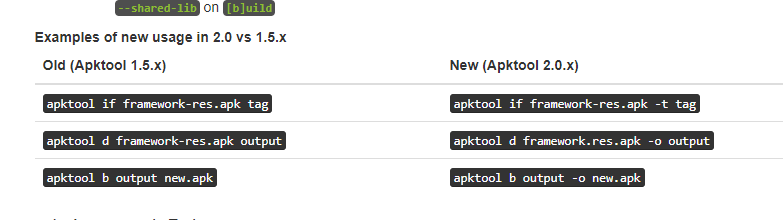
目的：打出的包要加密，之前一直用的androidkiller，但是比较麻烦，用这个命令行自动解包-加密-替换-打包

网上很多教程都是老版本的，我下的是2.4 很多命令都变了

安装教程 <https://ibotpeaches.github.io/Apktool/install/>



这是命令变化的地方，网址是<https://ibotpeaches.github.io/Apktool/documentation/>

主要就是后面的输出目录要加个 -o

在解包的时候会报个警告，大意是输出目录不稳定啥的，这个东西应该是记录了一些信息，所以目前做法是调用自己的文件夹 -p E:\TestApk\111

解包命令：apktool d -f E:\TestApk\7.4\_yijie\_final.apk -o E:\TestApk\222 -p E:\TestAp\111

-f是强制替换

打包命令：apktool b -c E:\TestApk\222 -o E:\TestApk\111\123.apk -p E:\TestApk\111

这么整完之后安装不了，我查了一下说没有签名

-c是官网说加了会把签名拷贝，就有签名了，但是还是安装失败

应该还是证书的问题，想了一下，如果可以反编译包，然后再用原证书可以反编译回去，不就可以随便改别人apk了么，所以androidkiller编译的时候会指定新的证书

**最终格式：**

解包：apktool d -f E:\TestApk\7.4\_yijie\_final.apk -o E:\TestApk\222 -p E:\TestAp\111

打包（不加-c，本身的包有签名的话，签名会失效）：apktool b E:\TestApk\222 -o E:\TestApk\111\123.apk -p E:\TestApk\111

签名：

1.生成新的apk

jarsigner -keystore E:\TestApk\111\androidkiller.keystore -signedjar E:\TestApk\111\sign.apk E:\TestApk\111\123.apk zykj

2.签名原apk

jarsigner -keystore E:\TestApk\111\androidkiller.keystore E:\TestApk\111\123.apk zykj

如果要自动密钥，则

jarsigner -keystore E:\TestApk\111\androidkiller.keystore -storepass 123456 E:\TestApk\111\123.apk zykj