随风而逝的白色相簿

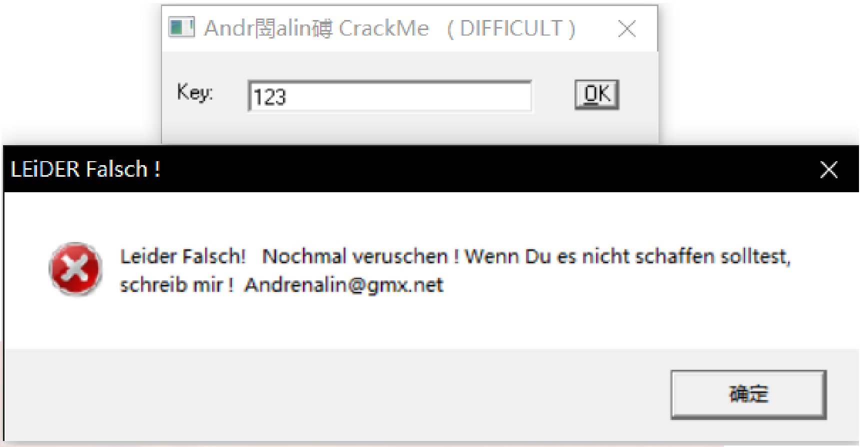
博客园 首页 新随笔 联系 订阅 管理

随笔 - 4 文章 - 0 评论 - 0

Crackme010

Crackme010 的逆向分析

1.程序观察



Andr閭alin磚 CrackMe （DIFFICULT）

Key: 123

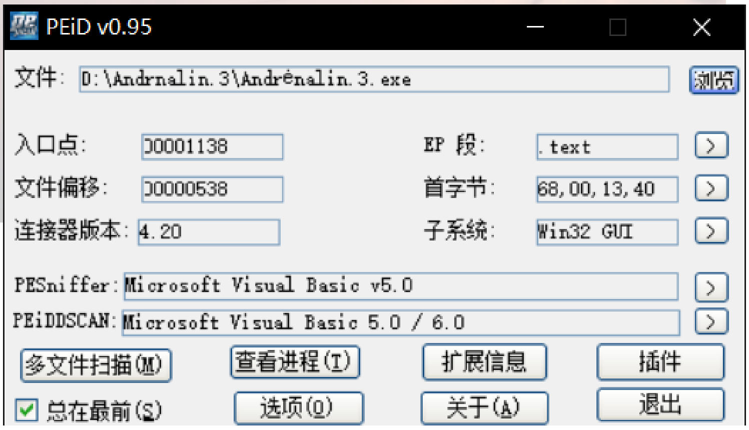
LEiDER Falsch !

Leider Falsch! Nochmal veruschen ! Wenn Du es nicht schaffen solltest, schreib mir ! Andrenalin@gmx.net

程序只有一个输入 Key 值的地方，尝试输入 "123"，程序和 008、009 一样出现的还是德语错误提示，因为这都是一个人制作的小程序。

虽然这个程序标注的是 3星难度，但是我感觉这个程序和 009 一样，只是稍微有一点不同，难度应该也是 1星才对。

2.简单查壳



PEiD v0.95

文件: D:\Andrnalin.3\Andrénalin.3.exe

入口点: 00001138 EP 段: .text

文件偏移: 00000538 首字节: 68,00,13,40

连接器版本: 4.20 子系统: Win32 GUI

PESniffer: Microsoft Visual Basic v5.0

PEiDDSCAN: Microsoft Visual Basic 5.0 / 6.0

公告

昵称：随风而逝的白色相簿
园龄：1年8个月
粉丝：1
关注：4

<			2019年9月			>
日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

随笔分类

160 Crackme(3)

随笔档案

2019年9月(3)
2018年10月(1)

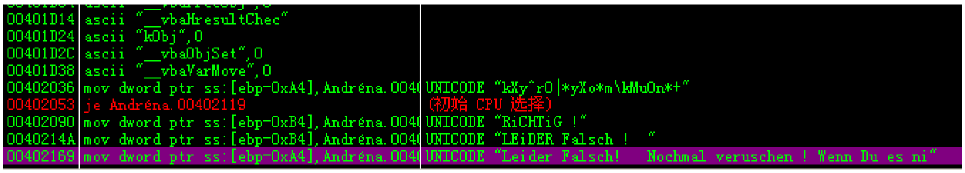
阅读排行榜

1. PHP一句话木马(6423)

无壳。

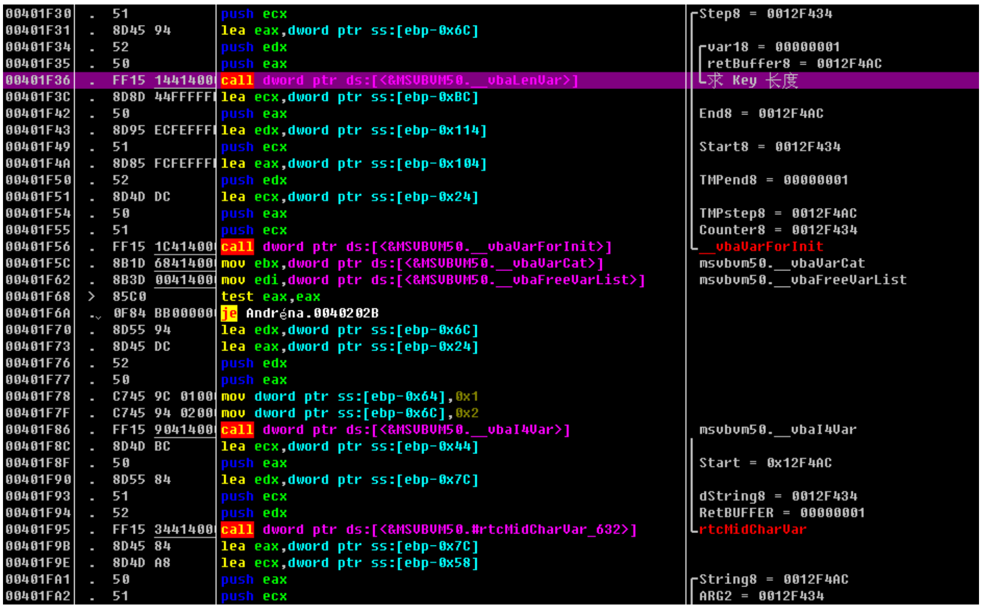
3.程序分析

使用 OD 载入程序，搜索字符串

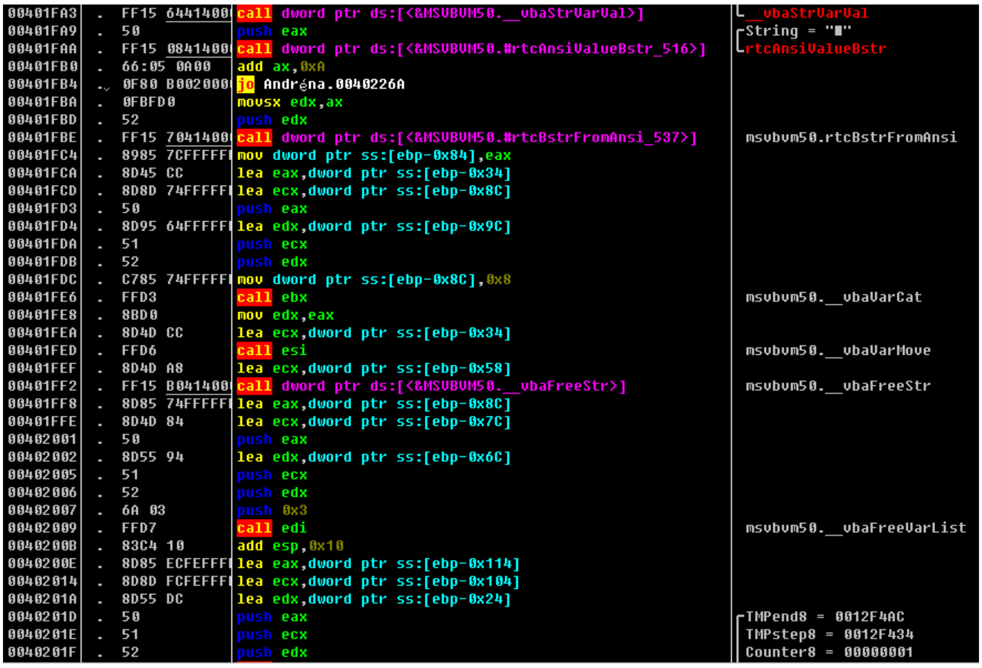


```
00401D14 ascii "__vbaHresultChec"
00401D24 ascii "kObj",0
00401D2C ascii "__vbaObjSet",0
00401D38 ascii "__vbaVarMove",0
00402036 mov dword ptr ss:[ebp-0xA4],Andréna.004 UNICODE "kXy^rO|*yXo*m\kMuOn*+"
00402053 je Andréna.00402119                     (初始 CPU 选择)
00402090 mov dword ptr ss:[ebp-0xB4],Andréna.004 UNICODE "RiCHTiG !"
0040214A mov dword ptr ss:[ebp-0xB4],Andréna.004 UNICODE "LEiDER Falsch !  "
00402169 mov dword ptr ss:[ebp-0xA4],Andréna.004 UNICODE "Leider Falsch!   Nochmal veruschen ! Wenn Du es ni"
```

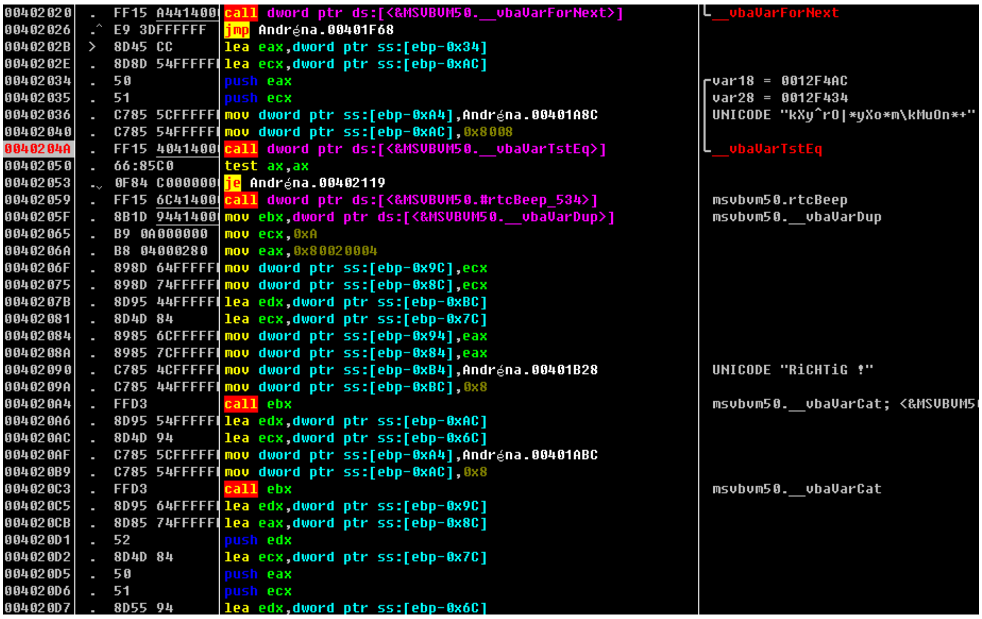
双击进入代码，可以看到，和 009 几乎是一模一样。



```
00401F30  .  51            push ecx                                           ┌Step8 = 0012F434
00401F31  .  8D45 94       lea eax,dword ptr ss:[ebp-0x6C]                    │
00401F34  .  52            push edx                                           │┌var18 = 00000001
00401F35  .  50            push eax                                           ││retBuffer8 = 0012F4AC
00401F36  .  FF15 1441400  call dword ptr ds:[<&MSVBVM50.__vbaLenVar>]        │└求 Key 长度
00401F3C  .  8D8D 44FFFFF  lea ecx,dword ptr ss:[ebp-0xBC]                    │
00401F42  .  50            push eax                                           │ End8 = 0012F4AC
00401F43  .  8D95 ECFEFFF  lea edx,dword ptr ss:[ebp-0x114]                   │
00401F49  .  51            push ecx                                           │ Start8 = 0012F434
00401F4A  .  8D85 FCFEFFF  lea eax,dword ptr ss:[ebp-0x104]                   │
00401F50  .  52            push edx                                           │ TMPend8 = 00000001
00401F51  .  8D4D DC       lea ecx,dword ptr ss:[ebp-0x24]                    │
00401F54  .  50            push eax                                           │ TMPstep8 = 0012F4AC
00401F55  .  51            push ecx                                           │ Counter8 = 0012F434
00401F56  .  FF15 1C41400  call dword ptr ds:[<&MSVBVM50.__vbaVarForInit>]    └__vbaVarForInit
00401F5C  .  8B1D 6841400  mov ebx,dword ptr ds:[<&MSVBVM50.__vbaVarCat>]      msvbvm50.__vbaVarCat
00401F62  .  8B3D 0041400  mov edi,dword ptr ds:[<&MSVBVM50.__vbaFreeVarList>] msvbvm50.__vbaFreeVarList
00401F68  >  85C0          test eax,eax
00401F6A  .v 0F84 BB00000  je Andréna.0040202B
00401F70  .  8D55 94       lea edx,dword ptr ss:[ebp-0x6C]
00401F73  .  8D45 DC       lea eax,dword ptr ss:[ebp-0x24]
00401F76  .  52            push edx
00401F77  .  50            push eax
00401F78  .  C745 9C 0100  mov dword ptr ss:[ebp-0x64],0x1
00401F7F  .  C745 94 0200  mov dword ptr ss:[ebp-0x6C],0x2
00401F86  .  FF15 9041400  call dword ptr ds:[<&MSVBVM50.__vbaI4Var>]          msvbvm50.__vbaI4Var
00401F8C  .  8D4D BC       lea ecx,dword ptr ss:[ebp-0x44]                    │
00401F8F  .  50            push eax                                           │ Start = 0x12F4AC
00401F90  .  8D55 84       lea edx,dword ptr ss:[ebp-0x7C]                    │
00401F93  .  51            push ecx                                           │ dString8 = 0012F434
00401F94  .  52            push edx                                           │ RetBUFFER = 00000001
00401F95  .  FF15 3441400  call dword ptr ds:[<&MSVBVM50.#rtcMidCharVar_632>] └rtcMidCharVar
00401F9B  .  8D45 84       lea eax,dword ptr ss:[ebp-0x7C]
00401F9E  .  8D4D A8       lea ecx,dword ptr ss:[ebp-0x58]
00401FA1  .  50            push eax                                           ┌String8 = 0012F4AC
00401FA2  .  51            push ecx                                           │ARG2 = 0012F434
```



```
00401FA3  .  FF15 6441400  call dword ptr ds:[<&MSVBVM50.__vbaStrVarVal>]           └__vbaStrVarVal
00401FA9  .  50            push eax                                                 ┌String = "■"
00401FAA  .  FF15 0841400  call dword ptr ds:[<&MSVBVM50.#rtcAnsiValueBstr_516>]    └rtcAnsiValueBstr
00401FB0  .  66:05 0A00    add ax,0xA
00401FB4  .v 0F80 B002000  jo Andréna.0040226A
00401FBA  .  0FBFD0        movsx edx,ax
00401FBD  .  52            push edx
00401FBE  .  FF15 7041400  call dword ptr ds:[<&MSVBVM50.#rtcBstrFromAnsi_537>]     msvbvm50.rtcBstrFromAnsi
00401FC4  .  8985 7CFFFFF  mov dword ptr ss:[ebp-0x84],eax
00401FCA  .  8D45 CC       lea eax,dword ptr ss:[ebp-0x34]
00401FCD  .  8D8D 74FFFFF  lea ecx,dword ptr ss:[ebp-0x8C]
00401FD3  .  50            push eax
00401FD4  .  8D95 64FFFFF  lea edx,dword ptr ss:[ebp-0x9C]
00401FDA  .  51            push ecx
00401FDB  .  52            push edx
00401FDC  .  C785 74FFFFF  mov dword ptr ss:[ebp-0x8C],0x8
00401FE6  .  FFD3          call ebx                                                 msvbvm50.__vbaVarCat
00401FE8  .  8BD0          mov edx,eax
00401FEA  .  8D4D CC       lea ecx,dword ptr ss:[ebp-0x34]
00401FED  .  FFD6          call esi                                                 msvbvm50.__vbaVarMove
00401FEF  .  8D4D A8       lea ecx,dword ptr ss:[ebp-0x58]
00401FF2  .  FF15 B041400  call dword ptr ds:[<&MSVBVM50.__vbaFreeStr>]             msvbvm50.__vbaFreeStr
00401FF8  .  8D85 74FFFFF  lea eax,dword ptr ss:[ebp-0x8C]
00401FFE  .  8D4D 84       lea ecx,dword ptr ss:[ebp-0x7C]
00402001  .  50            push eax
00402002  .  8D55 94       lea edx,dword ptr ss:[ebp-0x6C]
00402005  .  51            push ecx
00402006  .  52            push edx
00402007  .  6A 03         push 0x3
00402009  .  FFD7          call edi                                                 msvbvm50.__vbaFreeVarList
0040200B  .  83C4 10       add esp,0x10
0040200E  .  8D85 ECFEFFF  lea eax,dword ptr ss:[ebp-0x114]
00402014  .  8D8D FCFEFFF  lea ecx,dword ptr ss:[ebp-0x104]
0040201A  .  8D55 DC       lea edx,dword ptr ss:[ebp-0x24]
0040201D  .  50            push eax                                                 ┌TMPend8 = 0012F4AC
0040201E  .  51            push ecx                                                 │TMPstep8 = 0012F434
0040201F  .  52            push edx                                                 │Counter8 = 00000001
```



```
00402020  .  FF15 A4414000 call dword ptr ds:[<&MSVBVM50.__vbaVarForNext>]   └__vbaVarForNext
00402026  .^ E9 3DFFFFFF   jmp Andrḗna.00401F68
0040202B  >  8D45 CC       lea eax,dword ptr ss:[ebp-0x34]
0040202E  .  8D8D 54FFFFFF lea ecx,dword ptr ss:[ebp-0xAC]
00402034  .  50            push eax                                          ┌var18 = 0012F4AC
00402035  .  51            push ecx                                          │var28 = 0012F434
00402036  .  C785 5CFFFFFF mov dword ptr ss:[ebp-0xA4],Andrḗna.00401A8C     │UNICODE "kXy^rO|*yXo*m\kMuOn*+"
00402040  .  C785 54FFFFFF mov dword ptr ss:[ebp-0xAC],0x8008                │
0040204A  .  FF15 40414000 call dword ptr ds:[<&MSVBVM50.__vbaVarTstEq>]     └__vbaVarTstEq
00402050  .  66:85C0       test ax,ax
00402053  .v 0F84 C0000000 je Andrḗna.00402119
00402059  .  FF15 6C414000 call dword ptr ds:[<&MSVBVM50.#rtcBeep_534>]      msvbvm50.rtcBeep
0040205F  .  8B1D 94414000 mov ebx,dword ptr ds:[<&MSVBVM50.__vbaVarDup>]    msvbvm50.__vbaVarDup
00402065  .  B9 0A000000   mov ecx,0xA
0040206A  .  B8 04000280   mov eax,0x80020004
0040206F  .  898D 64FFFFFF mov dword ptr ss:[ebp-0x9C],ecx
00402075  .  898D 74FFFFFF mov dword ptr ss:[ebp-0x8C],ecx
0040207B  .  8D95 44FFFFFF lea edx,dword ptr ss:[ebp-0xBC]
00402081  .  8D4D 84       lea ecx,dword ptr ss:[ebp-0x7C]
00402084  .  8985 6CFFFFFF mov dword ptr ss:[ebp-0x94],eax
0040208A  .  8985 7CFFFFFF mov dword ptr ss:[ebp-0x84],eax
00402090  .  C785 4CFFFFFF mov dword ptr ss:[ebp-0xB4],Andrḗna.00401B28      UNICODE "RiCHTiG !"
0040209A  .  C785 44FFFFFF mov dword ptr ss:[ebp-0xBC],0x8
004020A4  .  FFD3          call ebx                                           msvbvm50.__vbaVarCat; <&MSVBVM5
004020A6  .  8D95 54FFFFFF lea edx,dword ptr ss:[ebp-0xAC]
004020AC  .  8D4D 94       lea ecx,dword ptr ss:[ebp-0x6C]
004020AF  .  C785 5CFFFFFF mov dword ptr ss:[ebp-0xA4],Andrḗna.00401ABC
004020B9  .  C785 54FFFFFF mov dword ptr ss:[ebp-0xAC],0x8
004020C3  .  FFD3          call ebx                                           msvbvm50.__vbaVarCat
004020C5  .  8D95 64FFFFFF lea edx,dword ptr ss:[ebp-0x9C]
004020CB  .  8D85 74FFFFFF lea eax,dword ptr ss:[ebp-0x8C]
004020D1  .  52            push edx
004020D2  .  8D4D 84       lea ecx,dword ptr ss:[ebp-0x7C]
004020D5  .  50            push eax
004020D6  .  51            push ecx
004020D7  .  8D55 94       lea edx,dword ptr ss:[ebp-0x6C]
```

程序流程大概就是：

1. 程序首先求得输入 Key 的长度。

2. 建立循环，循环次数为 Key 的长度。

3. 每次取我们输入的 Key 的一个字符。

4. 将字符转换为 ASCII 再加上 0xA。

5. 再次转换为字符形式。

6. 将转换过的字符连接起来。

在最后程序将转换过的字符串和字符串 "kXy^rO|*yXo*m\kMuOn*+"进行比较，如果相同进提示正确，不相同就提示错误。

4.写出注册机

既然已经知道了转换的步骤，也有了正确的转换后的字符串，那么反推出正确的 Key 就是很简单的了。

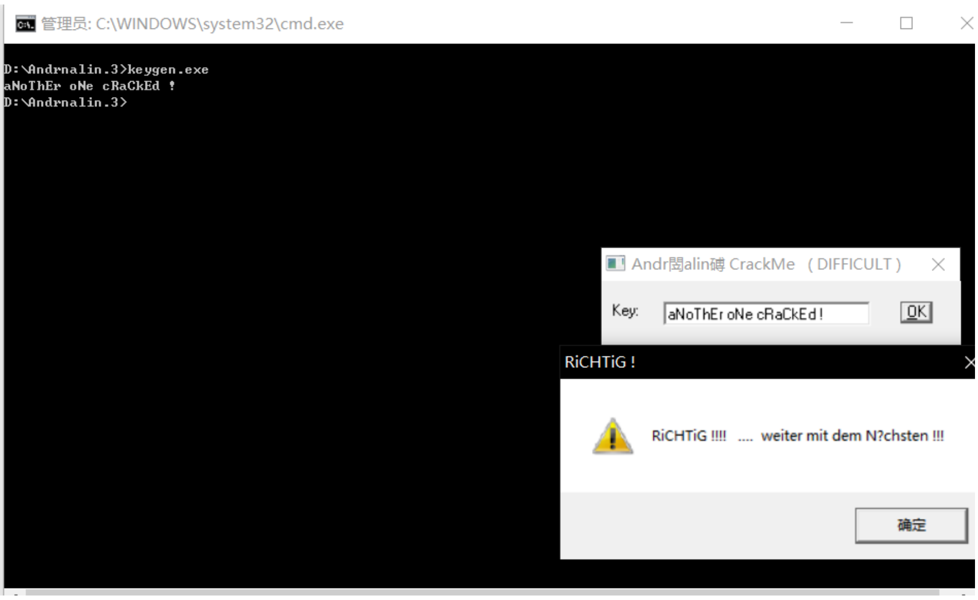
```
#include <stdio.h>
#include <string.h>


int Key()
{
    char szKey[30] = "kXy^rO|*yXo*m\\kMuOn*+";
    int NameLen = 0;

    NameLen = strlen(szKey);

    for (int i = 0; i < NameLen; i++)
    {
        szKey[i] -= 0xa;
    }
    printf("%s", szKey);
    return 0;
}

int main(int argc, char* argv[])
{
    Key();
    return 0;
}
```



管理员: C:\WINDOWS\system32\cmd.exe

```
D:\Andrnalin.3>keygen.exe
aNoThEr oNe cRaCkEd !
D:\Andrnalin.3>
```

Andr閻alin磚 CrackMe （DIFFICULT）

Key: aNoThEr oNe cRaCkEd !

OK

RiCHTiG !

RiCHTiG !!!! .... weiter mit dem N?chsten !!!

确定

相关文件在我的 Github：https://github.com/UnreachableLove/160-Crackme/tree/master/Crackme010

2019-09-10 19:15:56

分类: 160 Crackme

好文要顶 关注我 收藏该文

随风而逝的白色相簿
关注 - 4
粉丝 - 1

« 上一篇： Crackme009

posted @ 2019-09-10 19:16 随风而逝的白色相簿 阅读(1) 评论(0) 编辑 收藏

刷新评论 刷新页面 返回顶部

发表评论

昵称： 随风而逝的白色相簿

评论内容：

提交评论 退出

[Ctrl+Enter快捷键提交]

0
推荐

0
推荐