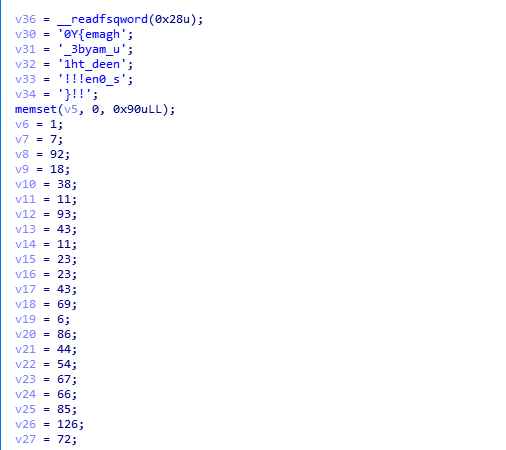
# 题目分析

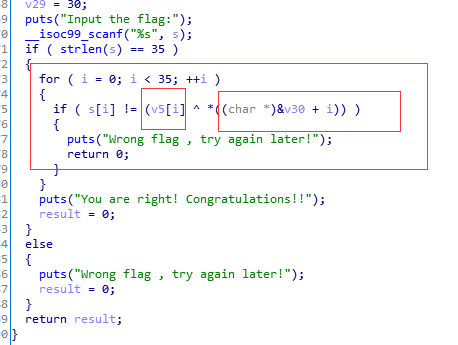
将文件拖进IDA反编译，我们可以发现前面的部分是数据处理部分。



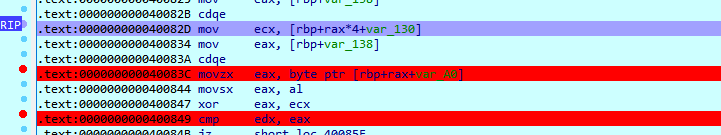
然后我们来看程序的关键算法部分，可以发现程序将输入的flag和一个异或运算得到的数据进行了对比。

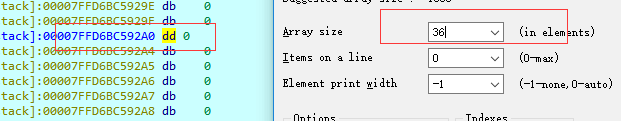
这个异或运算的数据就是前面处理的数据，可以发现数据是什么不太好认出来。

我们使用IDA远程调试ELF文件来看数据是什么。

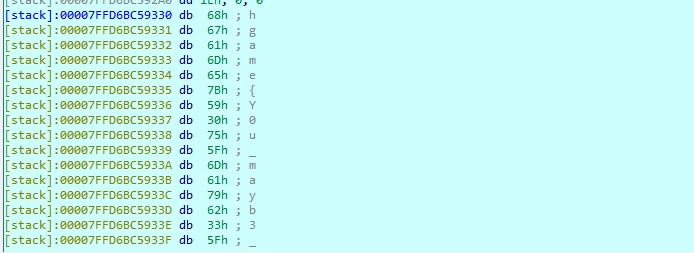


使用IDA远程动态调试ELF文件。





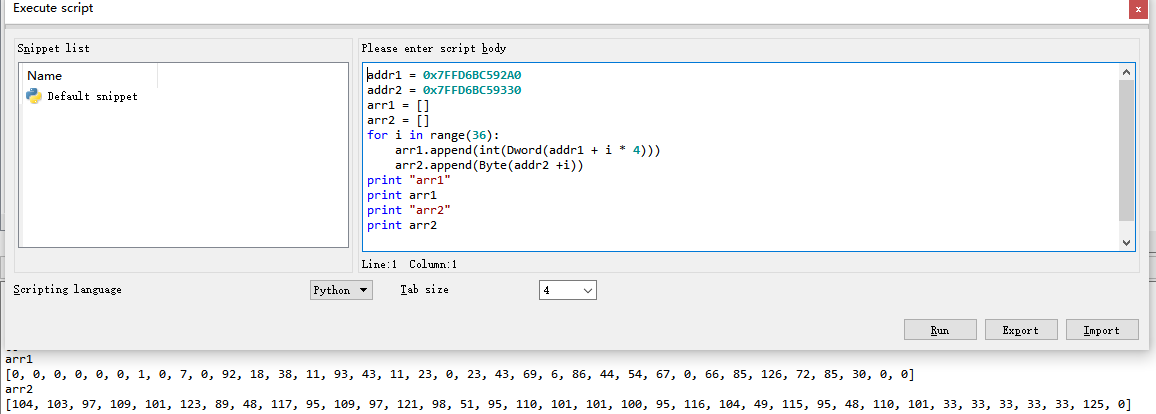
C:\Users\asus\Documents\Tencent Files\2929812422\Image\C2C\Image1\PLL`OL__PNLCEX0_T7X9]PY.png



我们可以很容易发现异或的数据是什么。

然后写IDAPython脚本将数据打印出来。

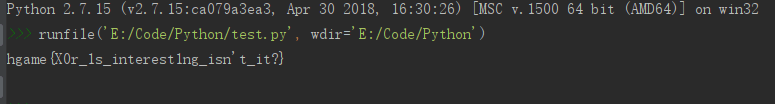
addr1 = 0x7FFD6BC592A0  
addr2 = 0x7FFD6BC59330  
arr1 = []  
arr2 = []  
for i in range(36):  
 arr1.append(int(Dword(addr1 + i \* 4)))  
 arr2.append(Byte(addr2 +i))  
print "arr1"  
print arr1  
print "arr2"  
print arr2



# 脚本

然后我们写出脚本

arr1 = [0, 0, 0, 0, 0, 0, 1, 0, 7, 0, 92, 18, 38, 11, 93, 43, 11, 23, 0, 23, 43, 69, 6, 86, 44, 54, 67, 0, 66, 85, 126, 72, 85, 30, 0, 0]  
arr2 = [104, 103, 97, 109, 101, 123, 89, 48, 117, 95, 109, 97, 121, 98, 51, 95, 110, 101, 101, 100, 95, 116, 104, 49, 115, 95, 48, 110, 101, 33, 33, 33, 33, 33, 125, 0]  
flag = ""  
for i in range(36):  
 flag += chr(arr1[i] ^ arr2[i])  
print flag



# C:\Users\asus\Documents\Tencent Files\2929812422\Image\C2C\Image1\JK~E6G}N)F1Z5]2KV`)`VQJ.png

# FLAG：hgame{X0r\_1s\_interest1ng\_isn't\_it?}