IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 4 Issue 1, January 2017
ISSN (Online) 2348 – 7968 | Impact Factor (2015) - 4.332
www.ijiset.com

Modified RSA Cryptosystem

Harsh Sahay
Department of Computer Science and Information Technology
Bit Sindri, Dhanbad, India (Jharkhand)
harshsahay_cit@rediffmail.com

Abstract

A traditional RSA Cryptosystem is based on only two prime numbers which is an efficient algorithm for preventing an unauthorized access over the internet. But there are some drawbacks in RSA cryptosystem, such as its high computational time. The primary motivation of our work is to reduce average computational time and provide better data security compared to traditional RSA. In this work we are modifying basic RSA cryptosystem algorithm by using three prime numbers which provides better data security as compared to standard RSA algorithm. Instead of applying RSA over each data unit, multiple data units are merged together to form one merged unit. The modified RSA is applied on the merged unit to form a cipher text which is sent by the sender. For merging multiple data units into single data unit, Cantor's pairing algorithm has been used. At the receiver's end the cipher text sent is received. The cipher text is deciphered using our modified RSA algorithm, which is the merged plain text (PT). Then this merged data unit is separated (unpaired) using Cantor's un-pairing algorithm. The highlight of this work is that, it increases the efficacy of the asynchronous cryptography(as compared to traditional RSA). The proposed framework increases security and reduces the average time taken for sending the data from sender to receiver

INDEX TERMS - Cryptography, RSA, public key, private key, pairing and unpairing algorithm.

INTRODUCTION

In recent trends internet provides communication between peoples, defense personals, gives facility to electronic payment and many others. This is reason behind much concern of privacy, identifying theft, security etc. Recently, due to the large losses from illegal data access, data security has become an important issue for public, private and defense organizations. In order to protect valuable data or information from unauthorized access, illegal modifications and reproduction, various types of cryptographic techniques are used. [1]There are two kinds of cryptography symmetric key cryptography and asymmetric key cryptography. In symmetric key cryptography same key is used between the sender and receiver. While in asymmetric key cryptography two different keys(public key and private key) are used between sender and receiver for encryption and decryption.RSA is most famous asymmetric cryptography algorithm.

RSA

The most common public key algorithm is RSA, named for his three inventors Rivest, Shamir and Adleman (RSA). [7] In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers i.e. the factoring problem. In a secure communication using public key cryptography (RSA algorithm), the sender encrypts the message using receiver's public key, this key is known to everyone in the communication network. The encrypted message is sent to the receiving end that will decrypt the message with its private key. Only the intended receiver can decrypt the message because only receiver knows the private key. Thus using RSA algorithm we can communicate in a secure way. [2]

RSA Algorithm

1. Choose two large prime numbers P and Q.
   Let it be P=7 and Q=17.
2. Calculate N=P*Q.
   We have N=7*17= 119
3. Select the public key i.e. encryption key) E such that it is not the factor of (P-1) and (Q-1).

Let us find (17-1)*(7-1) =96
Factor of 96 are 2, 2,2,2,2 and 3(96=2*2*2*2*2*3).Thus, we have to choose E none of the factor of E is 2 and 3.As a few example we can't choose E as 4(because it has 2 as a factor), 15(because it has 3 as a factor),6(because it has 2 and 3 both as Factor).Let us choose E as 5 (it could have been any other number that does not its factors as 2 and 3).

4. Select a private key (i.e. decryption key) D such that the following equation is true
(D*E) mod (P-1)*(Q-1) =1
Let us substitute the value of E, P and Q in the equation.
We have: (D*5) mod (7-1)*(17-1) =1.
That is, (D*5) mod (6 * 16) =1.
That is, (D*5) mod (96) = 1.
After some calculation let us take D=77.Then the following is true:
(77*5) mod (96) = 385 mod 96 = 1.Which is what we wanted.

5. For encryption, calculate the cipher text CT from plain text PT as follows:
CT= (PT^E) mod N
Let us assume that we want to encrypt plain text 10.Then we have, CT= (10^5) mod 119=100000 mod 119 = 40. Send CT as a cipher text to the receiver. Send 40 as a cipher text to the receiver.

6. For decryption, calculate plain text PT from cipher text CT as follows,
PT= (CT^D) mod N.
We perform the following
PT= (CT^D) mod N. That is, PT= (40^77) mod 119=10 which is original plain text.[8]

Related Work

R. Rivest, A. Shamir, and L. Adleman has proposed a method for implementing a public key cryptosystem whose security rest in a part on the difficulty of factoring the large numbers. It permits secure communication to be established without the use of couriers to carry key and it also permits one to 'sign' digitized documents [2].

Alaa Hussein, Al-Hamami and Ibrahem Abdallah Aldariseh proposed enhancing the RSA algorithm; in this RSA algorithm they used additional third prime number in the composition of the private and public key. Because of additional prime number the factoring complexity of variable (n) is also increase. [3]

Vivek Choudhary and Mr. N. Praveen have proposed a secure algorithm in their work, which includes the architectural design and enhanced form of RSA algorithm through the use of third prime number in order to make a modulus n which is not easily decomposable by intruders. Further, their approach eliminates the need to transfer n, the product of two random but essentially big prime numbers, in the public key due to which it becomes difficult for the intruder to guess the factors of n and hence the encrypted message remains safe from the hackers. [4]

Problem concerning RSA cryptosystem: The security of RSA public key cryptosystem is based on the assumption that it is easy to multiply two large prime numbers, but it is extremely difficult to factor the product of two large prime numbers (modulus). In RSA if one can factor N into its prime numbers then the private key can be detected and security of RSA can be broken.[5]Even the limitation of using public key cryptography for encryption and decryption is its speed of computation. Its computation takes time to compute the mathematical operation of RSA algorithm. It means it takes more time for sending the messages from sender to receiver by RSA.

Solution methodology to concerned system: The security of RSA algorithm can be compromised in the network. To increase the security of computation of RSA algorithm there is a need to modify RSA algorithm. Here, in this work we are modifying the RSA algorithm by using three prime numbers, Instead of two as used in standard RSA algorithm. It gives efficient security. After multiplication of three prime numbers the value of N is so big that it is almost impossible to factorize. Even we are reducing the speed of mathematical computation of RSA algorithm .Here in this work we are using Cantor's pairing algorithm by which the RSA generates only a single integer number for messages which is sent to the receiver.

Cantor’s Pairing Function

A pairing function on set A associates each pair of members from A and generates a single integer number. Here is a classic example of a pairing function. When x and y are non_negative integers, Pair (x_, y_) outputs a single non-negative integer that is uniquely associated with that pair.

Here it is a classic example of a pairing function When x and y are non_negative integers, Pair@[x_,y_] outputs a single non_negative integer that is uniquely associated with that pair.

Pair@[x_,y_]:=Z=(x^2+3*x+2*x*y+y+y^2)/2;

The inverse function- Unpair@[Z_]:=

i= (-1+sqrt (1+8*Z))/2);
x= Z-i(I+i)/2 , y=i(3+i)/2-Z [6]

Modified RSA Algorithm

The modified RSA takes four phases for delivering its data to the destination

Phase (i): Pairing Algorithm:

Here by cantor’s pairing function, data as a number of characters (chunk wise) generates only single integer number. This single integer number is transferred in the form of plain text.

Phase (ii): Encryption:

Encryption converts Plain Text (clear text) into cipher text .Encryption sends confidential message over an insecure channel. This process needs a key (public key) and encryption algorithm. The encryption takes to the sender side as given below

CT= (PT^E) mod N.

Here CT stands for cipher text.
PT stands for plain text.
N stands for product of three prime numbers.
E stands for public key.

Phase (iii): Decryption:

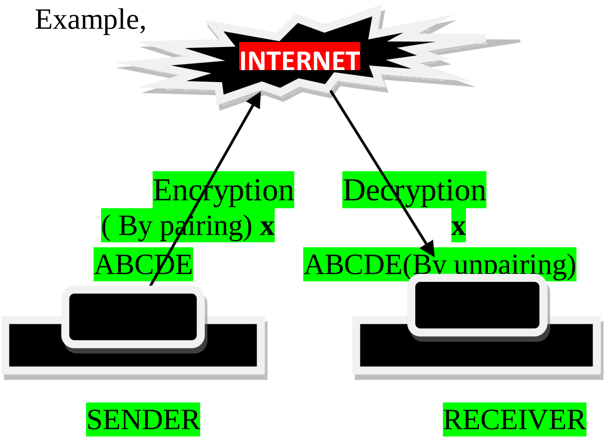
The process of decryption converts cipher text into plain text. This is reverse process of encryption. This is algorithm acts to the receiver side for converting messages from non readable format into the readable format. Process of decryption needs a key (private key) and decryption algorithm as given below

PT= (CT^D) mod N.
PT stands for plain text.
CT stands for cipher text.
N stands for product of three prime numbers.
D stands for private key.

Phase (iv): Unpairing algorithm:

Unpairing algorithm displays the single integer number(generated by the pairing function) as a group of characters to the receiver side those were actually sent.

Example,



As the diagram shown above a message ABCDE is sent by the sender to the receiver .The message ABCDE is converted first into an integer number **x** by pairing function. After encryption it is converted into a cipher text. By decryption it is converted into the plain text **x** (which have been generated by pairing function) to the receiver side. And by unpairing function to the integer number **x**, the message ABCDE is displayed to the receiver side (in the form of individual data item).The integer number **x** which is generated by pairing function is large, which is not easy to break by any kind of attack. As we increase the number of characters the generated integer number by pairing function becomes very large.

Output

ABCDE

275770849973326590396551516958.

(Generation of integer number by cantor's pairing function to the message ABCDE to the sender side as a plain text)

P1=108086391056891903

P2=179426083

P3=179426089

(P1, P2 and P3 are prime numbers)

N=3479703045951393714159943169103461

(N=P1*P2*P3)

Q=3479703007164357595645579286804832

(Q= (P1-1) * (P2-1) * (P3-1))

E =5

(E is public key, E= (E.gcd (Q) ===1))

D=1391881202865743038258231714721933

(D is private key, D = E.modInverse(Q))

CT=2385823539298465703514908727249098

(CT is cipher text, CT= PT^E mod N)

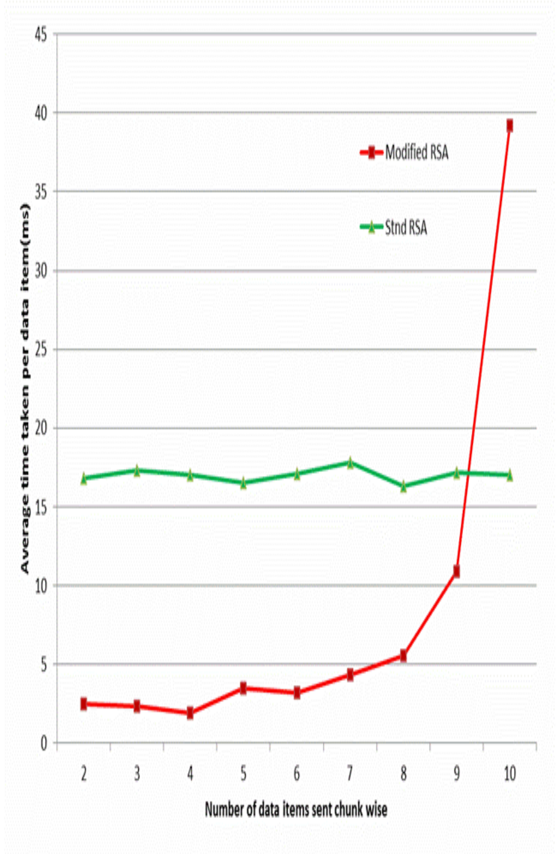
PT=275770849973326590396551516958

(PT is a plain text to the receiver side, PT=CT^D mod N)

By cantor's unpairing function generation of message ABCDE will be displayed to the receiver side.

Results and Discussion

In our experiment we have modified RSA cryptosystem by using three prime numbers instead of two as used in standard RSA. The three prime numbers are large, which provides excellent security through the network [4]. In this work we have minimized computation time in our proposed framework compared to standard RSA algorithm. Group of data items are combined together to form a chunk. Then the chunk is encrypted to form cipher text. Then the cipher text is sent to the receiver. At receiver, the cipher text is decrypted to form the plain text (integer number by the chunk of data items). Then the chunk is unpaired to form individual data items.



Graph: Comparative analysis item of proposed framework & standard RSA (with respect to average time taken per data)

In graph we have plot average time taken per data item vs. number of data items sent for our proposed algorithm and standard RSA. From the graph we can conclude that our proposed algorithm outperforms standard RSA as long as the chunk size is nine. Beyond chunk size of nine our algorithm starts taking more time. This graph clearly states that the optimal size of data item that can be clubbed together to form a chunk is nine for the modified RSA.

Modified RSA works well up to nine data items per chunk with three prime numbers in comparison to standard RSA takes less time for delivering its data to the destination. Even security provided by Modified RSA is excellent as compared to standard RSA.[4]

References

[1] Abdel-karim, Ahmed F. Shalash & Naglaa F developed "Modification on RSA cryptosystem using Genetic optimization"

[2] "Implementation of Modified RSA Cryptosystem Based on Offline Storage and Prime Number" by Ms. Ritu Patidar and Mrs. Rupali Bhartiya IJCAT International Journal of Computing and Technology, Volume 1, Issue 2, March 2014 ISSN: 2348 – 6090 www.IJCAT.org

[3] Al-Hamami, A. H., & Aldariseh, I. A. (2012, November). Enhanced Method for RSA Cryptosystem Algorithm. In Advanced Computer Science Applications and Technologies (ACSAT), 2012 International Conference on (pp. 402-408). IEEE.

[4] Vivek Choudhary1 and Mr. N. praveen2 "Enhanced RSA Cryptosystem Based On Three Prime Numbers" 1 Post Graduate Scholar, Department of Computer Science & Engineering, SRM University, Chennai, Tamilnadu, India 2 Assistant Professor, Department of Computer Science & Engineering, SRM University, Chennai, Tamilnadu, India.

[5] "Introduction to RSA and to Authentication" Fall 2006 Christensen MAT/CSC 483.

[6] "An Elegant Pairing Function", Matthew Szudzik, Wolfram Research, Inc. NKS 2006Wolfram Science Conference

[7]BEHROUZ A FOROUZAN "Data communication and networking" TMH Publication

[8] Atul Kahate, Cryptography and Network Security, Tata McGraw-Hill Publishing Company Limited.