International Journal of Electrical and Computer Engineering (IJECE)
Vol. 8, No. 5, October 2018, pp. 3949~3953
ISSN: 2088-8708, DOI: 10.11591/ijece.v8i5.pp3949-3953 

# Enhanced RSA Cryptosystem based on Multiplicity of Public and Private Keys

**Ahmed Eskander Mezher**
Department of Informatics Systems Management, Businesses Informatics College,
University of Information Technology and Communications, Baghdad, Iraq

**Article Info**

***Article history:***

Received Oct 20, 2017
Revised Jan 20, 2018
Accepted Jul 7, 2018

***Keyword:***

Computer science
Computer security
Data security
Public cryptography
RSA algorithm

**ABSTRACT**

Security is one of the most important concern to the information and data sharing for companies, banks, organizations and government facilities. RSA is a public cryptographic algorithm that is designed specifically for authentication and data encryption. One of the most powerful reasons makes RSA more secure is that the avoidance of key exchange in the encryption and decryption processes. Standard RSA algorithm depends on the key length only to protect systems. However, RSA key is broken from time to another due to the development of computers hardware such as high speed processors and advanced technology. RSA developers have increased a key length or size of a key periodically to maintain a high security and privacy to systems that are protected by the RSA. In this paper, a method has been designed and implemented to strengthen the RSA algorithm by using multiple public and private keys. Therefore, in this method the security of RSA not only depends on the key size, but also relies on the multiplicity of public and private keys.



***Corresponding Author:***

Ahmed Eskander Mezher,
Department of Informatics Systems Management,
Businesses Informatics College,
University of Information Technology and Communications,
Baghdad, Iraq.
Email: ahmed.mezher@uoitc.edu.iq

## 1. INTRODUCTION

Encryption is the way of transforming a message to another form that is completely different from its original one and it is difficult to read by an intruder. This enables people to secure sensitive information to be sent over the network. There are two main kinds of cryptography which are symmetric and asymmetric. In symmetric cryptography, the key used for encryption is the same key used for decryption process [1], [2]. There are many well-known algorithms of this kind of cryptography such as DES, AES, etc. While the other kind of cryptography is asymmetric. In asymmetric cryptography, there are two keys which are public key and private key, one for encryption and the other used for decryption [3]. The last type of cryptography is considered the most revolutionary change in the cryptography science [4], [5]. RSA is considered one of the most effective algorithm that can be used for both encryption and digital signature [6]-[8]. The strength of the algorithm depends on the factorization problem. The factorization is considered as challenging problem to mathematicians for many decades because it is NP complete problem [9]. Therefore, RSA strength comes from the previous point. In addition, the algorithm is well-known cryptographic algorithm that is invented by Shamir and Adelman in 1977 [10]. In the RSA algorithm, encryption and decryption can be performed by using two keys, one of them is public while the other one is private. To encrypt a message, the public key is used, then the cipher text and the public key are sent to a receiver. The receiver use their own private key to decrypt the cipher text. The private key will not send over a network during the sending process, and this is how public cryptography works.

### 1.1. Encryption of standard RSA

a. The algorithm uses two prime numbers X and Y.
b. N=X*Y.
c. Find Ø (N), which is (X-1) (Y-1).
d. Find K, which is GCD (K, Ø (N) =1. ( K is the public key)
e. $C=M^K$ mod N

### 1.2. Decryption of standard RSA

a. Find D, which is D*K mod Ø (N) =1.
b. M= $C^D$ mod N.

Where:

M is the plain text or the original message.
C is the cipher text or secret message.

Standard RSA algorithm was designed by Adleman and Shamir in 1977. The algorithm is considered the first public cryptographic system [11]. Rivek and Praveen proposed a method to enhance the security of RSA. The method stated that using three prime numbers instead of two would increase the security of RSA. Therefore, N is the composite number of multiple three prime numbers P, Q and R. They increased the strength of RSA by using third prime number [12]. Patidar and Bahritya proposed a method to increase the speed of RSA encryption and decryption because mathematical operations in the RSA algorithm take too much significant time. Their method depends on offline storage and three prime numbers. All parameters of RSA such as N, Ø (N), Public key (E), and private key (D) are stored in a table. All previous values of RSA are stored in a big database before the establishment of RSA process [13]. Mila Bahadori, Mohammed Reza, Omid Sarbishei, Mojtaba Atarodi and Mohammed Sharif implemented a method to increase the speed of public key and private key generation. A method was based on using a smart card which contained mini processor and random number generator. The smart card calculated public and private keys before encryption and decryption processes establishment [14]. Yi-Shiung Yeh and Chia-Yaochen implemented a method to reduce time consumption of RSA decryption process since the decryption process takes more computations than the encryption process. The method increased the speed of traditional decryption process three times faster than the traditional decryption algorithm based on Chinese Remainder Theorem only [15].

## 2. NEW PROPOSED ALGORITHM

In this paper, we have proposed new RSA algorithm using multiple public keys and multiple private keys. The algorithm is summarized as follows:

a. Choose two prime numbers R1 and R2.
b. N=R1*R2.
c. Find Ø (N) =Ø (R1)* Ø (R2). Since R1 and R2 are prime numbers, so Ø (R1) = (R1-1) and Ø (R2) = (R2-1).
d. Choose a series of public keys, not just one public key as in the standard RSA such as E1, E2, and E3… Em. Where E is a public key and m is the number of public keys that is used to encrypt a message. All public keys E1, E2, E3… Em are relatively prime to Ø (N). That means there are no factors between the above public keys and Ø (N). This step is so important in order to make sure each public key has an inverse number (i.e. Private Key) which will be used later in the decryption process to decrypt the message.
e. Calculates multiple secret keys D1, D2, D3…Dm. Where D is an inverse of E and m is the number of private keys.

We demonstrated the above steps as a flowchart in Figure 1. In the encryption process, a message would be encrypted several times using multiple different public keys in our suggested improvement to the RSA algorithm. How many times the message should be encrypted precisely? The message would be encrypted by using all possible public keys or some of them. We know that the public key E is any number that has no factors with Ø (N). Thus, in our suggested algorithm we selected all numbers that has no factors with Ø (N) to be our list of public keys, and we encrypted the message using all the previous public keys or some of them. We have introduced a mathematical formula to count the number of encryptions that is used in our suggested improvement to the RSA algorithm. (Ø (Ø (N)-1) is the number of numbers that have no factors with Ø (N). Therefore, (Ø (Ø (N)-1) is the number of public keys that would be used to encrypt the

message. The cryptographer has the choice to choose some public keys from the above formula to be used in the encryption process. For Example: N=13, Ø (N) =12, Ø (Ø (N)-1) =3. Now, three numbers only in a set of 12 qualified to be the public key E which are 5, 7, and 11. Therefore, we would use those numbers to encrypt the message three times using three public keys 5, 7, and 11. In the decryption process, we should find how many times to encrypt the messages in order to generate private keys, which is equal to the number of public keys that is used to encrypt the message. In the same previous example, if N=13 then (Ø (Ø (N)-1) = 3. Then we should find three private keys which are d1, d2 and d3. After that, we would use the private keys to decrypt the message in a reverse order. Therefore, our improvement to the algorithm makes RSA depends not only on the key length to decrypt the message, but also depends on the several public keys and private keys. For example, if the attacker by some way knew the two prime numbers of the composite number N, then he couldn't decrypt the message because the message is encrypted several times using multiple public keys. To decrypt the message, the attacker needs to find all private keys not just one private key as in the standard algorithm.

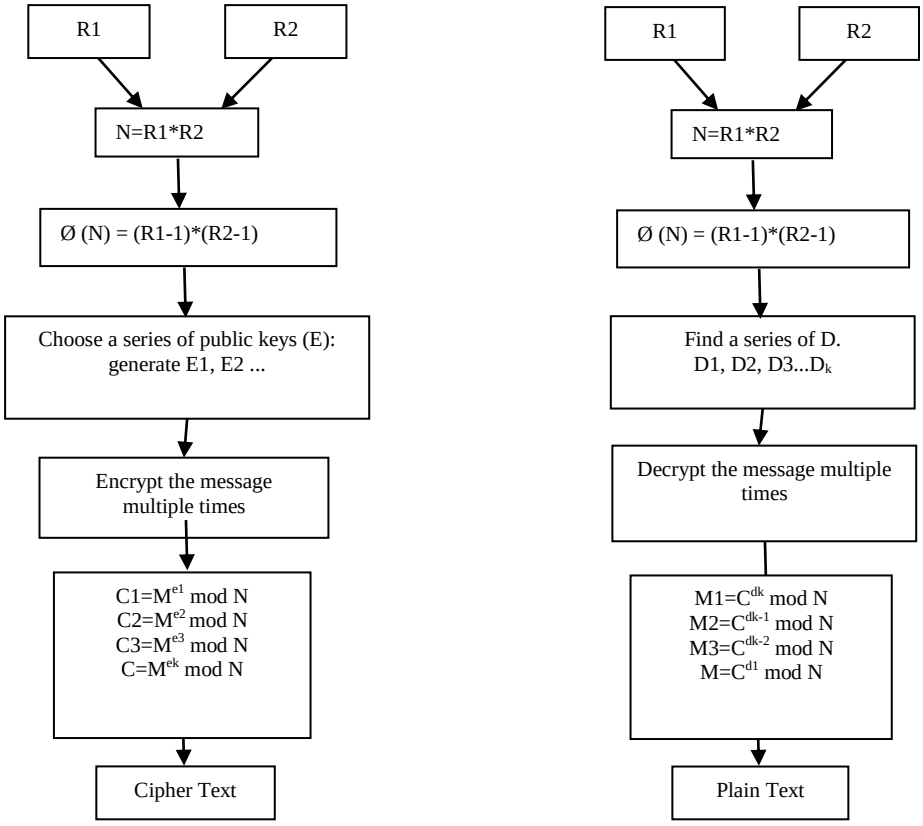


Figure 1. The new proposed RSA algorithm for Encryption and decryption processes

## 3. RESEARCH AND METHOD

In order to see the effectiveness of the new enhanced algorithm, we tested the powerful of the algorithm against brute force attack. The brute force is considered one of the most well-known attacks that used by hackers to break security systems. We did the implementation to see which algorithm takes much time to break than the other. We implemented the brute force attack in C++ to attack our new algorithm and the standard RSA algorithm. The brute force attack written in C++ to factor the composite number N for two prime numbers and finding the private key. In the standard algorithm, the brute force attack will factor N and one private key only then eventually terminated. While in the new algorithm, the brute force will factor N and finding several private keys. Because of data is encrypted several times using several public keys in the algorithm, the brute force attack will decrypt the message several times in order to get the original data.

## 4. RESULTS AND DISCUSSION

In this paper, we intentionally attack RSA to see the powerful of our algorithm. We used brute force attack to attack standard RSA and our improved RSA using different key size and size of N to see which is stronger than the other. The strength of RSA depends on the private key. To attack RSA, we needs to attack the private key. Therefore, in our research, we attacked a text that encrypted using standard RSA and the improved RSA algorithm. The results of attacking standard and improved RSA algorithms are showed in Table 1, Table 2 and Figure 2 respectively. From the results in Table 1 and Table 2, it is clearly obvious that our improved technique makes RSA stronger than the standard one. When the size of N is 12 in Table 1 and Table 2 that means N is a composite number consists of 12 digits in length, this number is a product of only two prime numbers.

In the standard RSA algorithm, it takes approximately 39 milliseconds to break the message while the improved algorithm takes approximately 347 milliseconds to break the message. In all different sizes of N (N is the length of private key) 7, 8, 9, 10, 11, and 12, the improved RSA is slower approximately 9 times than the standard RSA when brute force attack is used. Therefore, an attacker needs much time to break a system when the new algorithm is used. In the standard RSA algorithm, encryption and decryption is done by using one public key and one private key. If the attacker by some way or using a program with high speed computer figure out the private key, the system would be broken. While in our new algorithm, the encryption and decryption is done using multi public and private keys. Therefore, the attacker needs to know all the private numbers to break the system. The results in Table 2 are showing slowness to break RSA algorithm while the results in Table 1 are showing speedup in attacking the standard RSA algorithm. Therefore, the new algorithm harder to break than the standard algorithm. V. Choudhary and N. Praveen improved RSA algorithm by using three prime numbers while the improvement in our algorithm is using multi public and private numbers (unlimited number) in order to be hard to break [12].

Table 1. Attacking Standard RSA using Brute Force

| Size of N | Time to break RSA in milliseconds |
|---|---|
| 7 | 0.002 |
| 8 | 0.002 |
| 9 | 0.561 |
| 10 | 4.206 |
| 11 | 12.110 |
| 12 | 38.561 |

Table 2. Attacking the Improved RSA using Brute Force

| Size of N | Time to break RSA in milliseconds |
|---|---|
| 7 | 0.018 |
| 8 | 0.02 |
| 9 | 6.171 |
| 10 | 37.854 |
| 11 | 133.21 |
| 12 | 347.049 |

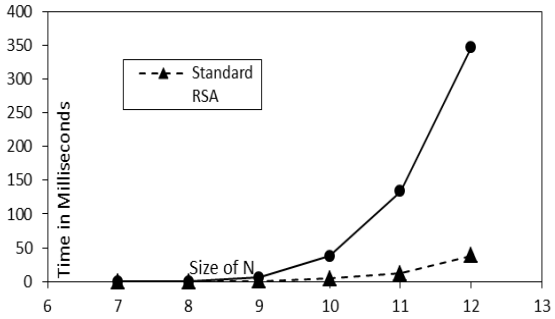


Figure 2. Comparison between Standard RSA and improved RSA

## 5. CONCLUSION

Encryption is the way of making data secure against different kinds of attacks and hackers. One of the most effective public cryptographic algorithms is RSA. Since RSA designed, there are many developments to the algorithm because of the advances in the computers and technology. The algorithm is broken several times. RSA laboratories increase the security strength of RSA by changing the length of private key from time to another. Rivek and Praveen enhanced the algorithm by using three prime numbers instead of two as in the original algorithm. Patidar and Bahritya improved the speed of the algorithm by using offline storage. In the standard RSA, one public key, one private key and N (Composite number of two primes) are the basic elements that are used in the encryption and decryption processes. In this paper, we suggested an improvement to the original algorithm to make it harder to break by using several public keys and several private keys for encryption and decryption processes. The brute force attack is used to attack both

algorithms (i.e. the standard and our improved algorithms). Our improved algorithm is more stable and stronger than standard algorithm against brute force attack. In addition, the improved algorithm is almost 9 times slower to break than standard algorithm when different sizes of keys are used.

## REFERENCES


[1] G. Simmons, "Symmetric and Asymmetric Encryption", *ACM Computing Surveys (CSUR)*, vol. 11, no. 4, pp. 305-330, 1979.

[2] PGP Corporation, "An introduction to Cryptography", Santa Clara, CA, USA, pp. 33, 45, 60, 2002.

[3] N. Babu, *et al*., "Implementation of High Security Cryptographic System with Improved Error Correction and Detection Rate using FPGA", *International Journal of Electrical and Computer Engineering*, vol. 6, no. 2, p. 602.

[4] Y. Kumar, *et al*., "Comparison of Symmetric and Asymmetric Cryptography with Existing Vulnerabilities and Countermeasures", *International Journal of Computer Science and Management Studies*, vol. 11, no. 3, 2011.

[5] D. Benne, "Cryptanalysis of RSA with small prime difference", *Applicable Algebra in Engineering, Communication and Computing*, vol. 13, no. 1, pp. 17-28, 2002.

[6] P. Singh and R. K. Chauhan, "A Survey on Comparisons of Cryptographic Algorithms using Certain Parameters in WSN", *International Journal of Electrical and Computer Engineering*, vol. 7, no. 4, pp. 2232, 2017.

[7] J. Espalmado and E. Arboleda, 'DARE Algorithm: A New Security Protocol by Integration of different Cryptographic Techniques", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 2, pp. 1032-1041, 2017.

[8] S. Bergmann, "Degenerate keys for RSA encryption", *ACM SIGCSE Bulletin*, vol. 41, no. 2, pp. 95-98, 2009.

[9] U. Somani, *et al*., "Implementing Digital Signature with RSA Encryption Algorithm to Enhance the Data Security of Cloud in Cloud Computing", *2010 First International Conference on Parallel, Distributed and Grid Computing*, 2010.

[10] R. Rivest, *et al*., "A Method for Obtaining Digital Signatures and Public-key Cryptosystems", *Communications of the ACM*, vol. 21, no. 2, pp. 120-126, 1978.

[11] A. Al-Hamami and I. Aldariseh, "Enhanced Method for RSA Cryptosystem Algorithm", *2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), Kuala Lumpur*, pp. 402-408, 2012.

[12] V. Choudhary and N. Praveen, "Enhanced RSA Cryptosystem based on Three Prime Numbers", *International Journal of Innovative Science, Engineering & Technology*, vol. 1, no. 10, 2014.

[13] R. Patidar, *et al*., "Modified RSA Cryptosystem based on Offline Storage and Prime Number", *Computational Intelligence and Computing Research (ICCIC), 2013 IEEE International Conference on, IEEE*, 2013.

[14] M. Bahadori, *et al*., "A novel approach for secure and fast generation of RSA public and private keys on SmartCard", *Proceedings of the 8th IEEE International NEWCAS Conference 2010, Montreal, QC*, pp. 265-268, 2010.

[15] R. Hwang, "An Efficient Decryption Method for RSA Cryptosystem," *Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on*, IEEE, vol. 1, pp. 585-590, 2005.


## BIOGRAPHY OF AUTHOR

**Ahmed Eskander Mezher** received the Bachelor's degree in Computer science from University of Technology, Baghdad-Iraq, in 2009, and the Master degree from University of Colorado Denver-United States of America in 2013. He is currently teacher assistant at the University of Information technology and Communications, College of Business informatics, Baghdad-Iraq. His research interest include Computer security, Network security, Networks, Algorithms for communication networks.