Technical Security Assessment Report

Target: OWASP Juice Shop
Tester: Ahmed Ali Mostafa Mohamed
Tools Used: Burp Suite, SQLmap, Browser DevTools, Manual Testing
Assessment Type: Black-box Web Application VAPT

---

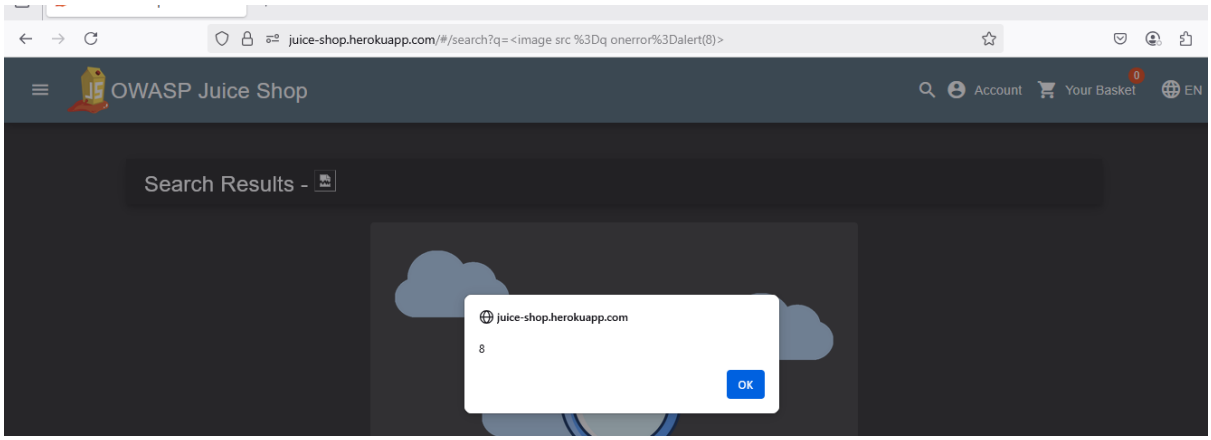
1. Reflected Cross-Site Scripting (XSS)

Description: User input in the search field is directly reflected in the page without proper sanitization, allowing JavaScript execution.

Vulnerable URL: https://juice-shop.herokuapp.com/#/

POC

1-Enter this payload in search block `<img src=x onerror=alert(8)>`



Severity: Medium

Remediation:

- Sanitize and encode all user input before reflecting it in the DOM.
- Use libraries such as DOMPurify.
- Implement strict Content Security Policy (CSP).

---

2. Broken Access Control – Feedback Submission Manipulation

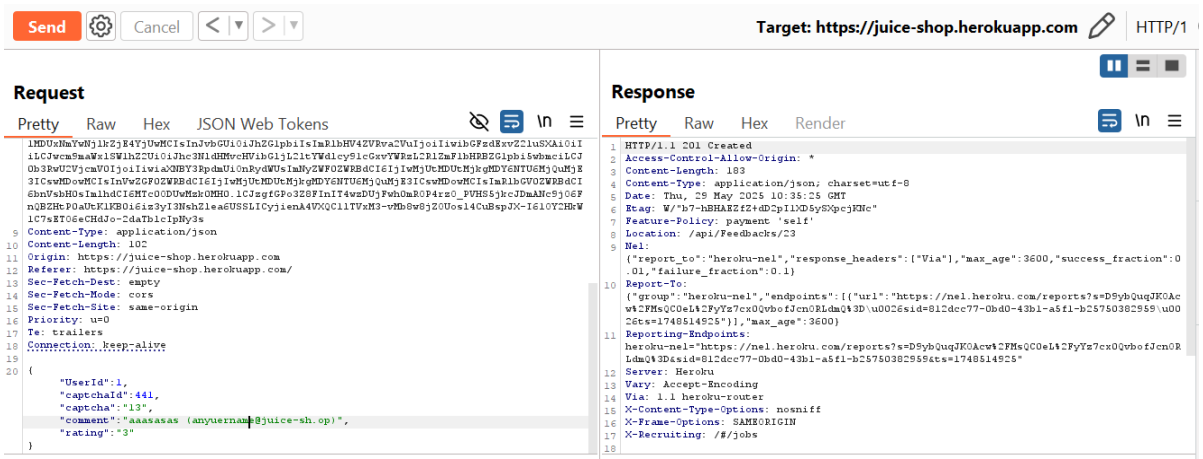
Description: The backend allows modification of the `user` parameter in feedback submissions. An attacker can impersonate other users.

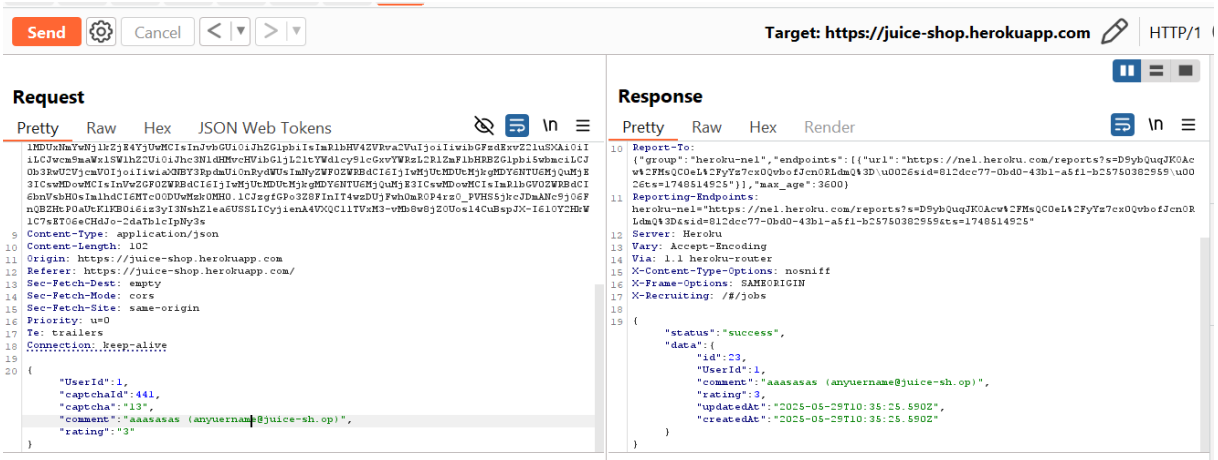
POC

1-Go to https://juice-shop.herokuapp.com/#/contact

2-Enter any massage and intercept request by burb suit

3-Changing the "email" value in the intercepted feedback request to another valid username results in feedback being stored under that identity.





Severity: High

Remediation:

- Never trust client-side user input for sensitive fields.
- Associate feedback with the authenticated user server-side only.
- Enforce proper authorization checks in backend APIs.

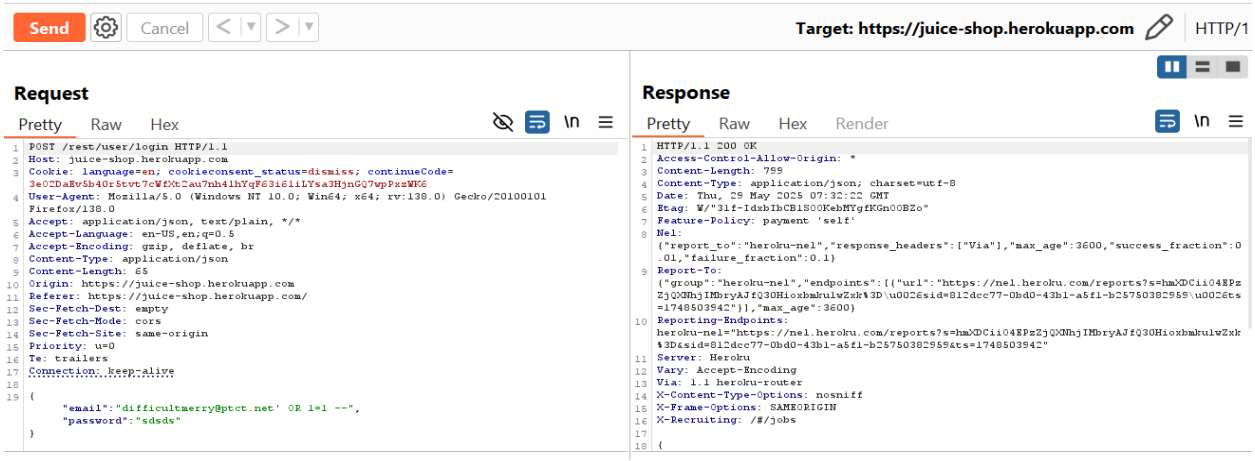
---

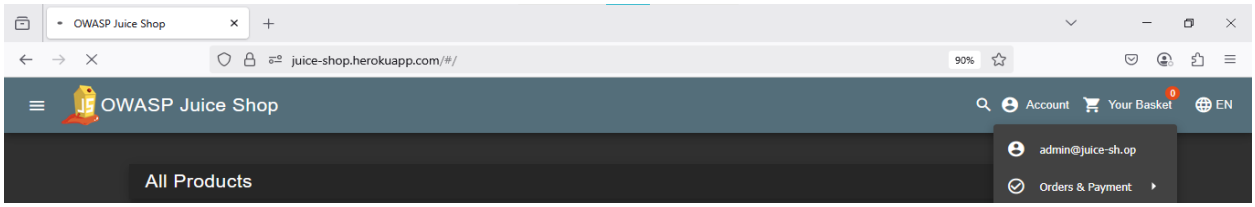
3. Broken access control Via SQL Injection – Login Bypass to Admin Account

Description: The login endpoint is vulnerable to SQL injection, allowing authentication bypass.

POC

1- **Go to login page**
2- **Enter difficultmerry@ptct.net' OR 1=1 --** and anything in password block.





Impact: Full access to admin account.

Severity: Critical

Remediation:

Use parameterized queries or ORM frameworks.

- Avoid dynamic SQL concatenation.
- Implement input validation and prepared statements.
- Monitor logs for suspicious login behavior.

---

4. Unprotected File Disclosure – `/ftp` Endpoint Via Information disclosure

Description: The application exposes a directory `/ftp` containing sensitive files such as `.bak`, `.pyc`, `.kdbx`, and internal documents.

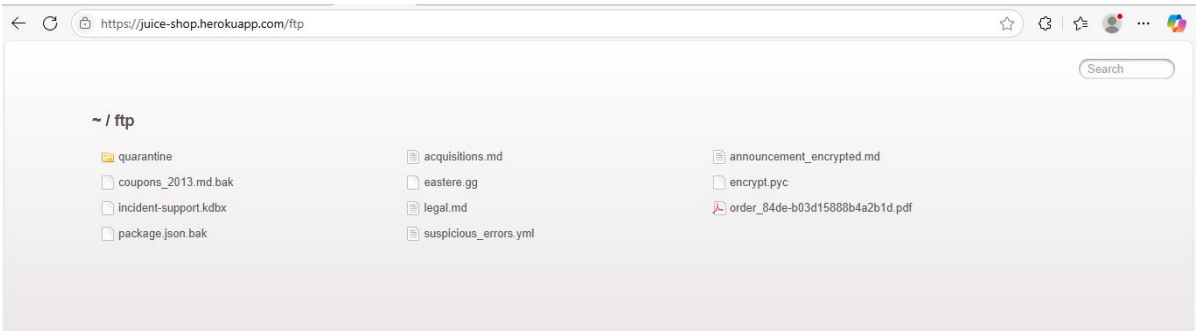
Files Observed:

- `package.json.bak`
- `incident-support.kdbx`
- `encrypt.pyc`
- `order_*.pdf`

POC

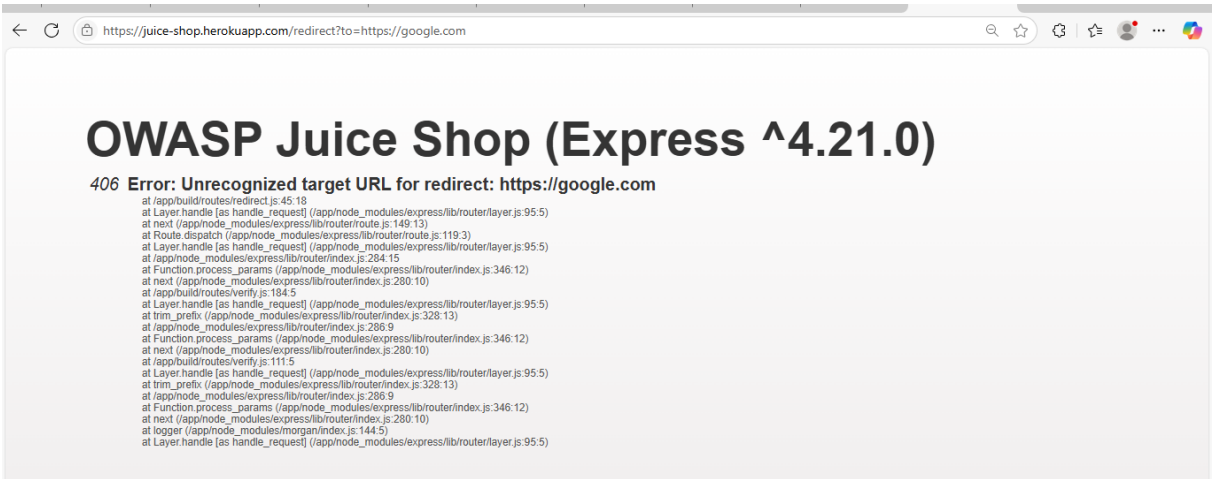
1-Go to https://juice-shop.herokuapp.com/robots.txt#/

2- Go to https://juice-shop.herokuapp.com/ftp



3- Go to https://juice-shop.herokuapp.com/redirect?to=https://github.com/juice-shop/juice-shop in github section.

4-repalce path github (information discloser based in error).



Security: High

Remediation:

- Disable directory listing in the web server.
- Restrict access to internal/backup/configuration files.
- Move sensitive files outside the web root.
- Require authentication for file access if necessary.

Executive Summary (Non-Technical Section)

This report highlights several high-impact vulnerabilities found in the Juice Shop web application. These issues can be exploited by attackers to:

- Steal user data or session information (via XSS).
- Forge feedback under another user's identity (Broken Access Control).
- Gain unauthorized access to admin-level features (SQL Injection).
- Access confidential internal files including passwords, source code, and sensitive documents.
- Information discloser

Business Impact:

- Loss of customer trust
- Potential data breaches
- Legal liabilities and regulatory penalties

Recommendation to Management:

- Immediately patch the vulnerabilities highlighted in this report.
- Educate developers on secure coding practices.
- Conduct regular security assessments and code reviews.
- Implement stronger access controls and input validation mechanisms.

Prepared by: Ahmed Ali Mostafa Mohamed
Digital Egypt Pioneers | Bug Bounty Researcher

EMAIL: omarahmedali569@gmail.com

Phone: +201006737824