

Activity: Analyze your first packet with Wireshark

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:

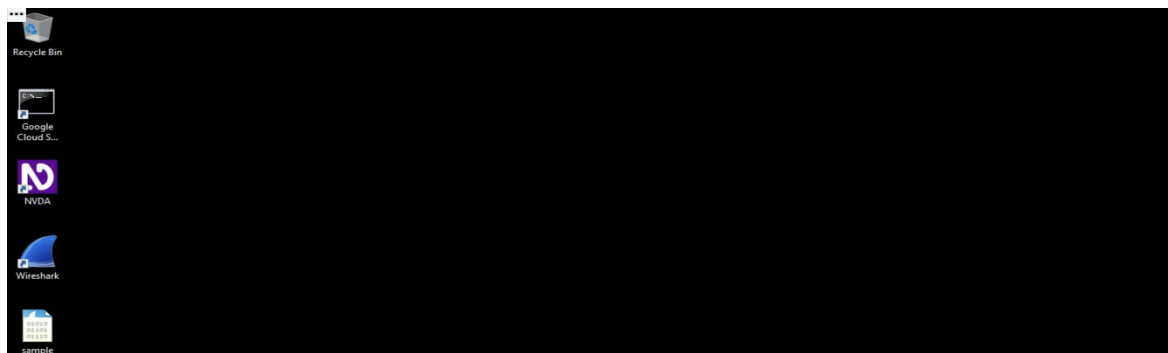
- identify the source and destination IP addresses involved in this web browsing session,
- examine the protocols that are used when the user makes the connection to the website, and
- analyze some of the data packets to identify the type of information sent and received by the systems that connect to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

Here's how you'll do this: **First**, you'll open the packet capture file and explore the basic Wireshark graphic user interface. **Second**, you'll open a detailed view of a single packet and explore how to examine the various protocol and data layers inside a network packet. **Third**, you'll apply filters to select and inspect packets based on specific criteria. **Fourth**, you'll filter and inspect UDP DNS traffic to examine protocol data. **Finally**, you'll apply filters to TCP packet data to search for specific payload text data.

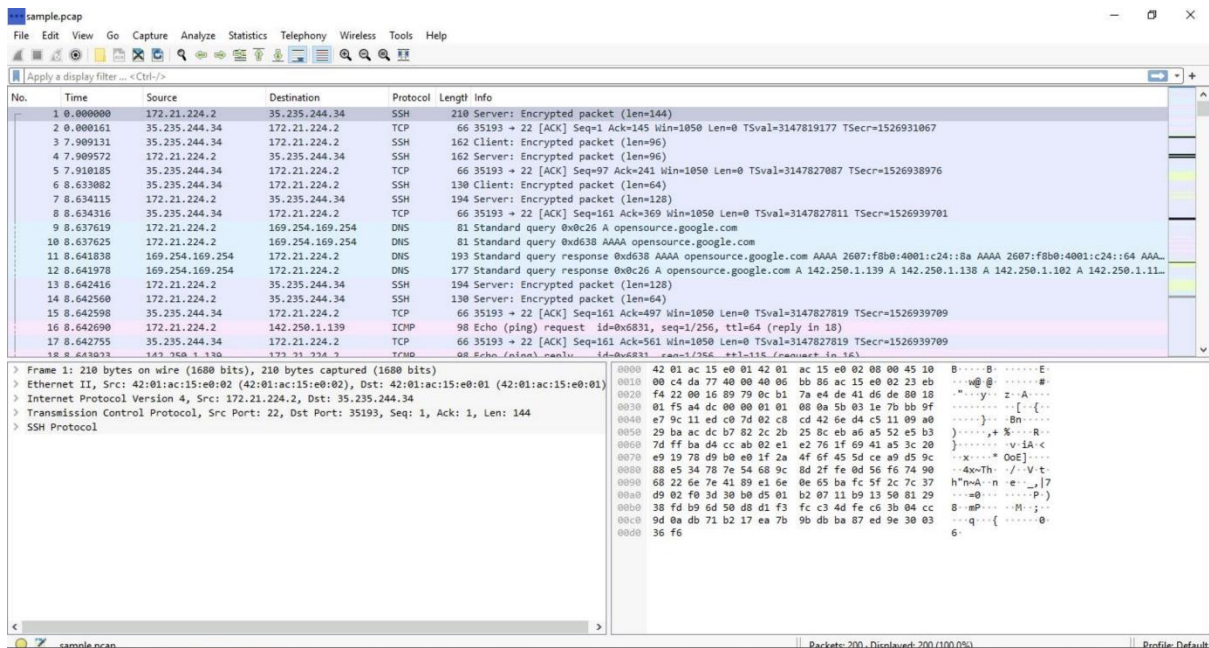
Task 1. Explore data with Wireshark

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.



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.

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

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

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=1/256, ttl=64 (reply in 18)
18	8.643923	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=1/256, ttl=115 (request in 16)
25	9.644712	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=2/512, ttl=64 (reply in 26)
26	9.645078	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=2/512, ttl=115 (request in 25)
31	10.646049	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=3/768, ttl=64 (reply in 32)
32	10.646563	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=3/768, ttl=115 (request in 31)
64	18.032768	172.21.224.2	142.250.1.139	TCP	74	49652 → 80 [SYN] Seq=0 Win=65320 Len=0 MSS=1420 SACK_PERM TSval=2804123005 TSecr=0 WS=128
65	18.034210	142.250.1.139	172.21.224.2	TCP	74	80 → 49652 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1420 SACK_PERM TSval=4069674930 TSecr=2804123005 WS=256
66	18.034238	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=1 Ack=1 Win=65408 Len=0 TSval=2804123006 TSecr=4069674930
67	18.034291	172.21.224.2	142.250.1.139	HTTP	151	GET / HTTP/1.1
68	18.034724	142.250.1.139	172.21.224.2	TCP	66	80 → 49652 [ACK] Seq=1 Ack=86 Win=65536 Len=0 TSval=4069674931 TSecr=2804123006
69	18.036927	142.250.1.139	172.21.224.2	HTTP	640	HTTP/1.1 301 Moved Permanently (text/html)
70	18.036941	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=86 Ack=583 Win=64896 Len=0 TSval=2804123009 TSecr=4069674934
79	18.037390	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [FIN, ACK] Seq=86 Ack=583 Win=64896 Len=0 TSval=2804123009 TSecr=4069674934
82	18.037927	142.250.1.139	172.21.224.2	TCP	66	80 → 49652 [FIN, ACK] Seq=583 Ack=87 Win=65536 Len=0 TSval=4069674935 TSecr=2804123009
83	18.037936	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=87 Ack=584 Win=64896 Len=0 TSval=2804123010 TSecr=4069674935

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

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 (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 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 01 03 03 07 }-----

No: 64 · Time: 18.032768 · Source: 172.21.224.2 · Destination: 142.250.1.139 · Protocol: TCP · Length: 74 · Info: 49652 → 80 [SYN] Seq=0 Win=65320 Len=0 MSS=1420 SACK_PERM TSval=2804123005 TSecr=0 WS=128

☒ Show packet bytes

Close Help

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.

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.

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.

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.

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.

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.

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**.

No.	Time	Source	Destination	Protocol	Length	Info
18	8.643923	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=1/256, ttl=115 (request in 16)
26	9.645078	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=2/512, ttl=115 (request in 25)
32	10.646563	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=3/768, ttl=115 (request in 31)
65	18.034210	142.250.1.139	172.21.224.2	TCP	74	80 → 49652 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1420 SACK_PERM TSval=4069674930 TSecr=2804123005 WS=256
68	18.034724	142.250.1.139	172.21.224.2	TCP	66	80 → 49652 [ACK] Seq=1 Ack=86 Win=65536 Len=0 TSval=4069674931 TSecr=2804123006
69	18.036927	142.250.1.139	172.21.224.2	HTTP	648	HTTP/1.1 301 Moved Permanently (text/html)
82	18.037927	142.250.1.139	172.21.224.2	TCP	66	80 → 49652 [FIN, ACK] Seq=583 Ack=87 Win=65536 Len=0 TSval=4069674935 TSecr=2804123009

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**.

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=1/256, ttl=64 (reply in 18)
25	9.644712	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=2/512, ttl=64 (reply in 26)
31	10.646049	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=3/768, ttl=64 (reply in 32)
64	18.032768	172.21.224.2	142.250.1.139	TCP	74	49652 → 80 [SYN] Seq=0 Win=65520 Len=0 MSS=1420 SACK_PERM TSval=2804123005 TSecr=0 WS=128
66	18.034238	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=1 Ack=1 Win=65488 Len=0 TSval=2804123006 TSecr=4069674930
67	18.034291	172.21.224.2	142.250.1.139	HTTP	151	GET / HTTP/1.1
70	18.036941	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=86 Ack=583 Win=64896 Len=0 TSval=2804123009 TSecr=4069674934
79	18.037398	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [FIN, ACK] Seq=86 Ack=583 Win=64896 Len=0 TSval=2804123009 TSecr=4069674934
83	18.037936	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=87 Ack=584 Win=64896 Len=0 TSval=2804123010 TSecr=4069674935

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.

No.	Time	Source	Destination	Protocol	Length	Info
23	8.647688	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq=161 Ack=705 Win=1050 Len=0 TSval=3147827824 TSecr=1526939714
24	8.647682	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq=161 Ack=709 Win=1050 Len=0 TSval=3147827824 TSecr=1526939714
25	9.644712	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=2/512, ttl=64 (reply in 26)
26	9.645078	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=2/512, ttl=115 (request in 25)
27	9.645214	172.21.224.2	169.254.169.254	DNS	86	Standard query 0x3cdc PTR 139.1.250.142.in-addr.arpa
28	9.645859	169.254.169.254	172.21.224.2	DNS	120	Standard query response 0x3cdc PTR 139.1.250.142.in-addr.arpa PTR jw-in-f139.1e100.net
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=1050 Len=0 TSval=3147828823 TSecr=1526940713
31	10.646049	172.21.224.2	142.250.1.139	ICMP	98	Echo (ping) request id=0x6831, seq=3/768, ttl=64 (reply in 32)
32	10.646563	142.250.1.139	172.21.224.2	ICMP	98	Echo (ping) reply id=0x6831, seq=3/768, ttl=115 (request in 31)
33	10.646715	172.21.224.2	169.254.169.254	DNS	86	Standard query 0x94d7 PTR 139.1.250.142.in-addr.arpa
34	10.647413	169.254.169.254	172.21.224.2	DNS	120	Standard query response 0x94d7 PTR 139.1.250.142.in-addr.arpa PTR jw-in-f139.1e100.net
35	10.647633	172.21.224.2	35.235.244.34	SSH	210	Server: Encrypted packet (len=144)
36	10.647821	35.235.244.34	172.21.224.2	TCP	66	35193 → 22 [ACK] Seq=161 Ack=1057 Win=1050 Len=0 TSval=3147829824 TSecr=1526941714
37	10.799238	172.21.224.2	169.254.169.254	TCP	54	56664 → 80 [ACK] Seq=1 Ack=1 Win=63814 Len=0
38	10.799668	169.254.169.254	172.21.224.2	TCP	54	[TCP ACKed unseen segment] 80 → 56664 [ACK] Seq=1 Ack=2 Win=65535 Len=0
39	11.169666	35.235.244.34	172.21.224.2	SSH	130	Client: Encrypted packet (len=64)
40	11.169886	172.21.224.2	35.235.244.34	SSH	130	Server: Encrypted packet (len=64)

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

9. 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.

```
▼ 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)
```

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.

```
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x10 (DSCP: Unknown, ECN: Not-ECT)
  Total Length: 196
  Identification: 0xda77 (55927)
> 010. .... = Flags: 0x2, Don't fragment
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 64
  Protocol: TCP (6)
  Header Checksum: 0xbb86 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 172.21.224.2
  Destination Address: 35.235.244.34
```

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.

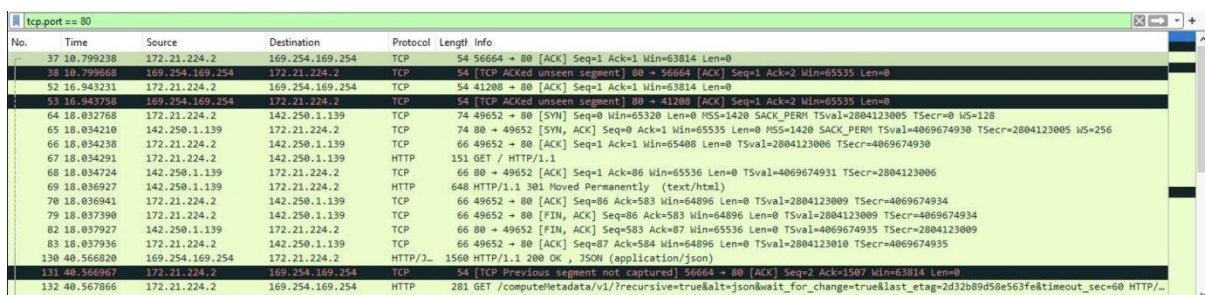
Task 5. Use filters to explore TCP packets

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.



The screenshot shows the Wireshark interface with the filter 'tcp.port == 80' applied. The packet list shows several packets, with the first packet (No. 37) selected. The packet details pane shows the structure of the selected packet: Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol. The packet bytes pane shows the raw data of the packet.

No.	Time	Source	Destination	Protocol	Length	Info
37	10.799238	172.21.224.2	169.254.169.254	TCP	54	56664 → 80 [ACK] Seq=1 Ack=1 Win=63814 Len=0
38	10.799668	169.254.169.254	172.21.224.2	TCP	54	[TCP ACKed unseen segment] 80 → 56664 [ACK] Seq=1 Ack=2 Win=65535 Len=0
52	16.943231	172.21.224.2	169.254.169.254	TCP	54	41288 → 80 [ACK] Seq=1 Ack=1 Win=63814 Len=0
53	16.943258	169.254.169.254	172.21.224.2	TCP	54	[TCP ACKed unseen segment] 80 → 41288 [ACK] Seq=1 Ack=2 Win=65535 Len=0
64	18.032768	172.21.224.2	142.250.1.139	TCP	74	49652 → 80 [SYN] Seq=0 Win=65536 Len=0 MSS=1420 SACK_PERM TSval=2804123005 TSecr=0 WS=128
65	18.034210	142.250.1.139	172.21.224.2	TCP	74	80 → 49652 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1420 SACK_PERM TSval=4069674930 TSecr=2804123005 WS=256
66	18.034238	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=1 Ack=1 Win=65488 Len=0 TSval=2804123006 TSecr=4069674930
67	18.034291	172.21.224.2	142.250.1.139	HTTP	151	GET / HTTP/1.1
68	18.034724	142.250.1.139	172.21.224.2	TCP	66	80 → 49652 [ACK] Seq=1 Ack=86 Win=65536 Len=0 TSval=4069674931 TSecr=2804123006
69	18.036927	142.250.1.139	172.21.224.2	HTTP	648	HTTP/1.1 301 Moved Permanently (text/html)
70	18.036941	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=86 Ack=583 Win=64896 Len=0 TSval=2804123009 TSecr=4069674934
79	18.037398	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [FIN, ACK] Seq=86 Ack=583 Win=64896 Len=0 TSval=2804123009 TSecr=4069674934
82	18.037927	142.250.1.139	172.21.224.2	TCP	66	80 → 49652 [FIN, ACK] Seq=583 Ack=87 Win=65536 Len=0 TSval=4069674935 TSecr=2804123009
83	18.037936	172.21.224.2	142.250.1.139	TCP	66	49652 → 80 [ACK] Seq=87 Ack=584 Win=64896 Len=0 TSval=2804123010 TSecr=4069674935
130	40.566820	169.254.169.254	172.21.224.2	HTTP/2	1560	HTTP/1.1 200 OK , JSON (application/json)
131	40.566967	172.21.224.2	169.254.169.254	TCP	54	[TCP Previous segment not captured] 56664 → 80 [ACK] Seq=2 Ack=1507 Win=63814 Len=0
132	40.567866	172.21.224.2	169.254.169.254	HTTP	281	GET /computeMetadata/v1/?recursive=true&alt=json&wait_for_change=true&last_etag=2d32b89d58e563fe&timeout_sec=60 HTTP/...

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**.



The screenshot shows the detailed view of packet 37. It displays the Ethernet II header, the Internet Protocol Version 4 header, and the Transmission Control Protocol header. The packet bytes pane shows the raw data of the packet.

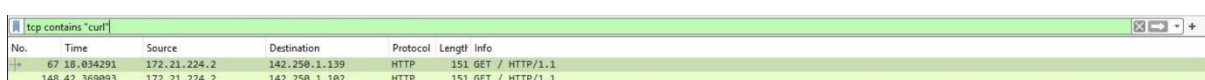
Frame 37: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
Ethernet II, Src: 42:01:ac:15:e0:02 (42:01:ac:15:e0:02), Dst: 42:01:ac:15:e0:01 (42:01:ac:15:e0:01)
Internet Protocol Version 4, Src: 172.21.224.2, Dst: 169.254.169.254
Transmission Control Protocol, Src Port: 56664, Dst Port: 80, Seq: 1, Ack: 1, Len: 0

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.



The screenshot shows the Wireshark interface with the filter 'tcp contains "curl"' applied. The packet list shows two packets, both of which are HTTP GET requests.

No.	Time	Source	Destination	Protocol	Length	Info
67	18.034291	172.21.224.2	142.250.1.139	HTTP	151	GET / HTTP/1.1
148	42.360993	172.21.224.2	142.250.1.102	HTTP	151	GET / HTTP/1.1