

BUUCTF-Reverse reverse1

原创

柒熠染云 于 2021-05-19 21:14:44 发布 79 收藏

分类专栏: [逆向工程](#) 文章标签: [反编译](#) [编程语言](#) [信息安全](#) [安全](#)

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/m0_46276093/article/details/117046791

版权



[逆向工程](#) 专栏收录该内容

10 篇文章 1 订阅

订阅专栏

BUUCTF-Reverse reverse1

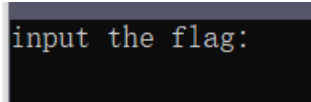
所用软件

1.IDA PRO (静态分析工具)

2.exeinfope (查壳工具)

逆向步骤

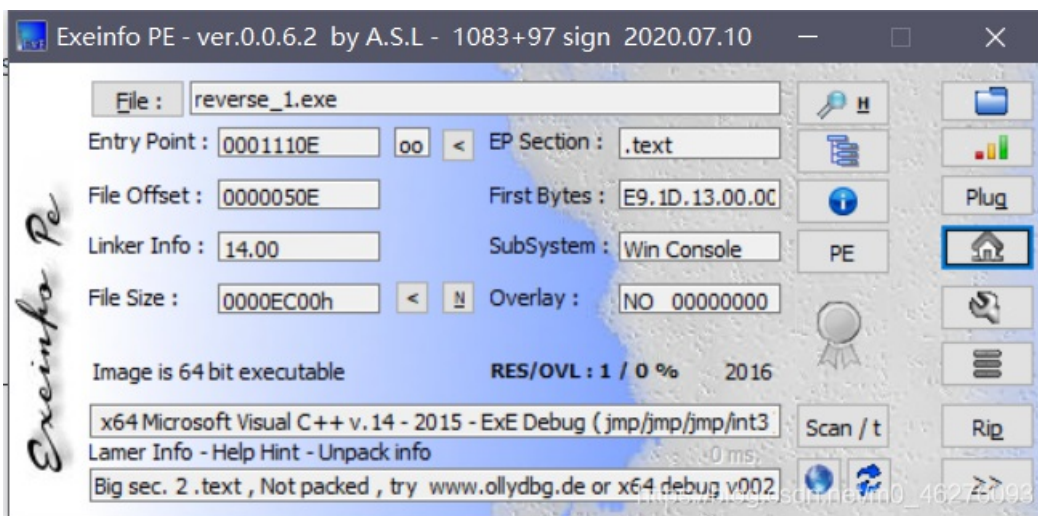
1.下载reverse_1.exe, 并将其打开



```
input the flag:
```

无可用信息, 继续下一步

2.将reverse_1.exe放入exeinfope中进行查壳



可知reverse_1.exe没有加壳。

3.将reverse_1.exe放入IDA中、

摁shift+F12查看程序字符串

```

009      C      _ArgList
00C      C      wrong flag\n
009      C      _ArgList
019      C      this is the right flag!\n
006      C      input
010      C      input the flag:
02B      C      ' is being used without being initializ
01C      C      Stack around the variable '

```

从最可疑的部分入手，并摠F5进行反编译。找到如下函数，此函数的功能大概率为将我们所输入的字符串与flag进行比较，我们沿这个思路继续查照

```

}  v3 = j_strlen(Str2);
}  if ( !strcmp(Str1, Str2, v3) )
L  sub_1400111D1("this is the right flag!\n");
2  else
3  sub_1400111D1("wrong flag\n");

4001C000      assume cs:_data
4001C000      ;org 14001C000h
4001C000 ; char Str2[]
4001C000 Str2      db '{hello_world}',0 ; DATA XREF: sub_1400111D1+0
4001C000      ; sub_1400118C0+67↑
4001C00E      align 10h
4001C010 ; uintptr_t _security_cookie
4001C010 ; security_cookie = 0000000000000000 ; DATA XREF: sub_1400111D1+0

```

我们发现变量Str2的内容为{hello_world}，也许这就是我们要找的flag？

不对，再等等。

```

7  }
8  for ( j = 0; ; ++j )
9  {
0  v8 = j;
1  v2 = j_strlen(Str2);
2  if ( v8 > v2 )
3  break;
4  if ( Str2[j] == 111 )
5  Str2[j] = 48;
6  }
7  sub_1400111D1("input the flag:");

```

程序在调用Str2进行比较之前进行了这一步操作，即为将字符串中的'o'换为'O'

所以我们最终得到的flag为flag{hell0_w0rld}