

***Alexandria National University***

***CYBER SECURITY ENGINEERING***

**Name : Mazen Mohamed gaber ID : 2305176**

**Name : ziad khaled ID : 2305148**

**Name : kamal Mohamed kamal ID 2305175**

**Web Penetration Testing Report: OWASP Juice Shop**

**Executive Summary**

**This penetration test was conducted to assess the vulnerability of the admin login page to brute force attacks using Burp Suite. Key findings include the lack of effective security controls such as rate-limiting, strong password enforcement, and failure to log and monitor suspicious activities. Recommendations include implementing rate-limiting mechanisms, strengthening password policies, and improving logging and alerting systems.**

**Scope and Methodology**

**Scope: This assessment tested the OWASP Juice Shop application, its APIs, and exposed endpoints.Methodology: The testing followed a black-box approach.Tools Used: The following tools were used - Hydra, Burp Suite,** Kali Linux

By: Mazen Mohamed Gaber:

**1. Brute Force on Admin Credentials**

**1-Description: When you run the Hydra command, the tool will test multiple passwords from your wordlist. If Hydra finds a valid combination of username and password, it will clearly display the result in the terminal.**

2-Tools and Environment Setup:

•Tools: Hydra, Kali Linux.

•Target: The OWASP Juice Shop admin login page.

1-Hydra command: hydra -l admin@juice-sh.op -P /home/kali/pass.txt -s 3000 localhost http-post-form "/#/admin/login:email=^USER^&password=^PASS^:Invalid email or password"

2-Explanation of the Command

1. -l admin@juice-sh.op: Specifies the target username (admin email in this case).

2. -P /path/to/passwordlist.txt: Specifies the path to your password list (e.g., rockyou.txt or common.txt).

3. -s 3000: Specifies the port number of the Juice Shop (default is 3000 if it’s running locally).

4. localhost: Specifies the target server (use localhost if the app is running on your system).

5. http-post-form: Tells Hydra to perform an HTTP POST request to the login form.

6. "/#/admin/login:email=^USER^&password=^PASS^:Invalid email or password":

• The first part (/#/admin/login) is the login endpoint.

• The form data fields (email=^USER^&password=^PASS^) indicate where Hydra will substitute the and password from the lists.

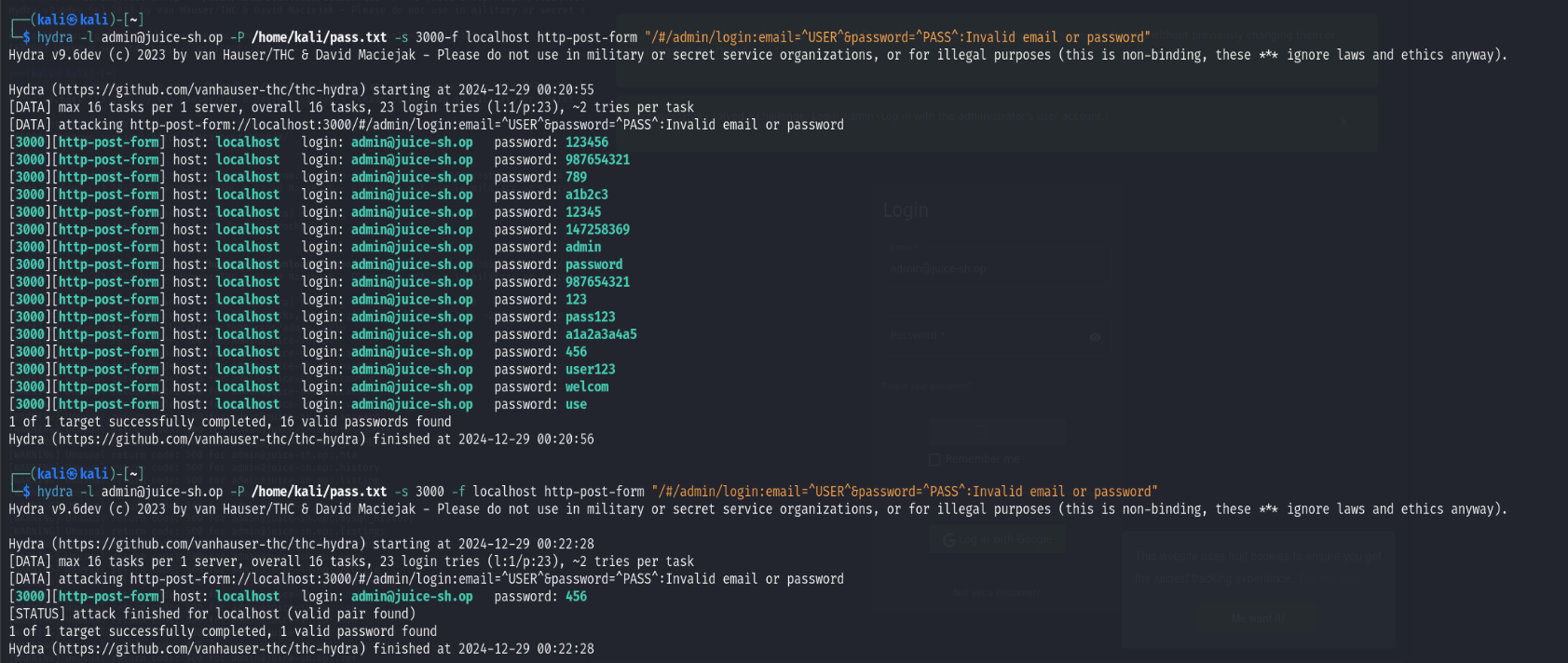
3- After Running the Command

•Success: If Hydra finds the correct password, it will display the successful credentials in the terminal.

•Failure: If no valid credentials are found, ensure your password list contains strong candidates.

4-Out put:

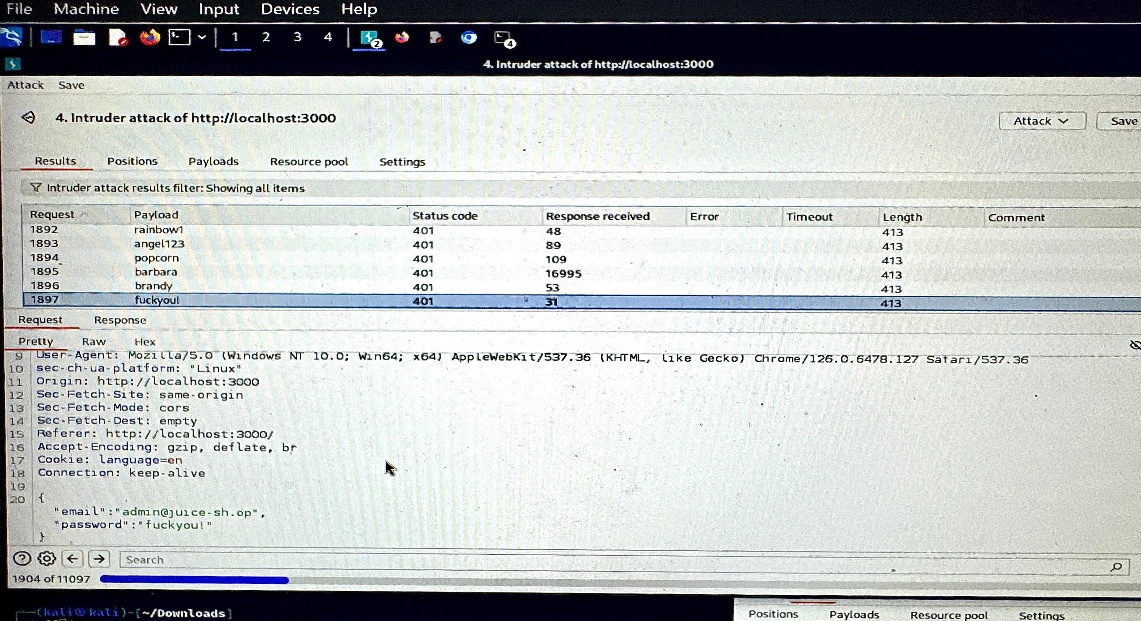
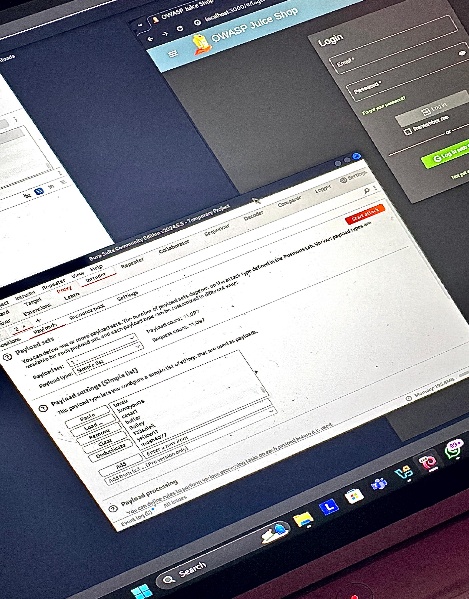
when the password is found, Hydra will show something like this:

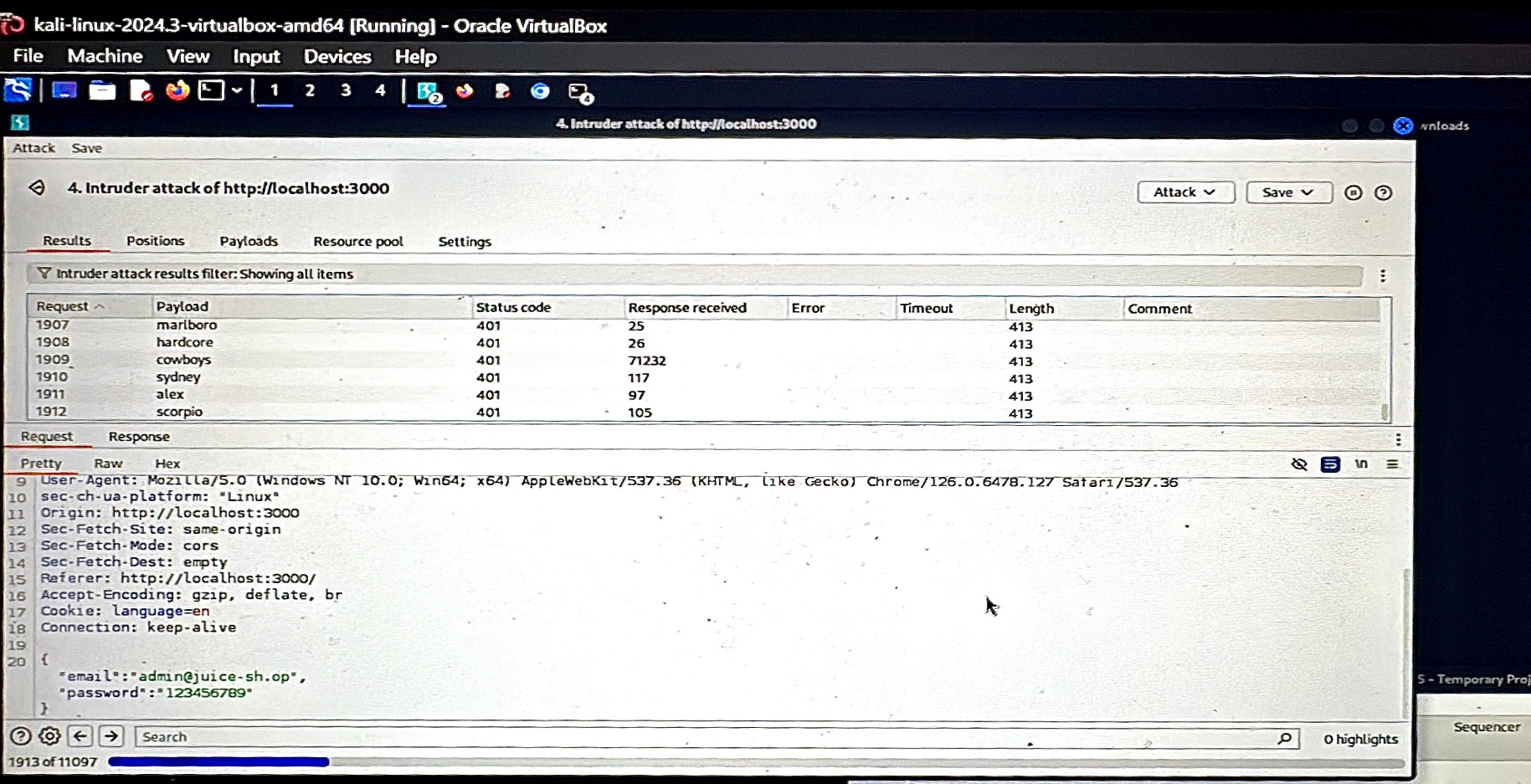
[3000][http-post-form] host: localhost login: admin@juice-sh.op password: admin123

Another tool: (Burp suite)

Description: A Brute Force attack on an Admin account using Burp Suite is a penetration testing technique where multiple combinations of usernames and passwords are systematically tried to gain unauthorized access to an account. Here’s how you can perform this process ethically and responsibly:

Out put: if status:401 worng password but 200: this is the password



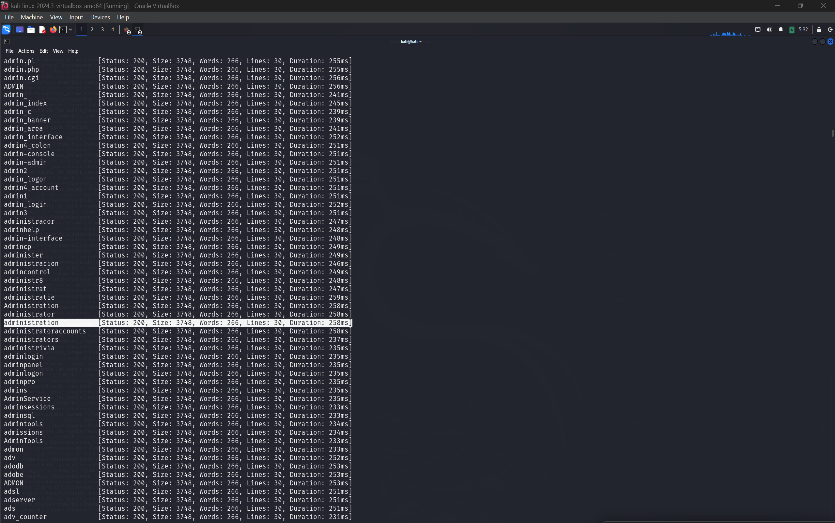
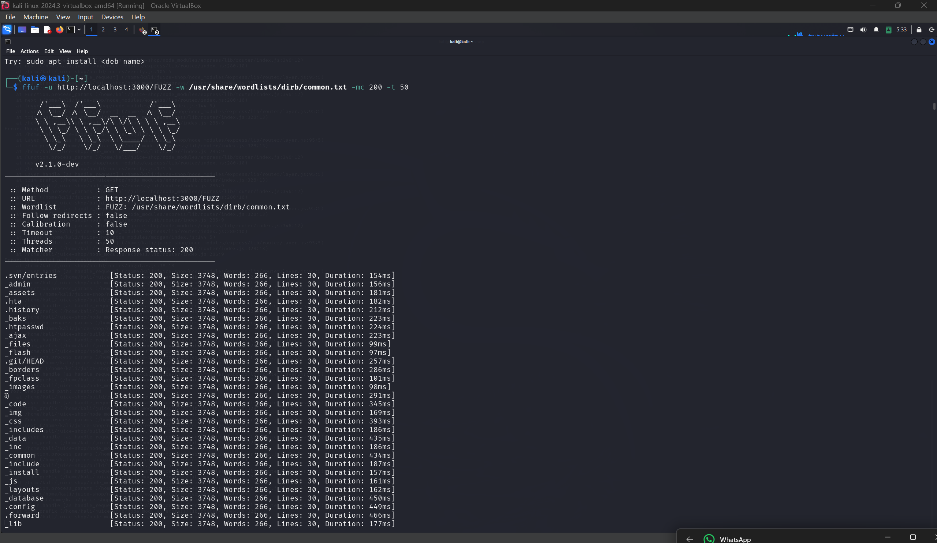


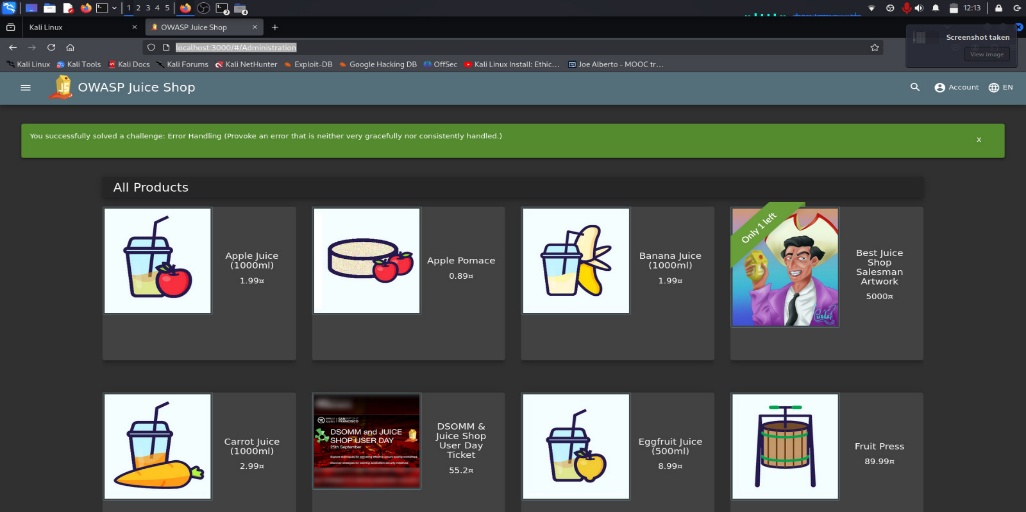
**By: ziad khalid fathy**

**2. Enumeration to Find Admin Path**

**Description: The attacker identifies hidden admin paths by examining the structure of URLs or utilizing tools such as FFUF (Fuzz Faster U Fool). This tool performs fast and efficient fuzzing to uncover hidden directories and endpoints within web applications, helping attackers exploit vulnerabilities .**

**Remediation: Secure sensitive URLs by obfuscating them and enforce strong authentication mechanisms for admin functionalities to prevent unauthorized access.**

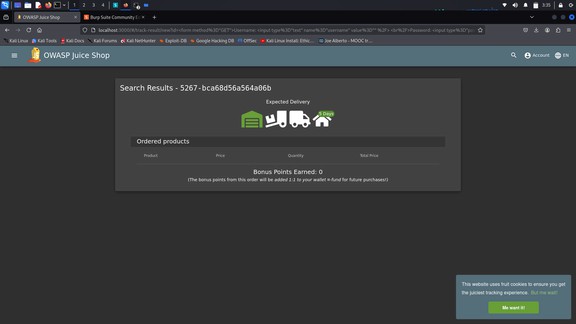
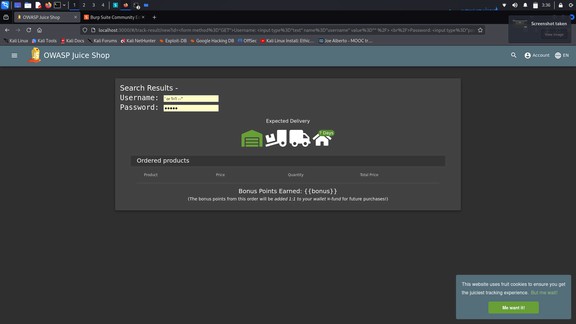
****

****

**BY: kamal Mohamed kamal**

**3. XSS in Product Search**

**Description: Malicious scripts are input into the product search bar, executing in the victim's browser.Risk: High. Enables session theft, redirection, or malicious activities.**



**Remediation: Sanitize user inputs and encode outputs.**

4-bonus Attacks:

1- missing logout fuctionallity

Description: Missing Logout Functionality: Verify if the application properly terminates the user session upon logout. Ensure no session tokens remain active after logging out.

