Design and implementation of mobile terminal anti-theft tracking system based on Android platform

Yonghong Luo[1, a], Jian Wang[2, b] and Chao Feng[3, c]

School of electronic science and engineering, National university of defense technology, Changsha 410000, China

[a]578834843@qq.com, [b]Jwang@nudt.edu.cn, [c]fengcaho@163.com

Keywords: Android, broadcast mechanism, backstage monitoring, anti-theft tracking.

Abstract. In order to solve security problems such as user privacy information revealed after mobile phone lost, based on Android broadcast mechanism and backstage service technology, improved data storage way and self-startup mode of original mobile phone security software, added SMS encryption、message broadcast priority interception and sandbox breakthrough technology, a mobile phone anti-theft tracking system with remote SMS management function was designed and implemented. The software not only owned basic anti-theft function of automatically remove sensitive information to protect user privacy security after mobile phone lost, but also silently monitored、GPS positioned for lost phone and got the latest contacts to track stolen phone. Real machine and simulator two experimental results show that the system is running well, can achieve desired design requirements.

Introduction

Maturing mobile communication technology makes mobile phones become more powerful, cellphones gradually get rid of image of traditional communication tool and become primary device for information in the mobile internet era, among them Android phones with open source、easy to use and powerful features get the majority of users support and love and its market share steadily increases year by year [1-3]. With the increase of computing power and storage capacity of smartphone, more and more users start using their phone to deal with personal business, they are accustomed to storing bank card account、payment code、video photos、personal diaries and other sensitive private information on the phone [4]. Once the phone lost, users will face a serious risk of loss of privacy, may not only cause loss of property but also have a negative impact on the life and work of their own and even families [5-7]. In response to these security issues, this paper designs an Android handset anti-theft tracking system which would automatically destroy private information and track cellphone location to protect user privacy and property security after the phone lost.

System Overall Design

System Workflow. If mobile security software runs on the cellphone for the first time you need to set initialization parameter, including register login username and password、set remote control phone number and save them together with initial sim card information, then system automatically opens anti-theft function; If not the first landing, you can close the software or reset the parameters by entering correct login username and password to access the system, otherwise the software will automatically quit login interface and continue to run anti-theft function in the background on condition that the password entered incorrectly more than three times. The security software will extract the current sim card information and compare it with the stored origin sim card information at each time the phone boot, if they are the same so nothing is done; if different, explain that the phone has been stolen for the sim card has been replaced so an alarm message will automatically sent to secure mobile phone by the software. Simultaneously the system backstage monitors received message in real time, then decrypts message content to perform corresponding operation in case of the message sent by secure phone, otherwise default does nothing.

The anti-theft tracking process of mobile security system is shown in figure 1.

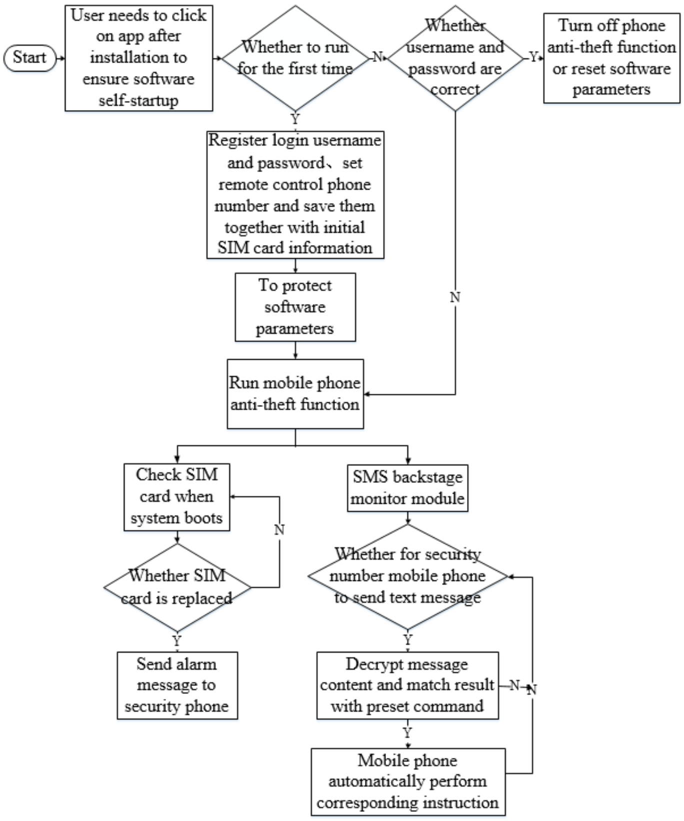
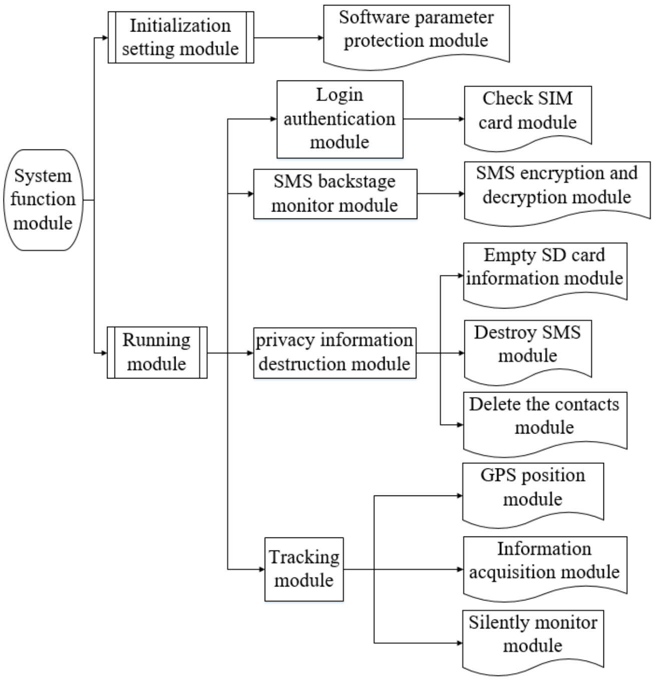

Start
User needs to click on app after installation to ensure software self-startup
Whether to run for the first time
N
Whether username and password are correct
Y
Turn off phone anti-theft function or reset software parameters
Y
Register login username and password、set remote control phone number and save them together with initial SIM card information
To protect software parameters
N
Run mobile phone anti-theft function
Check SIM card when system boots
SMS backstage monitor module
N
Whether SIM card is replaced
Whether for security number mobile phone to send text message
Y
Y
Send alarm message to security phone
Decrypt message content and match result with preset command
N
N
Y
Mobile phone automatically perform corresponding instruction

Fig. 1 Anti-theft tracking process

System Functional Architecture. Mobile security system based on Android broadcast mechanism and backstage service technology consists of initialization setting module and running module. In detail, software parameter protection module protects basic parameter information set by initial module from attack in initial stage, and then SMS decryption module decrypts message content captured by SMS backstage monitor module in operational phase, according to corresponding SMS commands privacy information destruction module is chosen to run to protect user privacy or tracking module is selected to retrieve lost mobile phone.

The function module structure of phone anti-theft system is shown in Figure 2.



System function module
Initialization setting module
Software parameter protection module
Login authentication module
Check SIM card module
SMS backstage monitor module
SMS encryption and decryption module
Empty SD card information module
Running module
privacy information destruction module
Destroy SMS module
Delete the contacts module
GPS position module
Tracking module
Information acquisition module
Silently monitor module

Fig. 2 The system function module structure

System Module Implementation

This article only describes software parameter protection module、self-startup module、SMS backstage monitor module and information acquisition module in detail because of excessive modules of security software.

Software Parameter Protection Module. Software parameter protection module is responsible for the protection of authenticity、privacy and integrity of basic parameters. There are five data storage ways under Android platform, namely sharedpreferences class storage、file storage、sqlite database storage、contentprovider class storage and network storage [8]. We consider that basic parameter information stored locally may be tampered or cracked reversely by attackers who even use a pseudo base station of fake security number to control user's phone, therefore this system uses network storage mode to update information at any time, or converts parameter information into md5 value and uses irreversibility of hash algorithm to ensure data security.

Self-startup Module. Mobile phone security software needs to startup for testing current sim card information at each boot time. Because Android system will send an "android.intent.action.BOOT_COMPLETED" broadcast to all applications after each boot, so before Android 3.1 we can register an broadcastreceiver whose <intent-filter> is equal to <action android: name= "android.intent.action.BOOT_COMPLETED" /> to achieve self-startup of phone software [9]. But after Android 3.1, packagemanager class of Android system has enhanced "stopped state" applications management, here "stopped state" applications refer to applications which have never been started after installation or forcibly stopped manually by user. Android system adds two flags namely FLAG_INCLUDE_STOPPED_PACKAGES and FLAG_EXCLUDE_STOPPED_PACKAGES to identify an application if its intent is activated so is in "stopped state" applications, when two flags are all set up or not, FLAG_INCLUDE_STOPPED_PACKAGES flag is efficient, so based on the new mechanism, applications which have never been started after installation will not receive "android.intent.action.BOOT_COMPLETED" broadcast.

To achieve self-startup of mobile phone security software after Android 3.1, we promote app privilege of software to system application level and handle it with system signature, then software will startup as Android system boot.

The core codes are as follows:

```
<uses-sdk android: minsdkversion= "7" android: shareduserid= "android.uid.system" />
<action android: name= "android.intent.action.BOOT_COMPLETED" />
<uses-permission android: name= "android.permission.RECEIVE_BOOT_COMPLETED" />
```

SMS Backstage Monitor Module. SMS backstage monitor module based on background non-interface components broadcastreceiver and service is an important component of mobile phone anti-theft system, can realize that master phone controls stolen phone by intercepting and monitoring short messages received by cell phone.

According to Android system broadcast mechanism, when received short messages Android system will send global broadcast “android.provider.Telephony.SMS_RECEIVED” to activate related receiver to process messages, this module processes messages by registering an broadcastreceiver whose <intent-filter> is equal to <action android: name= “android.provider.Telephony.SMS_RECEIVED” /> and calling its own onReceive() function to launch a service component [10,11].

Blocking system SMS receiving function is crucial to this module, for message broadcast sent by Context.sendOrderedBroadcast() method is ordered broadcast so high-priority broadcast receiver firstly receives broadcast and can interrupt this broadcast transmission makes subsequent low-priority radio receiver cannot receive broadcast, we use SMS broadcast priority interception technology which sets broadcast receiver for maximum priority: android: priority=“1000”, and call abortBroadcast() function in onReceive() method to suspend broadcast to spread.

Finally, in view of possible messages interception failure situation, using triple des encryption algorithm to encrypt message contents, such even if illegal phone owner sees command messages, they will not suspect because of garbled message contents. In the first place, a heavy des encryption operation puts generated random number object, selected operating mode and encrypted password as parameters of getCipher() method to generate one cipher class instance, and then we perform decryption operation on this instance; Finally, a heavy des encryption algorithm is applied to final realization of triple des encryption operation, note that sequence of their key are in reverse order.

Information Acquisition Module. The realization of information retrieval function requires mobile phone security software to access documents of messages、contacts、sd card and other applications. But Android sandbox mechanism achieves the mutual isolation between different applications and processes, namely, by default, the application does not have permission to access system resources or other application resource, each app and system processes are assigned unique and fixed user id and run in the independent dalvik virtual machine with independent address space and resources, any application if you want to access system resources or other application resources must be statement permissions in its own manifest file or shared uid.

So in order to break Android sandbox mechanism to achieve data exchange under different applications, this module uses sandbox breakthrough technology which makes use of external database access interface provided by contentprovider and takes advantage of contentresolver class to complete operation standard defined by contentprovider to access data between programs [12].

System Functional Test

In order to test and further improve design scheme of mobile phone security software, now testing basic functions of mobile security system under real machine and simulator two experimental environments, wherein real machine environment uses Android 4.4 huawei honor 6 smartphone, and simulator platform loads Android 4.4 system. The experimental contents and results are shown in table 1.

Table 1 Experimental contents and results

Experiments	Function tests	SMS commands to send	Test results
Experiment one	Delete the contacts	01	Good
Experiment two	Get the latest records	02	Good
Experiment three	Silently monitor	03	Good
Experiment four	GPS position	04	Good

On the simulator test platform, master phone sends “01” command message to controlled mobile phone, and controlled mobile phone will call delete() method of contentresolver object to automatically delete all contact data after received message. Controlled handset simulator status before and after contact information destroyed are shown in figure 3.

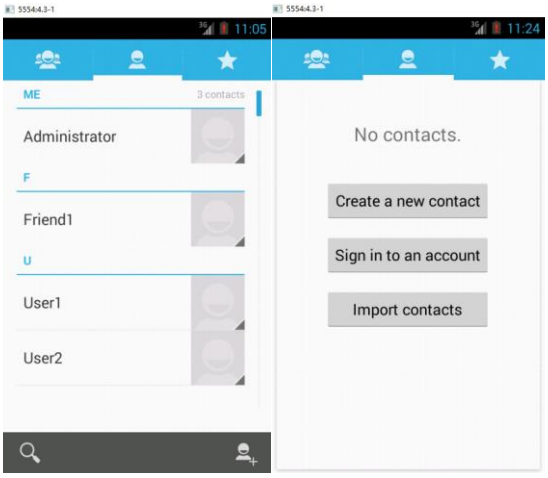
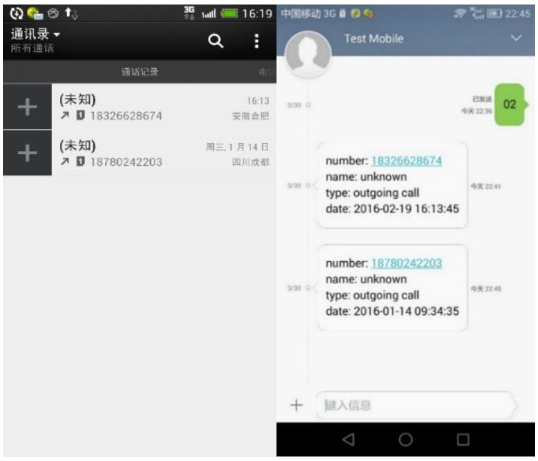


Fig. 3 Controlled handset simulator status before and after contact information destroyed

In the real machine test environment, master phone sends “02” command message to controlled mobile phone, and then controlled mobile phone will use ContentProvider.query() method of contentresolver object to query call records, finally returns to one Cursor object. Get the latest call records function test results are shown in figure 4, wherein the left side is controlled phone and the right side is master phone.



02

number: 18326628674
name: unknown
type: outgoing call
date: 2016-02-19 16:13:45

number: 18780242203
name: unknown
type: outgoing call
date: 2016-01-14 09:34:35

Fig. 4 Get the latest call records function test

On the simulator test platform, master phone sends “03” command message to controlled mobile phone, and then controlled mobile phone will call to master phone, specifically, setting phone number to call by instruction of Uri uri=Uri.parse(“tel:”+telStr), specifying Action type of Intent object is equal to ACTION_CALL to represent direct dial action, finally calling startActivity() method to launch system dialer; or automatically using mediarecorder class to record conversation content and sending it to network server when telephone connected [13]. Silent monitor function test process is shown in figure 5, wherein from left to right in turn are outgoing call、incoming call、call connected and call end four states.

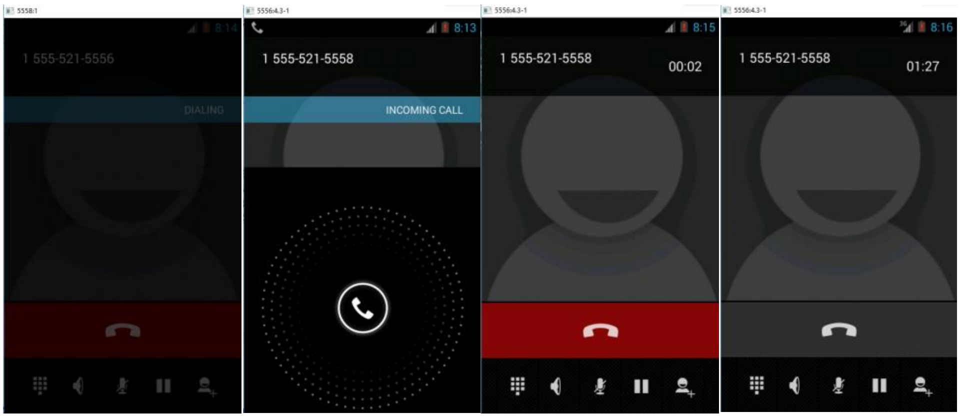
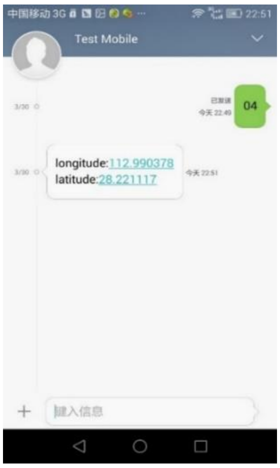


Fig. 5 Silent monitor function test

In the real machine test environment, master phone sends “04” command message to controlled mobile phone, and controlled mobile phone will call Context.getSystemService(Context.LOCATION_SERVICE) method to get locationmanager instance, after that calling getLastKnownLocation() method to get Location object which contains longitude、latitude、altitude and a series of location informations, finally using geocoding API with reverse geocoding to obtain intuitive location information [14]. GPS positioning function test results are shown in figure 6.



Test Mobile

04

longitude:112.990378
latitude:28.221117

Fig. 6 GPS positioning function test

Summary

In this paper SMS backstage monitor technology is used to achieve remote SMS management function of Android mobile phone anti-theft system, adds software parameter protection module、self-startup module and SMS encryption module on the basis of original mobile phone security software to enhance its functionality. The experimental results show that the system is designed reasonably and effectively, security software which meets expected anti-theft tracking function runs normally and has a certain practical significance and utility value for protecting user privacy and property security.

References

[1] Rogers R, Lombardo J, Mednieks Z, et al. Android Application Development - Programming with the Google SDK.[J]. Wiley & Sons, 2012.

[2] Kang H Y, Fan Y. Intelligent firewall for Android smartphone[J]. Journal of Beijing Information Science & Technology University, 2014.

[3] Yan M, Peng X G. Permission detection system based on android security mechanism[J]. Computer Engineering & Design, 2013, 34(3):854-858.

[4] Liu X, Jiang Z. Mobile-phone Customization based on Information Security[J]. Communications Technology, 2013.

[5] Gu Q, Li J, Gong X. DESIGN AND IMPLEMENTATION OF ANDROID SMART PHONES-BASED PRIVACY MANAGEMENT SYSTEM[J]. Computer Applications & Software, 2014.

[6] Yeh K H, Lo N W, Fan C Y. An analysis framework for information loss and privacy leakage on Android applications[C]// Consumer Electronics (GCCE), 2014 IEEE 3rd Global Conference on. IEEE, 2014:216 - 218.

[7] Mittal P, Dhruv B, Kumar P, et al. Analysis of security trends and control methods in Android platform[C]// Computational Intelligence on Power, Energy and Controls with their impact on Humanity (CIPECH), 2014 Innovative Applications of. IEEE, 2014:75 - 79.

[8] Zhang F, Jiang B, Huang J, et al. Android-based Personal Cloud Secure Storage System[J]. Science Technology & Engineering, 2012.

[9] Hao M. Mobile Phone Anti-theft and Obtaining Evidence Based on Android[J]. Research & Exploration in Laboratory, 2014.

[10] Umamaheswari M, Devapriya S P, Sriya A, et al. Android Mobile Security with Auto boot Application[J]. International Journal of Engineering & Technology (0975-4024), 2013, 5(3).

[11] Zhang H, Chen S Y. Implementation of Mobile Phone Anti-theft Tracking Based on Android[J]. Jiangxi Science, 2011.

[12] Jiang Y, Huang H. Security Solution for Mobile Anti-theft System Based on Android[J]. Microcontrollers & Embedded Systems, 2013.

[13] Han-Fei M O, Wang C D, Feng C R, et al. Research for Mobile phone anti-theft tracking and privacy control system based on android system[J]. Journal of Tianjin University of Technology, 2014.

[14] Cai L B. Intelligent Mobile Device GPS Positioning System Design and Implementation Based on Android[J]. Computer Knowledge & Technology, 2012.