Cybersecurity Vulnerability Assessment Report

Target: OWASP Juice Shop (Test Environment)
Tool Used: Burp Suite Professional
Submitted By: Gourav Swaroop
Date: 11/08/2025

Executive Summary

A penetration test was conducted on the OWASP Juice Shop application in a controlled lab environment. The objective was to identify vulnerabilities aligned with the OWASP Top 10 (2021).

During the assessment, four critical vulnerabilities were discovered:

- **SQL Injection (SQLi)** – High
- **Cross-Site Scripting (XSS)** – High
- **Cross-Site Request Forgery (CSRF)** – High
- **Security Misconfiguration** – Medium

These findings demonstrate common web application weaknesses that could allow attackers to bypass authentication, execute malicious scripts, or manipulate user sessions.

Methodology

The following methodology, based on the **OWASP Testing Guide**, was employed:

1. Environment Setup

- OWASP Juice Shop deployed locally: http://127.0.0.1:3000
- Burp Suite Professional configured as proxy: 127.0.0.1:8080

2. Reconnaissance & Scoping

- Defined target scope in Burp Suite.
- Manually explored the application to identify input vectors (login forms, search bar, feedback forms).

3. Automated Scanning

- Performed **Crawl + Audit (Balanced Mode)** scan in Burp Suite.
- Filtered results for SQLi, XSS, CSRF, and misconfigurations.

4. Manual Verification & Exploitation

- Verified vulnerabilities using Burp Repeater, Proxy, and manual browser tests.
- Created PoCs for XSS and CSRF.

5. Documentation

- Mapped findings to **OWASP Top 10 (2021)** categories.
- Captured screenshots as proof-of-concept.
- Provided mitigation recommendations.

Findings

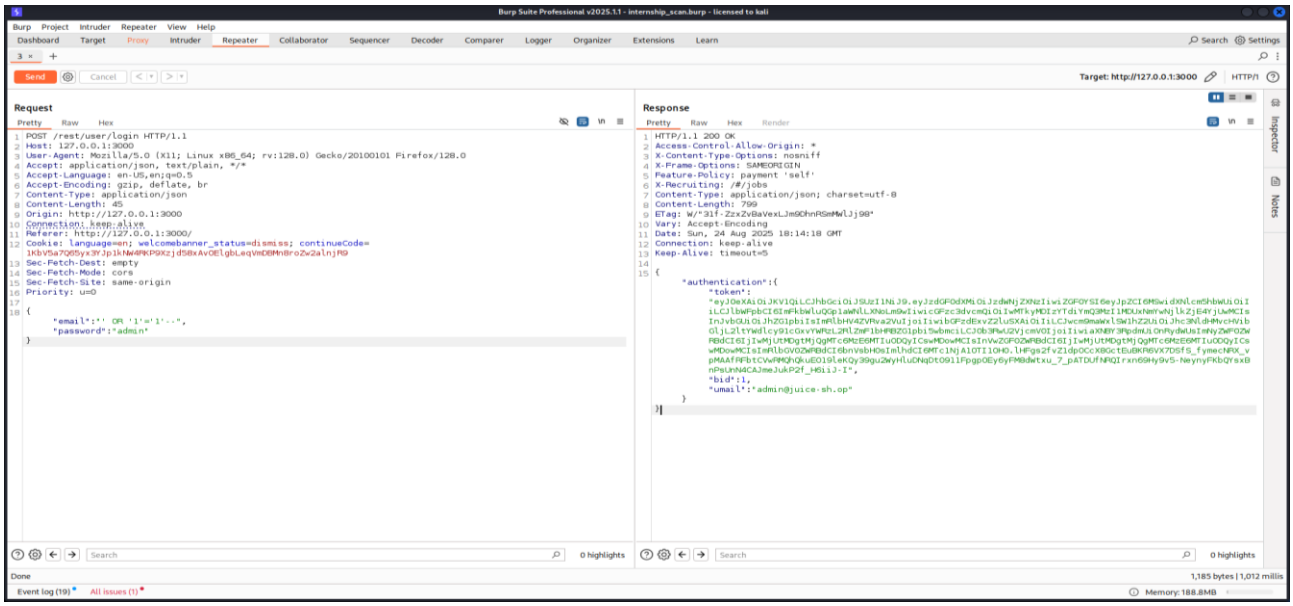
1. SQL Injection (SQLi)

Description:
Login form did not sanitize user inputs, allowing SQL injection payloads to bypass authentication.

Evidence (PoC):

Request payload: {"email":"' OR '1'='1'--","password":"admin"}

Response: 200 OK with valid authentication token



Screenshot: Figure 1 – Finding-SQLi.png

Impact:

- Attackers can bypass login, access admin accounts, or extract sensitive data.

Risk Rating: High

Mitigation:

- Use parameterized queries / prepared statements.
- Implement server-side input validation.
- Employ a Web Application Firewall (WAF) to filter malicious queries.

---

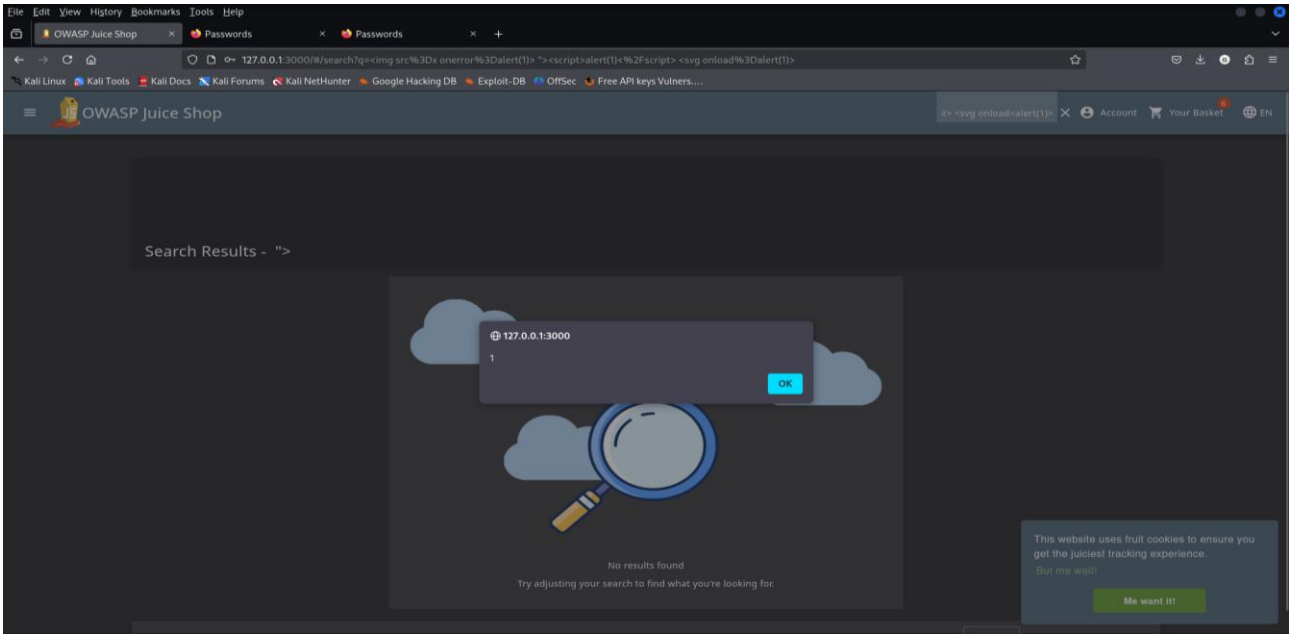
2. Cross-Site Scripting (XSS)

Description:
Search functionality failed to sanitize user input, allowing JavaScript injection.

Evidence (PoC):

Payload entered: <script>alert(1)</script>

Result: Browser displayed JavaScript alert



Screenshot: Figure 2 – Finding-XSS.png

Impact:

- Attackers can steal session cookies, deface content, or execute malicious scripts.

Risk Rating: High

Mitigation:

- Apply output encoding (HTML entity encoding).
- Sanitize user input with libraries like DOMPurify.
- Enforce a strict Content Security Policy (CSP).

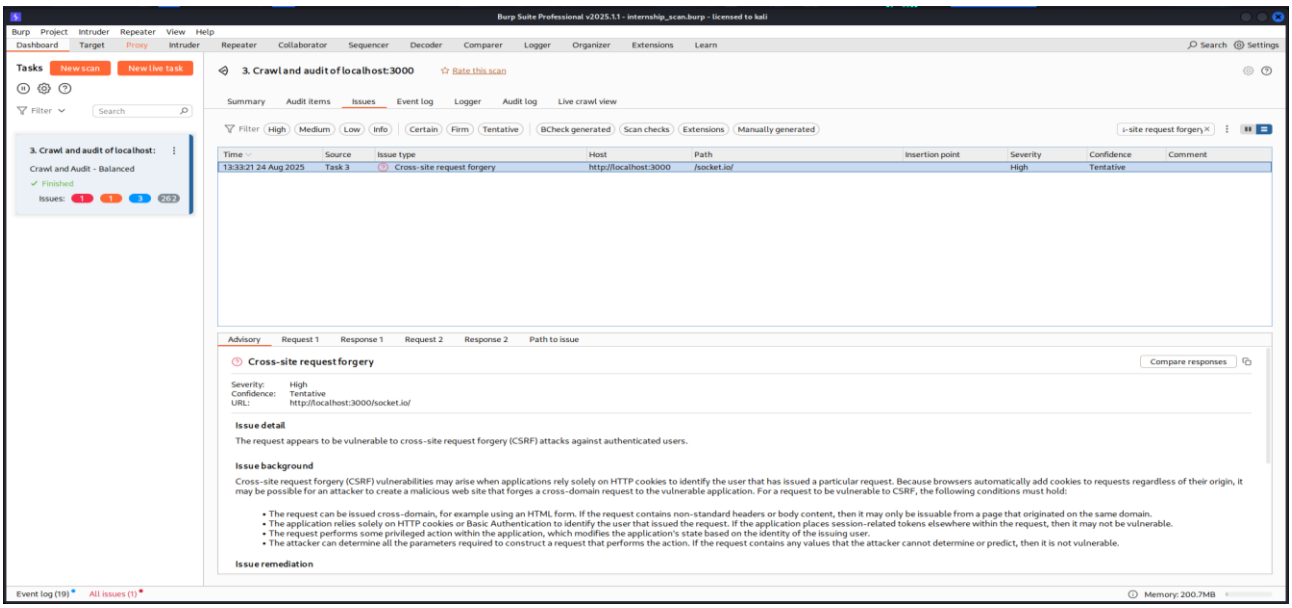
---

3. Cross-Site Request Forgery (CSRF)

Description:
The application relies solely on cookies for authentication and lacks anti-CSRF tokens for state-changing requests.

Evidence (PoC):

- Malicious HTML form submitted without user re-authentication, successfully changing account state.



Screenshot: Figure 3 – Finding-CSRF.png

Impact:

- Attackers can trick authenticated users into performing unwanted actions, such as changing passwords or modifying data.

Risk Rating: High

Mitigation:

- Implement anti-CSRF tokens for all state-changing requests.
- Use SameSite cookies to prevent cross-origin requests.
- Validate Referer or Origin headers on server-side.

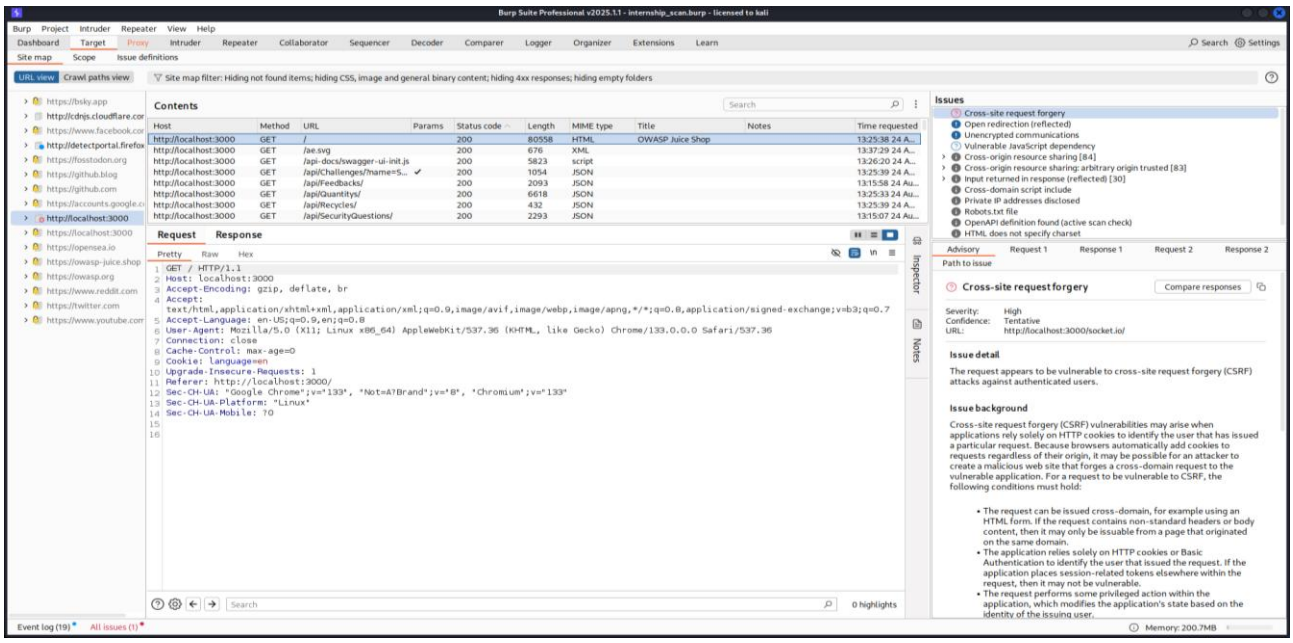
---

4. Security Misconfiguration

Description:
Server response headers revealed missing security directives and the application ran over HTTP.

Evidence (PoC):

- Missing headers: Content-Security-Policy, Strict-Transport-Security, X-Content-Type-Options
- Application accessible via HTTP



Screenshot: Figure 4 – Finding-SecurityMisconfig.png

Impact:

- Increases exposure to clickjacking, MIME sniffing, and MITM attacks.

Risk Rating: Medium

Mitigation:

- Enable HTTPS with HSTS.
- Configure headers:
    - X-Frame-Options: DENY
    - X-Content-Type-Options: nosniff
    - Content-Security-Policy with strict rules
- Regularly harden server configurations.

OWASP Top 10 Mapping

Vulnerability	OWASP Top 10 Category (2021)	Risk Level
SQL Injection	A03: Injection	High
Cross-Site Scripting (XSS)	A03: Injection / A07: XSS	High
Cross-Site Request Forgery (CSRF)	A08: Software & Data Integrity Failures	High
Security Misconfiguration	A05: Security Misconfiguration	Medium

---

Conclusion

The assessment identified multiple **critical vulnerabilities** in the test application. Exploiting these weaknesses could allow attackers to bypass authentication, inject malicious scripts, or manipulate user sessions.

Implementing the recommended mitigations will significantly improve the application's **resilience against OWASP Top 10 threats**.

Login form did not sanitize user inputs, allowing SQL injection payloads to bypass authentication.

Evidence (PoC):