

在 Android 平台下 利用 WiFi 技术实现即时通讯

文/肖 洋 郭 平 王 莹

摘 要:Android 平台在移动嵌入式开发上的应用日益广泛。在未来的无线网络,仅仅依靠 3G 技术无法解决热点地区的大规模数据应用,而必须辅之以传输速率更快的 WLAN 接入技术,本文从 Android 的特征、架构出发,利用其对 WiFi 技术的支持来实现一定范围内的即时通讯。

关键词:Android;WiFi;即时通讯

随着当前数据处理中心移动化的浪潮,移动互联网应用正以其方便、轻巧、快捷、简单等特性成为越来越多公司和个人的选择。以 Linux 为基础的 Android 平台,相对于 Symbian、Windows Mobile 和 Linux 等操作平台,对开发者来讲拥有更大的自由空间,其开源的理念直接促进了 Android 的应用不断扩大。而 WiFi 解禁所带来的网络速度提升和通讯便利,使得无线局域网蓬勃发展,比如在医院处方管理、仓储资源调配、公司内部通讯以及个人联系等各方面,无线通讯

相较传统通讯而言占据相当大的优势。

1.Android 平台简介

1.1 Android 特征

提供应用程序框架

定制的 Dalvik 虚拟机

集成了基于 Webkit 的开源项目的浏览器

提供 SQLite 数据库用于结构化数据存储

提供对音频、视频和图片等媒体的支持

支持蓝牙、EDGE、3G 和 WiFi

WiFi 的最大优点就是传输速度较高,可以达到 11Mbps,另外它与已有的各种 802.11 DSSS 设备兼容支持 GSM 电话控制。其主要特性为:速度快、可靠性高。在开放性区域,通讯距离可达 305 米;在封闭性区域,通讯距离为 76 米到 122 米,方便与现有的有线以太网网络整合,组网的成本更低。

(7) 强大的开发环境 本文实例采用 Eclipse3.6(HELIOS)作为集成开发环境,配置 JDK1.6.0-20 以及 Android 的最新 SDK: android-sdk-r05-windows, 以及

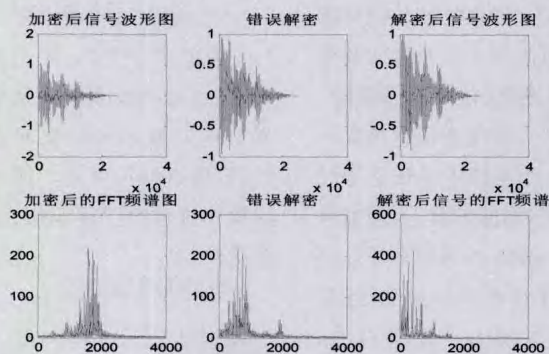


图4 解密仿真结果

4.系统的密码学分析

声音信号具有平稳变化的性质,频域加密要优于时域加密。本文对 87 个 FFT 系数进行了置乱,选择合理的密

钥,加密后的信号剩余可懂度会很低。实际应用中,考虑窃听者掌握了加密解密方法,并拥有足够的设备进行破解,那么最大需要进行 87! 次分析才能得

到结果。因此,FFT 方法拥有较强的保密能力。

5.结束语

本文基于 FFT 信号处理方法,设计了基于 FFT 的语音信号加密算法。对语音进行频域置乱。对仿真结果进行分析,可观察到加密后的语音已完全不同于源信号。在输入不正确的密钥时,不能正确的解密。系统的密钥空间为 87!,因此系统能抵抗一定程度的密码学攻击。解密后的信号具有较好的通话质量。本系统的整体结构,对研究更高级别的话音加密提供了成熟的理论基础,该系统具有一定的推广应用价值。□

(作者单位:后勤工程学院)

Android 平台自带的设备模拟器, 调试、内存和性能检测等工具, 以及 EclipseIDE 的开发插件 ADT-0.9.7。

1.2 Android 架构

Android 平台是在 Linux2.6.25 版本的基础上改造的, 不同的是 Android 的目标平台是 ARM 平台, 而不是通常的 i386 平台。Android 的模拟器是基于 Qemu0.8.2 和 SDL 进行开发的模拟环境。图 1 为 Android 平台的架构图:

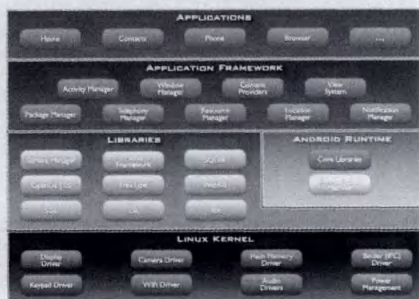


图 1 Android 平台架构图

2. WiFi 技术简介

WiFi 是由 AP(Access Point)和无线网卡组成的无线网络。AP 一般称为网络桥接器或接入点, 其工作原理相当于一个内置无线发射器的 HUB 或者是路由, 而无线网卡则是负责接收由 AP 所发射信号的 CLIENT 端设备。

WiFi 最主要的优势在于不需要布线, 可以不受布线条件的限制, 因此非常适合移动办公用户的需要, 具有广阔市场前景。目前它已经从传统的医疗保健、库存控制和管理服务等特殊行业向更多行业拓展开去, 甚至开始进入家庭以及教育机构等领域。另外, 无线网络的发射功率实际只有约 60~70 毫瓦, 手机的发射功率约 200 毫瓦至 1 瓦, 手持式对讲

机高达 5 瓦, 而且无线网络功率低, 是更安全的。

3. 采用 WiFi 技术实现即时通讯

3.1 系统概述

本文工作中实现的是在 Android 平台上开发利用 WiFi 实现即时通讯, 它能够让使用 Android 平台并支持 WiFi 功能的手机或电脑用户方便的、迅捷的相互联系, 而不会产生各类资讯费用。据最新报道, 在硬件支持的条件下, 比如使用最新型 Ubiquiti 公司的设备可以让传送距离达到 304 公里。

3.2 编程实现

项目的创建: 启动 Eclipse, 选中 File>New>Android Project, 创建项目 WiFiConnect, 选择 Google APIs 2.2, 并创建基于此库的 AVD: WiFiAVD。Android 平台中提供了 android.net.wifi 来支持 WiFi 的应用, 表 1 是要用到的主要类/接口的说明。

(1) WiFi 使用许可 为了获取 WiFi 网络状态, 必须在 AndroidManifest.xml 中添加访问许可: `<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>`

(2) WiFi 管理 获取 WiFi 管理器接口的代码如下:

```
WifiManager manager = (WifiManager) (this.getSystemService(Context.WIFI_SERVICE));
```

(3) 网络状态及扫描结果 WiFi 管理器可以通过 getScanResult 方法来获得最新访问点的 BSSID、capabilities、frequency、level、SSID 等信息:

```
List<ScanResult> list_result = service.getScanResult();
```

表 1 android.net.wifi 的主要类/接口说明

类/接口	说明
WifiManager	提供了管理所有 WiFi 连接的 API
WifiInfo	描述了 WiFi 连接状态
WifiConfiguration	代表了一个已配置的 WiFi 网络, 包括安全配置
ScanResult	用于描述探测到的存取点的信息

管理器也可以通过 isWifiEnabled 方法来获得 WiFi 网络是否可用: `if(manager.isWifiEnabled())`来判定。

(4) Telephony 和 SMS 管理 Android 平台中提供了 android.telephony 支持电话和短信的应用, 其中 android.telephony.TelephonyManager 提供了访问电话设备信息的方法:

```
TelephonyManager telmanager = (TelephonyManager) (this.getSystemService(Context.TELEPHONY_SERVICE));
```

android.telephony.SmsManager 则提供了短信的管理方法:

```
SmsManager sms = SmsManager.getDefault();
```

同样, 电话和短信的功能需要分别在 AndroidManifest.xml 中声明使用 android.permission.CALL_PRIVILEGED 和 SEND_SMS, 同时为了获得电话设备的信息 android.permission.READ_PHONE_STATE。

(5) 与发现的 WiFi 设备连接

在这里只介绍采用 SMS 方式连接已经发现的 WiFi 设备: `PendingIntent pi = PendingIntent.getActivity(this, 0, new Intent(this, Sendsms.class), 0); sms.sendMessage(deviceID, null, message, pi, null);` 其中 deviceID 是发现的设备 ID 号, message 是所需要发送的内容, pi 是提示发送完成的提示参数。

4. 结论和展望

Android 应用程序开发着眼于它的整个体系构架, 是一项复杂的工程。本文分析了 Android 体系的层次架构, 利用其对 WiFi 的支持开发一个即时通讯软件, 在下一步工作中, 将继续深入研究 WiFi 技术的安全以及带宽对通讯的影响。从当前国内移动通讯的浪潮来看, 使用 WiFi 的即时通讯技术不仅在民用方面有较大应用前景, 在军事上也将有较大的发展空间, 包括构建战地 Ad-Hoc 网络等应用。■

(作者单位: 后勤工程学院/海军 92117 部队)