re学习笔记（40）i春秋2020 GYCTF-re-吃鸡神器

原创

Forgo7ten 于 2020-02-22 22:02:26 发布 568 收藏

分类专栏： ctf小白成长ing # reverse 文章标签： i春秋 CTF reverse qt5

ctf小白成长ing 同时被 2 个专栏收录

112 篇文章 6 订阅

订阅专栏

reverse

113 篇文章 6 订阅

订阅专栏

i春秋2020新春战疫赛-re-吃鸡神器

新手一枚，如有错误（不足）请指正，谢谢！！

个人博客：点击进入

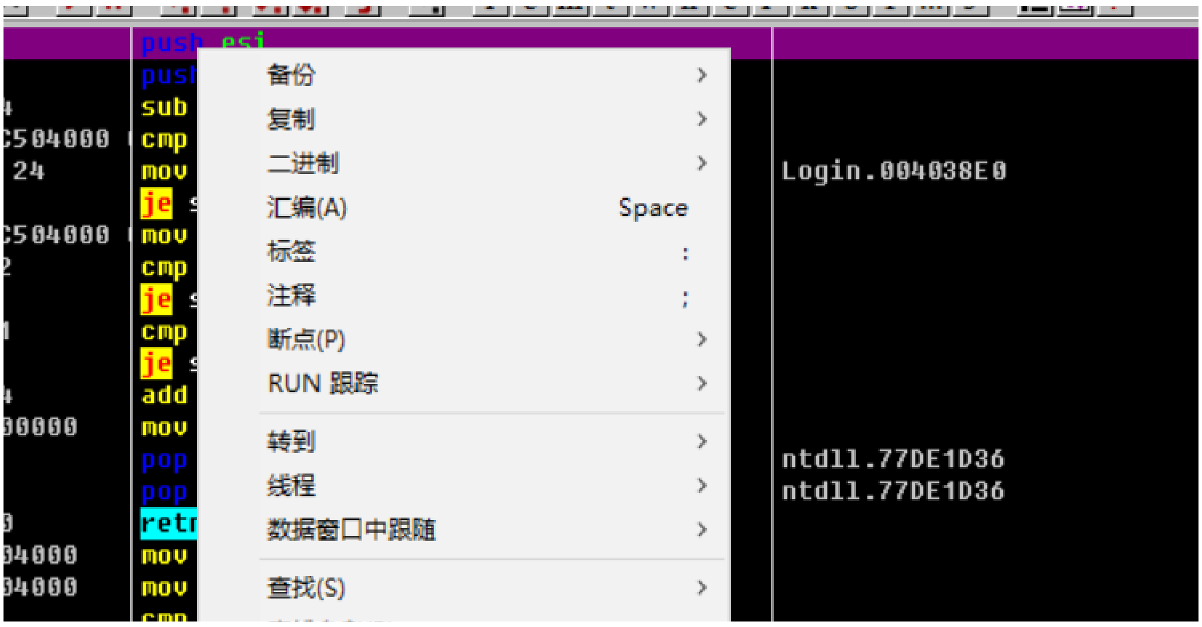
题目描述：

> 卢姥爷收到了朋友发来的"吃鸡神器"，但是朋友忘了告诉他登录账户和密码，□且卢姥爷也不好意思去问。所以 请为卢姥爷以 "lubenwei" 为□户名注册个账户。flag格式为 "flag{对应密码}"

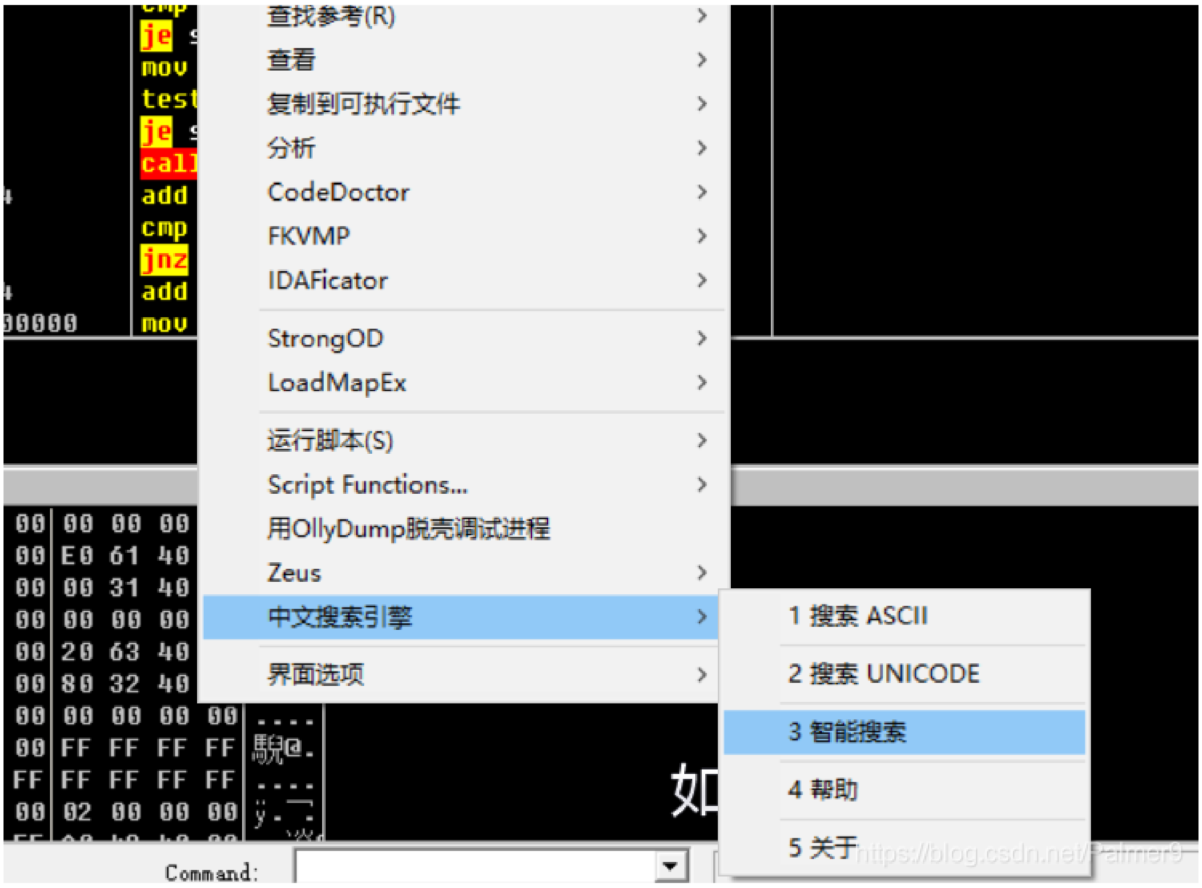
题目下载：

> 链接: https://pan.baidu.com/s/1gVE158CY6VmSV4qUwtXXqA 提取码: 2020

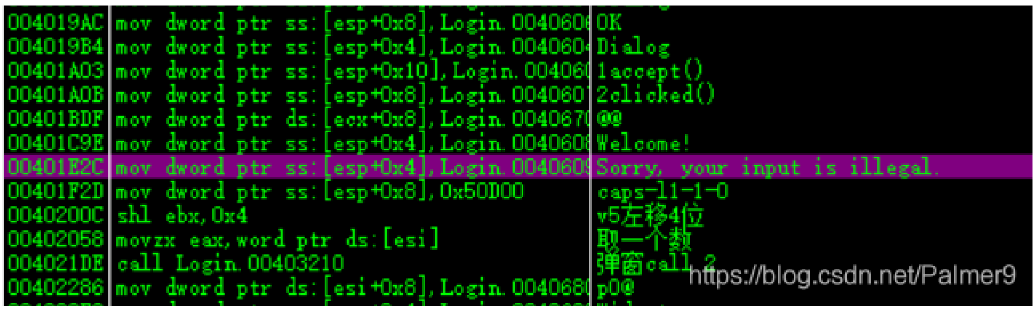
OD载入，字符串搜索



```
push esi
push
sub
cmp
mov      Login.004038E0
je
mov
cmp
je
cmp
je
add
mov
pop      ntdll.77DE1D36
pop      ntdll.77DE1D36
retn
mov
mov
```

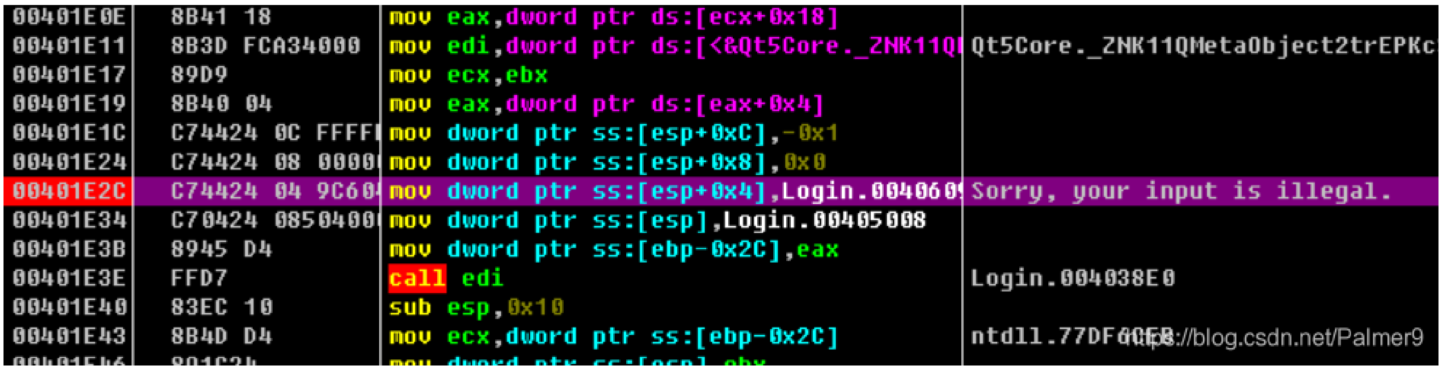


找到关键字符串，双击进入



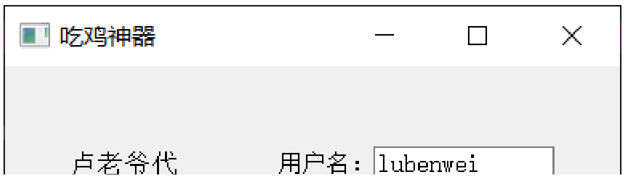
```
004019AC mov dword ptr ss:[esp+0x8],Login.004060 OK
004019B4 mov dword ptr ss:[esp+0x4],Login.004060 Dialog
00401A03 mov dword ptr ss:[esp+0x10],Login.004060 1accept()
00401A0B mov dword ptr ss:[esp+0x8],Login.004060 2clicked()
00401BDF mov dword ptr ds:[ecx+0x8],Login.004067 @@
00401C9E mov dword ptr ss:[esp+0x4],Login.004060 Welcome!
00401E2C mov dword ptr ss:[esp+0x4],Login.004060 Sorry, your input is illegal.
00401F2D mov dword ptr ss:[esp+0x8],0x50D00      caps-l1-1-0
0040200C shl ebx,0x4                             v5左移4位
00402058 movzx eax,word ptr ds:[esi]             取一个数
004021DE call Login.00403210                     弹窗call 2
00402286 mov dword ptr ds:[esi+0x8],Login.004068 p0@
```

F2下断点，运行程序，



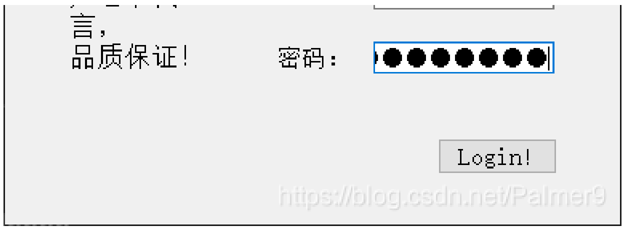
```
00401E0E   8B41 18         mov eax,dword ptr ds:[ecx+0x18]
00401E11   8B3D FCA34000   mov edi,dword ptr ds:[<&Qt5Core._ZNK11Q  Qt5Core._ZNK11QMetaObject2trEPKc
00401E17   89D9            mov ecx,ebx
00401E19   8B40 04         mov eax,dword ptr ds:[eax+0x4]
00401E1C   C74424 0C FFFF  mov dword ptr ss:[esp+0xC],-0x1
00401E24   C74424 08 0000  mov dword ptr ss:[esp+0x8],0x0
00401E2C   C74424 04 9C60  mov dword ptr ss:[esp+0x4],Login.004060  Sorry, your input is illegal.
00401E34   C70424 0850400  mov dword ptr ss:[esp],Login.00405008
00401E3B   8945 D4         mov dword ptr ss:[ebp-0x2C],eax
00401E3E   FFD7            call edi                                 Login.004038E0
00401E40   83EC 10         sub esp,0x10
00401E43   8B4D D4         mov ecx,dword ptr ss:[ebp-0x2C]          ntdll.77DF
```

输入用户名和假码，进行login

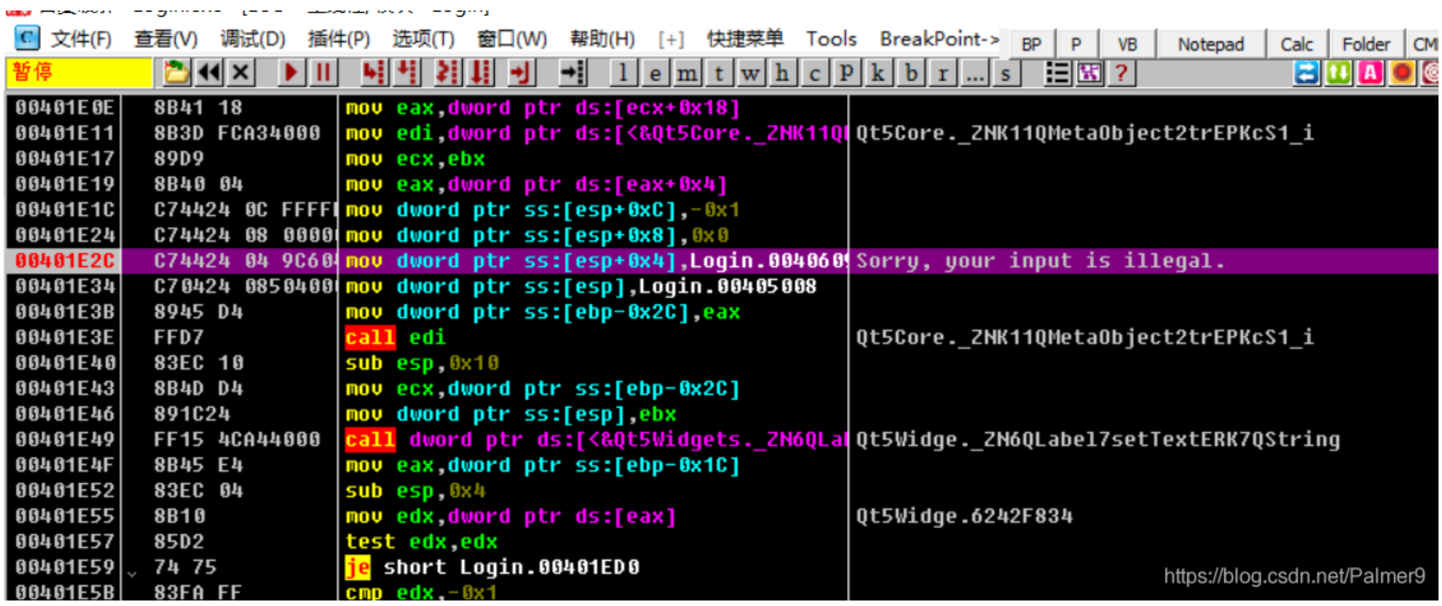


吃鸡神器

卢老爷代　　用户名: lubenwei

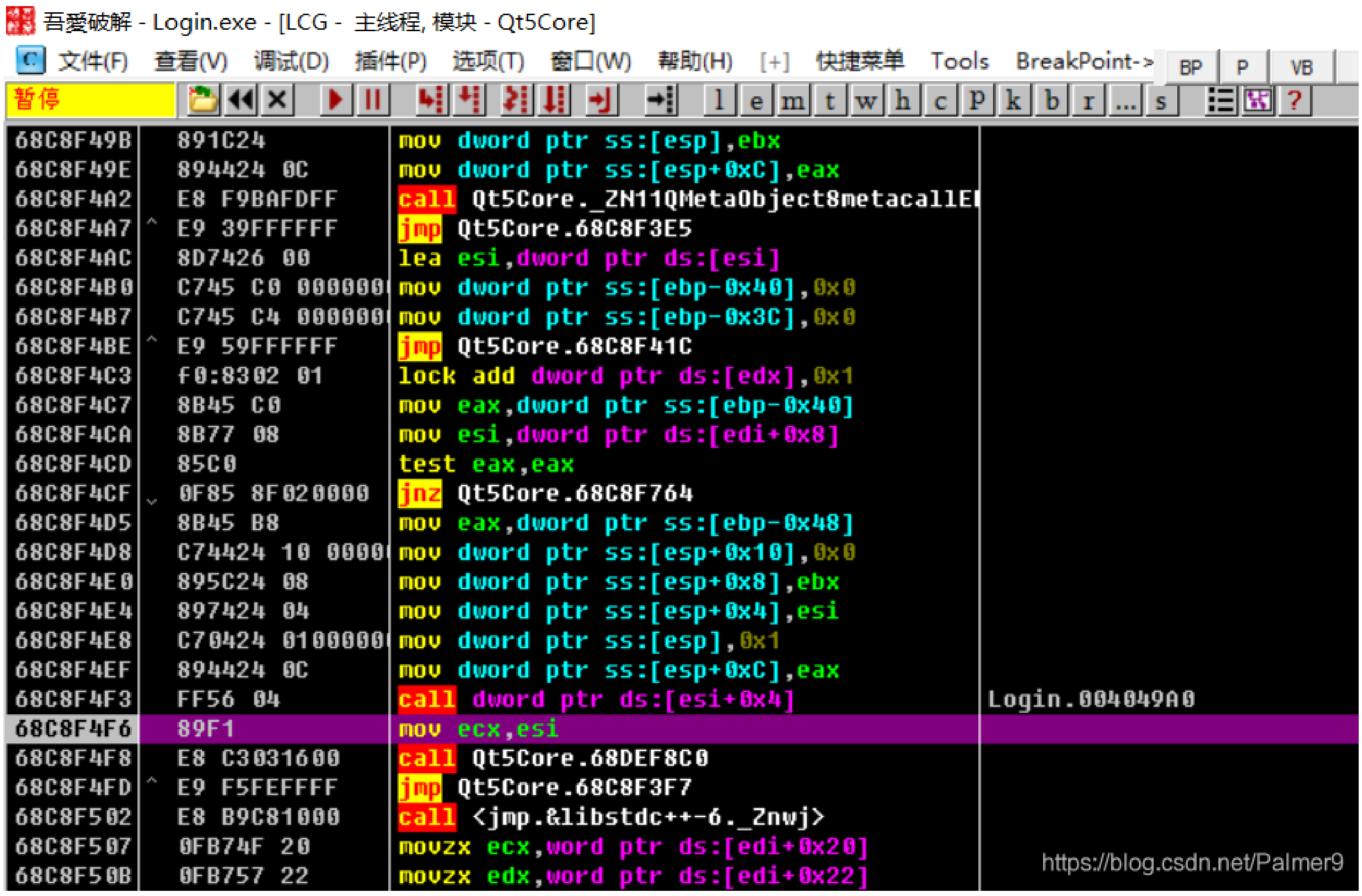


程序被断下来。


文件(F) 查看(V) 调试(D) 插件(P) 选项(T) 窗口(W) 帮助(H) [+] 快捷菜单 Tools BreakPoint-> BP P VB Notepad Calc Folder CM
暂停
l e m t w h c P k b r ... s

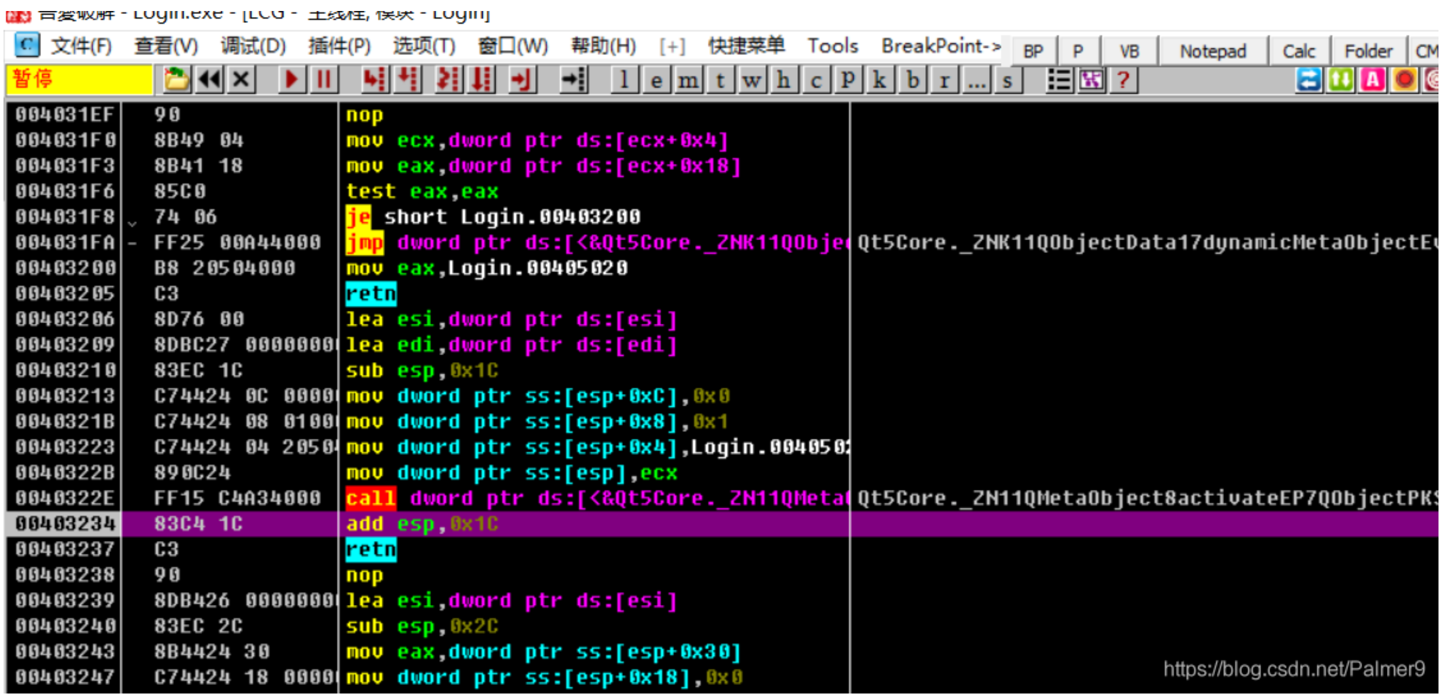
```
00401E0E   8B41 18          mov eax,dword ptr ds:[ecx+0x18]
00401E11   8B3D FCA34000    mov edi,dword ptr ds:[<&Qt5Core._ZNK11Q  Qt5Core._ZNK11QMetaObject2trEPKcS1_i
00401E17   89D9             mov ecx,ebx
00401E19   8B40 04          mov eax,dword ptr ds:[eax+0x4]
00401E1C   C74424 0C FFFF   mov dword ptr ss:[esp+0xC],-0x1
00401E24   C74424 08 0000   mov dword ptr ss:[esp+0x8],0x0
00401E2C   C74424 04 9C60   mov dword ptr ss:[esp+0x4],Login.004060  Sorry, your input is illegal.
00401E34   C70424 08504000  mov dword ptr ss:[esp],Login.00405008
00401E3B   8945 D4          mov dword ptr ss:[ebp-0x2C],eax
00401E3E   FFD7             call edi                                 Qt5Core._ZNK11QMetaObject2trEPKcS1_i
00401E40   83EC 10          sub esp,0x10
00401E43   8B4D D4          mov ecx,dword ptr ss:[ebp-0x2C]
00401E46   891C24           mov dword ptr ss:[esp],ebx
00401E49   FF15 4CA44000    call dword ptr ds:[<&Qt5Widgets._ZN6QLa  Qt5Widge._ZN6QLabel7setTextERK7QString
00401E4F   8B45 E4          mov eax,dword ptr ss:[ebp-0x1C]
00401E52   83EC 04          sub esp,0x4
00401E55   8B10             mov edx,dword ptr ds:[eax]               Qt5Widge.6242F834
00401E57   85D2             test edx,edx
00401E59 ˅ 74 75            je short Login.00401ED0
00401E5B   83FA FF          cmp edx,-0x1
```

ctrl+F9执行到返回，返回上层函数


吾爱破解 - Login.exe - [LCG - 主线程, 模块 - Qt5Core]
文件(F) 查看(V) 调试(D) 插件(P) 选项(T) 窗口(W) 帮助(H) [+] 快捷菜单 Tools BreakPoint-> BP P VB
暂停
l e m t w h c P k b r ... s

```
68C8F49B   891C24           mov dword ptr ss:[esp],ebx
68C8F49E   894424 0C        mov dword ptr ss:[esp+0xC],eax
68C8F4A2   E8 F9BAFDFF      call Qt5Core._ZN11QMetaObject8metacallE
68C8F4A7 ^ E9 39FFFFFF      jmp Qt5Core.68C8F3E5
68C8F4AC   8D7426 00        lea esi,dword ptr ds:[esi]
68C8F4B0   C745 C0 000000   mov dword ptr ss:[ebp-0x40],0x0
68C8F4B7   C745 C4 000000   mov dword ptr ss:[ebp-0x3C],0x0
68C8F4BE ^ E9 59FFFFFF      jmp Qt5Core.68C8F41C
68C8F4C3   F0:8302 01       lock add dword ptr ds:[edx],0x1
68C8F4C7   8B45 C0          mov eax,dword ptr ss:[ebp-0x40]
68C8F4CA   8B77 08          mov esi,dword ptr ds:[edi+0x8]
68C8F4CD   85C0             test eax,eax
68C8F4CF ˅ 0F85 8F020000    jnz Qt5Core.68C8F764
68C8F4D5   8B45 B8          mov eax,dword ptr ss:[ebp-0x48]
68C8F4D8   C74424 10 0000   mov dword ptr ss:[esp+0x10],0x0
68C8F4E0   895C24 08        mov dword ptr ss:[esp+0x8],ebx
68C8F4E4   897424 04        mov dword ptr ss:[esp+0x4],esi
68C8F4E8   C70424 0100000   mov dword ptr ss:[esp],0x1
68C8F4EF   894424 0C        mov dword ptr ss:[esp+0xC],eax
68C8F4F3   FF56 04          call dword ptr ds:[esi+0x4]              Login.004049A0
68C8F4F6   89F1             mov ecx,esi
68C8F4F8   E8 C3031600      call Qt5Core.68DEF8C0
68C8F4FD ^ E9 F5FEFFFF      jmp Qt5Core.68C8F3F7
68C8F502   E8 B9C81000      call <jmp.&libstdc++-6._Znwj>
68C8F507   0FB74F 20        movzx ecx,word ptr ds:[edi+0x20]
68C8F50B   0FB757 22        movzx edx,word ptr ds:[edi+0x22]
```

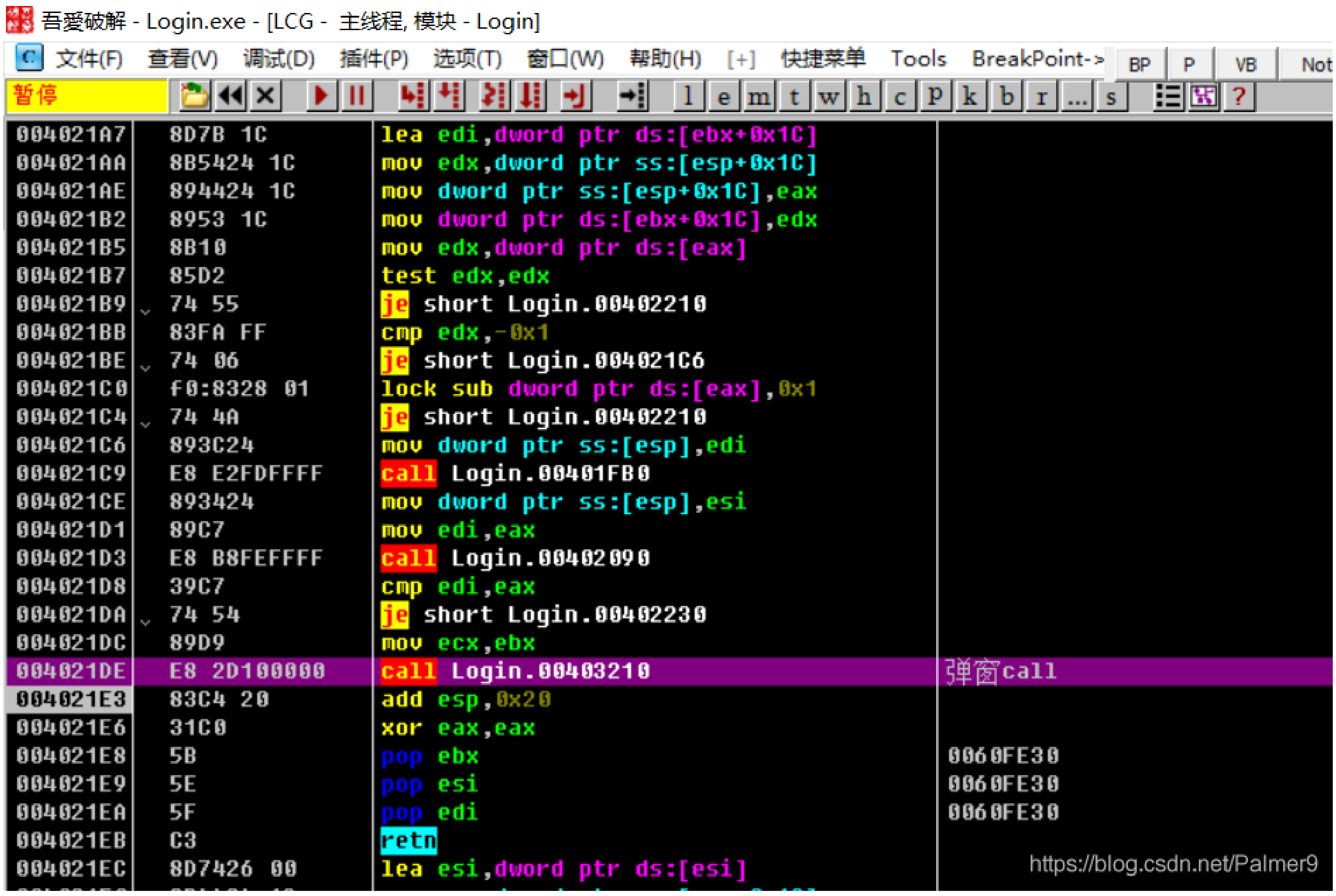
发现所在模块变成了一个dll，继续ctrl+F9执行到返回。发现来到了Login主模块



```
004031EF    90                nop
004031F0    8B49 04           mov ecx,dword ptr ds:[ecx+0x4]
004031F3    8B41 18           mov eax,dword ptr ds:[ecx+0x18]
004031F6    85C0              test eax,eax
004031F8  ⌄ 74 06             je short Login.00403200
004031FA  - FF25 00A44000     jmp dword ptr ds:[<&Qt5Core._ZNK11QObje  Qt5Core._ZNK11QObjectData17dynamicMetaObjectE
00403200    B8 20504000       mov eax,Login.00405020
00403205    C3                retn
00403206    8D76 00           lea esi,dword ptr ds:[esi]
00403209    8DBC27 0000000    lea edi,dword ptr ds:[edi]
00403210    83EC 1C           sub esp,0x1C
00403213    C74424 0C 0000    mov dword ptr ss:[esp+0xC],0x0
0040321B    C74424 08 0100    mov dword ptr ss:[esp+0x8],0x1
00403223    C74424 04 2050    mov dword ptr ss:[esp+0x4],Login.004050
0040322B    890C24            mov dword ptr ss:[esp],ecx
0040322E    FF15 C4A34000     call dword ptr ds:[<&Qt5Core._ZN11QMeta  Qt5Core._ZN11QMetaObject8activateEP7QObjectPK
00403234    83C4 1C           add esp,0x1C
00403237    C3                retn
00403238    90                nop
00403239    8DB426 0000000    lea esi,dword ptr ds:[esi]
00403240    83EC 2C           sub esp,0x2C
00403243    8B4424 30         mov eax,dword ptr ss:[esp+0x30]
00403247    C74424 18 0000    mov dword ptr ss:[esp+0x18],0x0
```

吾爱破解 - Login.exe - [LCG - 主线程, 模块 - Login]
文件(F) 查看(V) 调试(D) 插件(P) 选项(T) 窗口(W) 帮助(H) [+] 快捷菜单 Tools BreakPoint-> BP P VB Notepad Calc Folder CM
暂停
https://blog.csdn.net/Palmer9

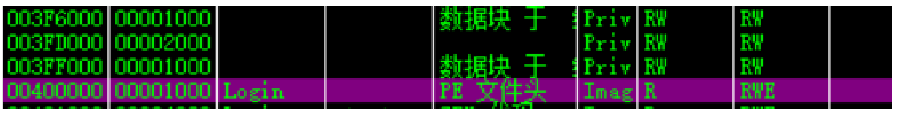
发现没有跳过去的跳转啥的。retn返回上层call
来到这里



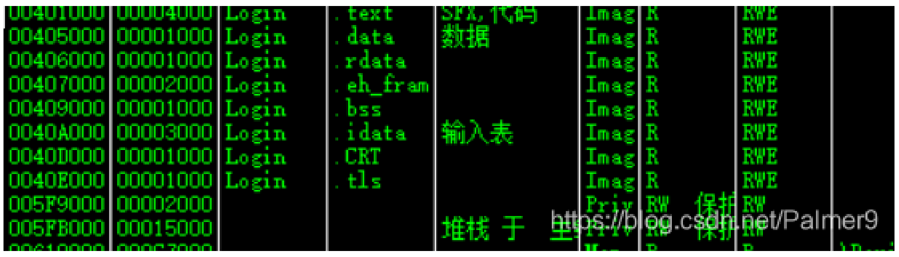
```
004021A7    8D7B 1C           lea edi,dword ptr ds:[ebx+0x1C]
004021AA    8B5424 1C         mov edx,dword ptr ss:[esp+0x1C]
004021AE    894424 1C         mov dword ptr ss:[esp+0x1C],eax
004021B2    8953 1C           mov dword ptr ds:[ebx+0x1C],edx
004021B5    8B10              mov edx,dword ptr ds:[eax]
004021B7    85D2              test edx,edx
004021B9  ⌄ 74 55             je short Login.00402210
004021BB    83FA FF           cmp edx,-0x1
004021BE  ⌄ 74 06             je short Login.004021C6
004021C0    F0:8328 01        lock sub dword ptr ds:[eax],0x1
004021C4  ⌄ 74 4A             je short Login.00402210
004021C6    893C24            mov dword ptr ss:[esp],edi
004021C9    E8 E2FDFFFF       call Login.00401FB0
004021CE    893424            mov dword ptr ss:[esp],esi
004021D1    89C7              mov edi,eax
004021D3    E8 B8FEFFFF       call Login.00402090
004021D8    39C7              cmp edi,eax
004021DA  ⌄ 74 54             je short Login.00402230
004021DC    89D9              mov ecx,ebx
004021DE    E8 2D100000       call Login.00403210         弹窗call
004021E3    83C4 20           add esp,0x20
004021E6    31C0              xor eax,eax
004021E8    5B                pop ebx                     0060FE30
004021E9    5E                pop esi                     0060FE30
004021EA    5F                pop edi                     0060FE30
004021EB    C3                retn
004021EC    8D7426 00         lea esi,dword ptr ds:[esi]
```

吾爱破解 - Login.exe - [LCG - 主线程, 模块 - Login]
文件(F) 查看(V) 调试(D) 插件(P) 选项(T) 窗口(W) 帮助(H) [+] 快捷菜单 Tools BreakPoint-> BP P VB Not
暂停
https://blog.csdn.net/Palmer9

这个程序IDA的基址和OD的基址一样，都是0x400000，用IDA配合查看

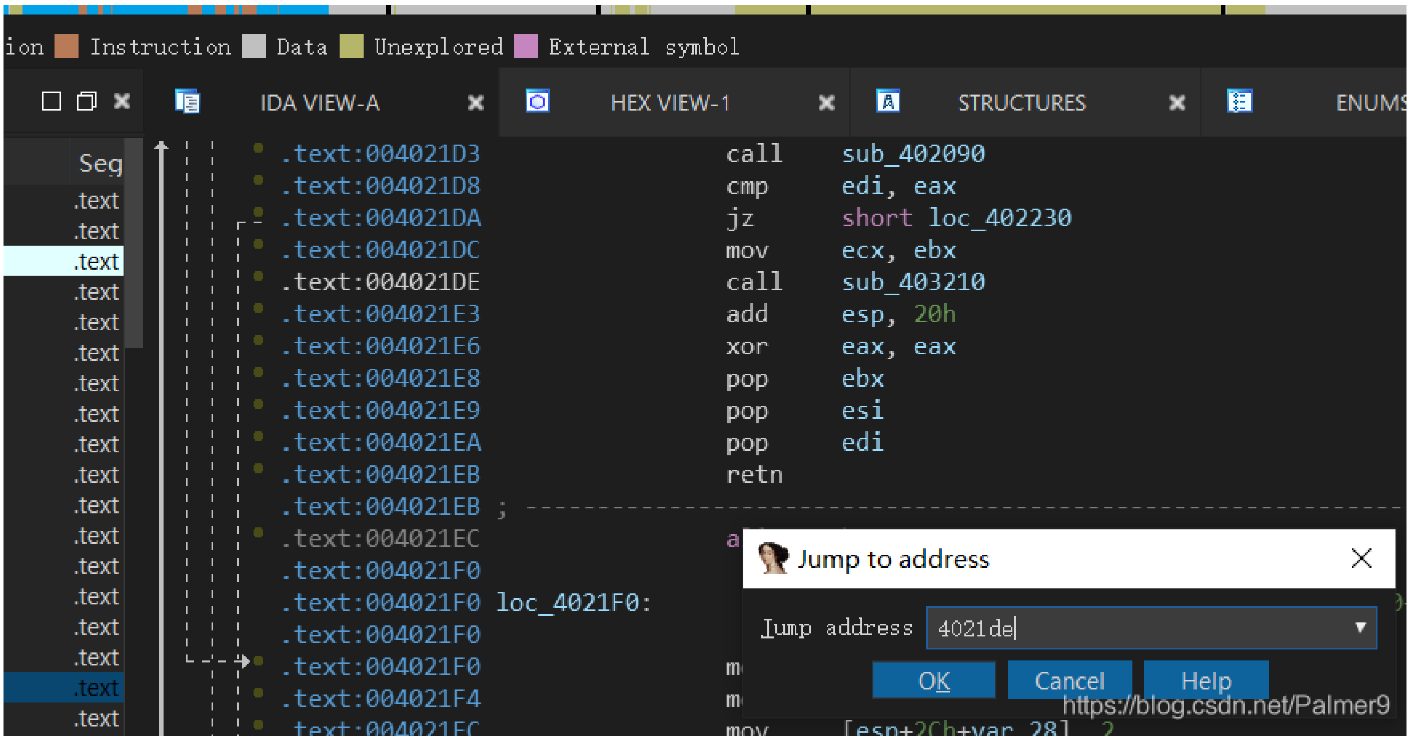


```
003F6000 00001000                  数据块 于  Priv RW    RW
003FD000 00002000                             Priv RW    RW
003FF000 00001000                  数据块 于  Priv RW    RW
00400000 00001000 Login            PE 文件头  Imag R     RWE
```



```
00401000 00004000 Login  .text     SFX,代码   Imag R   RWE
00405000 00001000 Login  .data     数据       Imag R   RWE
00406000 00001000 Login  .rdata               Imag R   RWE
00407000 00002000 Login  .eh_fram             Imag R   RWE
00409000 00001000 Login  .bss                 Imag R   RWE
0040A000 00003000 Login  .idata    输入表     Imag R   RWE
0040D000 00001000 Login  .CRT                 Imag R   RWE
0040E000 00001000 Login  .tls                 Imag R   RWE
005F9000 00002000                             Priv RW  保护 RW
005FB000 00015000                  堆栈 于    Priv RW  保护 RW
```

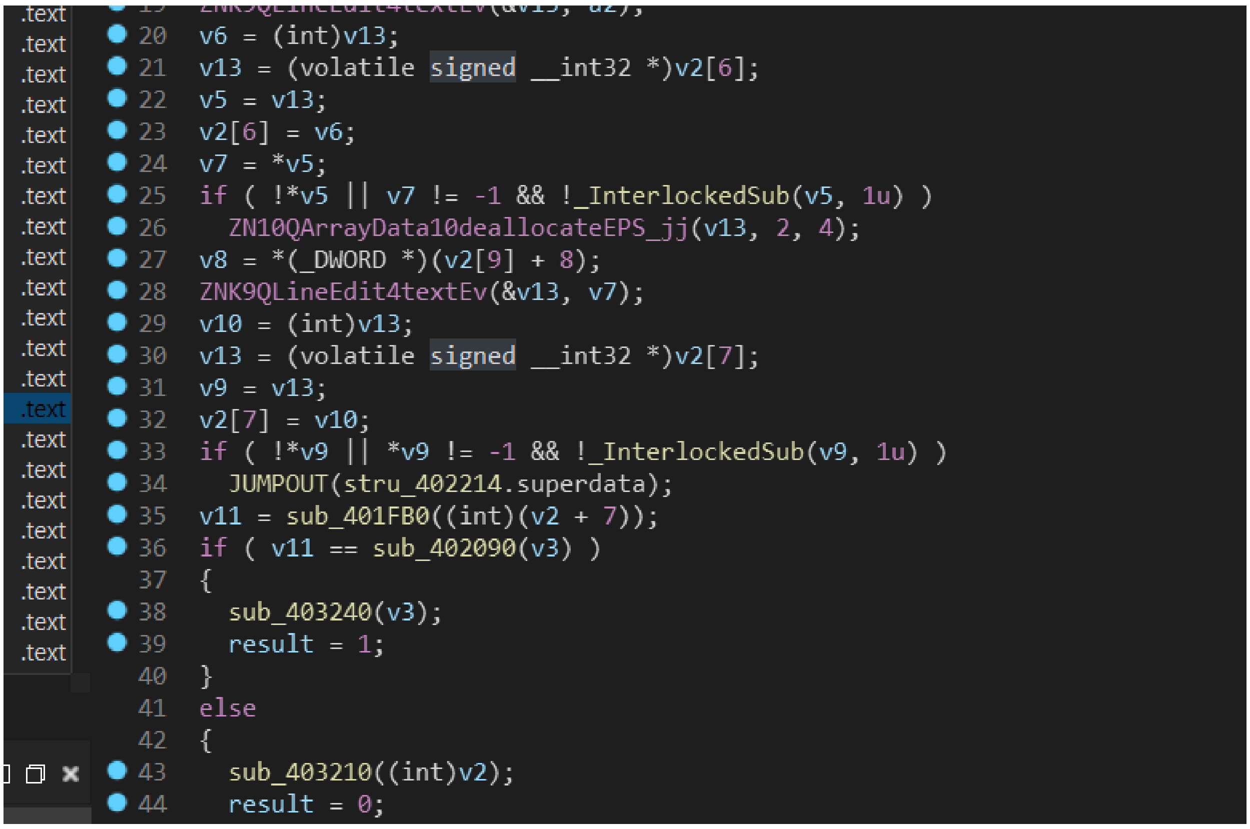
用IDA找到弹窗call地址0x00402ade
快捷键G打开跳转地址窗口



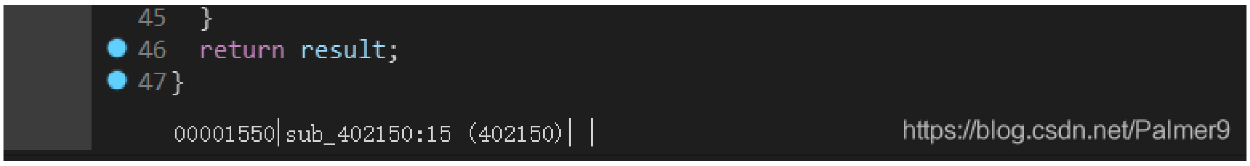
```
.text:004021D3                  call    sub_402090
.text:004021D8                  cmp     edi, eax
.text:004021DA                  jz      short loc_402230
.text:004021DC                  mov     ecx, ebx
.text:004021DE                  call    sub_403210
.text:004021E3                  add     esp, 20h
.text:004021E6                  xor     eax, eax
.text:004021E8                  pop     ebx
.text:004021E9                  pop     esi
.text:004021EA                  pop     edi
.text:004021EB                  retn
.text:004021EB ; ---------------------------------------------------------------
.text:004021EC
.text:004021F0
.text:004021F0 loc_4021F0:
.text:004021F0
.text:004021F0
.text:004021F4
```

Jump to address
Jump address 4021de
OK Cancel Help

F5查看伪代码



```
v6 = (int)v13;
v13 = (volatile signed __int32 *)v2[6];
v5 = v13;
v2[6] = v6;
v7 = *v5;
if ( !*v5 || v7 != -1 && !_InterlockedSub(v5, 1u) )
  ZN10QArrayData10deallocateEPS_jj(v13, 2, 4);
v8 = *(_DWORD *)(v2[9] + 8);
ZNK9QLineEdit4textEv(&v13, v7);
v10 = (int)v13;
v13 = (volatile signed __int32 *)v2[7];
v9 = v13;
v2[7] = v10;
if ( !*v9 || *v9 != -1 && !_InterlockedSub(v9, 1u) )
  JUMPOUT(stru_402214.superdata);
v11 = sub_401FB0((int)(v2 + 7));
if ( v11 == sub_402090(v3) )
{
  sub_403240(v3);
  result = 1;
}
else
{
  sub_403210((int)v2);
  result = 0;
```

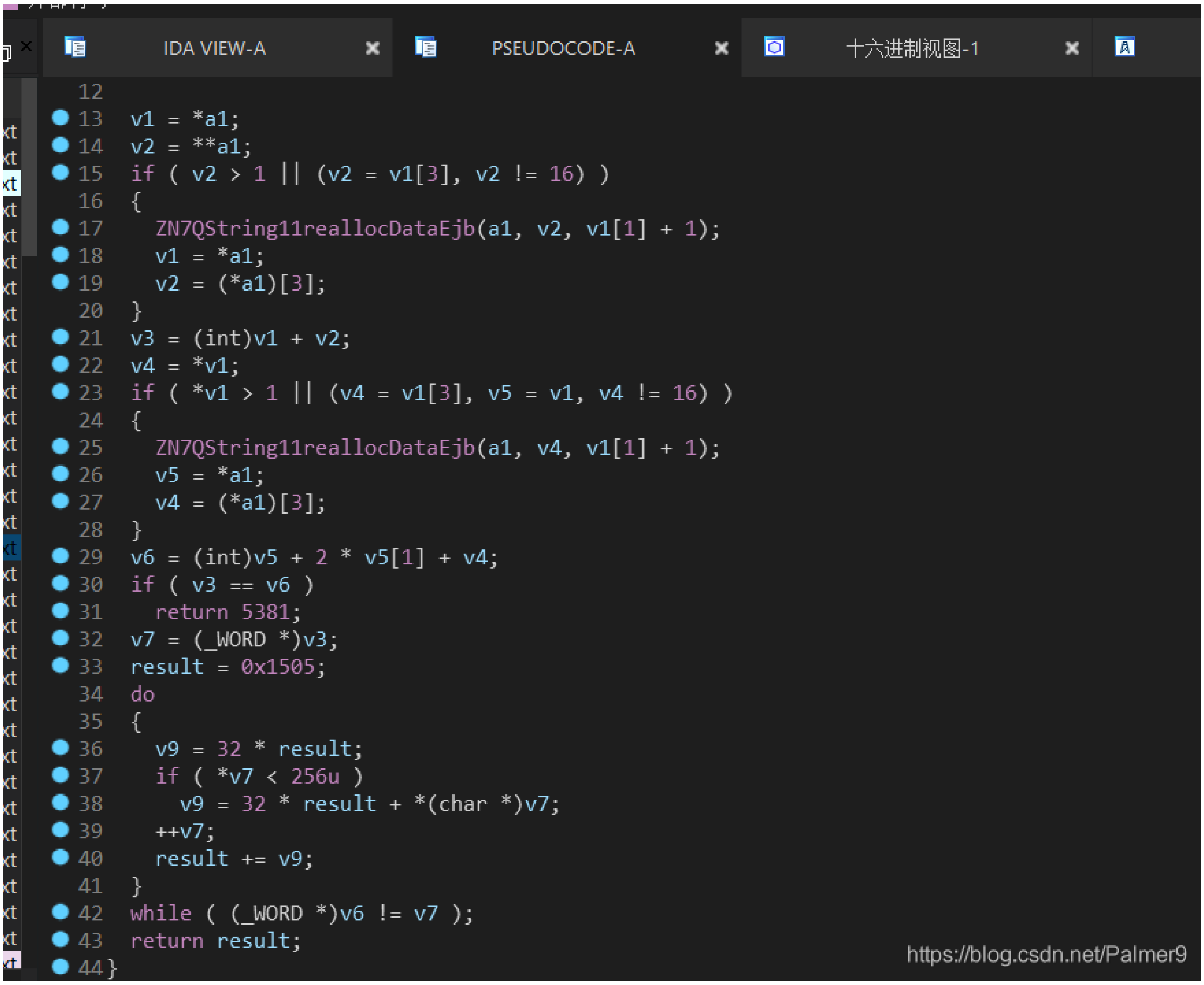


```
45   }
46   return result;
47 }
```

```
00001550|sub_402150:15 (402150)|
```

https://blog.csdn.net/Palmer9

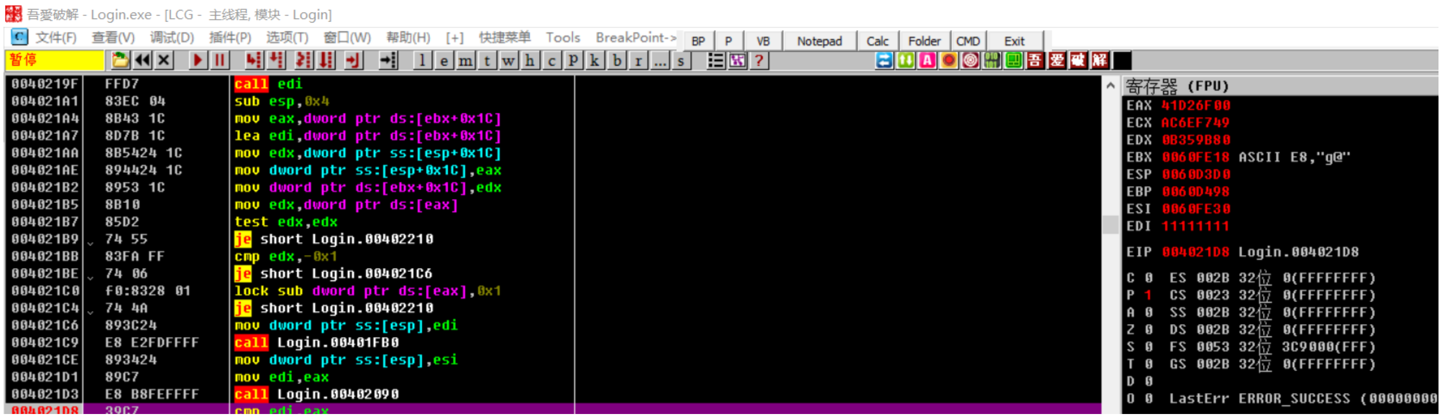
发现判断是否进行错误弹窗代码的条件是402090函数参与的，双击进入查看伪代码



```
12
13  v1 = *a1;
14  v2 = **a1;
15  if ( v2 > 1 || (v2 = v1[3], v2 != 16) )
16  {
17    ZN7QString11reallocDataEjb(a1, v2, v1[1] + 1);
18    v1 = *a1;
19    v2 = (*a1)[3];
20  }
21  v3 = (int)v1 + v2;
22  v4 = *v1;
23  if ( *v1 > 1 || (v4 = v1[3], v5 = v1, v4 != 16) )
24  {
25    ZN7QString11reallocDataEjb(a1, v4, v1[1] + 1);
26    v5 = *a1;
27    v4 = (*a1)[3];
28  }
29  v6 = (int)v5 + 2 * v5[1] + v4;
30  if ( v3 == v6 )
31    return 5381;
32  v7 = (_WORD *)v3;
33  result = 0x1505;
34  do
35  {
36    v9 = 32 * result;
37    if ( *v7 < 256u )
38      v9 = 32 * result + *(char *)v7;
39    ++v7;
40    result += v9;
41  }
42  while ( (_WORD *)v6 != v7 );
43  return result;
44 }
```

https://blog.csdn.net/Palmer9

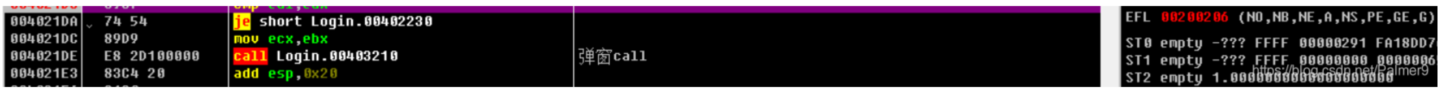
加上OD调试可知此处将输入的用户名的每个字符取出，在35~41行的循环里，对result进行变换最终返回result
因为用户名题目已经给出，所以返回的result为一个定值`0x41d26f00`
返回OD，也就是eax的值为`0x41d26f00`，而eax与edi进行比较，来控制下面的跳转，edi是在0x4021d1地址处被eax赋值的。
而eax是0x4021c9地址处调用的`0x401fb0`call的返回值。



吾爱破解 - Login.exe - [LCG - 主线程, 模块 - Login]
文件(F) 查看(V) 调试(D) 插件(P) 选项(T) 窗口(W) 帮助(H) [+] 快捷菜单 Tools BreakPoint-> BP P VB Notepad Calc Folder CMD Exit
暂停
l e m t w h c P k b r ... s

```
0040219F    FFD7            call edi
004021A1    83EC 04         sub esp,0x4
004021A4    8B43 1C         mov eax,dword ptr ds:[ebx+0x1C]
004021A7    8D7B 1C         lea edi,dword ptr ds:[ebx+0x1C]
004021AA    8B5424 1C       mov edx,dword ptr ss:[esp+0x1C]
004021AE    894424 1C       mov dword ptr ss:[esp+0x1C],eax
004021B2    8953 1C         mov dword ptr ds:[ebx+0x1C],edx
004021B5    8B10            mov edx,dword ptr ds:[eax]
004021B7    85D2            test edx,edx
004021B9  ⌄ 74 55           je short Login.00402210
004021BB    83FA FF         cmp edx,-0x1
004021BE  ⌄ 74 06           je short Login.004021C6
004021C0    F0:8328 01      lock sub dword ptr ds:[eax],0x1
004021C4  ⌄ 74 4A           je short Login.00402210
004021C6    893C24          mov dword ptr ss:[esp],edi
004021C9    E8 E2FDFFFF     call Login.00401FB0
004021CE    893424          mov dword ptr ss:[esp],esi
004021D1    89C7            mov edi,eax
004021D3    E8 B8FEFFFF     call Login.00402090
004021D8    39C7            cmp edi,eax
```

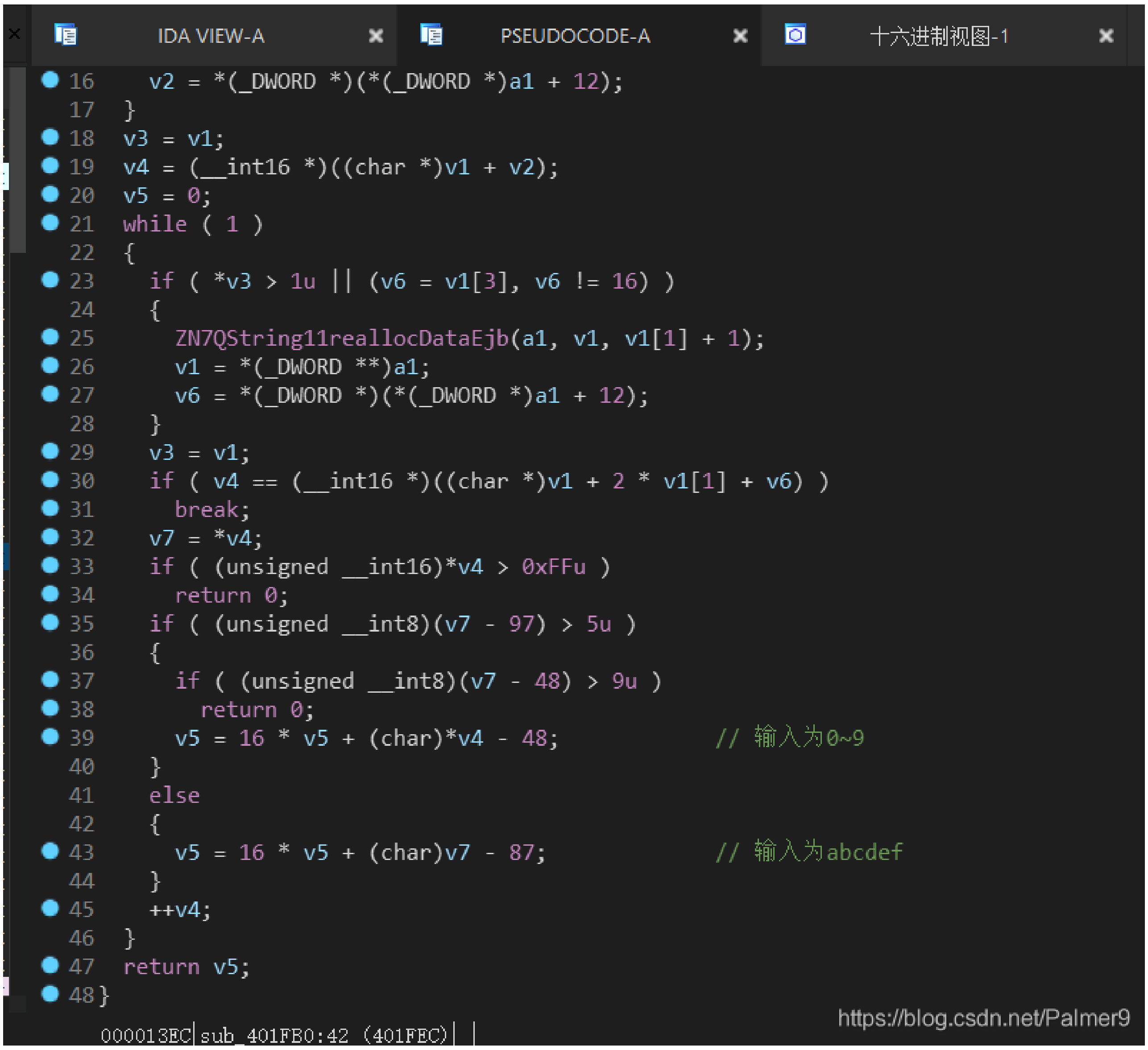
```
寄存器 (FPU)
EAX 41D26F00
ECX AC6EF749
EDX 0B359B80
EBX 0060FE18 ASCII E8,"g@"
ESP 0060D3D0
EBP 0060D498
ESI 0060FE30
EDI 11111111

EIP 004021D8 Login.004021D8

C 0  ES 002B 32位 0(FFFFFFFF)
P 1  CS 0023 32位 0(FFFFFFFF)
A 0  SS 002B 32位 0(FFFFFFFF)
Z 0  DS 002B 32位 0(FFFFFFFF)
S 0  FS 0053 32位 3C9000(FFF)
T 0  GS 002B 32位 0(FFFFFFFF)
D 0
O 0  LastErr ERROR_SUCCESS (00000000
```



```
004021DA   74 54        je short Login.00402230
004021DC   89D9         mov ecx,ebx
004021DE   E8 2D100000  call Login.00403210          弹窗call
004021E3   83C4 20      add esp,0x20
```

```
EFL 00200206 (NO,NB,NE,A,NS,PE,GE,G)
ST0 empty -??? FFFF 00000291 FA18DD7
ST1 empty -??? FFFF 00000000 0000006
ST2 empty 1.00000000000000000000
```

IDA查看401fb0()函数

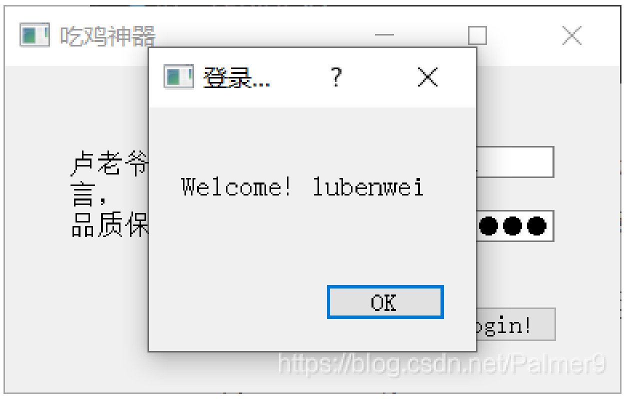


```
    v2 = *(_DWORD *)(*(_DWORD *)a1 + 12);
  }
  v3 = v1;
  v4 = (__int16 *)((char *)v1 + v2);
  v5 = 0;
  while ( 1 )
  {
    if ( *v3 > 1u || (v6 = v1[3], v6 != 16) )
    {
      ZN7QString11reallocDataEjb(a1, v1, v1[1] + 1);
      v1 = *(_DWORD **)a1;
      v6 = *(_DWORD *)(*(_DWORD *)a1 + 12);
    }
    v3 = v1;
    if ( v4 == (__int16 *)((char *)v1 + 2 * v1[1] + v6) )
      break;
    v7 = *v4;
    if ( (unsigned __int16)*v4 > 0xFFu )
      return 0;
    if ( (unsigned __int8)(v7 - 97) > 5u )
    {
      if ( (unsigned __int8)(v7 - 48) > 9u )
        return 0;
      v5 = 16 * v5 + (char)*v4 - 48;             // 输入为0~9
    }
    else
    {
      v5 = 16 * v5 + (char)v7 - 87;              // 输入为abcdef
    }
    ++v4;
  }
  return v5;
}
```

```
000013EC sub_401FB0:42 (401FEC)
```

使用OD调试可知这部分是对输入的密码进行处理。输入只能是0~9,a~f。
（其实就是将输入的十六进制字符串转换为十六进制数值然后返回

若想成功，也就是对密码处理后的返回值是`0x41d26f00`，也就是密码是字符串`"41d26f00"`



Welcome! lubenwei

根据题干输入密码即为flag
最终flag为`flag{41d26f00}`