**Task-5**

**Capture And Analyze Network Traffic Using Wireshark**

INTRODUCTION:-

Wireshark, a tool used for creating and analyzing PCAPs (network packet capture files), is commonly used as one of the best packet analysis tools. In this room, we will look at the basics of installing Wireshark and using it to perform basic packet analysis and take a deep look at each common networking protocol.PCAPs used in this room have been sourced from the Wireshark Sample Captures Page as well as captures from various members of the community. All credit goes to the respective owners.Before completing this room we recommend completing the 'Introductory Networking'. If you have a general knowledge of networking basics then you will be ready to begin.

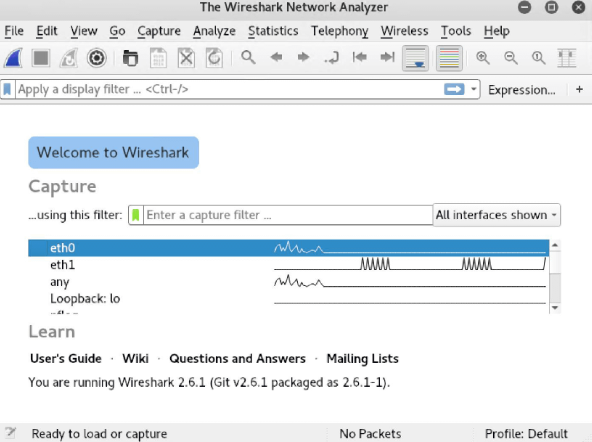
INSTALLATION:-

The installation for Wireshark is very easy and typically comes with a packaged GUI wizard. Luckily if you're using Kali Linux then it is already installed on your machine. Wireshark can run on Windows, macOS, and Linux. To begin installing Wireshark on a Windows or macOS device you will need to first grab an installer from the Wireshark website. Once you have downloaded an installer, simply run it and follow the GUI wizard.If you are using Linux you can install Wireshark with apt-get install wireshark or a similar package manager.

WIRESHARK OVERVIEW:-

The first screen that we are greeted by when opening Wireshark is the main page that will allow us to specify our interface(s) as well as apply filters to narrow down traffic that we are capturing.

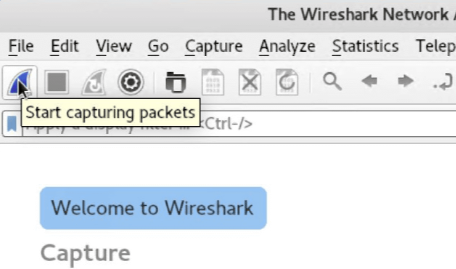
Here you can see that I have multiple interfaces to filter from you may have more or fewer interfaces than I have. From here we can choose whether we want to perform a live capture on our interface(s) or load a PCAP for analysis.

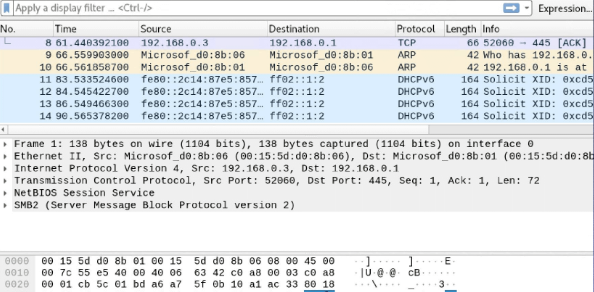


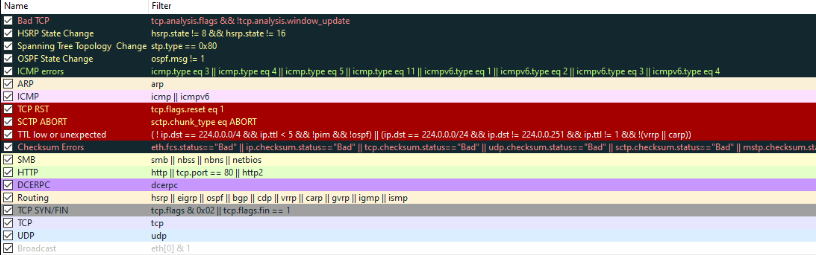
LIVE PACKET CAPTURES:-

If we begin by navigating to the green ribbon in Wireshark and select Manage Capture Filters we can view a list of available filters.

You do not have to select a filter, it will only help to bring down the number of packets being brought in and organize the capture. This is only a brief introduction to filters for more information about filters go to Task12 or go to the Wireshark Website.

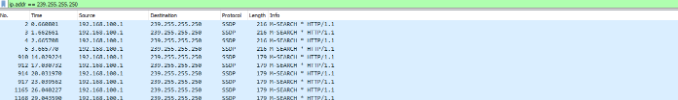
Once you have any capture filters you want selected, you can begin a capture on an interface by double-clicking the interface or by right-clicking and navigating to Start Capture.Depending on the network activity you may see no packets coming in or you may see packets streaming in very quickly.Once you're done gathering the packets you need or want, you can click the red square to stop capturing, and then you can begin your analysis.

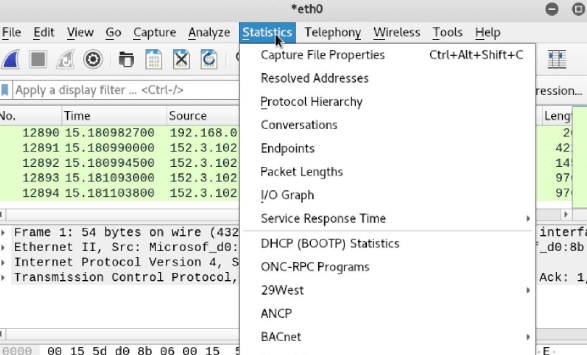




FILTERING CAPTURE:- Packet Filtering is a very important part of packet analysis especially when you have a very large number of packet sometimes even 100,000 plus. In task 3 capture filters were briefly covered however there is a second type of filter that is often thought of as more powerful and easier to use. This second method is known as display filters, you can apply display filters in two ways: through the analyze tab and at the filter bar at the top of the packet capture.

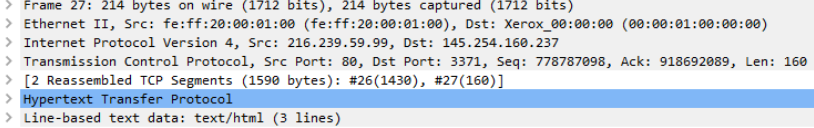
BASIC FILTERING:-

Filtering gives us a very large scope of what we can do with the packets, because of this there can be a lot of different filtering syntax options. We will only be covering the very basics in this room such as filtering by IP, protocol, etc. for more information on filtering check out the Wireshark filtering documentation.There is a general syntax to the filter commands however they can be a little silly at times. The basic syntax of Wireshark filters is some kind of service or protocol like ip or tcp, followed by a dot then whatever is being filtered for example an address, MAC, SRC, protocol, etc.Filtering by IP: The first filter we will look at is ip.addr, this filter will allow you to comb through the traffic and only see packets with a specific IP address contained in those packets, whether it be from the source or destination.Syntax: ip.addr == <IP Address>

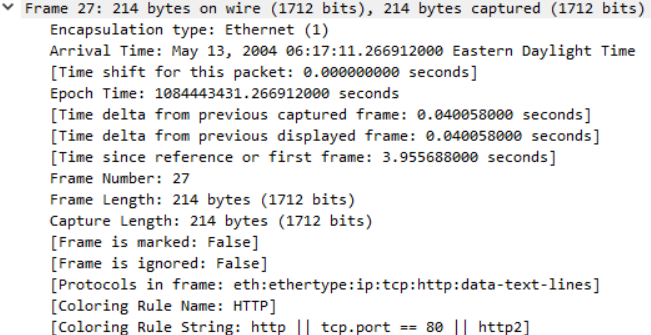
****

PACKET DETAILl:-

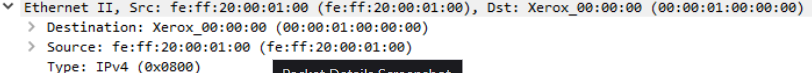
You can double click on a packet in capture to open its details. Packets consist of 5 to 7 layers based on the OSI model. We will go over all of them in an HTTP packet from a sample capture.

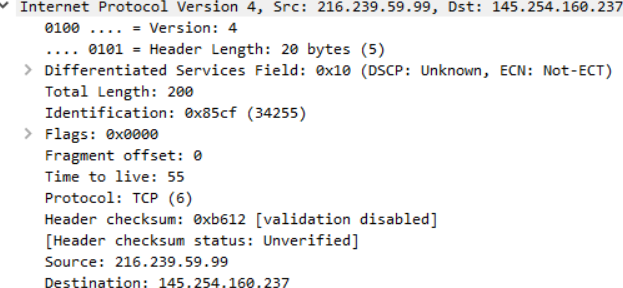


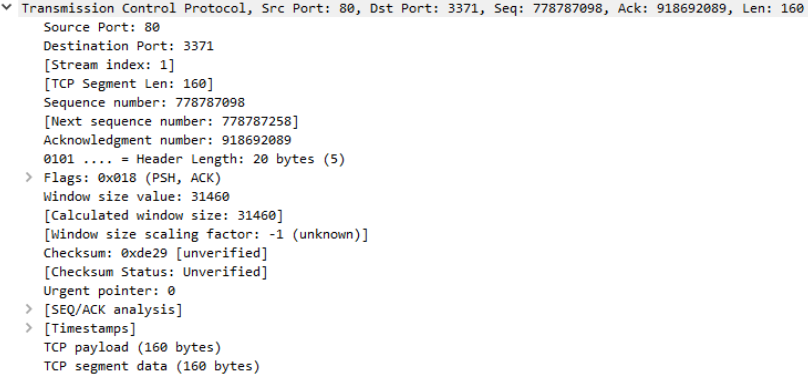
Looking above we can see 7 distinct layers to the packet: frame/packet, source [MAC], source [IP], protocol, protocol errors, application protocol, and application data. Below we will go over the layers in more detail.

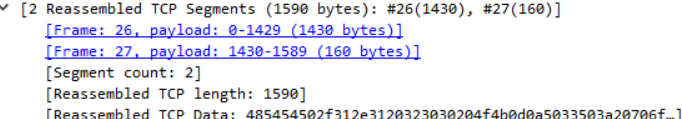
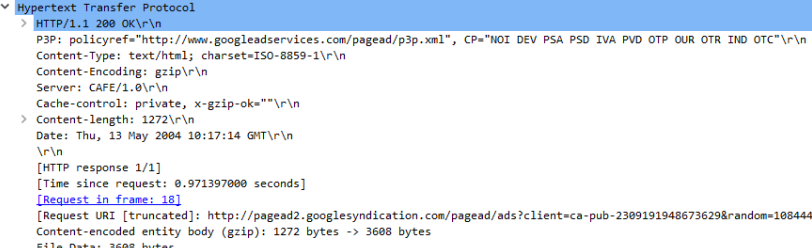
Frame (Layer 1) -- This will show you what frame / packet you are looking at as well as details specific to the Physical layer of the OSI model.

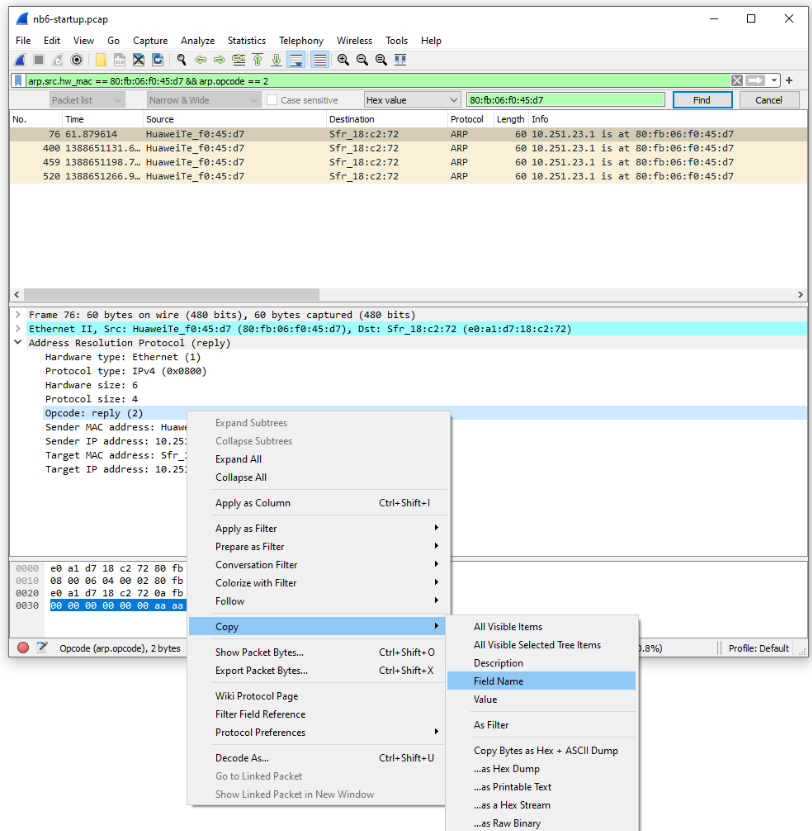
Source [MAC] (Layer 2) -- This will show you the source and destination MAC Addresses; from the Data Link layer of the OSI model.



Source [IP] (Layer 3) -- This will show you the source and destination IPv4 Addresses; from the Network layer of the OSI model.

Protocol (Layer 4) -- This will show you details of the protocol used (UDP/TCP) along with source and destination ports; from the Transport layer of the OSI model.

Protocol Errors -- This is a continuation of the 4th layer showing specific segments from TCP that needed to be reassembled.Application Protocol (Layer 5) -- This will show details specific to the protocol being used such HTTP, FTP, SMB, etc. From the Application layer of the OSI model.



CONCLUSION:- Want to learn more? There are multiple courses and certifications that take a deep dive into Wireshark. The first free resource that I would recommend checking out is the Wireshark online documentation. It is very detailed and can help you understand all the nuances that come with learning Wireshark. Along with the written docs they also provide some videos to help you out along the way.