Exemplar: Analyze your first packet with Wireshark

Lab by Cole Gentry
Level Intermediate
Last updated 2 days ago
4.8 (4,611)

This lab may incorporate AI tools to support your learning.

Activity overview

As a security analyst, you'll need to analyze network traffic in order to learn what type of traffic is being used in your local systems or the network, and for working with

Previously, you learned about packets and protocols. Analyzing captured packets can be a useful way to learn about the network traffic that is being used in your local systems. Wireshark is an open-source tool in Wireshark, which has a graphical user interface (GUI) and offers you a variety of powerful tools for analyzing your investigations. Wireshark's graphical user interface allows you to see packets and protocols in real-time, and it provides you with a variety of filters to analyze packets and filter the data in order to focus on the network traffic that is most relevant to your security investigations.

Now you'll use Wireshark to inspect packet data and apply filters to sort through packet information efficiently.

In this lab activity, you'll use Wireshark to examine a sample packet capture file and filter the network traffic data.

Scenario

In this scenario, you're a security analyst investigating a network.

You'll analyze a network packet capture file that contains traffic data related to a user connecting to an internal site. The ability to filter network traffic using packet filters is a skill that security analysts need to be proficient in.

You have three tasks to complete in this lab:

- Identify the source and destination IP addresses involved in this web browsing session.
- Identify the protocols that are used when the user loads the connection to the website site.
- Analyze some of the data packets to identify the type of information sent and received by the systems that connect to "www.ch" when the network data is captured.

Here's how you'll do this: First, you'll open the packet capture file and explore the Wireshark graphic user interface. Second, you'll apply a detailed view of a single packet and use Wireshark's tools to view protocol and packet information. Third, you'll apply filters to view protocol and packet information. Finally, you'll use the filters to sort through packets, filter and inspect UDP DNS traffic to examine protocol data. Finally, you'll apply filters to TCP packets and inspect the TCP packet data.

You're ready to use Wireshark to inspect network packet data!

Disclaimer: Due optimal performance and compatibility, it is recommended to use either **Google Chrome** or **Mozilla Firefox** browsers when accessing this lab.

Start the lab:

You'll need to start the lab before you can access the materials. To do this, click the green "Start Lab" button at the top of this screen.

Start Lab

After you click the "Start Lab" button, you will see a panel appear below where the start lab button was that has an **Open Workspace** button.

Click the **Open Workspace** button and a new tab will open with a cloud interface for accessing the Wireshark and the packet capture files. Now you're ready to begin!

Note: If the Windows OS becomes stuck and disconnects automatically, you can reconnect to the Windows Kali Linux by:

1. Return to the lab page, and click **Open Workspace** tab.
2. Click the "Reconnect" button.
3. Use the following credentials on the login page and leave the other fields unchanged:

Username	localhost
Hostname	address
Password	Password1!

Task 1. Explore data with Wireshark

In this task, you learn about Wireshark's graphic user interface (GUI) and how you can collect from a system that made web requests to a site. You need to open the data with Wireshark to get an overview of the data that was presented in the application.

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

The packet capture file has the Wireshark packet capture file type, which shows a **shark fin** icon. Wireshark's default file type is the pcapng file extension. The pcapng file extension that is hidden by default by Windows and Linux operating systems.

Note: If **Software Update** dialog box has appeared, notifying you that a new version of Wireshark is available, **Click Skip this version**.

2. Double-click the Wireshark file icon to the **sample.pcap** filename to open the Wireshark interface and capture data.

A lot of unknown packets traffic is listed. Which is the **pcap** filename in the file information needed in an upcoming step.

For now, here's 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:** Provides information about the data in the packet (this payload) as interpreted by Wireshark

Not all the data packets are the same size. Coloring rules are used to provide high-level visual cues to help you quickly identify the different types of data. Once relevant packets are identified, you can use Wireshark's filters to focus on the packets that quickly identify the data that is relevant to you. The example packet list is a group of light blue packets that all contain DNS traffic, followed by a group of purple packets that contain TCP traffic.

3. Scroll down the packet list until a packet is listed where the 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"?

- ARP
- ICMP
- DNS
- TCP

Submit

Answer: ICMP is the protocol type listed for the first (and all) packets that contain "Echo (ping) request" in the info column.

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 == 10.200.7.100
```

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

The list of packets displayed is now significantly reduced and contains only packets that contain the specified IP address. Notice that the packet numbers are not consecutive. Wireshark's default view is to sort **packet** numbers in ascending order, and **Ctrl + Shift + Left** groups the TCP and UDP traffic in a subset of TCP packets.

3. Double-click the first packet from the TCP protocol.

This opens a packet details pane window.

The **packet** contents of the entire conversation process displays in a new window with an analysis of the entire process of the network packet. The lower section of the window contains the raw packet data displayed in hexadecimal and ASCII text. Theses is a useful feature for security analysts who need to see the raw data in order to identify the **payload**.

Note: The details pane is located at the bottom portion of the main Wireshark window. It can also be accessed at a later window by double-clicking a packet.

4. Double-click the first subnet in the upper section. This starts with the word **Frame**.

This provides you with details about the overall network packet, a frame, a network interface, and the protocols contained in the packet. At the top, it provides a summary, listed in the order of the packet.

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

This item contains details about the packet at the Ethernet level. It contains the initial source and destination MAC addresses and the type of Ethernet protocol that the network packet contains.

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

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, the Internet Protocol Version for example, TCP or UDP, and the packet's time to live (TTL) value.

Note: The Internet Protocol Version 4 subnet is a lower Protocol layer of Ethernet. The Ethernet subnet often reflects the protocol.

The source and destination IP addresses shown here match the source and destination IP addresses in the summary display for this packet in the middle window.

7. Double-click **Internet Protocol Version 4** again to collapse that subnet and then double-click the **Transmission Control Protocol** subnet.

This provides details about the Transmission Control Protocol (TCP) packet, including the source and destination TCP ports, the TCP sequence and TCP acknowledgment numbers.

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 top of all the packets in the main Wireshark window.

What is the TCP destination port of this TCP packet?

- 80
- 200
- 49
- 53

Submit

Answer: Port 80 is the TCP destination port for this packet. It contains the initial web request to an HTTP website that will typically be listening on TCP port 80.

8. In the **Transmission Control Protocol** subnet, 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 window.
10. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.

All the packets have returned to the display.

If you are accidentally clicked 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 examine how to select packets using either their attached Ethernet Media Access Control (MAC) address or their Internet Protocol (IP) address.

1. Enter the following filter to select packets that, in a specific source IP address only. Enter this into the **Apply a display filter...** text box immediately above the list of packets:

```
ip.src == 10.200.7.100
```

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

A filtered list is returned with fewer entries than before. It contains only packets that used from **10.200.7.100**.

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 to a specific destination IP address only:

```
ip.dst == 10.200.7.100
```

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

A filtered list is returned that contains only packets that were sent to **10.200.7.100**.

6. Click the **X Clear display filter** icon in the Wireshark filter bar to clear the IP address filter.

Keep in mind that this is also useful to sort or focus specific Ethernet MAC address. This filters traffic related to one MAC address, regardless of the other protocols involved.

```
eth.addr == ac:dc:ac:1d:ab:82
```

8. Press **ENTER** or click the **Apply display filter** icon to the filter text box.
9. Double-click the first packet in the list. You may need to scroll down to display the first packet in the Ethernet II.
10. Double-click the **Ethernet II** subnet to see destination and source.

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

11. Double-click the **Ethernet II** subnet to collapse it.
12. Double-click the **Internet Protocol Version 4** subnet to expand it and scroll down until the **Time to Live** and **Protocol** field appear.

The **Protocol** field in the **Internet Protocol Version 4** subnet indicates which IP Internet protocol is contained in the packet.

What is the protocol contained in the Internet Protocol Version 4 subnet from the first packet listed in MAC address ac:dc:ac:1d:ab:82?

- ICMP
- TCP
- UDP
- ICMP

Submit

Answer: TCP is the internal protocol contained in the first packet from MAC address ac:dc:ac:1d:ab:82.

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 specific DNS traffic, you'll drill down into the protocol to examine how the DNS protocol works. Analyzing DNS traffic is a useful skill for security analysts because it helps to identify IP addresses that are being sent back by a DNS server when a server is successfully resolved.

1. Enter the following filter to select UDP port **53** traffic. DNS traffic uses UDP port 53 to communicate. 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 to 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 **Answers**, which is in the **Domain Name System (query)** subtree.

The Answers data includes the name that each packet (**ipannounce.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 ipannounce.google.com?

- 195.1.200.142
- 100.200.100.234
- 142.200.1.139
- 173.21.224.1

Submit

Answer: The IP address 195.200.1.200 is displayed in the expanded Answers section for the DNS query for **ipannounce.google.com**.

6. Click the **X** icon to close the detailed packet window.
7. 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 explore TCP traffic and TCP packets. Sometimes there is reason for tech that a present is potential radio connected inside network packets. TCP is a connection-based protocol in establishing such a connection, it ensures that the IP datagram is sent.

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 to the filter text box.

Quite a few packets were created when the user accessed the web page http://www.ch.com.

1. Scroll down to the packet's Internet Protocol, look for the **Identification** section of this packet's **IPv4 Header**.

What is the Time to Live value of the packet as specified in the Internet Protocol Version 4 section?

- 7h
- 8h
- 128
- 64

Submit

Answer: The Time to Live value is 64. This property is contained in the Internet Protocol Version 4 subtree, which is the third subtree listed in the detailed packet inspection window.

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

- 60 bytes
- 54 bytes
- 74 bytes
- 40 bytes

Submit

Answer: The Frame Length is 54 bytes. This property is contained in the Frame subtree, which is the first subtree listed in the detailed packet inspection window.

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

- 12 bytes
- 14 bytes
- 20 bytes
- 40 bytes

Submit

Answer: The Header Length is 20 bytes. This property is defined in the Internet Protocol Version 4 subtree, which is the fourth subtree listed in the detailed packet inspection window.

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

- 172.31.136.23
- 142.250.1.106
- 144.254.130.254
- 239.1.255.250

Submit

Answer: The Destination Address is 142.250.1.106. This property is defined in the Internet Protocol Version 4 subtree, which is the third subtree listed in the detailed packet inspection window.

4. Click the **X** button 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 "secret"
```

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

This filter is packet-counting and requires reads with the `-w` command in the sample packet capture file.

Conclusion

Great work!

You now have practical experience using Wireshark to:

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

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

End your 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 Kali shell. You won't be able to access the work you've completed if it ends.
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.