

# Reverse step1 writeup

原创

zh\_explorer 于 2015-04-24 21:54:41 发布 784 收藏

分类专栏: [hduisa内部平台writeup](#) 文章标签: [ctf reverse writeup](#)

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/zh\\_explorer/article/details/45251995](https://blog.csdn.net/zh_explorer/article/details/45251995)

版权



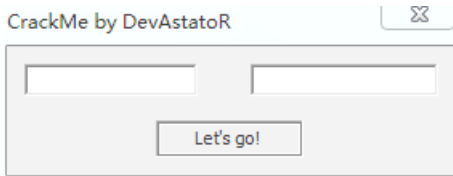
[hduisa内部平台writeup](#) 专栏收录该内容

6 篇文章 0 订阅

订阅专栏

不要吐槽题目的名字, 发题目的就是这么取的=,=

某天半夜出现在协会平台里面的Reverse简单题。这几天很忙, 就随便随便写写这题的writeup。



界面很简单, 两个edit框和一个按钮。直接丢进od。看这紧凑的代码估计是直接写汇编的。程序很简单, 不像高级语言编译的程序, 没有一点多余的代码。结合hint: Do you know SendMessage ?很快就可以发现异常的地方。

```
15 1020400 | call dword ptr ds:[<&SHLWAPI.StrToIntA>] | StrToIntA
4D E8      | lea ecx,[local.6]
          | push ecx
0C        | push 0xC
E4000200  | sub eax,0x200E4
          | push eax
          | push esi
07        | call edi
78204000  | mov esi,MyCrackI.00402070
```

```
1Param
wParam = C

Message
hWnd
SendMessageA
```

把第一个框的输入减去0x200E4作为SendMessage的消息。脑补0xD的WM\_GETTEXT消息。算出要输入的参数是"131313"好吧, 这随意的参数很有某人的风格=,=。

继续向下, SendMessage消息得到第二个edit框的字符, 然后加密。

```
> 33C9      | xor ecx,ecx
> 8BC1      | mov eax,ecx
- 99        | cdq
- F7FE      | idiv esi
- 8A4415 D8 | mov al,byte ptr ss:[ebp+edx-0x28]
- 30440D E8 | xor byte ptr ss:[ebp+ecx-0x18],al
- 41        | inc ecx
- 83F9 0B   | cmp ecx,0xB
^ 7C ED     | | XMyCrackI.004010AA
```

把输入前11个字符和0018FA24 24 0B 15 1C 13 0A 04 36 06 17 2F亦或。然后作为参数调用GetProcAddress。找找返回的地址在哪使用。

0040115F	- 53	push ebx	
00401160	- 68 40204000	push MyCrackI.00402040	ASCII "Well done."
00401165	- 8D45 B4	lea eax,[local.19]	
00401168	- 50	push eax	
00401169	- 53	push ebx	
0040116A	- FF55 10	call [arg.3]	

这里，看这4个参数再次脑补MessageBoxA。反向亦或得到输入参数information输入，自动弹出flag

程序会根据两个输入框的输入动态计算出flag，所以强行用od更改流程会弹出错误的flag，只有正确的输入才能得到正确的flag。