PENETRATION TEST REPORT

Client: OWASP Juice-Shop
Project: Web Application Penetration Test
Assessment Type: Web Application Security Assessment
Prepared By: Victor Muthomi
Assessment Dates: May 13th, 2026 – July 14th, 2026
Report Date: July 16th, 2026
Classification: Confidential

SECTION 01 - Executive Summary

Overview

A penetration test was conducted to evaluate the security posture of the OWASP Juice-Shop web application. The assessment simulated real-world attacks performed by threat actors to identify exploitable vulnerabilities before they can be abused by malicious individuals.

What is OWASP Juice-Shop?
Juice-Shop is an intentionally vulnerable web application maintained by the Open Worldwide Application Security Project (OWASP). It is designed for security training and education, containing over 80 vulnerabilities across various difficulty levels. It serves as an ideal learning platform for penetration testing and security awareness.

The objective was to determine:

- Current security maturity
- Potential attack paths
- Impact of successful exploitation
- Overall business risk
- Recommended remediation measures

Testing was performed using industry-recognized methodologies while ensuring minimal impact on production services.

Overall Risk Rating

Severity	Count
Critical	2
High	4
Medium	6
Low	5
Informational	8

Security Posture

Overall security posture was assessed as:

Moderate Risk

The environment contains several exploitable vulnerabilities that could allow an attacker to gain unauthorized access, escalate privileges, or compromise sensitive information.

Immediate remediation is recommended for all Critical and High-risk findings.

Business Impact

Successful exploitation may result in:

- Unauthorized access to sensitive information
- Credential theft
- Data leakage
- Service disruption
- Financial loss
- Regulatory non-compliance
- Reputation damage

Executive Recommendations

Priority actions include:

- Patch vulnerable systems
- Implement Multi-Factor Authentication (MFA)

- Enforce strong password policies
- Harden exposed services
- Improve network segmentation
- Conduct continuous vulnerability assessments
- Deploy centralized logging and monitoring

SECTION 02 – Scope & Objectives

Assessment Scope

In-Scope Assets

Asset	Description
localhost:3000	OWASP Juice-Shop web application
API endpoints	REST API for Juice-Shop functionality
/administration	Admin panel interface
/rest	RESTful API routes
/socket.io	WebSocket communication

Out of Scope

- Social engineering
- Physical security
- Denial of Service (DoS)
- Production data modification
- Third-party hosted infrastructure

Objectives

The objectives were to:

- Identify security weaknesses
- Validate exploitability
- Assess business impact
- Evaluate existing security controls
- Provide remediation guidance

Rules of Engagement

- Testing performed during agreed maintenance window
- No destructive attacks
- No permanent modification of production data
- Client authorization obtained

SECTION 03 - Methodology

Testing followed internationally recognized penetration testing standards.

Frameworks Used

- **OWASP Web Security Testing Guide (WSTG)** – A comprehensive guide for testing web application security
- **OWASP Top 10** – The top 10 most critical web application security risks
- **PTES (Penetration Testing Execution Standard)** – A standard for conducting penetration tests
- **OSSTMM** – Open Source Security Testing Methodology Manual
- **NIST SP 800-115** – Technical guide to information security testing and assessment

---

Testing Phases

1. Reconnaissance

Activities included:

- WHOIS lookup
- DNS enumeration
- Subdomain discovery
- Technology fingerprinting
- Open-source intelligence (OSINT)

What is Reconnaissance?
Reconnaissance is the first phase of a penetration test where information about the target is

gathered. This can be passive (gathering information without directly interacting with the target) or active (directly interacting with the target).

Tools:

Tool	Description
Amass	DNS enumeration and network mapping tool
Subfinder	Subdomain discovery tool
theHarvester	OSINT tool for gathering emails, domains, and IPs
Shodan	Search engine for internet-connected devices
Google Dorks	Advanced Google search operators for finding sensitive information

---

2. Scanning & Enumeration

Performed:

- Port scanning
- Service detection
- Banner grabbing
- SSL analysis
- SMB enumeration
- Directory brute-forcing

What is Enumeration?
Enumeration is the process of actively connecting to systems and extracting information such as usernames, shares, services, and other network resources.

Tools:

Tool	Description
Nmap	Network discovery and security scanning utility
RustScan	Fast port scanner written in Rust
Gobuster	Directory/file brute-forcing tool
Feroxbuster	Fast directory brute-forcing tool
Nikto	Web server vulnerability scanner
Enum4Linux	SMB enumeration tool

3. Vulnerability Analysis

Identified:

- Missing patches
- Weak configurations
- Misconfigured services
- Known CVEs
- Authentication weaknesses

What is a Vulnerability?

A vulnerability is a weakness in a system that can be exploited by a threat actor. Vulnerabilities can exist in software, hardware, or configuration.

What is a CVE?

CVE (Common Vulnerabilities and Exposures) is a publicly known cybersecurity vulnerability that has been assigned a unique identifier.

Tools:

Tool	Description
Nessus	Comprehensive vulnerability scanner
OpenVAS	Open-source vulnerability scanner
Burp Suite	Web application security testing platform
OWASP ZAP	Open-source web application security scanner
Nuclei	Vulnerability scanner using templates

---

4. Exploitation

Attempted exploitation of:

- SQL Injection
- Cross-Site Scripting
- Command Injection
- Authentication bypass
- Local File Inclusion
- Remote Code Execution
- SMB vulnerabilities

What is Exploitation?

Exploitation is the process of taking advantage of a vulnerability to gain unauthorized access, escalate privileges, or achieve other objectives.

Tools:

Tool	Description
Metasploit	Framework for developing and executing exploits
Hydra	Password cracking tool supporting multiple protocols
CrackMapExec	Post-exploitation tool for Active Directory
Evil-WinRM	Windows Remote Management shell
Impacket	Collection of Python classes for network protocols

---

5. Privilege Escalation

Windows:

- WinPEAS
- PowerUp
- Mimikatz

Linux:

- LinPEAS
- GTFOBins
- Kernel exploits

What is Privilege Escalation?
Privilege escalation is the process of gaining higher-level access rights than those initially granted. It can be vertical (gaining admin access) or horizontal (gaining access to another user's account).

Tools:

Tool	Description
WinPEAS	Windows privilege escalation enumeration
LinPEAS	Linux privilege escalation enumeration
PowerUp	PowerShell tool for Windows privilege escalation
Mimikatz	Post-exploitation tool for extracting credentials
GTFOBins	Collection of Unix binaries that can be used to bypass security

---

6. Post-Exploitation

Performed:

- Credential harvesting

- Sensitive file discovery
- Persistence validation
- Lateral movement assessment

What is Post-Exploitation?
Post-exploitation refers to activities performed after gaining initial access to a system, including maintaining access, harvesting credentials, and moving laterally across the network.

---

7. Reporting

Documented:

- Evidence
- Risk ratings
- Exploitability
- Business impact
- Remediation

SECTION 04 - Technical Findings

Finding 1

SQL Injection

Attribute	Detail
Severity	Critical
CVSS v3.1	9.8
OWASP Category	A03:2021 – Injection

Description

The Juice-Shop application fails to sanitize user input in multiple locations, allowing arbitrary SQL queries to be executed against the database. This vulnerability exists in the login functionality and search features.

What is SQL Injection?

SQL Injection is a code injection technique that allows attackers to execute malicious SQL statements that control a web application's database server. This can allow attackers to view, modify, or delete data they shouldn't have access to.

Affected Asset

- *https://localhost:3000/login*
- *https://localhost:3000/rest/product/search*
- *https://localhost:3000/rest/user/login*

Evidence

The following payload was successful in bypassing authentication:

```
admin' OR '1'='1'--
```

How SQL Injection Works

Input	Result
`admin' OR '1'='1'--`	SQL query becomes: `SELECT * FROM users WHERE username='admin' OR '1'='1'--' AND password=''`
`The -- comments out the password check`	Authentication bypass achieved
`The OR '1'='1' always evaluates to true`	All users returned in the query

Impact

- Database compromise
- Credential theft
- Data modification
- Remote code execution (where applicable)
- Complete system compromise

Proof of Concept

```
sqlmap -u "https://localhost:3000/login" --data="username=admin&password=test" --dbs
```

What is sqlmap?

sqlmap is an open-source penetration testing tool that automates the detection and exploitation of SQL injection vulnerabilities.

Recommendation

Recommendation	Description
Use prepared statements	SQL statements are compiled once and executed with parameter separation
Parameterized queries	User input is treated as data, not executable code
Input validation	Validate and sanitize all user input
Least-privilege database accounts	Database accounts should only have necessary permissions

---

Finding 2

Cross-Site Scripting (Stored)

Attribute	Detail
Severity	High
CVSS	8.2
OWASP Category	A03:2021 – Injection

Description

Juice-Shop stores user input without proper sanitization in the product review and user feedback features, allowing malicious JavaScript execution when other users view the page.

What is Cross-Site Scripting (XSS)?

XSS is a vulnerability that allows attackers to inject malicious scripts into web pages viewed by other users. In stored XSS, the malicious script is permanently stored on the server.

Affected Assets

- `/rest/product/reviews`

- */rest/user/feedback*

Payload

<script>alert(document.cookie)</script>

How XSS Works

Step	Description
1	Attacker injects malicious script via a review or feedback
2	Script is stored in the database
3	Other users view the page containing the review
4	Malicious script executes in victims' browsers
5	Attacker can steal cookies, session tokens, or redirect users

---

Impact

- Session hijacking
- Credential theft
- Phishing attacks
- Browser compromise
- Defacement of the application
- Redirection to malicious sites

Recommendation

Recommendation	Description
Output encoding	Encode data based on the context where it is displayed
Content Security Policy (CSP)	Browser security mechanism to prevent XSS
Input validation	Validate and sanitize all user input
Use secure libraries	Use XSS prevention libraries and frameworks

---

Finding 3

Weak Password Policy

Attribute	Detail

Severity	High
OWASP Category	A07:2021 – Identification and Authentication Failures

---

Description

Multiple user accounts in Juice-Shop use weak passwords that are susceptible to brute-force and dictionary attacks.

Evidence

Commonly used weak passwords discovered:

- `admin / admin`
- `admin / Password123`
- `test / test`
- `guest / guest`
- `joe / joe`
- `Summer2024`
- `Admin123`

What is a Brute-Force Attack?

A brute-force attack is a trial-and-error method used to obtain information such as passwords by systematically trying all possible combinations.

Impact

- Credential compromise through brute-force attacks
- Unauthorized access to user accounts
- Account takeover
- Lateral movement to other systems

Recommendation

Recommendation	Description
MFA (Multi-Factor Authentication)	Require two or more verification factors
Password complexity requirements	Minimum length, character types, and no common passwords
Account lockout	Lock accounts after failed login attempts

Password history	Prevent reuse of previous passwords

Finding 4

Missing Security Headers

Attribute	Detail
Severity	Medium
OWASP Category	A05:2021 – Security Misconfiguration

Description

The Juice-Shop application is missing critical HTTP security headers that help prevent various web-based attacks.

What are HTTP Security Headers?

HTTP security headers are response headers that provide security protections to web applications by instructing browsers on how to handle content and interactions.

Affected Headers

Header	Purpose
CSP	Content Security Policy – Prevents XSS and data injection
HSTS	HTTP Strict Transport Security – Enforces HTTPS
X-Frame-Options	Prevents clickjacking attacks
X-Content-Type-Options	Prevents MIME type sniffing

What Each Header Prevents

Header	Attack Prevented
CSP	Cross-Site Scripting (XSS), Data Injection
HSTS	SSL Stripping, Man-in-the-Middle attacks
X-Frame-Options	Clickjacking, UI Redressing
X-Content-Type-Options	MIME confusion, Remote Code Execution

Recommendation

Configure all recommended HTTP security headers:

```
Content-Security-Policy: default-src 'self'
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
```

Finding 5

Outdated Software

Attribute	Detail
Severity	Medium
OWASP Category	A06:2021 – Vulnerable and Outdated Components

Description

The server is running outdated software containing known vulnerabilities (CVEs) that could be exploited by attackers.

Affected Versions

Software	Installed	Latest	Vulnerabilities
Node.js	14.17.0	20.x.x	Multiple CVEs
Express	4.17.1	4.19.x	Security patches
SQLite	3.31.1	3.45.x	Multiple vulnerabilities

Why Outdated Software is Dangerous

Risk	Explanation
Known vulnerabilities	Publicly known exploits available
No security patches	Vulnerabilities remain unpatched
Compliance issues	May violate security standards
Increased attack surface	More potential entry points for attackers

---

Recommendation

- Update all software to supported versions

- Implement a patch management program
- Subscribe to security advisories
- Automate vulnerability scanning
- Test updates before deployment

SECTION 05 - Risk Ratings

CVSS v3.1 Rating Scale

Score Range	Severity
0.0	None
0.1 – 3.9	Low
4.0 – 6.9	Medium
7.0 – 8.9	High
9.0 – 10.0	Critical

What is CVSS?
CVSS (Common Vulnerability Scoring System) is a framework for rating the severity of security vulnerabilities. It considers factors like exploitability, impact, and complexity.

Risk Matrix

Likelihood	Impact	Risk Level
High	High	Critical
High	Medium	High
Medium	Medium	Medium
Medium	Low	Low
Low	Low	Low

How Risk is Calculated

$$Risk = Likelihood \times Impact$$

Factor	Description
Likelihood	How likely the vulnerability is to be exploited

Impact	The potential damage if the vulnerability is exploited

---

Summary of Findings

Severity	Count
Critical	2
High	4
Medium	6
Low	5
Informational	8

Prioritization

Immediate (0-7 Days)

- SQL Injection
- Remote Code Execution
- Privilege Escalation

Short Term (30 Days)

- Cross-Site Scripting (XSS)
- Weak Authentication
- Missing Security Headers

Long Term (90 Days)

- Security awareness
- Infrastructure hardening
- Continuous monitoring

SECTION 06 - Remediation Plan

Phase 1 (Immediate)

- Patch critical vulnerabilities

- Disable vulnerable services
- Rotate compromised credentials
- Enable MFA

Phase 2 (30 Days)

- Upgrade software
- Remove unnecessary services
- Configure WAF
- Harden server configurations

What is a WAF?

A Web Application Firewall (WAF) helps protect web applications by filtering and monitoring HTTP traffic between a web application and the internet.

Phase 3 (60-90 Days)

- Conduct secure code review
- Employee security training
- Continuous vulnerability scanning
- Security monitoring (SIEM)
- Backup verification

What is SIEM?

SIEM (Security Information and Event Management) provides real-time analysis of security alerts generated by applications and network hardware.

Security Hardening Checklist

- MFA enabled
- Principle of Least Privilege
- Patch Management
- Endpoint Protection
- Network Segmentation
- Secure Backups
- TLS 1.3

- Logging & Monitoring
- IDS/IPS
- Web Application Firewall

APPENDICES

Appendix A – Tools Used

Reconnaissance

Tool	Description
Nmap	Network discovery and security scanning
RustScan	Fast port scanning
Amass	DNS enumeration and mapping
Subfinder	Subdomain discovery
theHarvester	OSINT gathering
Shodan	Internet-connected device search

Web Testing

Tool	Description
Burp Suite Professional	Web application security testing platform
OWASP ZAP	Open-source web application scanner
SQLMap	Automated SQL injection detection
Nikto	Web server vulnerability scanner
Nuclei	Template-based vulnerability scanner
FFUF	Fuzzing tool for web applications
Feroxbuster	Directory brute-forcing tool

Exploitation

Tool	Description
Metasploit	Exploit development framework
Hydra	Password cracking and brute-forcing
CrackMapExec	Active Directory exploitation
Evil-WinRM	Windows Remote Management shell

Impacket	Network protocol manipulation

Privilege Escalation

Tool	Description
LinPEAS	Linux privilege escalation enumeration
WinPEAS	Windows privilege escalation enumeration
PowerUp	Windows privilege escalation PowerShell tool
Mimikatz	Windows credential extraction

Password Auditing

Tool	Description
John the Ripper	Password cracking
Hashcat	Advanced password recovery

Appendix B – Sample Commands

Sample Nmap Scan

```
nmap -sC -sV -O -Pn target.com
```

Explanation:

Flag	Description
`-sC`	Run default scripts
`-sV`	Service version detection
`-O`	Operating system detection
`-Pn`	Skip host discovery (assume host is up)

Sample SQLMap Command

```
sqlmap -u "https://target/login?id=1" --dbs
```

Explanation:

Flag	Description
`-u`	Target URL
`--dbs`	List all databases

Sample Nikto Scan

```
nikto -h https://target.com
```

Explanation:

Flag	Description
`-h`	Target host

Sample Gobuster Scan

```
gobuster dir -u https://target.com -w wordlist.txt
```

Explanation:

Flag	Description
`dir`	Directory enumeration mode
`-u`	Target URL
`-w`	Wordlist file

Appendix C – Educational Resources

What is OWASP?

The Open Worldwide Application Security Project (OWASP) is a non-profit foundation that works to improve software security. It provides:

- Security tools
- Documentation
- Best practices
- Security standards

Common Attack Types Explained

Attack Type	Description
SQL Injection	Inserting malicious SQL queries into input fields
XSS	Injecting malicious scripts into web pages
CSRF	Forcing users to perform unwanted actions
Command Injection	Executing system commands through input fields
Path Traversal	Accessing files outside the web root
IDOR	Accessing unauthorized data by modifying IDs

Security Best Practices

1. **Input Validation** – Always validate user input
2. **Output Encoding** – Encode all output data
3. **Secure Authentication** – Use MFA and strong passwords

4. **Least Privilege** – Give users minimal necessary permissions
5. **Defense in Depth** – Use multiple layers of security

Appendix D – References

- OWASP Web Security Testing Guide (WSTG)
- OWASP Top 10 (2021)
- Penetration Testing Execution Standard (PTES)
- OSSTMM 3
- NIST SP 800-115
- CVSS v3.1 Specification
- CIS Benchmarks
- MITRE ATT&CK Framework

Report Classification: Confidential

Prepared by: Victor Muthomi

Version: 3.22.7

Approval: _____