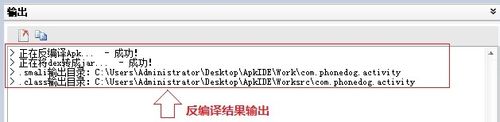
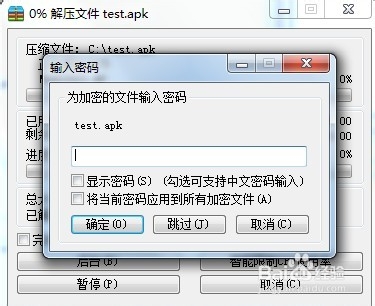
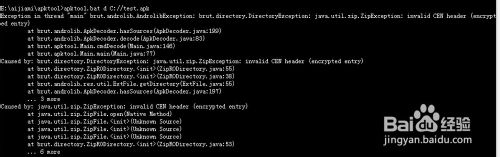
Android代码混淆防反编译解决方案研究

做Android开发的都知道要做混淆去防apk被反编译、破解，通过proguard进行Java代码混淆。但是，Android代码混淆真的能起到实质性的作用吗？看下面分析

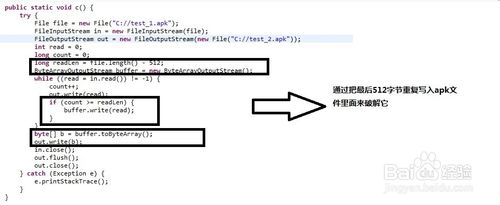
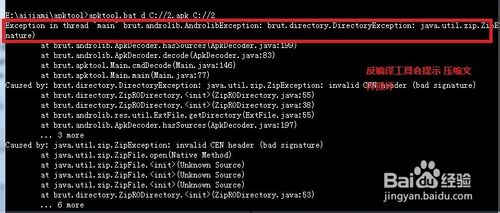
Android代码混淆

* 1 如下图，对Android 代码进行混淆后混淆器将代码中的所有变量、函数、类的名称加密为简短的英文字母代号，在APP被破解后增加破解者对代码的阅读难度。  [](http://jingyan.baidu.com/album/54b6b9c0c0c3772d583b470e.html?picindex=1)
* [](http://jingyan.baidu.com/album/54b6b9c0c0c3772d583b470e.html?picindex=2)
* 2 但是混淆的功效只能运作在APP已经被破解后，而且只是增加破解者的难度时间，对其防止破解的作用意义不是很大。
* 3 那么，Android代码混淆不能从根本上防破解，还有什么方法呢？接下来看：  END

反工具破解之伪加密

* 1 伪加密是Android4.2.x系统发布前最流行的加密方式之一，通过java代码对APK(压缩文件)进行伪加密，其修改原理是修改连续4位字节标记为”P K 01 02”的后第5位字节，奇数表示不加密偶数表示加密。伪加密后的APK不但可以防止PC端对它的解压和查看也同样能防止反编译工具编译。  [](http://jingyan.baidu.com/album/54b6b9c0c0c3772d583b470e.html?picindex=3)
* [](http://jingyan.baidu.com/album/54b6b9c0c0c3772d583b470e.html?picindex=4)
* 2 但是伪加密对其APK加密后市场也无法对其进行安全检测，部分市场会拒绝这类APK上传市场。伪加密的加密方式和解密方式也早已公布导致它的安全程度也大大降低。Android4.2.x系统无法安装伪加密的APK。
* 3 从上可看，Android代码混淆确实不如伪加密。但是Android代码混淆、伪加密也不是最靠谱的方法。接下来继续看。  END

反工具破解之APK压缩文件破解

* 1 APK在PC上面可以看作一个压缩文件，在Android系统里面它就是一个手机系统软件文件。Android系统对APK的识别是从标志头到标志尾，其他多余数据都会无视。所以说在标志尾添加其他数据对把APK看做压缩文件的PC端来说这个文件被破坏了，所以你要对其进行解压或者查看都会提示文件已损坏，用反编译工具也会提示文件已损坏，但是它却不会影响在Android系统里面的正常运行和安装而且也能兼容到所有系统。  [](http://jingyan.baidu.com/album/54b6b9c0c0c3772d583b470e.html?picindex=5)
* [](http://jingyan.baidu.com/album/54b6b9c0c0c3772d583b470e.html?picindex=6)
* [](http://jingyan.baidu.com/album/54b6b9c0c0c3772d583b470e.html?picindex=7)
* 2 但是这种APK压缩包破坏存在APK伪加密一样的问题，个别市场会不能识别导致不能上传市场。使用压缩文件修复工具也能把它修复好让我们做的保护消失。
* 3 三个分析，Android代码混淆、伪加密、压缩文件破解等均无法从根本上解决Android代码混淆防反编译、破解的问题。那么，我们继续探讨。  END

使用第三方平台加密

* 1 Google对Android APP的安全处理是给APK简单加密、java层源码加壳保护，对核心so库、资源文件、主配文件、第三方jar包却没有任何保护处理。所做的保护也早已被黑客攻破但并没有做任何升级维护导致Android APP现在没有任何安全性。Android代码混淆
* 2 拿移动应用安全行业的第三方平台——“爱加密”来说，在Android代码混淆的基础上，以源码加壳保护、so库核心代码加壳保护、资源文件签名保护、APK防二次打包保护等方面来对APK进行全方面保护。而且”爱加密”加密后的APP完全不会影响其运行效率和用户体验，兼容性是最优的。

3 综上所述，Android代码只做混淆是不行不行的，要真正保护Android代码的安全，还是要寻求更安全的加密保护技术！表只做Android代码混淆啦！