**Activity: Analyse your first packet with Wireshark**

**Overview:** As a security analyst, you’ll need to analyse network traffic in order to learn what type of traffic is being sent to and from systems on the networks you’ll be working with. Analysing packets can help security teams interpret and understand network communications. Network protocol analysers such as Wireshark, which has a graphical user interface or GUI, can help you examine packet data during your investigations. Since network packet data is complex, network protocol analysers (packet sniffers) like Wireshark are designed to help you find patterns and filter the data in order to focus on the network traffic that is most relevant to your security investigations.

**Scenario:** In this scenario, you’re a security analyst investigating traffic to a website. You’ll analyse 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.

**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.

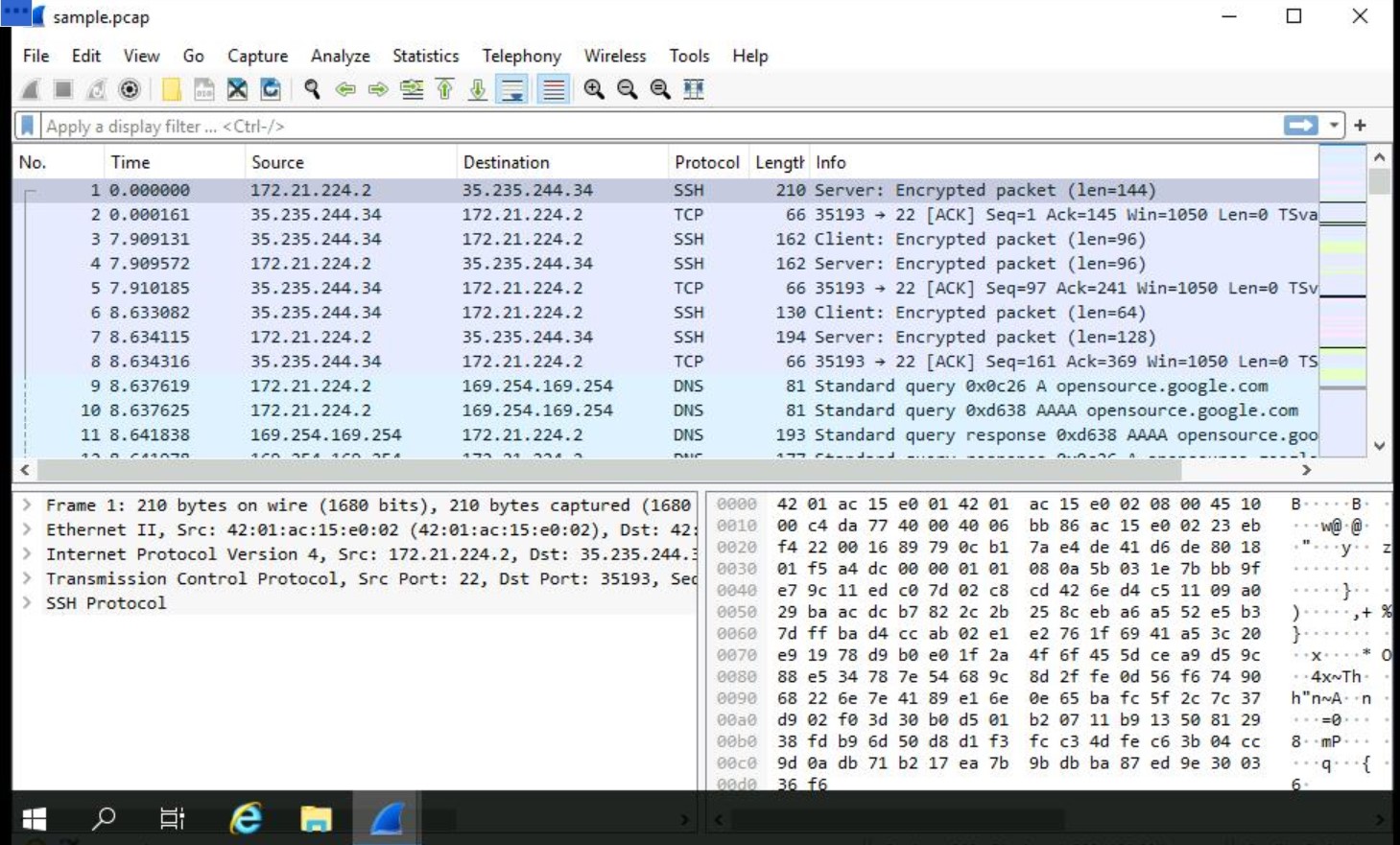
**Start your lab:** click on “start lab” to start the lab. Click the **Windows VM** button to launch the Windows virtual machine in a new browser tab.

**Task 1. Explore data with Wireshark**

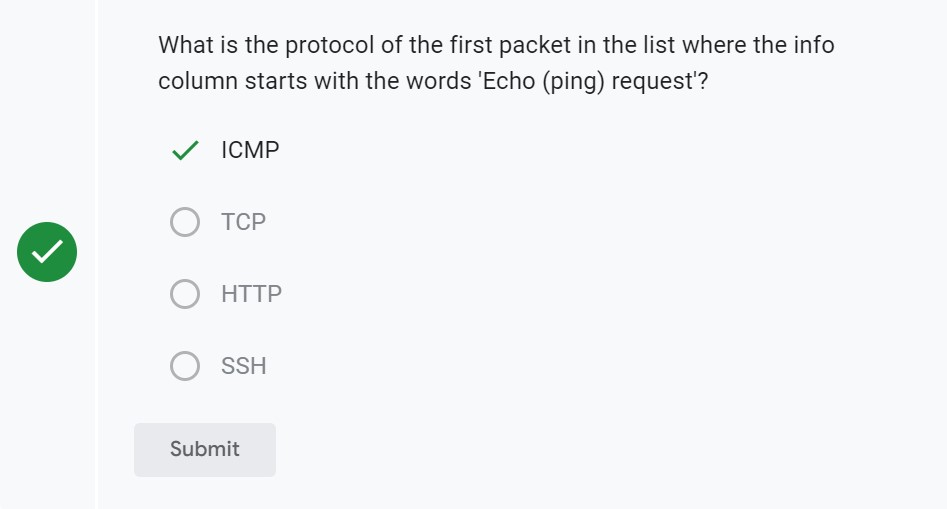
In this task, you must open a network packet capture file that contains data captured from a system that made web requests to a site. You need to open this data with Wireshark to get an overview of how the data is presented in the application.

1. 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.

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

1. Scroll down the packet list until a packet is listed where the info column starts with the words 'Echo (ping) request'.

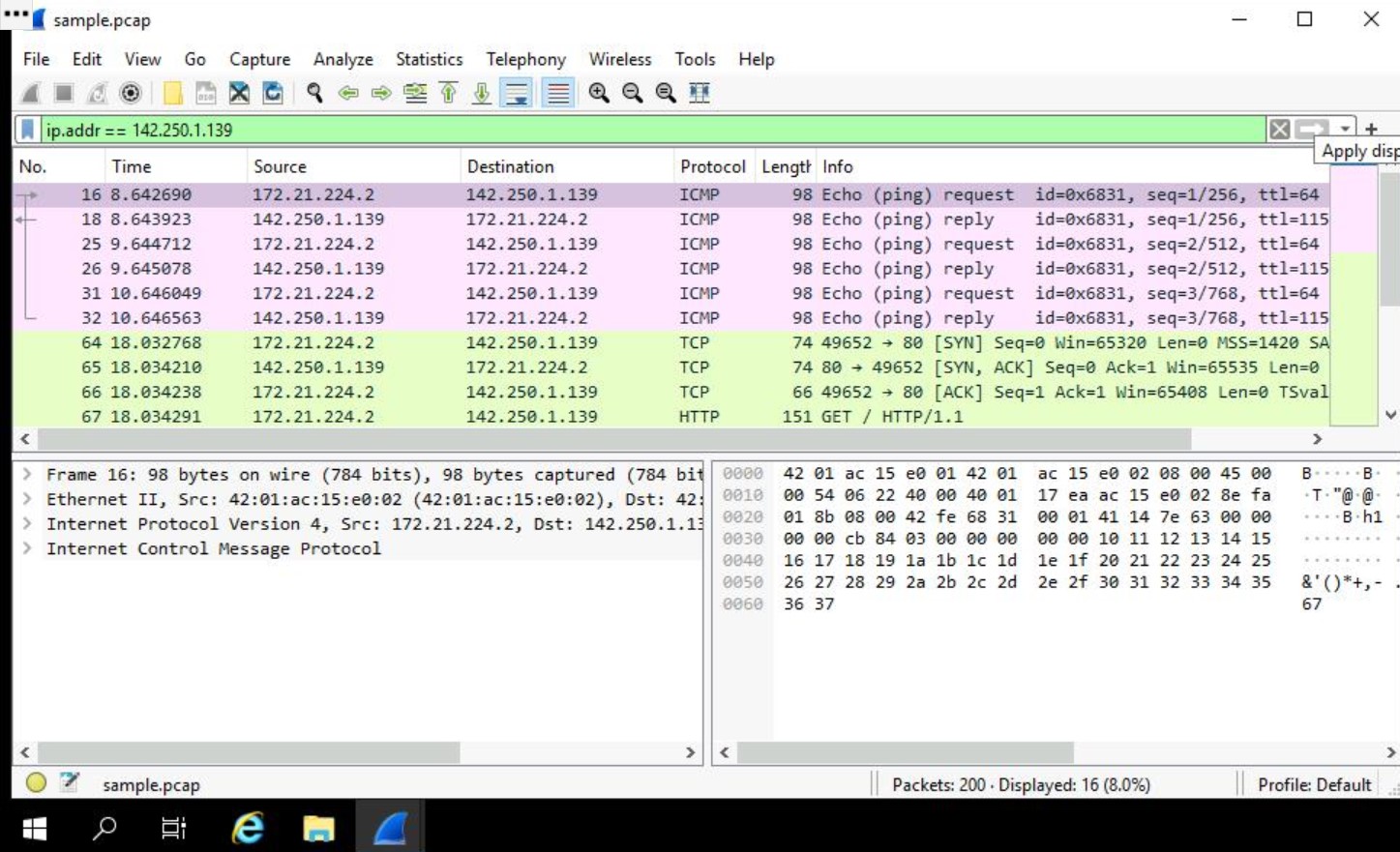


## **Task 2. Apply a basic Wireshark filter and inspect a packet**

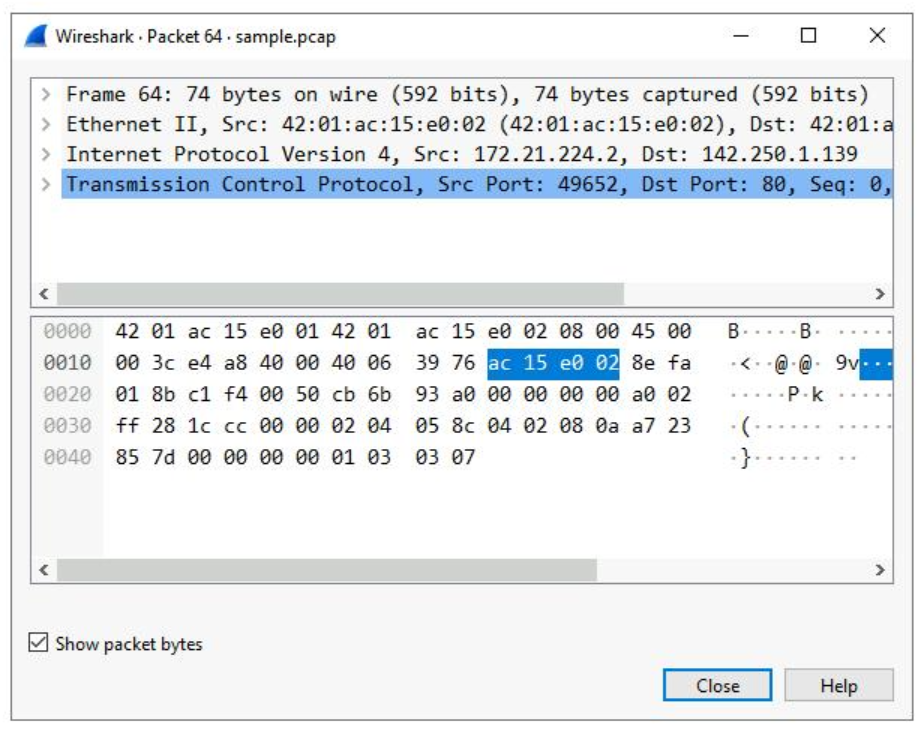
In this task, you’ll open a packet in Wireshark for more detailed exploration and filter the data to inspect the network layers and protocols contained in the packet.

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

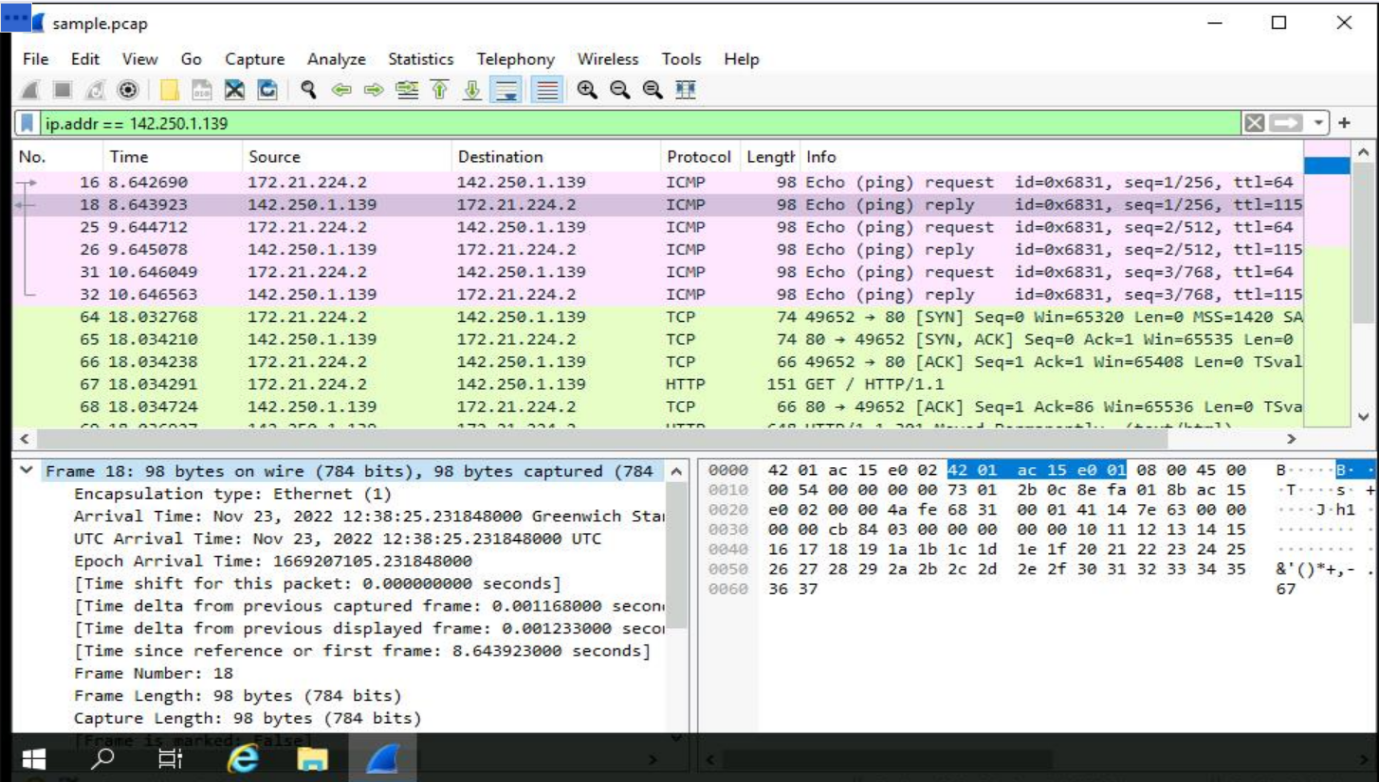


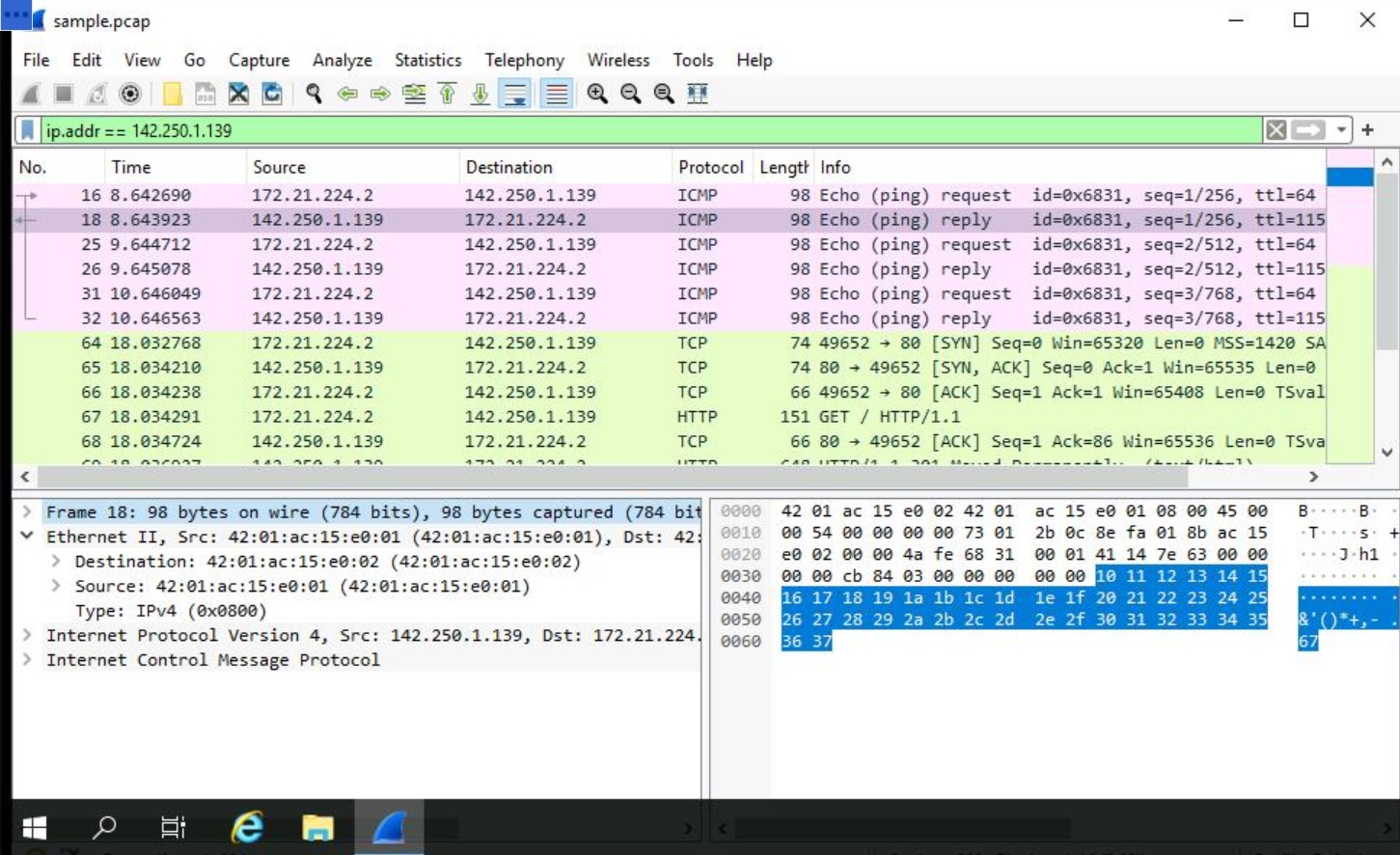
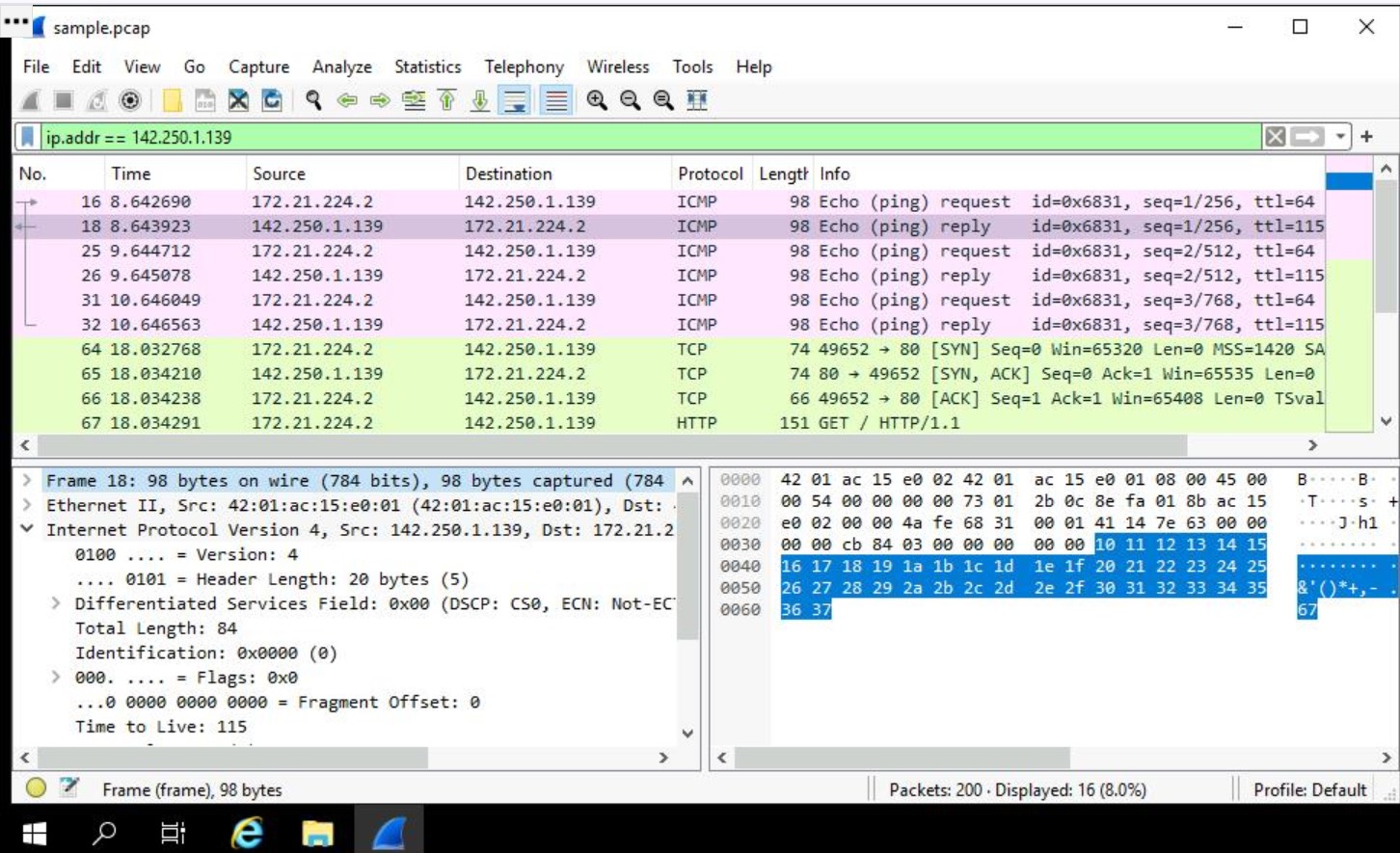
1. Press **ENTER** or click the **Apply display filter** icon in the filter text box.
2. Double-click the first packet that lists **TCP** as the protocol.



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 (“.”).

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

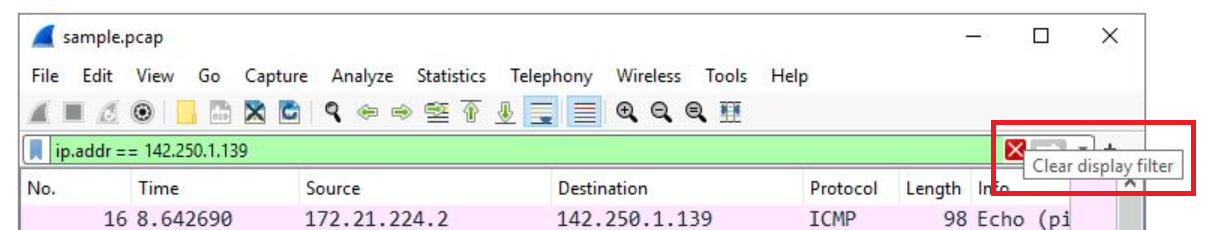
This provides you with details about the overall network packet, or frame, including the frame length and the arrival time of the packet. At this level, you’re viewing information about the entire packet of data.

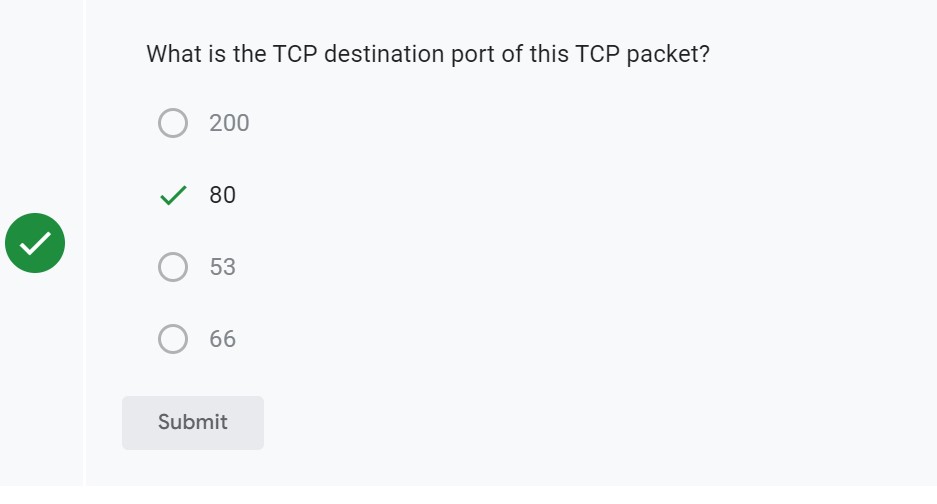
1. Double-click **Frame** again to collapse the subtree and then double-click the **Ethernet II** subtree. This item contains details about the packet at the Ethernet level, including the source and destination MAC addresses and the type of internal protocol that the Ethernet packet contains.
2. Double-click **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.

1. 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.





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

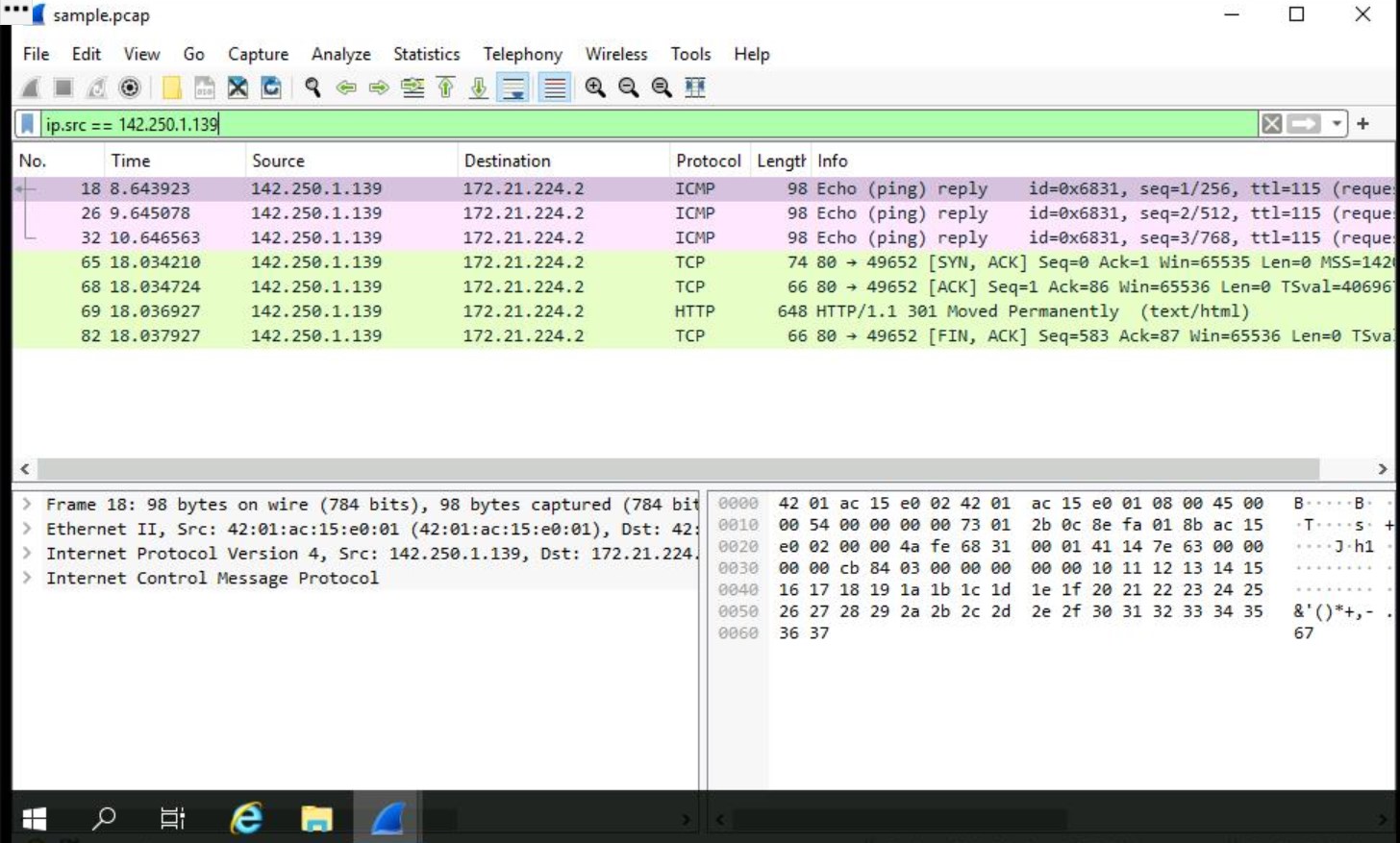
This provides a detailed view of the TCP flags set in this packet.

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

**Task 3. Use filters to select packets**

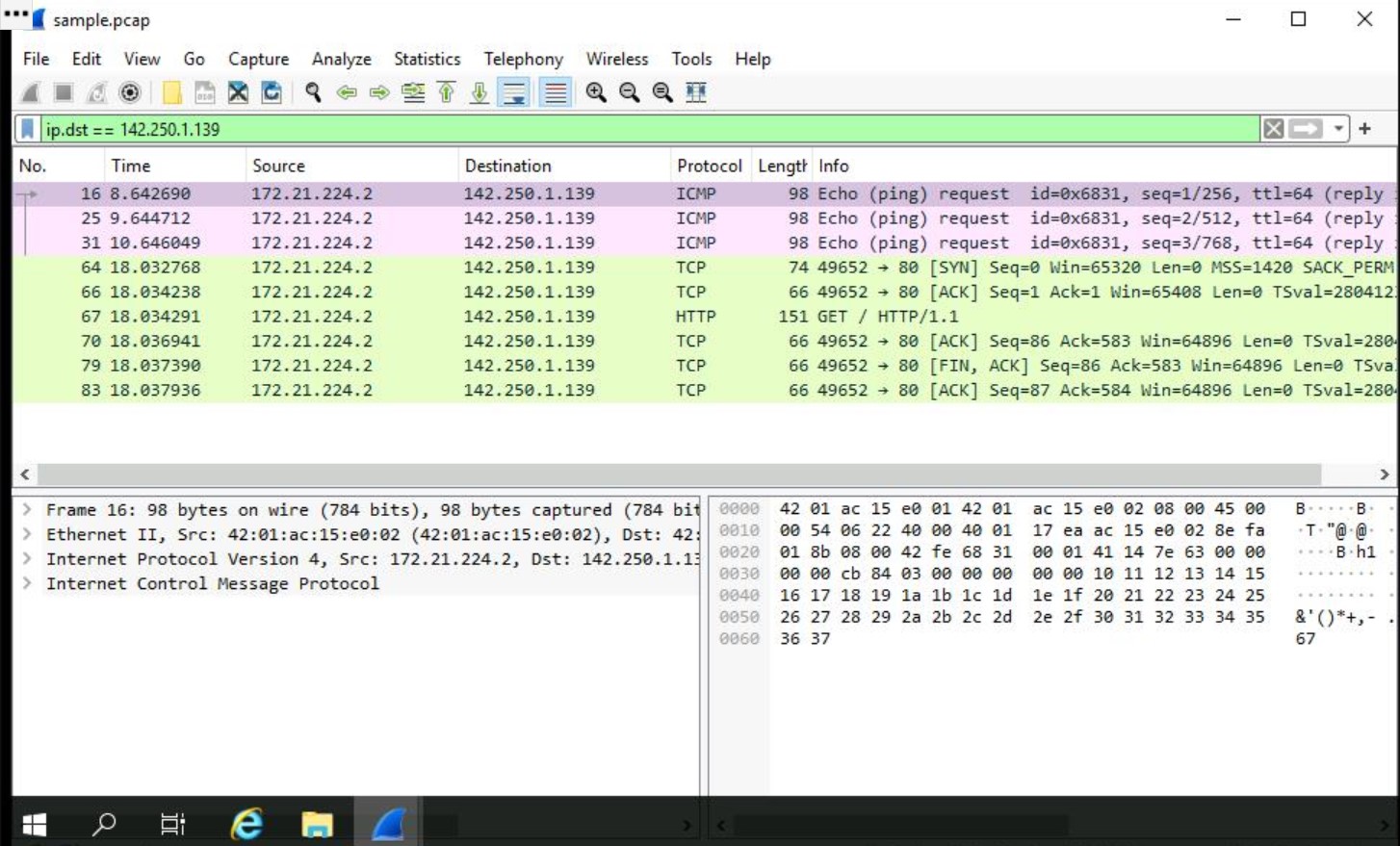
In this task, you’ll use filters to analyze specific network packets based on where the packets came from or where they were sent to. You’ll explore how to select packets using either their physical Ethernet Media Access Control (MAC) address or their Internet Protocol (IP) address.

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

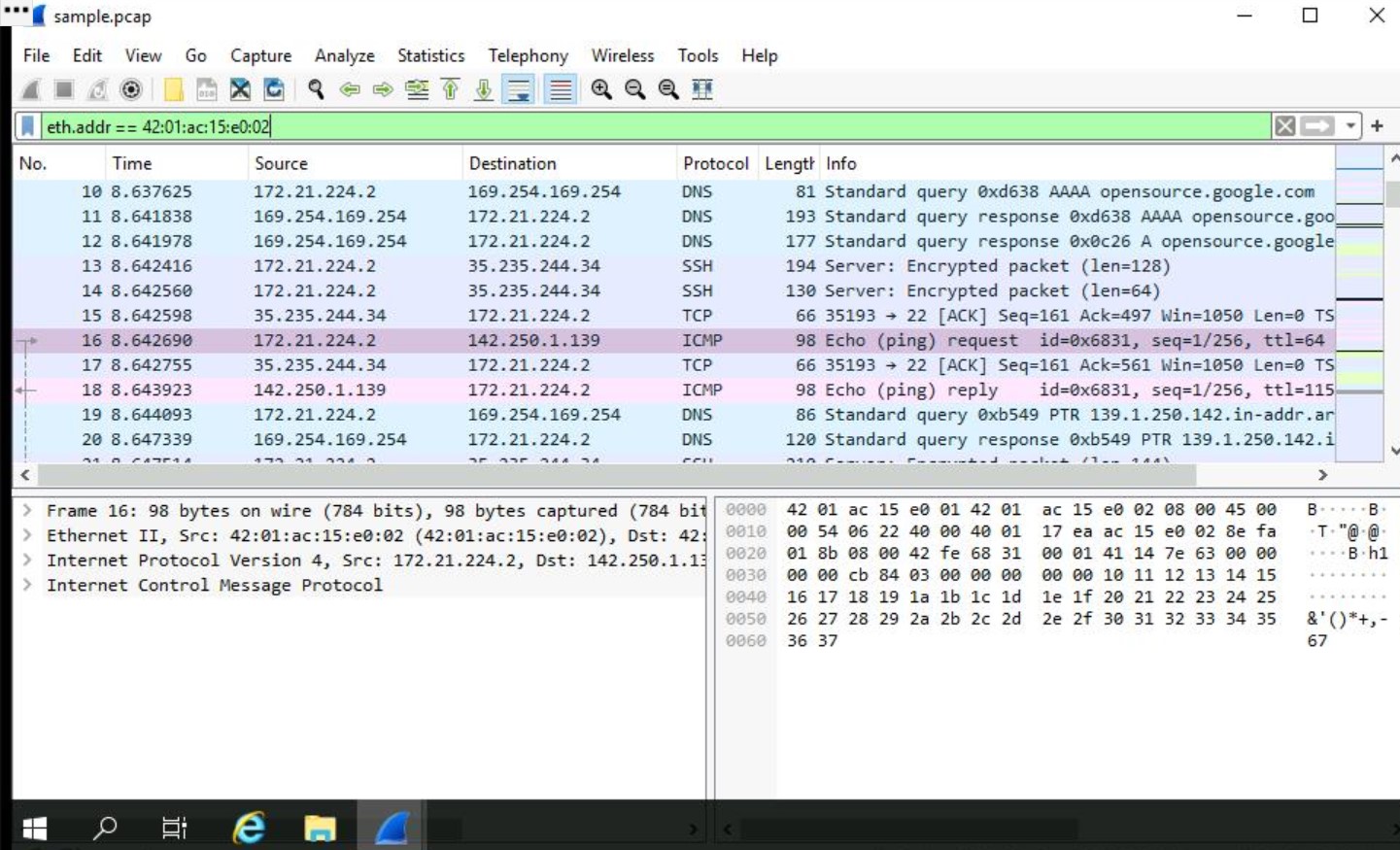
1. Press **ENTER** or click the **Apply display filter** icon in the filter text box.
2. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.
3. Enter the following filter to select traffic for a specific destination IP address only:

ip.dst == 142.250.1.139

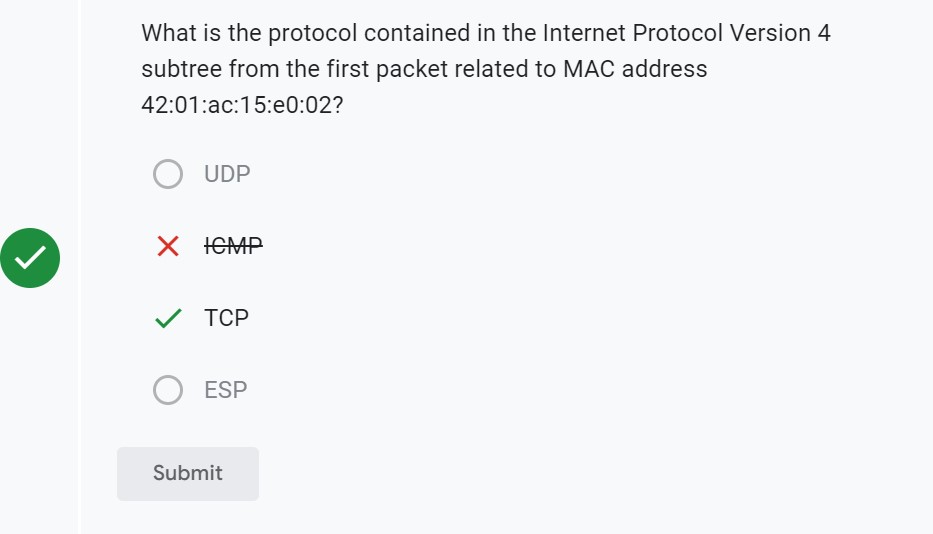


1. Press **ENTER** or click the **Apply display filter** icon in the filter text box.
2. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.
3. Enter the following filter to select traffic to or from a specific Ethernet MAC address. This filters traffic related to one MAC address, regardless of the other protocols involved:

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



1. Press **ENTER** or click the **Apply display filter** icon in the filter text box.
2. Double-click the first packet in the list. You may need to scroll back to display the first packet in the filtered list.
3. Double-click the **Ethernet II** subtree if it is not already open.
4. Double-click the **Ethernet II** subtree to close it.
5. Double-click the **Internet Protocol Version 4** subtree to expand it and scroll down until the **Time to Live** and **Protocol** fields appear.
6. Click the **X** icon to close the detailed packet inspection window.
7. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the MAC address filter.



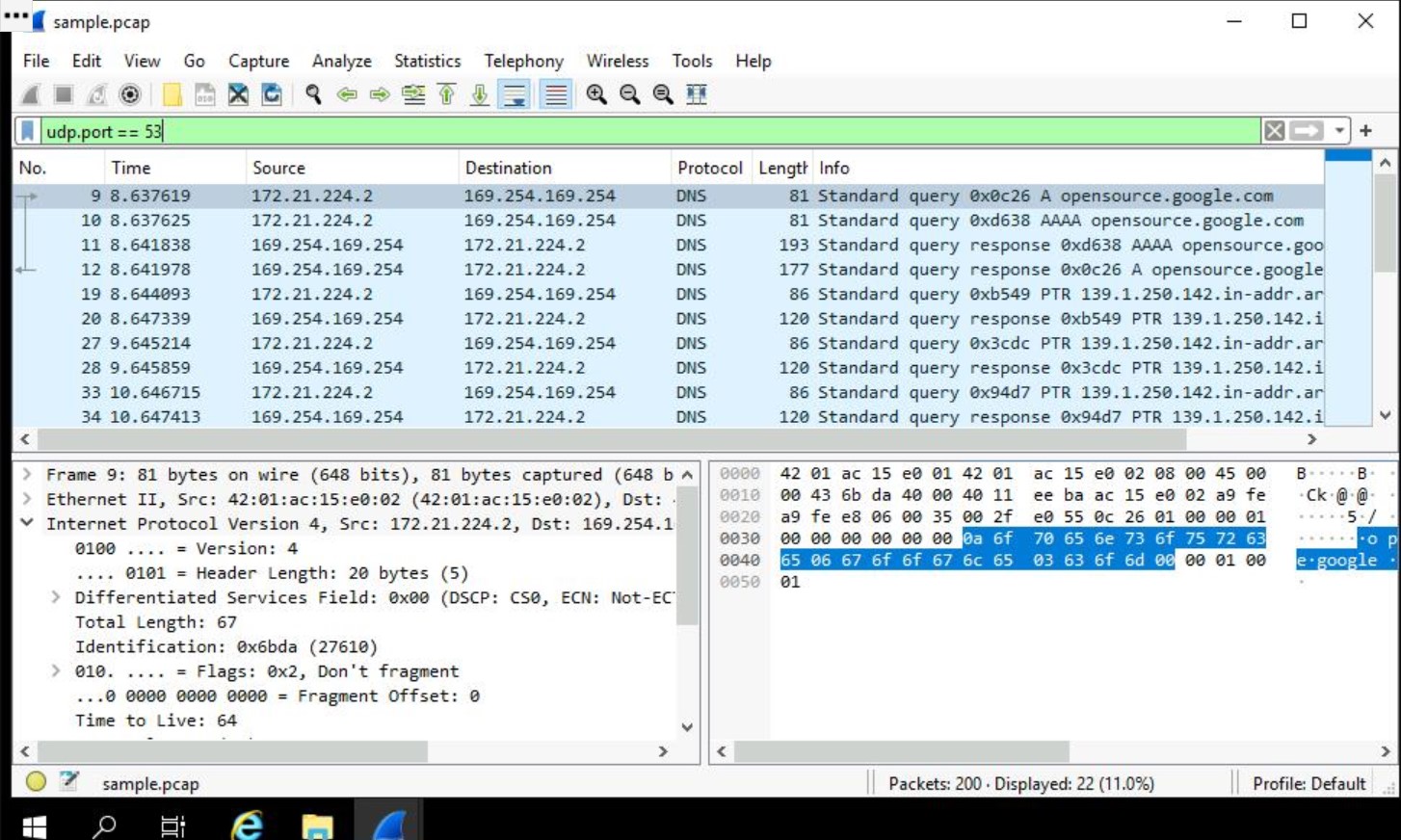
## **Task 4. Use filters to explore DNS packets**

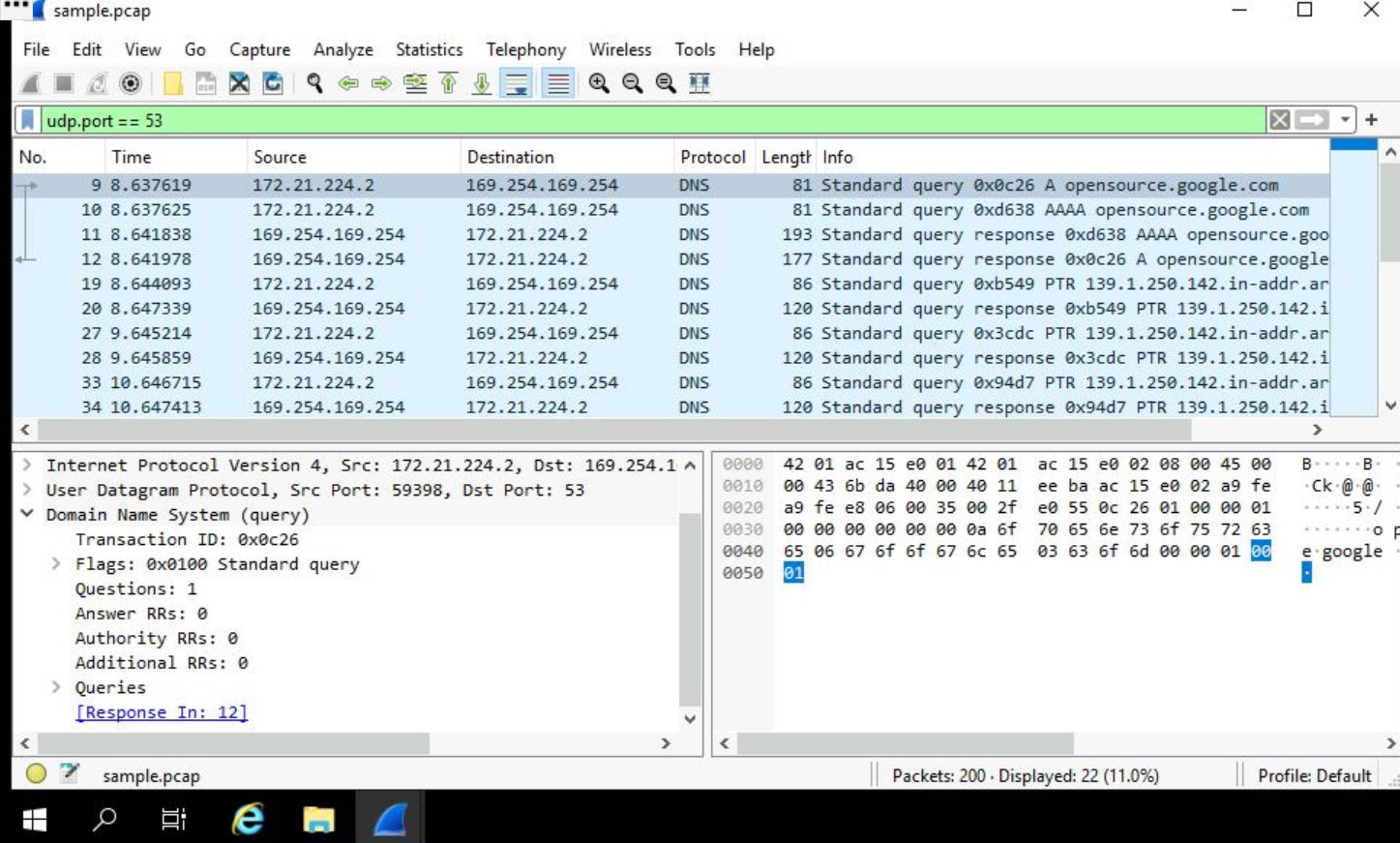
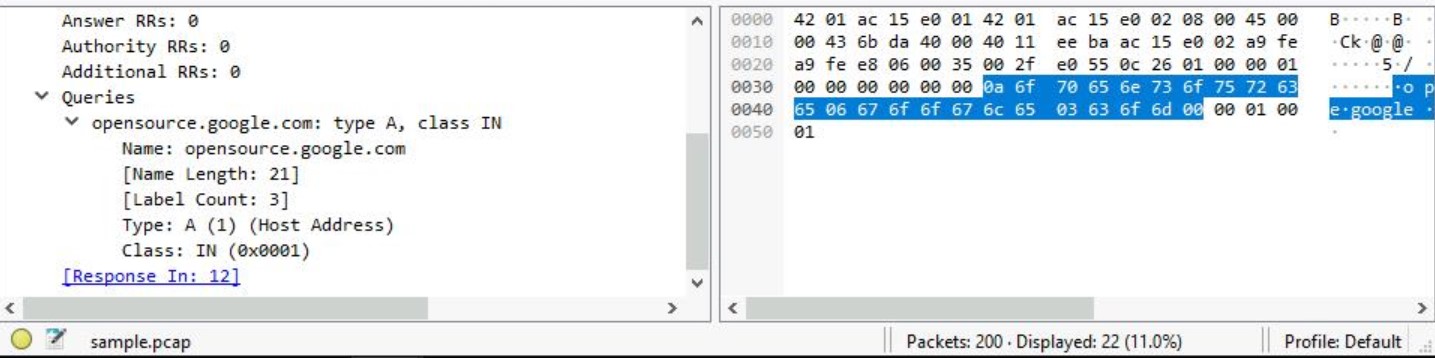
In this task, you’ll use filters to select and examine DNS traffic. Once you‘ve selected sample DNS traffic, you’ll drill down into the protocol to examine how the DNS packet data contains both queries (names of internet sites that are being looked up) and answers (IP addresses that are being sent back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

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



* 1. Double-click the first packet in the list to open the detailed packet window.
  2. Scroll down and double-click the **Domain Name System (query)** subtree to expand it.
  3. Scroll down and double-click **Queries**.
  4. Click the **X** icon to close the detailed packet inspection window.
  5. Double-click the fourth packet in the list to open the detailed packet window.
  6. Scroll down and double-click the **Domain Name System (query)** subtree to expand it.
  7. Scroll down and double-click **Answers**, which is in the **Domain Name System (query)** subtree.
  8. Click the **X** icon to close the detailed packet inspection window.
  9. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the filter.

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



139.1.250.142



172.21.224.1



142.250.1.139



169.254.169.254

**Task 5. Use filters to explore TCP packets**

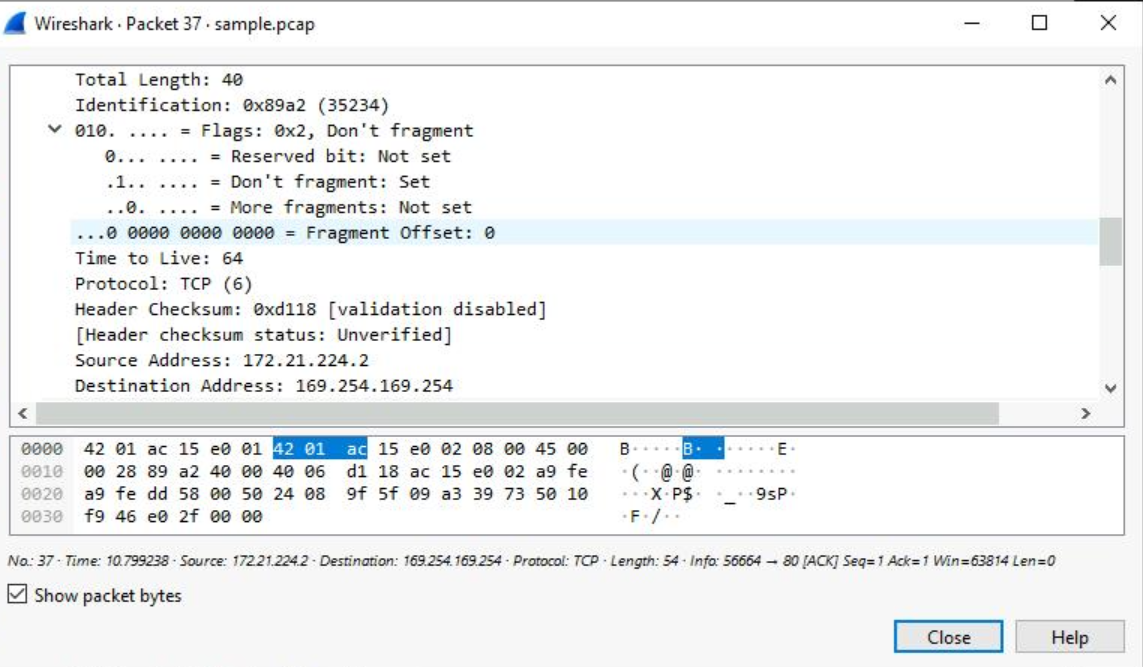
In this task, you’ll use additional filters to select and examine TCP packets. You’ll learn how to search for text that is present in payload data contained inside network packets. This will locate packets based on something such as a name or some other text that is of interest to you.

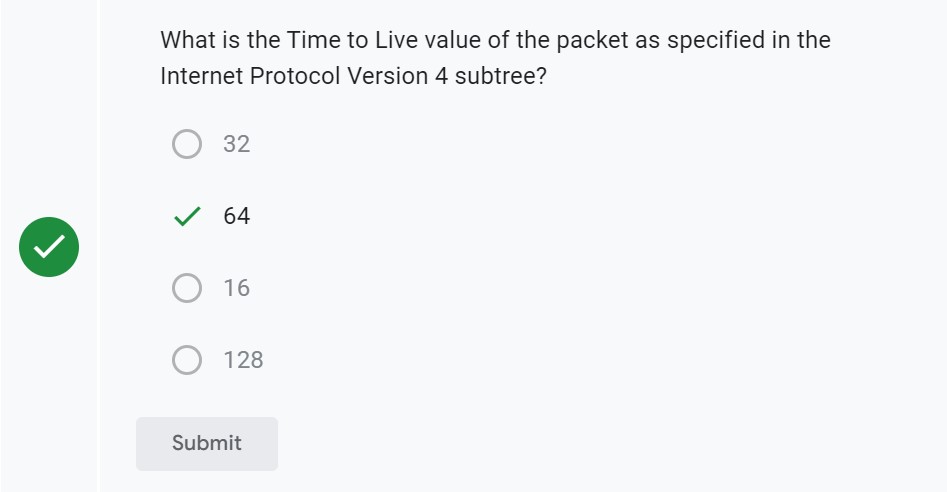
1. Enter the following filter to select TCP port 80 traffic. TCP port 80 is the default port that is associated with web traffic:

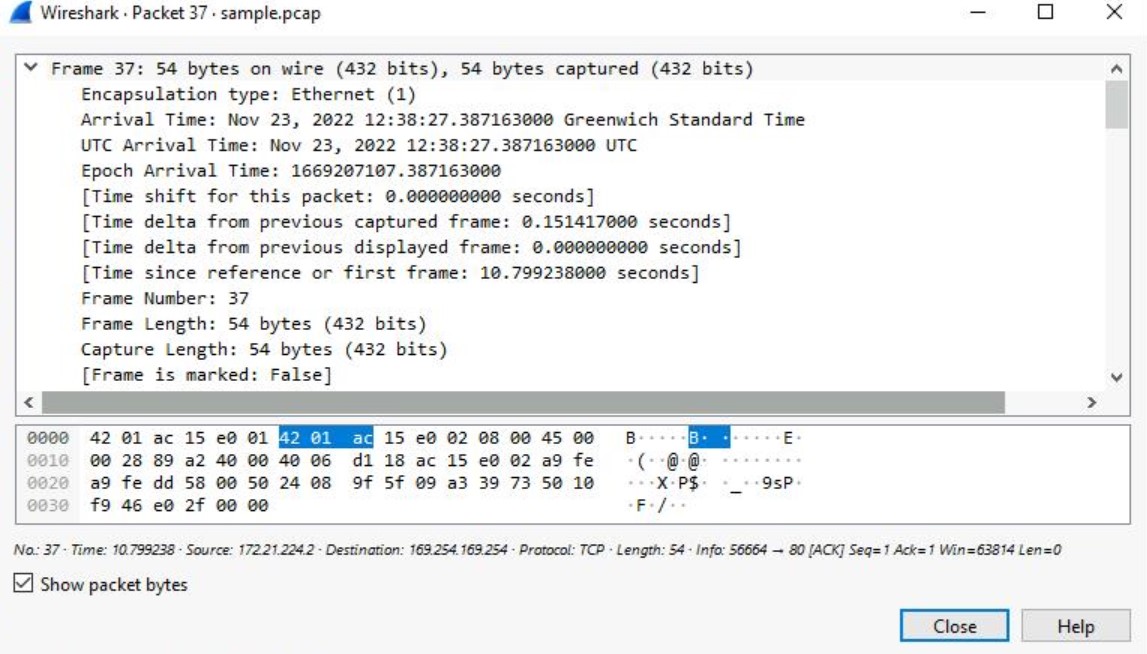
tcp.port == 80

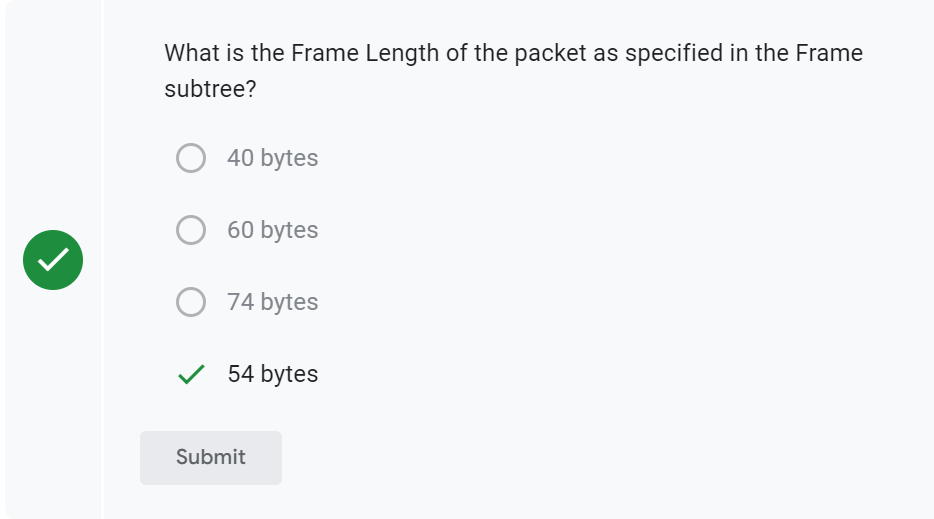


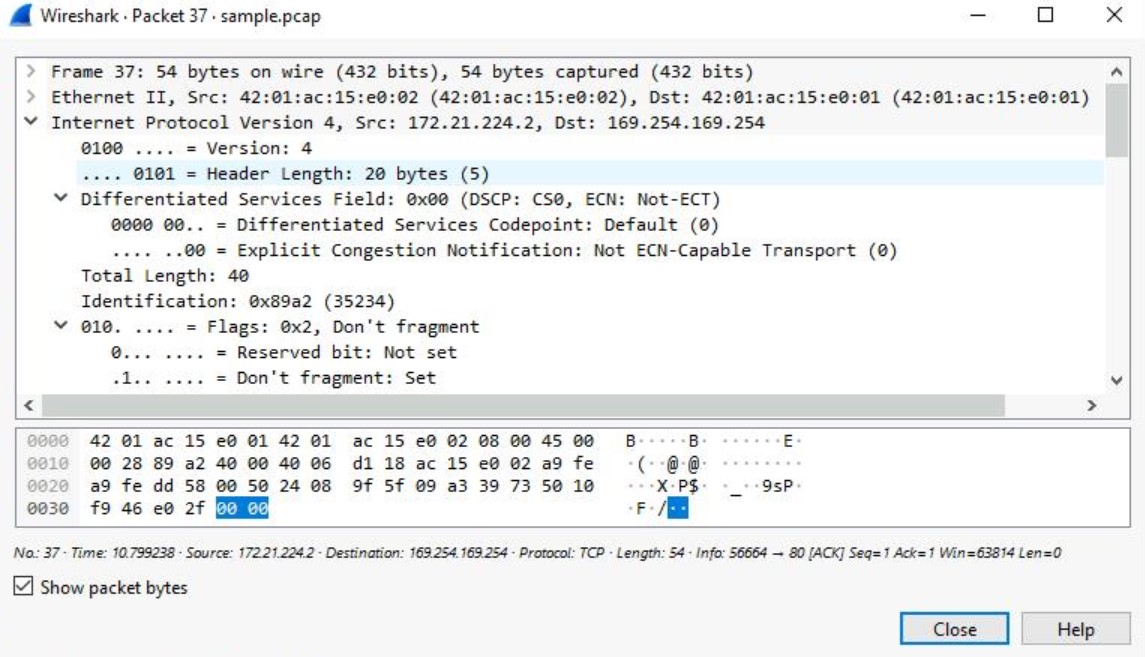
1. Press **ENTER** or click the **Apply display filter** icon in the filter text box.
2. Double-click the first packet in the list. The **Destination** IP address of this packet is 169.254.169.254.

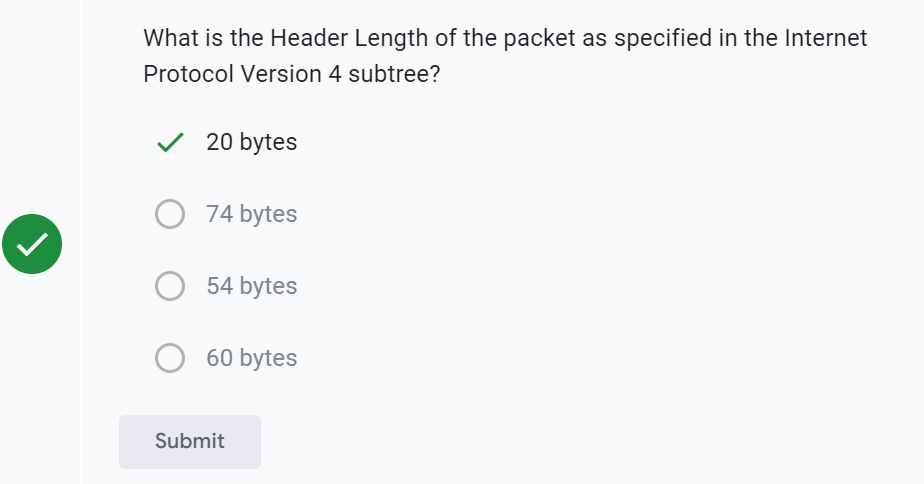






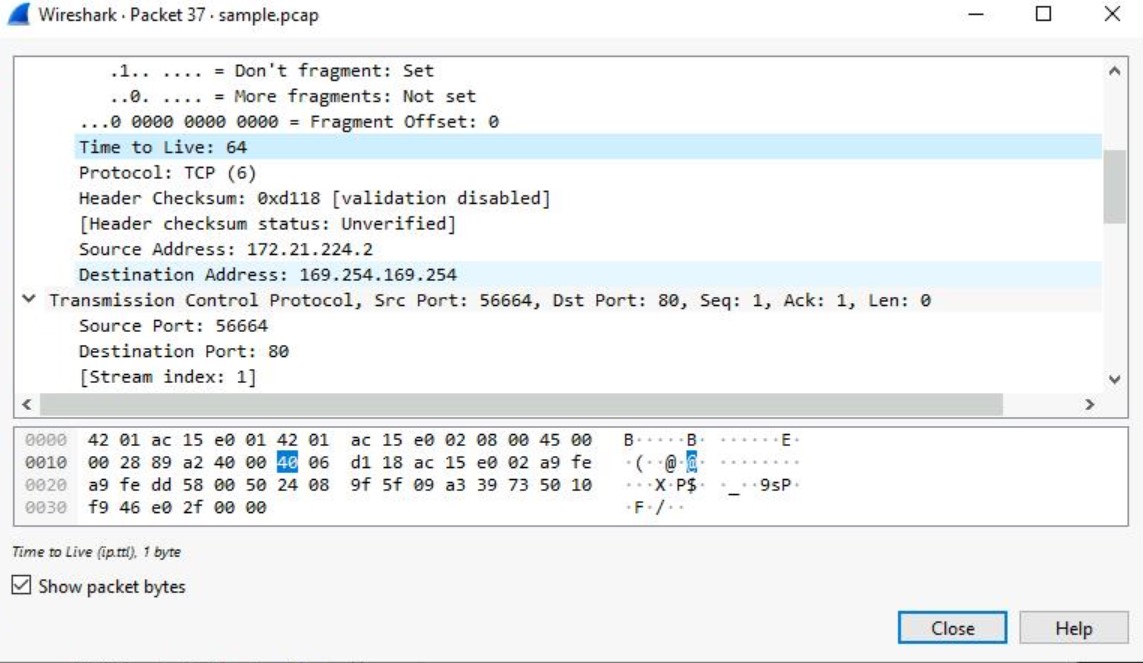


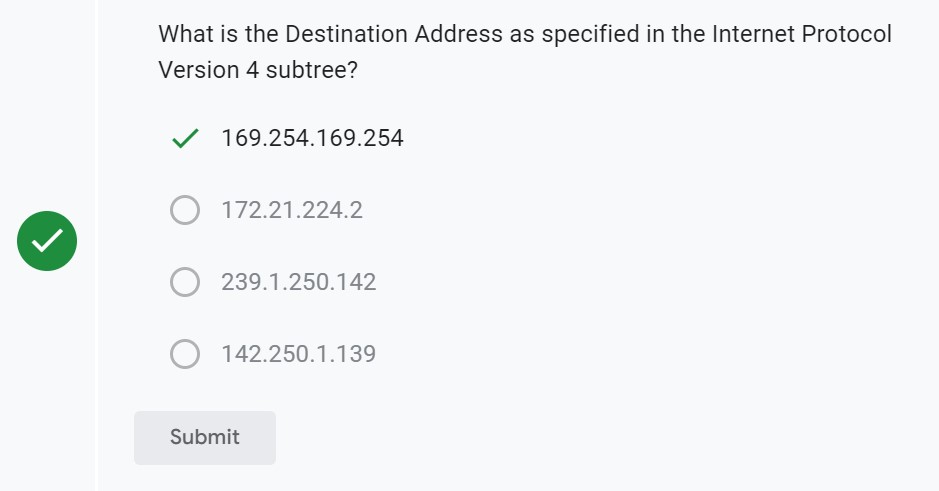




1. Click the **X** icon to close the detailed packet inspection window.
2. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the filter.
3. Enter the following filter to select TCP packet data that contains specific text data.

tcp contains "curl"





1. 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**

You now have practical experience using Wireshark to

* open saved packet capture files,
* view high-level packet data, and
* use filters to inspect detailed packet data.