Mobei Security

银行移动应用安全性调查报告

（v 0.02）

上海墨贝网络科技有限公司

2015/06/07

■ 版权声明

目录

1. 前言

继银行卡支付，网上支付（PC 端）之后，中国消费者已快速进入了移动支付时代。据 CNNIC 发布的《第 33 次中国互联网络发展状况统计报告》数据显示：截至 2013 年 12 月，我国手机网民规模达 5 亿，较 2012 年底增加 8009 万人；手机支付用户规模达到 1.25 亿，同比增长了 126.9%，占手机网民总量的 25.1%。可见，手机支付用户的增长速度远远高于手机网民规模的增长速度。移动支付的时代已经到来，与此同时，移动安全上的隐患和威胁也被放大。

对银行业来说，为强化移动银行的安全环境，可以通过开展多样化的安全检测来实现风险防控，保障客户利益。例如，可以对移动银行应用进行安全评估，发现其安全隐患，通过升级来弥补漏洞；对移动银行应用进行安全加固，防止二次打包、在应用中植入恶意代码等恶意行为；开展对移动银行应用的 24 小时监控，一旦发现漏洞攻击或二次打包等情况，及时进行上报；建设风控体系，及时发现异常支付，并采取应对策略等。通过如上形式，给消费者提供真实可靠的支付渠道，保障移动银行业务安全。

目前，对移动银行应用进行安全混淆、对渠道进行监控、建设风控体系等，已有成熟的产品、服务和解决方案，但这些手段对发现黑客攻击具有一定的滞后性。提前发现、修复移动应用漏洞，对缓解针对移动应用的攻击有很大的帮助，是保证移动银行安全的一个重要先决条件。

攻击方针对移动应用的攻击主要是基于移动应用的漏洞来进行。实际操作发现，事后对移动应用漏洞进行修复需要付出巨大的代价，包括对渠道包的更新、版本兼容性的考虑等，都必须付出巨大的人力、物力、财力，移动应用提供方常常会陷入漏洞无法修复的尴尬境地。因此，为缓解移动应用漏洞的产生，必须在开发流程中、开发结束后，对移动应用的安全进行及时、整体的评估，减少移动应用可能暴露的攻击面。

在本报告发布之前，已经有部分机构对银行类移动应用进行过安全评估[1]，而我们将采用更严苛的评估方法，配合具体案例，来说明移动应用安全的严峻性。

2. 银行移动应用安全评估方法

本次评估从移动应用代码审计、移动应用安全加固两个角度进行。具体来说，移动应用代码审计包括：程序配置缺陷检测、本地接口安全检测、本地存储安全检测、远程接口安全检测，具体的审计项目尽量以owasp mobile security top10[2]为依据。移动应用安全加固包括：本地运行环境安全检测，代码保护能力检测、混淆方法识别。

评估过程所使用的工具包括：墨贝科技自研产品akana[3]、excavator[4]和一些内部开发使用的辅助工具等。

评估对象为随机选取的20个银行类应用。考虑到此类移动应用对安全性要求比较高，本次测试使用了相对严格的测试选项。

测试过程中，如果遇到有使用第三方混淆的应用，我们使用人工参与的方法，首先恢复出原始代码，再进行安全评估。

注：

1、样例的选取主要为配合2015信息安全可控XX选取，评估对象具有一定的针对性，不免出现遗漏。

2、选取了20个银行移动应用，涉及到12家银行移动应用产品，就样本空间来说，具有一定的普遍适用性。

3、评估过程使用了动静结合的方法，部分静态检测结果已经得到了动态确认。

3. 银行移动应用安全评测结果

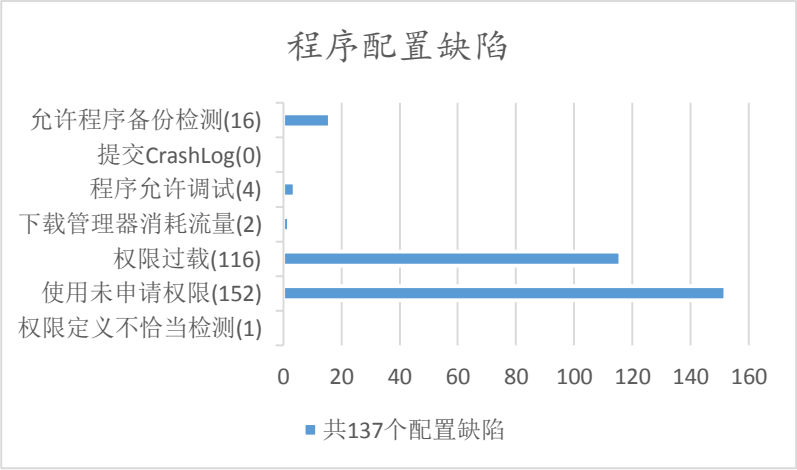
本次银行类移动应用安全评测主要以 owasp mobile security top10[2] 为依据，而不是以某些机构的自定义红线为基准。测试过程在一台 Ubuntu server、16G 内存、i7-4790 上完成，整个静态检测过程耗时 2 个小时。

3.1 移动应用代码审计

移动应用代码安全审计主要通过使用伪代码，建立伪代码运行的上下文关系，进行规则的匹配来进行的。报告中涉及到的漏洞术语、漏洞可能造成的危害、修正建议等信息可参考墨贝科技的移动应用安全知识库[5]或 owasp mobile security top10[2]，静态测试的主要测试范围为主应用代码，未对子包代码进行评估。为了对静态检测结果进行确认，我们使用了动态的方法进行了测试。

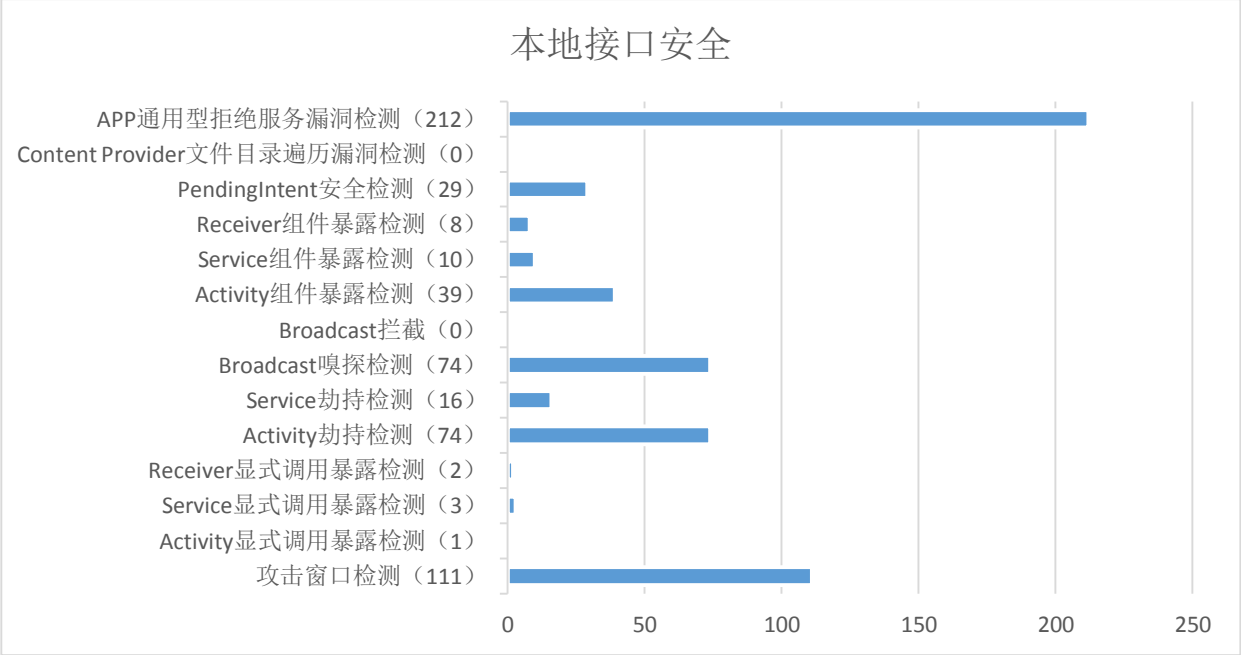
3.1.1 程序配置缺陷检测

程序配置缺陷指针对程序代码运行环境的一些设置的缺陷检测。具体包括：权限定义不恰当检测、使用未申请权限检测、权限过载检测、下载管理器消耗流量检测、程序允许调试检测、提交 CrashLog 检测、程序允许备份检测。检测结果如下：



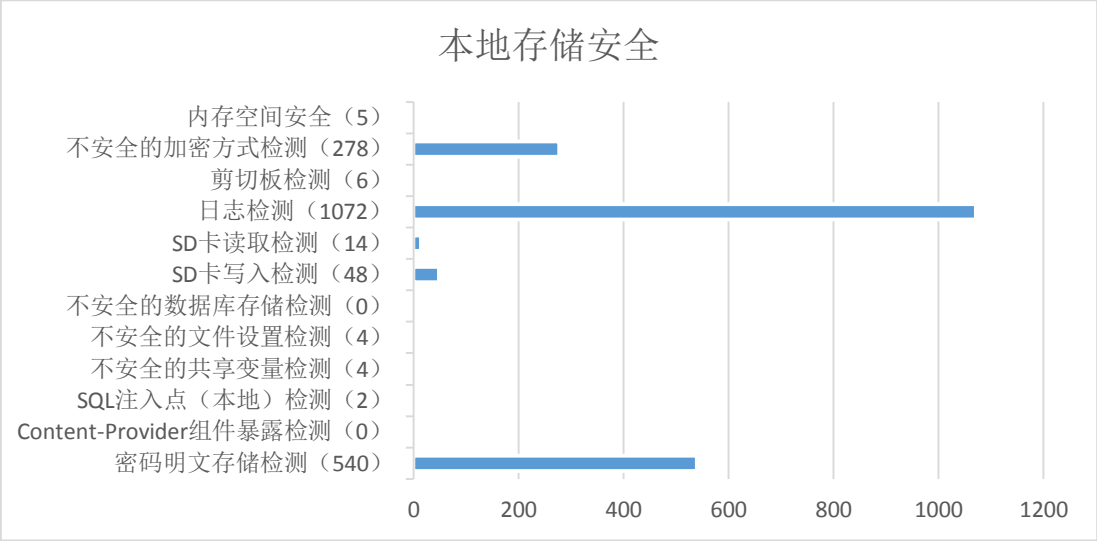
3.1.2 本地接口安全检测

本地接口安全指程序未对的本地接口进行严格的保护，导致程序可被本地其它应用攻击。具体的评估项目包括：攻击窗口检测、Activity 显式调用暴露检测、Service 显式调用暴露检测、Receiver 显式调用暴露检测、Activity 劫持检测、Service 劫持检测、Broadcast 嗅探检测、Broadcast 拦截检测、Activity 组件暴露检测、Service 组件暴露检测、Receiver 组件暴露检测、PendingIntent 安全检测、Content Provider 文件目录遍历漏洞检测、APP 通用型拒绝服务漏洞检测。检测结果如下：



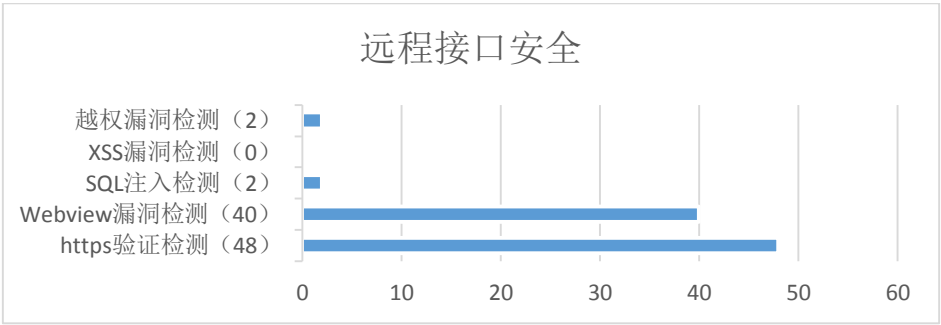
3.1.3 本地存储安全检测

考虑到银行应用的特殊性，对交互信息进行了严格检测，具体的检测项包括：密码明文存储检测、Content-Provider 组件暴露检测、SQL 注入点（本地）检测、不安全的共享变量检测、不安全的文件设置检测、不安全的数据库存储检测、SD 卡写入检测、SD 卡读取检测、日志检测、剪切板检测、不安全的加密方式检测，内存空间安全检测。检测结果如下：



3.1.4 远程接口安全检测

移动应用的远程接口检测主要测试移动应用与服务器的通信通道是否安全、对应的服务器是否安全。具体检测项目包括：https 验证检测、Webview 漏洞检测、SQL 注入检、XSS 漏洞检测、越权检测，检测结果如下：



3.2 移动应用安全加固

本文中所提到的加固与通常所所说的移动应用加固有所不同，我们更倾向于将提升移动应用整体安全性的手段称为加固，而将通常所提及的加固称为代码混淆。

3.2.1 本地运行环境安全检测

本地运行环境的安全性指，移动应用运行后，可以有效感知当前运行环境，清理潜在威胁，确保交易环境的安全。本项测试以手工的方式进行，测试发现，20 款银行类软件中，只有 4 款移动应用内置了第三方具备清场功能的反恶意程序引擎，用来保护运行环境的安全。

3.2.2 代码保护能力检测

移动应用必须采用有效的校验来防止移动应用被篡改。被篡改的移动应用不仅影响用户体验，损害 app 发布者形象，更严重的是，这个缺陷将引入新的安全威胁。测试过程中，我们发现大部分银行移动应用未做代码的有效校验，可被插入额外的代码、篡改原始代码。为方便测试，我们使用了第三方重打包工具 ninebox[6]对这些产品进行了插入代码操作，并进行测试，如果目标程序能正常运行，则表明此移动应用未能进行有效的校验，测试结果表明，45%的移动应用未做完整性校验，可以被重打包。

为探究未进行完整性校验的银行类移动应用被重打包的情况，我们抽取其中一个应用发现，在某分发渠道中，该银行的应用共出现 11 款同类产品，其中，只有 3 款是正版官方应用，其余 8 款，则是被第三方重打包的应用。

在测试的 20 个样本中发现了分别使用了 360 加固保[7]、阿里聚安全[8]混淆的 4 款移动应用。使用第三方混淆方案可以将代码的校验工作交由第三方来完成，并且，使用这种方法提高了攻击者的分析门槛。

4. 银行移动应用安全性分析

4.1 移动应用代码审计结果分析

银行类移动应用对安全性的要求相比其它应用更高，在依托操作系统提供的安全性的同时，安全边界必须拓展到移动应用本身。然而，测试发现，银行类移动应用与其它移动应用并无差别。

在程序的配置缺陷测试方面，我们发现，除了一些常见的默认配置缺陷外，特别的程序配置缺陷，如：程序允许调试，是由第三方混淆方案引入的。

本地接口安全性方面，由于银行类移动应用结构相对简单，这方面的漏洞与常见的小型移动应用并无太大差别，相对大型移动应用，银行类移动应用的本地接口方面的问题并不是很突出。

本地存储方面，我们考虑到银行类 app 对交易环境的高要求，使用了相对苛刻的检测选项，检测结果并不理想，如不少应用采用了弱加密算法，甚至出现了未正确使用加密算法的案例。

令我们感到吃惊的是，银行类交互的过程中，存在相当多的不合理的地方，如 https 协议的不合理使用，另外，webview 漏洞也大量存在于这些移动应用中。

4.2 移动应用安全加固结果分析

通过对移动应用安全加固进行分析，我们发现只有少量的银行类 app 使用了具备清场功能的反恶意程序引擎，相比安天[9]的分析引擎来说，这些引擎在检测方面的能力偏弱；另外在程序混淆方面，来自团队的内部测试结果表明，相比其它专业的混淆方案，几大互联网公司的混淆方案的混淆强度相对来说还是比较低的，从技术的角度出发，我们更倾向如娜迦科技[10]等专业的第三方混淆方案。

4.3 移动应用安全问题案例分析

为了证明移动应用中存在大量的安全漏洞，这里，我们尽量以一个机构的 app 为例，分析其存在的各个方面的问题。

例一：本地接口安全问题。该应用本地接口存在问题，导致任意第三方应用可以插入假消息，欺骗使用该应用的用户。

例二：远程数据短信接口安全问题。该应用远程数据短信接口存在问题，导致任意其它用户可以通过发送伪造的数据短信给安装有该应用的用户，欺骗使用该应用的用户。

例三：水平权限提升问题。该应用的后台服务器未对提交数据进行严格鉴权，导致攻击者可以在其它用户手机上生成任意订单。

5. 结论

从案例中我们可以发现，目前的银行类 app 总体安全性不高，除了传统的服务器端的安全性问题外，新型的客户端上的安全问题也普遍存在。

随着移动互联网的发展，攻防双方的主体正逐步往移动终端上迁移。银行类移动应用产品做为金融行业的载体，势必会越来越受到攻击方的关注。随着攻击方对移动应用安全的逐步理解，针对银行类移动应用的攻击会逐步显现。在这种环境下，有必要采取手段来保护银行类移动应用产品，消除任何一个可能导致攻击的攻击面，确保移动支付环境的安全。

6. 参考文献

1. 奇虎科技. *银行类手机APP安全性堪忧*. Available from: http://cdwb.newssc.org/html/2014-07/23/content_2086531.htm.
2. OWASP. *Top 10 Mobile Risks*. Available from: https://www.owasp.org/index.php/OWASP_Mobile_Security_Project#tab=Top_10_Mobile_Risks.
3. 墨贝科技. *Akana – 安卓移动应用静态审计环境*. Available from: http://www.mobeisecurity.com/?p=164.
4. 墨贝科技. *Excavator – 安卓移动应用动态安全审计环境*. Available from: http://www.mobeisecurity.com/?p=245.
5. 墨贝科技. *移动应用安全安全编码实践*. Available from: http://program-analysis.oicp.net/mediawiki/index.php/.
6. ninebox. Available from: http://www.ninebox.cn/.
7. 360加固保. *加固方案*. Available from: http://jiagu.360.cn/.
8. 阿里聚安全. *加固方案*. Available from: http://jaq.alibaba.com/.
9. 安天. *安天杀毒引擎*. Available from: http://www.antiy.cn/.
10. 娜迦科技. *加固方案*. Available from: http://www.nagain.com/application/.