OWASP Juice Shop Exploratory Test Report

1. Core Functionality Testing

Navigation & Layout

- Header contains logo, search bar, account button, and language selector
- Hamburger menu provides access to main navigation options
- Product grid layout on homepage with pagination
- Cookie consent popup appears on first visit
- Success notification banner appears for completed security challenges

Product Management

Product Display

- Products shown in grid with images, names, and prices
- Pagination shows 12 items per page (37 total items)
- Scroll-triggered pagination when clicking next page
- Product cards clickable with detailed popup view

Product Reviews

- Reviews expandable/collapsible in product popup
- Shows reviewer email and comment text
- Like functionality available for each review
- Review count displayed (e.g., "Reviews (2)")

Customer Interaction

Feedback System

- Anonymous feedback submission available
- Rating slider implementation
- 160-character limit on comments
- Basic CAPTCHA implementation (math equation)

- Character counter for comment field

About Us Section

- Corporate history with Lorem ipsum content
- Customer feedback carousel
- Social media integration (Twitter, Facebook, Slack, Reddit)
- Press Kit and NFT options available

Photo Wall

- Gallery layout with multiple images
- Product and promotional imagery
- Hashtag functionality (#raischi #whoneedsfourlegs)

2. Security Vulnerabilities

Critical Issues

- SQL injection vulnerability in login functionality
- Exposed user credentials through SQL injection
- Weak CAPTCHA implementation
- Email addresses publicly visible in reviews
- No rate limiting on feedback submission

UI/UX Issues

- Unconventional scroll-triggered pagination
- Non-standard cookie consent implementation
- Inconsistent error handling
- Form validation issues

3. Performance Concerns

Response Time

- Delayed loading of paginated content
- Image loading performance issues

- UI lag during cart updates

Browser Compatibility

- Layout inconsistencies across browsers
- Form submission handling varies
- Image scaling issues in mobile view

4. Recommendations

Security Improvements

- Implement proper input validation
- Protect against SQL injection
- Hide user email addresses in reviews
- Strengthen CAPTCHA implementation
- Add rate limiting for submissions

UX Enhancements

- Improve pagination implementation
- Standardize error handling
- Optimize image loading
- Enhance mobile responsiveness
- Implement proper form validation

This report documents findings from exploratory testing of the OWASP Juice Shop application, identifying both intentional vulnerabilities (as part of its security training purpose) and general functionality issues requiring attention.

OWASP Juice Shop Bug Report

1. Security Vulnerabilities

SQL Injection

- Severity: Critical
- Description: Application vulnerable to SQL injection attacks in login functionality
- Impact: Unauthorized access to user credentials and sensitive data
- Steps to Reproduce:

```
Sql {email=' OR 1=1-- password=any}
```

Authentication Bypass

- Severity: Critical
- Description: Login form vulnerable to authentication bypass
- Impact: Unauthorized access to user accounts
- Steps to Reproduce: Use SQL injection to bypass password verification

Weak Input Validation

- Severity: High
- Description: Registration form lacks proper input validation
- Impact: Potential for malicious data injection
- Steps to Reproduce: Submit registration form with invalid data formats

2. UI/UX Issues

Pagination Implementation

- Severity: Medium
- Description: Content only loads when scrolled to bottom after clicking next page
- Impact: Poor user experience and confusion
- Steps to Reproduce: Navigate through product pages using pagination controls

HTTP Status Codes

- Severity: Medium
- Description: All pages return 200 OK status regardless of existence
- Impact: Misleading response codes affecting error handling
- Steps to Reproduce: Access any non-existent URL

Cookie Consent

- Severity: Low
- Description: Non-standard cookie consent implementation
- Impact: Potential non-compliance with privacy regulations
- Steps to Reproduce: Observe cookie consent popup on first visit

3. Functional Issues

Review System

- Severity: Medium
- Description: Email addresses publicly exposed in review section
- Impact: Privacy concern and potential for user targeting
- Steps to Reproduce: View product reviews

CAPTCHA Implementation

- Severity: Medium
- Description: Weak CAPTCHA using simple math equation
- Impact: Easily bypassed by automated scripts
- Steps to Reproduce: Observe CAPTCHA in feedback form

Cart Manipulation

- Severity: High
- Description: Shopping cart vulnerable to price manipulation
- Impact: Potential financial loss
- Steps to Reproduce: Modify cart parameters through API endpoints

4. Performance Issues

Response Time

- Severity: Medium
- Description: Delayed loading of paginated content
- Impact: Poor user experience
- Steps to Reproduce: Navigate through multiple product pages

Server Stability

- Severity: High
- Description: Application crashes under automated tool load
- Impact: Service availability affected
- Steps to Reproduce: Execute multiple automated requests simultaneously

These bugs were identified through exploratory testing of the application's core functionality, security features, and user interface components.

Security and Logical Vulnerabilities Analysis

Endpoint: /client_registeration

1. SQL Injection Vulnerability (Risk Score: Critical - 9.8/10)

- The email parameter is directly concatenated into SQL query
- Exploit: Attacker can inject malicious SQL through the email field
- Impact: Database compromise, data theft
- Fix: Use parameterized queries instead of string concatenation

2. No Password Hashing (Risk Score: High - 8.5/10)

- Passwords are stored in plaintext
- Impact: Password exposure if database is compromised
- Fix: Implement bcrypt or Argon2 password hashing

3. No Input Validation (Risk Score: Medium - 6.5/10)

- Minimal validation only checks for empty fields

- Impact: Malformed data storage, potential XSS
- Fix: Implement proper input validation and sanitization

Endpoint: /client_login

1. Insecure JWT Implementation (Risk Score: Critical - 9.5/10)

- Hardcoded secret key ('123456')
- No signature verification in JWT decode
- Impact: Token forgery possible
- Fix: Use strong secret key from environment variables

2. SQL Injection in Login (Risk Score: Critical - 9.8/10)

- Direct string concatenation in login queries
- Impact: Authentication bypass possible
- Fix: Use parameterized queries