Vulnerability Assessment and Penetration Testing (VAPT) on OWASP Juice Shop

Name: Ashwani Maurya

Institution/Organization: Kashi Institute of Technology

Course Name: B.Tech (CSE)

Date: 22/07/2025

Supervisor: [Your Supervisor Name Here]

Abstract

This project report presents a comprehensive Vulnerability Assessment and Penetration Testing (VAPT) conducted on OWASP Juice Shop, a deliberately vulnerable web application. The aim is to simulate real-world cyberattacks to identify, exploit, and document security flaws within the application. The methodology included reconnaissance, vulnerability scanning using tools such as OWASP ZAP and Burp Suite, and manual exploitation of common web vulnerabilities like SQL Injection (SQLi), Cross-Site Scripting (XSS), and Cross-Site Request Forgery (CSRF). The results revealed multiple critical vulnerabilities, which were analyzed in terms of severity, impact, and mitigation strategies. This project not only enhances understanding of offensive security but also emphasizes the importance of secure coding and web application hardening. Future recommendations are provided to secure similar applications in real-world environments.

Table of Contents

List of Figures and Tables

Introduction

This project is based on a security assessment of OWASP Juice Shop, a web application intentionally developed with multiple vulnerabilities. The objective was to identify and exploit these weaknesses to understand their risks and develop appropriate mitigation techniques. This kind of testing is crucial in real-world applications to reduce attack surfaces and prevent data breaches. The tools used included Burp Suite, OWASP ZAP, and Nikto, all of which are commonly used in cybersecurity assessments.

Methodology

6.1 Approach

The goal was to perform a white-box penetration test on OWASP Juice Shop. The application was hosted locally using Docker. Each vulnerability was approached manually and with automated tools.

6.2 Tools Used

- Burp Suite: Intercept and manipulate HTTP requests.
- OWASP ZAP: Active and passive vulnerability scanning.
- Nikto: Scan for outdated server configurations.
- Docker: Host the Juice Shop instance locally.
- Kali Linux: Environment for penetration testing.

6.3 Step-by-Step Process

1. Setup: OWASP Juice Shop was deployed locally using Docker.
2. Reconnaissance: Identified technologies, server headers, and directories.
3. Scanning: Ran scans using OWASP ZAP and Nikto.
4. Exploitation: Performed manual testing for SQLi, XSS, CSRF, Broken Auth.
5. Documentation: Recorded vulnerabilities, evidence (screenshots), and CVSS scores.
6. Mitigation: Suggested fixes and best practices for each vulnerability.

Summary of Alerts

Risk Level	Number of Alerts
High	2
Medium	4
Low	8
Informational	6

Alerts

Name	Risk Level	Number of Instances
Anti-CSRF Tokens Check	High	10
Cross Site Scripting (Reflected)	High	2
Buffer Overflow	Medium	529
Content Security Policy (CSP) Header Not Set	Medium	58
Example Passive Scan Rule: Denial of Service	Medium	7
X-Frame-Options Header Not Set	Medium	55
Absence of Anti-CSRF Tokens	Low	73
Application Error Disclosure	Low	1
Cookie No HttpOnly Flag	Low	1
Cookie without SameSite Attribute	Low	2
In Page Banner Information Leak	Low	3
Information Disclosure - Debug Error Messages	Low	1
Permissions Policy Header Not Set	Low	59
X-Content-Type-Options Header Missing	Low	62
Information Disclosure - Suspicious Comments	Informational	58
Loosely Scoped Cookie	Informational	3
Modern Web Application	Informational	33
Non-Storable Content	Informational	2
Storable and Cacheable Content	Informational	64
User Controllable HTML Element Attribute (Potential XSS)	Informational	32

Alert Detail

High	Anti-CSRF Tokens Check
	A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an