

Project report on IP packet analysis using **Wireshark**

July 9th, 2018



Instructor:-Sir Noor Alam
Cyber Security and Ethical Hacking
Azure Skynet

By:-Adrian Clive Prasad
Email:wrathlustpride@gmail.com
Location:Bangalore

ACKNOWLEDGEMENT:

I would like to express our profound gratitude to our project guide Mr Noor Alam Sir and Azure Skynet Solutions Company for their support, encouragement, supervision, and useful suggestions throughout this project. Their moral support and continuous guidance enabled us to complete our work successfully

Introduction

Wireshark is a network protocol analyser, formerly released under the name Ethereal. As a result of certain copyright restrictions, when the primary developer left his former **company**, Ethereal changed its name to Wireshark, but remains the same program and has many of the same core developers that worked on Ethereal. This program is able to intercept packets transmitted over the network and compile statistics about network usage, allow the user to view content that is being accessed by other network users, and store usage information for offline access.

Wireshark has built-in color-coding features that help the user to identify particular types of network traffic, such as DNS in blue and HTTP in green. Most of the information displayed in the figure can be used to set up sorting filters, simplifying the process of analysing data.

Filters can often be set up to cover anything from protocol type to source or destination address, and even to focus on packets that lack certain data. The versatility of these filters makes sorting through the data much simpler, but the process still requires a keen understanding of what information is displayed and how to interpret it.

Wireshark is an open-source program, with an active support and development community, and held its fourth Annual Developer and User Conference in June 2011. With the support of this community, Wireshark has expanded over the years to offer support on hundreds of network protocols, with more being added all the time. As a result, Wireshark has established itself as the standard among commercial and educational institutions for network analysis.

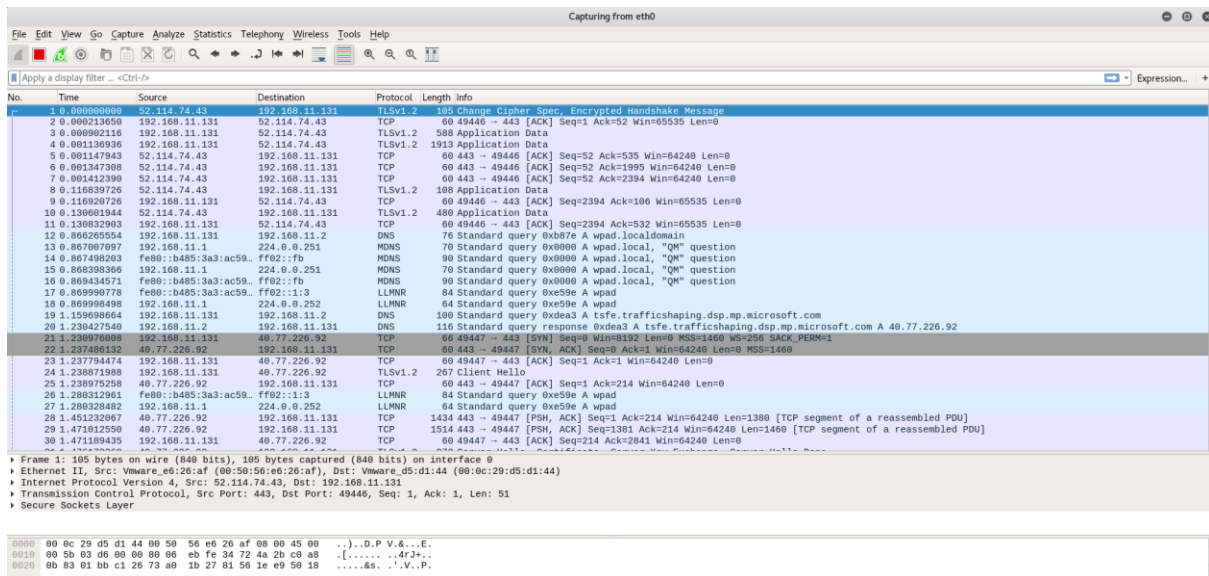


Figure showing Wireshark capturing the packets

Wireshark software has been developed to work on Microsoft Windows, Linux, Solaris, and Mac OS X. Support for all these major operating systems has further increased the market strength of Wireshark.

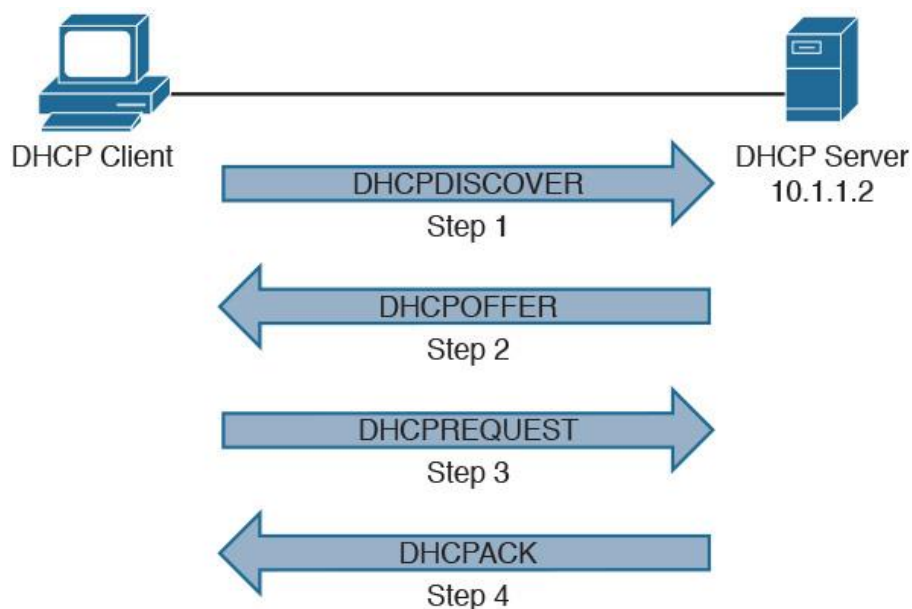
Wireshark can be used to view packet information obtained by many other packet capture programs.

Wireshark is very similar to tcpdump, but has a graphical front-end, plus some integrated sorting and filtering options.

Wireshark lets the user put network interface controllers into promiscuous mode (if supported by the network interface controller), so they can see all the traffic visible on that interface including unicast traffic not sent to that network interface controller's MAC address. However, when capturing with a packet analyzer in promiscuous mode on a port on a network switch, not all traffic through the switch is necessarily sent to the port where the capture is done, so capturing in promiscuous mode is not necessarily sufficient to see all network traffic.

DHCP Handshake:

DHCP is a client-server protocol in which servers manage a pool of unique IP addresses, as well as information about client configuration parameters, and assign addresses out of those address pools. DHCP-enabled clients send a request to the DHCP server whenever they connect to a network.



DNS Protocol :

DNS stands for Domain Name System. The main function of DNS is to translate domain names into IP Addresses, which computers can understand. It also provides a list of mail servers which accept Emails for each domain name. Each domain name in DNS will nominate a set of name servers to be authoritative for its DNS records. This is where all other name servers will be pointed when looking for information about the domain name. Name servers are a program or computer server that implements a name-service protocol.

Requirements:

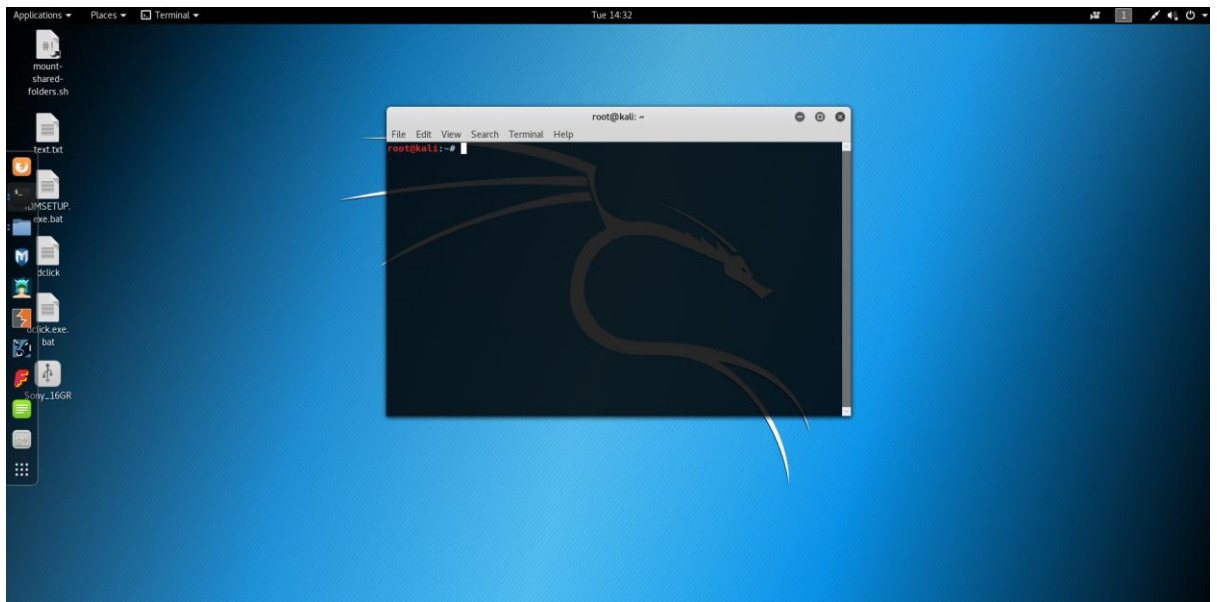
- A system using any Operating system
(in this project Kali Linux and its the related commands are used)
- A target system (preferably within the same LAN connection as the host)
- Wireshark software on the host system
- Nmap in host system to use scanning and generate desired signal packets

Since Kali is the host operating system, there is no need of installing Nmap and Wireshark because these softwares come pre-loaded in the Kali operating System.

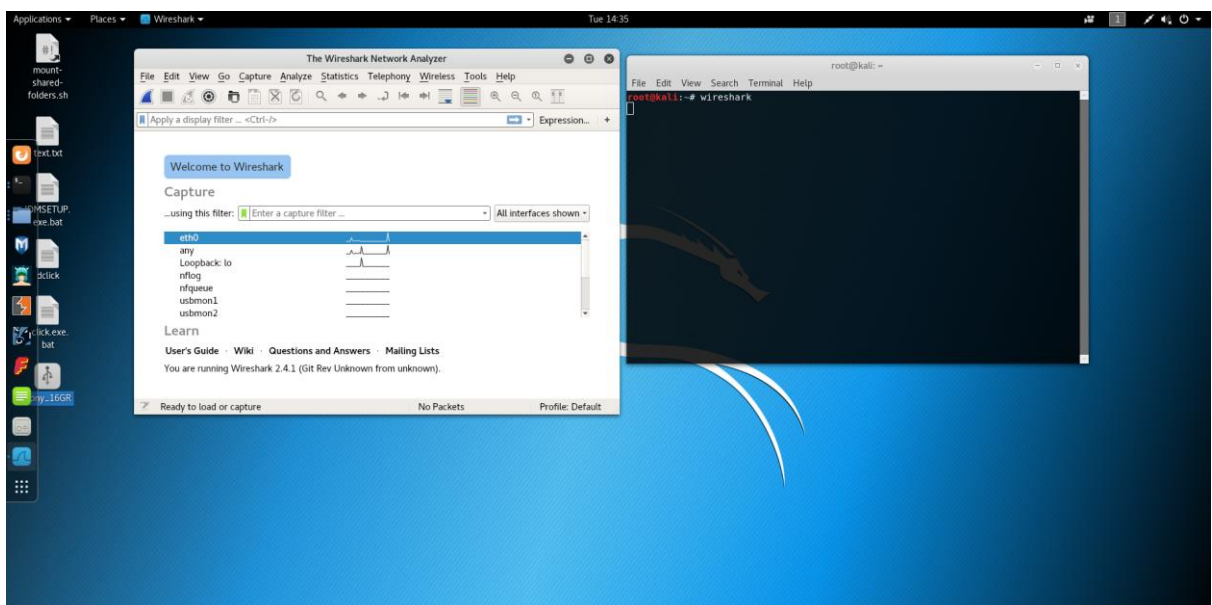
Using Wireshark to capture Wireless Packets

To Start the process follow the steps

1. Boot up the host operating system(kali linux in this case)
2. Open a terminal.
3. Type “ifconfig “ in the terminal to get the host’s I.P address.



4. Enter the command “wireshark” in the terminal (Wireshark application starts running).



5. Use the application to monitor packets.

WORKING:

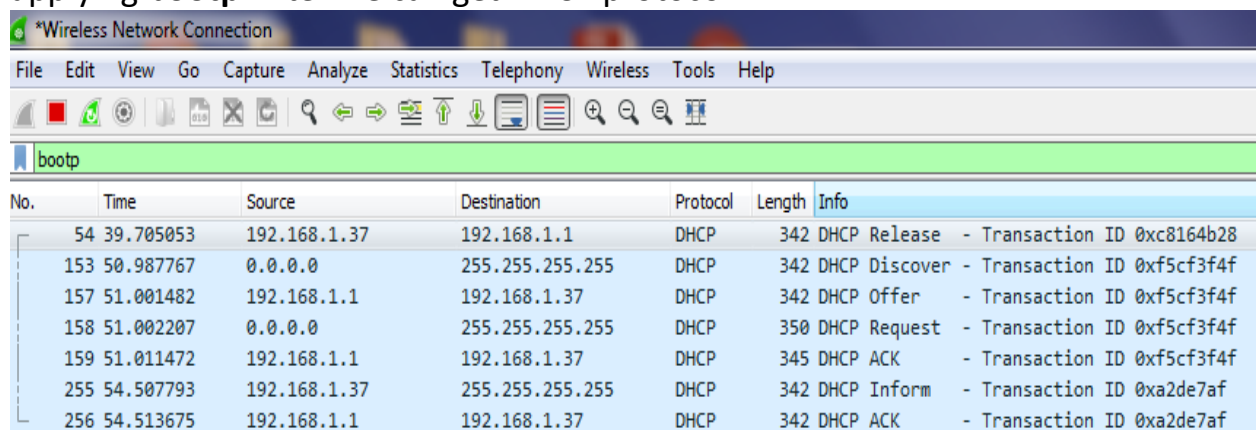
Connect WIFI to the laptop and open WireShark software . The Software starts collecting the traffic .

DHCP PROTOCOL:

DHCP Protocol provides IP address , so to get IP address from DHCP server , release the IP address and renew it using command prompt with the following commands.

1. Pc>ipconfig/release
2. Pc>ipconfig/renew

In this process the Wireshark collects the traffic due to IP address renewal , by applying **bootp** filter we can get DHCP protocol .



The image shows a Wireshark packet capture window titled '*Wireless Network Connection'. The filter bar at the top contains the text 'bootp'. Below the filter bar, a list of captured packets is displayed. The packets are DHCP messages between a client (192.168.1.37) and a server (192.168.1.1). The packets include a Release, Discover, Offer, Request, ACK, Inform, and another ACK. The 'Info' column shows the transaction ID for each message.

No.	Time	Source	Destination	Protocol	Length	Info
54	39.705053	192.168.1.37	192.168.1.1	DHCP	342	DHCP Release - Transaction ID 0xc8164b28
153	50.987767	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0xf5cf3f4f
157	51.001482	192.168.1.1	192.168.1.37	DHCP	342	DHCP Offer - Transaction ID 0xf5cf3f4f
158	51.002207	0.0.0.0	255.255.255.255	DHCP	350	DHCP Request - Transaction ID 0xf5cf3f4f
159	51.011472	192.168.1.1	192.168.1.37	DHCP	345	DHCP ACK - Transaction ID 0xf5cf3f4f
255	54.507793	192.168.1.37	255.255.255.255	DHCP	342	DHCP Inform - Transaction ID 0xa2de7af
256	54.513675	192.168.1.1	192.168.1.37	DHCP	342	DHCP ACK - Transaction ID 0xa2de7af

In the above figure:

IP address of PC is 192.168.1.37

IP address of DHCP Server is 192.168.1.1

The DHCP protocol can be clearly visualized with the following steps :

- DHCP Discover: DHCP client sends a DHCP Discover broadcast on the network for finding a DHCP server. If there is no respond from a DHCP server, the client assigns itself an Automatic Private IPv4 address (APIPA).
- DHCP Offer: DHCP servers on a network that receive a DHCP Discover message respond with a DHCP Offer message, which offers the client an IPv4 address lease.

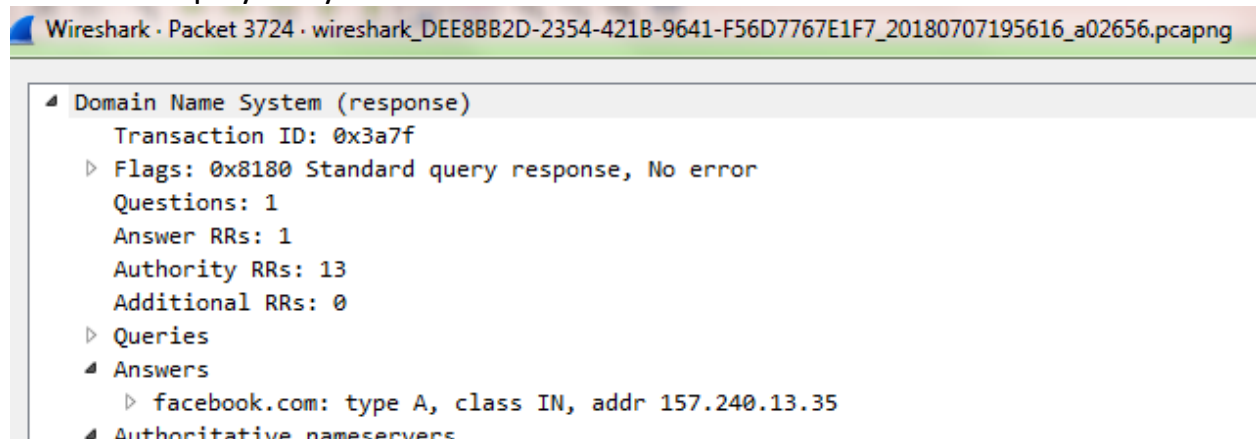
- DHCP Request: Clients accept the first offer received by broadcasting a DHCP Request message for the offered IPv4 address.
- DHCP Acknowledgement: The server accepts the request by sending the client a DHCP Acknowledgment message.

DNS PROTOCOL:

Open any website using any Web Browser . The wireshark collects the traffic of this process . By apply **dns** filter we can get the IP address of the website opened.

Let the website opened be FACEBOOK.

The data displayed by the filter is as shown below



Here the IP address of facebook is 157.240.13.35

The process that is followed to get the IP address is as follows:

- Step 1:** Request information.
- Step 2:** Ask the recursive **DNS** servers.
- Step 3:** Ask the root nameservers.
- Step 4:** Ask the TLD nameservers.
- Step 5:** Ask the authoritative **DNS** servers.
- Step 6:** Retrieve the record.
- Step 7:** Receive the answer.

SSL PROTOCOL:

Open any website that uses SSL protocol and then SSL filter in the Wireshark to get the process happening .

No.	Time	Source	Destination	Protocol	Length	Info
57398	4654.886041	192.168.1.37	136.147.103.128	TLSv1.2	699	Application Data
57400	4655.157479	136.147.103.128	192.168.1.37	TLSv1.2	581	Application Data
57401	4655.157798	136.147.103.128	192.168.1.37	TLSv1.2	85	Encrypted Alert
57475	4705.471930	136.147.103.128	192.168.1.37	SSL	61	Continuation Data
57479	4705.480361	192.168.1.37	136.147.103.128	TLSv1	571	Client Hello
57482	4705.748991	136.147.103.128	192.168.1.37	TLSv1.2	152	Server Hello, Change Cipher Spec
57483	4705.749334	136.147.103.128	192.168.1.37	TLSv1.2	99	Encrypted Handshake Message
57485	4705.750378	192.168.1.37	136.147.103.128	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
57486	4705.773716	192.168.1.37	136.147.103.128	TLSv1.2	699	Application Data

1. **Client Hello:** Information that the server needs to communicate with the client using SSL. This includes the SSL version number, cipher settings, session-specific data.

2. .Server Hello:

Information that the server needs to communicate with the client using SSL. This includes the SSL version number, cipher settings, session-specific data.

3. Encryption with Session Key.

4. Authentication and Pre-Master Secret.

TCP PROTOCOL:

Open any Website and apply TCP filter in the Wireshark.

No.	Time	Source	Destination	Protocol	Length	Info
1635	804.241535	192.168.1.37	103.102.166.224	TLSv1.2	571	Client Hello
1636	804.246689	103.102.166.224	192.168.1.37	TCP	66	443 → 49703 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=512
1637	804.246846	192.168.1.37	103.102.166.224	TCP	54	49703 → 443 [ACK] Seq=1 Ack=1 Win=17408 Len=0
1638	804.498728	103.102.166.224	192.168.1.37	TCP	54	443 → 49702 [ACK] Seq=1 Ack=518 Win=30720 Len=0
1639	804.498987	103.102.166.224	192.168.1.37	TLSv1.2	200	Server Hello, Change Cipher Spec, Encrypted Handshake Message
1640	804.501745	192.168.1.37	103.102.166.224	TLSv1.2	97	Change Cipher Spec, Encrypted Handshake Message
1641	804.516638	192.168.1.37	103.102.166.224	TLSv1.2	139	Application Data

Steps followed in TCP Protocol:

Host A sends a TCP SYNchronize packet to Host B

Host B receives A's SYN

Host B sends a SYNchronize-ACKnowledgement

Host A receives B's SYN-ACK

Host A sends ACKnowledge
Host B receives ACK.
TCP socket connection is ESTABLISHED.

APPLICATIONS:

1. Whenever there is a network related problem, we need some basic clues so that we can start addressing the problems. We will be able to get these clues from the packet sniffer software that is installed in the network. Therefore, packet sniffer software will not only help us monitor the network, but it will also help us analyze the network traffic so that we can identify any problem that crops up in the shortest time possible.
2. If there are any unauthorized intrusions, we will be able to detect the intrusions in good time. This will help us protect our network from the hackers.
3. We will be able to monitor the usage levels of the network at any given time. This will help us optimize the usage if we need to.
4. Using packet sniffer software we can keep a tab on each user in the network and gather sensitive information including passwords.

RESULT: The incoming and outgoing data is successfully monitored and sniffed and gives an idea of the followed protocol and the various handshake sequences

