ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 2, August 2014

Secure Messaging System over GSM Based on Third Party Support

Saja T. Ahmed, Loay E. George
University of Baghdad/ College of Science/ Dept. of Computer Science/Iraq

Abstract-The number of mobile users is increasing rapidly and expected to reach the saturation point with at least one mobile for each person. SMS is an important application of mobile system but when sensitive information is exchanged via SMS and through a well-known security flaws in GSM network, a security solution is required. In this paper a secure indirect messaging system based on third party forwarding is presented to protect SMS privacy from eavesdropping as well as to ensure that the message is sent by legitimate sender. Also, the system perceives the phone numbers and instantaneous location of recipients. The users of proposed system can communicate exclusively through a third party (i.e., server) since there is no direct conversation between subscribers. Transmitted message is encrypted using a stream cipher RC4 with P2P encryption key; along with some proposed steps are added to meet diffusion, confusion and to immune the introduces security solution.

Keywords: **diffusion; SMS messaging; mobiles security; confusion; randomness test.**

I. INTRODUCTION

Mobile phone is an important part of the lives since mobile users do not leave to anywhere without their mobile. The(February 2013), estimated that there are 6.8 billion mobile subscriptions worldwide, that is equivalent to 96 percent of the world population (7.1 billion according to the ITU), and this indicate a huge increase from 6.0 billion mobile subscribers in 2011 and 5.4 billion in 2010.Portio Research – in the excellent free Mobile Fact book 2013- predicted that mobile subscribers worldwide will reach 7.5 billion by the end of 2014 and 8.5 billion by the end of 2016. The Global System for Mobile Communications (GSM) is a standard set developed by the European Telecommunications Standards Institute (ETSI) to describe protocols for second generation (2G) digital cellular networks used by mobile phones [1]. Short Message Services (SMS) is a vital part of GSM services that provides a mechanism to transmit short messages to, and from mobile devices in GSM and other cellular network, the maximum message's length is 160 characters. SMS uses only a set of characters as data type; it is a quit simple and requires very low bandwidth and it is low in cost compared with other services which makes it suitable for quick communication [2]. Although GSM traffic is usually encrypted, there is little security in exchanging SMS; and in the cases where the device is lost, stolen or otherwise accessed by an adversary no security barrier will met to stop the confidentiality harm. In such case messages stored in the device can easily be read and misused by the adversary [3]. Many researchers have investigated GSM and SMS security threats and proposed some cryptographic algorithms to achieve secure messages communication.Bramhe [4] and Rayarikaretal[5] have proposed a secure SMS based on symmetric cryptographic technique using AES algorithm. Chavan and Sabnees[6] and Rangarajanetal[7]proposed a secured SMS based on asymmetric cryptography using Elliptic Curve Cryptography (ECC).Computational load consumption of encryption operations must be taken into account when choosing encryption scheme. In our proposed system asymmetric encryption is avoided because of its high computational overhead; SMS encryption in proposed system depending on symmetric cryptography. It is good for mobile devices due to their limited resources (i.e., less computing power, insufficient memory and limited power energy. In this paper, a system is introduced for providing secure channels that enable mobile users to send and receive secure SMS,indirectly, using android mobile devices through a third party where no peer-to-peer communication among clients.T he system stores encrypted SMS locally in mobile devices memory. SMS will consist of the message body and overhead information that represent priority of SMS and destination client ID. Each originated secure SMS is sent to the main server over GSM. Then the main server receives the sent SMS from sender and directs it to the destination depending on their ID. Another channel is dedicated for main server to communicate with the intended destination (taking into account the mobility of destination client); this channel is used for sending email to guaranty SMS delivery as quick as possible. Every exchanged SMS between one of the clients and main server is encrypted using the stream cipher Rivest Cipher (RC4) beside to some other proposed encryption methods to ensure the secrecy of SMS.

II. PROPOSED METHODS

Encryption alone is not enough as long as the system uses an encryption algorithm that publicly available and depends only on using one key only between the clients and server; when attacker discover the encryption key this will collapse encryption system of the tracked client. The proposed system introduces some encryption methods to support messages privacy immunity and to make the system components safer. These encryption methods are performed at sender side and

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 2, August 2014

the corresponding decryption at recipient side, as explained in figure (1).The proposed methods are described in the following subsections:

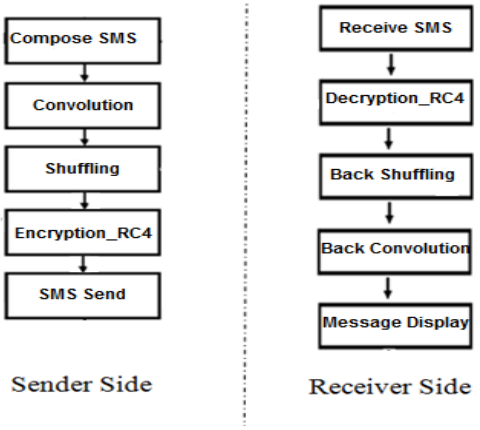

Fig.1: The proposed methods at sender and destination client

A. Convolution

The main aim of convolution is to make the task of finding the key as extremely hard task as possible; even if attacker has large number of plaintext-cipher text pairs produced using the same key. Also, the system applies substitution on certain elements of the message with other elements, depending on certain rules. The overall effect due to these two mechanisms is that "each byte of cipher text depends on the bytes of plaintext and key, but in different ways. In particular, changing one byte of the key should change the cipher text completely, by this way attacker hardly can get or deduce the information can lead him to discover key or parts of it even if he/she has some statistics about the cipher text. The proposed convolution algorithm uses a complicated compensation (substitution) method between message bytes and sender phone number bytes, as given in the following steps:

Step1: Convert the phone number into sequence of bytes:

1. Remove the redundant first four digits of phone number (i.e., for the mobile number 07701234567 the used part will be 1234567), the reduced number is called phone number.
2. Apply padding with specific characters if the phone array is less than message length.

Step2: Substitute message bytes with phone number bytes depending on R:

```
R =3 ///can be any number
K = message length
For all i Do {where 0 ≤i<K}
For all j Do {where -R ≤j ≤R}
Set P ←i+j
If (P < 0)Set P ← P + K
If (P > K) Set P← P - K
If ((P! = I) && (P >= 0) && (P <= K))
Set Con [I] ← Phone [P] XOR Message [I]
EndFor
EndFor
```

Step3: Return Con.

The convolution algorithm that explained above is used on sender side as first step to encrypt message. At recipient side the original message must be retrieved, so the back convolution algorithm must be applied, as a last step, in order to decrypt message. In back convolution same steps are used as in convolution algorithm except that the input will be the encrypted message and output will be the retrieved original message; this is possible because the applied convolution algorithm uses bitwise operator XOR.

B. Shuffling

The complex relationship between cipher text and symmetric key is generally implemented through a well-defined and repeatable series of substitutions and permutations. Substitution has been achieved through using convolution, as explained in above section, and permutation or transposition refers to manipulation of the order of bits according to some algorithm. To be effective, any symmetric structures may exist in the plaintext bits needs to be reorganized to ensure much complicated structures in the cipher text; this makes the symmetric structure harder to be detected. The proposed shuffling algorithm is a second stage that is performed at sender side to make cipher text hard to be intercepted by making the statistical structure of the plaintext is dissipated into long-range statistics of the cipher text this is to meet the diffusion requirements. The algorithm steps are as follow:

Step1: Generate an array of three prime elements depending on the phone number:

1. Convert phone number array of bytes to array of bits.
2. From the array of bits generate three numbers depending on specific sets of bits belong to the array of bits.

```
K=Phone. Length
For all i1 Do {where 0 ≤ i1<K/6}
a = a+Phone[i1]*2<<i1;
End For
For all i2 Do {where 8 ≤ i2<K /4}
b = b+Phone[i2]*2<<i2;
End For
For all i3 Do {where 20 ≤ i3<K /2}
c = c+Phone[i3]*2<<i3;
End For
```

3. Shuffle each number to closesprime integer

```
If (a is even) a = a + 1;
While (isprime(a) == false)a = a + 2;
If (b is even) b = b + 1;
While (isprime(b)== false)b = b + 2;
If (c is even) c = c + 1;
While (isprime(c) == false)c = c + 2;
Pr[0] = a;Pr[1] = b;Pr[2] = c;
```

Step2: Perform transposition:

```
K =Con.length
For all i Do {where 0 ≤ i<K}
Set FF ←((FF*Pr[0]+Pr[1]) mod Pr[2])mod K
Swap(Con[i],Con[FF])
```

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 2, August 2014

End For

Step3: Return Con as array of bytes.

When the recipient receives the encrypted message, one of the steps to decrypt message is to apply back shuffling algorithm which has same step-1 in shuffling, while the corresponding step2 and step3 are:

Step 2: Perform transposition:

Set M←Convert received decrypted SMS (as array of bytes) to bits
Set K ←M.length
For all I Do {where 0 ≤i<K}
Set FF ←((FF * Pr[0] + Pr[1]) modPr[2]) mod K
F[I] = FF;
End For
For all j Do {where k-1≥ j>0}
Swap(M[j],M[F[j]])

Step3: Return M as array of bytes.

C. P2P Encryption Key

The proposed system uses RC4 which is a stream cipher algorithm remarkable for its simplicity and speed in software makes it most suitable encryption algorithm to be applied on mobile device. Each member in the proposed system has a dedicated P2P RC4 key with the main server. This encryption scheme provides both message confidentiality and authentic-ation;so,if there is a security breach in one of the clients its impact will stay with that client, it does not affect other parts of the system and put the whole system in danger. Figure (2) illustrates the SMS data exchange between the elements of the proposed system.

III. CSMS WORK FLOW

The proposed system consists of two basic types of modules: (i) server module and (ii) client modules. The client module consists of two sub modules (i.e., the sender client and destination client),each of them performs certain set of tasks. Every subscribed client to the system should have a mobile device supported by android operating system, and a

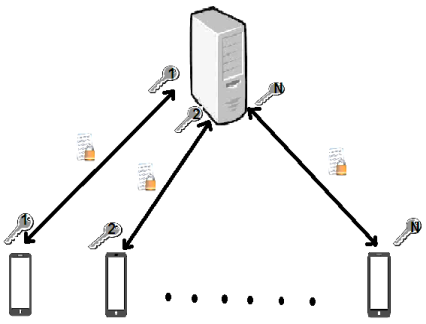

Fig.2: the Centralized Encryption Scheme

Special secure messaging software application which should installed on the device. The application has been developed using Java application development tools. The sender should enter the SMS body and the destination phone number only when he want to send a normal SMS using the built in SMS application, but when he use the proposed secure message application the sender should enter other information including:(i) message priority (it takes one byte because it is classified either as important set to 1 or normal set to 0), (ii)destination identification number that corresponds to his/her phone numbers and emails address at the server (it takes two bytes to accommodate two symbols on the most; which is retrieved from Sqlite database table).So, the sender can enter 157 character, as maximum, as a text message with 3 characters as overhead information; taking into consideration that the maximum length of the whole message body is 160 characters. The above two information are important parameters needed by server when it receive a message, send by authorized sender, to manipulate it and direct it successfully to its destination. The overhead information and the size of each element are shown in Figure (3).

Fig.3: Secure Message Payload

As shown in figure (4), the proposed system flow begins at sender client and passes through: Authorized client should compose his SMS, encrypts it, enter it to sqlite database, and send it to server, then waiting for SMS delivery acknowledgment, When server receive the encrypted SMS, it decrypts the content, analysis SMS, stores it encrypted using its own key into the main database, forwards it to destination client in its encrypted form using P2P key, then the server waits for SMS delivery acknowledgment, When destination client receives SMS; it stores its content in the local sqlite database, decrypts and displays the decrypted SMS.

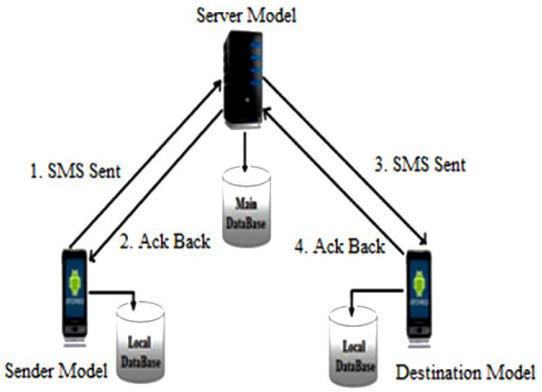

Fig.4: The proposed system workflow

IV. IMPLEMENTATION AND TEST RESULT

The software of the established system servers (i.e., main server, DB and GSM servers) is developed using Visual C# dot net programming language. The clients' application is

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 2, August 2014

developed using Java programming language and Eclipse SDK, working in Android environment. The clients' application consists of three main activities. The first activity is allowing the installed application to verify itself as an authorized party; this activity is enabled after the user assigns his email and password to server. If the client is not defined as authorized client this causes the first activity disabled. A snapshot of first activity is presented in figure (5).

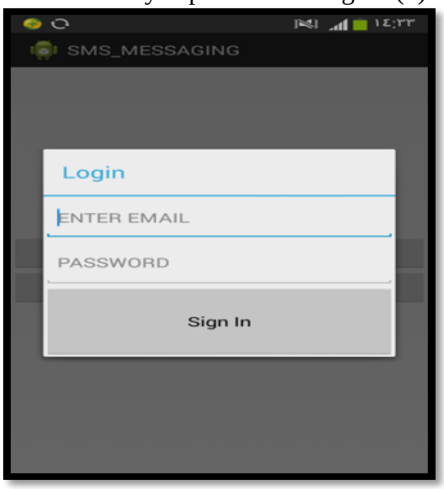

Fig.5: login activity

The second activity is a list view of three rows as follows: Compose SMS, Inbox and Outbox. Pressing on any row causes the second activity disappears to start with next activity; a snapshot of second activity is shown in figure (6).

Fig.6: list view activity

Inbox and outbox activities are used for storing encrypted incoming and outgoing messages in Sqlite database, respectively. Pressing on compose SMS row will initiate the third activity that allows user to enter server phone number, message body and overhead information (i.e., priority of SMS plus the destination client identification code); these information will be attached as a part of SMS body to be send to server. A snapshot of third activity is shown in figure (7). The goal of the proposed encryption system is to achieve randomness in the encrypted message; such that it is difficult for an eavesdropper to use statistical analysis to get a specific pattern that can lead him to discover the original message or key. Any successful encryption process is that can produce throughput that is statistically random and does not contain recognizable patterns or regularities.

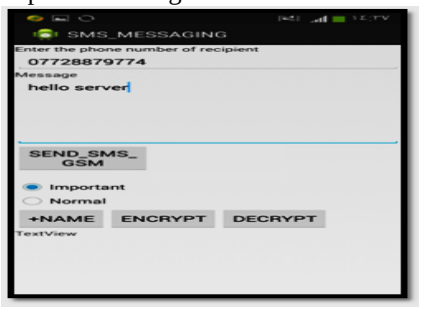

Fig.7: Compose SMS Activity

In this work the three NIST (National Institute of Standards and Technology) randomness tests (i.e., the frequency test, block test and runstest) have been applied using the encryption key "***ab78495fdjk6210d***" and without using hash value to examine the relationship between message and cipher text. Five plaintext messages each contains certain repeated pattern plus one normal message have been taken as test cases. Table (1) shows that encrypted messages meet the diffusion requirement because all produced cipher texts passes the three randomness tests.

Table (1) Diffusion Test Results

#	Plain Text	Cipher Text in hexadecimal	Random-ness test
1	aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaaaaaaaaaa aaaa	0cd327e7e62e957c167efae1 b503170e7bcd0b98af500c7 a433eeeac857d324124502c 8171b8bb57a96ca0d983a11 02baf06fc877c2ada4bbc5e7 31035eb8d3d9bd219ad6573 55a1a33aa3a5b6d9a50da82 c22a54474b501b135415c05 ea4410fc97c9aea9b066d64 1a5bdb939864407db1f72b9 b24b8ee4117689b9b27432e 85e863ee339e76d05d109a2 5069b1d673f6a3396c31860 90f3f9bd477d8569bc6	Freq. test: 0.6956(pass) Block test: 0.9632 (pass) Run test: 0.3397 (pass)
2	abababababa bababababab abababababa bababababab abababababa bababababab abababababa bababababab abababababa bababababab abababababa bababababab abababababa bababababab abbaba	4e9365e7a42ed77e147ebce 1f703550e39cd0998e9504e 7a413eacbc877d304124522 c8271a9bb54ad6ca6dbc1a2 103aaf07f8857e28da49ac5f 631131e98b3fd9d019ac657 215a3a138a1a7a4d8b70caa 2c26a50676b500b134015e0 7e84412fc96d9afadb260d4 03a7bdba39870005d91d70b bb04a9ce517768fb9f07232 eb5e877ee139e56d07d10ab 25169b3d230b4a1396f3187 490d3f98d675da5789c7	Freq. test: 0.4674 (pass) Block test: 0.8208 (pass) Run test: 0.5660 (pass)
3	aaabbbaaabb baaabbbaaab bbaaabbbaaa bbbaaabbbaa abbbaaabbba	4ad325e7a62e973e147ebea 1f501554a7bcd099aa9504e 7a417cacbe853d304324502 c8673adbb57ad7ca5db81a1 103aad07fc877e28d949ac5	Freq. test: 0.6956 (pass) Block test: 0.9517

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 2, August 2014

	aabbbaaabbb aaabbbaaabb baaabbbaaab bbaaabbbaaa bbbaaabbbaa abbbaaabbba aabbbaaabbb aaabbbaaabb baaabbbaaab bbaaab	c731133e98b3dd9d01be8657115a3a238a1a5a4d8b50ca6e22a50776b503b136435e07ea4412fe92d9aeadb263d441a1bdba3b870007d91d71bbb2498ce713768fb9f27630ad5e867ee13ae56d05d108b05369f1d470b5a1396c31870b0d3f9bd475da5189c6	(pass) Run test: 0.1301 (pass)
4	abcabcabcab cabcabcabca bcabcabcabc abcabcabcab cabcabcabca bcabcabcabc abcabcabcab cabcabcabca bcabcabcabc abcabcabcab cabcabcabca bcabca	0cd165e5e42ed73e167cf8a1b701174839cf099aef520c7a417ceeae873d344324522e8573bcbb54ab7ca3dbc1a21229ad06f8857c2ad949bc5d611237e98f3f9bd21be9657057a1a038a1a7b6dba70daa6e24a74576b502b337035e07e84610fe93c9afabb065d403a3bfb93b844405db1f73bbb0489ce615748bb9b07430ac5e853ce33ae56d07d30ba05269f3d233f5a1396f33864b0d3f98d677da509bc5	Freq. test: 0.7373 (pass) Block test: 0.8576 (pass) Run test: 0.8206 (pass)
5	aaaaaaaaaaa aaaaaaaaaaa aaaaaaaaaaa aaaaaaaaaaa aaaaaaaaaaa aaaaaaaaaaa aaaaaaaabbb bbbbbbbbbb bbbbbbbbbb bbbbbbbbbb bbbbbbbbbb bbbbbbbbbb bbbbbbbbbb bbbbbbbbbb bbbbbb	0a9165e5a66a953e167ebea1b703570a3bcf0bdaa9424e784178e8be877f340326502c8571babb55af7ca2d983a51039af06f8877e2ada4fac5c731031eb8f3d9bd219af657355a1a13aa3a5a6dba50da82c20a54474b503b135415c07ea4410fc95c9aeadb064d641a1bdbb39864017d81d73bfb24c8ee515348eb9f17230ef5c853ce13ae56e07d30eb0536bf3d130b5a13b2b32854b0d3c98d775da5188c5	Freq. test: 0.6956 (pass) Block test: 0.9109 (pass) Run test: 0.0729 (pass)
6	Helloahmedim here where are you?	693f717f7feca877cf84864d59a661d2ded6cc74cf56a68a32732c1d17bdd26d61a02f90	Freq. test: 0.2386 (pass) Block test: 0.6120 (pass) Run test: 0.6617 (pass)

Table (2) presents the results of applied three randomness tests for cipher text by taking three keys each have certain repeated pattern encryption key, in addition to two random encryption keys; one of them contains digits and alphabets; and the second contains symbols only; the tests conducted without using hash function to explain the relationship between key and cipher text. The original message is taken "***hello server im in baghdad***". The results of all tests demonstrated that the encrypted messages, using different keys, have passed the randomness tests. So, we can say that the proposed system meets the confusion requirement.

Table (2) Confusion Test Results

#	Encryption Key	Cipher Text	Randomness Test
1	aaaaaaaaaaaaaaa	05e7c1491f83fa5b6d38ed220f59c3950973dba24ddd1b244f10	Frequency test: 0.8897 (pass) Block test: 0.3835(pass) Run test: 0.6783 (pass)
2	abababababababab	3b278124488350a386603ca0b44e62706d4e52cf85f2bfac38de	Frequency test: 0.2120 (pass) Block test: 0.4908 (pass) Run test: 0.5553(pass)
3	1ab3e8nq6j7lp0z9	2e599076c31703fa7e4e9676739550537b253f890f7e9f86dee5	Frequency test: 0.2120 (pass) Block test: 0.6045 (pass) Run test: 0.6527 (pass)
4	~#@!%$^&*()?/}{+	4df33988433a59770dee450596f439a4cb27852cbd8a2c0452c6	Frequency test: 0.4054 (pass) Block test: 0.6887 (pass) Run test: 0.4570 (pass)
5	aaaaaaaabbbbbbbb	8b16a0075a2bf37a27393b3fbbd21eda88c827c68ddc44aaeb76	Frequency test: 0.5791 (pass) Block test: 0.8758 (pass) Run test: 0.5640 (pass)

REFERENCES

[1] Saxena, N. and Chaudhari, N.S., "A Secure Digital Signature Approach for SMS Security", International Journal of Computer Application (IJCA) .www.ijcaonline.org, 2011.

[2] Rangarajan, S., Ram, N. S., Krishna, N. V., "Securing SMS using Cryptography", International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 4, Issue 2, Pp. 285–288, 2013.

[3] Hassinen, M., Laitinen, P., "End-to-end Encryption for SMS Messages in the Health Care Domain", Connecting Medical Informatics and Bio-Informatics, R. Engelbrecht et al. (Eds.) ENMI, 2005.

[4] Bramhe, M.V., "SMS Based Secure Mobile Banking", International Journal of Engineering and Technology Vol.3, Issue 6, Pp.472-479, and ISSN: 0975-4024, 2011.

[5] Rayarikar, R., Upadhyay, S., Pimpale, P.,"SMS Encryption Using AES Algorithm on Android", International Journal of Computer Applications, Vol. 50, No.19, ISSN: 0975–8887, 2012.

[6] Chavan, R.R., Sabnees, M. "Secured Mobile Messaging", International Conference on Computing, Electronics and Electrical Technologies (ICCEET), 2012.

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 2, August 2014

[7] Rangarajan, S., Ram, N. S., Krishna, N. V., "Securing SMS Using Cryptography", International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 4, Issue 2, Pp. 285-288, 2013.