

# 【CTF整理】Who are you (2017强网杯web题)

原创

久违° 于 2020-04-20 10:16:02 发布 682 收藏

分类专栏: [CTF](#) 文章标签: [web](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/weixin\\_42742658/article/details/105627667](https://blog.csdn.net/weixin_42742658/article/details/105627667)

版权



[CTF 专栏收录该内容](#)

7 篇文章 1 订阅

订阅专栏

## 【CTF整理】Who are you (2017强网杯web题)

### 别人思路总结:

#### 0x01 初探

打开网页就是一句“Sorry. You have no permissions.”

按照惯例看看网页源码, 发现没有提示;

#### 0x02 初步思考

既然没有提示, 也没有其他的链接, 那么可能有以下几种可能:

1、敏感文件泄漏

2、跳转

3、cookie / session

第一个想法在经过扫描器扫描之后就放弃了, 因为只看到index.php, 还有/upload/, 但是在访问的时候是403

第二个在抓包的时候也没有看到有跳转

只剩下第三个

#### 0x03 cookie中的role

在查看cookie的时候发现了“Cookie: role=Zjo1OiJ0aHJmZyl7”, 后面那串第一个想法就是base64, 尝试过后得到

"f:5:"thrfg";". 一时间没有看懂这个thrfg是什么, 然后暴力猜测了一下, 发现是guest, 也就是rot-13. 于是把它改成admin的rot13过后的值就进去了。

#### 0x04 upload

进去之后在源码里有提示“<!-- \$filename = \$\_POST['filename']; \$data = \$\_POST['data']; -->”

这里应该是模拟一个文件上传，但是用post方法来弄的。

这里经过一番测试，发现在发送有<的时候会显示no no no。所以猜测源代码中有个正则表达式，用来匹配

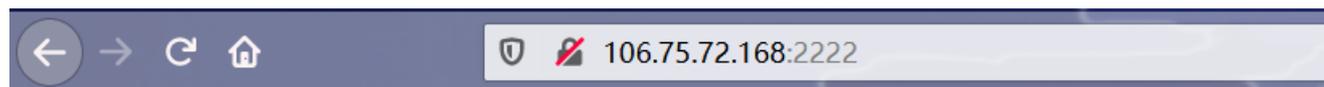
这里我用data[]=的方法，把data从字符串变成数组，导致绕过正则匹配。

上传之后能够发现它返回了文件的地址，访问它就得到flag

## 自己复现

访问：http://106.75.72.168:2222

得到：Sorry. You have no permissions.



Sorry. You have no permissions.

[https://blog.csdn.net/weixin\\_42742658](https://blog.csdn.net/weixin_42742658)

御剑扫描，无法访问

ID	地址
1	<a href="http://106.75.72.168:2222/uploads/">http://106.75.72.168:2222/uploads/</a>
2	<a href="http://106.75.72.168:2222/index.php">http://106.75.72.168:2222/index.php</a>

查看cookie:

-	Name	Value	Domain	Pat Expires / Ma...	Size	Http Ho: Sec	Sec
▼ http://106.75.72.168:2222 (1)							
<input type="button" value="add a new cookie"/>							
<input type="checkbox"/>	role	Zjo10iJ0aHJmZyI7	106.75.72.168		20	✓	✓

Name	<input type="text" value="role"/>	Zjo10iJ0aHJmZyI7
Domain	<input type="text" value="106.75.72.168"/>	
Path	<input type="text" value="/"/>	

Expiration (ISO)

HostOnly  Session  
 Secure  HttpOnly

base64解码cookie:

Base64:

解密Base64:

经过百度得知这玩意叫

ROT13编码:

# ROT13

 本词条由“科普中国”科学百科词条编写与应用工作项目 审核。

**ROT13**（回转13位，**rotate by 13 places**，有时中间加了个连字符称作**ROT-13**）是一种简易的替换式密码。它是一种在英文网络论坛用作隐藏八卦（spoiler）、妙句、谜题解答以及某些脏话的工具，目的是逃过版主或管理员的匆匆一瞥。ROT13被描述成“杂志字谜上下颠倒解答的Usenet点对点体”。ROT13也是过去在古罗马开发的凯撒加密的一种变体。[/blog.csdn.net/weixin\\_42742658](http://blog.csdn.net/weixin_42742658)

然后在线ROT-13解码:

字符串

```
f:5:"thrfg";
```

计算

解码结果

```
s:5:"guest";
```

复制

[https://blog.csdn.net/weixin\\_42742658](https://blog.csdn.net/weixin_42742658)

看到guest改成admin

尝试:



ROT13编码计算器

字符串

```
s:5:"admin";
```

计算

编码结果

```
f:5:"nqzva";
```

复制

[https://blog.csdn.net/weixin\\_42742658](https://blog.csdn.net/weixin_42742658)

对f:5:"nqzva";进行

base64编码得到Zjo1OiJucXp2YSI7, 用于替换原来的cookie

要转的:

f:5:"nqzva";

给我转!

URL格式

%66%3A%35%3A%22%6E%71%7A%76%61%22%3B

还原

SQL\_En:

0x66003A0035003A0022006E0071007A007600610022003B00

还原

Hex:

0x663A353A226E717A7661223B

还原

Asc:

102 58 53 58 34 110 113 122 118 97 34 59

单个还原

MD5\_32:

61B5668616E54453A498E38801A7DD59

MD5\_16:

16E54453A498E388

Base64:

Zjo10iJucXpZYSi7

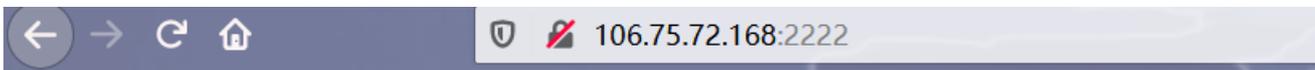
解密 Base64

解密Base64:

f:5:"nqzva";

[https://blog.csdn.net/weixin\\_42742658](https://blog.csdn.net/weixin_42742658)

用新的cookie访问，进入管理员界面

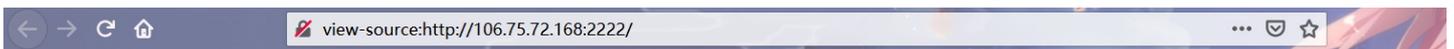


Hello admin, now you can upload something you are easy to forget.

[https://blog.csdn.net/weixin\\_42742658](https://blog.csdn.net/weixin_42742658)

[查看源](#)

码:



```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5 </head>
6 <body>
7 <!-- $filename = $_POST['filename']; $data = $_POST['data']; -->Hello admin, now you can upload something you are easy to forget.</body>
8 </html>
9

```

[https://blog.csdn.net/weixin\\_42742658](https://blog.csdn.net/weixin_42742658)

构造post提交:

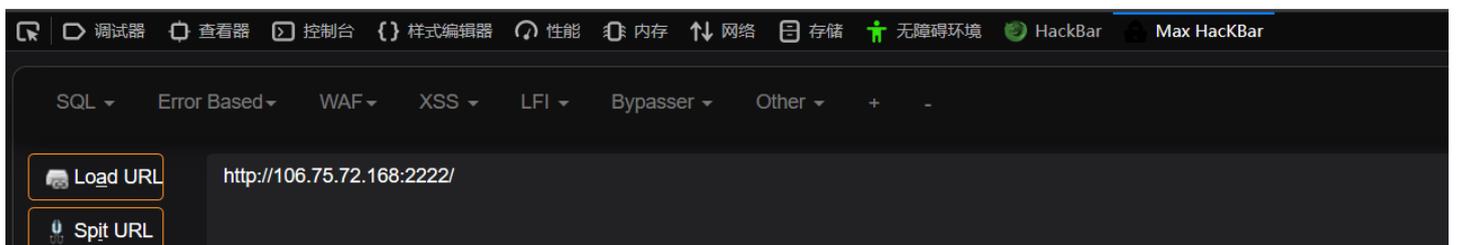
No No No!

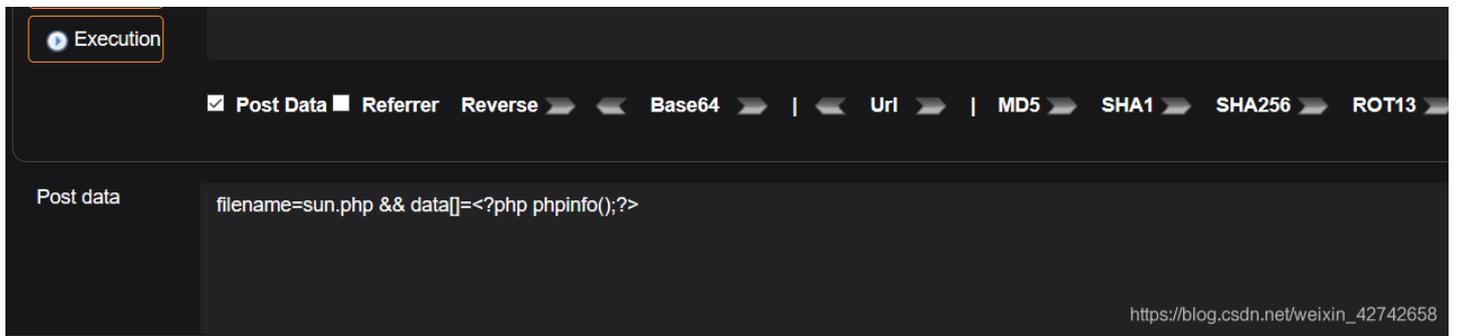


需要绕

过:

your file is in ./uploads/42a430a5f79d46be10e92ed9efd987besun.php





访问返回的网址得到flag:



**flag{e07cd440-8eed-11e7-997d-7efc09eb6c59}**

可能太久了，我加上返回的网址访问不到结果，光保留uploads可以访问。

by 久违 2020.4.19