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.

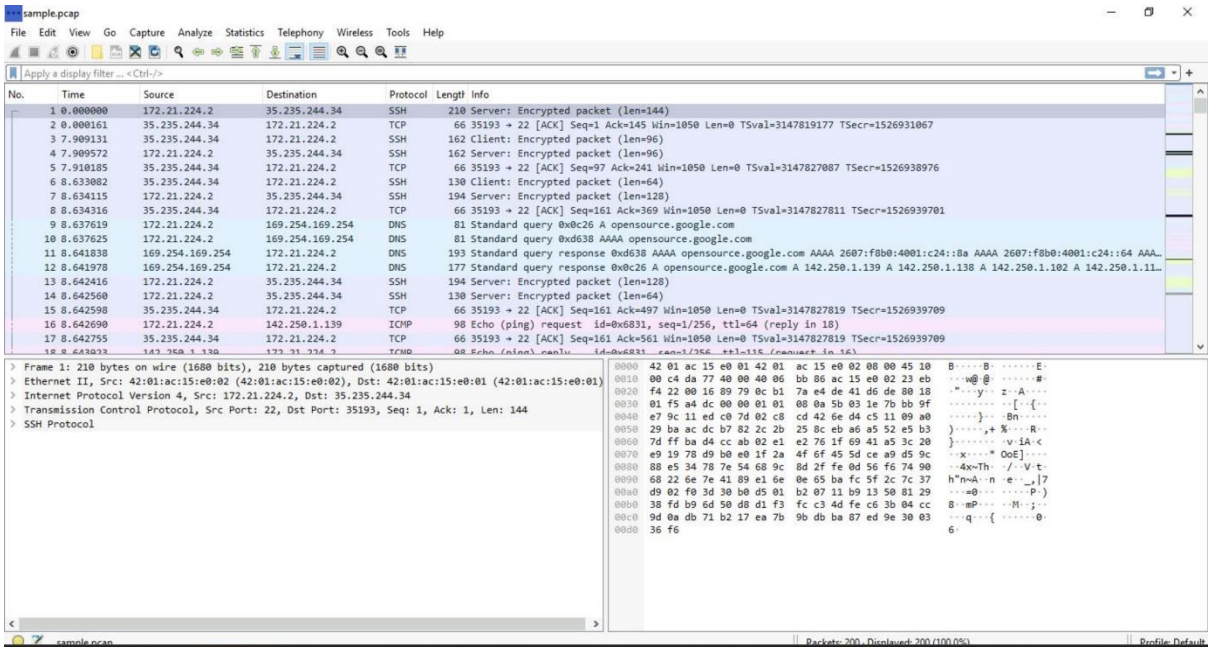
Here's how you'll do this: **First**, you'll open the packet capture file and explore the basic Wireshark graphic user interface. **Second**, you'll open a detailed view of a single packet and explore how to examine the various protocol and data layers inside a network packet. **Third**, you'll apply 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:

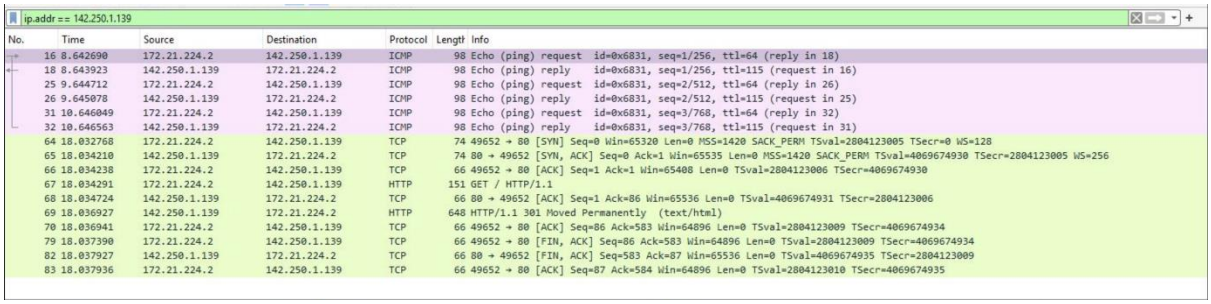
- **No.** : The index number of the packet in this packet capture file
- **Time**: The timestamp of the packet
- **Source**: The source IP address
- **Destination**: The destination IP address
- **Protocol**: The protocol contained in the packet
- **Length**: The total length of the packet
- **Info**: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

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

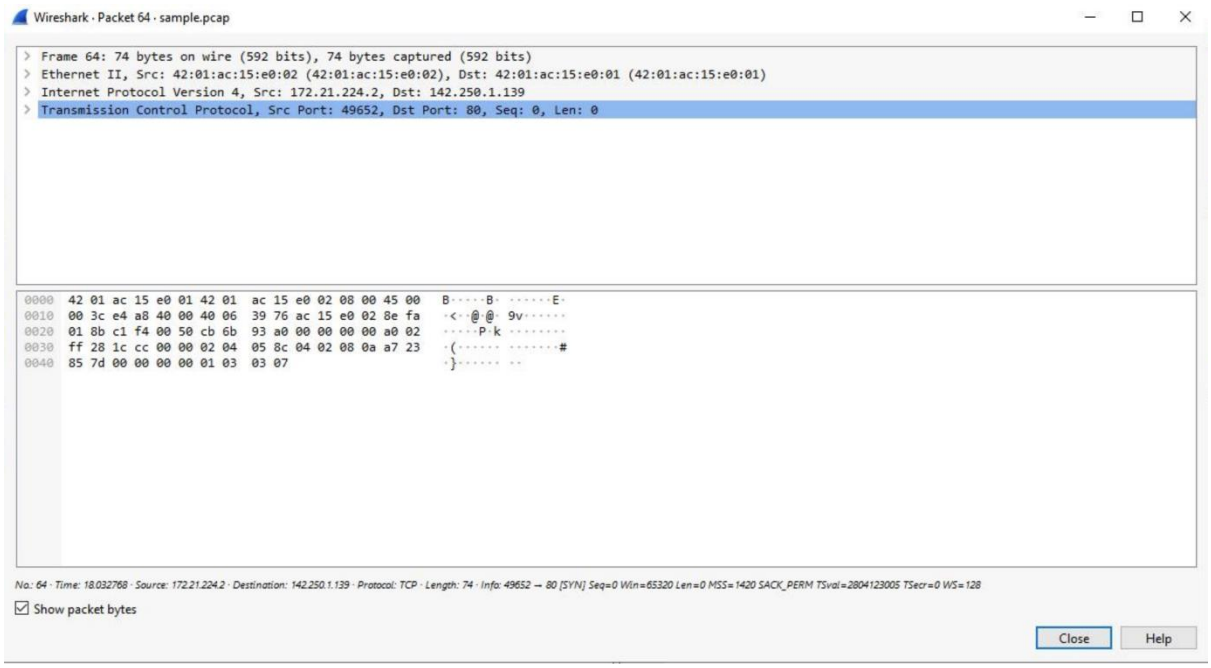
1. Enter the following filter for traffic associated with a specific IP address. Enter this into the **Apply a display filter...** text box immediately above the list of packets:

ip.addr == 142.250.1.139

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



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



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

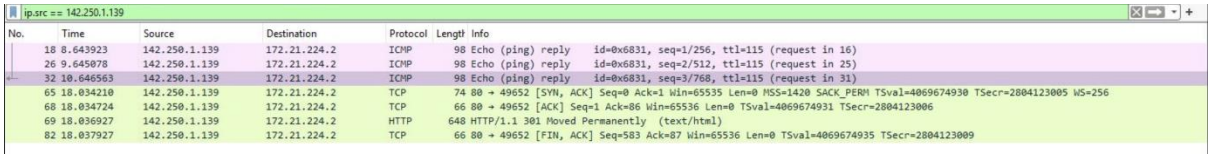
In this task, you'll use filters to analyze specific network packets based on where the packets came from or where they were sent to. You'll explore how to select packets using either their physical Ethernet Media Access Control (MAC) address or their Internet Protocol (IP) address.

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**.

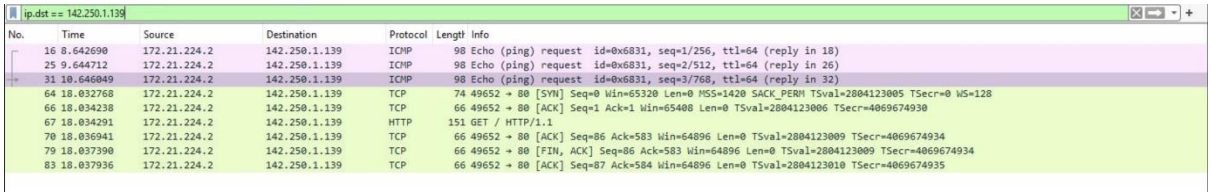


3. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.
4. Enter the following filter to select traffic for a specific destination IP address only:

ip.dst == 142.250.1.139

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

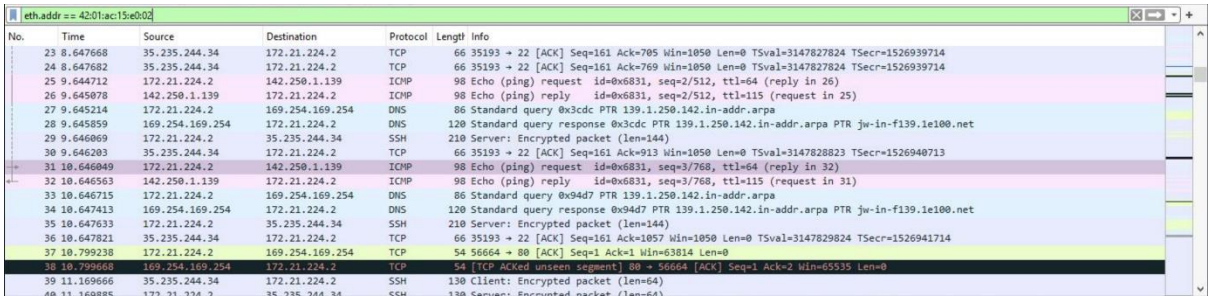
A filtered list is returned that contains only packets that were sent to **142.250.1.139**.



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:

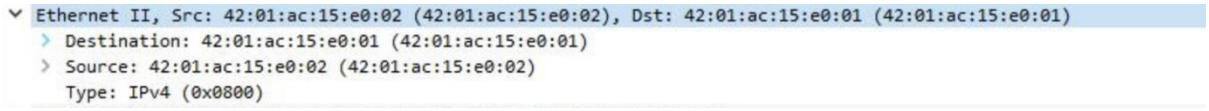
eth.addr == 42:01:ac:15:e0:02

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



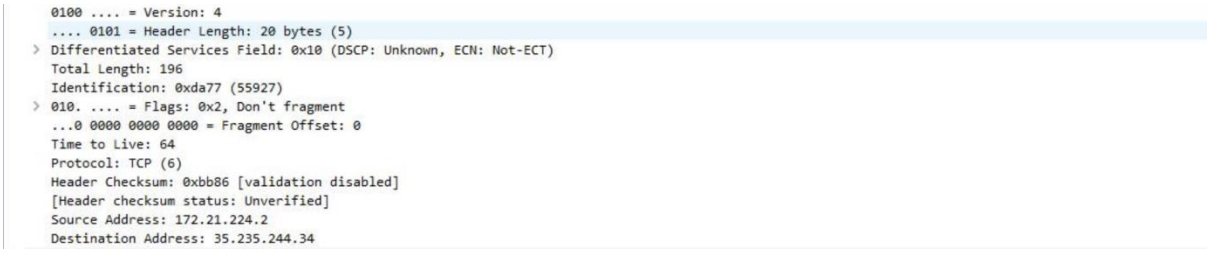
8. Double-click the first packet in the list. You may need to scroll back to display the first packet in the filtered list.
9. Double-click the **Ethernet II** subtree if it is not already open.

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



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.



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.

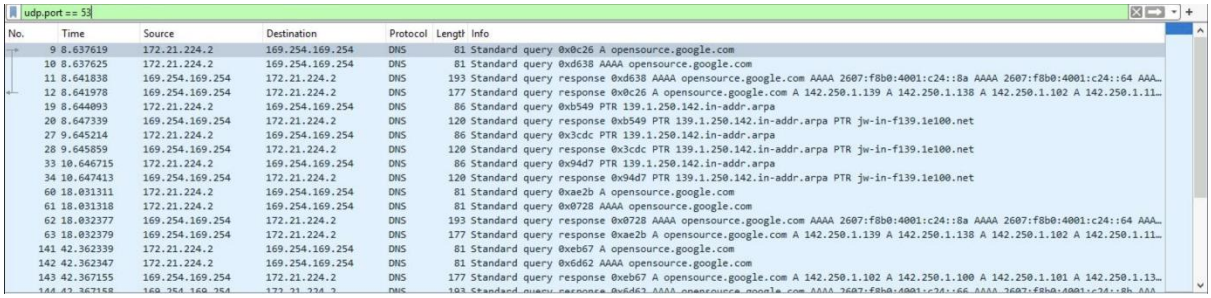
Task 4. Use filters to explore DNS packets

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

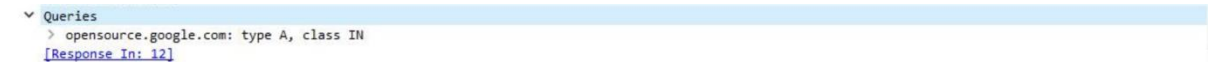
2. Press **ENTER** or click the **Apply display filter** icon in the filter text box.



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.

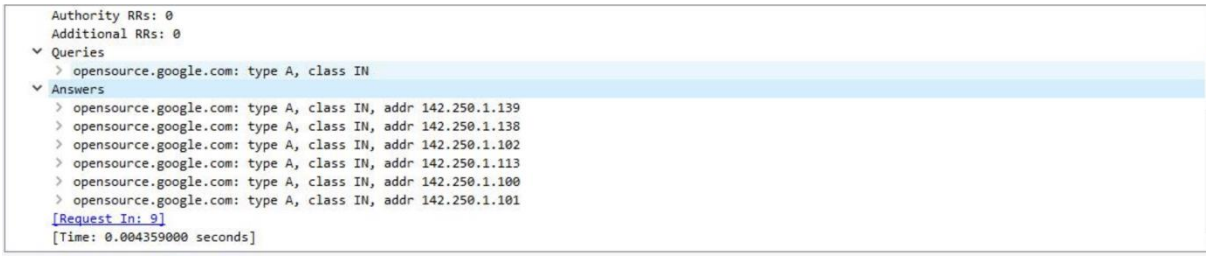
5. Scroll down and double-click **Queries**.

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.



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.

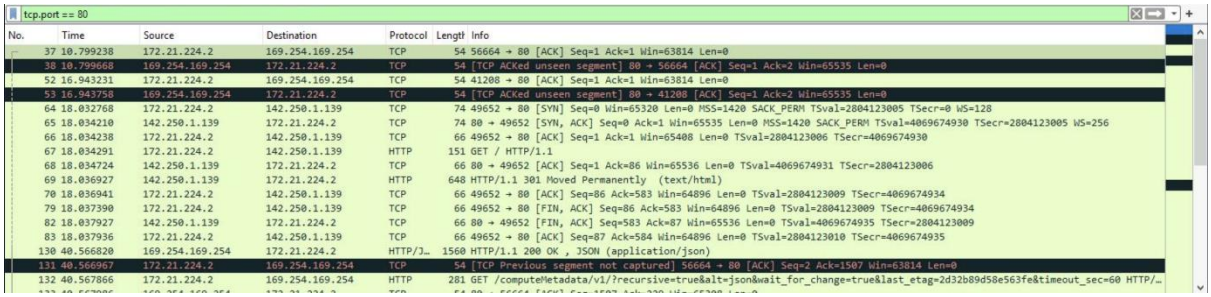
Task 5. Use filters to explore TCP packets

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:

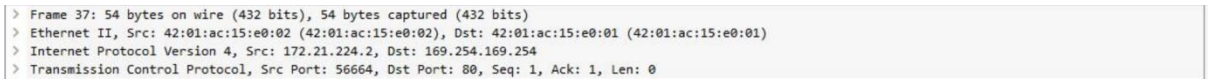
tcp.port == 80

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**.



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.