Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 6, June 2013, pg.53 – 59

RESEARCH ARTICLE

A Lookup Table Based Secure Cryptographic SMS Communication on Android Environment

Nishika[1], Rahul Kumar Yadav[2]
[1]PDM College of Engineering, Bahadurgarh, India
[2]PDM College of Engineering, Bahadurgarh, India

[1] ngulia5101989@gmail.com; [2] er13rahulyadav@gmail.com

Abstract— Information security is one of the major concerns while transferring some information publicly. With the advancement in the communication medium there are chances of more threats in the communication system. In this work, we have focused on the threats associated with SMS (Short Message Service) Communication in android environment. The work is about to define an effective symmetric cryptographic system that can encode the SMS text on sender side by using the minimum memory resources. As of the presence of limited resources on mobile phone, this light weight cryptography solution provides the secure communication by using dynamic and the lookup table. The work is implemented on Android 4.0 environments and tested on android mobile. The obtained results show the reliable communication on mobile devices.

Key Terms: - Android; Cryptography; Lookup Table; SMS; Symmetric

I. INTRODUCTION

Cryptography is one of the core approaches to provide the authenticated communication between the parties. Cryptography enables a user to perform secure communication over the network. Most of the electronic communication is performed under the cryptographic security. It saves the user information from any kind of theft. The main objective of any cryptography function is to encode the text or message by implementing some encoding technique. Cryptography can be performed on different media information such as text, audio, video etc. As the cryptography is performed, the encoded text called cipher text is obtained. This cipher text is not transmitted during the communication phase and provides the confidentiality of information. Now data can transmitted safely in unsafe networks. As the data is received by the specified receiver, the decryption algorithm is implemented to retrieve the data back from the cipher text. This decryption process is completely reverse to the encryption process and it is required to collect the information back in plain or original form [1,2].

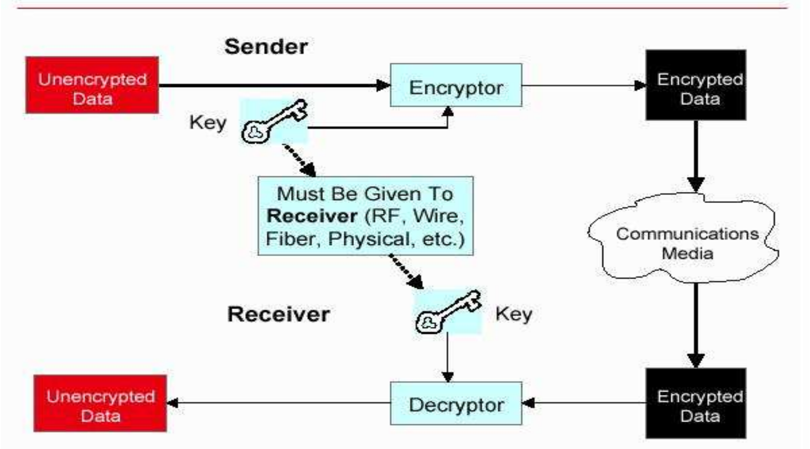


Fig 1 Symmetric Cryptography

The cryptographic algorithms are divided in two major categories called symmetric and asymmetric key cryptography. In case or symmetric key cryptography, same key is been used to perform the encryption and decryption. The security of these systems depends on the shared symmetric key. If the key is exposed to third person, the information can be extracted by that person. Here figure 1 is showing the symmetric cryptography process. In a public network, where the security is not that much necessary these kind of authenticated services are used. But when the security is concern to some organization or to some central server, in such case high level security services are required. In such case, the public key cryptography is been used to achieve the security goals [3]. These asymmetric cryptography algorithms are implemented to gain the optimum security goals. In this cryptographic algorithm, two keys are present in the cryptographic process, one for the encryption and other for decryption process. The public key cryptography process is shown in figure 2.

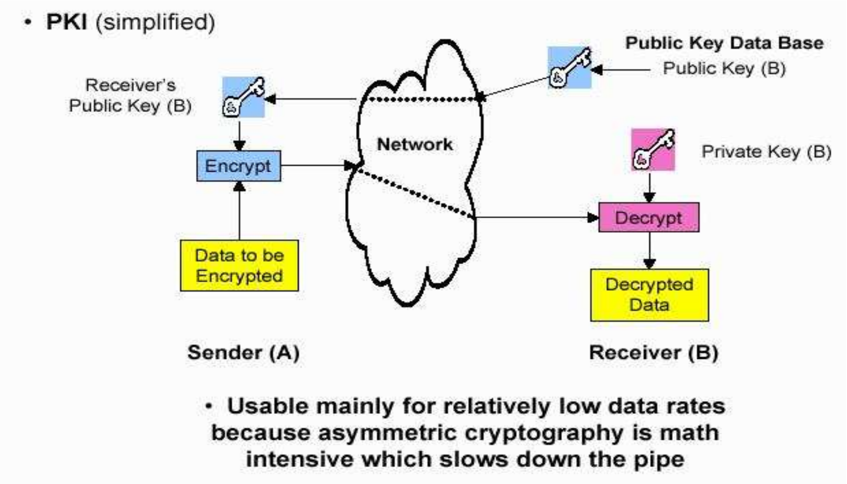


Fig 2 Asymmetric Cryptography

A. Digital Signature

Digital signature is some kind of finger prints that are attached with actual message to maintain the message integrity and to provide the effective authenticated messaging between the client and the server. To perform this signature marching different algorithmic approaches are used such as hash algorithm, cryptographic checksum and the digital fingerprints. The digital signature based message authentication system begins with the generation of a random key that can perform the data encryption. Now the random key will be encrypted and combined with the digital signature and perform the message encryption. This encrypted packet will be transmitted on an unsecured network. On the receiver side, the encrypted message and random key will be separated. Here on receiver side, the random key will be decrypted by using the private key of receiver. Now the message will be decrypted using the receiver random key. Finally the message will be retrieved as the plain message [6].

II. SMARTPHONES

Smart phones are having great importance in mobile computing. It is defined under the mobile applications that run on the smart phones and provide the support to vast market for the communication, entertainment and ecommerce. The presented system is defined for the mobile applications as well as provides the collaborative service to the standalone systems as well as to the application based model so that the services will run effectively on the mobile systems. It will allow the system to work on the real environment. Mobile phone applications are shifting from standalone designs to a collaborative (service) model. In this emerging environment, applications expose selected internal features to other applications, and use those provided by others. In the latter case, applications simply search and use appropriate providers of a service type at run-time, rather than bind itself to specific implementations during development. This allows a rich culture of “use and extend” development that has led to an explosion of innovative applications. This culture is possibly best illustrated in the Android1 operating system community [4,5].

The security model of the Android system is “system centric”. Applications statically identify the permissions that govern the rights to their data and interfaces at installation time. However, the application/developer has limited ability thereafter to govern to whom those rights are given or how they are later exercised. In essence, permissions are asserted as often vague suggestions on what kinds of protections the application desires. The application must take on faith that the operating system and user make good choices about which applications to give those permissions—which in many cases is impossible because they do not have sufficient context to do so.

Simply put, the Android system protects the phone from malicious applications, but provides severely limited infrastructure for applications to protect themselves. Based on extensive development of Android applications, observe three essential application policies not available to applications in the Android security framework [7,8]:

1) *Permission assignment policy:* Applications have limited ability to control to which permissions for accessing their interfaces are granted, e.g., white or black-list applications.
2) *Interface exposure policy:* Android provides only rudimentary facilities for applications to control how their interfaces are used by other applications.
3) *Interface use policy:* Applications have limited means of selecting, at run-time, which application’s interfaces they use.

III. PROPOSED MODEL

As smartphones become more ubiquitous, attackers make them their primary targets trying to either access the data stored on them or use them as an attack vector against business networks that those smartphones have access to. As a result, smartphones must be protected in both cases. They must be secured against network attacks where network can include 3G/4G networks, WiFi networks and even short range Bluetooth. The data stored in their internal memory and removable storage cards must be also sufficiently secured against unauthorized access whether this is network access via a malware application or loss/theft of the smartphone device itself. Third-party security applications are actively developed to provide the required security level.

In this work a dynamic key based symmetric cryptography approach is been implemented for the SMS text. The presented work is divided in two main layers, first to perform the compression to build the lookup table and to reduce the size of actual communication text. Once the lookup table generated, the encoding mechanism will use this lookup table to perform the cryptography by using the dynamic key. The encoded SMS will be communicated by using the android based smart phone. While performing the communication the key will be padded with text. On the receiver side, the encrypted key is obtained and then the decryption process will be obtained. On this decoded text, the decompression algorithm will be implemented to obtain the actual text. The block diagram of presented application model is given in figure 3.

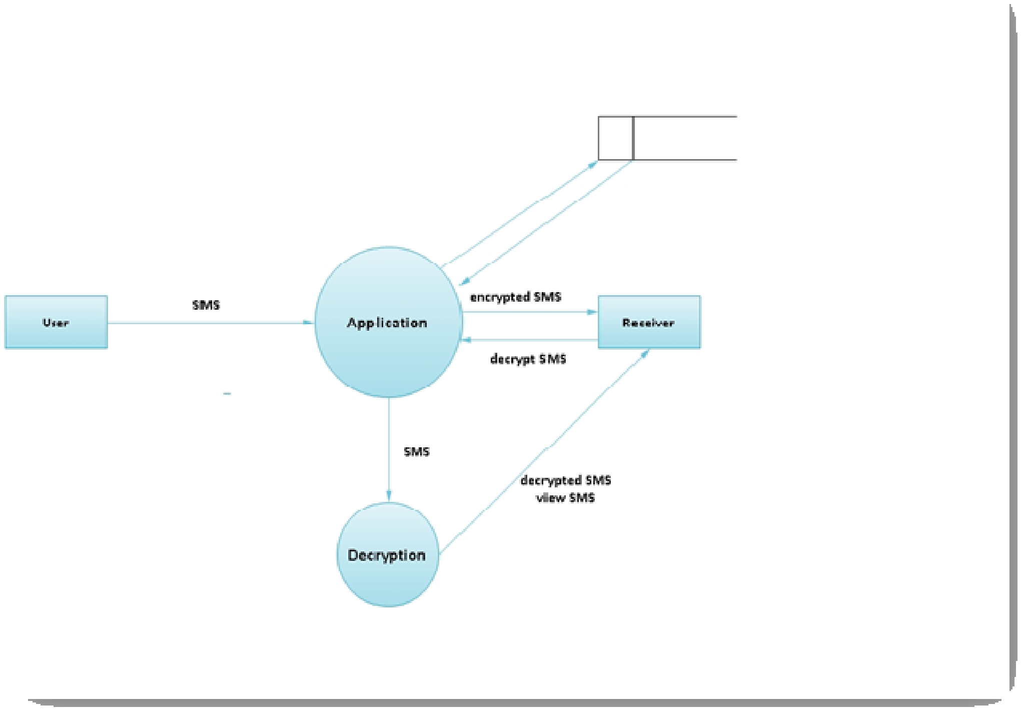


Fig 3 Presented Model

As we can see, the application is attached to the sender and the receiver and the application is defined under the cryptographic approach. As the user enters to the application, an SMS is communicated by the user. As the application receive the text SMS from the user. It will perform the encryption process on it and send to the receiver side. Now as the receiver read this encrypted message, the decryption algorithm will be implemented by the application. This application will process on the encrypted message and convert it back to the plain user oriented text form.

The cryptographic model is based on two main concepts, one is the lookup table and other is the dynamic key. The look up table is the temporary database system that is available on both the sender and the receiver application. Another concept associated with this work is the dynamic key. Here the dynamic key will be generated based on the text message analysis. The used algorithm for the cryptographic process is defined here under.

A. *Encryption Algorithm*

1. Accept the user SMS text as input to the cryptographic system.
2. Create an empty lookup table on sender side.
3. Perform the analysis on SMS text and obtain the text statistics. This statistics includes the

(i) Number of consonants message called C1
(ii) Number of vowels called V1
(iii) Number of spaces called S1
(iv) Number of symbols called Sy

4. Generate the dynamic key based on this statistics

(i) L=length(message)
(ii) den=(L-C1)+(L-V1)+(L-S1)+(L-Sy)
(iii) num=Complement(L)
(iv) Key = num/den

5. Now perform the Key Encryption by using following steps

(i) Convert the key to Binary form
(ii) Perform Left Circular Shift on each Nibble
(iii) Convert Each to Hexadecimal

6. Now read each character of text one by one and Add key to it and obtain the dynamic text

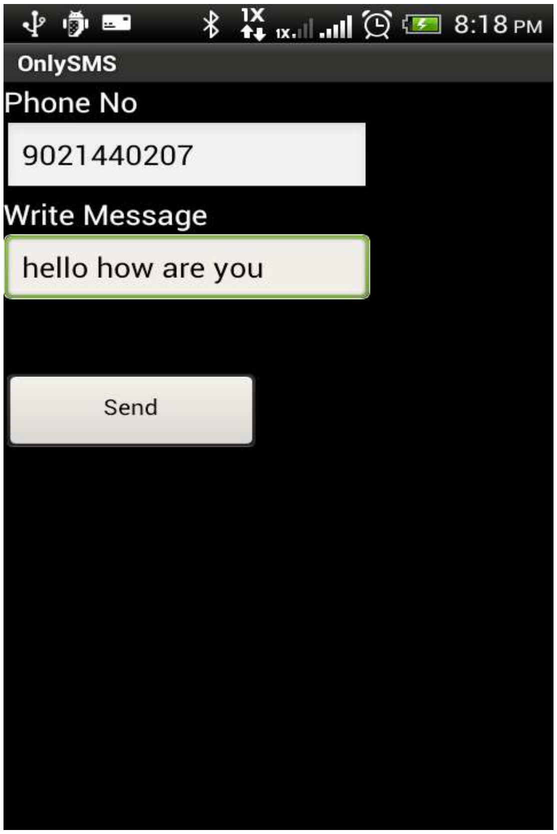
7. Obtain ASCII code of each character of Dynamic Text
8. Convert the ASCII value to Binary
9. Perform Circular Left Shift on Text
10. Convert each nibble to Hex
11. Add Padding to the text to include the key in message
12. Generate the Encrypted Text

B. *Decryption Algorithm*

1. Get the Cipher text as the Input Text to application.
2. Perform the Key Decryption of Message.
    - (i) Convert hex information to binary
    - (ii) Perform the circular right shift on each nibble of cipher text
    - (iii) Convert each nibble to Decimal
3. Retrieve character in the ASCII form.
4. Arrange this encrypted in the form of a string..
5. Convert the cipher text to the dynamic character by separating the padding from the text.
6. Read each dynamic character and convert it to binary form.
7. Perform the circular right shift on each nibble once.
8. Now subtract the key from each the ASCII code of dynamic character.
9. Obtain the character equivalent of each encoded character in lookup table.
10. Append the character in the form of String.

IV. Results

The work is implemented in Android 4.0 environment. Android is the latest technology and is an open-source platform developed by Google and the Open Handset [9] Alliance on which interesting and powerful new applications can be quickly developed and distributed to many mobile devices. The results driven from the system are given as under.

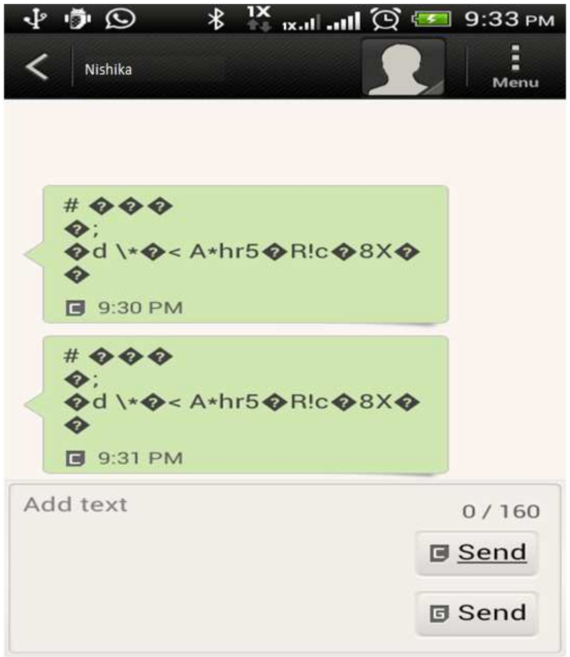


9021440207

hello how are you

Fig 4 Input Message (Android)

Here figure 4 is showing the input message using the Android based Smart phone. As we can say, the input screen will accept two parameters one is the receiver mobile number and other is input message. As the send button will be pressed, the input message will be communicated to the receiver side.

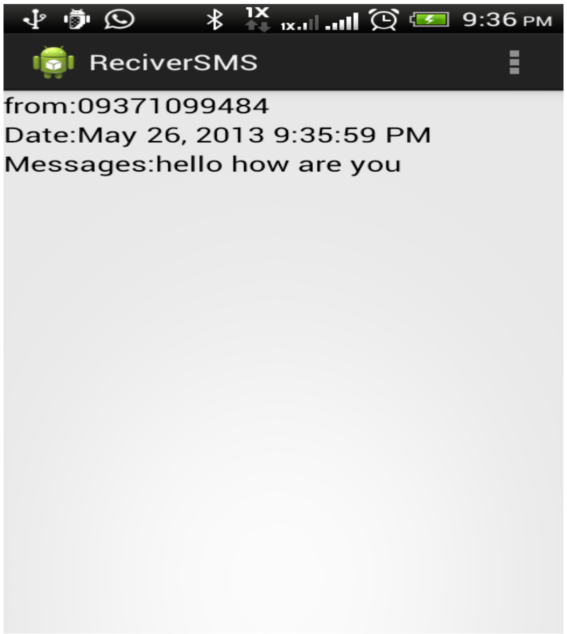


```
# ���
�;
�d \*�< A*hr5�R!c�8X�
�
9:30 PM

# ���
�;
�d \*�< A*hr5�R!c�8X�
�
9:31 PM
```

Fig 5 Intermediate Output

Here figure 5 is showing the processing output that will be drawn on the intermediate engine. As the input is received by the application and the encryption is performed. Here, figure is showing the cipher text.



```
from:09371099484
Date:May 26, 2013 9:35:59 PM
Messages:hello how are you
```

Fig 6 Receiver Side

Here figure 6 is showing the receiver side where the decoding of input message is performed and the result message is obtained. The figure is sowing the successful decoding of message by the application.

V. CONCLUSION

The security of text becomes major issue especially in case of mobile banking; message carrying any military information; M-Commerce etc. Asymmetric ciphers are very difficult to implement because two different keys need to be generated for both encryption and decryption.

In this work, a symmetric key encryption based application model is presented for the Android environment. The presented system is capable to encode and decode any SMS text for the smart phone. The obtained results show the effective encoding and decoding outcome driven from the system.

ACKNOWLEDGEMENT

I would like to thank almighty and my family for their constant support during this work. I would also like to thank my Guide Mr. Rahul Kumar Yadav for his support and encouragement to complete our work.

REFERENCES

[1] Artemios G. Voyiatzis, “Increasing Lifetime of Cryptographic Keys on Smartphone Platforms with the Controlled Randomness Protocol”.

[2] Adrienne Porter Felt, “Android Permissions: User Attention, Comprehension, and Behavior” , Symposium on Usable Privacy and Security (SOUPS) 2012, July 11-13,2012, Washington, DC, USA

[3] Manisha Madhwani, “Cryptography on Android Message Application Using Look Up Table And Dynamic Key (Cama)”, IOSR Journal of Computer Engineering (IOSRJCE) ISSN: 2278-0661, ISBN: 2278-8727 Volume 6, Issue 2 (Sep-Oct. 2012), PP 54-59

[4] Nathaniel Husted, “Smartphone Security Limitations: Conflicting Traditions”, GTIP ’11 Dec. 6, 2011, Orlando, Florida USA ACM 978-1-4503-1082-6/11/12 (pp 5-12)

[5] Bhaskar Sarma, “Android Permissions: A Perspective Combining Risks and Benefits”, SACMAT’12, June 20–22, 2012, Newark, New Jersey, USA. ACM 978-1-4503-1295-0/12/06 (pp 13-22)

[6] Liu Yang, “Short Paper: Enhancing Users’ Comprehension of Android Permissions”, SPSM’12, October 19, 2012, Raleigh, North Carolina, USA. ACM 978-1-4503-1666-8/12/10 (pp 21-26)

[7] Xuetao Wei, “Permission Evolution in the Android Ecosystem”, ACSAC ’12 Dec. 3-7, 2012, Orlando, Florida USA ACM 978-1-4503-1312-4/12/12 (pp 31-40)

[8] Sven Bugiel, “Practical and Lightweight Domain Isolation on Android”, SPSM’11, October 17, 2011, Chicago, Illinois, USA. ACM 978-1-4503-1000-0/11/10 (pp 51-62)

[9] David Barrera, “Understanding and Improving App Installation Security Mechanisms through Empirical Analysis of Android”, SPSM’12, October 19, 2012, Raleigh, North Carolina, USA. ACM 978-1-4503-1666-8/12/10 (pp 81-92)

[10] Rohan Rayarikar, “SMS Encryption using AES Algorithm on Android”, International Journal of Computer Applications (0975 – 8887) Volume 50– No.19, July 2012 (pp 12-17)

[11] Machigar Ongtang, “Porscha: Policy Oriented Secure Content Handling in Android”, ACSAC ’10 Dec. 6-10, 2010, Austin, Texas USA ACM 978-1-4503-0133-6/10/12 (pp 221-230)

[12] Marko Hassinen, SafeSMS- End-to-End Encrption for SMS Messages, IEEE International Conference on Telecommunications, 2008, 359-365.

[13] Roland Schloglhofer “Secure and Usable Authentication on Mobile Devices”, MoMM2012, 3-5 December, 2012, Bali, Indonesia. ACM 978-1-4503-1307-0/12/12 (pp 257-262)

[14] Na Qi Jink Pan Qun Ding, The Implementation of FPGA-based RSA Public-Key Application and Its Application in Mobile -Phone SMS Encryption System, IEEE International Conference on Instrumentation, Measurement, Computer, Communication and Control, 2011, 700-703.

[15] Mark H. Goadrich, “Smart Smartphone Development: iOS versus Android”, SIGCSE’11, March 9–12, 2011, Dallas, Texas, USA. ACM 978-1-4503-0500-6/11/03 (pp 607-612)

[16] 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.

[17] Himani Agrawal and Monisha Sharma, Implementation and analysis of various symmetric cryptosystems, Indian Journal of Science and Technology, Vol. 3 No. 12 (Dec 2010) ISSN: 0974- 6846