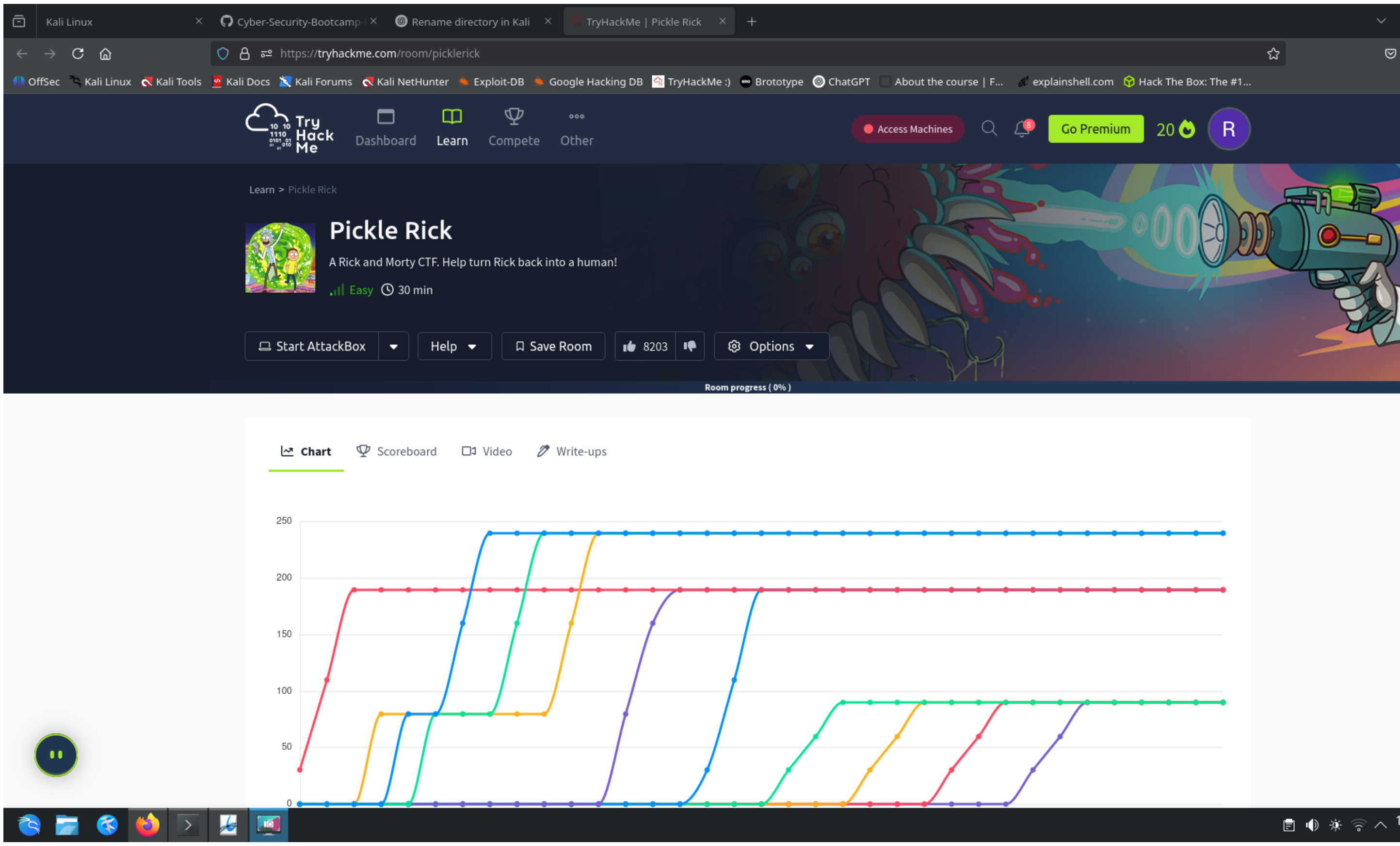
TryHackMe CTF Write-up

Room: Pickle Rick

Platform: TryHackMe

Submitted by:

R S ABHINAV



Pickle Rick

A Rick and Morty CTF. Help turn Rick back into a human!

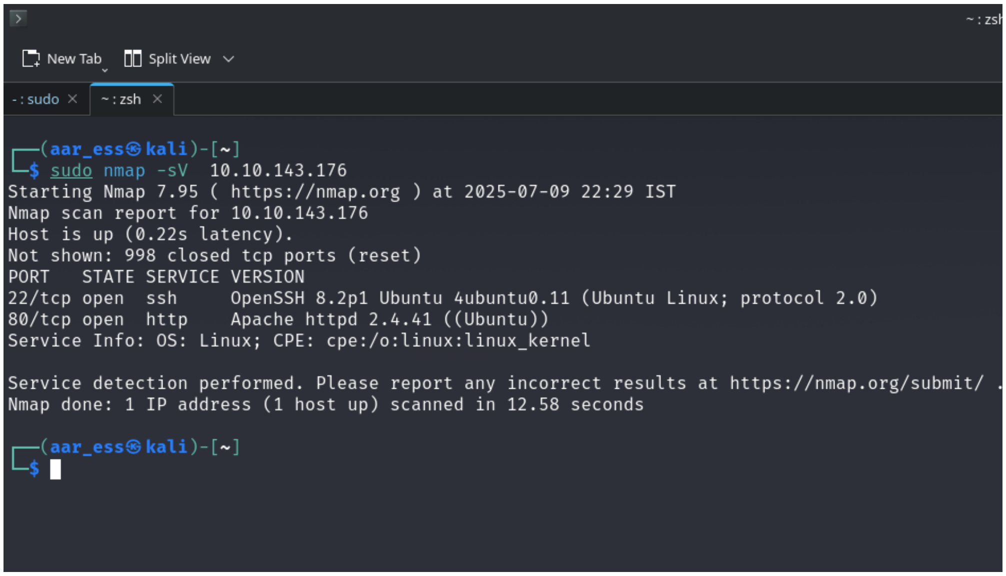
Easy 30 min

1. Connecting to the Room

After deploying the **Pickle Rick** room on TryHackMe, I connected to the target machine using the OpenVPN configuration file. The connection was successfully established.

2. Scanning the Target

I performed an Nmap scan on the target IP address `10.10.143.176` to identify open ports and services.

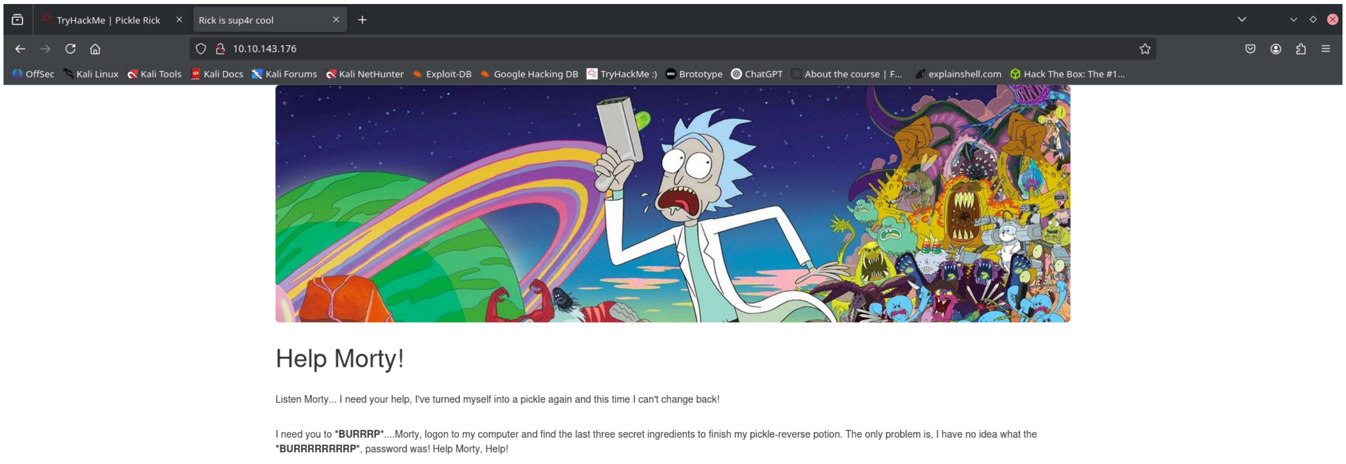


```
┌──(aar_ess㉿kali)-[~]
└─$ sudo nmap -sV  10.10.143.176
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-09 22:29 IST
Nmap scan report for 10.10.143.176
Host is up (0.22s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.11 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.58 seconds

┌──(aar_ess㉿kali)-[~]
└─$ 
```

The scan showed open ports for HTTP (80) and SSH (22). Based on this, I proceeded to explore the HTTP service in a browser.

3. Accessing the Website

Navigating to `http://10.10.143.176` led to a basic web page.



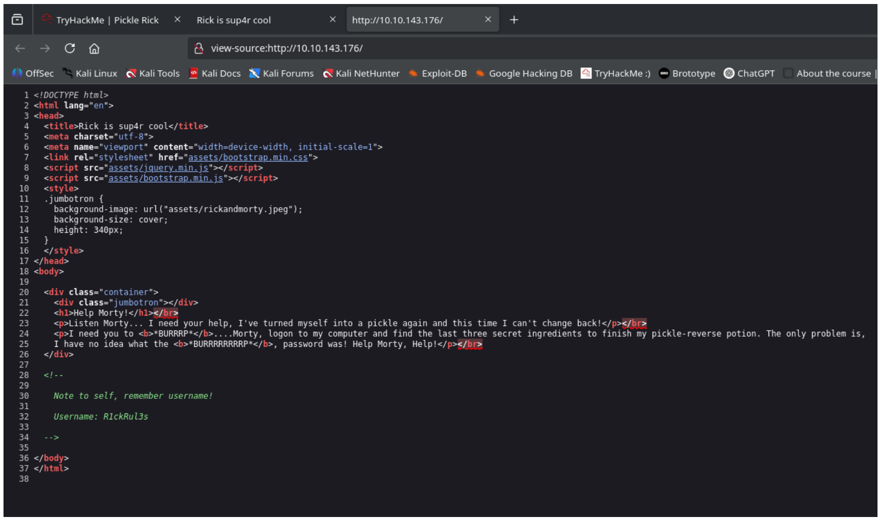
Help Morty!

Listen Morty... I need your help, I've turned myself into a pickle again and this time I can't change back!

I need you to ***BURRRP***....Morty, logon to my computer and find the last three secret ingredients to finish my pickle-reverse potion. The only problem is, I have no idea what the ***BURRRRRRRRP***, password was! Help Morty, Help!

4. Inspecting Elements for Clues

Upon inspecting the page elements using the browser's Developer Tools, I discovered a username embedded within the HTML code.

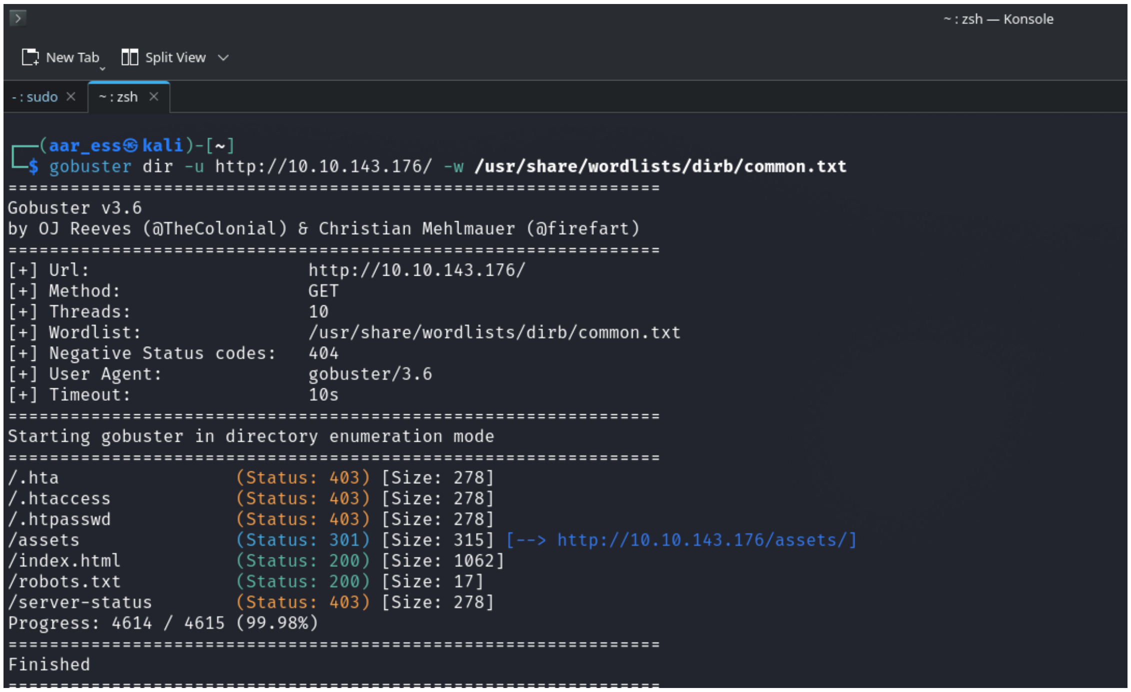


```html
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Rick is sup4r cool</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="assets/bootstrap.min.css">
  <script src="assets/jquery.min.js"></script>
  <script src="assets/bootstrap.min.js"></script>
  <style>
  .jumbotron {
    background-image: url("assets/rickandmorty.jpeg");
    background-size: cover;
    height: 340px;
  }
  </style>
</head>
<body>

  <div class="container">
    <div class="jumbotron"></div>
    <h1>Help Morty!</h1></br>
    <p>Listen Morty... I need your help, I've turned myself into a pickle again and this time I can't change back!</p></br>
    <p>I need you to <b>*BURRRP*</b>....Morty, logon to my computer and find the last three secret ingredients to finish my pickle-reverse potion. The only problem is,
    I have no idea what the <b>*BURRRRRRRRP*</b>, password was! Help Morty, Help!</p></br>
  </div>

  <!--

    Note to self, remember username!

    Username: R1ckRul3s

  -->

</body>
</html>
```

5. Directory Bruteforce with Gobuster

I ran Gobuster on the site to enumerate hidden directories. This revealed several accessible paths such as:

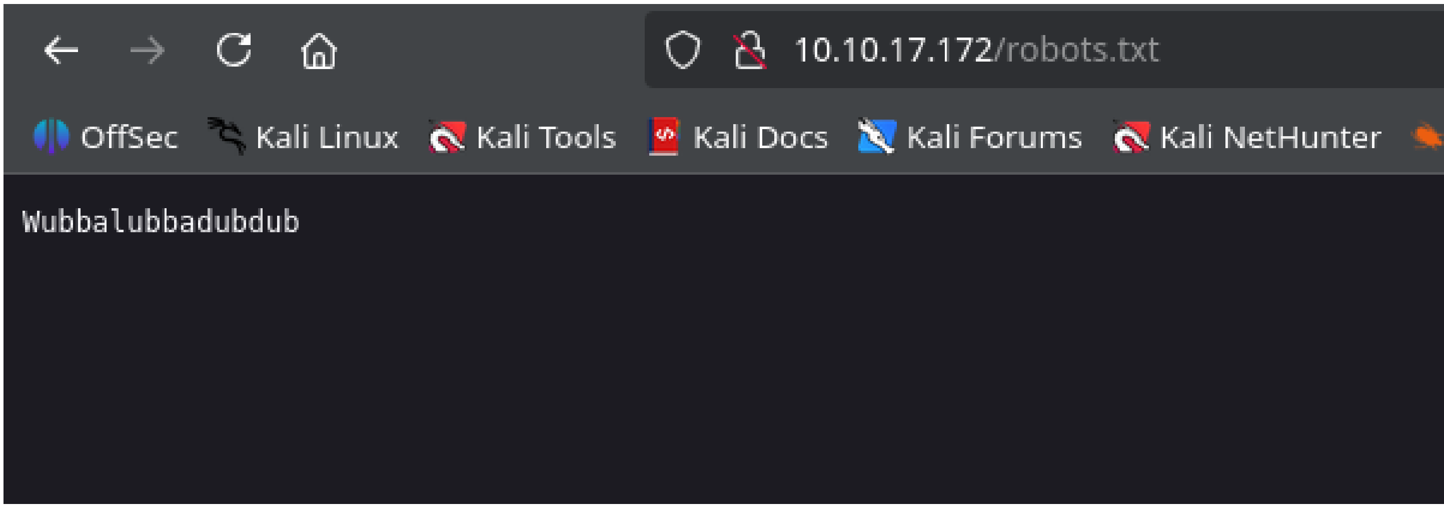
- `/robots.txt`
- `/assets`



```
┌──(aar_ess㉿kali)-[~]
└─$ gobuster dir -u http://10.10.143.176/ -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.143.176/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 278]
/.htaccess            (Status: 403) [Size: 278]
/.htpasswd            (Status: 403) [Size: 278]
/assets               (Status: 301) [Size: 315] [--> http://10.10.143.176/assets/]
/index.html           (Status: 200) [Size: 1062]
/robots.txt           (Status: 200) [Size: 17]
/server-status        (Status: 403) [Size: 278]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

6. Exploring robots.txt

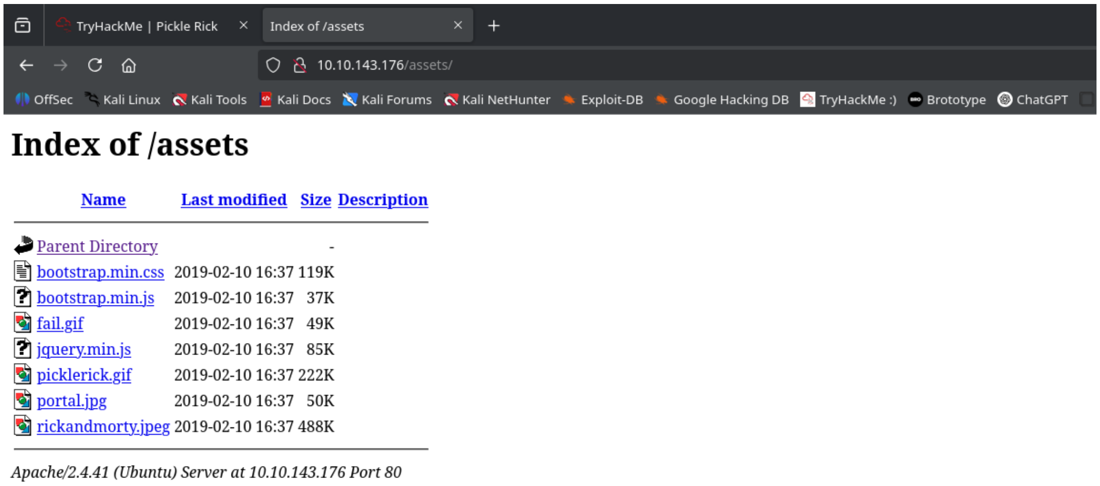
Accessing `/robots.txt` revealed a password, possibly for login or decoding purposes.



```
Wubbalubbadubdub
```

7. Analyzing assets/ Directory

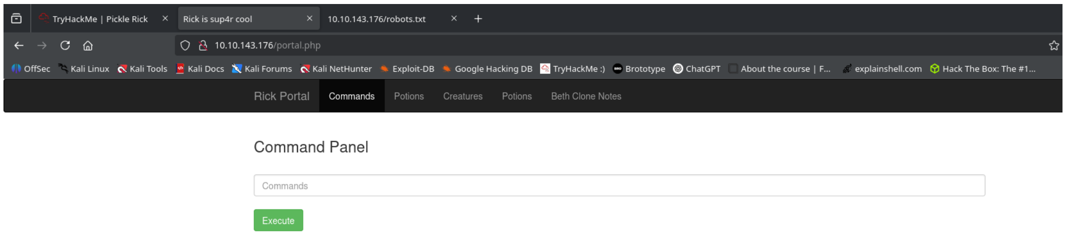
Inside the `/assets` directory, I found several files and images. One of them was `portal.jpg`, which hinted at another path.



```
Index of /assets

      Name             Last modified     Size Description
Parent Directory                           -
bootstrap.min.css  2019-02-10 16:37 119K
bootstrap.min.js   2019-02-10 16:37  37K
fail.gif           2019-02-10 16:37  49K
jquery.min.js      2019-02-10 16:37  85K
picklerick.gif     2019-02-10 16:37 222K
portal.jpg         2019-02-10 16:37  50K
rickandmorty.jpeg  2019-02-10 16:37 488K

Apache/2.4.41 (Ubuntu) Server at 10.10.143.176 Port 80
```

8. Accessing the Portal

Based on the hint from `portal.jpg` found in the `/assets` directory, I predicted the possible existence of a page named `portal.php`. Navigating to `/portal.php` brought me to a login page. I used the username obtained by inspecting the website's HTML source and the password found in `/robots.txt` to successfully log in. Upon logging in, I was redirected to a command panel where I could execute commands on the target machine.



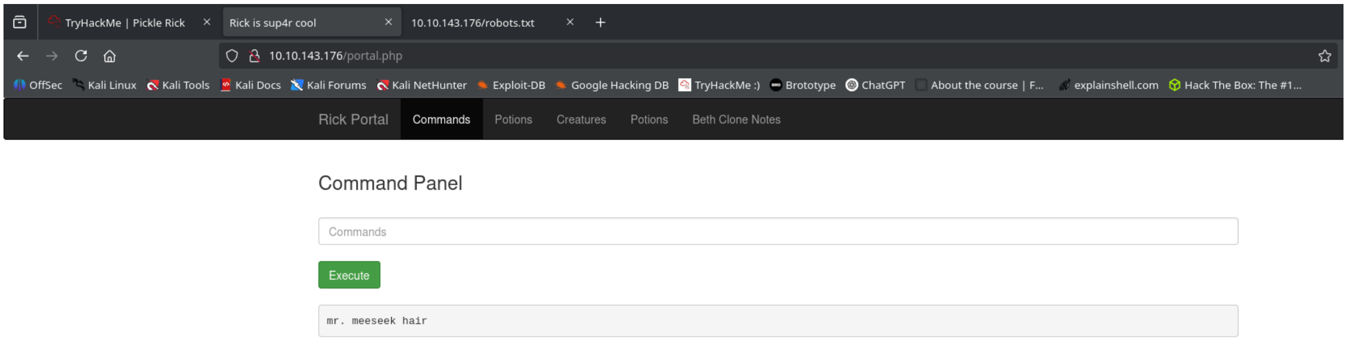
9. Finding the First Ingredient

Inside the portal's command panel, I first ran the `ls` command to list the available files. This showed a file named `Sup3rS3cretPickl3Ingred.txt`. I used the following command to read its contents:

```
less Sup3rS3cretPickl3Ingred.txt
```

This revealed the first ingredient:

1st Ingredient: mr. meeseek hair



```
mr. meeseek hair
```

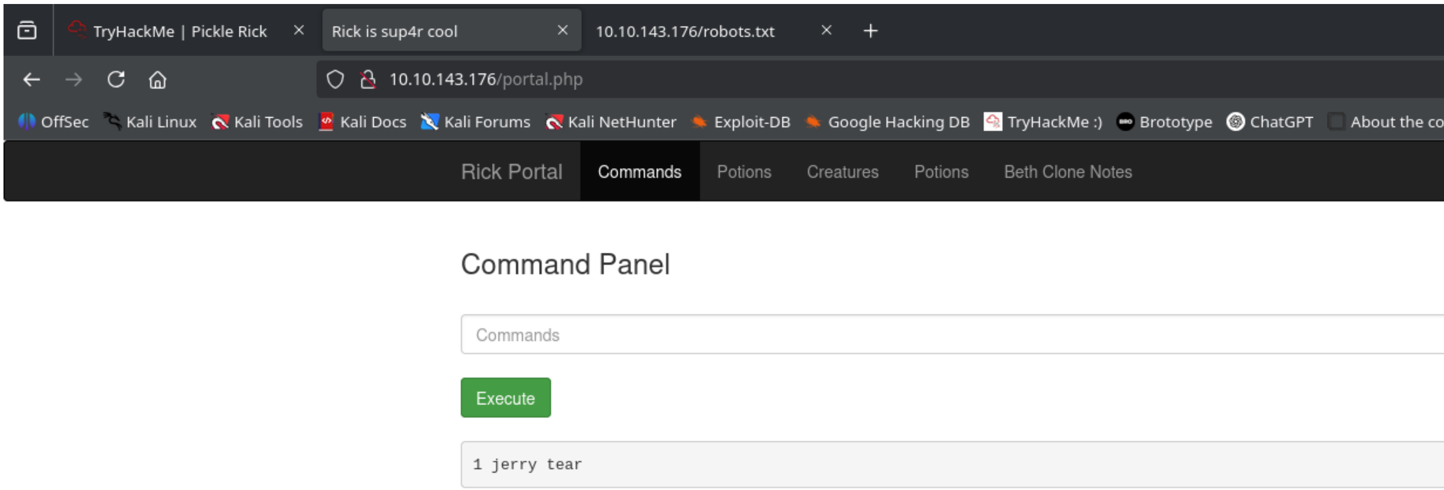
10. Finding the Second Ingredient

I navigated to the home directory and used the following commands to discover and read the second ingredient:

```
ls /home
ls /home/rick
less -l /home/rick/"second ingredients"
```

This revealed the second ingredient:

2nd Ingredient: 1 jerry tear



Command Panel

```
1 jerry tear
```

11. Finding the Third Ingredient (Root Access)

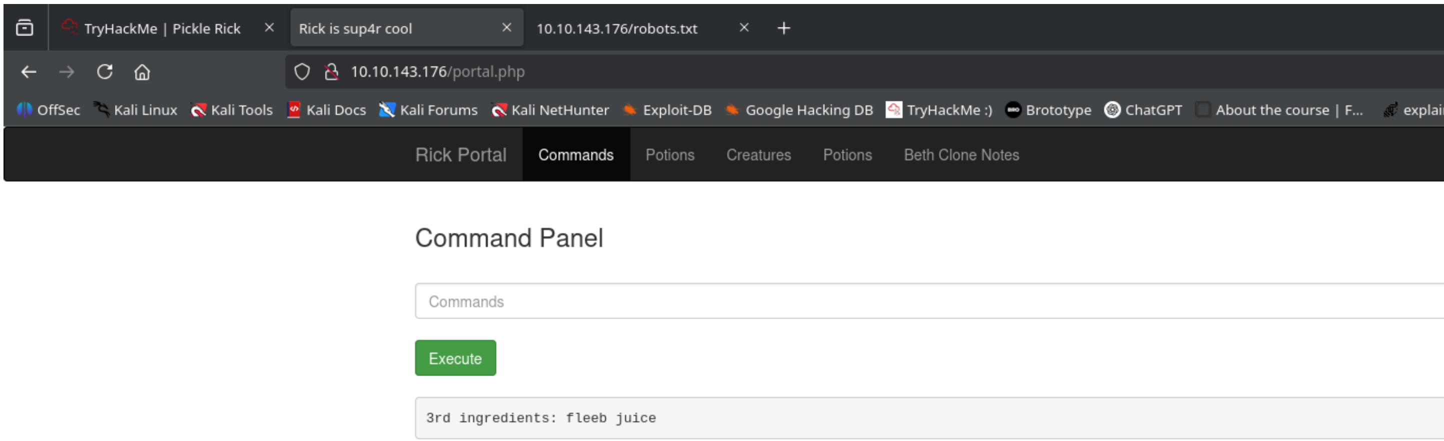
To search for the third ingredient, I checked if I had any 'sudo' privileges. Running:

```
sudo -l
```

showed that I could run commands as root without a password. Therefore, I accessed the root directory directly using:

```
sudo less /root/3rd.txt
```

Inside the file `3rd.txt`, I found the third and final ingredient: **fleeb juice**.



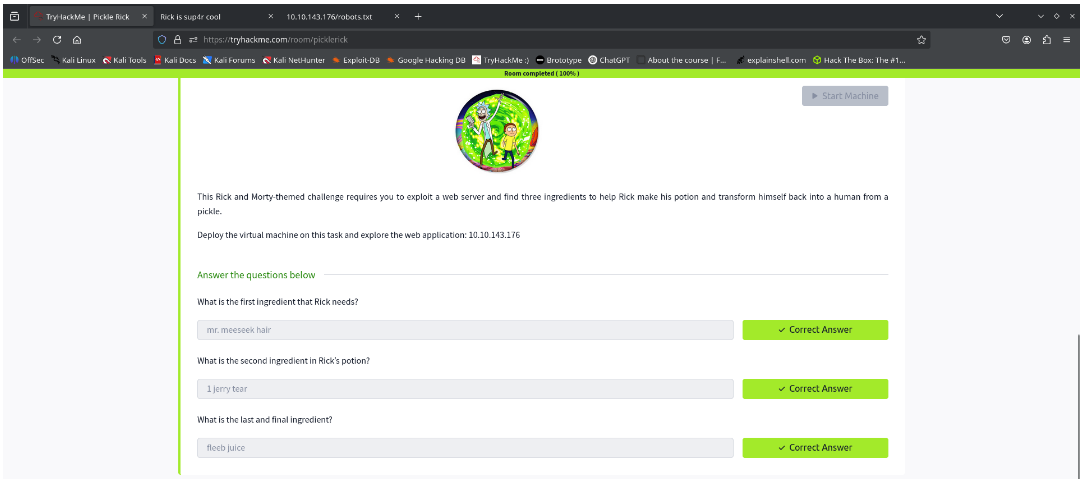
Command Panel

```
3rd ingredients: fleeb juice
```

12. Submitting the Ingredients

After collecting all three ingredients, I returned to the TryHackMe room interface and submitted them in the corresponding answer fields:

- **1st Ingredient:** mr. meeseek hair
- **2nd Ingredient:** 1 jerry tear
- **3rd Ingredient:** fleeb juice



This Rick and Morty-themed challenge requires you to exploit a web server and find three ingredients to help Rick make his potion and transform himself back into a human from a pickle.

Deploy the virtual machine on this task and explore the web application: 10.10.143.176

Answer the questions below

What is the first ingredient that Rick needs?

mr. meeseek hair — ✓ Correct Answer

What is the second ingredient in Rick's potion?

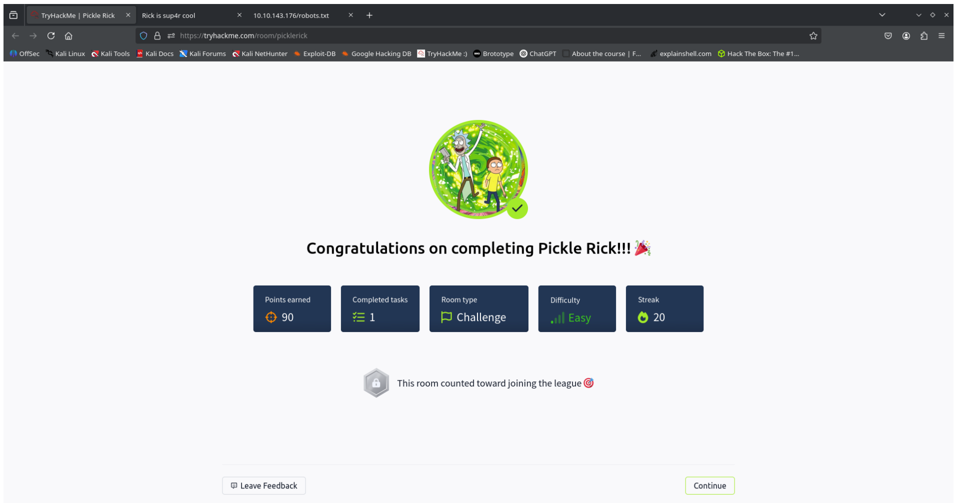
1 jerry tear — ✓ Correct Answer

What is the last and final ingredient?

fleeb juice — ✓ Correct Answer

13. Room Completed

After submitting all the correct ingredients, the platform validated my answers. Each submission field displayed a green checkmark indicating correctness. Finally, the room was marked as complete, confirming successful completion of the challenge.



Congratulations on completing Pickle Rick!!!

Points earned	Completed tasks	Room type	Difficulty	Streak
90	1	Challenge	Easy	20

This room counted toward joining the league

14. Tools Used

- OpenVPN – to connect to the THM network
- Nmap – for port scanning
- Browser Developer Tools – for inspecting HTML elements
- Gobuster – for brute-forcing hidden paths

15. Conclusion

This challenge provided hands-on practice with basic enumeration, web inspection, and privilege escalation. It reinforced the importance of exploring hidden files and using system permissions effectively. All three ingredients were found, and the room was successfully completed.