Introduction to Wireshark (GUI)

If you've already taken our Introduction to Network Analysis course then you'll have a good understanding of how packet captures (PCAPs) can be analyzed using Wireshark and TCPDump. If you haven't, don't worry! This lesson will cover how to use Wireshark and is followed by an exercise where you'll be investigating PCAPs to determine malicious activity and collect indicators of compromise.

WIRESHARK GUI

Wireshark is a very popular and free software that is used by many security experts around the world to capture and analyze network traffic in great detail. Wireshark comes with the official Kali Linux distribution, alternatively, it can be downloaded for free at: https://www.wireshark.org/#download. Once you have Wireshark installed, this section will help you familiarise yourself with the GUI.

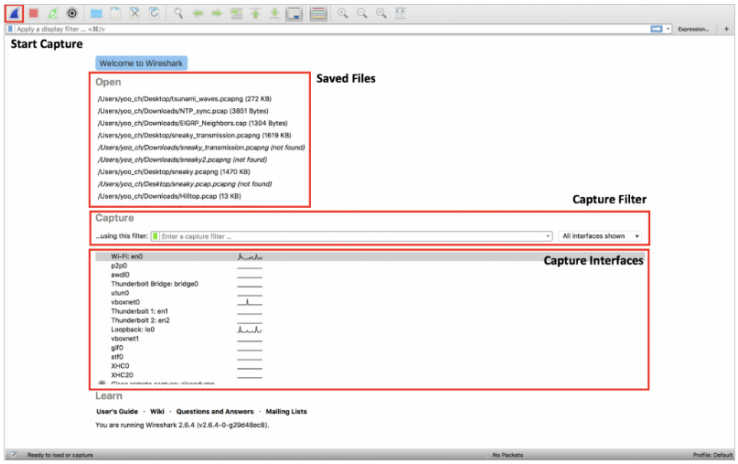
The GUI can be categorized into two screens

- ***Startup Window***, which is displayed when Wireshark is launched.
- ***Main Window***, which is displayed when a capture has been started or loaded.

You do not need to learn all of the settings and options, but knowing the major options will allow you to use Wireshark for network analysis with relative ease.

WIRESHARK STARTUP WINDOW

The Startup Window is the screen that pops up when the user starts Wireshark. It allows the user to start or load a network traffic capture and configure some capture settings.



[1] Start Capture: The blue button in the top left corner starts capturing inbound and outbound packets, with the specified capture filters, on the specified interface.

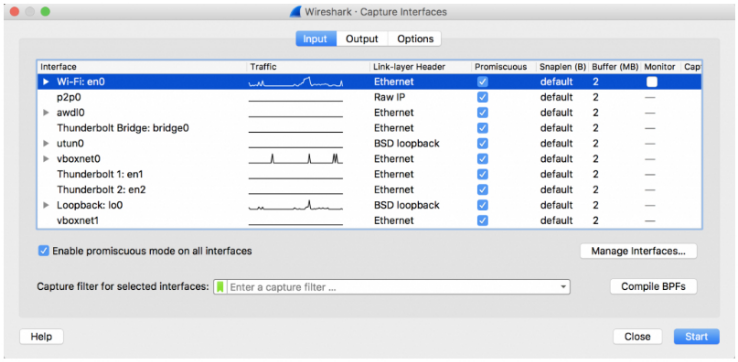
[2] Open Saved Files: Wireshark traffic capture files can be saved in several formats, such as .cap, .pcap or .pcapng, and can be opened in the Main Window for analysis.

and can be opened in the Main Window for analysis.

[3] Capture Filter: You can write expressions in the capture filter to limit the types of packets that Wireshark captures. For example, if you specify the **not arp** capture filter, Wireshark avoid capturing Address Resolution Protocol packets. Details on how to construct capture filters are described in the next section. Capture filters can be saved for reuse at a later time.

[4] Capture Interface Selection: Wireshark lists all available interfaces that it can capture on, with a graph of the recent network activity on each of those interfaces. You can select an interface that you want to capture traffic on, such as en0 for Wi-Fi traffic and vboxnet0 for virtual network traffic, in the above image.

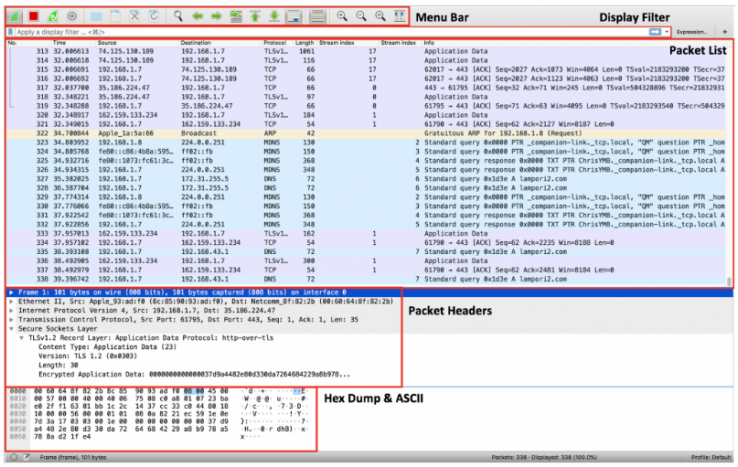
It is recommended that promiscuous mode be turned on for capturing interfaces. Promiscuous mode allows Wireshark to capture packets that are received on an interface but not actually addressed to the host, for example, frames transmitted on a wireless network with different MAC addresses. This allows Wireshark to capture other hosts' traffic and have a broader picture of the network.



Promiscuous mode can be managed by clicking on the cog-shaped button in the top menu bar, and toggling the setting for a specific interface or for all interfaces.

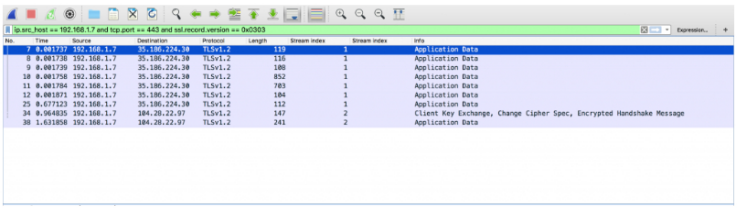
WIRESHARK MAIN WINDOW

The Main Window is where all of the capturing and analysis happens. There are dozens upon dozens of options, parameters, statistics and detailed information available on the traffic being captured. The user can view the network traffic, from the individual bytes of a single packet, to a statistical overview of protocols within the capture.



[1] Menu Bar: The menu bar located at the top of the window is used to manage the capture. In the far left section, you can start, stop and restart the capture, and manage capture interface settings. In the next section, you can open, save and close the capture file. The magnifying glass icon is used to find a specific packet using a display filter or by a string or bytes within the packet.

[2] Display Filter: The display filter is used to display only specified packets. You can construct an expression by specifying header fields and optionally, the values that they should match. Logical operators can also be used to chain expressions. If a packet contains the specified header field, or if the header field has a value that is specified, the packet will be shown in the packet list – otherwise, it will not be shown.

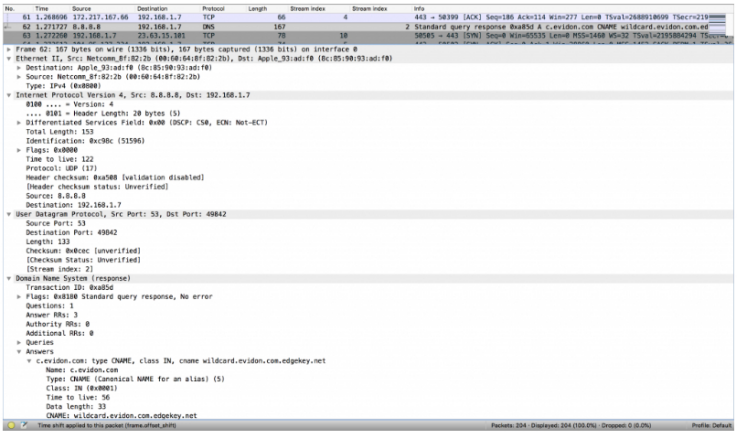


For example, in the capture above, a display filter of `ip.src_host == 192.168.1.7 and tcp.port == 443 and ssl.record.version == 0x0303` has been applied. The first statement matches packets with a source IP address of 192.168.1.7. The second statement matches packets with a source or destination TCP port 443 (SSL/TLS). The third statement matches packets that are using TLS version 1.2. The three statements are conjoined by the '**and**' logical operator, which means a packet must satisfy all three of those statements for them to be displayed in the packet list. A more in-depth look at the display filter is described in upcoming sections.

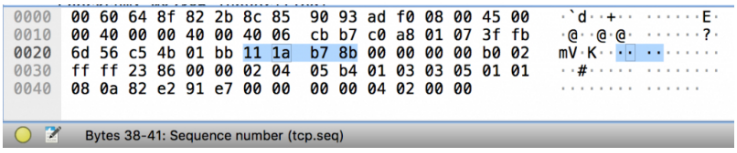
[3] Panes: The Wireshark Main Window has three main panes: packet list, packet headers and the hex dump and ASCII representation of the packet bytes.

[4] Packet List: The packet list aggregates major information on the packets that Wireshark captures, in columns. Generally, the packet list should display the packet number (the later the packet was captured, the higher this number is), time since the start of capture, the source and destination IP addresses, the protocol, the packet length and a summary of the packet headers or contents. You can easily get a picture of the network flow and protocol conversations being captured.

[5] Packet Headers: The packet headers section provides a wealth of information on each individual packet, and organizes packet header fields and values in layers of easy-to-view drop-down menus – from Layer 1 frame information to Layer 7 protocol contents.



The above image shows the sheer amount of detailed information Wireshark organizes and displays on a single DNS query response. You can see the source and destination MAC & IP addresses and UDP ports, DNS flags and DNS query response answers in full detail.



– **Hex Dump & ASCII**: On the bottom pane, you can see the hexadecimal and ASCII representation of the entire packet.

Hovering over the hex dump or ASCII highlights a section of the packet and displays what field of the packet is being highlighted on the bottom bar. In the above image, the TCP sequence number has been highlighted. You may also notice that the expression for the field is also displayed – in this case, **tcp.seq**. This is especially helpful when you are constructing display filters and do not know the specific Wireshark term for specifying a certain packet field.

This section covered the basic GUI fields in Wireshark's Startup & Main windows and what each one of them does. In the following sections, with the basic knowledge of the Wireshark GUI at hand, we will take a look at capturing live network traffic and analyzing .pcap capture files.

Previous Topic

Mark Complete

Next Topic

Back to Lesson