OWASP Juice Shop

Open for Business!

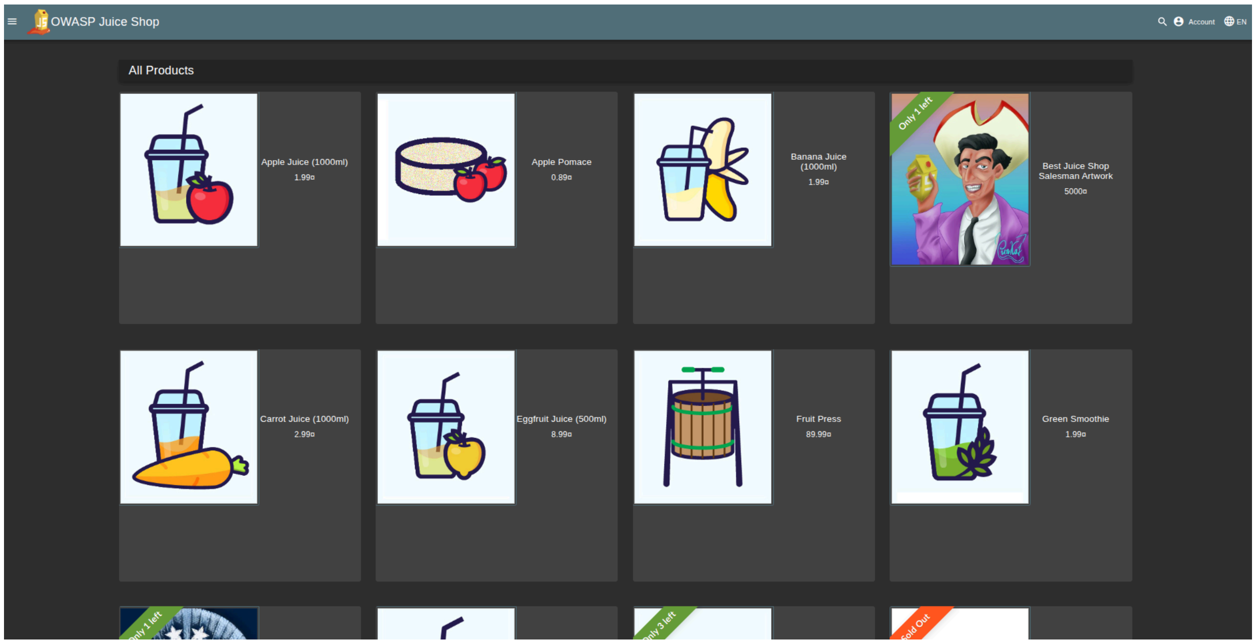
Within this room, we will look at OWASP's TOP 10 vulnerabilities in web applications. You will find these in all types of web applications. But for today we will be looking at OWASP's own creation, Juice Shop!

Juice Shop is a large application so we will not be covering every topic from the top 10.

We will, however, cover the following topics which we recommend you take a look at as you progress through this room.

- Injection
- Broken Authentication
- Sensitive Data Exposure
- Broken Access Control
- Cross-Site Scripting XSS

Lets Go on an Adventure!

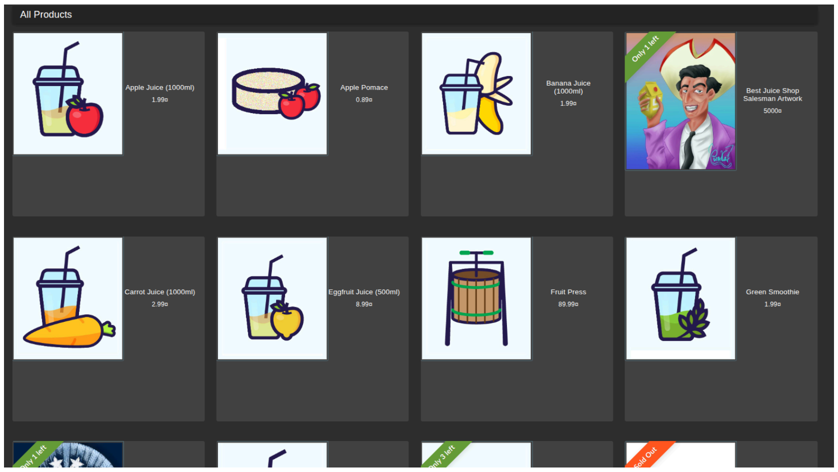


Before we get into the actual hacking part, it's good to have a look around. In Burp, set the Intercept mode to off and then browse around the site. This allows Burp to log different requests from the server that may be helpful later.

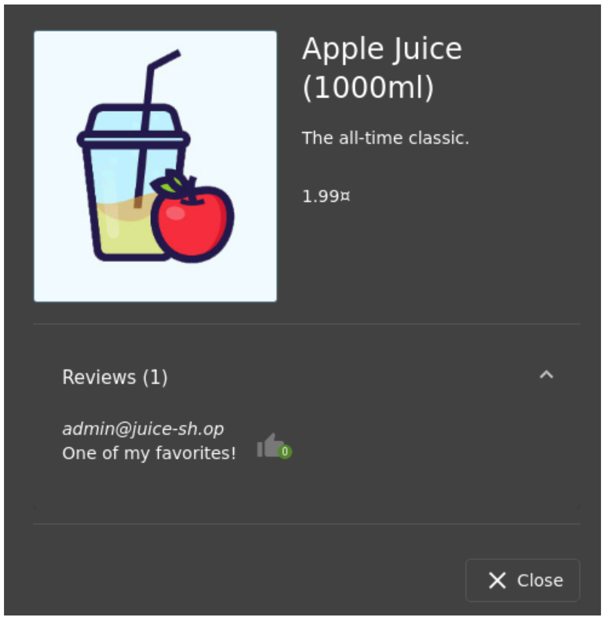
This is called walking through the application, which is also a form of reconnaissance!
****

Answer the questions below:

What's the Administrator's email address?



The reviews show each user's email address. Which, by clicking on the Apple Juice product, shows us the Admin email!



Apple Juice (1000ml)

The all-time classic.

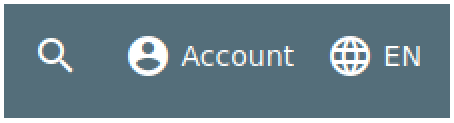
1.99¤

Reviews (1)

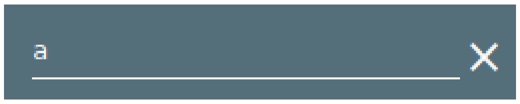
admin@juice-sh.op
One of my favorites!

Answer: admin@juice-sh.op

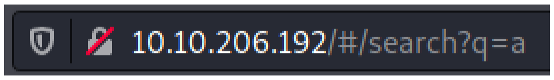
What parameter is used for searching?



Click on the magnifying glass in the top right of the application will pop out a search bar.



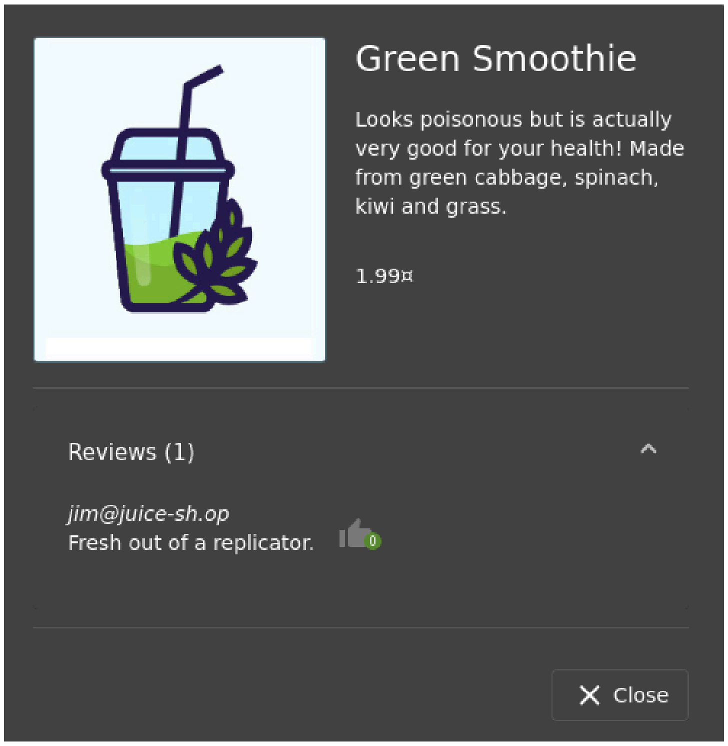
We can then input some text and by pressing Enter will search for the text which was just input.
Now pay attention to the URL which will now update with the text we just entered.



10.10.206.192/#/search?q=a

We can now see the search parameter after the /#/search? the letter q
Answer: q

What show does Jim reference in his review?
Jim did a review on the Green Smoothie product. We can see that he mentions a replicator.



Green Smoothie

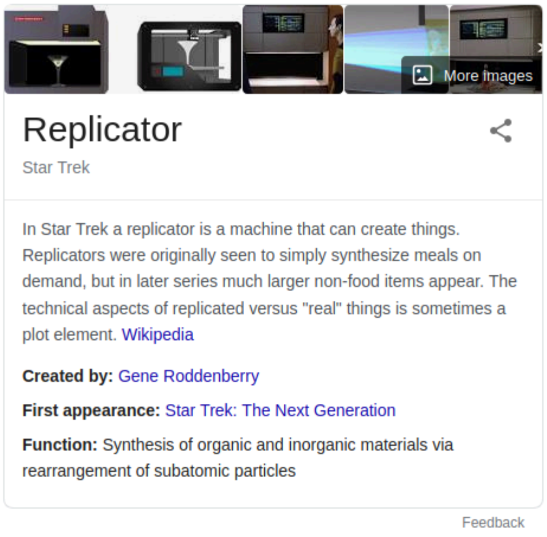
Looks poisonous but is actually very good for your health! Made from green cabbage, spinach, kiwi and grass.

1.99¤

Reviews (1)

jim@juice-sh.op
Fresh out of a replicator.

If we google "replicator" we will get the results indicating that it is from a TV show called Star Trek



Replicator

Star Trek

In Star Trek a replicator is a machine that can create things. Replicators were originally seen to simply synthesize meals on demand, but in later series much larger non-food items appear. The technical aspects of replicated versus "real" things is sometimes a plot element. Wikipedia

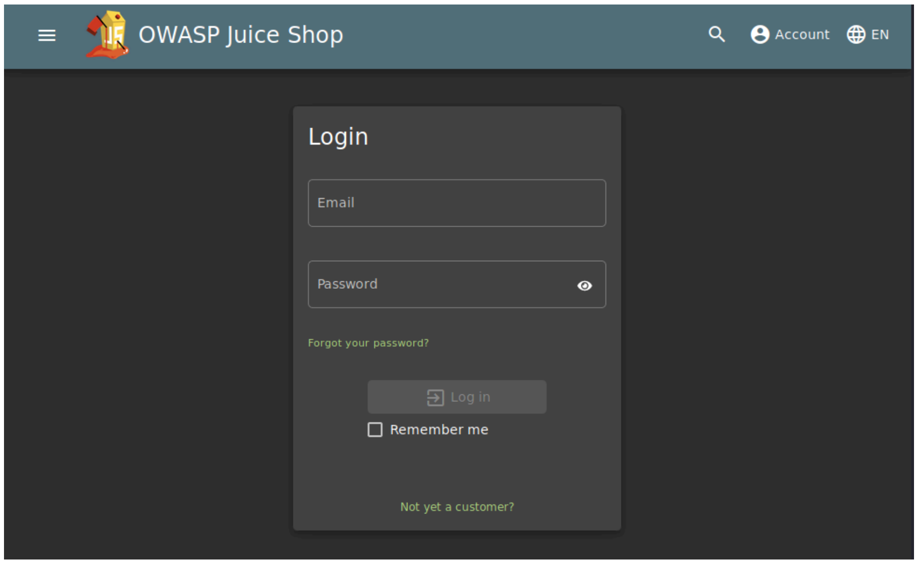
Created by: Gene Roddenberry

First appearance: Star Trek: The Next Generation

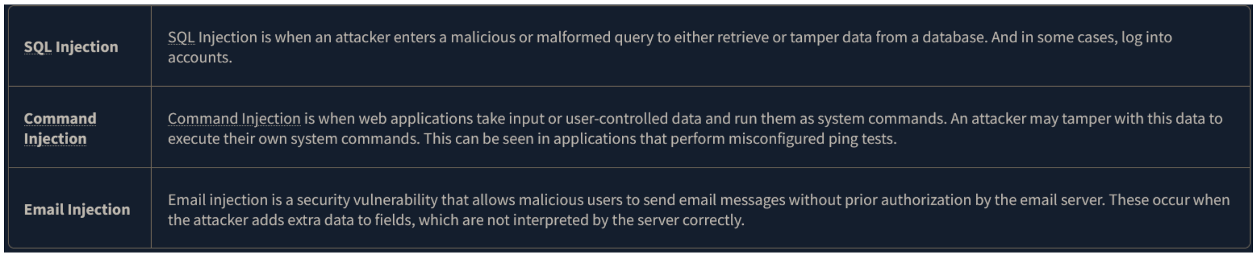
Function: Synthesis of organic and inorganic materials via rearrangement of subatomic particles

Answer: Star Trek
****

Inject the Juice



This task will be focusing on injection vulnerabilities. Injection vulnerabilities are quite dangerous to a company as they can potentially cause downtime and/or loss of data. Identifying injection points within a web application is usually quite simple, as most of them will return an error. There are many types of injection attacks, some of them are:



SQL Injection	SQL Injection is when an attacker enters a malicious or malformed query to either retrieve or tamper data from a database. And in some cases, log into accounts.
Command Injection	Command Injection is when web applications take input or user-controlled data and run them as system commands. An attacker may tamper with this data to execute their own system commands. This can be seen in applications that perform misconfigured ping tests.
Email Injection	Email injection is a security vulnerability that allows malicious users to send email messages without prior authorization by the email server. These occur when the attacker adds extra data to fields, which are not interpreted by the server correctly.

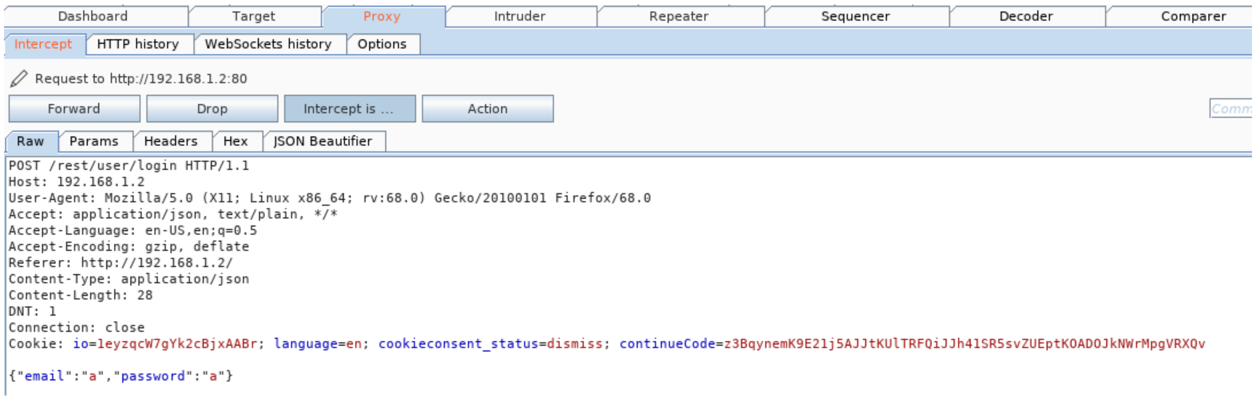
****

Answer the questions below:

Log into the administrator account!

After we navigate to the login page, enter some data into the email and password fields.

Before clicking submit, make sure Intercept mode is on. This will allow us to see the data being sent to the server!

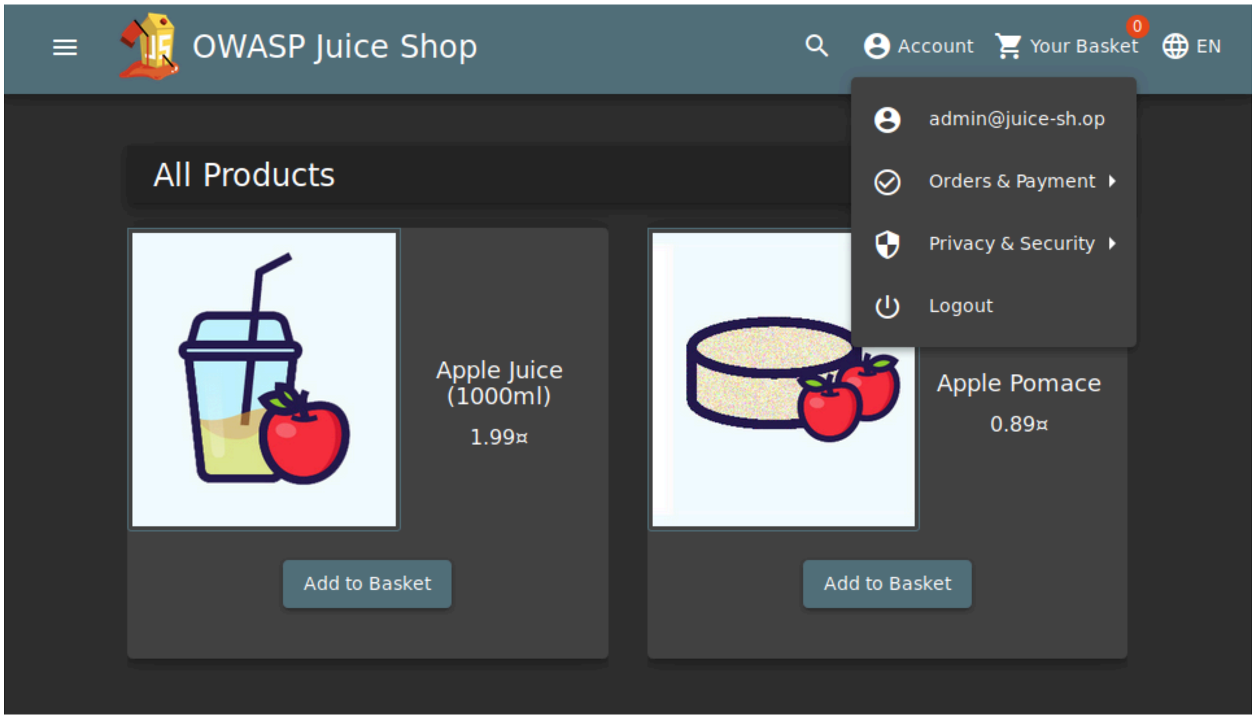


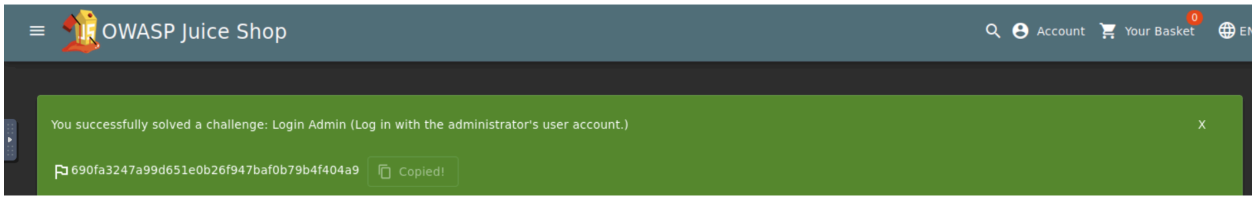
```
POST /rest/user/login HTTP/1.1
Host: 192.168.1.2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.2/
Content-Type: application/json
Content-Length: 28
DNT: 1
Connection: close
Cookie: io=leyzqcW7gYk2cBjxAABr; language=en; cookieconsent_status=dismiss; continueCode=z3BqynemK9E21j5AJJtKUlTRFQiJJh41SR5svZUEptKOADOJkNWrMpgVRXQv

{"email":"a","password":"a"}
```

We will now change the "a" next to the email to: ' or 1=1-- and forward it to the server.

Why does this work?

1. The character ' will close the brackets in the SQL query
2. 'OR' in a SQL statement will return true if either side of it is true. As 1=1 is always true, the whole statement is true. Thus it will tell the server that the email is valid, and log us into user id 0, which happens to be the administrator account.
3. The -- character is used in SQL to comment out data, any restrictions on the login will no longer work as they are interpreted as a comment. This is like the # and // comment in python and javascript respectively.





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

690fa3247a99d651e0b26f947baf0b79b4f404a9 Copied!

Answer: 690fa3247a99d651e0b26f947baf0b79b4f404a9

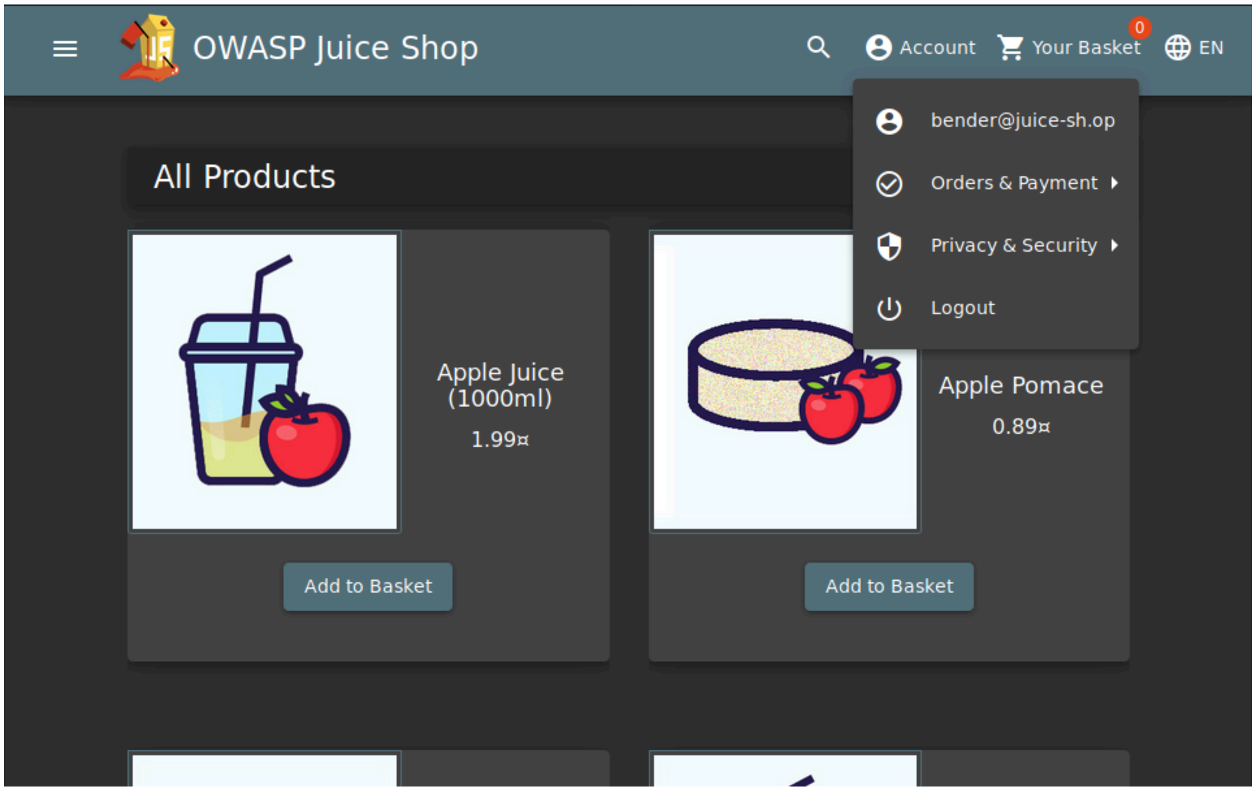
Log into the Bender account!

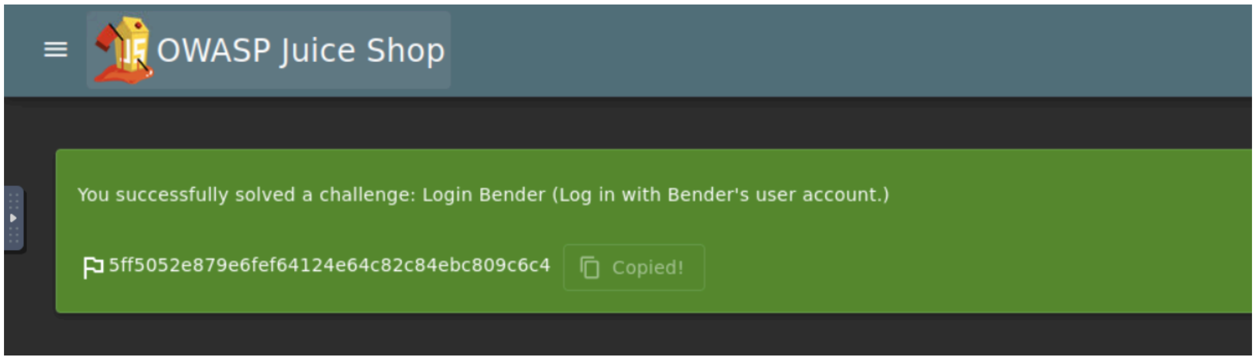
Similar to what we did in Question #1, we will now log into Bender's account! Capture the login request again, but this time we will put: bender@juice-sh.op'-- as the email.

```
{"email":"bender@juice-sh.op'--","password":"a"}
```

Now, forward that to the server!
But why don't we put the 1=1?

Well, as the email address is valid (which will return true), we do not need to force it to be true. Thus we are able to use '-- to bypass the login system. Note the 1=1 can be used when the email or username is not known or invalid.





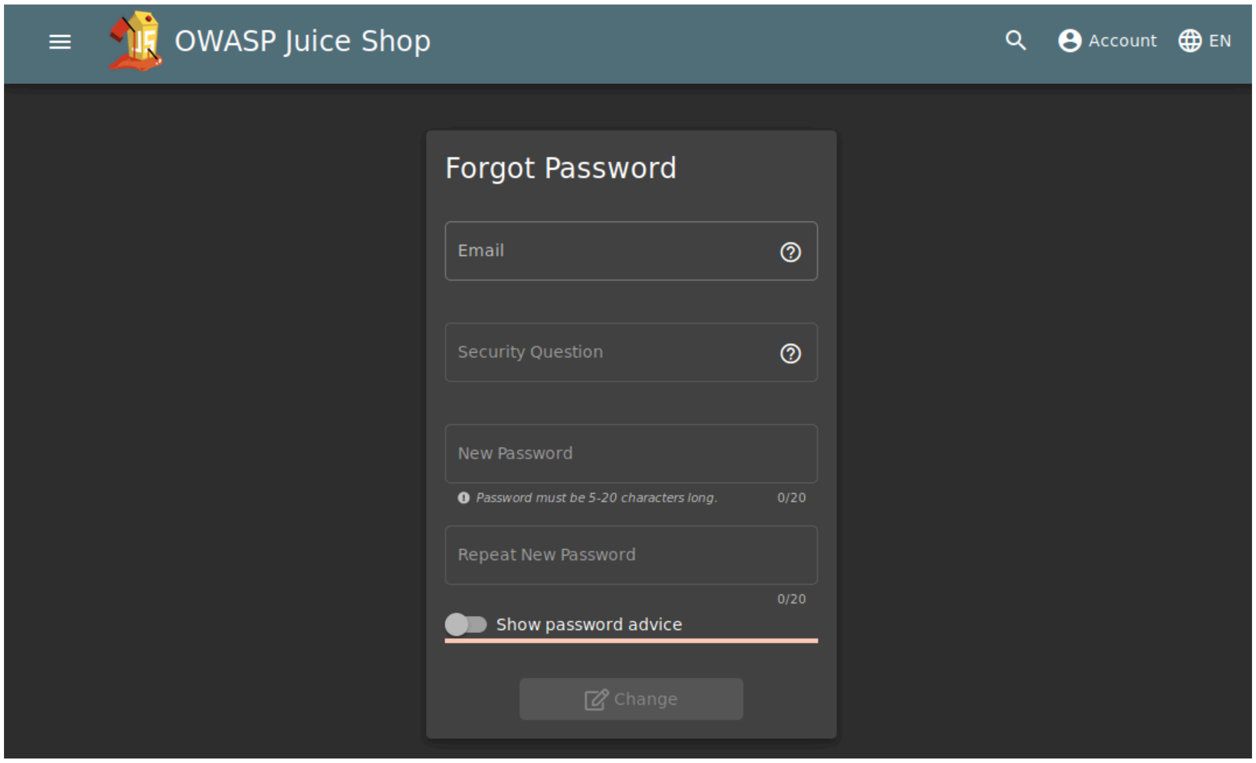
You successfully solved a challenge: Login Bender (Log in with Bender's user account.)

5ff5052e879e6fef64124e64c82c84ebc809c6c4

Answer: 5ff5052e879e6fef64124e64c82c84ebc809c6c4

****

Who Broke My Lock?



In this task, we will look at exploiting authentication through different flaws. When talking about flaws within authentication, we include mechanisms that are vulnerable to manipulation. These mechanisms, listed below, are what we will be exploiting.

- Weak passwords in high privileged accounts
- Forgotten password pages

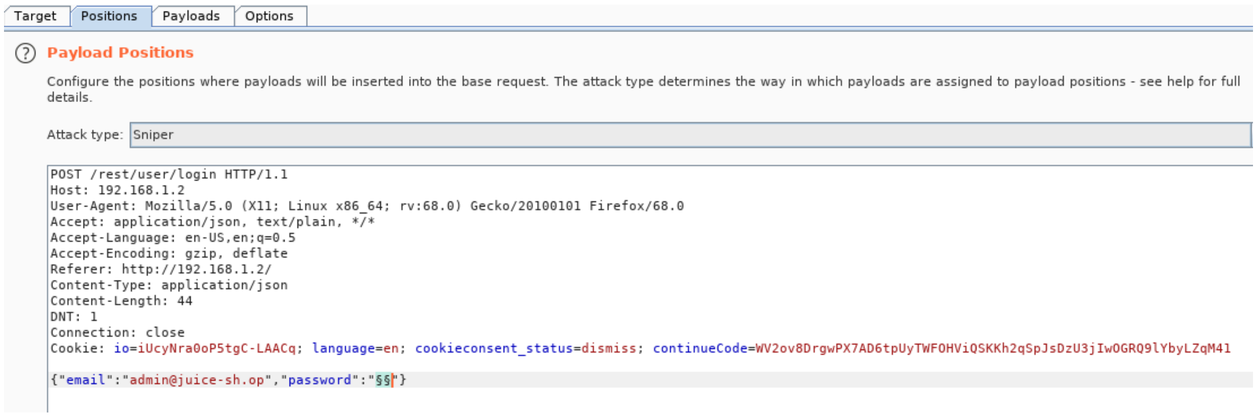
****

Answer the questions below:

Bruteforce the Administrator account's password!

We have used SQL Injection to log into the Administrator account but we still don't know the password. Let's try a brute-force attack! We will once again capture a login request, but instead of sending it through the proxy, we will send it to Intruder.

Go to Positions and then select the Clear § button. In the password field place two § inside the quotes. To clarify, the § § is not two sperate inputs but rather Burp's implementation of quotations e.g. "". The request should look like the image below.



Payload Positions

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

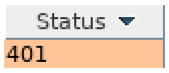
Attack type: Sniper

```
POST /rest/user/login HTTP/1.1
Host: 192.168.1.2
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.2/
Content-Type: application/json
Content-Length: 44
DNT: 1
Connection: close
Cookie: io=iUcyNra0oP5tgC-LAACq; language=en; cookieconsent_status=dismiss; continueCode=WV2ov8DrgwPX7AD6tpUyTWFOHViQSKKh2qSpJsDzU3jIwOGRQ9lYbyLZqM41

{"email":"admin@juice-sh.op","password":"§§"}
```

For the payload, we will be using the best1050.txt from Seclists. (Which can be installed via: apt-get install seclists)

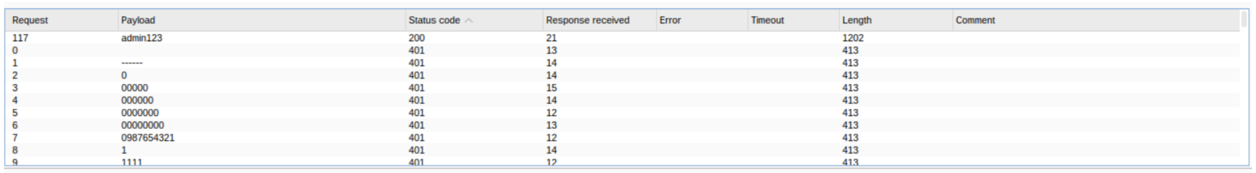
You can load the list from:
/usr/share/wordlists/SecLists/Passwords/Common-Credentials/best1050.txt

Once the file is loaded into Burp, start the attack. You will want to filter for the request by status.

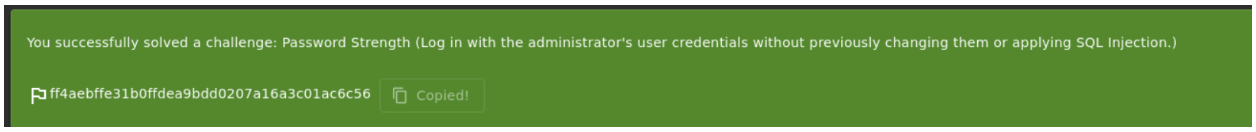
- A failed request will receive a 401 Unauthorized. 

- Whereas a successful request will return a 200 OK.

Once completed, login to the account with the password.



Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
117	admin123	200	21			1202	
0		401	13			413	
1	------	401	14			413	
2	0	401	14			413	
3	00000	401	15			413	
4	000000	401	14			413	
5	0000000	401	12			413	
6	00000000	401	13			413	
7	0987654321	401	12			413	
8	1	401	14			413	
9	1111	401	12			413	



You successfully solved a challenge: Password Strength (Log in with the administrator's user credentials without previously changing them or applying SQL Injection.)

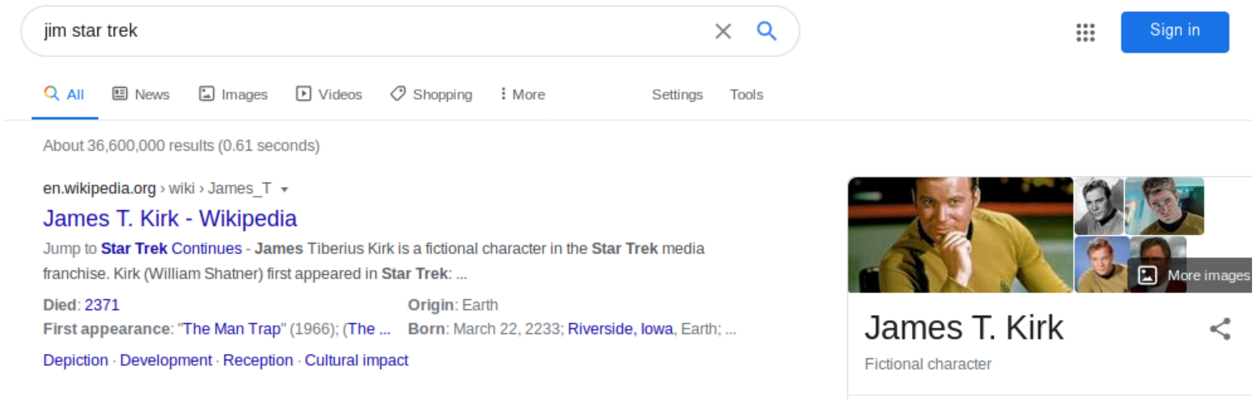
ff4aebffe31b0ffdea9bdd0207a16a3c01ac6c56 Copied!

Answer: ff4aebffe31b0ffdea9bdd0207a16a3c01ac6c56

Reset Jim's password!

Believe it or not, the reset password mechanism can also be exploited! When input into the email field in the Forgot Password page, Jim's security question is set to "Your eldest siblings middle name?".

In Task 2, we found that Jim might have something to do with Star Trek. Googling "Jim Star Trek" gives us a wiki page for Jame T. Kirk from Star Trek.



jim star trek

All · News · Images · Videos · Shopping · More · Settings · Tools

About 36,600,000 results (0.61 seconds)

en.wikipedia.org › wiki › James_T

James T. Kirk - Wikipedia

Jump to Star Trek Continues - James Tiberius Kirk is a fictional character in the Star Trek media franchise. Kirk (William Shatner) first appeared in Star Trek: ...

Died: 2371 Origin: Earth

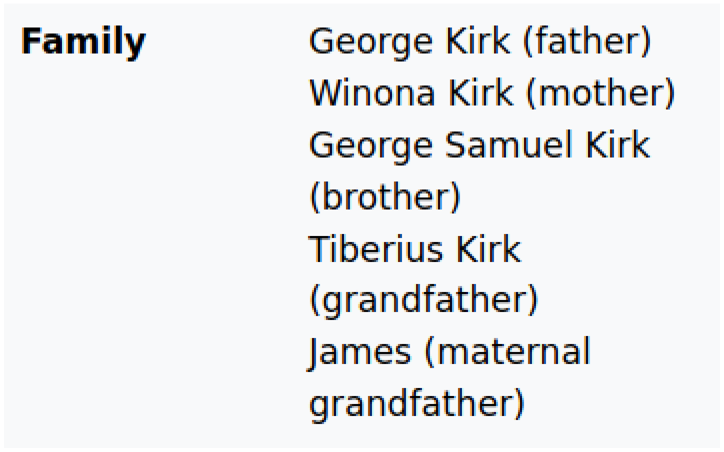
First appearance: "The Man Trap" (1966); (The ... Born: March 22, 2233; Riverside, Iowa, Earth; ...

Depiction · Development · Reception · Cultural impact

James T. Kirk

Fictional character

Looking through the wiki page we find that he has a brother

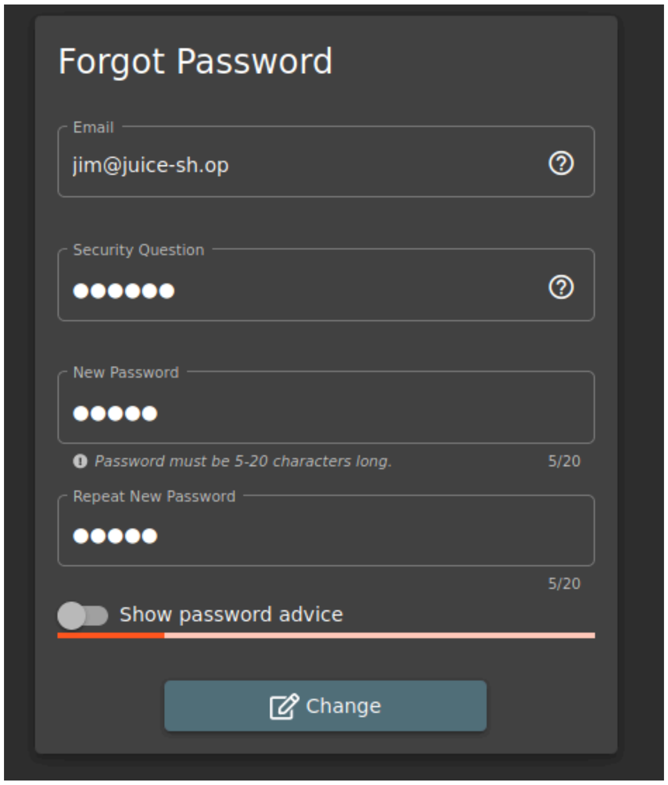


Family	George Kirk (father) Winona Kirk (mother) George Samuel Kirk (brother) Tiberius Kirk (grandfather) James (maternal grandfather)

Looks like his brother's middle name is Samuel

Inputting that into the Forgot Password page allows you to successfully change his password.

You can change it to anything you want!


Forgot Password

Email
jim@juice-sh.op

Security Question
●●●●●●

New Password
●●●●●

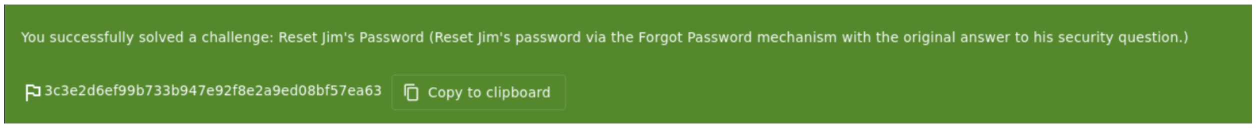
Password must be 5-20 characters long. 5/20

Repeat New Password
●●●●●

5/20

Show password advice

Change

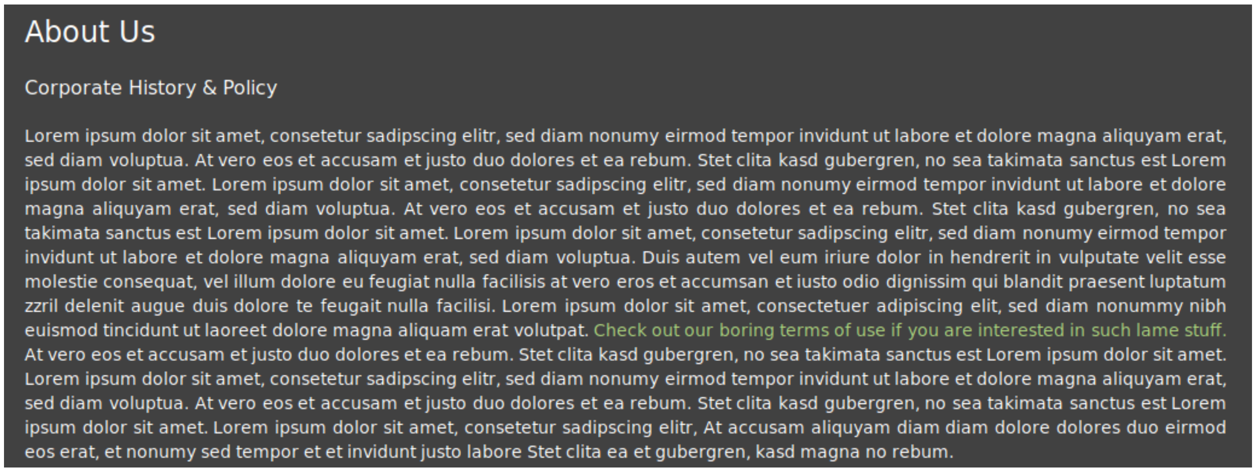

You successfully solved a challenge: Reset Jim's Password (Reset Jim's password via the Forgot Password mechanism with the original answer to his security question.)

3c3e2d6ef99b733b947e92f8e2a9ed08bf57ea63 Copy to clipboard

Answer: 3c3e2d6ef99b733b947e92f8e2a9ed08bf57ea63

****

Ah! Don't Look!


About Us

Corporate History & Policy

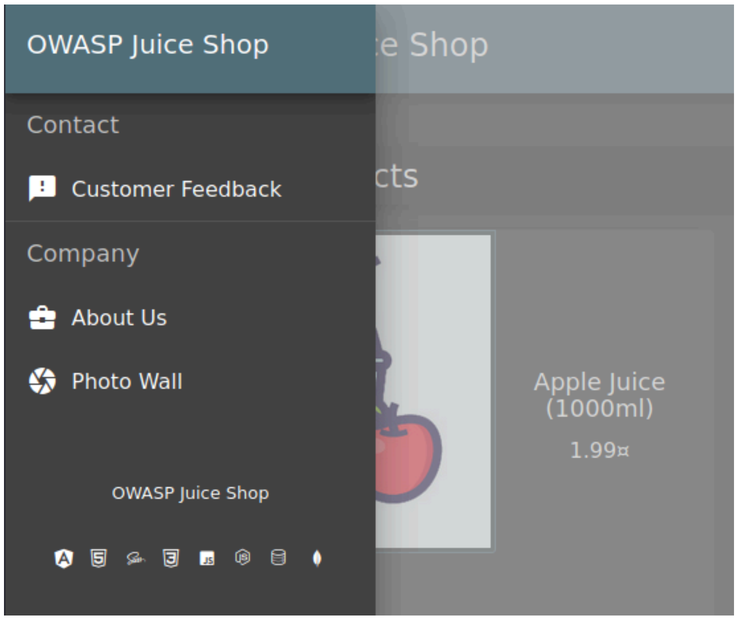
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Check out our boring terms of use if you are interested in such lame stuff. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, At accusam aliquyam diam diam dolore dolores duo eirmod eos erat, et nonumy sed tempor et et invidunt justo labore Stet clita ea et gubergren, kasd magna no rebum.

A web application should store and transmit sensitive data safely and securely. But in some cases, the developer may not correctly protect their sensitive data, making it vulnerable.

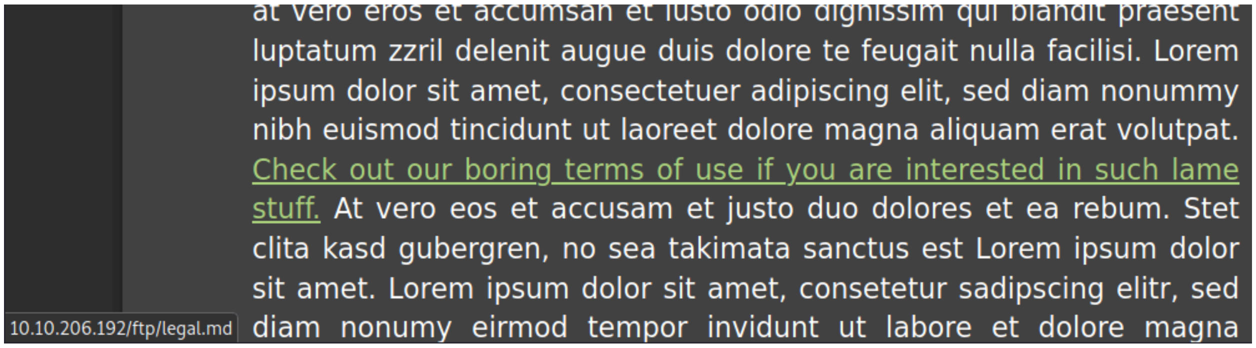
Most of the time, data protection is not applied consistently across the web application making certain pages accessible to the public. Other times information is leaked to the public without the knowledge of the developer, making the web application vulnerable to an attack.
****

Answer the questions below:

Access the Confidential Document!



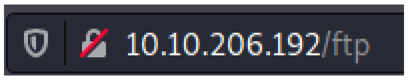
Navigate to the About Us page, and hover over the "Check out our terms of use".



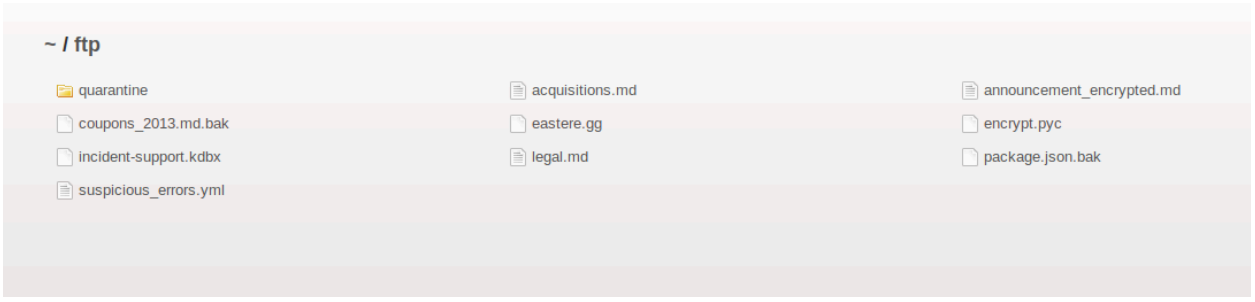
at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Check out our boring terms of use if you are interested in such lame stuff. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna

10.10.206.192/ftp/legal.md

You will see that it links to http://10.201.80.65/ftp/legal.md. Navigating to that /ftp/ directory reveals that it is exposed to the public!



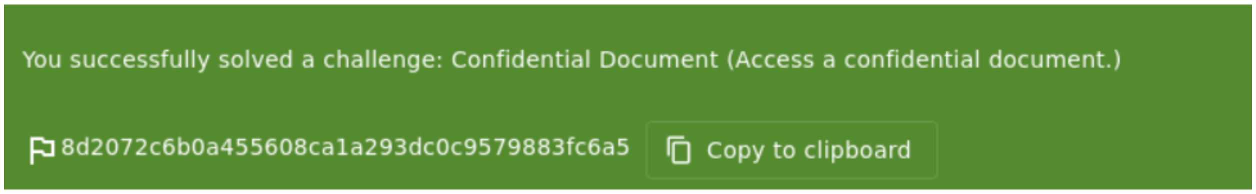
10.10.206.192/ftp


~ / ftp

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

We will download the acquisitions.md and save it. It looks like there are other files of interest here as well.

After downloading it, navigate to the home page to receive the flag!


You successfully solved a challenge: Confidential Document (Access a confidential document.)

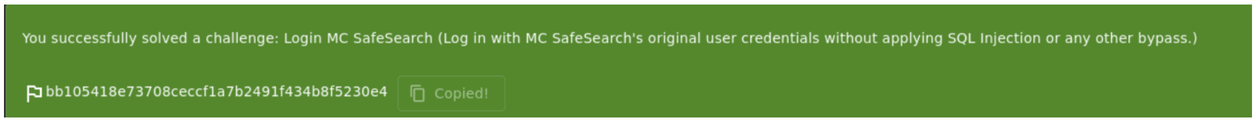
8d2072c6b0a455608ca1a293dc0c9579883fc6a5

Answer: 8d2072c6b0a455608ca1a293dc0c9579883fc6a5

Log into MC SafeSearch's account!
https://youtu.be/v59CX2DiX0Y
After watching the video there are certain parts of the song that stand out.

He notes that his password is "Mr. Noodles" but he has replaced some "vowels into zeros", meaning that he just replaced the o's into 0's.

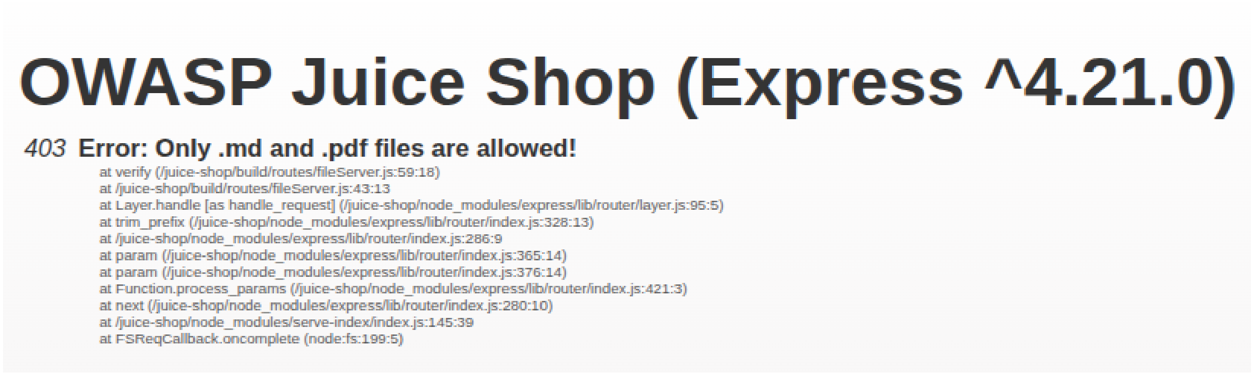
We now know the password to the mc.safesearch@juice-sh.op account is "Mr. N00dles"


You successfully solved a challenge: Login MC SafeSearch (Log in with MC SafeSearch's original user credentials without applying SQL Injection or any other bypass.)

bb105418e73708ceccf1a7b2491f434b8f5230e4

Answer: bb105418e73708ceccf1a7b2491f434b8f5230e4

Download the Backup file!
We will now go back to the http://10.201.80.65/ftp/ folder and try to download package.json.bak. But it seems we are met with a 403 which says that only .md and .pdf files can be downloaded.

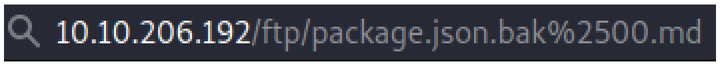


```
OWASP Juice Shop (Express ^4.21.0)

403 Error: Only .md and .pdf files are allowed!
    at verify (/juice-shop/build/routes/fileServer.js:59:18)
    at /juice-shop/build/routes/fileServer.js:43:13
    at Layer.handle [as handle_request] (/juice-shop/node_modules/express/lib/router/layer.js:95:5)
    at trim_prefix (/juice-shop/node_modules/express/lib/router/index.js:328:13)
    at /juice-shop/node_modules/express/lib/router/index.js:286:9
    at param (/juice-shop/node_modules/express/lib/router/index.js:365:14)
    at param (/juice-shop/node_modules/express/lib/router/index.js:376:14)
    at Function.process_params (/juice-shop/node_modules/express/lib/router/index.js:421:3)
    at next (/juice-shop/node_modules/express/lib/router/index.js:280:10)
    at /juice-shop/node_modules/serve-index/index.js:145:39
    at FSReqCallback.oncomplete (node:fs:199:5)
```

To get around this, we will use a character bypass called "Poison Null Byte". A Poison Null Byte looks like this: %00.

Note: as we can download it using the url, we will need to encode this into a url encoded format.

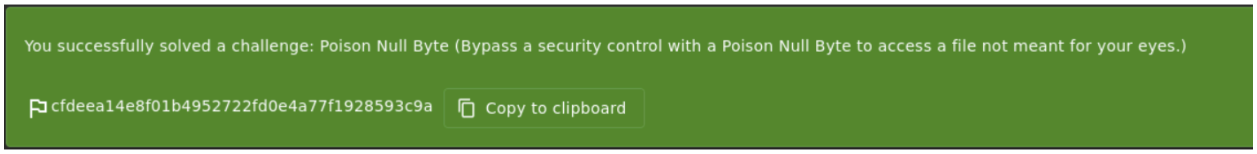
The Poison Null Byte will now look like this: %2500. Adding this and then a .md to the end will bypass the 403 error!



```
10.10.206.192/ftp/package.json.bak%2500.md
```

Why does this work?

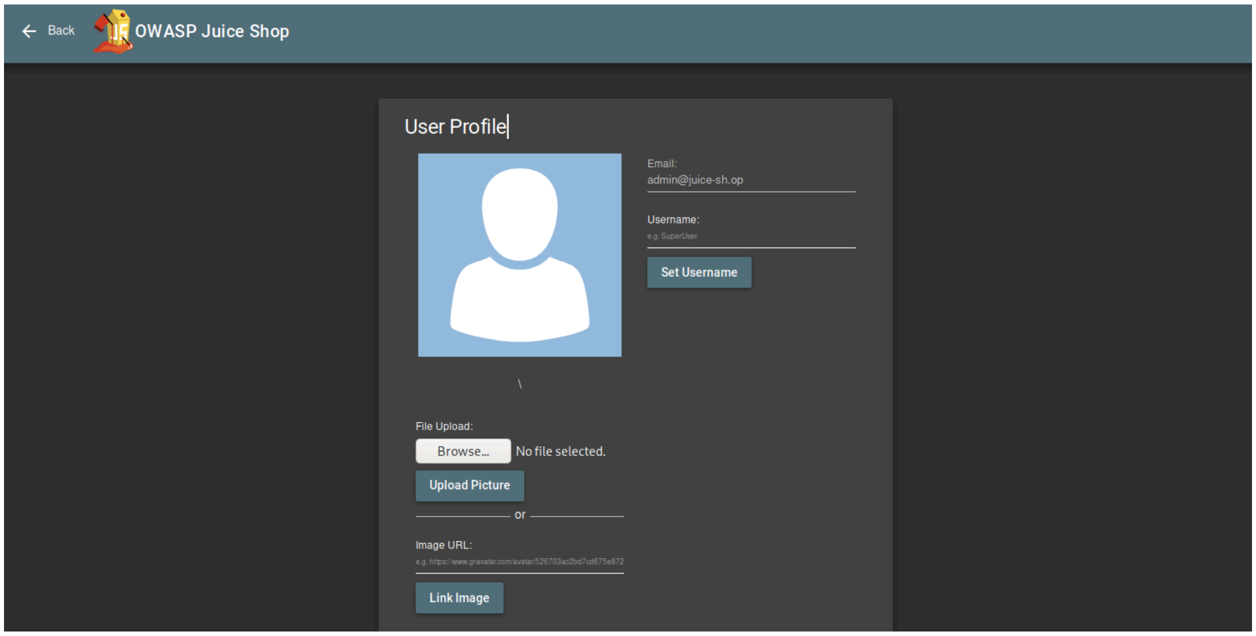
A Poison Null Byte is actually a NULL terminator. By placing a NULL character in the string at a certain byte, the string will tell the server to terminate at that point, nulling the rest of the string.



```
You successfully solved a challenge: Poison Null Byte (Bypass a security control with a Poison Null Byte to access a file not meant for your eyes.)

cfdeea14e8f01b4952722fd0e4a77f1928593c9a    Copy to clipboard
```

Answer: cfdeea14e8f01b4952722fd0e4a77f1928593c9a
****

Who’s Flying This Thing?



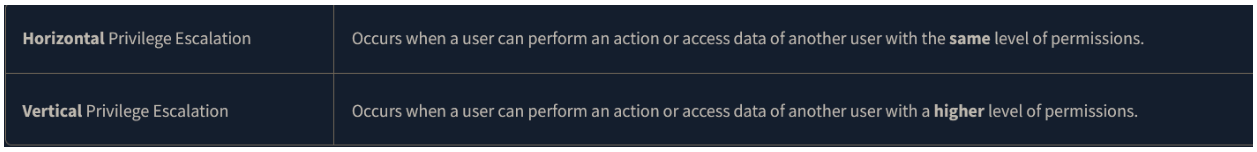
Email: admin@juice-sh.op

Username: e.g. SuperUser

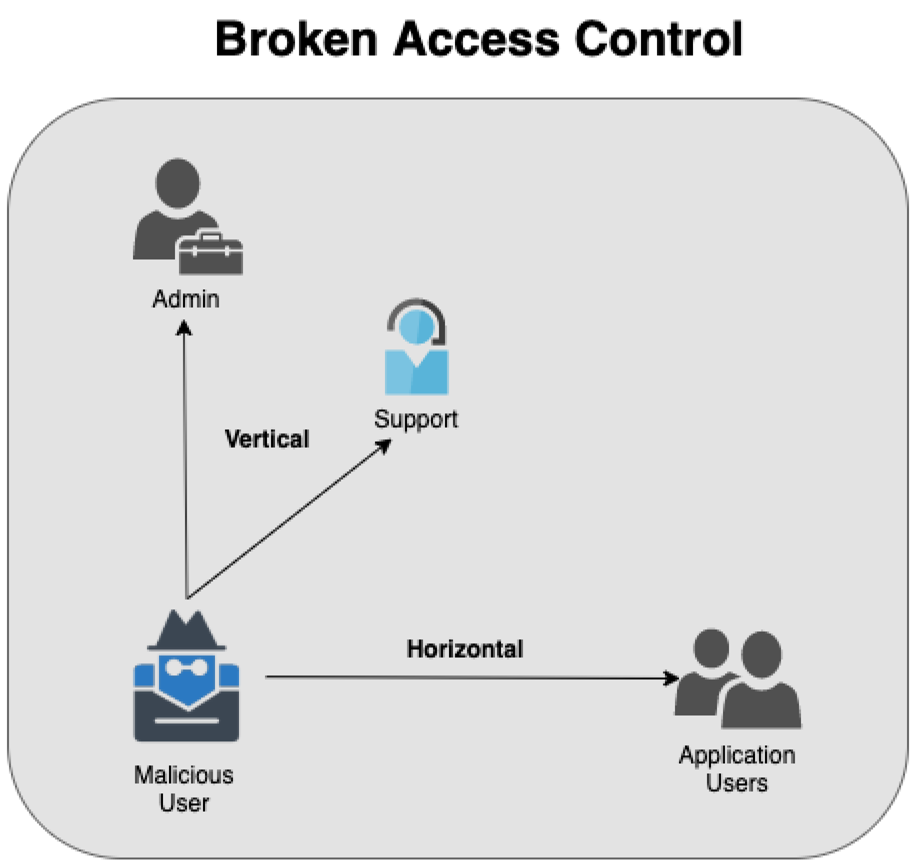
Image URL: e.g. https://www.gravatar.com/avatar/526703ac2bd7cd675e872

Modern-day systems will allow for multiple users to have access to different pages. Administrators most commonly use an administration page to edit, add and remove different elements of a website. You might use these when you are building a website with programs such as Weebly or Wix.

When Broken Access Control exploits or bugs are found, it will be categorised into one of two types:



Horizontal Privilege Escalation	Occurs when a user can perform an action or access data of another user with the **same** level of permissions.
Vertical Privilege Escalation	Occurs when a user can perform an action or access data of another user with a **higher** level of permissions.

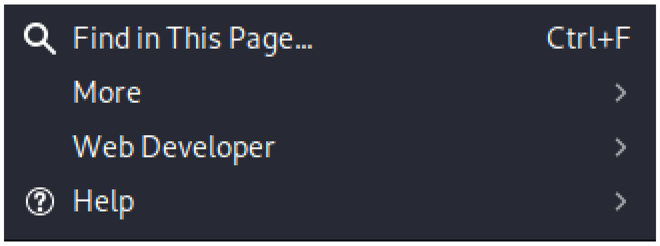


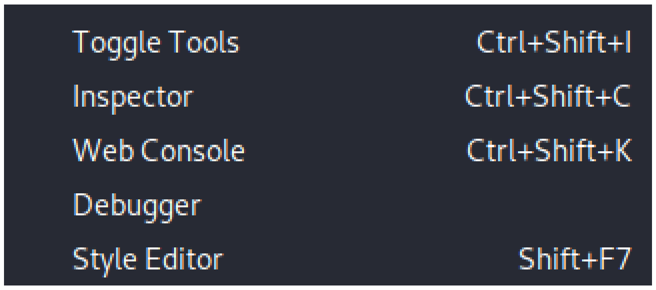
****

Answer the questions below:

Access the administration page!
First, we are going to open the Debugger on Firefox(Or Sources on Chrome).

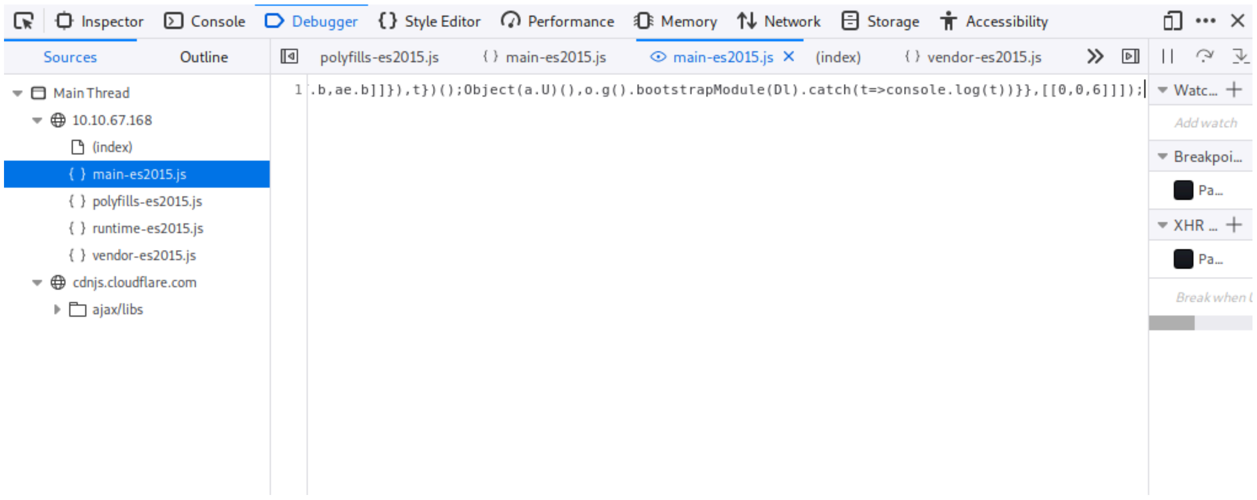
This can be done by navigating to it in the Web Developers menu.





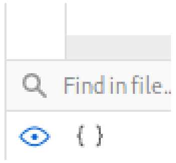
We are then going to refresh the page and look for a javascript file for main-es2015.js

We will then go to that page at: http://10.201.80.65/main-es2015.js



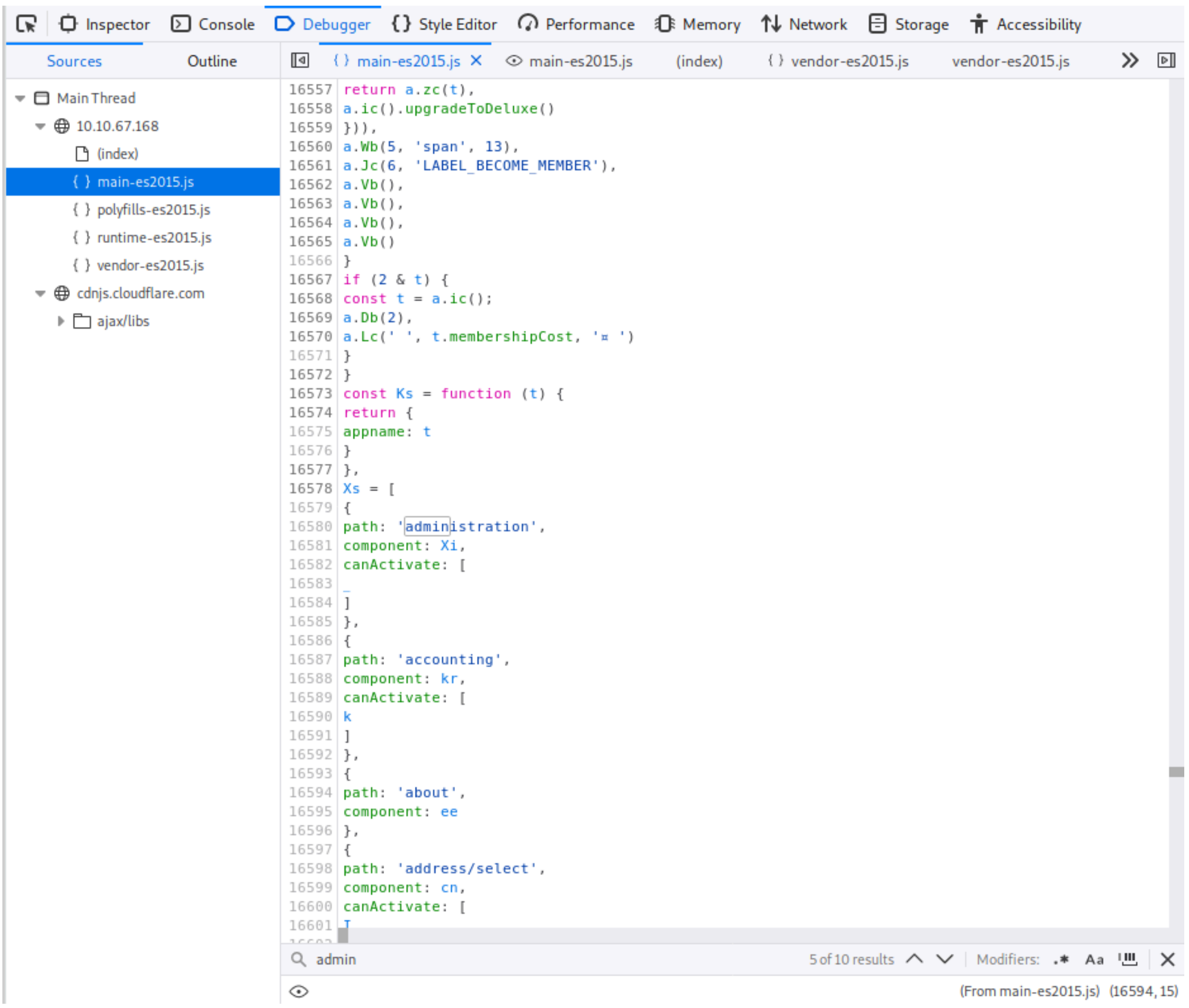
```
.b,ae.b]]}),t})();Object(a.U)(),o.g().bootstrapModule(Dl).catch(t=>console.log(t))}},[[0,0,6]]]);
```

To get this into a format we can read, click the { } button at the bottom



Now search for the term "admin"

You will come across a couple of different words containing "admin" but the one we are looking for is "path: administration”

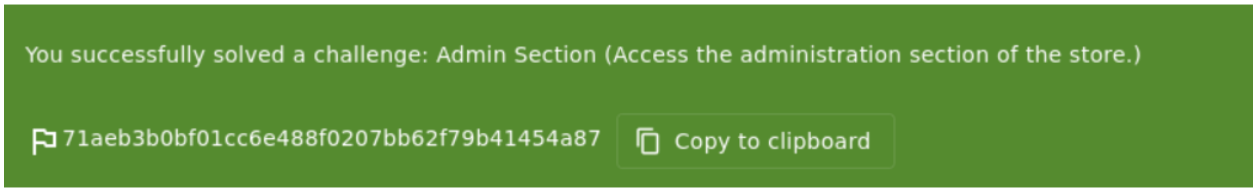


```
return a.zc(t),
a.ic().upgradeToDeluxe()
})),
a.Wb(5, 'span', 13),
a.Jc(6, 'LABEL_BECOME_MEMBER'),
a.Vb(),
a.Vb(),
a.Vb(),
a.Vb()
}
if (2 & t) {
const t = a.ic();
a.Db(2),
a.Lc(' ', t.membershipCost, '¤ ')
}
}
const Ks = function (t) {
return {
appname: t
}
},
Xs = [
{
path: 'administration',
component: Xi,
canActivate: [
_
]
},
{
path: 'accounting',
component: kr,
canActivate: [
k
]
},
{
path: 'about',
component: ee
},
{
path: 'address/select',
component: cn,
canActivate: [
T
```

This hints towards a page called "/#/administration" as can be seen by the about path a couple lines below, but going there while not logged in doesn't work.

As this is an Administrator page, it makes sense that we need to be in the Admin account in order to view it.

A good way to stop users from accessing this is to only load parts of the application that need to be used by them. This stops sensitive information such as an admin page from being leaked or viewed.


You successfully solved a challenge: Admin Section (Access the administration section of the store.)

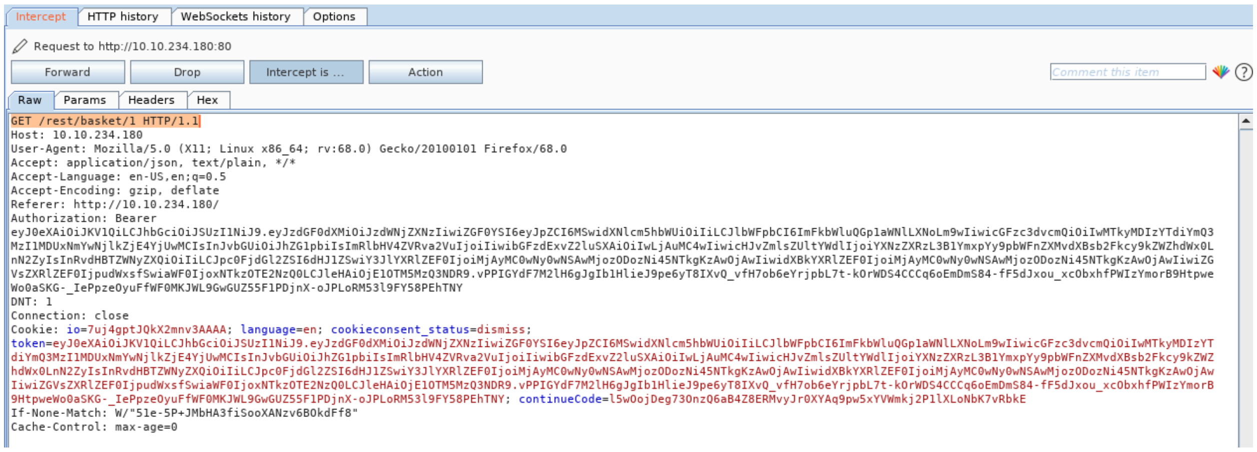
71aeb3b0bf01cc6e488f0207bb62f79b41454a87

Answer: 71aeb3b0bf01cc6e488f0207bb62f79b41454a87

View another user's shopping basket!

Login to the Admin account and click on 'Your Basket'. Make sure Burp is running so you can capture the request!

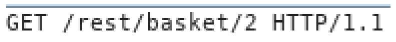
Forward each request until you see: GET /rest/basket/1 HTTP/1.1



Request to http://10.10.234.180:80

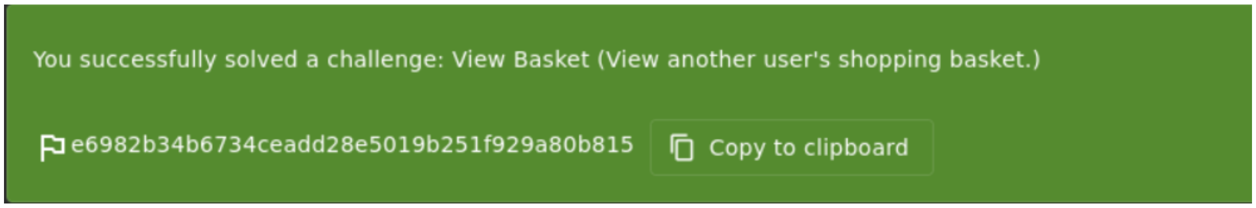
```
GET /rest/basket/1 HTTP/1.1
Host: 10.10.234.180
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.10.234.180/
Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3
MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmlsZUltYWdlIjoiYXNzZXRzL3B1YmxpYy9pbWFnZXMvdXBsb2Fkcy9kZWZhdWx0L
nN2ZyIsInRvdHBTZWNyZXQiOiIiLCJpc0FjdGl2ZSI6dHJ1ZSwiY3JlYXRlZEF0IjoiMjAyMC0wNy0wNSAwMjozODozNi45NTkgKzAwOjAwIiwidXBkYXRlZEF0IjoiMjAyMC0wNy0wNSAwMjozODozNi45NTkgKzAwOjAwIiwiZG
VsZXRlZEF0IjpudWxsfSwiaWF0IjoxNTkzOTE2NzQ0LCJleHAiOjE1OTM5MzQ3NDR9.vPPIGYdF7M2lH6gJgIb1HlieJ9pe6yT8IXvQ_vfH7ob6eYrjpbL7t-kOrWDS4CCCq6oEmDmS84-fF5dJxou_xcObxhfPWIzYmorB9Htpwe
Wo0aSKG-_IePpzeOyuFfWF0MKJWL9GwGUZ55F1PDjnX-oJPLoRM53l9FY58PEhTNY
DNT: 1
Connection: close
Cookie: io=7uj4gptJQkX2mnv3AAAA; language=en; cookieconsent_status=dismiss;
token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYT
diYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmlsZUltYWdlIjoiYXNzZXRzL3B1YmxpYy9pbWFnZXMvdXBsb2Fkcy9kZWZ
hdWx0LnN2ZyIsInRvdHBTZWNyZXQiOiIiLCJpc0FjdGl2ZSI6dHJ1ZSwiY3JlYXRlZEF0IjoiMjAyMC0wNy0wNSAwMjozODozNi45NTkgKzAwOjAwIiwidXBkYXRlZEF0IjoiMjAyMC0wNy0wNSAwMjozODozNi45NTkgKzAwOjAw
IiwiZGVsZXRlZEF0IjpudWxsfSwiaWF0IjoxNTkzOTE2NzQ0LCJleHAiOjE1OTM5MzQ3NDR9.vPPIGYdF7M2lH6gJgIb1HlieJ9pe6yT8IXvQ_vfH7ob6eYrjpbL7t-kOrWDS4CCCq6oEmDmS84-fF5dJxou_xcObxhfPWIzYmorB
9HtpweWo0aSKG-_IePpzeOyuFfWF0MKJWL9GwGUZ55F1PDjnX-oJPLoRM53l9FY58PEhTNY; continueCode=l5wOojDeg73OnzQ6aB4Z8ERMvyJr0XYAq9pw5xYVWmkj2P1lXLoNbK7vRbkE
If-None-Match: W/"51e-5P+JMbHA3fiSooXANzv6BOkdFf8"
Cache-Control: max-age=0
```

Now, we are going to change the number 1 after /basket/ to 2



```
GET /rest/basket/2 HTTP/1.1
```

It will now show you the basket of UserID 2. You can do this for other UserIDs as well, provided that they have one!

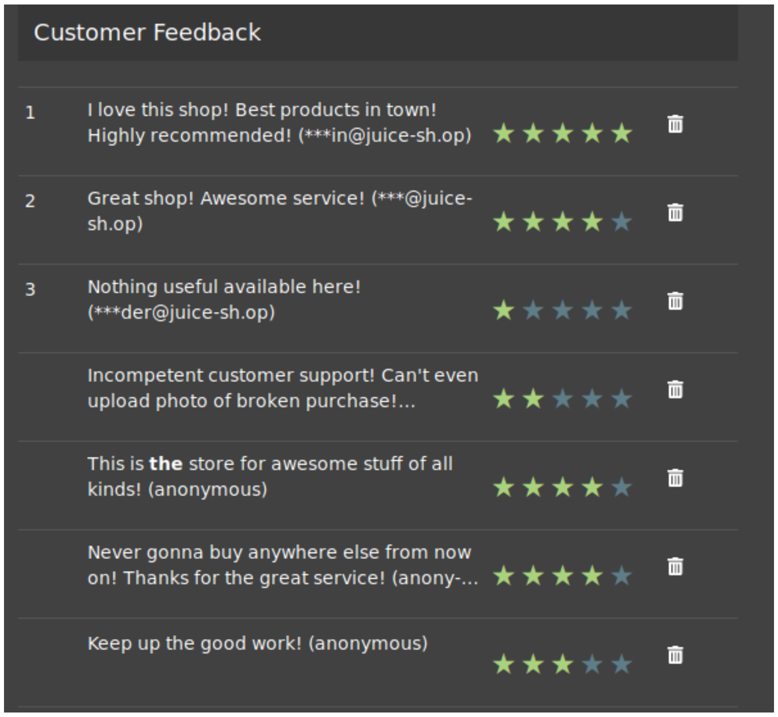

You successfully solved a challenge: View Basket (View another user's shopping basket.)

e6982b34b6734ceadd28e5019b251f929a80b815

Answer: e6982b34b6734ceadd28e5019b251f929a80b815

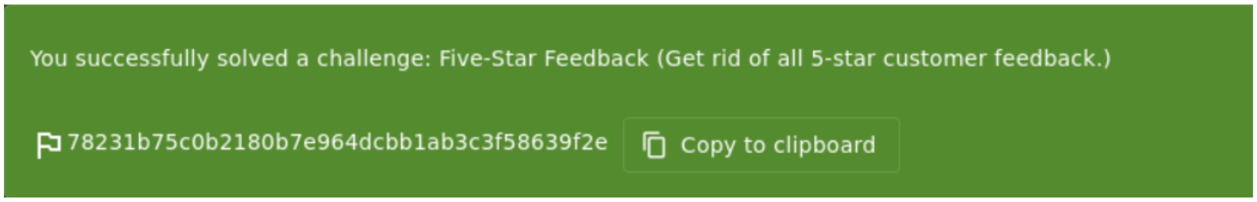
Remove all 5-star reviews!

Navigate to the http://10.201.80.65/#/administration page again and click the bin icon next to the review with 5 stars!



Customer Feedback

1	I love this shop! Best products in town! Highly recommended! (***in@juice-sh.op)	★★★★★	🗑
2	Great shop! Awesome service! (***@juice-sh.op)	★★★★☆	🗑
3	Nothing useful available here! (***der@juice-sh.op)	★☆☆☆☆	🗑
	Incompetent customer support! Can't even upload photo of broken purchase!...	★★☆☆☆	🗑
	This is **the** store for awesome stuff of all kinds! (anonymous)	★★★★☆	🗑
	Never gonna buy anywhere else from now on! Thanks for the great service! (anony-...	★★★★☆	🗑
	Keep up the good work! (anonymous)	★★★☆☆	🗑



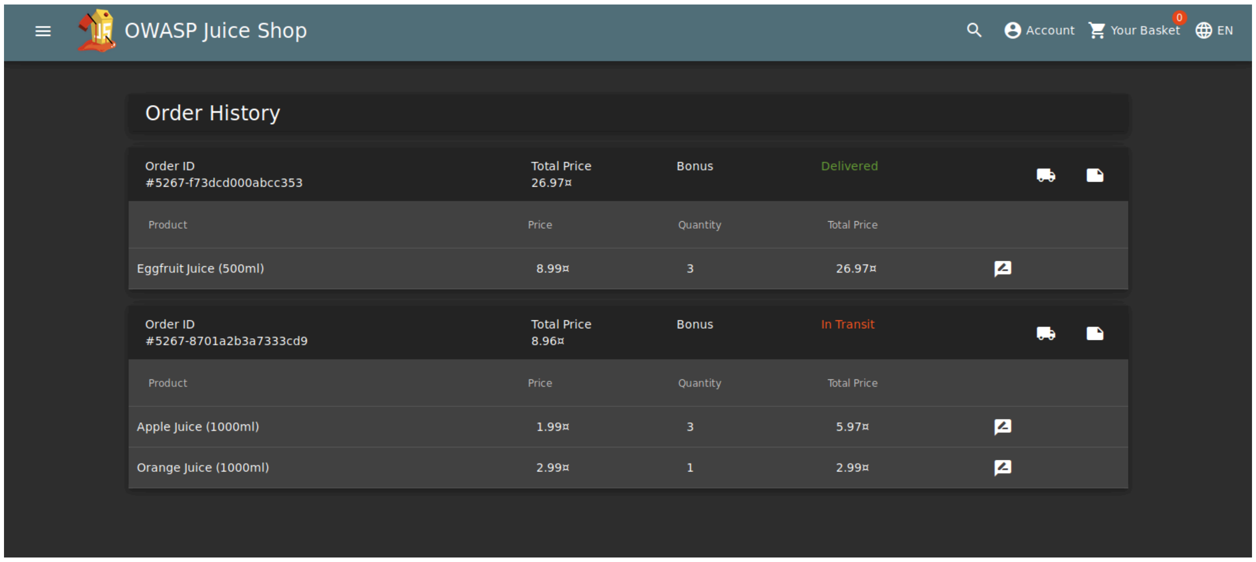
You successfully solved a challenge: Five-Star Feedback (Get rid of all 5-star customer feedback.)

78231b75c0b2180b7e964dcbb1ab3c3f58639f2e

Answer: 78231b75c0b2180b7e964dcbb1ab3c3f58639f2e

****

Where Did This Come From?



Order History

Order ID	Total Price	Bonus	Delivered
#5267-f73dcd000abcc353	26.97¤		

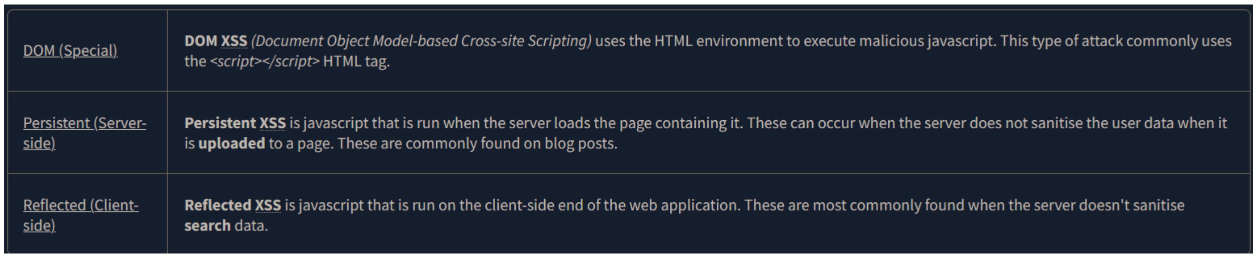
Product	Price	Quantity	Total Price
Eggfruit Juice (500ml)	8.99¤	3	26.97¤

Order ID	Total Price	Bonus	In Transit
#5267-8701a2b3a7333cd9	8.96¤		

Product	Price	Quantity	Total Price
Apple Juice (1000ml)	1.99¤	3	5.97¤
Orange Juice (1000ml)	2.99¤	1	2.99¤

XSS or Cross-site scripting is a vulnerability that allows attackers to run javascript in web applications. These are one of the most found bugs in web applications. Their complexity ranges from easy to extremely hard, as each web application parses the queries in a different way.

There are three major types of XSS attacks:



DOM (Special)	**DOM XSS** *(Document Object Model-based Cross-site Scripting)* uses the HTML environment to execute malicious javascript. This type of attack commonly uses the *<script></script>* HTML tag.
Persistent (Server-side)	**Persistent XSS** is javascript that is run when the server loads the page containing it. These can occur when the server does not sanitise the user data when it is **uploaded** to a page. These are commonly found on blog posts.
Reflected (Client-side)	**Reflected XSS** is javascript that is run on the client-side end of the web application. These are most commonly found when the server doesn't sanitise **search** data.

****

Answer the questions below:

Perform a DOM XSS!

We will be using the iframe element with a javascript alert tag:

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

Inputting this into the search bar will trigger the alert.

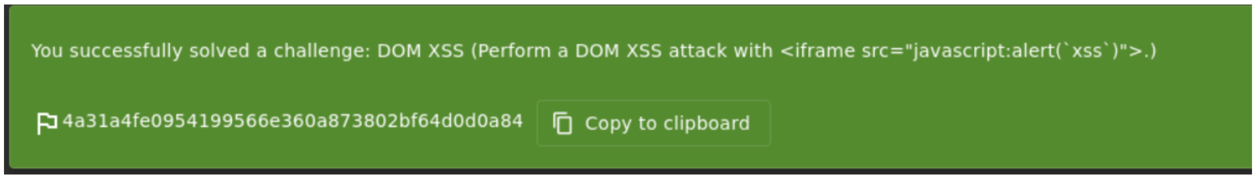
Note that we are using iframe which is a common HTML element found in many web applications, there are others which also produce the same result.

This type of XSS is also called XFS (Cross-Frame Scripting), is one of the most common forms of detecting XSS within web applications.

Websites that allow the user to modify the iframe or other DOM elements will most likely be vulnerable to XSS.

Why does this work?

It is common practice that the search bar will send a request to the server in which it will then send back the related information, but this is where the flaw lies. Without correct input sanitation, we are able to perform an XSS attack against the search bar.


You successfully solved a challenge: DOM XSS (Perform a DOM XSS attack with <iframe src="javascript:alert(`xss`)">.)

4a31a4fe0954199566e360a873802bf64d0d0a84

Answer: 4a31a4fe0954199566e360a873802bf64d0d0a84

Perform a persistent XSS!
First, login to the admin account.

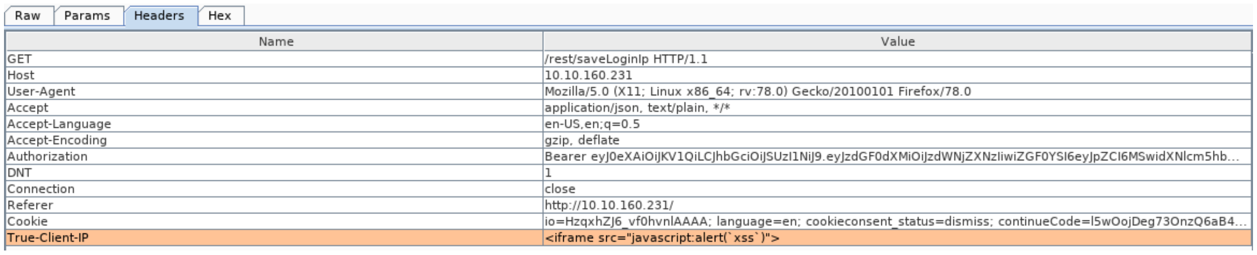
We are going to navigate to the "Last Login IP" page for this attack.

It should say the last IP Address is 0.0.0.0 or 10.x.x.x

As it logs the 'last' login IP we will now logout so that it logs the 'new' IP.

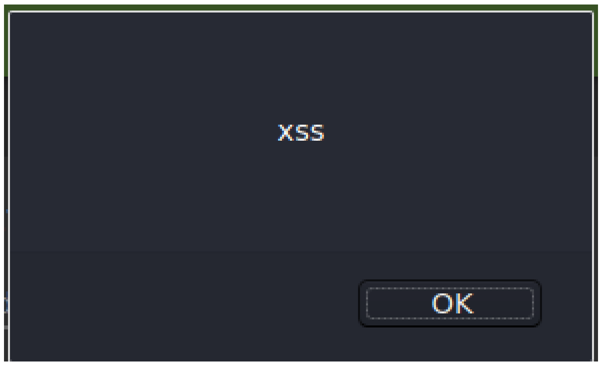
Make sure that Burp intercept is on, so it will catch the logout request.

We will then head over to the Headers tab where we will add a new header:



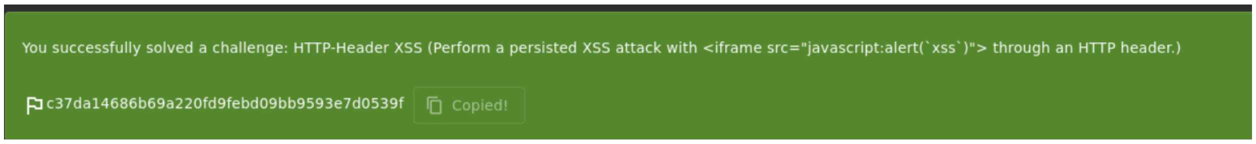
Name	Value
GET	/rest/saveLoginIp HTTP/1.1
Host	10.10.160.231
User-Agent	Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept	application/json, text/plain, */*
Accept-Language	en-US,en;q=0.5
Accept-Encoding	gzip, deflate
Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hb...
DNT	1
Connection	close
Referer	http://10.10.160.231/
Cookie	io=HzqxhZJ6_vf0hvnlAAAA; language=en; cookieconsent_status=dismiss; continueCode=l5wOojDeg73OnzQ6aB4...
True-Client-IP	<iframe src="javascript:alert(`xss`)">

Then forward the request to the server!
When signing back into the admin account and navigating to the Last Login IP page again, we will see the XSS alert!



Why do we have to send this Header?

The True-Client-IP header is similar to the X-Forwarded-For header, both tell the server or proxy what the IP of the client is. Due to there being no sanitation in the header we are able to perform an XSS attack.



You successfully solved a challenge: HTTP-Header XSS (Perform a persisted XSS attack with <iframe src="javascript:alert(`xss`)"> through an HTTP header.)

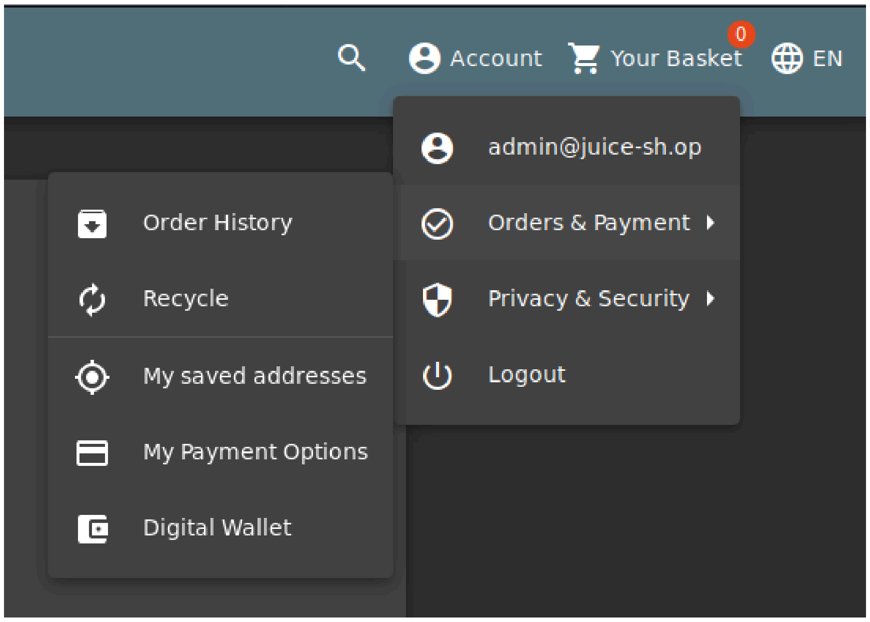
c37da14686b69a220fd9febd09bb9593e7d0539f

Answer: c37da14686b69a220fd9febd09bb9593e7d0539f

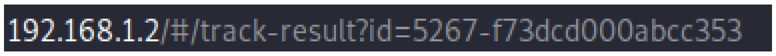
Perform a reflected XSS!

First, we are going to need to be on the right page to perform the reflected XSS!

Login into the admin account and navigate to the 'Order History' page.



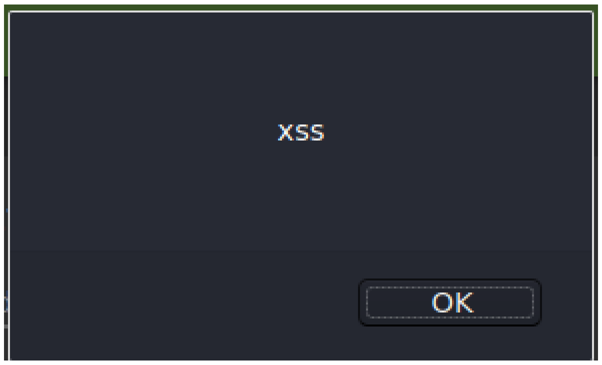
From there you will see a "Truck" icon, clicking on that will bring you to the track result page. You will also see that there is an id paired with the order.



```
192.168.1.2/#/track-result?id=5267-f73dcd000abcc353
```

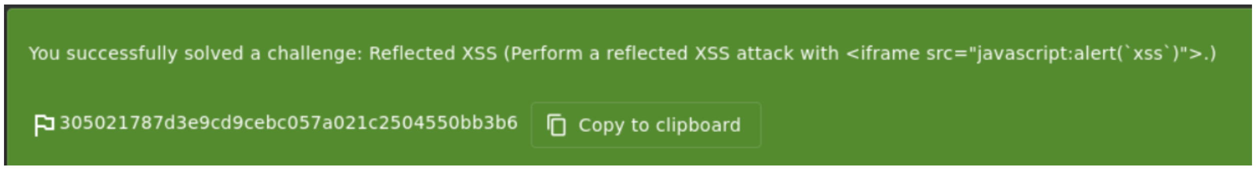
We will use the iframe XSS, <iframe src="javascript:alert(`xss`)">, in the place of the 5267-f73dcd000abcc353

After submitting the URL, refresh the page and you will then get an alert saying XSS!



Why does this work?

The server will have a lookup table or database (depending on the type of server) for each tracking ID. As the 'id' parameter is not sanitised before it is sent to the server, we are able to perform an XSS attack.



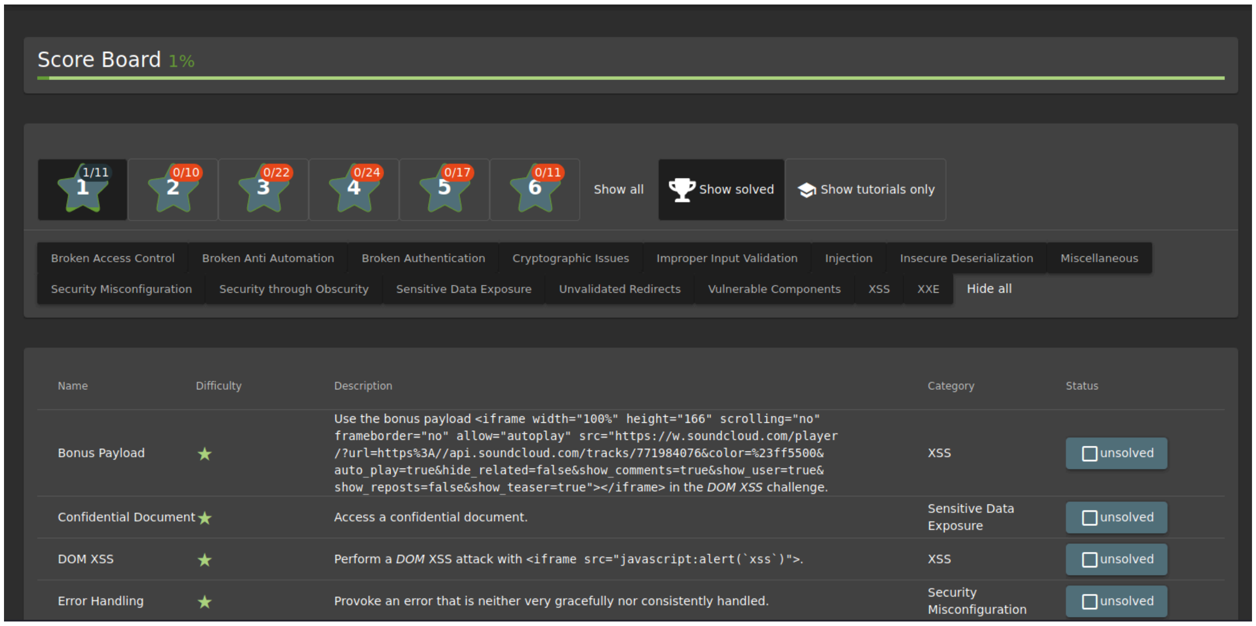
You successfully solved a challenge: Reflected XSS (Perform a reflected XSS attack with <iframe src="javascript:alert(`xss`)">.)

305021787d3e9cd9cebc057a021c2504550bb3b6

Answer: 305021787d3e9cd9cebc057a021c2504550bb3b6

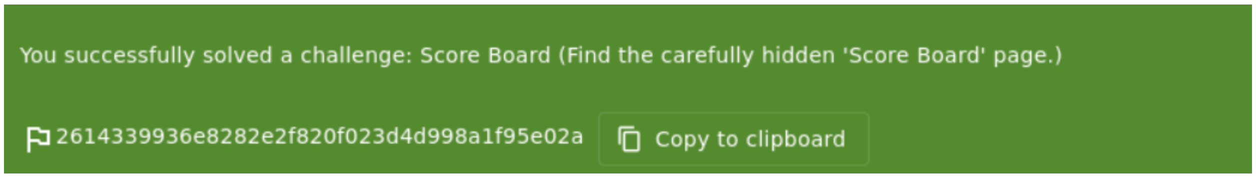
****

Exploration!



Name	Difficulty	Description	Category	Status
Bonus Payload	★	Use the bonus payload `<iframe width="100%" height="166" scrolling="no" frameborder="no" allow="autoplay" src="https://w.soundcloud.com/player/?url=https%3A//api.soundcloud.com/tracks/771984076&color=%23ff5500&auto_play=true&hide_related=false&show_comments=true&show_user=true&show_reposts=false&show_teaser=true"></iframe>` in the *DOM XSS* challenge.	XSS	☐ unsolved
Confidential Document	★	Access a confidential document.	Sensitive Data Exposure	☐ unsolved
DOM XSS	★	Perform a *DOM* XSS attack with `<iframe src="javascript:alert(`xss`)">`.	XSS	☐ unsolved
Error Handling	★	Provoke an error that is neither very gracefully nor consistently handled.	Security Misconfiguration	☐ unsolved

If you wish to tackle some of the harder challenges that were not covered within this room, check out the /#/score-board/ section on Juice-shop. Here you can see your completed tasks as well as other tasks in varying difficulty.



You successfully solved a challenge: Score Board (Find the carefully hidden 'Score Board' page.)

2614339936e8282e2f820f023d4d998a1f95e02a

Answer: 2614339936e8282e2f820f023d4d998a1f95e02a