VULNERABILITY ASSESSMENT & PENETRATION TESTING (VAPT) REPORT

Capstone Project – Week 4

Engagement Information

Field	Details
Assessment Type	Vulnerability Assessment & Penetration Testing
Methodology	PTES (Penetration Testing Execution Standard)
Attacker Machine	Kali Linux
Attacker IP	`192.168.44.130`
Target System	Mr. Robot VM
Target IP	`192.168.44.132`
Network Type	Isolated Lab Environment
Performed By	Akhilesh Palve
Date	*2025-12-29*

---

1. Executive Summary

A full-scope Vulnerability Assessment and Penetration Test (VAPT) was conducted against the **Mr. Robot virtual machine** in a controlled laboratory environment. The objective of this assessment was to identify vulnerabilities, validate exploitability, and demonstrate potential impact through a real-world attack simulation.

The engagement resulted in a **complete system compromise**, including unauthorized web access, credential disclosure, lateral movement, and root-level privilege escalation. Multiple weaknesses were identified, including insecure web application configuration, weak authentication controls, exposed sensitive files, and improper SUID binary permissions.

These findings demonstrate how chained vulnerabilities can escalate from low-severity issues to **critical business impact** when not remediated properly.

---

2. Scope of Engagement

In Scope

- Web application hosted on ports 80 and 443
- Authentication mechanisms
- Local privilege escalation vectors
- File system and user access

Out of Scope

- Denial-of-Service attacks
- Attacks on external or production systems
- Brute force outside discovered credentials

---

3. Methodology

The engagement followed the **PTES framework**:

1. Reconnaissance
2. Enumeration
3. Vulnerability Analysis
4. Exploitation
5. Post-Exploitation
6. Privilege Escalation
7. Reporting & Remediation

---

4. Reconnaissance & Enumeration

4.1 Network Scanning (Nmap)

A service discovery scan was performed against the target host.

```
nmap -sC -sV -T4 -oA nmap-report 192.168.44.132
```

Open Ports Identified

Port	Service	Observation
22	SSH	Closed
80	HTTP	Apache web server
443	HTTPS	Apache with SSL

The presence of a publicly accessible web service significantly increased the attack surface.

---

5. Web Application Enumeration

5.1 robots.txt Discovery

Accessing `/robots.txt` revealed:

- A partial key file (`key-1-of-3.txt`)
- A custom dictionary file (`fsocity.dic`)

The first flag was successfully retrieved during this phase.

5.2 Wordlist Analysis

The discovered dictionary file contained over **850,000 entries**, many of which were duplicates.

```
wc fsocity.dic
sort fsocity.dic | uniq | wc -l
```

After removing duplicates, the wordlist was reduced to **11,451 unique entries**, improving efficiency for subsequent authentication testing.

---

6. Vulnerability Scanning (Nikto)

A Nikto scan identified multiple security issues, including:

- WordPress installation exposure
- Admin login pages
- Sensitive configuration file references

- Misconfigured HTTP headers

```
nikto -h http://192.168.44.132 -output nikto_results.txt
```

These findings confirmed the presence of a vulnerable WordPress environment.

---

7. Authentication Attacks

7.1 Username Enumeration

Using the cleaned wordlist, username enumeration revealed a valid WordPress user account.

```
hydra -L fsocity.dic.uniq -p gibberish 192.168.44.132 http-post-form ...
```

Discovered Username: `elliot`

7.2 Password Discovery

A password spray using the same wordlist successfully identified valid credentials.

```
hydra -l elliot -P fsocity.dic.uniq 192.168.44.132 http-post-form ...
```

Valid Credentials:
`elliot : ER28-0652`

---

8. Exploitation

8.1 WordPress Admin Exploitation

Authenticated access to WordPress enabled exploitation via an admin-level plugin upload vulnerability.

```
Metasploit Module:
exploit/unix/webapp/wp_admin_shell_upload
```

Successful exploitation resulted in a **Meterpreter session** running as the `daemon` user.

---

9. Post-Exploitation

9.1 Sensitive File Disclosure

The following files were identified:

```
/home/robot/key-2-of-3.txt
/home/robot/password.raw-md5
```

The password hash was extracted and successfully cracked using a dictionary attack.

```
robot : abcdefghijklmnopqrstuvwxyz
```

The second flag was retrieved after switching to the `robot` user.

---

10. Privilege Escalation

10.1 SUID Binary Misconfiguration

Enumeration revealed an **outdated SUID-enabled Nmap binary** owned by root.

```
/usr/local/bin/nmap
```

The legacy interactive mode allowed execution of shell commands with elevated privileges.

```
nmap --interactive
!sh
```

This resulted in **root-level access** and retrieval of the final flag.

---

11. Impact Analysis

Security Property	Impact
Confidentiality	Critical
Integrity	Critical

Availability	Medium
Overall Risk	**Critical**

The system was fully compromised, allowing an attacker to read sensitive files, escalate privileges, and maintain control.

---

12. Remediation Recommendations

Immediate Actions

- Secure WordPress configuration
- Enforce strong authentication and rate-limiting
- Restrict access to sensitive files
- Remove legacy SUID binaries

Long-Term Improvements

- Regular vulnerability scanning
- Principle of Least Privilege
- Patch management
- Web Application Firewall (WAF) deployment

---

13. Key Learnings (Internship Reflection)

This engagement demonstrated how multiple low-severity vulnerabilities can be chained into a full system compromise. It reinforced the importance of secure web configuration, credential protection, and proper privilege management. The exercise also strengthened practical skills in professional reporting and structured penetration testing.

---

14. Conclusion

The Mr. Robot VM was found to be critically vulnerable due to multiple misconfigurations and weak security controls. The successful compromise highlights the real-world risks associated with poor security hygiene. Implementing the recommended remediation steps would significantly improve the system’s security posture.