## Memo-rap-check

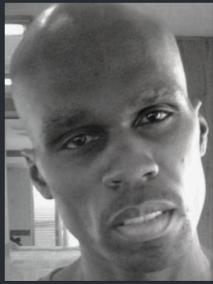
## Haters delight

Due to some risky et dolore magna Bitcoin "investments" the diam voluptua. At networth of rapper vero eos et 50 Cent is once accusam et justo again higher then duo dolores et ea 50 cent. After a rebum. Stet clita managed to come sanctus est Lorem back full swing.

recent interview the rapper plans to invest the money into the community into the community by funding invidunt ut labore supplying crack et dolore magna dealers with the latest and newest

amet,

aliquyam erat, sed ipsum dolor sit According to a dolor sit amet, amet. Lorem ipsum consetetur sadipscing elitr. eirmod tempor diam voluptua. At vero eos et. Lorem ipsum dolor accusam et justo duo dolores et ea



Rapper 50 cent back on track..

## Hot this month Tekashi69 sues Takeshi's Castle for copiright infringement. Rapper allegedly states: I'm clearly in \*heavy

mubmling\* and

therefore this is a

clear and shut case.

Main page of Memo rap, the first thing we notice is the dude sure is stoned:)

## Task descriptions

- Browse the application. Make note of any endpoints which might process user input.
- You can find the flag within the route "/flag". Within the source code, find the reason why you can't access it.
- Within the source, find out how and by whom your inputs are processed.
- Exploit the application to retrieve the flag remotely. For debuggin purposes you might want to temporarily patch the source, for example by commenting out parts of the code.

The first thing we do is obviously nmap -sV

```
--$ <u>sudo</u> nmap -sV 172.17.0.2
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-13 10:56 CET
Nmap scan report for 172.17.0.2
Host is up (0.0000040s latency).
Not shown: 999 closed tcp ports (reset)
                 STATE SERVICE VERSION
80/tcp open http
1 service unrecognized despite returning data. If you know the service/version,
ollowing fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port80-TCP:V=7.93%I=7%D=1/13%Time=63C12AE6%P=x86_64-pc-linux-gnu%r(GetR
SF:equest, E81, "HTTP/1\.1\x20200\x200K\r\nX-DNS-Prefetch-Control:\x20off\r\
SF:nExpect-CT:\x20max-age=0\r\nX-Frame-Options:\x20SAMEORIGIN\r\nStrict-Tr
SF:ansport-Security:\x20max-age=15552000;\x20includeSubDomains\r\nX-Downlo
SF:ad-Options:\x20noopen\r\nX-Content-Type-Options:\x20nosniff\r\nX-Permit
SF:ted-Cross-Domain-Policies:\x20none\r\nReferrer-Policy:\x20no-referrer\r
SF:\nX-XSS-Protection:\x200\r\ncontent-type:\x20text/html\r\ncontent-lengt
SF:h:\x203294\r\nDate:\x20Fri,\x2013\x20Jan\x202023\x2009:56:54\x20GMT\r\n
SF: Connection: \\ x20close \\ r\\ n\\ r\\ n< html\\ x20lang = \\ "en\\ ">\\ n\\ x20\\ x20\\ x20\\ x20< head>\\ n
SF:20\x20<title>Memo\x20Rap\x20Check</title>\n\x20\x20\x20\x20\x20\x20<metoptime to the control of the contro
SF: a \times 20 name = \ ''viewport \ '' \times 20 content = \ ''width = device - width, \ 'x20 initial - scal' + a \times 20 name = \ ''viewport \ 
SF:ons,20A,"HTTP/1\.1\x20404\x20Not\x20Found\r\nX-DNS-Prefetch-Control:\x2
SF:0off\r\nExpect-CT:\x20max-age=0\r\nX-Frame-Options:\x20SAMEORIGIN\r\nSt
SF:rict-Transport-Security:\x20max-age=15552000;\x20includeSubDomains\r\nX
SF:-Download-Options:\x20noopen\r\nX-Content-Type-Options:\x20nosniff\r\nX
SF:-Permitted-Cross-Domain-Policies:\x20none\r\nReferrer-Policy:\x20no-ref
SF:errer\r\nX-XSS-Protection:\x200\r\ncontent-type:\x20application/json;\x
```

we see there is a http server running on port 80, which is common, and the other output, which is uncommon

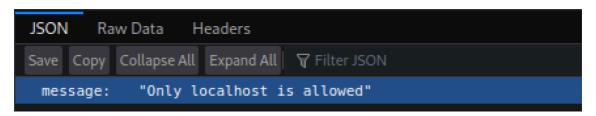
now let's try to dirb it, maybe we find something interesting

```
—— Scanning URL: http://172.17.0.2/ ——
+ http://172.17.0.2/feedback (CODE:200|SIZE:2690)
+ http://172.17.0.2/flag (CODE:401|SIZE:39)
+ http://172.17.0.2/list (CODE:401|SIZE:39)
```

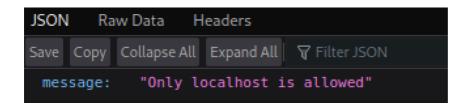
we see 3 direcotries, 1 of them has to contain the flag



Feedback directory: the text field might be exploitable, but let's check other directories first



The list directory is only allowed to be accessed with localhost, so only if I am the one who runs the server

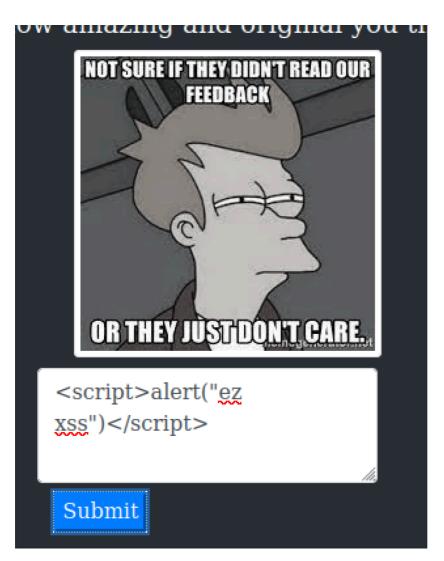


Same thing with /flag directory

Let's get back to our /feedback page and see if we can inject some code in that chat box

```
</aside>
v<article id="article">
v<article id="ar
```

from what I see the submit button creates POST request



First thing that comes to mind is to try and exploit the comment box, which didn't work:(

```
return reply.type('text/html').send(fs.readFileSync('views/feedb
ack.html',{encoding:'utf8', flag:'r'}));
});
                               fastify.post('/api/submit', async (request, reply) => {
    let { feedback } = request.body;
                                        if (feedback) {
    return db.addFeedback(feedback)
                                                          .then(() => {
    bot.purgeData(db);
                      reply.send({ message: 'Our intern has wo rked tirelessly to process your feedback.' });
Inspector Consc
                      .catch(() => reply.send({ message: 'Ooops, could
n\'t process your feedback.', error: 1}));
Q. Search HTML
        ▼<div class="row
          ▼<div class="d
                                        return reply.send({ message: 'Missing parameters.', error: 1 });
          ▼ <div class=
            ▼<form id=
                               fastify.get('/list', async (request, reply) => {
   if (request.ip != '127.0.0.1') {
      return reply.code(401).send({ message: 'Only localhost i
             wediv id=
                      s allowed'});
                <butto
              </div>
             </form>
                                        return await db.getFeedback()
                                                 .then(feedback => {
            </div>
                                                          if (feedback) {
          </div>
         </div>
                                                                   return reply.view('views/list.pug', { fe
        </section>
                      edback: feedback });
      </article>
     </div>
```

In the container we can encounter the source code

We can notice what addFeedback() function is called when we press the button

We could have commented out the function and check /list in the browser, but for what if we already got the flag:

flag\_you\_w0uldn't\_copy\_paste\_content\_Would\_u?