Content Retrieval and Protection of Smartphones through Remote Access

Ashish B. Nair, Ishita M. Raut, Sangeeta Joseph, Prof. Dipali Koshti

Abstract— From being a gadget of luxury and sophistication, smartphones have gone on to become a broad-based phenomenon in the mobile phone market. Today, there are around 1.91 billion smartphone users in across the globe. The dramatic growth in the usage of smartphones is driven by a desire among users to stay connected and have instant access to social networking sites. With smartphones becoming an indispensable tool, the imminent need for mobile protection is increasingly important over the past few years. As annoying as it is to lose their mobile phones, a major percentage of the victims considered the loss of contact information and exposure or loss of private information the worst part of the experience and also a huge inconvenience.

This Project proposes and implements a low cost mobile tracking system and provides an option to access the smartphone using another phone using SMS (Short Message Service) and/or GPRS (General Packet Radio Service).

Index Terms— Data Retrieval, GPS Tracking, Mobile Security, Remote Access, SMS

I. INTRODUCTION

India ranks third among the top countries for smartphone users with an estimated 117 million subscribers [1], behind only China and the US, to grow 45% in 2014. The number of users using internet in India is 243,198,922 [2] (present number) which is around 8.33% of the world's internet users. Smartphone's are now at the peak of popularity in their usage of accessing the internet which includes mail access, social networking, mobile shopping and mobile banking. Along with smartphones comes the need to protect them. Your cell phone is lost or stolen, now what do you do? A stolen cell phone is more than just the absence of the device itself. It's the loss of important information and the potential for identity theft.

Given these features the problem arises when the phone is left at someplace and we want to retrieve some data from the Smartphone. Remote accessing of mobile becomes necessary in such cases. To deal with this problem, we thought of developing an application that permits remote accessing of necessary data in the lost phone. It includes getting the incoming call numbers, incoming messages, accessing call logs, changing phone's profile setting, locking the phone as well as retrieving the contacts.
This application is designed for android users and is useful in the following scenarios:

Manuscript received April 20, 2015.
Ashish B. Nair, Department of Computer Engineering, Fr. Conceicao Rodrigues College of Engineering, Mumbai, India.
Ishita M. Raut, Department of Computer Engineering, Fr. Conceicao Rodrigues College of Engineering, Mumbai, India.
Sangeeta Joseph, Department of Computer Engineering, Fr. Conceicao Rodrigues College of Engineering, Mumbai, India.
Prof. Dipali Koshti, Assistant Professor, Department of Computer Engineering, Fr. Conceicao Rodrigues College of Engineering, Mumbai, India.

1) You have misplaced you phone and can't find it. There are two possibilities to this situation that either the phone has been stolen or forgotten somewhere. With the help of this app, you can track the location of the phone via a friend's smartphone/normal phone.
2) You reach your office and realize that you have forgotten your phone at your home. Need to urgently call a client whose phone number is stored in mobile. Not a problem, with this app on your phone you can easily access and retrieve the address book of your phone remotely from yours or your friend's any model cell phone. This app will take the effort to find your required number and SMS it back to you on your sent mobile number.

II. LITERATURE SURVEY

With the advent of new technology the way of communication is also changed. Mobiles are the latest invention and common way to communicate now-a-days. Mobile phones have evolved from simple voice terminals into highly-capable, general-purpose computing platforms. While people are becoming increasingly more dependent on such devices to perform sensitive operations, protect secret data, and be available for emergency use, it is clear that the security aspects of mobile devices are another important research area with many research models in it.

Remote phone access [3] is an android application which allows the user to track the mobile using any browser. It also allows many other facilities like;

- To upload or download any file from the mobile phone
- To change the setting i.e. Bluetooth, Wi-Fi, Sync, and GPS
- To check call logs
- To reset application password

It also allows the user to send a SMS, retrieve contacts, and lock the phone remotely; but are the paid features of the application.

Another Application named Remote Phone [4] enables the user to manage the android from any browser directly. Remote Phone requires a Dropbox account to use the app. The application enables the user

- To track the location of the mobile phone
- To retrieve contacts
- To send and receive SMS
- To make a call

The paid features of the application allows to lock the device, erase the data; change the ringing mode, change the Wi-Fi and

Bluetooth settings, and allows the user to start and access the audio recording.

III. PROPOSED MODEL

Many existing system were studied [3],[4] and almost all of the required a browser to access the lost phone. The proposed model is a low cost tracking system made using the latest technologies, to make it available to the common people owning smartphones. The model is a mobile application for Android smartphone users with a plethora of features which helps the user to remotely access his phone and also track it. The application must be installed on the phones for it to work. The basic idea of this project is to track the lost phone and retrieve data either via SMS or Internet.

The various features that the proposed system will offer are as follows:

1. Track the smartphone
2. Lock the device
3. Change the ringing mode i.e. normal, silent, vibrate
4. Change Wi-Fi, Bluetooth, Mobile Data, GPS settings
5. Erase Gallery data
6. Retrieve contacts
7. Retrieve and delete SMS
8. Retrieve and delete call logs
9. Take a picture from front and back camera
10. Record audio
11. Activate a siren
12. Message and call alert
13. Send an on-display message to the phone
14. SIM card change notification

IV. SYSTEM ARCHITECTURE

The model has two components, the client and the server integrated into one application. The user can access the services by logging in as a client or as the server. The server is a dedicated component to run on smartphone which has to be remotely accessed and protected and the client component is another smartphone or a normal phone which is used to access the server component [5]. In the case of normal mobile phones, users will only communicate directly through SMS. So the server handles the request from the client smart phone as well as from normal phones. In a case where both the components are smartphones, both should have the proposed system installed.

The block diagram of the server component is shown in Fig. 1 [5]. The SMS listener and parser in the server will receive all the incoming messages and check for the command in the SMS. After authentication, SMS command received will be processed if it is from the authenticated number. There, the request handler will process the command and response will be made. Call handler will read the incoming call number and inform the client number by the call alert SMS. Boot handler is used to start the application as the phone is booted. Changing of SIM card in that smartphone will be detected by the boot up listener and will be informed to the remote user.
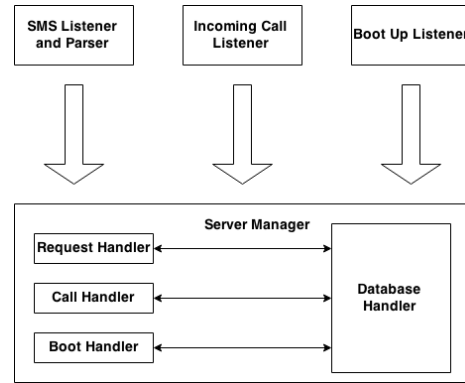

Fig. 1 Architecture of the Server

Similarly the block diagram of client component is shown in Fig. 2 [5]. The SMS listener is used to confirm the authentication process and to retrieve response of various services. The interface is used to send the command and display the result of various commands.
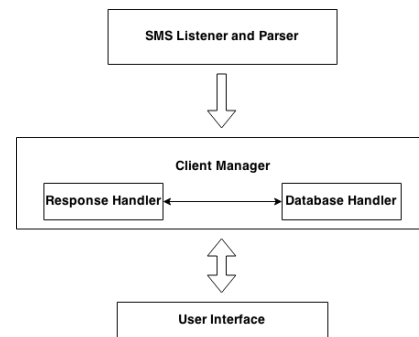

Fig. 2 Architecture of the Client

The list of commands used in this application are showed below:

Table I: List of Commands

COMMANDS	DESCRIPTION
%%connect	Establishes a connection between client & server for a client smartphone
%connect	Establishes a connection between client & server for a normal phone.
%%lock password	Locks the server phone
%%contacts name_of_contact	Retrieves contacts
%%delcontact name_of_contact	Deletes contacts
%%getmessages count	Retrieves messages
%%deletesms sms_id	Deletes messages
%%sendsms cell_number message	Sends message to the cell number via server phone
%%log no_of_logs	Retrieves call logs
%%deletelog log_id	Deletes call logs
%%location	Retrieves the co-ordinates of the current location of the server
%%backpic	Takes a picture using the back camera

%%frontpic	Takes a picture using the front camera
%%record	Records voice
%%bluetoothon	Switches the Bluetooth on
%%bluetoothoff	Switches the Bluetooth off
%%dataon	Switches the Mobile data on
%%dataoff	Switches the Mobile data off
%%siren	Activates a siren on the server phone
%%malert	Notifies client about incoming messages
%%calert	Notifies client about incoming/missed calls
%%display message	Sends an on-display message to server phone
%%callforward cell_number	Forwards call to the cell number provided

V. IMPLEMENTATION

A. Authentication and Connection Establishment

The first step that the application does after the user enters his credentials is authentication. The actual password for user authentication is entered by the user at the time of installation of the server application. The password is converted into the message digest by using MD5 algorithm [6] and is stored in the database.

For a smart phone when the client wants to establish a remote connection, he/she will enter the password in the form field. The password is encrypted using DES encryption algorithm in the client smartphone. The key is generated randomly. The encrypted password and the key is sent to the Server via SMS. The Server decrypts the password with the help of the key in order to get the original password. The original password is then converted into a message digest using MD5 algorithm [6] and is compared with the message digest stored in the database. Fig. 3(a) below depicts the flow for user authentication for smartphone client.

For normal phones the client enter the command %connect with the password and sent to the server phone. Similarly the password is retrieved at the server side and converted into message digest and is compared with the stored message digest.

If authentication is successful then it will respond to this SMS by switching on the internet connection if it is off. Once Internet is on, it will create a socket for connection [7] and returns a '%%connect' SMS along with its IP address for a smartphone client. The client will connect to the socket with the help of the IP Address and connection is established. Once the connection has been established, the user can access all the features of the application. Fig. 3(b) depicts the process of establishing a connection.
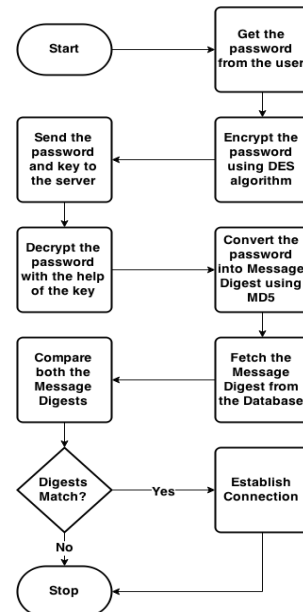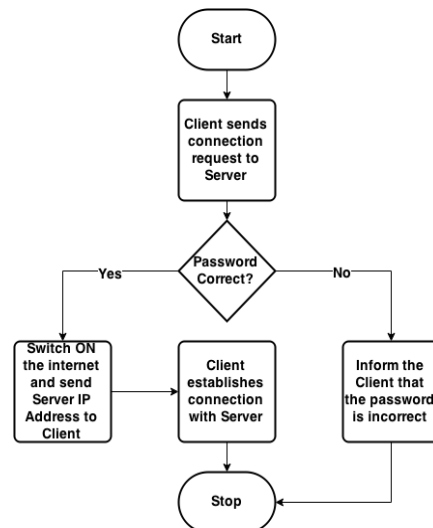

Fig. 3(a) Flowchart for Authentication

Fig. 3(b) Flowchart for Connection Establishment

Fig. 4 is the screen shot of client smartphone after authentication and connection establishment.
The 1st option is for location tracking, then message, settings, call and log, snapshot, recording, lock and extras i.e. ringing modes and siren etc.
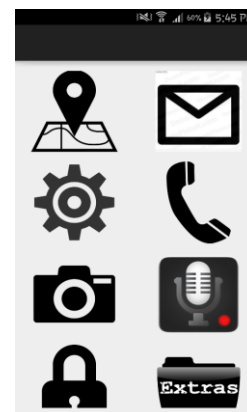

Fig. 4 Menu Screen

B. Lock Screen

Security is of the highest concern when it comes to smartphones since we store all the important data in our phones. With this application, users can remotely lock their phone so that their data is safe. If the user is accessing his stolen phone from a normal phone, he/she needs to send an SMS to his smartphone with the command '%% lock password'. In case of a smartphone, the user just has to click on the 'lock' button provided in the application.

C. Contacts

With this feature, the user can retrieve his/her contacts and view them. He/she can also wipe out all the contacts from his stolen phone remotely. The contact name and telephone numbers are sent over remotely and displayed on the client phone's screen. The command for contact retrieval is '%%contacts name_of_contact'. This command will give all the contacts containing the string name_of_contact. For a smartphone client, he/she has an option of searching contacts. Fig. 5 shows the screen shot for contacts retrieval where the contact to be searched was entered as 'Dad'.
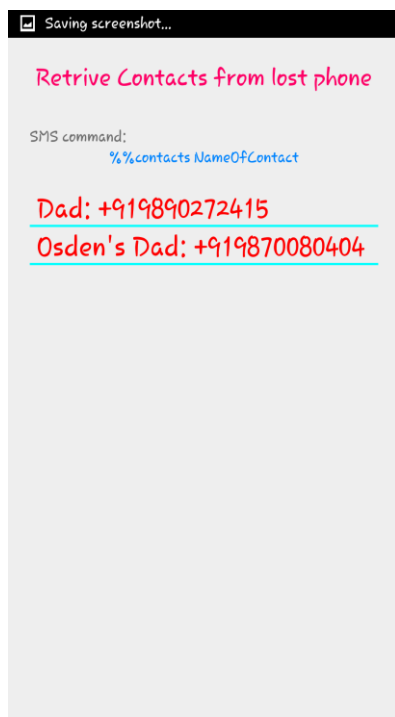

Fig. 5 Contact Retrieval

D. Messages

The user can retrieve messages and delete them from the server phone using this feature. A normal client has to type the command '%%smsget number_of_messages' to retrieve them. This command display the SMS ID, the sender's number and the content of the message. A smartphone client just has to type the number of messages he/she wants to retrieve and they'll be sent over the internet. The user can delete messages from the server phone using the SMS command '%%delsms smsid' which will delete message pertaining to that id. Each SMS has a unique id. The id of the message can be known using the %%smsget command. In a smartphone client the user just need to select the message that needs to be deleted. Fig. 6 is the screen shot of the list of retrieved messages.
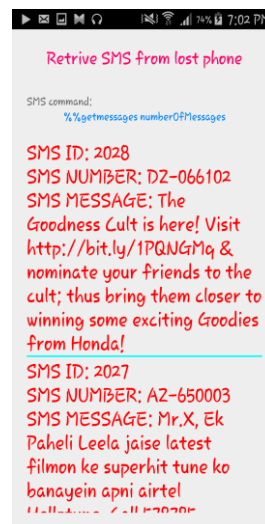

Fig. 6 SMS Retrieval

E. Call Logs

This feature works similar to retrieving and deleting messages. The normal client must type '%%log' to retrieve his/her call logs and '%%dellog' to delete them. A smartphone client has to enter the number of call logs he/she wants to retrieve and then click on the 'retrieve logs' button. The content displayed on the smartphone client consists of the phone number, contact name, type of call i.e. missed, outgoing or incoming, call date and time and the call duration. Fig. 7 is a screen shot of the list of retrieved logs from the server's phone.
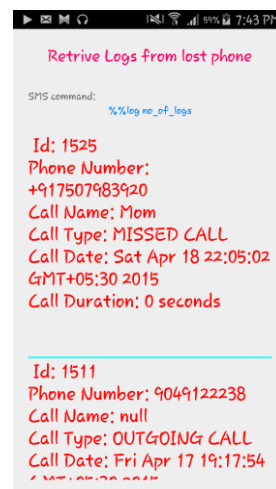

Fig. 7 Call Log Retrieval

F. Tracking

Tracking the server phone is the first thing a user will be in order to search for it. This application provides the user with live tracking every time he/she attempts to track the server phone. Once the server phone gets the request from the client for tracking, it sends its current coordinates back to the client and then displays it on the map. A normal client must send a message to the server using '%%location' to receive the coordinates and see it live on the map. A smartphone client just needs to click on the 'get location' button to receive coordinates. Fig. 8 is a screen shot of the map where the drop like figure is the location of the lost server's phone and the blue dot is the location of the client's smartphone.

Fig. 8 Tracking

Fig. 9 Settings

G. Snapshots

This is one of the most important features of this application. This feature allows the user to take snapshots using the front and/or back camera of the server phone, remotely. This feature will help identify the thief or the surroundings where the user's smartphone is in. A normal client needs to send a message with the command '%%backpic' for using the back camera and '%%frontpic' for using the front camera. The user will receive the pictures as an email. A smartphone client just needs to click on the buttons provided in the application and it will receive the images in his phone.

H. Record Audio

This feature enables the user to record sound in the server's surroundings. It will help the user in terms of security and he/she can provide the police with the recording to help them identify area or the thief. Using the command '%%record', a normal client can get the recording. A smartphone client just needs to click on the button 'record audio' to receive it. The audio recording is of the duration of ten seconds.

I. Ringing Mode

With this feature, the user can change the ringing mode of the server phone. The user can change it to silent if he does not wish for the server phone to ring or he/she can change it to ringing mode. The user can also change from vibration to off vibration mode and vice versa. The commands for changing the ringing mode and illustrated in table 1 given above. Smartphone clients need to just click on the necessary buttons provided in the application.

J. Settings

A few settings like bluetooth, mobile data, WiFi, GPS, etc. can be toggled using this application. The user can switch on or switch off these settings as per his/her wish. The commands to toggle these settings are illustrated in table 1. Fig. 9 shows how the settings can be toggled in the server phone.
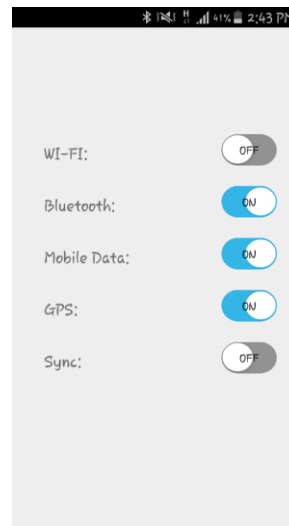
K. Wipe Out Data

This feature is extremely handy as it wipes out all the data from the server phone. It ensures that no one will have access to the user's photos, videos, audios and other media. A normal client can send the command '%%wipeout' to erase all data. A smartphone client should click on the button saying 'wipe out data'.

L. Siren

A siren is used in cases where one needs to alert surrounding people regarding something. If a smartphone is stolen and a siren is activated, the people around will realise that something is wrong and there is a probability that they might report it. Another case might be when the user has lost his/her and wants to check if it's in his/her vicinity. In this case, a siren will help the user find his/her phone immediately. User can activate the siren on the server phone for a period of 20 seconds by sending a command '%%siren'.

M. Message and Call Alert

This feature allows the user to receive message and call alerts to the client phone. If there are any incoming messages or calls, it will be notified to the user. In case of an incoming message, the user will be sent the details of the message along with the sender name and number, date and time. If the server phone receives any calls, it will be notified to the client along with call details such as the contact name (if any), contact number, date, time and the duration of the call, if it was picked up. A normal will have to send the command '%%malert' for messages and '%%calert' for calls.

N. On-display Message

An on-display message can be sent to the server phone using this application. A normal client must send a '%%display message' command to the server. The 'message' field will contain the message that the user wants to display on the server's screen. This feature could be helpful to inform a particular person that the phone he found is a lost phone and he needs to contact the owner.

O. SIM card change notification

This is another important feature of the application. Usually when a phone is stolen, the thieves tend to discard the SIM and put another one. Thus, it will become very difficult to locate the phone. This feature will notify the user on the client phone as soon as a new SIM card is inserted and activated in the server phone. The user will also be notified about the new SIM card number.

VI. CONCLUSION

The proposed model has successfully been implemented for android smartphone users. The model has been tested on android versions Jelly Bean, Kitkat and Lollipop [8]–[10]. This application could be useful in many situations and doesn't require a browser like the other existing application do. The only feature of the application being installed on the phone even after it is factory reset is the future scope for the project.

ACKNOWLEDGMENT

REFERENCES

[1] Survey about the number of smartphone users in India: http://techcircle.vccircle.com/2014/05/29/

[2] Survey about the number of internet users in India: http://en.wikipedia.org/wiki/List_of_countries_by_number_of_Internet_users

[3] Remote phone access: https://play.google.com/store/apps/details?id=com.goupilapps.remoteaccessandroid

[4] Remote Phone: https://play.google.com/store/apps/details?id=lovetere.remotephone

[5] K.S. Kuppusamy, Senthilraja.R, G. Aghila, A Model for Remote Access and Protection of Smartphones using Short Message Service, International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol.2, No.1, February 2012

[6] MD5 Research: Zhao Yong-Xia ; Inf. Eng. Sch., Wuhan Univ. of Sci. & Technol., Wuhan, China ; Zhen Ge. Published in Multimedia and Information Technology (MMIT), 2010 Second International Conference at Kaifeng

[7] Socket Programming: LimiKalita. Published in (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3), 2014, 4802-4807

[8] Android SDK: http://developer.android.com/sdk/android-2.3.html

[9] Android Emulator: http://developer.android.com/guide/developing/tools/emulator.html

[10] Operating System: http://en.wikipedia.org/wiki/Android_%28operating_system%29