BUUCTF CREAKRTF

原创

一夜通宵程序员 于 2021-05-20 22:29:33 发布 54 收藏

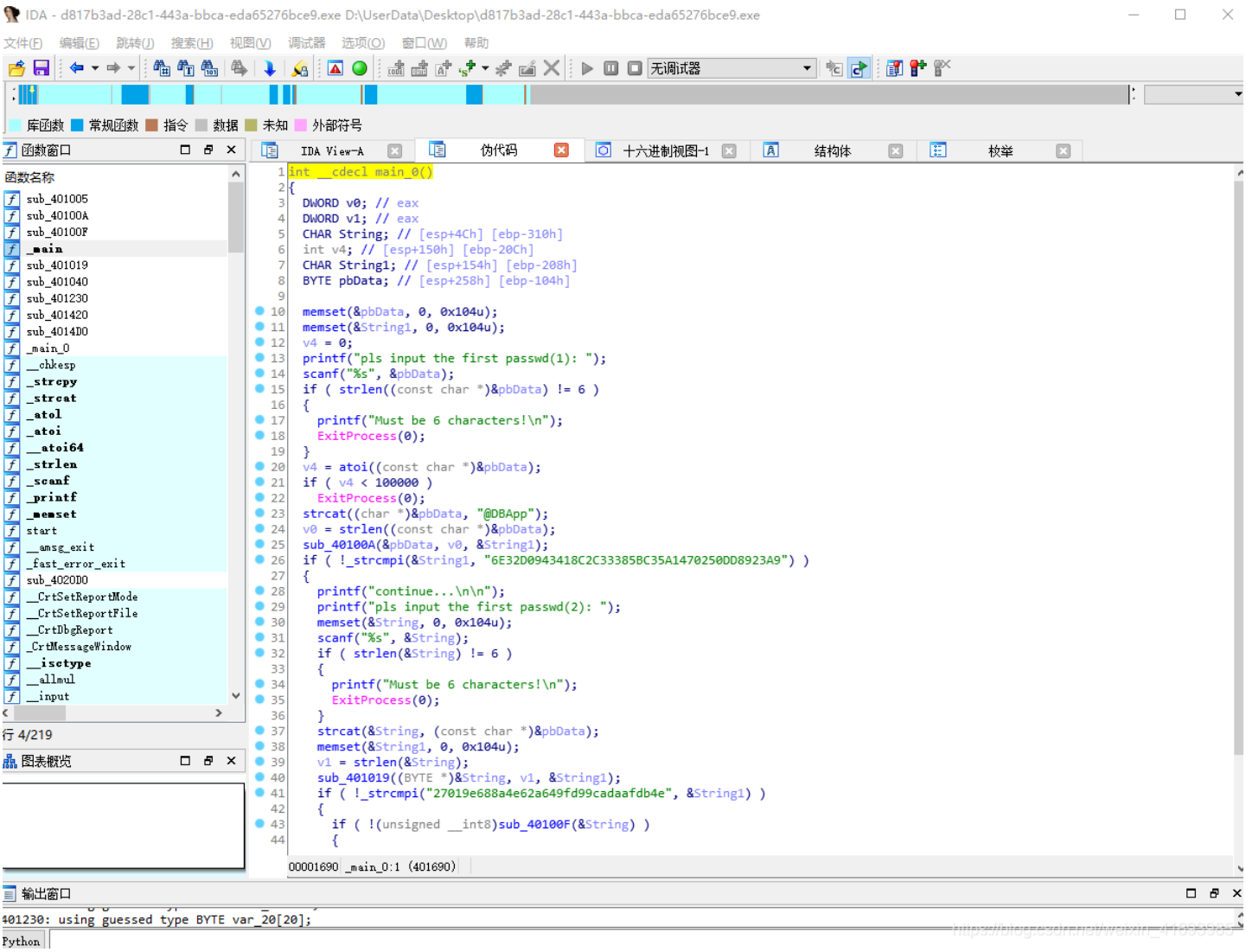
文章标签： 安全

本文链接：https://blog.csdn.net/weixin_41693985/article/details/117092582

BUUCTF CREAKRTF

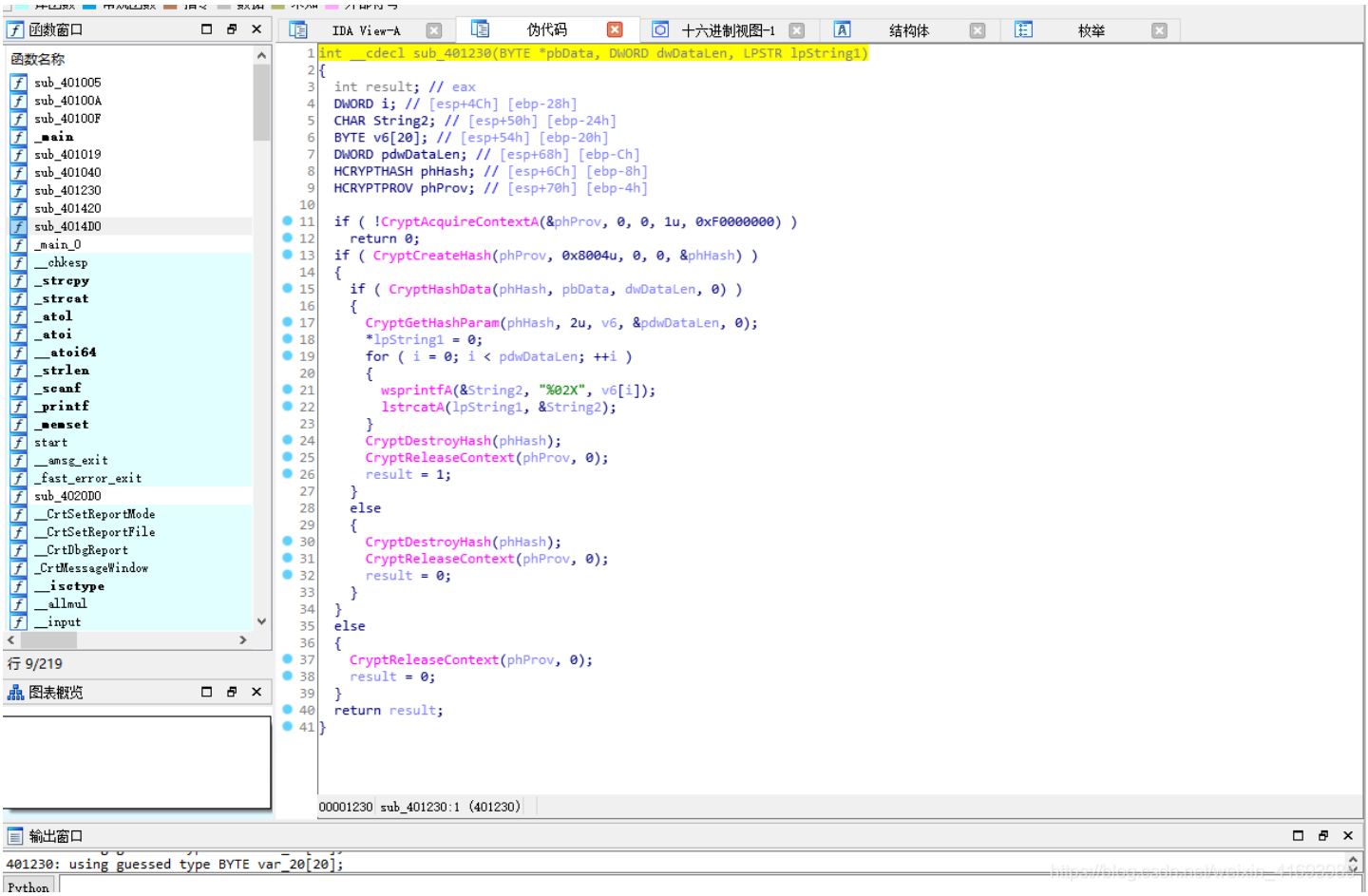
拿到程序，首先查壳，无壳，是32位程序

拖入IDA，进main函数



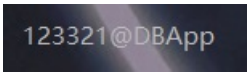
分析代码，首先是要我们输入一串字符且长度为6，再往下分析可知，atoi是把字符串变为整数型，所有可知，我们输入的字符是大于100000小于999999的

ok，继续往下走，可以看到strcat把我们输入的字符串和@DBApp拼接，然后在下一个函数进行加密得到的字符串就是"6E32D0943418C2C33385BC35A1470250DD8923A9"，进入加密函数

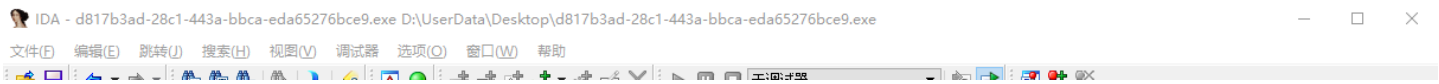


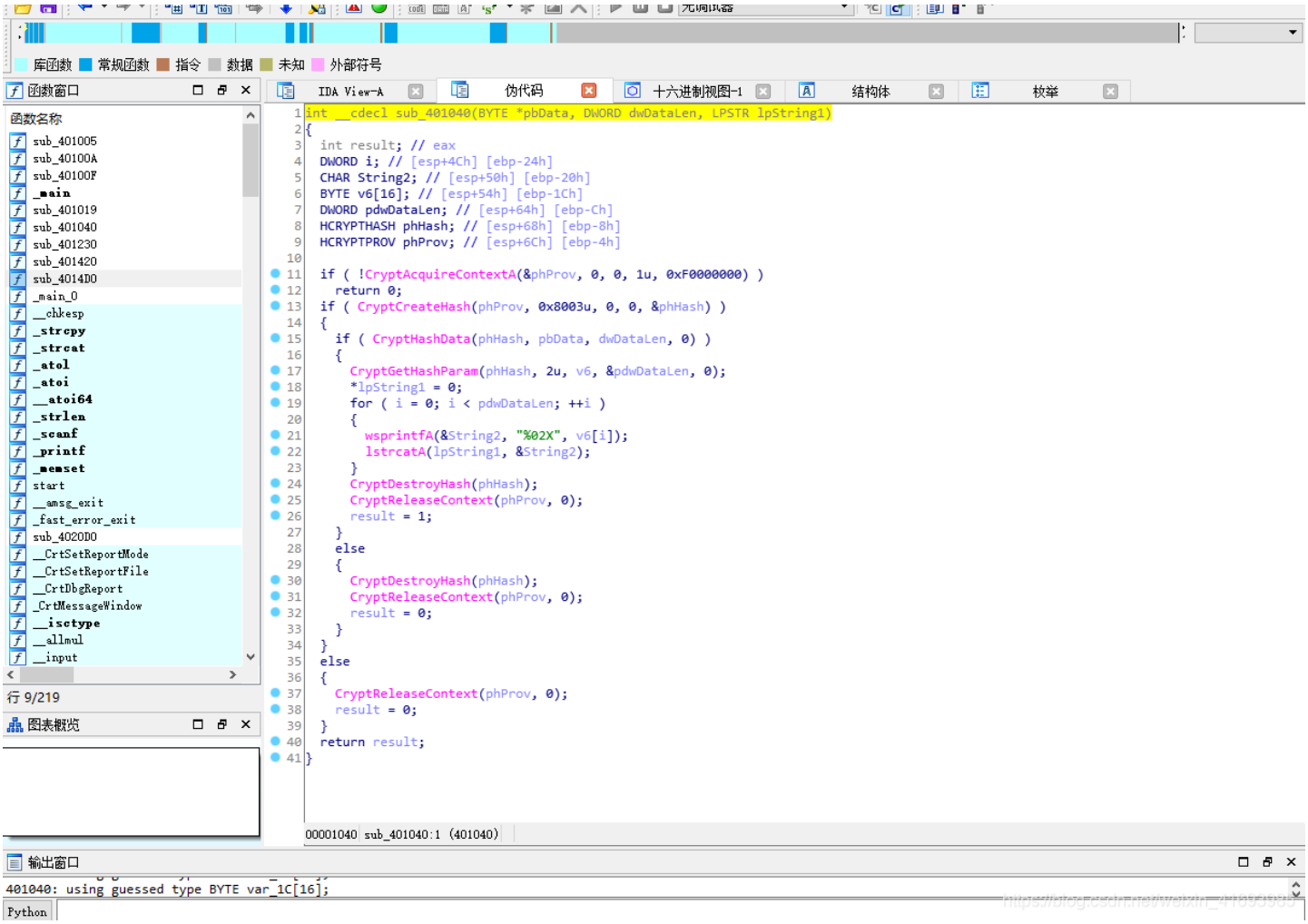
直接百度搜索CryptAcquireContextA了解这是一个加密函数，而8004u是标识符，可知是SHA1，好像是不可逆加密，不确定，所以直接暴力破解，脚本如下：

输出：

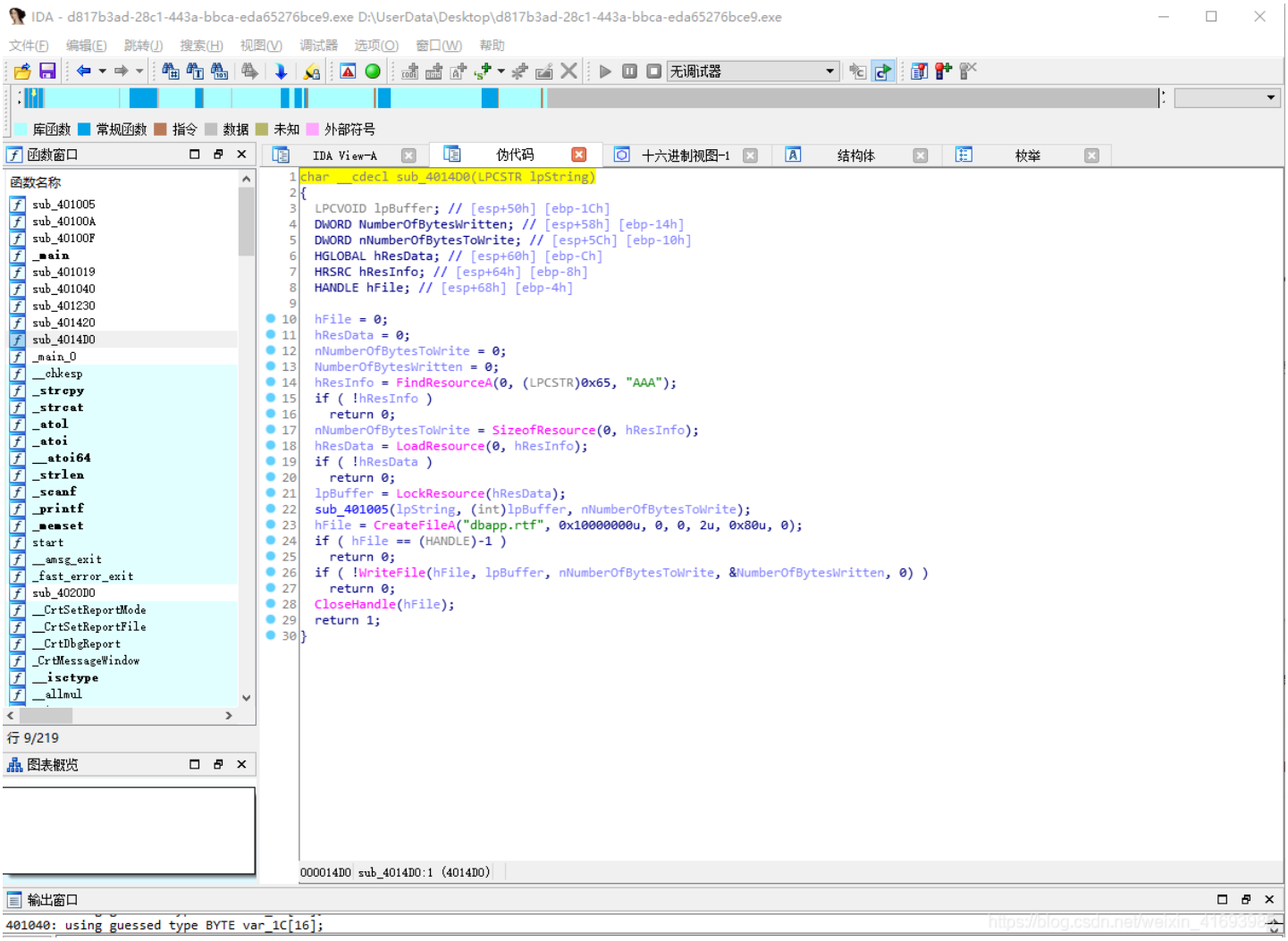


下面的代码和上面的就差不多了，区别在于下面的代码不能进行爆破破解，所以只能进入加密函数里面去看看

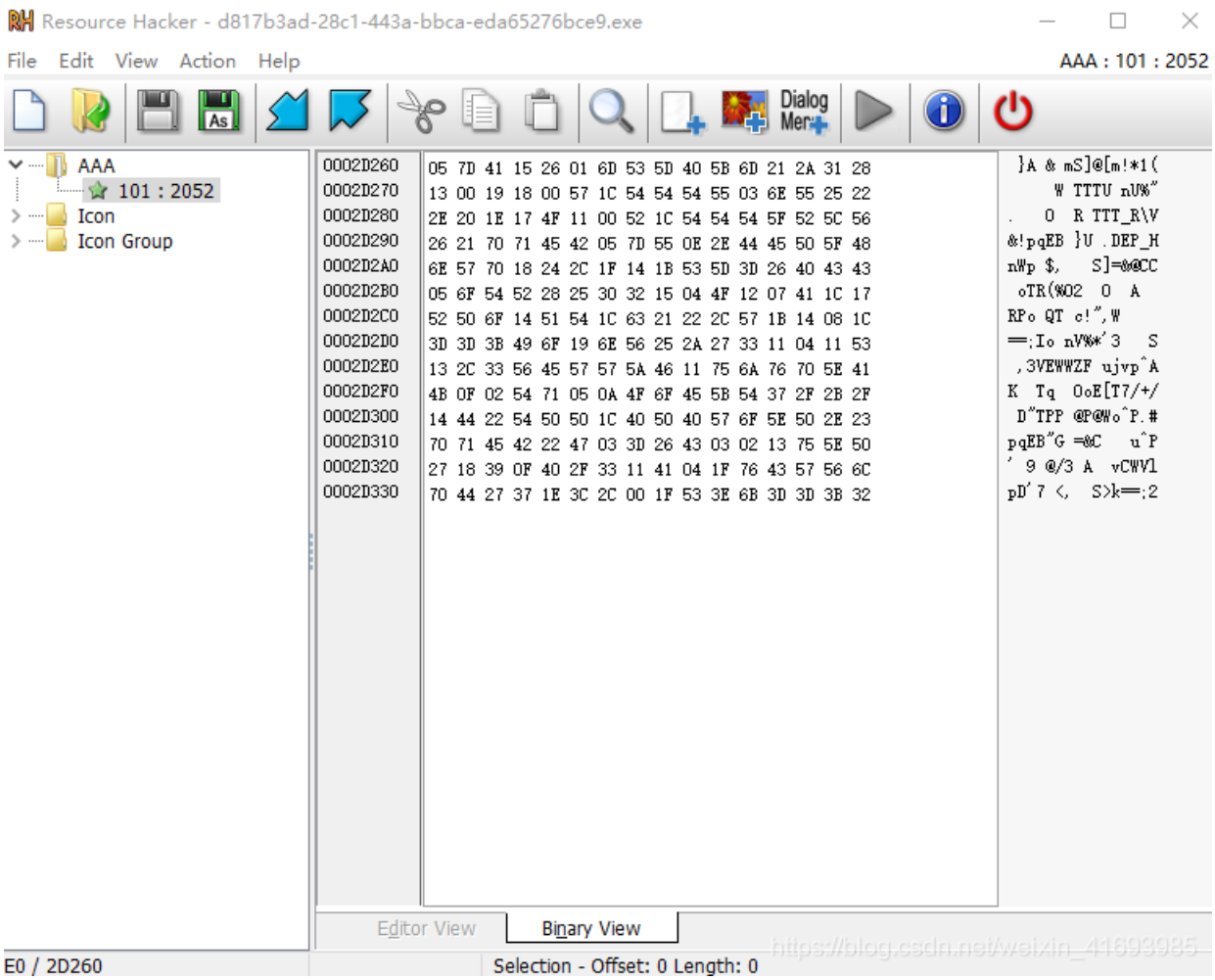




一看标识符，不会，跳过向下分析下一个sub函数



在这里呢就用到一款工具Resource Hacker，打开AAA的文件如下：



然后查看sub_401005函数

```
unsigned int __cdecl sub_401420(LPCSTR lpString, int a2, int a3)
{
  unsigned int result; // eax
  unsigned int i; // [esp+4Ch] [ebp-Ch]
  unsigned int v5; // [esp+54h] [ebp-4h]

  v5 = lstrlenA(lpString);
  for ( i = 0; ; ++i )
  {
    result = i;
    if ( i >= a3 )
      break;
    *(_BYTE *)(i + a2) ^= lpString[i % v5];
  }
  return result;
}
```


到这里差不多就懂了，异或嘛，逆算一下就好了，脚本如下：

```
s = "{\\rtf1"

a = [0x05,0x7D,0x41,0x15,0x26,0x01]

flag = ""
for i in range(0,len(s)):
    x = ord(s[i]) ^ a[i]
    flag += chr(x)
print(flag)
```


结果：

```
~!3a@0
```

将两次得到的数据输入程序就可以得到一个world文档，打开即可获得flag（我的创建的文档不知道跑哪去，我还以为我程序坏了，结果是创建到其他目录去了）