# 0X00 字符串加密

加密.data端中的数据

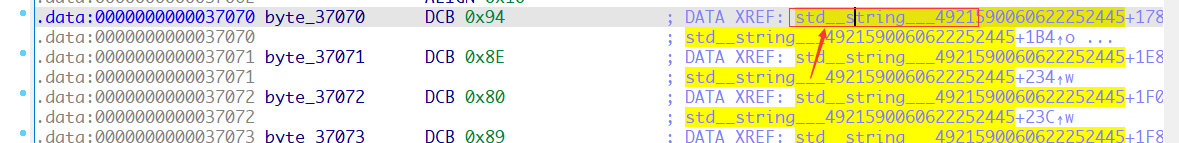
## 0x0000 识别

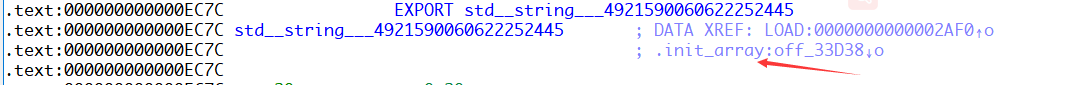
主要是看.data段有没有被.init\_array这种先执行的段应用。

1. 明显的datadiv\_decode前缀，直接上的原版ollvm



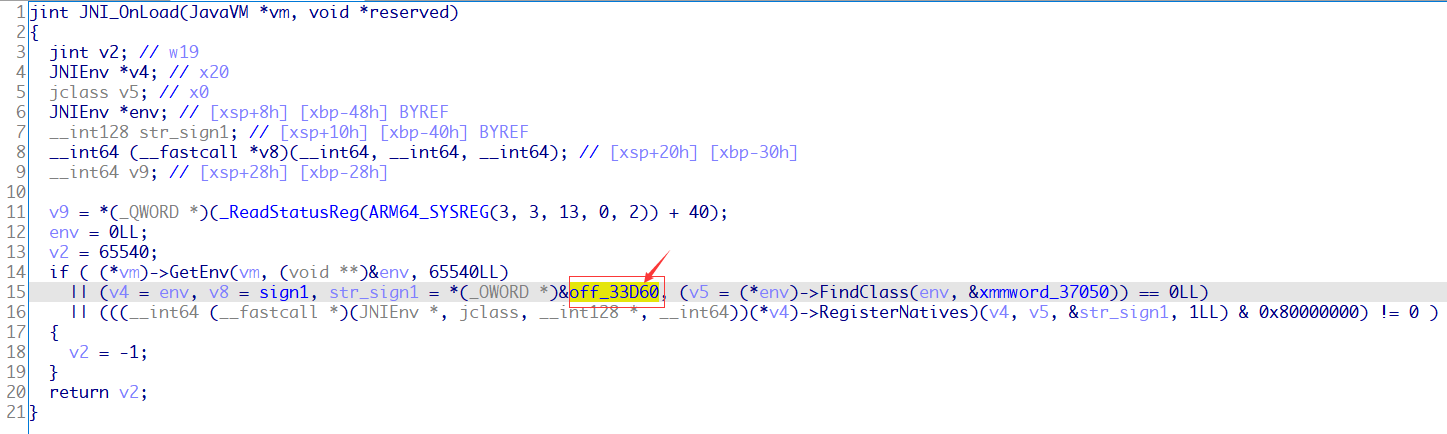
1. 自定义的解密函数名，方法其实是一样的看有没被.init\_array这种段引用





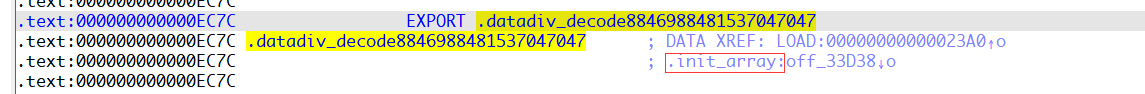
## 0x0001 JNI\_Onload

加密JNI\_Onload中的字符串



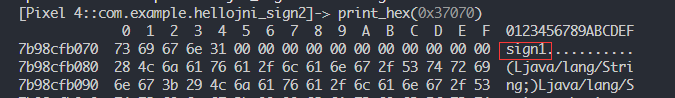
进入相应的data段，发现被.init\_array段中的函数调用





应对方法：

直接加载后用frida去hexdump相应的地址



或者像要得到jni函数的函数名，签名和相对地址可以直接hook RegisterNatives函数。

## 0x0002 data段

其实准确的来说是加密data端的数据，jni\_onload用到的.data只是一起被加密了而已

# 0x01 Control Flow Flattening

主要是将if else块转换成switch case块