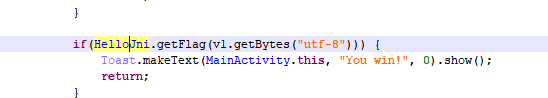
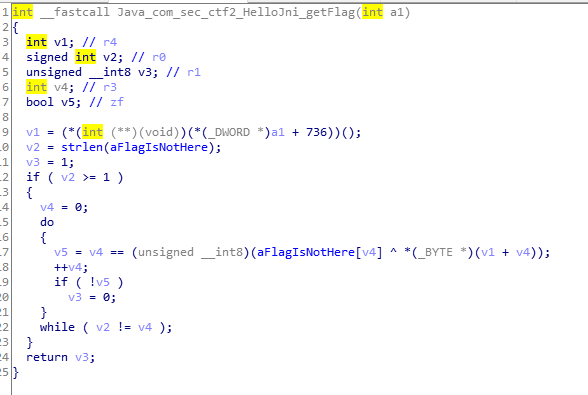
## 题目分析

把apk拖进jeb里面进行反编译，可以找到成功的地方

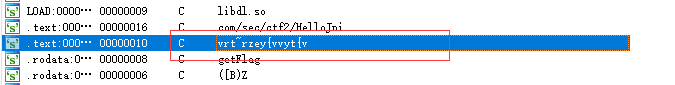


发现是调用了so文件里面的函数，然后我们加压apk，将apk的so文件拖进IDA进行分析，找到调用函数。

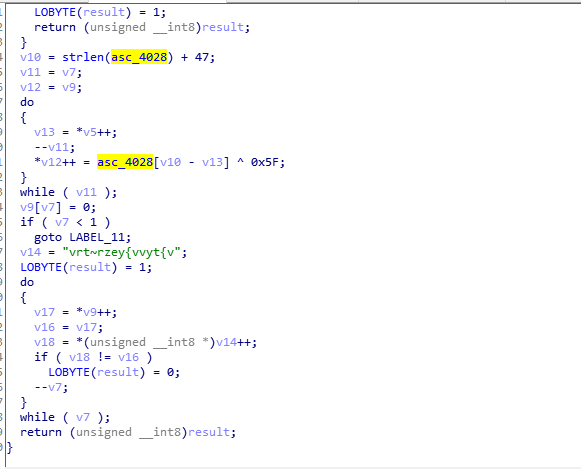


但是我们分析这个函数写脚本时候发现得不到正确的flag，其实这里的字符串提示的也很清楚了，FlagIsNotHere，这是因为真正的函数在启动时候在JNI\_onLoad函数里被替换掉了。

如果我们不想动态调试，在IDA字符串列表中可以找到一个可疑的字符串，



我们找到调用它的地方，即是真正调用的so函数。即sub\_1104。



然后分析函数逻辑写出脚本即可。

## 脚本

str1 = "vrt~rzey{vvyt{v"  
str2 = "!:#"+chr(0x24)+"%&()+-"  
flag = ""  
num = len(str2) + 47  
for i in range(len(str1)):  
 flag += chr(num - str2.find(chr(ord(str1[i]) ^ 0x5f)))  
print flag