1. Summary

This practical assessment demonstrates a complete hands-on exploitation lifecycle in a controlled virtual lab environment. A vulnerable target machine (Metasploitable2) was attacked from Kali Linux using a combination of reconnaissance, web application testing, exploit chaining, Python PoC customization, post-exploitation enumeration, evidence collection, and formal reporting.
A reflected XSS vulnerability was first identified on DVWA, followed by successful exploitation and session escalation to a Meterpreter shell using Metasploit. Post-exploitation enumeration confirmed full root access. A Python PoC from Exploit-DB was also customized and tested for a specific CVE. Network traffic was captured and verified using cryptographic hashing to maintain forensic integrity.
This exercise successfully demonstrates real-world penetration testing workflow covering exploitation, evidence handling, vulnerability documentation, and management reporting.

---

2. Scope and Methodology

Scope

- **Attacker Machine:** Kali Linux
- **Target Machine:** Metasploitable2
- **Target IP:** 192.168.56.102
- **Web Application:** DVWA
- **Tools Used:** Metasploit, Nmap, Burp Suite, sqlmap, Python, Exploit-DB, Wireshark, Google Docs

Methodology Followed (PTES Aligned)

1. Reconnaissance using Nmap
2. Web vulnerability testing (XSS, SQLi)
3. Exploit chaining (XSS → RCE via Metasploit)
4. Python PoC customization
5. Post-exploitation enumeration
6. Evidence collection & hashing
7. Reporting & remediation

---

3. Technical Findings

3.1 Advanced Exploitation Lab – Exploit Chain

Step-1: Service Reconnaissance

```
nmap -sC -sV 192.168.56.102
```

Key services identified:

- Port 80 – Apache Web Server

- Port 21 – vsFTPd 2.3.4
- Port 3306 – MySQL
- Port 8180 – Apache Tomcat

Step-2: XSS Identification

Target: http://192.168.56.102/dvwa

Payload: <script>alert('XSS')</script>

JavaScript executed → **Reflected XSS Confirmed**

Step-3: Exploit Chain (XSS → RCE using Metasploit)

Session upgraded using:
post/multi/manage/shell_to_meterpreter

Meterpreter session obtained as ROOT
Verification:
getuid → root
sysinfo → Ubuntu 8.04, Kernel 2.6.24

Exploit Chain Log Table

Exploit ID	Description	Target IP	Status	Payload
004	XSS → RCE Chain	192.168.56.102	Success	Meterpreter

Step-4: Python PoC Customization (Exploit-DB)

PoC Used:
34992.py – Drupalgeddon (CVE-2014-3704)

Execution Result:
[X] NOT Vulnerable

Target was not running vulnerable Drupal

PoC Customization Summary
The downloaded Python PoC from Exploit-DB was modified to hardcode the target IP address 192.168.56.102 and use the correct HTTP port. Additional exception handling was added to gracefully manage connection errors and unreachable services, ensuring reliable execution within the isolated Metasploitable2 test environment.

Step-5: Post-Exploitation Enumeration

Commands executed:

- getuid
- sysinfo
- ipconfig
- ls
- cat /etc/passwd
- ps
- netstat -antp

Full system enumeration completed with root privileges

---

4. Web Application Testing Lab – DVWA

Testing Log

Test ID	Vulnerability	Severity	Target URL
001	SQL Injection	Critical	http://127.0.0.1/dvwa/vulnerabilities/sqli/
002	XSS Reflected	Medium	http://127.0.0.1/dvwa/vulnerabilities/xss_r/

Burp Suite Testing

- Session token intercepted via proxy
- Security level parameter tampered
- Manual request manipulation performed

sqlmap Result

- Databases enumerated:
    - dvwa
    - information_schema

Automated SQL Injection confirmed

Web Testing Summary

The DVWA application was successfully tested for OWASP Top 10 vulnerabilities using manual and automated techniques. SQL Injection and reflected XSS were confirmed through Burp Suite manipulation and sqlmap automation. Session interception was also demonstrated, highlighting insecure authentication and insufficient input validation controls.

---

5. Reporting Practice – Findings & CVSS

Finding ID	Vulnerability	CVSS Score	Remediation
F001	SQL Injection	9.1	Input validation, prepared statements
F002	XSS Reflected	6.1	Output encoding, input sanitization
F003	vsftpd Backdoor	9.8	Upgrade FTP service
F004	Weak Session Security	7.5	Secure cookies, HTTPS

6. Post-Exploitation & Evidence Collection

Traffic Capture Tool
Wireshark

SHA-256 Hash Verification
0da00471d072f9715639167fd15bab901efa92b408d2a5f84b7ea1ba6905383e

Evidence Log

Item	Description	Collected By	Date	Hash Value
Traffic Log	HTTP Traffic	VAPT Analyst	2025-08-25	0da00471d072f9715639167fd15bab901efa92b408d2a5f84b7ea1ba6905383e

Evidence Summary
Network traffic was captured using Wireshark during live interaction with the DVWA application. The capture was saved in PCAPNG format and cryptographically verified using SHA-256 hashing. This ensured forensic integrity, proper chain-of-custody, and reliable evidence preservation for further analysis and reporting.

7. Capstone Project – Full VAPT Cycle (PTES Aligned)

7.1 Capstone Objective
The objective of this Capstone Project was to simulate a complete Vulnerability Assessment and Penetration Testing (VAPT) lifecycle in a controlled lab environment using **Kali Linux** as

the attacker and Metasploitable2 as the target. The engagement covered reconnaissance, exploitation, detection, remediation planning, verification, and final reporting in alignment with the PTES framework.

7.2 Capstone Simulation – Exploitation Phase

The exploitation phase utilized the same confirmed vulnerabilities documented earlier in this report:

- vsftpd 2.3.4 backdoor (network-level RCE)
- SQL Injection on DVWA
- Reflected XSS on DVWA

Successful exploitation resulted in:

- Remote command execution
- Privilege escalation to root
- Full system compromise of the target host

This confirms the attack simulation requirement of the Capstone project was successfully achieved.

7.3 Capstone Detection – Vulnerability Logging (OpenVAS/Nmap Correlation)

Timestamp	Target IP	Vulnerability	PTES Phase
2025-08-25 12:45:00	192.168.56.102	vsftpd Backdoor	Exploitation
2025-08-25 12:55:00	192.168.56.102	SQL Injection	Web Application
2025-08-25 13:05:00	192.168.56.102	Reflected XSS	Web Application
2025-08-25 13:15:00	192.168.56.102	Weak Session Handling	Post-Exploitation

Detection is validated using:

- Nmap service discover
- Burp Suite interception
- sqlmap automation

7.4 Capstone Remediation & Verification Plan

Vulnerability	Recommended Patch / Control
vsftpd 2.3.4	Upgrade to latest stable FTP version

SQL Injection	Use prepared statements & ORM
XSS	Apply strict output encoding
Telnet	Disable and enforce SSH-only access
Cookies	Enable HttpOnly, Secure, SameSite

Post-Remediation Verification (Rescan Plan):

- Run Nmap to confirm closed/filtered ports
- Re-test DVWA SQL queries after sanitization
- Validate XSS payload blocking
- Confirm session cookies are secured

7.5 PTES Executive Summary (Capstone Report)

This Capstone Project demonstrates a complete penetration testing lifecycle conducted on a deliberately vulnerable virtual environment using Kali Linux as the attacker and Metasploitable2 as the target. The engagement followed the Penetration Testing Execution Standard (PTES) and included reconnaissance, vulnerability discovery, exploitation, post-exploitation, evidence collection, and remediation planning.

Network scanning identified multiple exposed services, including FTP, HTTP, MySQL, and Tomcat. Web application testing on DVWA confirmed SQL Injection and reflected Cross-Site Scripting vulnerabilities. Exploitation of a known vsftpd backdoor and web-layer weaknesses resulted in full remote code execution and privilege escalation to root access.

Post-exploitation enumeration validated total system compromise.

Forensic network traffic was captured and cryptographically verified using SHA-256 hashing to maintain evidence integrity. Findings were risk-rated using CVSS and mapped to OWASP Top 10. Remediation strategies were proposed, including service upgrades, secure coding practices, and access hardening.

This project effectively simulates a real-world enterprise penetration test and demonstrates strong practical proficiency in offensive security, evidence handling, and professional reporting.

7.6 Non-Technical Management Brief (Capstone)

A controlled security assessment was conducted to evaluate how an attacker could compromise an organization’s systems. Multiple serious security weaknesses were identified, including unsafe web application coding and outdated network services. These flaws allowed full system access to be gained with administrator-level control.

Such vulnerabilities could lead to data theft, service disruption, and regulatory violations in a real-world environment. Immediate security hardening, regular updates, and continuous security testing are strongly recommended. This assessment highlights the importance of proactive cybersecurity investment to prevent costly breaches and operational downtime.

8. Recommendations Summary

- Sanitize all web inputs to prevent XSS
- Use prepared statements to prevent SQL Injection
- Upgrade vsftpd 2.3.4 immediately
- Disable Telnet and unused services
- Implement firewall and IDS/IPS
- Enforce secure cookie flags (HttpOnly, Secure)
- Conduct periodic VAPT and rescan after patching

9. Conclusion

This practical engagement successfully demonstrated a complete penetration testing lifecycle. The lab confirmed that multiple critical vulnerabilities exist at both the network and application layers. A full exploit chain was executed from web-level XSS to system-level root access. Post-exploitation enumeration and forensic evidence collection validated attacker persistence and impact. The findings clearly show how a real attacker could compromise a production environment if proper security controls are not maintained.