某某应用安全开发规范

梆梆移动应用安全服务平台

2016 年 9 月 20 日

概述

基于应用创建时提交的相关信息，由梆梆移动应用安全服务平台，基于梆梆多年移动安全经验积累，及技术沉淀，设计的规则算法，自动化生成此应用安全开发规范文档，服务于应用开发阶段。

应用信息

应用名称：	民生手机银行
版本：	v1.0
开发者：	北京研发中心一部
联系人：	李小凡
联系电话：	13689736291
联系邮箱：	Xiaofan.li@126.com

网络通讯安全保护（适用于 Android、IOS）

由于本应用包含基于网络通讯的服务，考虑当前基于网络通讯的恶意攻击形式多样，如数据截取、篡改、身份仿冒等攻击手段。建议对网络通讯进行如下形式的保护：

建议方案	
1	使用加密的网络协议传输数据，如：HTTPS。
2	加密网络传输的数据，确保及时数被盗取，也无法破解利用。
3	应用客户端与服务端进行双向的身份验证，防止攻击者仿冒欺诈行为。
4	建立通讯消息的数据完整性验证机制，确保数据不被攻击者篡改。

本地数据存储安全保护（适用于 Android、IOS）

对于应用客户端本地数据，无论是文件形态还是存储在数据库中，建议均进行相应的保护，因为在非 root 的情况下，由于 Android 本身的沙箱机制，其它程序是无法读取这些数据的。但是在手机 root 的情况下，恶意程序可以直接读取这些数据文件。

建议方案	
1	对本地数据进行加密（如：配置文件、核心数据文件等），仅在应用使用数据时进行解密，保护本地数据即使被窃取也无法使用。
2	通过 API 对 Sqlite 数据库中的域进行加密，若需要，可将 Sqlite 数据库文件进行全文件加密，使数据库表结构也受到保护。

验证码安全保护（适用于 Android、IOS）

短信验证码：利用手机短信进行身份安全验证，是目前最主流的一种身份验证和支付授权机制；但同时由于此方式的普遍应用性，针对其出现的短信拦截、窃取的移动木马病毒种类，已超过 4000 种，成为病毒威胁的重灾区。建议对短信验证码实施针对性的防护措施。

建议方案	
1	应验证发出申请的手机与使用验证码的账户是否匹配，确保不被攻击者劫持利用。
2	应设置应用启动密码，防止手机丢失后通过短信验证码进行恶意操作；并且，应用启动密码应与账户绑定存储在服务端，防止通过重装应用绕过启动密码。

图形验证码：利用图形验证码验证用户是否为合法用户，是目前最主流的剔除机器人程序的验证手段。故对图形验证码的能力要求，就具有较强的针对性。

建议方案	
1	图形验证码的识别建议：机器程序难以识别，越难越好；人类可以识别，越容易越好。 对于验证方式，常用的输入、点选的方式比较容易被程序模拟，建议使用拖动等屏幕操作性强的验证方式。

信息输入安全保护（适用于 Android、IOS）

界面安全：由于金融 APP 的登陆、转账、支付等页面，涉及直接的利益关系，往往是攻击者进行信息窃取、截获的主要供攻击对象。攻击者通过界面的覆盖劫持，从而获取用户主动输入的账户等信息。

建议方案	
1	对自身 Activity 实时监控，发现界面被异常覆盖，提出警示并中断操作。
2	对一些已知手机、系统、App 本身存在的日志漏洞，进行有效的主动检测并提供修补措施，避免攻击者通过监听 Logocat 日志劫持界面。

键盘安全：作为输入信息的第一入口的“移动 App 键盘”，建议在本 App 的信息输入时，尽量不使用 Android 系统自带（或用户默认设置）的输入法；存在被攻击者植入键盘钩子，进行键盘监听的风险，从而导致用户核心数据的泄露。

建议方案	
1	建议随机键盘的字母、数字的位置，规避输入点位置监听恶意软件的威胁。
2	在用户输入时，屏蔽掉回显输入信息，规避屏幕截图的恶意软件攻击。
3	将输入的数据信息，在数据存储过程、内存数据换算过程中，应全程加密，防护底层 Dump 和内存读取等攻击手段。

多渠道发布安全保护（适用于 Android）

若应用存在多渠道发布的需要，考虑到各渠道对应用安全性的把控能力不同，建议对渠道中发布的应用，进行如下安全措施的保护：

建议方案	
1	应对客户端做完整性校验，校验对象包括：代码、资源文件、配置文件等原包中所有的文件。避免应用被植入病毒代码、广告 SDK 等恶意程序。
2	对渠道实行监测机制，在应用发布以后，对各发布渠道的应用情况进行实施监测，及时发现并解决潜在威胁。

源代码安全保护建议（适用于 Android、IOS）

由于苹果公司的审核需要，IOS 应用无法进行二进制可执行文件层面的加固，但基于源代码层面的安全加固保护，是 Android 应用和 IOS 应用均支持的，具体方案如下：

建议方案	
1	源代码混淆，可考虑利用代码混淆器，在应用编译发布前将源代码混淆，增加逆向调试难度。
2	插入花指令，在代码源文件中插入各种不会被执行的无效字节码，使逆向分析工具进行字节码解析时崩溃。
3	将控制流平坦化，在保证不改变源代码功能的前提下，将源码中的 if、while、for、do 等控制语句转化为 switch 分支选择语句。
4	将控制流不透明化，对于跳转控制条件和分支语句，在保持原程序逻辑关系的前提下，可随机确定控制块的执行顺序，达到模糊程序控制逻辑、隐藏程序控制流的目的。
5	进行代码完整性校验 ,可在混淆源码时植入 check 因子 ,在程序执行时校验同因子映射对应的代码，保证代码执行时的完整性。

SDK 安全使用建议（适用于 Android、IOS）

SDK 往往涉及很多关键功能，并且与其他核心模块存在强耦合，而 SDK 的 so 文件中包括算法、密钥、后台 API 接口、调用逻辑、漏洞等很多机密信息。对于所使用的 SDK，建议进行相应的保护措施。

建议方案	
1	SDK的代码和资源文件进行加密处理 ,整体加密甚至是函数方法级的加密。仅在应用调用某函数时，解密对应的函数方法，并且在执行后立即清除内存中的残留。
2	应用和 SDK 进行绑定，避免 SDK 被恶意调试，较高程度上防范破解、逆向分析。
3	对于 SDK 中涉及加解密服务的，对其加密算法及内嵌的密钥应进行有效的加密保护。
4	对于 SDK 已知漏洞的修补，往往通过升级实现。而能够及时的升级，则对 SDK 安全性具有非常大的影响。建议 SDK 能够独立自动化升级，不依赖于应用的版本升级。

密钥的安全管理 & 保护（适用于 Android、IOS）

应用使用加解密服务，势必会涉及对密钥的管理使用，而密钥（Key）是安全加密机制中最重要的元素。一旦密钥未受到有效的保护，那么使用密钥来实现保护的通讯协议、代码、业务逻辑、重要数据等核心资产都将面临被破解的风险。

建议方案	
1	密钥的访问权限应受到严格控制，仅应用某些服务时放能够访问，并且对于访问的权限具备验证机制。
2	密钥应为密文存储，即使被拷贝，也不应该能够被利用。
3	应使用加密的网络协议（如：https）对密钥进行可信的网络传输。
4	在加密网络协议传输基础上，进一步将密钥加密后进行传输，保证即使网络流量被截获破解，也无法得到明文密钥。
5	应用在使用密钥进行加解密操作时，内存中的密钥应受到严格的保护，避免被攻击者从内存中读取。

HTML 功能页面保护（适用于 Android、IOS）

对于 HTML、JavaScript 等脚本文件，实现上是以明文的方式存放安装包的资源文件或者运行时的本地数据中，攻击者甚至不需要逆向就可以直接拿到源代码。建议对此部分代码也进行相应的保护。

建议方案	
1	建议脚本文件进行加密存储，仅在使用时，进行动态解密。
2	对于 HTML 页面中包含的输入操作，如：账户、密码；应使用具备加密能力的安全键盘控件，保证数据在输入、传输过程中的安全性。

应用环境保护建议（适用于 Android、IOS）

应用软件所运行的环境是多样的、未知的，至于环境是否安全，是否已经存在运行中的木马病毒？

虽然对于移动客户端已集成各方面的安全保护措施，极大的保护了应用的自身安全。但作为金融类移动端产品，其本身就是攻击的重灾区，对于运行环境的安全可靠，相对于其它类型的应用，需要更主动的关注和维护。

建议方案	
1	在应用中集成轻量级的反病毒引擎，从而对运行环境的安全提供有效的保障。

更多服务

梆梆安全作为国内最大的移动金融安全服务供应商，梆梆安全率先在全球推出 App 安全加固服务，成为国内唯一覆盖 Android、IOS 平台的移动应用安全公司。梆梆安全专注为金融、政府、企业、移动互联网及物联网行业提供移动应用安全整体解决方案，以及安全咨询、渗透测试、源码审计、安全应急响应、安全培训等技术服务支持。

地址：北京市海淀区学院路 30 号天工大厦 A 座 6 层
服务电话：4008-881-881
公司网站：https://www.bangcle.com/