安全-eval（BugkuCTF）

原创

小狐狸FM 于 2021-07-14 22:29:21 发布 577 收藏 7

分类专栏： 安全 # CTF夺旗 文章标签： php ctf eval

本文链接：https://blog.csdn.net/smallfox233/article/details/118514806

安全 同时被 2 个专栏收录

91 篇文章 8 订阅

订阅专栏

CTF夺旗

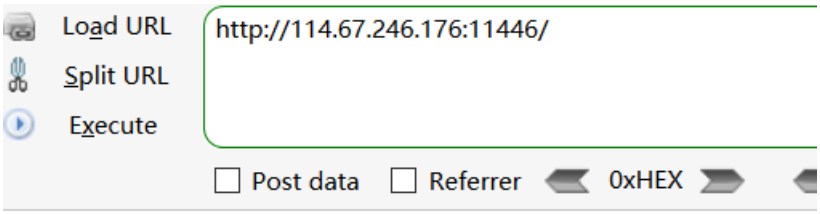
38 篇文章 0 订阅

订阅专栏

文章目录

一、题目

原题链接




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


二、WriteUp

> - 在代码中先是通过 `include` 包含了 `flag.php` 文件
> - 然后使用了 `$REQUEST['hello']` 来获取用户传给服务器的 `hello` 参数

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

> - `var_dump()` 会将传入的变量打印到页面，`eval()` 会执行传入到其中的 `php` 代码，`eval("var_dump($a);")` 就是把 `$a` 的内容打印出来
> - PHP:var_dump - Manual
>   PHP var_dump() 函数-W3Cschool

语法

```
void var_dump ( mixed $expression [, mixed $... ] )
```

此函数显示关于一个或多个表达式的结构信息，包括表达式的类型与值。数组将递归展开值，通过缩进显示其结构。

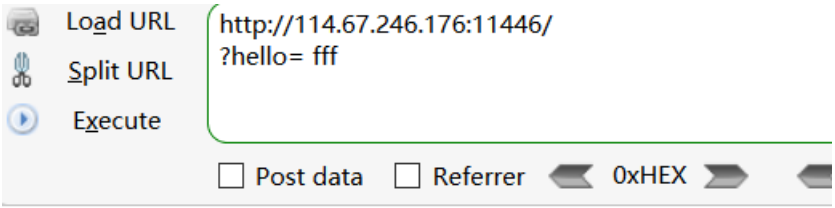
> **Tip** 和直接将结果输出到浏览器一样，可使用输出控制函数来捕获当前函数的输出，然后(例如)保存到一个 string 中。

参数	描述
expression	你要打印的变量。

技术细节

返回值:	没有返回值。
PHP 版本:	PHP 4, PHP 5, PHP 7

给 `hello` 赋值，执行后会将变量的类型和内容打印出来




```
string(3) "fff" <?php
        include  "flag.php";
        $a  =  @$_REQUEST['hello'];
        eval(  "var_dump($a);");
        show_source(__FILE__);
?>
```


- 可以使用 `file()` 函数先将 `flag.php` 的内容存入数组，然后这个数组会被 `var_dump()` 和 `eval()` 函数输出到页面
- PHP:file - Manual
  PHP file()函数-W3school

定义和用法

file() 函数把整个文件读入一个数组中。

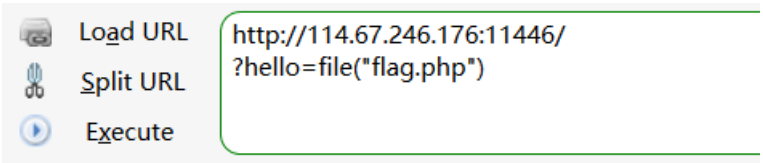
与 file_get_contents() 类似，不同的是 file() 将文件作为一个数组返回。数组中的每个单元都是文件中相应的一行，包括换行符在内。

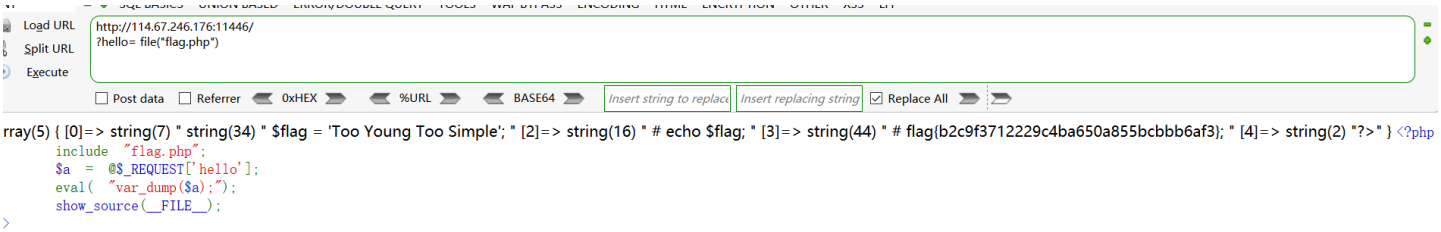
如果失败，则返回 false。

语法

```
file(path,include_path,context)
```

参数	描述
path	必需。规定要读取的文件。
include_path	可选。如果也想在 include_path 中搜寻文件的话，可以将该参数设为 "1"。
context	可选。规定文件句柄的环境。 context 是一套可以修改流的行为的选项。若使用 null，则忽略。





rray(5) { [0]=> string(7) " string(34) " $flag = 'Too Young Too Simple'; " [2]=> string(16) " # echo $flag; " [3]=> string(44) " # flag{b2c9f3712229c4ba650a855bcbbb6af3}; " [4]=> string(2) "?>" } <?php

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