IOSR Journal of Computer Engineering (IOSRJCE)
ISSN: 2278-0661, ISBN: 2278-8727 Volume 6, Issue 2 (Sep-Oct. 2012), PP 54-59
www.iosrjournals.org

Cryptography On Android Message Application Using Look Up Table And Dynamic Key (Cama)

Manisha Madhwani[1], Kavyashree C.V.[2] , Dr. Jossy P. George[3]
[1,2,3](Department of Computer Science, Christ University, India)

Abstract : *Most popular shortest and cheapest textual form of communication is short message service (SMS).To ensure the security of the texts which is sent, many algorithm are available. In this paper we propose an efficient algorithm for cryptography which is based on static Look Up table and Dynamic Key. Symmetric encryption and decryption is used in this algorithm. The proposed algorithm is more secure and simple to implement. This application makes use of built in android Intents and SMS Manager to send and receive messages.*
Keywords – *Secure SMS, Cryptography, SMS Encryption, SMS Decryption*

I. INTRODUCTION

Communication is the better way to exchange feelings, ideas and expressions. Communication involves a sender and a receiver conveying information through a communication channel. Senders and receivers are a vital part of communication. In face-to-face communication the roles of the sender and receiver are not distinct as both parties communicate with each other, even if in very subtle ways such as through eye-contact (or lack of) and general body language. There are many other subtle ways that we communicate with others, for example the tone of our voice can give clues to our mood or emotional state, whilst hand signals or gestures can add to a spoken message.

The two modes of communication are Verbal and Non-Verbal. Verbal communication includes text messages, presentations, discussions, and aspects of interpersonal communication. Non Verbal Messages can be communicated through gestures and touch, by body language or posture, by facial expression and eye contact.

Short Message Service (SMS) is a textual form of communication which is of precise length. SMS falls under the category of verbal communication. SMS are sent via mobile phone or through web. SMS enables users to exchange text messages economically in comparison to call rates.

Each messages in a mobile communication can contain at most 140 bytes (1120 bits) of data, the equivalent of up to 160 English characters [1]. SMS security ensures security of messages from the access of unauthorized users. Various SMS security options are used to provide the flexibility, control and interoperability that are required in the varied environments that SMS is used [2].

Some of the important applications of android message applications are Handcent, PANSI, ebuddy, LiveProfile and Kik Messanger.

Contribution: In this paper CAMA model is proposed. An efficient encryption and decryption for SMS is explained. The process of transforming plain text to cipher for data security is also proposed.

Organization: Introduction is given in section I, the existing research papers are discussed in section II, proposed model in section III, the algorithm is described in section IV, the performance analysis and user interface are described in section V and finally conclusion is given in section VI.

II. LITERATURE SURVEY

Mohsen Toorani et al., [3] provided the introduction of new Secure SMS Messaging Protocol (SSMS) for the Mobile-Payment systems. It being an application-layer protocol is intended for GSM users as a secure bearer in the mobile payment systems. It uses elliptic curve-based public key solution which uses public key as secret key for symmetric encryption.

Marko Hassinen [4] provided application solution named “Safe SMS”, using java for achieving confidentiality, integrity and authentication in SMS without any additional hardware for ensuring message is not tempered and authenticates sender.

SafeSMS has two methods for encrypting via Quasigroup and Blowfish. S. H. Shah Newaz et al., [5] proposed scheme for the enhancement of SMS security system for GSM users. Thereby, incorporating digital signature over cipher which is converted so by existing encryption schemes is made compatible to GSM security infrastructure. Encryption can be done with the existing GSM encryption algorithm, called A8. Then the encrypted message will create hash and finally it will be digitally signed. Thus, signed encrypted message will be transmitted.

Mary Agoyi et al., [6] evaluated encryption and decryption time for three algorithms RSA, Elliptic-curve and ElGamal to which plain text of different sizes is provided based on results one is chosen for further encryption. Their performance evaluation in securing SMS shows that key generation, encryption and decryption time increases with an increase in key size. Large key size algorithms are not suitable for SMS encryption due to small memory and low computational power of mobile phones.

Na Qi Jing Pan Qun Ding [7] did improvements on RSA algorithm because the SMSC will filter out the characters which are out of prescribed limit, thus the cipher text can't reach the destination. Thus, they also used FPGA based on high speed processing tools to implement the RSA algorithms and apply it in mobile phone short message encryption system.

Ch. Rupa et al., [8] proposed accost effective scheme which uses a concept called Cheating Text. The original message is embedded in a meaningful text called cheating text. Here, index table called (Real Message Index File) RIF file is hashed and sent to the receiver along with the cheating text in which the original message is embedded. Authentication is achieved by verifying the hash value of the plain text.

Rishav Ray et al., [9] proposed a scheme to encrypt messages using randomized data hiding algorithm to encrypt message using modified generalized Cipher Method. For encrypting secret message, a new algorithm called Modified Generalized Vernam Cipher Method (MGVCM) is used. For hiding this secret message, bits of each character of secret message are inserted in the LSB of eight randomly selected bytes of the cover file. The randomized embedding of message in a cover file provides an additional layer of security over the encryption.

Hongbo Zhou et al., [10] proposed a scheme which uses threshold cryptography based Defense Against Cyber Attacks (DCA) for MANET for solving problems of lower communication overhead. Invulnerability to mobile tolerance to missing or faulty server nodes, cryptography based DCA scheme is ideal.

David Lisoněk et al., [11] proposed an algorithm to send message through GSM using an asymmetric Rivest, Shamir and Adleman (RSA) cipher. This application prevents tapping and substituting techniques to secure SMS. It is achieved by storing the public key in a certificate which can be signed by the certification authority.

III. MODEL

In this section definition of different parameters of performance analysis of proposed CAMA model are discussed.

1 Definition

(i) ***Symmetric-key cryptography****:* Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key [12].

(ii) ***Cryptography:*** Cryptography is the practice and study of techniques for secure communication in the presence of third parties [13].

Encryption: Encryption is the coding or scrambling of information so that it can only be decoded and read by someone who has the correct decoding key [14]. The encrypted values can be obtained by using the Equation 1.

Static Character=126-(Plain Character ASCII – 32) (1)

(iii) ***Decryption:*** The process of decoding data that has been encrypted into a secret format [15]. The decrypted values can be obtained by using the Equation 2.

Plain Character=Static Character ASCII – 94 (2)

Digital Signature: A digital signature or digital signature scheme is a mathematical scheme for demonstrating the authenticity of a digital message or document. A valid digital signature gives a recipient reason to believe that the message was created by a known sender, and that it was not altered in transit [16].

2. Proposed CAMA Model

This proposed scheme is based on static Look Up table and Dynamic Key. It makes use of symmetric key encryption and decryption. The scheme is cost effective, simple and easy to implement. It is applied on message application to provide security to texts being sent from an android mobile to another. Look up table consists of characters starting from ASCII value 32 to 126 and 126 to 32. Working of CAMA model has been generalized in Fig. 1 below:

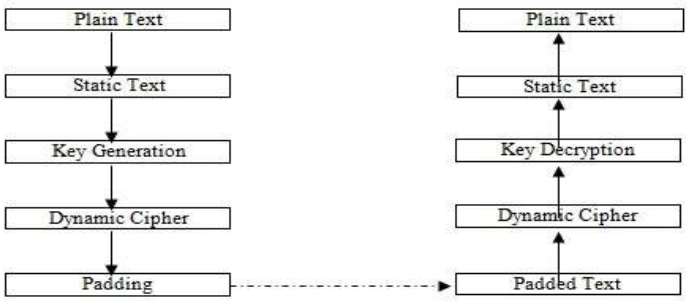

Fig. 1 CAMA Block Diagram

In Fig. 2 the message format of the proposed scheme is shown. Message length is length of the entered message. It is in hexadecimal number format. Data pair is the actual cipher pair. Character position is the position of the cipher pair in actual message. Encrypted key is the encrypted dynamic key which is used for decrypting the message.

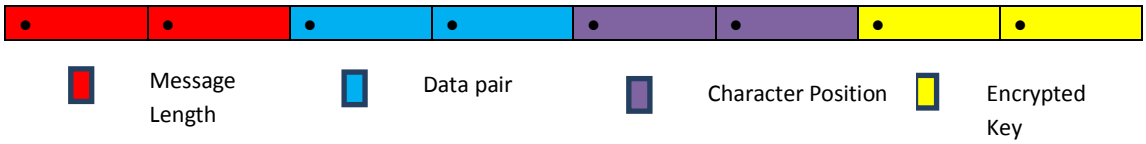

Fig. 2 Message Format

IV. ALGORITHM

4.1 Encryption

The process of transforming plain text to an unreadable format using a cipher is called encryption. The proposed cryptography scheme does encryption of plain text into cipher text incorporating generated dynamic key on static look-up table. In Table 1, algorithm for encryption of message is explained.

Table 1: Encryption of Message

Step1: Convert plain text to static text using Lookup table.
Step2: Generate Dynamic key.
A1= Contains length of index positions of all consonants
B1= Contains length of index positions of all vowels
C1= Contains length of index positions of all spaces
D1= Contains length of index positions of all special characters
L5 Contains actual Message length
denominator=[L5 -A1]+[L5 -B1]+[L5 –C1]+[L5 -D1]
numerator= 1's Complement(L5 in 8-bit binary)
M= numerator / denominator
Step 3: Key Encryption.
Convert key into binary.
Circulate each nibble left shift once.
Convert each nibble into hexadecimal.
Step 4: To each character in static text add key recursively and get dynamic text.
Step 5: Get ASCII value of each character of dynamic text.
Step 6: Convert each ASCII value to binary.
Step 7: Perform circular left shift.
Step 8: Convert each nibble to hexadecimal i.e. each character to cipher pair.
Step 9: Pad each cipher pair in following format
Message len + Cipher pair + position + encrypted key.

4.2 Decryption

The symmetric encrypted key received embedded in cipher is decrypted to get key which could thus decrypt cipher to retrieve plain text. Steps involved in the decryption of encrypted input text are explained in Table 2 supported by an example.

Table 2: Decryption of Message

Step 1: Extract cipher pair and the encrypted key.
eg. extracted cipher pair =36,c8,ae,ae,a8,63,e1,a8,a2,ae,ca and encrypted key=0e
Step 2: Key decryption.
Convert each hexadecimal value to binary.
eg. key 0e in decimal=014
key first part(0)=0000
key second part(14)=1110
Circular right-shift each nibble once.
eg. key first part on cir. Right shift rotation=0000
key second part on cir. right shift rotation=0111
Convert each nibble to decimal.
eg. 0000=> 0

0111=> 7
thus, decrypted key (07)10 = 7
eg. 1001 | 0011 => 9|3
a. Retrieve character of corresponding ASCII.
eg. 9|3 =>] (dynamic character)
b. Append above dynamic character to form string.
Step 3: Reassemble encrypted string in sequential order.
eg. assembled encrypted string
0b36010e0bc8020e0bae030e0bae040e0ba8050e0b63060e0be1070e0ba8080e0ba2090
e0bae0a0e0bca0b0e
Step 4: Convert cipher pair to dynamic char.
For each 8-byte pad do
eg. pad 1 holds: 0b36010e i.e. 0b|36|01|0e
a. Convert to binary.
eg. 3|6 to binary => 0011 | 0110
b. Circular right-shift each nibble once.
eg. 0011 | 0110 => 1001 | 0011
c. Convert to decimal i.e. the ASCII value of dynamic char.
Step 5: For each dynamic character do
a. Obtain static character by subtracting key from dynamic character's ASCII.
eg. 93-7 => 86 => V (static character)
b. Append above static character to form string.
done
Step 6: For each static character do
a. Retrieve corresponding plain character for static character from look-up table.
eg. V=>H
b. Append above static character to form string.
done

V. PERFORMANCE ANALYSIS

5.1 Study of Algorithm

Existing system uses very complicated algorithms to perform encryption. We have come up with an efficient algorithm which is easy to implement and to understand. Table 3 shows some performance analysis of the proposed system.

Table 3 Performance Matrix

Basis	Existing System	Proposed System
Space Efficiency	Consumes more space, since it uses table and index files. Less space efficient.	Consumes less space, since static look up table is computed not stored. More space efficient.
Ease Of Use	Quite Complicated	Simple and easy to operate and implement.
CPU Utilization	More	Less
Security	High	Average
Confidentiality	Depends	High

Plain Character v/s Cipher graph is showing cipher value comprising of encrypted data padded with message length, position and encrypted key for each character falling in ASCII range of 32-126 of look up table is drawn in

Fig. 3. Thus, message length=1, for all alphabets.

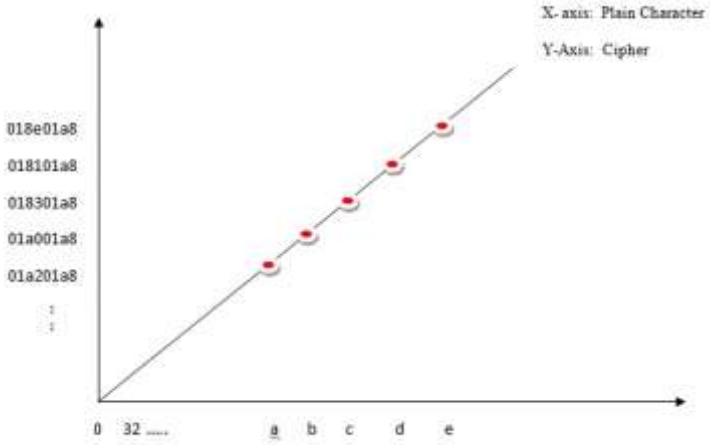

Fig. 3 Plain Character v/s Cipher graph

5.2 User Interface

User interface at sender's end is shown in Fig. 4. If phone number or message is not entered, a toast message is displayed as shown in the figure. This is a part of front end validation employed in the developed project.

Fig. 4 Application at Sender's End

Acknowledgement of message is very important to know if the message has reached the receiver or not. Fig. 5 shows the acknowledgement received by the sender after sending message.

Fig. 5 Acknowledgement of Sent Message

VI. CONCLUSION

The security of text becomes major issue especially in case of mobile banking; message carrying any military information etc. In the proposed scheme an algorithm for cryptography is proposed which is based on static Lookup table and dynamic key. It makes use of symmetric key encryption and decryption. This application makes use of built in android Intents and SMS manager to send and receive messages. The decrypted message is received on our application at the receivers end. Hence, this application is cost effective, simple and easy to use.

REFERENCES

[1] Information Security, http://www.infosec.gov.hk/english/technical/ files/ short.pdf.
[2] Micro System Center, http://technet.microsoft.com/en-us/library/ cc181234.aspx.
[3] M. Toorani and A. A. Beheshti Shirzai, SSMS - A Secure SMS Messaging Protocol for the M-Payment Systems, *IEEE Symposium on Computers and Communications*, 2012, 700-705.
[4] Marko Hassinen, SafeSMS - End-to-End Encryption for SMS Messages, *IEEE International Conference on Telecommunications*, 2008, 359-365.
[5] S. Jahan, M. M. Hussain,M. R.Amin and S. H. Shah Newaz, A Proposal for Enhancing the Security System of Short Message Service in GSM, *IEEE International Conference on Anti-counterfeiting Security and Identification*, 2008, 235-240.
[6] Mary Agoyi and Devrim Seral, SMS Security: An Asymmetric Encryption Approach, *IEEE International Conference on Wireless and Mobile Communications*, 2010, 448-452.
[7] Na Qi Jing Pan Qun Ding, The Implementation of FPGA-based RSA Public-Key Algorithm and Its Application in Mobile-Phone SMS Encryption System, *IEEE International Conference on Instrumentation*, Measurement, Computer, Communication and Control, 2011, 700-703.
[8] Ch. Rupa and P.S. Avadhani, Message Encryption Scheme Using Cheating Text, *IEEE International Conference on Information Technology*, 2009, 470-474.
[9] Rishav Ray, Jeeyan Sanyal, Tripti Das, Kaushik Goswami, Sankar Das and Asoke Nath, A new Randomized Data Hiding Algorithm with Encrypted Secret Message using Modified Generalized Vernam Cipher Method: RAN-SEC algorithm, *IEEE Information and Communication Technologies World Congress*, 2011, 1211-1216.
[10] Hongbo Zhou, Mutka and Lionel M. Ni, Multiple-key Cryptography-based Distributed Certificate Authority in Mobile Ad-hoc Networks, *IEEE proceedings of GLOBECOM*, 2005, 1681-1685.
[11] David Lisoněk and Martin Drahanský, SMS Encryption for Mobile Communication, *IEEE International Conference on Security Technology*, 2008, 198 – 201.
[12] Symmetric key Cryptography, http://en.wikipedia.org/wiki/Cryptography.
[13] An Overview of Cryptography, http://www.garykessler.net/library/cryp to.html.
[14] Encryption, http://www.techterms.com/definition/encryption.
[15] Decryption, http://www.webopedia.com/TERM/D/decryption.html.
[16] Digital Signature, http://en.wikipedia.org/wiki/Digital_signature.

Author's Profiles

Manisha Madhwani obtained her BA from University of Allahabad with Advanced National Level Diploma in Computer Applications from National Institute of Electronics and Information Technology and currently pursuing Postgraduation in Masters Of Computer Application (MCA) in Department of Computer Science, Christ University, Bangalore. Her projects include "One Pass Assembler For 8085 Using Java", "FMCG Online Management Store" and "iBook Management Store".

KavyaShree C. V. obtained her BCA from Sri Bhagwan Mahaveer Jain College, Bangalore and currently pursuing Post graduation in Masters of Computer Application (MCA), Department of Computer Science, Christ University, Bangalore. Her project undertaken includes Macro Weaver a "Two pass Macro Processor for IBM 360/370 machine", "Hotel Management System", and "Mobile store management".

Dr. Jossy P. George currently serves as Assistant Professor of the Department of Computer Science at Christ University, Bangalore. He received the B. Sc. In Computer Science, Bachelor of Philosophy (B. Ph), Bachelor of Theology (B. Th) and Master of Computer Application (MCA). He has done his FDPM from IIM, Ahmedabad. He was awarded Ph.D. in Computer Science from Christ University, Bangalore. His research interests include Image Processing, Biometrics, Computer Networks and Android Applications. He is a member of International Association of Computer Science and Information Technology (IACSIT) and Computer Society of India (CSI).