J. Bangladesh Electron. 11 (1-2); 1-8, 2011

A Higher Level Technique for Secured Message Transactions

Anisur Rahman[*1], M. Ismail Jabiullah [2] and M. Lutfar Rahman [3]

[*1-3] *Department of Computer Science and Engineering, Daffodil International University*
[2]*Department of Software Engineering, Daffodil International University.*

Abstract

A higher level secured message transaction technique has been designed, developed, analyzed and implemented for vulnerable communication channels. For this, a plaintext message is converted by using asymmetric key algorithm RSA and one-time pad as symmetric key algorithm to the ciphertext message. Three levels of encryption processes are imposed to produce the ciphertext that is to be sent to the destination. First, the original message is encrypted by the private key of RSA algorithm of the sender and then the output of the first encryption is again encrypted with the public key RSA algorithm of the targeted receiver and finally the output of the second encryption is further encrypted by using a shared secret key using one-time pad algorithm. After the above three encryptions a final ciphertext is generated that is to be sent to the receiver through the communication channels. In the receiving end, the receiver first decrypts the ciphertext by the shared secret key using one-time pad algorithm. Later, the receiver again decrypts the output of the first decrypted message with receiver's private key and RSA algorithm. Finally, the receiver decrypts the output of the second decryption process with the public key of the sender produced by RSA algorithm; and hence retrieves the original plaintext message successfully. In this process, shared secret-key and private-public key of the communicants establish the confidentiality, authentication, authorization and non-repudiation in the message transactions that ensures the stronger security of the message transactions between the sender and receiver. It can be applied where a stronger security service of the communicating messages are needed.

Keywords: Public Key, Private Key, Shared Secret Key, Confidentiality, Authentication, Integrity, Non-repudiation and Authorization.

1. INTRODUCTION

Security is the key for earning the trust of the communicating bodies in the electronic transaction process. It is becoming increasingly important for electronic transactions in network security arena [1,2,3]. If high security is ensured for making electronic transactions, then it can earn the confidence of the parties involved in the communication process. Now-a-days the communication channels become more vulnerable to the security threats. As a result effective measures are necessary to provide security to the messages sent through the communication media. The cryptographic techniques are the key ingredients to provide security to the electronic messages [4]. The security of the cryptographic techniques heavily relies on strong encryption and decryption algorithms and increased complexity associated to the algorithms [5]. In virtually all distributed environments, electronic mail is the most heavily used network-based application [6]. Users expect to be able to send mail to others who are connected directly or indirectly to the internet [7].

Table 1: Functions, Algorithms and their Descriptions

Function	Algorithms	Descriptions
Digital signature	DSS/SHA or RSA/SHA	A hash code of a message is created using SHA-1. This message digest is encrypted using DSS or RSA with the sender's private key and included with the message.
Message encryption	CAST or IDEA or Three-key Triple DES with Diffie-Hellman or RSA	A message is encrypted using CAST-128 or IDEA or 3DES with a one-time session key generated by the sender. The session key is encrypted using Diffie-Hellman or RSA with the recipient's public key and included with the message.
Compression	ZIP	A message may be compressed, for storage or transmission, using ZIP.
Email compatibility	Radix 64 conversion	To provide transparency for email applications, an encrypted message may be converted to an ASCII string using radix 64 conversions.
Segmentation		To accommodate maximum message size limitations, PGP performs segmentation and reassembly.

PGP (Pretty Good Privacy) provides a confidentiality and authentication service that can be used for electronic mail and file storage applications [8, 9].

In this paper, a higher level secured message transaction process has been designed, developed, analysed and implemented in programming language with a comparative security service analysis. The security measurement approaches of the proposed system have also been presented.

2. CONVENTIONAL METHODS

Someone wishing to send a secure mail message could first transform the file to be mailed using PGP, then e-mail the transformed file using a traditional mailer. Similarly, if one were to receive a PGP-encrypted mail message, one could treat the received message as a file and feed it to PGP to process. PGP uses public key cryptography for personal keys. The operation of PGP consists of five services as shown in Table 1 [7]. All the services are authentication, confidentiality, compression, e-mail compatibility, and segmentation.

Operational Description:

In message transmission process, the PGP entity performs the following steps **[9, 10]**:

1. Signing the message
   (a) PGP retrieves the sender’s private key from the private key ring using sender_userid as an index. If sender_userid was not provided in the command, the first private key on the ring is retrieved.
   (b) PGP prompts the user for the passphrase to recover the unencrypted private key.
   (c) The signature component of the message is constructed.
2. Encrypting the message
   (a) PGP generates a session key and encrypts the message.
   (b) PGP retrieves the recipient’s public key from the public-key ring using receiver_userid as an index.
   (c) The session key component of the message is constructed

The receiving PGP entity performs the following steps [11,12,13]:

1. Decrypting the message
   (a) PGP retrieves the receiver's private key from the private-key ring, using the key ID field in the signature key component of the message as an index.
   (b) PGP recovers the transmitted message digest.
   (c) PGP computes the message digest for the received message and compares it to the transmitted message digest to authenticate.
2. Authenticating the message
   (a) PGP retrieves the sender's public key from the public-key ring, using the key ID field in the signature key component of the message as an index.
   (b) PGP recovers the transmitted message digest.
   (c) PGP computes the message digest for the received message and compares it to the transmitted message digest to authenticate.

3. PROPOSED SYSTEM

The proposed system composed of three mathematical techniques and they are prime number generation, RSA key generation based on the given prime numbers and the corresponding secret key generation. All the techniques have been presented in the subsequent sections. The data flow diagram, encryption algorithm, decryption algorithm have also been presented.

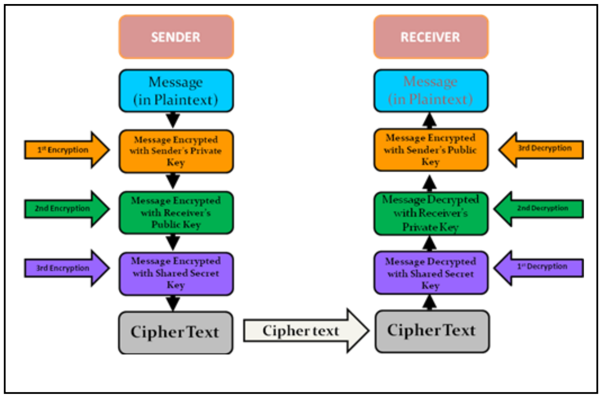

SENDER
RECEIVER
Message (in Plaintext)
Message (in Plaintext)
1st Encryption
Message Encrypted with Sender's Private Key
Message Encrypted with Sender's Public Key
3rd Decryption
2nd Encryption
Message Encrypted with Receiver's Public Key
Message Decrypted with Receiver's Private Key
2nd Decryption
3rd Encryption
Message Encrypted with Shared Secret Key
Message Decrypted with Shared Secret Key
1st Decryption
Cipher Text
Cipher text
Cipher Text

Fig. 1: Message Transmission Process in Proposed System

3.1 Encryption Algorithm

The encryption algorithm composed of the following two parts (Fig. 1); part A represents the acronyms of the message contents and part B depicts the steps of the encryption algorithm.

Part A:

Here, M = the intended plaintext message, PRs = Sender's Private Key, PUr = Receiver's public key, E = Encryption Algorithm, M' = First encrypted message with sender's private key, M" = Second encrypted message of M' with receiver's public key, SK = Shared Secret Key, C = Cipher text.

Part B:

Step 1: M' ← EPRs (M)

Step 2: M" ← EPUr (M')

Step 3: C ← ESK (M")

Step 4: C is sent to the destination.

3.2 Decryption Algorithm

The decryption algorithm also composed of the following two parts; part A is common to both encryption and decryption processes. Part C depicts the steps of the decryption algorithm.

Part C:

Step 1: M" ← DSK(C)

Step 2: M' ← DPUs (M")

Step 3: M ← DPRr (M’)

Step 4: M is the original plaintext message

4. IMPLEMENTATION

The proposed system has been implemented in three phases, viz. Key Generation, Encryption, and Decryption

RSA Key Generation/Public-Private Key Generation: The RSA algorithm generates a pair of public key PU = {e, n} and private key Pr = {d, n}. Here the sender knows the value of e, and the receiver knows the value of d [12]. The parameters e, d and n can be generated as following steps.

(a) Select two prime numbers, p and q and $p \neq q$

(b) Calculate $n = p \times q$

(c) Calculate $\Phi (n) = (p-1) (q-1)$

(d) Select integer e such that e is relatively prime to $\Phi(n)$ i.e., $\gcd (\Phi(n), e) = 1$; $1< e < \Phi (n)$

(e) Determine d such that $d = e^{-1} \pmod{\Phi (n)}$

As a result of the preceding steps a public key PU = {e, n} and a private key PR = {d, n} is formed.

4.1 Secret Key Generation

In the secret-key generation process, the communicants: sender and receiver to agree on the secret key without anyone else finding out. This requires a method by which the two parties can communicate without fear of eavesdropping [14, 15]. However, the advantage of secret key cryptography is that it is generally faster than public key cryptography. In our research a 32 bit random number is generated and used as a shared secret key for the both sender and receiver. This secret key is used in the third phase of encryption and decryption process [17].

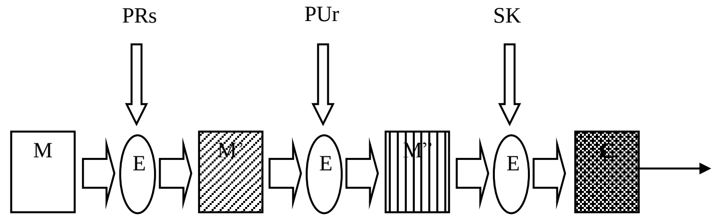


Fig. 2: Encryption Process of the Proposed System

4.2 Data Flow Diagram for Encryption

In the proposed encryption system, the message M is encrypted with the sender’s private key which is absolutely unknown to the outside world except the sender [16,17]. After the first encryption a new message M’ is produced. In the second encryption process, the encrypted message M’ is further encrypted with the public key of the receiver which is known to everybody and a new cipher text M’’ is produced. In the third step the cipher text M’’ is once again encrypted with the shared secret key which generates a final cipher text that will be sent through the insecure communication channel to the receiver. Fig. 3 shows the whole encryption process that has been done in the sender site.

The corresponding legends of the Fig. 2 are given below:

M = Message

M’ = Encrypted Message by PRs

M” = Encrypted M’ by PUr

PRs = Private Key of Sender

PUr = Public Key of Receiver

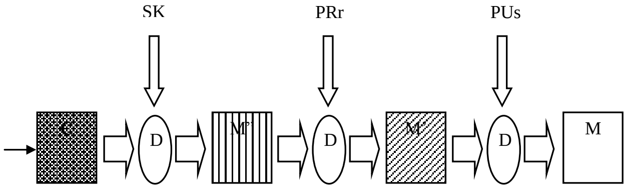


Fig. 3: Decryption Process of the Proposed System

E = Encryption Algorithm

SK = Shared Secret Key

C = Produced cipher text

4.3 Data Flow Diagram for Decryption

The received cipher text is first decrypted with the shared secret key that produces the cipher text M" and then decrypt it with the receiver's private key and retrieve encrypted message M'. After that the receiver decrypts the retrieved cipher text message M' with the sender's public key and finally get the intended message M. Fig. 3 shows the whole decryption process that has been done in the receiver side.

The corresponding legends of the Fig. 3 are given below:

M = Message

M' = Encrypted Message by PRs

M" = Encrypted M' by PUr

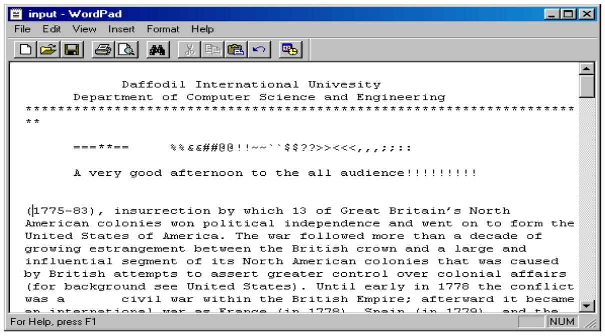
PRs = Private Key of Sender

PUr = Public Key of Receiver

E = Encryption Algorithm

SK = Shared Secret Key

C = Produced cipher text



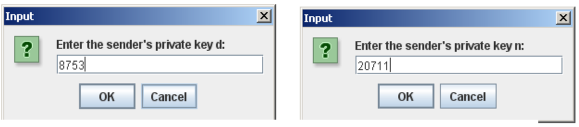
```
            Daffodil International Univesity
      Department of Computer Science and Engineering
**********************************************************************
**

      ===**==     %%&&##@@!!~~``$$??>><<<,,,;;::

      A very good afternoon to the all audience!!!!!!!!!


(1775-83), insurrection by which 13 of Great Britain's North
American colonies won political independence and went on to form the
United States of America. The war followed more than a decade of
growing estrangement between the British crown and a large and
influential segment of its North American colonies that was caused
by British attempts to assert greater control over colonial affairs
(for background see United States). Until early in 1778 the conflict
was a       civil war within the British Empire; afterward it became
an international war as France (in 1778), Spain (in 1779), and the
```

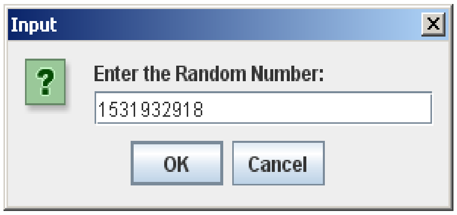
Fig. 4: Plaintext Message



Enter the sender's private key d:
8753

Enter the sender's private key n:
20711

Fig. 5: Encryption with Sender's Private Key



Enter the Random Number:
1531932918

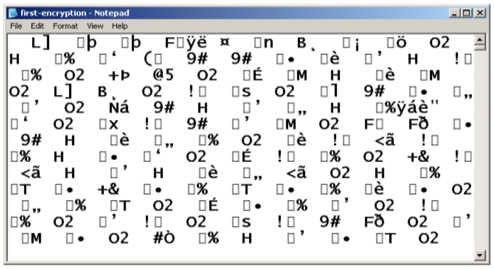
Fig. 6: Encryption with Secret Key

5. INPUT-OUTPUT ANALYSIS

The proposed system has been implemented in Java programming language. Java programming language has been chosen as it is object oriented and platform independent, strong security features, distributive and multithreaded nature influences us to work on [18, 19]. The Note Pad program of Microsoft windows has been used as an input and output file format. In the experiment, any length of note pad text file considered as a plaintext (Fig. 4) used and the corresponding output file is also stored in note pad format.

In Fig. 4, a plaintext message written on notepad is encrypted with sender private key in Fig. 5 and then it produces an encrypted cipher text as shown in Fig. 7.

In the second stage, the ciphertext produced that is the output of the first stage is further encrypted with the receiver's public key is again encrypted in the third stage.



```
  L]   □þ   □þ   F□ÿë ¤   □n   B¸    □¡  □ö   O2
H    □%   □‘   (□   9#   9#   □•   □è   □’   H    !□
 □%   O2   +Þ   @5   O2   □É   □M   H    □è   □M
O2   L]   B̀   O2   !□   □s   O2   □˥   9#   □•   □„
 □’   O2   Ṅá   9#   H    □’   □„   H    □%ÿáè"
□‘   O2   □x   !□   9#   □’   □M   O2   F□   Fð   □•
 9#   H    □è   □„   □%   O2   □è   !□   <ã   !□
□%   H    □•   □‘   O2   □É   !□   □%   O2   +&   !□
 <ã   H    □’   H    □è   □„   <ã   O2   H    □%
□T   □•   +&   □•   □%   □T   □•   □%   □è   □•   O2
 □„   □%   □T   O2   □É   □•   □%   □’   O2   !□
□%   O2   □’   !□   O2   □s   !□   9#   Fð   O2   □’
 □M   □•   O2   #Ò   □%   H    □’   □•   □T   O2
```

Fig. 7: Encrypted Plaintext with Sender's Private Key

In the third stage, the output ciphertext found in second stage is again encrypted with a randomly generated shared secret key (Fig. 6) and it ultimately produces the final ciphertext (Fig. 7) to be sent to the receiver.

In the receiving end, the cipher text is first decrypted with the same shared secret key which was used in the sending end and it retrieves the message which is the same as the second stage's encrypted output.

In the second steps of decryption process, the retrieved cipher text is further decrypted with the receiver's private key and it also retrieves the message which is the same as the first stage's encryption output.

In the third and final stage is decryption process the retrieved cipher text is further decrypted with the sender public key and it ultimately retrieves the original plaintext message.

6. COMPARATIVE ANALYSIS

A comparison between the proposed system and the existence system has been done and presented in the following Table 2. The comparative analysis is fully based on the realization of the fundamental security services: confidentiality, integrity, authentication, non-repudiation with authorization. All the services are established by our proposed system [19].

Table 2: Comparative Security Services of the Proposed System with the Existing System

System	Confidentiality	Authentication	Integrity	Non-Repudiation	Authorization
PGP	√	X	X	√	X
Proposed System	√	√	√	√	√

The security services performed by the proposed system are analyzed in the following:

- Confidentiality: The system performs encryption with the sender's private key that is not compromised by anyone in the communication and so it establishes first layer confidentiality of the message transaction. Since the system finally again encrypts the cipher text with the shared secret key for the communicants. Hence, it establishes second layer confidentiality in the communicating message.
- Authentication: The system performs encryption with the sender's private key and then again encrypts them with receiver's public key in the sender site. It is decrypted with sender's public key in the receiver's site, so sender can not deny that is not sent by the sender. Since, the message is decrypted with the receiver's private key that is related with the receiver's public key that is used in the sender's site for encryption. No one can decrypt the cipher text other than the intended receiver, since the receiver's private key is not compromised. Hence, it establishes the both way authentication for the communicating message.
- Integrity: The system encrypts the message contents through the sender's private key related to his public key. So, anyone who knows the sender's public key can decrypts the cipher text and retrieves the message. But the system again encrypts the cipher text by using the receiver's public key and so none other than the intending receiver can not decrypt the received cipher text and hence can not retrieve the message. So, thus it establishes integrity in the communicating message.
- Non-repudiation: The proposed system first encrypts the message with sender's private key, then encrypts with receiver's public key and finally encrypts the contents with a shared secret key. One can compromise the shared secret key and retrieves the cipher text that needs to retrieve the receiver's private key and that is not possible. Since it requires retrieving the message the receiver's private key, sender's public key and a sender-receiver shared secret key, so it is impossible to repudiate the transaction to each of the participants simultaneously. Hence, it establishes non-repudiation in the communicating message.

- Authorization: In this system, only the authorized sender and receiver can communicate the messages, since they have their public keys. No one other than the valid users can transact the message properly. Thus, it establishes the authorization in the communicating message.

CONCLUSIONS

A three-level encryption process is imposed to produce a ciphertext which is to be sent to the intended destination. First, the original plaintext message is encrypted with the private key of sender then again encrypted the output of the first encryption with the public key of the receiver; and finally the output of the second encryption is further encrypted by the shared secret key. After the above three encryption a final cipher text is generated and then is sent to the destination through the communication channels. In the receiving end, the receiver first decrypts the cipher text by the shared secret key and produce an intermediate text. Later, the receiver again decrypts the intermediary text that is the output of the first decrypted message with the private key. Finally, the receiver decrypts the output of the second decryption with the public key of the sender and retrieved original plaintext message successfully. In this process, shared secret key and private-public key of the communicants establish some security services: the confidentiality, authentication and non-repudiation successfully that ensure the strong security of the message transactions. It can be applied where a stronger security services are required.

REFERENCES

[1] A. Menezes, P. Van Oorschot and S. Vanstone, “Handbook of Applied Cryptography”, CRC Press, Inc., 1997.

[2] Gollmann, D. Compuer Security. New York: Wiley, 1999.

[3] Menezes, A.J., van Oorschot, P.C., and Vanstone, S.A., “Handbook of Applied Cryptography”, CRC Press, 1997.

[4] B. Schneir, "Applied Cryptography", John Wiley, ISBN: 9971-51-348-X, Pages: 169-180, 2nd Edition, 1996.

[5] B.A. Forouzan, "Data Communications and Networking", 3rd Edition, Tata-McGraw Hill, 2004.

[6] A. Tanenbaum, "Computer Network", 4th Edition, 2003.

[7] http://www.rsa.com/rsalabs/node.asp?id=2166

[8] W. Stallings, "Cryptography and Network Security", Principles and Practice, 3rd Edition, ISBN: 81-7808-902-5, Pearson and Education, 2003, 4th Indian Reprint, 2004.

[9] S. M. Bellovin and M. Merritt. “Encrypted Key Exchange: Password based protocols secure against dictionary attacks”. In Proceedings 1992 IEEE Symposium on Research in Security and Privacy, pp: 72–84. IEEE Computer Society, May 1992.

[10] Kimmo J¨ arvinen “Studies on Efficient Implementation of Cryptographic Algorithms”, Ph.D. thesis, Helsinki University of Technology

[11] http://java.sun.com/j2se.s

[12] http://www.asciitable.com

[13] Bellare, M, and Rogaway, P. “Optimal Asymmetric Encryption: How to Encrypt with RSA.” Proceedings, Eurocrypt ’94, 1994.

[14] M. Ismail Jabiullah, Kamrul Ahsan, Jahangir Alam, ANM Khaleqdad Khan and M. Lutfar Rahman, “Elliptic Curve Cryptographic Technique Implementation of Textmessage (SMS) Transaction in Mobile Phone”, In the Proceedings of the Annual Conference, Central Auditorium, Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, Page: 70, May 04-05, 2007.

[15] M. Ismail Jabiullah, Shamima Ahsan, ANM Khaleqdad Khan and M. Lutfar Rahman, “Emerging Applications of Cryptography in Bluetooth Networks”, In the Proceedings of the Annual Conference, Central Auditorium, Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, Page: 71, May 04-05, 2007.

[16] M. Ismail Jabiullah, Ahmed Fazle Rabbi and Shahida Rafique, “Privacy Mechanisms for Session Initial Protocol Message Over VoIP”, In the Proceedings of the Annual Conference, Central Auditorium, Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, Page: 71, May 04-05, 2007.

[17] M. Ismail Jabiullah, ANM Khaleqdad Khan and M. Lutfar Rahman, “Secured Session-key Distribution through Prime Field Elliptic Curve Cryptography”, In the Proceedings of the Annual Conference, Central Auditorium, Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, Page: 73, May 04-05, 2007.

[18] M. Ismail Jabiullah, ANM Khaleqdad Khan and M. Lutfar Rahman, “An Improved Session-key Distribution Technique for the Key Distribution Center (KDC)”, In the Proceedings of the Annual Conference, Central Auditorium, Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, Page: 73, May 04-05, 2007.

[19] A.N.M. Khaleqdad Khan, M. Ismail Jabiullah, Rosy Sharmin and M. Lutfar Rahman, “A Trusted-Based Novel Approach for Secured Mutual Distributions of Session-key and Message”, In the Proceedings of the National Conference on Electronics, Information and Telecommunication, Rajshahi University, Rajshahi, Bangladesh, A-132, ISBN: 984-300-000645-7, Page: 206, June 29-30, 2007.