Ministry of Higher Education and Scientific Research

University Kasdi Merbah Ouargla

Faculty The New Technologies of Information and Communication
Department of Computer Science and Information Technologies

Memory Academic Master

Domain: Computer and Information Technology

Track: Computer Science

Specialty: Fundamental Computer Science

Implementation of an application for detect and protection against MitM attack

Presented by :

Gherbi Borhan eddin
Beggari Ahmed said

supervised by :

Mr.Khaldi Amin

Session 2018/2019

Abstract

Information security has become a very important and indispensable element in people's lives and in work of different institutions, and it is now necessary to provide protection against various threats. Among these well known threats is a kind of attack called the man in the middle attack which it of the most dangerous attacks. And used by hacker in local networks allows the attacker to put himself between users and the router where it can intercept traffic and spy on various communications and control it,redirect, and steal sensitive information, etc ..

The goal of our project is to achieve an application that enables us to protect against these attacks so that detect the source, type of attack and cancel it if possible.

Key-words :

MitM, ARP, DNS, DHCP, Spoofing, Poisoning, Detection, Attack, Security

Table of content

List of Figures

chapter 1 : Mechanisms of attack on computer systems

List of table

General Introduction

Today’s technology has become an essential element in the world. Scientists have built the Internet network in order to transfer information and exchange resources, and then it has evolved into a global network and expanded to include most areas. It is used in commercial matters, scientific research and the source of information and more. The Internet consists of a complex and distributed networks around the world that connect a variety of devices such as computers, routers and switches connected to each other via wires, Through wired and wireless media, processed and used a lot of programs.

The use of the Internet in several domains such as financial resources led to the emergence risks and therefore the emergence of Cyber criminals. Among these security weaknesses there are attacks called man in the middle attack (MITM), most of which occur in the local networks enables the attacker to spy on communications between normal user and router (default Gateway) and inject payloads stealing,break manipulation of the network and redirect communications.

An attacker can exploit a vulnerability that affects a ARP protocol that enables an attacker to make all victim's traffic pass through him and steal the username and passwords that may be affiliated with a bank,There is a high probability of skipping ssl/Tls encryption. An attacker could also exploit a vulnerability of DNS protocol where attacker able to redirect communications to malicious sites. Also, an attacker can sink the router with requests and then build a malicious DHCP server that forces the victims to connect to their server and then redirect traffic to main router. A network scan is a pre-attack process that allows the attacker to enumerate devices in the network. With the emergence of these risks has made the security aspect has become very important in the information technology is indispensable to most customers and services.

In our project we will realize an application that will detect the attacks of a man in the middle and defend against them if possible, the application work in dedicated system in promiscuous mode based in network-IDS and host-IDS,where we will develop a set of functions based on the principle of monitoring the connection to the technology of packets sniffing.

In the first method we will create an algorithm that reveals spoofing ARP packets with the source MAC address of the attacker and the target victim In addition to detecting network scanning. A second method would be detecting a suspicious movement. We will analyze the DNS packets by DNS response if they are harmful or safe. The third method we will achieve is an algorithm that detects DHCP attack.

Our work will extend over three chapters respectively titled and detailed as follows:

The first chapter presents the types of known attacks in cyberspace and types of hacker in addition to some offensive techniques and famous vulnerabilities, At the end of the chapter we will talk about the vulnerabilities (ARP,DHCP,DNS) protocols and how they occur, which causes dangerous attacks such ARP cache Poisoning and DNS Spoofing, DHCP Spoofing.

Then we will discuss in the second chapter the concepts of information security and identification methods of data protection and identify the infrastructure of network protocols DNS,DHCP and HTTPS, and how to work in the reception and transmission data.

In the last chapter, we will provide our approach in order to enforce the protection of communications in the local network by applying mechanisms to detect and prevent MITM attacks.

I Chapter 1 :

Mechanisms of attack on computer systems

I.1 -Introduction

Any computer connected to a computer network is likely to be vulnerable to attack. an attack is to exploit an error in the computer system (operating system, program or even user) for unknown purposes of the system operator and are generally harmful. Internet attacks occur continuously, with multiple attacks per minute on each connected device. Most of these attacks are automatically launched from Infected devices (by viruses, Trojans, worms, etc.), without the owner's knowledge.

I.2 -Type of hackers

People think hackers are negative people in the online world, but we all know there's more to the story. Just like there are good and bad guys in the real world with different shades of their personality, the types of hackers vary by their agenda, methodologies and skill practice .In this post, I'll introduce the well-known and lesser-known kinds of hackers you should know. [1]

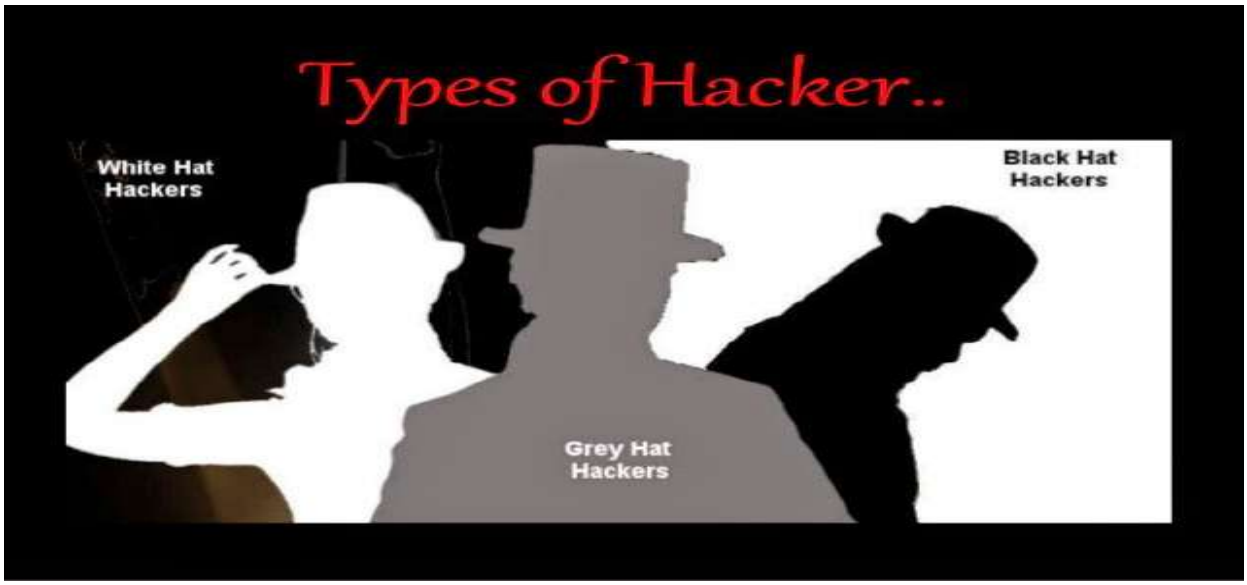

Figure 1-1 : Famous three types of hacker

I.2.1 -Black hat hackers

Taking credit for the negative persona around "hacking," these guys are your culprits. While their agenda may be monetary most of the time, it's not always just that. These hackers look for vulnerabilities in individual PCs, organizations and bank systems. Using any loopholes they may find, they can hack into your network and get access to your personal, business and financial information.

I.2.2 -White hat hackers

Meet the right guys on the dark web. White hat hackers, also known as ethical hackers are the Cybersecurity experts who help the Govt and organizations by performing penetration testing and identifying loopholes in their Cybersecurity. They even do other methodologies and ensure protection from black hat hackers and other malicious Cyber crimes. these are the right people who are on your side. They will hack into your system with the good intention of finding vulnerabilities and help you remove virus and malware from your system.

I.2.3 -Gray hat hackers

Gray hat hackers fall somewhere in between white hat and black hat hackers. While they may not use their skills for personal gain, they can, however, have both good and bad intentions. For instance, a hacker who hacks into an organization and finds some vulnerability may leak it over the Internet or inform the organization about it. It all depends upon the hacker. Nevertheless, as soon as hackers use their hacking skills for personal gain they become black hat hackers. There is a fine line between these two.

I.3 -Type of attack

Computer systems implement different components, ranging from electricity to power machines to software running via the operating system and using the network. Attacks can occur at each link of this chain, if there is an exploitable vulnerability [1]. Hackers use several attack techniques. These attacks can be grouped into three different families :

I.3.1 -Direct attack

It's the simplest of attacks,The hacker attacks his victim from his computer,Most script kiddies use this technique. Indeed, the hack programs they use are only weakly configurable, and many of these programs send packets directly to the victim. If you are attacked in this way, there is a good chance that you can trace back to the origin of the attack, identifying at the same time the identity of the attacker.

I.3.2 -Indirect attacks by rebound

The principle itself is simple, the attack packets are sent to the intermediate computer, which passes the attack to the victim. Hence the term rebound, This attack is very popular with hackers. Indeed, the rebound has two advantages :

- Hide the identity (the IP address) of the hacker.

- Possibly, use the resources of the intermediate computer because it is more powerful (CPU,bandwidth,..) to attack.

I.3.3 -Indirect attacks by response

This attack is a derivative of the rebound attack,It offers the same benefits, from the point of view of the hacker. But instead of sending an attack to the intermediate computer for it to echo, the attacker will send him a request,And it is this response to the request that will be sent to the victim.

I.4 -Techniques of attack

I.4.1 -Denial-of-Service attack (DoS attack)

A denial-of-service (DoS) attack occurs when legitimate users are unable to access information systems, devices, or other network resources due to the actions of a malicious Cyber threat actor. Services affected may include email, websites, online accounts (e.g., banking), or other services that rely on the affected computer or network. A denial-of-service is accomplished by flooding the targeted host or network with traffic until the target cannot respond or simply crashes, preventing access for legitimate users. DoS attacks can cost an organization both time and money while their resources and services are inaccessible.[2]

Distributed Denial-of-Service attack (DDoS)

A distributed denial-of-service (DDoS) attack occurs when multiple machines are operating together to attack one target. DDoS allows for exponentially more requests to be sent to the target, therefore increasing the attack power. It also increases the difficulty of attribution, as the true source of the attack is harder to identify. DDoS attackers often leverage the use of a botnet a group of hijacked internet connected devices to carry out large scale attacks. Attackers take advantage of security vulnerabilities or device weaknesses to control numerous devices using command and control software [3]. Once in control, an attacker can command their botnet to conduct DDoS on a target. In this case, the infected devices are also victims of the attack.

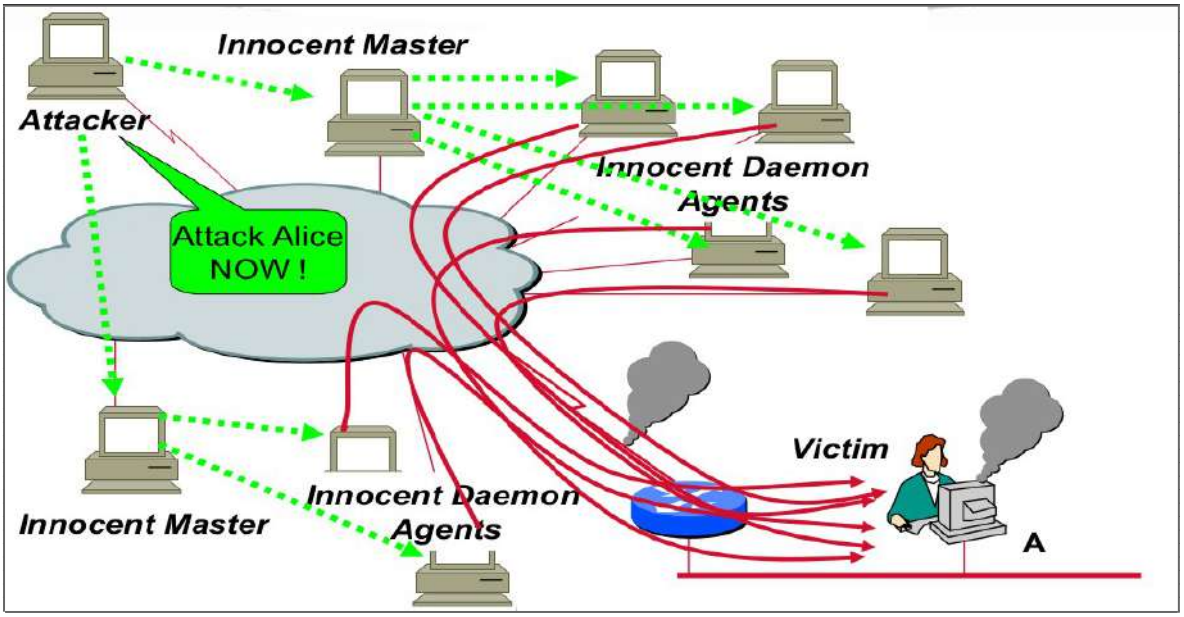

Figure 1-2 : DDoS Attack used Botnet machines

Types of DoS and DDoS Attacks

Over the years, denial-of-service attacks have evolved to encompass a number of attack vectors and mechanisms :

I.4.1.A -Smurf Attack

The attacker sends Internet Control Message Protocol broadcast packets to a number of hosts with a spoofed source Internet Protocol (IP) address that belongs to the target machine [3]. The recipients of these spoofed packets will then respond, and the targeted host will be flooded with those responses. The scenario of such an attack is as follows :

- The attacking machine ping one or more broadcast servers by falsifying the source IP address and providing the IP address of a target machine.
- The broadcast server echoes the request on the set network.
- All machines in the network send a response to the broadcast service.
- Broadcast server redirects responses to the machine target.

Thus, when the attacking machine sends a request to several broadcast servers located on different networks, the set of responses of the computers of the different networks will be routed on the target machine.

I.4.1.B -NTP Amplification Attack

The Network Time Protocol (NTP) used by machines connected to the Internet to set their clocks. In a NTP Amplification attack, DDoS attackers take advantage of NTP flood. Attackers spoof a victim's NTP infrastructure and use Open NTP servers, which send (MON_GETLIST) very small requests resulting in a very high-volume of NTP responses (Amplification Factor). Since attackers spoof a victim's NTP infrastructure, all of the reflected/amplified responses flood a victim's NTP server, which take them offline or flood the network and take it offline as well. This attack is rarely detectable by deep packet inspection technologies because the NTP requests and responses seem to be 100% normal [4]. An NTP amplification attack can be broken down into four steps :

- The attacker uses a botnet to send UDP packets with spoofed IP addresses to a NTP server which has its monlist command enabled. The spoofed IP address on each packet points to the real IP address of the victim.
- Each UDP packet makes a request to the NTP server using its monlist command, resulting in a large response.
- The server then responds to the spoofed address with the resulting data.
- The IP address of the target receives the response and the surrounding network infrastructure becomes overwhelmed with the deluge of traffic, resulting in a denial-of-service.

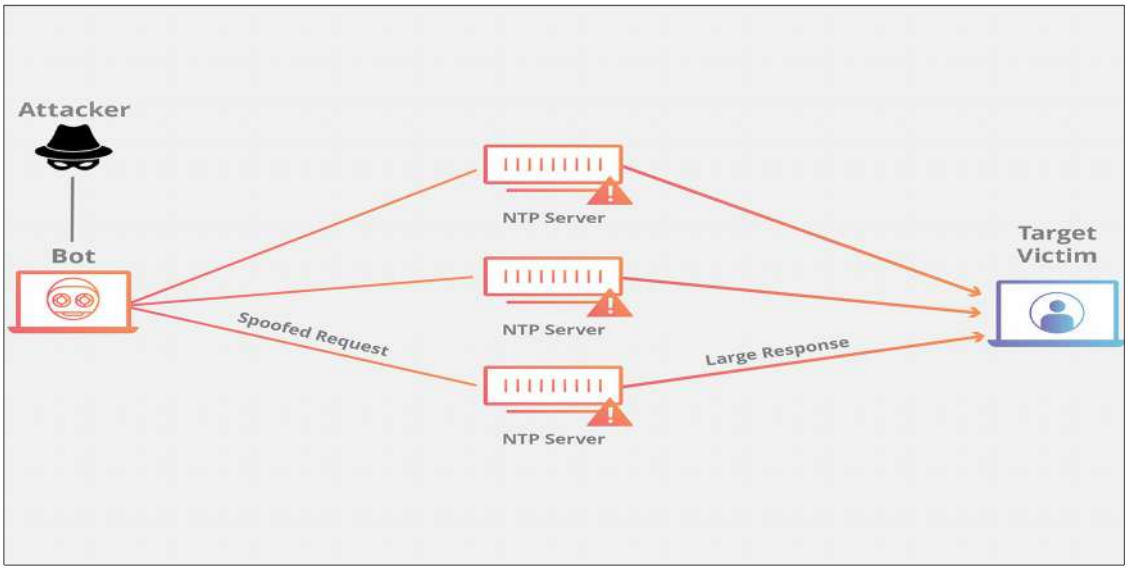

Figure 1-3 : DDoS attack through NTP servers

I.4.1.C -SYN flood attack

A SYN flood (half-open attack) is a type of denial-of-service (DDoS) attack which aims to make a server unavailable to legitimate traffic by consuming all available server resources. By repeatedly

sending initial connection request (SYN) packets, the attacker is able to overwhelm all available ports on a targeted server machine, causing the targeted device to respond to legitimate traffic sluggishly or not at all. How does a SYN flood attack work?

To create denial-of-service, an attacker exploits the fact that after an initial SYN packet has been received, the server will respond back with one or more SYN/ACK packets and wait for the final step in the handshake. Here's how it works.[3]

- The attacker sends a high volume of SYN packets to the targeted server, often with spoofed IP addresses.
- The server then responds to each one of the connection requests and leaves an open port ready to receive the response.
- While the server waits for the final ACK packet, which never arrives, the attacker continues to send more SYN packets. The arrival of each new SYN packet causes the server to temporarily maintain a new open port connection for a certain length of time, and once all the available ports have been utilized the server is unable to function normally.

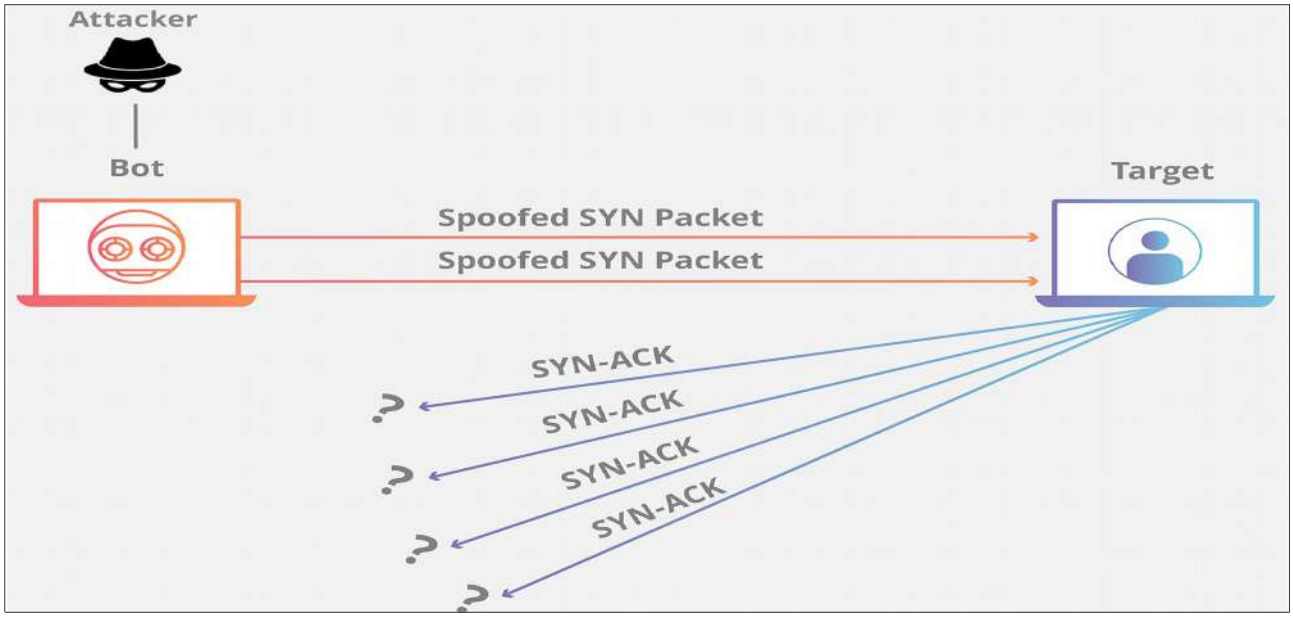

Figure 1-4 : DDoS attack with syn flooding

I.4.1.D -HTTP flood DDoS attack

An HTTP flood attack is a type of volumetric distributed denial-of-service (DDoS) attack designed to overwhelm a targeted server with HTTP requests . Once the target has been saturated with requests and is unable to respond to normal traffic, denial-of-service will occur for additional requests from actual users [3]. How does an HTTP flood attack work?

HTTP flood attacks are a type of “layer 7” DDoS attack. Layer 7 is the application layer of the OSI model, and refers to internet protocols such as HTTP. HTTP is the basis of browser-based internet requests, and is commonly used to load web pages or to send form contents over the Internet. Mitigating application layer attacks is particularly complex, as the malicious traffic is difficult to distinguish from normal traffic. In order to achieve maximum efficiency, malicious actors will commonly employ or create botnets in order to maximize the impact of their attack. By utilizing many devices infected with malware, an attacker is able to leverage their efforts by launching a larger volume of attack traffic.

There are two varieties of HTTP flood attacks :

- **HTTP GET attack** - in this form of attack, multiple computers or other devices are coordinated to send multiple requests for images, files, or some other asset from a targeted server. When the target is inundated with incoming requests and responses, denial-of-service will occur to additional requests from legitimate traffic sources.
- **HTTP POST attack** - typically when a form is submitted on a website, the server must handle the incoming request and push the data into a persistence layer, most often a database. The process of handling the form data and running the necessary database commands is relatively intensive compared to the amount of processing power and bandwidth required to send the POST request. This attack utilizes the disparity in relative resource consumption, by sending many post requests directly to a targeted server until it's capacity is saturated and denial-of-service occurs.

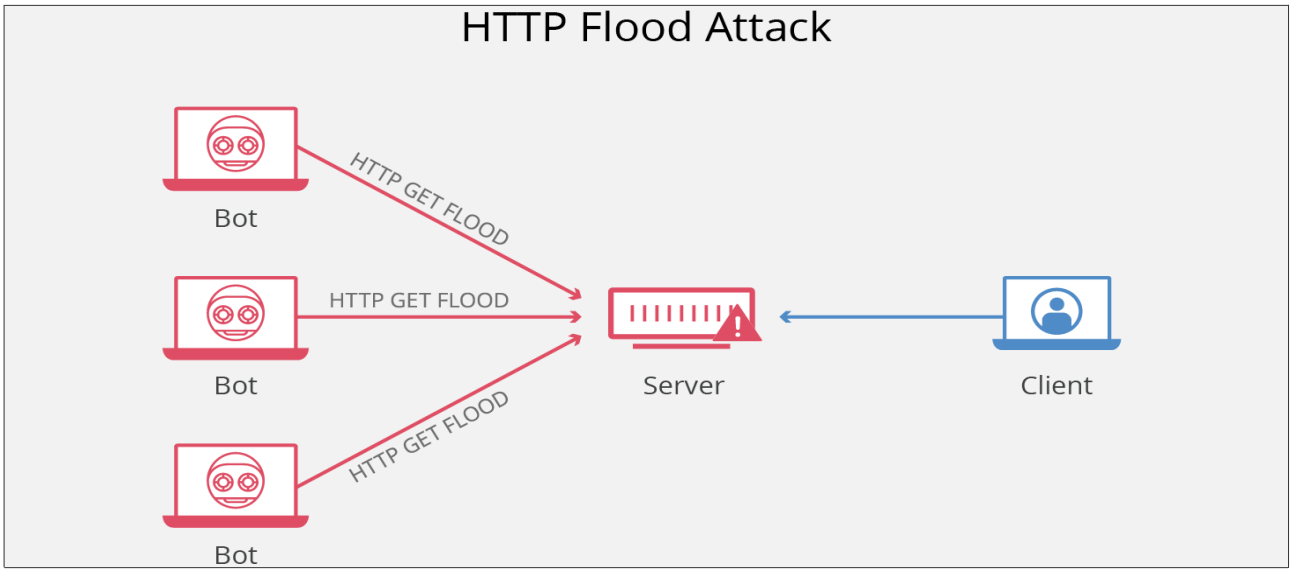

Figure 1-5 : HTTP flood request cause DDoS attack

I.4.2 -Remote Code Execution (buffer overflow)

Compiled software that runs in a native execution environment has historically been plagued by vulnerabilities like buffer overflows and format string bugs. One of the most significant advantages of languages such as C# and Java is that programmers do not need to worry about the kind of buffer management and pointer arithmetic problems that have affected software developed in native languages such as C and C++, and have given rise to the majority of critical bugs found in that software.[5]

I.4.2.A -Buffer overflow Vulnerabilities

Buffer overflow vulnerabilities occur when an application copies user controllable data into a memory buffer that is not sufficiently large to accommodate it. The destination buffer is overflowed, resulting in adjacent memory being overwritten with the user's data. Depending on the nature of the vulnerability, an attacker may be able to exploit it to execute arbitrary code on the server or perform other unauthorized actions. Buffer overflow vulnerabilities have been hugely prevalent in native software over the years, and have been widely regarded as the Public Enemy Number One that developers of such software need to avoid. we shall cover two main categories of classic software vulnerability buffer overflows:stack buffer overflow, heap-based overflow.

I.4.2.B -Stack Overflows

Buffer overflows typically arise when an application uses an unbounded copy operation (such as strcpy in C) to copy a variable-size buffer into a fixed-size buffer without verifying that the fixed sized buffer is large enough. For example,the following function copies the username string into a fixed-size buffer allocated on the stack :

```
bool CheckLogin(char* username, char* password)
        {
                char* _username = (char*) malloc(32);
                strcpy(_username, username);
                ...
```

Figure 1- 6 : Vulnerable code that cause Stack overflow

If the username string contains more than 32 characters, the _username buffer is overflowed, and the attacker will overwrite the data in adjacent memory. In a stack-based buffer overflow, a

successful exploit typically involves overwriting the saved return address on the stack. When the CheckLogin function is called, the processor pushes onto the stack the address of the instruction following the call [5]. When CheckLogin is finished, the processor pops this address back off the stack and returns execution to that instruction. In the meantime, the CheckLogin function allocates the _username buffer on the stack right next to the saved return address. If an attacker can overflow the _username buffer, he can overwrite the saved return address with a value of his choosing, thereby causing the processor to jump to this address and execute arbitrary code.[5]

I.4.2.C -Heap Overflows

Heap-based buffer overflows essentially involve the same kind of unsafe operation as described previously, except that the overflowed destination buffer is allocated on the heap, not the stack :

```
bool CheckLogin(char* username, char* password)
        {
                char* _username = (char*) malloc(32);
                strcpy(_username, username);
                ...
```

Figure 1- 7 : Vulnerable code that cause heap overflow

In a heap-based buffer overflow, what is typically adjacent to the destination buffer is not any saved return address but other blocks of heap memory, separated by heap control structures. The heap is implemented as a doubly linked list: each block is preceded in memory by a control structure that contains the size of the block, a pointer to the previous block on the heap, and a pointer to the next block on the heap. When a heap buffer is overflowed, the control structure of an adjacent heap block is overwritten with user controllable data. This type of vulnerability is less straightforward to exploit than a stack-based overflow.

I.4.3 -Malware attack

Manual attacks on systems do not happen as much as they did in the past. Today hackers automate their attacks by creating a piece of malicious software (malware) that can compromise thousands of systems at one time with more precision. most malware is created to obtain sensitive information (credit card data, Social Security numbers, credentials,etc.), gain unauthorized access to systems. The most commonly used schemes for making money through malware are as follows [6]

- Systems are compromised with bots and are later used in distributed denial-of-service (DDoS) attacks, spam distribution, or as part of a botnet's command and control system.
- Ransomware encrypts some or all of the users' files with keys that are only given to the users after they pay a ransom, typically using cryptocurrencies.
- Spyware collects personal data for the malware developer to resell to others.
- Malware redirects web traffic so that people are pointed toward a specific product for purchase.
- Malware installs key loggers, which collect sensitive financial information for the malware author to use.
- Malware is used to carry out phishing attacks, fraudulent activities, identity theft steps, and information warfare activities.

There are several types of malicious code, or malware such as viruses, worms, trojan horses, and logic bombs are as follows :

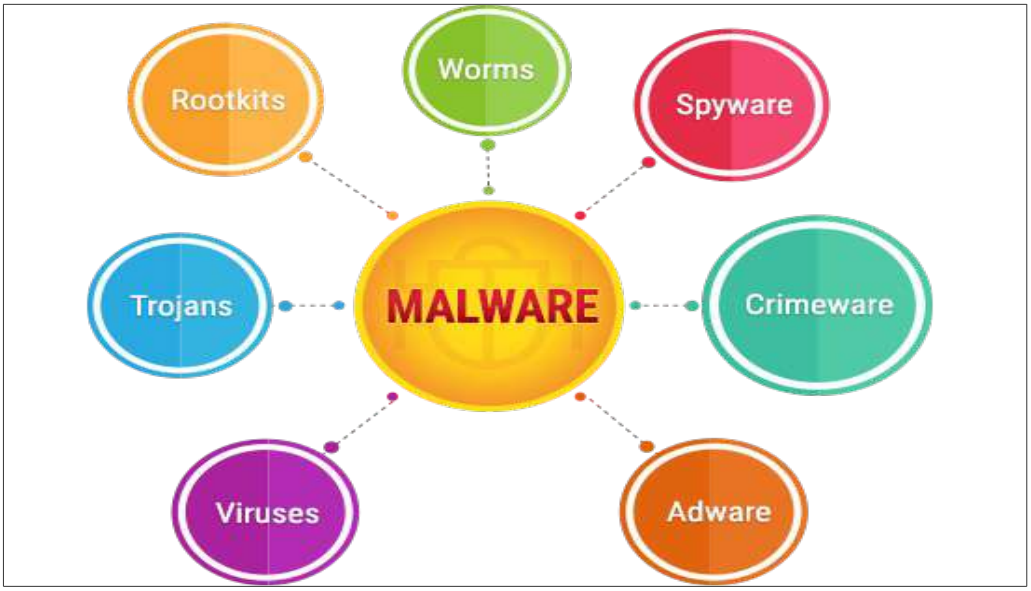

Figure 1-8 : Famous types of malware

I.4.3.A -Viruses

A virus is a small application, or string of code, that infects software. The main function of a virus is to reproduce and deliver its payload, and it requires a host application to do this. In other words, viruses cannot replicate on their own. A virus infects a file by inserting or attaching a copy of itself to the file. The virus is just the "delivery mechanism." [6] It can have any type of payload

(deleting system files, displaying specific messages, re-configuring systems, stealing sensitive data, installing a sniffer or back door).

I.4.3.B -Worms

Worms are different from viruses in that they can reproduce on their own without a host application, and are self-contained programs. In the digital world,worms are just little programs, and like viruses they are used to transport and deliver malicious payloads. One of the most famous computer worms is Stuxnet, which targeted Siemens supervisory control and data acquisition (SCADA) software and equipment. It has a highly specialized payload that was used against Iran's uranium enrichment infrastructures with the goal of damaging the country's nuclear program.

I.4.3.C -Trojan Horses

A Trojan horse is a program that is disguised as another program. For example [6], a Trojan horse can be named Notepad.exe and have the same icon as the regular Notepad program. However, when a user executes Notepad.exe, the program can delete system files. Trojan horses perform a useful functionality in addition to the malicious functionality in the background. So the Trojan horse named Notepad.exe may still run the Notepad program for the user, but in the background it will manipulate files or cause other malicious acts. Users are commonly tricked into downloading some type of software from a website that is actually malicious. The Trojan horse can then set up a back door, install keystroke loggers, implement rootkits, upload files from the victim's system, install bot software, and perform many other types of malicious acts.

I.4.3.D -Ransomware

Ransomware is one of the most dangerous malware that, once it's taken over your computer, threatens you with harm, usually by denying you access to your data. The attacker demands a ransom from the victim, promising not always truthfully to restore access to the data upon payment. Users are shown instructions for how to pay a fee to get the decryption key. The costs can range from a few hundred dollars to thousands, payable to Cyber criminals in Bitcoin. There are number of vectors ransomware can take to access a computer [11]. One of the most common delivery systems is phishing spam attachments that come to the victim in an email, masquerading as a file they should trust. Once they're downloaded and opened, they can take over the victim's computer, especially if they have built-in social engineering tools that trick users into allowing administrative access. Some other, more aggressive forms of ransomware, like WannaCry, exploit security holes to infect computers without needing to trick users.

I.4.4 -Web Application Attack

Web application attacks. Despite their advantages, web applications do raise a number of security concerns stemming from improper coding. Serious weaknesses or vulnerabilities, allow hackers to gain direct and public access to databases in order to churn sensitive data this is known as a web application attack [9]. Many of these databases contain valuable information (e.g., personal and financial details) making them a frequent target of hackers. Although such acts of vandalism as defacing corporate websites are still commonplace, nowadays, hackers prefer gaining access to the sensitive data residing on the database server because of the immense pay-offs in selling the data.In the framework described below, it is easy to see how a hacker can quickly access the data residing on the database through a dose of creativity and, with luck, negligence or human error, leading to vulnerabilities in the web applications.

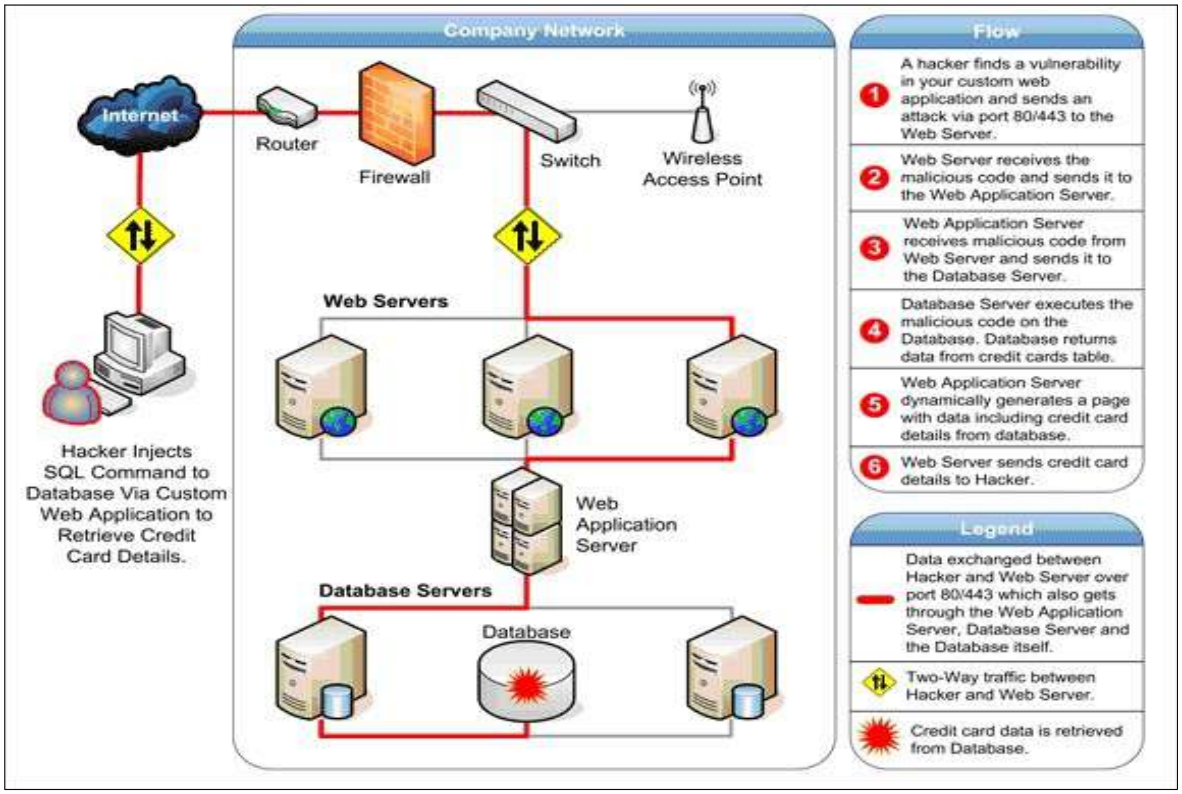

Figure 1-9 : Phases of attack web application-level

I.4.4.A -Cross-Site Scripting Attack (XSS)

Cross-Site Scripting (XSS) attacks are a type of injection, in which malicious scripts are injected into otherwise benign and trusted websites. XSS attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user. Flaws that allow these attacks to succeed are quite widespread and occur anywhere a web application uses input from a user within the output it generates without validating or encoding it.

An attacker can use XSS to send a malicious script to an unsuspecting user [7]. The end user's browser has no way to know that the script should not be trusted, and will execute the script. Because it thinks the script came from a trusted source, the malicious script can access any cookies, session tokens, or other sensitive information retained by the browser and used with that site. These scripts can even rewrite the content of the HTML page,as you see the Figure below it shows the attack mechanism .

I.4.4.A.1 How the example attack works

The figure below illustrates how this example attack can be performed by an attacker :

1. The attacker uses one of the website's forms to insert a malicious string into the website's database.
2. The victim requests a page from the website.
3. The website includes the malicious string from the database in the response and sends it to the victim.
4. The victim's browser executes the malicious script inside the response, sending the victim's cookies to the attacker's server.

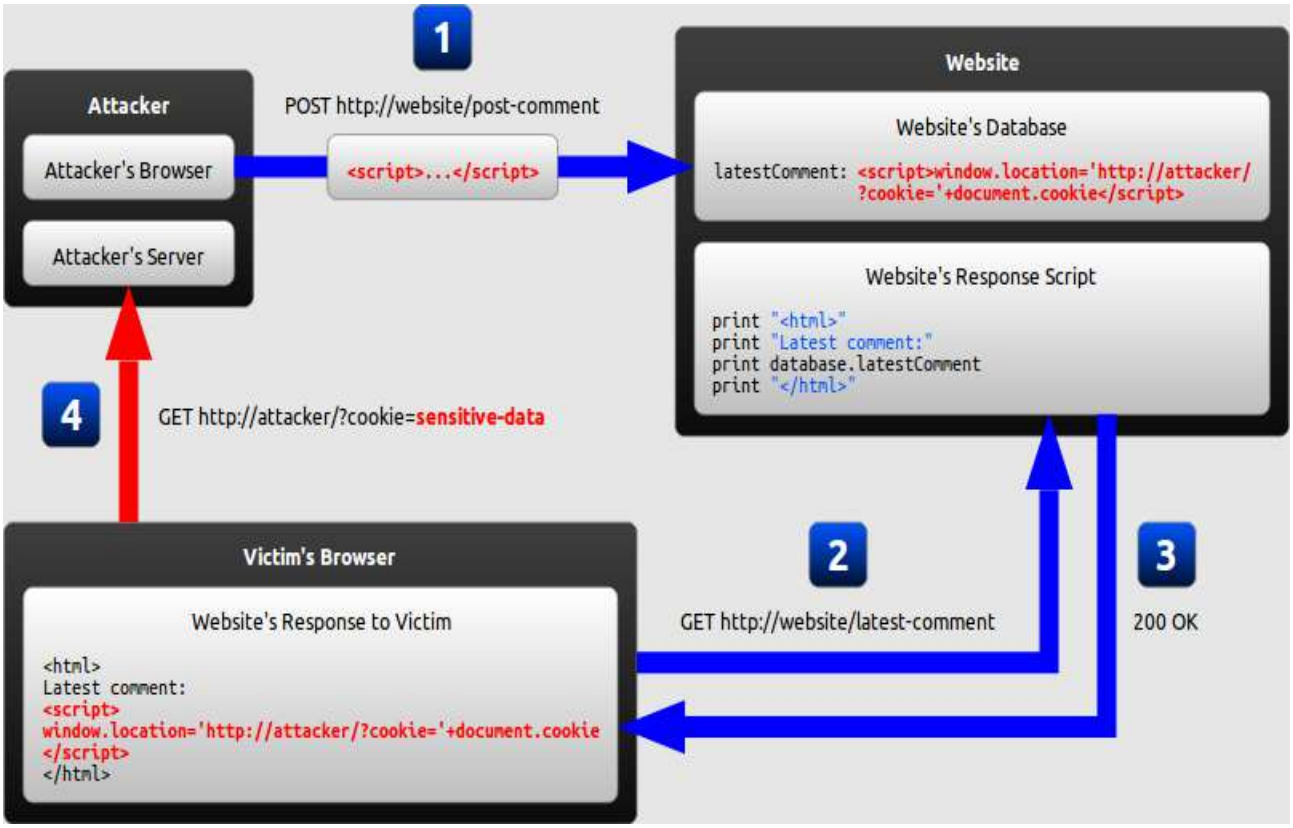

Figure 1-10 : Stages of exploit XSS vulnerability

I.4.4.A.2 The consequences of malicious JavaScript

Among many other things, the ability to execute arbitrary JavaScript in another user's browser allows an attacker to perform the following types of attacks:

- Cookie theft: The attacker can access the victim's cookies associated with the website using `document.cookie DOM`, send them to his own server, and use them to extract sensitive information like session IDs.
- Keylogging: The attacker can register a keyboard event listener using `addEventListener` and then send all of the user's keystrokes to his own server, potentially recording sensitive information such as passwords and credit card numbers.
- Phishing: The attacker can insert a fake login form into the page using DOM manipulation, set the form's `action` attribute to target his own server, and then trick the user into submitting sensitive information.

I.4.4.B -SQL Injection Attack

SQL injection (SQLi) is an application security weakness that allows attackers to control an application's database – letting them access or delete data, change an application's data-driven behavior, and do other undesirable things – by tricking the application into sending unexpected SQL commands. SQL injections are among the most frequent threats to data security [8]. SQL injection weaknesses occur when an application uses untrusted data, such as data entered into web form fields, as part of a database query. When an application fails to properly sanitize this untrusted data before adding it to a SQL query, an attacker can include their own SQL commands which the database will execute. Such SQLi vulnerabilities are easy to prevent, yet SQLi remains a leading web application risk, and many organizations remain vulnerable to potentially damaging data breaches resulting from SQL injection.

I.4.4.B.1 SQL Injection Attack Performed

To make an SQL Injection attack, an attacker must first find vulnerable user inputs within the web page or web application. A web page or web application that has an SQL Injection vulnerability uses such user input directly in an SQL query. The attacker can create input content. Such content is often called a malicious payload and is the key part of the attack. After the attacker sends this content, malicious SQL commands are executed in the database. SQL is a query language that was designed to manage data stored in relational databases. You can use it to access, modify, and delete data. Many web applications and websites store all the data in SQL databases. In some cases, you can also use SQL commands to run operating system commands. Therefore, a successful SQL Injection attack can have very serious consequences.

- Attackers can use SQL Injections to find the credentials of other users in the database. They can then impersonate these users. The impersonated user may be a database administrator with all database privileges.
- SQL allow you select and output data from the database. An SQL Injection vulnerability could allow the attacker to gain complete access to all data in a database server.

- SQL also lets you alter data in a database and add new data. For example, in a financial application, an attacker could use SQL Injection to alter balances, void transactions, or transfer money to their account.
- You can use SQL to delete records from a database, even drop tables. Even if the administrator makes database backups, deletion of data could affect application availability until the database is restored. Also, backups may not cover the most recent data.
- In some database servers, you can access the operating system using the database server. This may be intentional or accidental. In such case, an attacker could use an SQL Injection as the initial vector and then attack the internal network behind a firewall.

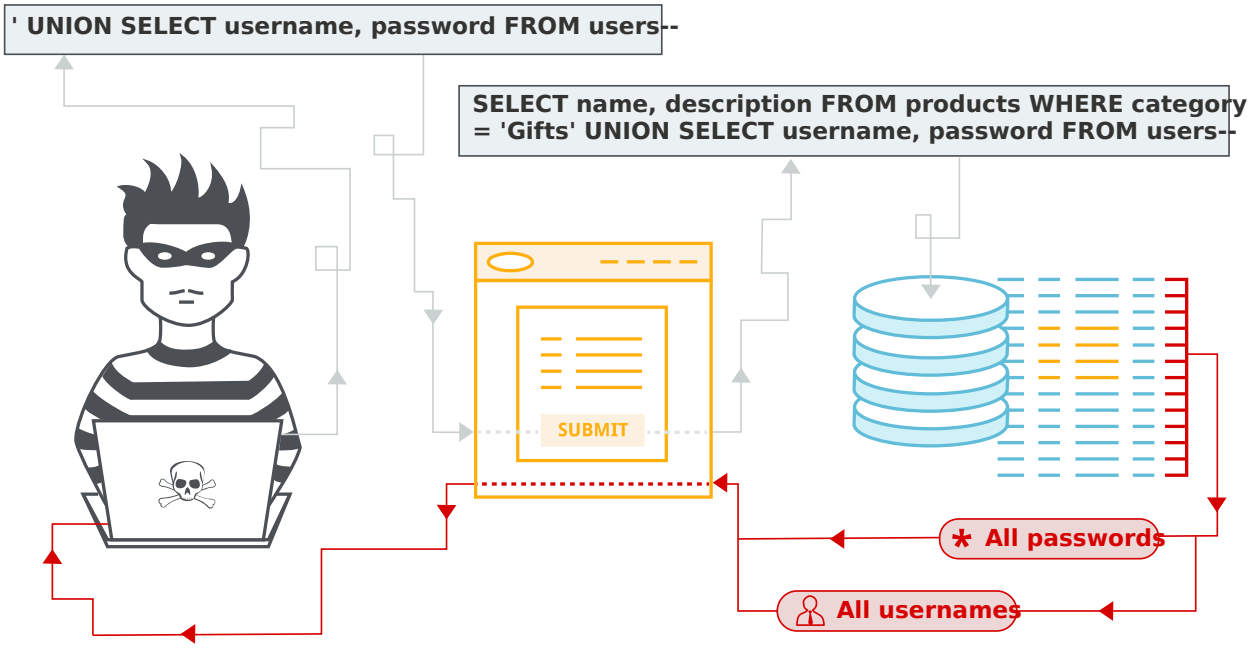

Figure 1-11 : Inject malicious query in Data Base

I.4.4.B.2 Simple SQL Injection Example

The first example is very simple [10]. It shows, how an attacker can use an SQL Injection vulnerability to go around application security and authenticate as the administrator The following

script is pseudo-code executed on a web server. It is a simple example of authenticating with a username and a password. The example database has a table named `users` with the following columns: `username` and `password`.

```
# Define POST variables
uname = request.POST['username']
passwd = request.POST['password']

# SQL query vulnerable to SQLi
sql = "SELECT id FROM users WHERE username='" + uname + "' AND password='" + passwd + "'"

# Execute the SQL statement
database.execute(sql)
```

These input fields are vulnerable to SQL Injection. An attacker could use SQL commands in the input in a way that would alter the SQL statement executed by the database server. For example, they could use a trick involving a single quote and set the `passwd` field to :

```
password' OR 1=1
```

As a result, the database server runs the following SQL query :

```
SELECT id FROM users WHERE username='username' AND password='password' OR 1=1'
```

Because of the `OR 1=1` statement, the `WHERE` clause returns the first `id` from the `users` table no matter what the `username` and `password` are. The first user `id` in a database is very often the administrator. In this way, the attacker not only bypasses authentication but also gains administrator privileges. They can also comment out the rest of the SQL statement to control the execution of the SQL query further :

```
-- MySQL, MSSQL, Oracle, PostgreSQL, SQLite
' OR '1'='1' --
' OR '1'='1' /*
-- MySQL
' OR '1'='1' #
-- Access (using null characters)
' OR '1'='1' %00
' OR '1'='1' %16
```

I.4.5 -Man-in-the-Middle Attack

Man-in-the-middle attacks Which will be the focus of our project happen at different levels and forms. However, its basic concept requires three key players: the victim, the entity which victim is trying to contact, and the man in the middle. The victim can be any user trying to access a website or a web application (the entity). On any typical connection, the user can directly connect to the website server and visit the site. The “man in the middle” inserts itself between the connection of the user and the website server [12]. It will try to mimic the website and pretend that normal communication is happening with the user. However, the “man in the middle” snoops in the conversation and gathers important information. The hacker will collect personal information such as login details, credit card numbers, and others.

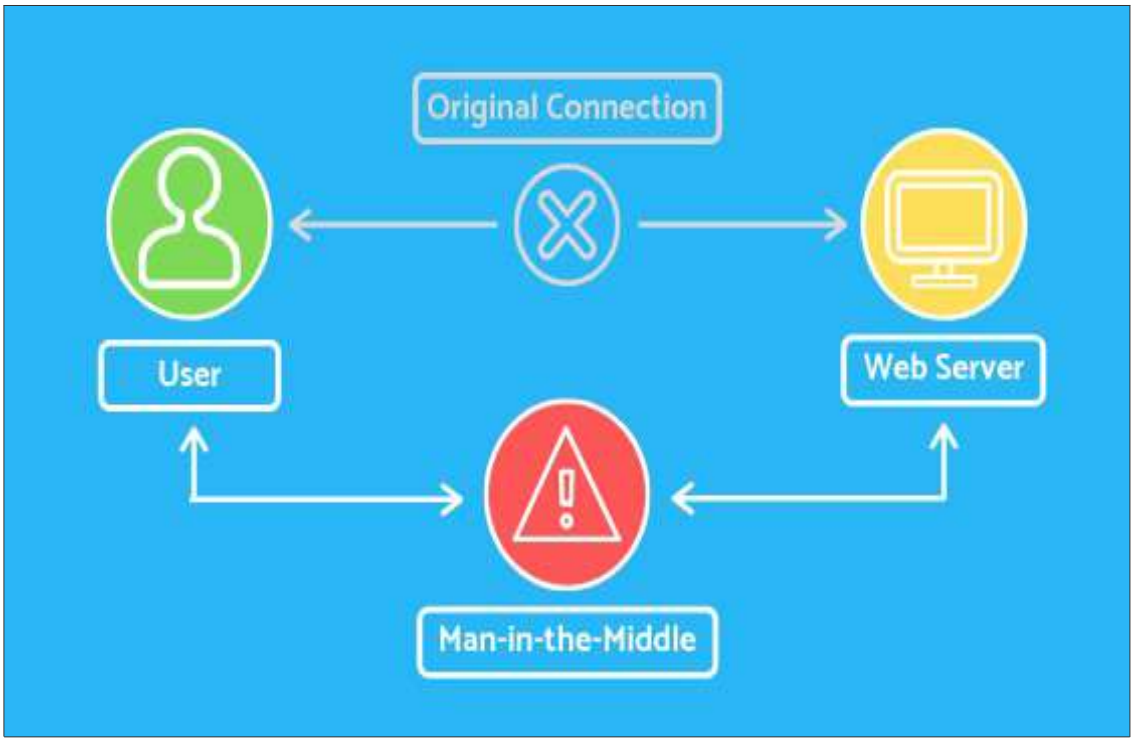

Figure 1-16 : MitM attack between User and Web Server

MitM attacks are not limited to crystallographic system [14], they include any network attack in which the attacker makes independent connections with two victims and starts relaying messages between the two victims without their knowledge. In other words, the attacker becomes an invisible proxy between the two victims. Common ways to execute network MITM attacks include :

- ARP poisoning
- DNS poisoning

- DHCP poisoning

I.4.5.A -ARP poisoning attack

One way for an attacker to execute a network MITM attack is to send gratuitous Address Resolution Protocol ARP packets (unsolicited ARP replies) to each victim node, thereby attempting to poison their ARP cache. The attacker specifically wants to replace the MAC address of the remote victim's IP address with the attacker's MAC address. [13]

In a common MITM attack, one of the target nodes is in the attacker's LAN, while other is in the Internet, such as when attacking computers in a wireless network. In order to execute the attack on the two victims, the attacker sends gratuitous ARP packets to the local victim and the local gateway, to ARP poison these two caches. The ARP cache of the local victim must have the attacker's MAC address for the gateway's IP address, and the ARP cache of the gateway must have the attacker's MAC address for the local target's IP address. This way, all communication from the target node to the Internet (via the gateway) passes through the attacker.

ARP poisoning in detail :

1. attacker scan range network for up devices and enumerate its.
2. Attacker selects two target nodes on the same local subnet (for example, a client and the gateway).
3. Attacker sends gratuitous ARP packets to the local node with the IP address of the default gateway but the attacker's MAC address.
4. Attacker sends gratuitous ARP packets to the default gateway with the IP address of the client node but the attacker's MAC address.
5. Attacker starts filtering IP packets so that only those coming from the local client node and local gateway targets are intercepted; this is necessary so that the attacker can differentiate traffic from the victim and other Internet-bound traffic.

Note : When an attacker perform this attack he has a high probability that he is skipping HTTPS encryption.[14]

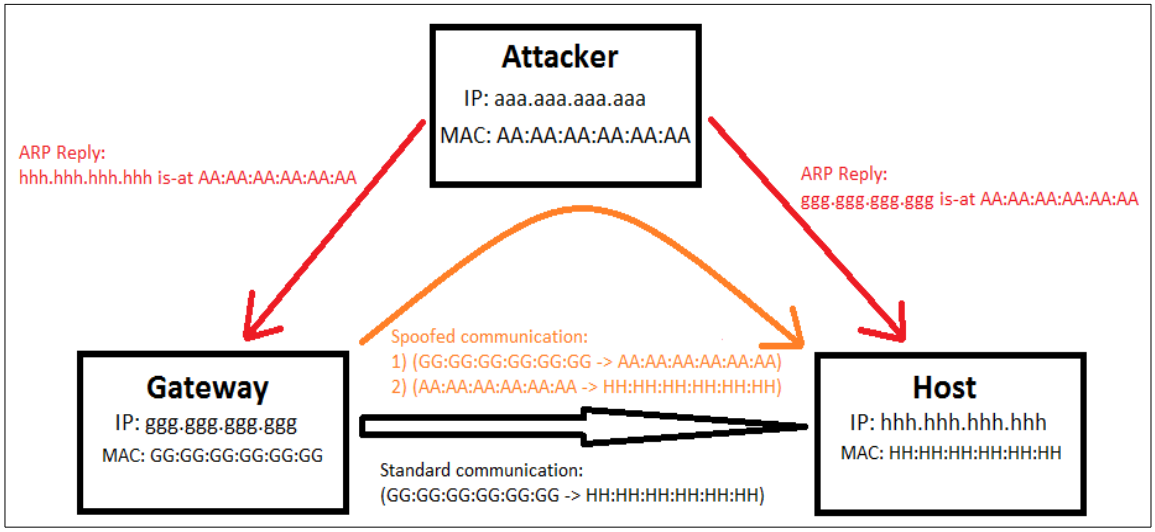

Figure 1-17 : ARP cache poisoning scenario attack

I.4.5.B -DNS spoofing attack

A DNS server provides the IP address associated with a given domain/host name. It is possible for an attacker to replace a valid domain name's IP with an attacker-controlled IP address [15]. By doing so, any victim that resolves the given domain/host name will receive a response that includes the attacker's address. Attackers can use this technique to execute MITM attacks on any of the DNS server's clients. Thus, this technique can be used to execute MITM attacks on different users simultaneously to, for example, execute phishing attacks.

DNS spoofing in detail :

1. Attacker finds the DNS server of one of the victim clients.
2. Attacker sets up a malicious DNS server that sends malicious IP address information for a valid domain/host name (*DN*).
3. Attacker tricks any of the clients of the DNS server found in step 1 to make a request to their malicious DNS server.
4. Attacker waits until server client makes a request to their malicious DNS server.
5. Attacker responds with a DNS response that indicates that domain name *DN* corresponds to an attacker's IP address.

6. Attacker waits until victim client connects to the domain name *DN*. If the attack succeeds, the attacker is able to intercept this connection, because the victim is actually connecting to the attacker-controlled server.

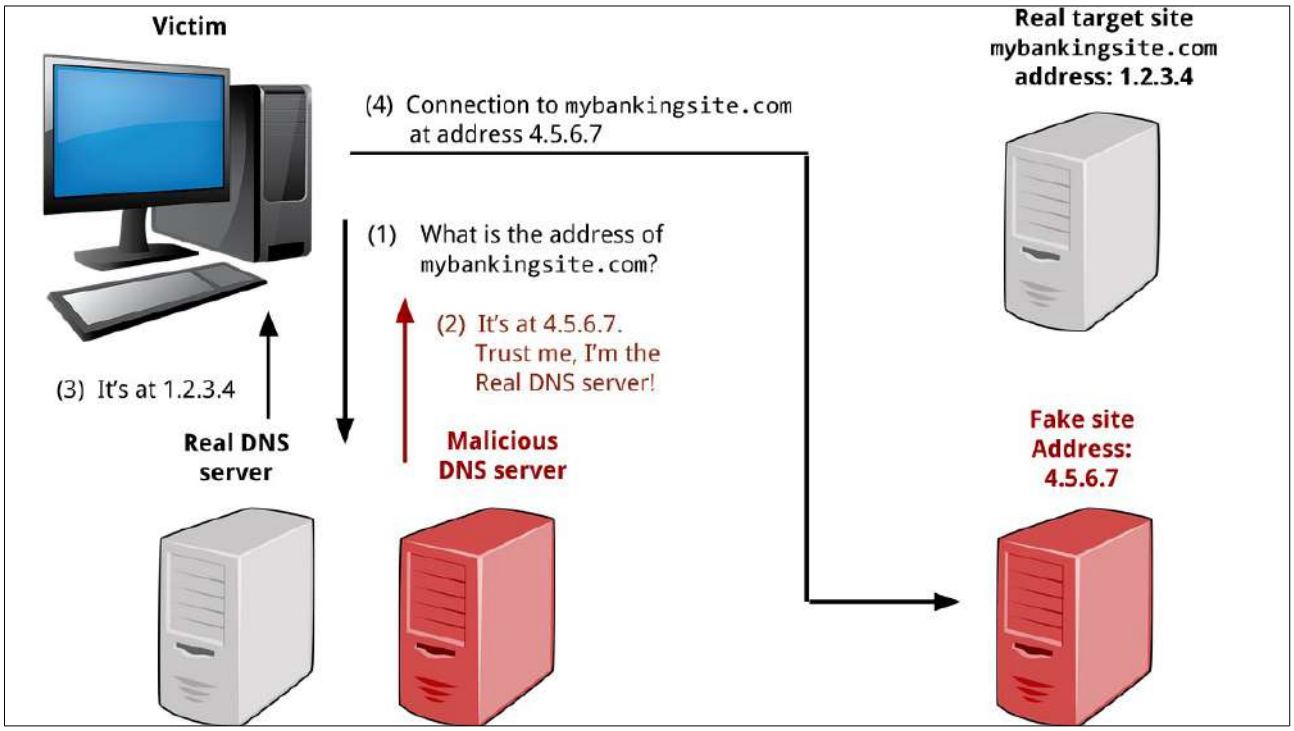

Figure 1-18 : DNS Spoofing attack level

I.4.5.C -DHCP spoofing attack

A DHCP server provides IP information such as the default gateway IP address to network nodes that join a local network. An attacker can pose as a DHCP server and send forged DHCP acknowledgments to any connecting nodes. If the connecting node receives the DHCP acknowledgment from the attacker before the real DHCP server, it uses the information provided by the attacker to resolve its IP configuration. The attacker can supply its own IP address for the default gateway address in forged DHCP requests in order to execute MITM attacks between the connecting node and either a local network node or a remote one.[15]

DHCP spoofing In detail :

1. Attacker starts DHCP server that sends the attacker's IP address for the default gateway IP address.
2. Victim A connects to the network (note that the victim must obtain their IP information via DHCP).

3. Attacker beats race condition against the real DHCP server and sends a forged DHCP acknowledgment to victim A.
4. Victim A receives the attacker's IP as the default gateway in the DHCP response message.
5. Attacker starts intercepting IP packets sent from victim A to victim B, as the attacker is now the gateway for victim A.

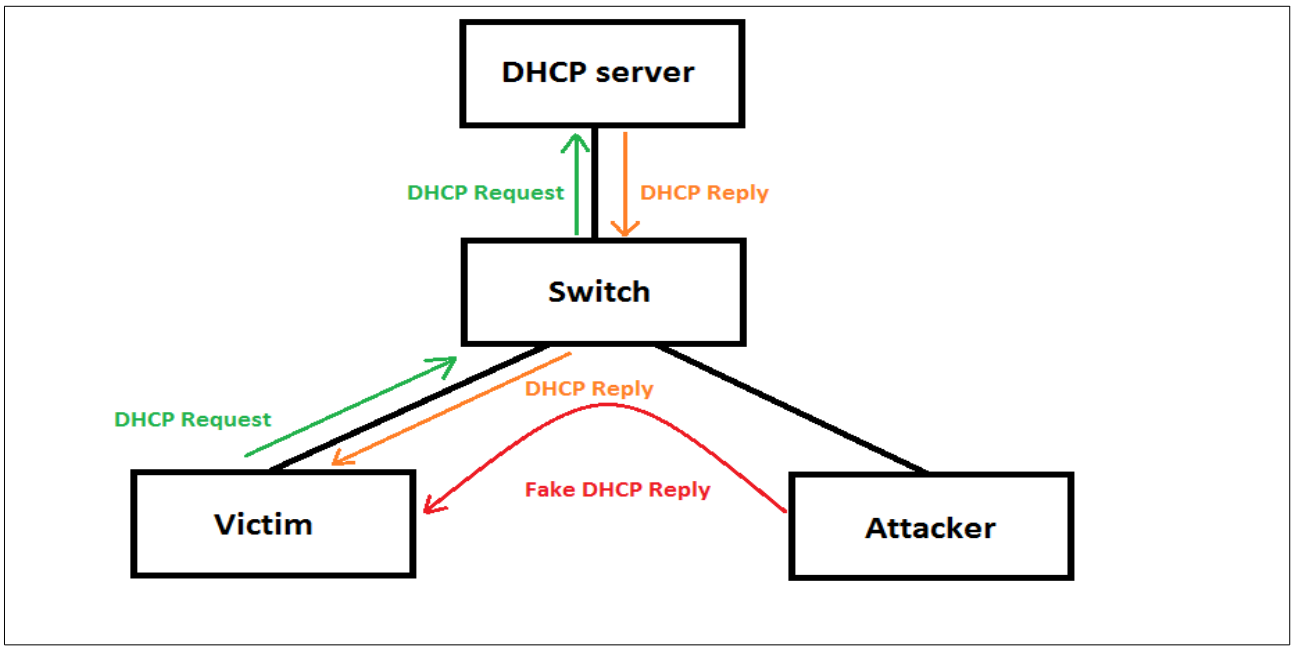

Figure 1-19 : DHCP Spoofing attack scenario

I.5 -Conclusion

Due to the dependence of a lot of institutions and people on information systems in various areas of information technology is still evolving and this happens a lot of gaps that can be exploited by hackers as they can also develop new methods of hacking and new types of attacks. Information security engineers also work to increase system immunity, introduce defensive methods, and reduce the likelihood of attacks by attackers.

II Chapter 2 : Mechanisms for protecting computer systems

II.1 -Introduction

In the previous chapter we talked about the most popular types of attack techniques used by hackers. in this chapter We will present the protection techniques currently used and the methods of defense against Cyber attacks. We will learn about the most important protocols and how they work. The use of techniques used by security managers to implement security plans and policies and the implementation of various protection programs and device.

II.2 -Antivirus

II.2.1 -Principle of operation

An antivirus is a program that can detect the presence of malware on a computer and, to the extent possible, disinfect it. This is called virus eradication to describe the procedure of cleaning the computer. Here's a detailed look at how antivirus software works, and how it stands guard to protect your PC or mobile device from these malicious and menacing threats.[16]

II.2.2 -Virus definitions

This is practically the first method traditional antivirus software employ to identify malware. The programs rely upon signatures to detect new malware. Provided the company has already analyzed and extracted a proper signature of the file that is then kept in a database. Threats are compared to this database, and devices are then protected in case the signatures match. But while this approach does prevent malware outbreaks, Cyber criminals try to stay one step ahead by writing viruses that either encrypt themselves or modify their code in order to disguise and not match virus definitions.

II.2.3 -Sandbox Detection

This is a behavioral based detection technique that executes the programs in a virtual environment, as opposed to detecting its fingerprint at run time. Antivirus software that come with this type of detection capabilities execute programs in a separate, virtual environment, and log the actions it performs to determine whether the programs are malicious or not. If found safe, a given program is then executed in the real environment [16]. As you can imagine, this technique is both heavy and slow, and its resource intensive nature means that it is rarely used in consumer antivirus solutions. End users may not always have the need for sandbox detection, but enterprises do, and antivirus solutions designed for corporate and network use offer this.

II.2.4 -Data mining

This is one of the latest approaches in malware detection that security vendors now provide with their antivirus and antimalware products.A series of features of files are extracted from files, and then data mining and machine learning algorithms are used to classify the behavior of a file and detect whether it has malicious intent or not. This is particularly helpful in detecting and defeating the newest forms of malware in the wild.

II.2.5 -Types of Scans

All these varying types of detection capabilities are fine, but another, equally important, measure of how successful an antivirus is in protecting a system is the types of scans it offers.[17]

II.2.5.A -Real-time Protection

Also known as memory-resident scanning or background guard.This type of scanning refers to the automatic protection that almost all modern antivirus programs offer. It basically monitors the system for any suspicious activity in real time, while data is loaded into the active memory. For example, when a USB drive is inserted, a browser is opened, or a downloaded file is executed. The price of this type of scanning is performance, but it offers increased protection, and more chances of catching malware before it does damage.

II.2.5.B -Smart Scan

These refer to an approach where an antivirus only scans selected files, that are more suspicious to be altered or infected. Smart scanning lowers the need of system resources, while protecting against the more common types of viruses, threats and risks

II.2.5.C -Startup Scan

Antivirus software often come with a special program that is designed to run every time the PC is booted up. It does a quick scan of the boot sectors and critical system files, instead of a full disk scan that takes a long time to finish. This comes in particularly handy to catch boot sector viruses, before they get a chance to spread.

II.3 -Firewalls

The Internet of today is in stark contrast to the close knit group of research networks that made up the Internet forty years ago [18]. As the Internet has grown, the need to protect networks and even individual computers has become a major concern. To this end, devices and software that fall

under the banner of “firewall” have become a necessity for any and all computers connected to the Internet that the user wants to remain safe.Firewalls have the ability to further enhance security by enabling granular control over what types of system functions and processes have access to networking resources. These firewalls can use various types of signatures and host conditions to allow or deny traffic. Although they sound complex, firewalls are relatively easy to install, setup and operate.

II.3.1 -What’s in a Name?

The question remains: what exactly is a firewall? Firewalls are network devices or software that separates one trusted network from an untrusted network by means of rule based filtering of network traffic as depicted in Figure 2-1. Despite the broad definition of a firewall, the specifics of what make up a firewall depend on the type of firewall. There are three basic types of firewall: packet-filtering firewalls, stateful firewalls, and application gateway firewalls.

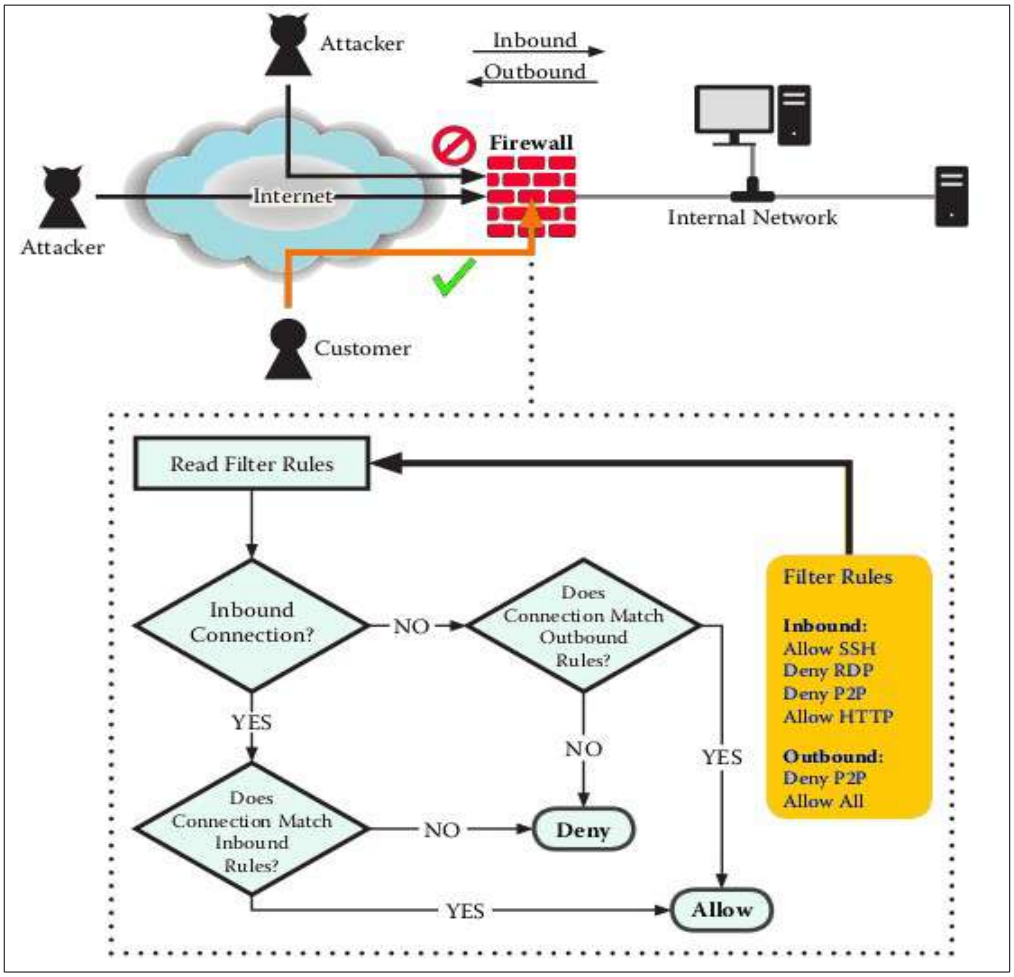

Figure 2-1 : Firewall implementation

II.3.2 -Packet-Filtering Firewalls

The most rudimentary of firewalls is the packet-filtering firewall. Packet-filtering firewalls work at the IP level of the network. Most routers integrate this type of firewall to perform basic filtering of packets based on an IP address. The principle behind packet-filtering firewalls is that the firewall bases the decision to allow a packet from one network into another network solely on the IP address of the source and destination of the packet. For instance, if the firewall administrator defines the following packet-filtering rules, and if a packet from host 1.1.1.1 is destined for host 2.2.2.2, the firewall allows the packet to pass.

ALLOW host 1.1.1.1 to host 2.2.2.2

DENY ALL

In this example, the firewall administrator has placed an ALLOW rule after the DENY ALL rule. This situation would prevent the processing of the last ALLOW rule.

II.3.3 -Stateful Firewalls

Simple packet-filtering firewalls suffer from one significant downside: they do not take into consideration the state of a connection, only the endpoints of the connection. Stateful firewalls allow only properly established connections to traverse the firewall's borders. While packet filtering is still a key element of these firewalls, the firewall also pays attention to the state of the connection.

II.3.4 -Application Gateway Firewalls

Application gateway firewalls, also known as proxies, are the most recent addition to the firewall family. These firewalls work in a similar manner to the stateful firewalls, but instead of only understanding the state of a TCP connection, these firewalls understand the protocol associated with a particular application or set of applications. A classic example of an application gateway firewall is a Web proxy or e-mail-filtering proxy. A Web proxy, for instance, understands the proper HTTP protocol and will prevent an improperly constructed request from passing. Likewise, an e-mail-filtering proxy will prevent certain e-mails from passing based on predefined conditions or heuristics (for example, if the e-mail is spam). These proxies also prevent unknown protocols from passing through. For example, a properly configured HTTP proxy will not understand an SSH connection and will prevent the establishment of the connection see Figure below :

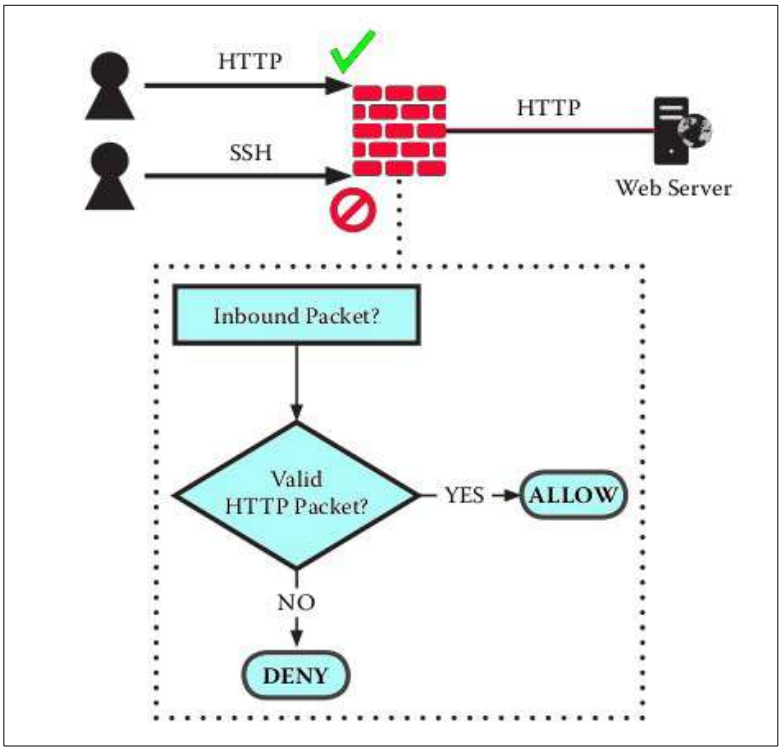

Figure 2-2 : Block SSH connection

II.4 -Proxy Server

A proxy server, also known as a "proxy" or "application-level gateway", is a computer that acts as a gateway between a local network (for example, all the computers at one company or in one building) and a larger-scale network such as the internet. Proxy servers provide increased performance and security. In some cases, they monitor employees' use of outside resources. A proxy server works by intercepting connections between sender and receiver [19]. All incoming data enters through one port and is forwarded to the rest of the network via another port. By blocking direct access between two networks. It is also used by the hacker to hide their traces on the Internet It is also used by users to bypass government blocking services on some Internet services.

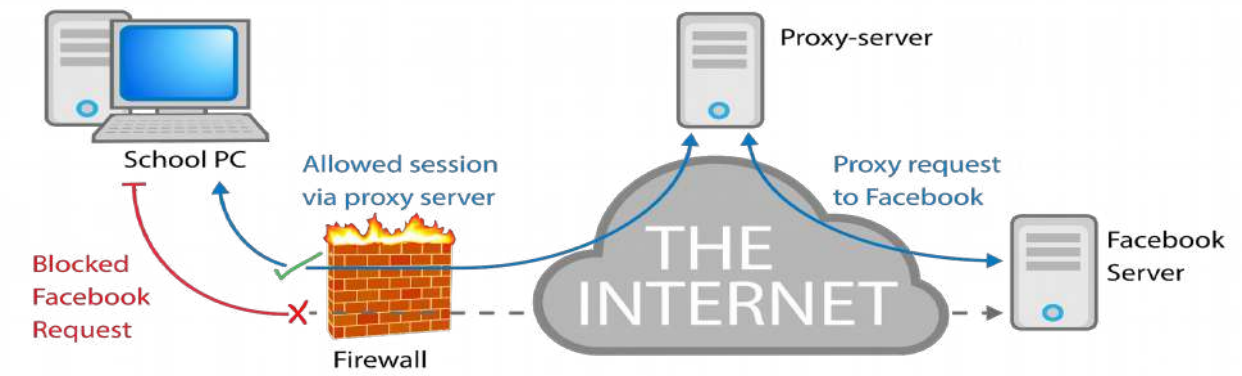

Figure 2-3 : Proxy Server workflow

II.4.1 -How Proxy Server Works?

Whenever the client connects to a web proxy server and makes a request for the resources for example, **example.html** that reside on a remote server **xyz.com**, the proxy server forwards this request to the target server on behalf of the client, so as to fetch the requested resource and deliver it back to the client. An example of client can be a user operated computer that is connected to the Internet.

II.4.2 -Reverse Proxy

As its name implies, a reverse proxy does the exact opposite of what a forward proxy does. While a forward proxy proxies in behalf of clients or requesting hosts, a reverse proxy proxies in behalf of servers. A reverse proxy accepts requests from external clients on behalf of servers stationed behind it just like what the figure below illustrates. The client is oblivious to the file transfer servers behind the proxy, which are actually providing those services. In effect, whereas a forward proxy hides the identities of clients, a reverse proxy hides the identities of servers.[20]

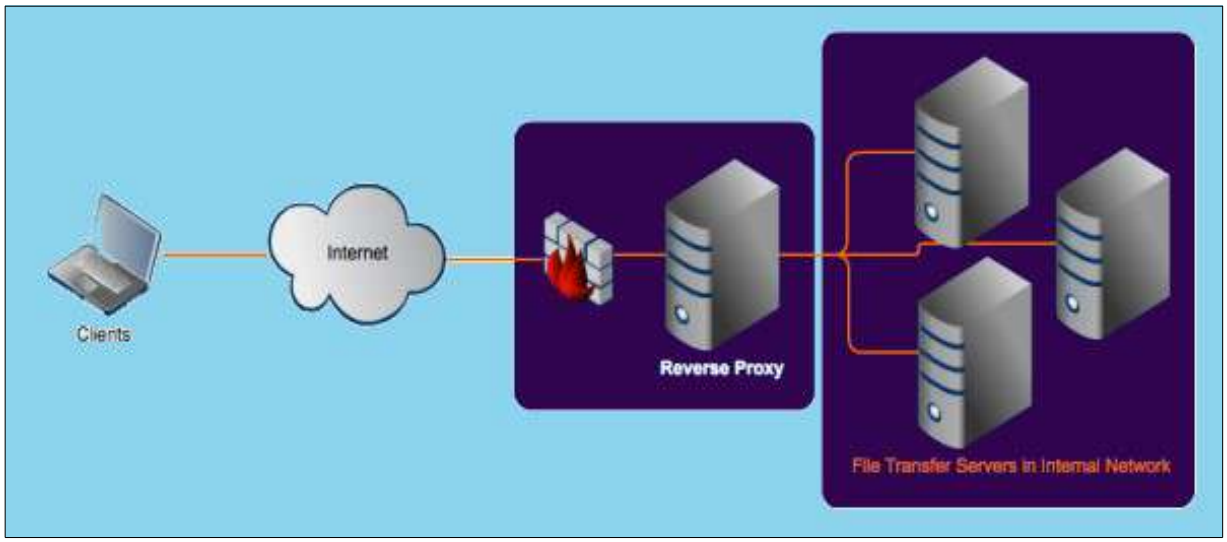

Figure 2-4 : Reverse Proxy behind internet network

Reverse proxy more security and performance.

An Internet-based attacker would therefore find it considerably more difficult to acquire data found in those file transfer servers than if he wouldn't have had to deal with a reverse proxy. In most cases, reverse proxy servers also act as load balancing for the servers behind it. Load balancing play a crucial role in providing high availability to network services that receive large volumes of requests. When a reverse proxy performs load balancing, it distributes incoming requests to a cluster of servers, all providing the same kind of service. So, for instance, a reverse proxy load balancing FTP services will have a cluster of FTP servers behind it.

II.4.3 -How Hackers Cover Their Tracks

Masking one's IP address is a standard practice when conducting illicit activities [19]. A well-configured proxy provides robust anonymity and does not log activity, thereby frustrating law enforcement efforts to identify the original location of the person(s) involved. Proxies are useful to attackers in many ways. Most attackers use proxies to hide their IP address and, therefore, their true physical location. In this way, attackers can conduct fraudulent financial transactions, launch attacks, or perform other actions with little risk. While law enforcement can visit a physical location identified by an IP address, attackers that use one or multiple proxies are more difficult to locate.

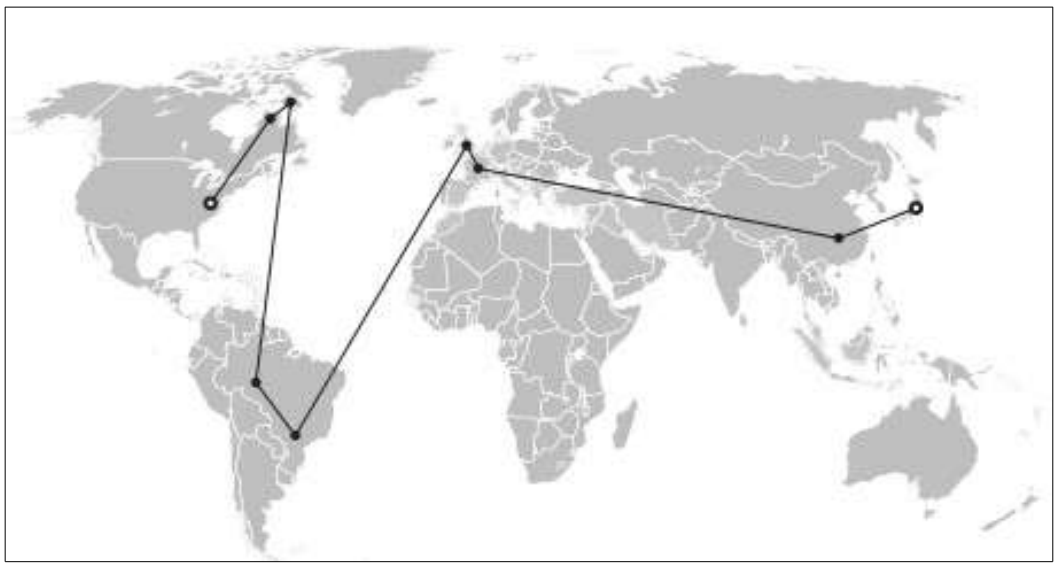

Figure 2-5 : Map of proxy chain

Here are some popular and powerful proxy types used by hackers and users to cover their traces and protect privacy.

II.4.3.A -The Onion Router Tor

Some lightweight proxies are written in scripting languages, which run with an HTTP server and are easier for attackers to modify [21]. Application proxies require configuration. Some applications either do not operate correctly through proxy services because the proxy server removes necessary information or cannot satisfy the request. Some services like The Onion Router Tor also give users the ability to proxy traffic and hide their original location from victims.

II.4.3.B -Virtual Private Network VPN

VPN acts as a more versatile proxy and supports more security features [15]. Instead of configuring the application to use a proxy, users can tunnel all traffic through the VPN ,it services

usually support strong authentication and are less likely to leak information that could identify the user of a proxy.

II.5 -IDS/IPS system

Intrusion Detection Systems IDS and Intrusion Prevention Systems IPS are both parts of the network infrastructure. IDS/IPS compare network packets to a Cyber threat database containing known signatures of Cyber attacks and flag any matching packets. The main difference between them is that IDS is a monitoring system, while IPS is a control system [1]. IDS doesn't alter the network packets in any way, whereas IPS prevents the packet from delivery based on the contents of the packet, much like how a firewall prevents traffic by IP address.

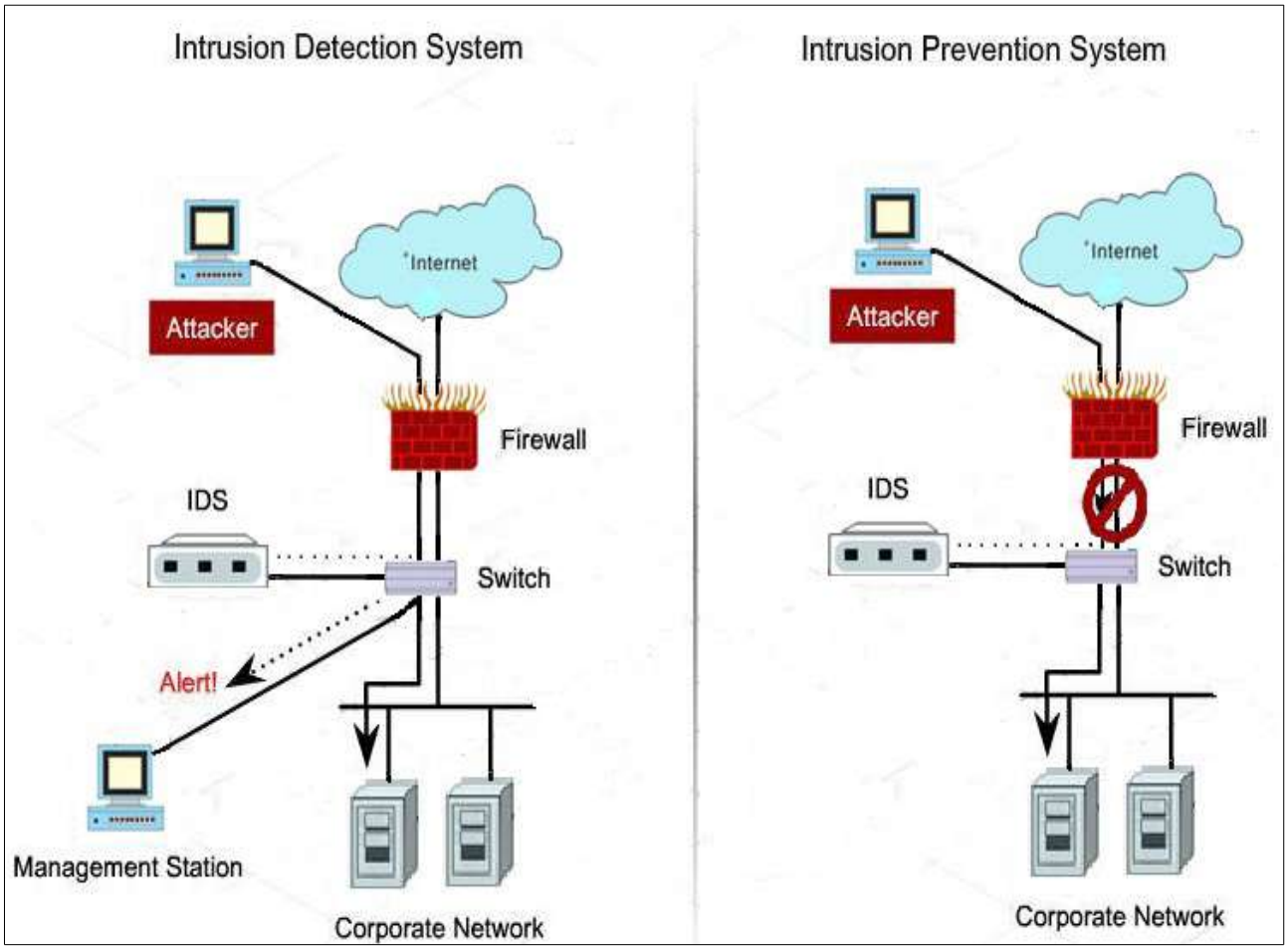

Figure 2-6 : IDS and IPS inside network

II.5.1 -Intrusion Detection System

An Intrusion Detection System (IDS) is a mechanism that sneaks into network traffic to identify abnormal or suspicious activity and thereby a prevention action on the risks of intrusion.

There are two major distinct families of IDS :

- Network Based Intrusion Detection System (N-IDS): they provide security at the network level.
- Host Based Intrusion Detection system (H-IDS): they provide security at the host level.

II.5.1.A -Network-IDS

An N-IDS requires dedicated hardware and is a system capable of controlling packets on one or more network links in order to discover whether a malicious or abnormal act is occurring. The N-IDS places one or more network interface cards of the dedicated system in promiscuous mode. It is common to find several IDS on the different parts of the network and in particular to place a probe outside the network to study attacks attempts and a probe internally to analyze the requests that crossed the screen -fire or conducted from the inside.

II.5.1.B -Host-IDS

The H-IDS resides on a particular host and the range of these software thus covers a large part of the operating systems such as Windows, Solaris, Linux, HP-UX, Aix, etc.The H-IDS analyzes particular information in log logs and also captures incoming / outgoing network packets from the host for intrusion signals (denial of service, backdoors, Trojans, non-access attempts). allowed, malicious code execution, buffer overflow attacks, etc..).[1]

II.5.1.C -Detection techniques

Network traffic is usually made up of IP datagrams. An N-IDS is able to capture packets as they travel on the physical links it is connected to. An N-IDS consists of a TCP / IP stack that reassembles IP datagrams and TCP connections. He can apply the following techniques to recognize intrusions :

- Checking the protocol stack (packet inspection).
- Verification of application protocols (protocol analysis).
- Recognition of pattern matching attacks (signature search)

II.5.1.D -Alert methods

The main methods used to report and block intrusions on N-IDS are :

- **Reconfiguration of third-party equipment** (firewall, ACL on routers) command sent by the N-IDS to a third party device (packet filter, firewall) for an immediate reconfiguration in

order to block an intruder. This reconfiguration is possible by passing information detailing an alert (at the top of packets).

- **Sending a SNMP trap to a third-party hypervisor:** sending the alert (and the details of its constituent information) in the format of an SNMP datagram to a third-party console.
- **Sending an email to one or more users:** sending an email to one or more mailboxes to notify a serious intrusion.
- **Log attack:** Backup details of the alert in a central database such as the following information: (timestamp, IP address of the intruder, IP address of the target, protocol used, payload).
- **Backup of suspicious packets:** backup of all captured packets and or only the packets that triggered an alert.
- **Starting an application:** launching an external program to perform a specific action (sending an SMS message, sending a hearing alert ...).
- **Sending a ResetKill:** construction of a TCP FIN packet to force the end of a connection (only valid on intrusion techniques using the TCP transport protocol).
- **Visual notification of the alert:** display of the alert in one or more management consoles.

II.5.2 -Intrusion Prevention System

An intrusion prevention system (IPS) is a tool for information system security specialists, similar to IDS, to take measures to reduce the impact of an attack. It is an active IDS, it detects an automated scan, the IPS can block the ports automatically. IPS can therefore counter known and unknown attacks. Like IDS, they are not 100% reliable and may even be false positive if they block legitimate traffic.There are two major distinct families of IPS :

II.5.2.A -Network-IPS

The network intrusion prevention system (NIPS) is an IPS that monitors network traffic and can take actions such as terminating a TCP session. A declination in WIPS (wireless intrusion prevention system) is sometimes used to evoke the protection of wireless networks.

II.5.2.B -Host-IPS

The host-based intrusion prevention system (HIPS), which is an IPS for monitoring the workstation through different techniques, monitors processes, drivers, .dll etc. In case of detection

of suspicious process HIPS can kill him to stop his actions. HIPS can therefore protect against buffer overflow attacks. There are also kernel intrusion prevention systems (KIs) that detect any intrusion attempts at the kernel level, but they are less used.

II.6 -Security protocols

A security protocol is an abstract or concrete protocol that performs a security function and implements encryption methods, often as primary cryptographic sequences [22]. The protocol describes how to use algorithms. The protocol includes detailed enough details about data structures and representations, and at this point it can be used to implement multiple interoperable versions of the program.

II.6.1 -SSL Protocol

SSL (Secure Sockets Layers), which could be translated by layer of secure sockets is a method of securing transactions made via the Internet. The SSL standard was developed by Netscape, in collaboration with Mastercard, Bank of America, MCI and Silicon Graphics. It is based on a key cryptography method to ensure the security of the transmission of data on Internet. Its principle is to establish a communication channel secure (encrypted) between two machines (a client and a server) after an authentication step.

II.6.2 -HTTPS Protocol

Hypertext Transfer Protocol Secure HTTPS is an extension of the Hypertext Transfer Protocol HTTP. It is used for secure communication over a computer network, and is widely used on the Internet.In HTTPS, the communication protocol is encrypted using Transport Layer Security TLS, or, formerly, its predecessor, Secure Sockets Layer SSL. The protocol is therefore also often referred to as HTTP over TLS,or HTTP over SSL.The principal motivation for HTTPS is authentication of the accessed website and protection of the privacy and integrity of the exchanged data while in transit [23]. It protects against man-in-the-middle attacks. The bidirectional encryption of communications between a client and server protects against eavesdropping and tampering of the communication. In practice, this provides a reasonable assurance that one is communicating without interference by attackers with the website that one intended to communicate with, as opposed to an impostor.

Unlike SSL working at the transport layer, HTTPS provides message based security over the HTTP protocol, by individually marking the HTML documents to using certificates. While SSL is

application independent used and encrypts the entire communication, HTTPS is very strongly bound to the HTTP protocol and individually encrypts each message.

HTTPS messages are based on three components :

- the HTTP message
- cryptographic preferences of the sender
- the recipient's preferences.

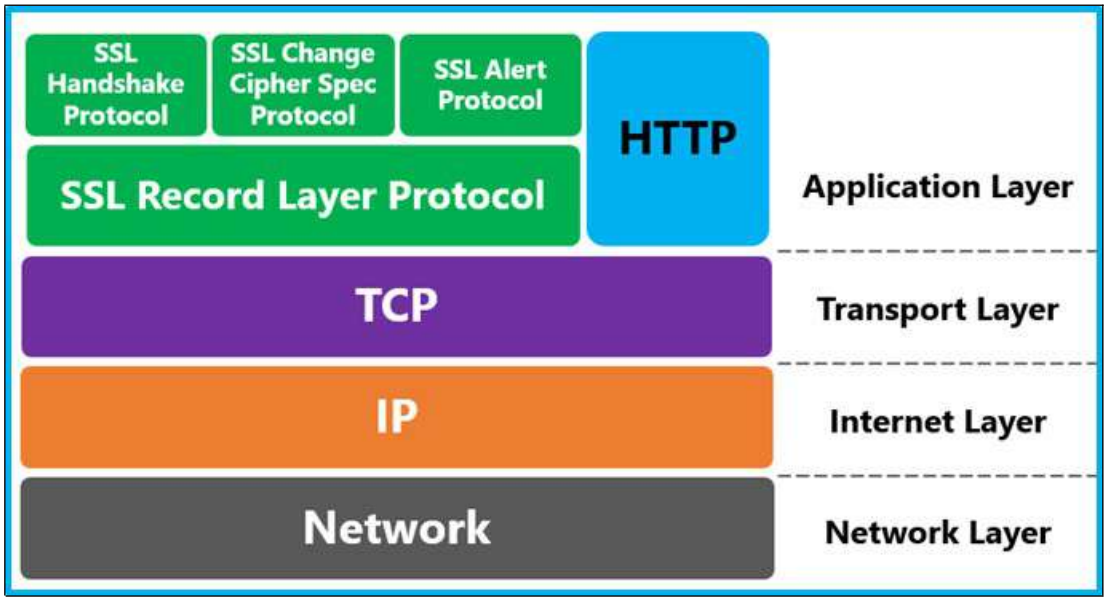

Figure 2-7 : Security HTTPS on TCP/IP model

II.6.3 -DNS Security

DNS servers are computers that structure the Web world. The players in this sector have therefore decided to secure transmissions from these servers to each other [22]. That's what the tent is trying to do DNSsec secure protocol (Domain Name System Security Extensions). DNSsec is one of the extensions of the DNS protocol and is supposed to ensure the authentication and integrity of the records of the DNS. Its operation is based on signature checks digital and encrypted data exchange. DNSsec thus allows to constitute a secure identification chain. The user could therefore better identify if this or that subdomain of a website is well the one he wants to address. DNSsec could be set up from systematic way to counter the flaws of the DNS protocol and the phishing phenomenon.

How Does DNSSEC Work?

DNSSEC helps ensure you’re reaching the site you intended to visit by using public keys and digital signatures to verify data.It does so by adding new records to the DNS settings following:

- **RRSIG:** holds cryptographic signatures.

- **DNSKEY:** holds public signing keys.
- **DS:** holds hashes of DNSKEY records.
- **NSEC and NSEC3:** provides denials-of-existence of DNS records.
- **CDNSKEY and CDS:** facilitates DS update requests between child and parent Zones

These records can be accessed in the same way as a regular DNS record such as a CNAME or A record however they are used to digitally sign a domain. DNSSEC also involves two main types of keys :

1. **Zone-signing keys, or ZSKs:** contain both a public and private key portion and validate specific record sets within a Zone
2. **Key-signing keys, or KSKs:** sign DNSKEY records

Each signed nameserver possesses one public key and one private key. When a client makes a request, the data they transmit is signed with a private key, which the recipient then opens with a public key. If a third party attempts to intervene without the public key, the recipient knows that the data is fraudulent. Because DNSSEC doesn't come with any encryption algorithms, it can't provide data confidentiality, it just helps the DNS server verify the authenticity of data requests.

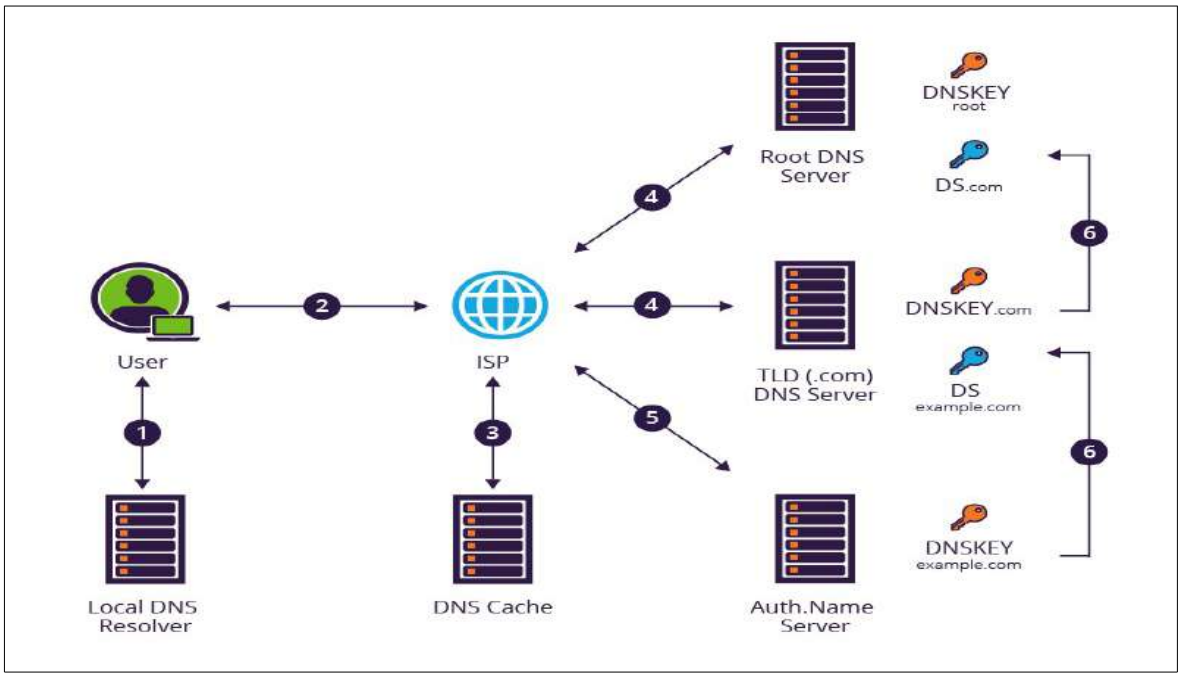

Figure 2-8 : Security DNS query and response over DNS security

II.6.4 -MIMES Protocol

MIMES (Secure Multipurpose Intenet Mail Extension) that could be translated by email extensions to multiple purposes and secure is a method of securing e-mail exchanges to ensure the confidentiality and non repudiation of electronic messages. MIMES is based on the MIME standard, which aims to allow to include attached files in email messages other than text files ASCII. It is thus thanks to the MIME standard that it is possible to add attachments of all types to emails.

The MIMES standard is based on the principle of encryption public key. MIMES makes it possible to encrypt the content of messages but do not encrypt the communication. The different parts of a electronic message, coded according to the MIME standard, are each encrypted is using a session key. In each header is inserted the session key, encrypted using the recipient's public key. Only the recipient can thus open the body the message, using its private key, which ensures confidentiality and the integrity of the message received.By the way, signing the message is encrypted using the private key of the sender. Every person intercepting the communication can read the contents of the signature of the message, but this ensures the recipient's identity the sender, because only the sender is able to decrypt the message with his private key.

II.6.5 -SSH Protocol

The Internet makes it possible to carry out a large number of remote operations, including the administration of servers or the transfer of files. The telnet protocol and BSD r-commands (rsh, rlogin and rexec) to perform these remote tasks have the disadvantage major to circulate in clear on the network the information exchanged, including the login and password for access to the remote machine. The SSH (Secure Shell) protocol addresses this problem by allowing users (or TCP / IP services) to access a machine through an encrypted communication called tunnel.[22]

Principle of operation

The SSH protocol was developed in 1995 by the Finn Tatu Ylönen.This is a protocol that allows a client (a user or a machine) to open an interactive session on a machine remote it called server to send commands or files from secure way :

- The data flowing between the client and the server is encrypted, which guarantees their confidentiality nobody else but the server or the client can not read the information transiting on the network.It is therefore not possible to listen to the network using a frame analyze.

- The client and the server mutually authenticate each other so to ensure that the two machines that communicate are well the ones that each of the parties think they are. he is no longer possible for a hacker to impersonate the client or server (spoofing).

SSH connection

Establishing an SSH connection is done in tow steps :

1. At first the server and the client identify themselves mutually to put in place a secure channel.
2. In a second step, the client authenticates with the server to get a session.

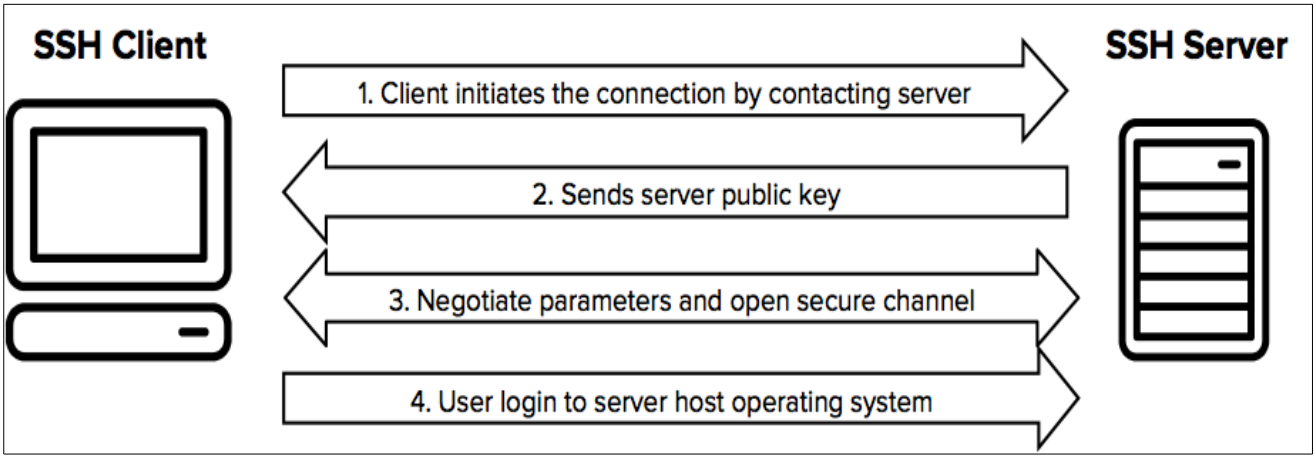

Figure 2-9 : Connection security over SSH client and server

II.6.6 -FTPS Protocol

File Transfer Protocol with SSL Security (FTPS) is an extension to the FTP protocol that adds Secure Socket Layer (SSL)/Transport Layer Security (TLS)-based mechanisms/capabilities on a standard FTP connection.It mainly enables performing or delivering standard FTP communication on top of an SSL-based security connection.FTPS is mainly used to provide secure server to server communication.however, it can also be used to access a server from desktop or end-user devices.
FTPS uses a combination of symmetric DES/AES encryption and asymmetric RSA (Rivest-Shamir-Adleman) and DSA (Digital Signature Algorithm) algorithms to deliver security and uses X.509 certificates for authentication [24].FTPS is delivered in two different forms :

- **Explicit FTPS:** Selected parts or components for communication are encrypted.connection starts on standard FTP port 21, switches to SSL or TLS based on FTP client requesting SSL encryption via AUTH SSL or AUTH TLS command respectively. Standards compliant to RFC 2228 - FTP Security Extensions.

- **Implicit FTPS:** All communications are encrypted.FTP connection starts on a designated port (usually 990),SSL is started at the beginning of the connection

Potential standards are covered in several Internet drafts. Explicit SSL should be used where standards compliance is mandated.

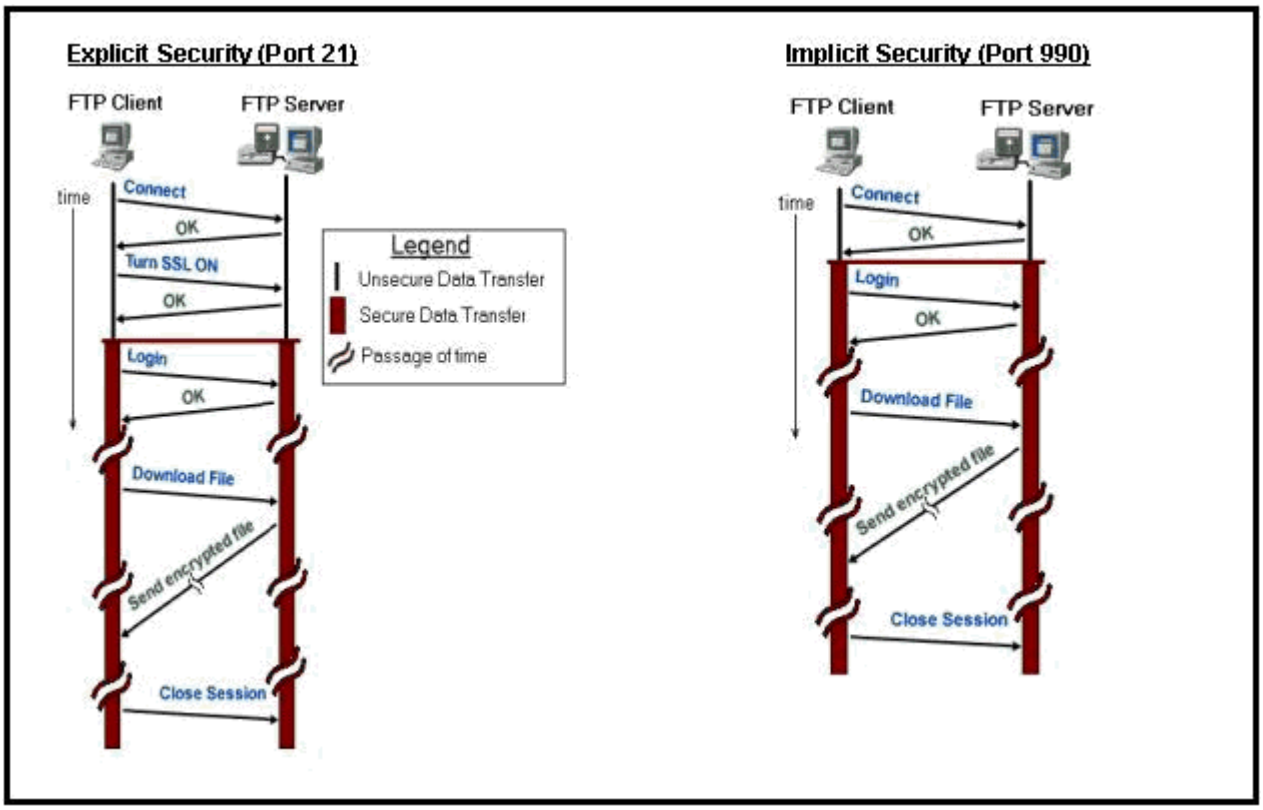

Figure 2-10 : Method connections enter different form's protocol FTPS security

II.6.7 -DHCP Protocol

DHCP is an abbreviation Dynamic Host Configuration Protocol. This protocol allows a computer that connects on a local network to automatically obtain its configuration. he Just tell your computer to use DHCP. The main goal being simplifying the administration of a network.DHCP is primarily used to distribute IP addresses over a network, but it has been originally designed as a complement to the Bootstrap Protocol (BOOTP) which is used to launch an installation OS over the network by downloading the necessary files by TFTP. A DHCP server can therefore return host-specific BOOTP or configuration parameters given.

How Dynamic Host Configuration Protocol (DHCP) works?

The Dynamic Host Configuration Protocol (DHCP) client TCP/IP software is not configured with a static IP address and it is configured to obtain an IP address dynamically from a DHCP Server. When a DHCP client device boots up, it not capable send and receive network traffic, because TCP/ IP is not configured. But it can participate in broadcast traffic. DHCP Clients and DHCP Servers

uses broadcast messages to communicate with each other. The scope of a broadcast message is only within the local broadcast domain. Broadcast messages will never cross the router to reach another network, because routers drop limited broadcast IP Address. Let us explain how the steps are taken :

- Host connecting to network (cable or wireless) sends DHCP discover message to all hosts in Layer 2 segment (destination address is FF:FF:FF:FF:FF:FF). Frame with this **DISCOVER** message hits the DHCP Server.
- After the DHCP Server receives discover message it suggests the IP addressing **OFFERING** to the client host by unicast.
- Now after the client receives the offer it requests the information officially sending **REQUEST** message to server this time by unicast.
- Server sends **ACKNOWLEDGE** message confirming the DHCP lease to client. Now client is allowed to use new IP settings.

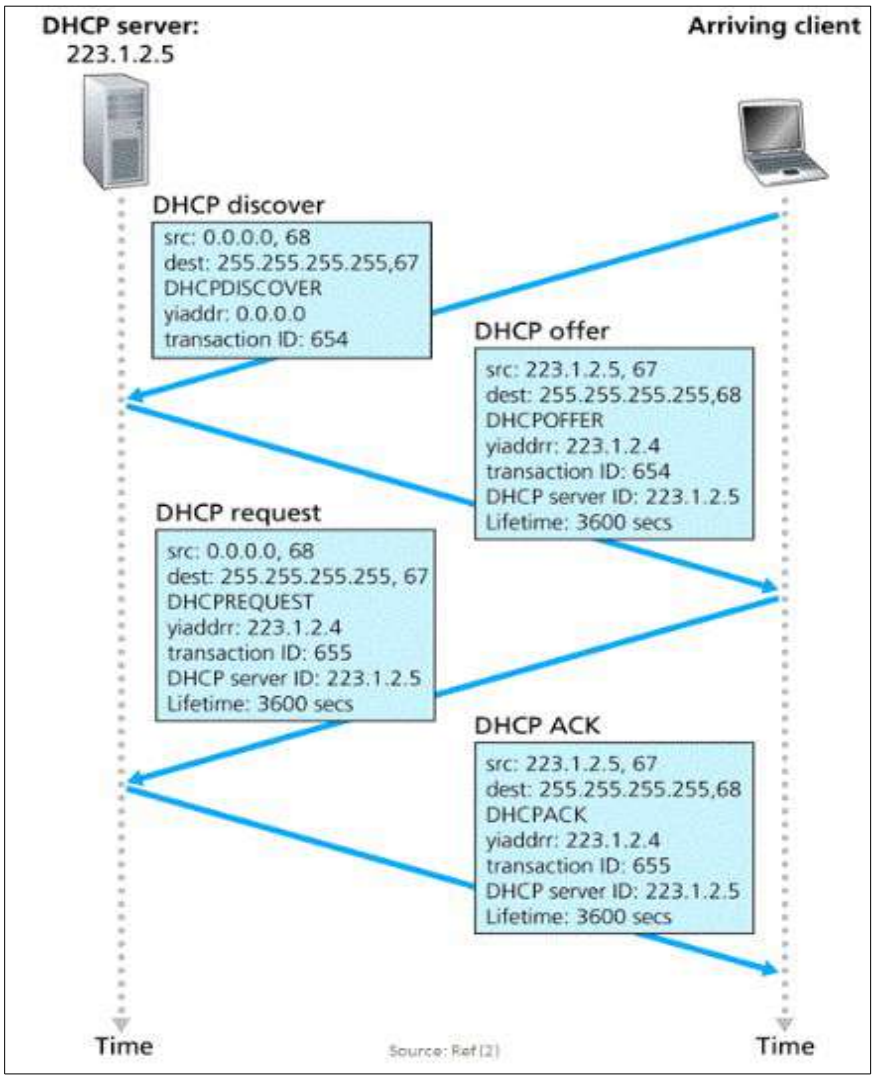

Figure 2-11 : Describe DHCP Functionality

II.7 -Conclusion

In this chapter, we discussed the basic defense techniques used in the institutions and discussed some of the characteristics, features and techniques used to minimize the damage that could be caused by hackers. However, what we have addressed is a simplified form of basic protection mechanisms. In the practical field, the protection aspect should be taken into account more deeply, strict protection policy measures and analysis of many things. And follow every new happening in this area because the failure to put security procedures may lead to the company or users to collapse and big loss . This is why huge sums of money have been spent in this domain.

III Chapter 3

Design and realization Mechanism of protection against MITM attack

III.1 -Introduction

In this chapter, we will represent the realization of our application, starting with the development and testing tools , the methods and mechanisms of protection used in it, Then we will give details about the experiments, and the results obtained in detecting MITM attack.

III.2 -Tools and methods used

III.2.1 -Python programming language

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics [25]. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

- Python is easy to learn and use. It is developer-friendly and high level programming language.
- Python language is more expressive and interpreted language.
- Python can run equally on different platforms such as Windows, Linux, Unix and Macintosh etc. So, we can say that Python is a portable language.
- Python language is freely available at offical web address.The source-code is also available. Therefore it is open source.
- Object-Oriented Language and extensible.
- Large Standard Library and GUI Programming Support
- In 2018/2019 Python is the most widely used in the world and is constantly increasing according to statistics.
- Used in artificial intelligence and by big company such as NASA and IBM,reddit,etc..

III.2.2 -Scapy python programming

Scapy is the basis that we depend it in development our application,so what is scapy?

Scapy is a Python program that enables the user to send, sniff and dissect and forge network packets. This capability allows construction of tools that can probe, scan or attack networks. In other words, Scapy is a powerful interactive packet manipulation program. It is able to forge or decode packets of a wide number of protocols, send them on the wire, capture them, match requests and replies, and much more. Scapy can easily handle most classical tasks like scanning, tracerouting, probing, unit tests, attacks or network discovery. It can replace hping, arpspoof, arp-sk, arping, p0f and even some parts of Nmap, tcpdump, and tshark.[26]

Scapy also performs very well on a lot of other specific tasks that most other tools can't handle, like sending invalid frames, injecting your own 802.11 frames, combining techniques (VLAN hopping+ARP cache poisoning, VOIP decoding on WEP encrypted channel,etc …),It is used by security researchers and hackers to build their tools.

Note : scapy runs natively on Linux, and on most Unixes with libpcap and its python wrappers

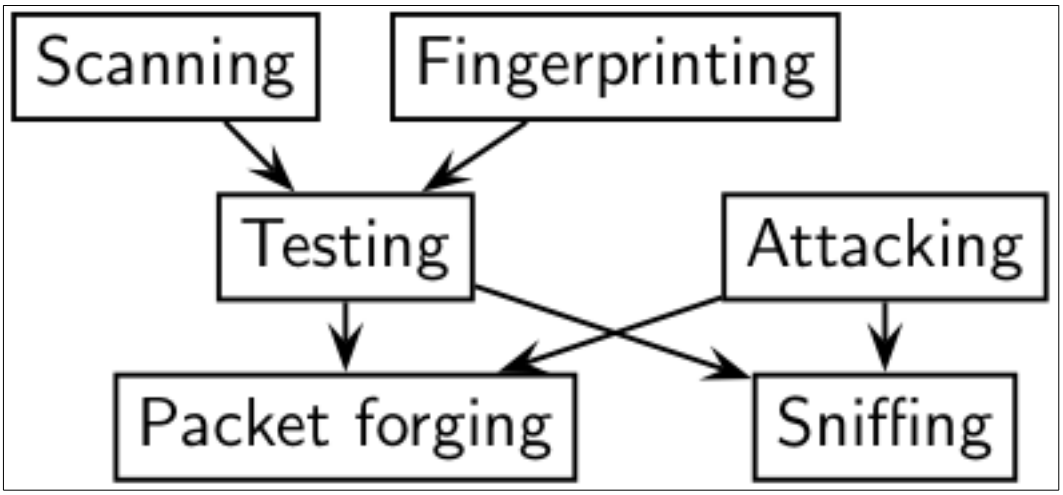

Figure 3-1 : Functionality Base on Scapy

III.2.3 -Wireshark

Wireshark is the world's foremost and widely-used network protocol analyzer [27]. It lets you see what's happening on your network at a microscopic level and is the de facto (and often de jure) standard across many commercial and non-profit enterprises, government agencies, and educational

institutions. Wireshark development thrives thanks to the volunteer contributions of networking experts around the globe and is the continuation of a project started by Gerald Combs in 1998.

Wireshark has a rich feature set which includes the following :

- Deep inspection of hundreds of protocols, with more being added all the time.
- Live capture and offline analysis.
- Standard three-pane packet browser.
- Multi-platform: Runs on Windows, Linux, macOS, Solaris, FreeBSD, NetBSD, and many others .
- Captured network data can be browsed via a GUI, or via the TTY-mode TShark utility.
- The most powerful display filters in the industry.
- Rich VoIP analysis.
- Capture files compressed with gzip can be decompressed on the fly and more.

III.2.4 -Kali linux test machine

Kali Linux is a popular penetration testing environment built on Debian distribution [28],Contains hundreds of powerful tools such as Metasploit-framework and wireshark,maltego,Nmap,ect. It created by Offensive Security company, the tools used to perform MITM attacks are as follows :

- **Ettercap-framework:** This is one of the most powerful tools currently used by hackers to perform MITM attacks,via GUI or command line.
- **arpspoof:** also used to execute arp attack spoofing,Via command line.
- Enable forwarding mode in kernel machine.
- **iptables:** is the firewall used in Linux that we will use to redirect communications,via the command line.

Note : ettercap implement all kind of MitM attack we need perform (arp,dns,dhcp) spoofing and enumerate devices in local network sslstrip for skipping sll encryption.

III.3 -Protect and detect methods

In this section we go processing solution to analyze malicious network activity. This application only supports Linux systems.

III.3.1 -ARP Poisoning Detect

Is an algorithm designed to detect Spoof ARP packets and scan network. This algorithm is based on the principle of capturing ARP packets, and then analyzes and checks them if there is a strange movement in the network. It can work in network mode, which analyzing all traffic in the network general ,or In host mode,which analyzing of only the ARP packets of the request type and response to the host IP address in particular, here steps as follows :

Detect Spoof ARP :

1. sniff ARP packets.
2. create the ARP Cache to contain all the IP addresses in the network corresponding to the MAC.
3. The monitoring is maintained as long as the connection is maintained.
4. The magic way to detect Spoof ARP packets is to compare the source IP address and its MAC in the ARP cache for program.
5. if an IP address already exists for a ARP packet but has a new MAC address that does not agree with the ARP Cache, we conclude that the packet is forged.
6. Specifies the attacker's MAC address.
7. Show results on screen.

III.3.1.A - Detect Network scan :

Before implementing Spoof ARP, the attacker will send hundreds of ARP request packets to the network's range to determine his targets. Based on the attacker's technique, we have created a method that reveals this behavior as follows :

1. Fill in a list of ARP packets.
2. Fill in another list of time values for ARP packets.
3. Check if the number of ARP packets is large and the time between packets is very short,This means the scanning process exists now.
4. Determine the IP address of the attacker.
5. Show results on screen.

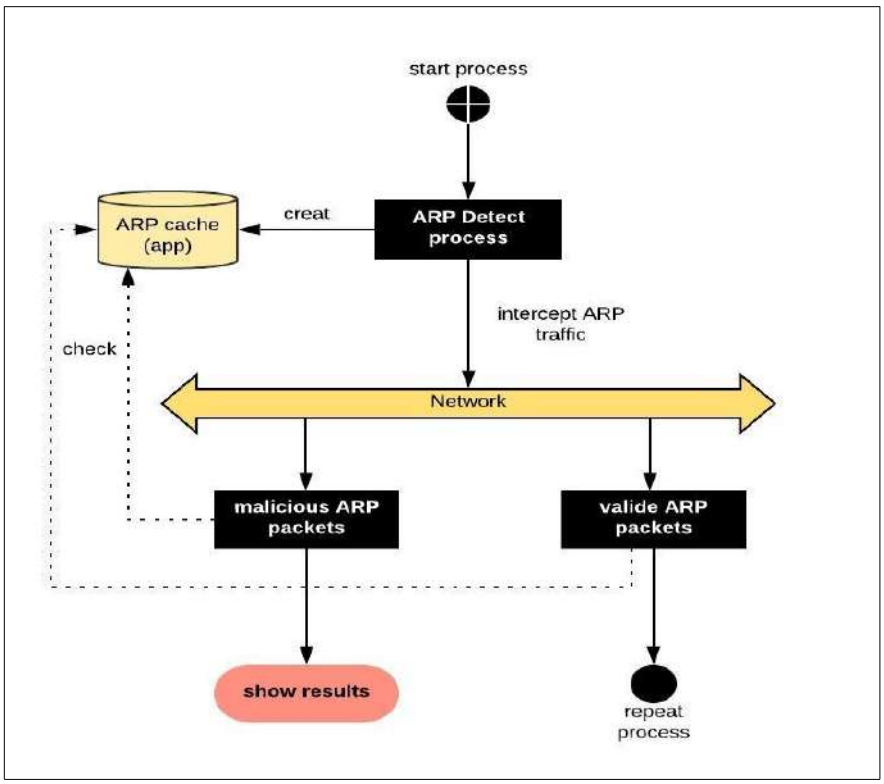

Figure 3-2 : Diagram describe of ARP-D method

III.3.2 -DNS Spoof Detect

The algorithm captures DNS packets, and then inspect them through mechanisms we going to explain in order to detect if there are malicious DNS response packets. It can work in network mode, which analyzing all DNS traffic in the network in general ,or In host mode,which analyzing of only the DNS packets of the response type to the host IP address in particular,here steps as follows :

1. Sniff DNS packets.
2. Collect approximately 100 DNS response packets and store them in a list.
3. Inspect the DNS records of type A and obtain the domain IP address for each packet.
4. Compare the IP addresses of the domains with each other.
5. If a duplicate IP address is found with a different domain, it means spoofing DNS responses for many domains.
6. Show results on screen.

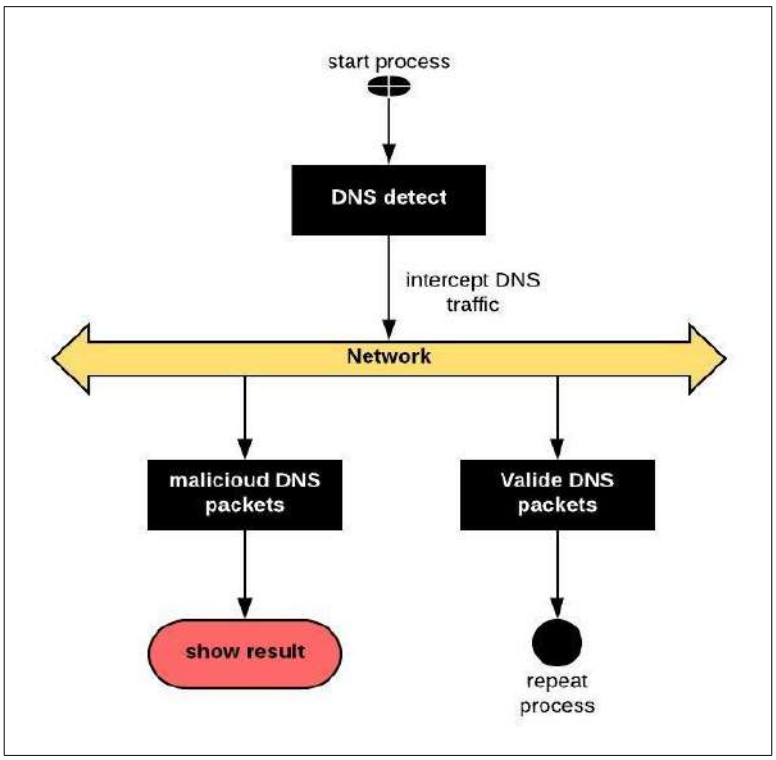

Figure 3-3 : Diagram describe of DNS-D method

III.3.3 -DHCP Spoof Detect

This algorithm works in two different ways depending on the specific mode,In network mode where you also intercept, examine, and analyze DHCP packets, Before the actual DHCP attack, the attacker sends thousands of DHCP requests to the router to consume the range of the IP addresses. Through this movement we can expect a DHCP spoofing attack. In the host mode, we perform two operations. The steps are as follows :

III.3.3.A -Network-DSD

In network mode, we can choose one of the following two processes.

- The first operation :

  1. sniff DHCP packets.
  2. Compare the IP address of a real DHCP server and DHCP packets captured from the DHCP offer or DHCP ACK type of packet.
  3. If there is a difference in IP addresses with a real DHCP server, we conclude that there are DHCP Spoof packets.
  4. Show results on screen.

- The second operation :

  1. Capture DHCP request packets and store them in a list.
  2. Calculate the time difference between packets.
  3. If there are a large number of DHCP packets and the time difference between them is very short, it means that it is a MAC Flooding attack.
  4. Show results.

III.3.3.B -Host-DSD

In host mode, the two operations will be performed together for better results.

1. We perform a traceroute and then check the second node if it has a private IP.
2. We then examine the first node device by performing the OS detection option scan.
3. If the operating system does not refer to the router it means that we are in a DHCP Spoofing attack.

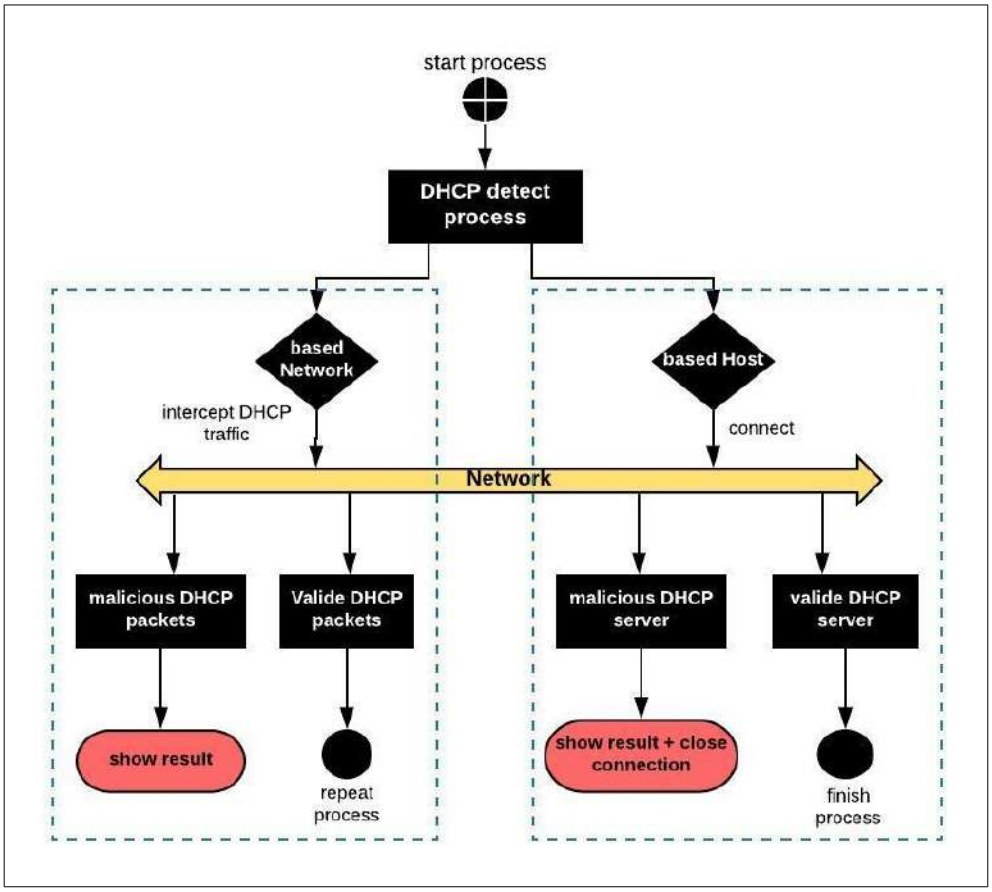

Figure 3-5 : Diagram describe of DHCP-D method

III.4 -Experimentation and Results

After the discovery methods were exposed to attack in three different main types in this section we will discuss experiments we have done for each method and show the results in detail in screenshots and form of tables to experiments conducted in several case study on different transmissions medium and network modes.

Running the application on Linux Terminal specified by the correct path to application directory, the directory contain app files, the root user is required for execution. see the following example :

root@kali$:sudo python main_MITM.py –help // option help show manual user

III.4.1 -Result of ARP Poisoning Detect

In order to get the desired results, we simulate hacker attacks through the most frequently used steps.

Note : We assume that we know the IP address.

- Enable forwarding Mode in Attacker machine.
- Redirect packets in iptables to a specific port.
- Run sslstrip to downgrade HTTPS to HTTP.
- Run arpspoofing by ettercap tool.

These are the results obtained by the application.

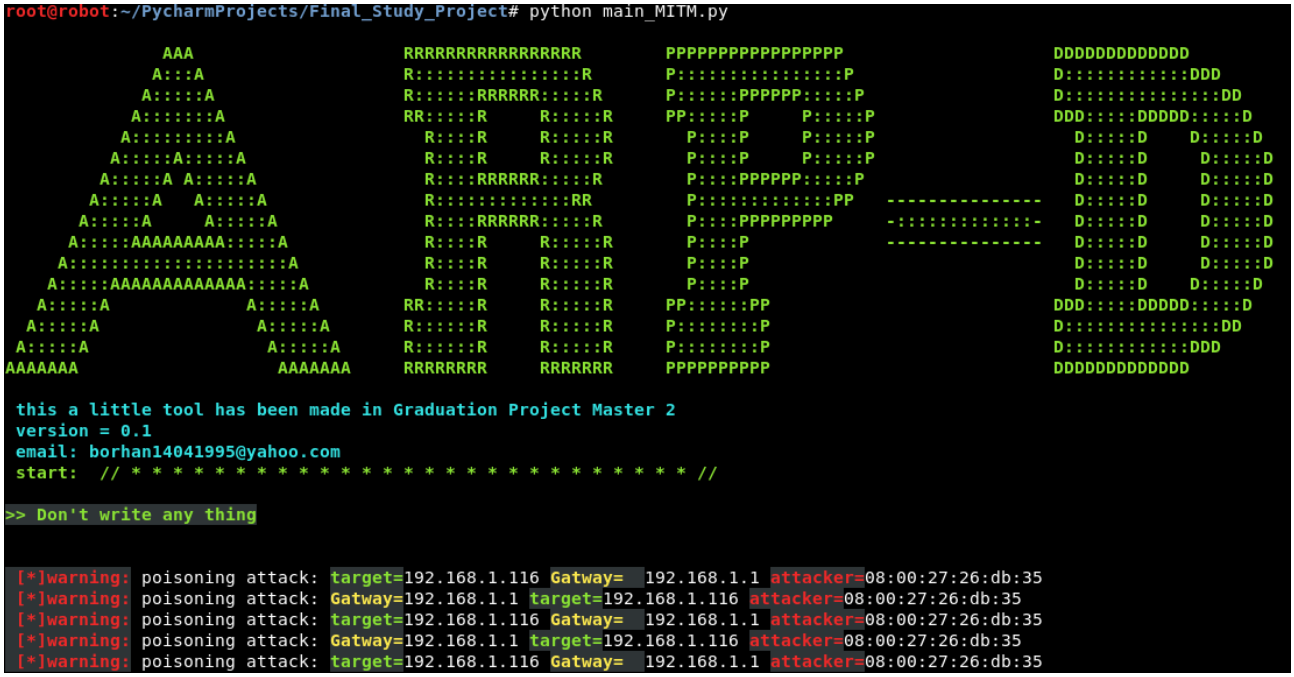

Figure 3-6 : Describe result of ARP-D method

This result is shown as the application works whatever mode, when we run ARP_Detect.py file, show us this interface in the terminal screen and then start to intercept all the traffic of ARP packets, when the traffic is normal will not show any results. currently it can detect the ARP spoof attack , here We have warning messages in red where each message corresponds to a captured malicious packet and the information in the message is as follows :

- ➔ In this scenario, the attacker sends a spoofed ARP replay to the target, ،in this case we are and my address is two 192.168.1.116.
- ➔ target = 192.168.1.116 ,Gateway =192.168.1.1 and MAC address of attacker machine.
- ➔ In this scenario the attacker sends a spoofed ARP replay to Gateway which is my router address 192.168.1.1.
- ➔ Gateway =192.168.1.1,target = 192.168.1.116 and MAC address of attacker machine.
- ➔ Messages continue to appear as long as the attack continues.

This table describes the number of trial times in different cases followed by statistics results :

	Based on host wireless	Based on Network wireless	Based on host wired	Based on Network wired
Test rate	15	12	8	11
ARP Poisoning detection attack	98%	80%	98%	95%

Table *3- 1* : Describe statistics result of ARP-D method enter different medium

III.4.1.A -Discover the network scan process

Although this is not a serious process, it is classified as medium, while ARP Spoofing is critical. It is considered preliminary in order to identify the targets or victims who will subjected a serious attack later on. Attack process will done by NMAP and these are the results obtained from the application.

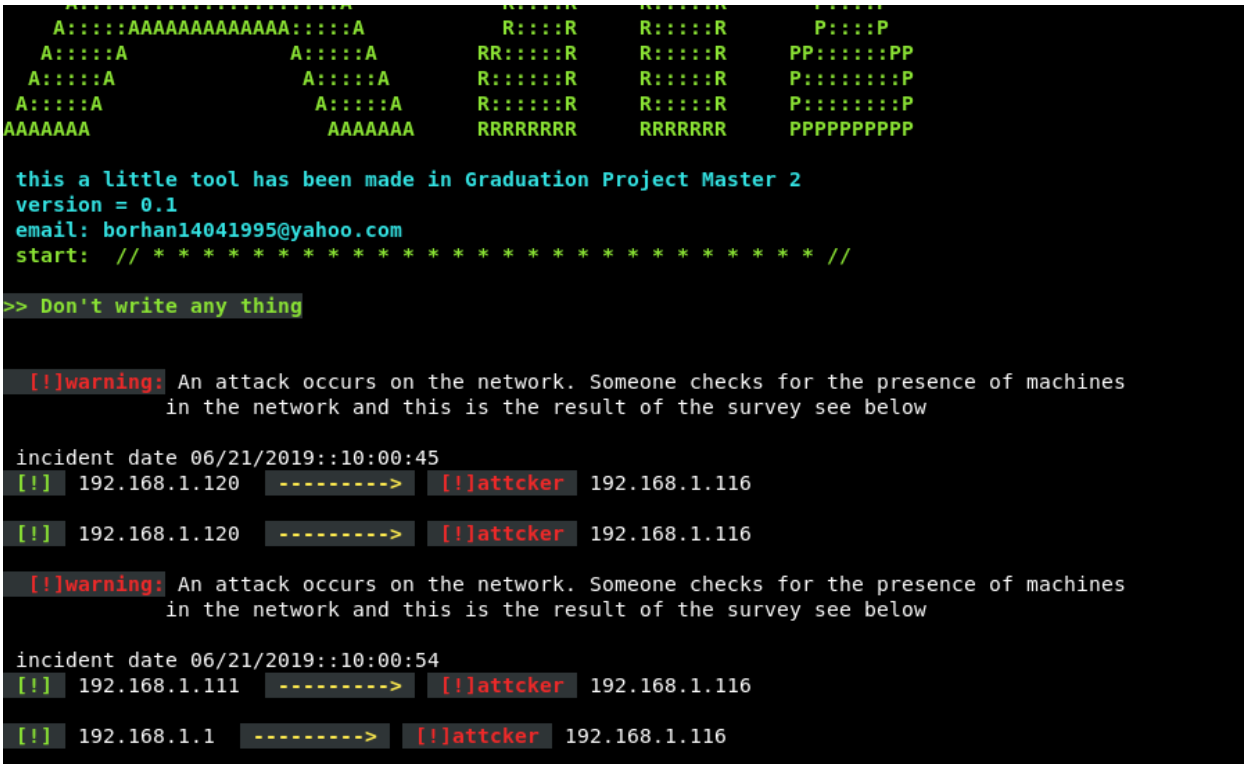

Figure 3-7 : Describe result of scan network process

The result means that the attacker sent more than 500 ARP packets in a short time, and that warning message appeared, meaning as follows :

- ➔ The first describes the warning, followed by the time of the event.
- ➔ set of IP addresses that the attacker gets during the scan.
- ➔ the red is IP address of the attacker.

III.4.2 -Result of Detect DNS Spoof

An attacker can set up a DNS malicious server in the network. The probability of success his attack is low . he tries to intercept traffic of DNS answer and then responds as quickly as possible. If the answer to the right DNS server is come first, the attack will have no effect. Some browsers also have mechanisms to detect any strange movement in DNS request and response that helps to reduce success of attacks. In order to get the desired results we simulate hacker attacks using some tools and here are the results.

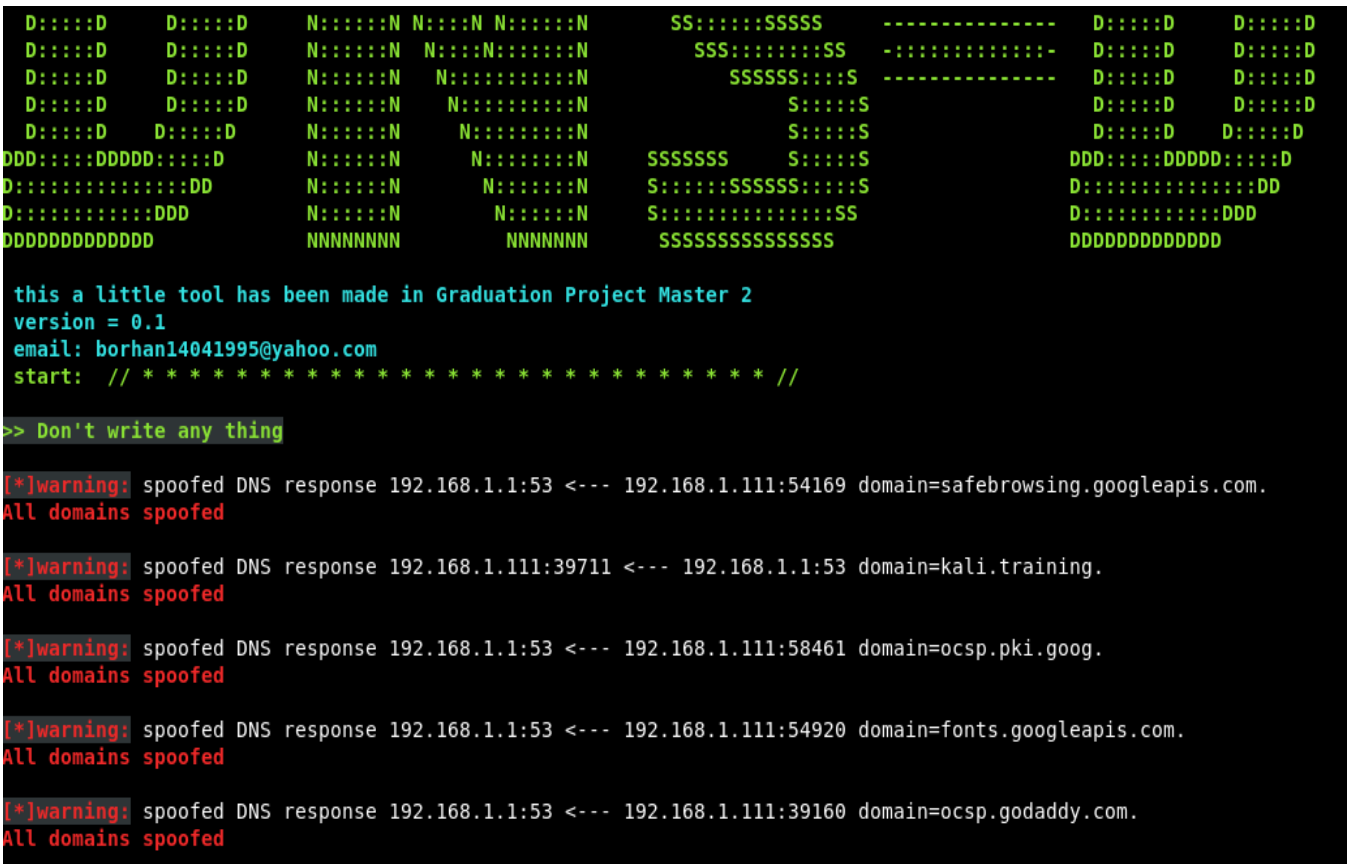

Figure 3-8 : Describe result of DNS-D method

This result is shown as the application works whatever mode, when we run DNS_Detect.py file, show us this interface in the terminal screen and then start to intercept all the traffic of response DNS packets, when the traffic is normal will not show any results. currently it can detect the DNS spoof attack , here We have warning messages in red where each message corresponds to a captured malicious packet and the information in the message is as follows :

➔ All are spoof DNS answers specific from a DNS server via target IP address followed by the domain name and address of the attacker's server.

This table describes the number of trial times in different cases followed by statistics results :

	Based on host wireless	Based on Network wireless	Based on host wired	Based on Network wired
Test rate	20	15	5	10
DNS spoofing detection attack	75%	50%	75%	50%

Table *3- 2* : Describe statistics result of DNS-D method enter different medium

III.4.3 -Result of Detect DHCP Spoof

In order to get the desired results we simulate hacker attacks using some tools and here are the results.

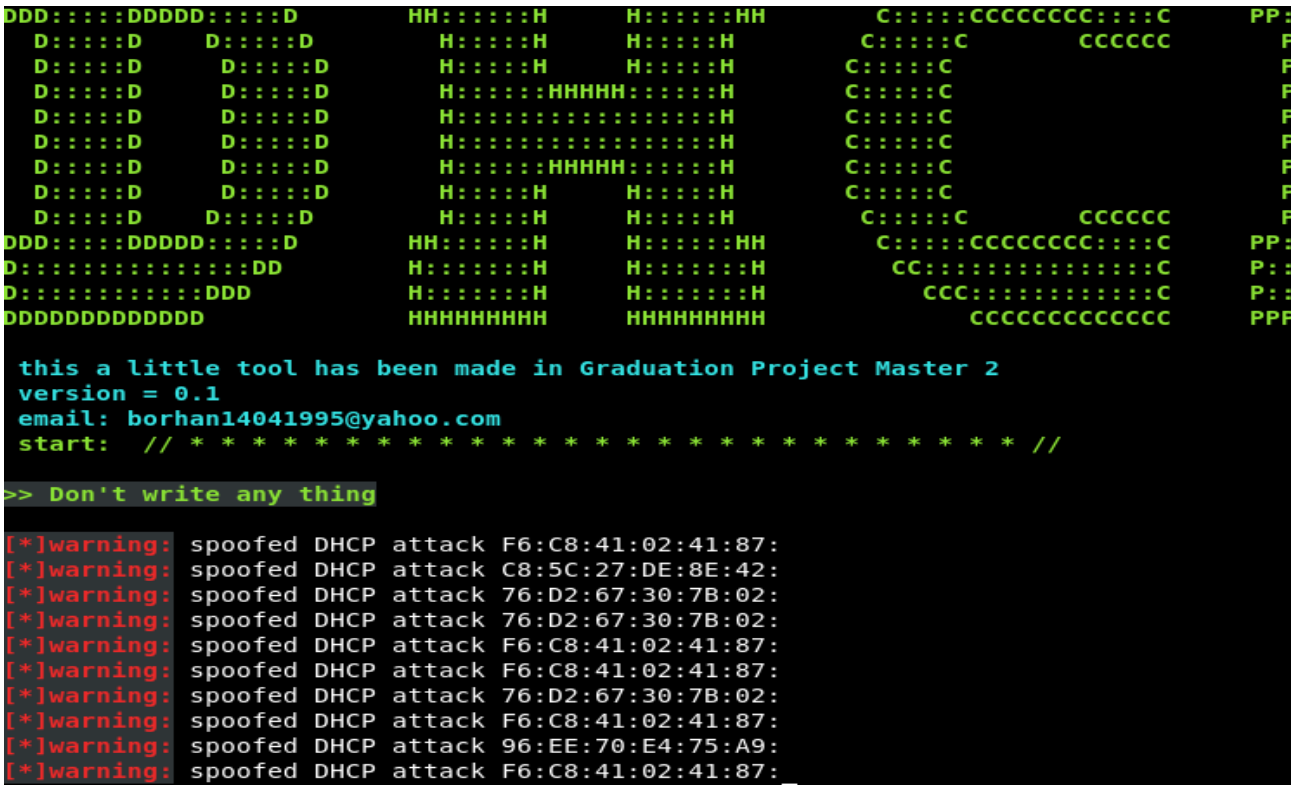

Figure 3- 5: Describe result of DHCP-D method

This result is shown as the application works whatever mode, when we run DHCP_Detect.py file, show us this interface in the terminal screen and then start to intercept all the traffic of request DHCP packets, when the traffic is normal will not show any results. currently it can detect the DHCP spoof attack , here We have warning messages in red where each message corresponds to a captured malicious packet and the information in the message is as follows :

- ➔ Random MAC addresses are drowned out by the attacker to router in order to drain all IP addresses.

This table describes the number of trial times in different cases followed by statistics results :

	Based on host wireless	Based on Network wireless	Based on host wired	Based on Network wired
Test rate	12	18	10	10
DHCP spoofing detection attack	35%	50%	60%	90%

Table *3- 3* : Describe statistics result of DHCP-D method enter different medium

III.5 -Conclusion

The phase of realization and design is the most important part of the application life cycle. This chapter has been dedicated to the presentation of architecture functional and technical of our application and tools used for the realization,We described the use of our application, and at the end of this chapter we showed the different parts of this work through some screenshots and tables that give stats about results of the application.

General Conclusion

As our dependence on computers and network constantly increases, comprehensive network security is of tremendous importance. A first requirement to be able to better protect networks assets is to gain a detailed understanding of malicious threats. The concept of detection man in the middle attack (MITM) was specifically invented to fill this task.

At the present time, however, the security aspect must be taken into account and given great importance. Threats that occur every day and the new techniques that hackers are currently using. By following the security guidelines and policies recommended by researchers and administrators, we can keep us safer and maintain the integrity and confidentiality of information.

In this dissertation, we provided a good structure to limit the attack of the man in the middle where we focused on the three most important factors where the vulnerability it Internet protocols (ARP, DNS, DHCP) ,With our understanding the process of exploitation,we were able, through our application, to develop the necessary protection mechanism using Scapy.

The solution to reduce attacks of a man in the middle is detection technique allowed us to have a deep knowledge of how the network protocols and data transmission work,through the continuous analysis with working of the application in addition to our study, we feel are able to develop application to work on different environments on infrastructure and adoption of smarter ways.

The experimental results of our application show how to detect attacks with some study statistics and this is to ensure that users take preventive measures to deter the actual attack.

Continuing the project will to cover most offensive techniques for MITM attack. In the future, we going to add the prevention methods we are currently working on. It is also will more controllable by enabling remote control using the (client / server) method, event logging, and logs management. Enable the sending of (alerts / notifications) via email in detail. Add methods of statistics and deep learning if possible. Intelligent network analysis for various protocols and devices, for example, svae the correct HTTP headers as digital signature from which we can identify the worst attacks such as social engineering and phishing attacks.

Bibliography

[1] Amin Khaldi.”Computer since security course ”.University kasdi merbah ouargle,Algerie

[2] Cybersecurity and Infrastructure Security Agency (CISA) of USA <Understanding Denial-of-Service Attacks> | US-CERT. (2009). Retrieved from , https://www.us-ce.rtgov/ncas/tips/ST04-015

[3] Department of Homeland Security (DHS),(Distrebution Denial-of-Service Attacks),document http://www.us-cert.gov/sites/default/files/publications/DDoS Quick Guide.pdf 14 /01/2014

[4] A. Furfaro, G. Malena, L. Molina and A. Parise, "A Simulation Model for the Analysis of DDoS Amplification Attacks," 2015 17th UKSim-AMSS International Conference on Modelling and Simulation (UKSim), Cambridge, 2015, pp. 267-272.

[5] jhon erickson,<THE ART OF EXPLOITATION, 2ND EDITION book>.No Starch Press, Inc 2008

[6] Harris, S., & Maymi, F. *CISSP All-in-One Exam Guide, 7th Edition* (7th ed., pp. Chapter 8 page 1187). New York : McGraw-Hill Education, 2016

[7] Dafydd Stuttard,Marcus Pinto:"The Web Application Hacker’s Handbook: Finding and Exploiting Security Flaws, Second Edition"(Chapter 12 page 431 XSS),Published by John Wiley & Sons, Inc.2011

[8] Dafydd Stuttard,Marcus Pinto:"The Web Application Hacker’s Handbook: Finding and Exploiting Security Flaws, Second Edition"(Chapter 9 page 287 SQL),Published by John Wiley & Sons, Inc.2011

[9] The Open Web Application Security Project (OWASP),available on https://www.owasp.org/index.php/Main_Page, OWASP Top 10 2017.

[10] Burleson, D. (2003). *Advanced SQL database programmers handbook*. [United States]: Rampant Techpress.

[11] Wiener, G. *Cyberterrorism and ransomware attacks*.

[12] G. Nath Nayak and S. Ghosh Samaddar, "Different flavours of Man-In-The-Middle attack, consequences and feasible solutions," 2010 3rd International Conference on Computer Science and Information Technology, Chengdu, 2010, pp. 491-495.

[13] S.Whalen,"An Introduction to ARP Spoofing", april 2001

[14] F. Callegati, W. Cerroni and M. Ramilli, "Man-in-the-Middle Attack to the HTTPS Protocol," in IEEE Security & Privacy, vol. 7, no. 1, pp. 78-81, Jan.-Feb. 2009

[15] Zouheir Trabelsi,Kadhim Hayawi,Arwa Al Braiki,Sujith Samuel Mathew."Network Attacks and Defenses",2013

[16] Abraham, S., Ciuca, I., Randhawa, E., C.T., L., Marara, L., blitherwart, y., & Singh, V. (2019). 12+ Types of Malware Explained with Examples (Complete List). Retrieved from https://www.malwarefox.com/malware-types/

[17] Koret, J., & Bachaalany, E. The antivirus hacker's handbook.

[18] James Graham,Richard Howard,Ryan Olson, "Cyber Security Essentials firewall",Auerbach Publications Taylor & Francis Group 2011

[19] James Graham,Richard Howard,Ryan Olson, "Cyber Security Essentials proxy server",Auerbach Publications Taylor & Francis Group 2011

[20] Reverse Proxy Guide - Apache HTTP Server Version 2.4. (2019). Retrieved from https://httpd.apache.org/docs/2.4/howto/reverse_proxy.html

[21] The Onion Router Project.Retrieved from ,https://2019.www.torproject.org/docs/documentation.html.en

[22] francois PILLOU,philippe BAY,"security protocole", Document:Toute sur la sucurité informatique

[23] HTTPS. (2019). Retrieved from https://en.wikipedia.org/wiki/HTTP_Secure

[24] Steve Tobias."Case Study In Secure File Transfer:Implementing Secure FTP with SSLIn a Healthcare Organization",GSEC SANS Institute,July 14 2004.

[25] Python 2.7.16 documentation. (2019). Retrieved from https://docs.python.org/2/index.html

[26] Philippe BIONDI."Network packet forgery with Scapy",phil(at)secdev.org|EADS Corporate Research Center SSI Department Suresnes, November 16, 2005

[27] Bullock, J., & Kadijk, J. (2017). Wireshark for Security Professionals. Somerset: John Wiley & Sons, Incorporated.

[28] Kali Linux | Penetration Testing and Ethical Hacking Linux. (2019). Retrieved from https://www.offensive-security.com/, creators kali linux.

نبذة مختصرة

أصبح أمن المعلومات عنصرا هاما للغاية ولا غنى عنه في حياة الناس و في عمل المؤسسات المختلفة ، و من اللازم حاليا توفير الحماية ضد التهديدات المختلفة

.من بين هذه التهديدات المعروفة نوع من الهجوم يسمى هجوم الرجل في الوسط من أكثر الهجمات خطورة

و المستخدمة من قبل المتسللين في الشبكات المحلية, يسمح للمهاجم بوضع نفسه بين المستخدمين و الراوتر حيث يتمكن من اعتراض حركة المرور والتجسس على مختلف الإتصالات والسيطرة عليها من تحكم واعادة توجيه، وسرقة المعلومات الحساسة ،وما الى ذلك

الهدف من مشروعنا هو تحقيق تطبيق يمكّننا من الحماية ضد هذه الهجمات بحيث يتم اكتشاف مصدر الهجوم ونوعه ومحاولة الغائه إن أمكن

Résumé

La sécurité de l'information est devenue un élément très important et indispensable dans la vie des gens et dans le travail de différentes institutions. Il est maintenant nécessaire d'assurer une protection contre diverses menaces.Parmi ces menaces bien connues se trouve une sorte d’attaque appelée l’homme du milieu (MitM en anglais) qui l’attaque parmi les attaques les plus dangereuses.Et utilisé par un pirate informatique dans les réseaux locaux permet à l’attaquant de se placer lui-même entre les utilisateurs et le routeur afin d’intercepter le trafic, d’espionner diverses communications, de le contrôler, de le redirection et de voler des informations sensibles, etc.

Le but de notre projet est de réaliser une application qui nous permette de nous protéger contre ces attaques afin de détecter la source, le type d’attaque et de l’annuler si possible.

Mots clés :

MitM, ARP, DNS, DHCP, Spoofing, Poisoning, Detection, Attack, Security