TryHackMe – Pickle Rick CTF Write-Up

Name: Sidhardh D
Platform: TryHackMe
Room: Pickle Rick
Target IP: `10.10.110.194`
Difficulty: Easy
Tags: Web, Enumeration, Linux, Privilege Escalation, RCE

Overview

The goal of this CTF is to recover three hidden ingredients that Rick needs to turn back into a human. The box is based on a simple Linux web server with several hidden directories, clues in HTML source code, command injection, and basic privilege escalation via sudo misconfiguration.

Tools Used

Tool	Use Case
`nmap`	Port and service scanning
`gobuster`	Directory brute-forcing
Firefox	Manual inspection of webpages
Bash Shell	Interaction with the target shell
`sudo`, `less`	Escalation and file access

Step 1: Port Scanning with Nmap

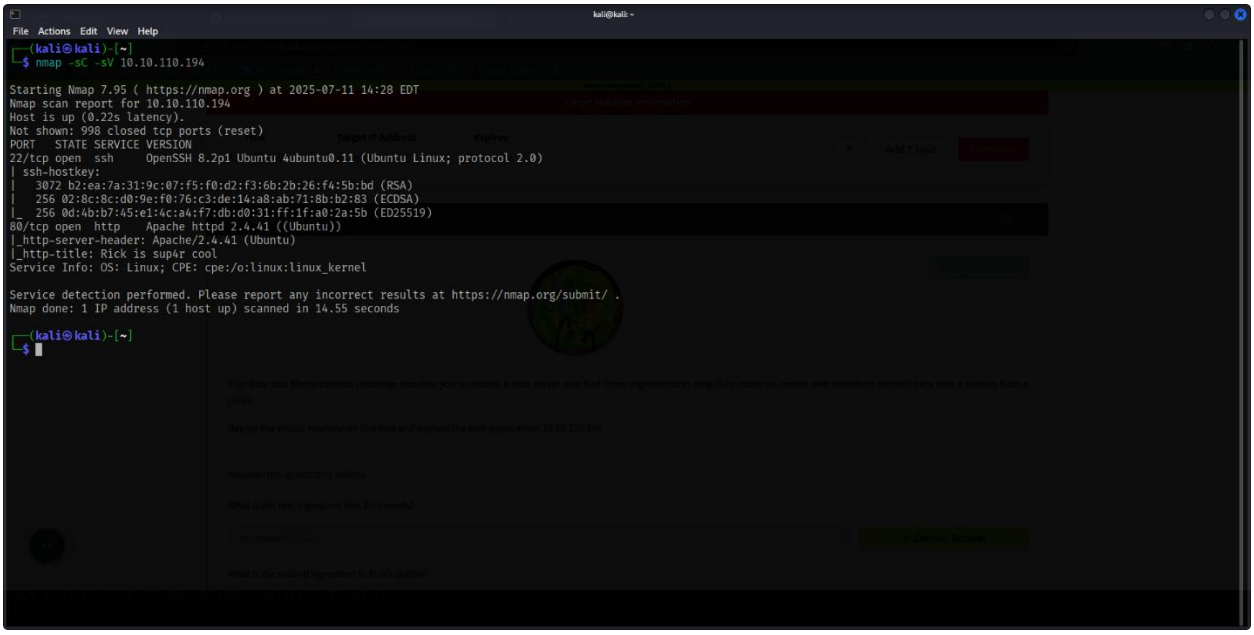
To begin, I scanned the target IP for open ports and running services using Nmap:

```
nmap -sC -sV 10.10.110.194
```

This command performs a default script scan (`-sC`) and attempts to determine service versions (`-sV`).

Scan Output:

```
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu
80/tcp open  http    Apache httpd 2.4.41 (Ubuntu)
```



The SSH port is open, but without credentials it's inaccessible at this point. The HTTP port is of interest and likely holds our initial foothold.

Step 2: Manual Web Exploration

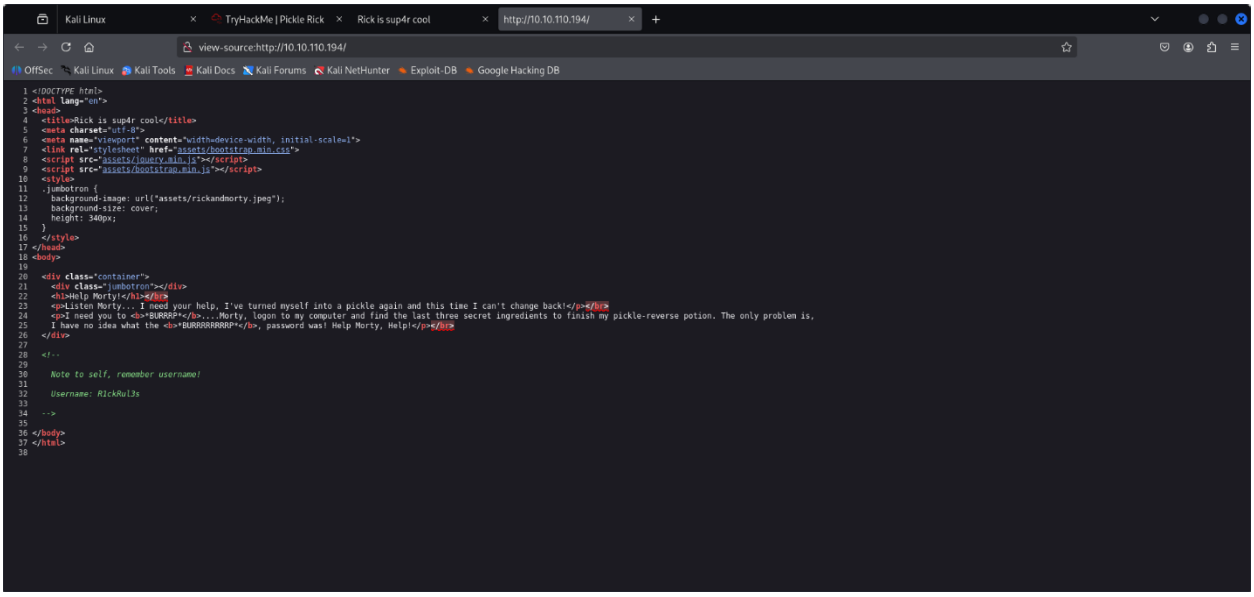
Navigating to the IP in a browser:

```
http://10.10.110.194
```

The homepage shows a Rick and Morty-themed message and nothing interactive. I performed basic checks like hovering over elements and looking for hidden input boxes but found none.

Next, I viewed the page source using the browser (`Ctrl + U`) and found a hardcoded comment:

```
Username: RickRul3s --
```



This is a clear hint that a login function exists somewhere and gives us the username upfront.

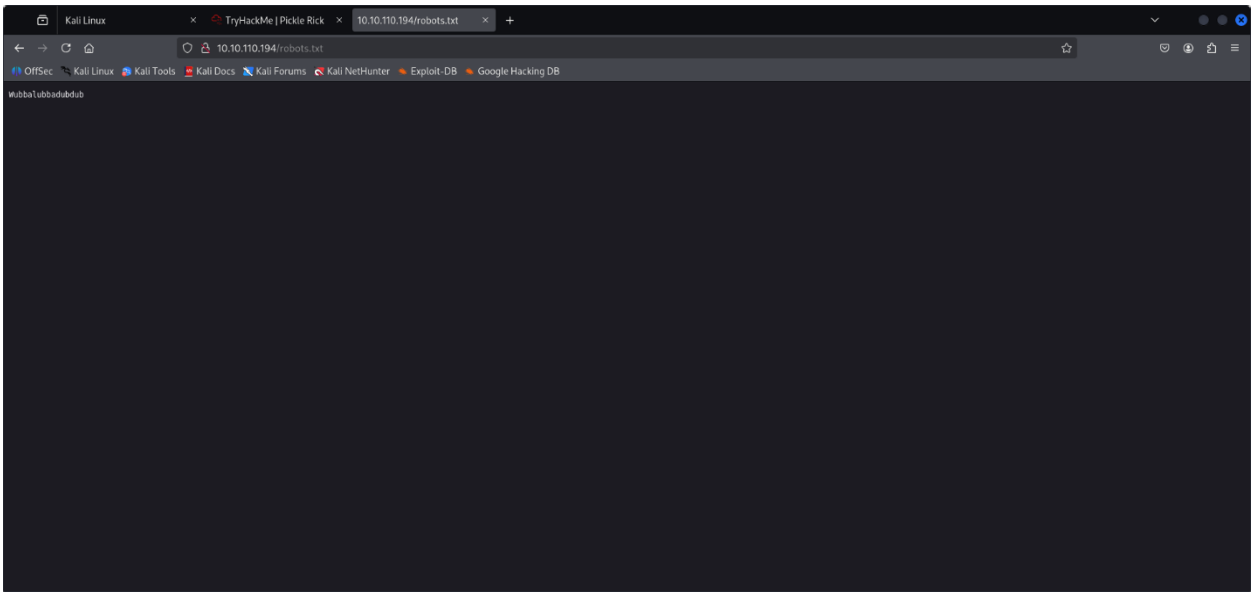
Step 3: Finding Password via robots.txt

A common misconfiguration is leaving sensitive information in `robots.txt`. I navigated to:

`http://10.10.110.194/robots.txt`

It contained the line:

`Wubbalubbadubdub`



Given the format, it's likely the password corresponding to the username found earlier.

So far, we have:

- **Username:** RickRul3s
- **Password:** Wubbalubbadubdub

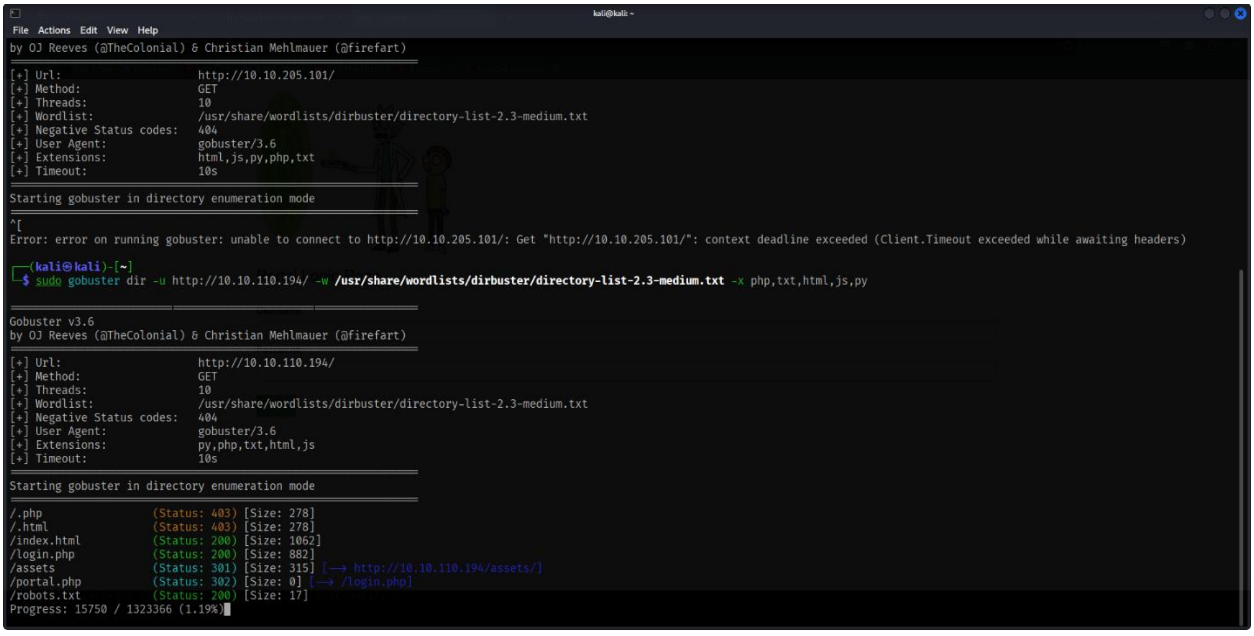
Step 4: Discovering Hidden Pages with Gobuster

To locate hidden directories or files, I ran Gobuster:

```
sudo gobuster dir -u http://10.10.110.194/ \-w
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt \
-x php,txt,html
```

Results:

- `/login.php` – likely the login page
- `/portal.php` – potentially a restricted or post-login page
- `/robots.txt` – already accessed
- `/assets/` – usually static files like CSS or images

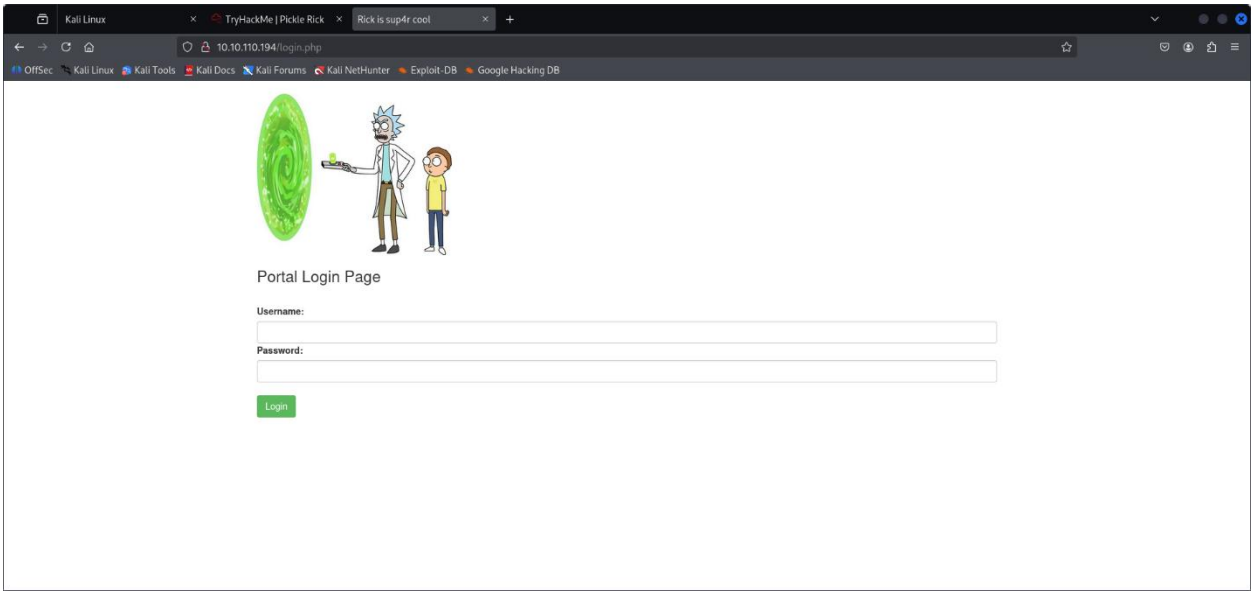


This confirmed that `/login.php` is where we can try the credentials.

Step 5: Logging In

I visited:

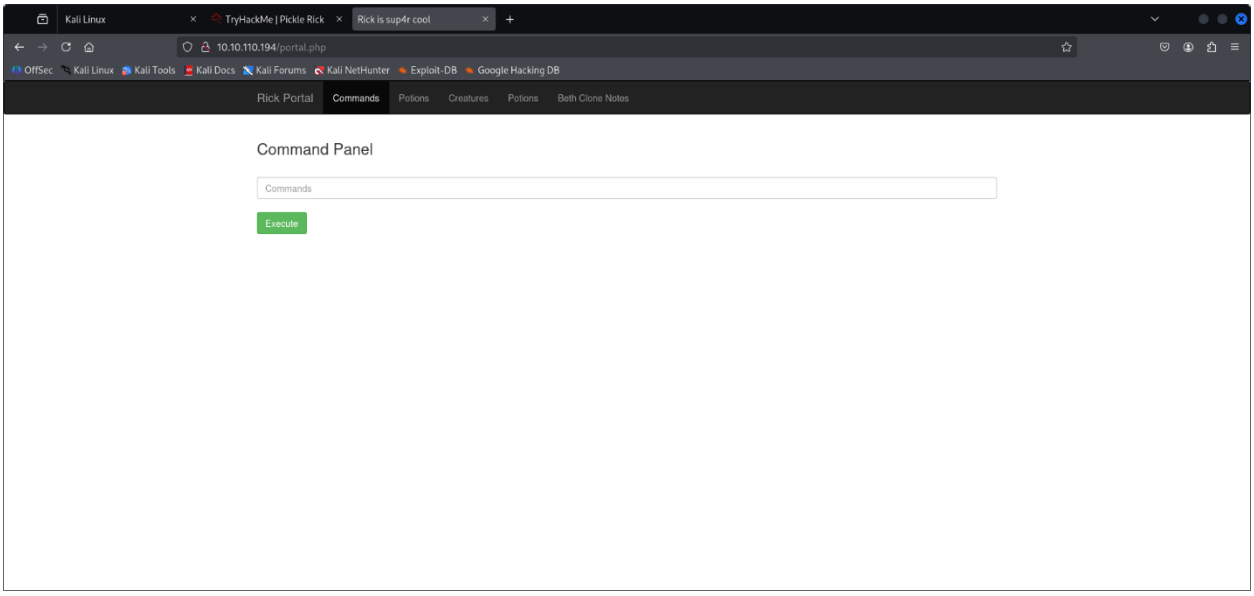
`http://10.10.110.194/login.php`



and entered the known credentials:

- **Username:** RickRul3s
- **Password:** Wubbalubbadubdub

The login succeeded and redirected me to `/portal.php`, which contains a simple form allowing command input.



Step 6: Exploring the Portal Command Interface

This page has a single-line text field which appears to execute commands on the server. I verified functionality with:

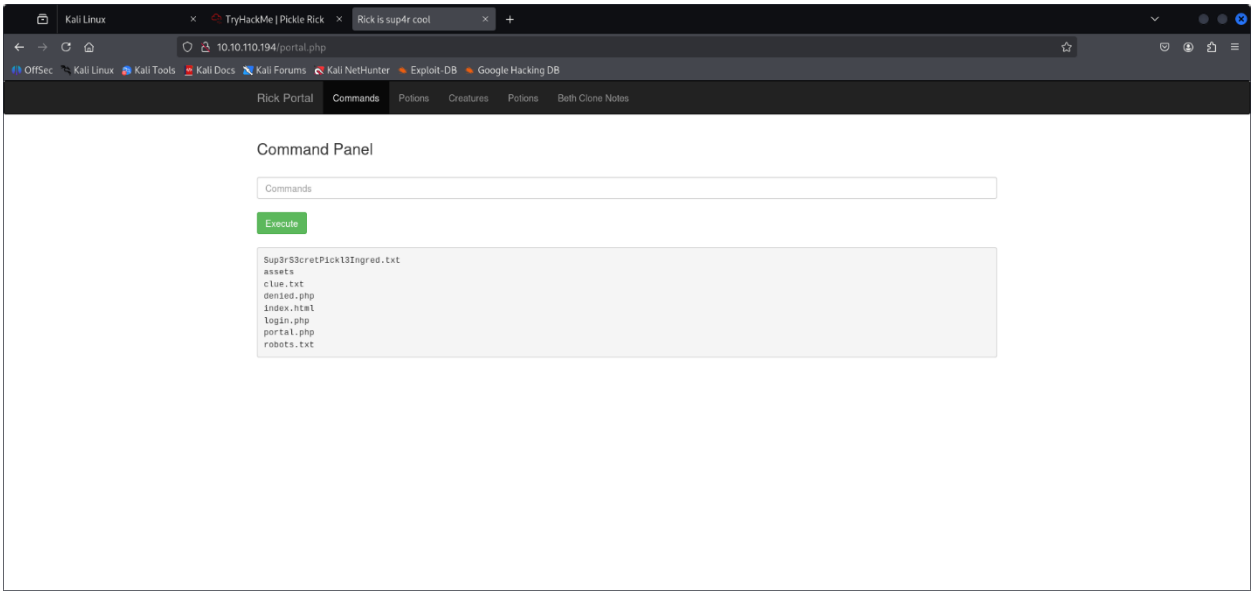
```
whoami
```

Result:

```
www-data
```

Then I listed directory contents:

```
ls
```



I saw the file `Sup3rS3cretPickl3Ingred.txt`.

I first attempted to use `cat`:

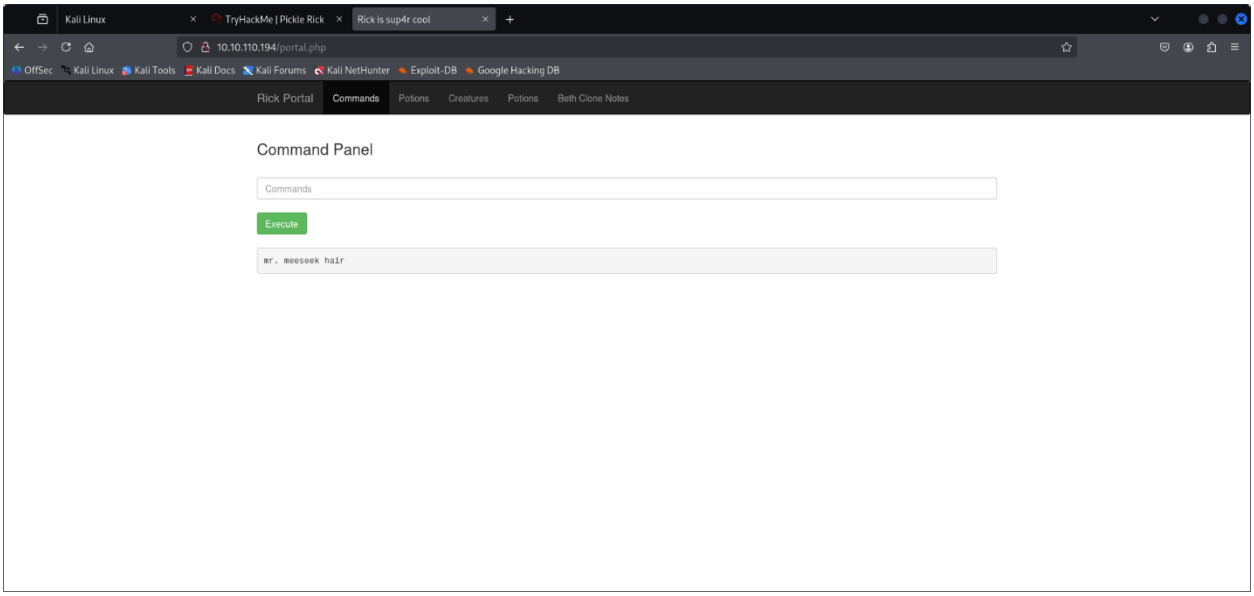
```
cat Sup3rS3cretPickl3Ingred.txt
```

It returned:

```
Command disabled
```

Then I tested `less`, which worked:

```
less Sup3rS3cretPickl3Ingred.txt
```



🔑 **Ingredient 1:** `mr. meeseek hair`

Step 7: Reading the Clue

Listing files again, I found:

`clue.txt`

Using the working `less` command:

```
less clue.txt
```

Content:

> "Look around the file system for the other ingredients."

This was a useful hint. It told me that the remaining ingredients aren't here in the web root but somewhere else in the file system. Based on this, I tried navigating to common system directories.

Step 8: Exploring User Home Directory

Since user data often resides in `/home`, I tried:

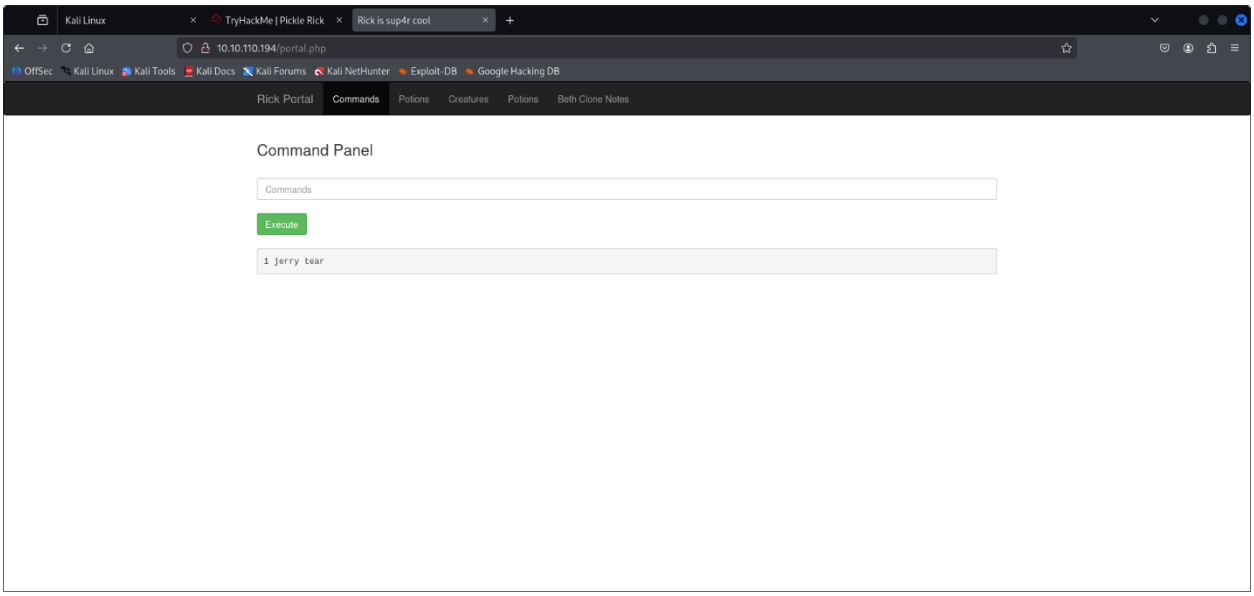
```
cd /home
ls
cd rick
ls
```

I found a file called:

`second ingredients`

Accessing it with:

```
less "second ingredients"
```



🔑 **Ingredient 2:** `1 jerry tear`

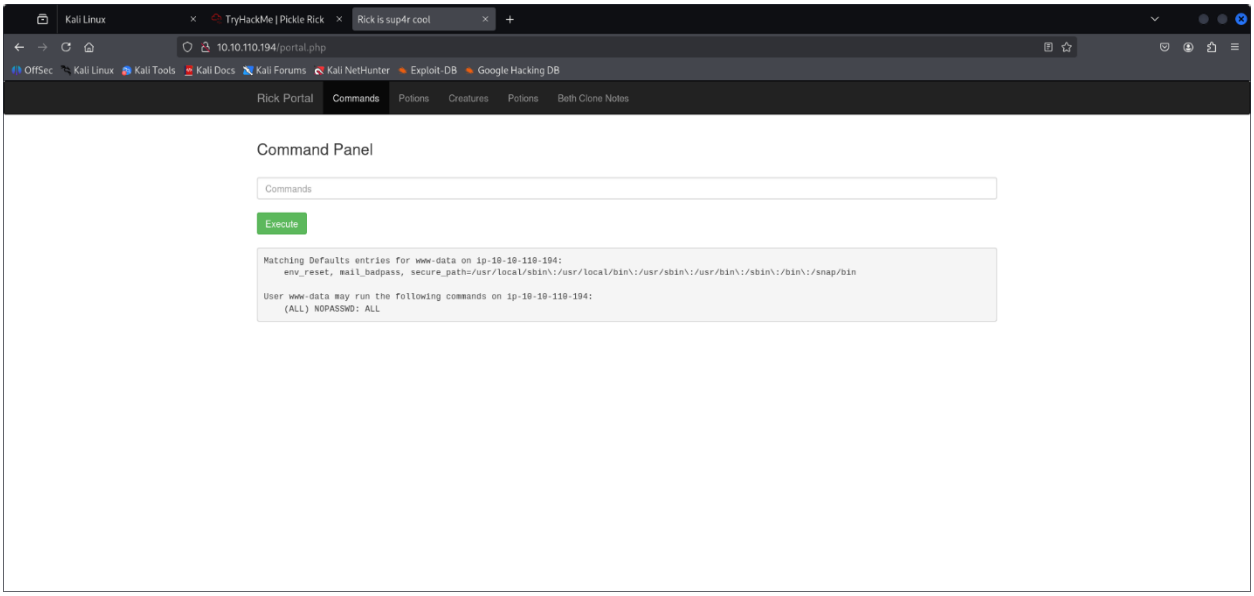
Step 9: Privilege Escalation

Next, I tested for elevated permissions:

```
sudo -l
```

It returned:

`(ALL) NOPASSWD: ALL`

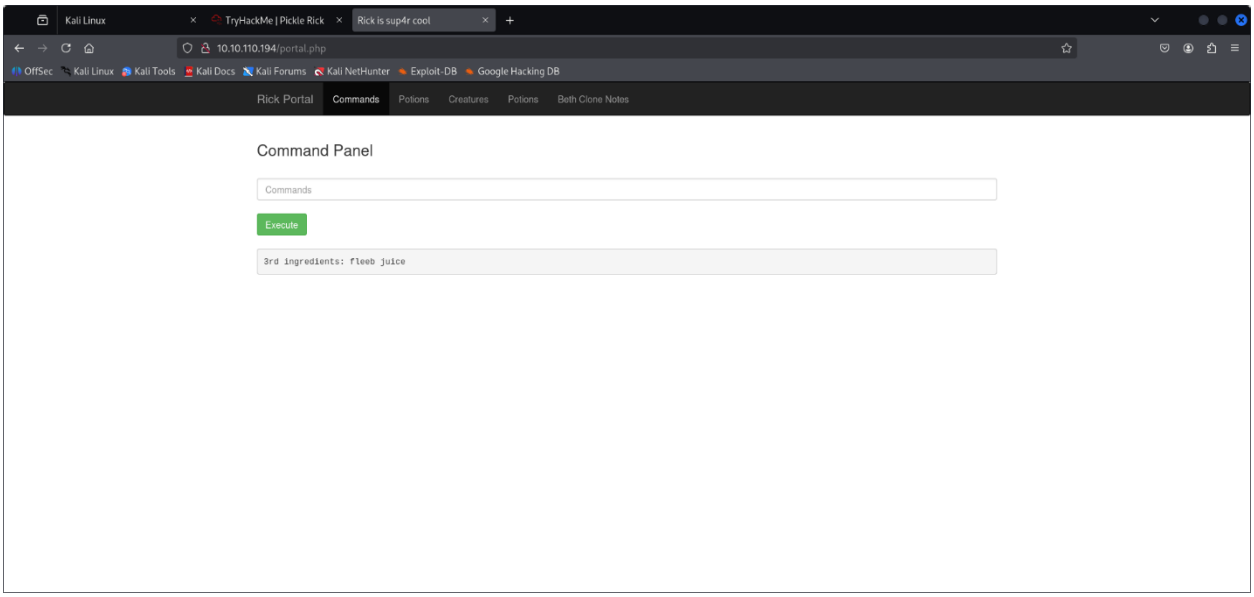


This confirmed I could run any command as root without needing a password. I escalated to root using:

```
sudo -i
```

Once as root, I checked:

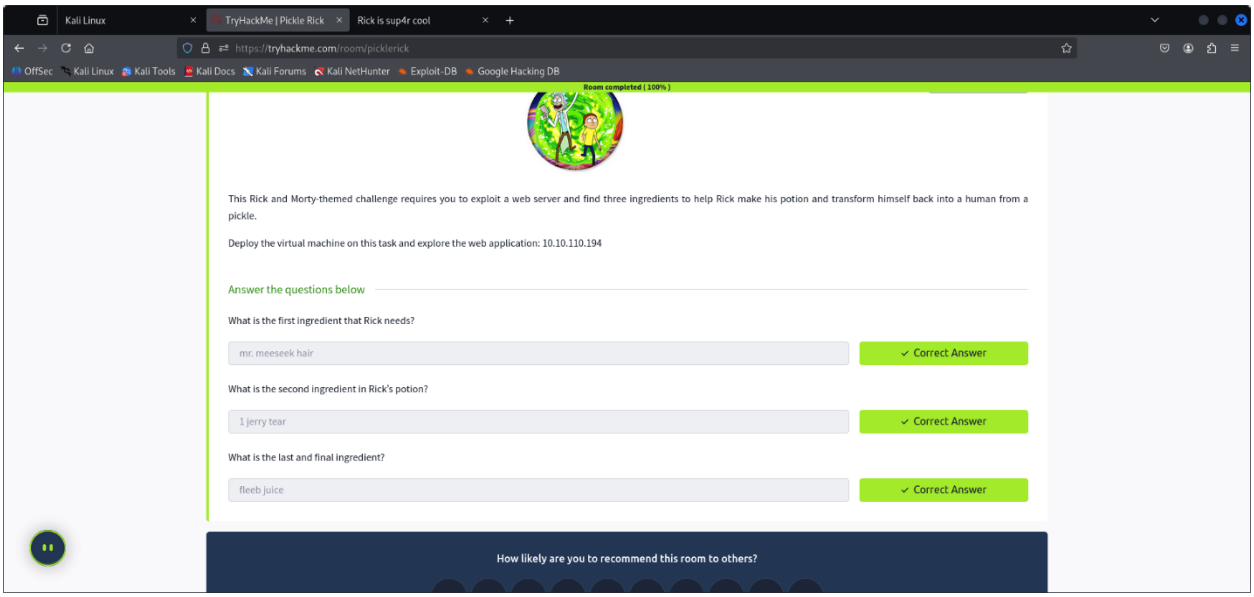
```
cd /root
ls
less 3rd.txt
```



🔑 **Ingredient 3:** `fleeb juice`

Final Summary

Ingredient	Value
1	mr. meeseek hair
2	1 jerry tear
3	fleeb juice



This concludes the Pickle Rick CTF. The room was simple but instructive in web enumeration, dealing with restricted shell environments, and basic privilege escalation via misconfigured sudo rules.