Ethical Hacking and Penetration Testing

Assignment 4

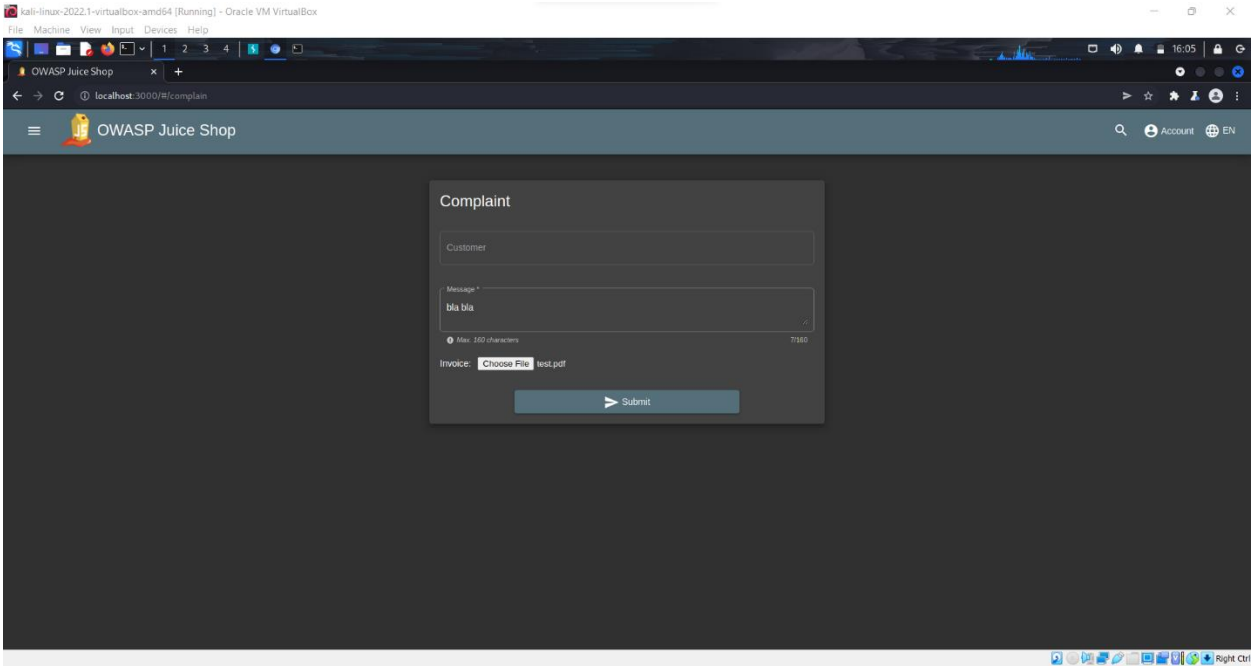
Web Exploitation (Juice-Shop)

Name: John Ehab

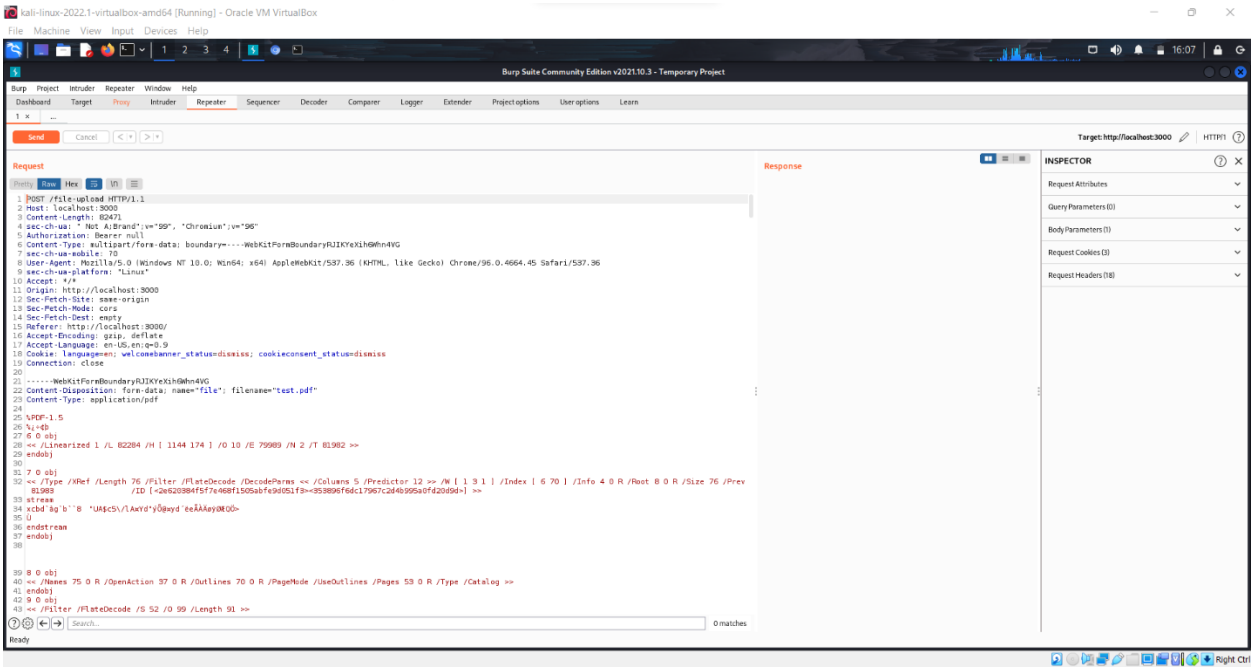
ID: 100-2096

Attack number 1

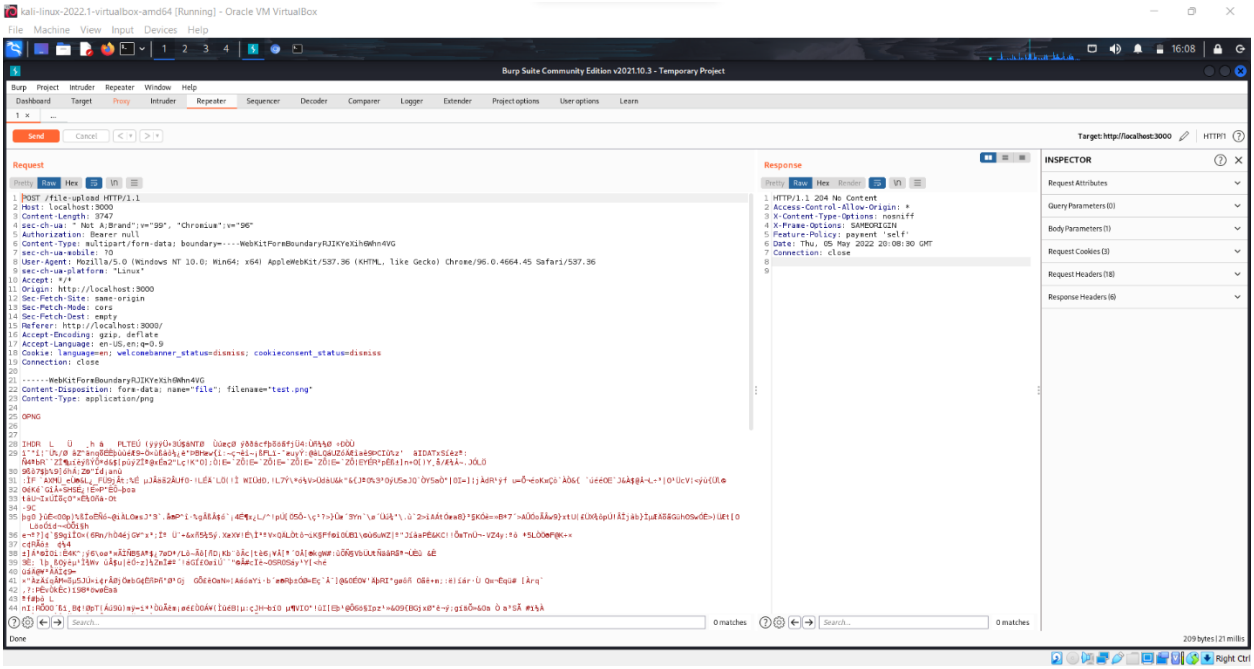
1- I'll open juiceshop and go to the complaint webpage, write any message and upload any .pdf file.

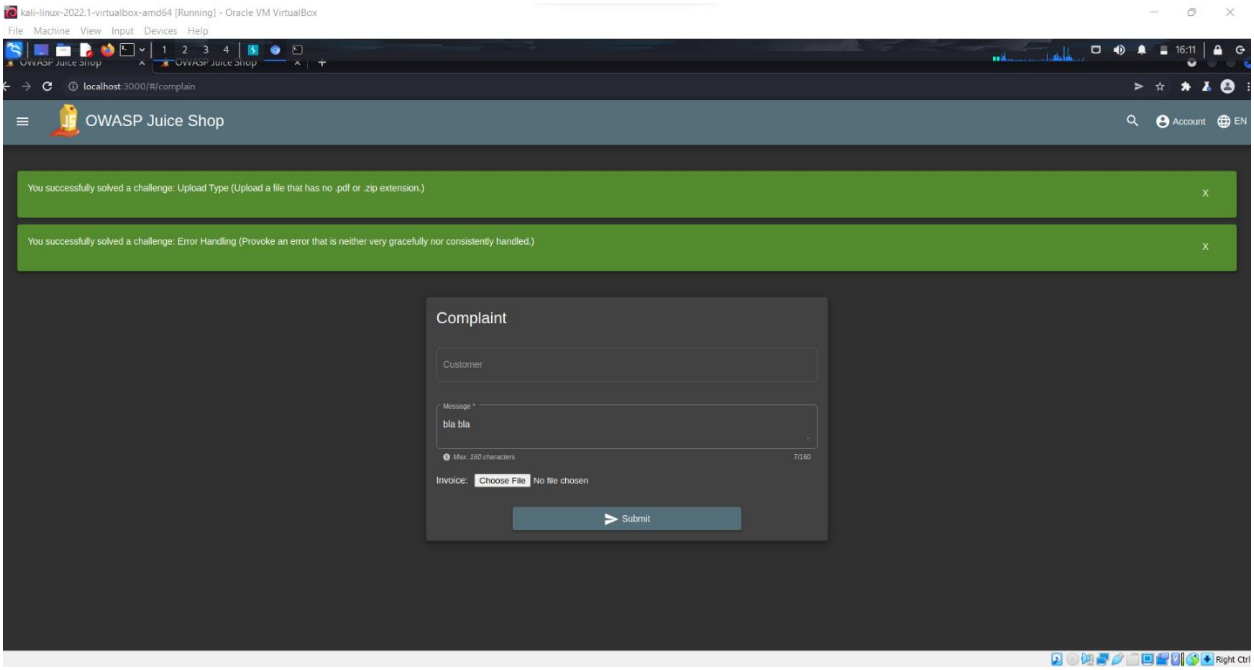


2- Then press submit and intercept the post request using burpsuite and send it to the repeater to manipulate it.



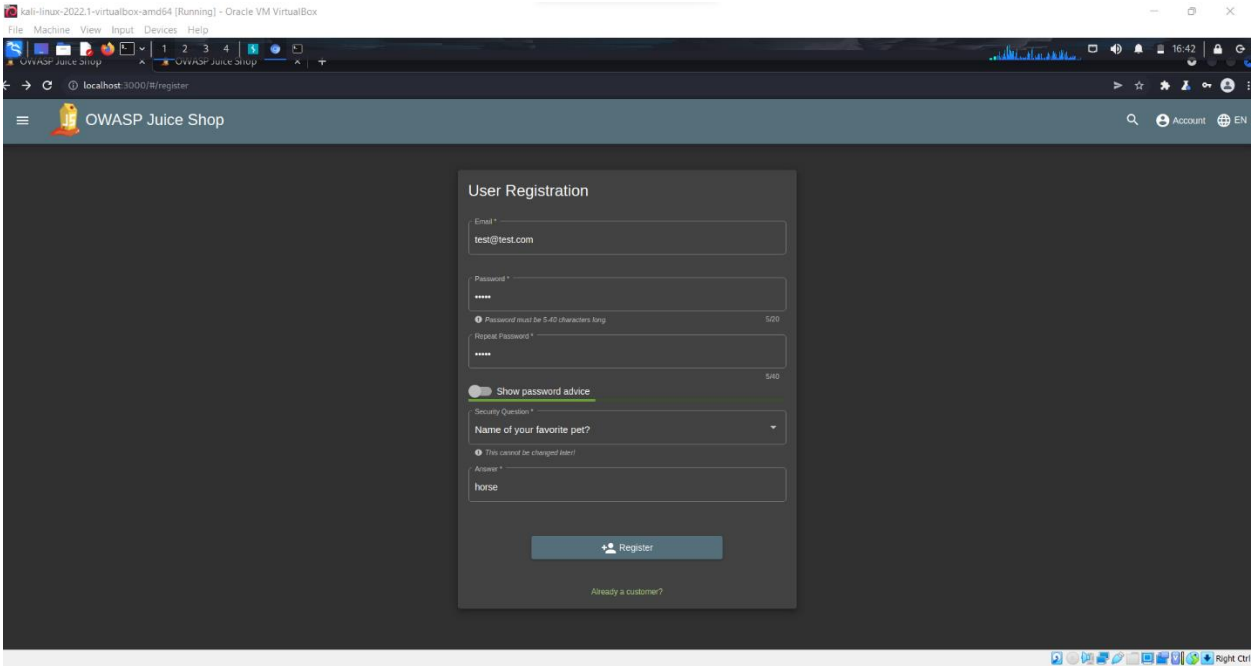
3- Then I opened the file I want to upload instead, in this case it's a photo with .png extension, with notepad. Then copy paste it to the data section in the post req instead of the data of the.pdf file. And changed the filename to test.png instead of test.pdf. And changed the content-type to application/png instead of application/pdf. Then I sent the request uploading the file successfully.



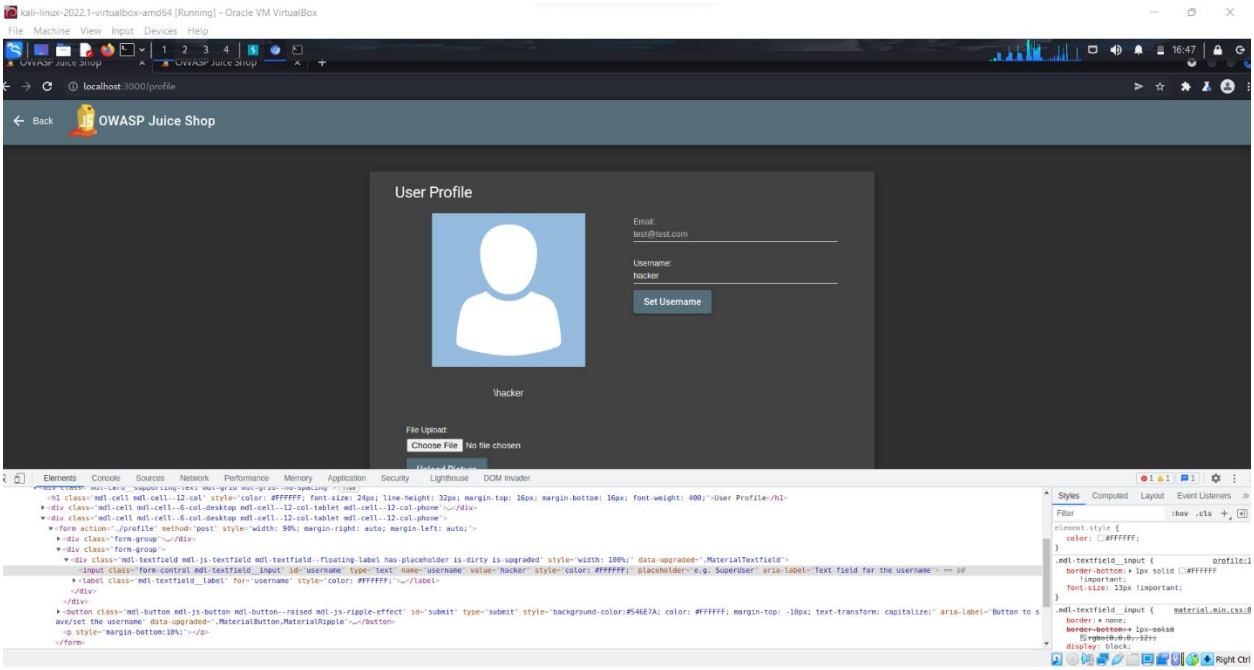


Attack number 2

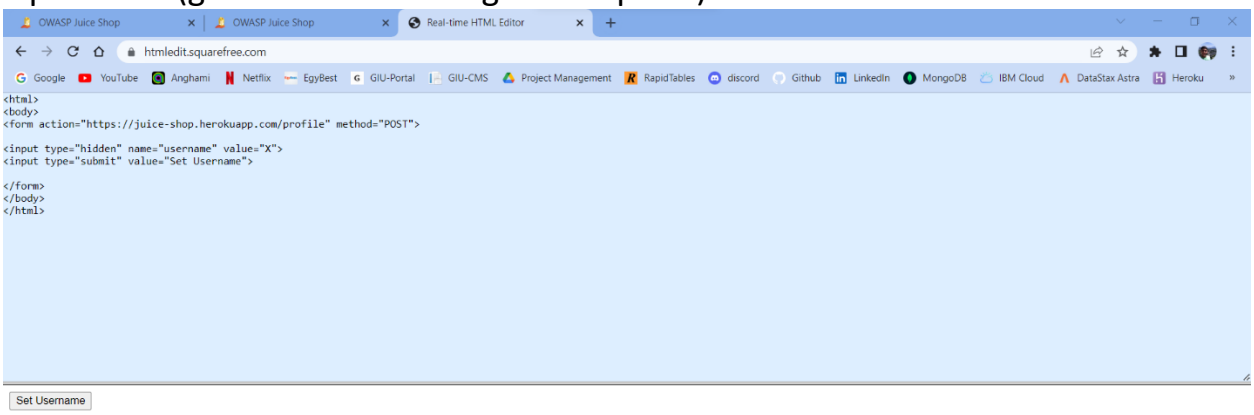
1- I'll go to the accounts login webpage in juiceshop, then I'll register as a new customer.

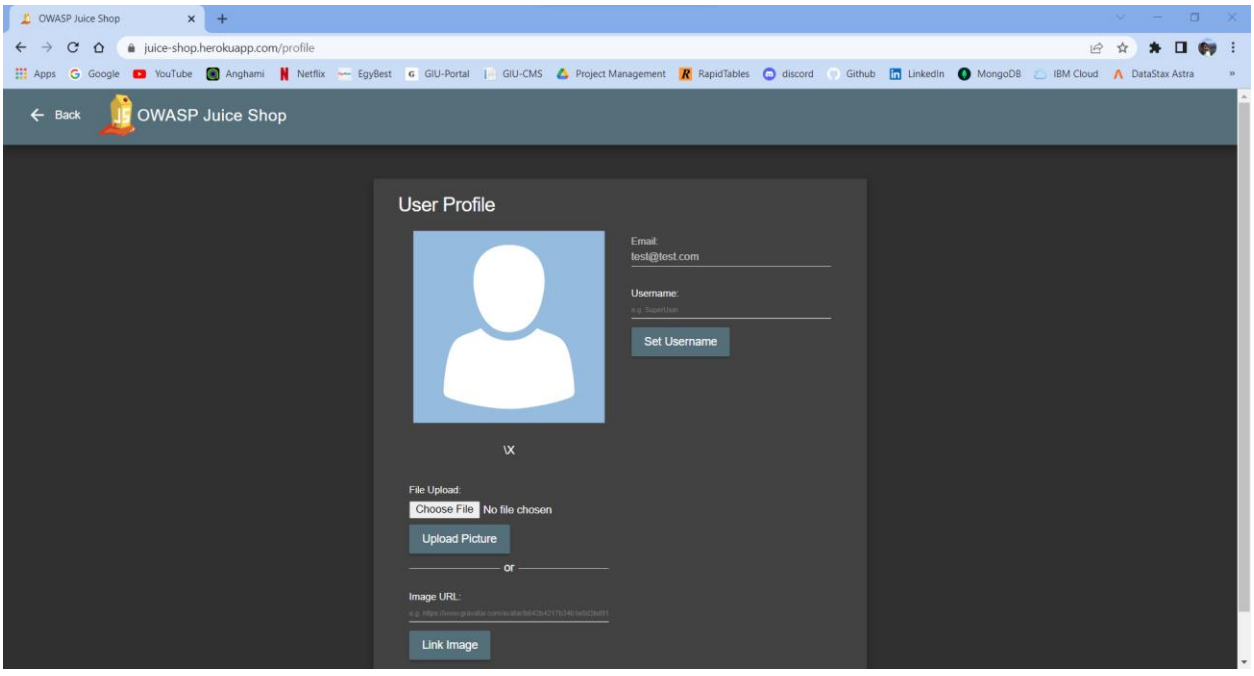


2- Then I'll login with this account to juiceshop, and I'll go to the user profile webpage. Then I'll write any value inside the username input box and I'll inspect the element to notice that the value of the input is changed in the html of the webpage and not sent using a post request.



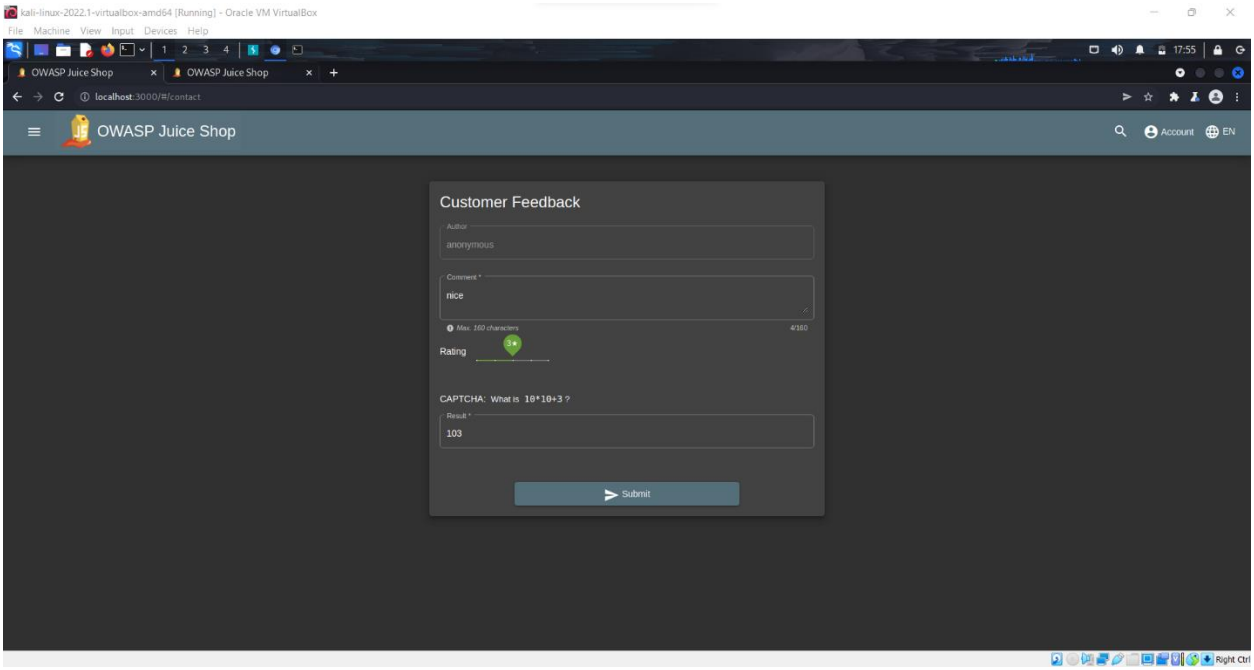
3- So I'll go to any online html editor (another origin) to perform my attack from there and change the username to any value, in my case I'll use squarefree (given in the challenge description) and I'll name him X.



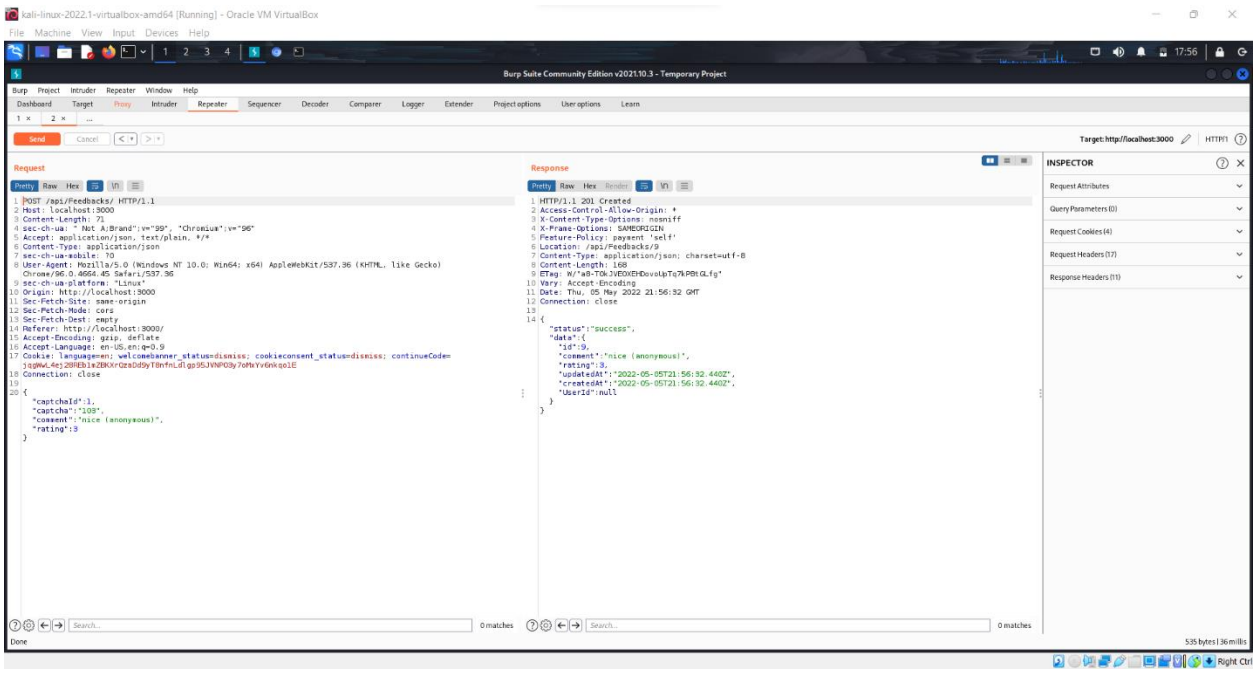


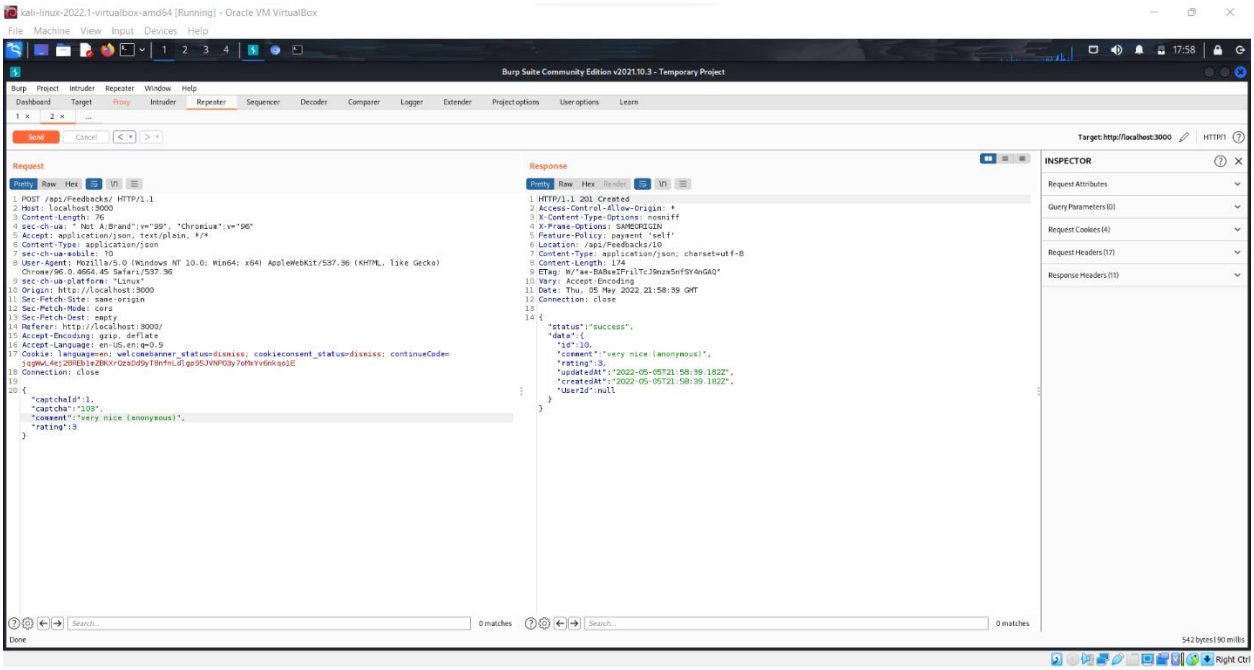
Attack number 3

1- First I'll go to the customer feedback webpage in juiceshop and fill the comment section and the rating with any value I want and solve the given captcha.

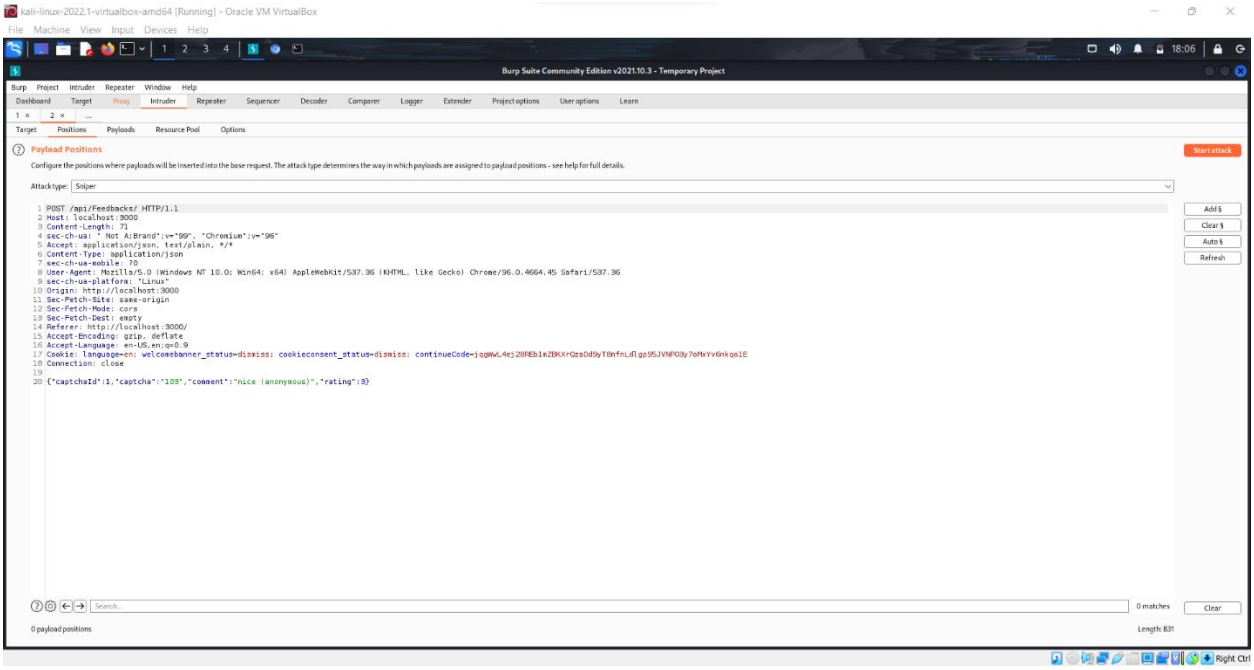


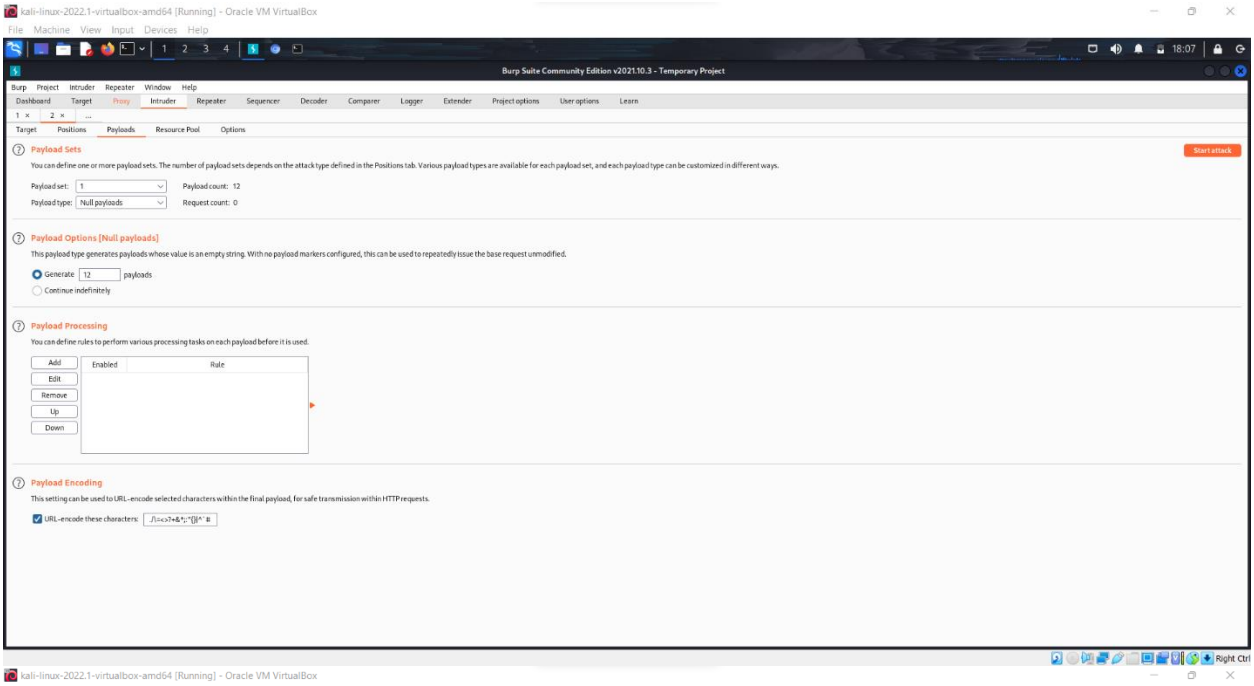
2- I'll intercept this post req and send it to the repeater and try to resend it many times with different comment values discovering that each time the response is still success, so the captchaId doesn't have to be unique for each comment, the req just needs to match the captchaId with the correct corresponding captcha value.

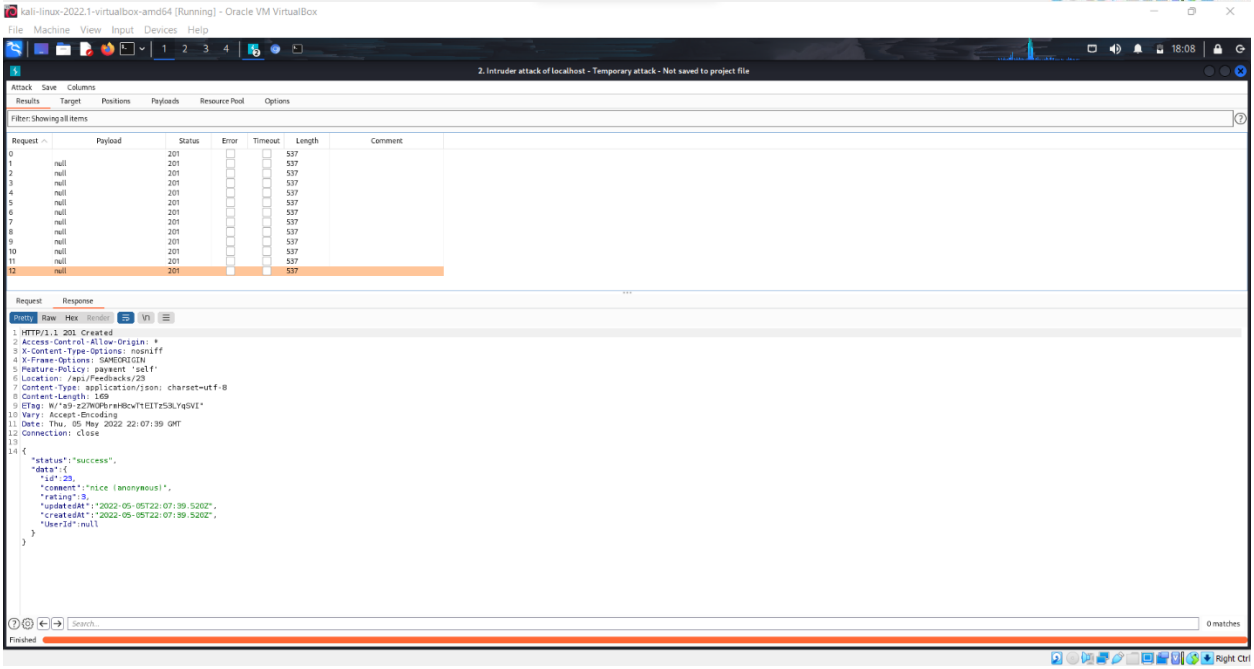


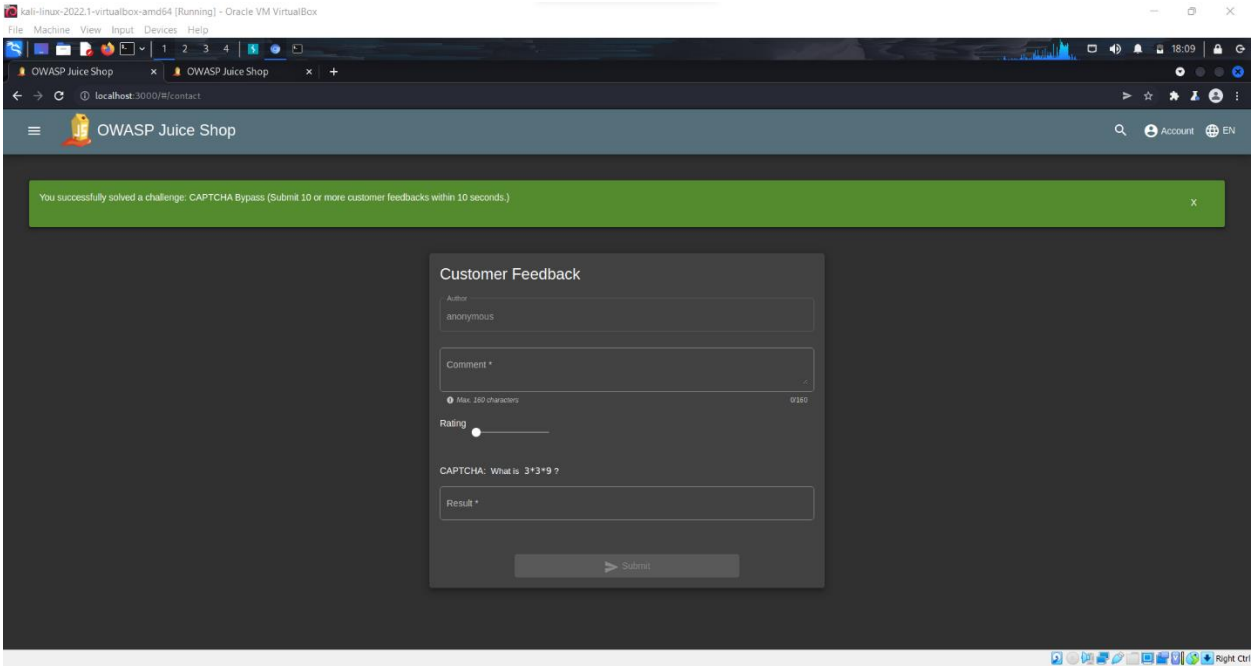


3- Then I'll send the req for the intruder to perform my attack using the sniper attack and choosing more than 10 payloads.



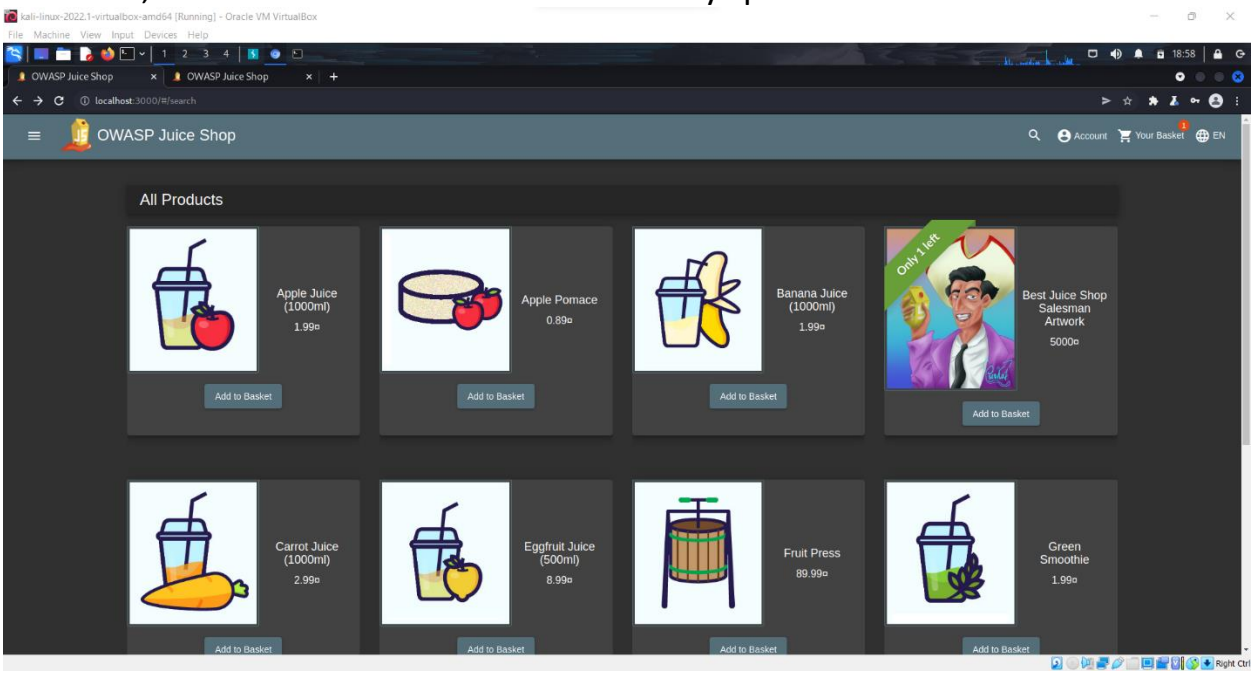

kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Burp Suite Community Edition v2021.10.3 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Target Positions Payloads Resource Pool Options
Payload Sets
Start attack
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set: 1
Payload count: 12
Payload type: Null payloads
Request count: 0
Payload Options [Null payloads]
This payload type generates payloads whose value is an empty string. With no payload markers configured, this can be used to repeatedly issue the base request unmodified.
Generate 12 payloads
Continue indefinitely
Payload Processing
You can define rules to perform various processing tasks on each payload before it is used.
Add
Edit
Remove
Up
Down
Enabled
Rule
Payload Encoding
This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.
URL-encode these characters:
Right Ctrl

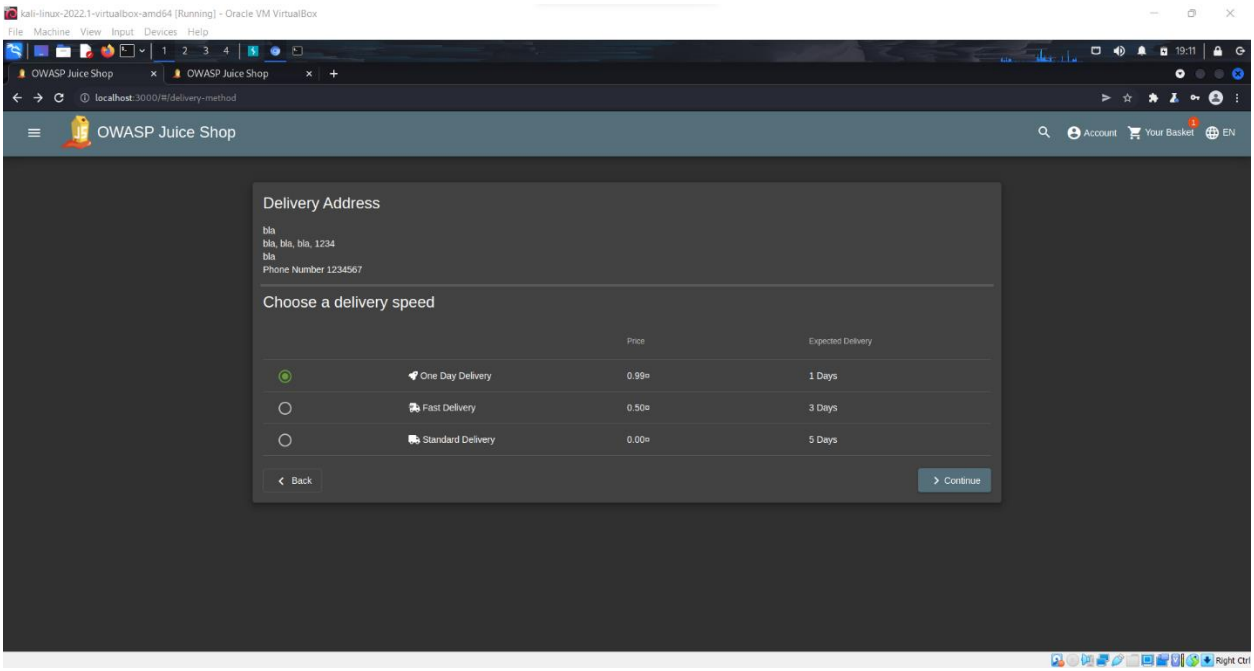

kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
2. Intruder attack of localhost - Temporary attack - Not saved to project file
Attack Save Columns
Results Target Positions Payloads Resource Pool Options
Filter: Showing all items
Request Payload Status Error Timeout Length Comment
0 201 537
1 null 201 537
2 null 201 537
3 null 201 537
4 null 201 537
5 null 201 537
6 null 201 537
7 null 201 537
8 null 201 537
9 null 201 537
10 null 201 537
11 null 201 537
12 null 201 537
Request Response
Pretty Raw Hex Render \n
HTTP/1.1 201 Created
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
Location: /api/Feedbacks/23
Content-Type: application/json; charset=utf-8
Content-Length: 169
ETag: W/"a9-z27KGPbreHBcwTtEITzS3LYqSVI"
Vary: Accept-Encoding
Date: Thu, 05 May 2022 22:07:39 GMT
Connection: close
{
"status":"success",
"data":{
"id":23,
"comment":"nice (anonymous)",
"rating":3,
"updatedAt":"2022-05-05T22:07:39.520Z",
"createdAt":"2022-05-05T22:07:39.520Z",
"UserId":null
}
}
Search...
0 matches
Finished
Right Ctrl


kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
OWASP Juice Shop
OWASP Juice Shop
localhost:3000/#/contact
OWASP Juice Shop
Account
EN
You successfully solved a challenge: CAPTCHA Bypass (Submit 10 or more customer feedbacks within 10 seconds.)
x
Customer Feedback
Author
anonymous
Comment *
Max. 160 characters
0/160
Rating
CAPTCHA: What is 3*3*9 ?
Result *
Submit
Right Ctrl

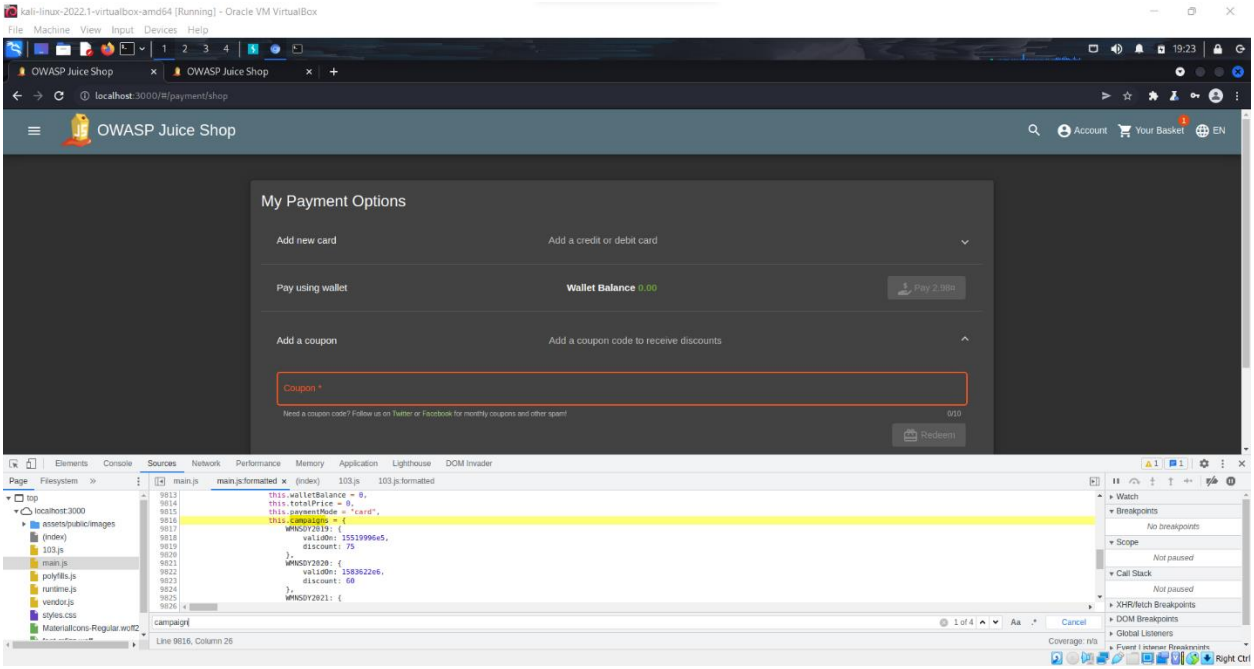
Attack number 4

1- I'll use the account I registered in attack number 2 to log in, I'll choose any product and add it to the basket, then view my basket, then proceed to checkout, then select an address and a delivery speed.

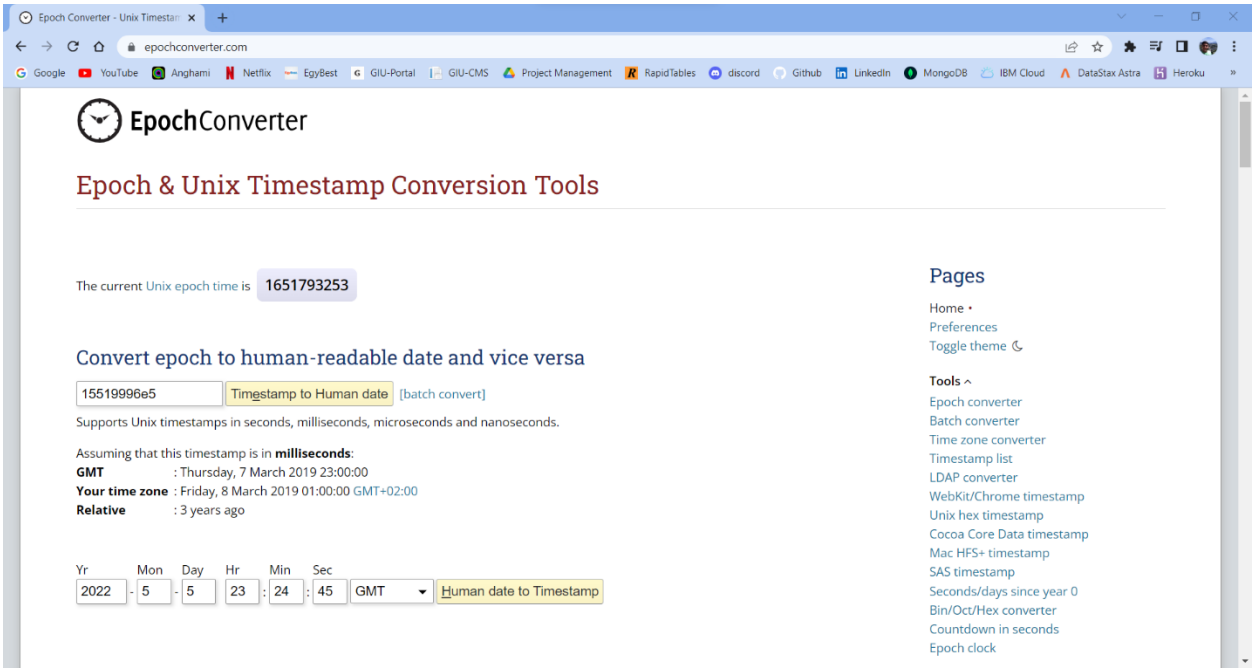




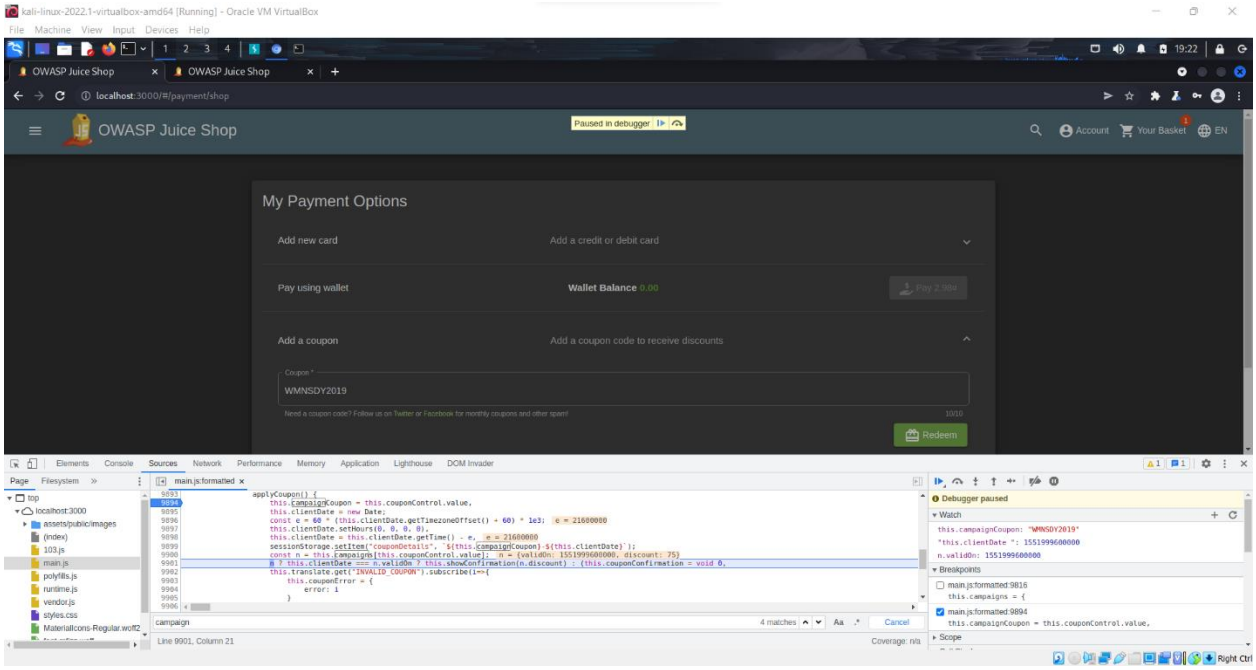
2- In the payment options I'll press on add a coupon, then to know the expired campaign coupon codes I'll open developer tools, then go to the sources tab, and open the "main.js" file, and press on the pretty print button, then search for the word campaign. I'll find each code with the corresponding timestamp for it.

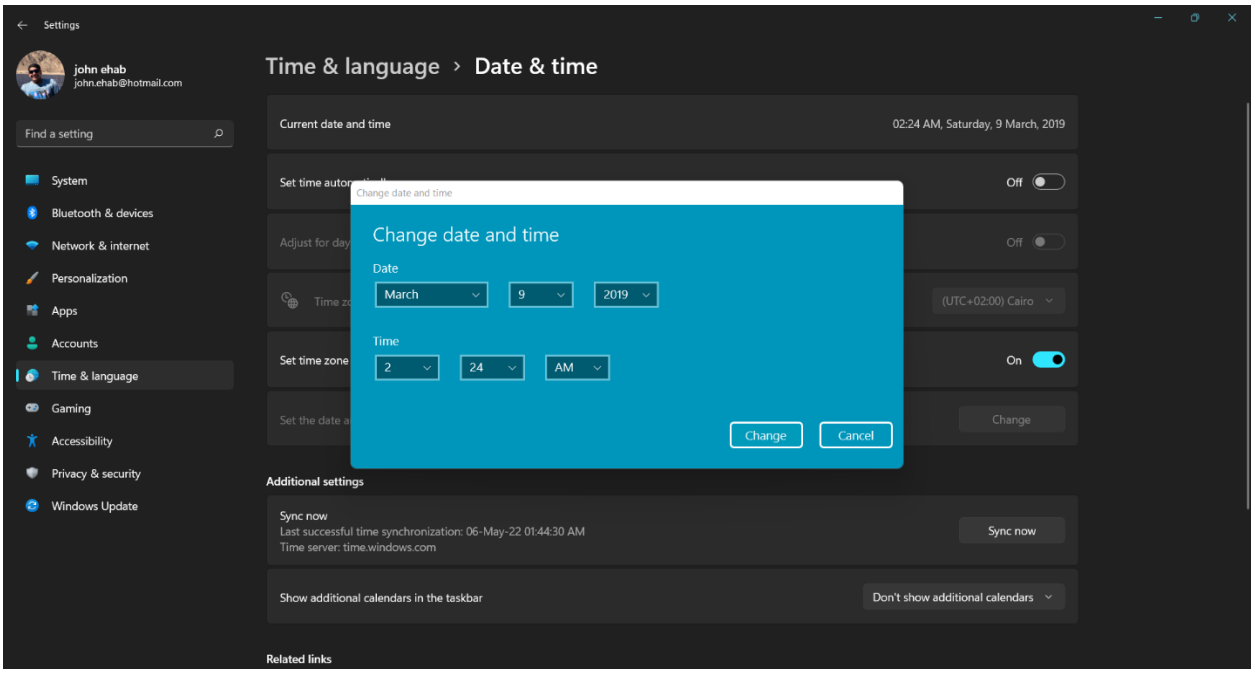


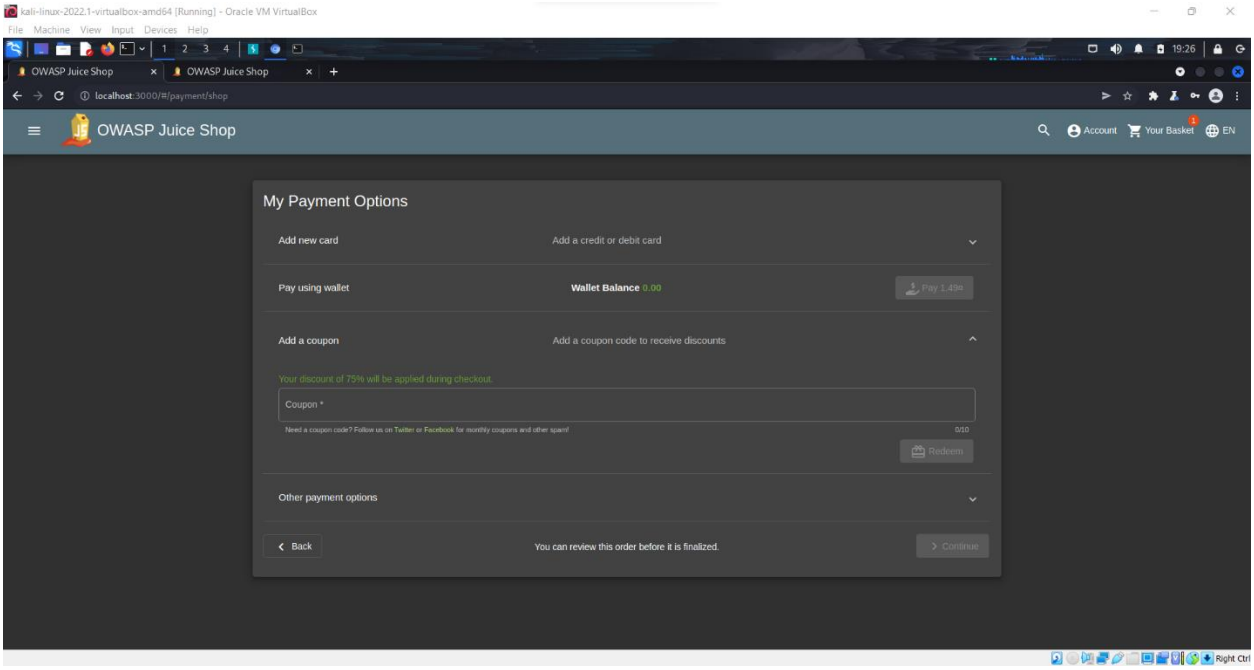
3- I'll select one coupon to work on, let's say the first one, its coupon code is "WMNSDY2019" and its validOn timestamp is "15519996e5". I'll use any online timestamp converter to see what date is it valid on.



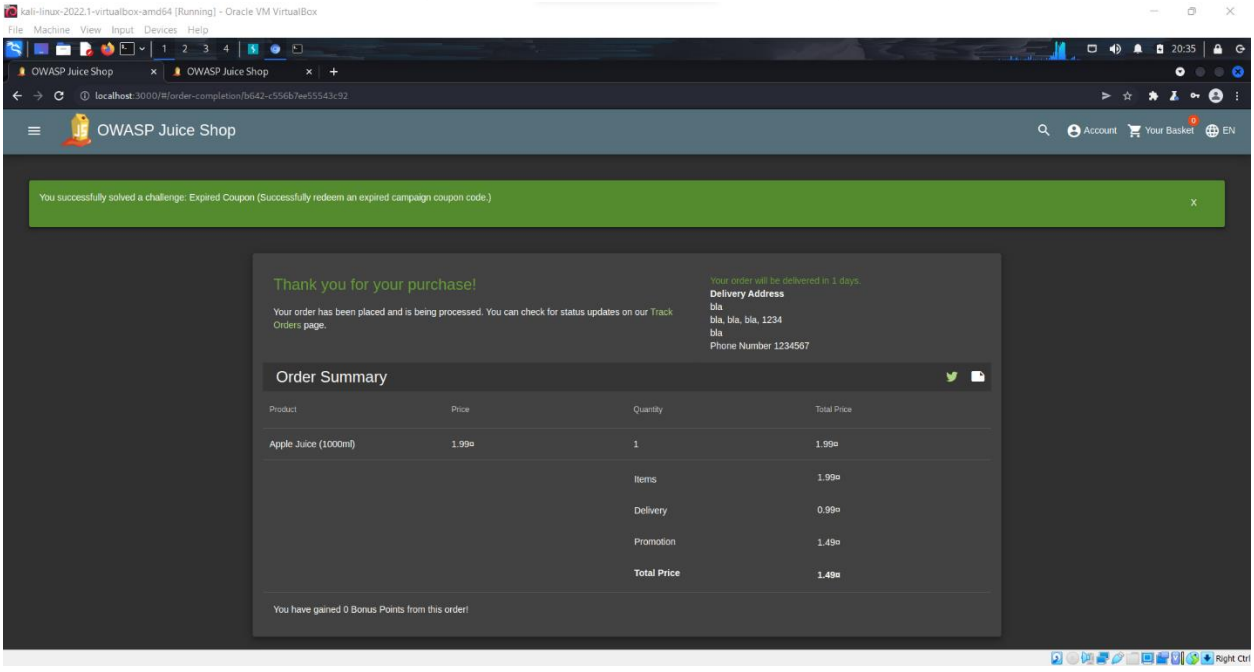
4- Then I'll write the coupon code on the payment options webpage, and I'll change my local machine date and time manually to different dates around 8/3/2019 or 7/3/2019 to see if any date matches with my time zone and then press redeem. The date that was successful with our time zone was 9 March 2019.





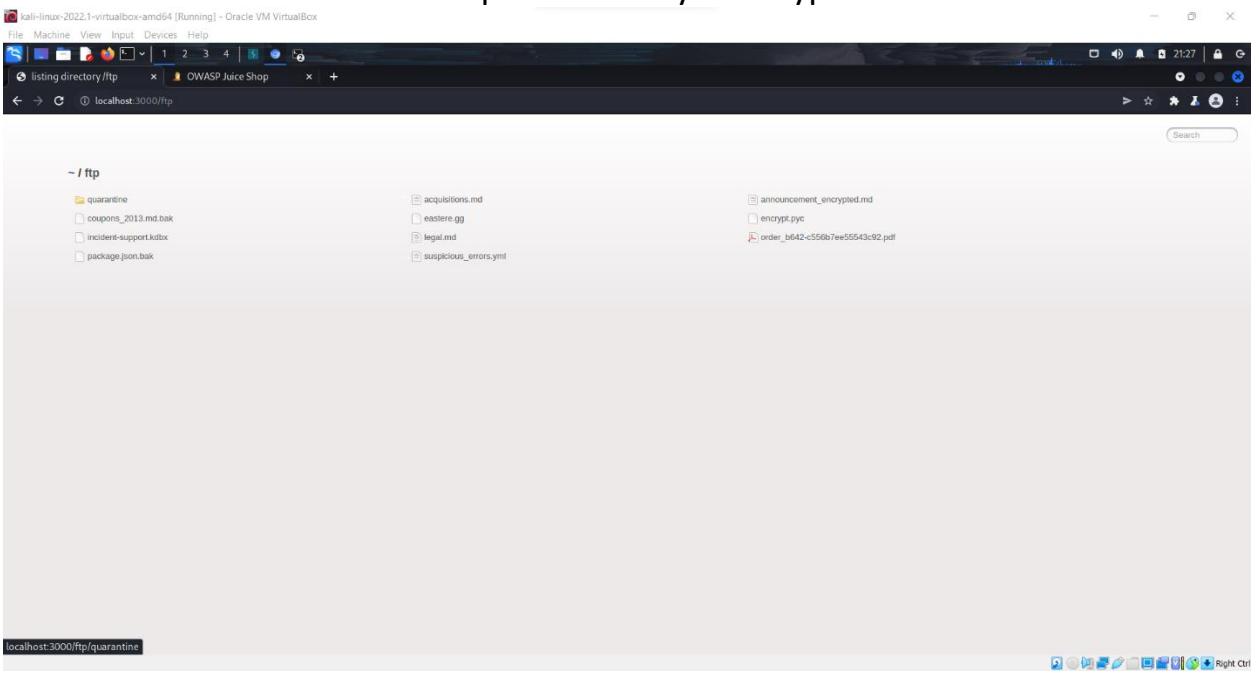


5- Then I'll add any values for a credit card and select it, then press on "place your order and pay" to complete the attack successfully.

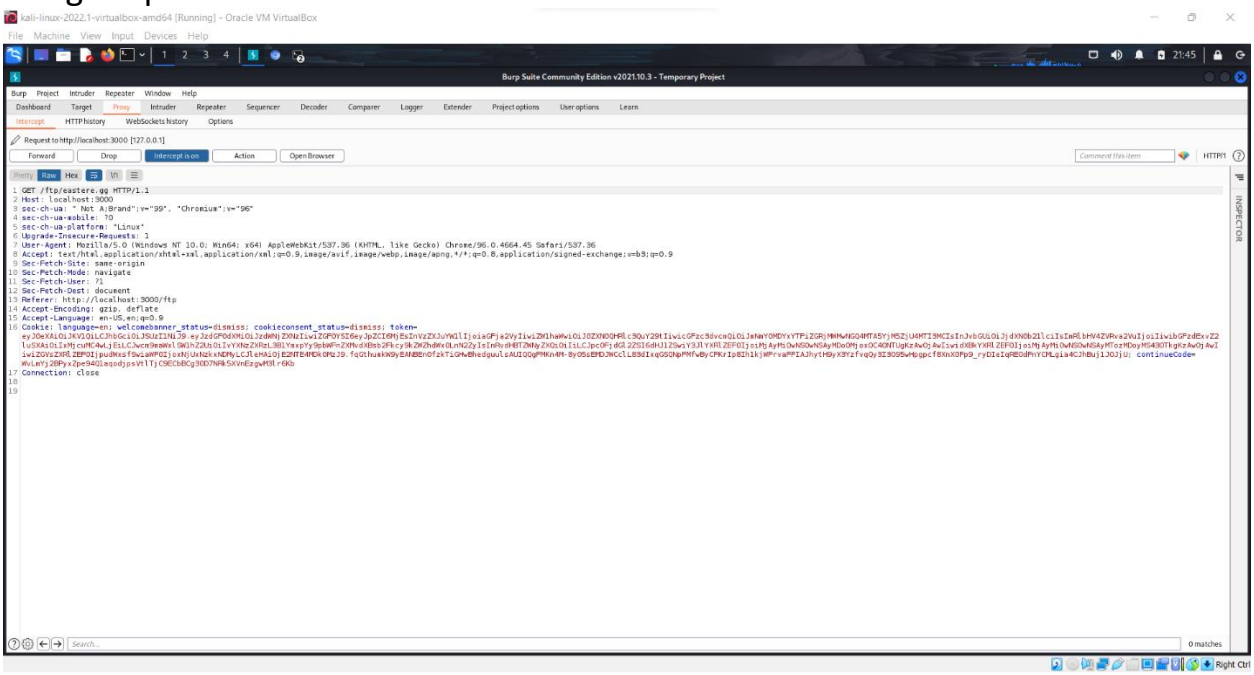


Attack number 5 & 6

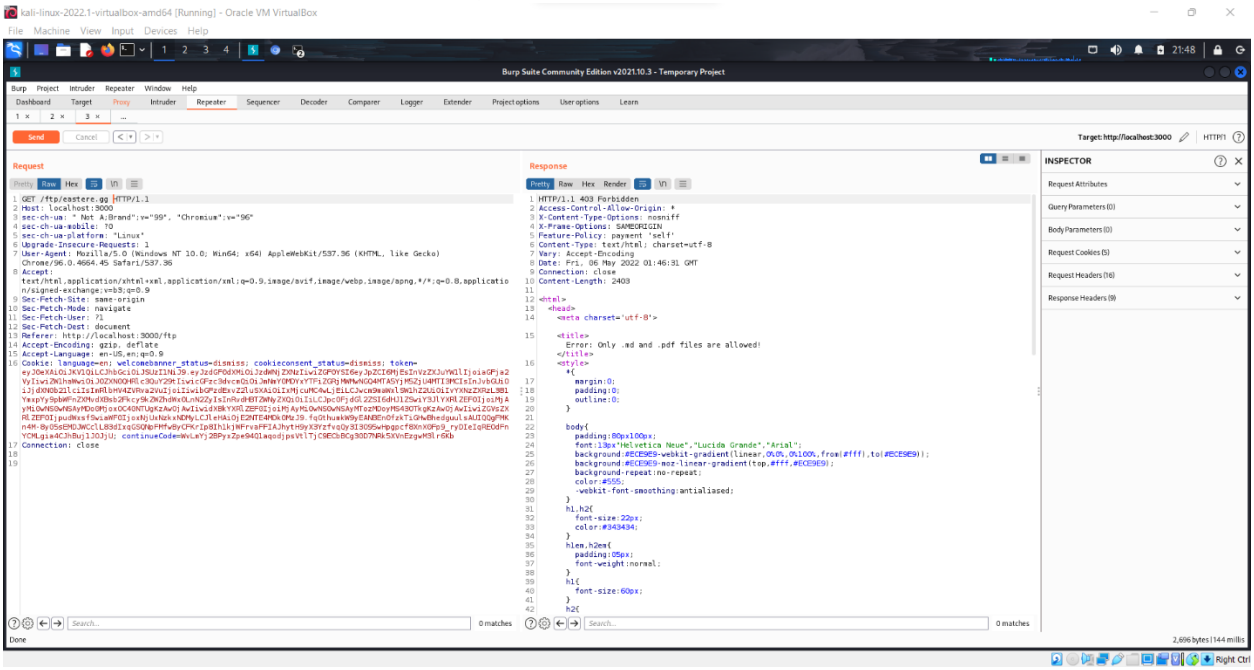
1- Firstly, open localhost:3000/ftp to show the files on the server, I want to transfer the eastere.gg file but the website doesn't allow the transfer of files of this extension so I'll use poison null byte to bypass it.

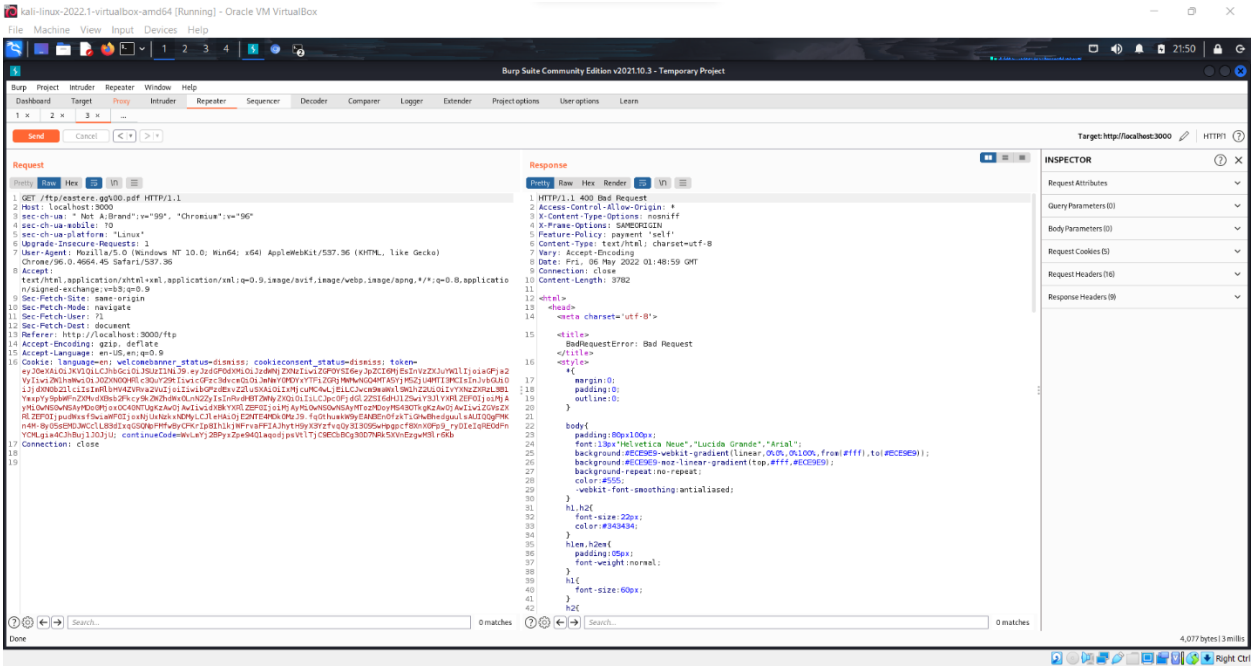


2- I'll press on eastere.gg to try to download it and intercept this get request using burpsuite.

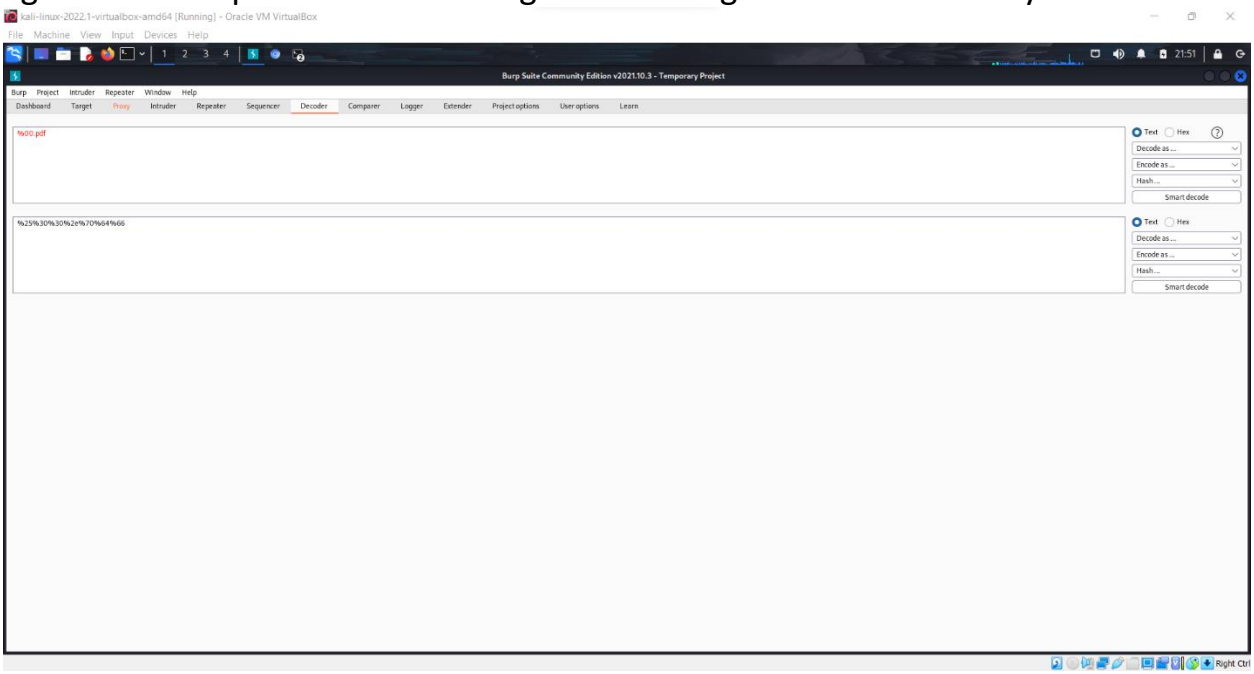


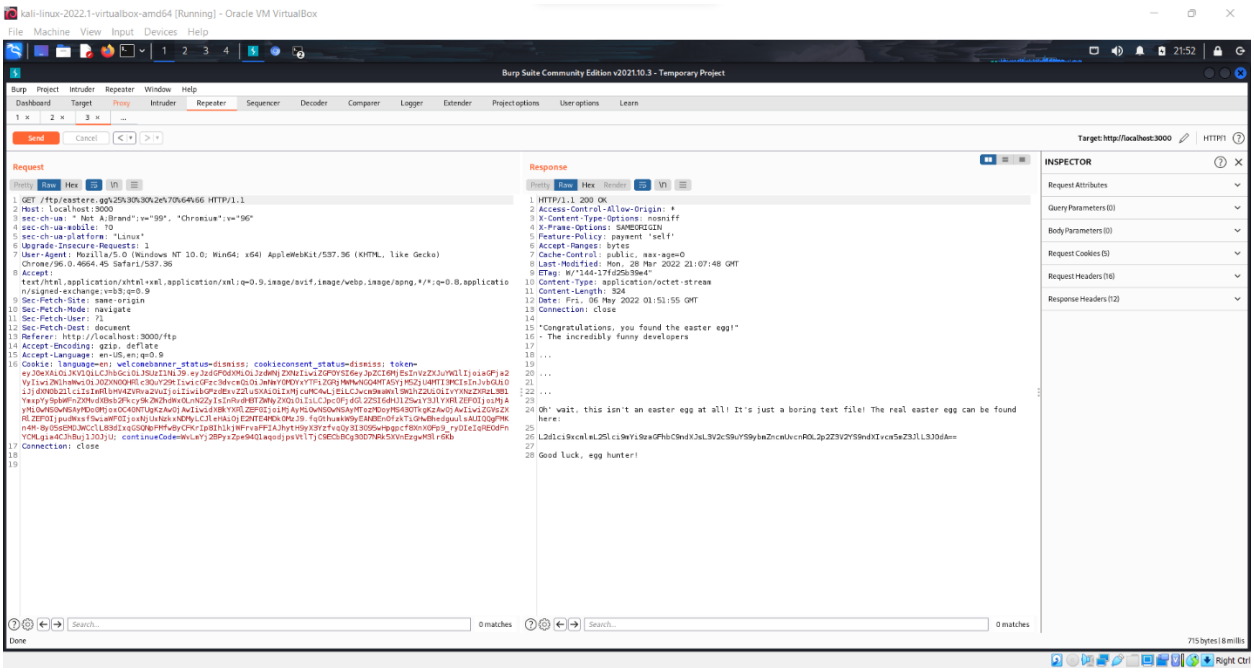
3- I'll send it without changing anything and will get an error 403 cause it's forbidden because of its extension, then I'll inject the null byte "%00" followed by an extension accepted by the server ".pdf" but also I'll get an error 400 because it's a bad req.



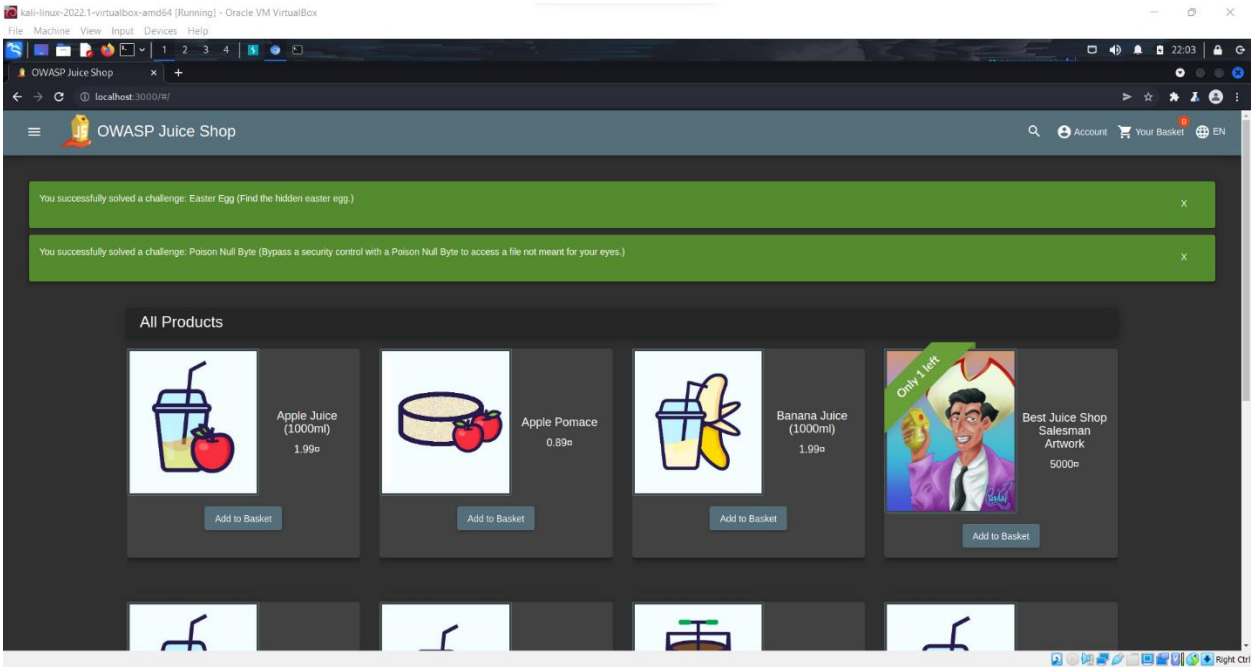


4- To inject it in a form of a request I'll use burpsuite decoder to encode %00.pdf as URL, then I'll concatenate the encoded text instead and send it again on the repeater downloading and showing the file successfully.



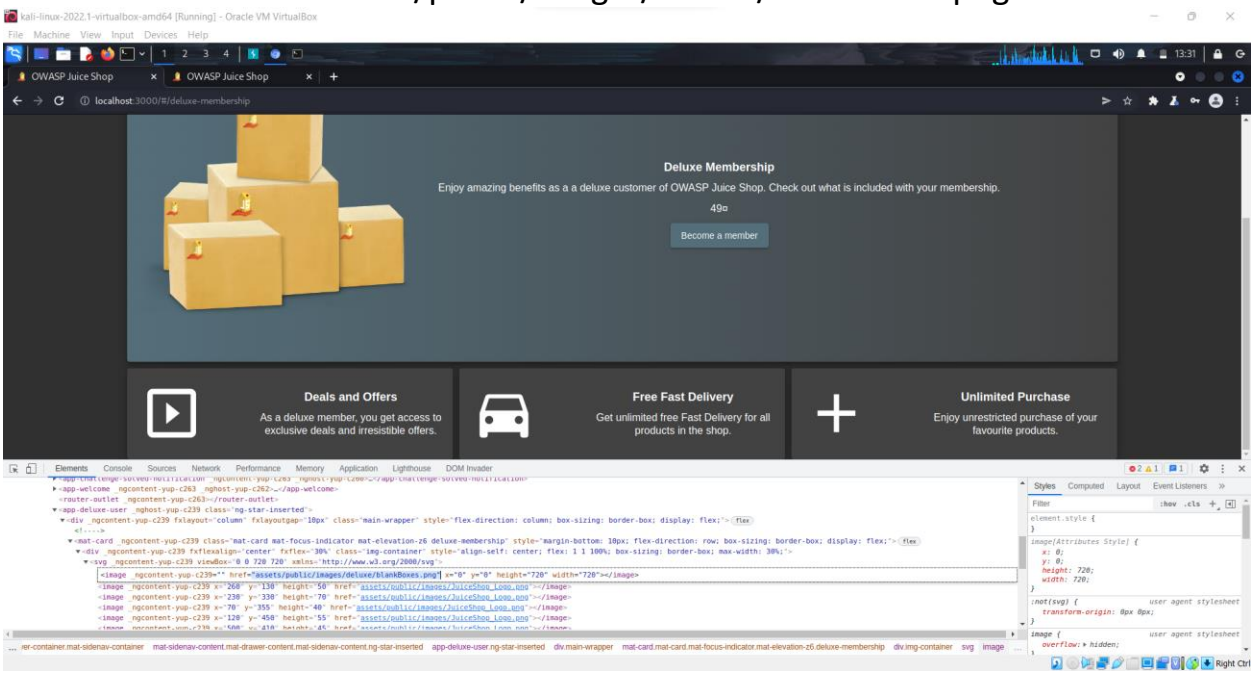


5- Then copy paste it back to the proxy and forward the req to complete the attack.

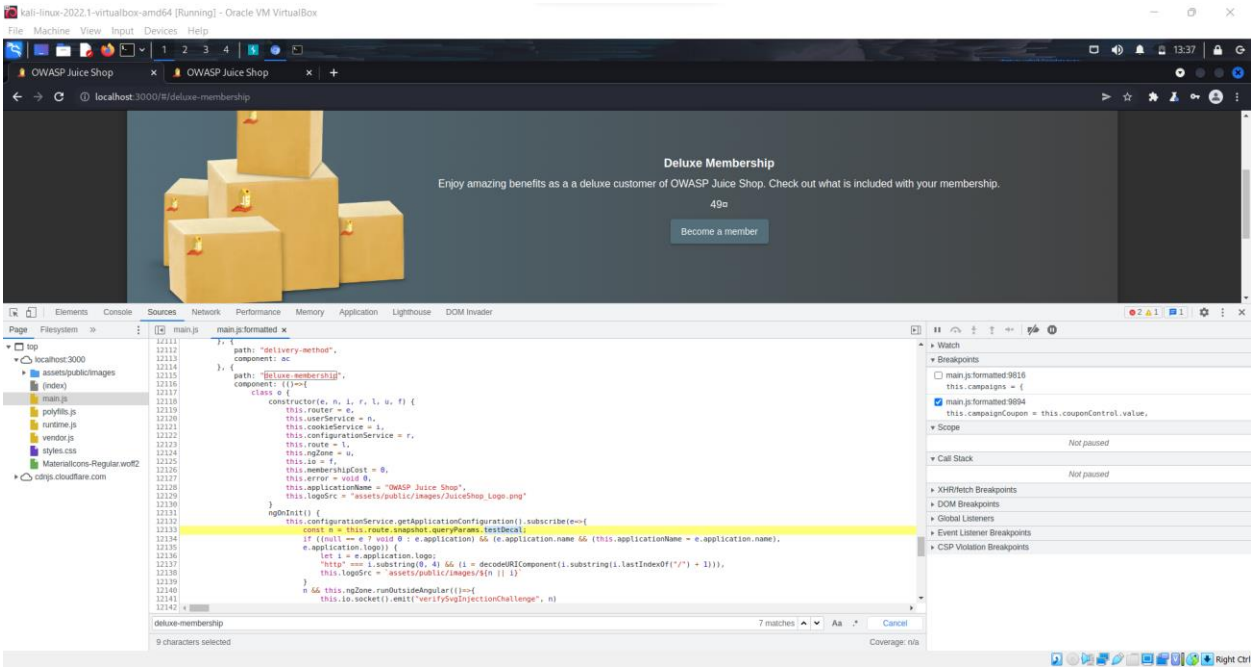


Attack number 7

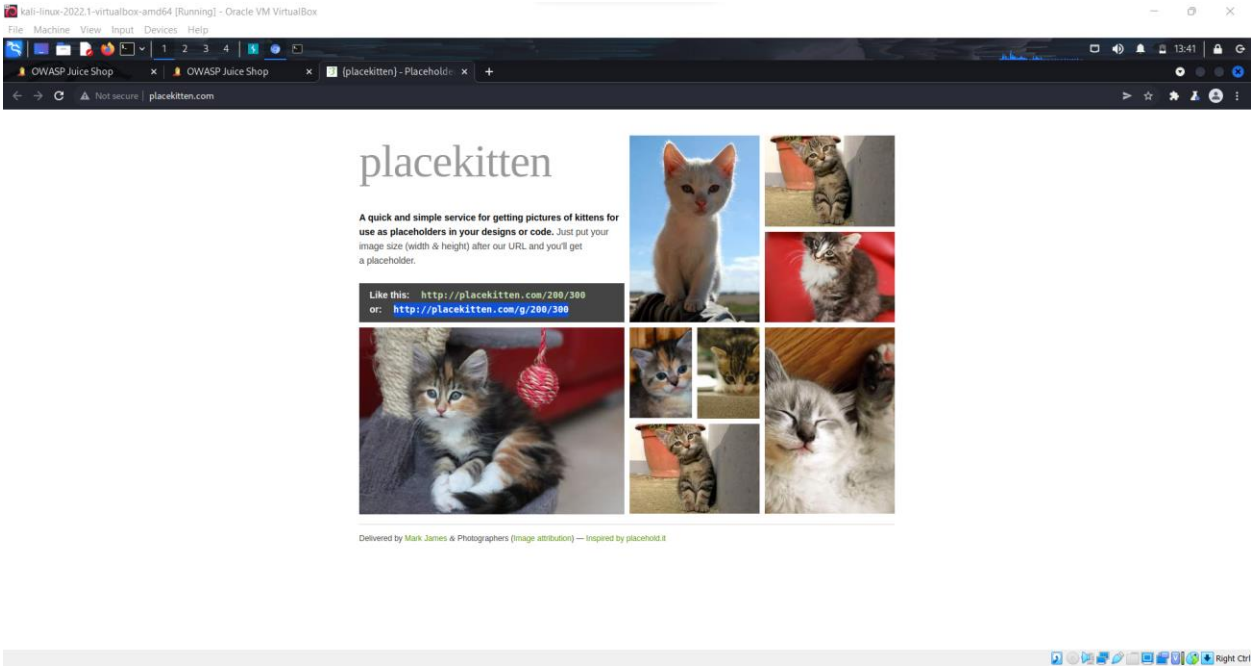
1- First I'll go to the deluxe-membership webpage to find an img for the delivery boxes on the website, then will inspect and search for the image till I find that its href is "assets/public/images/deluxe/blankBoxes.png"



2- Then I'll open the sources tab then the main.js formatted file then I'll search for "deluxe-membership" till I find the parameter's name which is "testDecal"



3- I'll get the url of the kittens that need to be placed from the challenge description. http://placekitten.com/g/200/300



4- Then I'll start trying to change the value of the testDecal parameter to try to embed or redirect to the placekitten url. I'll then notice the behaviours of the different parameter values and do my analysis till the challenge is successfully completed using the url:
http://localhost:3000/#/deluxe-membership?testDecal=../../../redirect?to=http://placekitten.com/g/400/500?x=http://github.com/juice-shop/juice-shop