IMPLEMENTING SECURITY ON ANDROID APPLICATION

1Kirandeep, 2Anu Garg

1Lovely School Of Computer Science and Engineering, Lovely Professional University, Chaheru, Punjab
2Lovely School Of Computer Science and Engineering, Lovely Professional University, Chaheru, Punjab

ABSTRACT

Android is an open platform which is becoming very popular operating system. Its open source code is easily handled by the users to get and use new contents and applications on their handsets. With the increasing popularity of these smartphones, additional privacy protection to these devices is required. Android is more flexible to become attractive targets for malicious attacks due to significant advances in both hardware and operating systems. Because malware on device can create number of risks, which creates problem while connectivity because of security issues. In this paper, it will be described that how security can be improve of Android Operating System so that users can safely used the android smart phones. In this thesis, I have analyzed the security goals of the Android operating system and tested its security. The thesis also contains a discussion about how secure the Android system is and how much trust can be placed on it while using it.

Keywords: Android; Dalvik Virtual Machine; Security; Encryption; Decryption; Open Handset Alliance

1. INTRODUCTION

The OHA allows phone makers to run Android on a suitable handset, without charge. In September 2008, T-Mobile released the first smartphone based on the Android Platform as well as a Software Development Kit (SDK). In October, the source code was made available under Apache’s open source license. The company released the platform’s full source code immediately after the first device hit the market. It allows developers to write managed code in a Java-like language that utilizes Google-developed Java libraries.
has released tool i.e. Google apps that implement under some security policies. There are so many facilities like password protection also implement in Android smart phones. Android is Linux based operating system. The architecture of Android operating system is designed in such manner so that communication at application level and end user will be quite easy. Android applications are written in Java, a programming language. But Android has its own virtual machine i.e. DVM, which is used for executing the Android applications. Designing of Android application is easy as compared to other applications of Iphones. Android was created in October 2003 by Andy Rubin, Rich Miner, Nick Sears, and Chris White. On August 17th, 2005 Google purchased the company for around $50,000,000 and all the founder went to work for Google. The unveiling of the Android platform on 5 November 2007 was announced with the founding of the Open Handset Alliance, a consortium of 34 hardware, software and telecom companies devoted to advancing open standards for mobile devices.

1.1 ANDROID FEATURES:

- Android code under the Apache License, a free software and open source license.
- A very important feature of Android OS is that it is open source nature, develop new application or update existing application.
- Each Android app runs within its own virtual machine and each virtual machine is isolated in its own Linux process.
- Each app is given unique user and group IDs
- All applications have full access to phone capabilities.
- All applications are permissions-based.
- It allows access to core mobile device functionality through standard API calls.
- A powerful SDK is available for development that contains libraries, tutorials, sample code and emulator.
- Should have no costs for using the platform, develop applications for the platform or publish own developed applications.

2. REVIEW

Burns\textsuperscript{[12]} et al., cellular phones are used to discuss sensitive personal and business information. In end-to-end encryption Android phone calls, current encrypted phone call solutions for this problem require an internet connection for VOIP or special handset. In this paper, architecture of encrypt phone calls as an addition to Android smart phones. In these days, there are so many applications and much information which is used to provide information to the user. For example, banking applications, any business details and also health information are easily provided by the users. So, in these days technology has moved forward, so there is need to provide security to the user. This paper presents a method for adding an encrypted communications stack to Android. Red phone from whispersys is Android based products which provide encrypted phone calls. But to access this application, internet facility is required. While communicating through Red phone from whispersys, it requires trusted central server for the secure communication. But there are security problems during as Android Phone calls. These all are explained as follows:
a) **Security problems in Android Operating System:** In Android operating system, there is Dalvik JVM and also sandboxing which is included through Dalvik JVM; it restricted tasks that required user consent at application installs time. With this security, Android has many related exploits granting root permissions. Although Google is in charge to distribute multitude of phone manufactures and wireless carriers, who must provide users with a patch after Google creates it.

b) **Security Problems in Cellular Networks:** When user communicate with another via phone call or any other source, then it is very important to secure the data over the network. So that unauthorized persona cannot listen or get any important information. Like in phone, when user communicate with another one then GSM which includes encryption is used. Unfortunately, these encryption schemes have long history of being adequate for dependable privacy protection.

c) **Security problems in Telephone Networks:** Cellular networks revert to unencrypted phone network outside of handset-to network link. If the path or network is more trusted then there will be no issue to communicate with one another. But here the network provider is not so trusted; this lack of security through phone calls can be a problem. Here is an Android telephony in which many components are used when user communicate with another one on Phone calls. The block diagram of Android Telephony is shown as below and explains its components:

![Block of Telephony Components](Burns I)

In this block diagram of Android Telephony, there is RIL (Radio Interface Layer) which starts interaction above baseband. Lib hardware contains various functions which are used to interact with GSM network properly. There are so many actions like dial numbers, hang up calls, accepting calls and also rejecting calls etc… The RIL library performs call backs into file. For this when request occur then Android packages that contain various classes dealing with controlling the phone. After accepting request, here are various commands which deal with the control state of phone calls.
Audio system and audio flinger does direct routing by default. Audio stream can be used for any input or output to/from any audio applications like speakers and microphones etc.

There are steps with which call is placed and the actual digital voice stream is set up by low level code provided vendor. When call is established, then phone puts in MODE_IN_CALL and then if Bluetooth or wired headset is connected, then Android perform phone call action. In Android Phone, it can’t handle phone calls by itself, in Android Phone; there is dumb phone which connects phone call through baseband and baseband responds when it is in phone call. According to I. Burns et al., there are so many difficulties while implementing this scenario, which are as follows:

a) Another difficulty is that the implementation also depends on data which is being sent through phone network that reproduces the original data bit-for-bit. If data lost then it is very difficult to recover that encrypted data.

b) Another difficulty is that to access the secure data, there is need of correct permission for all actions need to be located. With cellular, wired and international phone networks involved, there is potential for significant latency in the transmission system.

c) Another difficulty is that as phone call is a real world application, encryption and transmission of data must be more timely than web browsing. The battery life is also concerned for the encryption of phone calls. For encryption, CPU utilization and power consumption are also very important to access the encrypted data during a call.

![Fig 1.2 Structure of Placing Phone Call](Burns I)
Fig 1.3 Structure of placing Encrypted Phone Call (Burns I)

In this diagram, the structure of placing an encrypted phone call will have its voice routed through the encryption module in the library layer. As per conclusion, I. Burns et al., this is sufficient to provide security but still an unauthorized person who knows about encryption can easily break that cipher.

3. PROPOSED MODEL FOR ANDROID APPLICATION

In this section, we will be describing the methods that we have proposed to develop the application. The flowchart describes the whole process of creating the application.

Application isolates in a sandbox environment. Each application executes in its own environment and is unable to influence or modify execution of any other application. After creating an application, each file is packaged in .apk format, which is Android package archive for installation. .apk file holds images, manifest for the application.

Android is sandboxed which means that each application executes within its own virtual machine. Sandboxing has features that one application cannot modify the data of another installed application.
3.1 CREATE THE APPLICATION

In this section, we will be discussing the steps that are used while developing the application. Get started on Android 2.2 and a MinSDKVersion of 4, which is provided with a valid name, package, and activity. The idea was to protect our Call Logs from attackers. Because if phone is in wrong hands then anyone can easily get the contacts numbers and misuse that sensitive data. The main objective is to provide facility to secure our information regarding the malicious behaviours.

Develop the Framework

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Name</td>
<td>Implementing Security On Android Application</td>
</tr>
<tr>
<td>Build Target</td>
<td>Android 2.3.3</td>
</tr>
<tr>
<td>Application Name</td>
<td>Call Log Encryption</td>
</tr>
<tr>
<td>Package Name</td>
<td>Com.CallLogEncrypts.activity</td>
</tr>
<tr>
<td>Create Activity</td>
<td>SplashScreen</td>
</tr>
<tr>
<td>Min SDK Version</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 1-1 Application framework

![Flowchart- Application Development Framework](image)

Fig 1.4 Flowchart- Application Development Framework
This table described the framework used in developing the application. For creating this application we use simple Android components and use some imported classes. Set up all the tools and frameworks necessary to develop the application. It is necessary to secure our sensitive information before falling into mislead hackers. In order to prevent our information we used some encryption and decryption method which is compatible with Android that will ensure the device and their valuable data it contains are secured against unwanted use or intrusion.

### 3.1.1 DESIGN OUR APPLICATION

For designing an application, firstly we create some activities which are the interaction part of application where any user can interact with the application. Different activities serve for different reasons for that we used different Android widgets like simple buttons, view, edit boxes, text and dialogbox etc which made the application user friendly and easy to handle. Splash screen is working for meaningful information regarding our application via a single screen without any user interaction. When splash screen comes in front one can easily understand about the application as it is running on the main screen. After splash screen main activities comes which gives us a user interface.

### 3.1.2 TEST THE APPLICATION

After developing the application we have to test it on different devices and it must be compatible with every device. Once our application is running on the emulator, it can use the services of the platform to invoke other applications. For testing purpose we used minimum SDK version which is 2.3.3 for our application. For compatibility with every device, we have mentioned its information in application’s manifest file. This is a core file of all type of information which is used in the application.

### 3.1.3 APPLICATION CALL LOG ENCRYPTION

The recent calls which are stored in call logs. Sometimes, phone lost then hackers can easily hack the data from the phone. From this stolen data, call logs is one of them. Sometimes, attacker can attacks on calls and misuse of these calls. We can say that if phones are on wrong hands then anyone can check the phone details like SMS, contacts, gallery etc... Malware usually destroys valuable and sensitive information in infected systems. Android developers upload their applications to Android official market that sometimes exploit to their infected devices by compromising their privacy. Thus our main focus is to prevent Android applications from performing illegitimate actions that may lead to user’s loss. If any malicious content or hacking takes place over there, our data get destroyed or will be not remain clean to get information from that infected data. That’s why we used here encryption and decryption method to protect our data from outside content so that not even a hacker can hack the data. When we transfer our data and at receiver end data get separated through same cipher data. Only receiver and sender can get pure data because only they know the key. In this section, we will be describing the snapshots of our developing application. We will discuss the functionality of each snapshot one by one.
Fig 1.5 Welcome Screen

It is welcome screen of Android application. After this login form is displayed in which user fill its username and password and then login successfully into application by entering credentials.

Fig 1.6 Login Screen
In Fig 4.2, it is a login screen in which if user fill wrong credentials then it will displayed message i.e. wrong credentials. If credentials match with username and password then it will successfully login by user.

In login screen, when user fill username and password when it matches with shared preferences then user can easily enter into application and use the application securely.

In Call Log Homepage, in which there are four buttons All Calls, Missed Calls, Outgoing Calls and Incoming Calls. User can click on any of the button and then open the list of contacts according to the selected button. E.g., if user select outgoing calls button then it will display the list of only outgoing calls by user.
When user click on outgoing calls then a new activity is opened and the list of all outgoing calls displays and now user can select any of contact from this list for encryption. And by clicking continue button it will goto new activity and after click on back button user can go back to homepage of this application.

In unselected list of outgoing calls is displayed. If user click on continue button without select any contact then it will display a message to select atleast single contact so that user can easily do encryption on that selected list.
When user select contact list with the help of checkboxes then after selecting contacts, user click on continue button and then a new list will be displayed in which only selected items are shown.

In Fig 4.8, selected contacts lists will be displayed on new activity. In this page only those contacts are displayed which user wants to encrypt on Call Log Encryption. In this page there are two buttons one is Encrypt and another is Decrypt, these both are used for encryption and decryption on contact list…
After selecting the contacts from Call log list, user click on Encrypt button then a new dialog box will be displayed in which user fill the password which he/she entered password at login time. It will match password from shared preferences then if password matches only then user can continue with encryption otherwise application will be stopped.

When user fill password, if it matches with shared preferences then if it corrects it will encrypt the whole list and update the contacts in encrypted form and a new list will be displayed on same list.
After encryption when user again want to decrypt the encrypted list then again it will asked for the password so that it can check that the user who uses this application is correct user.

When user filled the correct password after matching with shared preferences then encrypted list will be easily decrypted and a list will be update after decryption and original contacts will be displayed to the user.
4. CONCLUSION

In Android, To make application more secure user can use encryption and decryption algorithms so that it will become very difficult for the attacker to decrypt the data of application. In Android, there are many security measures, which have been applied to face the challenges of openness of smart phones. Attackers can easily capture the data through network. Malwares make capability to show difference between the previous and current threats to exploit the vulnerabilities. Sometimes users accidently gives the permissions to execute the malicious code because of awareness, so protect the application from these types of attacks we implement the application and provide security to that application through encryption and decryption algorithms and also protect the sensitive data to transfer securely without getting any privacy loss.

5. FUTURE WORK

For the security purpose, the work was carried out with aim to prevent the application by using encryption and decryption algorithms. I used to allow static user to enter in Call Log Encrypter application, but in future scope we can use SQLite database and some new services for entering the new users in the application and those new user’s database will save information with the help of SQLite database. And also when we encrypt and decrypt the data using encryption algorithms then we can also save that encrypted list in any location like Internal and External memory in SD Card and also we can save it on cloud. So that if users want his/her data back then user can easily retrieve that encrypted and decrypted data from any save location where they put that data like on cloud and in phone memory itself. So, the main purpose of this application is to provide the security to the user while using Android application.

6. REFERENCES

1) Zhou Yajin, Zhang Xinwen, Jiang Xuxian and W. Freeh Vincent "Taming Information-Stealing Smartphone Applications (on Android)", Department of Computer Science, NC State University.
4) Burns I, Gabert K, Zheng J “End-to-End Encryption Android Phone Calls”, Department of Computer Science and Engineering, New Mexico Institute of Mining And Technology, NM, USA.