Juice-Shop exploring Report

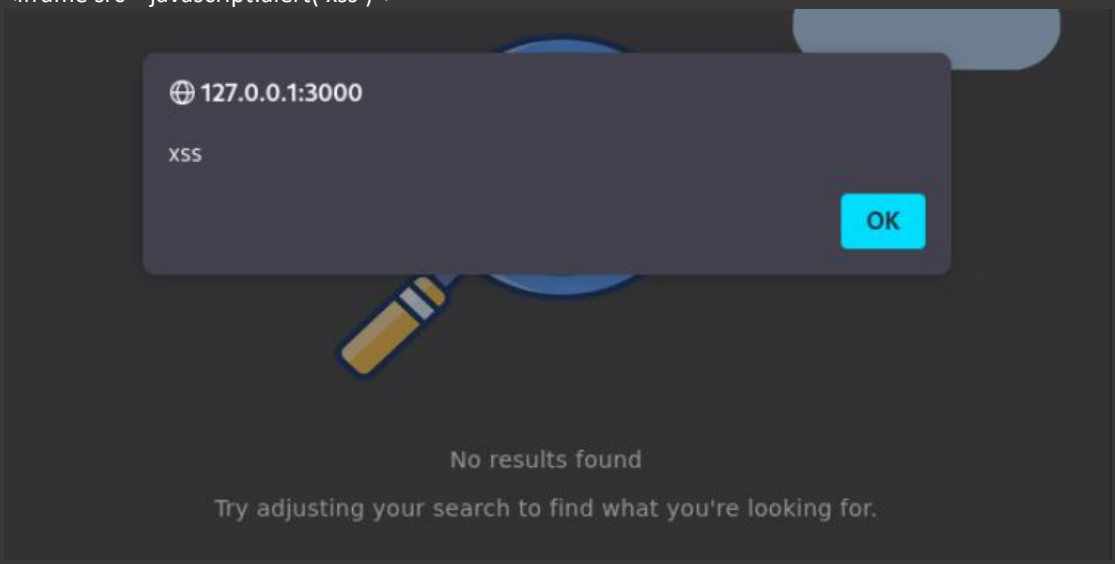
Juice-shop - interactive vulnerable platform, where entry-level pentesters can apply their skills and knowledge, put teory to practice.

First that we need to do - install npm server. This will allow us to have access to site every time when we need it. There are several steps to achieve this.
1. Install nodejs through terminal
Sudo nodejs install
2. Instal npm
Sudo npm install
3. Install juice-shop from github
Git clone https://github.con/juice-shop/juice-shop.git
4. Enter to the fold Juice-shop
Cd juice-shop
5. Start npm-server
Npm start
6. Open site with browser on the port 3000.
127.0.0.1:3000/

Tasks level EASY

1. XSS-injection.

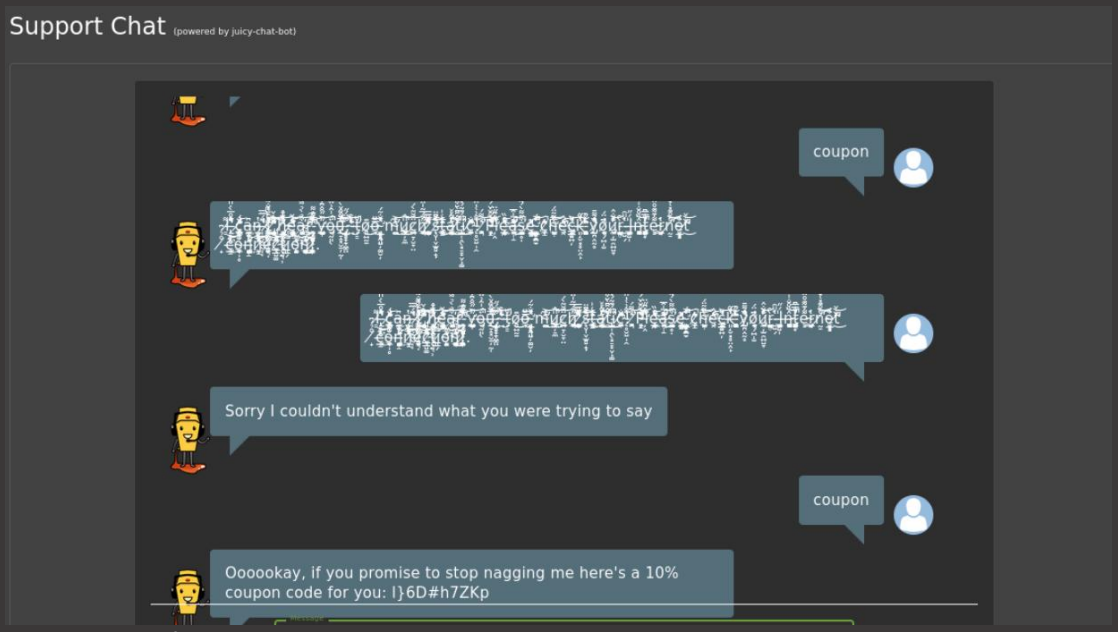
This term means a web-security vulnerability where an attacker can inject malicious scripts into trusted web-sites.
For our understanding if site vulnerable to XSS-injection or not we can use search-field. For this we can enter <h1>owasp. We can see that site do not resist. So we can use next script:
<iframe src=”javascript:alert(‘xss’)”>



As result we can see next changes in the address field: 127.0.0.1:3000/#/search?q=<iframe src%3D”javascript:alert(‘xss)”>

2. Brutforce chat-bot

Bruteforce this is trial-or-error method, that implies repetitive tries with every possible combinations. Using this method while talking with site’s bot, we can reach next results:



3. Searching for a hidden directories.

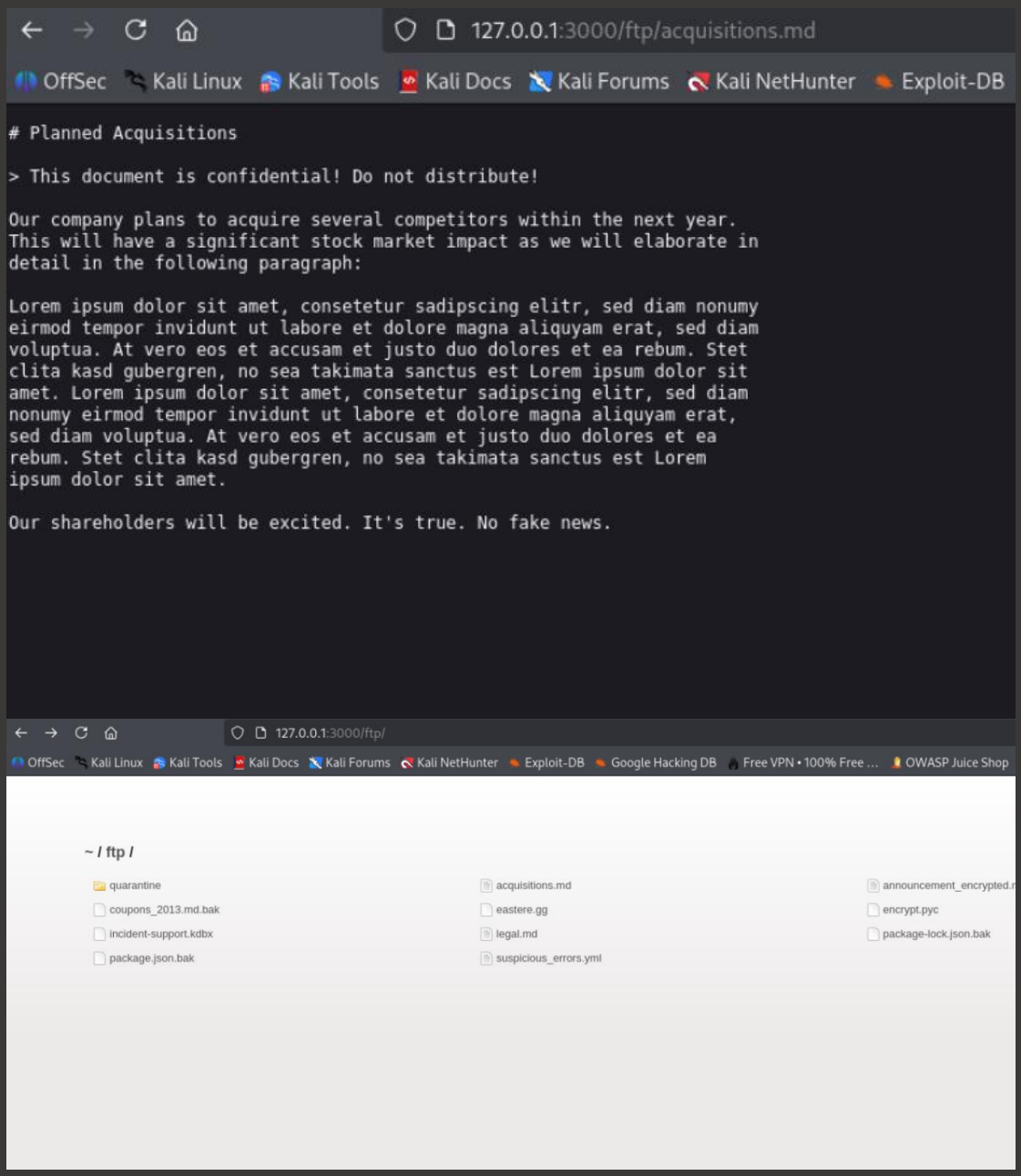
For this purpose we can use powerful utilita dirb.

Dirb http://127.0.0.1:3000 -f

```
┌──(vbox㉿vbox)-[~/Downloads/juice-shop]
└─$ dirb http://127.0.0.1:3000 -f

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Thu Dec  4 16:45:22 2025
URL_BASE: http://127.0.0.1:3000/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Fine tunning of NOT_FOUND detection

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://127.0.0.1:3000/ ----
+ http://127.0.0.1:3000/assets (CODE:301|SIZE:156)
+ http://127.0.0.1:3000/ftp (CODE:200|SIZE:941)
+ http://127.0.0.1:3000/profile (CODE:500|SIZE:1061)
+ http://127.0.0.1:3000/promotion (CODE:200|SIZE:692)
+ http://127.0.0.1:3000/redirect (CODE:500|SIZE:3559)
+ http://127.0.0.1:3000/robots.txt (CODE:200|SIZE:3)
+ http://127.0.0.1:3000/video (CODE:200|SIZE:263788)
+ http://127.0.0.1:3000/Video (CODE:200|SIZE:263788)

(!) FATAL: Too many errors connecting to host
    (Possible cause: COULDNT CONNECT)

-----------------
END_TIME: Thu Dec  4 16:47:48 2025
DOWNLOADED: 4589 - FOUND: 8
```

Code 200 allows us to see which directories are active and exists in the access zone.

One of them is directory /ftp, where we can find a document acquisitions.md, that suppose to be confidential.

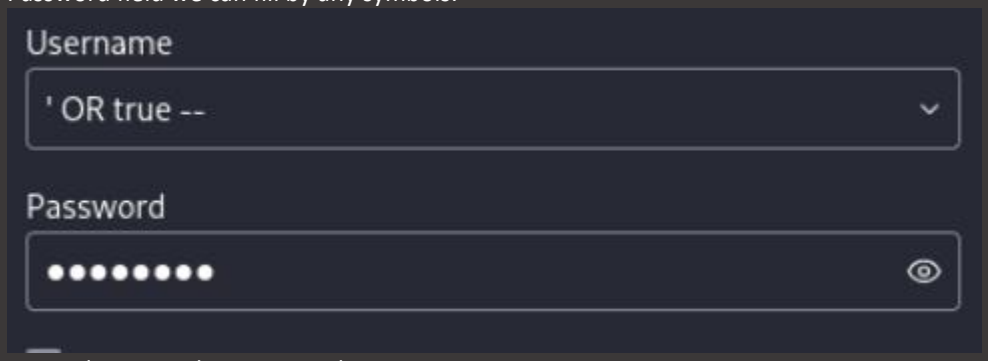


4. SQL-injection for password bypassing

We can start our penetration in the registration form. For this we can put next to the username field:

- ‘
- ‘ or true
- ‘ or true --

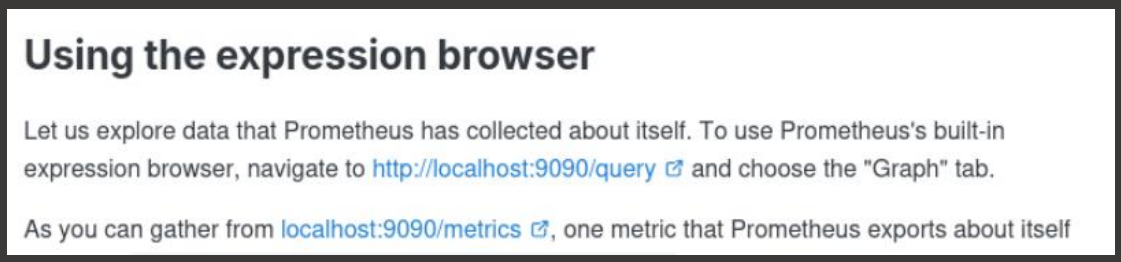
Password field we can fill by any symbols.



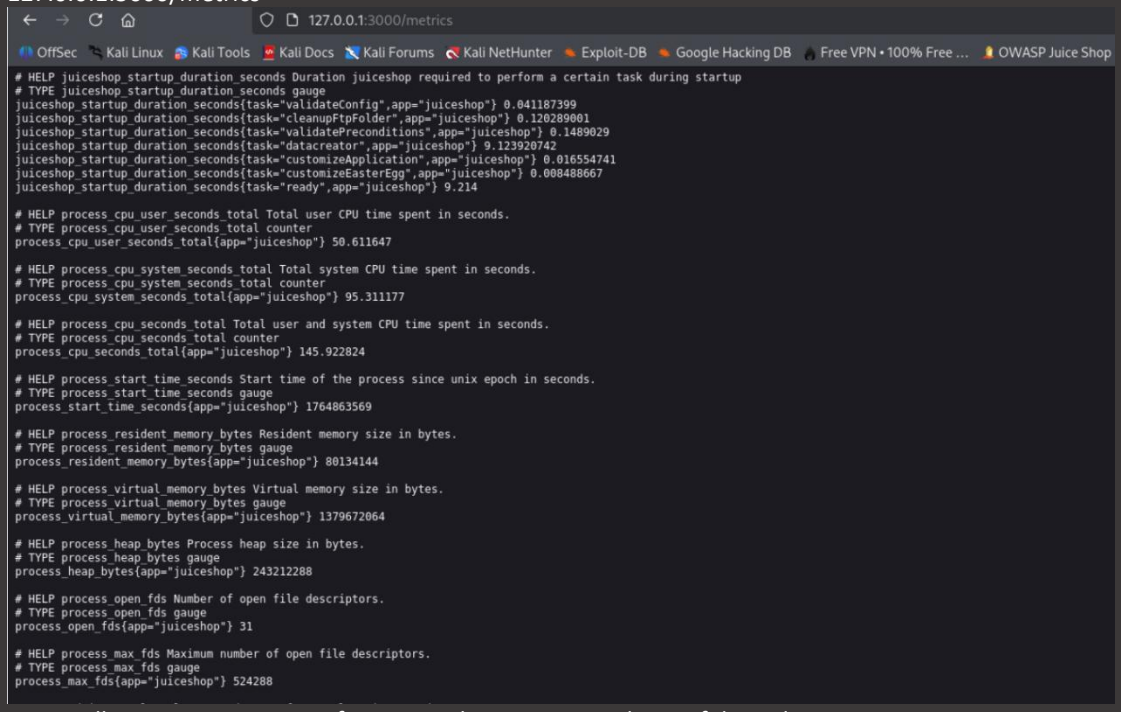
As result, we can login as an adiministrator.

5. Metrics

Metrics are very important entity. They can provide information about errors, requests, time and memory, that server needs. Prometheus - system of monitoring - can disclose information about exposed metrics on site.



So, based on the above, we can use next link:
127.0.0.1:3000/metrics

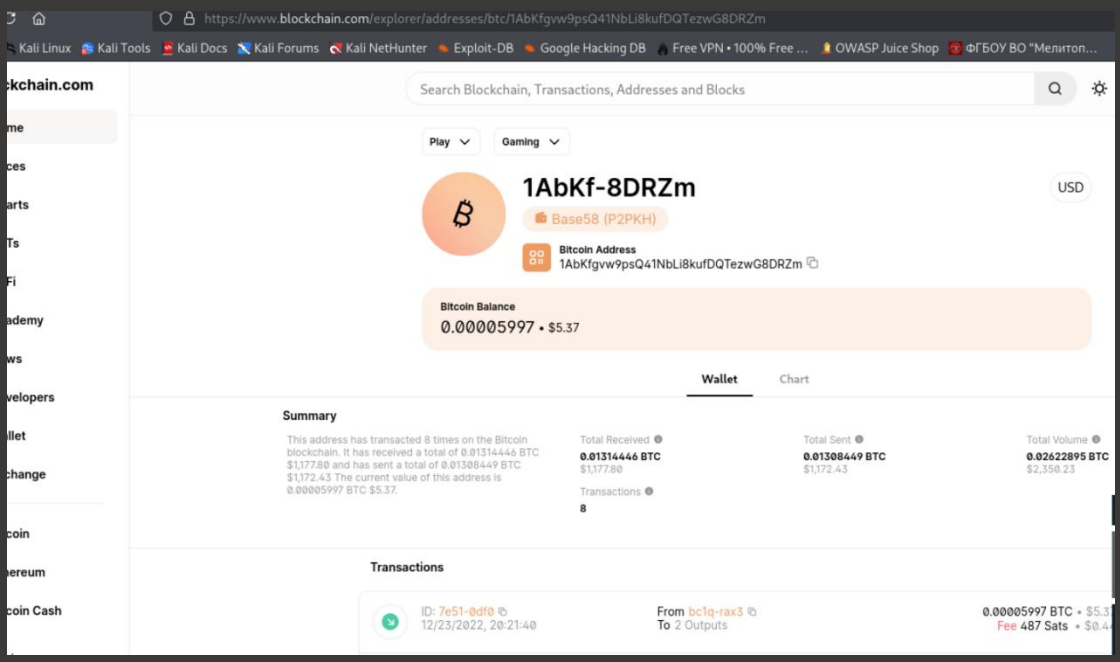


Eventually, we get access to information that suppose to be confidential.

6. Searching for hidden list.

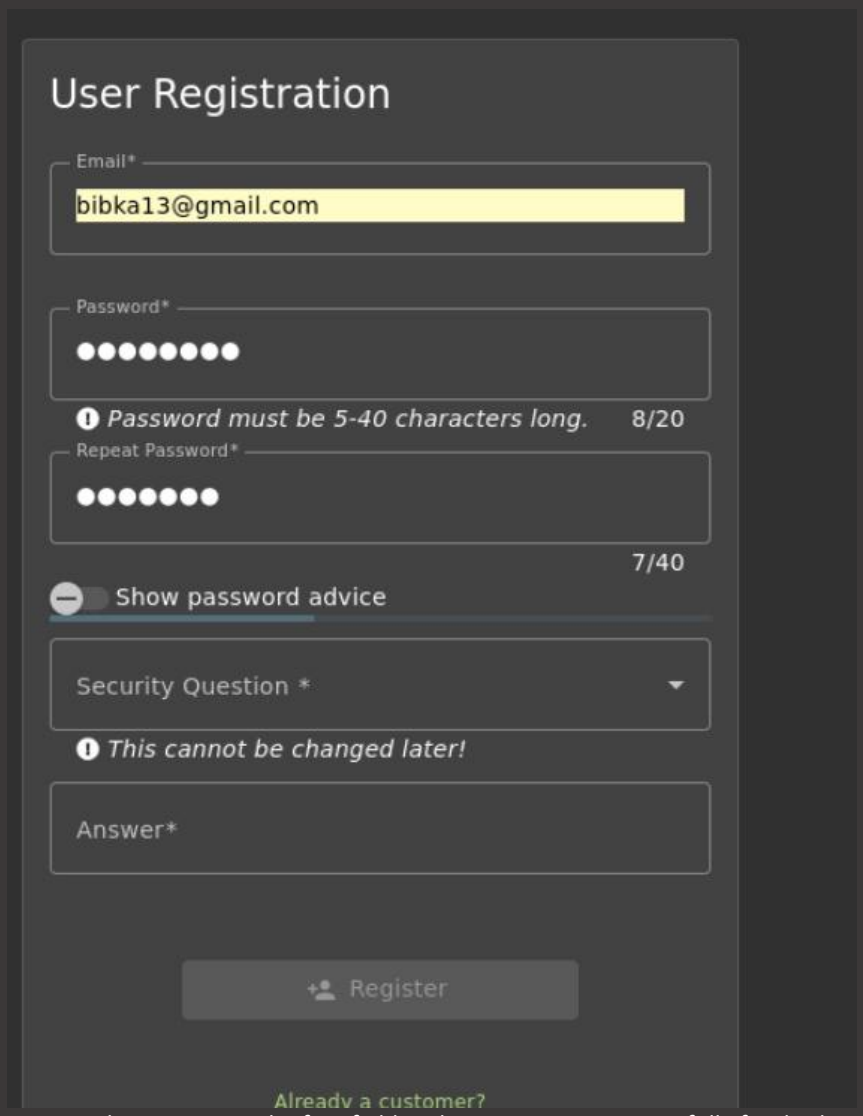
For this purpose we can save the file main.js and looking carefully into the text.
There we can see hidden link

This link leads us to cryptowallet.



7. DRY principle

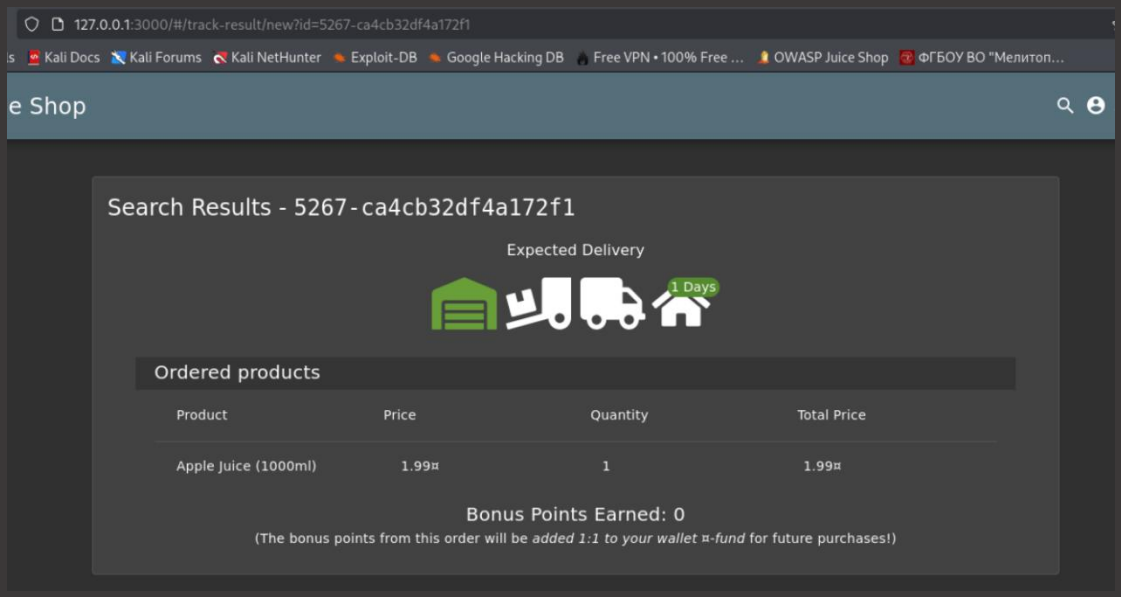
Dry principle stands for “don’t repeate yourself”.



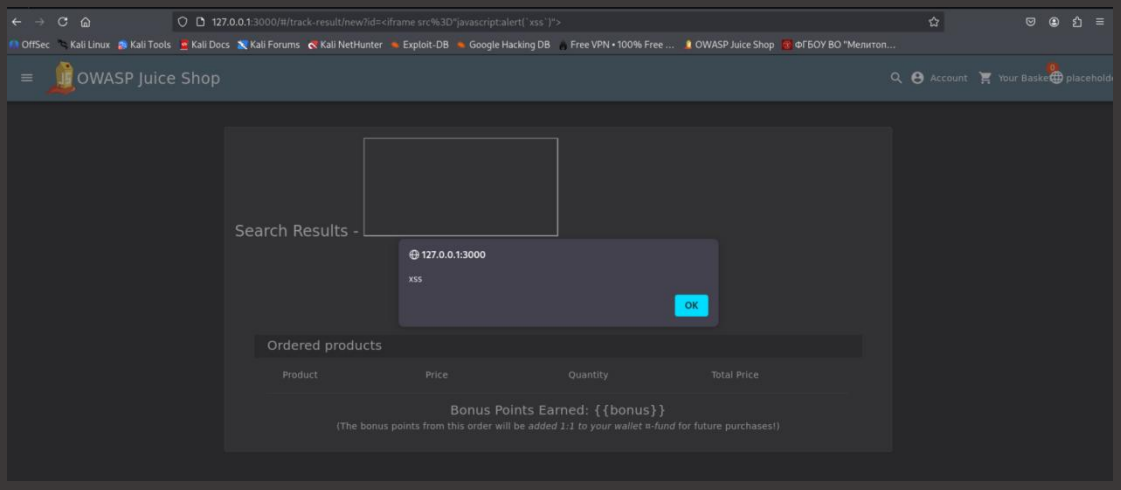
We just change pass in the first field and registration is successfully finished

8. Reflected XSS Attack

Reflected XSS attack implies injection of malicious script into a website via user input (like URL parameter).
We suppose to find a reference that has parameter “id” in it’s structure. Exploring whole site we see it on the track-result page.

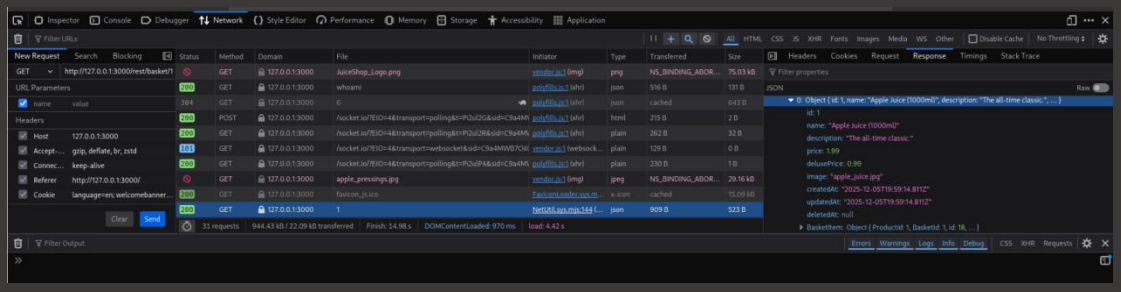


Here we change an id parameter value for <iframe src=javascript:alert(‘xss’)”>, and after reloading of page we can see pop-up window, that testifies that script is done



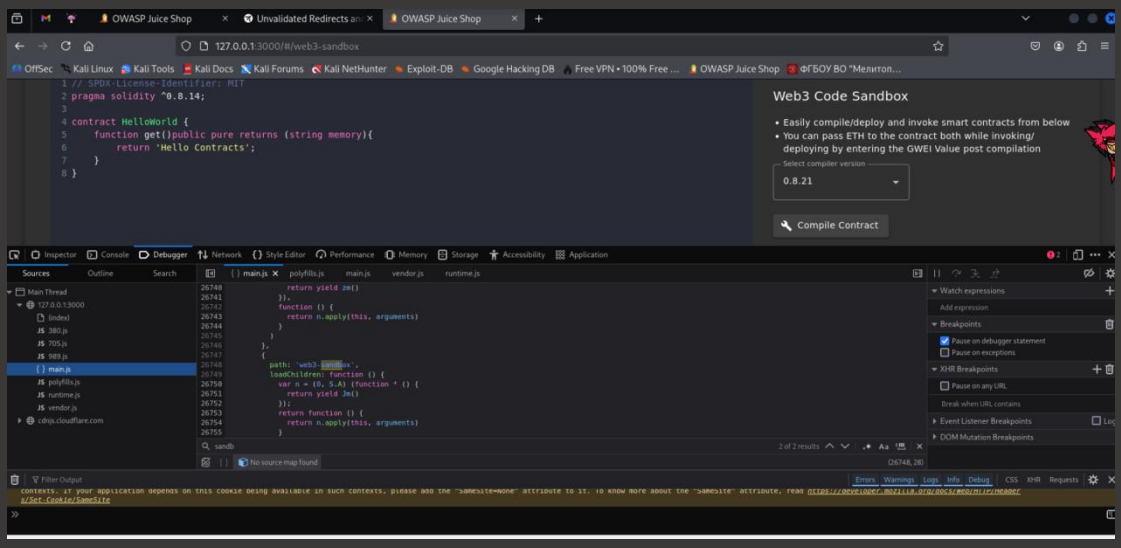
9. Another user’s basket viewing

For achieving this we use the developer mode, go to the network tab and through requests chanching basket’s id in th GET request.



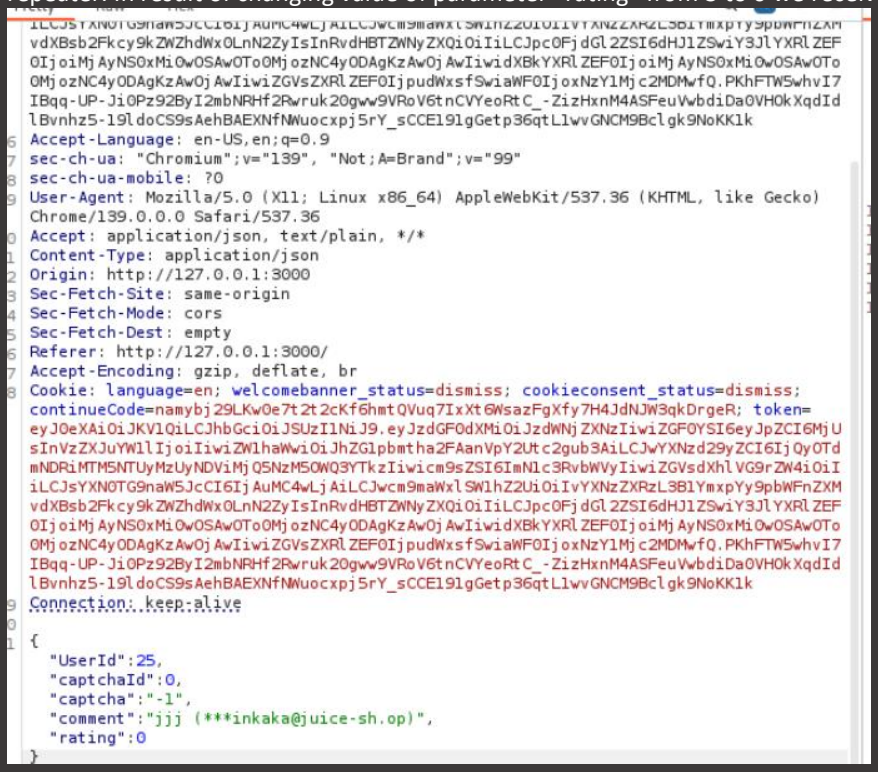
10. Hidden sandbox

We can fin d hidden path in the main.js.

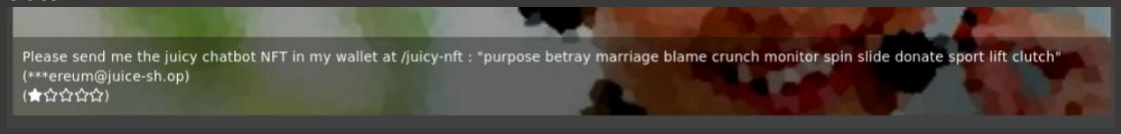


11. 0 starts feedback

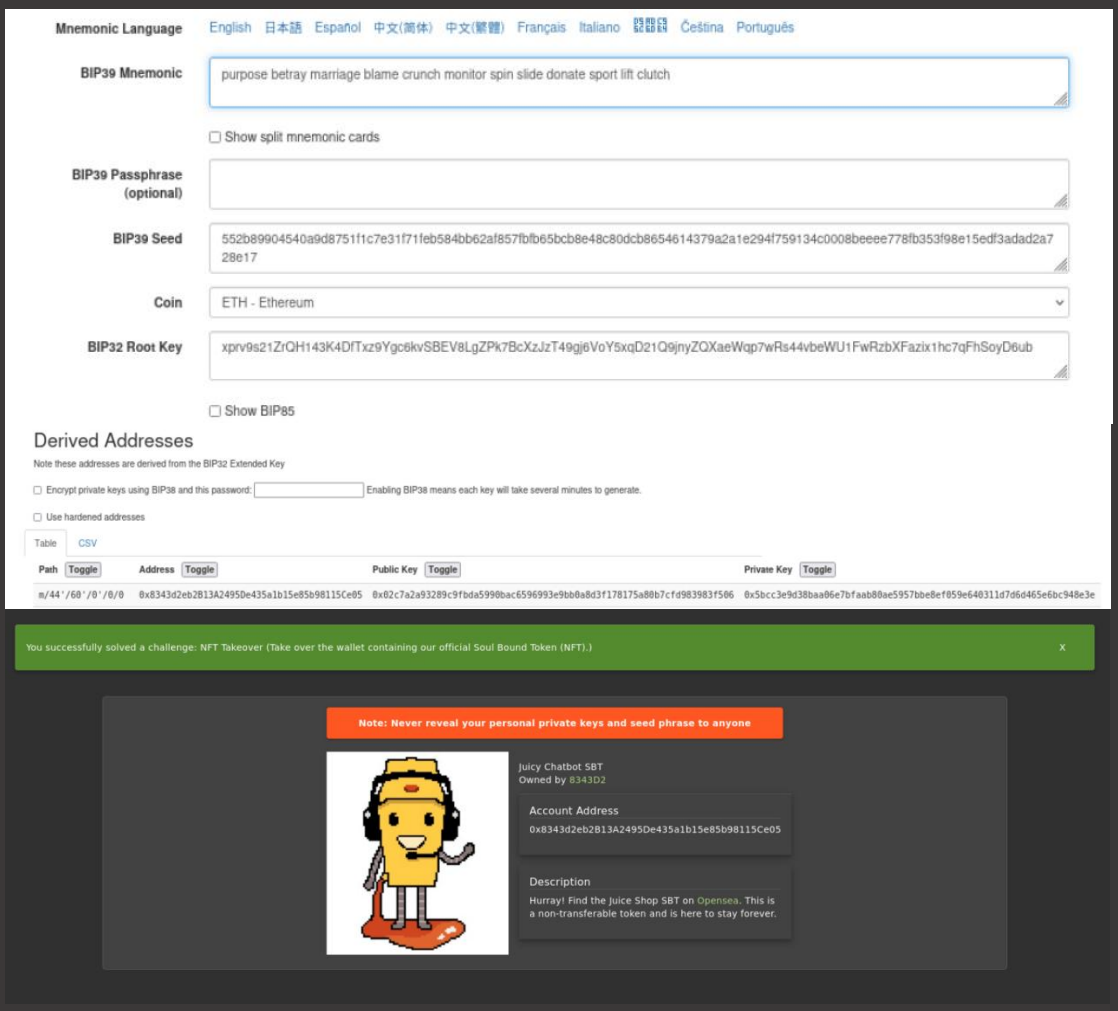
For this purpose we used BurpSuit. With this tool we caught POST request and change it with repeater. In result of changing value of parameter “rating” from 3 to 0 we receive executing a request.



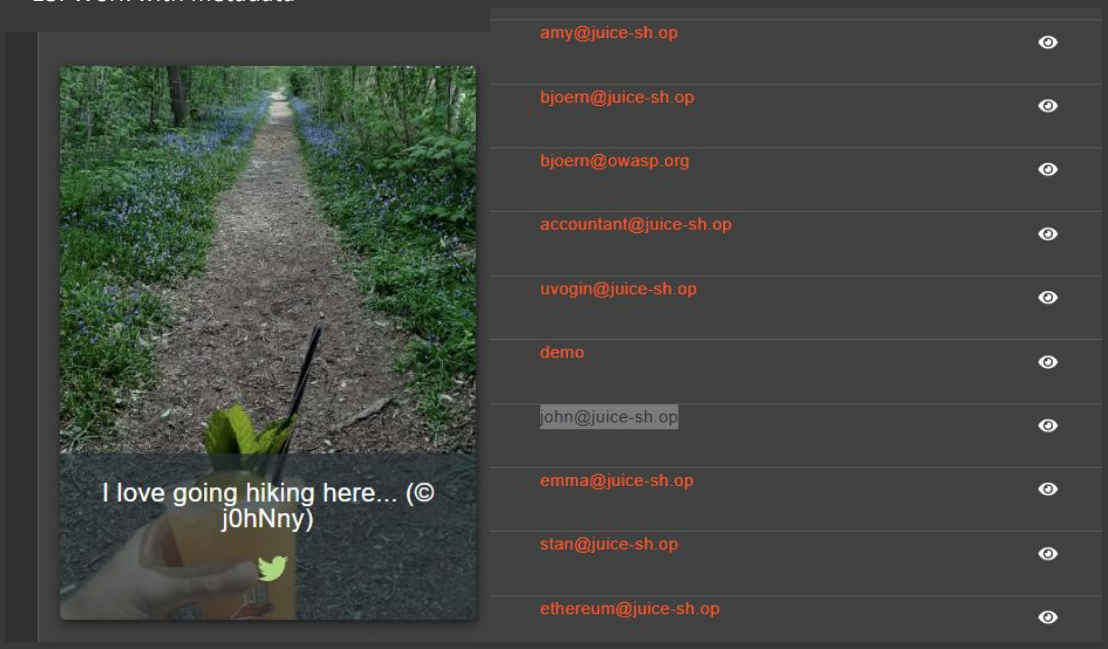
12. To get acces to NFT-wallet data we find seedphrase, that disclosed on the feedbacks page. We see the seed phrase: purpose betray marriage blame crunch monitor spin slide donate sport lift clutch.



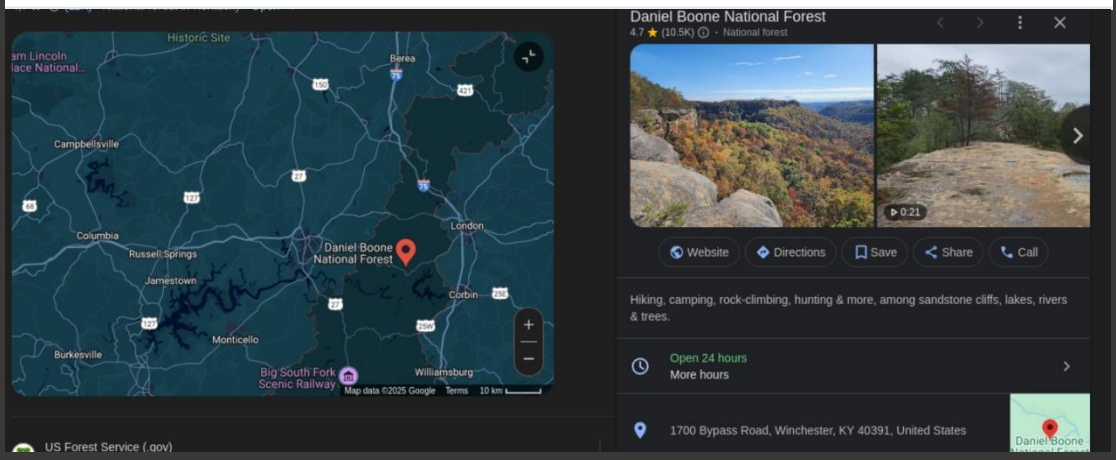
Web-tool iancoleman.io helps to convert it to BIP39 Seed.



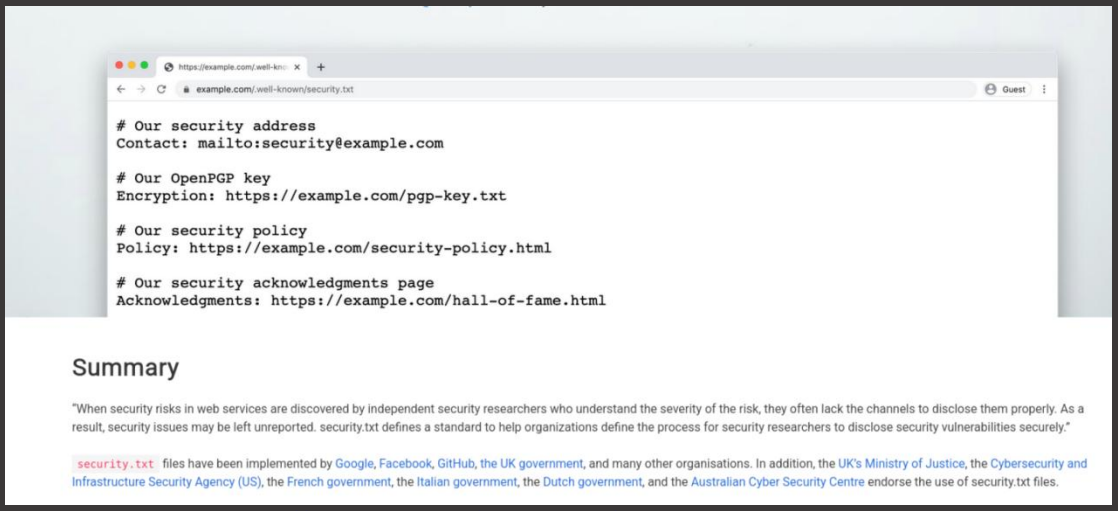
13. Work with metadata



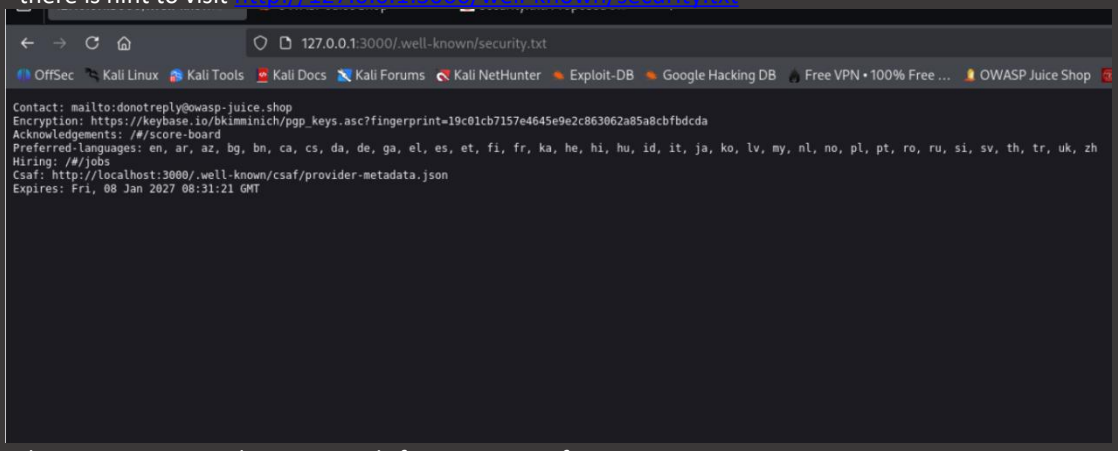
thumbnail_length	4531
srgb_rendering	Perceptual
gamma	2.2
pixels_per_unit_x	3779
pixels_per_unit_y	3779
pixel_units	meters
image_size	471x627
megapixels	0.295
thumbnail_image	(Binary data 4531 bytes)
gps_latitude	36 deg 57' 31.38" N
gps_longitude	84 deg 20' 53.58" W
gps_position	36 deg 57' 31.38" N, 84 deg 20' 53.58" W
category	image



14. White-hat behavior.
- visit the site securitytxt.org



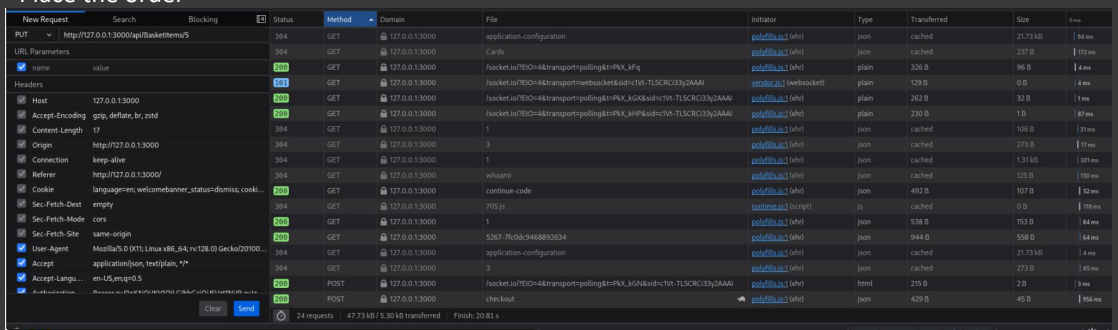
- there is hint to visit http://127.0.0.1:3000/well-known/security.txt

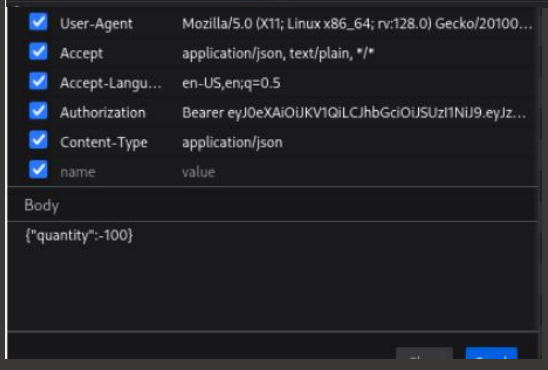


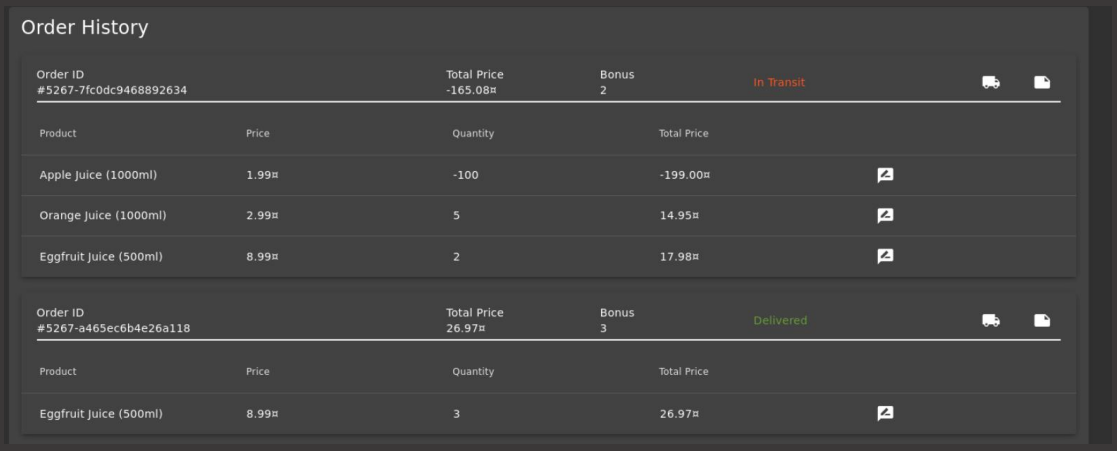
- there are contacts, that are stands for agreement for pentesting.

15. Add a negative quantity to the cart using edit code.

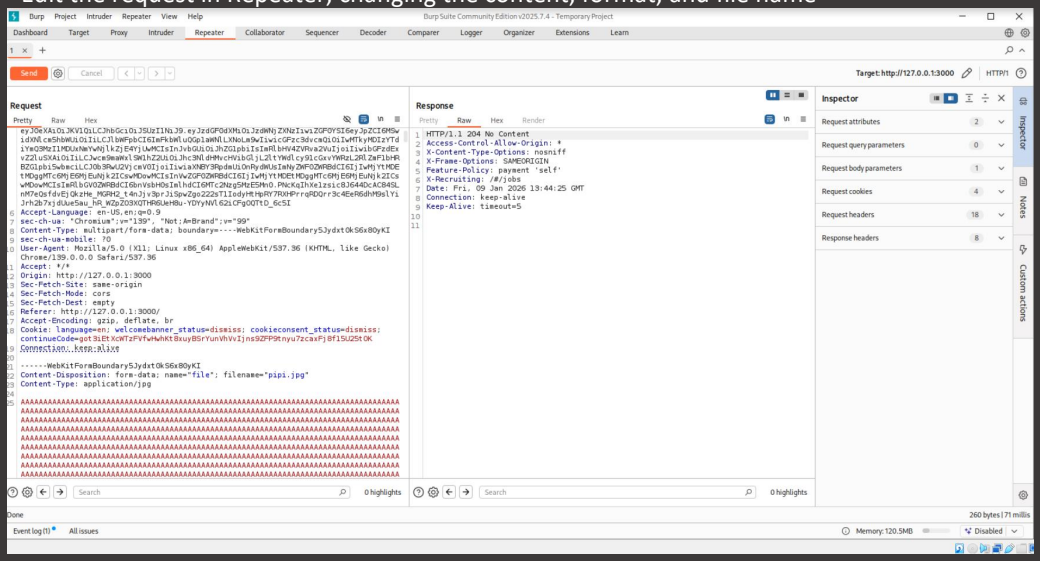
- To do this, modify the PUT request so it looks like this:
- The product ID matches our product
- Change the Authorization token to any other one from another PUT
- In the body, change the quantity to -100
- Place the order





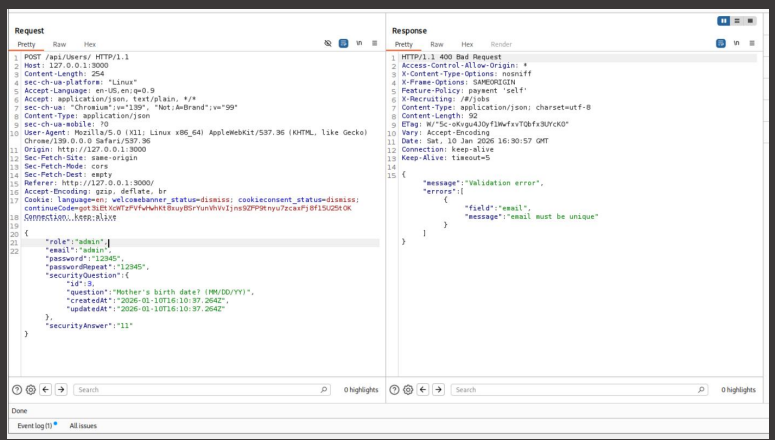


16.- Intercept the POST request when sending a file that doesn't violate the size limit with burp suit - Edit the request in Repeater, changing the content, format, and file name



---

17. For creating an account with role that will be “admin” is possible to do next steps:
1) Open BurpSuite >> proxy >> 127.0.0.1:3000
2) Intercept the POST request that has connection with API named Users and send it to repeater.
3) Modify it by adding new parameter named “role”
4) Assing the value for this parametr named “admin”
5) Send the modifyed request vie the repeater



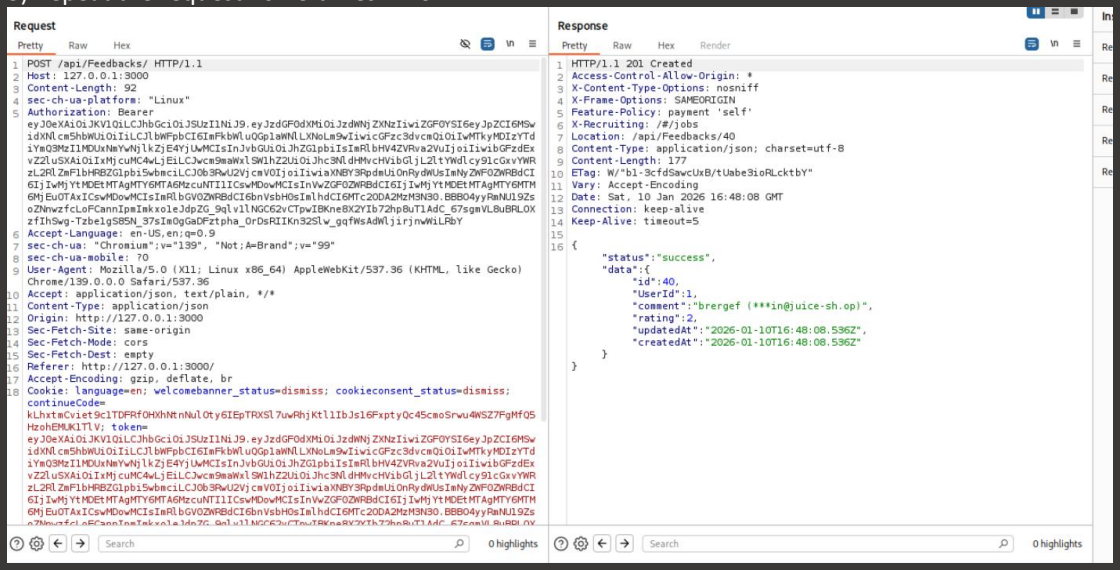
18. Captcha Bypass

This task also can be complete with help of BurpSuite. And it can be done in next few steps.

1) Intercept the request POST, while sending the normal feedback

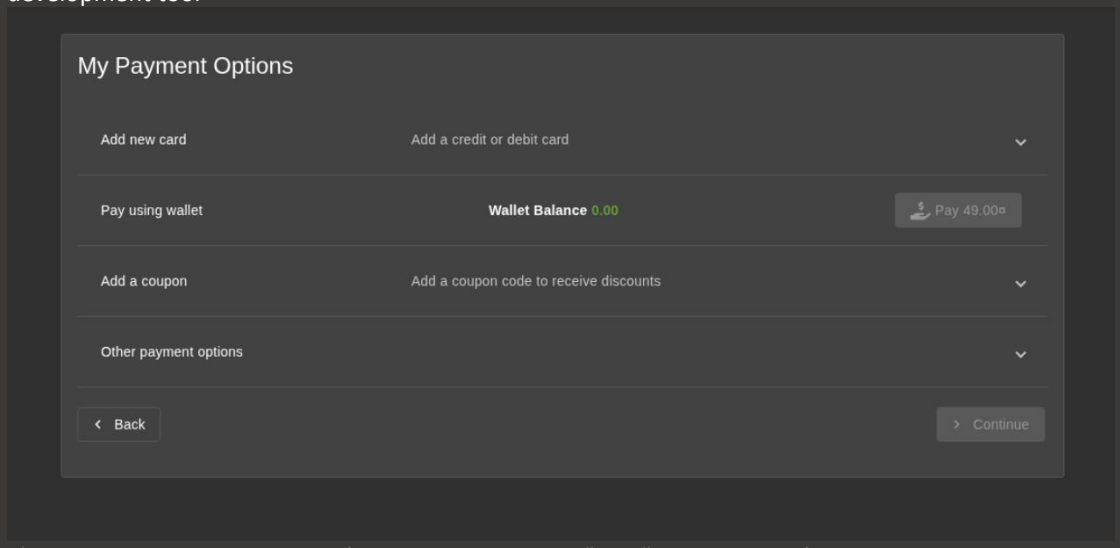
2) Sent it to repeater

3) Repeat the request 10-15 times in row

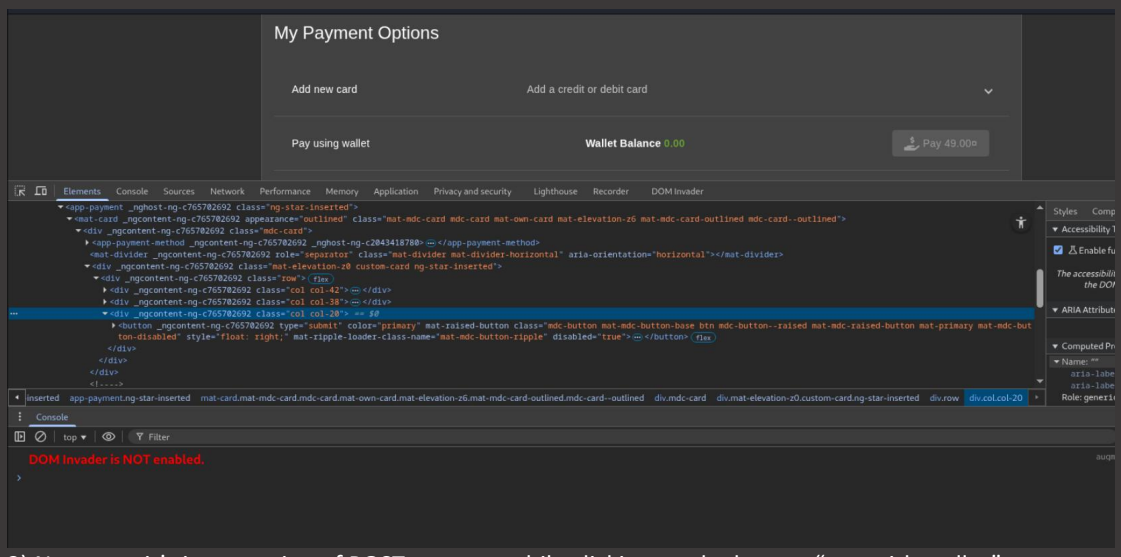


19. Become deluxe-member without paying for it

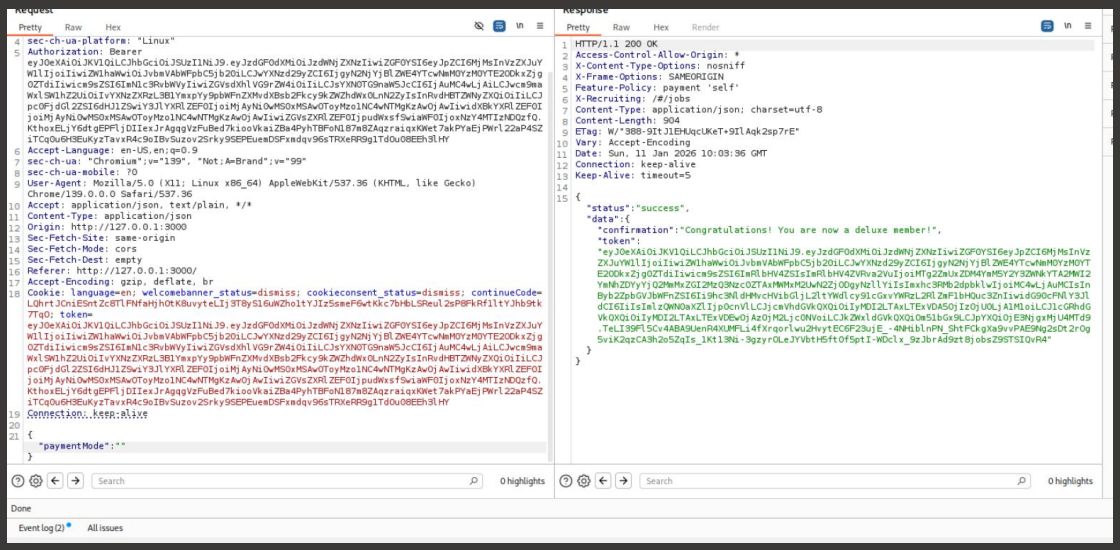
1) Button in the section pay using wallet is not available. That suppose to be fixed through the development tool



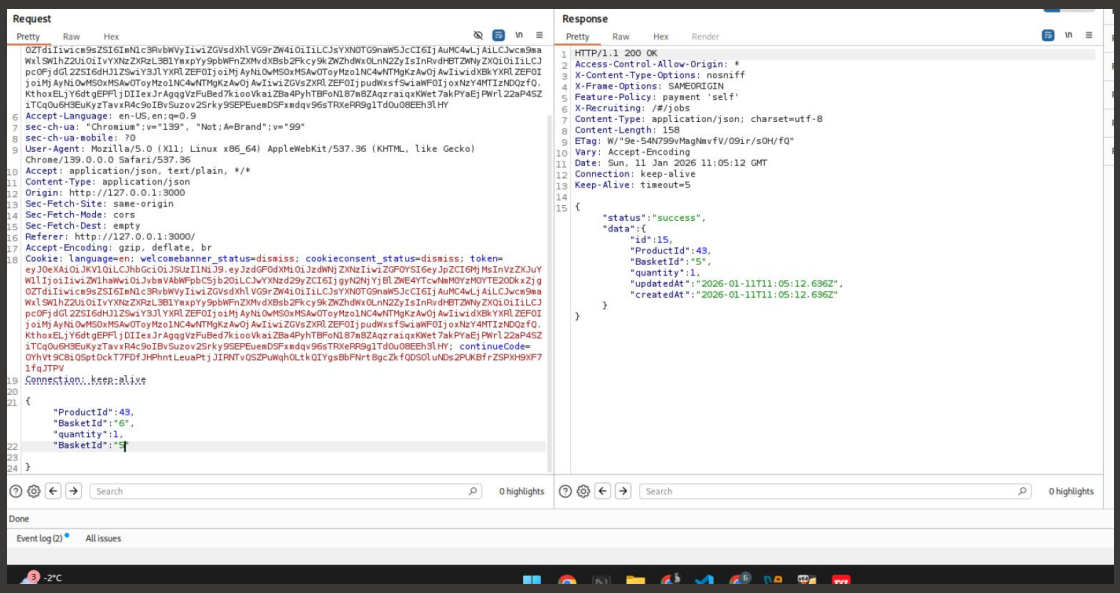
2) Using the inspector helps to find attribute disable=”true” and delete it for making button active



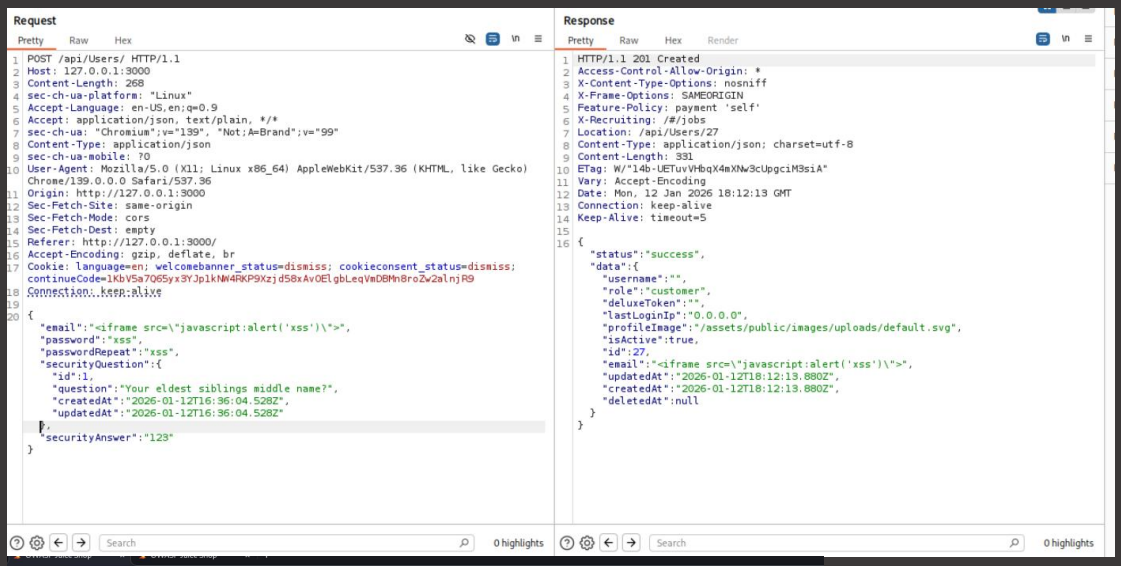
3) Next step it's intercepting of POST request, while clicking on the button "pay with wallet"
4) Using repeator in the BurpSuit to modify request by deleting the value of attribute "paymentMode".
5) Send modyfied request and off the intercepting

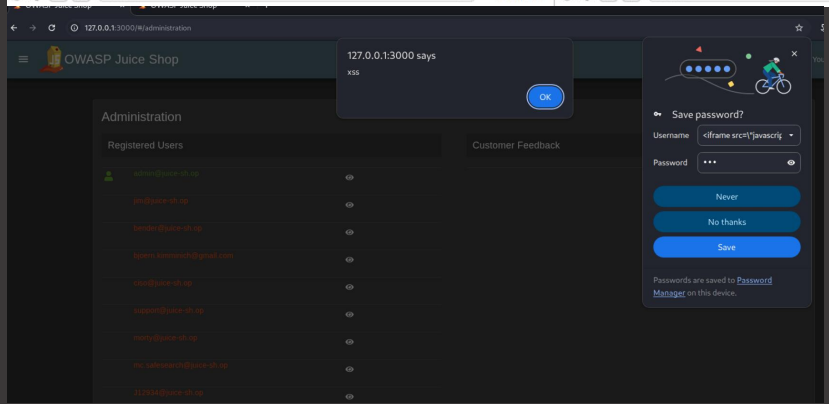


20. Put the product in another user's basket.
there are some steps for achieving this purpose
1) Intercept the POST request via the BurpSuite tool. It suppose to have such parameters: ProductId, BasketId, Quantity
2) Modify it by adding extra parameter BasketId. It is going to be another user's basket
3) Important: it has to be number ID less, than original BasketID
4) Send the modifyied request through Repeater



21. Perform a persisted XSS attack bypassing a client-side security mechanism
This can be implemented by following next steps:
- Intercept POST request that has link api/user via registration a new user
- change next values:
1) Email: {"email": "<iframe src=\"javascript:alert(`xss)\">",
2) Password: {"password": "xss"}`
- send modifyied request through repeator
- login to the site as an admin
- follow the link that leads do /administration
- check accounts that are registrated





22. Perform a persisted XSS attack through an HTTP header
For this task next steps are important:
- find the link /rest/saveLoginIP
- add the header True-Client-IP with value <iframe src=”javascript:alert(‘xss’)”> (header X-Forwarded-For was not succesful)