Client- Server Communication in Android os using Dynamic key Security Algorithm

[1]Kalaivant P,[2]Dr.Ramesh R,[3]Gayatheri S
[1]ME Student, [2]Associate Professor, [3] Research scholar,
[1, 2 3] Electrical and Electronics Engg Department,
[1, 2 3] CEG, Anna University

***Abstract* - Android is a smart mobile phone platform for developing new generation embedded devices. The Android ADK provides the tools and API necessary to begin developing embedded applications on the Android platform using the Java programming language. The paper presents a client – server based application which makes use of powerful Java technology to achieve its functionality. The server is interfaced to a real time device and the server stores the data of the real time application running in the device. The android client communicates with the database server through a secure channel. A secure channel is a way of transferring data that is resistant to overhearing and tampering. It also aims at providing a secure channel to transmit the data for the android application by implementing a security protocol for the communication channel at the server or either client application. The application offers a dynamic, efficient and secured remote access to data from any android device**

***Keywords* - Android, Security, Client Server, WIFI, Encryption**

I. INTRODUCTION

There is a growing interest in adopting android in embedded real time environments. Android's well supported, open source development environment eases application development. The Android ADK provides the tools and API necessary to begin developing applications on the Android platform using the Java programming language. The paper presents a client – server based android application which makes use of powerful java technology to achieve its functionality. The server is interfaced to a real time device and the server stores the data of the real time application. The android client communicates with the database server through a secure channel. A secure channel is a way of transferring data that is resistant to overhearing and tampering. It aims at providing a secure channel to transmit the data for the android application by implementing a security protocol for the communication channel at the server or either client application. The protocol is based on the user related parameters such as PIN and Unique Identification number and the device hardware identifier. Hence the security protocol is highly resistant to tampering. Thef application offers a dynamic, efficient and secured remote access to data from any android device.

II.ANDROID OS

Android is a mobile operating system based on the Linux kernel and currently developed by Google. With an user interface based on direct manipulation, Android is designed primarily for touch screen mobile devices such as smart phones and tablet computers, with specialized user interfaces for televisions (Android TV), cars (Android Auto), and wrist watches (Android Wear). The OS uses touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects, and a virtual keyboard. Android's source code is released by Google under open source licenses, although most Android devices ultimately ship with a combination of open source and proprietary software. Initially developed by Android, Inc., which Google backed financially and later bought in 2005,Android unveiled in 2007 along with the founding of the Open Handset Alliance -a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. The Architecture of Android OS is shown in Fig.1.

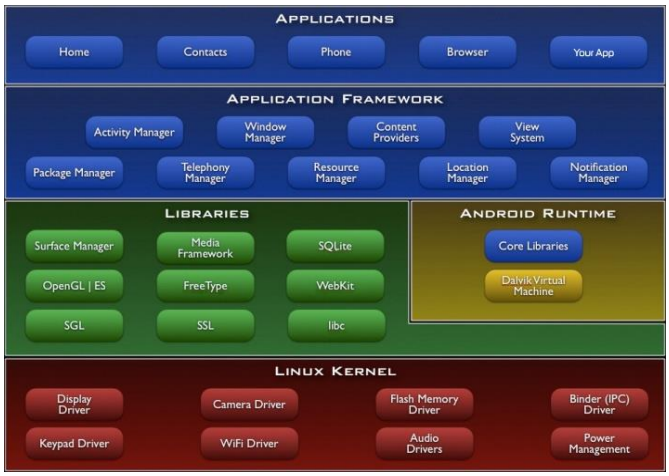


Figure.1. Architecture of Android OS

The Android OS can be referred to as a software stack of different layers, where each layer is a group of several program components. Together it includes operating system, middleware and important applications. Each layer in the architecture provides different services to the layer just above it. The basic layer is the Linux kernel. The whole Android OS is built on top of the Linux 2.6 Kernel with some further architectural changes made by Google. It is the Linux that interacts with the hardware and contains all the essential hardware drivers. Drivers are programs that control and communicate with the hardware.

The libraries are written in c or c++ language and are specific for a particular hardware. SQLite is the database engine used in android for data storage purposes WebKit is the browser engine used to display HTML content. OpenGL Is used to render 2D or 3D graphics content to the screen. Android Run time consists of Dalvik Virtual machine and Core Java libraries. It is a type of JVM used in android devices to run apps and is optimized for low processing power and low memory environments. Unlike the JVM, the Dalvik Virtual Machine doesn't run .class files, instead it runs .dex files. .dex files are built from .class file at the time of compilation and provide higher efficiency in low resource environments. The Dalvik VM allows multiple instance of Virtual machine to be created simultaneously providing security, isolation, memory management and threading support

III.PROPOSED SECURITY MECHANISM AND COMMUNICATION

The proposed security system and the communication establishment between mobile client and server is explained below:

A. Security Mechanism

The proposed security mechanism implemented for the Mobile to Server communication is shown in Fig. 2.

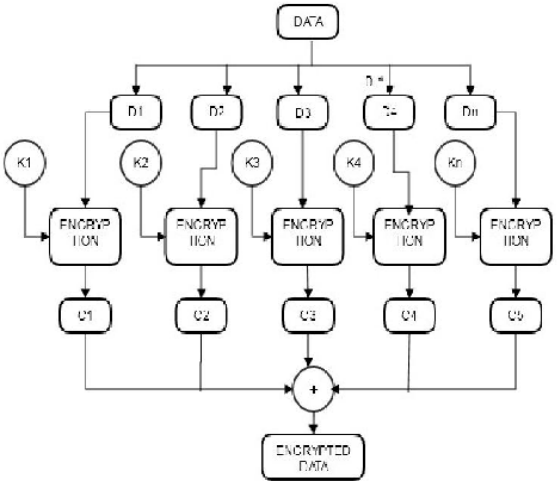


Figure.2. Encryption Mechanism flow diagram

The Encryption algorithm implements AES for encryption and decryption. The proposed encryption algorithm uses 128 bit key for encryption and decryption process as shown in figure 4 Encryption algorithm.

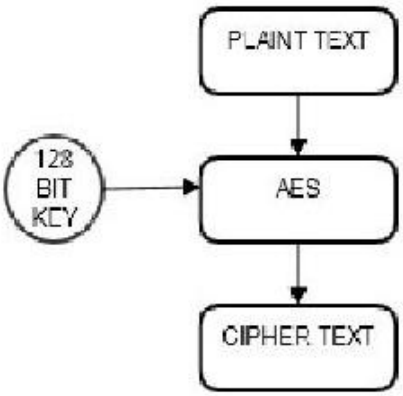
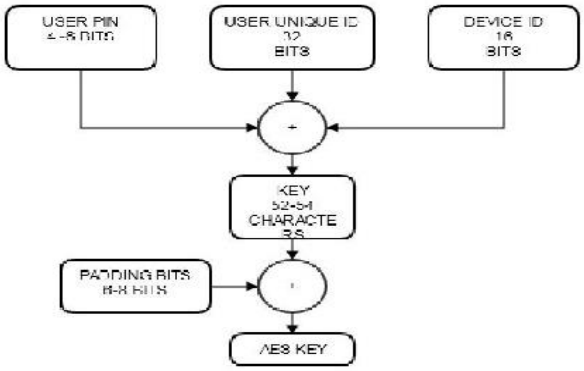


Figure.3. Encryption Algorithm

The key used for encryption is generated dynamically based on the User Parameters – Unique Identfier, User Device Id and User PIN. The UuiqueID, DeviceId and Pin are concatenated to form a string and the whole string is hashed,. The derived hash is the whole KEY K used for encryption and decryption. The Key generation process is detailed in Figure 4 Key Generation.



F

Figure.4 Key Generation

Based on the user PIN length, the user data D is split and correspondingly the key is also split. For n pin, the data varies from d1 d2…. dn and the key is split into k1,k2…..kn. For d1, k1 key is used for encryption , d2, k2 and for dn,kn is used for encryption. The encrypted data c1,c2..cn are combined by applying the user encrypted password as the delimiter to get the whole Cipher text C and stored in the database.

Again while decryption, the data is derived by applying the reverse process, The cipher text c1,c2..cn is derived from the cipher text C by applying the encrypted password as the delimiter. The original data d1 is recovered by applying the decryption with the key k1 to the cipher c1. Similarly, the data d2 is derived by applying k2 to c2 and dn is derived applying kn to cn. The data d1,d2..dn are combined to form the original data D.

For the give data, the same encryption algorithm is applied with different keys based on the user PIN length. The user PIN is restricted to minimum of 4 and a maximum of 6 characters. When the user realizes that his application security is compromised, then the user PIN can be changed. Hence the key also changes. For the same user, if he is going to use the application from a different device, then the key generated is also different since the device id changes. Hence the security offered by the algorithm is robust against overhearing and other network attacks [10].

B.Communication Establishment

The android client communicates tot he server sing HttpURLconnection class.of the java.net namespace. An java servlet would be running on the server. The android client has an upload screen which uploads the encrypted file and sent to the servlet. The servlet stores the file in the server. To view the file contents, the encrypted file would be downloaded and the decryption mechanism is applied at the client side ans the file contents are shown

IV. IMPLEMENTATION RESULTS

An android application ClientServerApp is built using the Android ADT. Android Development Tools (ADT) is a plug-in for the Eclipse IDE that is designed to give a powerful, integrated environment in which to build Android applications. Users using the ClientServerApp from their Device register to the database using the Signup screen. The sign up screen receives the user name, password, user pin and the device id is auto populated. The ClientServerApp home and signin screens are shown in figure.5 -ClientServerApp screens

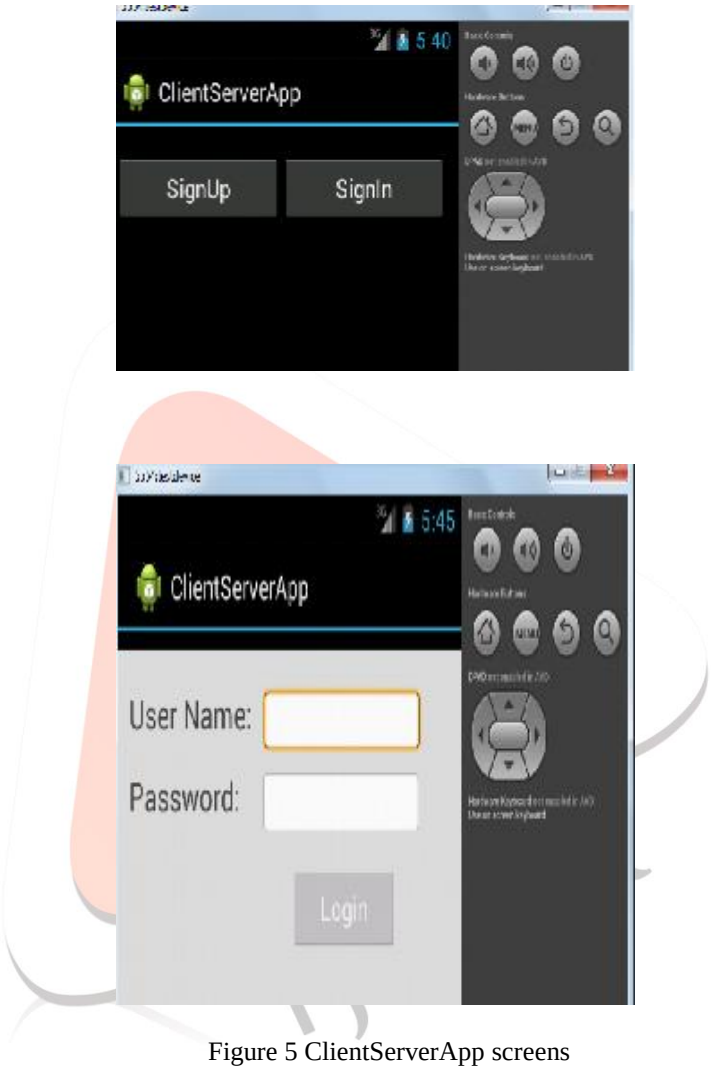


Figure 5 ClientServerApp screens

For each user on sign up, an unique Id is generated using the UUID class of System.Java. The UUID is a 32 bit field and the device Id from which the user registers is retrieved using the android.provider.Settings.Secure class parameter called the ANROID_ID.

Each user can have their own PIN which 4–6 characters and it accepts only numbers. Dynamic key is generated using the combination of UUID, DeviceID and User PIN. The Leas common multiple of 4,5,6 is 60 and the key generated using the GeneratewholeKey() method is padded to the length of 60. for an user pin of length 4, the data and key is split into 15 chunks . For each chunk, the encryption is applied. For an user pin of length 5, the chunk size is 12 and for user in of length, the chunk size is 10. Refer to the Table-1 for the padding bits used to create he key based on the user pin length.

S.no	User Pin Length	Dynamic Key length (UniqueID+DeviceID = 48)	Padding Bits
1	4	52	**"0101010"**
2	5	53	"0101010"
3	6	54	"010101"

Table 1 Dynamic Key – Padding Bit details

For a file size of 10 KB and user pin length 6, the data is split into chunks of each 1KB and for each chunk encryption is applied and the resulting encrypted chunks are merged to form the encrypted file. The figure 6 shows the files split and stored for encryption and decryption.

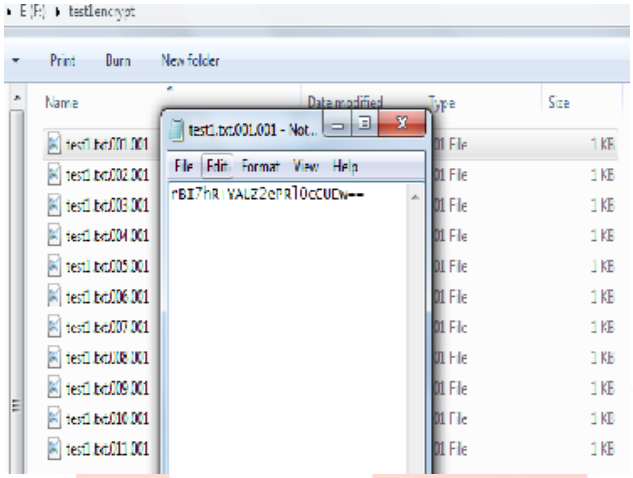


Figure 6 Encrypted and Decrypted files

To decrypt, the encrypted file ias again split inot chunks and for each chunk separately the key is applied for decryption. The decrypted files would be stored in folder and these files are merged together to form the actual revered file. The encryption an decryption methods and the class are shown in figure 6 - Encryptor Class Methods. D

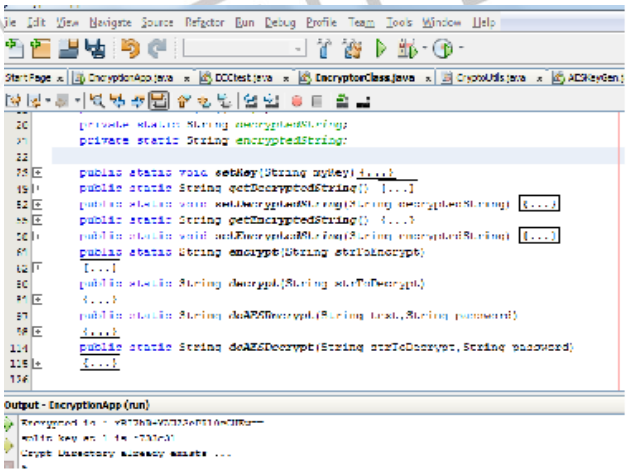


Figure 6 Encryptor Class Methods

The file is uploaded to the server using the File Upload screen.

The encrypted data is sent to the server via the HTTP do post method. The UploadFile Servlet stores the encrypted file. To view the content of the stored file, the uploaded file is received to the android client and the decryption is applied. The resulting decrypted file is saved to the client

The key itself is not stored rather only the parameters used for key generation are sent via the communication channel. Hence the user overhearing these parameters cannot decrypt the data directly

V.ACKNOWLEDGEMENT

Kalaivani thanks Dr.R.Ramesh and Dr.P.VanajaRanjan for their full support and the continuous encouragement in completing the proposed work and for extending in near future.

VI.CONCLUSION AND FUTURE WORK

Mobile to Server secure communication over is achieved in Android. The data overheard in the communication channel cannot be decrypted , because of the dynamic key and the complexity of the algorithm. Future work would be to improve the performance of the algorithm implemented

REFERENCES

[1] http://android-er.blogspot.in/2014/02/android-sercerclient-example client.html
[2] http://www.android-app-market.com/android-architecture.html
[3] http://www.android-app-market.com/first-android-app.html
[4] http://en.wikipedia.org/wiki/Android_version_history
http://www.android-app-market.com/android-development-tutorial
[5] http://ayurveda.hubpages.com/hub/Types-of-Network-Attacks]
[6] Kirti . P. Lokhande,Prof. Avinash . P.Wadhe;"Security in Android File System", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 12, December 2013 ISSN: 2277 128X Research Paper
[7] Attacking"Design of Chatting Application Based on Android Bluetooth", International Journal of Computer Science and Mobile Computing. Prof.P.Rama Bayapa Reddy,Dr.K.Soundararajan,Dr.M.H.M.Krishna Prasad,"Prevention of Attacks in internet controlled Embedded Applications"
[8] Poonam Mandavkar, Gauri Patil, Chetna Shetty, Vishal Parkar," SMS Security for Android Mobile Using Combine Cryptographic Algorithms ",International Journal of Advanced Research in Computer and Communication Engineering",Vol. 3, Issue 4, April 2014
[9] Software test attacks to break embedded & mobile devices – John Sri Parameswaran • Tilman Wolf,Embedded systems security—an overview.