《Android软件安全与逆向分析》

图书基本信息

书名：《Android软件安全与逆向分析》

13位ISBN编号：9787115308153

10位ISBN编号：7115308152

出版时间：2013-2

出版社：人民邮电出版社

作者：丰生强

页数：407

《Android软件安全与逆向分析》

内容概要

本书由浅入深、循序渐进地讲解了Android 系统的软件安全、逆向分析与加密解密技术。包括Android 软件逆向分析和系统安全方面的必备知识及概念、如何静态分析Android 软件、如何动态调试Android 软件、Android 软件的破解与反破解技术的探讨，以及对典型Android 病毒的全面剖析。
本书适合所有Android 应用开发者、Android 系统开发工程师、Android 系统安全工作者阅读学习。

《Android软件安全与逆向分析》

作者简介

丰生强（网名非虫）
Android软件安全专家。看雪论坛Android安全版版主；安卓巴士开发交流版版主。
对Android软件与系统安全有狂热的爱好和独到的见解，对Android系统的全部源代码进行过深入地研究和分析。逆向分析实战经验丰富。
在国内信息安全杂志上发表过多篇有价值的软件安全文章，目前就职于国内某Android开发企业，常年混迹于看雪论坛（ID非虫）。
作者邮箱：fei_cong@hotmail.com
愿与国内安全爱好者共同交流与探讨安全技术。

《Android软件安全与逆向分析》

书籍目录

章节摘录

版权页： 插图： Android NDK的platforms＼〈android版本〉＼arch—arm＼usr＼include＼jni.h头文件中，声明了所有可以使用到的JNI接口函数。该文件中有两个重要的结构体JNINativeInterface与JNIInvokeInterface，JNINativeInterface是JNI本地接口，实际上它是一个接口函数指针表，里面每一项都为JNI接口的函数指针，所有的原生代码都可以调用这些接口函数；而JNIInvokeInterface则是JNI调用接口，该结构目前只有3个保留项与5个函数指针，这5个函数用于访问全局的JNI接口，多用于原生多线程程序开发。既然JNI为开发人员提供的API就在这两个接口内，那么掌握了这些API的含义与使用方法是不是就可以理解为掌握了Android NDK开发了呢？笔者认为大多数时候是这样的。

《Android软件安全与逆向分析》

编辑推荐

国内第一本Android软件安全书别让你的代码成为别人的炮灰eoe全球最大中文Android开发者社区、看雪论坛、安卓巴士　　推荐每一位Android开发者的必备之书！在Android这个平台，我们已面临诸多威胁！2013年超过1800万台Android设备会遭遇某种形式的恶意软件的攻击。 恶意代码和病毒数量呈指数增长； 应用软件和数字内容的版权不断遭到侵害； 软件破解、篡改、广告库修改和植入、应用内付费破解等普遍存在； 软件本身的安全漏洞也频繁出现在国内外互联网企业的产品中； 数据泄露和账户被盗等潜在风险让人担忧； 官方系统、第三方定制系统和预装软件的漏洞不断被发现。要掌握主动，免除威胁，你应了解真相！安全技术几乎都是双刃剑，它们既能协助我们开发更有效的保护技术，也几乎必定会被攻击者学习和参考。这里的问题是，大量安全技术的首次大范围公开，是否会带来广泛的模仿和学习，从而引发更多的攻击？在这个问题上，安全界一直存在争议。这是任何一本里程碑式的安全书籍都无法绕开的话题。在《信息安全工程》中，Ross Anderson说："尽管一些恶意分子会从这样的书中获益，但他们大都已经知道了这些技巧，而好人们获得的收益会多得多。"正是基于对这种观念的认同，才使得这本书呈现于此。强实践性。这本书的几乎每一个部分，都结合实际例子，一步步讲解如何操作。缺乏可操作性，是Android安全方面现有论文、白皮书、技术文章最大的问题之一，很多人读到最后可能对内容有了一些概念，却不知道从何下手。强时效性。作者在写作的同时，持续跟随业界最新进展，刚刚发布不久的Santoku虚拟机、APIMonitor等工具，以及Androguard的新特性等，已然出现在了这本书中。

《Android软件安全与逆向分析》

精彩短评

1、很多工具都讲解了，比较贴近实际
2、 满满的干货，值得细看
3、知识涵盖全面，但有一定难度。
4、目前写android安全的书不多，此书应该是目前最佳的一本了
5、内容非常丰富，安卓逆向科普类型的书。
6、挺全挺详细的，还和写序的人一起吃过饭，挺不错的人~
7、朋友推荐，看起来不错。好评！
8、入门图书
9、这本书还是更加侧重软件本身加固的，对于代码改进的帮助不是很多。Android安全大火ing
10、内容深入浅出，图文并茂！很好的一本！
11、讲安全的书少，内容还行，不过需要深入的学习，目前看了小部分，了解了之前未知和未懂的地方。
12、这种书太坏啦!!!!
13、写的挺好，看的轻松
14、适合入门级
15、花了两天时间粗略的读了一遍。作者太厉害了！非常值得阅读并实践里面的例子
16、看雪网上的高手，长期专注和实战
17、非常适合逆向的入门，对于Android应用开发的一些底层dx也有比较详细的介绍，适合一看。
18、因为工作的缘故涉及到破解等方面的内容，所以快速的翻阅了大部分章节。应该说写的内容还算丰富，不过笔墨分配还可以斟酌。介绍的工具较多，需要参考其它资料文档，并且加以实例练习才能掌握。
某些章节还会随时研究。
19、8gjlo roa c7 5o7fb jm6f 6f 4a 9m nom ea rgfb bgfb tgm n6f b7 4m 87f ea 5om fa pm ta ea 4a6f h6 tm ea7 h6 9ma 86f 4a6f ta o7 46g om nogm 9m6f 4m b6g bm6f tm 96aqac o7 nh6ada roa ea7 no6g o6a nm6f taf noa pm t6g noa ogm tgfb no6fb roa t6g ia6f hafb rom6 d6g eam o6a tgm rma ogm roa eam hafb 96g 56g y7a noa 4m y6f t7
20、不错，质量很好，正版书！
21、好书！
22、google的说明文档里面都有的，只不过是英文的！不过还是非常感谢，
23、我默默的说下，花了三个小时浏览了一遍~~本书讲的比较细，看得出作者试图从底层、从原理来说明一些问题。。但是估计考虑到整个书的定位上，所以系统机制、软件内部机理还是不是很多~~内容上有不小篇幅介绍了smali、dex、arm指令内容。。最后想说的是，本书是Android软件逆向、调试、分析的入门好书，值的本人这样菜鸟学习~~
24、Android逆向良心之作
25、感觉就是作者博客零碎的结合体，内容都是泛泛带过，没有深入。虽然作者是大牛但是在写这部内容上还少了我们需要看到的东西，就是深入理解
26、这本就是我最想研究的那部分，包括反编译，类似个人笔记
27、想入门不错
28、真正认识Android，这是特别优秀的一本书。
29、很全面的一本书，适合初学者入门
30、内存不错，要是有个光碟附送源码就好了
31、在看雪上看非虫出书了 买一个~
32、非常不错，android安全的经典之作，感觉比国外的好多了，有基础、有核心、有广度、有深度的四有好书，不需要累述的决不浪费笔墨，图书馆续借两次后果断买了一本。
33、感觉笔误的地方比较多啊,是不是应该好好校对一下...
34、多看点安全的书，coding也就会多注意这方面的问题。
35、这本书毋庸置疑会成为逆向工程领域的经典著作。
36、论坛的博客，都是一个软件的介绍和怎么用，底层的东西都没有，没有什么有用的东西

37、没得说，干货很多，需要慢慢研究啊！
38、Android没什么好玩的。从入门到放弃
39、和不错的介绍Android逆向工程的书籍，条理清晰，实战性很强。对自己现在进行的任务很有帮助。虽然自己不是这个方向的大牛，但是确实觉得这款技术非常有趣！强烈推荐！
40、如果是刚学android的话看这本书会很头疼。。。。不过里面还是有不少干货的。
41、基础底子不行，看的有些吃力
42、对反编译和保护手段的介绍都无甚稀奇，对编译和安装等流程介绍也停留在代码流程的讲解上。关于 dalvik 指令 和 smali 的介绍还算殷实，需要的时候应该用的上。无论如何，只要签名抓牢，就还有最后的稻草。
43、居然有个人比我还早读过，肯定是编辑，哼……
44、涵盖内容丰富，但整体思维与主要处理思路方法解释不详。
也可能是系统安全技术，很多是crack
45、本来是为了毕业答辩买的这本书，可惜答辩完了也没用上，不过书还是不错的，有时间再抽空看看吧
46、还是要和伪机器码死磕才行啊。。。
47、挺好
48、快速入门书
49、很实用，内容详细，介绍的很清楚
50、很好的一本书。从这本书我知道了Java语言有多么的不安全。
51、介绍得很细，实例也挺多

《Android软件安全与逆向分析》

精彩书评

1、以前没有整体地从事过这方面的学习与研究, 感謝作者提供了这样一个详实的学习材料.全书内容主要分为APP Crack 与 系统的安全防护.各章节的例子非常适合打手研究与体会.建议将各章节的实例都实践一下, 从中掌握工具的使用与问题分析的方法思路.以前做过很长一段时间的系统兼容性的工作, 也许早点接触本书可以更快地对应用进行定位与分析.

《Android软件安全与逆向分析》

章节试读

1、《Android软件安全与逆向分析》的笔记-第98页

经过混淆的dex文件，反编译出来的smali代码可能没有源文件信息，因此，“.source”行的代码可能为空。

2、《Android软件安全与逆向分析》的笔记-第321页

里面主要讲了ROOT的机制原理.
目前可以通过X-ray工具进行系统安全性测试.

ROOT原理见http://qianjigui.iteye.com/blog/1990068

3、《Android软件安全与逆向分析》的笔记-第347页

外部存储安全, 内部存储安全: 将数据存放在系统的私有区或公共区. 如果非法程序具有对这些文件的读写权限(ROOT状态等)就可能进行恶意处理.
对于敏感数据, 需要对数据进行加密处理.

数据通信安全:
如同网络sniffer一样, Android系统内的Intent交互可能被第三方劫持或读取.
类似于一种通信组件的防护.
这种情况下, 一定要保证指定Intent的接收者与数据安全性.

4、《Android软件安全与逆向分析》的笔记-第329页

Android的权限检查机制, 主要有下层的系统资源权限检查与framework层的组件与服务调用检查.
其中下层的检查工作主要是通过linux自身有权限体系, 通过用户组进行资源的管理与分配.
而这个用户组又与应用自身的运行时状态关联.

串谋权限攻击是通过调用其它程序的组件完成提升自身权限的事情.
这其中涉及到Android系统的组件安全.
其原理基本上是合法程序的某些组件(A)具有非法程序希望使用的权限功能, 而A自身在被其它组件调用时并没有Check调用者来源, 从而导致了权限的外放.

一般的检测方法:
Mercury for Android

5、《Android软件安全与逆向分析》的笔记-第329页

<permission name="android.permission.INTERNET">
<group gid="inet" /> </permission>gid指定了所关联的用户组。声明了该权限后，运行时的进程所属的用户就会添加到inet用户组。

6、《Android软件安全与逆向分析》的笔记-第310页

《Android软件安全与逆向分析》

分析dex2jar, 利用特殊指令使其工作异常. 目前不晓得是否可以构造一个无法运行到的代码块, 放在程序中dex2jar无法识别.

7、《Android软件安全与逆向分析》的笔记-第96页

在分析Android程序过程中，我们需要先查看该程序是否具有Application类，如果有，就要看看它的OnCreate()方法中是否做了一些影响到逆向分析的初始化工作。譬如，商业软件的授权验证代码，检测软件的购买状态等等。

8、《Android软件安全与逆向分析》的笔记-第363页

利用DroidBox工具对病毒进行完整的分析.

9、《Android软件安全与逆向分析》的笔记-第339页

Activity 被劫持: 在后台运行一个监控程序, 发现需要进行劫持的Activity后, 立刻调用一个钓鱼Activity让用户进行输入.

```
  public void onBackgroundNewTask(View view) {
//后台服务发现指定程序后,启动自己
```

```
    TimerTask task = new TimerTask() {
      @Override
      public void run() {
        try {
          Thread.sleep(10000);
        } catch (InterruptedException e) {
          e.printStackTrace();
        }
        Log.d(TAG, "To Front");
        Intent intent = new Intent(getBaseContext(), MainActivity.class);
        //If set, this activity will become the start of a new task on this history stack.
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        getApplication().startActivity(intent);
      }
    };
    task.run();
  }

  public void onBacktoOldActivityBtn(View view){
    moveTaskToBack(true);
  }
```

并且将伪装的Activity设置为不在History列表中显示:

```
 &lt;activity
        android:name="com.example.app.MainActivity"
        android:label="@string/app_name"
        android:excludeFromRecents="true"&gt;
&lt;/activity&gt;
```

Broadcast Receiver 安全:
发送者安全: 一般的广播过程是全域可见, 如果广播发送者在发送与接收广播結果时,并没有对返回結果的回应者进行Check,就有可能收到非法发送者的假消息. 可以指定接收者, 或者检查返回的結果来源.
广播接收者安全: 可能广播被劫持, 非法程序构造假的广播消息. 接收者需要check来源.

Service安全:
组件对外提供了相关调用与处理服务功能, 这种情况下就需要Check调用者. 通过Android自带的自定义权限进行权限间的管理与共享.

Content Provider安全:
该组件用于数据间共享与操作, 需要关注该组件的可读写权限. 如果可能被所有程序访问, 就需要关注数据的安全性.

10、《Android软件安全与逆向分析》的笔记-第20页

在Android系统软件安全中,主要涉及APP自身防护与系统整体安全.
前半部分关注APP攻击与自身防护.

进行攻击(crack)的主要流程是: 获得APP->反编译->修改->重新编译->发布.

其中Dalvik Smali代码是可以进行来回编译的一个标准中间层.其相关语法见书中内容, 外围介绍见:

《Android软件安全与逆向分析》

http://qianjigui.iteye.com/blog/2000153

11、《Android软件安全与逆向分析》的笔记-第94页

5.2.1 反编译apk程序 94
关键描述信息集合: AndroidManifest.xml

5.2.2 程序的主Activity 95
<intent-filter>
<action android:name="android.intent.action.MAIN"/>
<category android:name="android.intent.category.LAUNCHER"/>
</intent-filter>
5.2.3 需重点关注的Application类 95

在整个程序刚开始时的应用入口,大量的状态与安全检查多出自这个地方.

5.2.4 如何定位关键代码——六种方法 96
信息反馈法: message, string.xml
特征函数法: Toast等进行状态反馈的代码
顺序查看法: 静态分析代码结构
代码注入法: 在分析后的代码的关键路径上插入用于调试的log
栈跟踪法: 动态获取运行程序的调用栈,用于分析程序的运行状态
Method Profiling: 动态的得到程序的运行状态

12、《Android软件安全与逆向分析》的笔记-第15页

第一章没有干货,失望,如果强说有的话,就是P15的Vendor ID和Product ID的自行添加了...

13、《Android软件安全与逆向分析》的笔记-第250页

一个直接调试Apk 文件的工具， 那就是AndBug， 它可以在没有源代码的情况下， 调试android上的java程序， 并支持断点、call stack查看、查看class、method信息等

AndBug的主页是：https://github.com/swdunlop/AndBug

整个工作原理就是利用了JDWP进行调试通信,好处是省去了源代码支持.另一方面,在Android Dalvik内部也确实是从dexcode出发的JDWP支持,所以没有影响.

常见问题:
安装时请先添加
sudo apt-get install python-dev python-bottle
如果出现: EOF问题,请关闭其他连接的DDMS(例如: Eclipse DDMS)

《Android软件安全与逆向分析》

版权说明