Web Penetration Testing

الطالبة : شهد احمد محمد شكري عودة الله

الطالبة : دجى مدالله محمود الرواجفة

DOCUMENT VERSION CONTROL

Data Classification *– **Client Confidential***

Client Name	
Project name	Web Penetration Testing
Authors	
Approved by	
Version	
Submission Date	

Table of contents

1: Executive Summary

This report: documents the findings after testing web application. A series of tests were conducted against the targeted scope, using testing tools and where appropriate, manual testing techniques, to establish the presence of actual or potentially exploitable security vulnerabilities, which if exploited could result in direct or indirect damage to Customer Name. These key findings (classified as "High", "Medium" or "Low") have been highlighted in this report. Please refer to the Detailed Penetration Testing Results of the report for in depth details the key findings, their associated risks, and recommended actions.

The following represents the definition and the description of each severity rate.

Impact	Description
High	A vulnerability that can be exploited by the attacker and cause huge damage to application components and data.
Medium	A vulnerability that can be exploited by the attacker and cause moderate damage to application components and data.
Low	A vulnerability that can be exploited by the attacker to understand application underlying technologies and versions which can be utilized in further attacks.

1.1 Scope Details

The scope of evaluation and testing covered the following assets:

#	Host	Platform
1		OWASP Juice Shop

2: Detailed Penetration Testing Results

2.1 Introduction

Simulated e-commerce web application with login, file upload, shopping cart, and comment features

2.2 Restrictions

practical penetration test on a vulnerable web application, identify key security flaws, and document findings using standard tools and methodologies

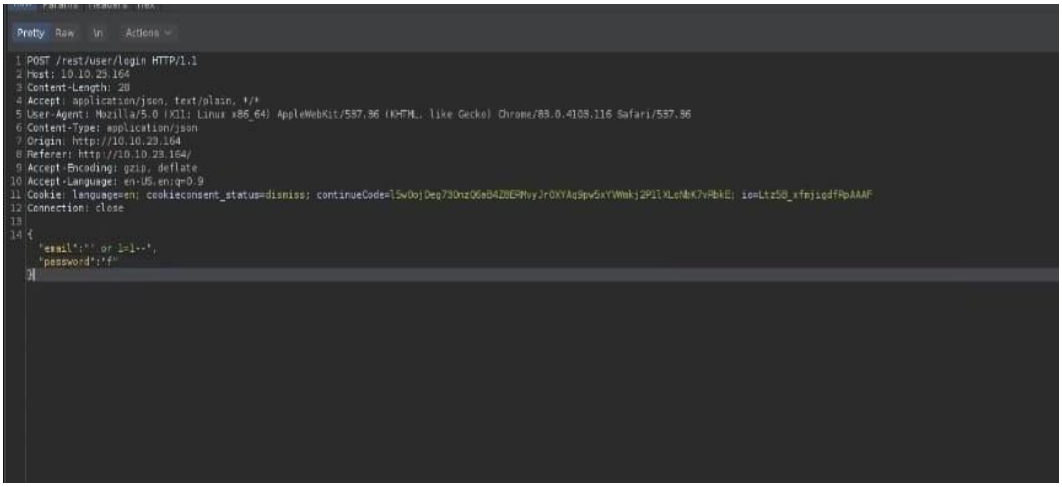
2.3 Tools

1. Burp Suite - for intercepting and modifying HTTP requests
2. OWASP ZAP - for automated vulnerability scanning
3. curl - for manual request testing
4. Chrome Developer Tools - for inspecting DOM and client-side behavior

2.4 Discovered vulnerability

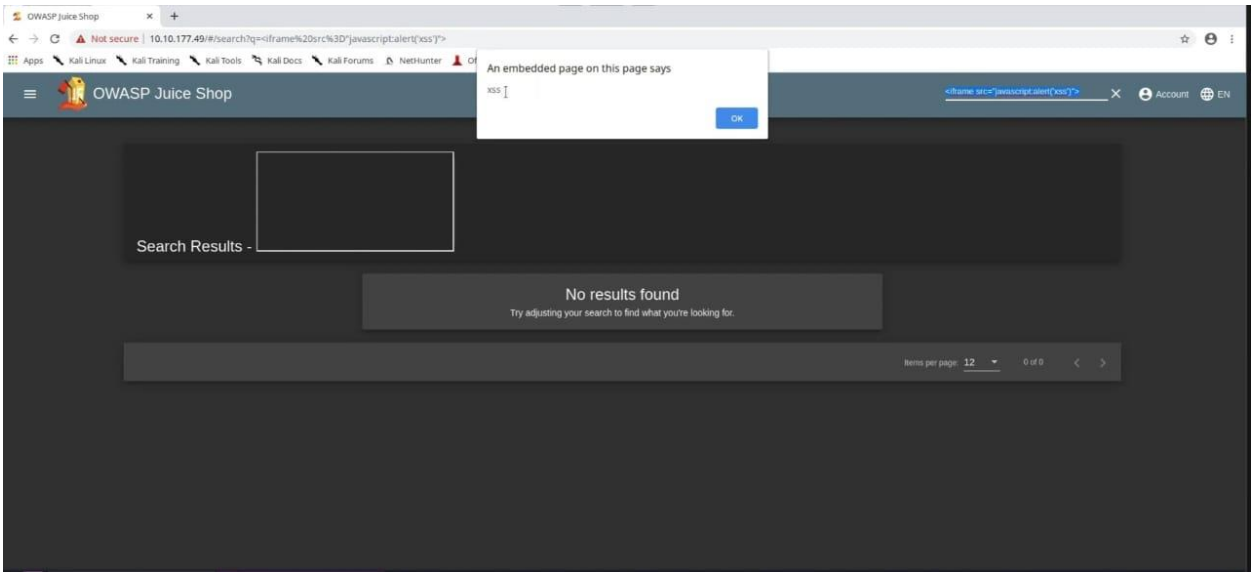
1. SQL Injection (**High**)

   Location: User login endpoint
   Discovery Method: Injected' OR 1=1--
   Exploitation: Allowing login without valid credentials
   Impact: Exposure of user and product data
   Recommended Fix: Implement parameterized queries (Prepared Statements)



2. Stored XSS (**High**)

Location: Product comment section
Discovery Method: injected <iframe src="javascript:alert('xss')">
Exploitation: Execution of malicious JavaScript in users' browsers
Impact: Session hijacking and UI manipulation
Recommended Fix: Sanitize user input and encode output using functions



3. File Upload Vulnerability (**High**)

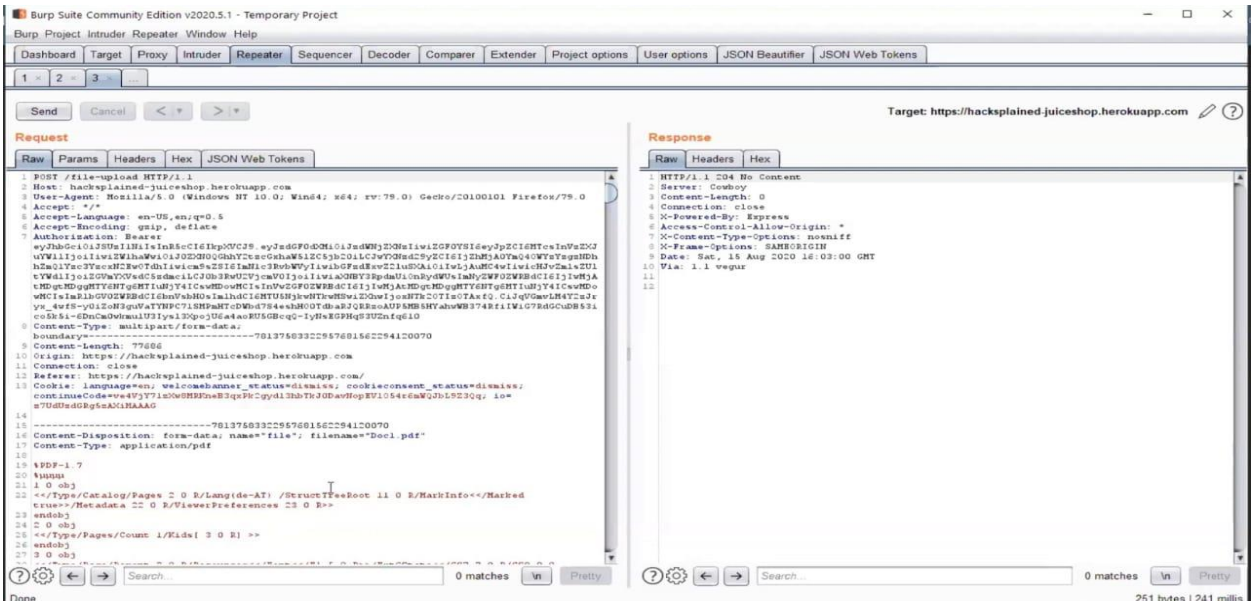
Location: File upload endpoint

Discovery Method: Uploaded a file containing executable code

Exploitation: Remote code execution on the server

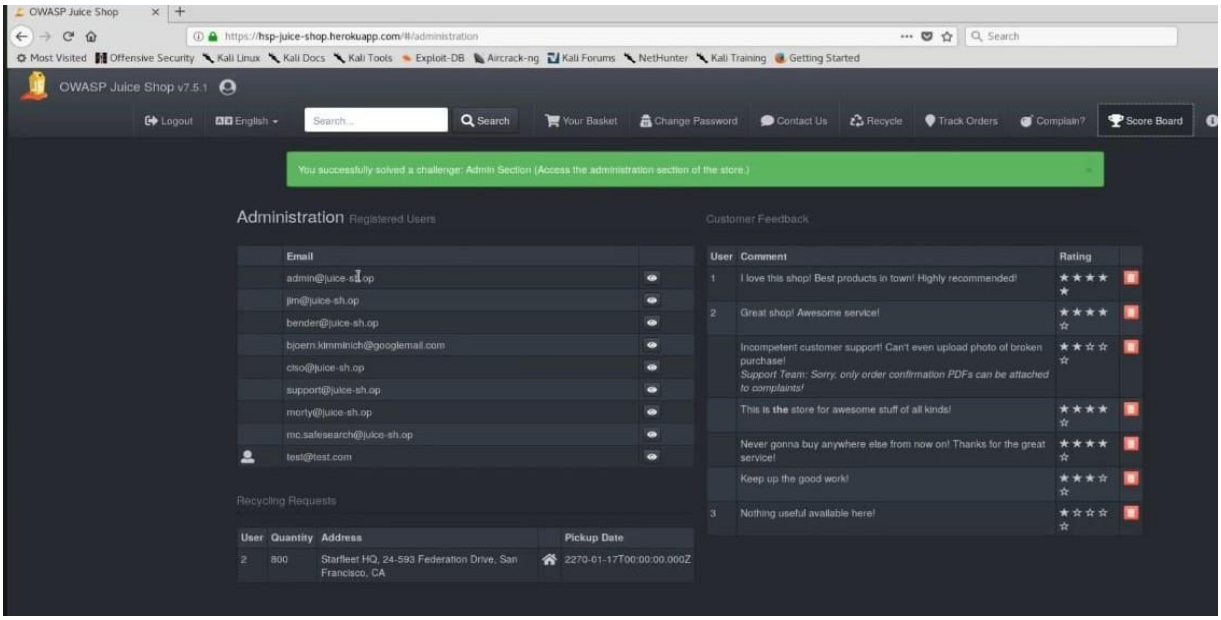
Impact: Full system compromise

Recommended Fix: Validate file type, extension, and content before saving



4. Broken Access Control (High)

Location: Administration section
Discovery Method: Direct access to the hidden `/administration` endpoint without proper authorization checks
Exploitation: Unauthorized data access and modification
Impact: Violation of user privacy and data integrity
Recommended Fix: Enforce server-side authorization checks for all administrative endpoints



3: Conclusion

A practical penetration test was conducted on the OWASP Juice Shop application, revealing multiple vulnerabilities across key functionalities. Each flaw was documented with exploitation techniques, and recommended mitigations.

These findings highlight the importance of secure coding practices and robust access control mechanisms in web applications.