Android Application For Secure File Transferring using Data Encryption Standard

Dattatreya Hadapad
M-techII year,GNDEC Bidar

Asst.Prof Steven Raj N
CSE Dept,GNDEC Bidar

Abstract

Android shook the world of smartphones. Which is created full of opportunity to embrace and make use of opportunity offered by the smartphones and this credit goes to the open model support by the Google. Unleashing the smartphone application market was also beneficial for end-users, while most of the application achieve their goals without abusing users privacy, an open and popular platform as android as perfect environment to exploit and disseminate security attacks.

To avoid Malicious application we are designing and developing an application which provides security for transferring the file/data with the help of DES algorithm, where it encrypts and decrypts the data with the secret key, here transmission is done in two ways, If the file/data is public then file is selected and sent to particular receiver, if the file/data is private it ask for security key and sent towards receiver, this way we achieve the secure transmission of information(file/data) between end-users.

Keywords: DES(Data Encryption Standard) ,DVM(Dalvik Virtual Machine), OHA(Open Handset Alliance).

1. Introduction

Thousands of Android phones are activating each day. The world of smart phones created an ecosystem which is full of developers where those developers are eager to embrace and make use of opportunity offered by smartphones and this credit goes to the open model support by Google, Developers are not forced to pay much for code certification fees or sharing significant percentage of their profit with application distribution points. The number of applications developed for android and the android smartphones have the similar growth.

The smartphones android application market was beneficial for end-users who can now choose among the large variety of applications. While most of the applications achieve their goals without affecting the users privacy, but some of them have

-recently arrived in the market to do the things in quite reverse way.

To overcome from such undesirable situations, An android incorporates security mechanisms and features that allow partial protection of user's privacy from malicious applications. However, developing an efficient and usable android security application model suitable for battery powered devices which is intended for use by a wide range of people is not so easy.

Google Android is a Linux-Based platform developed by Open Handset Alliance (OHA). Most Of the Android Applications are developed in Java and compiled in to a custom-byte code that is run by Dalvik Virtual Machine(DVM). In particular, each Android is executed in its own address space and in separate DVM. Android applications are built combining any of the following four basic components. *Activities* represents a user interface; *Service* executes background process; *Broadcast Receiver* are mailboxes for communication within components of the same application or belonging to the different applications; *Content Provider* Store and share application data application's data. Application components communicate through message called *intents.*

In this paper we are going to develop application model which provides a security for the file/data transfer with the help of DES algorithm where this algorithm encrypts and decrypts the data with the secret key, here transmission is done in mainly two ways, Private and Public, If the file/data is public then the relevant file is selected and sent towards the particular receiver with that receivers ip-address, if in case the file/data is private it asks for a private key and sent towards the particular receiver with their ip-address.

A. *Motivation*

Despite of existing attempts to address the privacy issue the solution proposed by MockDroid[7], TISSA [23], and AppFence[8] are to coarse-gained.

These approaches apply indiscriminate to the whole data of a content provider.

Our Main Motivation is to develop an application which provides a security to the users by avoiding the malicious applications to leak the user's private data. If the malicious application is leveraging a vulnerable of a legitimate application then this type of attacks is often referred to as confused deputy attacks. However, Developing application for the android market is quite simple (just pay the fee $25), Designing colluding applications that on purpose provide to other application permission Without the user being aware of it is become increasingly popular.

In This Paper We are developing an application for the purpose of providing the security to the user data or the private information of the End-User.

2. Related Work

In Case of Android Platform , permissions can be requested and granted only at installation, to remove this limitations there are many approaches have been proposed to address the problem of specifying and changing fine-grained policies during run-time or during selecting a private file/data.

Nauman et al. [10] and Conti et al. [11] proposed security extensions to support context-related policy enforcement at run-time. Bai et al. [12] has further extended the android security model to support part of the UCON model.

More recent papers [7],[8]Concentrated on the users private data. [7] Introduces a system which can limit the access of the installed applications to the data and the components of the Android OS. Applications are agnostic of these limitations. For instance, an application querying the contacts' provider may receive no results if the provider is not empty this process is widened by Zhou et al.[11]. Their work provides user with the ability to define the accuracy level of the information revealed to the application.

WU Feng-xiang, SUNXin-sheng, YUAN Ying-chum et al.[3] presented a paper on development and implementation of Eclipse-based file transfer for android smartphones where this illustrates to understand the communication between them via socket based on the platform.

Machigar Ongtang, Stephen Mclaughlim, William Enck, and Patric McDaniel et al [5] proposed Recent evolution of mobile technologies has opened new use of possibilities for mobile devices. At the same time information security problems related to mobile devices have become more serious. This study investigates a possibility of remotely and securely transferring files between the mobile devices and home computer via the Internet, and the information security threats related. This paper deals with a setup for secure remote file management. Security was evaluated with threat analysis and the file server was tested with security testing tools. Based on the evaluation and testing the system construction was considered to be secured when properly used. However new software vulnerabilities are discovered daily also for mobile devices there fore the security of the set up get week if that setup is not updated frequently. Also Changes made to Configuration file by the user poor management of login information and careless use of devices are the problems which cause information security threats.

To overcome from these problems we are generating an application for the sender and the receiver side so that there should be a secure file transmission should be done between two remote smartphone users through Wi-Fi. And this will be the much use full application in the corporate world like companies with a Wi-Fi where the employees can share their personal/private information or file with the particular end user. With whom he/she wants to share the private data/information.

3. Existing System.

The study of existing system helps for a new system to develop new by overcoming from the existing limitations. Currently there are no such systems available for file transfer in android smartphones. If any of the available systems use the TCP/IP protocol for communication and also for transferring of file from one device to the another.

Earlier Transfer of file between two android Smartphones has done with the help of Bluetooth

To the transmission area limited within the range the speed of transferring the file is less and there was no security mechanisms for users private data/file the receiver can directly access and install the data/file sent from the sender. The limitations of the existing system we proposed a new application which is having more area coverage range with the help of Wi-Fi and with considerably high file transfer between two devices with a security mechanism to protect the users private data/file leakage to the unauthorized end-user.

4. Proposed System

In our proposed system two hardware's are used which are two android based smartphones,

with the help of which Users can transfer any kind of data/file from one smart phone to another in a secured manner. One is known as sender and another one is known as receiver. First the sender application is

installed in one android based smartphone and the receiver installed in another android Based Smartphone. Then user who wants to send the data

have to open the sender application and have to select the file or the data which has to be sent towards the another end user, and the end users also should open the receiver application. Once the application opened in the sender's android smartphone the application will automatically search for the IP's in existing Wi-Fi range and list all the existing Wi-Fi based android smartphones IP's. sender has to select the particular IP-address's among the list of existing devices browse the file and transfer to the desired receiver end.

There are two transferring mode one is public and another one is private. if the user selects the public then the file directly sends to the receiver. for transferring file we use the medium which is Wi-Fi connection first will connect both the smart phones android devices and both smartphones or Android devices here java socket programming to connect these two sender and the receiver android devices.

While transferring the file, first sender connects with receiver using socket programming then sender should select the mode of transfer that is private mode or public mode if the mode is private then the application will ask for a secret key. For the purpose of security the DES algorithm is used and the length of the key should be 8-bits. here the DES encrypts the key in to unreadable form and then the sender sends the encrypted file/data in to unreadable form and sends towards the receiver where the file is decrypted only with the help of that 8-bit secret key. If the entered secret key is correct then the file is received and decrypts the file and save that file to its original form towards receiver end.

5. System Design

The purpose of the system design is to plan a solution for a problem specified by the required document this is the first step in moving from the problem domain to the solution domain. Here in this phase system block diagram will b built and that will be helpful to understand the behaviour of the system. And it is the process or art of defining the architecture, components, model and relevant software and hardware needs..

5.1 Proposed System Architecture

A system Architecture is the conceptual model that defines the structure behaviour and more views of the system, organizing in the way that supports reasoning about the structure of the system which comprise the system components the relationship between them and provides a plan for which product can be produced and the system developed that will work together to implement overall system.

The bellow diagram will represents the system architecture of our design methodology.

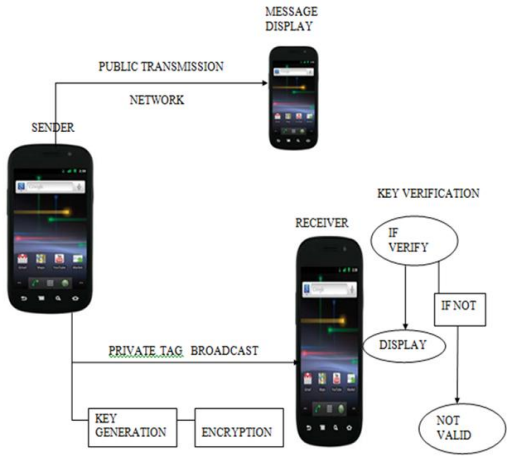

Fig 5.a System Design Architecture

The above diagram represents the system architecture, the main components of the system are android devices or android smartphones , transfer mode , GUI, encryption, decryption, scan IP ,and random key generation.

Here android based devices or smartphones with Wi-Fi are used as hardware which works as sender and receiver in which the user should have installed with relevant applications on their devices ,

Basically file transferring of the data contains two transferring modes

- Private
- Public

Public transmission is done without any security where we need browse a file/data from the source device to send no security key is needed for the public file or data. Once the transmission mode is selected as public and the relevant file/data is browsed from the source device we need to choose the IP-address from the randomly generated IP-address list of the receiver device to which the data is to be sent publically and the transfer the file to the receiver device.

In case of private transmission first we need to choose the file/data which is to be sent towards receiver side and then transmission mode as private after that the scan-ip process will generate the list of the device ip in that Wi-Fi range, from that we have to select desired

ip-address of the receiver device then the randomly generated secret key of 8-bit and The key verification process shows that if the key is valid (i.e key should be of length equal to 8-bit) then it displays and sent to the receivers , phone number. Otherwise if the key is not valid then it will show not valid message. The DES algorithm is used to encrypt and decrypt.

6. Implementation

The implementation of application is developed using the java programming in eclipse framework. The architecture can be implemented using the followings

- Programming language: Java
- Frame work: Eclipse
- Supporting tool: Android SDK
- Platform: Windows XP/Vista/07/08

The programming language is selected as java because it supports android, java is an platform independent language ,java is a portable language, and the java socket programming will allow the remote connectivity between two devices.

Platform as windows XP/Vista/07/08 is used to manage the computer resources. Where the operating system is used to perform the basic task such as controlling and allocating memory, prioritizing, system request, controlling the internal system resources as a service to user and programs of the system, The security concern are large and require that the system being developed to be robust and safe from attack windows 7 analyzes the performance impact of visual effect and uses this for networking and managing files. An operating system processes system data and users input to produce the desired output.

Framework uses the Eclipse IDE (Integrated Development Environment) is a multi-language software development environment compromising an IDE and an establishing plug-in systems. It is written in mostly in Java language and that can be used to develop android application in java and by means of various plugins and their programming languages including ADA , C,C++, Perl, PHP, Python, Ruby.. The IDE is often called Eclipse ADT for ADA, Eclipse CDT for C/C++, Eclipse PDT for PHP and Eclipse JDT for Java. The Eclipse SDK includes the Eclipse Java Development Tool(JDT) offering an IDE with a built-in increment Java compiler and a full model of the Java source files. This allows for advanced refactoring techniques and code analysis. The IDE also used for a work space , in case of the set of the metadata over a flat file space allowing external file modification as along as the corresponding workspace "resource" is refreshed.

6.1 Implementing Modules of File Transfer.

Implementation of file transfer mode is mainly divided in to four modules, each having its own functionality as follows.

i) *GUI(Graphical User Interface)*: This Module Gives the application layout for both sender and receiver where each field shows to browse the file transfer mode selection among the private and public, a scroll button to show the list of IP-addresses of the available Android devices in that particular Wi-Fi range, and a field which generate random 8-bit security key where we can select a randomly generated key and bellow this we have a text box where we have to enter the phone number of the specific end user to whom that secret key of the private file is to be sent. Finally in GUI we have a Transfer file option which will be used to send the file to the desired receiver-end.

The following functionalities are used to generate the required GUI,

Example codes:

```
Et1= (EditText) findViewById();
  Et2 = (EditText) findViewById();
  Btn = (Button) findViewById();
Randomkey=(Spinner)findViewById();
  Iplpin = (Spinner) findViewById();
Buttonscan = (Button) findViewById();
 Scan.setOnClickListener(ne  OnClickListener()
```

Fig. 6a Graphical Interface mode Implementation

ii*)* *Transfer of Mode:* Here transfer mode is very important module, which contains two main modules, *private* and *public*. in public mode user can send the data directly towards the receiving device ,in the private mode security process is taken place with the help of secret key and DES algorithm.

iii) *Socket Connection:* here the socket connects the users to transfer the file, sender writes the in the output stream and receiver reads in the socket IP sream. Connection is made here to connect within the Wi-Fi.

Example:
Socket1socket=newSocket(selectedip,1234)

vi) Data Encryption Algorithm(DES): DES is used for the security purpose where we encrypt and decrypt the data/file using a secret key to avoid the access to the un authorized users.
The following figure shows the entire module implementation

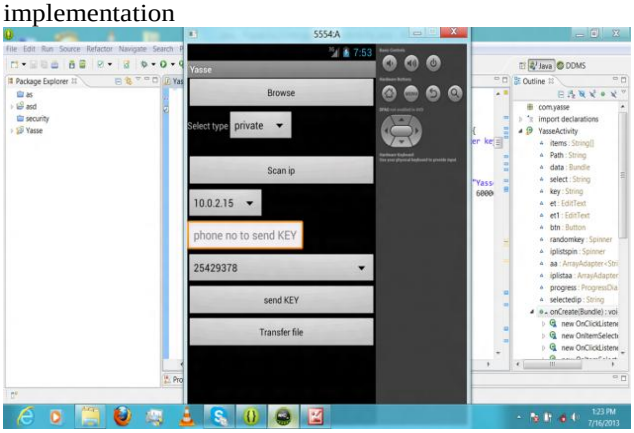

Fig.6b A complete sender Application model.

The receiver model contain the toast functional model where the pop-up menu will be raised to enter the DES secret key if the data is private or if the data/file is public then the file received message in the receiver application model.as shown bellow

Fig.6b A complete receiver Application model.

10. Conclusion

In This paper We are proposed and Implemented An attempt to present a file transfer application for the Android based mobile devices using WI_FI has been achieved. This application allows users to send a file or data (i.e. video, audio, images, text, and files) to the other android device in a secure manner. Application is able to filter out data tagged with user-defined labels (such as public, private, confidential).

In this way, applications can still access the data without reaching for user's sensitive information. This application can be useful in many real time events and applications for an enhanced user support.

8. References

[1] Android malware steals info from one million phoneowners.http://nakedsecurity.sophos.com/2010/07/29/android-malware-steals-info-million-phone-owners/.

[2] Android Project. http://www.android.com.

[3]ARM Trustzone Technology. http://www.arm.com/products/processors/technologies/trustzone.php.

[4] Gartner says android to command nearly half of worldwide smartphoneoperating system market by year-end 2012.http://www.gartner.com/it/page.jspid=1622614.

[5] These 26 Android Apps Will Steal Your Phone'sInformation. http://www.businessinsider.com/up-to-120000-android-phones-have-been-infected-with-malware-2011-5.

[6] Guangdong Bai, Liang Gu, Tao Feng, Yao Guo, and Xiangqun Chen.Context-aware usage control for android. In *Proc. SecureComm 2010*,pages 326–343, 2010.

[7] Alastair R Beresford, Andrew Rice, and Nicholas Skehin. MockDroid:trading privacy for application functionality on smartphones. In *Proc.HotMobile '11*, 2011. to be published.

[8] Sven Bugiel, Lucas Davi, Alexandra Dmitrienko, Thomas Fischer,and Ahmad-Reza Sadeghi. Xmandroid: A new android evolutionto mitigate privilege escalation attacks. Technical report, TechnischeUniversit¨at Darmstadt, D-64293 Darmstadt, Germany, June2011.

[9] M. Conti, V.T.N. Nguyen, and B. Crispo. Crepe: context-related policy enforcement for android. In *Proc. ISC '10*, pages 331–345, 2010.

[10] Mauro Conti, Vu Thien Nga Nguyen, and Bruno Crispo. Crepe: contextrelated policy enforcement for android. In *Proceedings of the 13th international conference on Information security*, ISC'10, pages 331–345, Berlin, Heidelberg, 2011. Springer-Verlag.

[11] Lucas Davi, Alexandra Dmitrienko, Ahmad-Reza Sadeghi, and Marcel Winandy. Privilege escalation attacks on android. In *Proceedings of the 13th international conference on Information security*, ISC'10, pages 346–360, Berlin, Heidelberg, 2011. Springer-Verlag.

[12] Michael Dietz, Shashi Shekhar, Yuliy Pisetsky, Anhei Shu, and Dan S.Wallach. Quire: Lightweight provenance for smart phone operating systems. In *20th USENIX Security Symposium*, 2011.

[13] William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol N. Sheth. TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones. In *Proceedings of OSDI 2010*, October 2010.

[14] William Enck, Machigar Ongtang, and Patrick McDaniel. On lightweight mobile phone application certification. In *Proc. CCS '09*, pages 235–245, 2009.

[15] William Enck, Machigar Ongtang, and Patrick McDaniel. Understanding android security. *IEEE Security and Privacy*, 7(1):50–57, 2009. [16] Peter Hornyack, Seungyeop Han, Jaeyeon Jung, Stuart Schechter, and David Wetherall. "these aren't the droids you're looking for": Retroffiting android to protect data from imperious applications. Technical report, April 2011. Available at: http://appfence.org/appfence.pdf.

[17] Anthony Lineberry, David Luke Richardson, and Tim Wyatt. These aren't the permissions you're looking, 2010. Available at: http://dtors. files.wordpress.com/2010/08/blackhat-2010-slides.pdf.

[18] Mohammad Nauman, Sohail Khan, and Xinwen Zhang. Apex: extending android permission model and enforcement with user-defined runtime constraints. In *Proc. ASIACCS '10*, pages 328–332, 2010.

[19] Machigar Ongtang, Stephen McLaughlin, William Enck, , and Patrick McDaniel. Semantically rich application-centric security in android. In *Proc. ACSAC '09*, pages 73–82, 2009.

[20] Georgios Portokalidis, Philip Homburg, Kostas Anagnostakis, and Herbert Bos. Paranoid android: Zero-day protection for smartphones using the cloud. Technical report, 2010. Available at: http://www.cs.vu.nl/ ~herbertb/papers/trpa10.pdf.

[21] Roman Schlegel, Kehuan Zhang, Xiaoyong Zhou, Mehool Intwala, Apu Kapadia, and XiaoFeng Wang. Soundcomber: A stealthy and contextaware sound trojan for smartphones. In *Proceedings of the 18th Annual Network & Distributed System Security Symposium*, NDSS '11, pages 17–33, 2011.

[22] Asaf Shabtai, Yuval Fledel, Uri Kanonov, Yuval Elovici, Shlomi Dolev, and Chanan Glezer. Google android: A comprehensive security assessment. *IEEE Security and Privacy*, 8:35–44, 2010.

[23] Yajin Zhou, Xinwen Zhang, Xuxian Jiang, and V.W. Freeh. Taming Information-Stealing Smartphone Applications (on Android). In *Proc. TRUST 2011*, 2011. to be published.