Start with a comprehensive nmap scan of the domain

nmap -Pn -sC -sV 10.10.208.44

- Pn
- sC
- sV

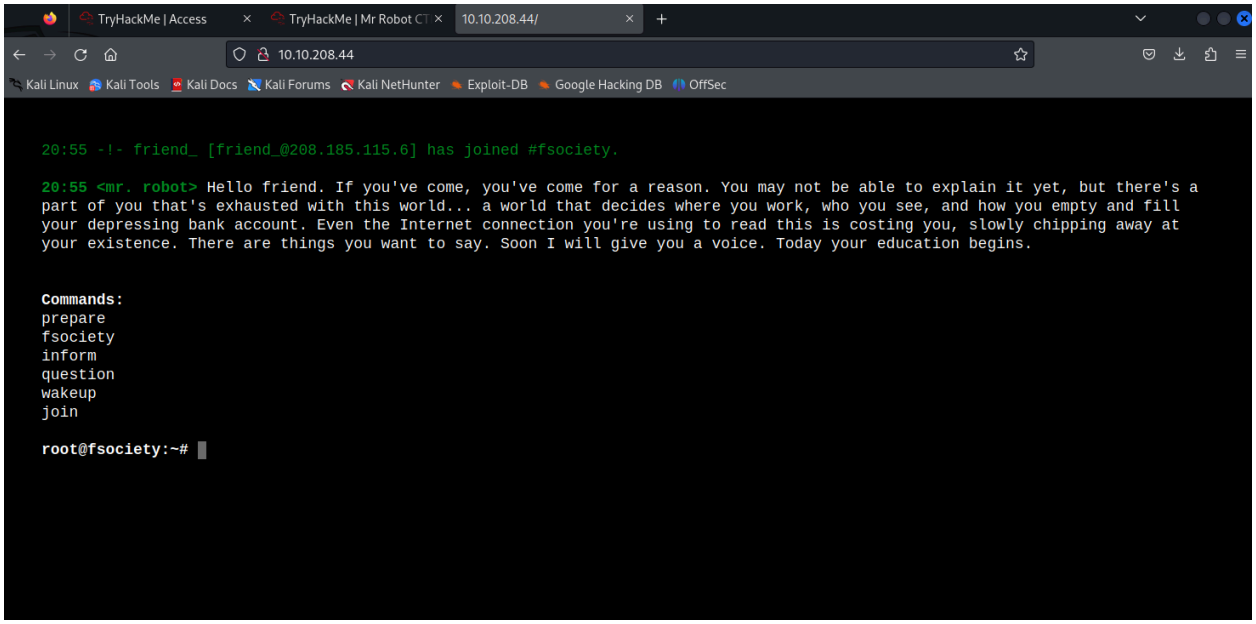
```
Starting Nmap 7.94 ( https://nmap.org ) at 2024-06-13 20:54 EDT
Stats: 0:00:20 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 0.00% done
Stats: 0:00:33 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.65% done; ETC: 20:54 (0:00:00 remaining)
Nmap scan report for 10.10.208.44
Host is up (0.17s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE  SERVICE  VERSION
22/tcp  closed ssh
80/tcp  open   http     Apache httpd
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache
443/tcp open   ssl/http Apache httpd
|_http-title: Site doesn't have a title (text/html).
| ssl-cert: Subject: commonName=www.example.com
| Not valid before: 2015-09-16T10:45:03
|_Not valid after:  2025-09-13T10:45:03
|_http-server-header: Apache

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 36.31 seconds

┌──(kali㉿kali)-[~]
└─$
```

Noticed that http on port 80 is open.

Type in http://10.10.208.44



Promptly leads to this site here. Where you can type in the commands but these don't do anything directly. My first step was to scan for url directories using gobuster.
(I also cloned in the following repository to get the necessary wordlists needed for the gobuster command https://github.com/danielmiessler/SecLists.git)

Gobuster is a tool used to bruteforce files and directories on web servers.
The following command I used is
gobuster dir -u http://10.10.208.44 -w SecLists/Discovery/Web-Content/common.txt

This basically checks to see if there are any active directories on the site and my goal was to look for anything that might've looked suspicious. After the gobuster scan i noticed /robots.txt going to this site led me to the first key.

```
/image                (Status: 301) [Size: 0] [--> http://10.10.208.44/image/]
/images               (Status: 301) [Size: 235] [--> http://10.10.208.44/images/]
/index.html           (Status: 200) [Size: 1188]
/index.php            (Status: 301) [Size: 0] [--> http://10.10.208.44/]
/js                   (Status: 301) [Size: 231] [--> http://10.10.208.44/js/]
/intro                (Status: 200) [Size: 516314]
/license              (Status: 200) [Size: 309]
/login                (Status: 302) [Size: 0] [--> http://10.10.208.44/wp-login.php]
/page1                (Status: 301) [Size: 0] [--> http://10.10.208.44/]
/phpmyadmin           (Status: 403) [Size: 94]
/rdf                  (Status: 301) [Size: 0] [--> http://10.10.208.44/feed/rdf/]
/readme               (Status: 200) [Size: 64]
/render/https://www.google.com (Status: 301) [Size: 0] [--> http://10.10.208.44/render/https:/www.google.com]
/robots.txt           (Status: 200) [Size: 41]
/robots               (Status: 200) [Size: 41]
/rss                  (Status: 301) [Size: 0] [--> http://10.10.208.44/feed/]
/rss2                 (Status: 301) [Size: 0] [--> http://10.10.208.44/feed/]
/sitemap              (Status: 200) [Size: 0]
/sitemap.xml          (Status: 200) [Size: 0]
/video                (Status: 301) [Size: 234] [--> http://10.10.208.44/video/]
/wp-admin             (Status: 301) [Size: 237] [--> http://10.10.208.44/wp-admin/]
/wp-content           (Status: 301) [Size: 239] [--> http://10.10.208.44/wp-content/]
/wp-cron              (Status: 200) [Size: 0]
/wp-config            (Status: 200) [Size: 0]
/wp-includes          (Status: 301) [Size: 240] [--> http://10.10.208.44/wp-includes/]
```

TryHackMe | Access × TryHackMe | Mr Robot CT × 10.10.208.44/robots.txt ×

10.10.208.44/robots.txt

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

```
User-agent: *
fsocity.dic
key-1-of-3.txt
```

TryHackMe | Access × TryHackMe | Mr Robot CT × 10.10.208.44/key-1-of-3.txt ×

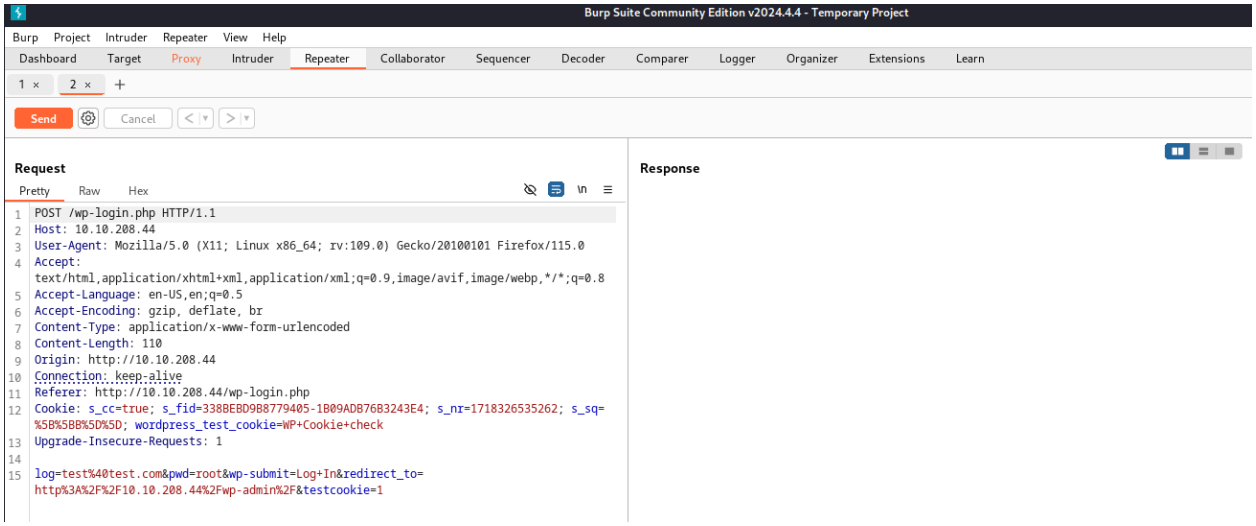
10.10.208.44/key-1-of-3.txt

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

```
073403c8a58a1f80d943455fb30724b9
```

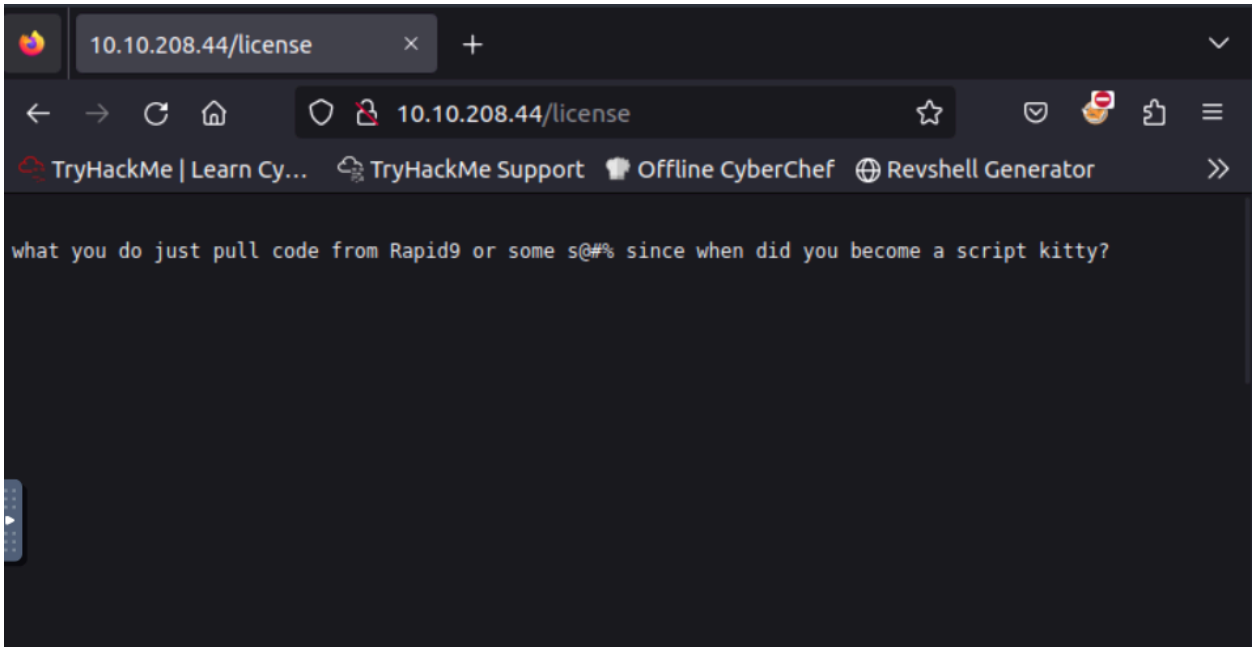
Next I went to the login page. I captured the proxy for the login through burp suite. The sent it to the repeater to see what I could manipulate.

The other thing I noticed was the fsocity.dic file and I immediately assumed that this is some type of password list.

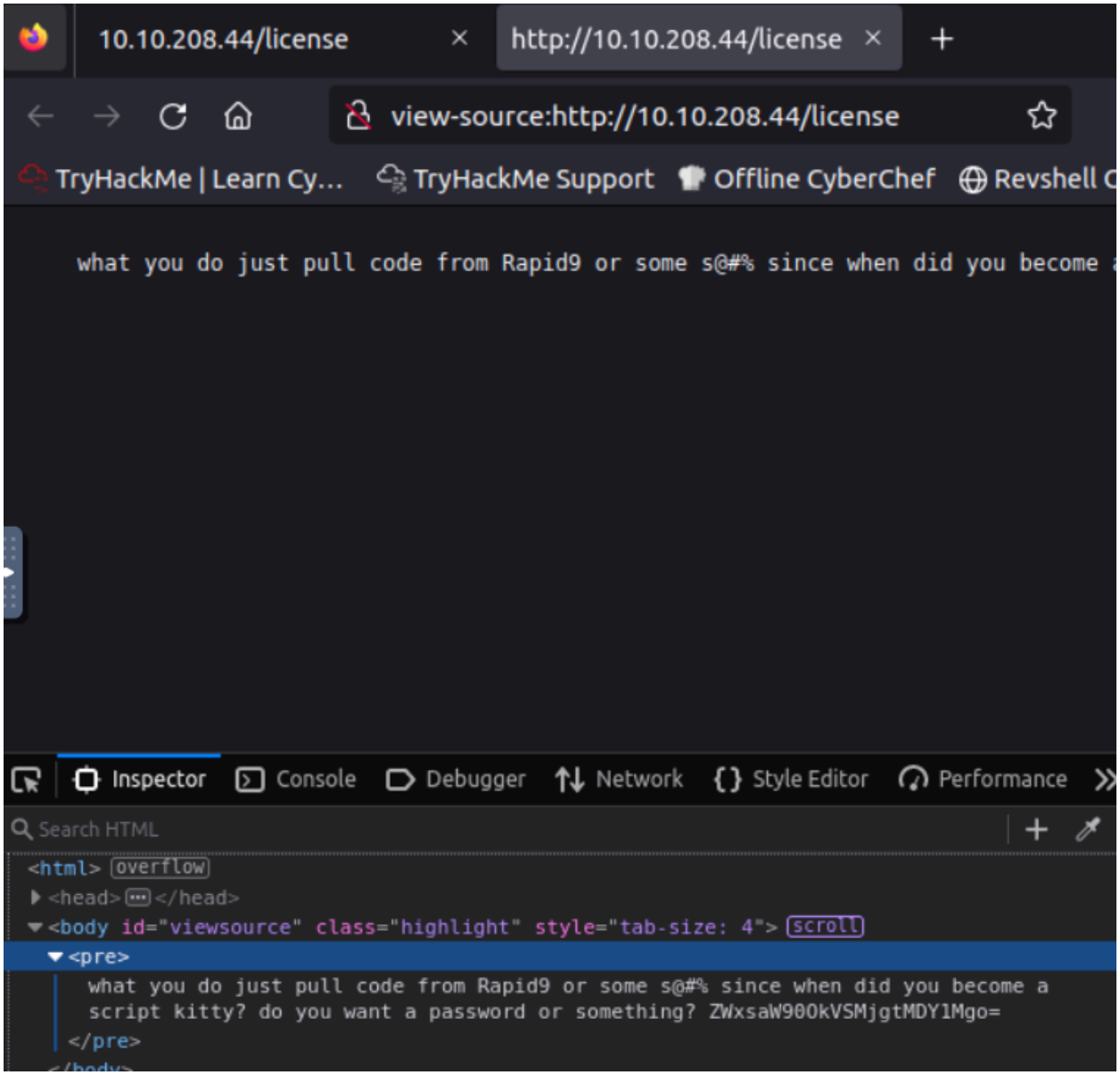


From here I got stuck so I had to use the walkthrough guidelines to see that I needed to use hydra to brute force the password and username.

I was guided to the /license directory which I had also scanned in gobuster.



Upon page inspection you get an encoded key.



I put this into cyberchef which gave me a username and a password: elliot:ER28-0652
I went back to the login page and put in this info.

From here I put a reverse shell payload into the editor appearance and was able to login to a user named daemon. We can download bin/bash for an interactive terminal.
I needed to be a robot however to access the flag. Daemon does have access to a hash and it was stated to be md5.

I ran this hashcat command to decrypt the hash within powershell :
./hashcat -m 0 -a 0 "c3fcd3d76192e4007dfb496cca67e13b" rockyou.txt

This gave me the password was abcdefghijklmnopqrstuvwxyz and I was able to log into the robot user and then retrieve the contents to key 2 of 3