Web Application Penetration Testing Report

Project: DVWA Penetration Testing

Prepared by: Prajakta Navnath Patole

Date: 01/09/2025

# Abstract

This project demonstrates a penetration test performed on the Damn Vulnerable Web Application (DVWA). The test focused on identifying common web application vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), and Session Fixation. Tools such as Kali Linux, Docker, Nmap, and Firefox Developer Tools were used to perform reconnaissance, scanning, exploitation, and vulnerability reporting.

# Introduction

DVWA is a deliberately vulnerable PHP/MySQL web application designed for security professionals to practice penetration testing techniques. This project aimed to demonstrate real-world web vulnerabilities and highlight remediation strategies to improve security.

# Setup & Tools

- Kali Linux (Virtual Machine)  
- Docker  
- DVWA (Damn Vulnerable Web Application)  
- Nmap for network reconnaissance  
- Firefox Developer Tools for inspecting cookies and sessions

# Testing Methodology

The penetration test followed a structured methodology:  
1. Reconnaissance: Identify services and open ports using Nmap.  
2. Scanning: Analyze web application structure and entry points.  
3. Exploitation: Perform SQL Injection, XSS, and Session Fixation tests.  
4. Post-Exploitation: Evaluate impact and capture evidence.  
5. Reporting: Document findings, screenshots, and remediation steps.

# Findings

The following table summarizes the identified vulnerabilities:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Vulnerability | Location | Risk Level | Evidence (Screenshot) | Fix (Summary) |
| F1 | SQL Injection | /vulnerabilities/sqli/ | High | Screenshot 1 | Use parameterized queries and input validation |
| F2 | Cross-Site Scripting (XSS) | /vulnerabilities/xss\_r/ | Medium | Screenshot 2 | Encode output, validate input |
| F3 | Session Fixation | Login Workflow | High | Screenshot 3 | Regenerate session IDs on login |

# Recommendations

- Use parameterized queries and input validation to prevent SQL Injection.  
- Implement proper output encoding and whitelist input to prevent XSS.  
- Regenerate session IDs after login to prevent Session Fixation.  
- Regularly update and patch systems, and conduct security testing periodically.

# Conclusion

The penetration test successfully demonstrated three major vulnerabilities in DVWA. By implementing the recommended fixes, organizations can significantly reduce security risks. This project highlights the importance of secure coding practices, regular testing, and proactive security measures.

# Appendix

Include screenshots of Nmap scans, SQLi exploitation, XSS popup, and session fixation proof here.







