APK逆向

**APK组成：**

asset文件夹:

<资源目录1：>

lib文件夹：

<so库存放位置，一般由NDK编译得到，常见于使用游戏引擎或JNI native调用的工程中>

ses文件夹:

<资源目录2：>

AndroidMainfest.xml:

<Android工程的基础配置属性文件>

classdex：

<Java代码编译得到的Dalvik VM能直接执行的文件>

Resources.arsc:

<对res目录下的资源的一个索引文件，保存了原工程中strings.xml等文件内容>

asset资源目录和res资源目录的不同处：

res目录下的资源文件在编译时会自动生成索引文件（R.java），在Java代码中用R.xxx.yyy来引用；而asset目录下的资源文件不需要生成索引，在Java代码中需要AssetManager来访问；

一般来说，除了音频和视频资源（需要放在raw或asset下），使用Java开发的Android工程使用到的资源文件都会放在res下；使用C++游戏引擎（或使用Lua Unity3D等）的资源文件均需要放在asset下。

**Dalvik字节码：**

Dalvik是google专门为Android操作系统设计的一个虚拟机，经过深度的优化。虽然在Android上的程序是使用java来开发的，但是Dalvik和标准的java虚拟机JVM是两回事。Dalvik VM 是基于寄存器的，而JVM是基于栈的；Dalvik有专属的文件执行格式dex，而JVM执行的是java字节码。Dalvik VM比JVM速度更快，占用空间更少。

通过Dalvik的字节码我们不能直接看到原来的逻辑代码，这时需要借助Apktool或dex2jar+jd-gui工具来帮助查看。但是，注意的事最终我们修改APK需要操作的文件是.smali文件，而不是导出来的Java文件重新编译。

**Smali简介（破解的重中之重）：**

Smali,baksmali分别是指安卓系统里的Java虚拟机（Dalvik）所使用的一种dex格式文件的汇编器，反汇编器。其语法是一种宽松式的Jasmin/dedexer语法，而且它实现了.dex格式所有功能（注解，调试信息，线路信息等）。

当我们对APK文件进行反编译后，遍会生成此类的文件。

**Android逆向重要工具：**

**apktool**

**dex2jar**

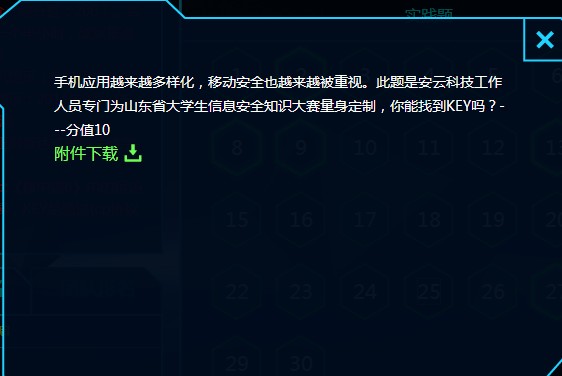
jd-gui

IDA

**apktool    作用：资源文件的获取，可以提取出图片文件和布局文件进行查看**

**dex2jar    作用：将apk反编译成**[**Java**](http://lib.csdn.net/base/java)**源码（classes.dex转化成jar文件）**

**jd-gui       作用：查看APK中的classes.dex转化成的jar文件，即源码文件**

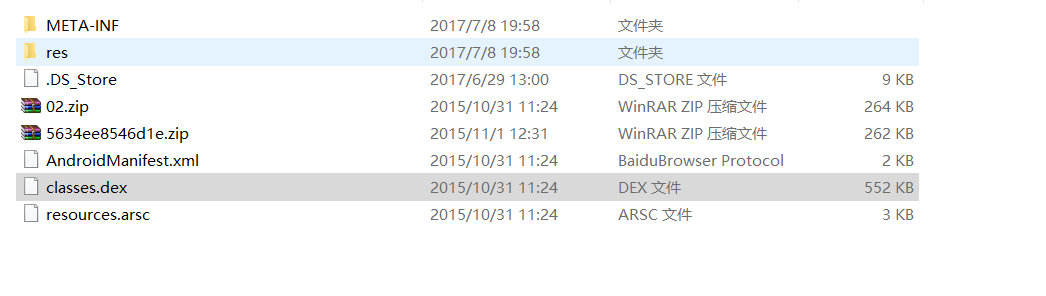
具体题目：

文件在附件中。

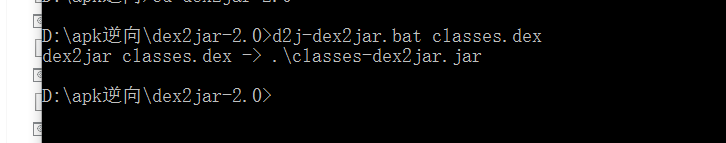
Writeup：

（需要提前安装好dex2jar,jd-gui,安装需要有Java环境）

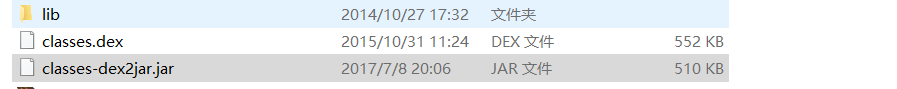
解压文件得到pak文件，将apk文件格式改为zip，解压得到



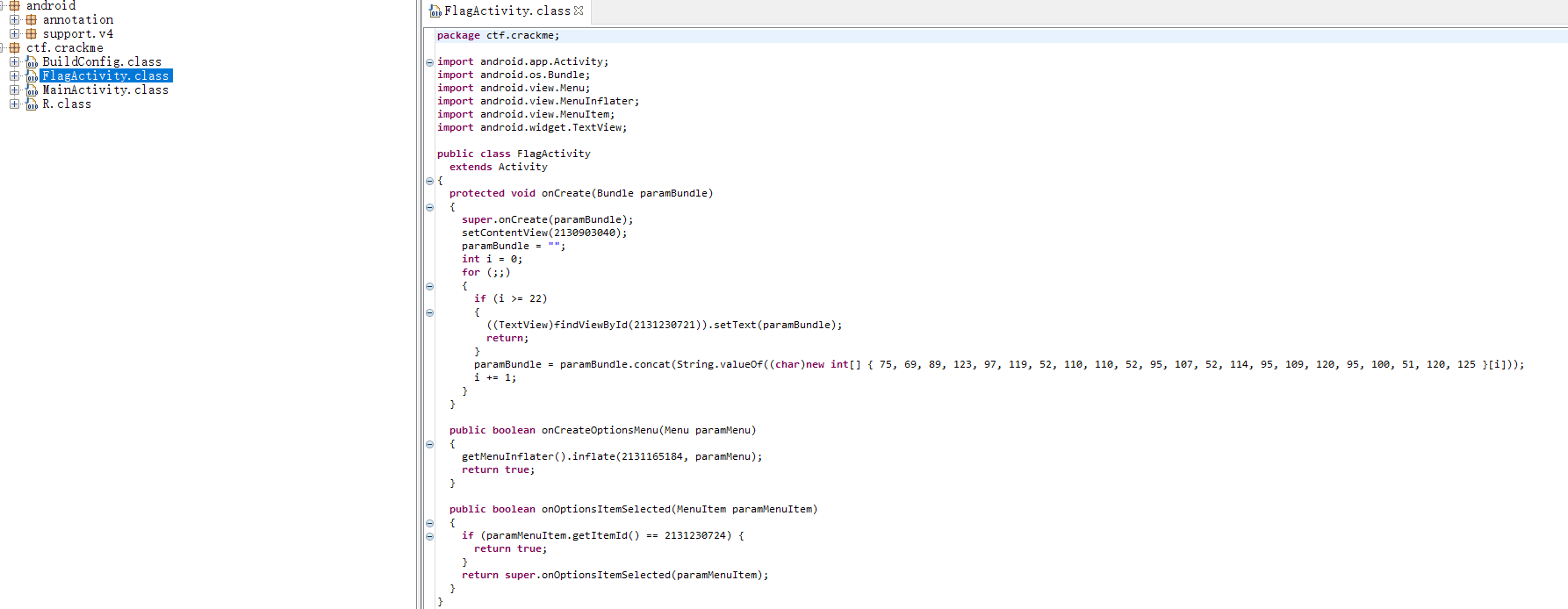
然后将解压得到的class.dex文件拷贝到与dex2jar文件目录下，之后操作如图



可以得到一个.jar文件



然后将.jar文件拖到jd-gui中



可以看到啊有一个FlagActivity.class类，阅读该代码可以发现key是由一串ascii码转为字符串后连接到一起得到key.

答案：KEY{am4nn4\_k4r\_mx\_d3x}