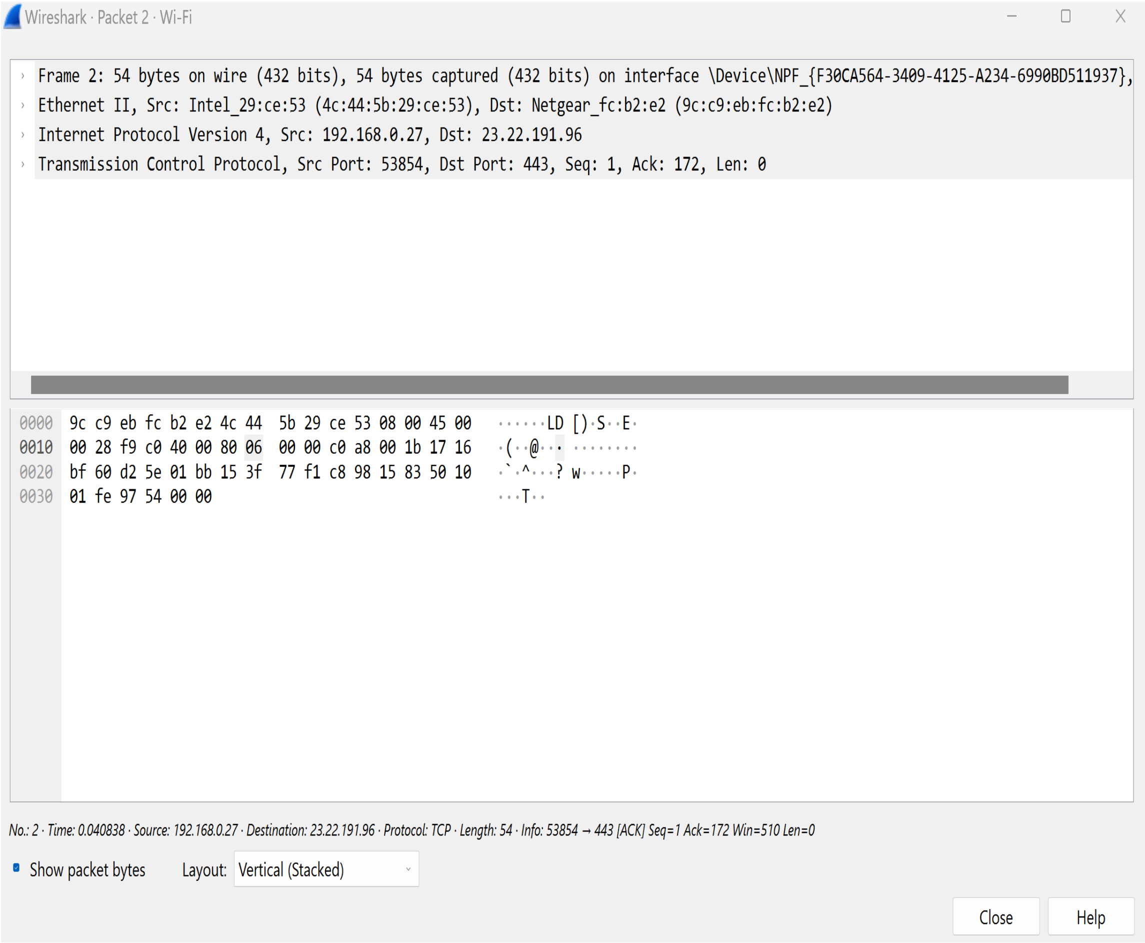
☑ Show packet bytes   Layout: Vertical (Stacked)

3. Notes: The list of displayed packets has been significantly reduced and now contains only packets where either the source or the destination IP address matches the address you entered. Wireshark uses only two packet colors: light pink for ICMP protocol packets and light green for TCP (and HTTP, which is a subset of TCP) packets.Double-click the first packet that lists **TCP** as the protocol.

This opens a packet details pane window:

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**.



```
› Frame 2: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{F30CA564-3409-4125-A234-6990BD511937},
› Ethernet II, Src: Intel_29:ce:53 (4c:44:5b:29:ce:53), Dst: Netgear_fc:b2:e2 (9c:c9:eb:fc:b2:e2)
› Internet Protocol Version 4, Src: 192.168.0.27, Dst: 23.22.191.96
› Transmission Control Protocol, Src Port: 53854, Dst Port: 443, Seq: 1, Ack: 172, Len: 0
```

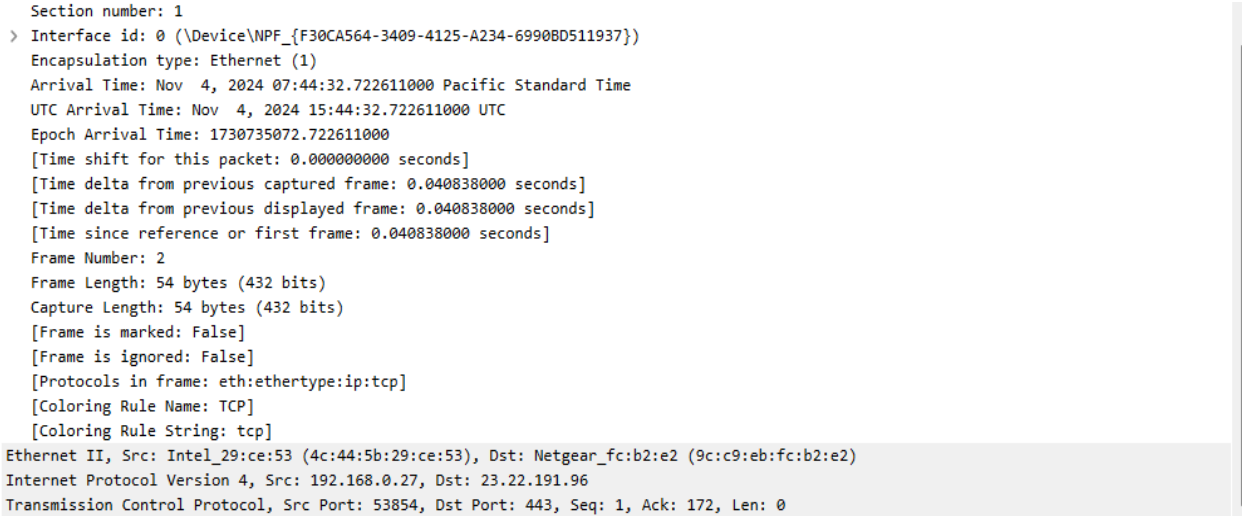
```
0000  9c c9 eb fc b2 e2 4c 44  5b 29 ce 53 08 00 45 00   ······LD [)·S··E·
0010  00 28 f9 c0 40 00 80 06  00 00 c0 a8 00 1b 17 16   ·(··@··· ········
0020  bf 60 d2 5e 01 bb 15 3f  77 f1 c8 98 15 83 50 10   ·`·^···? w·····P·
0030  01 fe 97 54 00 00                                  ···T··
```

No.: 2 · Time: 0.040838 · Source: 192.168.0.27 · Destination: 23.22.191.96 · Protocol: TCP · Length: 54 · Info: 53854 → 443 [ACK] Seq=1 Ack=172 Win=510 Len=0

☑ Show packet bytes   Layout: Vertical (Stacked)

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

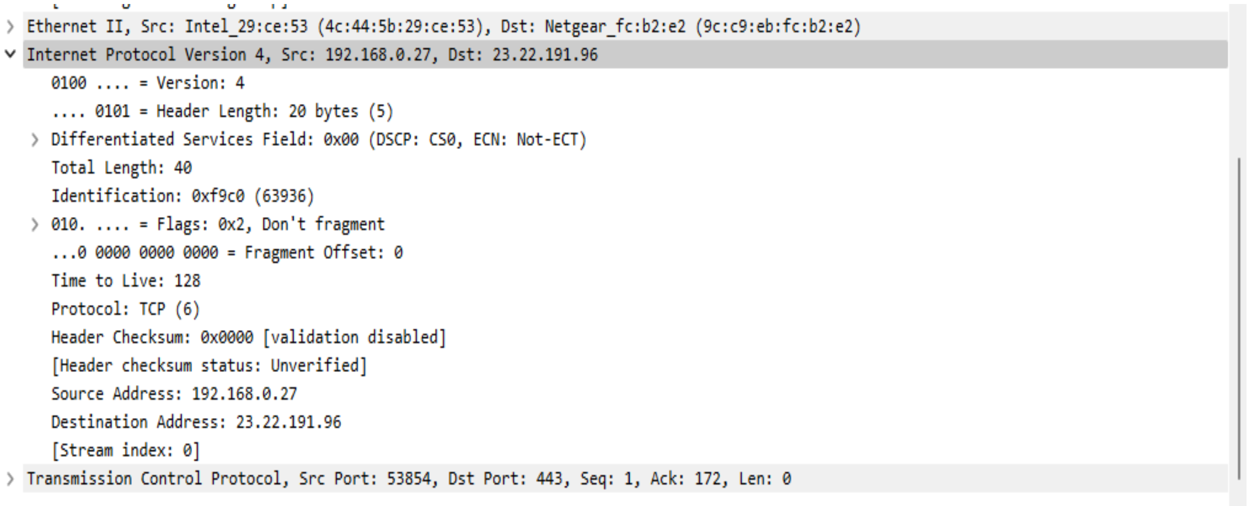
5. Double-click **Frame** again to collapse the subtree and then double-click the **Ethernet II** subtree.



```
    Section number: 1
  > Interface id: 0 (\Device\NPF_{F30CA564-3409-4125-A234-6990BD511937})
    Encapsulation type: Ethernet (1)
    Arrival Time: Nov  4, 2024 07:44:32.722611000 Pacific Standard Time
    UTC Arrival Time: Nov  4, 2024 15:44:32.722611000 UTC
    Epoch Arrival Time: 1730735072.722611000
    [Time shift for this packet: 0.000000000 seconds]
    [Time delta from previous captured frame: 0.040838000 seconds]
    [Time delta from previous displayed frame: 0.040838000 seconds]
    [Time since reference or first frame: 0.040838000 seconds]
    Frame Number: 2
    Frame Length: 54 bytes (432 bits)
    Capture Length: 54 bytes (432 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ip:tcp]
    [Coloring Rule Name: TCP]
    [Coloring Rule String: tcp]
Ethernet II, Src: Intel_29:ce:53 (4c:44:5b:29:ce:53), Dst: Netgear_fc:b2:e2 (9c:c9:eb:fc:b2:e2)
Internet Protocol Version 4, Src: 192.168.0.27, Dst: 23.22.191.96
Transmission Control Protocol, Src Port: 53854, Dst Port: 443, Seq: 1, Ack: 172, Len: 0
```

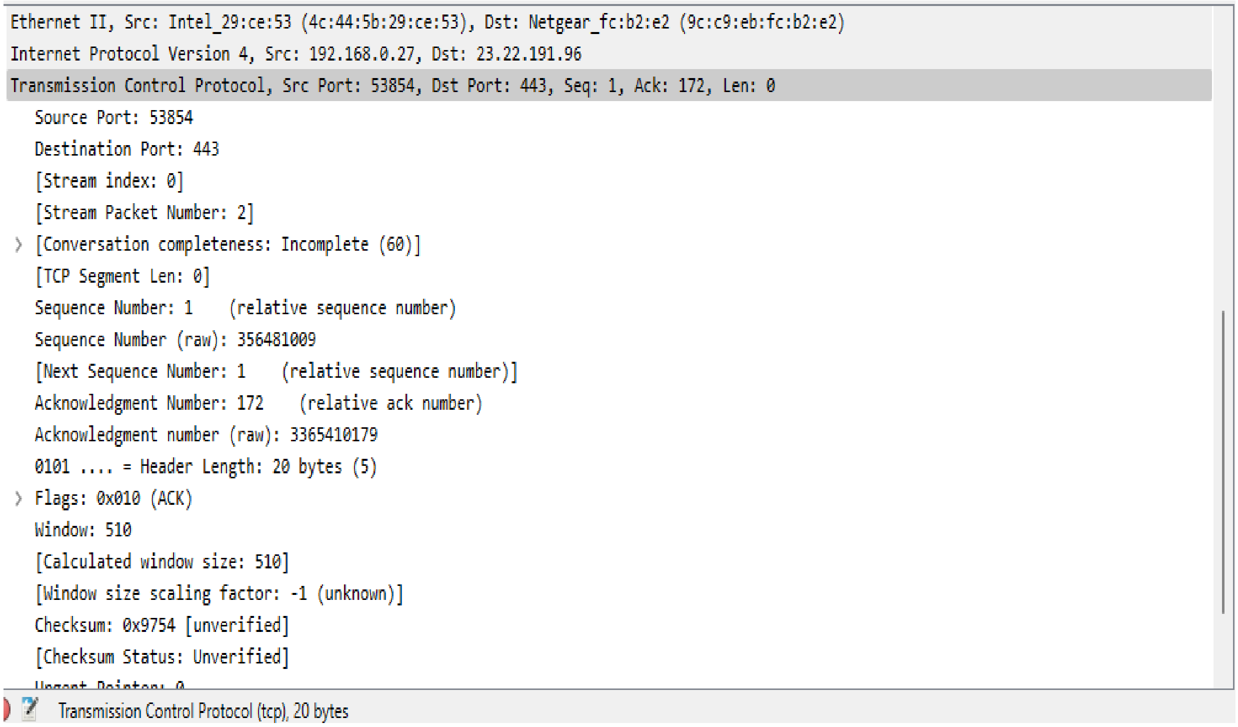
This item contains details about the Ethernet packet, including the source and destination MAC addresses and the type of internal protocol it contains.

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



```
> Ethernet II, Src: Intel_29:ce:53 (4c:44:5b:29:ce:53), Dst: Netgear_fc:b2:e2 (9c:c9:eb:fc:b2:e2)
v Internet Protocol Version 4, Src: 192.168.0.27, Dst: 23.22.191.96
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 40
    Identification: 0xf9c0 (63936)
  > 010. .... = Flags: 0x2, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: TCP (6)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.0.27
    Destination Address: 23.22.191.96
    [Stream index: 0]
> Transmission Control Protocol, Src Port: 53854, Dst Port: 443, Seq: 1, Ack: 172, Len: 0
```

Double-clicking Ethernet II and Internet Protocol Version 4 provides packet data about the Internet Protocol (IP) data contained in the Ethernet packet. This data includes information such as the source and destination IP addresses, and the Internal Protocol (for example, TCP or UDP) carried inside the IP packet.Double-click **Internet Protocol Version 4** again to collapse that subtree and then double-click the **Transmission Control Protocol** subtree.



```
Ethernet II, Src: Intel_29:ce:53 (4c:44:5b:29:ce:53), Dst: Netgear_fc:b2:e2 (9c:c9:eb:fc:b2:e2)
Internet Protocol Version 4, Src: 192.168.0.27, Dst: 23.22.191.96
Transmission Control Protocol, Src Port: 53854, Dst Port: 443, Seq: 1, Ack: 172, Len: 0
    Source Port: 53854
    Destination Port: 443
    [Stream index: 0]
    [Stream Packet Number: 2]
  › [Conversation completeness: Incomplete (60)]
    [TCP Segment Len: 0]
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 356481009
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 172    (relative ack number)
    Acknowledgment number (raw): 3365410179
    0101 .... = Header Length: 20 bytes (5)
  › Flags: 0x010 (ACK)
    Window: 510
    [Calculated window size: 510]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0x9754 [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
Transmission Control Protocol (tcp), 20 bytes
```

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

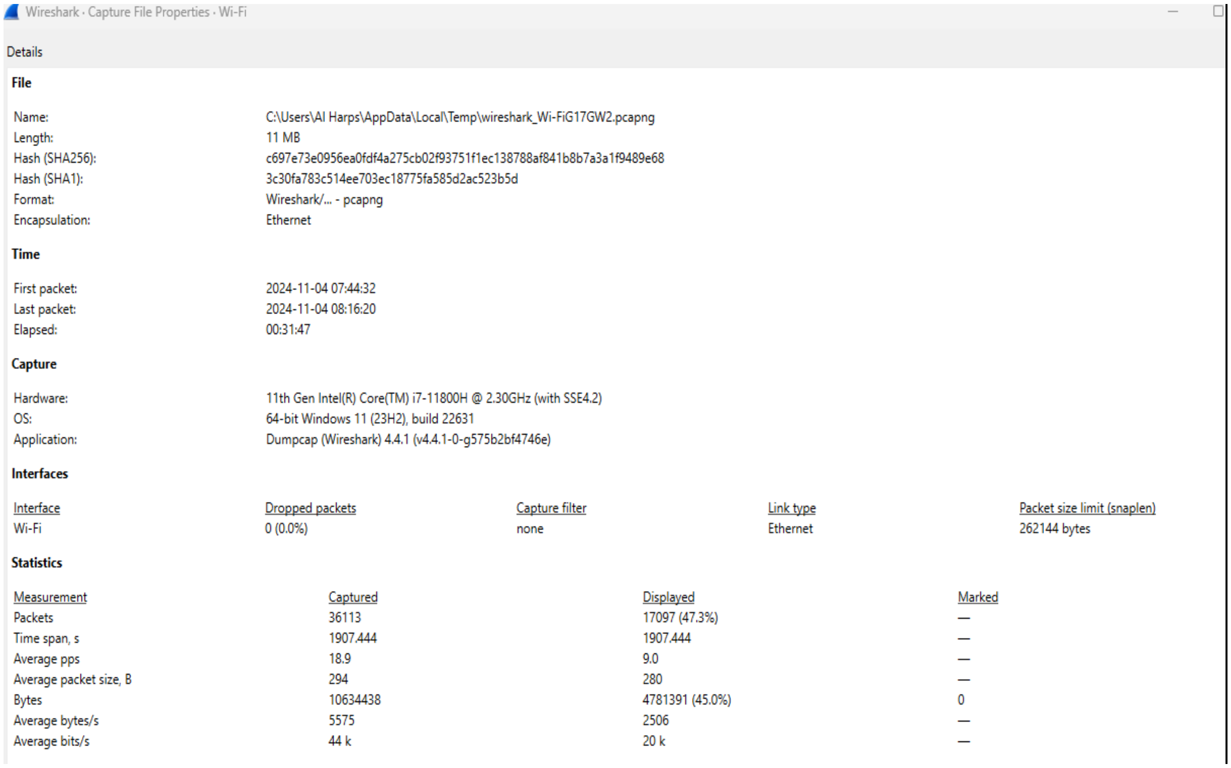
The source port and destination port listed here match the source and destination ports in the info column of the summary display for this packet in the list of all of the packets in the main Wireshark window.

What is the TCP destination port of this TCP packet?

- ☐53
- ☐200
- ☑443
- ☐66

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.



Wireshark · Capture File Properties · Wi-Fi

Details

File

Name:	C:\Users\Al Harps\AppData\Local\Temp\wireshark_Wi-FiG17GW2.pcapng
Length:	11 MB
Hash (SHA256):	c697e73e0956ea0fdf4a275cb02f93751f1ec138788af841b8b7a3a1f9489e68
Hash (SHA1):	3c30fa783c514ee703ec18775fa585d2ac523b5d
Format:	Wireshark/... - pcapng
Encapsulation:	Ethernet

Time

First packet:	2024-11-04 07:44:32
Last packet:	2024-11-04 08:16:20
Elapsed:	00:31:47

Capture

Hardware:	11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz (with SSE4.2)
OS:	64-bit Windows 11 (23H2), build 22631
Application:	Dumpcap (Wireshark) 4.4.1 (v4.4.1-0-g575b2bf4746e)

Interfaces

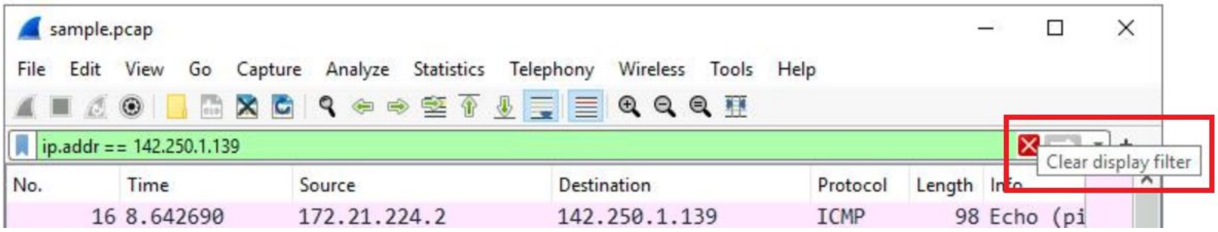
Interface	Dropped packets	Capture filter	Link type	Packet size limit (snaplen)
Wi-Fi	0 (0.0%)	none	Ethernet	262144 bytes

Statistics

Measurement	Captured	Displayed	Marked
Packets	36113	17097 (47.3%)	—
Time span, s	1907.444	1907.444	—
Average pps	18.9	9.0	—
Average packet size, B	294	280	—
Bytes	10634438	4781391 (45.0%)	0
Average bytes/s	5575	2506	—
Average bits/s	44 k	20 k	—

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.



ip.addr == 142.250.1.139

```
No.     Time        Source          Destination      Protocol  Length  Info
     16 8.642690    172.21.224.2    142.250.1.139    ICMP          98  Echo (pi
```

All the packets have returned to the display.

If you ever accidentally close the Wireshark application, you can reopen it by double-clicking the **sample** file on the desktop.

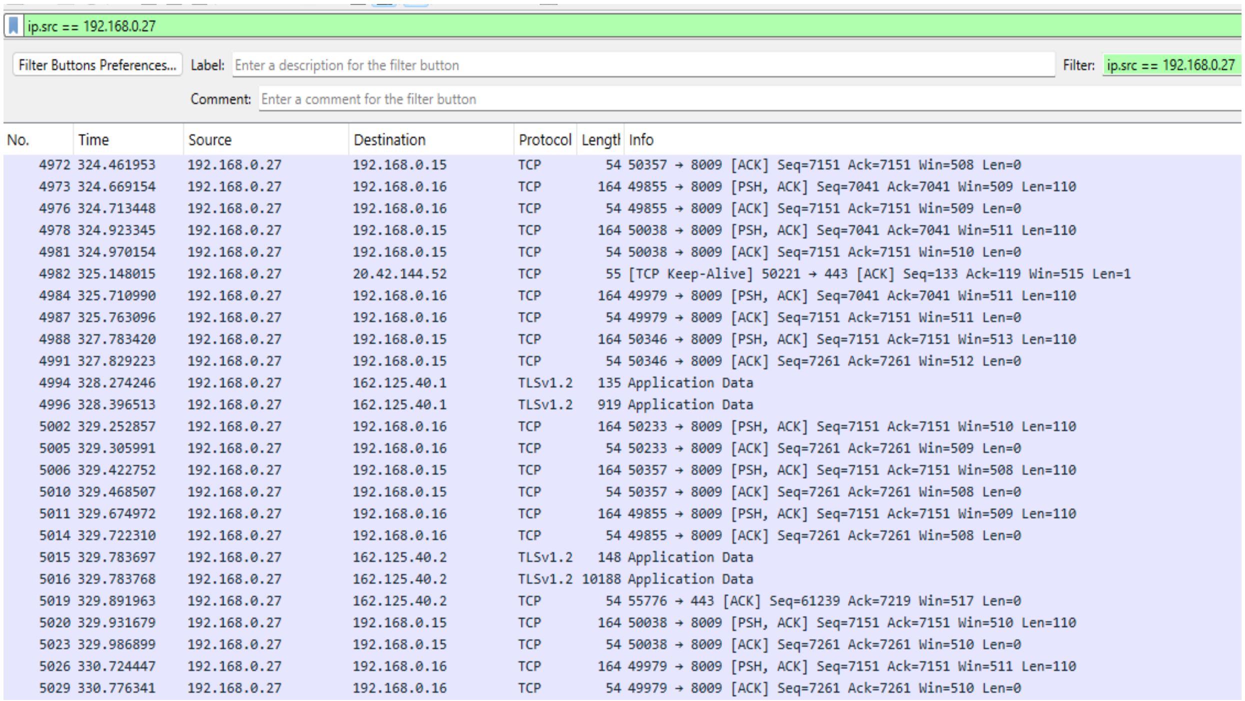
Task 3. Use filters to select packets

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 == 192.168.0.27

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 **192.168.0.27**.



ip.src == 192.168.0.27

Label: Enter a description for the filter button

Filter: ip.src == 192.168.0.27

Comment: Enter a comment for the filter button

```
No.   Time        Source        Destination    Protocol Length Info
 4972 324.461953  192.168.0.27  192.168.0.15   TCP         54 50357 → 8009 [ACK] Seq=7151 Ack=7151 Win=508 Len=0
 4973 324.669154  192.168.0.27  192.168.0.16   TCP        164 49855 → 8009 [PSH, ACK] Seq=7041 Ack=7041 Win=509 Len=110
 4976 324.713448  192.168.0.27  192.168.0.16   TCP         54 49855 → 8009 [ACK] Seq=7151 Ack=7151 Win=509 Len=0
 4978 324.923345  192.168.0.27  192.168.0.15   TCP        164 50038 → 8009 [PSH, ACK] Seq=7041 Ack=7041 Win=511 Len=110
 4981 324.970154  192.168.0.27  192.168.0.15   TCP         54 50038 → 8009 [ACK] Seq=7151 Ack=7151 Win=510 Len=0
 4982 325.148015  192.168.0.27  20.42.144.52   TCP         55 [TCP Keep-Alive] 50221 → 443 [ACK] Seq=133 Ack=119 Win=515 Len=1
 4984 325.710990  192.168.0.27  192.168.0.16   TCP        164 49979 → 8009 [PSH, ACK] Seq=7041 Ack=7041 Win=511 Len=110
 4987 325.763096  192.168.0.27  192.168.0.16   TCP         54 49979 → 8009 [ACK] Seq=7151 Ack=7151 Win=511 Len=0
 4988 327.783420  192.168.0.27  192.168.0.15   TCP        164 50346 → 8009 [PSH, ACK] Seq=7151 Ack=7151 Win=513 Len=110
 4991 327.829223  192.168.0.27  192.168.0.15   TCP         54 50346 → 8009 [ACK] Seq=7261 Ack=7261 Win=512 Len=0
 4994 328.274246  192.168.0.27  162.125.40.1   TLSv1.2    135 Application Data
 4996 328.396513  192.168.0.27  162.125.40.1   TLSv1.2    919 Application Data
 5002 329.252857  192.168.0.27  192.168.0.16   TCP        164 50233 → 8009 [PSH, ACK] Seq=7151 Ack=7151 Win=510 Len=110
 5005 329.305991  192.168.0.27  192.168.0.16   TCP         54 50233 → 8009 [ACK] Seq=7261 Ack=7261 Win=509 Len=0
 5006 329.422752  192.168.0.27  192.168.0.15   TCP        164 50357 → 8009 [PSH, ACK] Seq=7151 Ack=7151 Win=508 Len=110
 5010 329.468507  192.168.0.27  192.168.0.15   TCP         54 50357 → 8009 [ACK] Seq=7261 Ack=7261 Win=508 Len=0
 5011 329.674972  192.168.0.27  192.168.0.16   TCP        164 49855 → 8009 [PSH, ACK] Seq=7151 Ack=7151 Win=509 Len=110
 5014 329.722310  192.168.0.27  192.168.0.16   TCP         54 49855 → 8009 [ACK] Seq=7261 Ack=7261 Win=508 Len=0
 5015 329.783697  192.168.0.27  162.125.40.2   TLSv1.2    148 Application Data
 5016 329.783768  192.168.0.27  162.125.40.2   TLSv1.2  10188 Application Data
 5019 329.891963  192.168.0.27  162.125.40.2   TCP         54 55776 → 443 [ACK] Seq=61239 Ack=7219 Win=517 Len=0
 5020 329.931679  192.168.0.27  192.168.0.15   TCP        164 50038 → 8009 [PSH, ACK] Seq=7151 Ack=7151 Win=510 Len=110
 5023 329.986899  192.168.0.27  192.168.0.15   TCP         54 50038 → 8009 [ACK] Seq=7261 Ack=7261 Win=510 Len=0
 5026 330.724447  192.168.0.27  192.168.0.16   TCP        164 49979 → 8009 [PSH, ACK] Seq=7151 Ack=7151 Win=511 Len=110
 5029 330.776341  192.168.0.27  192.168.0.16   TCP         54 49979 → 8009 [ACK] Seq=7261 Ack=7261 Win=510 Len=0
```

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 == 192.168.0.27

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 **192.168.0.27**.

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

- ☑TCP
- ☐ICMP
- ☐ESP
- ☐UDP

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

14. 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**) and the addresses that are associated with that name.

Which of these IP addresses is displayed in the expanded Answers section for the DNS query for opensource.google.com?

- □169.254.169.254
- ☑192.168.0.27
- □172.21.224.1
- □139.1.250.142

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
- □16
- □32
- □128

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

- □74 bytes
- ☑54 bytes
- □60 bytes
- □40 bytes

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

- ☑20 bytes
- □60 bytes
- □74 bytes

- □54 bytes

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

- □172.21.224.2
- ☑169.254.169.254
- □239.1.250.142
- □142.250.1.139

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.

Conclusion: Great work! You now have practical experience using Wireshark to:

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

This is an important milestone on your journey toward understanding how to use network packet analysis tools to examine network traffic!

End of Lab: Before you end the lab, make sure you’re satisfied that you’ve completed all the tasks, and follow these steps:

1. Click End Lab. A pop-up box will appear. Click Submit to confirm that you're done. Ending the lab will remove your access to the Bash shell. You won’t be able to access the work you've completed in it again.
2. Another pop-up box will ask you to rate the lab and provide feedback comments. You can complete this if you choose to.
3. Close the browser tab containing the lab to return to your course.
4. Refresh the browser tab for the course to mark the lab as complete.