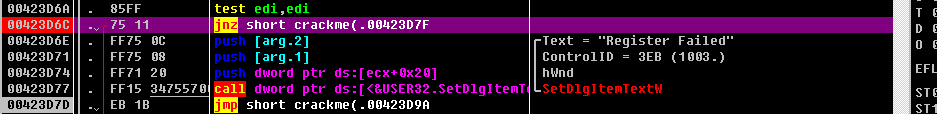


两个不知道的什么段 有tls

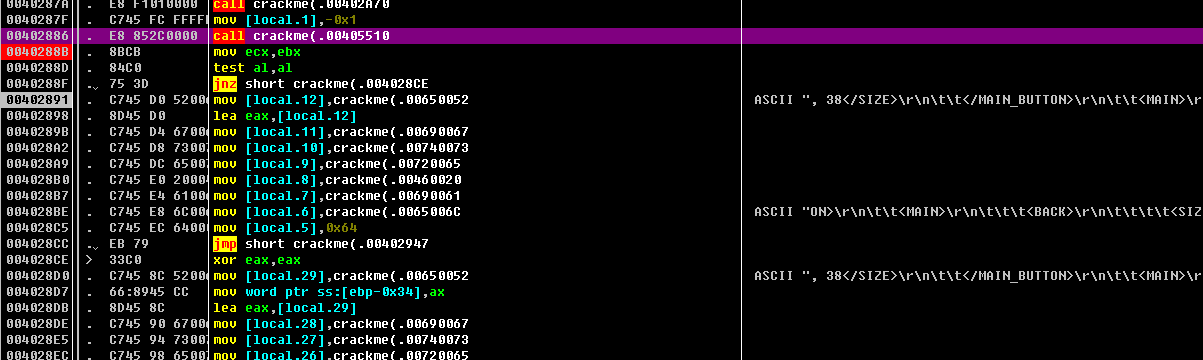
MFC程序 BP GetDlgItem 获取输入



重要分支（待定）

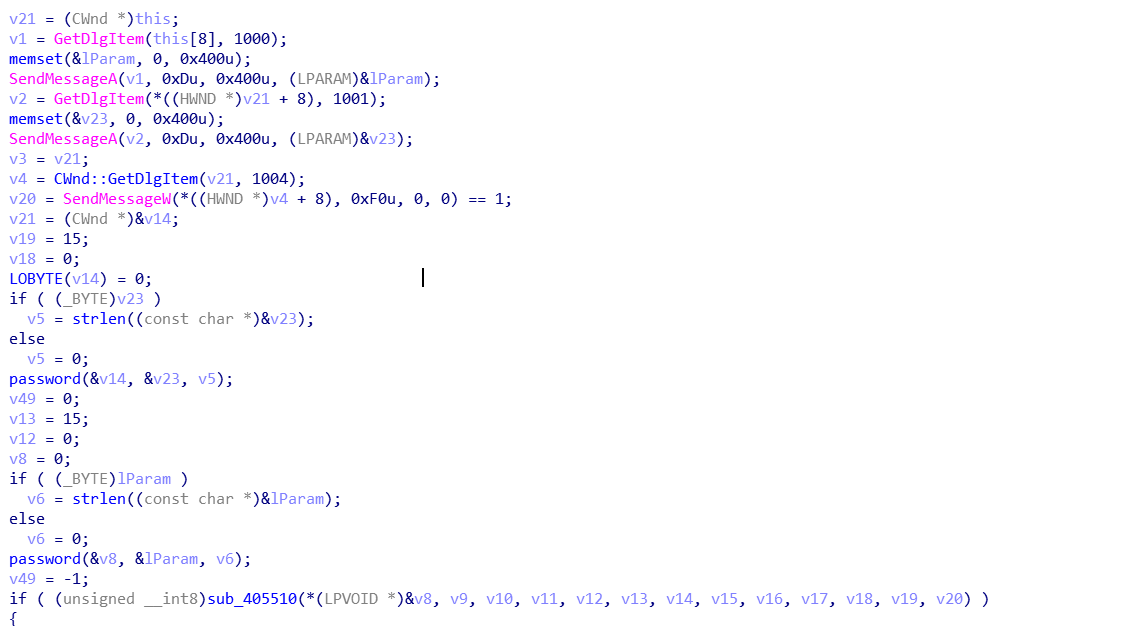
对输入内容内存下内存访问

00402A70 跟踪



这个有点怪的， 补充：实际是 结果判断

检索到主流程



坐实了，那个分支就是结果关键



算法分析

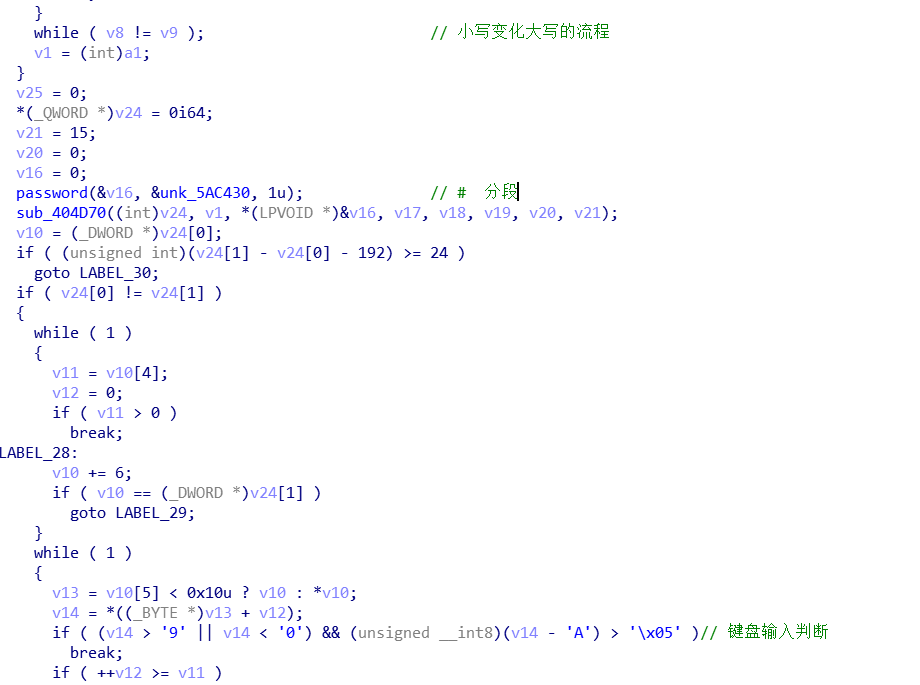
实际是用户名长度判断

补位

帐号要求39位



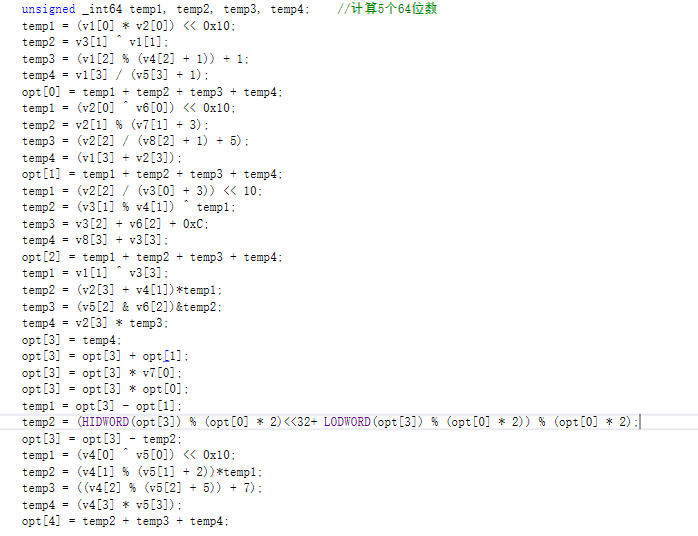
条件 



37-7位 # 末尾补 # 8组 然后

头皮发麻的计算

返回64位数 5个



Emmm，base80

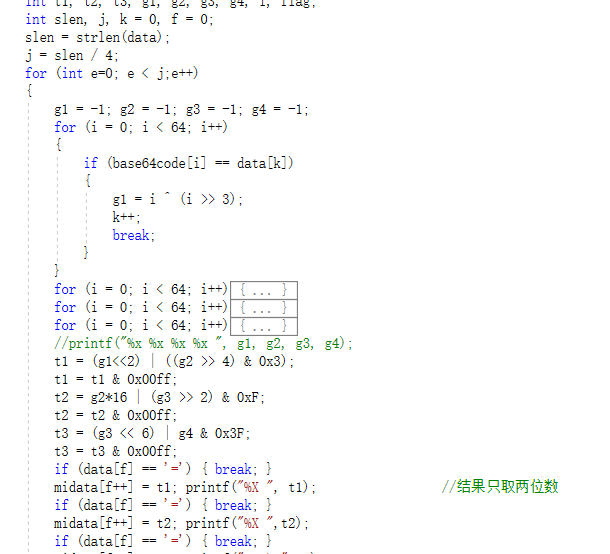
对以下字符串

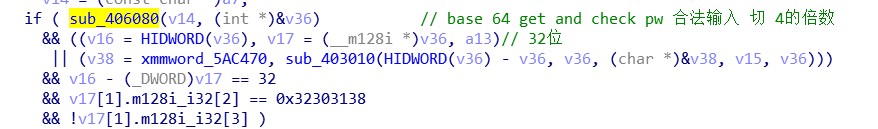
YtTJUVEn0$HI34y#8rsFewxlm+/u5a^2welcomegslab2018zQfghDRSG@di\*kABZO6Kq79L&CPWvNop

实际截64位

ZO6Kq79L&CPWvNopzQfghDRSG@di\*kAB8rsFewxlm+/u5a^2YtTJUVEn0$HI34y#

对注册码b64计算





对于加密后的regcode作要求

32位 即 输入的48位 最后一位是 =

加密后的regcode 第25-28位 是 8102 （反的 反的 反的 说三遍）

加密后的 regcode 第 29-32 为 0

已知的 1.第30到32位 为0 0 asc 0x30 0x30 对应B64 ZZZ

2.第28到30位 为 0x32 0 0 对应B64 NeZZ

3.第25-28位为 810 对应B64 pKqE

4.最后一组最后一位为 ‘=’

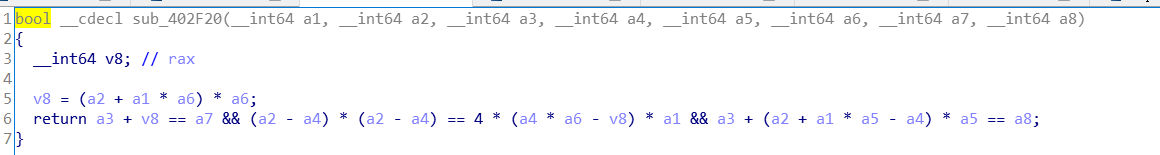
pKqENeZZZZZ=

//吐槽一下这个程序在 xp 和 win10 下不一样的，好毒



对前面username 5组 每组 对应高低位 把 加密后的regcode 也推进站

然后血崩之数学题 结果满足应该就返回true 了



满足以上公式即可



注册码并不唯一 改程序只取计算结果的低64位 作为 公式对比