XCTF-攻防世界CTF平台-Reverse逆向类——59、mfc逆向-200（MFC编程逆向）

原创

大灬白 于 2021-09-08 14:32:34 发布 180 收藏 5

分类专栏： #VC++逆向 XCTF-攻防世界-Reverse逆向类题目 文章标签： mfc c++ c语言 VMProtect DES加密算法

本文链接：https://blog.csdn.net/Onlyone_1314/article/details/120179345

VC++逆向 同时被 2 个专栏收录

2 篇文章 0 订阅

订阅专栏

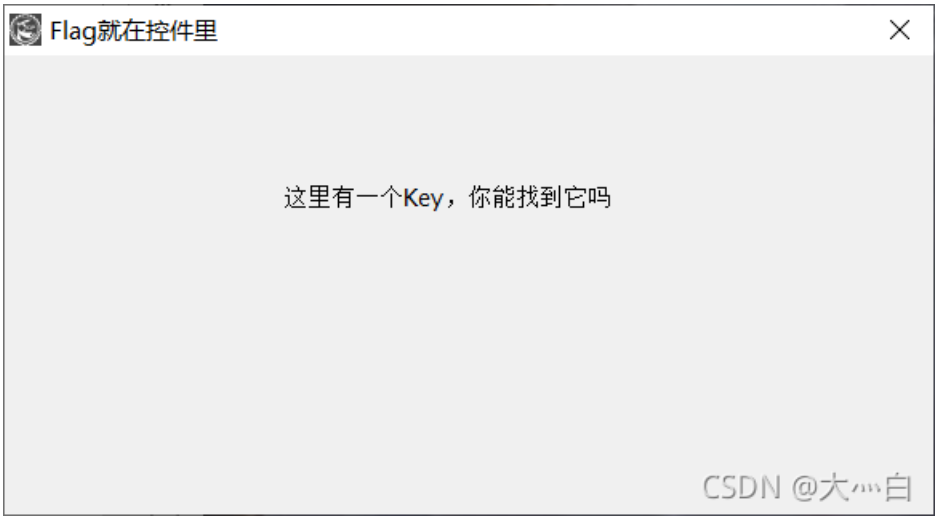
XCTF-攻防世界-Reverse逆向类题目

11 篇文章 1 订阅

订阅专栏

先查看程序信息：

Windows 32 位GUI程序，符合题目中MFC编程的程序，被加了VMProtect。
运行程序：



用xspy查看程序控件信息：

获得程序窗口句柄：00170E52
类名：944c8d100f82f0c18b682f63e4dbaa207a2f1e72581c2f1b
再观察程序的消息处理函数：

接受到以下消息：
WM_SYSCOMMAND（0x112，274）：当用户选择窗口菜单的一条命令或当用户选择最大化或最小化时那个窗口会收到此消息；
WM_PAINT（0xf，15）： 要求一个窗口重画自己；
0x0464：用户自定义的消息；
WM_QUERYDRAGICON（0x37，55）：此消息发送给最小化窗口，当此窗口将要被拖放而它的类中没有定义图标，应用程序能返回一个图标或光标的句柄，当用户拖放图标时系统显示这个图标或光标；
之后调用相应的函数。
我们看到程序自定义了一个消息0x0464（1124），所以我们用代码给程序发送一个该消息，观察它接受到这个消息之后会调用函数做什么处理：
SendMessage.cpp

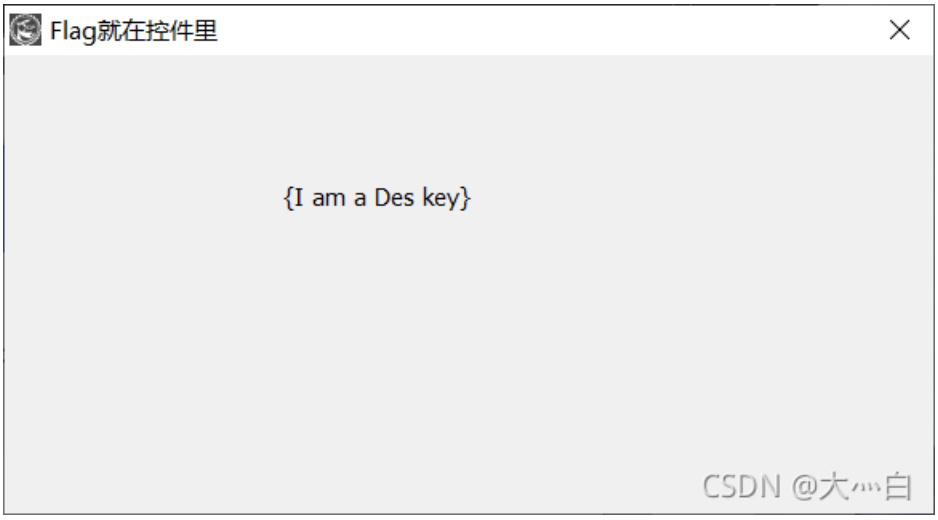
```
#include <stdio.h>
#include <stdlib.h>
#include <Windows.h>

int main()
{
    //获取窗口句柄
    HWND hwnd = ::FindWindowA(NULL, "Flag就在控件里");
    if (hwnd)
    {
        //向指定句柄程序发送一条消息0x464
        SendMessage(hwnd, 0x464, NULL, NULL);
    }else{
        printf("没有找到接收窗口");
    }
    system("pause");
    return 0;
}
```

其中

> LRESULT SendMessage（HWND hWnd，UINT wMsg，WPARAM wParam，LPARAM lParam);
> 参数：
> hWnd：其窗口程序将接收消息的窗口的句柄。如果此参数为HWND_BROADCAST，则消息将被发送到系统中所有顶层窗口，包括无效或不可见的非自身拥有的窗口、被覆盖的窗口和弹出式窗口，但消息不被发送到子窗口；
> wMsg用于区别其他消息的常量值，这些常量可以是Windows单元中预定义的常量，也可以是自定义的常量；
> wParam通常是一个与消息有关的常量值，也可能是窗口或控件的句柄；
> lParam通常是一个指向内存中数据的指针。由于WParm、lParam和Pointer都是32位的，因此，它们之间可以相互转换 返回值：返回值指定消息处理的结果，依赖于所发送的消息。

也可以直接用软件给程序发送消息：

之后我们的程序就调用函数显示了{I am a Des key}：



说明它是一个DES算法的密钥，我们开始看到类名：944c8d100f82f0c18b682f63e4dbaa207a2f1e72581c2f1b，就是一个64位密文与DES算法输出的密文位数相等，所以我们利用密钥将密文解密：

得到明文：`thIs_Is_real_kEy_hahaaa`，就是flag。