## 加密程序：

### 加密要求：

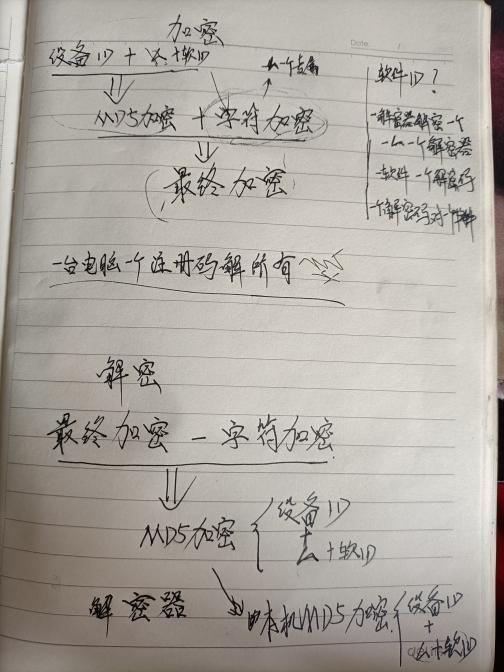
1. 多家公司解密不可混用
2. 一个公司对应一个解密软件。
3. 一个解密器解密该公司所有软件
4. 一台电脑上一个解密文件对应一个软件

### 加密设置：

1. 设备ID+人+软件ID （备注：“人”可不填，所有软件ID不可重复）
2. MD5+字符加密（备注：加密字符需要32位（int），字符加密：代表公司，不可重复）

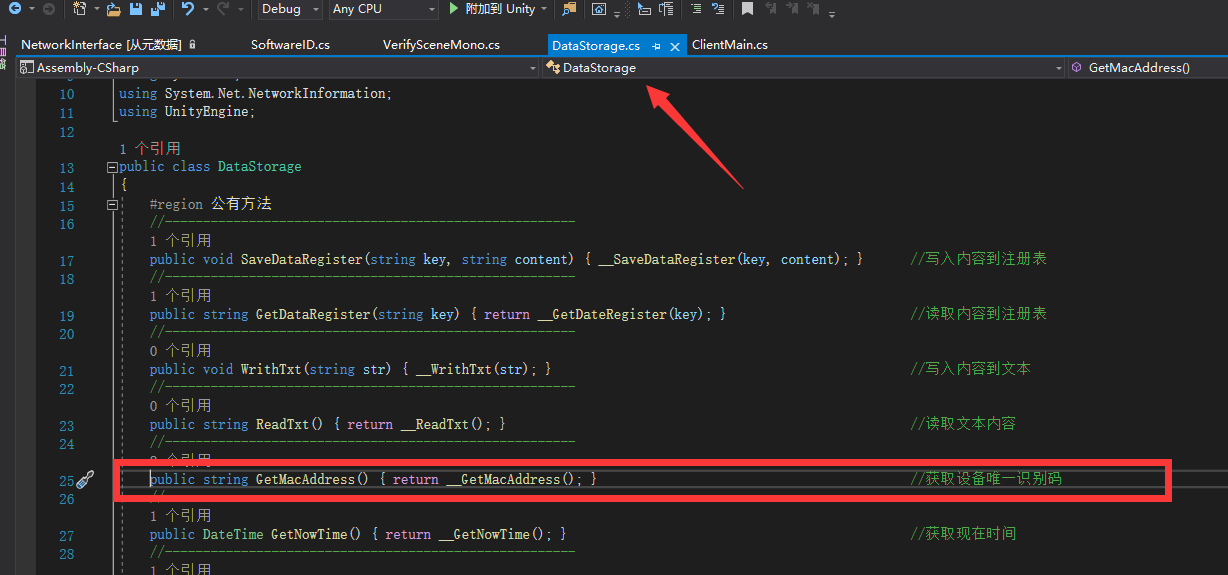
### 解密设置：

1. 解密器的解密字符需要根据加密字符单独设置

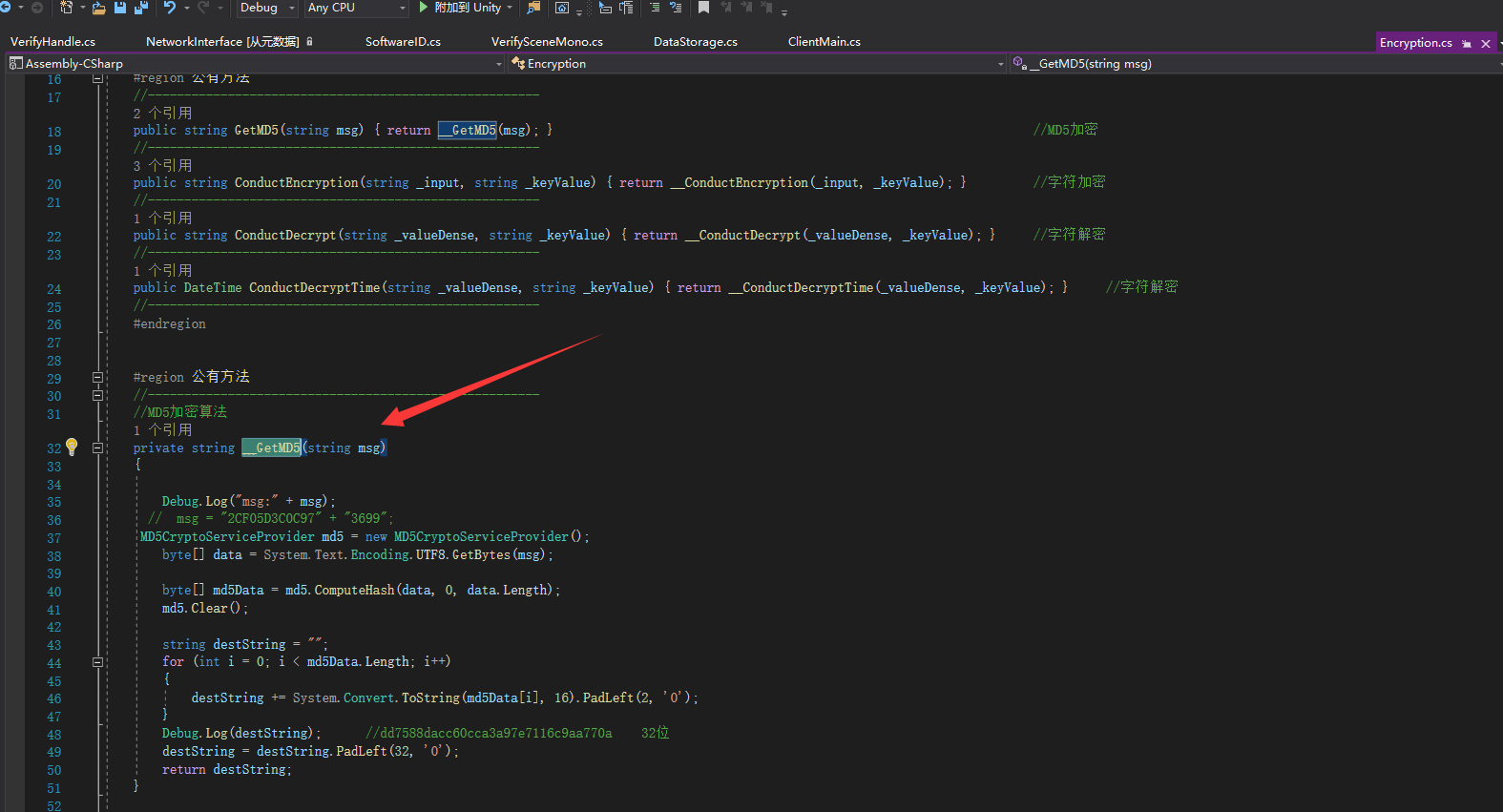


### 一：加密部分

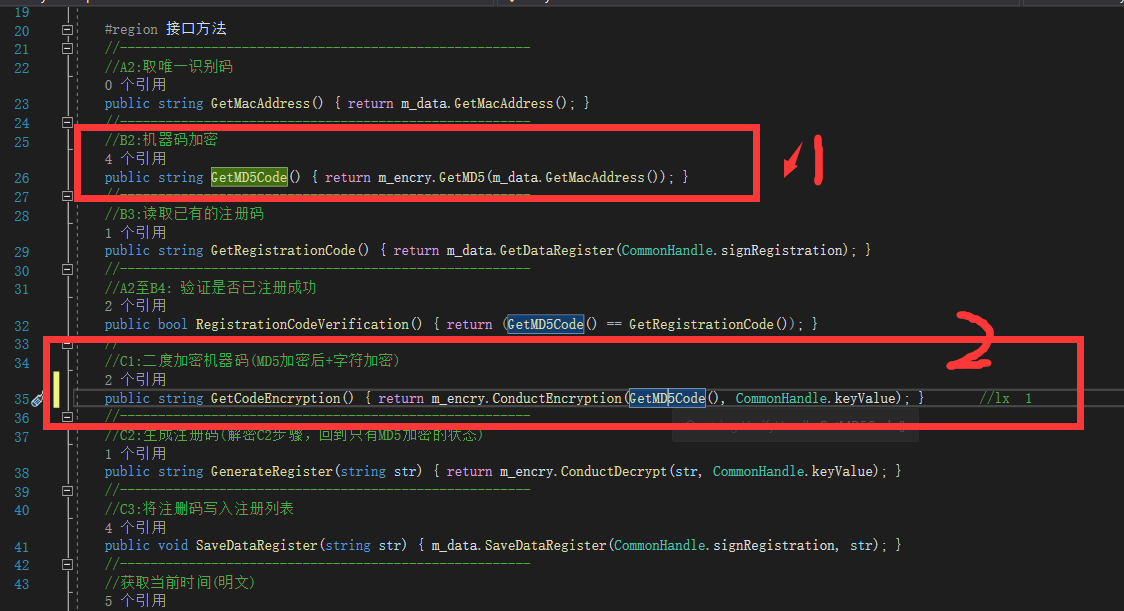
1. 首先获取设备唯一识别码：



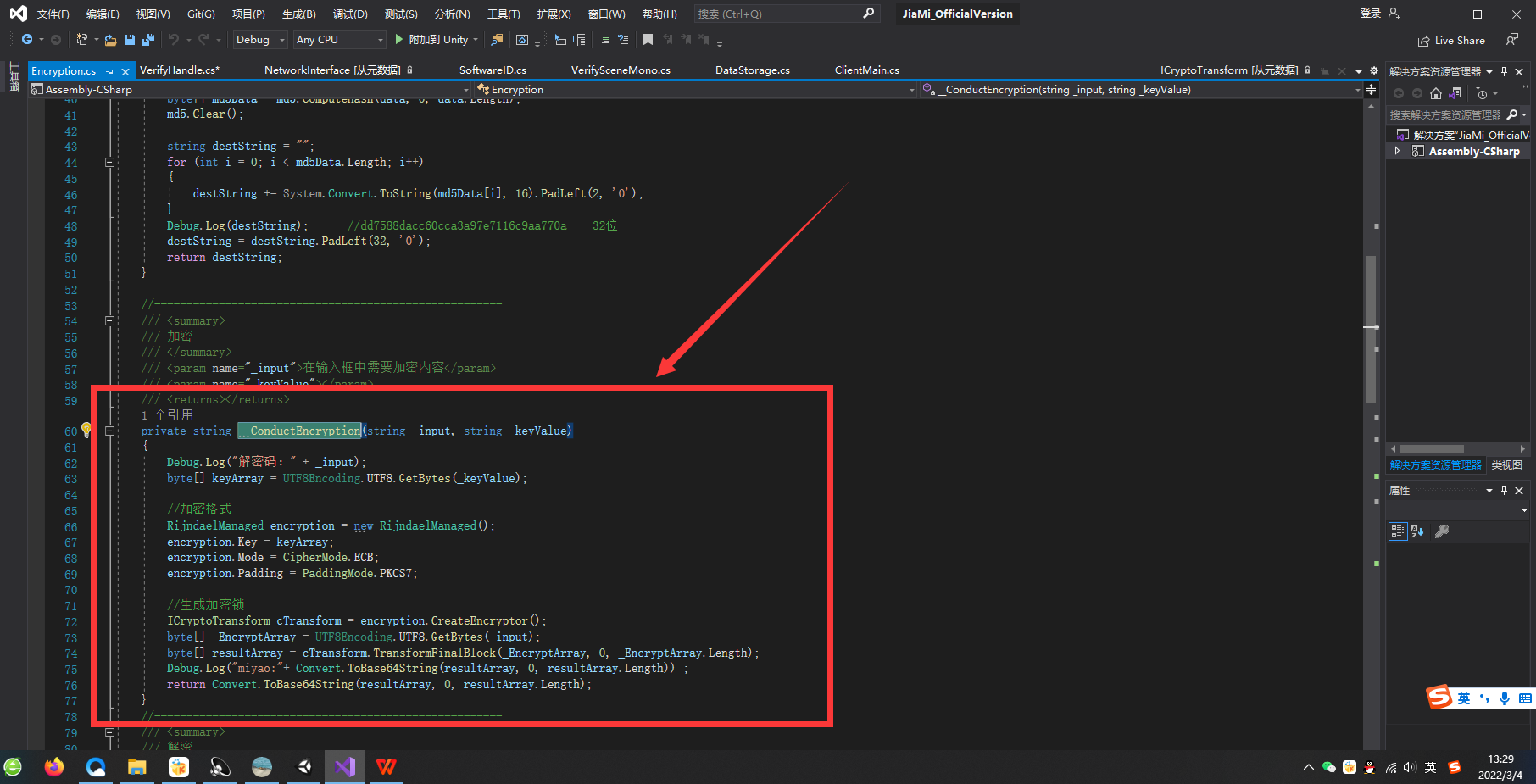
1. 获取到唯一识别码以后（+软件ID），在通过md5加密算法进行加密，获得加密后的32位字符串



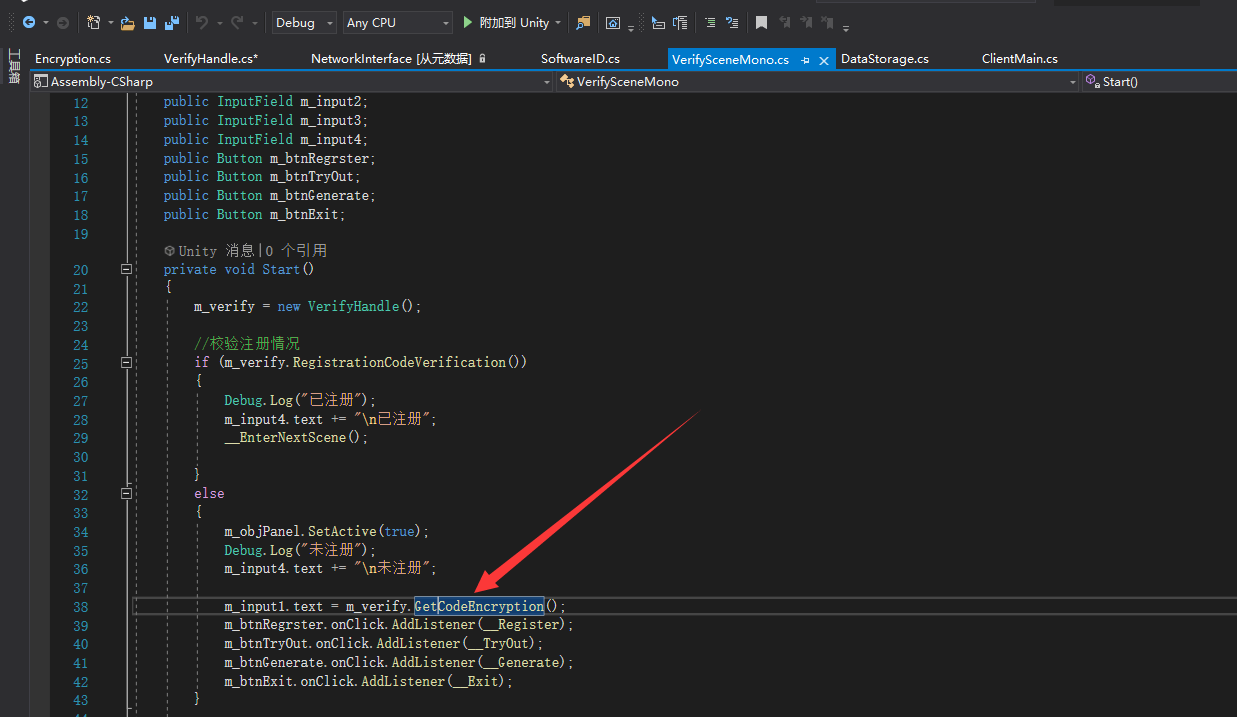
1. md5加密后的机器加密码，在进行二度加密，附加自己设置的32位字符串；



1. 最终生成一个秘钥

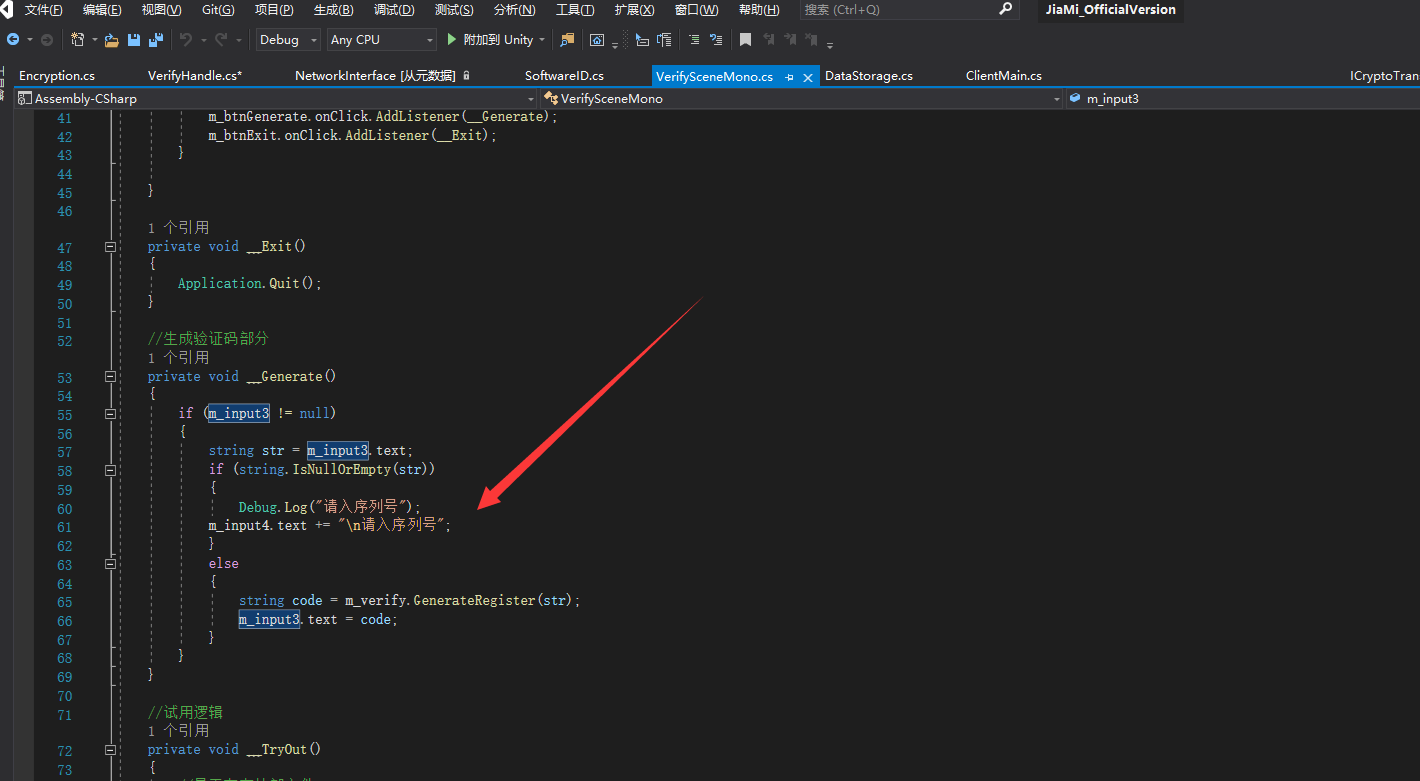


1. 然后在主页拿到这个秘钥，通过这个秘钥去获取解密码

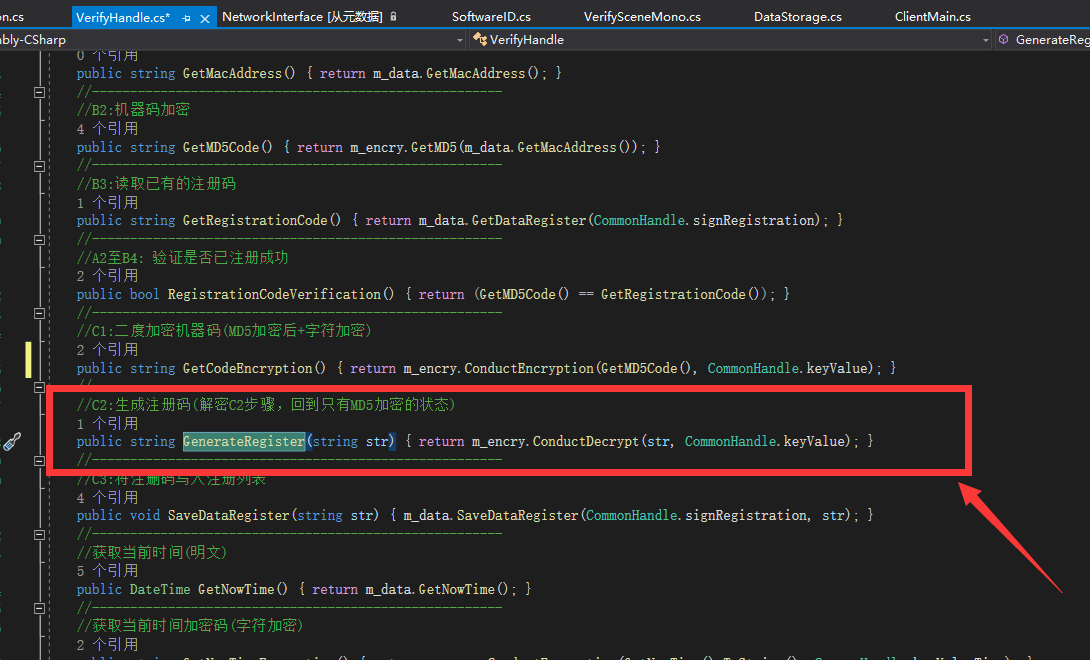


### 二：解密部分

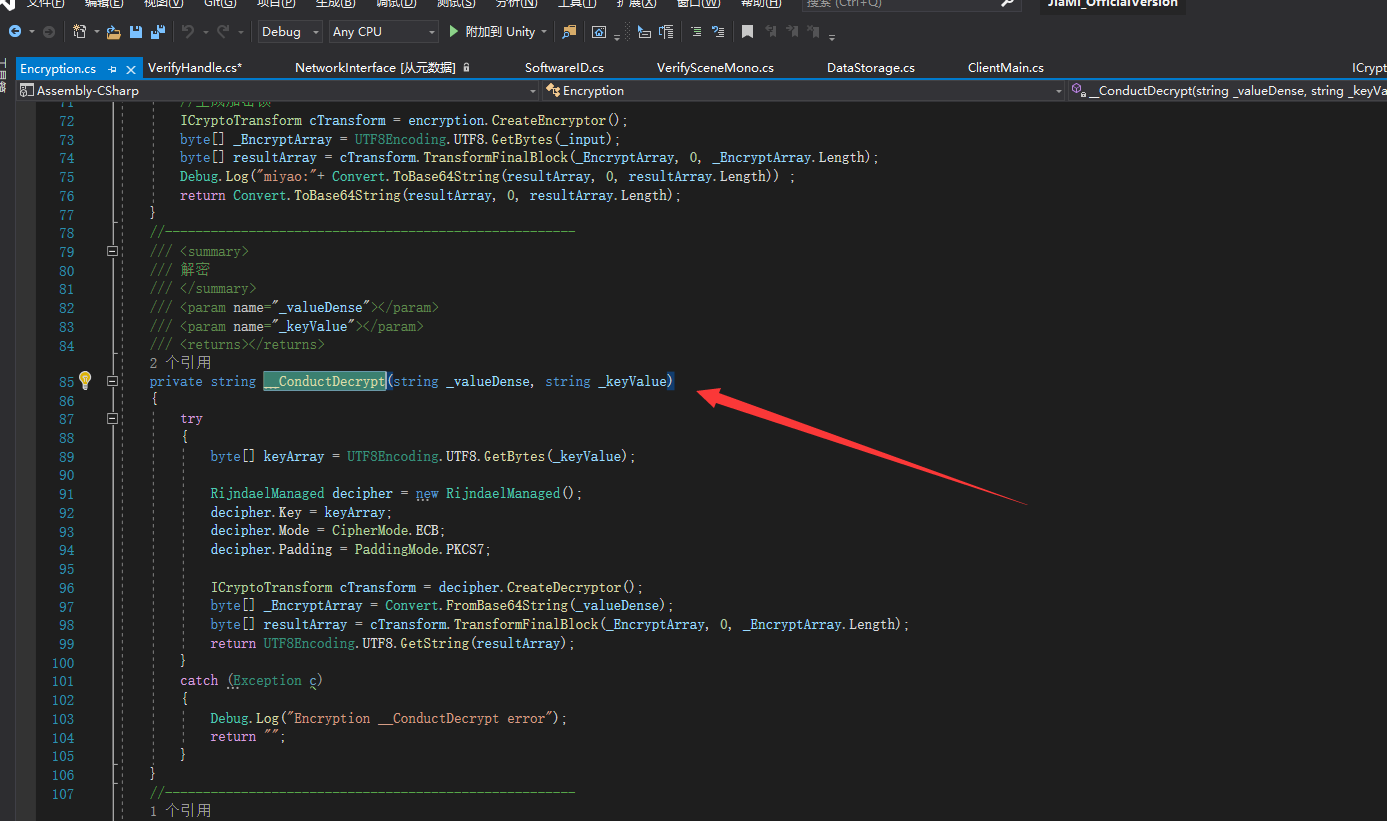
1. 拿到加密的秘钥，放到解码器部分，点击生成验证码（通过解密来生成注册码）



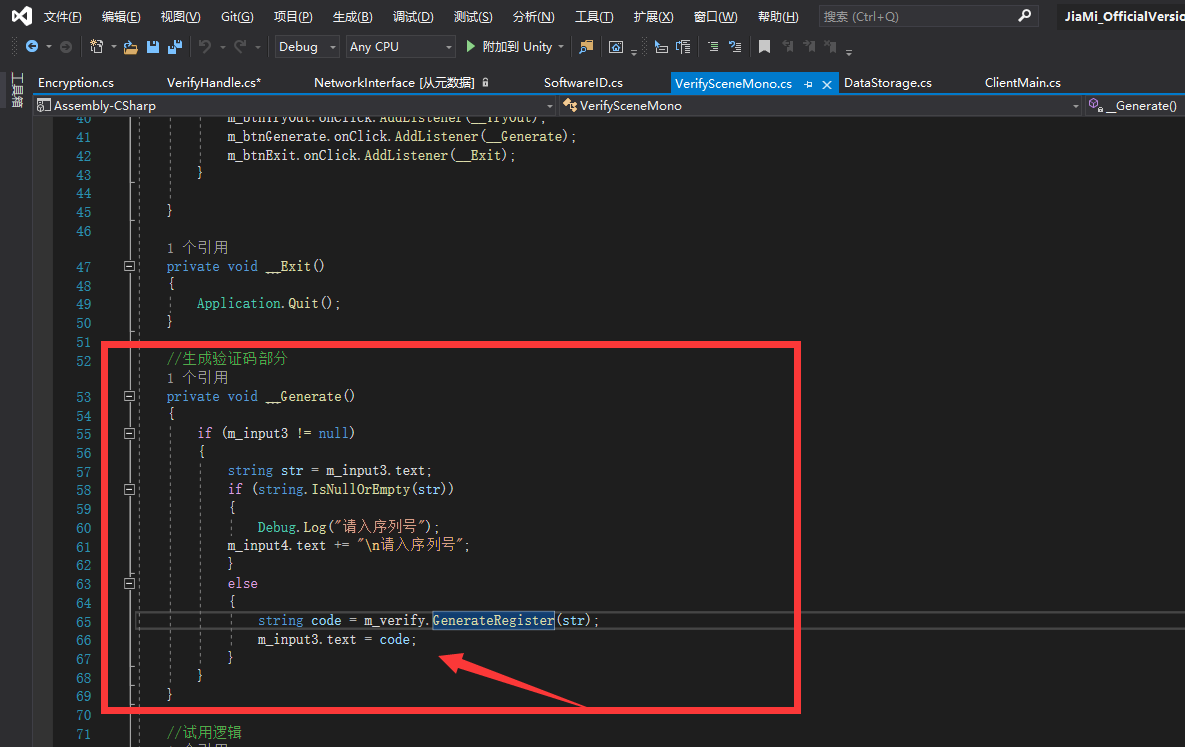
1. 将秘钥解密到md5状态，



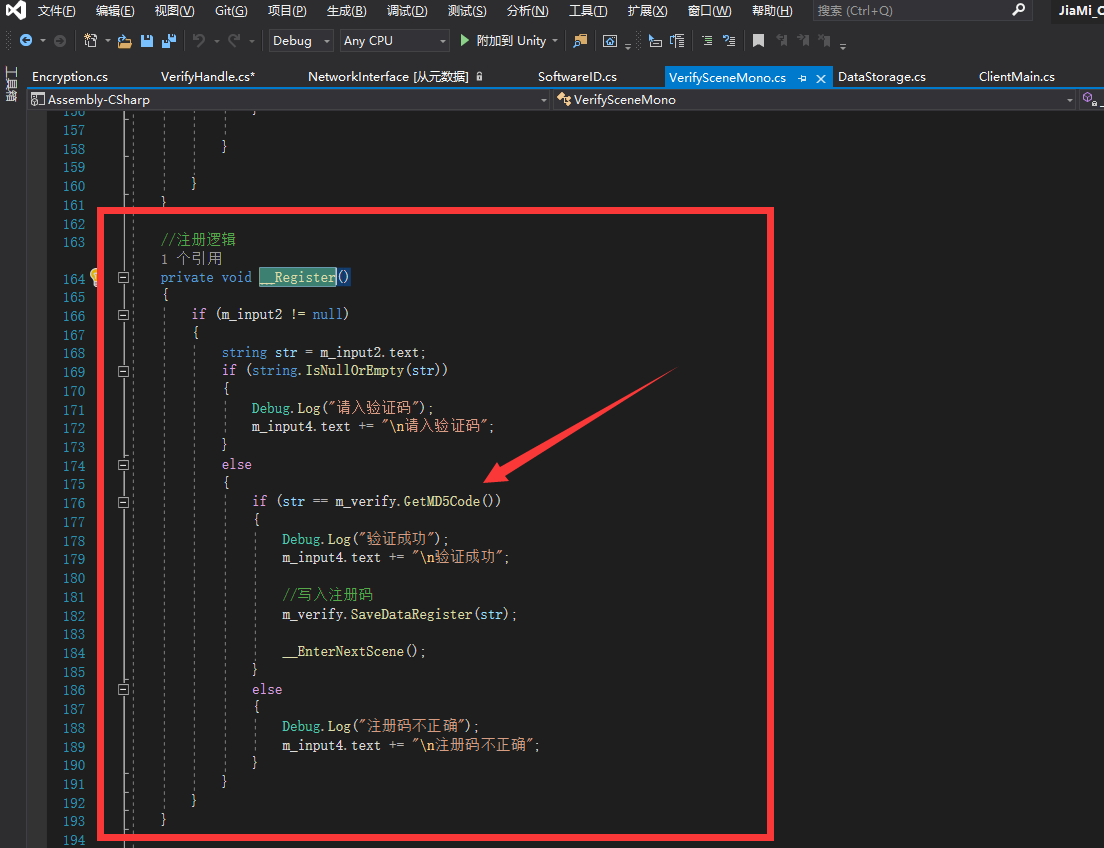
1. 首先对之前加密的“字符加密”进行解密（回到只有MD5加密的状态）



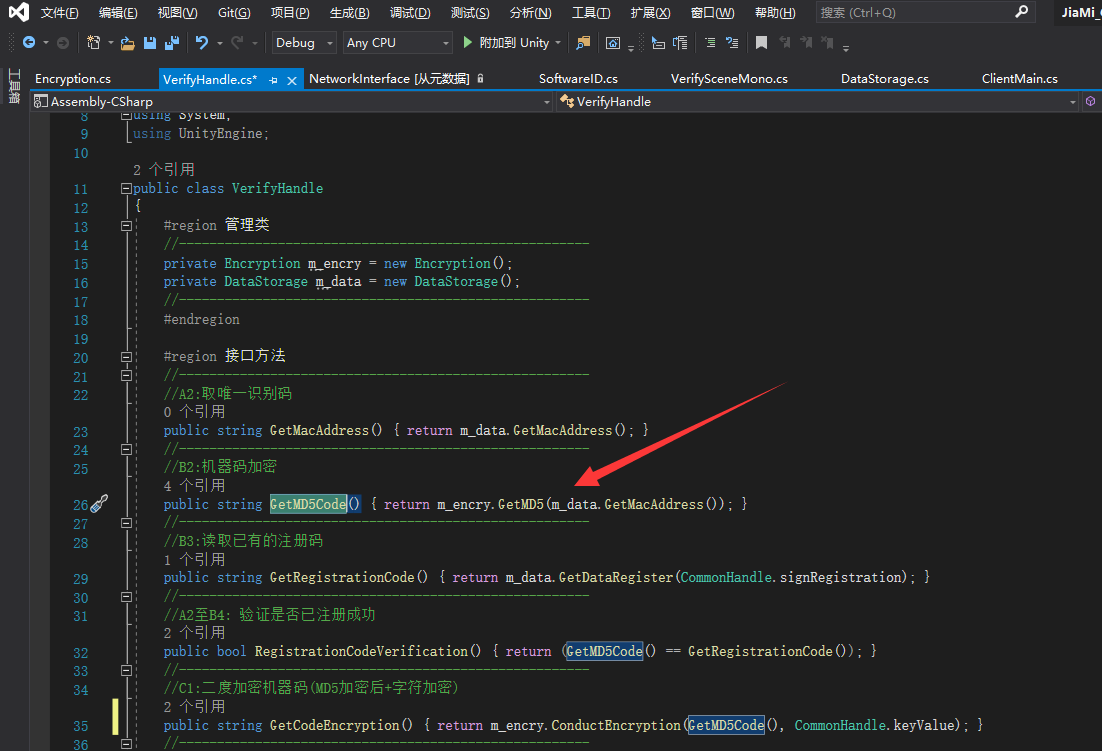
1. 此时解码器部分便得到了验证码



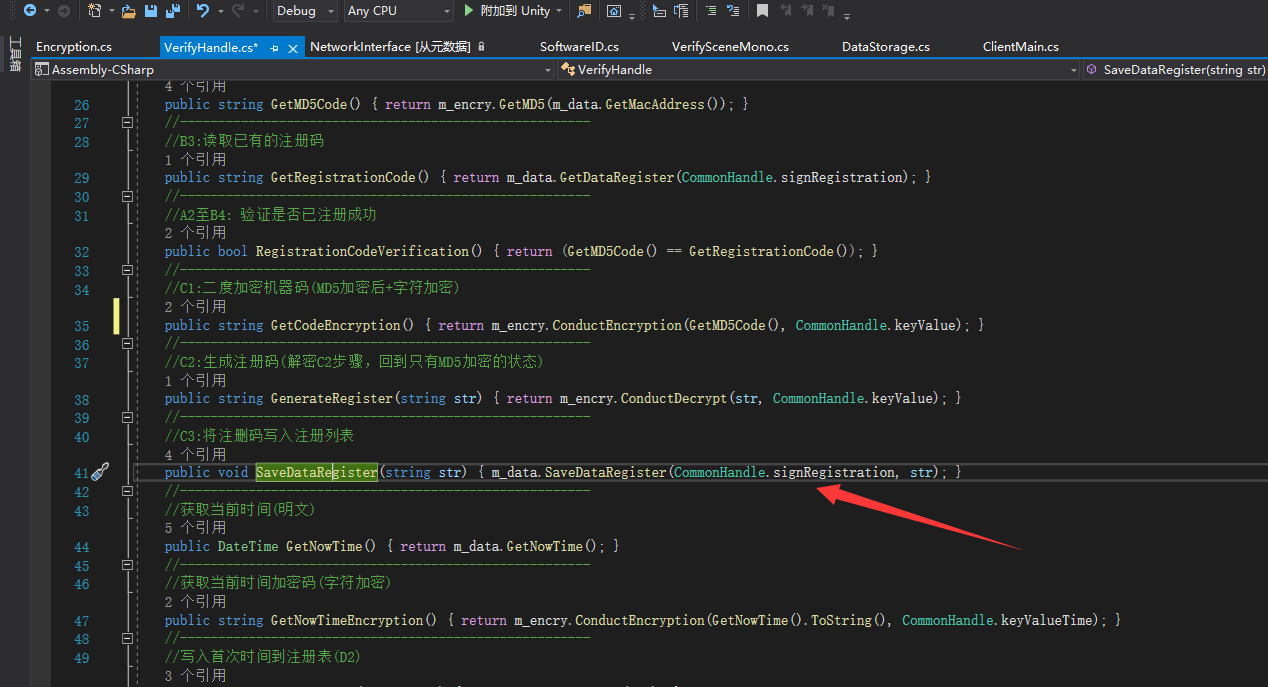
1. 将得到的验证码，放入注册码位置，进行比对验证



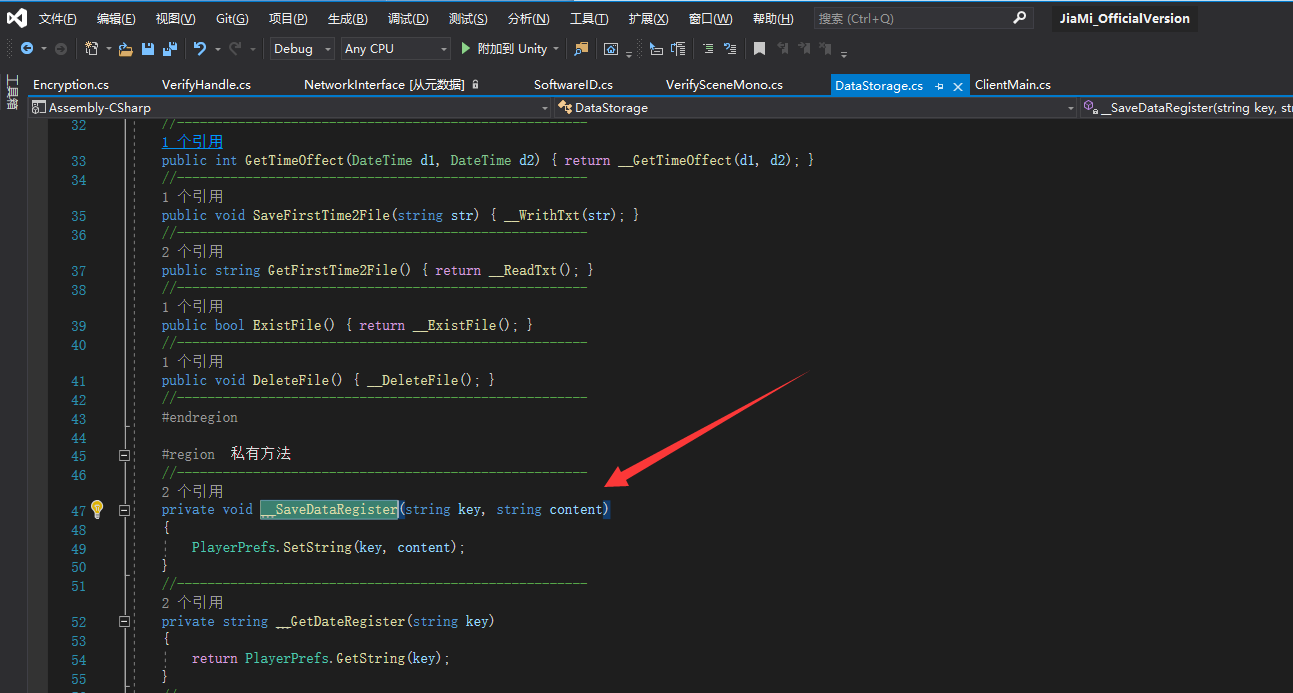
1. 比对的码仍为 机器经过md5加密后的比对码，



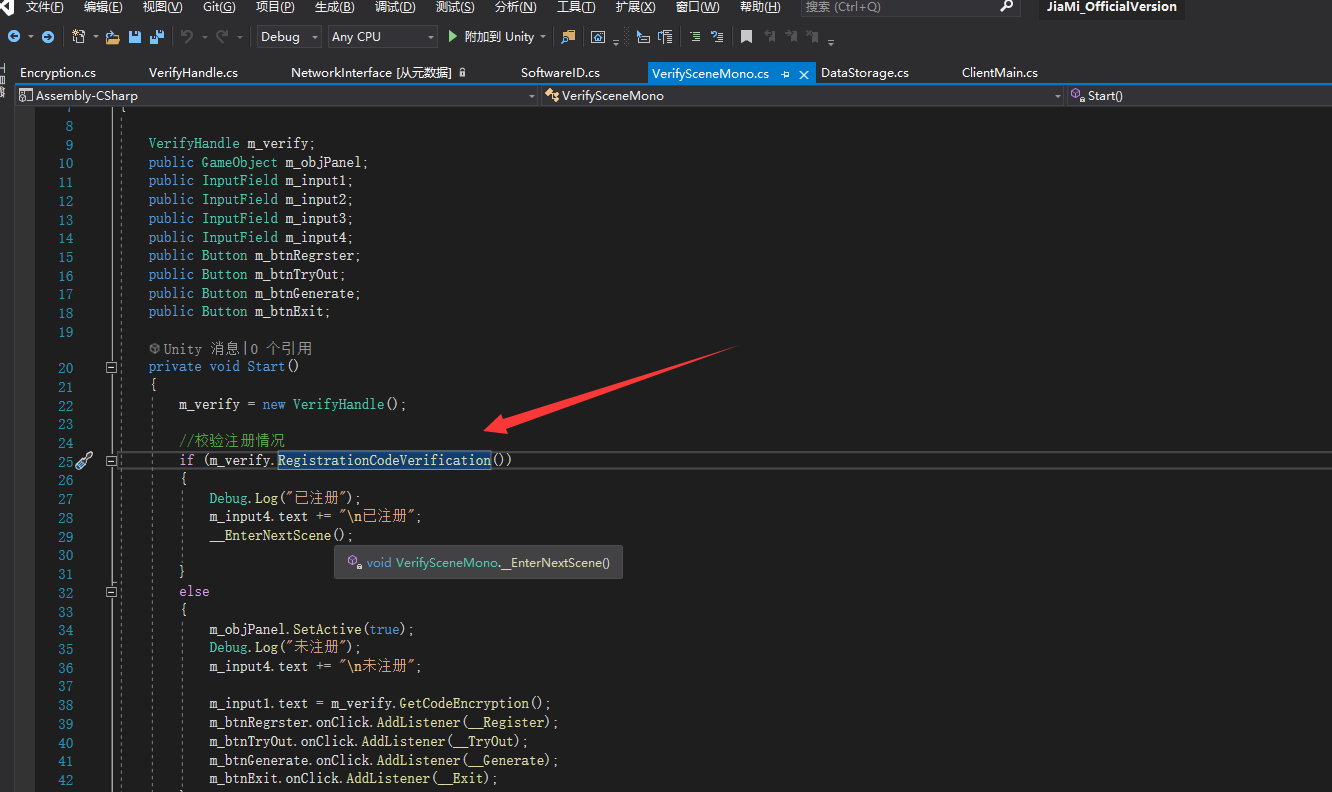
1. 比对成功后，将 注册码 写入注册表（存储在本地）



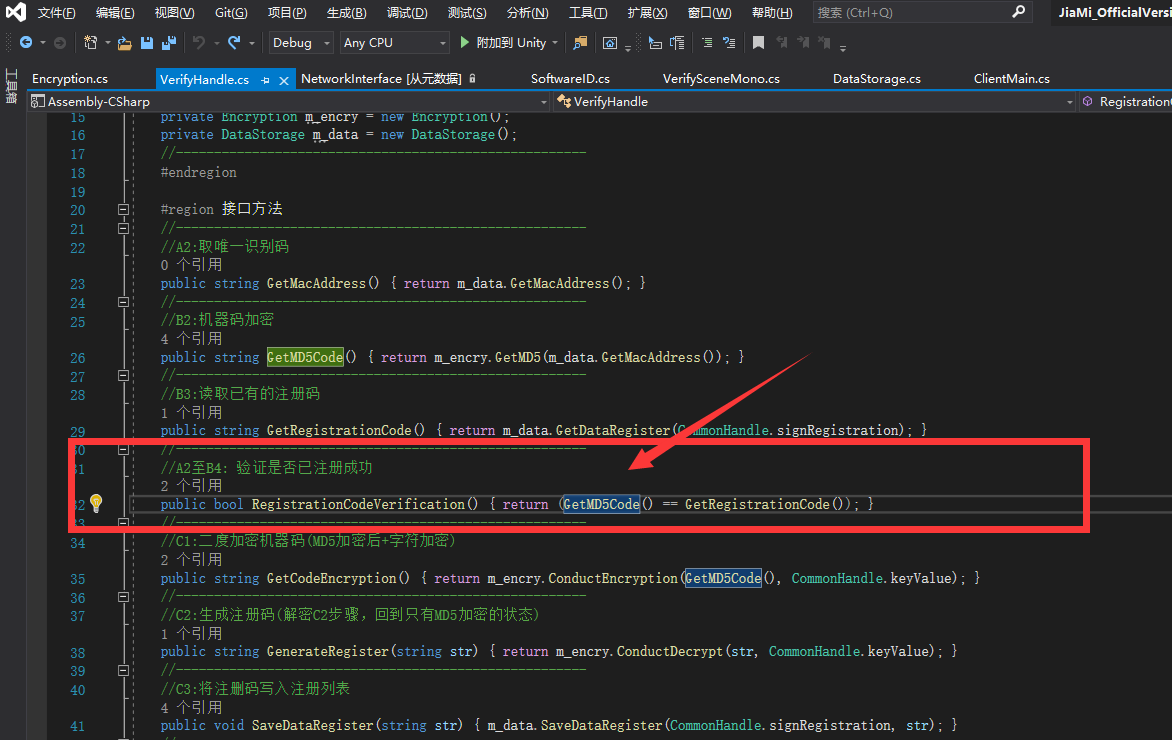
1. 调用PlayerPrefs.SetString(key, content);方法进行存储



1. 二次进入后校验注册情况

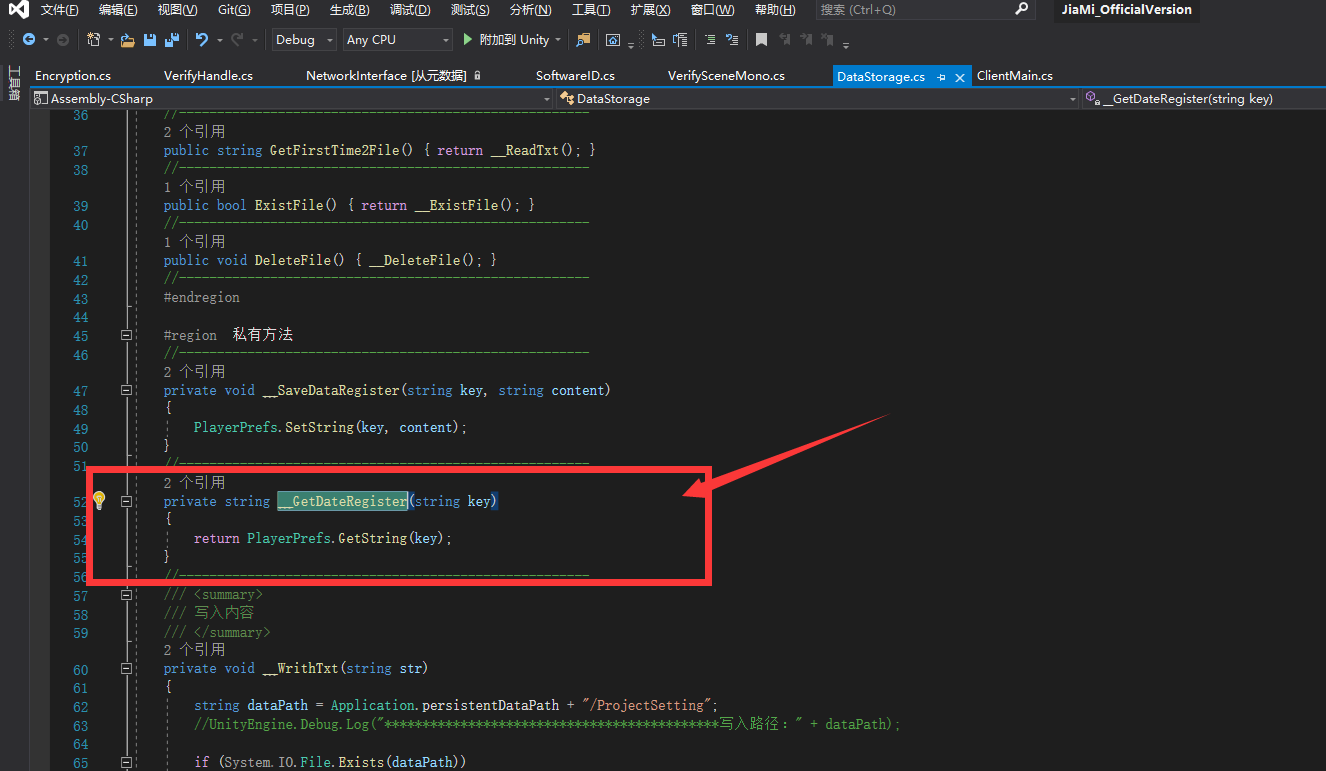


1. 重新获取md5注册码与存储的注册码进行比对



1. 读取当时写入的注册码

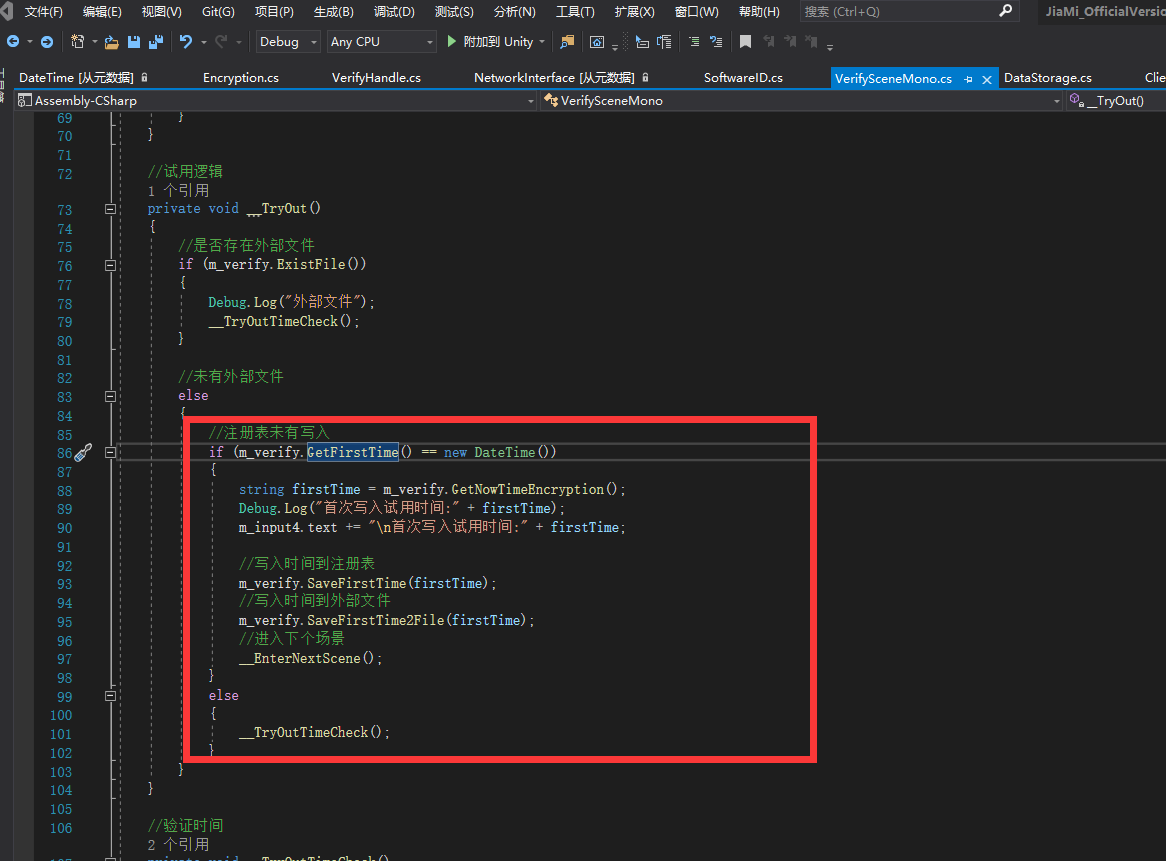
（存储位置https://blog.csdn.net/Monkey\_Xuan/article/details/115518561）



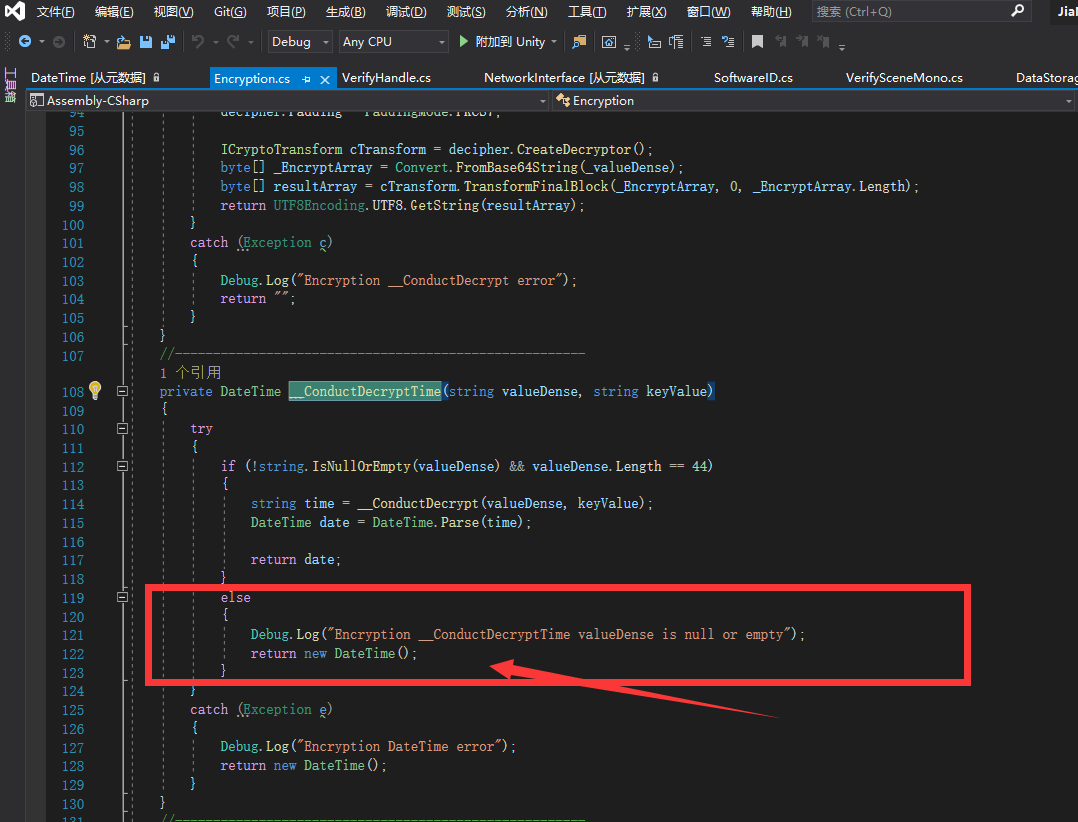
### 三：试用部分

1. 第一次试用，未有外部文件

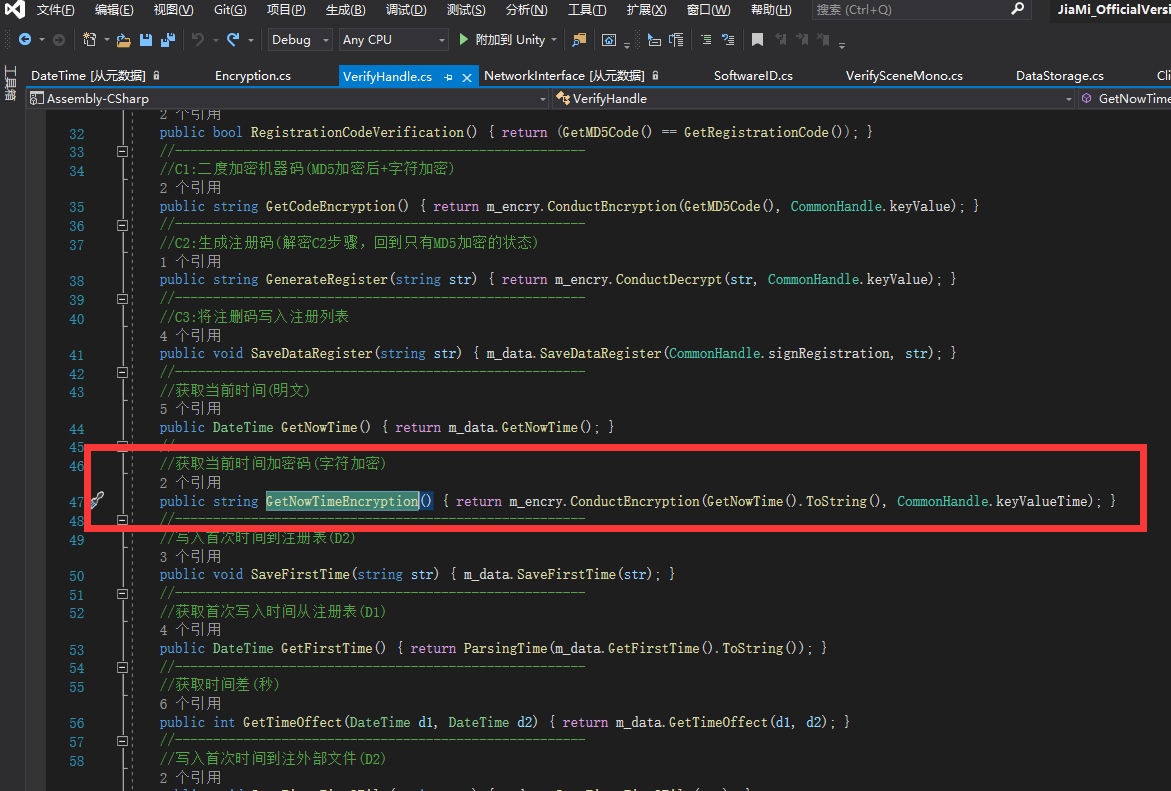
（外部文件位置：C:\Users\Admin\AppData\LocalLow\DefaultCompany\JiaMi\_OfficialVersion）



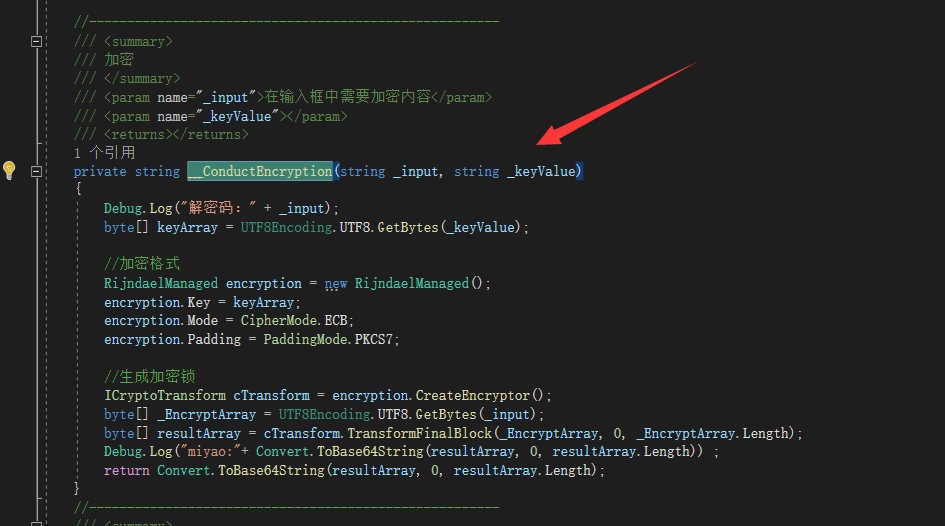
1. 判断中GetFirstTime为返回的当前时间



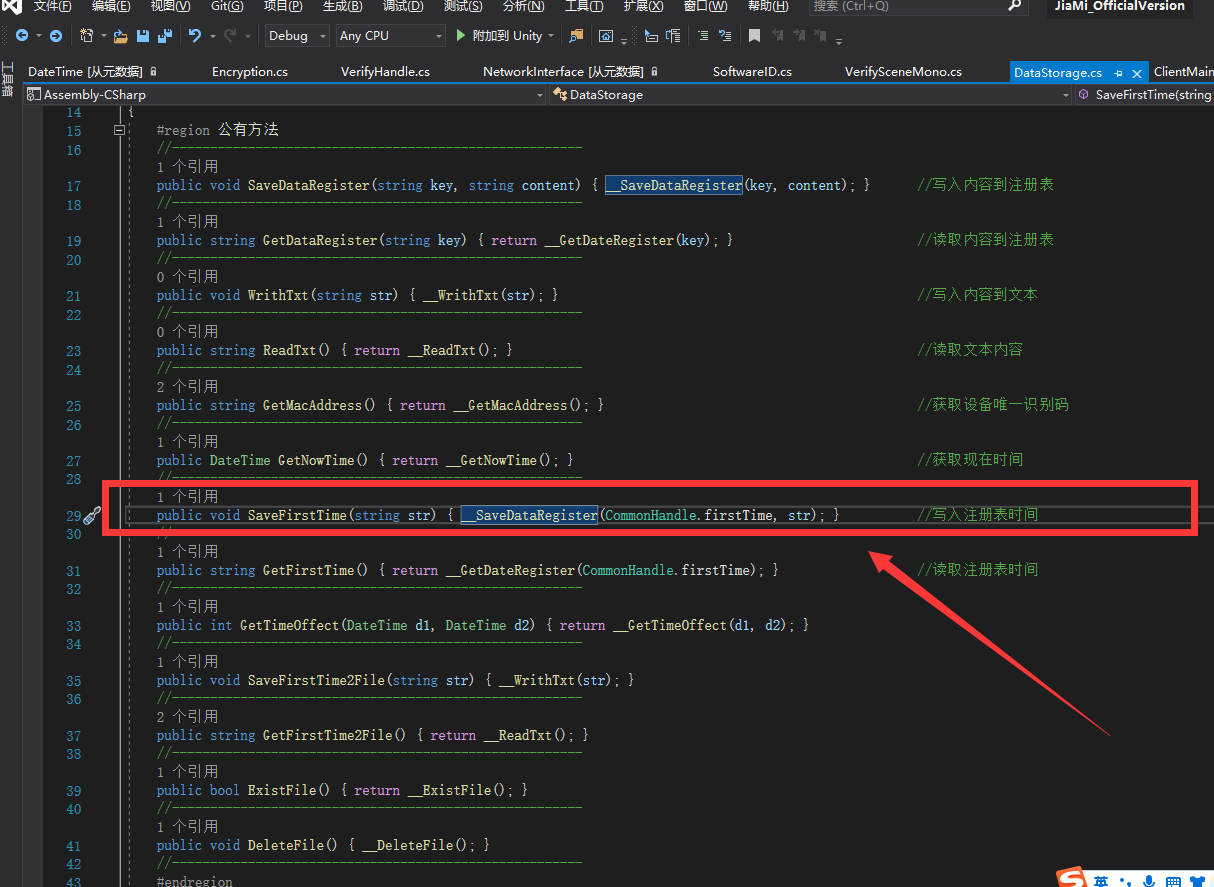
1. 获取当前时间进行加密



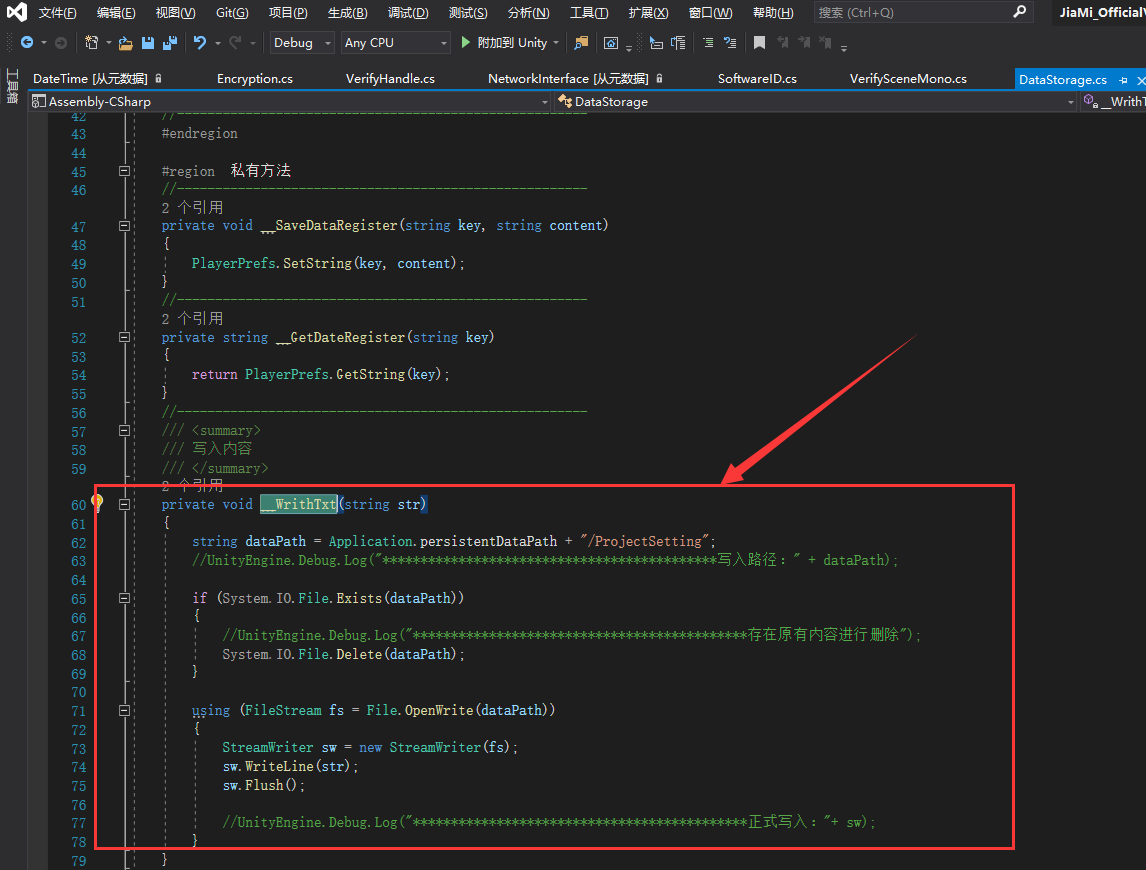
1. 使用之前的加密算法，加密时间



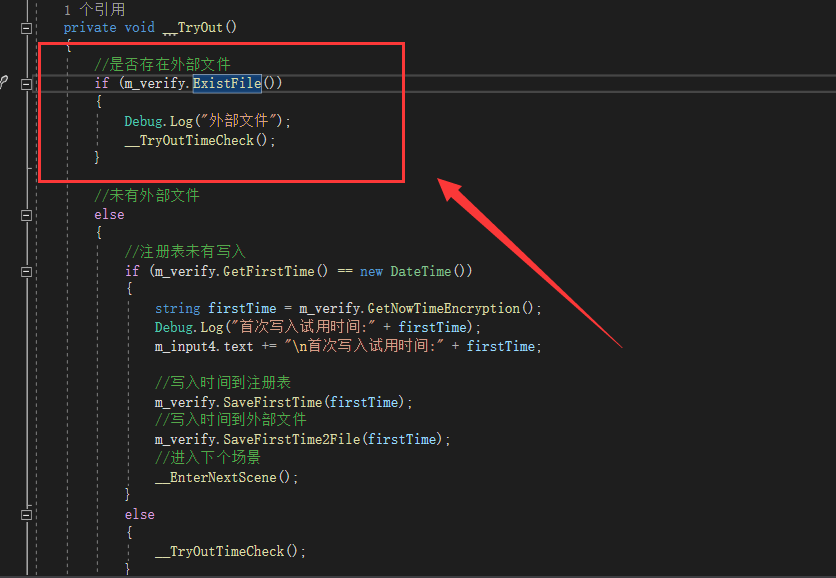
1. 拿到当前的加密时间，将加密时间写入注册表



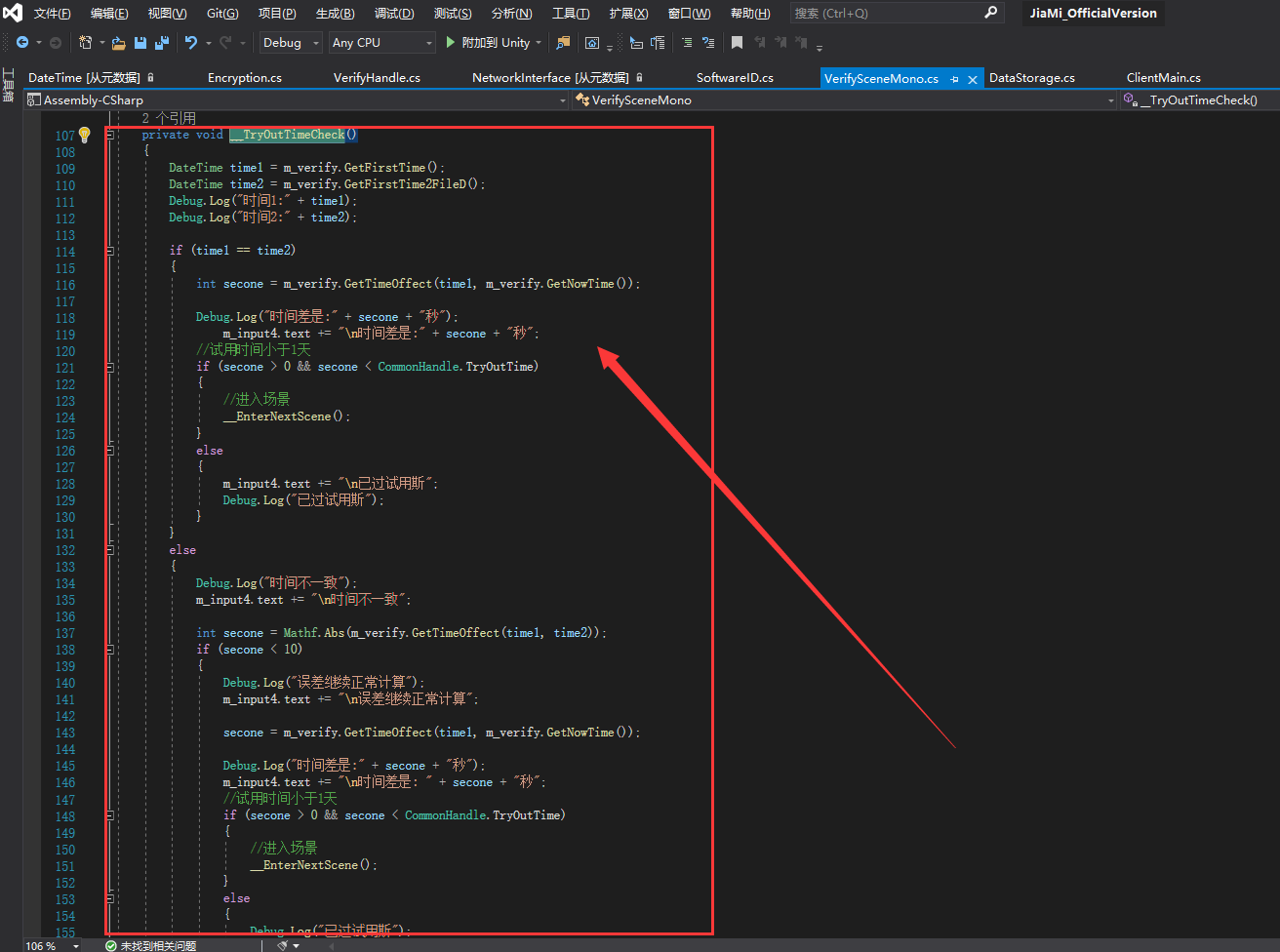
1. 同时将时间写入外部文件：



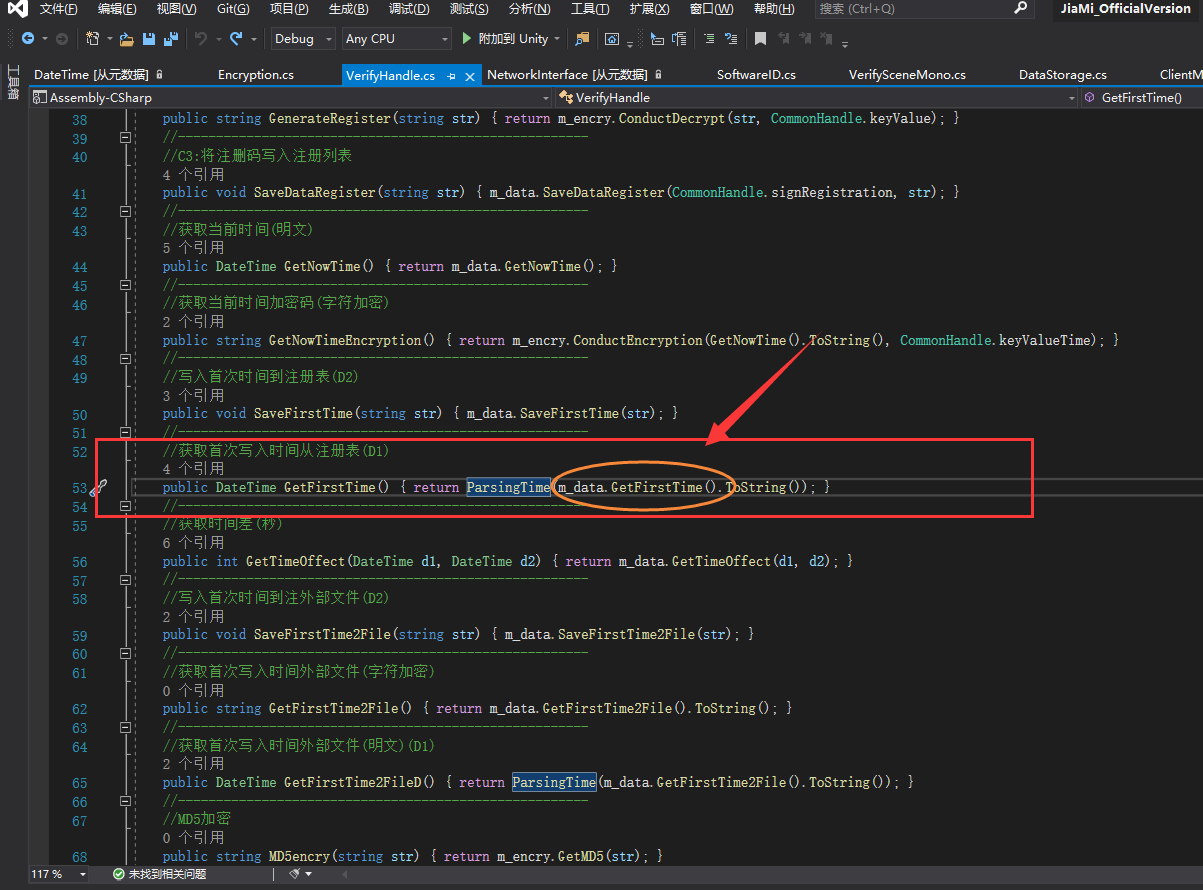
1. 第二次试用，检查外部文件



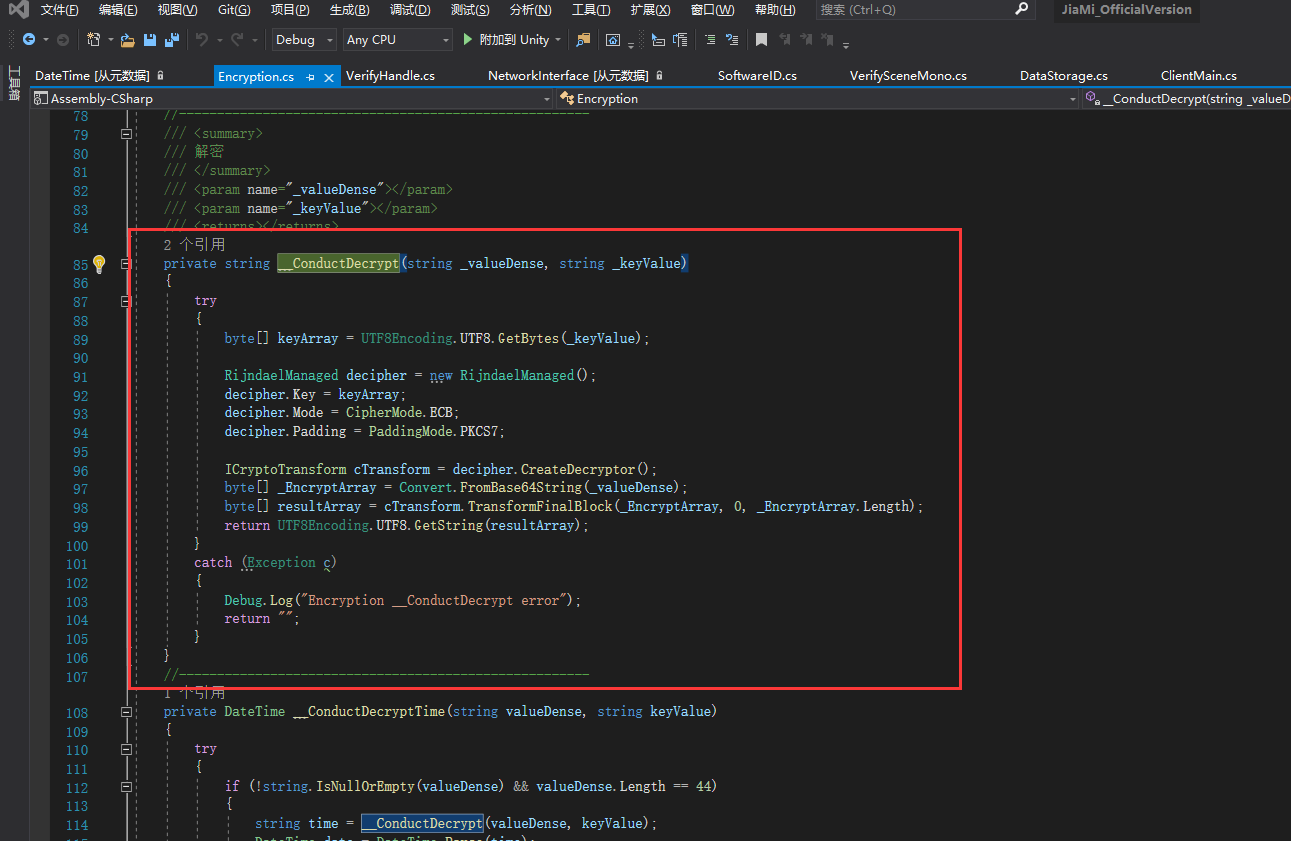
1. 当检查到外部文件时，尝试时间比对



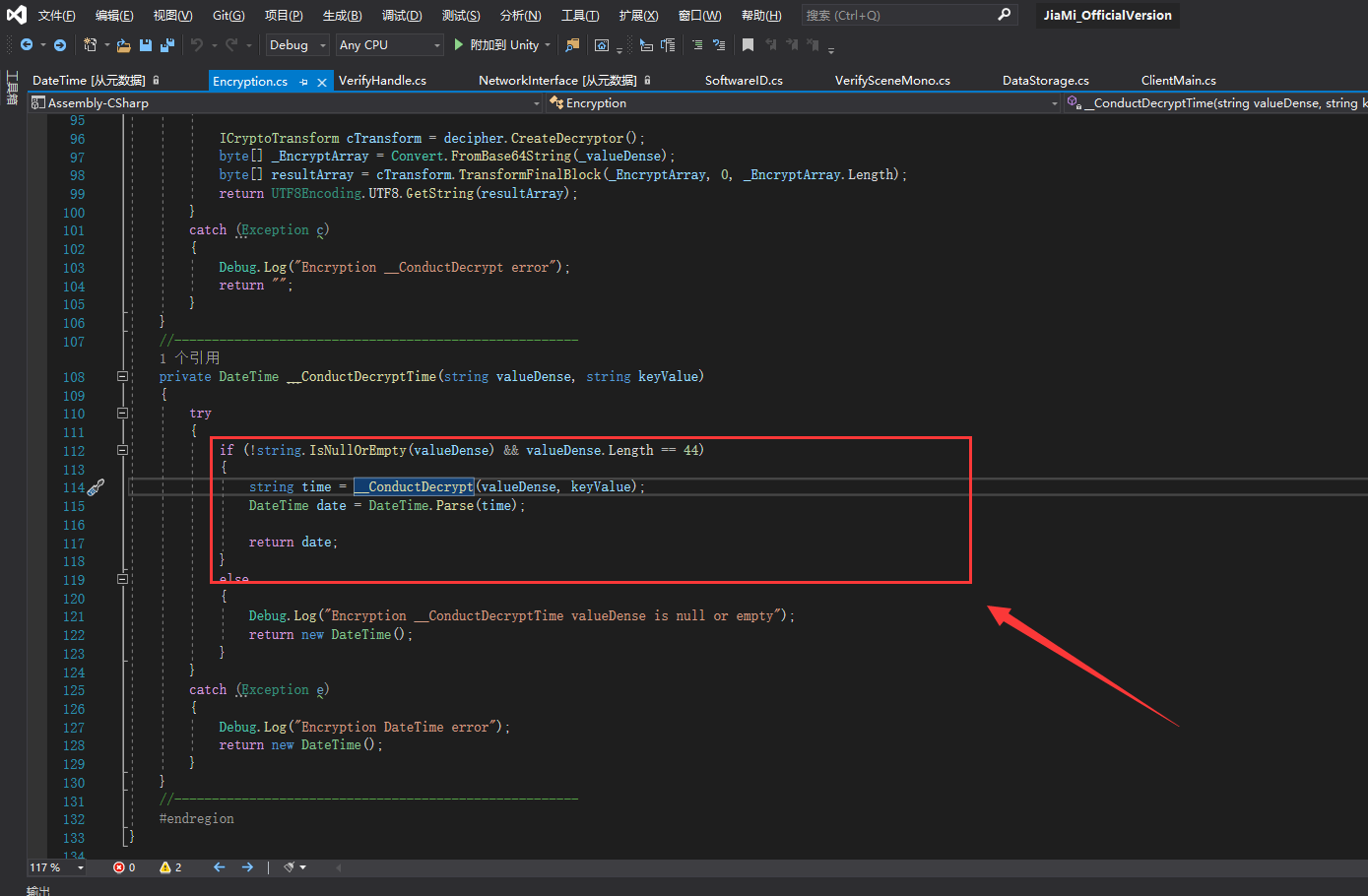
1. 从注册表获取首次写入的加密时间，送入ParsingTime去解密



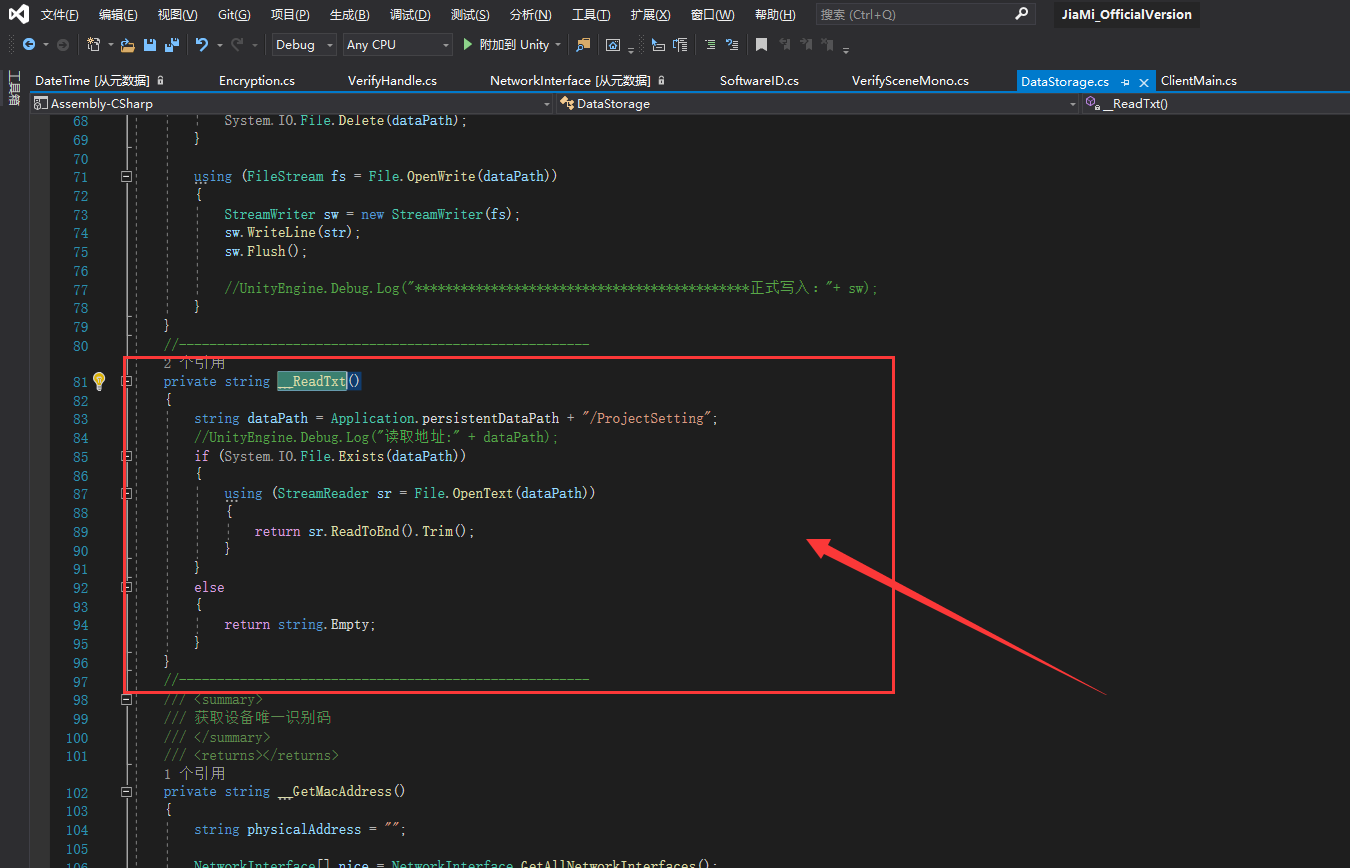
1. 调用字符解密程序对加密时间进行解密



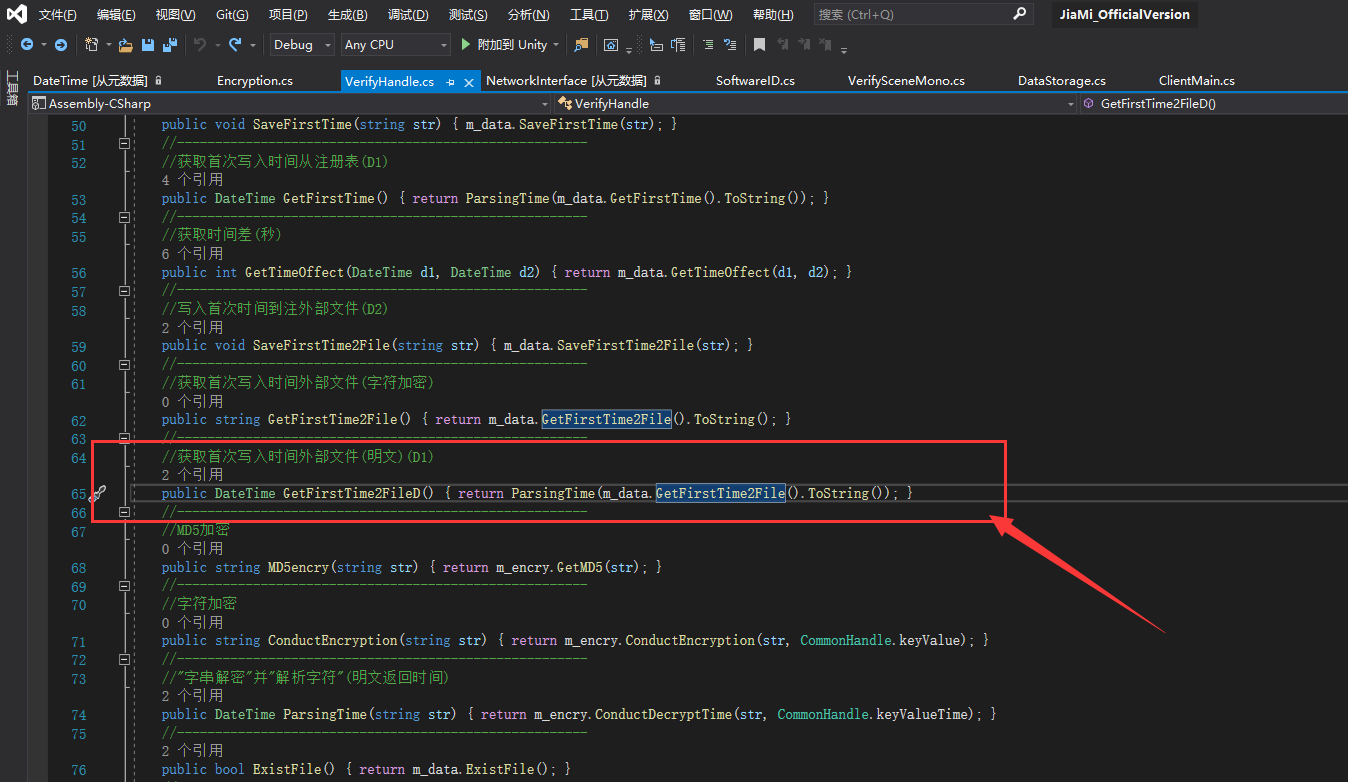
1. 对解密后的时间进行解析，然后返回，此时时间为明文



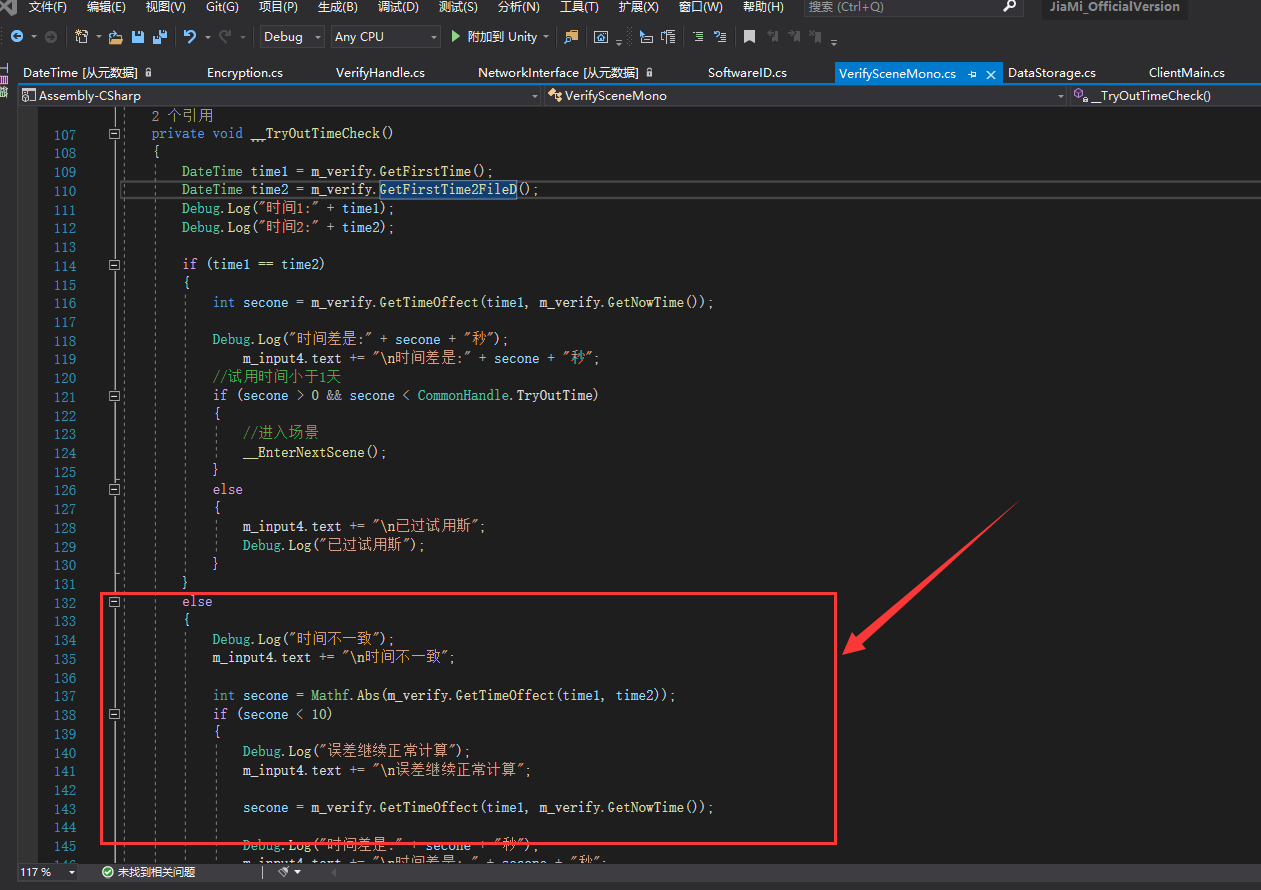
1. 读取外部文件写入时间



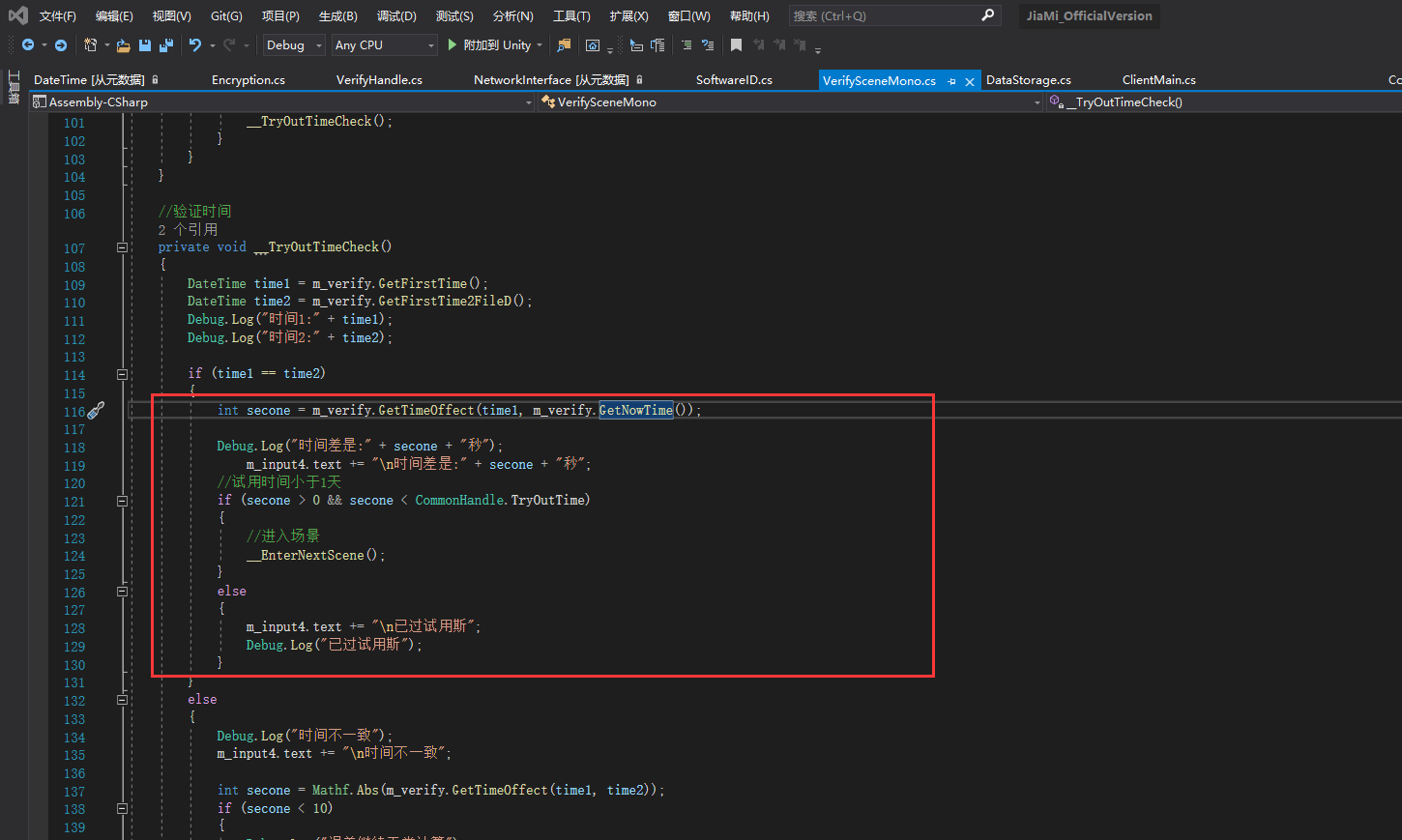
1. 同样调用字符解密程序，对外部文件 进行解密，，拿到解析后的外部文件时间（明文）



1. 当两个时间不一致时，误差小于10秒时可以通过，否则说明有改动，试用不予通过；



1. 两个时间一致时，判断当前试用时间是否过期，获取到的明文时间与当前时间进行比对



### 四：试用部分意外情况

1. 当外部文件或者注册表时间被意外删除时，将当前时间予以返回，否则解析加密时间后返回

