系统签名

有时候，我们开发的apk需要用到系统权限，需要在AndroidManifest.xml中添加共享系统进程属性：

android:sharedUserId="android.uid.system"

android:sharedUserId="android.uid.shared"

android:sharedUserId="android.media"

这时候apk的签名就需要是系统签名(platform、shared或media)才能正常使用。否则报INSTALL\_FAILED\_SHARED\_USER\_INCOMPATIBLE错

# 常用系统签名方式

## Android源码环境下签名

这种方式比较麻烦，你需要有编译过的源码环境，并按如下步骤：

1、拷贝App源码到Android源码的packages/apps/目录下，且App源码是普通(Eclipse)格式的  
2、配置Android.mk，在其中添加

LOCAL\_CERTIFICATE := platform 或 shared 或 media

3、使用mm编译App，生成的apk即系统签名

## 手动重新签名

这种方式比在源码环境下签名简单，App可以在Eclipse或Android Studio下编译，然后给apk重新签名即可。  
但这种方式在频繁调试的时候比较痛苦，即使写成脚本，也需要重复一样的操作。

### 相关文件

platform.x509.pem、platform.pk8、signapk.jar

../build/target/product/security/platform.x509.pem

../build/target/product/security/platform.pk8

../out/host/linux-x86/framework/signapk.jar

signapk源码路径: ../build/tools/signapk

### 签名命令

java -jar signapk.jar platform.x509.pem platform.pk8 old.apk new.apk

步骤

1、将相关文件及源apk文件置于同一路径下  
2、检查源apk包，去掉META-INF/CERT.SF 和 META-INF/CERT.RSA 文件（可以忽略这一步）  
3、执行签名命令即可

## Android Studio系统签名

让Android Studio集成系统签名，需要用到一个工具keytool-importkeypair，详见下文。

keytool-importkeypair is A shell script to import key/certificate pairs into an existing Java keystore

这个工具的作用是将系统签名的相关信息导入到已有的签名文件里。可从[这里](https://github.com/getfatday/keytool-importkeypair" \t "_blank)下载。工具的使用方法可以通过--help或README.textile来寻求帮助，或参考[使用keytool工具](http://czj4451.iteye.com/blog/1487684" \t "_blank)这篇文章。

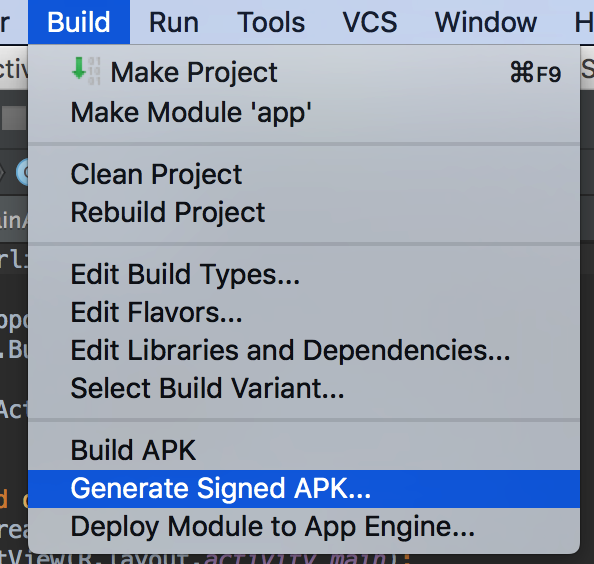
### 相关文件

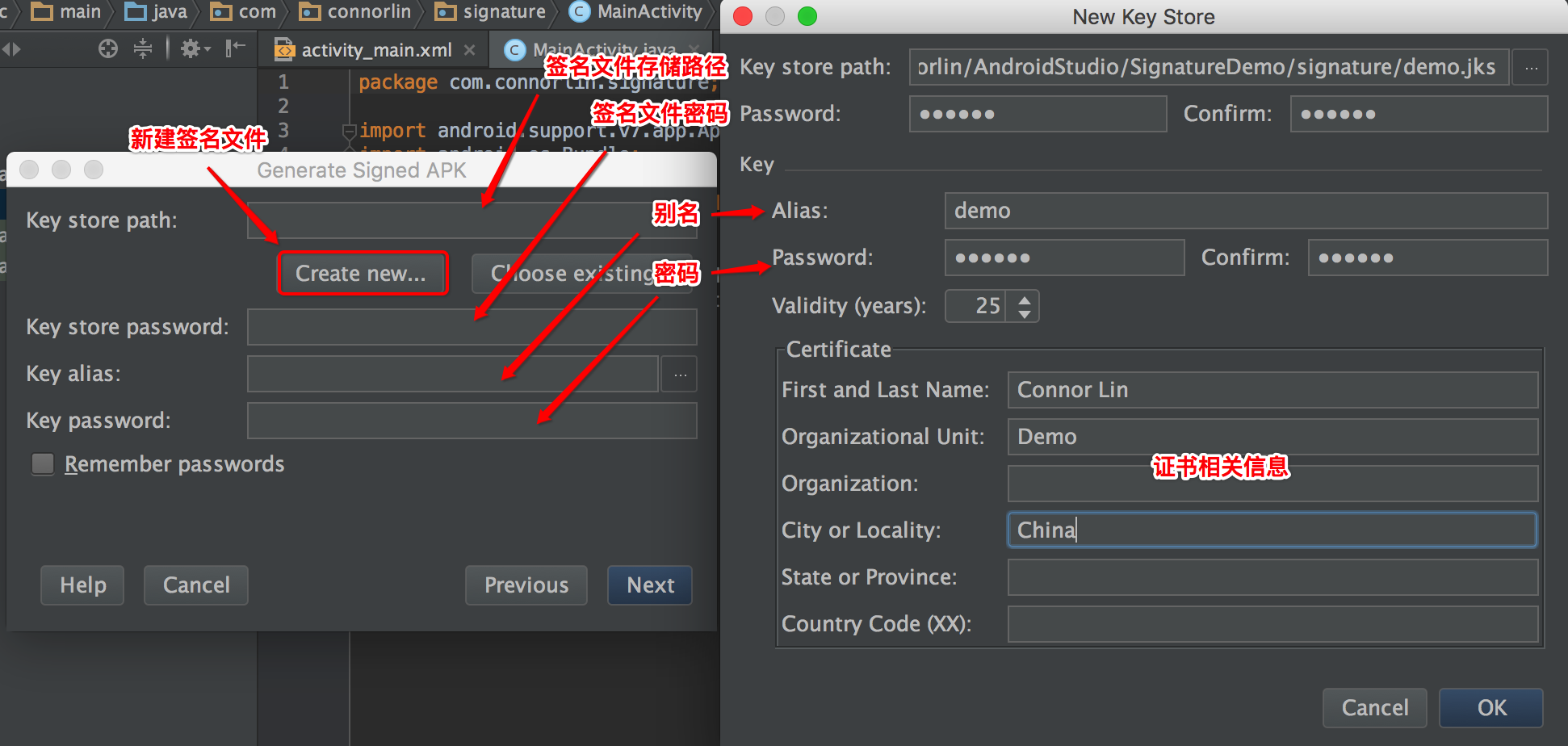
platform.x509.pem、platform.pk8、keytool-importkeypair、demo.jks、signature.sh

我的做法是在App根目录新建Signature文件夹专门存放签名相关文件。

### 步骤

1、生成demo.jks签名文件





2、编写签名脚本signature.sh，内容如下：

#!/bin/sh

# 转换平台签名命令

./keytool-importkeypair -k demo.jks -p 123456 -pk8 platform.pk8 -cert platform.x509.pem -alias demo

# demo.jks : 签名文件

# 123456 : 签名文件密码

# platform.pk8、platform.x509.pem : 系统签名文件

# demo : 签名文件别名

为脚本文件添加可执行权限：（在linux下执行）

$ sudo chmod a+x signature.sh

执行脚本：

$ ./signature.sh

3、配置builde.gradle

在android区域下(与defaultConfig同级)添加配置：

signingConfigs {

release {

storeFile file("../signature/demo.jks")

storePassword '123456'

keyAlias 'demo'

keyPassword '123456'

}

debug {

storeFile file("../signature/demo.jks")

storePassword '123456'

keyAlias 'demo'

keyPassword '123456'

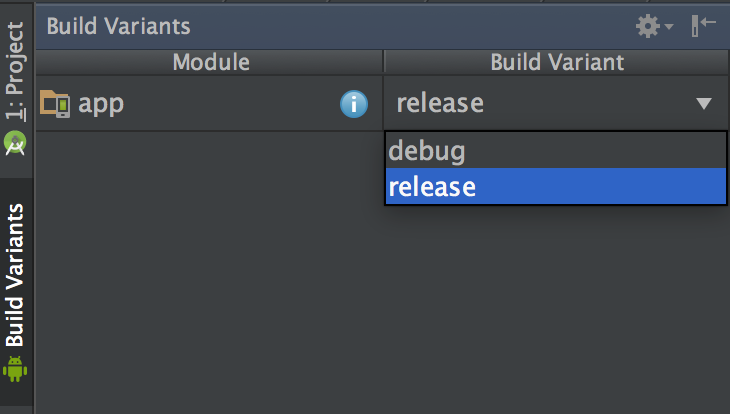
}

}

这样debug或release apk就带有系统签名了。

如果想直接Run app就是release版且带系统签名的apk，还需修改：

1) 变体app



2) 修改buildTypes：

buildTypes {

release {

minifyEnabled false

proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.txt'

signingConfig signingConfigs.release

}

}

这样直接Run app就是带系统签名的release版apk了。

**That's all, Enjoy it!**

参考：

让Android Studio支持系统签名(证书)

http://www.jianshu.com/p/47265c8899b5