新人的reverse学习(6) I春秋 百度杯”CTF比赛 十一月场_CrackMe01

原创

fayinq 于 2021-04-26 21:46:12 发布 73 收藏

分类专栏： CTF Reverse 笔记

本文链接：https://blog.csdn.net/fayinq/article/details/116167225

CTF 同时被 3 个专栏收录

12 篇文章 0 订阅

订阅专栏

Reverse

7 篇文章 0 订阅

订阅专栏

笔记

7 篇文章 0 订阅

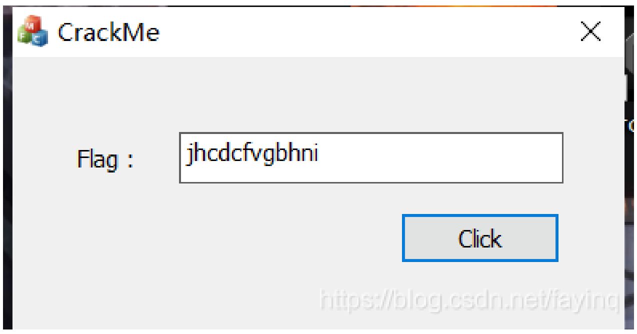
订阅专栏

题目链接

CrackMe01

1.用exeinfo查壳，无壳，32位

2.运行一下，发现是个窗口，而且输入值之后没有报错信息。



3.因为出现了窗口，所以说，我们要知道Windows中与窗口有关的一些函数。

MessageBox ()
功能是显示一个消息对话框，一般会传入4个参数，其功能分别为：

```
1.窗口句柄
2.消息框主体显示的字符串
3.消息框标题上的字符串
4.样式（就是下面有几个按钮，分别是啥东西）
```

一般为你输入一串字符串后用来提示你对不对。~~但是这个题没有，我就是想把它记下来~~

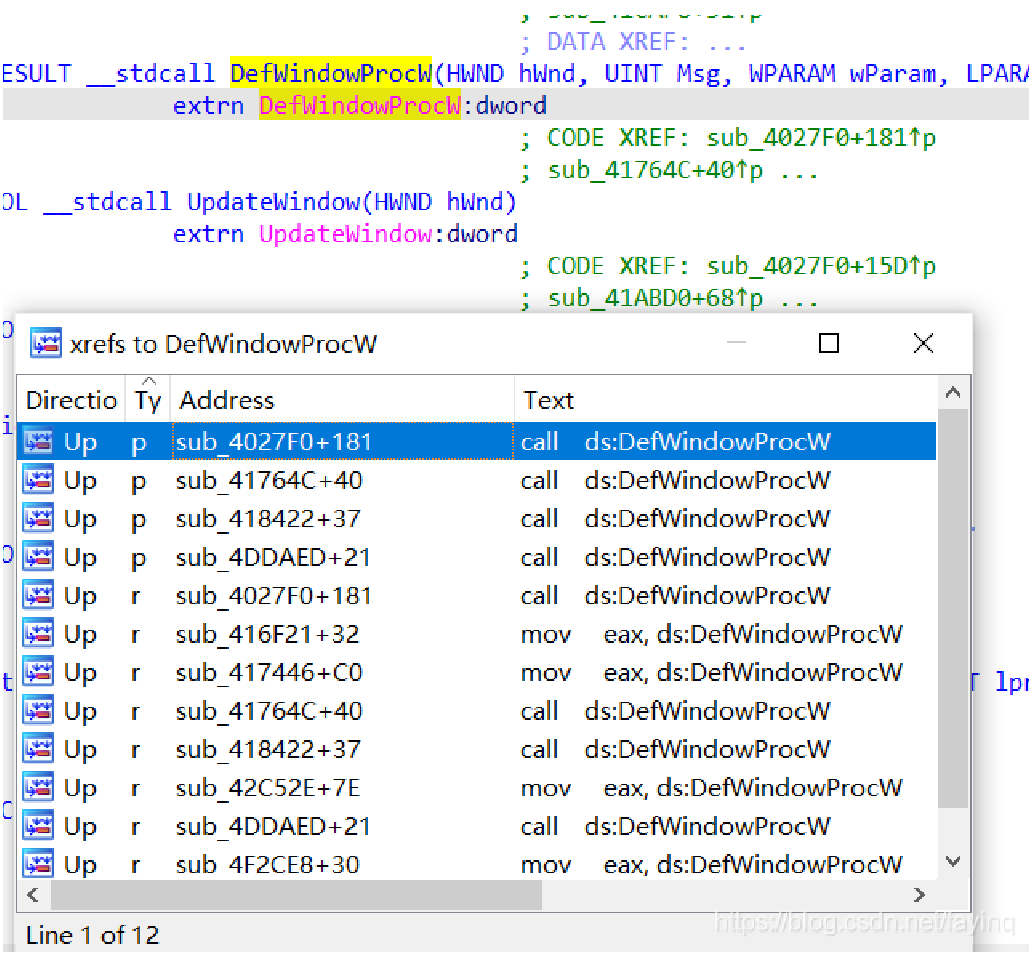
DefWindowProc（HWND hWnd，UINT Msg，WPARAM wParam，LPARAM lParam）
DefWindowProc这个函数是默认的窗口处理函数
一般有啥窗口（未进行特殊定义的窗口都是他）可以去找找这个函数。参数作用：

```
1.hWnd：指向接收消息的窗口过程的句柄。
2.Msg：指定消息类型
3.wParam：指定其余的、消息特定的信息。该参数的内容与Msg参数值有关
4.IParam：指定其余的、消息特定的信息。该参数的内容与Msg参数值有关
```

ShowWindow(HWND hWnd，int nCmdShow)

```
1.hwnd：没什么用
2.ncmdshow：窗口的形式
```

4。 在IDA的import窗口中查找DefWindowProc函数，并且寻找其交叉引用的位置。



5 进入第一个函数看看，然后好像看到了一些看起来比较重要的操作

```
  int v14; // eax
  struct tagRECT v15; // [esp+18h] [ebp-68h] BYREF
  struct tagRECT Rect; // [esp+28h] [ebp-58h] BYREF
  struct tagPAINTSTRUCT Paint; // [esp+38h] [ebp-48h] BYREF

  v4 = Msg;
  v5 = lParam;
  v6 = 0;
  v7 = (void *)wParam;
  if ( Msg > 0xF )
  {
    if ( Msg != 4097 )
      return DefWindowProcW((HWND)a1, v4, (WPARAM)v7, v5);
    v10 = 0;
    do
    {
      v11 = *(_WORD *)(wParam + 2 * v10++);
      v6 += v11;
    }
    while ( v10 <= lParam );
    for ( i = 0; i < 22; ++i )
      chText[i] ^= v6;
    GetWindowRect((HWND)a1, &v15);
    v13 = (v15.left - v15.right + GetSystemMetrics(16)) / 2;
    v14 = GetSystemMetrics(17);
    SetWindowPos((HWND)a1, HWND_MESSAGE|0x2, v13, (v15.top - v15.bottom + v14
    SetWindowPos((HWND)a1, (HWND)0xFFFFFFFE, 0, 0, 0, 0, 3u);
    if ( (v6 & 0xF00) == 1024 && (v6 & 0xF0) == 0xB0 && (v6 & 6) == 6 )
    {
      ShowWindow((HWND)a1, 5);
```



```
for ( i = 0; i < 22; ++i )
  chText[i] ^= v6;
```

```
if ( (v6 & 0xF00) == 1024 && (v6 & 0xF0) == 0xB0 && (v6 & 6) == 6 )
{
  ShowWindow((HWND)a1, 5);
  UpdateWindow((HWND)a1);
}
```

我们能在此看到有之前提到的一个重要函数，就是showwindow（推测一下，应该是会出现一个类似与正确的提示），所以说我们可以推测出v6的值。而后就可以推测出chetext的值。

6 查看chText的值

```
04F0 04DA 04D7 04D1 048C 04FF 04F5 04FE 04E3 04F8 04E7 04FF 04E3 04E9 04F0 04F3
0485 0480 0484 04F2 04F4 04F3 0000 0000 EE5C 0057 0000 0000 3F2E 5641 4E43 546F
```

接下来写脚本得出v6的值和chtext就行了

上代码

v6:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
 for(int i=1;i<=2000;i++)
  if((i & 0xF00) == 1024 && (i & 0xF0) == 0xB0 && (i & 6) == 6)
  {
   cout<<i;
   break;
  }
 return 0;
}
```

chText:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
 int a[30]={0x4F0, 0x4DA, 0x4D7, 0x4D1, 0x48C, 0x4FF, 0x4F5, 0x4FE, 0x4E3, 0x4F8, 0x4E7, 0x4FF, 0x4E3, 0x4E9, 0x4F0,0x4F3, 0x485, 0x480, 0x484, 0x4F2, 0x4F4,0x4F3};
 for(int i=0;i<22;i++)
  a[i]^=1206;
 for(int i=0;i<22;i++)
  cout<<char(a[i]);
 return 0;
}
```

flag{ICHUNQIU_FE362DBE}