Vulnerability Assessment & Pentesting Lab - DVWA

Abstract

This project demonstrates a web-based vulnerability assessment and penetration testing lab using Damn Vulnerable Web Application (DVWA) hosted on Metasploitable, with Kali Linux as the attacker machine. It showcases real-world OWASP Top 10 vulnerabilities and how to exploit them using industry-standard tools like SQLmap, Hydra, Burp Suite, and browser-based manual testing. This structured lab covers everything from reconnaissance to exploitation and proof of concept documentation.

1. Lab Setup

- Kali Linux (Attacker): Tools used - SQLmap, Hydra, Burp Suite, Nmap, Firefox
- Metasploitable (Target): DVWA pre-installed
- Network: Bridged adapter, same subnet (e.g., 192.168.146.0/24)
- DVWA Security Level: Initially Low, later set to Medium for CSRF testing

2. Attack Summary

Attack Type	Tool Used	Result / Notes
SQL Injection	SQLmap	Dumped databases via GET injection with session cookies
Command Injection	Browser/Burp	Executed system commands via ping form
Auth Bypass	Manual	Used classic 'OR 1=1-- to bypass login
FTP Brute Force	Hydra	Cracked 'msfadmin' credentials using rockyou.txt
XSS (Reflected)	Browser	Triggered popup via search input
XSS (Stored)	Browser	Payload executed after relogin via feedback form
CSRF (Medium)	Custom HTML	Forged GET request changed password without token

🔎 3. Detailed Attack Walkthrough

➤ SQL Injection

Dumped databases via GET injection with session cookies

➤ Command Injection

Executed system commands via ping form

➤ Auth Bypass

Used classic 'OR 1=1-- to bypass login

➤ FTP Brute Force

Cracked 'msfadmin' credentials using rockyou.txt

➤ XSS (Reflected)

Triggered popup via search input

➤ XSS (Stored)

Payload executed after relogin via feedback form

➤ CSRF (Medium)

Forged GET request changed password without token

📌 4. Observations

DVWA on Metasploitable allowed exploitation of multiple OWASP Top 10 vulnerabilities, including SQLi, XSS, and CSRF. Despite Medium security being selected, the CSRF protection was incomplete, allowing tokenless attacks. Screenshots were collected during all phases of the testing.

✅ 5. Conclusion

This pentesting lab demonstrates key web security flaws and how attackers exploit them using both automated and manual methods. This project provides a strong demonstration of web security awareness, tool mastery, and documentation — highly relevant for red teaming, security testing, or SOC roles.