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:

1. Identify the source and destination IP addresses involved in this web browsing session.
2. Examine the protocols that are used when the user makes the connection to the website.
3. Analyze the data packet to identify the type of information sent and received by the systems that connect to each other when the network data is captured.

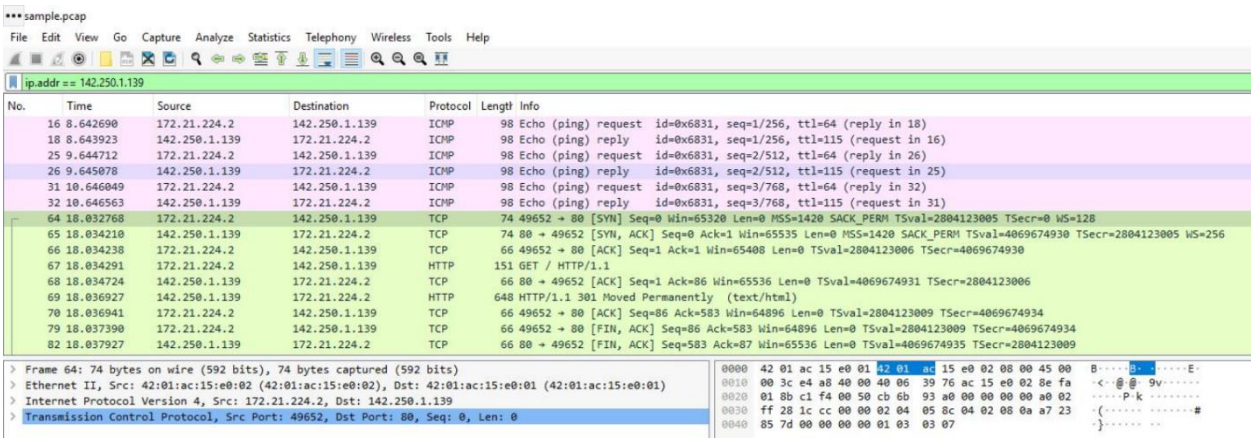
An overview of the key property columns listed for each packet:

- 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.

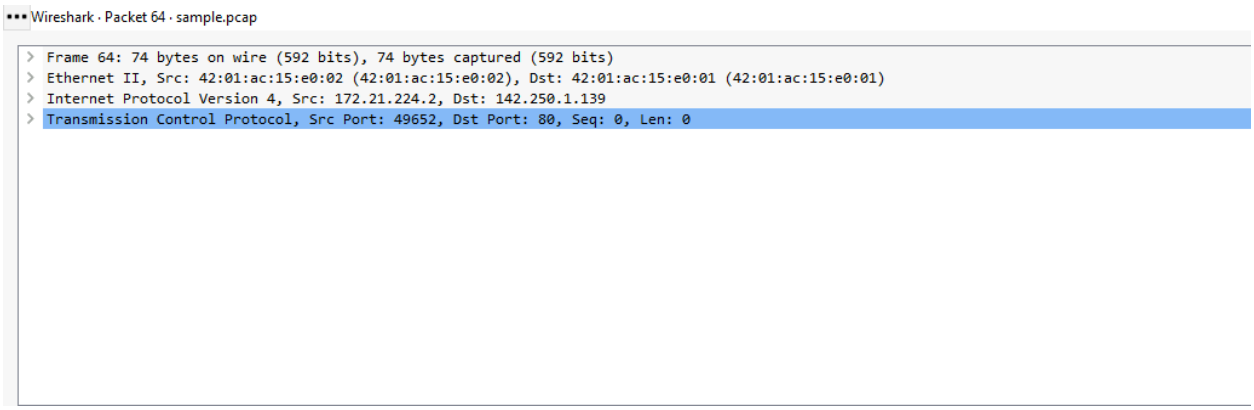
Solutions

Task 1. Apply a basic Wireshark filter and inspect a packet

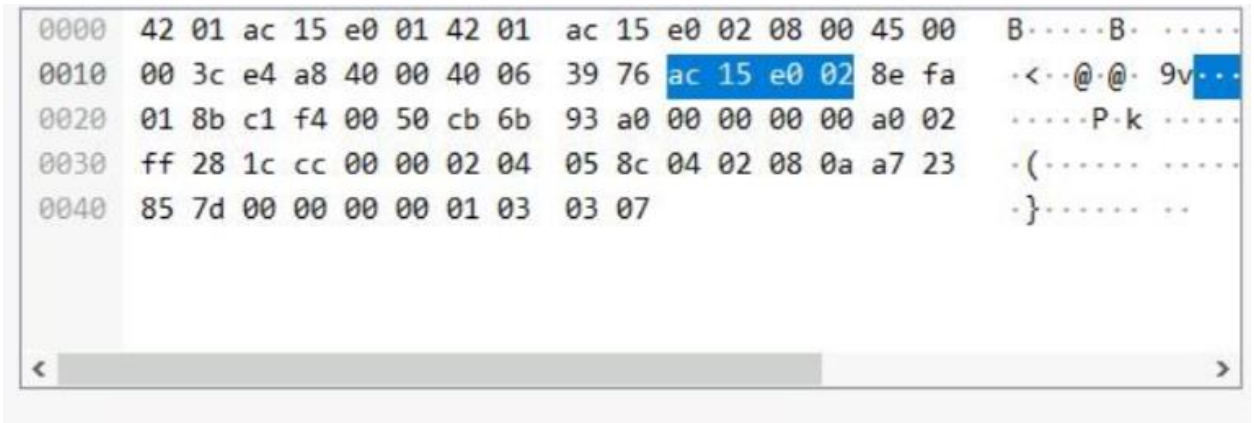
In this task, you'll open a packet in Wireshark for more detailed exploration and filter the data to inspect the network layers and protocols contained in the packet.



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.



The upper section of this window contains subtrees where Wireshark will provide you with an analysis of the various parts of the network packet.



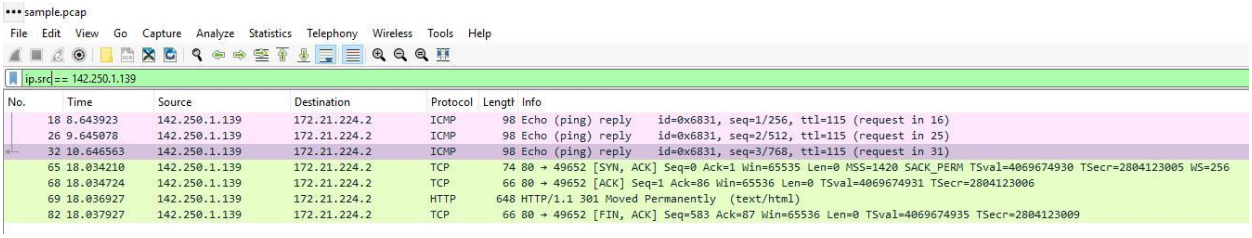
The lower section of the window contains the raw packet data displayed in hexadecimal and ASCII text. There is also placeholder text for fields where the character data does not apply, as indicated by the dot (“.”).

Task 2. 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.

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

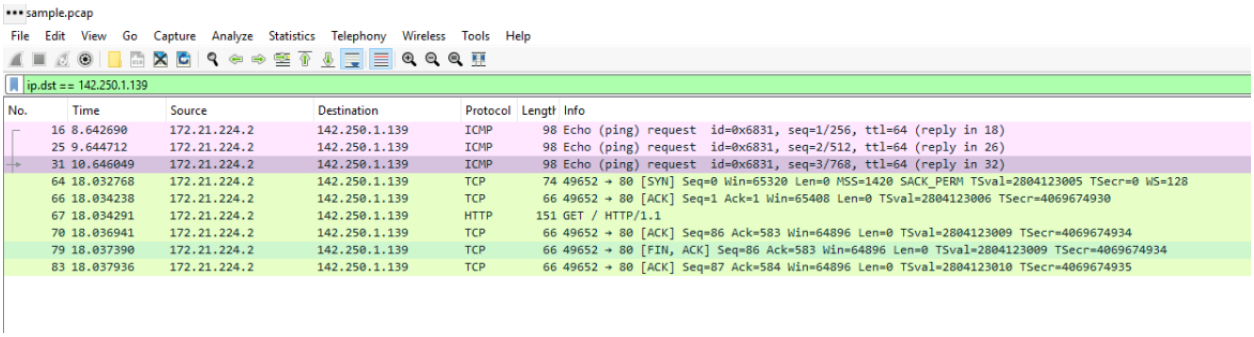
ip.src == 142.250.1.139



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

Enter the following filter to select traffic for a specific destination IP address only:

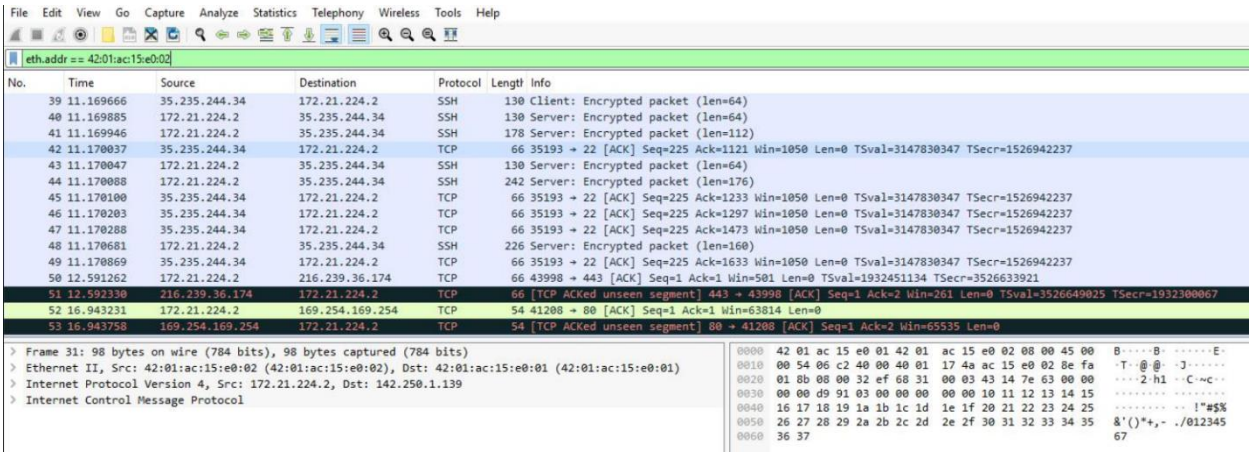
ip.dst == 142.250.1.139

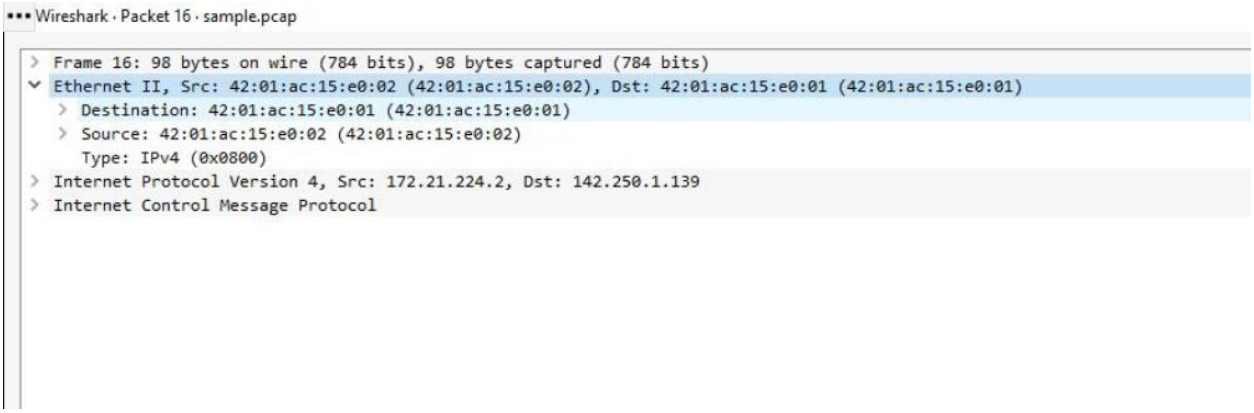


A filtered list is returned with fewer entries than before. It contains only packets that came from **142.250.1.139**.

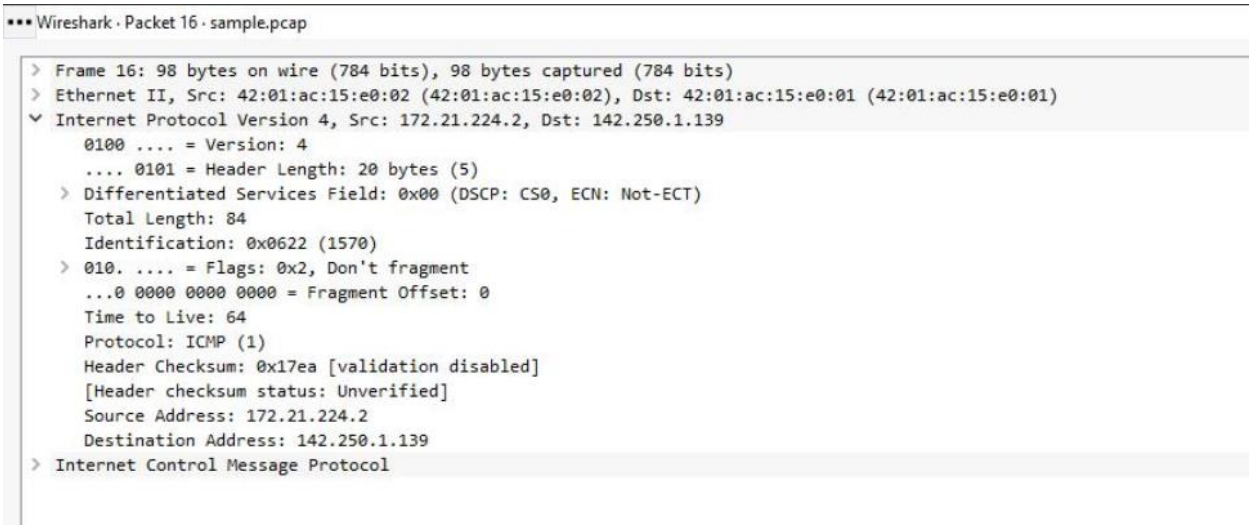
Enter the following filter to select traffic to or from a specific Ethernet MAC address. This filters traffic related to one MAC address, regardless of the other protocols involved:

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





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



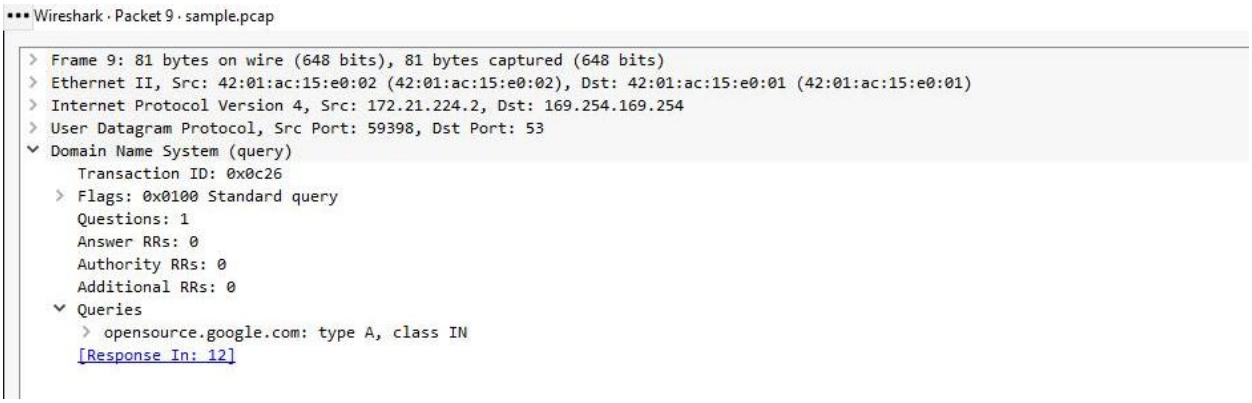
Task 3. Use filters to explore DNS packets

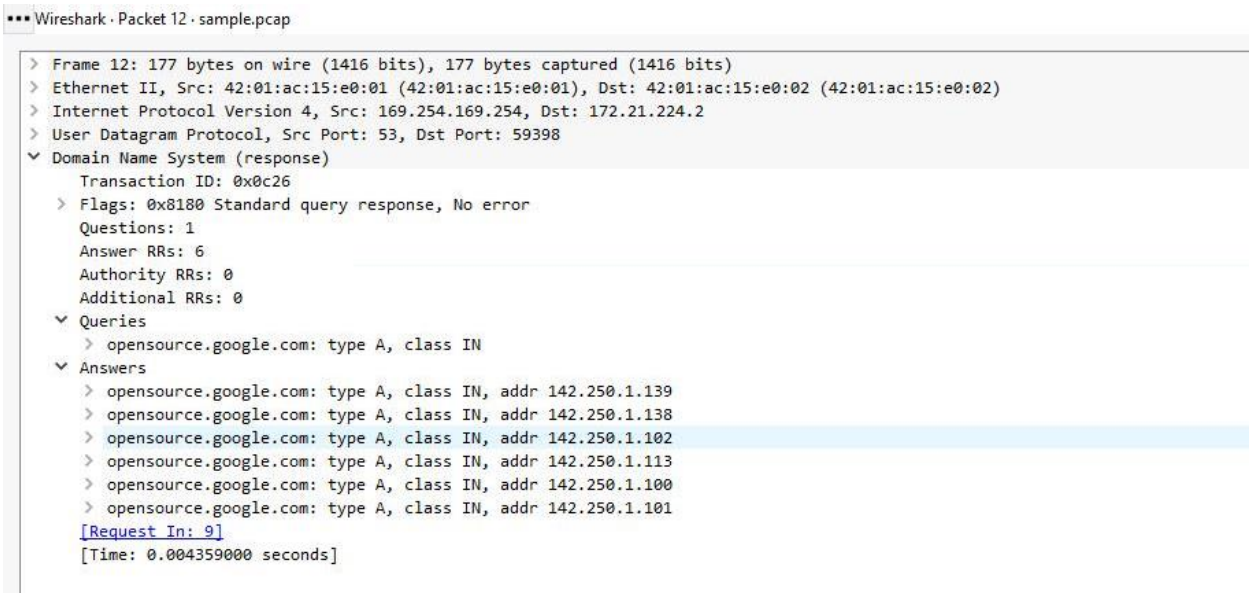
In this task, you'll use filters to select and examine DNS traffic. Once you've selected sample DNS traffic, you'll drill down into the protocol to examine how the DNS packet data contains

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).

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
```





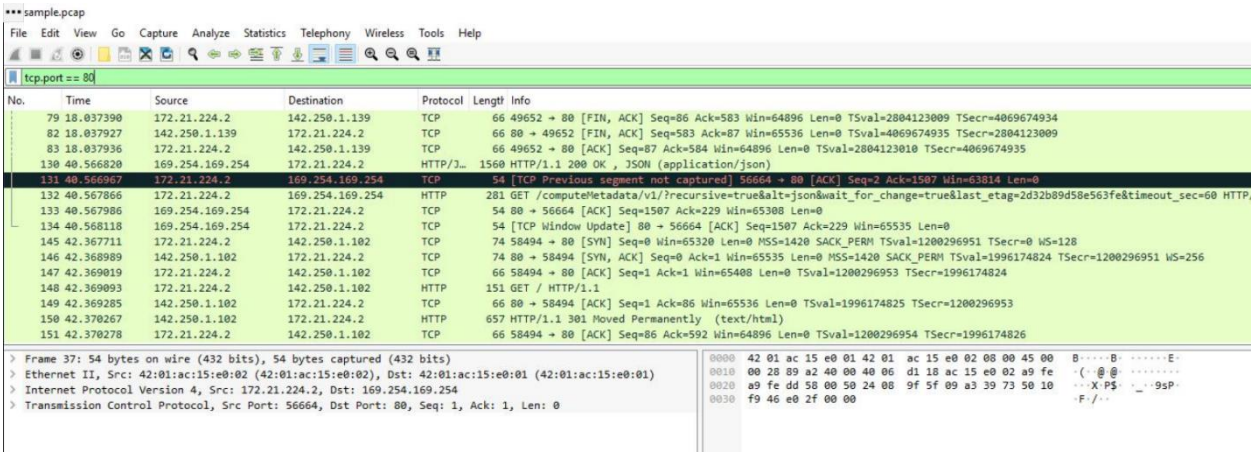
The IP address 142.250.1.139 is displayed in the expanded Answers section for the DNS query for **opensource.google.com**.

Task 4. 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.

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



Enter the following filter to select TCP packet data that contains specific text data.

tcp contains "curl"