[插件开发之360 DroidPlugin源码分析（一）初识](http://blog.csdn.net/hejjunlin/article/details/52123450)

DroidPlugin的是什么？

      一种新的插件机制，一种免安装的运行机制，是一个沙箱（就是对于使用者来说，并不知道他会把apk怎么样）， 是模块化的基础。

DroidPlugin的缺点是什么？

     a.通知栏限制（无法在插件中发送具有自定义资源的Notification，例如： 1. 带自定义RemoteLayout的Notification 2. 图标通过R.drawable.XXX指定的通知（插件系统会自动将其转化为Bitmap）

     b.安全性担忧（可以修改，hook一些重要信息）

c.机型适配（不是所有机器上都能行，因为大量用反射相关，如果rom厂商深度定制了framework层，反射的方法或者类不在，容易插件运用失败）

    d. 需要预先注册权限（在Library中申请了原生系统所有的权限）

    e. 无法在插件中注册一些具有特殊Intent Filter的Service、Activity、BroadcastReceiver、ContentProvider等组件以供Android系统、已经安装的其他APP调用。

f. 缺乏对Native层的Hook，对某些带native代码的apk支持不好，可能无法运行。比如一部分游戏无法当作插件运行。

DroidPlugin的特点是什么？

    a.免安装（就是如果只要从网上下载一个apk，不用安装apk，在插件机制下，就能运行）

    b.无需修改源码（因为大量反射，代理，Binder相关，这些足以骗过framework层）

    c.二进制级别隔离

    d.插件之间可以相互调用

    e.解除耦合

    f.静默安装，就是前面说的不用安装，就可在插件机制中运行apk

    g.崩溃隔离，插件崩溃，对主应用来说，不会有明显影响

    h.还原插件自己的多进程机制，适配性

    i.模块隔离，如可以把UI和控制逻辑进行隔离，控制逻辑可用插件化的方式

官方特点说明：

1. 支持Androd 2.3以上系统
2. 插件APK完全不需做任何修改，可以独立安装运行、也可以做插件运行。要以插件模式运行某个APK，你**无需**重新编译、无需知道其源码。
3. 插件的四大组件完全不需要在Host程序中注册，支持Service、Activity、BroadcastReceiver、ContentProvider四大组件
4. 插件之间、Host程序与插件之间会互相认为对方已经"安装"在系统上了。
5. API低侵入性：极少的API。HOST程序只是需要一行代码即可集成Droid Plugin
6. 超强隔离：插件之间、插件与Host之间完全的代码级别的隔离：不能互相调用对方的代码。通讯只能使用Android系统级别的通讯方法。
7. 支持所有系统API
8. 资源完全隔离：插件之间、与Host之间实现了资源完全隔离，不会出现资源窜用的情况。
9. 实现了进程管理，插件的空进程会被及时回收，占用内存低。
10. 插件的静态广播会被当作动态处理，如果插件没有运行（即没有插件进程运行），其静态广播也永远不会被触发。

DroidPlugin的的基本原理是什么？

    a.共享进程：为android提供一个进程运行多个apk的机制，通过API欺骗机制瞒过系统

    b.占坑：通过预先占坑的方式实现不用在mainfest注册，通过一带多的方式实现服务管理

    c.Hook机制：动态代理实现函数hook，Binder代理绕过部分系统服务限制，IO重定向（先获取原始Object-->Read,然后动态代理Hook Object后-->Write回去，达到瞒天过海的目的）

插件Host的程序架构：

