SQL Injection Penetration Test Report on OWASP Juice Shop

Prepared by: Elroy Fernandes
Date: January 13, 2026
Environment: Kali Linux (VM)
Target: OWASP Juice Shop v19.1.1
Test Type: SQL Injection Assessment

Table of content

Executive Summary

This report documents a successful SQL injection attack against the OWASP Juice Shop login page conducted in a controlled, local environment. The vulnerability allowed authentication bypass using a simple SQL payload (‘ OR 1=1- -), granting unauthorized administrative access. This assessment was performed for educational purposes to demonstrate common web application vulnerabilities and remediation strategies.

***Key Finding**: High Severity SQL Injection (CWE-89) on login functionality.*
***Impact:** Complete authentication bypass, full admin privileges.*
***Status:** Proof of Concept Achieved*

Scope and Objectives

Scope

- **Application:** OWASP Juice Shop (Intentionally vulnerable web application)
- **Environment:** Local Kali Linux deployment
- **Target Component:** User Login page
- **Testing Method:** Manual penetration testing

Objectives

- Identify and exploit authentication vulnerabilities
- Demonstrate SQL injection techniques
- Document findings and remediation steps
- Showcase ethical security testing methodology

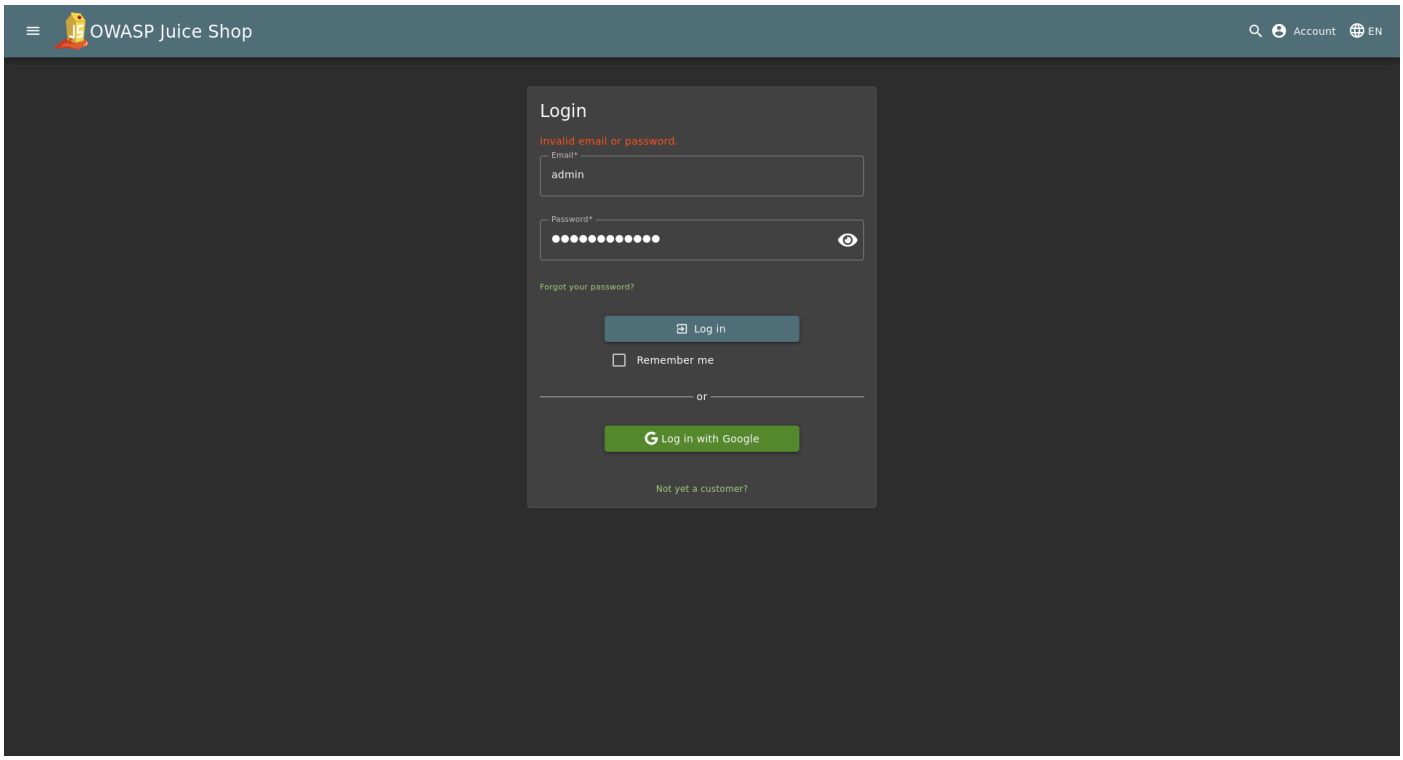
Technical Finding: SQL Injection (Auth Bypass)

Description

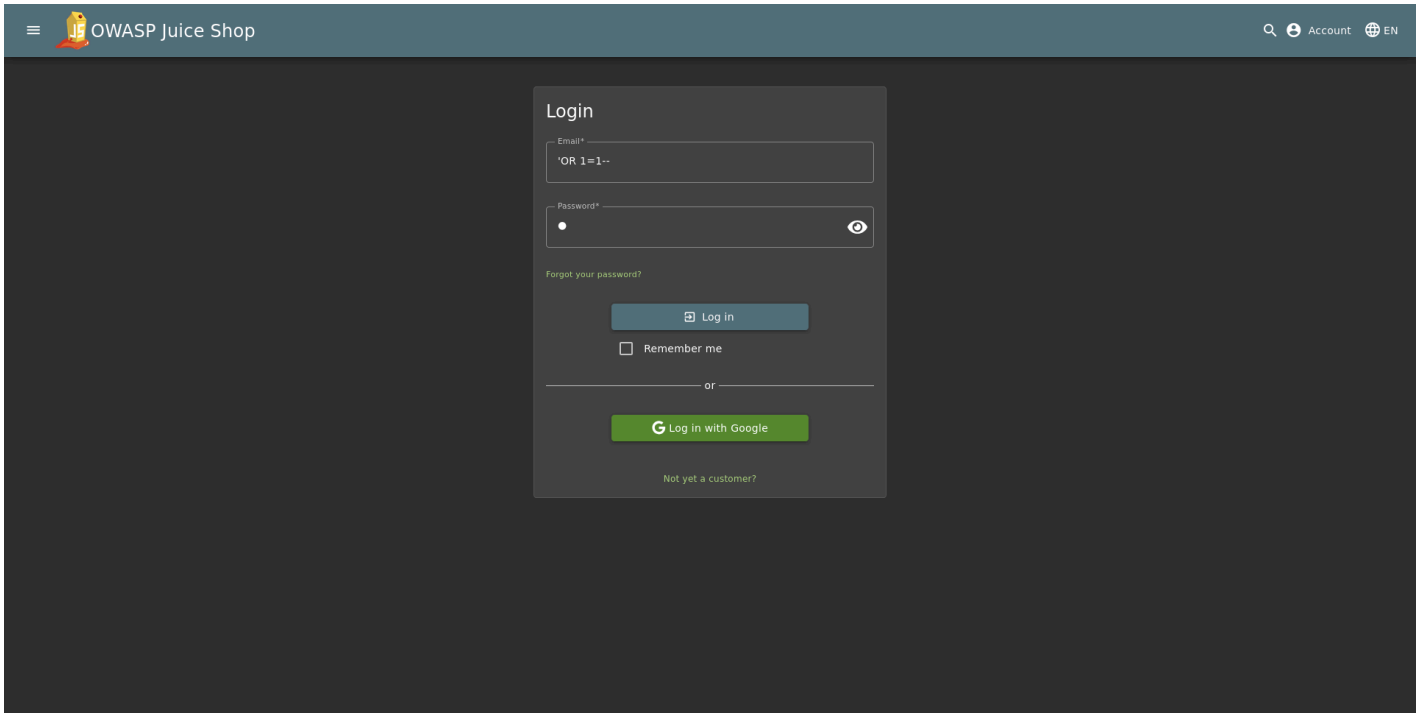
The login form does not use parameterized queries or prepared statements when validating user credentials. By injecting SQL syntax into the email/username field, an attacker can easily manipulate the back-end database query logic to always return “True”, effectively logging the attacker into the first account in the database (typically the Admin)

Steps to Reproduce (Proof of Concept)

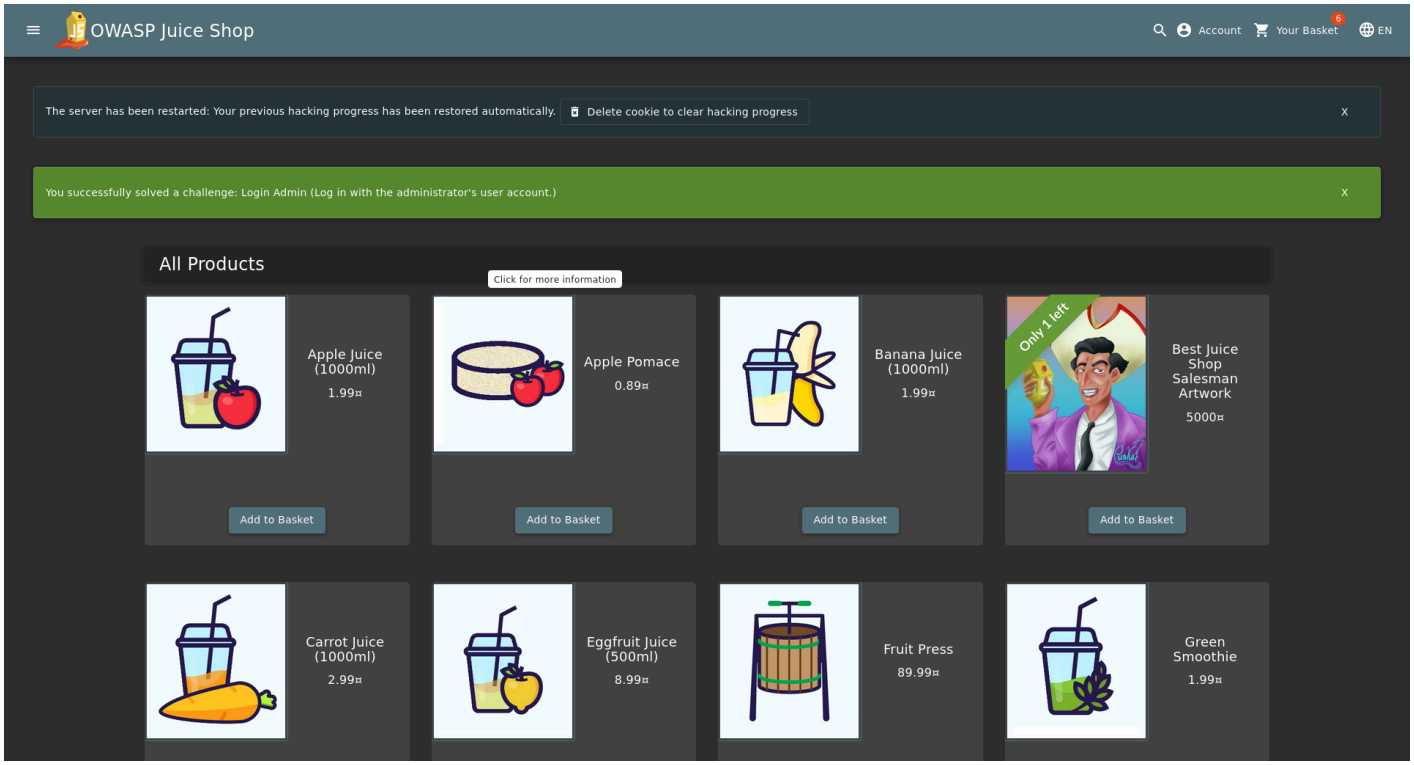
1. Navigate to the login page of the OWASP Juice Shop.

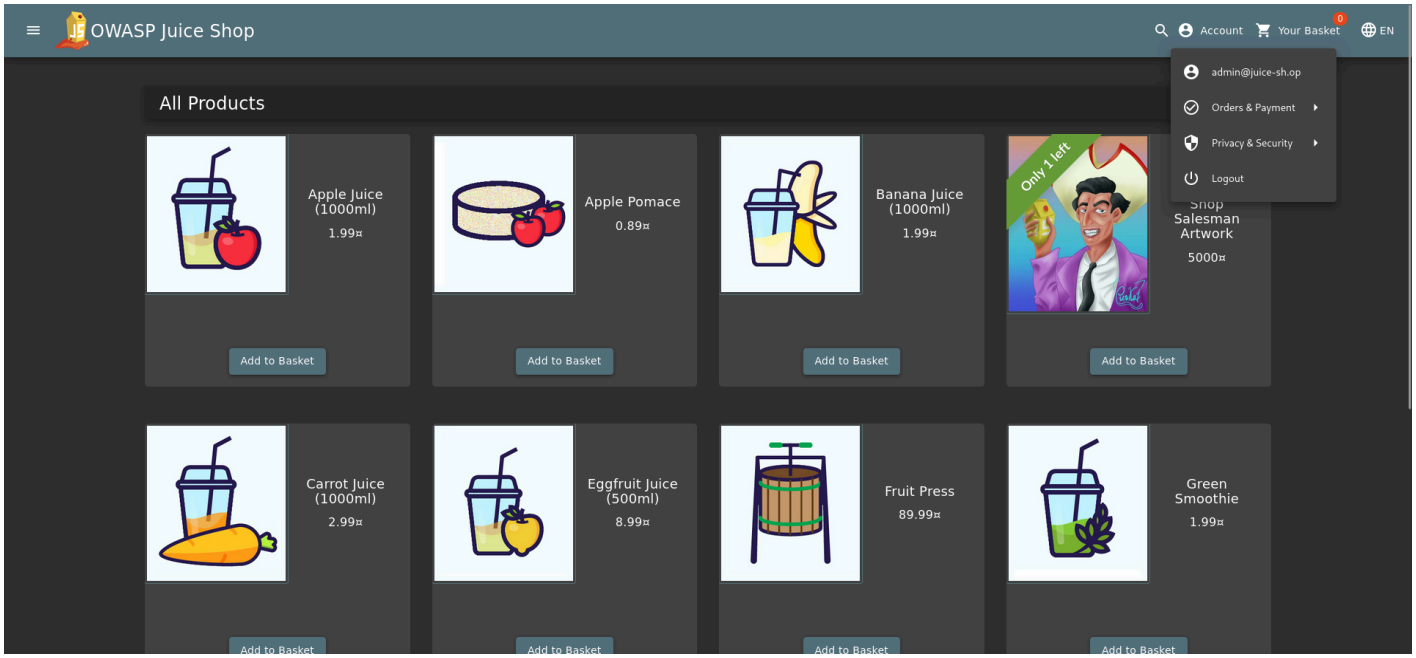


2. In the Email/Username field, enter the following payload ‘ **OR 1=1 - -**



3. Enter any arbitrary string in the Password field (e.g: SomethingPassword1)
4. Click Login.
5. ***Result***:The Application processes the query, ignores the password check due to the comment operator (- -), and authenticates the user as the admin@juice-sh.op





Impact

- **Severity:** Critical
- **Confidentiality:** High (Access to all user data and PII)
- **Integrity:** High (Ability to modify products, reviews, and user account)
- **Availability:** Medium (Potential to delete records or drop tables)

Technical Analysis

The SQL injection vulnerability exists because:

1. **No Input Validation:** Username field accepts special SQL characters (‘, - -,etc.)
2. **No Parameterized Queries:** Application uses string concatenation instead of prepared statements.

Remediation recommendations

To prevent this, the development team should implement the following:

1. **Use PreparedStatements:** This ensures the database treats the input as data only, not as executable code.
2. **Input Validation:** Whitelist allowed characters in username field (alphanumeric + underscore only). Reject inputs containing SQL keywords (SELECT, OR, - -,etc.)

Conclusion

This penetration test successfully demonstrated a critical SQL injection vulnerability in the OWASP Juice shop authenticated mechanism. The vulnerability allows an unauthenticated attacker to bypass login controls and gain full administrative access using a simple payload. Implementation of parameterized queries and input validation will effectively remediate this issue. This assessment highlights the importance of secure coding practices and the need for regular security testing in application deployment.

***Report Status**: Complete*
***Severity Level:** High*
***Remediation:** Timeline: Immediate*