

Android 移动设备边信道隐私推测研究综述

史福田, 毛剑, 刘建伟

(北京航空航天大学电子信息工程学院, 北京 100191)

摘要: 针对存在敏感信息泄露的 Android 边信道, 基于隐私信息关联对象进行分类分析; 对主要攻击模型进行了分析总结, 包括推测隐私信息的核心方法及各个推测攻击的应用场景; 对针对边信道攻击的检测和抑制方法进行了总结, 分析了现有防御机制存在的问题。最后, 展望了边信道隐私推测攻击的发展和研究方向。

关键词: Android; 边信道攻击; 隐私推测; 移动安全

中图分类号: TP393

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2018034

Review of side-channel privacy inference of Android mobile devices

SHI Futian, MAO Jian, LIU Jianwei

School of Electronic & Information Engineering, Beihang University, Beijing 100191, China

Abstract: Android side channels may leak users' sensitive information. The existing Android side-channel attacks were analyzed and classified according to the type of the privacy-sensitive information with their inference models. The mainstream privacy inference approaches were analyzed and the deployed scenarios of these attacks were summarized. In addition, the mechanisms were discussed to detect or suppress the side-channel-based attacks and disclosed their limitations. Finally, the potential directions of research on side-channel-based privacy inference in Android systems were proposed.

Key words: Android, side-channel attack, privacy inference, mobile security

1 引言

边信道隐私推测是指利用计算设备、软件实现、信息传输等环节中无意间泄露的敏感信息, 推测用户隐私的攻击行为。基于该原理, Kocher^[1]于 1996 年提出了第一个攻击智能卡加密算法实现的边信道隐私推测攻击, 之后出现了众多通过时间、耗电量、电磁辐射等边信道信息攻击智能卡的研究工作^[2-4]。直至云计算时代, 边信道信息

泄露一直威胁着用户的信息安全。

随着智能手机、智能穿戴设备的普及, 智能移动设备存储、处理着大量隐私信息, 如用户密码、身份、位置等。在所有智能移动设备中, 运行 Android 操作系统的设备占比超过了 72%^[5]。然而, Android 系统的开放性使 Android 移动设备成为恶意应用攻击的主要对象。近年来, 机器学习进一步为边信道攻击的隐私推测能力提供了支持, 使基于边信道隐私推测的攻击愈发严重。因

收稿日期: 2018-02-02; 修回日期: 2018-03-20

通信作者: 毛剑, maojian@buaa.edu.cn

基金项目: 国家重点研发计划基金资助项目 (No.2017YFB0802400)

Foundation Item: The National Key R&D Program China (No.2017YFB0802400)

此, 深入研究 Android 移动设备的边信道隐私推测攻击对保护用户的信息安全具有重要意义。

目前, 针对 Android 移动设备边信道隐私推测的研究可以分为 3 个方面: 边信道攻击研究、边信道检测研究和边信道抑制研究。边信道攻击研究尝试发现潜在的边信道, 通过构建隐私推测场景判断潜在信道是否能够被实际利用以及如何利用。边信道检测研究尝试分析应用程序的代码和行为, 发现实施边信道攻击的恶意应用。边信道抑制研究尝试通过限制边信道泄露敏感信息的能力, 破坏边信道攻击的有效性。

2 Android 移动设备边信道分类分析

用户隐私信息是指可以用来识别、联系、定位、区分一个人的敏感信息^[6]。用户在使用 Android 移动设备的同时, 也将隐私信息暴露给了 Android 移动设备。其中包括直接静态存储在设备上的信息, 如账号、相册等; 此外, 还有诸如应用运行状态、传感器状态等隐私敏感(privacy sensitive)信息。对于直接存储在设备上的信息, Android 系统通过沙箱机制和权限机制进行了严格的访问控制。但对于隐私敏感信息(如应用程序运行状态等), Android 系统尚未进行细致的保护。应用程序通常可以访问这些隐私敏感信息, 且无需申请特定权限或只需要申请少量常规权限。攻击者利用这类未受保护的隐私敏感信息推断用户隐私, 即 Android 移动设备边信道隐私推测攻击。

在 Android 移动设备中, 用户隐私的关联对象有两种: 应用状态和设备状态。用户隐私与应用状态关联体现为: 用户个人(隐私)行为直接影响应用程序的某些运行状态。例如, 用户拍摄小视频时, 相关程序的 CPU 使用时间、占用内存大小等状态会显著增加等。用户隐私与设备状态关联体现为: 用户的个人(隐私)行为直接影响硬件设备的状态, 如用户跑步时引起加速度传感器的状态数据变化。边信道隐私推测则是利用应用状态和设备状态等与用户隐私有关联的敏感信息推测用户的隐私。如图 1 所示, 边信道隐私推测攻击流程可以分为两步: 1) 收集应用状态、设备状态等边信道泄露的敏感信息, 2) 利用边信道泄露的敏感信息推测用户隐私。

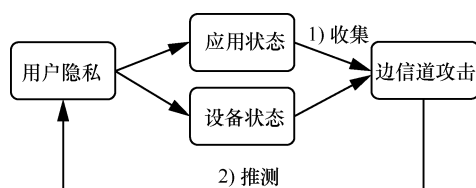


图 1 边信道隐私推测攻击流程

本文将 Android 移动设备中泄露隐私敏感信息的边信道按隐私关联对象分为应用状态边信道和设备状态边信道。

2.1 应用状态边信道攻击研究

Android 是基于 Linux 内核开发的移动操作系统, 它的安全模型也是建立在 Linux 内核安全模型的基础上^[7]。Android 系统对应用程序运行状态的记录既有 Linux 内核提供的 proc 文件系统(procfs, proc file system)和 sys 文件系统(sysfs, sys file system), 也有 Andorid 系统提供的应用程序接口(API, application programming interface)。本文按应用状态信息的来源将应用状态边信道信息来源分为 Linux 层公共文件资源边信道(简称 Linux 层边信道)和 Android 层应用程序接口边信道(简称 Android 层边信道)。

2.1.1 Linux 层公共文件资源边信道攻击研究

Linux 层边信道主要来自 Linux 内核的公共文件资源, 包括 procfs 和 sysfs。它们是 Android 继承 Linux 的虚拟文件系统, 用来记录系统和应用的运行状态。procfs 包含了应用进程和系统进程的运行时状态统计信息, 如进程的 CPU 使用时间、内存占用大小、网络使用统计等。sysfs 包含了内核子系统信息、设备信息、网络环境信息等。Android 应用程序接口没有提供应用程序的运行状态信息, Android 系统保留 Linux 内核的 procfs 和 sysfs 的初衷是支持应用程序开发者了解应用程序的运行状态。然而, 这些应用状态信息在提供有用信息的同时, 也成为边信道隐私推测攻击收集利用的对象, 给用户隐私带来了安全隐患。

利用 procfs 提供的应用内存使用情况推测用户的浏览记录是一种典型的边信道隐私推测攻击。用户浏览记录被认为是一种用户隐私(利用浏览记录, 攻击者可以了解用户的喜好、行为特点等信息), 因此浏览记录通常受到严密保护^[8]。应用进程的内存使用信息保存在 proc 文件系统的

/proc/<pid>/statm 文件中,主要数据包括虚拟内存页数(statm.size)、物理内存页数(statm.resident)、共享物理内存页数(statm.share)、数据段物理内存页数(statm.drs)等。文献[9]中指出,Android系统浏览器的内存变化与浏览器正在渲染的网页有关。研究人员利用浏览器进程的数据段物理内存页数(statm.drs)记录浏览器在渲染不同页面时的内存变化足迹,并以此区分不同的网页,实现了对用户网页浏览记录的推测攻击。文献[10]发现Android系统在进行用户界面切换时,应用进程和窗口组合进程(SurfaceFlinger)的共享内存会增大,变化量与应用进程图形缓存(GraphicBuffer)的大小有关,这是由应用进程和窗口组合进程进行进程间通信造成的。研究人员通过观测应用进程共享虚拟内存(statm.size-statm.drs)变化来检测Android应用的界面(UI, user interface)切换行为,之后结合/proc/<pid>/stat文件提供的CPU使用时间等信息推测出用户正在浏览的应用界面,实现对用户应用浏览记录的推测攻击。

利用sysfs提供的电池功耗信息推测用户隐私也是一种典型的边信道隐私推测攻击。Android移动设备的电池功耗信息保存在sys文件系统的/sys/class/power_supply/battery文件夹下,主要数据包括当前电池的工作电压(voltage_now)、工作电流(current_now)等。用户的不同行为导致不同的移动设备功耗,如用户录制小视频时的功耗比浏览网页时的功耗大。移动设备的功耗信息反映了应用的状态和用户的行为,是一种可利用的应用状态数据边信道。利用sysfs提供的电压、电流信息观测移动设备的功耗变化,可进行4种边信道隐私推测攻击^[11],其中包括:1)通过分析应用程序在启动阶段的功耗模式,识别出前端运行的应用程序;2)根据应用程序用户界面的功耗模式,实现对用户正在浏览的应用程序界面的推测;3)根据用户输入密码时的功耗剧烈变化次数,推测出用户输入的密码长度;4)通过分析移动设备在特定位置的功耗特征,实现对用户地理位置的推测。此外,另一种攻击则将移动设备的功耗与蜂窝信号强度关联,据此分析用户位置与基站的距离,实现对用户位置的推测^[12]。

2.1.2 Android层应用程序接口边信道攻击研究

Android层边信道主要来自Android系统提供的应用程序接口。除了Linux内核提供的公共文件资源,Android系统也提供了应用程序接口以帮助应用程序访问系统资源。其中,统计Android应用程序流量使用情况的android.net.TrafficStats和接收Android应用公共广播内容的android.content.BroadcastReceiver是2个典型的Android层边信道。

Android移动设备的网络数据使用情况保存在proc文件系统的/proc/uid_stat/<uid>/和sys文件系统的/sys/class/net/<interface>/statics/中,其中,前者以应用程序为统计单元,后者以网络通信接口为统计单元。应用程序通过TCP接收和发送网络数据分组时,Android将数据分组的长度值累计到应用程序的网络流量统计中。应用程序接收网络数据统计保存在/proc/uid_stat/<uid>/tcp_rcv文件中,发送网络数据统计保存在/proc/uid_stat/<uid>/tcp_snd文件中。同时,Android应用程序还可以通过公共API类android.net.TrafficStats访问网络流量统计数据。文献[13]设计了流量序列对比方法用于推测应用行为。该方法首先利用TrafficStats对目标应用程序行为建立网络流量指纹模型,再将实际监听到的流量序列与流量指纹对比,实现对特定应用行为的识别。研究人员发现,Twitter用户在发表推文时的POST数据分组大小在580~720 byte之间变化,这是因为推文长度在0~140 byte之间。攻击通过Twitter提供的API查找与受害者具有相同地点、相同推文发送时间、相同推文长度的用户ID,实现对用户身份的推测。此外,WebMD应用网络数据的不同使用反映了用户在该应用上的不同选择,而这些选择反映了用户所关心的疾病,由此可实施推测用户健康状况的边信道攻击。虽然安全套接字层(SSL, secure sockets layer)协议及传输层安全(TSL, transport layer security)协议对网络数据分组的加密隐藏了数据分组中数据内容的长度,但仍可以利用网络数据传输方向、数据分组大小等特征区分不同应用的网络行为模式,推测出用户安装的应用程序状态^[14]。

广播(broadcast)是Android核心组件,用于在不同应用之间传递数据或在同一应用的不同

组件之间传递数据。广播类似“发布-订阅”模式,应用程序可以通过 `sendBroadcast` 方法发布广播通知其他应用,也可以通过 `registerReceiver` 方法订阅指定的广播以接收其他应用的通知。例如,Android 系统在启动完成后会发送“ACTION_BOOT_COMPLETED”广播,注册此广播的应用就会收到该广播并执行 `BroadcastReceiver.onReceive` 方法。攻击者可利用 Android 系统的电池电量变化广播构建 2 个恶意应用(如 M-app1、M-app2)之间传递敏感信息的边信道^[15]。恶意应用 M-app1 将敏感信息编码,然后通过 CPU、屏幕等设备改变电池电量的变化速度;恶意应用 M-app2 则根据接收到的电量变化广播的时间间隔,推测出收到的敏感信息。文献[16]利用 Android 系统的音量变化广播实现了类似的边信道攻击。

2.2 设备状态边信道攻击研究

设备状态边信道主要指以传感器为代表的硬件边信道(Hardware 层边信道)。随着移动设备内置传感器种类增多、功能增强,越来越多的 Android 应用程序需要依赖传感器采集用户信息,如健康管理、路径导航等。同时,越来越多的用户隐私信息在 Android 移动设备中处理,这为边信道隐私推测攻击提供了潜在的攻击目标。目前,利用传感器的用户隐私推测攻击主要有屏幕输入推测和位置信息推测。

屏幕输入推测是指利用传感器数据推测用户在屏幕上输入的解锁密码、解锁图案、文本等隐私信息。此类攻击实施主要基于两点:1) Android 应用访问传感器数据不需要单独申请权限;2) 用户在使用屏幕上输入信息时会轻微地敲击屏幕、转动设备。现有的屏幕输入推测攻击主要如下。1) 通过用户在屏幕上输入的数字与加速度传感器之间存在关联,推测用户的解锁密码^[17],以及在此基础上的单个字母推测攻击^[18]。2) 利用智能手表的加速度传感器实现推测解锁密码的攻击^[19]。3) 结合加速度传感器和陀螺仪传感器进行推测文本攻击^[20]。4) 基于用户输入时设备轻微转动导致的光线传感器数据变化,实现解锁密码推测攻击^[21]。此外,也有研究利用话筒和前置摄像头推测用户输入,但利用这 2 个设备需要单独申请权限。

位置信息推测是基于用户随身携带移动设备这一假设进行的。对于用户位置,应用程序可以直接访问 GPS 传感器获取,但这需要单独申请位置信息访问权限。当没有权限时,应用可以通过边信道推测用户的位置。典型攻击包括:通过加速度传感器推测用户使用的交通工具^[22];利用气压传感器和地形地貌信息推测出用户的驾驶路线^[23];基于加速度传感器和陀螺仪传感器推测用户的驾驶路线^[24]等。

综上,对 Android 移动设备中泄露隐私敏感信息的边信道按隐私关联对象分为应用状态边信道和设备状态边信道。表 1 对这 2 类边信道的敏感信息来源和隐私推测目标进行了分析总结。

表 1 Android 移动设备边信道研究分析总结

类型	信息来源	推测目标	主要研究
应用状态边信道	Linux 层 (procfs, sysfs)	浏览记录	文献[9~11]
		地理位置	文献[11,12]
		密码长度	文献[11]
	Android 层 (Android API)	用户身份	文献[13]
		浏览记录	文献[13,14]
共谋应用消息		文献[15,16]	
设备状态边信道	Hardware 层 (传感器、摄像头、话筒等)	解锁密码、图案	文献[17,19,21]
		软键盘字符输入	文献[18,20]
		地理位置	文献[22~24]

3 边信道隐私推测方法研究分析

边信道隐私推测攻击通过边信道收集足够的隐私敏感信息后,通过提取敏感信息特征、设计隐私推测分类器展开隐私推测。隐私推测的输入是攻击应用从边信道收集的敏感信息,输出是用户的隐私信息。隐私推测针对不同的边信道和不同的隐私展开研究,主要解决 2 个问题:1) 从敏感信息中提取哪些特征;2) 运用什么方法推测用户隐私。提取特征需要根据实际的攻击场景具体研究。隐私推测运用的方法通常是典型的匹配、识别、分类算法,常用的方法有 Jaccard 指数(Jaccard index)、动态时间归整(DTW, dynamic time warping)、随机森林(random forest)、支持向量机(SVM, support vector machine)、隐马尔可夫模型(HMM, hidden markov model)、循环神经网络(RNN, recurrent neural network)。

Jaccard 指数^[25]又称为 Jaccard 相似系数,用于比较有限样本集之间的相似性与差异性。

Jaccard 指数值越大, 样本相似度越高。文献[9]实现了从网页的内存足迹推测网页的攻击。攻击首先构建网页内存足迹(memprint)数据库, 每个网页的内存足迹 m 表示为一组 (E, c) 。其中, E 代表内存大小, c 代表 E 的出现次数。攻击使用式 $J(m_1, m_2) = \frac{m_1 \cap m_2}{m_1 \cup m_2}$ 计算网页内存足迹 m_1 和 m_2

的 Jaccard 相似系数。然后, 分别计算待测目标网页的内存足迹与已知网页的内存足迹的 Jaccard 相似系数。最后判定相似系数大于阈值的已知网页是用户正在浏览的网页。

DTW^[26]用于孤立词识别, 该算法基于动态规划(DP, dynamic programming)的思想, 解决了发音长短不一的模板匹配问题, 是语音识别中出现较早、较为经典的一种算法。在边信道攻击研究中, DTW 被用来匹配时序数据, 如匹配用户输入不同数字时运动传感器的时序数据。文献[17]利用加速度传感器的 x 轴数据 v_x 和 y 轴数据 v_y , 提取某一时刻的加速度特征 $\arctan\left(\frac{v_y}{v_x}\right) \frac{180}{\pi}$, 随后

利用 DTW 查找加速度特征时序数据的最佳匹配。

随机森林和支持向量机是典型的机器学习分类器。随机森林^[27]是一个包含多个决策树的分类器, 且其输出类别由个别树输出类别的众数而定。支持向量机^[28]针对线性可分情况进行分析, 对于线性不可分的情况, 通过使用非线性映射算法将低维输入空间线性不可分的样本转化为高维特征空间使其线性可分。在边信道攻击中, 机器学习(machine learning)分类器可以利用边信道数据的统计特性推测用户隐私。文献[14]通过记录每组网络数据分组的上传流量、下载流量、总流量, 并计算这 3 组数据的最小值、最大值、平均值、中位数、标准差、方差、数据分组个数等统计特征, 最后利用随机森林、支持向量机等机器学习分类器对这些统计特征进行分类, 推测用户安装的应用程序。

HMM^[29]用来描述一个含有隐含未知参数的马尔可夫过程, 其从可观察的参数中确定该过程的隐含参数, 然后利用这些参数做进一步的分析, 如模式识别等。在边信道攻击研究中, HMM 被用来识别序列数据中隐藏的隐私信息。文献[30]实现了利用中断信息推测用户解锁图案的攻击。

攻击首先根据采集到的中断间隔推断出解锁图案中每条直线的长度得到长度序列, 再使用 Viterbi 算法计算出最可能的隐含状态序列, 即解锁图案。

RNN 也用来识别序列数据中隐藏的隐私信息。RNN^[31]是一种通过隐藏层节点周期性连接捕捉序列化数据中动态信息的神经网络, 可以对序列化的数据进行分类, 常用于手写识别、语音识别。文献[32]利用中断数据识别用户用手势输入方式输入的长文本。实验表明, 使用 SVM 对中断数据进行处理, 由于 SVM 以单词为单位进行识别, 不考虑上下文, 丢失了长文本的语义语境信息, 识别的准确率小于 10%。而 RNN 能自动学习训练集中的单词、语境, 通过训练 RNN 分类器, 可将长文本中的单词识别准确率达到 92%。

表 2 总结了边信道隐私推测方法的典型应用。

表 2 边信道隐私推测方法研究总结

方法	用途	典型应用
Jaccard index	签名相似度	文献[9]
DTW	时序数据匹配	文献[17]
machine learning	统计特征分类	文献[14]
HMM	序列数据推测	文献[30]
RNN	序列数据推测	文献[32]

4 边信道攻击检测机制研究分析

针对 Android 系统安全问题, 已有多种恶意应用、恶意攻击行为检测机制^[33]。虽然常规恶意应用检测机制并未针对边信道攻击, 但是由于边信道攻击应用也具有类似攻击特征, 常规恶意应用检测机制也常被集合用于边信道攻击检测。边信道攻击检测设计的通用分析机制主要有数据流分析检测和控制流分析检测。

4.1 数据流分析

数据流分析检测通过分析敏感信息在应用程序中的传播, 检测应用程序外传敏感信息的行为, 如通过网络套接字向外界发送敏感信息的行为等。数据流分析检测主要有静态数据流分析和动态污点跟踪^[33]。

静态数据流分析的主要原理是分析应用程序源码中变量的关系, 检测与敏感信息相关的代码。文献[34]提出 FlowDroid 静态数据流分析方法。该

方法利用反编译得到的可执行代码文件、布局文件以及配置文件,分析应用程序组件的生命周期和回调函数,生成函数调用图,并通过分析数据在函数调用图中的传播发现可疑的数据流。

动态污点跟踪技术的主要原理是使用污点标记应用程序中的敏感数据,随着程序的运行将污点传播至依赖于敏感数据的对象中。文献[35]提出的 TaintDroid 就是通过动态污点跟踪保护 Android 系统中的隐私数据。TaintDroid 将位置传感器、加速度传感器、话筒、相机、通信录、短信、设备识别码等敏感信息作为源污点,从指令层、应用间消息层、文件层等不同层跟踪污点在内存中的传播。当污点出现在与外界联系的接口(如网络通信接口等)中时,TaintDroid 就会发起敏感信息泄露警告。

4.2 控制流分析

控制流分析检测是通过分析应用程序的 API 调用及 API 调用的触发条件来获取应用程序行为的相关信息,以此判断应用程序是否存在边信道攻击行为。

DroidAPIMiner^[36]就是一种控制流分析方法。该方法首先分析恶意应用频繁调用的敏感 API,通过反编译 APK 文件提取应用程序的敏感 API 调用信息,最后通过训练基于 KNN (k-nearest neighbor)、SVM 等多种机器学习算法的分类器检测恶意应用。该方法的恶意应用检出率最高达到 97.8%。

MAMADROID^[37]也是一种基于控制流分析的检测机制。首先反编译 APK 文件,分析应用程序的 API 调用信息生成 API 调用图;根据 API 调用图提取 API 调用序列,进而构建 API 调用的马尔可夫链。得到的马尔可夫链有 340 个状态和 115 600 个可能的转化,接着以状态转化的概率作为特征向量,利用主成分分析对特征向量降维,通过随机森林、1-NN、3-NN、SVM 这 4 种分类算法对恶意应用进行分类检测。该方法的恶意应用检出率最高达到 99%。

4.3 边信道攻击应用检测机制

现有针对边信道攻击应用的检测机制的研究是在常规恶意应用检测基础上,增加对边信道特征的检测。这部分研究工作可以看作是常规恶意

应用检测机制的补丁。

常规恶意应用检测机制可以检测部分边信道攻击,但还有部分边信道攻击可以轻松绕过常规检测机制。例如,文献[38]提出了多种可以绕过 TaintDroid 污点跟踪检测的边信道攻击。这是由于 TaintDroid 可以跟踪隐私数据在内存中的传播,却不能跟踪数据在边信道中的传播。这类攻击通过 2 个攻击应用(如 M-app1、M-app2)共谋实现。应用 M-app1 将获取的敏感信息隐藏在文件元数据、剪切板等公共资源或程序休眠时间等应用行为中。应用 M-app2 通过观测这些公共资源或应用行为获取 M-app1 发送的敏感信息。由于这类数据流的终点是公共资源或应用行为,TaintDroid 无法检测。

边信道攻击应用检测研究多从修补常规恶意应用检测漏洞的角度出发。例如,针对文献[38]提出的边信道攻击,文献[39]在 TaintDroid 基础上构建了多种边信道的污点传播规则,实现了对边信道信息泄露的检测。该机制构建的污点传播规则有:当应用调用 sleep 函数时,将时间相关信息作为污点;污点传播到公共资源,则将访问公共资源的函数作为污点等。

5 边信道攻击抑制方法研究分析

针对 Android 边信道隐私推测攻击的检测研究,提高了恶意应用利用边信道收集敏感信息的难度,增强了对 Android 系统中用户隐私的保护。然而,一旦恶意攻击应用绕过了检测机制的防御,就仍然可以发起边信道攻击。为全面保护 Android 用户的信息安全,有必要对边信道进行抑制,降低其泄露信息的能力。目前,抑制边信道信息泄露的研究工作可以分为权限控制、干扰信源、限制采样、限制运行这 4 个研究方向。

权限控制指在 Android 权限模型基础上加强对敏感信息的访问控制。通过细粒度的权限模型保护被边信道攻击所收集利用的敏感信息是最直接有效的抑制方法。文献[40]针对利用传感器的信息泄露提出了一种细粒度的传感器访问权限控制方法。Android 应用程序访问硬件层传感器设备需要经过 Application 层、Framework 层、Library 层、Kernel 层和 Hardware 层。该方法通过在

Framework 层增加权限检查实现了对传感器访问权限的细粒度控制。

干扰信源指通过干扰边信道收集敏感信息,降低隐私推测攻击的准确性。典型的干扰信源的方式是注入噪声。文献[41]提出一种针对通过传感器推测屏幕输入内容攻击的限制方法。该方法在用户打开软键盘输入内容时,向/dev/input 文件夹下的加速度传感器和陀螺仪传感器相应的文件以 7~12 ms 的随机时间间隔注入随机值,实验结果证明,该方法能够显著降低文本输入推测的准确性。文献[42]提出一种针对利用 procfs 提供的中断信息推测用户解锁图案攻击的限制方法。该方法通过修改内核实现向/dev/input 注入假中断,修改 Widget 组件的实现代码设置隐藏窗口以接收假中断,既保证了应用窗口能够正常接收中断,又干扰了隐私推测攻击。文献[18]提出打乱软键盘字符位置的方法,研究表明该方法可以消除输入推测攻击的安全隐患,代价是降低了软键盘的可用性。

限制采样指通过限制应用程序访问敏感信息或降低应用程序对敏感信息采样频率的方式,降低隐私推测攻击的准确性。文献[43]提出一种 Android 系统音频资源管理方案,该方案可以在用户通话时限制第三方应用访问麦克风,从而抑制了利用用户通话音频推测用户隐私的攻击。文献[9]在利用浏览器进程的内存足迹推测用户的浏览记录时,对比了不同内存信息采样率下的推测准确性。实验结果表明,对应用程序访问 procfs 的频率进行限制能够显著降低隐私推测的准确性。

限制运行指通过查杀恶意应用进程的方式保护敏感信息。用户在使用 Android 移动设备时产生的敏感信息通过边信道被恶意应用收集,恶意应用收集敏感信息的行为在 Android 后台隐蔽进行。文献[44]提出了一种边信道抑制方法,该方法在暴露隐私的前台应用运行时,中止信息收集应用进程;在前台应用运行结束后,启动信息收集应用。该方法要中止信息收集应用运行,需要申请“KILL_BACKGROUND_PROCESSES”权限。

抑制边信道收集隐私敏感信息的主要方法、应用细节和典型工作总结如表 3 所示。

表 3 Android 边信道抑制方法研究总结

方法	细节	典型工作
权限控制	敏感信息细粒度权限控制	文献[40]
干扰信源	注入噪声、随机软键盘	文献[18,41,42]
限制采样	限制敏感信息访问频率	文献[43]
限制运行	查杀后台数据收集应用	文献[44]

6 存在的问题与研究展望

6.1 边信道攻击检测和抑制机制存在的问题

Android 移动设备存储、处理着大量用户隐私,边信道隐私推测攻击通过收集边信道泄露的隐私敏感信息推测用户隐私,给 Android 用户的信息安全带来了严重的威胁。目前已有很多针对 Android 边信道隐私推测攻击的检测和抑制研究,这些研究对防御边信道攻击、保护用户隐私起到了重要作用。但仍有一些不足之处,具体如下。

1) 针对特定边信道攻击带来的问题

目前,边信道攻击检测抑制机制多针对特定边信道攻击,要想防御所有的边信道攻击,需要研究应用多种边信道攻击检测抑制机制。这带来 3 个问题:①只对现有特定的边信道有效,不能广泛防御未知的边信道攻击;②需要研究验证多种检测抑制机制的有效性;③部署应用大量检测抑制机制使运行负载增加。

在最近的研究中,文献[44]提出了能检测抑制多种收集敏感信息的边信道攻击方案 App-Guardian,但该方案也存在问题:①该方案通过 CPU 调度频率检测边信道攻击应用的准确性有待进一步验证;②该方案通过查杀应用进程的方式抑制边信道收集敏感信息,存在导致误杀正常应用的可能性。

2) 检测抑制机制实际应用过程中的问题

目前,检测抑制机制实际应用有 2 种方式:一种是安装第三方安全应用;另一种是改进 Android 系统的安全机制。第三方安全应用的主要问题是运行负载。随着边信道攻击的增多,安全应用需要多种边信道抑制检测机制,安全应用的运行负载也随之增大。改进 Android 系统安全机制的典型做法是对隐私敏感信息增加细粒度权限控制。例如,Android M 细化了应用访问 procfs 的权限,使应用不能再访问 procfs 中其他进程的

相关文件。然而, 由于可利用的边信道仍在不断被发现, 对每个被利用的边信道都增加细粒度的权限管理很难实现。也有研究指出^[45], 由于有的用户不理解特定权限的实际意义, 有的用户不关心应用请求了哪些权限, 导致通过权限机制也很难确保所有 Android 用户的信息安全。

6.2 边信道隐私推测攻击的发展趋势

Android 移动设备边信道攻击的检测抑制研究不断完善增加了边信道隐私推测的难度。同时, 物联网和智能家居的迅速发展也为边信道隐私推测提供了更多的隐私敏感信息。在这种背景下, 边信道隐私推测攻击未来可能的发展趋势如下。

1) 组合边信道攻击

组合多个边信道可以使隐私推测在更多敏感信息的基础上进行, 一方面可以提高已有边信道攻击隐私推测的准确率, 另一方面可以通过组合多种敏感信息推测新的隐私信息。在最近的研究中, 文献[46]组合缓存信息和传感器状态信息, 用以提高推测用户输入内容的准确性; 文献[47]组合用户浏览网页时的流量统计信息和输入用户名、密码时的传感器状态信息, 提出了可以推测用户网站登录凭证的边信道攻击。

2) 智能家居边信道攻击

智能家居中的智能设备都通过网络互联和访问, 从边信道攻击的角度来看, 这些智能设备可以作为边信道用来收集敏感信息, 这增强了 Android 边信道获取隐私敏感信息的能力。随着攻击者对这些智能家居设备利用潜力的挖掘, 边信道隐私推测攻击在智能家居环境下将会更容易实现。事实上, 文献[44]已经提出了一种智能家居边信道攻击, 该文利用智能家居的 Wi-Fi 摄像头配合 Android 系统的应用流量边信道, 推测用户是否在家。文献[48]则提出通过对睡眠监视器的流量分析推测用户的睡眠状态。

7 结束语

隐私信息泄露是目前 Android 移动设备最严重的安全威胁之一。本文对 Android 边信道隐私推测攻击进行了系统研究, 主要从典型敏感信息边信道、常用隐私关联推测方法、边信道攻击应用检测、边信道信息收集抑制这 4 个角度展开。

本文按隐私关联对象对 Android 移动设备边信道进行了分类分析, 总结了边信道隐私推测中用到的匹配、识别、分类算法, 分析了现有边信道攻击检测抑制机制的主要方法和缺陷, 分析了边信道隐私推测攻击的发展趋势。

参考文献:

- [1] KOCHER P C. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems[C]//Annual International Cryptology Conference. 1996: 104-113.
- [2] KOCHER P, JAFFE J, JUN B. Differential power analysis[C]//Annual International Cryptology Conference. 1999: 388-397.
- [3] QUISQUATER J J, SAMYDE D. Electromagnetic analysis (ema): measures and counter-measures for smart cards[M]//Smart Card Programming and Security. Berlin: Springer. 2001: 200-210.
- [4] MANGARD S, OSWALD E, POPP T. Power analysis attacks: revealing the secrets of smart cards[M]. Springer Science & Business Media, 2008.
- [5] 国家计算机网络应急技术处理协调中心. 2017 年第四季度国内操作系统及浏览器占比情况分析[R]. 2018.
- [6] CNCERT. Analysis of the occupation ratio of domestic operating system and browser in the fourth quarter of 2017[R]. 2018.
- [7] THURM S, KANE Y I. Your Apps are watching you[J]. The Wall Street Journal, 2010, 17(1).
- [8] 张玉清, 王凯, 杨欢, 等. Android 安全综述[J]. 计算机研究与发展, 2014, 51(7): 1385-1396.
- [9] ZHANG Y Q, WANG K, YANG H, et al. Survey of Android OS security[J]. Journal of Computer Research and Development, 2014, 51(7): 1385-1396.
- [10] MAO J, CHEN Y, SHI F, et al. Toward exposing timing-based probing attacks in Web applications[J]. Sensors, 2017, 17(3): 464.
- [11] JANA S, SHMATIKOV V. Memento: learning secrets from process footprints[C]//2012 IEEE Symposium on Security and Privacy (SP). 2012: 143-157.
- [12] CHEN Q A, QIAN Z, MAO Z M. Peeking into your App without actually seeing it: UI state inference and novel Android attacks[C]//USENIX Security Symposium. 2014: 1037-1052.
- [13] YAN L, GUO Y, CHEN X, et al. A study on power side channels on mobile devices[C]//The 7th Asia-Pacific Symposium on Internetwork. 2015: 30-38.
- [14] MICHALEVSKY Y, SCHULMAN A, VEERAPANDIAN G A, et al. PowerSpy: location tracking using mobile device power analysis[C]//USENIX Security Symposium. 2015: 785-800.
- [15] ZHOU X, DEMETRIOU S, HE D, et al. Identity, location, disease and more: inferring your secrets from android public resources[C]//The 2013 ACM SIGSAC Conference on Computer & Communications Security. 2013: 1017-1028.

- [14] TAYLOR V F, SPOLAOR R, CONTI M, et al. Robust smartphone app identification via encrypted network traffic analysis[J]. IEEE Transactions on Information Forensics and Security, 2018, 13(1): 63-78.
- [15] CHANDRA S, LIN Z, KUNDU A, et al. Towards a systematic study of the covert channel attacks in smartphones[C]// International Conference on Security and Privacy in Communication Systems. 2014:427-435.
- [16] MARFORIO C, RITZDORF H, CAPKUN S. Analysis of the communication between colluding applications on modern smartphones[C]//Computer Security Applications Conference. 2012:51-60.
- [17] CAI L, CHEN H. TouchLogger: inferring keystrokes on touch screen from smartphone motion[C]//Usenix Conference on Hot Topics in Security. 2011:9-9.
- [18] OWUSU E, HAN J, DAS S, et al. ACCessory: password inference using accelerometers on smartphones[C]//The Twelfth Workshop on Mobile Computing Systems & Applications. ACM, 2012: 9.
- [19] 宋晨光, 刘建伟, 伍前红, 等. 基于智能手表运动传感器的新型攻击及其防范[J]. 通信学报, 2015, 36(Z1): 235-242.
- SONG C G, LIU J W, WU Q H, et al. New attack based on smartwatch motion sensors and the protection method research[J]. Journal on Communications, 2015, 36(Z1): 235-242.
- [20] PING D, SUN X, MAO B. Textlogger: inferring longer inputs on touch screen using motion sensors[C]//The 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks. 2015: 24.
- [21] SPREITZER R. Pin skimming: exploiting the ambient-light sensor in mobile devices[C]//The 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices. 2014: 51-62.
- [22] HEMMINKI S, NURMI P, TARKOMA S. Accelerometer-based transportation mode detection on smartphones[C]//The 11th ACM Conference on Embedded Networked Sensor Systems. 2013: 13.
- [23] HO B J, MARTIN P, SWAMINATHAN P, et al. From pressure to path: barometer-based vehicle tracking[C]//The 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments. 2015: 65-74.
- [24] NARAIN S, VOHUU T D, BLOCK K, et al. Inferring user routes and locations using zero-permission mobile sensors[C]//Security and Privacy. 2016:397-413.
- [25] HAMERS L. Similarity measures in scientometric research: the Jaccard index versus Salton's cosine formula[J]. Information Processing and Management, 1989, 25(3): 315-18.
- [26] MÜLLER M. Dynamic time warping[J]. Information Retrieval For Music and Motion, 2007: 69-84.
- [27] LIAW A, WIENER M. Classification and regression by random-forest[J]. Rnews, 2002, 2(3): 18-22.
- [28] SUYKENS J A K, VANDEWALLE J. Least squares support vector machine classifiers[J]. Neural Processing Letters, 1999, 9(3): 293-300.
- [29] EDDY S R. Hidden markov models[J]. Current Opinion in Structural Biology, 1996, 6(3): 361-365.
- [30] DIAO W, LIU X, LI Z, et al. No pardon for the interruption: new inference attacks on android through interrupt timing analysis[C]// Security and Privacy. IEEE, 2016:414-432.
- [31] MIKOLOV T, KARAFIÁT M, BURGET L, et al. Recurrent neural network based language model[C]//The 11th Annual Conference of the International Speech Communication Association. 2010.
- [32] SIMON L, XU W, ANDERSON R. Don't interrupt me while i type: inferring text entered through gesture typing on Android keyboards[J]. Proceedings on Privacy Enhancing Technologies, 2016(3):136-154.
- [33] 卿斯汉.Android 安全研究进展[J]. 软件学报, 2016, 27(1):45-71.
- QING S H. Research progress on Android Security[J]. Journal of Software, 2016, 27(1):45-71.
- [34] ARZT S, RASTHOFFER S, FRITZ C, et al. FlowDroid: precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for Android apps[C]//ACM Sigplan Conference on Programming Language Design and Implementation. 2014:259-269.
- [35] ENCK W, GILBERT P, CHUN B G, et al. TaintDroid: an information flow tracking system for real-time privacy monitoring on smartphones[J]. ACM Transactions on Computer Systems, 2010, 32(2):1-29.
- [36] AAFER Y, DU W, YIN H. DroidAPIMiner: mining API-level features for robust malware detection in Android[C]// International Conference on Security and Privacy in Communication Systems. 2013:86-103.
- [37] MARICONTI E, ONWUZURIKE L, ANDRIOTIS P, et al. MamaDroid: detecting android malware by building Markov chains of behavioral models[C]//Network and Distributed Systems Security Symposium(NDSS'17). 2017.
- [38] BABIL G S, MEHANI O, BORELI R, et al. On the effectiveness of dynamic taint analysis for protecting against private information leaks on Android-based devices[C]// IEEE International Conference on Security and Cryptography. 2015:1-8.
- [39] GRAA M, CUPPENS-BOULAHIA N, CUPPENS F, et al. Detection of side channel attacks based on data tainting in Android systems[C]//IFIP International Conference on ICT Systems Security and Privacy Protection. 2017: 205-218.
- [40] 吴敬征, 武延军, 罗天悦, 等. 一种基于权限控制机制的 Android 系统隐蔽信道限制方法[J]. 中国科学院大学学报, 2015, 32(5): 667-675.
- WU J Z, WU Y J, LUO T Y, et al. A new mitigation approach for covert channel of Android operating system based on permission mechanism[J]. Journal of University of Chinese Academy of Sci-

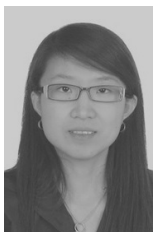
- ences, 2015, 32(5): 667-675.
- [41] SHRESTHA P, MOHAMED M, SAXENA N. Slogger: smashing motion-based touchstroke logging with transparent system noise[C]//ACM Conference on Security & Privacy in Wireless and Mobile Networks. 2016:67-77.
- [42] SCHWARZ M, LIPP M, GRUSS D, et al. KeyDrown: eliminating software-based keystroke timing side-channel attacks[C]// Network and Distributed System Security Symposium. 2018.
- [43] PETRACCA G, SUN Y, JAEGER T, et al. Android: preventing attacks on audio channels in mobile devices[C]// Computer Security Applications Conference. 2015:181-190.
- [44] ZHANG N, YUAN K, NAVEED M, et al. Leave me alone: app-level protection against runtime information gathering on Android[C]//IEEE Symposium on Security and Privacy. 2015: 915-930.
- [45] FELT A P, HA E, EGELMAN S, et al. Android permissions: user attention, comprehension, and behavior[C]//The Eighth Symposium on Usable Privacy and Security. 2012: 3.
- [46] LIPP M, GRUSS D, SPREITZER R, et al. ARMageddon: cache attacks on mobile devices[C]//USENIX Security Symposium. 2016: 549-564.
- [47] SPREITZER R, GRIESMAYR S, KORAK T, et al. Exploiting data-usage statistics for website fingerprinting attacks on Android[C]//The 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks. 2016: 49-60.
- [48] APHORPE N, REISMAN D, FEAMSTER N. A smart home is no

castle: privacy vulnerabilities of encrypted IoT traffic[C]// Workshop on Data and Algorithmic Transparency. 2016.

[作者简介]



史福田 (1991-), 男, 山东青岛人, 北京航空航天大学硕士生, 主要研究方向为移动应用安全。



毛剑 (1978-), 女, 江苏溧水人, 博士, 北京航空航天大学讲师, 主要研究方向为 Web 安全、移动终端安全、大数据安全与隐私保护。



刘建伟 (1964-), 男, 山东莱州人, 博士, 北京航空航天大学教授、博士生导师, 主要研究方向为密码学、网络安全。