使用IDA Pro静态分析Android程序

# 基本操作

找到Apk解压出的classes.dex文件，然后打开IDA Pro，将classes.dex拖放到IDA Pro的主窗口，会弹出如下对话框，IDA Pro解析出的文件属性是“Android DEX File”，保持默认属性点击OK按钮。

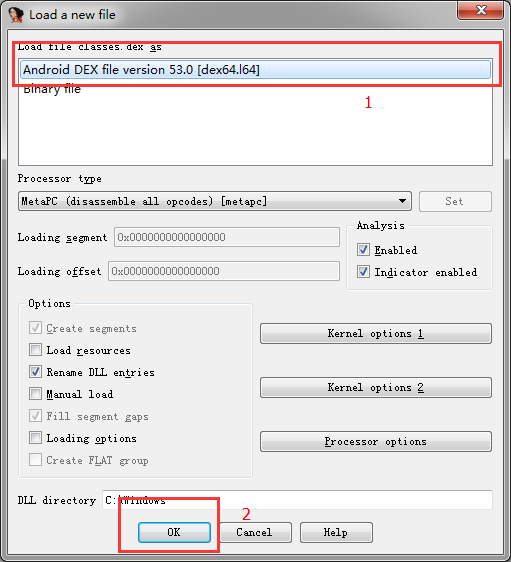


图 1

在工具栏下面便是工作窗口。主要的窗口分页有”IDA View-A”、”Hew View-A”、”Structures”、”Enums”、”Imports”和”Exports”。



## IDA View-A

分析窗口，支持两种显示模式，除了常见的反汇编模式之后，还提供图形视图以及其他有趣的功能。

快捷键G：通过快捷键G弹出的地址跳转对话框，可以跳转到dex文件相应的位置上。

点击菜单项“Jump→Jump to segment”，或者按下快捷键CTRL+S，弹出段选择对话框，如图2所示。IDA Pro将dex文件一共分成如下几个段。

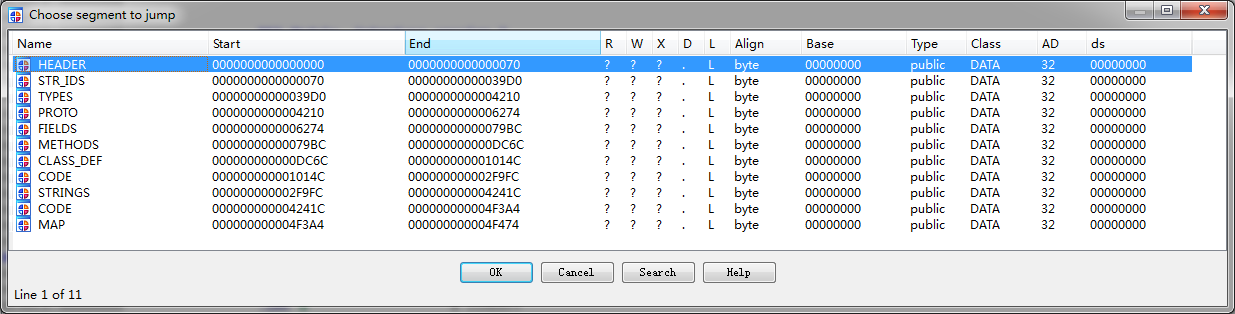


图 2

## Exports选项卡

dex文件中所有方法可以点击Exports选项卡查看。方法的命名规则为“类名.方法名@方法声明”。如图3所示，在Exports选项卡中随便选择一项，例如选择名为SimpleCursorA [dapter.swapCursor@LL](mailto:dapter.swapCursor@LL)的方法，然后双击跳转到相应的反汇编代码处。

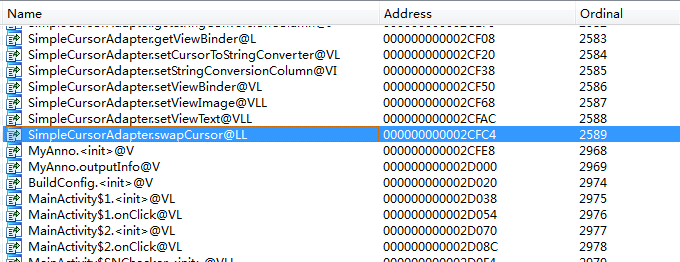
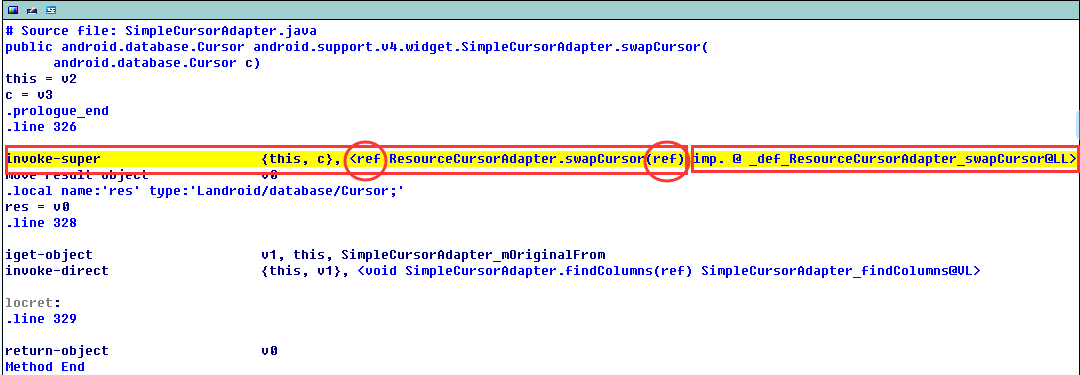


图 3

为方便可按空格键切换到视图的模式来查看，结果如图4所示



方法第1行的invoke-super指令的前半部分如下：

invoke-super {this, p0}, <ref ResourceCursorAdapter.swapCursor(ref)

IDA Pro的反汇编代码使用ref关键字来表示非Java标准类型的引用，对上面一行代码的分析，前面ref是swapCoursor()方法的返回类型，后面括号中的ref是参数类型。

方法第1行的invoke-super指令的后半部分如下：

imp. @ \_def\_ResourceCursorAdapter\_swapCursor@LL

这一部分是IDA Pro智能识别的。IDA Pro能智能识别Android SDK的API函数，并使用imp关键字标识出来，@后面的部分为API的声明，类名与方法名之间使用下划线分隔。

# 定位关键代码

## 定位字符串

如上面所述打开段选择框，双击STREING段类型，跳转到String类型数据段中。然后点击菜单项 “Search→text”，或者按下快捷键ALT+T，打开文本搜索对话框，在String旁边的文本框中输入要搜索的字符串后点击OK按钮，稍等片刻 就会定位到搜索结果。

## 定位关键API

打开段选择框，双击第一个CODE段跳转到数据起始段，然后点击菜单项 “Search→text”，或者按下快捷键ALT+T，打开文本搜索对话框，在String旁边的文本框中输入要搜索的API名称后点击OK按钮，稍等 片刻就会定位到搜索结果。如果API被调用多次，可以按下快捷键CTRL+T来搜索下一项。

## 定位方法名

如上面所说，所有的方法名均在Exports选项卡中。单击Exports选项卡，以程序的主界面MainActivity为例。在Exports选项卡页面上输入Main，代码定位到以Main开头的所有行，如图4所示。双击可以看到该函数反编译后具体的内容。

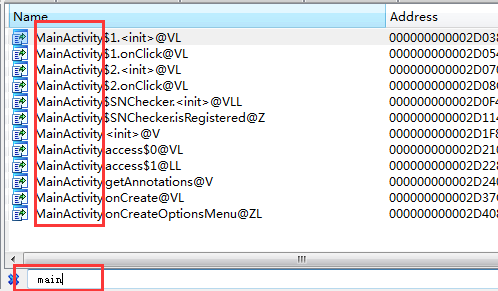


图 4

## 对MainActivity的简单分析即破解

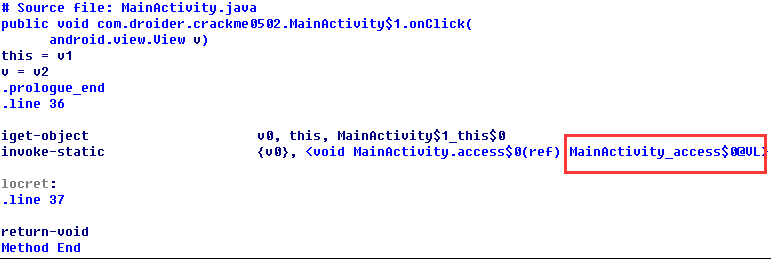


图 5 图 6

图5为该APK的主界面，可以发现有2个按钮。这对应了图4中的两个OnClick()的方法。查看OnClick1()的代码可以看到如图6中一行代码，调用了MainActivity.access$0()方法。

在Exports中找到该方法，双击查看代码，得知该方法调用了getAnnotations()方法，故OnClick1()为“获取注解”按钮，同理可知OnClick2()为“检测注释码”按钮。查看OnClick2()代码，切换至视图模式如图7所示，代码的“分水岭”就是“if-eqz v2, loc\_2D0DC”，左边红色箭头表示条件不满足时执行的路线，右边的绿色箭头是条件满足时执行的路线。

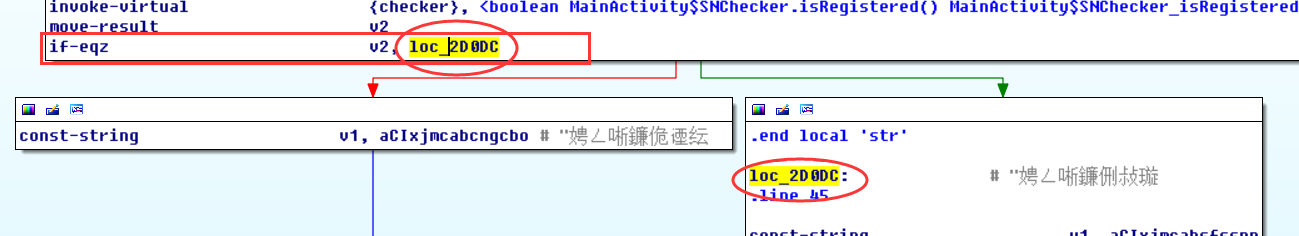


图 7

由上图可知直接修改if-eqz即可将程序破解。将鼠标放置if-eqz所在行，点击IDA Pro主界面的“Hex View-A”选项卡，如图8所示，可看到这条指令所在的文件偏移为0x2D0BE，相应的字节码为“38 02 0f 00”，修改字节码，重新打包即可完成破解。



图 8