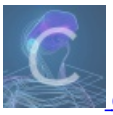
XCTF中IgniteMe

原创

Most3R-03 于 2020-01-13 17:04:23 发布 255 收藏

分类专栏： ctf安全实验室逆向

本文链接：https://blog.csdn.net/recordliu/article/details/103960854



总结知识：

关于异或如下

位运算符家族中，最常用的，某过于异或运算符。

异或运算符是指：

参与运算的两个值，如果两个相应位相同，则结果为0，否则为1。

即：0^ 0=0， 1^ 0=1， 0^ 1=1， 1^1=0

例如：10100001^ 00010001=10110000

0^ 0=0,0^ 1=1 可理解为：0异或任何数，其结果=任何数

1^ 0=1,1^ 1=0 可理解为： 1异或任何数，其结果=任何数取反

任何数异或自己，等于把自己置零

（1)按位异或可以用来使某些特定的位翻转，如对数10100001的第1位和第2位翻转，可以将数与00000110进行按位异或运算。

10100001^ 00000110=10100111 用十六进制表示：0xA1 ^ 0x06= 0xA7

(2)通过按位异或运算，可以实现两个值的交换，而不必使用临时变量。

例如交换两个整数a，b的值，可通过下列语句实现： a=10100001, b=00000110 a=a^ b； //a=10100111 b=b^ a；

//b=10100001 a=a^ b； //a=00000110

(3)异或运算符的特点是：数a两次异或同一个数b（a=a^ b^ b）仍然为原值a.

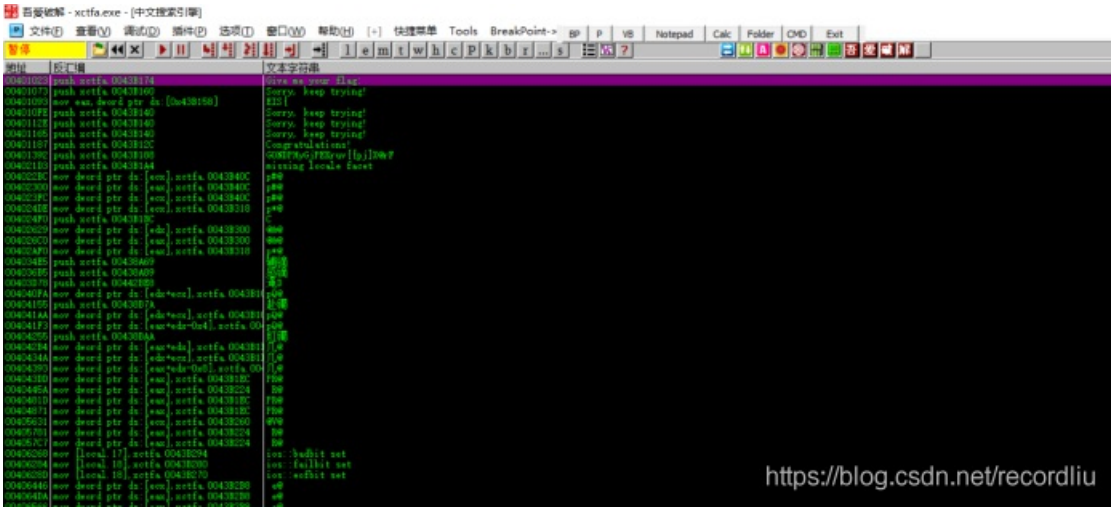
利用异或进行两个值的交换异或有两个很重要的性质：

1、 A^ A = 0;2、A^0 = A;

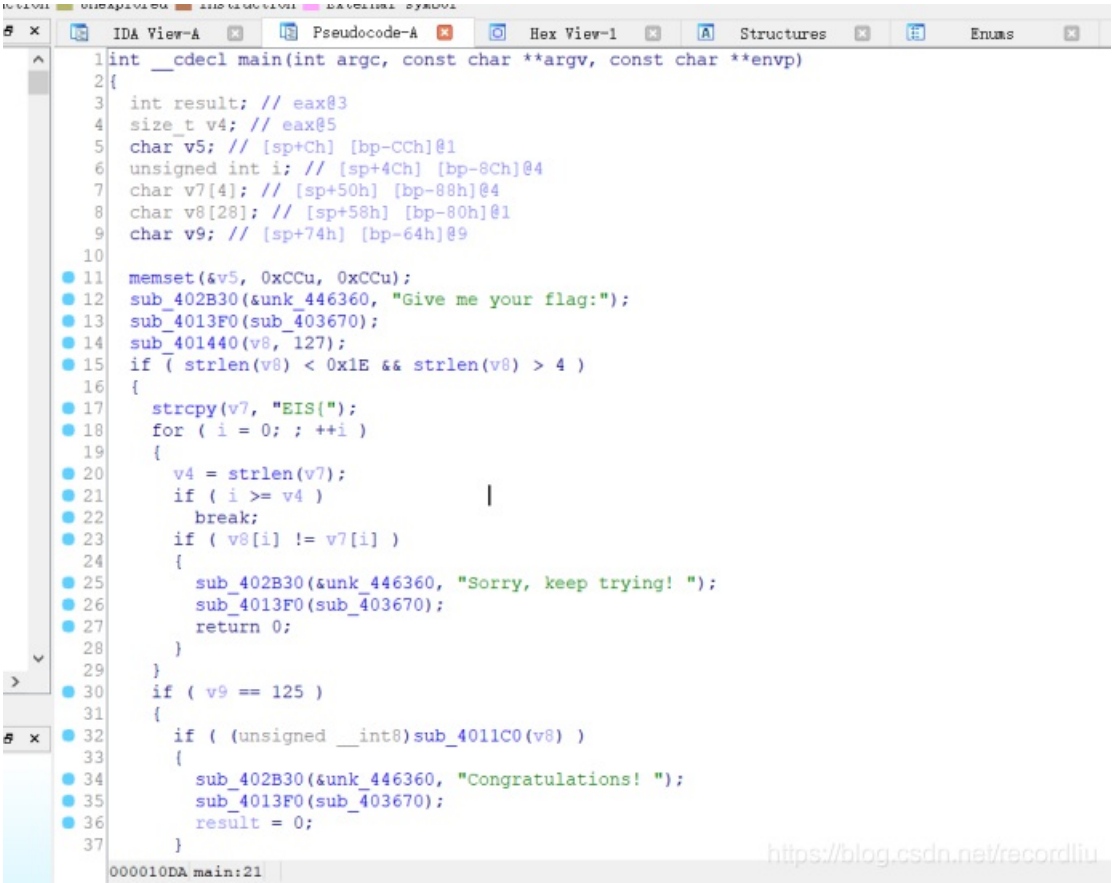
开始逆向

好，进入正题，开始逆向。

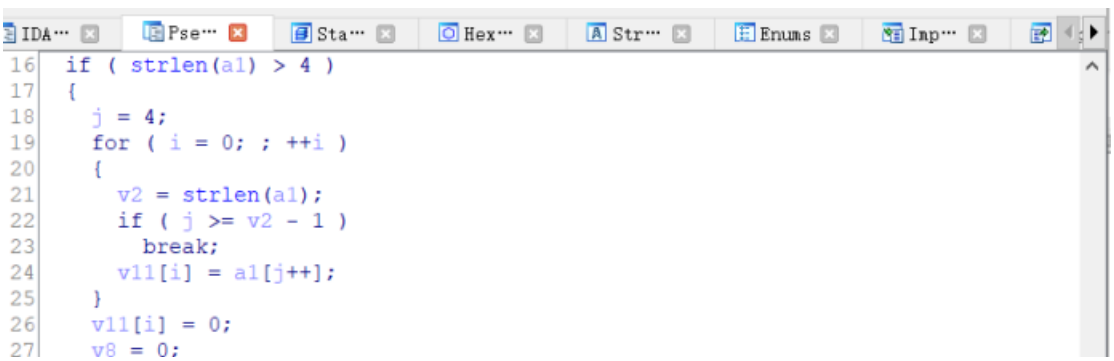
首先，载入OD，查找字符串



没有什么有用的，载入ida



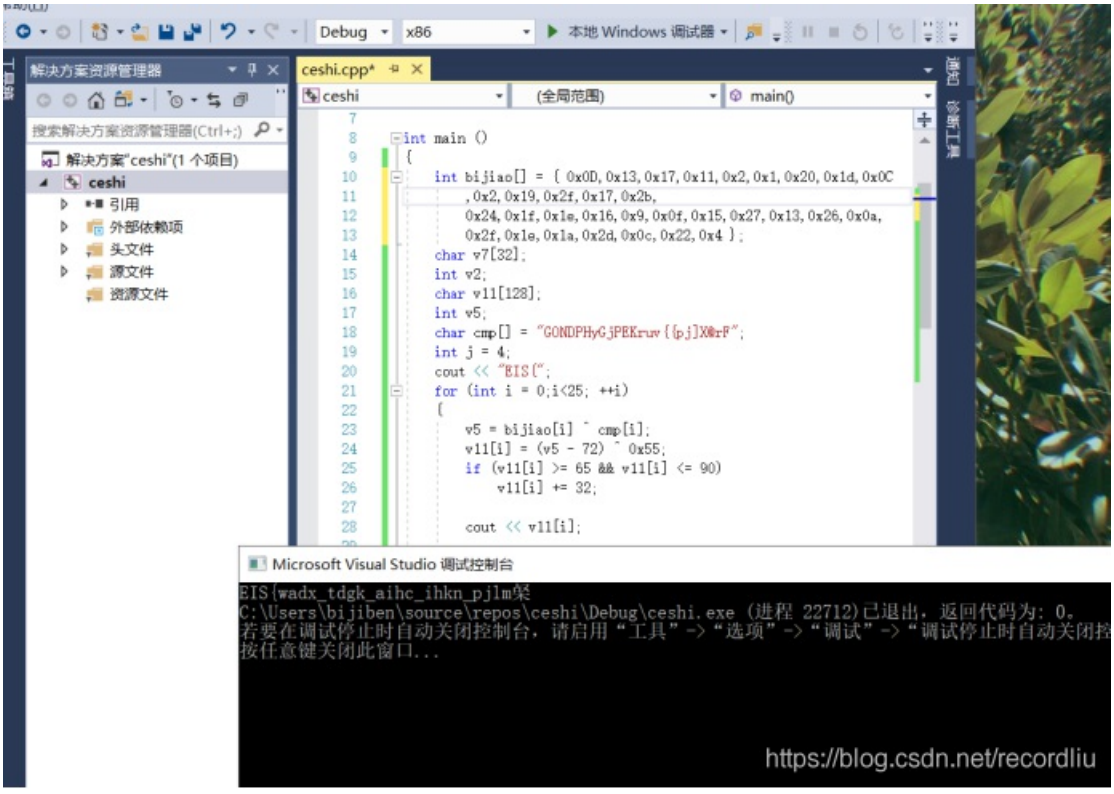
经过分析 sub_4011c0函数可能是关键函数，点开看看！



```
28    v6 = 0;
29    memset(v7, 0, 0x20u);
30    for ( j = 0; ; ++j )
31    {
32      v3 = strlen(v11);
33      if ( j >= v3 )
34        break;
35      if ( v11[j] >= 97 && v11[j] <= 122 )
36      {
37        v11[j] -= 32;
38        v6 = 1;
39      }
40      if ( !v6 && v11[j] >= 65 && v11[j] <= 90 )
41        v11[j] += 32;
42      v5 = sub_4013C0(v11[j]);
43      v7[j] = byte_4420B0[j] ^ v5;
44      v6 = 0;
45    }
46    result = strcmp("GONDPHyGjPEKruv{{pj]X@rF", v7) == 0;
47  }
48  else
49  {
```

000012F1 sub_4011C0:39

进行解题。



EIS{}字符串在main里有加。