# **Final project (report-01)**

# 

**Team leader : Omar Abdelaal Saad**

**ID: 2305165**

**Member 2 : Mustafa Aly Mustafa**   
**ID : 2305616**   
**Member 3 : Mohammed Osama Mohammed**   
**ID : 2305180**

**Executive Summary**   
  
**Purpose of the Test**:  
  
We identify the vulnerabilities in Juice Shop and analyze how attackers can exploit them.  
  
**High-level impact and critical vulnerabilities.**

1. Presence of an XSS vulnerability.
2. Lack of rate-limiting led to multiple brute-force attacks, resulting in system compromise.
3. Hidden administrative paths can be discovered through random URL enumeration.

**Summary of Recommendations:**

1. Implement login attempt rate-limiting.
2. Restrict access to administrative paths.

**Scope and Methodology**  
  
**Tools Used:**

* **Burp Suite**
* **Hydra X**
* **Kali Linux**

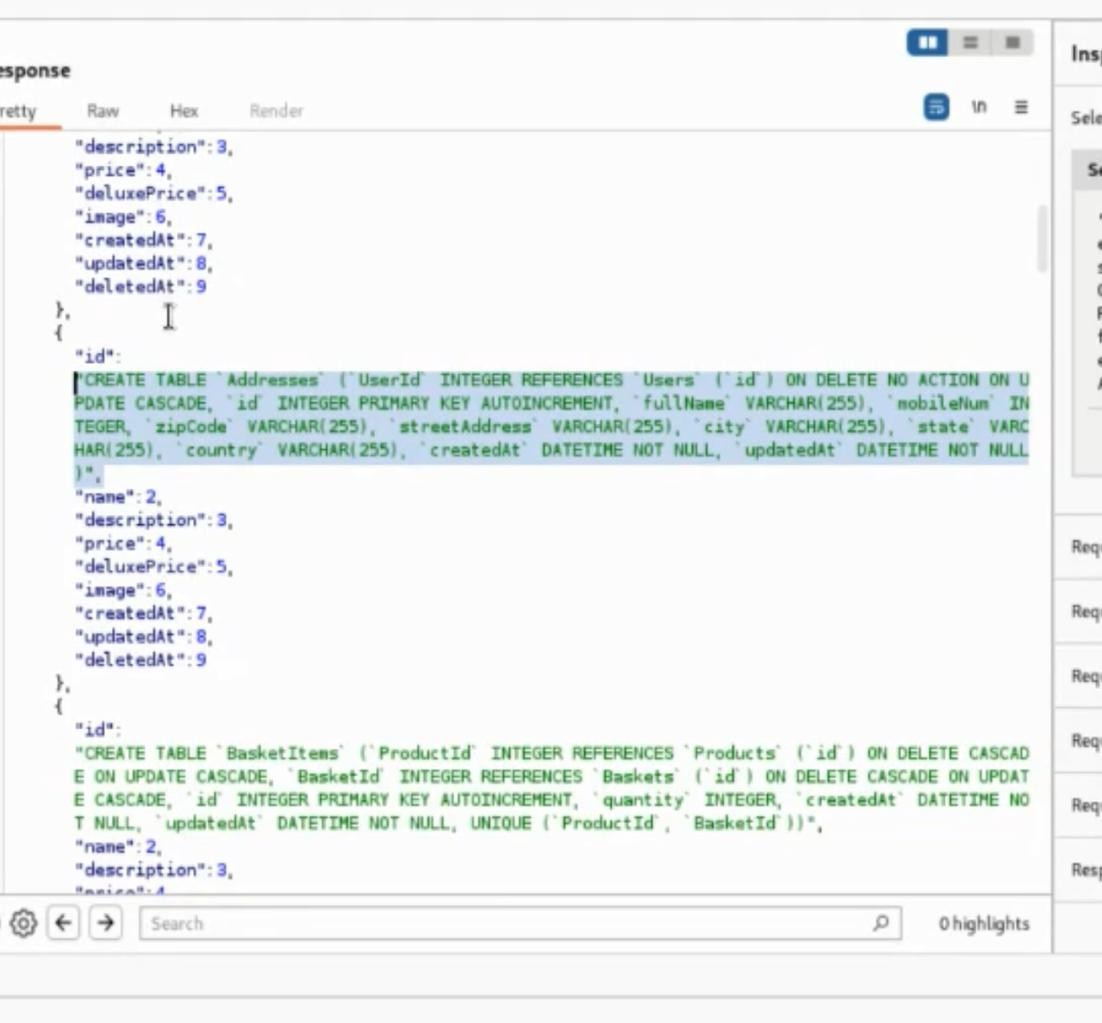
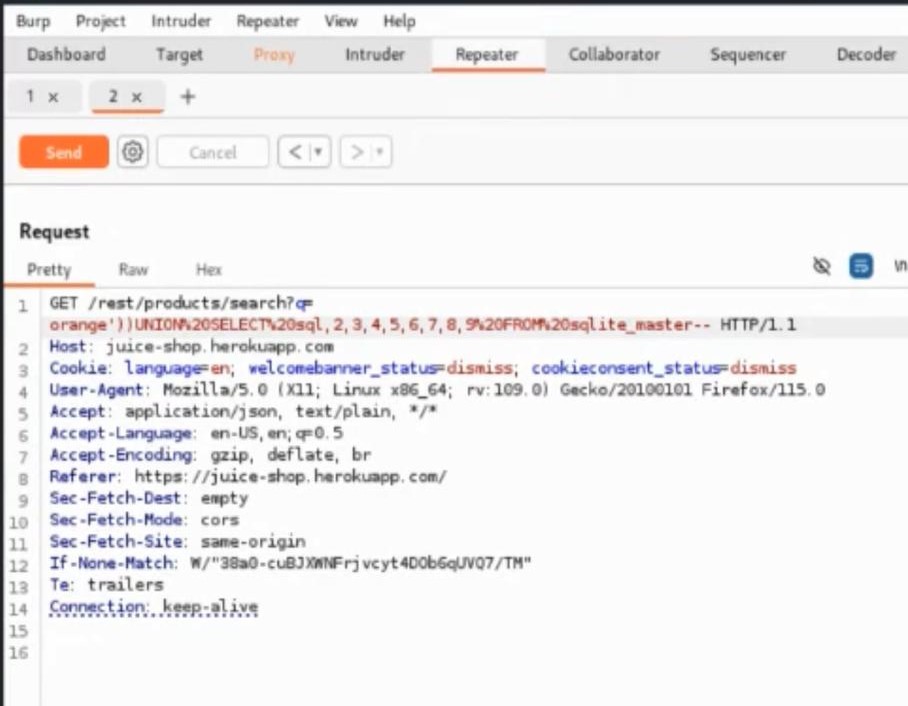
**Vulnerability Findings**

**1. Critical Vulnerability:** **Enumeration**

**a. Description:**  
  
The vulnerability allows me to know all the contents of the schema and identify the name of the database the website is operating on.

**b. Risk & Impact:**  
  
It becomes easy to know everything related to the database, meaning I can have full access to it through SQL Injection.

**c. Remediation:**   
  
Using regular expressions.

**d. Evidence**  


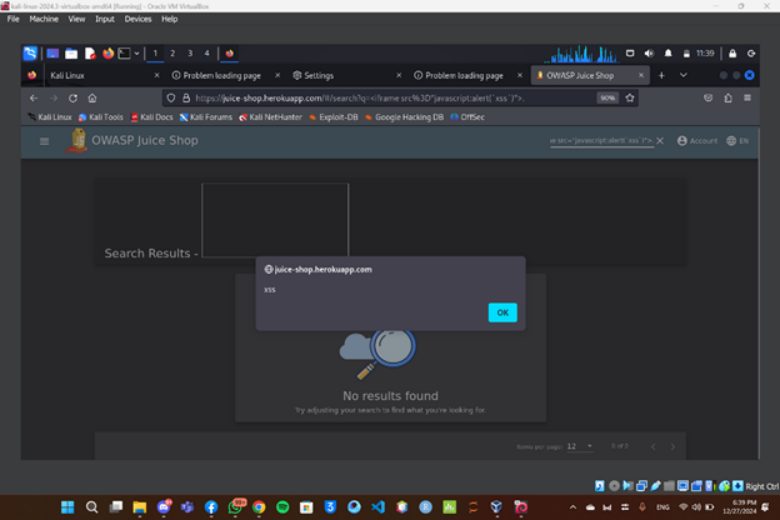
**2. Critical Vulnerability:** **XSS in Product Search**

**a. Description:**   
  
The vulnerability is exploited by injecting a malicious JavaScript payload (<iframe src=javascript:alert('xss')>) into the search bar. The application reflects this input back without any sanitization or validation, causing the script to execute in the user's browser.

**b. Risk & Impact:**   
  
This flaw allows an attacker to:  
Execute arbitrary JavaScript in the victim’s browser.  
Steal session cookies or other sensitive data.  
Redirect users to malicious websites.

**c. Remediation:**   
  
Use Security Tools: Regularly test your app with tools like Burp Suite to catch issues early.

**d. Evidence:**

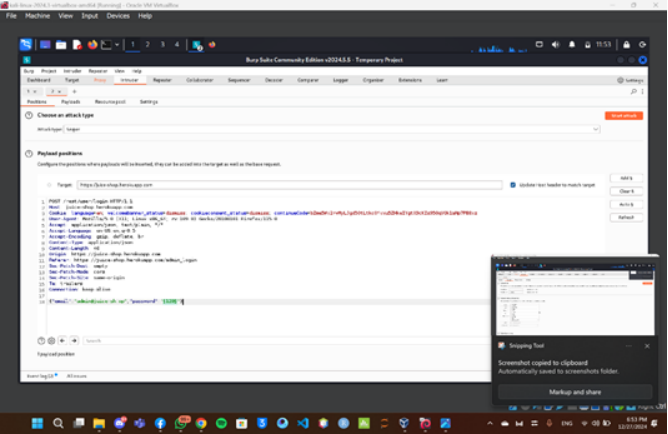


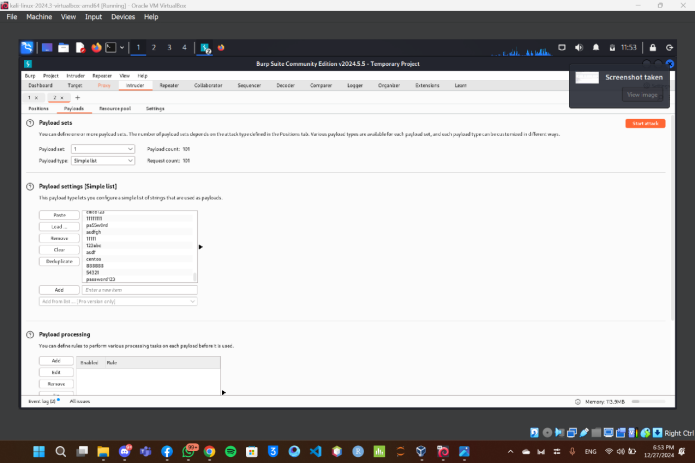
**3. Critical Vulnerability:** **Brute Force on Login Page**

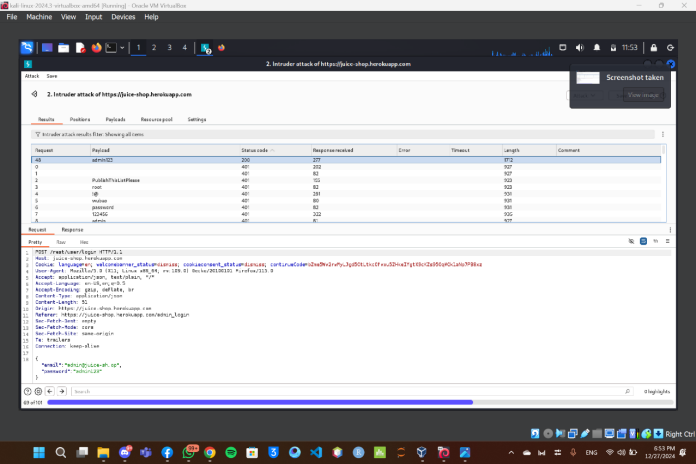
**a. Description:**   
  
The login page lacks adequate protection mechanisms, such as rate-limiting or account lockouts. This allows attackers to use automated tools (like Burp Suite) to try various username-password combinations until a valid one is found.

**b. Risk & Impact:**   
  
1. Credential Compromise: Attackers can gain unauthorized access to accounts, including admin accounts.  
2. System Abuse: A successful brute-force attack could lead to data theft, privilege escalation, and compromised user trust.

**c. Remediation:**  
  
1. Logging and Monitoring: Detect and respond to brute-force attempts by monitoring login activity.  
2. Multi-Factor Authentication (MFA): Add an extra layer of security to reduce reliance on password strength alone.  
3. Make a counter for the authentication trials.

**d. Evidence:**  
  






**4. Critical Vulnerability:** **SQL Injection in Login Page (bonus)**

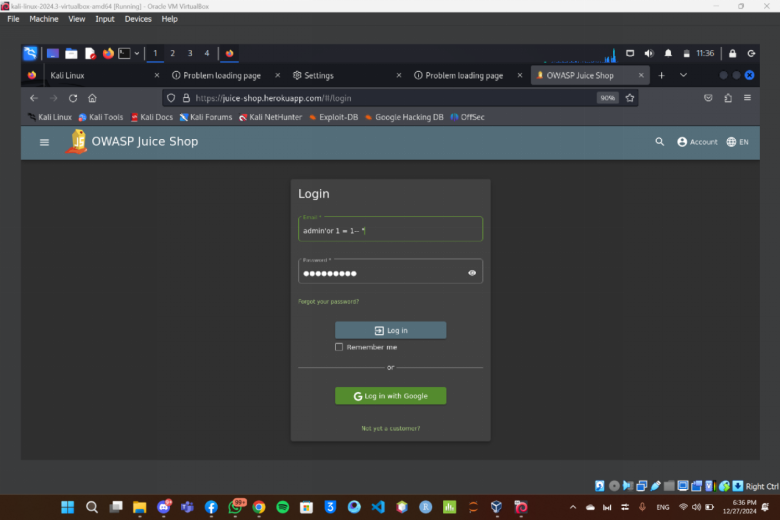
**a. Description:**   
  
The login page is vulnerable to SQL Injection, allowing attackers to bypass authentication. By entering malicious SQL code  
such as: admin' OR 1=1 --

The attacker forces the database to accept the condition as always true, granting unauthorized access to the admin account.

**b. Risk & Impact:**   
  
1. Authentication Bypass: Attackers can gain admin-level access without valid credentials.  
2. Data Breach: Sensitive user data and application integrity are at risk

**c. Remediation:**

Monitor logs for suspicious activity to detect and respond to attacks.

**c. Evidence:**  
  


**5. Critical Vulnerability:** **view basket (bonus)**

**a. Description:**   
This technique allows me to access the user's basket by inspecting the website. I go to Inspect → Network → Reload, then search for the ID of my basket, change it, and send the request. This opens a different basket for the user whose ID I entered. If I want to access the admin basket, I would use ID 1, as it's the first account created and assigned the first ID on the website.

**b. Risk & Impact:**   
  
1. Sensitive information like user details and product prices can be leaked.

2.Data manipulation is possible, allowing price changes, unauthorized discounts, or access to discount codes tied to specific baskets.

3.This can aid in XSS attacks, where cookies can be stolen, or the user can be redirected to other pages to steal their data.

**c. Remediation:**

Monitor logs for suspicious activity to detect and respond to attacks.

**Exploitation and attacks**  
  
We used:

* **Burp Suite** to perform brute-force attacks.

The owner's account was successfully accessed through brute-force attacks, allowing us to gain full control over the website.

**Conclusion**  
  
The OWASP Juice Shop website contains numerous critical vulnerabilities that pose significant risks to both the site and its registered users if exploited.

**This link contains the team's videos.**  
[**https://drive.google.com/drive/folders/16Uowzsb1rUOqjfxevjLQDuxttRBfnIXW**](https://drive.google.com/drive/folders/16Uowzsb1rUOqjfxevjLQDuxttRBfnIXW)