**Android 将第三方应用转系统应用**

[](https://www.jianshu.com/u/0ff9b1fc0dfb)

[雾滴](https://www.jianshu.com/u/0ff9b1fc0dfb) 关注

2017.05.03 16:36\* 字数 197 阅读 1441评论 1喜欢 2

防止自己app被杀死或者获取某些系统权限，那就把app变为系统权限app吧

**1.打包生成已经签名app**

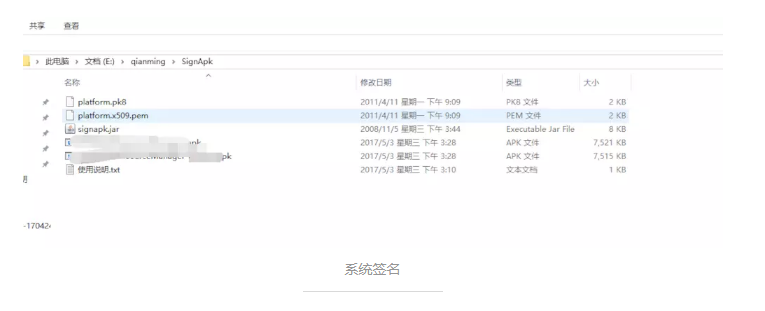
添加：android:sharedUserId="android.uid.system"



配置文件

**2.使用通用签名来重新给apk文件签名。**

准备好platform.pk8、platform.x509.pem和签名工具signapk.jar（3个文件的下载地址），还有自己的apk，放在同一个文件夹下



进入当前文件目录执行：SignApk.jar platform.x509.pem platform.pk8 input.apk out.apk

资源下载：[友情下载](https://link.jianshu.com/?t=http://download.csdn.net/download/qq_14876677/9173447" \t "_blank)

**3.获取root权限 push app到系统 重启**

adb remount

adb push yourapk.apk system/app

adb shell

chmod yourapk.akp 777

exit

adb reboot;

其它方法：1.用第三方软件 2.重新编译 刷入系统应用 待尝试

# Android 将自己的应用改为系统应用

2016年11月30日 16:19:11 [薛瑄](https://me.csdn.net/xx326664162) 阅读数：39463

版权声明：本文为博主原创文章，转载请标明出处。 https://blog.csdn.net/xx326664162/article/details/53406933

转载请标明出处：<http://blog.csdn.net/xx326664162/article/details/53406933> 文章出自：[薛瑄的博客](http://blog.csdn.net/xx326664162)

所谓系统程序就是system/app目录中的程序，普通应用转换成系统程序后有稳定、减少内存（DATA）空间占用、恢复出厂设置后不会消失、修改系统时间、调用隐藏方法、系统关机重启、静默安装升级卸载应用等等等等优点，想知道怎么操作？接下来我们介绍三种方法。

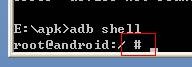
# 第一种：使用ADB命令将app安装在system/app目录下

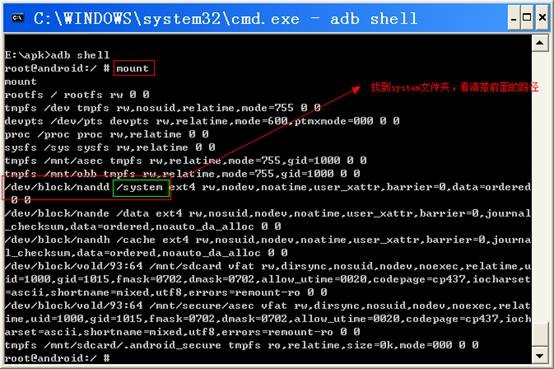
参考：[android 将自己的应用改为系统应用](http://www.apkbus.com/thread-130756-1-1.html)

## 这种方法的原理就是：

1、把apk文件移动到system/app目录，   
2、.so文件移动到system/lib目录。   
3、修改相应的权限

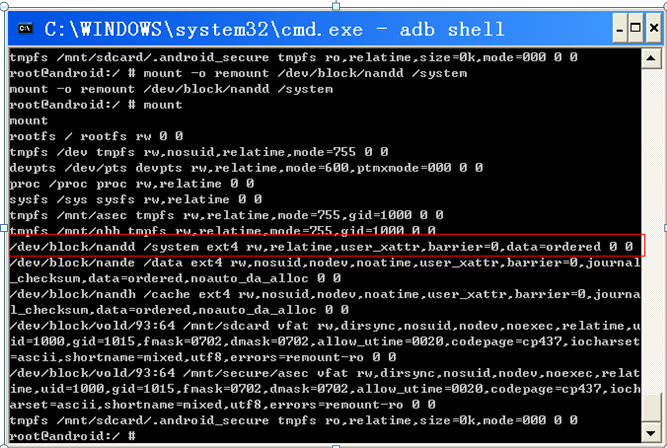
## 操作步骤：

**1. 将你的手机数据线，插上，把你的设备设置为允许usb调试**   
**2. 打开命令终端cmd**   
**3. 输入命令 adb shell**   
**4. 确定能进入系统**   


**5. 输入命令 mount**   


**6. 因为system默认是只读文件夹，所以根据上面的提示输入下面命令，使其变为可读写**   
**mount -o remount /dev/block/nandd /system** （图）   
这里写图片描述

**再出输入 mount 查看system和上面的不一样了，说明正确**



**7. 输入 exit 退出android系统终端**

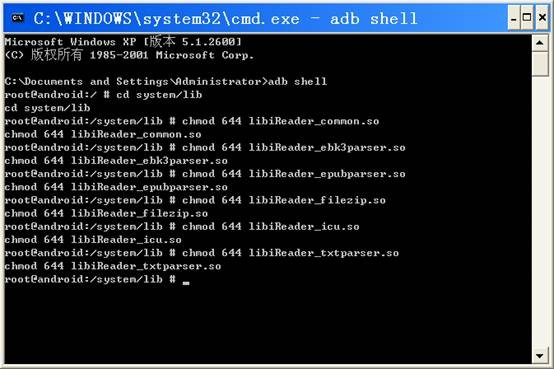
**8. 解压你的apk文件，进入查看lib/armeabi文件夹下有没有 .so文件，如果没有这种库文件的话，直接跳到第10步，（因为有些apk文件是要调用动态链接库的，你不拷贝的话，就没有办法运行！会报错）如果有的话， 将这些\*.so文件都拷贝到/system/lib文件夹下：**

命令：adb push libiReader\_txtparser.so system/lib   
这里写图片描述

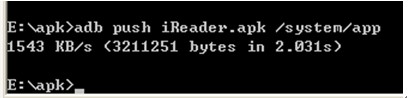
**9、拷贝完了之后呢，要给这些库文件添加权限，看看别的库文件权限是几**

chmod 644 xxxxx.so

* 1

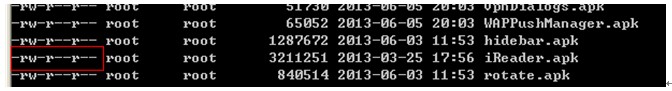


**10. 将你的apk文件拷贝进入/system/app（该文件夹里存放着所以系统级别的apk），图中我是将iReader.apk拷贝过去的**



**11. 再次进入android终端 adb shell**   
**12. 进入system/app文件夹 cd system/app**   
**13. 查看其他apk的权限 ll 能看出区别**



**14. 修改iReader.apk权限使其和其他的一样 chmod 644 iReader.apk**   


**15. 搞定这些之后，重启设备 reboot**   
**16. 看看系统里面是不是安装好了该应用，点击一下，看是否正常运行，可以的话，再检测是否无法卸载！**

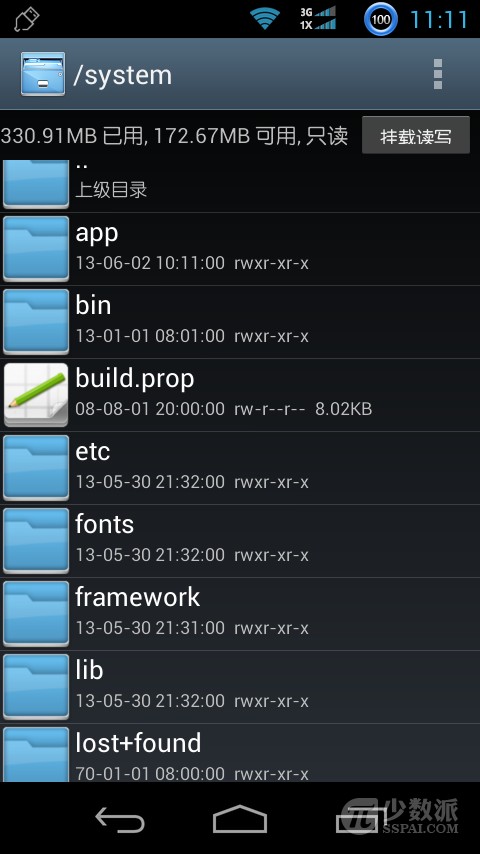
# 第二种：借助工具把app转为系统应用（原理和方法一一样）

转载：[安卓进阶教程：怎样把应用转换成系统程序](http://sspai.com/23427)

RE管理器转换和LINK2SD都可以实现，任选其一即可

## 使用RE管理器转换

1、首先我们把需要转换的程序在电脑上用压缩软件打开 ，看有没有lib这个目录。如果有，再把lib目录打开，直到出现以.so结尾的文件，把文件都拖出来备用。



2、把需要转换的应用（apk文件）连同刚拖出的.so文件（如果有），放到手机内存卡，   
3、用RE管理器复制到system目录，把权限更改如图，   
4、把更改权限后的apk文件移动到system/app目录，.so文件移动到system/lib目录。   
5、完成后重启手机，应用就转换成系统程序了。

## 使用LINK2SD转换



如果感觉以上方法麻烦，也可借助工具来操作，LINK2SD、钛备份等软件都可以把普通程序转换为系统程序，以LINK2SD为例，打开LINK2SD，找到需要软件的程序，点击，再点操作，选择转换系统应用，接着会有个确认窗口，确认后，重启手机程序就转换好了。

# 第三种：使用signapk打包成系统应用

参考：   
[android之使用signapk打包成系统应用，获取系统权限](http://blog.csdn.net/wangsfine/article/details/51205564)   
[使用platform密钥来给apk文件签名的命令](http://www.devdiv.com/platform_apk_-article-1853-1.html)   
[Android安全开发之通用签名风险](https://segmentfault.com/a/1190000006205334)   
[关于android:sharedUserId=”android.uid.system”这个系统级权限](http://www.eoeandroid.com/thread-329398-1-1.html)   
[安装APK 时, 提示” 共享用户权限不完整” , 不能安装成功, 如何解决?](http://bbs.csdn.net/topics/390105214)   
<https://github.com/android/platform_build/tree/master/target/product/security>

为了更好地理解下面介绍的两种方法的原理，先来学习几个概念：

## Android应用签名机制

Android系统要求安装的应用必须用数字证书进行签名后才能安装，并且签名证书的私钥由应用开发者保存。签名证书的生成也由开发者自己生成。在应用安装时会校验包名（package name）和签名，如果系统中已经存在了一个相同的包名和签名的应用，将会用新安装的应用替换旧的；如果包名相同但是签名不同，则会安装失败。

**为什么需要数字签名？**

数字签名是防止要保护的内容被篡改，用非对称加密算法。先对要保护的内容进行消息摘要，用私钥对消息摘要进行加密，生成数字签名，将数字签名和要保护的内容一起分发出去。 内容接收者用公钥对数字签名解密得到发送者给的消息摘要A，内容接收者对接收到的内容进行用相同的消息摘要算法处理得到消息摘要B，对比A和B是否相同，来判定传送的内容是否被篡改。 正常的APK文件是个ZIP压缩文件，除了应用的可执行文件、资源文件，还包括这些可执行文件、资源文件的摘要信息，数字证书的公钥信息等。并且通过这些签名信息可以确定APP和其开发者的关系。

**进行签名需要的工具有哪些？**

对apk进行签名需要用到签名证书和签名工具。Android系统要求对APP进行签名的数字证书可以由开发者自己生成。签名工具有jarsigner和signapk。jarsigner是Java本身自带的一个工具，他也可以对jar进行签名的；而signapk是专门为了Android应用程序apk进行签名的工具。二者的区别是：jarsigner工具签名时使用的是keystore签名文件，signapk工具签名时使用的是pk8，x509.pem文件。

**签名后的文件都有哪些？**

应用签名完后在应用的META-INF目录下会有三个文件：

CERT.RSA、CERT.SF和MANIFEST.MF。

**MANIFEST.MF**中保存了所有其他文件的SHA1摘要并base64编码后的值。

**CERT.SF**文件 是对MANIFEST.MF文件中的每项中的每行加上“rn”后，再次SHA1摘要并base64编码后的值（这是为了防止通过篡改文件和其在MANIFEST.MF中对应的SHA1摘要值来篡改APK，要对MANIFEST的内容再进行一次数字摘要）。

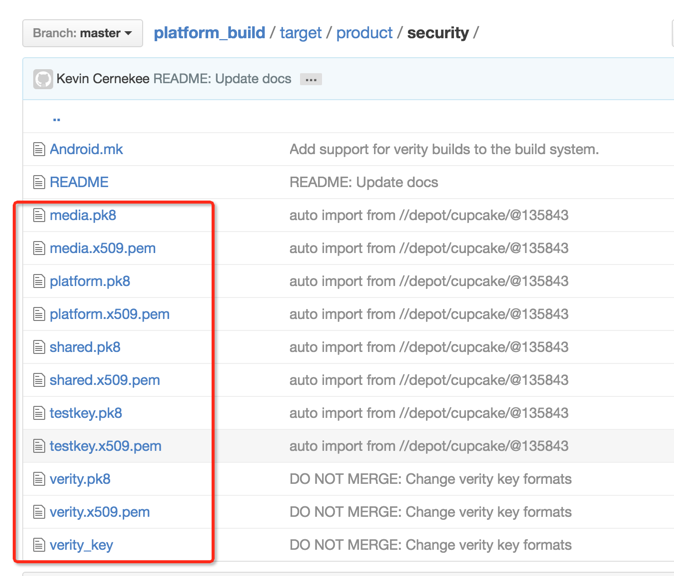
**CERT.RSA** 文件：包含了签名证书的公钥信息和发布机构信息。

对安装包的校验过程在源码的frameworks/base/core/java/android/content/pm/PackageParser.java类中可以看到

**什么是通用签名？**

搭建好Android开发环境后（使用Eclipse或Android Studio），对APK签名的默认密钥存在debug.keystore文件中。在linux和Mac上debug.keystore文件位置是在**~/.android**路径下，在windows目录下文件位置是**C:\user\用户名.android**路径下。

除了debug.keystore外，在AOSP发布的Android源码中，还有以下几个证书是公开的，任何人都可以获取，在源码的**build/target/product/security**目录中：



这几个证书的作用：

**testkey**

Generic default key for packages that do not otherwise specify a key.

**platform**

Test key for packages that are part of the core platform.

**shared**

Test key for things that are shared in the home/contacts process.

**media**

Test key for packages that are part of the media/download system.

**verity**

Test Key for verifiedboot system imagein Android Lollipop. Sign boot.img,sign verity metadata in system.img.

**通用签名风险：**

（1）如果攻击者的应用包名与目标应用相同，又使用了相同的密钥对应用进行签名，攻击者的应用就可以替换掉目标应用；

（2）另外目标应用的自定义权限android:protectionlevel为“signature”或者“signatureOrSystem”时，保护就形同虚设；

（3）如果设备使用的是第三方ROM，而第三方ROM的系统也是用AOSP默认的签名，那么使用如果使用系统级签名文件签名过的应用，权限就得到了提升。

## 具体的实现方法：

### 第一种是需要在Android系统源码的环境下用make来编译：

1. 在应用程序的AndroidManifest.xml中的manifest节点中加入android:sharedUserId=”android.uid.system”这个属性。
2. 修改Android.mk文件，加入LOCAL\_CERTIFICATE := platform这一行
3. 使用mm命令来编译，生成的apk就有修改系统时间的权限了。

### 第二种：

下面着重介绍一个这个方法：

**1. 加入android:sharedUserId=”android.uid.system”这个属性。**

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.example.jant.addview"

android:sharedUserId="android.uid.system" >

<application

...（省略若干代码）

</application>

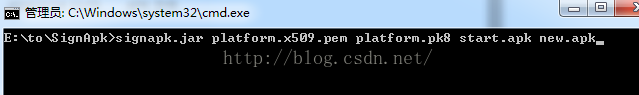
</manifest>

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9

**2. 使用自己的签名文件，生成apk**

**3. 使用通用签名来重新给apk文件签名。**

3.1 .准备好platform.pk8、platform.x509.pem和签名工具signapk.jar（3个文件的下载地址），还有自己的apk，放在同一个文件夹下。

3.2 在cmd下进入到该文件夹后，使用如下命令：   


3.3 回车后我们的文件夹下已经多了一个new.apk文件了，这就将我们的应用打包成系统应用

你也可以在github中去下载，但是下载的是SignApk.java，需要进行一些处理，如下

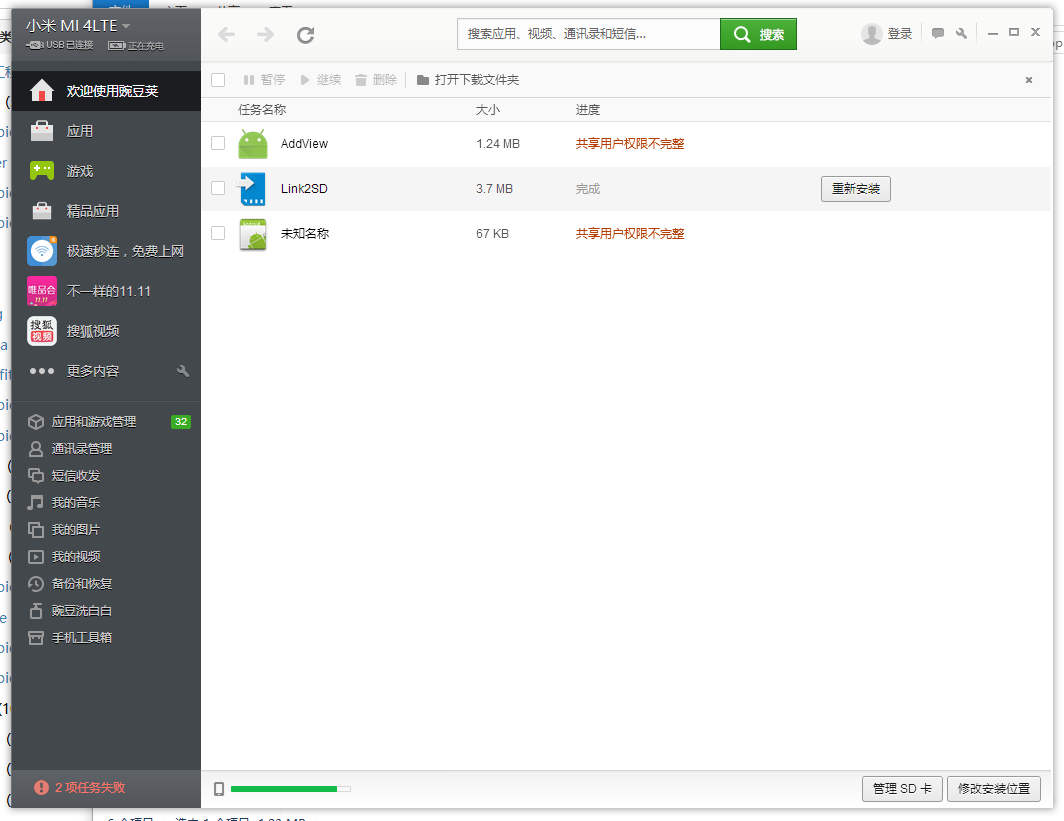
1.1.进入\build\target\product\security,找到【platform.pk8】和【platform.x509.pem】系统密钥。   
1.2.进入\build\tools\signapk找到SignApk.java，运行 javac编译成SignApk.class   
1.3.执行命令java com.android.signapk.SignApk platform.x509.pem platform.pk8 input.apk output.apk

## 最后解释一下原理

首先加入android:sharedUserId=”android.uid.system”这个属性。通过Shared User id,拥有同一个User id的多个APK可以配置成运行在同一个进程中。那么把程序的UID配成android.uid.system，也就是要让程序运行在系统进程中，也就是系统应用。

只是加入UID还不够，如果这时候安装APK的话发现无法安装，提示签名不符，原因是程序想要运行在系统进程中还要有目标系统的platform key，就是上面第二个方法提到的platform.pk8和platform.x509.pem两个文件。用这两个key签名后apk才真正可以放入系统进程中。第一个方法中加入LOCAL\_CERTIFICATE := platform其实就是用这两个key来签名。

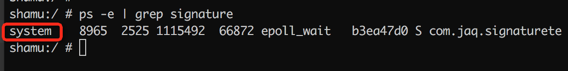
这也有一个问题，就是这样生成的程序只有在原始的Android系统或者是自己编译的系统中才可以用，因为这样的系统才可以拿到platform.pk8和platform.x509.pem两个文件。要是别家公司做的Android上连安装都安装不了。试试原始的Android中的key来签名，程序在模拟器上运行OK，不过放到小米四上安装uibl，如下图，这样也是保护了系统的安全。



最后还说下，这个android:sharedUserId属性不只可以把apk放到系统进程中，也可以配置多个APK运行在一个进程中，这样可以共享数据，应该会很有用的。   
你在Manifest.xml里声明使用了shareuserid 或者一些特殊permission，比如你shareuserid到uid.system，就必须使用系统platform签名来签你的apk，否则是不能安装的，同理share到其他用户id或者其他process上也是得用跟那个process运行的apk一样的签名   
一般签名肯定是厂商私有的，你肯定是没办法了，除非机器烧的是开发版本(eng)

# 检验app是否已经是系统应用

查看应用的进程属性，如果是system用户组，说明已经是系统应用。



**关注我的公众号，轻松了解和学习更多技术**   


# Android App安装成系统应用

2016年06月13日 23:27:00 [ElonSpace](https://me.csdn.net/elonspace" \t "_blank) 阅读数：5693

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/elonspace/article/details/51661349

**简单介绍一下如何将Android App安装成系统应用：**

**1、首先你要确保你手中的机器已经root过**

**2、将APK copy到sd卡根目录下**

**3、进入终端**

adb shell

**4、切换超级用户**

su

**5、挂载**

mount -o remount /system

**6、将APK copy到系统目录下**

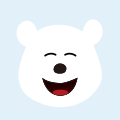
busybox mv /storage/sdcard0/xxx.apk /system/app

**7、安装APK**

adb install /system/app/xxx.apk

**8、重启机器**

# 将第三方apk变成系统apk

[](http://blog.51cto.com/670176656)

[671076656](http://blog.51cto.com/670176656)0人评论[1919人阅读2015-01-27 11:02:23](javascript:;)

将第三方apk变成系统apk

# 1. 前提条件：有root权限；

# 2. 具体步骤

## 2.1设置FLAG\_PERSISTENT

在AndroidManifest.xml文件中，加入android:persistent="true" ；

## 2.2设置FLAG\_SYSTEM

将你写的应用程序放到/system/app/目录下即可。具体操作如下：

### 2.2.1 Copy apk到/system/app/目录下

使用Cygwin或cmd进入你工程所在的bin目录中，然后执行如下两条命令：

adb remount

adb push yourpackage.apk /system/app/

命令adb remount是为了获取系统目录的临时操作权限，然后使用push命令将自己写的程序包上传到系统应用程序目录中。

### 2.2.2 Copy apk的so文件（如果apk有so的话）

复制它们到system/lib中，方法：

2.2.2.1 使用Eclipse的文件浏览器

       网上查找

2.2.2.2 使用手机助手

    网上查找

2.2.2.3 使用adb 命令

      使用cmd进入so文件所在目录，执行如下命令（有多个文件的话，要多次执行把每个so文件copy到目录下）：

    adb push 你的so文件名.so /system/lib

## 2.3 重启设备

    进行完以上操作，重启设备后，你的apk就变成系统的apk了。

# 安卓进阶教程：怎样把应用转换成系统程序

2013年06月03日

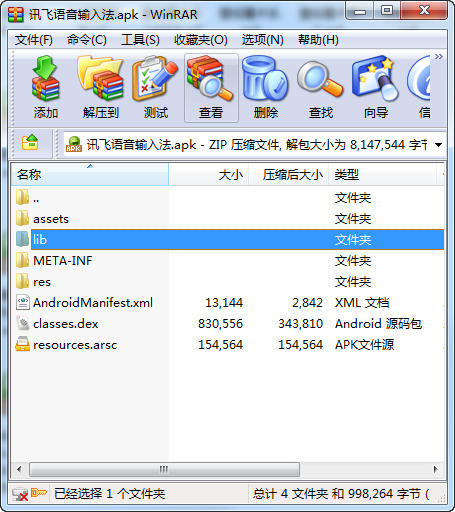
[穿越](https://sspai.com/user/65)

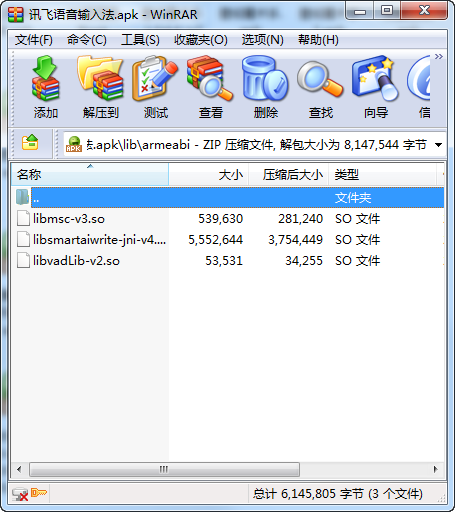
#### [穿越](https://sspai.com/user/65)

所谓系统程序就是system/app目录中的程序，普通应用转换成系统程序后有稳定、减少内存（DATA）空间占用、恢复出厂设置后不会消失等等优点，想知道怎么操作？认真看教程吧。

本教程以下内容需要有ROOT权限，并且已安装RE管理器等工具。首先我们需要判断什么样的程序适合转换成系统程序，转换后的优点。上面已说过，转换后也会有缺点，就是系统程序更新比较麻烦。如果让其自动更新，更新后程序还是会安装在DATA空间，对于系统程序的更新我们一般是用下面的方法再转换一次，所以我们通常选择那些更新不是很频繁、又需要安装在手机内存中，对于稳定运行环境要求较高的应用。

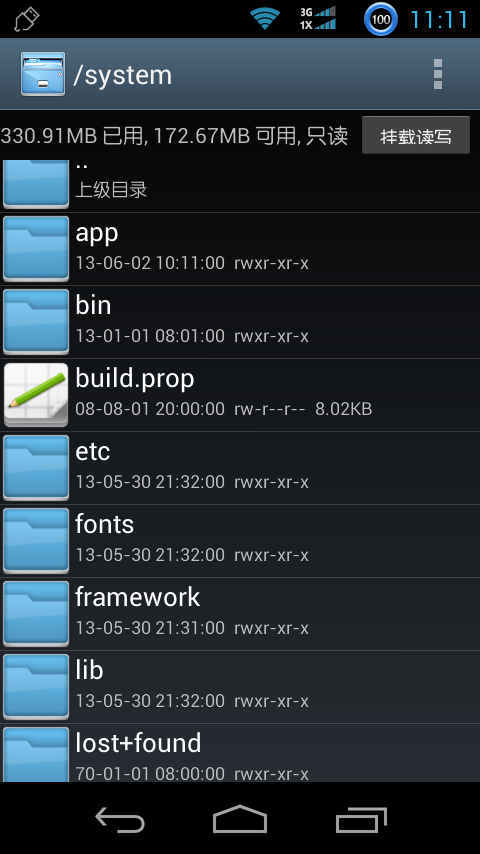
## 1.使用RE管理器转换





首先我们把需要转换的程序在电脑上用压缩软件打开 ，看有没有lib这个目录。如果有，再把lib目录打开，直到出现以.so结尾的文件，把文件都拖出来备用。





把需要转换的应用（apk文件）连同刚拖出的.so文件（如果有），放到手机内存卡，再用RE管理器复制到system目录，把权限更改如图，再把更改权限后的apk文件移动到system/app目录，.so文件移动到system/lib目录。完成后重启手机，应用就转换成系统程序了。

## 2.使用LINK2SD转换





如果感觉以上方法麻烦，也可借助工具来操作，LINK2SD、钛备份等软件都可以把普通程序转换为系统程序，以LINK2SD为例，打开LINK2SD，找到需要软件的程序，点击，再点操作，选择转换系统应用，接着会有个确认窗口，确认后，重启手机程序就转换好了。

## 注意：

并不是所有的程序都可以按以上方法转换的，有些程序转换后会出错，只能放弃。有些apk文件lib目录有armeabi、mips、x86三个目录，这是根据CPU不同而准备的不同的.so文件，一般这些文件不提取也可以。

## 参考资料：

[安卓教程：手机电脑传输文件各方法讲解](https://sspai.com/23282)

[安卓新手教程：RE管理器RootExplorer使用 I](https://sspai.com/23271)

[安卓新手教程：RE管理器RootExplorer使用 Ⅱ](https://sspai.com/23272)

[Android ROOT后必备应用：LINK2SD](https://sspai.com/23209)

[Android安装系统应用](https://yuanyao.iteye.com/blog/811866)

**博客分类：**

* [Android](https://yuanyao.iteye.com/category/79956)

[Android](http://www.iteye.com/blogs/tag/Android)

  测试需要，将应用内置在手机当中，作为系统应用来启动。

  1.adb remount

  2.adb push \*\*.apk /system/app

  3.sync system(如果不执行，可能找不到应用启动的图标)

  可以去settings->app manager去看一下，在All里可以查到你安装的App。

## Android App 安装成系统应用

## 原

[[XngPro](https://my.oschina.net/XngPro)](https://my.oschina.net/XngPro)

[XngPro](https://my.oschina.net/XngPro) 发布于 2017/02/09 19:10

字数 108

阅读 70

收藏 0

点赞 0

[评论 0](https://my.oschina.net/XngPro/blog/834960#comments)

**简单介绍一下如何将Android App安装成系统应用：**

### ****1、首先你要确保你手中的机器已经root过****

### ****2、将APK copy到sd卡根目录下****

### ****3、进入终端****

adb shell

### ****4、切换超级用户****

su

### ****5、挂载****

mount -o remount /system

### ****6、将APK copy到系统目录下****

busybox mv /storage/sdcard0/xxx.apk /system/app

### ****7、安装APK****

adb install /system/app/xxx.apk

### ****8、重启机器****

# [[教程]](http://www.miui.com/forum.php?mod=forumdisplay&fid=370&filter=typeid&typeid=1940) [教你轻松把用户级软件变为系统级应用（图文教程）](http://www.miui.com/thread-2502520-1-1.html)

[楼主](http://www.miui.com/forum.php?mod=viewthread&tid=2502520)

996521|

2015-4-21 17:12 | 来自PC

|

[复制](http://www.miui.com/thread-2502520-1-1.html)

|

|

跳转 

|  |
| --- |
| 国际惯例！教程开始之前先唠嗑一下！                                   本教程的目的是什么?     有什么好处？             本教程可以把一个用户级（自己安装的）的软件改变成系统级软件（系统自带的）            好处是什么？我举个例子。有些软件是不会自启的。比如“蝰蛇音效”，用户级安            装的蝰蛇音效在你重启之后必须要点开它之后才会生效。而系统级的蝰蛇音效则不            用，因为系统在启动的时候回自动读取system/app下（系统级）的软件，之后应              用这些软件的设置。                                                          所需工具及条件：安卓手机一台（这是废话），手机必须具有root（安卓系统最高级权                                  限）， root管理器。                                          好了就唠嗑这些！下面开始正式教程！                                             以搜狗输入法为例！       第一步：安装root管理器，授予root权限，打开re管理器，进入用户级软件的apk存放      目录（data/app），找到搜狗输入法（com.sohu.inputmethod.sougou.apk），单      击他，然后点击查看，再点开lib目录，看看有没有动态链接库（.so文件），如果没有      的话请忽略第二步，有的话请一定要执行第二步（重中之重！！）。  https://attach.bbs.miui.com/forum/201504/21/163434tcbzo7hrsrellkib.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/163550ig5m0vks4z4dvann.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/163626k3fyww9yy3wjy613.png.thumb.jpg  https://attach.bbs.miui.com/forum/201504/21/163803pl4l8cqcqdlxdtl0.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/163743i7fpe2eouzjp0flu.png.thumb.jpg       第二步：刚才说了没有.so文件的请忽略这一步，有的话请一定要执行这一步。                     第一步过后请移步到用户级软件的动态链接库目录（data/data）。找到与搜                    狗输入法同名的文件夹（com.sohu.inputmethod.sougou），点进去，找                      到“lib”这个文件夹。然后把里面的所有文件拷贝到system/lib里面。  https://attach.bbs.miui.com/forum/201504/21/165318tdwvbzcobdrww93b.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/164734cd1uoor26ztoo6oo.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/164738pbk6xgsgux7ojkzs.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/164744vyr2cxl1fb1z6b63.png.thumb.jpg  https://attach.bbs.miui.com/forum/201504/21/164747kcw0negv0cnnygnd.png.thumb.jpg      第三步：移动完.so文件之后就可以正式移动apk文件了，打开data/app目录，找到搜                    狗输入法（com.sohu.inputmethod.sougou.apk），把它剪切到system/app                  目录。然后重启就ok了！  https://attach.bbs.miui.com/forum/201504/21/165935i3lzxeem2celmajl.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/165937djy3jm0sl0crdc3j.png.thumb.jpghttps://attach.bbs.miui.com/forum/201504/21/165941x14n8ikeinb49eo8.png.thumb.jpg          最后在唠嗑一下，并不是所有软件都带有动态链接库文件的（.so），带有的必须移          动到system/lib里面，没有的可以忽略！切记                  拒绝伸手党！加个分吧！亲！                                                 本教程完！ |

如果想要获取一些诸如WRITE\_SECURE\_SETTINGS，STATUS\_BAR之类的权限（signature|privileged）的话，可以直接吧APK放到/system/priv-app里面，不需要框架签名（Root直接拷进去重启，比如系统插件）。  
如果要拿到BRICK这类的权限（signature，话说要brick干什么，跑），就需要框架签名了，自己弄个ROM吧..