Computer Science and Application 计算机科学与应用, 2021, 11(5), 1474-1483
Published Online May 2021 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2021.115151

Hans汉斯

基于攻击的安卓应用安全检测及评估标准研究

喻毫博，甘　刚

成都信息工程大学网络空间安全学院，四川　成都
Email: hong9741@163.com, test_me@cuit.edu.cn

收稿日期：2021年4月25日；录用日期：2021年5月20日；发布日期：2021年5月27日

摘　要

随着手机的发展，开源的安卓系统也慢慢填补了自己的劣势，在市场占有上大大超越了苹果手机。同样因为安卓的开源性，各种鱼龙混杂的安卓应用都进入了安卓市场，所以安卓应用安全性研究就成了近年来的重中之重。本文就现阶段安卓应用安全检测需求进行深度剖析，从攻击者的角度出发，结合现有检测指标，设计出详细全面的安卓应用安全检测评估指标。针对部分评估指标，给出相应的测试内容。按照当前主流的安全检测指标权重，计算出一个简单的线性检测评估模型。此模型可以有效发现安卓应用在不同评估指标下的安全强度，以及安卓应用所面对的安全风险。

关键词

移动安全，安卓应用，安全检测，评估

Research on Security Detection and Evaluation Criteria of Android Application Based on Attack

Haobo Yu, Gang Gan

School of Cybersecurity, Chengdu University of Information and Technology, Chengdu Sichuan
Email: hong9741@163.com, test_me@cuit.edu.cn

Received: Apr. 25th, 2021; accepted: May 20th, 2021; published: May 27th, 2021

Abstract

With the development of mobile phones, the open source Android system has gradually filled its disadvantage and greatly surpassed Apple's mobile phone in market share. Also because of the

文章引用: 喻毫博, 甘刚. 基于攻击的安卓应用安全检测及评估标准研究[J]. 计算机科学与应用, 2021, 11(5): 1474-1483. DOI: 10.12677/csa.2021.115151

open source nature of Android, a variety of mixed Android applications have entered the Android market, so Android application security research has become a top priority in recent years. This paper analyzed the current Android application security detection requirements in depth, and from the perspective of attackers, combined with the existing detection indicators, designed detailed and comprehensive Android application security detection evaluation indicators. According to some evaluation indexes, the corresponding test contents were given. According to the current mainstream safety detection index weight, a simple linear detection evaluation model was calculated. This model can effectively discover the security strength of Android applications under different evaluation indicators, as well as the security risks Android applications face.

Keywords

Mobile Security, Android Applications, Security Detection, Evaluation

1. 引言

安卓作为 Linux 为内核的开源操作系统，因为其开放性，受到大量开发人员的追捧，同时因为开发门槛低，安卓迅速占领市场。就权威调研机构 IDC 公布的《中国 2019 第三季度智能手机出货量》[1]报告来看，国产安卓手机已经大量占据市场。因此，安卓应用的安全性问题就显得尤为重要。在腾讯安全国际技术峰会上会发布每年的各种最新技术白皮书，其中腾讯科恩实验室发布的《Android 应用安全白皮书》表示，安卓应用目前存在很高的风险，98%的应用都不安全。ApkPecker 作为权威的安卓应用自动化漏洞扫描系统，也公布了他们的发现。安卓应用主要是在场景漏洞利用、前后台漏洞等部分存在安全隐患。其中，《白皮书》在通过大样本分析后发现，检测样本中的安卓应用大多缺乏用户信息保密机制，这给移动应用带来了更大的安全风险。因为用户信息保密机制的缺乏带来的安全事件给用户的信息和资金带来了巨大危害。分析主要原因，还是因为在开发中出现了隐患，后期的监测和修复又跟不上等。安卓应用的安全面临着极大的考验。Android 系统由于其开源的属性，市场上针对开源代码定制的 ROM 参差不齐，在系统层面的安全防范和易损性都不一样，安卓应用的发布审核也比较宽松，导致漏洞较多。虽然目前市场上的安卓应用都有一些安全防范意识，但是并不完善，对安全的重视程度不够；而且由于开发安全并不属于开发的范畴中，很多开发人员并没有足够的安全技术，防范措施很有限制。为了用户的安全，对安卓应用的安全性检测至关重要。因此急需一套合理有效的安全检测评估体系，此标准可以对未上线的 app 进行全面的安全评估，加强 app 上线后的安全性，并从多方面杜绝攻击。此研究对保障安卓应用市场安全，提升开发人员对安全性的重视具有重要意义。

2. 安卓应用安全检测研究现状

当前常见的安全检测技术，分别基于动态检测和静态检测[1]，动态检测方面有基于应用程序的行为特征进行的，提出的方法很多[2] [3]，比较有特点的是重庆师范大学刘玮等人[4]提出一个通用的软件检测框架。这个检测框架通过安卓逆向技术从安卓应用中获取各项安全信息的特征，并建立了特征信息库。在此基础上，通过机器学习来建立了检测模型，通过分类检测的方式来进行检测。这个检测框架让软件在安装前能够进行较为正确的安全检测，且扩展性较高。动态数据安全方面王喆[5]分析了移动终端数据

存储和传输两个重要环节中的安全防护要素，并基于此研究提出了一种安卓移动终端数据安全检测评价方法。静态检测方面，则是对 APK 源码进行反编译，再对比敏感代码进行分析[6] [7] [8]，由此来判断被检测应用的安全性，相比动态检测更加直接有效率[9]。Zhejun Fang 等人[10]设计了一种针对安卓组件间通讯漏洞的静态检测进行了研究，该研究提出了一种旨在检测 Android 应用程序中输入验证漏洞的新颖方法，并实现了一个名为 IVDroid 的原型，该原型提供了对 Java 源代码的实用静态分析，通过重复验证行为挖掘来检测未知模式的缺陷。

但是，当前存在的安全检测技术都存在两个问题，第一都是从单方面的检测，并没有结合安全机制，恶意代码检测[11]和安全加固[12]的全面的检测，第二是缺乏一个相对完整的综合性的评估指标体系和模型。而在各大厂商提供的安卓安全检测系统中，只是对相应的指标进行了检测，一方面检测内容不一定完善，另一方面检测后不知道问题出在哪。要想知道怎么让 app 更安全，就必须从攻击者的角度出发看问题。所以不管是学术研究还是工业检测方面，都缺少一个从攻击角度入手的相对完整且能为直观看出 app 安全问题的安全检测评估模型及系统。本文将结合安卓动态检测和静态检测，结合安卓安全机制从攻击角度入手，展开对安卓应用安全检测评估方法的研究，分析各项指标性能，量化指标，构建一个直观且相对完善的评估模型，并实现安卓应用安全检测评估系统。

3. 基于攻击的 Android 应用安全检测评估方案设计

3.1. Android 应用安全检测和评估指标

针对安卓应用，从攻击者的角度出发，通常从以下 6 个方向来进行检测：反编译检测(Against-Compiling)，反篡改检测(Against-Tamper)，漏洞检测(Vulnerability)，组件安全检测(Components-Security)，数据安全检测(Data-Security)和敏感操作检测(Sensitive-Operations)，后文中为便捷使用，将 AC 表示为反编译检测，将 AT 表示为反篡改检测，将 VB 表示为漏洞检测，将 CS 表示为组件安全检测，将 DS 表示为数据安全检测，将 SO 表示为敏感操作检测。

反编译检测主要针对安卓应用加固方面的检测，检测内容如表 1 所示。

Table 1. Decompile the checkpoint list of detection indicators
表 1. 反编译检测指标检测点表

反编译检测指标项	主要检测点
dex 文件加固(AC1)	dex 文件是否加密
so 文件加固和安全检测(AC2)	so 文件加密或隐藏
混淆率检测(AC3)	代码混淆率
反编译检测指标项	主要检测点

反篡改检测主要检测是否有攻击者能够利用破解工具或其他技术将安卓应用的代码或资源文件篡改，检测 manifset 文件中的设置是否安全。检测内容如表 2 所示。

Table 2. Anti-tamper detection index detection points list
表 2. 反篡改检测指标检测点表

反篡改检测指标项	主要检测点
资源文件加固检测(AT1)	资源文件是否加密
文件信息检测(AT2)	文件信息是否容易篡改
证书检测(AT3)	证书是否安全且一致

Continued

manifest 文件检测(AT4)		PermissionGroup 项检测
		系统权限使用检测
		sharedUserId 检测
		allowBackup 标志检测
		Debuggable 配置检测
		非必要权限检测
		最低支持版本检测

漏洞检测一方面检测常见的系统漏洞，另一方面在服务端，分为 http 协议和 tcp/udp 协议来检测容易出现的漏洞。如表 3 所示。

Table 3. List of detection points of vulnerability detection indicators
表 3. 漏洞检测指标检测点表

漏洞检测指标项		主要检测点
系统漏洞检测(VB1)		fragment 注入漏洞检测
		sqlite 数据库日志泄露漏洞检测
		随机数生成漏洞检测
服务端漏洞检测(VB2)	http 协议	水平权限风险检测
		垂直权限风险检测
		SQL 注入漏洞检测
		XSS 漏洞检测
		敏感信息检测
	tcp/udp 协议	敏感信息检测
		Html5 漏洞检测
provider 注入漏洞检测(VB3)		provider 注入漏洞检测

组件安全分四方面，一是检测第三方库和 SDK 库是否安全。二是检测安卓应用本身所带的组件，扫描各种漏洞，检测组件信息等。三是针对 Webview 组件检测各类漏洞。最后是对 SQlite 组件检测加密情况和安全性。如表 4 所示。

Table 4. List of component safety detection index detection points
表 4. 组件安全检测指标检测点表

反编译检测指标项	主要检测点
第三方库和 SDK 组件检测(CS1)	第三方组件和 SDK 库安全检测
组件安全检测指标(CS2)	4 大组件导出检测
	ContentProvider 目录遍历漏洞检测
	Implicit Service 漏洞检测
	grant-uri-permission 属性检测
	Intent-Based 攻击检测
	Intent Scheme URI 漏洞攻击检测
	应用本地拒绝服务漏洞检测
	manifest 中定义组件未实现检测
	Debug 或 Test 敏感测试组件泄露检测
	Intent 不安全反射风险检测

Continued

webview 组件安全检测(CS3)	远程执行漏洞检测 潜在 XSS 攻击检测 本地文件访问漏洞检测 密码明文存储漏洞检测 主机名弱校验检测 证书弱校验检测 中间人攻击漏洞检测 不校验证书漏洞检测 组件系统隐藏接口未移除漏洞
sqlite 安全检测(CS4)	SQLite 数据库加密(SQLCipher)检测 SQLite 数据库(SQLite Encryption Extension (SEE))检测 SQLite 数据库的对称密钥(PRAGMA key)检测 SQLiteDatabase Transaction Deprecated 检测 Databases 任意读写漏洞检测

数据安全检测项从数据安全的角度出发，分为网络通信安全，弱加密风险，一般数据安全和 Hook 技术检测。网络通信安全检测网络通信中可能出现的风险及漏洞。弱加密风险检测各处运用加密的部分是否正常使用了加密或是否使用了弱加密。一般数据安全检测针对可能出现数据泄露的地方检测泄露风险。Hook 安全主要检测各信息接收或发送部件的数据信息是否正常。如表 5 所示。

Table 5. Table of data security detection index detection points
表 5. 数据安全检测指标检测点表

数据安全检测指标项	主要检测点
网络通信安全检测(DS1)	SSL 连接检测 SSL 组件安全检测 Host 检测 HttpURLConnection 漏洞检测 网络端口开放威胁检测
弱加密风险检测(DS2)	弱加密算法风险检测 不安全密钥长度检测 ECB 弱加密风险检测 不安全初始化向量检测 RSA 中 Padding 风险检测 是否使用密码保护检测
数据安全检测(DS3)	敏感数据检测 剪切板敏感信息泄露风险检测 Intent 组件数据泄露风险检测 PendingIntent 误用风险检测 密钥硬编码风险检测 程序和数据加载检测 BASE64 安全检测 全局文件读写漏洞检测 日志泄露风险检测

Continued

数据安全检测(DS3)	外部加载 Dex 检测
	外部存储路径检测
	明文证书风险检测
	第三方 oAuth 风险检测
hook 技术检测(DS4)	SQlite 存储信息检测
	SharePreferences 存储信息检测
	Contert Provider 存储信息检测
	File 存储信息检测
	Intent 数据内容检测
	Log 日志打印检测
	System.print*输出检测
	剪贴板敏感信息检测

最后敏感操作检测，主要是从静态代码检测和 DDOS 攻击两方面检测。静态代码主要检测应用的权限信息，代码中各类敏感函数的使用情况，在这个方面并没有使用现在常用的机器学习方式进行检测，只是检测了常见的敏感函数类型。详情如表 6 所示。

Table 6. List of sensitive operation detection indicators
表 6. 敏感操作检测指标检测点表

敏感操作检测指标项	主要检测点
权限信息检测(SO1)	应用权限检测
敏感函数调用检测(SO2)	安全相关函数检测
	安全相关类检测
	运行命令检测
	Native Library 加载检测
	外部动态加载 Dex 检测
	Root 代码检测
	获取 IMEI 和 Device ID 敏感信息代码检测
	获取 Android ID 敏感信息代码检测
	发送 SMS 敏感代码检测
	文件删除代码检测
	signature 代码检测
DDOS 攻击检测(SO3)	组件 DDOS 攻击检测
	Native crash 检测
	Java crash 检测

3.2. Android 应用安全评估模型权重

在安全检测指标的基础上，通过层次分析法，利用当前常用的安卓安全检测权值进行定量计算，所有评估指标权值加起来为结果为 1，最终检测得到的结构越接近 1 说明这个安卓应用的安全性能越高。根据主流安全检测权重指标和实验总结后，得出 Android 应用安全检测评估模型指标权重。如表 7 所示：

Table 7. Attack-based Android application security detection evaluation index comprehensive weight table
表 7. 基于攻击的安卓应用安全检测评估指标综合权重表

主准则	权重	次准则	综合权重
反编译检测(AC)	0.1052	dex 文件加固(AC1)	0.0496
		so 文件加固和安全检测(AC2)	0.0233
		混淆率检测(AC3)	0.0323
反篡改检测(AT)	0.3074	资源文件加固检测(AT1)	0.0354
		文件信息检测(AT2)	0.0542
		证书检测(AT3)	0.0719
		manifest 文件检测(AT4)	0.1459
漏洞检测(VB)	0.2188	系统漏洞检测(VB1)	0.1118
		服务端漏洞检测(VB2)	0.0759
		provider 注入漏洞检测(VB3)	0.0311
组件安全检测(CS)	0.1052	第三方库和 SDK 组件检测(CS1)	0.0171
		组件安全检测指标(CS2)	0.0386
		webview 组件安全检测(CS3)	0.0329
		sqlite 安全检测(CS4)	0.0166
数据安全检测(DS)	0.1582	网络通信安全检测(DS1)	0.0389
		弱加密风险检测(DS2)	0.0206
		数据安全检测(DS3)	0.0683
		hook 技术检测(DS4)	0.0304
敏感操作检测(SO)	0.1052	权限信息检测(SO1)	0.0618
		敏感函数调用检测(SO2)	0.0301
		DDOS 攻击检测(SO3)	0.0133

4. 实验分析

4.1. 实验内容

本文首先从实验室安卓应用数据库和网络上收集测试用安卓应用，本次采取的应用都是未进行加固或系统能去加固的应用，因此可能大体样本在加固检测方面得分较低。在收集到的安卓应用中，将应用分为游戏类应用，金融类应用和其他类应用，分别选取 40 个作为实验检测评估对象，利用本文实现的安卓应用安全检测评估系统，在安卓应用安全检测评估指标准则的基础上对安卓应用进行安全测试，根据测试情况给出评估结果。根据实验测试统计，不同类别的安卓应用评价得分趋势图如图 1 所示。

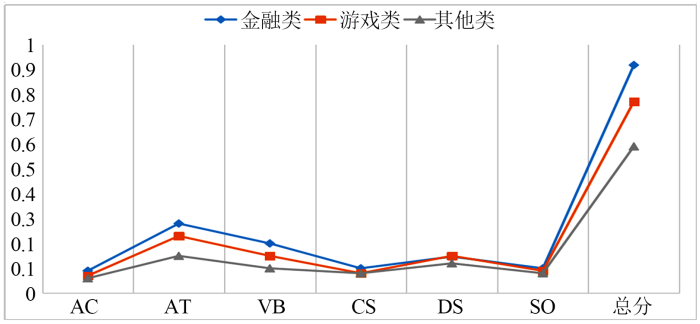

Figure 1. Android application security assessment score trend chart
图 1. 安卓应用安全检测评估得分趋势图

由于实验中选取的测试样本较多，本文选取不同类别下典型的安卓应用作为案例进行对比，作为代表的典型安卓应用可以反映出不同类的安卓应用的安全强度，表示出不同的安全性。金融类安卓应用中，银行类产品一般都选择了第三方加固，所以只能进行加固检测，无法脱壳。本文中选取的金融类产品多为借贷类，可脱壳进行后续的静态检测和动态检测。金融类安卓应用选取宜享贷，游戏类安卓应用选取地牢年代记，其他类安卓应用选取每日心情。安卓应用安全检测评估得分具体得分结果如表 8 所示。由于列出检测不通过原因可能导致恶意者利用这些问题去对安卓应用造成损害，因此本文不列出未通过检测的原因，只列出是否通过检测。表中 √ 代表通过检测，×代表为通过检测。

Table 8. Typical Android application security assessment score sheet
表 8. 典型安卓应用安全检测评估得分表

检测指标	宜享贷	地牢年代记	每日心情
AC1	√	√	×
AC2	√	×	×
AC3	√	×	×
AT1	×	√	×
AT2	√	√	√
AT3	√	√	√
AT4	√	√	√
VB1	√	√	×
VB2	√	√	√
VB3	√	√	√
CS1	√	×	√
CS2	√	×	√
CS3	√	√	√
CS4	√	√	√
DS1	√	√	√
DS2	√	√	×
DS3	√	√	×
DS4	×	×	×
SO1	√	√	√
SO2	×	×	×
SO3	√	×	×
检测得分	0.9041	0.8149	0.5849

4.2. 实验结果分析

从安卓应用安全检测评估实验的结果来分析，由于金融类安卓应用对安全的需求是最高的，且银行类安卓应用大多采用了第三方加固，金融类安卓应用基本在各方面都采取了安全措施，自身安全性最强。根据图 1 所示可以得出，在安全检测评估中金融类得分普遍高于其他两种类型，特别是在反篡改检测指标和漏洞检测指标方面。本文列出的典型案例中，宜享贷问题主要出在为对资源文件进行加密或者隐藏，这可能是开发者一点疏忽，其他大部分金融类应用都做到了对资源文件的加密或隐藏。金融类安卓应用如果被攻击可能导致用户资金亏损问题，所以对于漏洞检测的重视程度大于其他两种类型。同时由于金融类安卓应用对需要对用户资料负责，安全相关类敏感函数调用过多也可能造成隐患，还需注意。在对

金融类安卓应用进行 Hook 框架测试时，部分应用会检测到 Hook 框架而弹出提示警告，并未退出，虽然能到达预警作用但还是可能造成危害。还有部分应用只能检测到一种 Hook 框架，另一种框架却能 Hook 成功，开发者应在开发时考虑完善，才能得到用户更多的信赖。

游戏类安卓应用中，本文选取了谷歌拥有百万下载量的地牢年代记来作为典型案例。可以发现游戏类安卓应用的加固检测得分往往都低于金融类，原因是游戏类安卓应用多数会使用自己研发的技术来对应用进行加固，这些技术往往考虑不一定周全，比如地牢年代记只是采用了较为检测的加壳技术隐藏了 Dex 文件，而没有对 So 文件进行加固，攻击者可以通过对 So 文件的检测判断出安卓应用加壳的模式与逻辑，从而将安卓应用进行脱壳处理。当然这个原因也在于游戏客户端需要大量的 So 文件和引用大量第三方组件，需要有比其他应用更快的更新速度有关。同时，也是因为这个原因，导致了游戏类安卓应用在组件安全类的得分会较低，大量的第三方组件不能保证每一个组件的安全性，可能会出现一些本文设计系统未知的组件或库，导致系统判定为检测不通过。还有个毕竟严重的问题在于，游戏类安卓应用由于组件过多，是的对 DDOS 攻击的防御能力普遍较低，当利用脚本向应用暴露组件发送畸形数据时，就会造成应用崩溃。

对于其他类安卓应用来说，安全性能就较低了，大部分应用甚至没有做安全加固。比如每日心情应用，完全没有经过加固就进行了上线。其他类安卓应用主要的安全得分点在于反篡改检测，漏洞检测和组件安全检测三方面。反篡改检测方面由于其他类安卓应用一般功能较为限定，并不会获取太多的权限，因而攻击者不大会去改变这类应用的信息等来进行攻击。漏洞检测方面则是由于技术的不断更新，大多数的安全漏洞都会随着安卓系统的版本提升而消除，本文选取的其他类应用大多为较新的软件，在漏洞扫描类得分都比较正常。同时由于其他类安卓应用不会使用太多的系统组件，反而提高了这类安卓应用在组件安全方面的得分，使攻击者无法通过组件方面的缺陷进行攻击。需要注意的是其他类应用也会保护用户的信息，但是这类应用在信息的安全保障上做的较差，各类数据基本处于无防护状态，面对 Hook 框架的 Hook 多少应用也毫无反应。

根据目前网上调研各类报告和各类安卓应用破解和攻击的文章综合分析来看，本文设计的安全应用安全检测评估模型基本符合客观规律。由于检测结果也会给出检测未通过的原因，可以有效的让开发者有针对性的进行修改，提高安卓应用的安全性。本系统能直观的看出安卓应用的安全强度，获得的评分越高则代表了安卓应用的安全强度越高，存在的安全风险也就越小。

测试中发现，测试应用大多缺乏了资源文件的加固指标，组件安全检测指标，Dex 文件加密或隐藏指标，敏感数据检测指标，畸形数据的拒绝服务攻击检测指标和 Hook 框架检测指标。这些指标的确实往往会造成严重的安全隐患。特别是大量的安卓应用并未对 Hook 框架进行检测，Hook 框架作为当前安卓应用安全事故产生较高频率的工具，能对安卓应用的用户信息造成极大的威胁。为保证用户的隐私信息，各大安卓应用都应该注意对 Hook 框架的检测。同时本文建议安卓应用在上线前最好进行完善的安卓应用加固工作，就算因为资金问题不能使用第三方加固，也需要自己设计尽量完善的加固方案，这不止是对用户的安全的负责，也是对自我产品的负责。

5. 结论

本文在分析安卓应用安全检测当前研究基础上，针对目前存在的没有一个成型的安卓应用安全检测模型的问题，从攻击者的角度出发，给出了一个相对完善的安卓应用安全检测评估标准。此标准从反编译检测、反篡改检测、漏洞检测、组件安全检测、数据安全检测和敏感操作检测 6 大方面进行了相对详细的检测说明。最后从实验数据库和市场上选取 120 个不同类别的应用进行安全检测和评估。并给出了不同类型的 3 种典型应用评估结果。实验结果表明该评估标准能有效地反应出 Android 应用的安全强度

高低。

本文主要是揭示出评估指标对于安卓应用安全性强度的重要性，检测评估方案目前划分较为简单，后续研究可以从以下方向进行：

安卓应用加固方案较多，检测内容也多，安卓应用加固的安全性会直接影响整个安卓应用，后续研究中可以将安卓应用加固检测单独进行研究，分析安卓应用加固在安卓应用整体安全性能中的重要程度。

安卓应用的安全指标并不是独立存在的，各项指标都有相互关联性，一个指标的缺失可能会影响其他指标，所以在评估模型的指标关联上还有所缺失，未来可以在这方面进行研究，使得评估结果更加准确。

参考文献

[1] 裘文成. 安卓 APP 安全性在线分析系统设计[J]. 电子世界, 2019(10): 141-142.

[2] 李涛, 张旭. 一种 Android 应用程序的安全检测方法及系统[P]. 中国专利, CN102831338A. 2012-06-28.

[3] 陈洋. 面向 Android 平台的软件安全检测技术研究[D]: [硕士学位论文]. 大连: 大连海事大学, 2015.

[4] 刘玮, 李蜀瑜. Android 移动应用检测研究[J]. 计算机应用与软件, 2019, 36(6): 322-326.

[5] 王喆. Android 移动终端数据安全检测方法研究[J]. 网络安全技术与应用, 2019(1): 63-64.

[6] 张静, 宋巍, 张炽华. 安卓应用服务泄露静态检测工具[J]. 电子设计工程, 2019, 27(13): 1-6.

[7] Martinelli, F., Mercaldo, F., Saracino, A., *et al*. (2016) I Find Your Behavior Disturbing: Static and Dynamic App Behavioral Analysis for Detection of Android Malware. 2016 14*th Annual Conference on Privacy, Security and Trust* (*PST*), Auckland, 12-14 December 2016, 129-136. https://doi.org/10.1109/PST.2016.7906947

[8] 齐林, 刘功申, 孟魁, 蔡逆水. 基于静态分析的虚假安卓 APP 分析与检测[J]. 通信技术, 2017, 50(12): 2840-2845.

[9] Wu, H., Yang, S. and Rountev, A. (2016) Static Detection of Energy Defect Patterns in Android Applications. *Proceedings of the* 25*th International Conference on Compiler Construction*, Barcelona, 17-18 March 2016, 185-195. https://doi.org/10.1145/2892208.2892218

[10] Fang, Z., Liu, Q., Zhang, Y., *et al.* (2015) IVDroid: Static Detection for Input Validation Vulnerability in Android Inter-Component Communication. Springer International Publishing, Springer, Cham. https://doi.org/10.1007/978-3-319-17533-1_26

[11] Narayanan, A., Chandramohan, M., Chen, L., *et al*. (2017) A Multi-View Context-Aware Approach to Android Malware Detection and Malicious Code Localization. *Empirical Software Engineering*, **6**, 1-53.

[12] 巫志文, 李炜. 基于 Android 平台的软件加固方案的设计与实现[J]. 电信工程技术与标准化, 2015(1): 33-37.