Computer Science and Application 计算机科学与应用, 2014, 4, 393-399
Published Online December 2014 in Hans. http://www.hanspub.org/journal/csa
http://dx.doi.org/10.12677/csa.2014.412052

Hans汉斯

Research on File Encryption of Intelligent Terminal Based on Android Platform

Xiaojie Liu[1], Zhansheng Chen[1,2*], Bingwen Su[1], Zijun Chen[1]

[1]School of Applied and Technology, Beijing Union University, Beijing
[2]School of Computer and Information Technology, Beijing Jiaotong University, Beijing
Email: *ldtchenzs@buu.edu.cn

Received: Nov. 8th, 2014; revised: Dec. 9th, 2014; accepted: Dec. 24th, 2014

Copyright © 2014 by authors and Hans Publishers Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
http://creativecommons.org/licenses/by/4.0/
Open Access

Abstract

The open source characteristic of the Android attracts numerous malware, and draws a large number of researchers that are focus on the security enhancement field of Android. The paper introduces the Android system architecture, security mechanisms and its safety defects. Aimed at file security enforcement of Android system, we use the existing mature file encryption algorithm, based on the learning of Android application development, design and develop a file encryption system.

Keywords

Android, Security Enhancement, Encryption Algorithm, System Architecture, Security Mechanism

基于Android平台智能终端文件加密的研究

刘晓杰[1]，陈战胜[1,2*]，苏炳文[1]，陈子君[1]

[1]北京联合大学应用科技学院，北京
[2]北京交通大学计算机与信息技术学院，北京
Email: *ldtchenzs@buu.edu.cn

收稿日期：2014年11月8日；修回日期：2014年12月9日；录用日期：2014年12月24日

*通讯作者。

摘　要

Android源代码开放的特性招致了众多恶意软件的攻击，吸引了大量研究人员对Android安全增强方面的关注。文中介绍了Android系统架构及安全机制，对其安全缺陷进行总结。针对智能手机文件安全增强方面，利用已有成熟的文件加密算法，在了解学习Android应用程序开发基础上，设计并开发了Android手机文件加密系统。

关键词

Android，安全增强，加密算法，系统架构，安全机制

1. 引言

移动通信技术的快速发展使得人们对智能手机的依赖性愈发凸现，用户通过手机开始使用越来越多的 APP 游戏、商用软件。这些 APP 应用在给用户带来便捷的同时，也涉及用户个人私密信息泄露的安全问题，主要表现在大部分 APP 应用都有隐私信息越权偷取问题，主要表现在调用通讯录、扫描用户信息并上传、强制用户安装应用、恶意扣费等恶意行为。

Android 系统于 2007 年由 Google 公司与开发手机联盟同期发布，其开放性和移动性特点受到用户喜爱，同时使得手机上的个人私密信息面临着被窃取的风险，这会给用户造成不同程度的损失。虽然 Android 系统提供了应用沙盒、签名、组权限等安全权限，但“以开发者为中心，依托用户”的粗粒度授权机制所引发的诸多问题，导致了平台安全性大幅降低。而文件加密作为一种保护用户敏感数据的有效手段，能够有效增强用户个人数据的安全性和可靠性。

近年来，众多学者早 Android 系统文件加密研究方面取得了一些成果。其中，曾炜[1]等人基于 Windows XP 操作系统，利用 VC++编程语言，实现了一个基于 AES 的加解密系统，能够对字符串和文件使用 AES 算法进行正确的加密和解密。张新中[2]等人利用 C#作为文件加密系统的开发平台，采用面向对象设计方法，基于 DES 加密算法实现了文件加解密等功能，测试表明该系统可对 ASCII 文件正确快速加解密。

针对 Android 智能手机 APP 的广泛应用及其存在个人隐私泄露和经济损失的安全威胁，本文在了解 Android 系统架构、安全机制及其缺陷的前提下，基于 Android 平台设计并开发了一个文件加密系统，为用户提供 DES、AES 等成熟文件加密算法，有效增强了 Android 智能手机文件的保密，便于用户使用。

2. Android 分层式系统架构

Android 系统作为移动设备软件栈，采用四层架构设计，从底层到上层分别是 Linux 内核层、系统运行库层、应用程序框架曾及应用程序层。

其中，Linux 内核层中包含摄像头驱动、内存驱动、键盘驱动、显示驱动等核心系统服务，是介于底层硬件和上层应用之间的抽象层，负责上下层之间的交互。Android 系统运行库层分为本地库和运行环境，本地库用户提供 Android 系统不同组件调用的函数库，便于开发者方便开发，主要包含 Surface Manager、SQLite、SSL 等九大部分；运行环境主要是 Dalvik 虚拟机和 Java 库组成，为应用程序提供运行环境。应用程序框架为 Android 开发的基础，提供应用程序 API，主要包含 Activity Manager、Window Manager、Location Manager 以及 Notification Manager 等九大部分。应用程序层运行在 Dalvik 虚拟机上，这些应用程序既包含 Android 平台自带的系列核心应用程序，如地图、邮件、联系人管理程序等，也包括用户自

行下载安装的应用程序。

3. Android 安全机制及其缺陷

本文主要研究 Android 就应用框架和应用程序层面提供的安全机制[3]，主要分为隔离机制、权限机制、虚拟机防护。

3.1. 应用程序隔离机制

Android 基于 Linux 内核，每个应用程序被赋予唯一的用户 ID，通过为其创建一个 Sandbox 运行于自己的 Linux 进程中，借由 Linux 自主访问控制机制使得应用程序不为其他应用程序所见，同时防止影响其他运行程序，起到了有效的隔离安全保护。

3.2. 应用框架权限机制

Android 系统定义了一系列与安全操作相关的权限标记，用于提供应用框架层中组件之间访问的强制访问控制权限机制。通过在配置文件 Manifest.xml 中设定应用所需求的权限，在用户授权后该应用中组件就可以继承相应的设定权限。

Android 系统提供四种权限保护等级，分别是 Normal (普通)、Dangerous (危险)、Signature (签名)和 Signature Or System (签名/系统)。其中，普通级别隐藏于目录或显示于屏幕，危险级别则在程序安装时会在屏幕显示，签名级别只有当请求程序与声明程序的签名相同时才会被授权。

3.3. 虚拟机防护

Android 采用进程隔离机制来保障应用软件与其数据文件的独立性和完整性。

在 Android 智能手机 APP 广泛应用中，其安全威胁[4]主要来自各种手机病毒而引发的威胁，主要表现在数据安全和系统安全两个方面，需要用户警惕的表现主要有手机个人私密信息丢失；恶意代码的植入；恶意代码利用服务提供者吸费模式，植入用户手机后隐秘的想 SP 号发送定制业务信息，屏蔽 10086 等扣费确认短信，并销毁记录等系列暗箱操作，造成通信费用的剧增；通过手机控制，传播不法信息等。就本质而言，安全威胁与 Android 开放性特点，对系统空间访问限制、应用商店的安全监管机制不完善有着密切的联系。

4. 加密算法的研究成果

加密算法研究层出不穷，但是为业界所承认并广泛使用的依然集中在几种经典加密算法，如对称加密算法、非对称加密算法等。在对称加密算法中，数据加密、解密采用同一个密钥，优点是加解密速度快，但其安全性主要取决于秘钥的安全性，常见算法有 DES、AES 等。非对称加密算法使用两个不同但又匹配的公钥和私钥，公钥用于加密私钥用于解密，常见算法有 RSA，DSA 等。

本文设计的文件加密算法采用对称加密算法，主要介绍 DES 算法和 AES 算法。

4.1. DES 加密算法

DES (Data Encryption Standard)加密算法是早期加密算法的研究基础，如后续的 LOKI、3DES、GOST 等。DES 加密算法采用 56 位密钥附加 8 位奇偶校验位。分组后的明文与 56 位密钥按位交换或替代的方式形成密文组的加密方法，其优点是运算速度快。随着 90 年年代初差分密码分析(differential cryptanalysis)和线性密码分析(linear cryptanalysis)的提出，以及 IDEA 密码的出现打破了 DES 类密码的垄断局面。

4.2. AES 加密算法

AES 是一种替代 DES 的高级加密算法，它的加密数据块分组长度为 128 bit，密钥长度最少 128 位，其特点是运算速度快，安全级别高。

NIST 将 Rijndael 算法作为高级加密标准，该算法运算容易，Rijndael 加密算法明文区块的初值是一个 4 × 4 的字节矩阵，简称为“体”，除了最后一轮加密过程之外，所有的 AES 加密循环都遵循四个步骤：1) AddRoundKey 操作，即采用子密钥生成方案生成的 Round KEY 与矩阵中字节进行 XOR 运算；2) SubBytes 操作，即基于某非线性替换函数，采用查找表方式对相应字节进行替换；3) ShiftRows 操作，即将矩阵中横列进行循环式移位；4) MixColums 操作，即启用线性变化混合每行中的四个字节。结尾的 MixCollums 步骤由 AddRoundKey 取代。

Rijndael 算法加解密流程如图 1 所示。

5. 加解密系统的设计与实现

5.1. 系统设计

本文基于 Android 平台设计开发一款 Android 智能手机文件加解密系统，系统功能模块如图 2 所示。

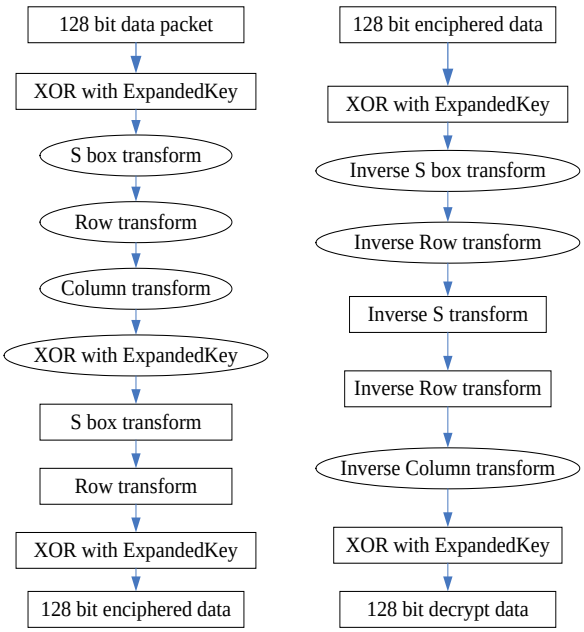

Figure 1. The basic flow of AES encryption algorithm
图 1. AES 加密算法基本流程

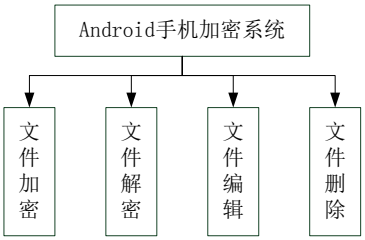

Figure 2. The function diagram of mobile encryption system
图 2. 手机加密系统功能图

如图 1 所示，本文设计的手机加密系统共包含文件加密、文件解密、文件编辑和文件删除四个功能。其中，文件加密功能可以对手机及外扩内存卡中任意格式文件进行加密。文件解密功能是将已加密文件进行还原，解密时需输入加密时设定的密码，此外，文件在解密时重新修改文件名，也可以不修改。文件编辑功能是针对用户想加密保护的文件，待编辑后保存即可。文件删除功能是为防止隐私文件被恶意复制，只要输入明文文件名即可删除。

此外，系统在用户界面采用人体视觉舒适的蓝色调，简洁大方。

5.2. 系统实现

本文使用 Eclipse 集成开发环境，采用 Java 语言，设计并实现了基于 Android 平台的文件加密系统，工作流程如图 3 所示。系统在真机上测试通过，运行结果如图 4 所示。

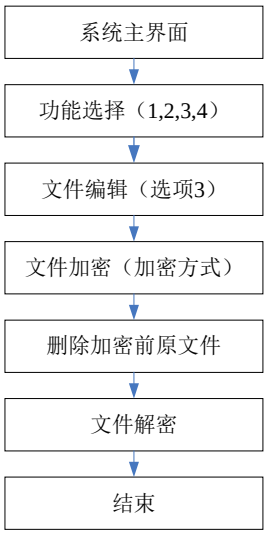

Figure 3. The workflow of mobile encryption system
图 3. 手机加密系统工作流程

(a) (b) (c) (d)

Figure 4. The run results of mobile encryption system
图 4. 手机加密系统运行结果

如图 4(a)所示，系统开始运行首先显示主界面，包含文件加密、文件解密、文件编辑和文件删除四大功能。测试中首先使用文件编辑功能，编辑文件 a.txt，内容如图 4(b)所示。然后对文件 a.txt 进行加密，单击图 4(a)中文件加密按钮，出现如图 4(c)所示界面，提示用户可以在 DES、DESede 和 AES 三种解密方式中任选其一，并提示用户输入加密秘钥以及输出的加密文件名。本文采用 AES 加密方式，加密后文件内容如图 4(d)所示。在文件解密过程中输出文件名可以为原文件名或自定义文件名。文件删除的作用是将加密后的明文文件删除。

由于不同的加密算法中密钥长度不同，在系统实现中通过分支判断语句 Switch()语句判断用户选择的加密算法，采用 While()循环保存输入的密码，采用 GetKeyPair()产生获得 DES 加密的密钥。其中，生成密钥保存到 DeskKey.xml 中的代码入下：

```
public static void saveDesKey(){
    try {
      SecureRandom sr = new SecureRandom();
    //生成一个 KeyGenerator 对象
    KeyGenerator
    kg = KeyGenerator.getInstance ("DES" );
    kg.init (sr);
    FileOutputStream
    fos=new ileOutputStream("E:/DesKey.xml");
   ObjectOutputStream
    oos = new ObjectOutputStream(fos);
  //生成密钥
    Key key = kg.generateKey();
   oos.writeObject(key);
   oos.close();
  } catch (Exception e) {
      e.printStackTrace();
 }
}
```

6. 结束语

Android 智能手机平台所具备的开放性及完整的系统架构，为广大 APP 应用开发爱好者提供了开源、便捷的开发平台。本文在介绍 Android 系统架构、安全权限和安全缺陷后，探究 Android 开发框架并设计研发了手机文件加密系统，优点是操作简单明了，不足在于文件编辑后明文中的空格会在加密中全部消除，将在后续工作中研究并加以改进。

此外，透明加密技术[5] [6]是在用户不改变操作习惯，毫无察觉的情况下完成文件的加密和解密，主要针对企业内部文件的保护。当系统监测到用户对受保护文件进行读写操作时，会自动对文件进行解密操作，将明文传送给应用程序以显示，几乎完全不影响用户的读写操作。当系统监测到用户对受保护的文件结束读写操作时，会对文件自动加密并传送至存储设备进行存储。其特点是与文件使用环境密切相关，受保护文件在硬盘上是密文，而在内存中是明文，环境一旦改变，加解密服务将无法完成，从而保证了文件的安全。

本文将会考虑研发基于透明加密技术的 Android 文件加密系统的设计与开发，为广大用户提供一种选择方案。

基金项目

北京联合大学“启明星”大学生科技创新项目(12222994701，12222994501)，北京联合大学新起点计划项目资助(zk10201303)，北京市职业院校教师素质提高工程资助项目(京教财[2012]21 号)。

参考文献 (References)

[1] 曾炜, 王科怀 (2014) 基于 AES 加解密系统的设计与实现. *软件导刊*, **13**, 132-134.

[2] 张新中, 吴玉祥 (2014) 基于 DES 的文件加密系统设计. *电脑开发与应用*, **27**, 43-44.

[3] 蒋绍林, 王金双, 张涛等 (2012) Android 安全研究综述. *计算机应用与软件*, **29**, 205-209.

[4] 黄小林 (2014) Android 系统数据保护关键技术研究. 西南理工大学, 成都.

[5] 唐铭若 (2012) 基于 Android 平台的文件透明加密的设计与实现. 电子科技大学, 成都.

[6] 王艳敏, 李永忠, 吕少伟 (2014) Android 平台下文件透明加密技术的研究与实现. *计算机技术与发展*, **24**, 137-140.