浅谈：APP有哪些常被黑客利用的安全漏洞

作者：asd19860
原文链接：https://ld246.com/article/1442577397592
来源网站：链滴

首先，说到APP的安全漏洞，身为程序猿的大家应该不陌生；如果抛开安卓自身开源的问题的话，其要产生的原因就是开发过程中疏忽或者代码不严谨引起的。但这些责任也不能怪在程序猿头上，有时因为BOSS时间催得紧等很多可观原因。由国内移动应用安全检测团队爱内测（www.ineice.com）的TO给我们浅谈关于Android 系统的开源设计以及生态环境。

1. 应用反编译

漏洞：APK 包非常容易被反编译成可读文件，稍加修改就能重新打包成新的 APK。

利用：软件破解，内购破解，软件逻辑修改，插入恶意代码，替换广告商 ID。

建议：使用 ProGuard 等工具混淆代码，重要逻辑用 NDK 实现。

例子：反编译重打包 FlappyBird，把广告商 ID 换了，游戏改加插一段恶意代码等等。

2. 数据的存储与传输

漏洞：外部存储（SD 卡）上的文件没有权限管理，所有应用都可读可写。开发者把敏感信息明文存在 SD 卡上，或者动态加载的 payload 放在 SD 卡上。

利用：窃取敏感信息，篡改配置文件，修改 payload 逻辑并重打包。

建议：不要把敏感信息放在外部存储上面；在动态加载外部资源的时候验证文件完整性。

漏洞：使用全局可读写（MODE_WORLD_READABLE，MODE_WORLD_WRITEABLE）的内部存储式，或明文存储敏感信息（用户账号密码等）。

利用：全局读写敏感信息，或 root 后读取明文信息。

建议：不适用全局可读写的内部存储方式，不明文存储用户账号密码。

3. 密码泄露

漏洞：密码明文存储，传输。

利用：

root 后可读写内部存储。

SD 卡全局可读写。

公共 WiFi 抓包获取账号密码。

建议：实用成熟的加密方案。不要把密码明文存储在 SD 卡上。

4. 组件暴露 (Activity, Service, Broadcast Receiver, Content Provider)

漏洞：

组件在被调用时未做验证。

在调用其他组件时未做验证。

利用：

调用暴露的组件，达到某种效果，获取某些信息，构造某些数据。（比如：调用暴露的组件发短信、博等）。

监听暴露组件，读取数据。

建议：验证输入信息、验证组件调用等。android:exported 设置为 false。使用 android:protectionevel="signature" 验证调用来源。

5. WebView

漏洞：

恶意 App 可以注入 JavaScript 代码进入 WebView 中的网页，网页未作验证。

恶意网页可以执行 JavaScript 反过来调用 App 中注册过的方法，或者使用资源。

利用：

恶意程序嵌入 Web App，然后窃取用户信息。

恶意网页远程调用 App 代码。更有甚者，通过 Java Reflection 调用 Runtime 执行任意代码。

建议：不使用 WebView 中的 setJavaScriptEnabled(true)，或者使用时对输入进行验证。

6. 其他漏洞

ROOT 后的手机可以修改 App 的内购，或者安装外挂 App 等。

Logcat 泄露用户敏感信息。

恶意的广告包。

利用 next Intent。

7. 总结

APP的漏洞大部分都是因为开发人员没有对输入信息做验证造成的，另外因为 Intent 这种特殊的机制需要过滤外部的各种恶意行为。再加上安卓应用市场混乱，开发人员水平参差不齐。所以现在 Androi 应用的漏洞，恶意软件，钓鱼等还在不断增多。

再加上 root 对于 App 沙箱的破坏，Android 升级的限制。国内的安卓环境一片混乱，惨不忍睹。所，如果想要保证你的应用没有安全漏洞，就要记住：永远不要相信外面的世界。