ISSN 2278-3091

Volume 8, No.4, July – August 2019

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse67842019.pdf

https://doi.org/10.30534/ijatcse/2019/67842019

Enhanced Homomorphic Encryption technique using RSA ALGORITHM with multiple keys

D.Chandravathi [1] , Prof.P.V.Lakshmi[2]

[1]GVP College for Degree & PG Courses (A), Rushikonda, Visakhapatnam-45.

[2] GITAM University, Rushikonda, Visakhapatnam-45.

ABSTRACT

The most demanding, in present scenario for wide range of applications is security for transmitting confidential information over the network. The confidential information is prone to many malicious attacks. Hence, the need for protecting confidential information has become a major challenge over the Internet. Cryptographic algorithms are the best choice for providing security against malicious attacks. By applying various techniques information can be protected. For securing information over the cloud is another major challenge, which is still has to be considered. The idea of Homomorphic Encryption is a promising method for securing information in the cloud. In this paper, RSA algorithm using multiple public key pairs with Homomorphic Encryption is proposed. The idea is to generate a key pair from multiple keys using RSA homomorphic encryption, which is partially homomorphic in nature, instead of a single key pair. This technique utilises one key for enciphering and other for deciphering.. The beauty of this scheme is that a single key pair is selected from multiple key pairs which communicate with other parties. This technique of multiple keys generation utilises some mathematical logic for obtaining public key directly, when compared to RSA scheme with single key. By doing so, the attacks for finding the private key are stopped.

Key words: Confidential information, Internet, Cryptographic algorithms, malicious attacks, RSA , Homomorphic Encryption, decryption ,multiple key pairs.

1. INTRODUCTION

In the recent days, securing data is one of the challenges over Internet. Network security plays an important role for providing security of information[1]. For protecting data and network during data transmission, cryptography plays predominant role. Communications over the Internet has increased a lot and many transformations took place for secure transmission. Cryptographic algorithms are considered to as one of the solution for providing security [2]. Cryptographic algorithms and its functions prevent adversaries from decoding which are confidential messages[5].

Modern techniques in cryptography look upon various security factors which includes integrity, confidentiality, authentication and non-repudiation. The primary goal of network security is protecting network from unauthorized accesses [3][6]. To achieve this, authentication is an important phase for providing security. This phase is also known as One-factor authentication scheme. Schemes like Two-Factor and Three-Factor authentication are also used for verification using security tokens or fingerprints. With the help of firewalls, the authorized users have the privilege to access the services[6][8].

Public key cryptography plays a crucial role for providing enhanced security for data. RSA technique serves an example for public key cryptosystems. RSA scheme uses factorization of large numbers[2].RSA cryptosystems makes use of two keys, one for encrypting the message to cipher text and other for deciphering to plain text. This technique achieves major factors like secrecy, key distribution and strong authentication. The complexity lies in exponential factor of large numbers and also the generation of large prime numbers because an increase in key size requires nearly exponential increase in time[4][5].

1.1 RSA Algorithm

The RSA scheme has involves three stages:

1. Key Generation process.
2. Encoding or Encryption process and
3. Decoding/Decryption process.

In RSA scheme, the process of encryption is followed by encoding plain text using public key of the sender and include it in public file which is denoted by ‘E(n)’. Later, decryption process is carried out using sender’s private key[2][4]. The resultant cipher text is denoted by ‘D(n)’. The properties of RSA scheme is as follows:

1. Decryption process is done with the expression as $D(n) * E(n(M)) = M$, to get the original plain message.
2. Derive the values of E(n) , D(n).
3. None can compute the value of ‘D(n)’ by revealing ‘E(n)’. The user calculates the value of ‘D(n)’ followed by decrypting messages which were encoded with ‘E(n)’.
4. Decoding the original message followed by encoding , results in original message i.e.,

$$En\ (Dn(M)) = M.$$

Researchers namely, Rivest, Shamir, and Adleman [7], observed that if a procedure satisfying third property is applied, it is extremely impractical for another user to try to decrypt the message by trying all possible messages until they find one such that $E(n(M)) = C$[8][9].

1.2 Types of Cryptography:

There are two types of cryptography:

i. Different key cryptography

ii. Same key cryptography.

i. Different key cryptography

The figure 1 below depicts cryptography which uses different keys for encryption and decryption [9][11].

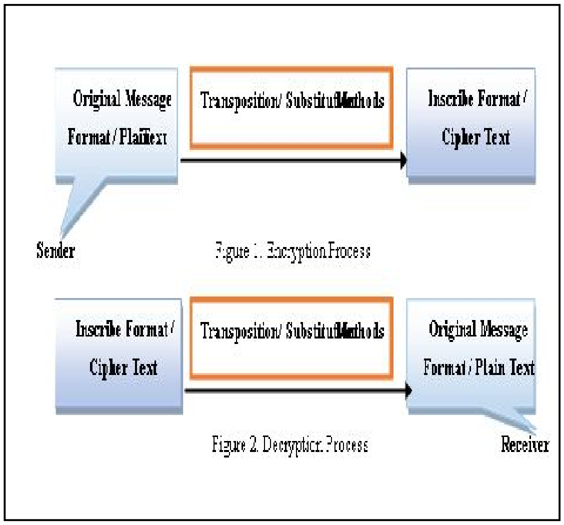

Figure 1: Different key Cryptography

ii. Same key cryptography

The Figure given below depicts same key cryptography, which uses same key both for encryption and decryption [10][15].

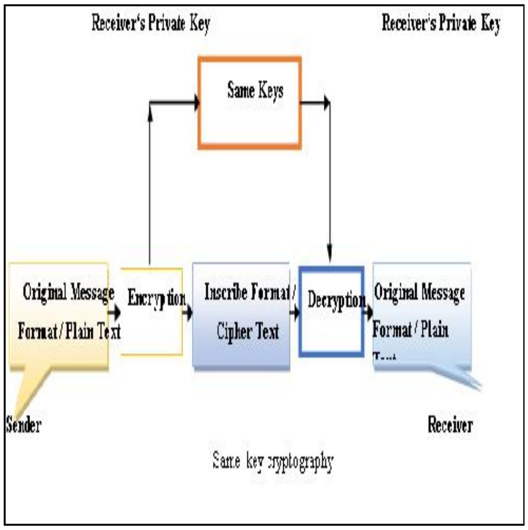

Figure 2 : Same key cryptography

2. HOMOMORPHIC ENCRYPTION SCHEME

Homomorphic encryption scheme aims at ensuring secure communication and storage with data privacy. We have several encryption schemes and one such scheme is RSA scheme which is multiplicative homomorphic[10] .RSA scheme computes the product of cipher text and generates the result that is equal to the product of original plain text[11].

In the year 2009, fully homomorphic encryption scheme was first introduced by the Gentry. He achieved "Somewhat Encryption Scheme "which limits only for few operations to be carried on encrypted data. The major operations are multiplication and addition. RSA does not support the addition operation[12][13]. RSA works with multiplication operation. Fully Homomorphic scheme can be implemented by performing both addition and multiplication operations on the cipher text.

Gentry observed that the encryption functions Enc (x1) and Enc (y1) are easily computed from Enc (x1 + y1) and Enc (x1 * y1). Since noise is attached with each of the operation on the cipher text, there where limitations on the operations. To avoid the noise problem in the cipher text, Craig introduced a technique of bootstrape which converts the scheme into the Fully Homomorphic Scheme. Gentry's idea was to build a Cryptosystem with usual encryption and decryption function that encodes and decodes cipher text from plain text and vice-versa[12]. The major applications of homomorphic encryption are in the cloud computing domain since the data is stored in encipherd format in cloud.

The figure3 given below describes the comparison between classical cryptosystem with respect to homomorphic cryptosystems. The idea is that a post processing is done so that the process of decryption is minimized.

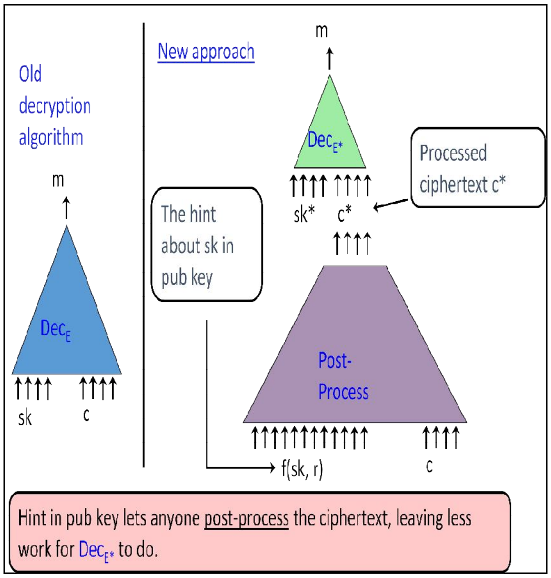

Figure 3: Traditional approach versus Homomorphic approach

3. OVERVIEW OF THE SYSTEM:

Figure4 describes the entire process of RSA homomorphic encryption technique using multiple keys**.**

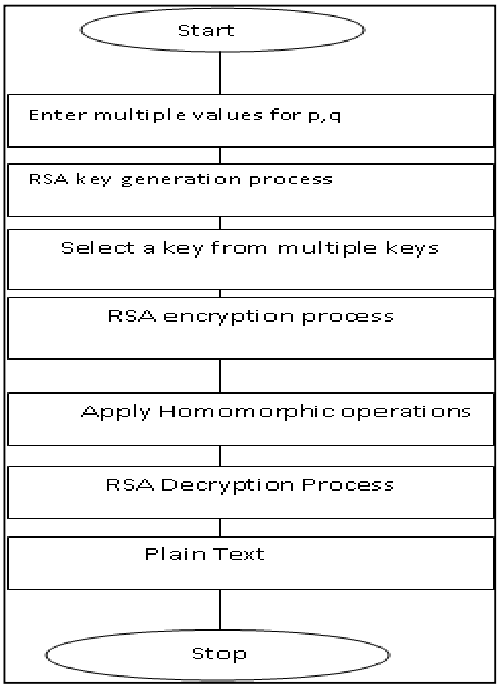
Start
Enter multiple values for p,q
RSA key generation process
Select a key from multiple keys
RSA encryption process
Apply Homomorphic operations
RSA Decryption Process
Plain Text
Stop

Figure 4: Overview of the system

4. ALGORITHM

The following are the mathematical foundations for RSA Scheme with digital signature [1]:

4.1 The RSA digital signature:

4.1.1 Theorem1: (fundamental theorem)

Let ‘*a’* be any positive integer which is denoted as ‘a_k’ such that $a_k = P_1, P_2, P_3, ..., P_n$, for all $P_1 > P_2 > P_3 \dots > P_T$ are prime numbers such that $a_k > 0$[3].

4.1.2 Theorem 2: (Euclid Theorem)

Suppose for two positive integers ‘x1’ , ‘y1’ , the greatest common factor ‘d’ is expressed as linear combination of integer coefficient p,q such that p,q € Z э d = xp + yq.

4.1.3 Theorem 3: (Fermat Theorem)

Select ‘p_1’ as a prime number and let ‘x’ be any positive integer then the Fermat theorem is given by [(x * (p_1-1)) ≡1 (mod p_1)].

Definition of Euler function (*n*) :

Let ‘n1’ be any integer such that n1 = 1 , [ϕ (1) = 1] for all n1 > 1, the value of ϕ (n1) is less than ‘n1’ as well as prime to n1.

4.1.4 Theorem 4

Suppose ‘p_1’ and ‘q_1’ are prime numbers such that ‘p1’ is not equal to q1 then , [ϕ (p_1* q_1)] is equal to [ϕ (p_1) * ϕ (q_1)] which in turn is equal to [(p_1 −1)(q_1 −1)][5].

4.1.5 Theorem 5: (Euler theorem)

Let ‘a1’, ’n1’ be two integers and if ‘a1’ is co-prime to ‘n1’, then [(a1 * ϕ(n1)) ≡ (mod m1)]. The following deductions are made from the above theorem:

i. According to Theorem 3(Fermat’s Theorem) ,let ‘p1’ be a prime number and if n1 = p1, then [(a1*(p1-1)) ≡ 1(mod p1)].

ii. Also we have [(a1 * ϕ(n1+1)) ≡ a1 (mod p1)].

iii. Suppose that if n1 = (p1*q1) such that ‘p1’ and ‘q1’ both are primes and p1≠ q1, for every m1,n1 0 <m1 <n1 and if GCD (m1, n1) = 1, then [(m*(p1- 1)*(q1-1)) +1 ≡ (mod n1)].

The above five theorems are used in of RSA cryptosystem as feasibility proof in upcoming section.

4.1.6 Theorem 6

Let ‘p1’ & ‘q1’ be two prime numbers such that p1≠ q1, then (r1* m1) ≡ 1 (mod (p1 − 1)(q1 − 1)) for some ‘r1’ , then if a, b, c belong to positive integers such that ‘b is congruent to [(a*m) mod (p*q)]) , ‘c’ is congruent to [(b* r) mod(p1* q1)] , then ‘c’ is congruent to [a mod (p1* q1)].

4.2 RSA Key Generation

1. Select any two random large primes numbers ‘p1’ and ‘q1’such that both p1 and q1 are of equal sizes and compute the value of n1 = [p1* q1] of required bit length.
2. Calculate the value of (φ) which is done by [(p-1)*(q-1)] [Theorem 4].

3. The value of public key component ‘e’ is calculated as GCD (e, phi) = 1, such that 1 < e < phi [Theorem 2].

4. The value of private key component which is denoted by ‘d’ is calculated using Theorem 6, by using the formula [e*d ≡ 1 (mod (φ))] such that1 < d < (φ) .

5. Hence, the public key is denoted as (n1, e) and the private key as (n1, d) and the values of d, p1, q1 and (φ) are kept secret[14].

4.3 Encryption Process

For process of encryption the following stops are performed by Sender ‘A‘:

1. Acquire the recipient, ‘B’s public key (n1, e).
2. Let ‘m’ denote plain text message.
3. After applying the public key to the plain text the cipher text is generated using the formula C= [m^e mod n1].
4. The generated cipher text ‘C’ is sent to B.

5. PROPOSED METHODOLOGY

Proposed methodology is a new modification to RSA cryptosystem. It is a modification that is done to RSA algorithm. The modified RSA technique works as classical RSA except the difference is in the generation of multiple keys. Classical RSA follows a single pair key generation process. The new modified RSA follows the concept of multiple key pairs. This is done by choosing a single key from ‘k’ derived multiple keys.

By doing so, RSA provides enhanced security which makes as the most secure public key cryptosystem. This can be done by increasing the number of the primes numbers .Further, composing the value of ‘n’ to ‘k’ given prime numbers. This technique provides RSA cryptosystem an extended feature of security. The hardness lies in finding the value of ‘k’ which is decomposed by ‘n’, which is not so easy to calculate. Hence, by calculating the number of private keys from a single key to multiple keys, makes the algorithm stronger and secure.

5.1 Algorithm

RSA Algorithm is well known as a block cipher [15]. The plaintext and the cipher text are integers which range from 0 and n- 1 for some ‘n’. In partial homomorphic encryption there will be two values i.e, v1 and v2.RSA is a partial homomorphic crypto system.We generates ciphers as C1and C2, which is computed as:

1. C1= [a^e mod n1] and C2= [b^e mod n1].
2. Encryption is given by C^e = [(C1.C2)^e mod n1].
3. The values of ‘n1’,’b’,’a’ are known to sender and receiver.
4. The receiver computes the value of ‘d’ . The public key is computed as K_U = {b,n1},{a}and private key by K_R = {d , n1} .The following can be computed for public key encryption :
   i. Compute the values of b, a, d, n1.
   ii. Derive the values M^e and C^d for all M < n1.
   iii. C^d is a multiple of ‘a’ and ‘b’ is the partial homomorphic technique in the normal RSA algorithm.
   iv. Then , the recipient ‘B’ calculates C = [(c1.c2)^e mod n] and transmits the value of C.
   v. On receiving cipher text, user A decrypts as [M = (c1.c2)^d mod n].

5.2 HOMOMORPHIC ENCRYPTION ALGORITHM

The process of encryption is done on behalf of sender ‘A’ as :

i. Receives the recipients B’s public key as (n1 , e).
ii. The two ciphers are generated as c1 and c2 as c1=[a^e mod n1] and c2 =[b^e mod n1] .
iii. The sender transmits the value of the cipher text using homomorphic encryption as :
C= [(c1*c2)^e mod n1] to B.

5.3 Decryption Algorithm

The Recipient B does as follows:

- Utilizes the received key (n1 , d) to compute the original message as M = [(c1*c2)^d mod n1].

6. RESULTS

The result of the proposed technique is shown below.Table1 describes encryption and decryption time with respect to different file size. The file sizes are in kilobytes(KB) and the time is calculated in milliseconds(msec).

File Size(KB)	eTime (msec)	dTime(msec)
10	8	5
20	12	10
30	15	13
40	18	17
50	20	18
80	27	23
100	32	29
120	43	35
150	65	51
200	110	93

Table 1 : Table for encryption and decryption time (msecs)with respect to file sizes(KB)

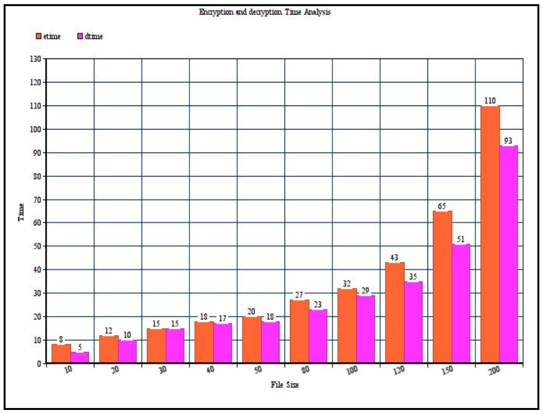

Figure 5: Chart representation for Homomorphic encryption and decryption process

From the above figure5, it is clear that the encryption time is more where as decryption time is less when compared to classical RSA to RSA homomorphic system. From the table1 it is clear that as the file size increases the time for encrypting with RSA homomorphic with multiple keys is increased .The time for decryption process is less when compared to encryption process. As discussed earlier homomorphic encryption is a good idea which minimises decryption process since operations are carried on the enciphered data[8][9].

7. CONCLUSION

Classical RSA using homomorphic encryption utilises a sing pair key for generation of public and private keys. Hence, security is at risk. In this paper, a new technique for implementing public-key cryptosystem using RSA Homomorphic encryption with multiple keys is proposed. The algorithm uses either of key from multiple keys generated for encryption, which is sent separately. Hence, the attacker doesn’t have much knowledge about the key and cannot decrypt the message. Homomorphic encryption technique applied to the ciphers generated by RSA helps to perform operations on the ciphers[8]. Homomorphic operations provide more security for data that is stored in the cloud provider. Hence privacy is preserved and security is enhanced.

REFERENCES

[1] Maheswari Losetti, Kanaka Raju Gariga "**An Enhanced Rsa Algorithm for Low Computational Devices"** International Journal of Advanced,Research and Innovations Vol.1, Issue .2, pp 114-118.

[2] Kuldeep Singh, Rajesh Verma, Ritika Chehal **"Modified Prime Number Factorization Algorithm (MPFA) For RSA Public Key Encryption",** International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-4, September 2012, pp 204-206.

[3] Sonal Sharma, Jitendra Singh Yadav and Prashant Sharma, **"Modified RSA Public Key Cryptosystem Using Short Range Natural Number Algorithm"** International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 8, August 2012, pp 134-138

[4] MJ Wiener. (1990), **"Cryptanalysis of short RSA secret exponents",** IEEE Transactions on Information Theory, Vol 36, No 3, pp 553-558. https://doi.org/10.1109/18.54902

[5] R Gennaro. (2000), **"RSA-Based Undeniable Signatures", Journal of Cryptology**, Vol 13, No. 4, pp 397-416. https://doi.org/10.1007/s001450010001

[6] Bulusu Rama, K Sai Prasad , P Sreeja, **"Secure k-NNquery on encrypted cloud data with multiple keys ",** International Journal of Advanced Trends in Computer Science and Engineering Volume 8, No.3, May -June 2019 ISSN 2278-3091, https://doi.org/10.30534/ijatcse/2019/82832019

[7] Syed.Karimunnisa, Dr.Vijaya Sri Kompalli, **"Cloud Computing: Review on Recent Research Progress and Issues" ,**April2019,International Journal of Advanced Trends in Computer Science and Engineering Volume 8, No.2, March ,ISSN 2278-3091. https://doi.org/10.30534/ijatcse/2019/18822019

[8] D.Chandravathi, Prof. P. V. Lakshmi , '**An Empirical Study on Homomorphic Encryption for Cloud Security'**, DOI 10.4010/2016.749 ISSN 2321 3361 © *2016 IJESC Research Article* Volume 6 Issue No. 4 **.**

[9] D.Chandravathi, Prof. P. V. Lakshmi ,**'A Novel Homomorphic Encryption Technique for Generation of Keys Using Cluster Classification for Cloud Security'**, *IJSER,*Vol 7, ISSN 2229-5518.

[10] D.Chandravathi, Prof. P. V. Lakshmi , **'Homomorphic encryption scheme using hill cipher for cloud data security'** , *International Journal of Advanced Information Science and Technology,* ISSN: 2319- 2682 .

[11] D.Chandravathi, Prof. P. V. Lakshmi , **'Performance Analysis of modified RSA and RSA Homomorphic encryption for cloud data security'** , *IJAR* , ISSN: 2320-5407 Int. J. Adv. Res. 5(2), 275-281 . https://doi.org/10.21474/IJAR01/3144

[12] D.Chandravathi, Prof. P. V. Lakshmi , **'A New Hybrid Homomorphic Encryption Scheme for Cloud Data Security ',** *Advances in Computational Sciences and Technology* ISSN 0973-6107 Volume 10, Number 5 (2017) pp. 825-837 © Research India Publications.

[13] D.Chandravathi, Prof. P. V. Lakshmi , **'Performance Analysis of Homomorphic Encryption algorithms for Cloud Data Security'** , *International Journal for Research in Applied Science & Engineering Technology (IJRASET)***,** ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887 Volume 6 Issue III, March 2018.

[14] D Boneh, M Franklin. (2001), "**Efficient generation of shared RSA keys",** Journal of the ACM, Vol 48, No. 4, pp 702-722. https://doi.org/10.1145/502090.502094

[15] Gagandeep shahi, Charanjit singh **"Cryptography and its two Implementation Approaches"** International Journal of Innovative Research in Computer and Communication Engineering ,Vol. 1, Issue 3, May 2013,PP 668-672.