

## Запускаю docker контейнер, в котором работает приложение OWASP Juice Shop

```
Командная строка
Microsoft Windows [Version 10.0.19045.4046]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\std_string>docker pull bkimminich/juice-shop
Pulling from bkimminich/juice-shop
latest: Pulling from bkimminich/juice-shop
3d78e577da35: Pull complete
bf5b9b82a0b6: Pull complete
4eff9a62d888: Pull complete
a62778643d56: Pull complete
a62778643d56: Pull complete
7c12895b77fb: Pull complete
3214acfc345c0: Pull complete
5664b15f108b: Pull complete
0bab15eeea81d: Pull complete
4aa0ea1413d3: Pull complete
d793a0a11111: Pull complete
04ee425378d2: Pull complete
d40c3200d929: Pull complete
221438ca359c: Pull complete
pb0f6cad3051: Pull complete
6f971e93c4e2: Pull complete
c83c31ce41af: Pull complete
0cb5c07f8ed: Pull complete
3137de757dd8: Pull complete
b78a88c4c4b5: Pull complete
bc0e9837a45c: Pull complete
93600000c7: Pull complete
Digest: sha256:db5bacfffb67d4bab08df1d7cf79aa57402eddfe526da37cdf0620a4131e82ca
Status: Downloaded newer image for bkimminich/juice-shop:latest
docker.io/bkimminich/juice-shop:latest

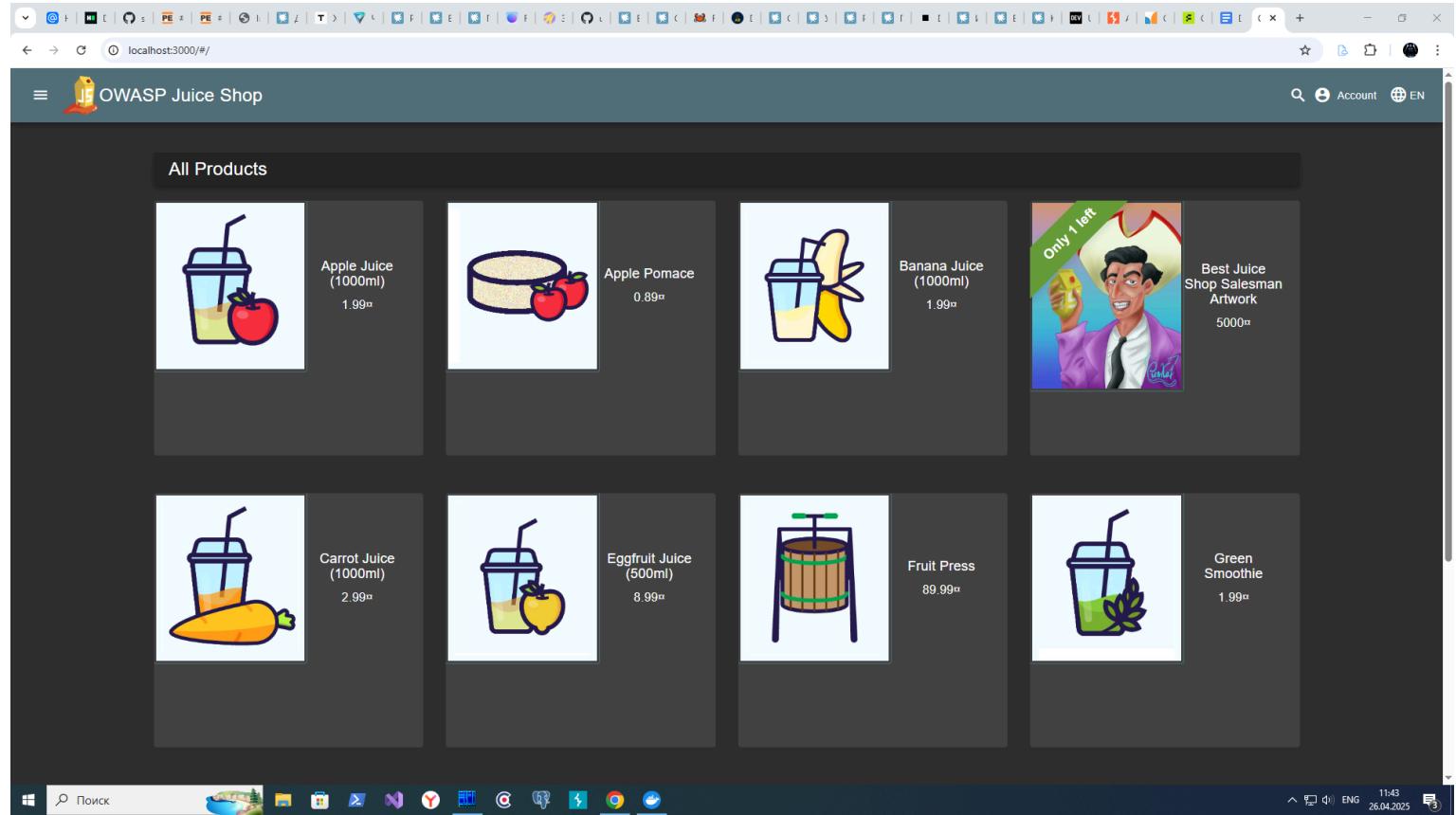
What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview bkimminich/juice-shop

C:\Users\std_string>docker run --rm -d --name juice -p 3000:3000 bkimminich/juice-shop
06cc0459623c707b73503a8d8bb1f8d3ff15cd041fc268e3513a68533a9dcfd

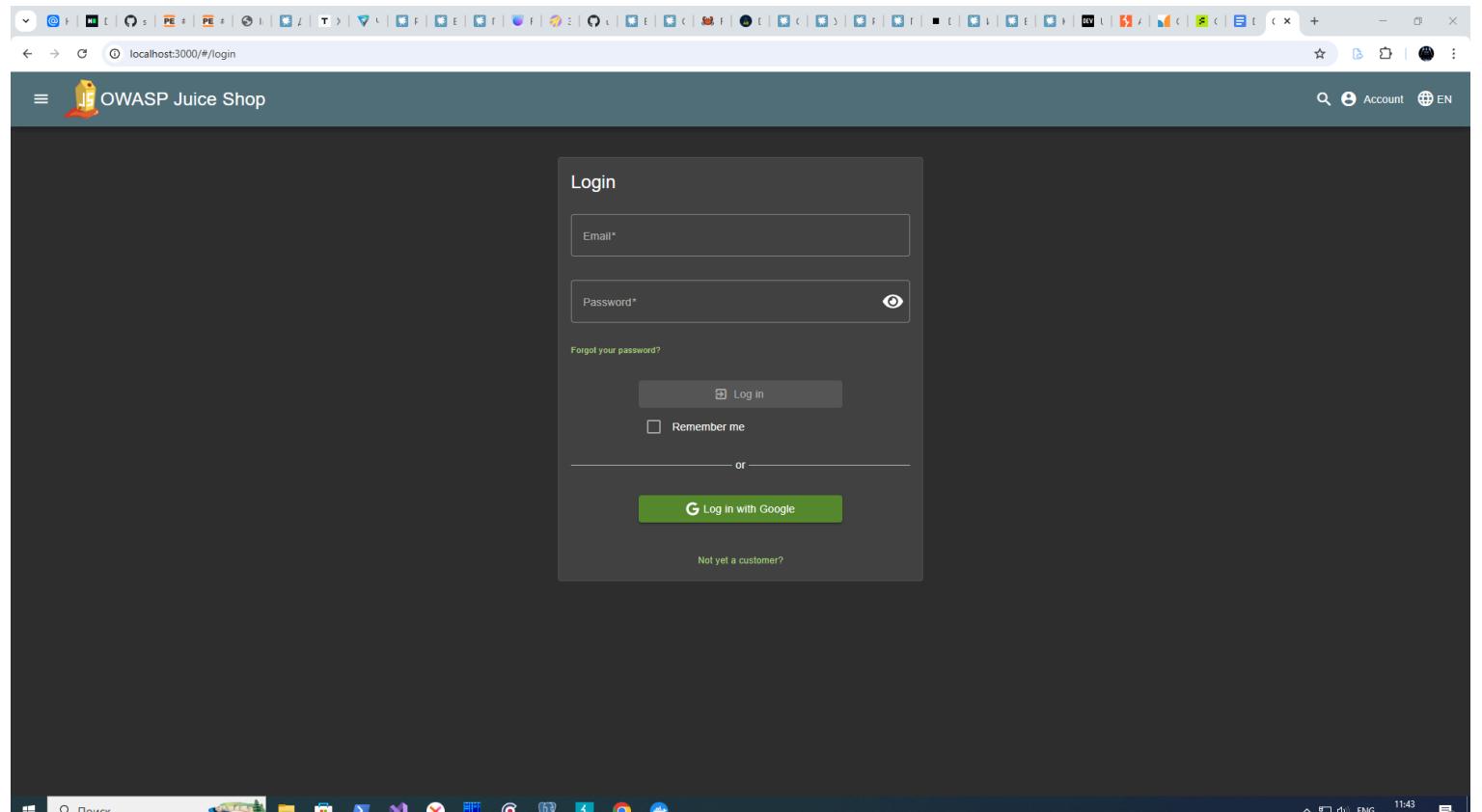
C:\Users\std_string>
```

## 1. SQL Injection-логин

Запускаю приложение



Захожу на форму для ввода данных учетной записи (через меню Account - Login)



Пробую форму ввода данных учетной записи на наличие SQL инъекции: в поле электронной почты пользователя я ввожу

' OR 1=1;--

Т.к. я не знаю электронную почту ни у одного из пользователей; в поле пароль ввожу произвольные данные.

The screenshot shows a browser window with the URL `localhost:3000/#/login`. The page title is "OWASP Juice Shop". The main content is a "Login" form. In the "Email\*" field, the value "' OR 1=1;--" is entered. Below the form, there is a message: "Forgot your password?". At the bottom of the form, there is a "Log in" button with a key icon and a "Remember me" checkbox. Below the form, there is a horizontal line with the word "or" and a green "G Log in with Google" button. At the very bottom of the page, there is a standard Windows taskbar with various icons and system status information.

Отправляю введенные данные на сервер (нажатием на кнопку Log in). Вижу, что данная SQL инъекция сработала: я попал в учетную запись администратора

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

### All Products

 Apple Juice (1000ml) 1.99¤  <a href="#">Add to Basket</a>	 Apple Pomace 0.89¤  <a href="#">Add to Basket</a>	 Banana Juice (1000ml) 1.99¤  <a href="#">Add to Basket</a>	 Only 1 left Best Juice Shop Salesman Artwork 5000¤  <a href="#">Add to Basket</a>
 Carrot Juice (1000ml) 2.99¤  <a href="#">Add to Basket</a>	 Eggfruit Juice (500ml) 8.99¤  <a href="#">Add to Basket</a>	 Fruit Press 89.99¤  <a href="#">Add to Basket</a>	 Green Smoothie 1.99¤  <a href="#">Add to Basket</a>

Смотрю профиль текущего пользователя; убеждаюсь, что это на самом деле учетная запись администратора

### User Profile

Email: admin@juice-sh.op

Username: e.g. SupportUser  
[Set Username](#)

File Upload:  
Выберите файл | Файл не выбран  
[Upload Picture](#)

OR

Image URL:  
<https://www.gravatar.com/avatar/5C77D9a2b770e079d220d4074408>  
[Link Image](#)

Как была найдена уязвимость: стандартная проверка наличия SQL инъекции в форме ввода данных учетной записи

Как можно защититься от данной уязвимости: всегда использовать средства для работы с БД, предназначенные для использования пользовательского ввода в SQL запросах. Это в первую очередь, параметризованные запросы. Конечно, можно попробовать защититься от этой уязвимости и с помощью WAF, но это лечение последствий, а не причины.

Какие риски несет данная уязвимость: несанкционированный вход под учетной записью администратора ставит под удар и конфиденциальность, и целостность и доступность данных.

## 2. Stored XSS

Запускаю приложение

The screenshot shows a web application interface for 'OWASP Juice Shop'. At the top, there's a navigation bar with various icons and a search bar. Below it is a header with the title 'All Products' and a logo of a juice glass. The main content area displays a grid of product cards:

Image	Name	Description	Price
	Apple Juice (1000ml)		1.99¤
	Apple Pomace		0.89¤
	Banana Juice (1000ml)		1.99¤
	Best Juice Shop Salesman Artwork	Only 1 left!	5000¤
	Carrot Juice (1000ml)		2.99¤
	Eggfruit Juice (500ml)		8.99¤
	Fruit Press		89.99¤
	Green Smoothie		1.99¤

The bottom of the screen shows a Windows taskbar with various pinned icons and system status information.

В строку поиска ввожу следующее значение:

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

Вижу, что появилось окно оповещения с текстом IDQD

A screenshot of a web browser window. The address bar shows the URL `localhost:30000/#/search?q=<iframe%20src%3D'javascript:alert('IDQD')">`. A modal dialog box is centered on the screen with the text "Подтвердите действие на localhost:30000" and "IDQD". Below the dialog, the main page content is visible, showing a search results page for "Search Results -". The search bar contains the query "No results found Try adjusting your search to find what you're looking for.". At the bottom right of the page, there are pagination controls: "Items per page: 12" and "0 of 0".

A screenshot of a web browser window. The address bar shows the URL `localhost:30000/#/search`. The main content area displays a grid of products under the heading "All Products". The products listed are:

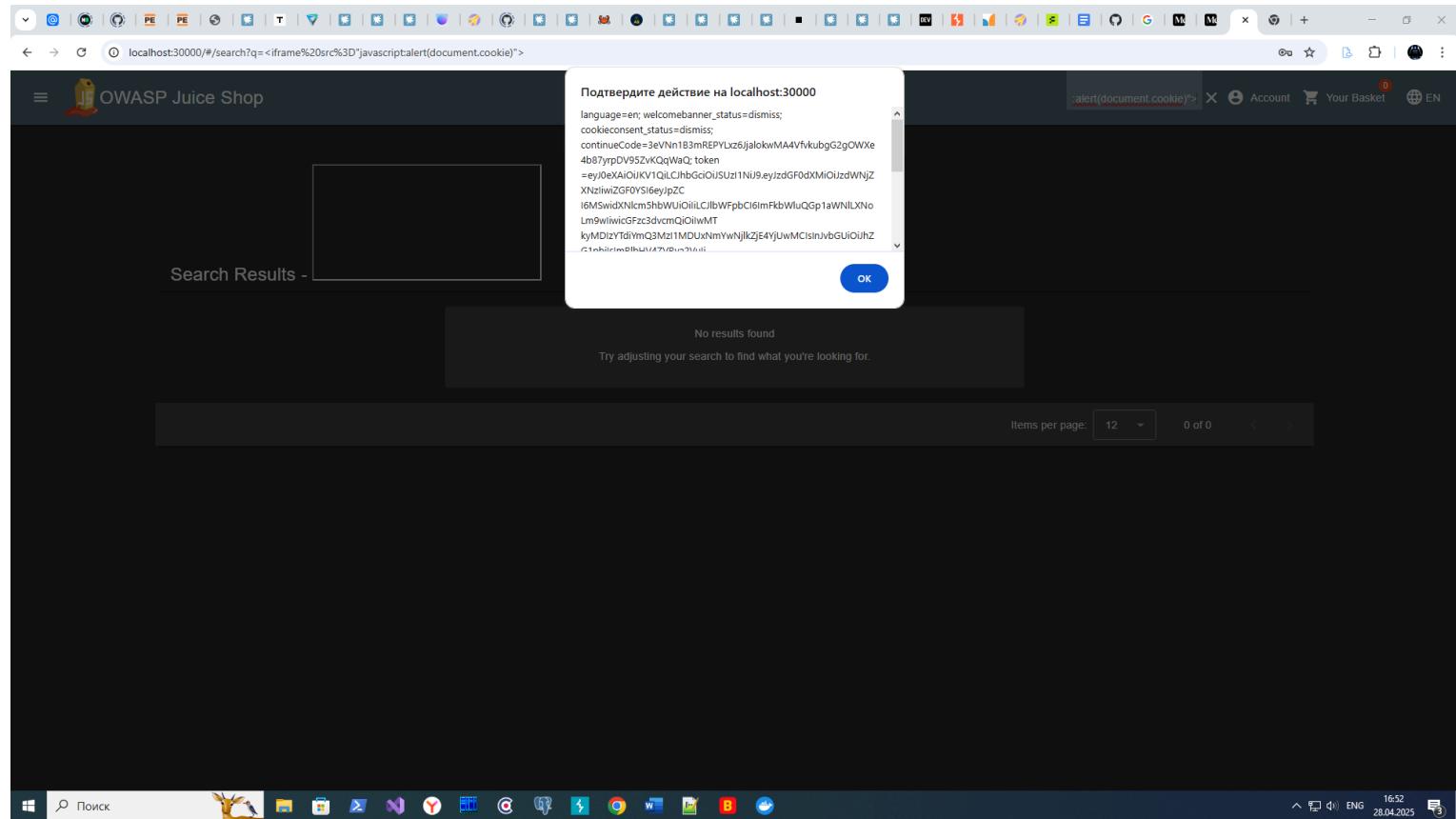
Image	Name	Description	Price
	Apple Juice (1000ml)	1.99¤	<button>Add to Basket</button>
	Apple Pomace	0.89¤	<button>Add to Basket</button>
	Banana Juice (1000ml)	1.99¤	<button>Add to Basket</button>
	Best Juice Shop Salesman Artwork	5000¤	<button>Add to Basket</button>
	Carrot Juice (1000ml)	2.99¤	<button>Add to Basket</button>
	Eggfruit Juice (500ml)	8.99¤	<button>Add to Basket</button>
	Fruit Press	89.99¤	<button>Add to Basket</button>
	Green Smoothie	1.99¤	<button>Add to Basket</button>

The status bar at the bottom of the browser window shows the date and time as "28.04.2025 16:51".

В строку поиска ввожу следующее значение:

```
<iframe src="javascript:alert(document.cookie)">
```

Вижу, что появилось окно оповещения со значением объекта cookie (которое содержится в свойстве document.cookie)



Наша XSS инъекция в строку поиска сработала

К сожалению другие XSS инъекции недоступны при запуске OWASP Juice Shop из docker контейнера - это видно из информации с доски результатов, расположенной по относительному пути score-board (я лично проверил несколько stored XSS и ни одна из них не сработала)

The screenshot shows a web-based challenge board for XSS attacks. At the top, there are tabs for 'Hacking Challenges' and 'Coding Challenges'. Below the tabs, there's a search bar and filters for 'Difficulty', 'Status', and 'Tags'. A message indicates 17 challenges are unavailable due to security concerns or technical incompatibility. The challenges listed are:

- DOM XSS**: Perform a DOM XSS attack with <iframe src="javascript:alert('xss')">. Difficulty: ★.
- Bonus Payload**: Use the bonus payload <iframe width="100%" height="166" scrolling="no" frameborder="no" allow="autoplay" src="https://w.soundcloud.com/player/>. Difficulty: ★★.
- Reflected XSS**: Perform a reflected XSS attack with <iframe src="javascript:alert('xss')>. (This challenge is potentially harmful on Docker!) Difficulty: ★★.
- API-only XSS**: Perform a persisted XSS attack with <iframe src="javascript:alert('xss')> without using the frontend application at all. (This challenge is potentially harmful on Docker!) Difficulty: ★★★.

Other challenges shown include:

- Client-side XSS Protection**: Perform a persisted XSS attack with <iframe src="javascript:alert('xss')> bypassing a client-side security mechanism. (This challenge is potentially harmful on Docker!) Difficulty: ★★★★.
- CSP Bypass**: Bypass the Content Security Policy and perform an XSS attack with <script>alert('xss')</script> on a legacy page within the application. (This challenge is potentially harmful on Docker!) Difficulty: ★★★★.
- HTTP-Header XSS**: Perform a persisted XSS attack with <iframe src="javascript:alert('xss')> through an HTTP header. (This challenge is potentially harmful on Docker!) Difficulty: ★★★★.
- Server-side XSS Protection**: Perform a persisted XSS attack with <iframe src="javascript:alert('xss')> bypassing a server-side security mechanism. (This challenge is potentially harmful on Docker!) Difficulty: ★★★★.
- Video XSS**: Embed an XSS payload </script><script>alert('xss')</script> into our promo video. (This challenge is potentially harmful on Docker!) Difficulty: ★★★★★.

Как была найдена уязвимость: с помощью ввода простейшей проверки на наличие XSS инъекции во все элементы управления, в которые можно ввести текст

Как можно защититься от данной уязвимости: экранирование пользовательского ввода перед его отображением на странице, использование библиотек для безопасного отображения HTML (например, DOMPurify).

Какие риски несет данная уязвимость: XSS инъекция (особенно stored XSS инъекция) позволяет злоумышленнику выполнить произвольный JavaScript-код в браузере жертвы, что может привести к краже сессий, конфиденциальной информации и другим атакам.

### 3. Price Manipulation

Запускаю burp suite, захожу на вкладку proxy (intercept off)

S Burp Project Intruder Repeater View Help

Burp Suite Community Edition v2025.3.3 - Temporary Project

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

Intercept HTTP history WebSockets history Match and replace Proxy settings

Intercept off Forward Drop

Time Type Direction Method URL Status code Length

**Intercept is off**

If you turn Intercept on, messages between Burp's browser and your target servers are held here. This enables you to analyze and modify these messages, before you forward them.

Learn more Open browser

Event log All issues

Memory: 113.3MB Disabled

Пойск

14:49 28.04.2025

Запускаю из burp suite браузер. Запускаю приложение

OWASP Juice Shop

localhost:30000/#/

OWASP Juice Shop

All Products

Image	Name	Description	Price
	Apple Juice (1000ml)	1.99¤	
	Apple Pomace	0.89¤	
	Banana Juice (1000ml)	1.99¤	
	Best Juice Shop Salesman Artwork	5000¤	Only 1 left
	Carrot Juice (1000ml)	2.99¤	
	Eggfruit Juice (500ml)	8.99¤	
	Fruit Press	89.99¤	
	Green Smoothie	1.99¤	

Захожу в систему как администратор (с помощью SQL инъекции)

OWASP Juice Shop

localhost:30000/#/search

All Products

	Apple Juice (1000ml) 1.99¤  <a href="#">Add to Basket</a>		Apple Pomace 0.89¤  <a href="#">Add to Basket</a>		Banana Juice (1000ml) 1.99¤  <a href="#">Add to Basket</a>		Only 1 left Best Juice Shop Salesman Artwork 5000¤  <a href="#">Add to Basket</a>
	Carrot Juice (1000ml) 2.99¤  <a href="#">Add to Basket</a>		Eggfruit Juice (500ml) 8.99¤  <a href="#">Add to Basket</a>		Fruit Press 89.99¤  <a href="#">Add to Basket</a>		Green Smoothie 1.99¤  <a href="#">Add to Basket</a>

Очищаю корзину и добавляю в нее Apple Juice (1000ml). Перехожу в burp suite на вкладке Proxy во вкладку Http history. В ней выбираю запрос по относительному URL /api/Products/1 и отправляю его в Repeater

Burp Suite Community Edition v2025.3.3 - Temporary Project

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

HTTP history WebSockets history Match and replace Proxy settings

Filter settings: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port	Start response t
67	http://localhost:30000	GET	/rest/basket/1			304	305				1 JVTs, 0 JVEs	127.0.0.1			14:53:40 28 Apr 2025	8080	21
68	http://localhost:30000	DELETE	/api/BasketItems/2			200	414	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:53:40 28 Apr 2025	8080	28
69	http://localhost:30000	GET	/rest/basket/1			200	926	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:53:41 28 Apr 2025	8080	23
70	http://localhost:30000	GET	/rest/basket/1			304	305				1 JVTs, 0 JVEs	127.0.0.1			14:53:41 28 Apr 2025	8080	19
71	http://localhost:30000	DELETE	/api/BasketItems/3			200	414	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:53:41 28 Apr 2025	8080	26
72	http://localhost:30000	GET	/rest/basket/1			200	538	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:53:41 28 Apr 2025	8080	23
73	http://localhost:30000	GET	/rest/basket/1			304	304				1 JVTs, 0 JVEs	127.0.0.1			14:53:42 28 Apr 2025	8080	21
74	http://localhost:30000	GET	/api/Quantities			304	306				1 JVTs, 0 JVEs	127.0.0.1			14:53:47 28 Apr 2025	8080	21
75	http://localhost:30000	GET	/rest/products/search?q=		✓	304	306				1 JVTs, 0 JVEs	127.0.0.1			14:53:48 28 Apr 2025	8080	11
86	http://localhost:30000	GET	/rest/user/whoami			304	304				1 JVTs, 0 JVEs	127.0.0.1			14:53:51 28 Apr 2025	8080	5
87	http://localhost:30000	GET	/rest/products/1/reviews			200	544	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:53:51 28 Apr 2025	8080	20
88	http://localhost:30000	GET	/rest/products/1/reviews			304	304				1 JVTs, 0 JVEs	127.0.0.1			14:53:51 28 Apr 2025	8080	11
89	http://localhost:30000	GET	/rest/products/1/reviews			304	304				1 JVTs, 0 JVEs	127.0.0.1			14:53:51 28 Apr 2025	8080	8
90	http://localhost:30000	GET	/rest/basket/1			304	304				1 JVTs, 0 JVEs	127.0.0.1			14:54:14 28 Apr 2025	8080	23
91	http://localhost:30000	POST	/api/BasketItems/		✓	200	541	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:54:14 28 Apr 2025	8080	10
92	http://localhost:30000	GET	/api/Products/?id=Mon%20Apr%2028%		✓	200	643	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:54:14 28 Apr 2025	8080	22
93	http://localhost:30000	GET	/rest/basket/1			200	908	JSON			1 JVTs, 0 JVEs	127.0.0.1			14:54:14 28 Apr 2025	8080	22
94	http://localhost:30000	GET	/rest/basket/1			304	305				1 JVTs, 0 JVEs	127.0.0.1			14:54:29 28 Apr 2025	8080	26

Request

Pretty Raw Hex JSON Web Token

Response

Pretty Raw Hex Render

Inspector

Request attributes 2

Request query parameters 1

Request cookies 5

Request headers 16

Response headers 12

В Repeater убираю из GET запроса параметр d, отправляю его на сервер и смотрю результат

Burp Suite Community Edition v2025.3.3 - Temporary Project

Target: http://localhost:30000 / HTTP/1.1

**Request**

```
GET /api/Products/1 HTTP/1.1
Host: localhost:3000
sec-ch-ua-platform: "Windows"
Authorization: Bearer eyJhbGciOiJIUzI1NiJ9.e3VzdG9tQDxDMiOjIzJwNjZKXNlIwiZ0POFyS1eeyjpZC16MwvdxN1cm
ShbWU1o1ilCjUbWpFc16ImhW1QnGpiaNLWLn0LmW1iwiGrc3dvcmQ1o1iWTrxM0IiYtd1YmQ3Mz1lMDUoN
mV7zE4Y3YjWmC1i1nvhGbU1i1nbH4V2RvaVujo1i1ibGpfa2z1u2s1i1LcJwvcm
D1o1i1iwiwNB73pdm10Bdc16i1j1wMjUtdcH1jgqNdkMc6MTyUttY21CsWdowMCis1m1b1GV02W8Bdc16b1vNbHs1mhd1C6M
TcONTgzsMzg1CX0...nTyoyLoc3pdmn1z23KCRt5k532FT1o4uq7lozthpPscCvjeftSX2hzGtcz1zSpLxJZDh-L66
Bx4c
Accept-Language: ru-RU,ru;q=0.9
sec-ch-ua: "Google Chrome";v="135", "Not-A-Brand";v="0"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/135.0.0.0 Safari/537.36
Accept: application/json, text/plain, */*
X-User-Email: admin@juice-shop. OR i=1--
See-Header: same-origin
See-For-Behind-Header: none
See-Fetch-Dest: empty
Referer: http://localhost:3000/
Accept-Encoding: gzip, deflate, br
Cookie: language=en; welcomebanner_status=dismiss; continueCode=1RDVsA7Q65x3TjW1kNW4RPXpXzd58sAx0ElgleqfVmbNsro2w2alnjR9; cookieconsent_status=dismiss;
token=eyJhbGciOiJIUzI1NiJ9.e3VzdG9tQDxDMiOjIzJwNjZKXNlIwiZ0POFyS1eeyjpZC16MwvdxN1cm
ShbWU1o1ilCjUbWpFc16ImhW1QnGpiaNLWLn0LmW1iwiGrc3dvcmQ1o1iWTrxM0IiYtd1YmQ3Mz1lMDUoN
mV7zE4Y3YjWmC1i1nvhGbU1i1nbH4V2RvaVujo1i1ibGpfa2z1u2s1i1LcJwvcm
D1o1i1iwiwNB73pdm10Bdc16i1j1wMjUtdcH1jgqNdkMc6MTyUttY21CsWdowMCis1m1b1GV02W8Bdc16b1vNbHs1mhd1C6M
TcONTgzsMzg1CX0...nTyoyLoc3pdmn1z23KCRt5k532FT1o4uq7lozthpPscCvjeftSX2hzGtcz1zSpLxJZDh-L66
Bx4c
Connection: keep-alive

```

**Response**

```
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
ETag: "101-d87f0c08f8f1"
Vary: Accept-Encoding
Date: Mon, 28 Apr 2025 09:55:46 GMT
Connection: keep-alive
Keep-Alive: timeout=5
status:"success",
"data":{
  "id":1,
  "name":"Apple Juice (1000ml)",
  "description":"The all-time classic.",
  "price":1.99,
  "taxRate":0.05,
  "image":"apple_juice.jpg",
  "createdAt": "2025-04-28T09:37:17.232Z",
  "updatedAt": "2025-04-28T09:37:17.232Z",
  "deletedAt":null
}

```

**Inspector**

- Request attributes: 2
- Request query parameters: 0
- Request body parameters: 0
- Request cookies: 5
- Request headers: 16
- Response headers: 12

**Notes**

**Custom actions**

Формирую PUT запрос со следующими параметрами:

- Свойство Content-Type: application/json
  - Тело запроса: {"price":0, "deluxePrice":0}

Отправляю запрос на сервер и получаю ответ из которого видно, что обе цены стали равны 0.

Burp Suite Community Edition v2025.3.3 - Temporary Project

Target: http://localhost:30000

**Request**

```
Pretty Raw Hex JSON Web Token
PUT /api/Products/1 HTTP/1.1
Host: localhost:30000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/135.0.0.0 Safari/135.36
Accept: application/json, text/plain, */*
X-User-Email: admin@juice-sh.op
Content-Type: application/json
Content-Length: 28
Cookie: language=en; welcomebanner_status=dismiss; continueCode=1E05VA7Q5yX3Up1kW4RFPXzjd58xAvOElgLeqWmBmBnroFwaijn9P; cookiecomment_status=dismiss; cookiecomment_time=1688303000
Authorization: Beater
Content-Type: application/json
Content-Length: 28
{
    "price": 0,
    "deluxePrice": 0
}
```

**Response**

```
Pretty Raw Hex Render
HTTP/1.1 200 OK
Date: Mon, 28 Apr 2025 09:57:34 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 28
Vary: Accept-Encoding
Date: Mon, 28 Apr 2025 09:57:34 GMT
Connection: keep-alive
Keep-Alive: timeout=5
{
    "status": "success",
    "data": {
        "id": 1,
        "name": "Apple Juice (1000ml)",
        "description": "The all-time classic.",
        "price": 0,
        "deluxePrice": 0,
        "image": "apple_juice.jpg",
        "createdAt": "2025-04-28T09:57:37.232Z",
        "updatedAt": "2025-04-28T09:57:34.689Z",
        "deletedAt": null
    }
}
```

**Inspector**

Selected text: ntent-Length: 28 \x \n \x \n ("price"

Decoded from: Select ✓ Cancel Apply changes

Request attributes: 2 ✓

Request query parameters: 0 ✓

Request cookies: 5 ✓

Request headers: 18 ✓

Response headers: 12 ✓



Проверяю в приложении в браузере, что цена на Apple Juice (1000ml) стала 0

OWASP Juice Shop

All Products

 Apple Juice (1000ml) 0¤ <a href="#">Add to Basket</a>	 Apple Pomace 0.89¤ <a href="#">Add to Basket</a>	 Banana Juice (1000ml) 1.99¤ <a href="#">Add to Basket</a>	 Only 1 left Best Juice Shop Salesman Artwork 5000¤ <a href="#">Add to Basket</a>
 Carrot Juice (1000ml) 2.99¤ <a href="#">Add to Basket</a>	 Eggfruit Juice (500ml) 8.99¤ <a href="#">Add to Basket</a>	 Fruit Press 89.99¤ <a href="#">Add to Basket</a>	 Green Smoothie 1.99¤ <a href="#">Add to Basket</a>

Добавляю Apple Juice (1000ml) в корзину

OWASP Juice Shop

localhost:30000/#/basket

OWASP Juice Shop

Your Basket (admin@juice-sh.op)

Apple Juice (1000ml)

1 0¤

Total Price: 0¤

Checkout

You will gain 0 Bonus Points from this order!

Placed Apple Juice (1000ml) into basket.

Покупаю Apple Juice (1000ml)

OWASP Juice Shop

localhost:30000/#/order-summary

OWASP Juice Shop

Delivery Address

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

Payment Method

Digital Wallet

Order Summary

Items	0.00¤
Delivery	0.00¤
Promotion	0.00¤
<b>Total Price</b>	<b>0.00¤</b>

Place your order and pay

You will gain 0 Bonus Points from this order!

Покупаю Apple Juice (1000ml)

Смотрю заказы - вижу, что моя покупка после изменения цены прошла успешно

The screenshot shows a browser window with the URL <localhost:30000/#/order-completion/5267-fc853d47d22da4f4>. The page title is "OWASP Juice Shop". The main content area displays a "Thank you for your purchase!" message and an "Order Summary" table. The summary table shows one item: "Apple Juice (1000ml)" at 0.00€, quantity 1. The total price is also 0.00€. Below the table, a note states: "You have gained 0 Bonus Points from this order!". To the right of the table, there is delivery information: "Your order will be delivered in 5 days.", "Delivery Address: Administrator, 0815 Test Street, Test, Test, 4711 Test, Phone Number 1234567890". There are also social sharing icons for Twitter and Facebook.

The screenshot shows a browser window with the URL <localhost:30000/#/order-history>. The page title is "OWASP Juice Shop". The main content area displays an "Order History" table. It lists three orders. Order #5267-fc853d47d22da4f4 has a total price of 0.00€, bonus 0, and status "In Transit". Order #5267-dcf71a157574bb85 has a total price of 26.97€, bonus 3, and status "Delivered". Order #5267-3b16edfa0931880f has a total price of 8.96€, bonus 0, and status "In Transit". Each order row contains a table with product details: Apple Juice (1000ml) for the first two orders, and Eggfruit Juice (500ml) and Orange Juice (1000ml) for the third order. There are edit and delete icons for each order row.

Как была найдена уязвимость: с помощью анализа взаимодействия с бэкэндом через burp suite  
Как можно защититься от данной уязвимости: добавить валидацию данных на сервере  
Какие риски несет данная уязвимость: финансовые потери и, потенциально, репутационные потери

#### 4. Administration Access

Запускаю приложение

The screenshot shows the 'All Products' page of the OWASP Juice Shop application. The page displays a grid of products:

Image	Name	Description	Price
	Apple Juice (1000ml)		1.99¤
	Apple Pomace		0.89¤
	Banana Juice (1000ml)		1.99¤
	Best Juice Shop Salesman Artwork		5000¤
	Carrot Juice (1000ml)		2.99¤
	Eggfruit Juice (500ml)		8.99¤
	Fruit Press		89.99¤
	Green Smoothie		1.99¤

Произвожу поиск (с помощью Инструментов разработчика в google chrome) по загруженным файлам. В файле main.js нахожу потенциально интересующий нас относительный путь /administration

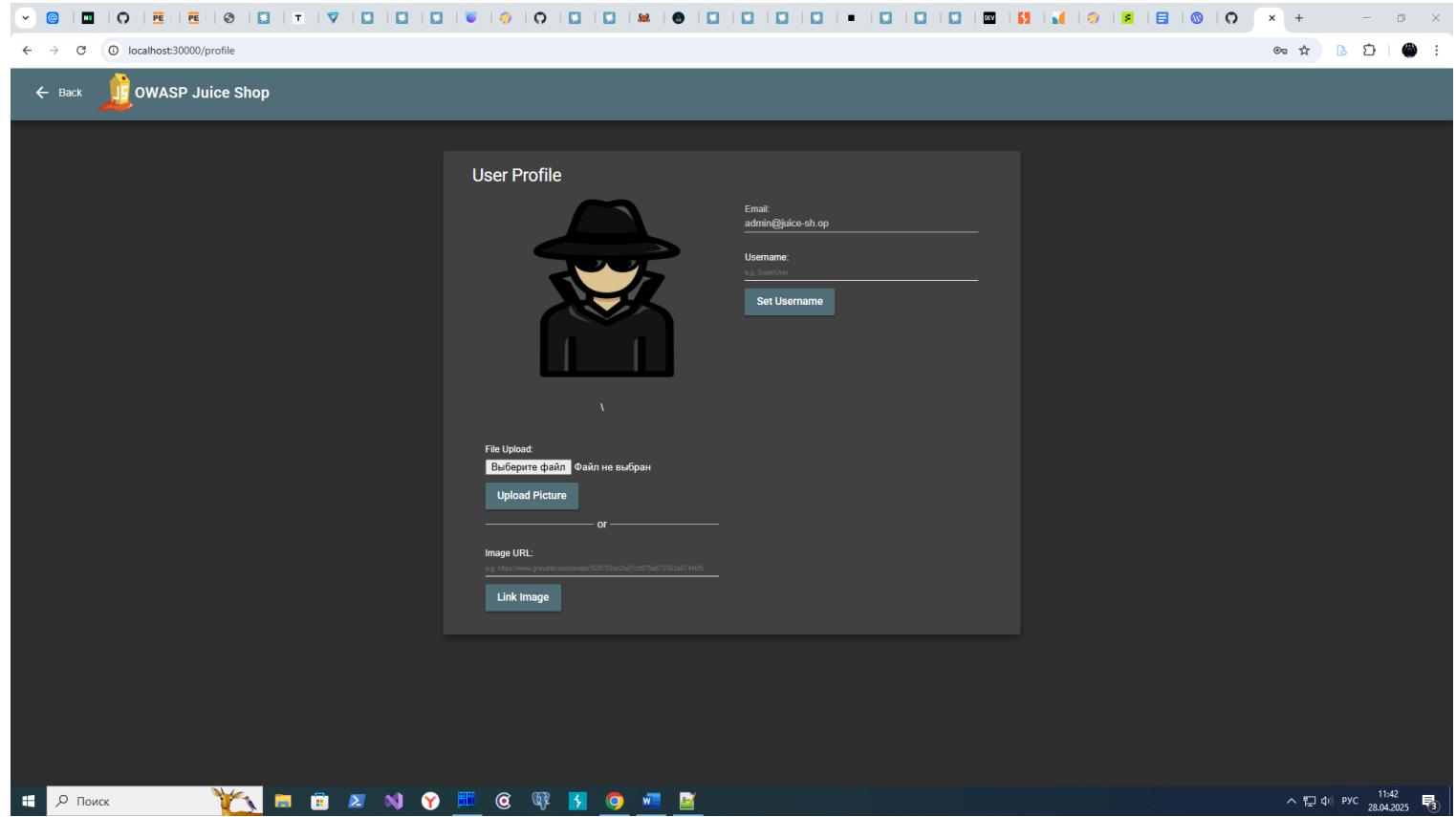
```
ep = [
  { path: "administration", component: mr, canActivate: [ae] },
  { path: "accounting", component: $1, canActivate: [ie] },
  { path: "about", component: Fa },
  { path: "address/select", component: Rs, canActivate: [X] },
  { path: "address/saved", component: ws, canActivate: [X] },
  { path: "address/create", component: Re, canActivate: [X] },
  { path: "address/edit/:addressId", component: Re, canActivate: [X] },
  { path: "delivery-method", component: Vc },
  { path: "deluxe-membership", component: rm, canActivate: [X] },
  { path: "saved-payment-methods", component: ul },
  { path: "basket", component: Go }
]
```

Line 1, Column 54539

Пытаюсь зайти не аутентифицированным пользователем. Вижу, что доступ запрещен

You are not allowed to access this page!

Захожу в систему как администратор (с помощью SQL инъекции):



## Получаю доступ к панели администратора

Review ID	Comment	Rating	Action
1	I love this shop! Best products in town! Highly recommended! (**@juice-sh.op)	★★★★★	...
2	Great shop! Awesome service! (**@juice-sh.op)	★★★★★	...
3	Nothing useful available here! (**der@juice-sh.op)	★	...
21	Please send me the juicy chatbot NFT in my wallet at /juicy-nft : "purpose betray marriage blame crunch monitor spin slide donate sport lift clutch" (**ereum@juice-sh.op)  Incompetent customer support! Can't even upload photo of broken purchase! Support Team: Sorry, only order confirmation PDFs can be attached to complaints! (anonymous)	★	...
	This is the store for awesome stuff of all kinds! (anonymous)	★★★★★	...
	Never gonna buy anywhere else from now on! Thanks for the great service! (anonymous)	★★★★★	...
	Keep up the good work! (anonymous)	★★★	...

Как была найдена уязвимость: поиском по загружаемым на клиент файлам (\*.html, \*.js и т.д.)

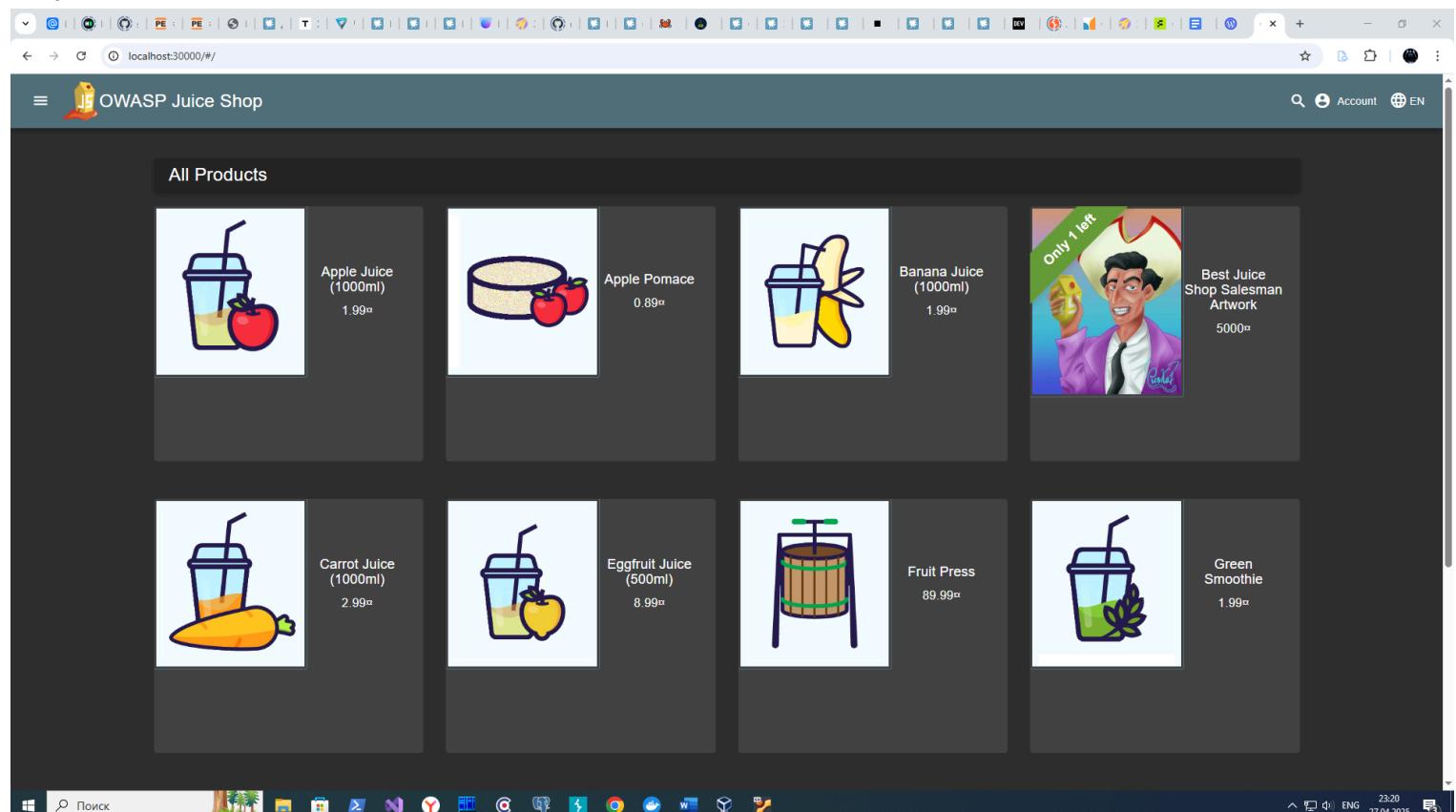
Как можно защититься от данной уязвимости:

- провести ревью кода на фронтенде и убрать из маршрутов в клиентских скриптах относительные пути на скрытую функциональность
- провести ревью кода на бекенде и убрать не используемую функциональность из маршрутов и из кода вообще
- провести ревью всего приложения с целью выявления (и последующего избавления) от не используемой функциональности (панель администратора доступна администратору только через ввод соответствующего URL в адресной строке браузера)
- можно попробовать защититься от этой уязвимости и с помощью WAF, но это лечение последствий, а не причины

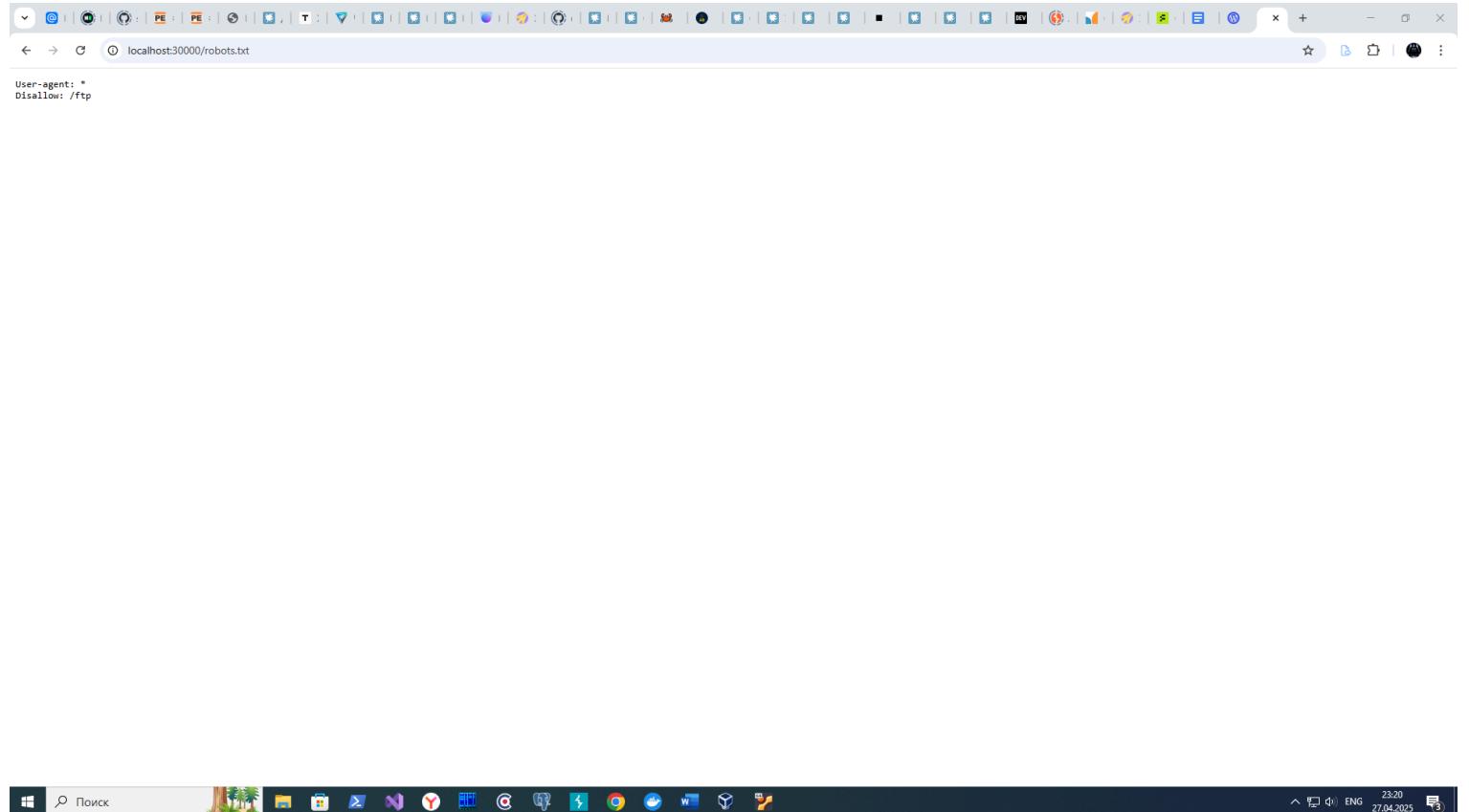
Какие риски несет данная уязвимость: по умолчанию у пользователя администратор не доступа к панели администратора через элементы UI - только через ввод соответствующего URL в адресной строке браузера, т.е. эта функциональность, по идеи вообще убрана у учетной записи администратора. Соответственно, данная скрытая функциональность повышает возможности злоумышленника, взломавшего учетную запись администратора, в нанесении потенциального вреда

## 5. Backup & Crack

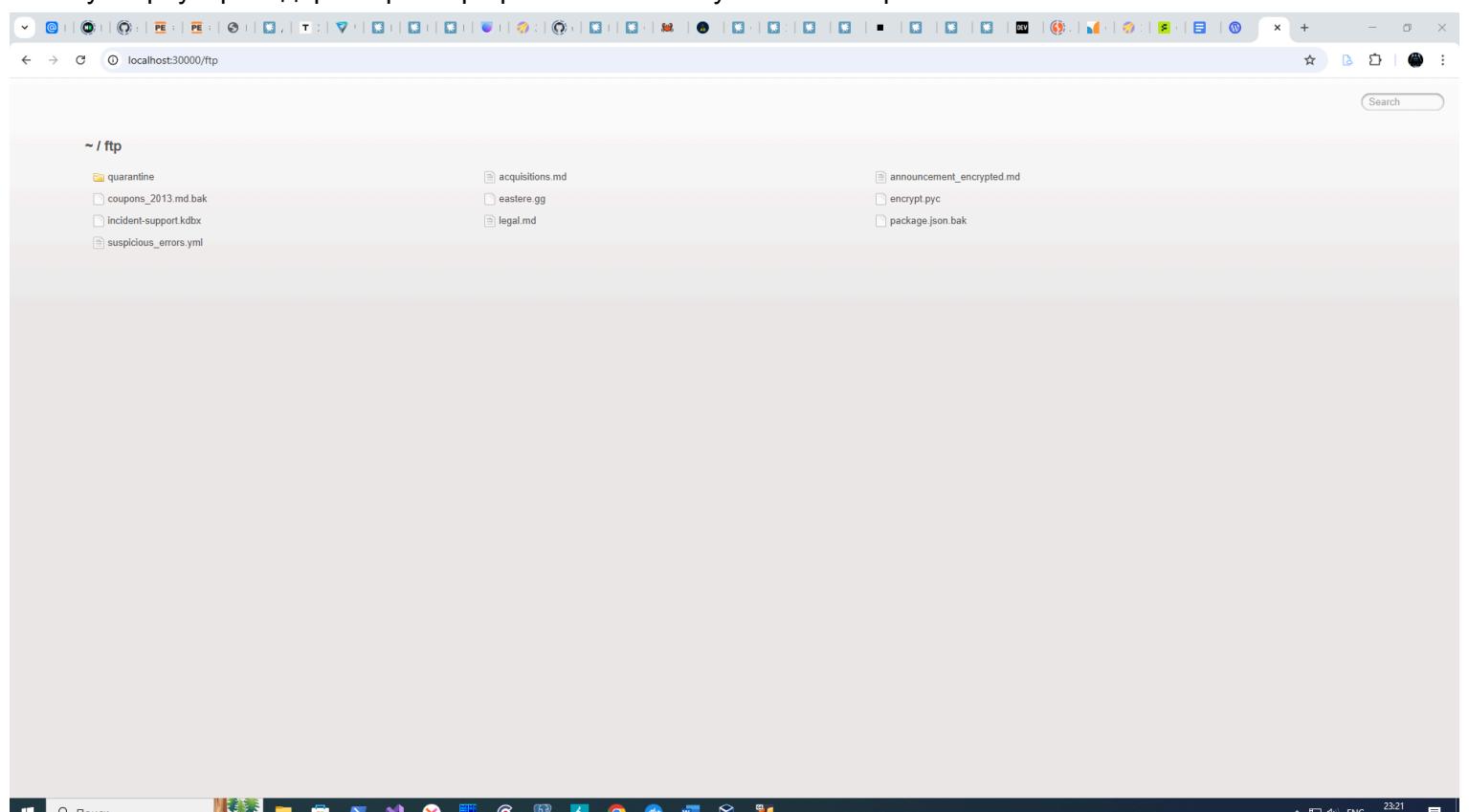
Запускаю приложение



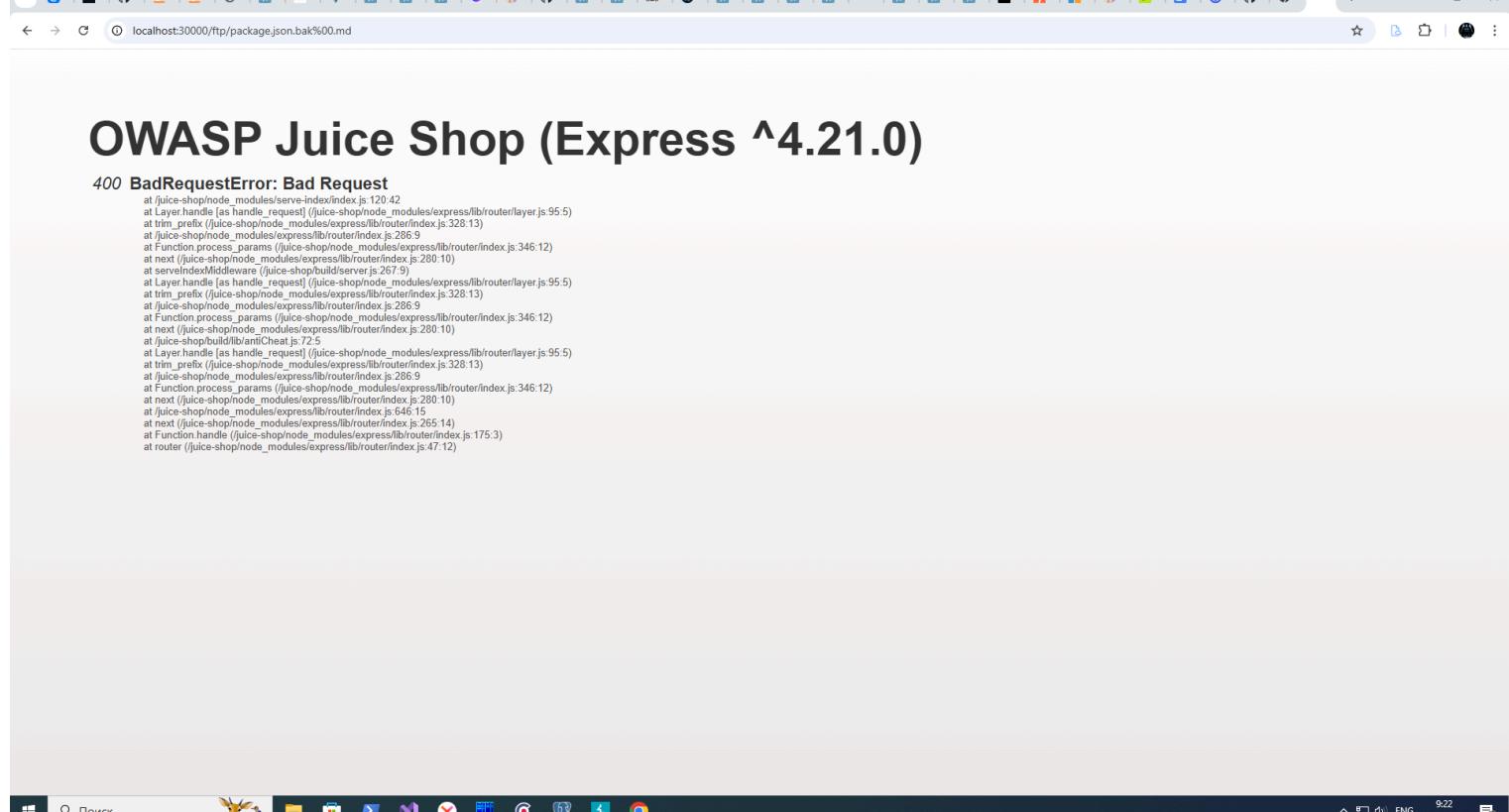
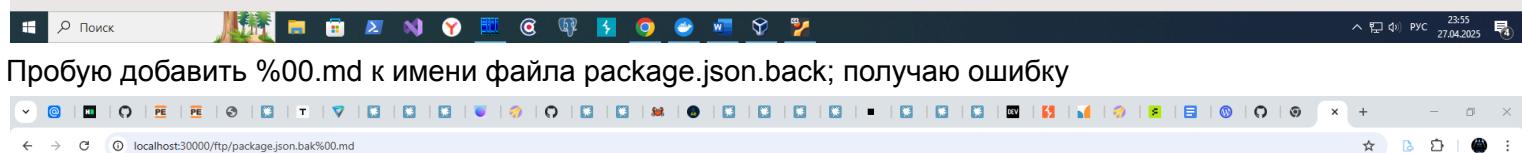
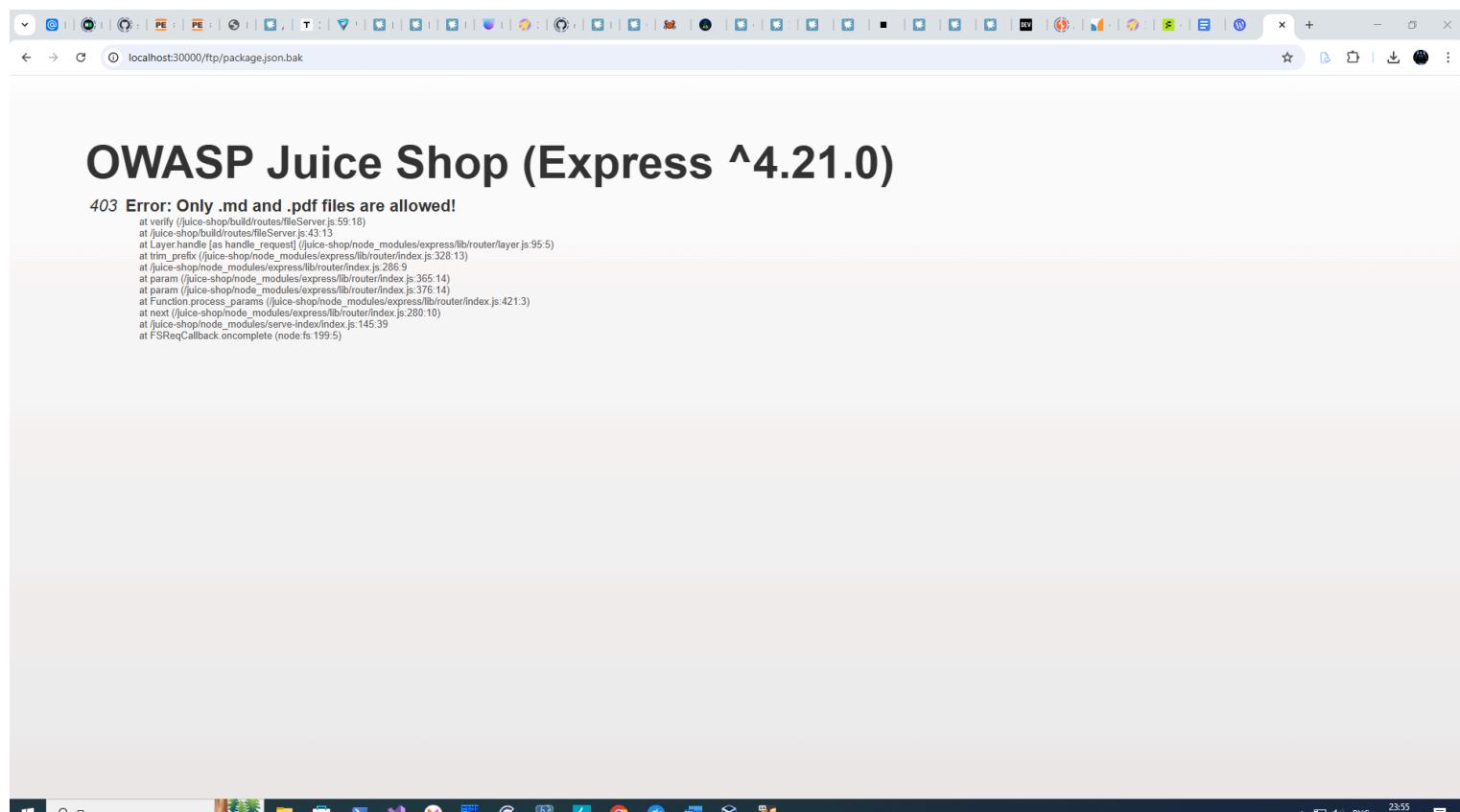
Смотрю, что есть в файле robots.txt (и есть ли он вообще). Вижу, что есть некоторая директория ftp



Захожу в браузере в директорию ftp приложения. Вижу несколько файлов.



Пробую скачать файл package.json.back; получаю ошибку



C:\Users\std\_string\Downloads\package.json.bak%00.md - Notepad++  
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?  
links.txt Makefile description.md Dockerfile weather\_control.yaml lights\_demo.yaml prepared-demo.sql new 1 package.json.bak%00.md  
1 (1)  
2 . "name": "juice-shop",  
3 . "version": "6.2.0-SNAPSHOT",  
4 . "description": "An intentionally insecure JavaScript Web Application",  
5 . "homepage": "http://owasp-juice.shop",  
6 . "author": "Björn Kimmiminich <bjoern.kimmiminich@owasp.org> (<https://kimmiminich.de>)",  
7 . "contributors": [  
8 . . . "Björn Kimmiminich",  
9 . . . "Jannis Hollenbach",  
10 . . . "Aashish693",  
11 . . . "greenkeeper[bot]",  
12 . . . "MarcLler",  
13 . . . "agrawalarpit14",  
14 . . . "Scar26",  
15 . . . "CaptainFreak",  
16 . . . "Supratik Das",  
17 . . . "JuiceShopBot",  
18 . . . "the-pro",  
19 . . . "Ziyang Li",  
20 . . . "aryan10",  
21 . . . "milc3",  
22 . . . "limo Page1",  
23 . . . "...",  
24 . ],  
25 . "private": true,  
26 . "keywords": [  
27 . . . "Web security",  
28 . . . "web application security",  
29 . . . "webappsec",  
30 . . . "owasp",  
31 . . . "pentest",  
32 . . . "penetration",  
33 . . . "security",  
34 . . . "vulnerable",  
35 . . . "vulnerability",  
36 . . . "broken",  
37 . . . "podgeit",  
38 . . . ],  
39 . "dependencies": {  
40 . . . "body-parser": "~1.18",  
41 . . . "colors": "~1.1",  
42 . . . "config": "~1.28",  
43 . . . "cookie-parser": "~1.4",  
44 . . . "cors": "~2.8",  
45 . . . "dottie": "~2.0",  
46 . . . "escodegen": "~0.7",  
47 . . . "errorhandler": "~1.5",  
48 . . . "express": "~4.16",  
49 . . . "express-jwt": "~0.1.3",  
50 . . . "expressss": "~4.0",  
51 . . . "glob": "~5.0",  
52 . . . "grunt": "~1.0",  
53 . . . "grunt-angular-templates": "~1.1",  
54 . . . "grunt-contrib-clean": "~1.1",  
55 }  
User Defined language file - Markdown (preinstalled) length: 4291 lines: 178 Ln: 1 Col: 1 Pos: 1 Unix (LF) UTF-8 INS  
Windows taskbar: Помощь Пуск Панель задач Документы Яндекс Браузер Калькулятор Папка Старт Помощь 23:59 ENG 27.04.2025

Нахожу в этом файле строку с хешем пароля администратора:

"admin": {"password": "0192023a7bbd73250516f069df18b500"}

Пробую подобрать по хешу пароль, исходя из предположения, что используется слабый алгоритм хеширования; например, md5.

The screenshot shows the iTools MD5-decrypt interface. In the 'ВХОД' (Input) section, the MD5 hash '0192023a7bbd73250516f069df18b500' is entered. In the 'ВЫХОД' (Output) section, the decrypted password 'admin123' is displayed. A message at the top states: 'Большое обновление нашего инструмента расшифровки MD5! Теперь брутфорс в 20 раз быстрее, и мы добавили еще 3 миллиона предварительно вычисленных хешей. Инструмент расшифровки MD5 теперь поддерживает предварительно вычисленные хеши с более чем 11 миллионами записей (необязательно, поэтому если отключено, процесс расшифровки теперь можно выполнить полностью в вашем браузере).'. On the right, there's a sidebar with various tools like 'Удаление фона' and 'Редактор изображений на базе ИИ'.

Получаю, что пароль администратора admin123

Проверяю, что данный пароль действителен (адрес электронной почты администратора я знаю из результатов эксплуатации SQL инъекции для входа - это [admin@juice-sh.op](mailto:admin@juice-sh.op))

The screenshot shows the OWASP Juice Shop login page. The URL in the address bar is 'localhost:30000/#/login'. The login form has 'Email\*' set to 'admin@juice-sh.op' and 'Password\*' set to 'admin123'. Below the form, a success message 'Login successful!' is displayed. The background of the page is dark.

Вход с такими данными был успешен. Смотрю профиль пользователя - это администратор

The screenshot shows a web browser window with the URL `localhost:30000/profile`. The page title is "User Profile" from "OWASP Juice Shop". On the left, there is a placeholder image of a hooded figure. To the right, the user's email is listed as `admin@juice-sh.op`, and the role is indicated as "SuperUser". A "Set Username" button is present. Below this, there are fields for "File Upload" (with a placeholder "Выберите файл" and note "Файл не выбран") and "Upload Picture" (a blue button). An "OR" separator leads to another section for "Image URL" (with a placeholder URL) and a "Link Image" button.

Как была найдена уязвимость: поиск интересных директорий в стандартных местах - таких, как robots.txt

Как можно защититься от данной уязвимости:

- у пользователей, не прошедших аутентификацию, должны быть минимальные права
- жесткий контроль того, что доступно через http, ftp и другие протоколы снаружи
- использование сложных паролей (чтобы было невозможно осуществить подбор по словарю) и сильных алгоритмов хеширования (причем с солью)

Какие риски несет данная уязвимость: Доступ к резервным копиям и другим приватным файлам может позволить злоумышленникам получить доступ к конфиденциальной информации, включая пароли и личные данные пользователей.