WEB APPLICATION SECURITY ASSESSMENT REPORT

(Comprehensive Vulnerability Analysis & Penetration Testing)

Project Details

Project Title: Web Application Security Testing – OWASP Juice Shop

Task Number: FUTURE_CS_01

Client / Organization: *Future Interns*

Application Under Test (AUT): OWASP Juice Shop (Docker Deployment)

CIN ID : FIT/OCT25/CS4249

Submission Date: *16/11/2025*

Prepared By: *Jeetesh Maurya)*

📚 TABLE OF CONTENTS

1. Executive Summary

This security assessment focused on identifying vulnerabilities in the OWASP Juice Shop web application, an intentionally insecure platform designed for hands-on security testing.
The assessment simulated real-world penetration testing activities that attackers commonly perform, including authentication testing, input validation checks, access control validation, and security misconfiguration assessment.

The testing revealed **three key vulnerabilities**, each of which could be exploited under real-world conditions:

- Username Enumeration
- Reflected Cross-Site Scripting
- Directory Browsing / Sensitive Data Exposure

While Juice Shop is intentionally vulnerable, these findings mimic realistic security flaws frequently found in modern web applications.
Addressing such vulnerabilities would significantly strengthen any system's security posture.

2. Assessment Objectives

The main objectives of this engagement were:

- To identify common and impactful vulnerabilities in a controlled environment
- To understand exploitation techniques used by attackers
- To learn and implement OWASP Top 10 testing standards
- To gain hands-on experience with security tools like Burp Suite, OWASP ZAP, and Docker
- To prepare a formal security assessment document suitable for professional portfolio use

3. Scope of Work

3.1 In-Scope Components

Component	Status
Application URL	http://localhost:3000
Authentication Pages	Included
Search Functionality	Included
Public Endpoints (e.g., /ftp directory)	Included
API Endpoints (as captured via Burp HTTP History)	Included

3.2 Out-of-Scope Components

Component	Reason
Denial of Service Attacks	Not permitted for local system stability
Backend Database Testing	Requires internal access
Source Code Review	Not included in black-box assessment
Server OS Exploitation	Beyond scope of web app testing

4. Testing Environment

Parameter	Details
Operating System	Windows 11
Deployment Method	Docker Desktop
Application Container	bkimminich/juice-shop
Browser Used	Burp-Suite Chromium
Network Configuration	Localhost (no VPN or proxy interference)

5. Tools and Resources Used

📌 Primary Tools

- **Burp Suite Community Edition** – Interception, manual testing, request manipulation
- **OWASP ZAP** – Automated scanning and spidering
- **Docker Desktop** – Running the Juice Shop container

📌 Supporting Tools

- Windows CMD / PowerShell
- Chromium browser (bundled with Burp Suite)
- Firefox (optional for comparison)

📌 Standards & Documentation

- OWASP Testing Guide v4
- OWASP Top 10 (2021)
- CVSS v3.1 Risk Calculator

6. Methodology

This assessment was executed in four main phases:

6.1 Reconnaissance

- Identified accessible endpoints through browsing and proxy logs
- Captured all HTTP/HTTPS requests using Burp Proxy
- Noted application behavior and error messages
- Mapped directory structures such as /ftp

6.2 Manual & Automated Testing

Techniques used:

- Input fuzzing
- HTML/JS payload injection
- Parameter tampering
- Directory traversal attempts
- Authentication bypass attempts

Automated Scans:

- OWASP ZAP passive scan
- Burp Suite issue alerts

6.3 Exploitation

Verified vulnerabilities using:

- Specially crafted payloads
- Manipulated HTTP request headers
- Path guessing
- Manual navigation to sensitive endpoint

6.4 Verification & Evidence Collection

- Screenshots of vulnerable pages
- Screenshot of impacted request in Burp Suite
- Consistent replication of vulnerabilities

7. Detailed Vulnerability Analysis

7.1 Finding F-01: Username Enumeration

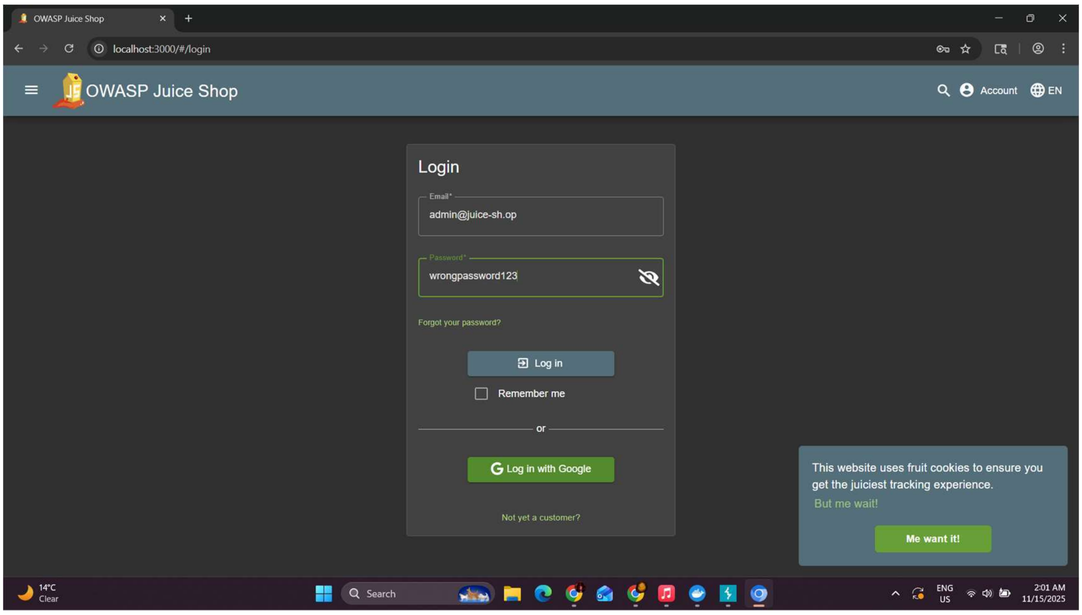
Severity: Medium
CVSS v3.1 Score: 5.3
OWASP Category: A07 – Identification & Authentication Failures

Description

The login functionality displays **different error messages** depending on whether a username exists.
This allows attackers to verify if an account is registered.

Technical Evidence

- Invalid user input → “User does not exist”
- Valid user input → “Invalid password”
  This discrepancy leaks user existence.



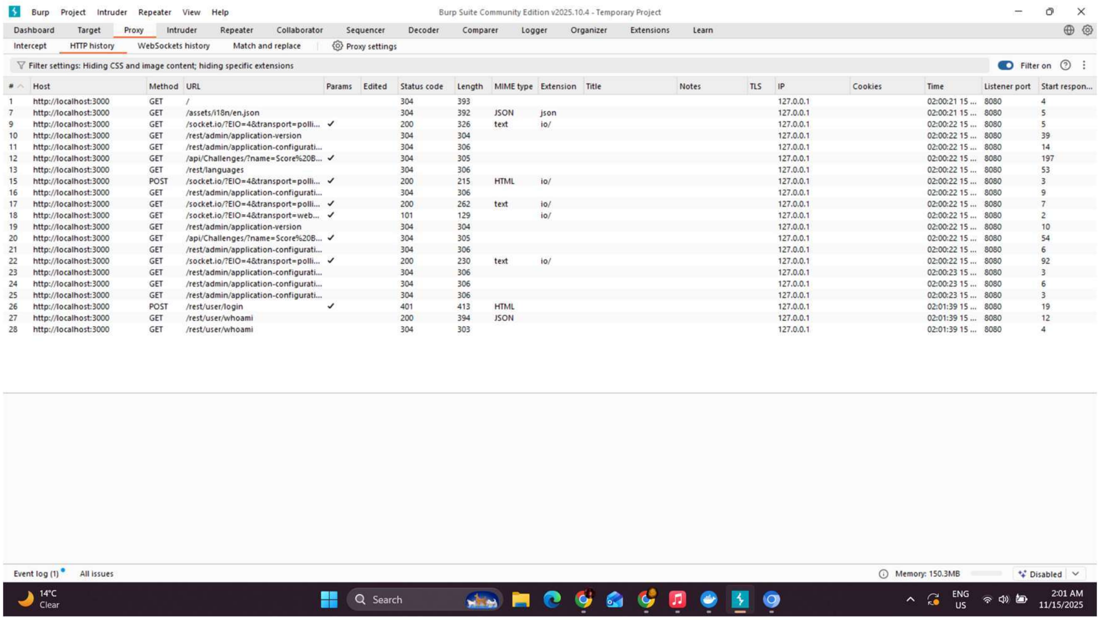
localhost:3000/#/login

Login

Email* admin@juice-sh.op

Password* wrongpassword123

This website uses fruit cookies to ensure you get the juiciest tracking experience.



Burp Suite Community Edition v2025.10.4 - Temporary Project

Filter settings: Hiding CSS and image content; hiding specific extensions

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port	Start respon...
1	http://localhost:3000	GET	/			304	393						127.0.0.1		02:00:21 15 ...	8080	4
7	http://localhost:3000	GET	/assets/i18n/en.json			304	392	JSON	json				127.0.0.1		02:00:21 15 ...	8080	5
9	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	326	text	io/				127.0.0.1		02:00:22 15 ...	8080	5
10	http://localhost:3000	GET	/rest/admin/application-version			304	304						127.0.0.1		02:00:22 15 ...	8080	39
11	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:22 15 ...	8080	14
12	http://localhost:3000	GET	/api/Challenges/?name=Score%20B...	✓		304	305						127.0.0.1		02:00:22 15 ...	8080	197
13	http://localhost:3000	GET	/rest/languages			304	306						127.0.0.1		02:00:22 15 ...	8080	53
15	http://localhost:3000	POST	/socket.io/?EIO=4&transport=polli...	✓		200	215	HTML	io/				127.0.0.1		02:00:22 15 ...	8080	3
16	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:22 15 ...	8080	9
17	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	262	text	io/				127.0.0.1		02:00:22 15 ...	8080	7
18	http://localhost:3000	GET	/socket.io/?EIO=4&transport=web...	✓		101	129		io/				127.0.0.1		02:00:22 15 ...	8080	2
19	http://localhost:3000	GET	/rest/admin/application-version			304	304						127.0.0.1		02:00:22 15 ...	8080	10
20	http://localhost:3000	GET	/api/Challenges/?name=Score%20B...	✓		304	305						127.0.0.1		02:00:22 15 ...	8080	54
21	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:22 15 ...	8080	6
22	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	230	text	io/				127.0.0.1		02:00:22 15 ...	8080	92
23	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:23 15 ...	8080	3
24	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:23 15 ...	8080	6
25	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:23 15 ...	8080	3
26	http://localhost:3000	POST	/rest/user/login	✓		401	413	HTML					127.0.0.1		02:01:39 15 ...	8080	19
27	http://localhost:3000	GET	/rest/user/whoami			200	394	JSON					127.0.0.1		02:01:39 15 ...	8080	12
28	http://localhost:3000	GET	/rest/user/whoami			304	303						127.0.0.1		02:01:39 15 ...	8080	4

Memory: 150.3MB

Impact

- Attackers can build a list of valid users
- Enables targeted password brute-force attacks
- Increases likelihood of credential stuffing attacks

Attack Scenario

An attacker automates username attempts through a bot:
If the system reveals correct usernames, attackers focus password attacks only on valid accounts.

Recommendations

- Standardize error messages to a generic:
  “Invalid username or password”
- Add CAPTCHA or rate-limiting
- Implement account lockout mechanisms

7.2 Finding F-02: Reflected Cross-Site Scripting (XSS)

Severity: Medium–High
CVSS v3.1 Score: 6.4
OWASP Category: A03 – Injection

Description

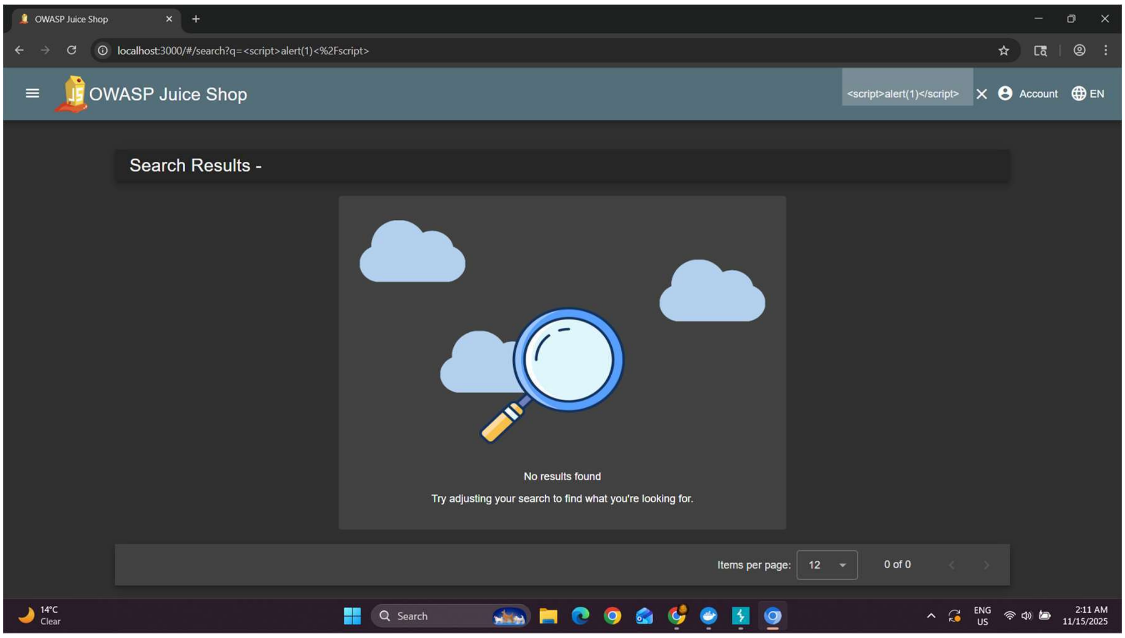
The search parameter reflects user input directly without sanitization. Testing with payload:

```
<script>alert(1)</script>
```

revealed potential XSS behavior (depending on version and context).

Impact

- Execution of attacker-injected JavaScript
- Theft of session cookies
- Redirection to malicious pages
- Potential account compromise



localhost:3000/#/search?q=<script>alert(1)<%2Fscript>

<script>alert(1)</script>

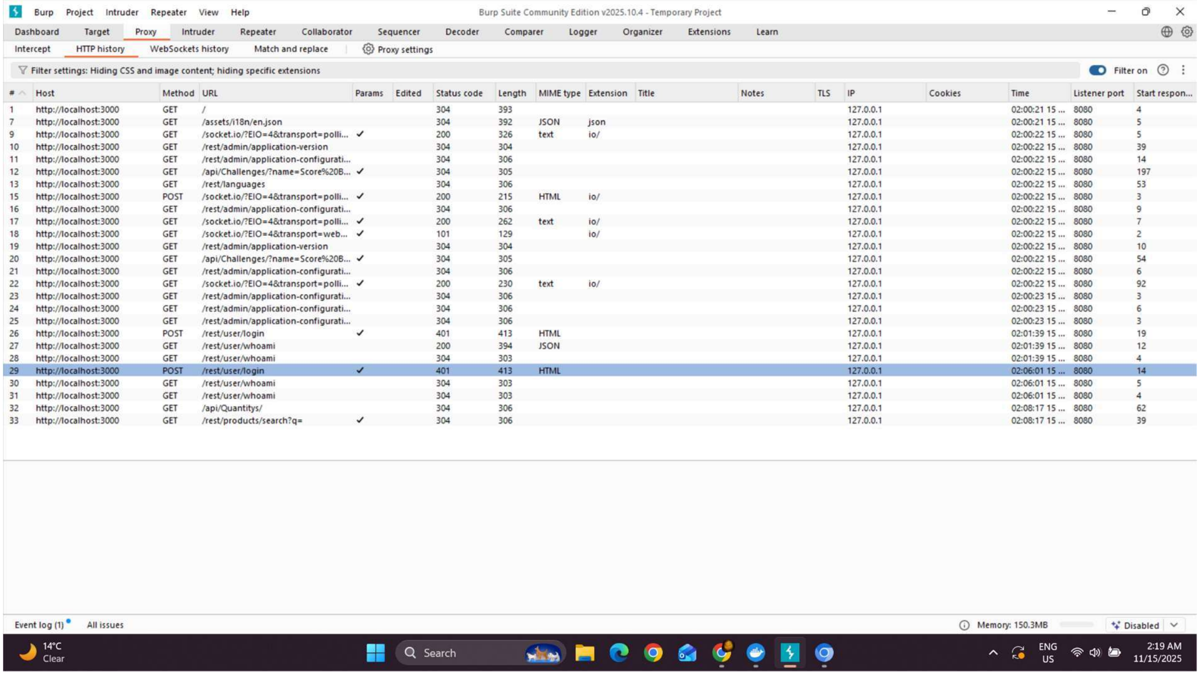
Search Results -

No results found

Try adjusting your search to find what you're looking for.

Items per page: 12

0 of 0


Burp Suite Community Edition v2025.10.4 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Intercept HTTP history WebSockets history Match and replace Proxy settings
Filter on
Event log (1) All issues
Memory: 150.3MB
Disabled
14°C Clear
Search
ENG US
2:19 AM 11/15/2025

Filter settings: Hiding CSS and image content; hiding specific extensions

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port	Start respon...
1	http://localhost:3000	GET	/			304	393						127.0.0.1		02:00:21 15 ...	8080	4
7	http://localhost:3000	GET	/assets/i18n/en.json			304	392	JSON	json				127.0.0.1		02:00:21 15 ...	8080	5
9	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	326	text	io/				127.0.0.1		02:00:22 15 ...	8080	5
10	http://localhost:3000	GET	/rest/admin/application-version			304	304						127.0.0.1		02:00:22 15 ...	8080	39
11	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:22 15 ...	8080	14
12	http://localhost:3000	GET	/api/Challenges/?name=Score%20B...	✓		304	305						127.0.0.1		02:00:22 15 ...	8080	197
13	http://localhost:3000	GET	/rest/languages			304	306						127.0.0.1		02:00:22 15 ...	8080	53
15	http://localhost:3000	POST	/socket.io/?EIO=4&transport=polli...	✓		200	215	HTML	io/				127.0.0.1		02:00:22 15 ...	8080	3
16	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:22 15 ...	8080	9
17	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	262	text	io/				127.0.0.1		02:00:22 15 ...	8080	7
18	http://localhost:3000	GET	/socket.io/?EIO=4&transport=web...	✓		101	129		io/				127.0.0.1		02:00:22 15 ...	8080	2
19	http://localhost:3000	GET	/rest/admin/application-version			304	304						127.0.0.1		02:00:22 15 ...	8080	10
20	http://localhost:3000	GET	/api/Challenges/?name=Score%20B...	✓		304	305						127.0.0.1		02:00:22 15 ...	8080	54
21	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:22 15 ...	8080	6
22	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	230	text	io/				127.0.0.1		02:00:22 15 ...	8080	92
23	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:23 15 ...	8080	3
24	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:23 15 ...	8080	6
25	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:00:23 15 ...	8080	3
26	http://localhost:3000	POST	/rest/user/login	✓		401	413	HTML					127.0.0.1		02:01:39 15 ...	8080	19
27	http://localhost:3000	GET	/rest/user/whoami			200	394	JSON					127.0.0.1		02:01:39 15 ...	8080	12
28	http://localhost:3000	GET	/rest/user/whoami			304	303						127.0.0.1		02:01:39 15 ...	8080	4
29	http://localhost:3000	POST	/rest/user/login	✓		401	413	HTML					127.0.0.1		02:06:01 15 ...	8080	14
30	http://localhost:3000	GET	/rest/user/whoami			304	303						127.0.0.1		02:06:01 15 ...	8080	5
31	http://localhost:3000	GET	/rest/user/whoami			304	303						127.0.0.1		02:06:01 15 ...	8080	4
32	http://localhost:3000	GET	/api/Quantitys/			304	306						127.0.0.1		02:08:17 15 ...	8080	62
33	http://localhost:3000	GET	/rest/products/search?q=	✓		304	306						127.0.0.1		02:08:17 15 ...	8080	39

Attack Scenario

A victim receives a link such as:

http://localhost:3000/#/search?q=<script>alert("pwned")</script>

If clicked, attacker code executes instantly.

Recommendations

- Use server-side sanitization libraries
- Encode all output using HTML entity encoding
- Deploy Content Security Policy (CSP)
- Validate inputs using whitelist filters

7.3 Finding F-03: Directory Browsing / Sensitive Data Exposure

Severity: High
CVSS v3.1 Score: 7.1

OWASP Category: A01 – Broken Access Control
Secondary Category: A05 – Security Misconfiguration

Description

The /ftp directory is publicly accessible and lists internal files:

http://localhost:3000/ftp

This is a common misconfiguration where a server exposes its internal filesystem.

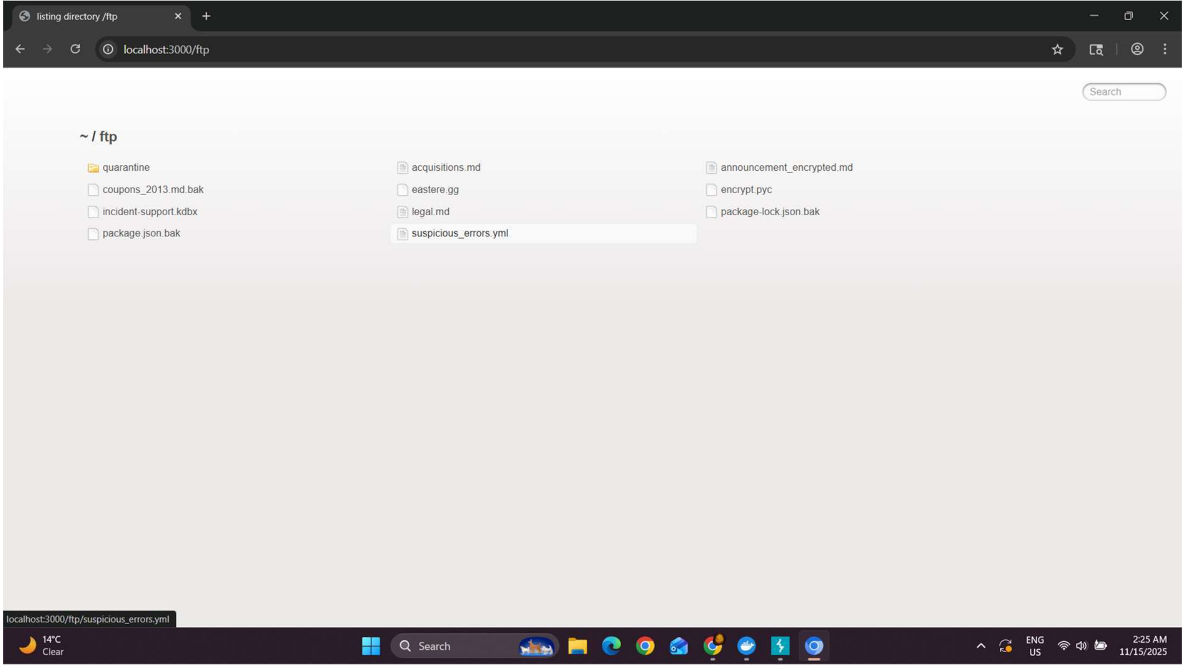
Technical Evidence

Accessible files included:

- Invoices
- Images
- Log files
- Data exports (depending on version)

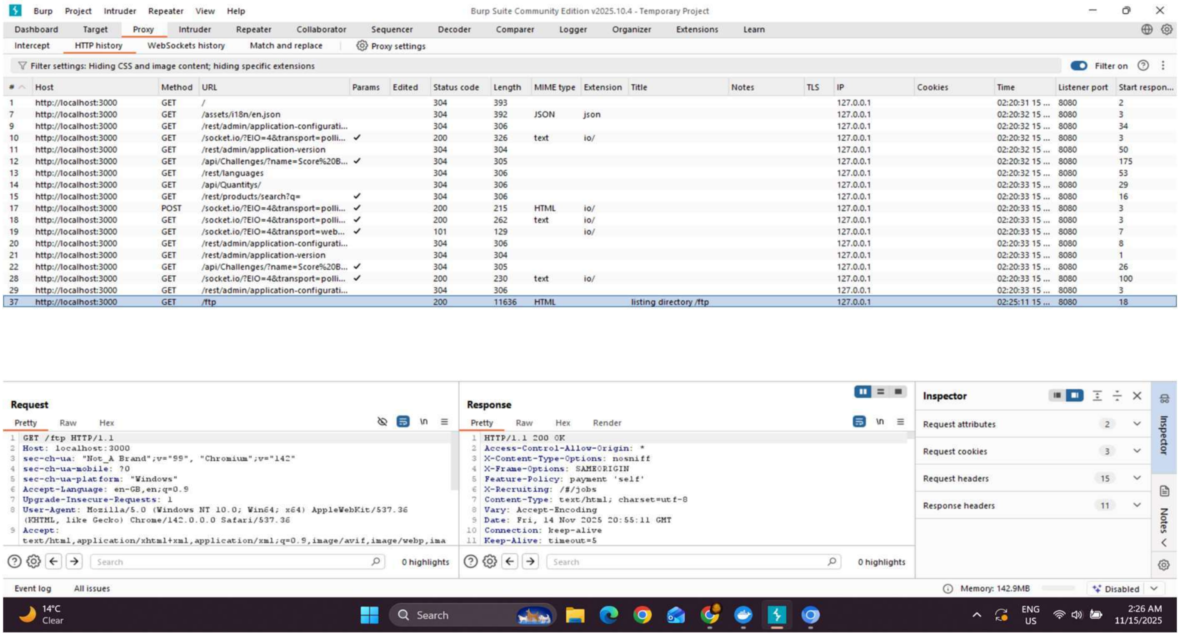
Impact

- Leakage of confidential information
- Data theft
- Increased attack surface
- Attackers may identify internal file paths and configurations



~ / ftp

quarantine
coupons_2013.md.bak
incident-support.kdbx
package.json.bak
acquisitions.md
eastere.gg
legal.md
suspicious_errors.yml
announcement_encrypted.md
encrypt.pyc
package-lock.json.bak


Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2025.10.4 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Intercept HTTP history WebSockets history Match and replace Proxy settings
Filter settings: Hiding CSS and image content; hiding specific extensions
Filter on
Inspector
Request attributes 2
Request cookies 3
Request headers 15
Response headers 11
Notes
Event log All issues
Memory: 142.9MB
Disabled
14°C Clear
Search
ENG US
2:26 AM 11/15/2025

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port	Start respon...
1	http://localhost:3000	GET	/			304	393						127.0.0.1		02:20:31 15 ...	8080	2
7	http://localhost:3000	GET	/assets/i18n/en.json			304	392	JSON	json				127.0.0.1		02:20:32 15 ...	8080	3
9	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:20:32 15 ...	8080	34
10	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	326	text	io/				127.0.0.1		02:20:32 15 ...	8080	3
11	http://localhost:3000	GET	/rest/admin/application-version			304	304						127.0.0.1		02:20:32 15 ...	8080	50
12	http://localhost:3000	GET	/api/Challenges/?name=Score%20B...	✓		304	305						127.0.0.1		02:20:32 15 ...	8080	175
13	http://localhost:3000	GET	/rest/languages			304	306						127.0.0.1		02:20:32 15 ...	8080	53
14	http://localhost:3000	GET	/api/Quantitys/			304	306						127.0.0.1		02:20:33 15 ...	8080	29
15	http://localhost:3000	GET	/rest/products/search?q=	✓		304	306						127.0.0.1		02:20:33 15 ...	8080	16
17	http://localhost:3000	POST	/socket.io/?EIO=4&transport=polli...	✓		200	215	HTML	io/				127.0.0.1		02:20:33 15 ...	8080	3
18	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	262	text	io/				127.0.0.1		02:20:33 15 ...	8080	3
19	http://localhost:3000	GET	/socket.io/?EIO=4&transport=web...	✓		101	129		io/				127.0.0.1		02:20:33 15 ...	8080	7
20	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:20:33 15 ...	8080	8
21	http://localhost:3000	GET	/rest/admin/application-version			304	304						127.0.0.1		02:20:33 15 ...	8080	1
22	http://localhost:3000	GET	/api/Challenges/?name=Score%20B...	✓		304	305						127.0.0.1		02:20:33 15 ...	8080	26
28	http://localhost:3000	GET	/socket.io/?EIO=4&transport=polli...	✓		200	230	text	io/				127.0.0.1		02:20:33 15 ...	8080	100
29	http://localhost:3000	GET	/rest/admin/application-configurati...			304	306						127.0.0.1		02:20:33 15 ...	8080	3
37	http://localhost:3000	GET	/ftp			200	11636	HTML		listing directory /ftp			127.0.0.1		02:25:11 15 ...	8080	18

Request

Pretty Raw Hex

```
GET /ftp HTTP/1.1
Host: localhost:3000
sec-ch-ua: "Not_A Brand";v="99", "Chromium";v="142"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Accept-Language: en-GB,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,ima
```

Search 0 highlights

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: text/html; charset=utf-8
Vary: Accept-Encoding
Date: Fri, 14 Nov 2025 20:55:11 GMT
Connection: keep-alive
Keep-Alive: timeout=5
```

Search 0 highlights

Attack Scenario

An attacker navigates to /ftp and downloads invoice files containing customer information.

Recommendations

- Disable autoindex / directory listing on the server
- Restrict /ftp to admin-only access
- Move sensitive files outside public webroot
- Use proper file permissions (600/640 on Linux systems)

8. OWASP Top 10 Mapping (2021)

Vulnerability	OWASP Category	Explanation
Username Enumeration	A07: Identification & Authentication Failures	Leaks authentication state information
Reflected XSS	A03: Injection	Unsanitized input directly reflected into HTML
Directory Browsing	A01: Broken Access Control / A05: Security Misconfiguration	Unauthorized access to internal directories

9. Risk Rating Matrix

Severity	Description
Low	Minimal impact, no sensitive data exposed
Medium	Potential information leakage, minor exploitation
High	Significant risk, sensitive data exposed, real exploitation possible
Critical	Full system compromise, remote code execution

All identified vulnerabilities fall under Medium to High-risk categories.

10. Recommendations & Best Practices

✓ Strengthen Authentication

- Implement login throttling
- Use generic error messages
- Add multi-factor authentication (MFA)

✓ Improve Input Security

- Enforce input validation
- Encode all outputs
- Implement CSP headers
- Avoid dynamic HTML generation

✓ Restrict File Access

- Disable directory indexing
- Implement RBAC for sensitive paths
- Enforce secure server configurations

✓ General Best Practices

- Conduct regular security audits
- Perform code reviews
- Keep frameworks updated
- Use automated scanners during CI/CD

Conclusion

The assessment successfully identified three realistic and high-impact vulnerabilities in the OWASP Juice Shop application. These findings reflect common security issues that occur in production environments due to misconfiguration, poor validation, or weak access control.

By following the recommended fixes, the application's overall security posture can be dramatically improved.
This project provided hands-on experience with industry-standard testing methodologies, tools, and vulnerability reporting practices.