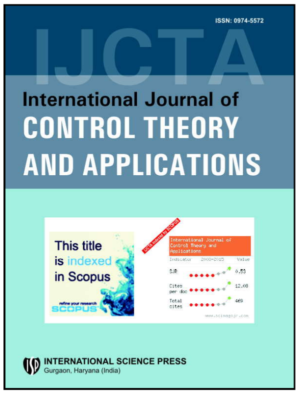

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 13 • 2017

Efficient RSA Cryptosystem with Key Generation using Matrix

Prerna Verma[1], Dindayal Mahto[2], Sudhanshu Kumar Jha[3] and Dilip Kumar Yadav[4]

[1,2,3,4] Department of Computer Applications National Institute of Technology Jamshedpur, Jharkhand (India), Email: prerna9verma@gmail.com, dindayal.mahto@gmail.com, sudhanshukumarjha@gmail.com, dkyadav.ca@nitjsr.ac.in

Abstract: RSA is one of the popular public key cryptography algorithms, whose security is based on factorization of RSA modulus, however, this may be vulnerable or invulnerable depends on the value of the modulus. In this paper, large RSA modulus is generated using a matrix. This paper exhibits very small order of the matrix, i.e. the matrix is of order four is sufficient to generate key (private key) and the modulus of more than 256 digits, hence instead of storing a large key, proposed method stores small order matrix and three random numbers of 10 digits, thus, in terms of space requirement, this algorithm is better than the conventional / standard RSA. In this paper small public keys are used during encryption to reduce the encryption time while Chinese Remainder Theorem (CRT) is used during decryption to reduce the decryption time. Hence, the proposed model is more secure and efficient in terms of space requirement and time complexity.

Keywords: RSA, Encryption, Decryption, Chinese Remainder Theorem, Asymmetric Key Cryptography

1. INTRODUCTION

Cryptography is an art and science of secure communication, in which mathematical concepts are applied to encode messages (i.e., plain text) into unintelligible form, known as cipher text. In the RSA cryptosystem there are two types of problem – hard problems and easy problems. Hard problems are those in which operations take time in the $O(2^{||a||})$ where "$||a||$" is the bit length (or number of digits present in the input) of the input "a". Examples of hard problems are based on factorization of numbers, based on discrete logarithm problems. Easy problems are those in which time taken by the operations are in the $O(||a||)$ where "$||a||$" is the bit length (or number of digits present in the input) of the input "a" . Example of easy problems based on multiplication operations, modular arithmetic operations.

As we know the security of the RSA cryptosystem depends on the factorization of the modulus "n" (which falls into hard problems category in cryptosystem) used in the RSA. But the computational power of the hardware has increased drastically and still increasing, so for secure RSA cryptosystem the "n" should be at least 100 digits. Hence, its two prime factors should be at least 50 digits.

RSA is popular public-key cryptosystem, because it has very simple algorithms to do operations like key generation, encryption and decryption. However, as per today's industry needs of better communication security, a large value of modulus of RSA is used; which in turn requires high computational cost. The other main disadvantage of RSA is the large space occupied by the key pairs. In addition, as the security level is increased the RSA key size grows at a much faster rate. For a detailed discussion about key sizes, see Lenstra [13]. In order to overcome these drawbacks, many researchers have proposed variants of RSA which either reduce the computational costs [8], [11], [14], [17], [18], [19], or reduce the (key) storage requirements [12], [30]. Hence, this paper focus in reducing the (key) storage requirement as well as computational cost of RSA by using small value of public key for fast encryption and CRT for fast decryption. Hence, this proposed model is more efficient in terms of time complexity and space requirement.

The remaining part of this paper is organized as follows Section 2 provides detailed of the proposed model with operations like key generation, encryption and decryption. Section 3 analyses the proposed model and compares it with the standard RSA. Section 4 concludes the paper.

2. PROPOSED MODEL

This proposed efficient RSA public cryptosystem is an enhanced version standard RSA cryptosystem. As it is an asymmetric cryptosystem, it uses pair of keys; one key is used to encrypt the data in such a way that it can only be decrypted with the other key. In this model, the key pairs are generated using matrix while to generate keys pair in standard RSA, random numbers is used. As per the Table 1, in order to provide highest level of security, the key size should be greater than or equal to 100 digits.

Table 1
Modulus size with their secured life period [2]

Digits	*Number of operations*	*Time Require*
50	$1.4 * 10^{10}$	3.9 hours
75	$9.0 * 10^{12}$	104 days
100	$2.3 * 10^{15}$	74 years
200	$1.2 * 10^{23}$	$3.8 * 10^{9}$ years
300	$1.5 * 10^{29}$	$4.9 * 10^{15}$ years
500	$1.3 * 10^{39}$	$4.2 * 10^{25}$ years

2.1. Key Generation Algorithm:

This algorithm comprises of two sub algorithm:

(i) Create a square matrix of random number (i.e., 5*5).

(ii) Generation of public-key and private-key of proposed model: In this sub – algorithm we obtain public and private key using special matrix obtained from the first sub algorithm.

(i) Generation of Square Matrix of Random Number

(a) Declare a matrix of size 5*5(or we take 8*8 or 16*16 i.e. square matrix of small order) , the purpose of using matrix of small size ,as we see later that the small matrix of size 2*2, can capable of generating large key size.

(ii) Generation of Public and Private Key

(A) In order to generate "N", proposed model does following steps:

(a) Takes the matrix which obtained from Algorithm (i) i.e. mat[5][5].

(b) Generates a random number

(c) Add this random number to each element of the given matrix. (Because in the subsequent steps use multiplication operation with matrix element, it prevents from vanish of result because matrix may contain element with zero value.

(d) Do multiplication operation within matrix elements.

(e) Generate a random number and add it with the resultant value.

(f) It is not necessary that it would be a prime number, so use next prime function to generate prime number which is greater than the above value.

This value is nothing but the value of the first prime i.e., P.

(g) In the similar manner we generate second prime number i.e., Q. We use same matrix for "P" and "Q", so to distinguish "P" and "Q" we use other random number which is different from those which was used in generation of "P".

(h) Now we calculate N=P*Q [4].

Generation of "Public Key" and "Private Key":

Select small value of public key i.e., "E" so that it is co-prime with phi(N),it means GCD(E, phi(N))=1 . Now we have a partial public key and we have to generate partial private key "D" for this we have to use inverse function.

2.2. RSA Encryption

$C = M^E \bmod N$ (where M is plain text and C is cipher text) [4]

2.3. RSA Decryption[15]

- Let $C_1 = C(\bmod P)$
- Let $C_2 = C(\bmod Q)$
- Let $D_1 = D (\bmod (P-1))$
- Let $D_2 = D (\bmod (Q-1))$
- Let $M_1 = C_1^{D1} (\bmod P)$
- Let $M_2 = C_2^{D2} (\bmod Q)$
- So , $M = [((M_2+Q-M_1)A)(\bmod Q)]P + M_1$
- Where $P < Q$ and $0<A<Q-1$ and $A_P = 1(\bmod Q)$.

3. ANALYSIS OF PROPOSED MODEL

3.1. Comparison between simple key generation and modified key generation algorithm

From the Figure 1 and Figure 2 we can compare the key length generated by the standard RSA and the proposed model. From the Figure 1 it can be observed that using 10 digits random numbers proposed model generate key of size 78 digits and the standard RSA generates 20 digits. Hence, the proposed model can generate the desired key size very efficiently than standard RSA.

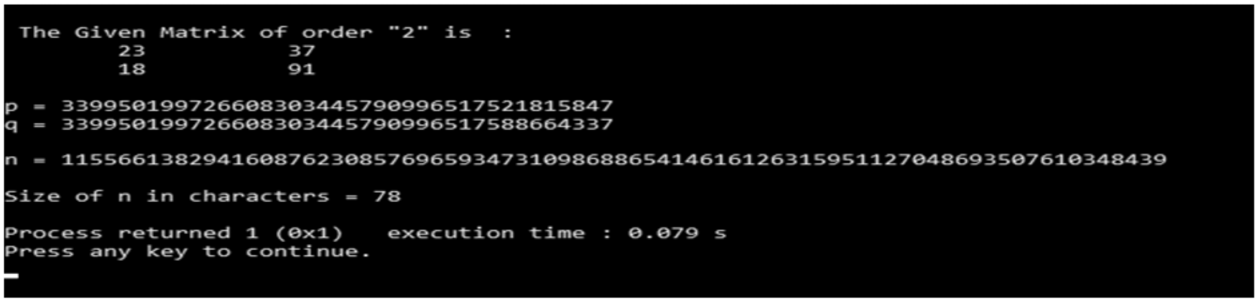

Figure 1: Snapshot of output of proposed model

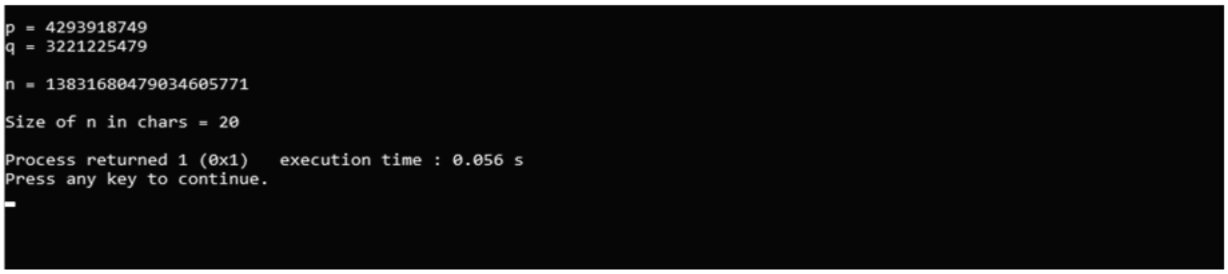

Figure 2: Snapshot of output of standard RSA

3.2. Key length comparison within modified key generation algorithm of matrix of same order with elements of different magnitude

From the Figure 3 and Figure 4 we can see that the key length does not depend on the values stored in the matrix. It means how large value (till the value less than 10 digits) we store in the matrix the key length is same until order of the matrix is same.

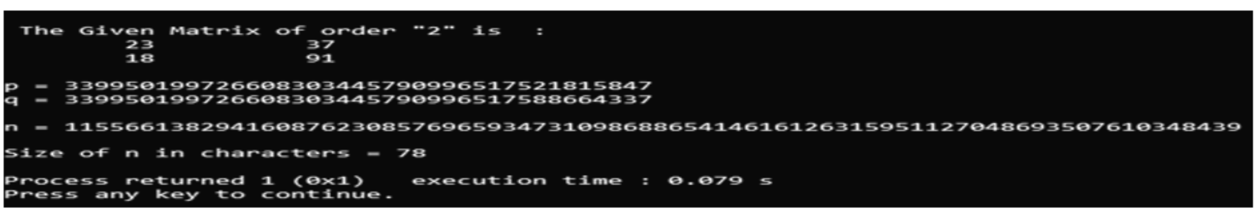

Figure 3: Snapshot of output of proposed model of order 2 with low values martix elements

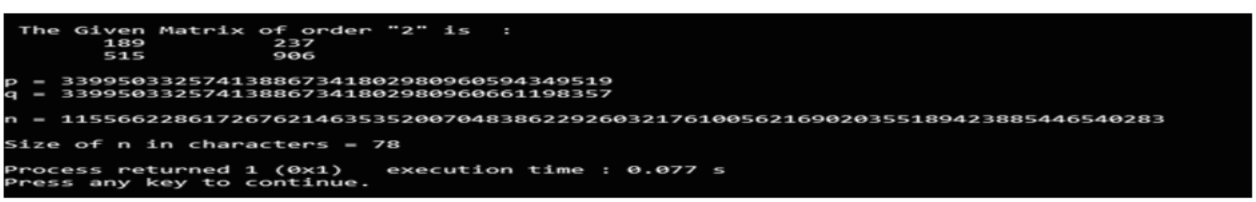

Figure 4: Snapshot of output of proposed model of order 2 with high values martix elements

3.3. Key length comparison within modified key generation algorithm with different matrix orders

From the Figures 5-7 we can compare the key length generated by the proposed model with different matrix orders. From the pictures we can see that the key length does depend on the order of the matrix. From the

Figure 5: Snapshot for key generation of key length 174 digits using matrix of order 3

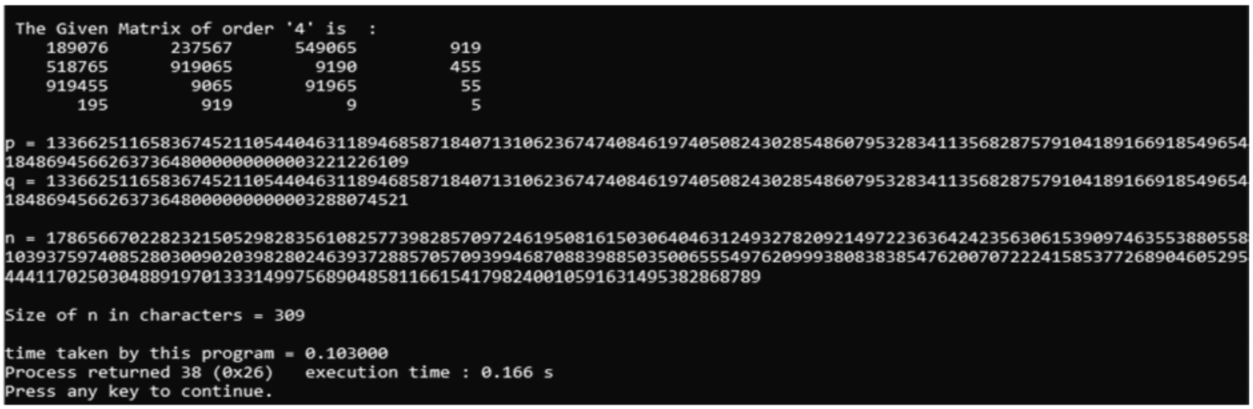

Figure 6: Snapshot for key generation of key length 309 digits using matrix of order 4

Figure 7: Snapshot for key generation of key length 501 digits using matrix of order 5

“Figure 5” we can see that matrix of order 3 is sufficient for generating required key length (i.e.,>100 digits as mentioned in the Table 1 the modulus of size more than 100 digits are more secure).

3.4. Average key generation running time of proposed model and standard RSA

From the Table 2 and Figure 8, we can observe that the proposed model has the same time complexity as the standard RSA.

Table 2
Time complexity of key generation algorithms

Modulus Length (in digits)	*Execution Time (in milliseconds)*	
	Proposed Model	*Standard RSA*
78	5.701	4.117
174	20.233	20.108
309	175.638	142.149
501	1054.094	1250.346

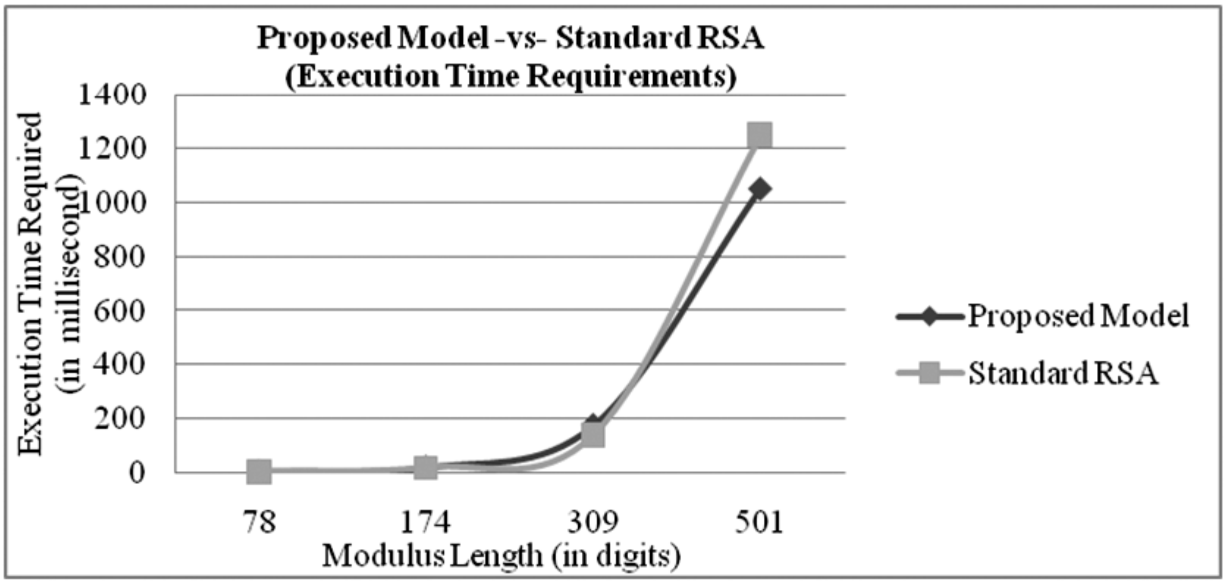

Figure 8: Time complexity of key generation algorithms

3.5. Space requirement of proposed algorithm

Instead of storing the key of 2048 bits length we store matrix of order 4 and three random numbers of 80 bits. But in standard RSA of 2048 bits length key, one needs storage of 2048 bits. Hence the proposed model is better in terms of space requirement than standard RSA. From the Table 3 and Figure 9, we can observe that the proposed model is more efficient in term of space requirement.

Table 3
Space requirements of proposed model and standard RSA

Modulus Length (in digits)	*Space Required (in digits)*	
	Proposed Model	*Standard RSA*
78	38	79
174	48	174
309	62	309
501	80	501

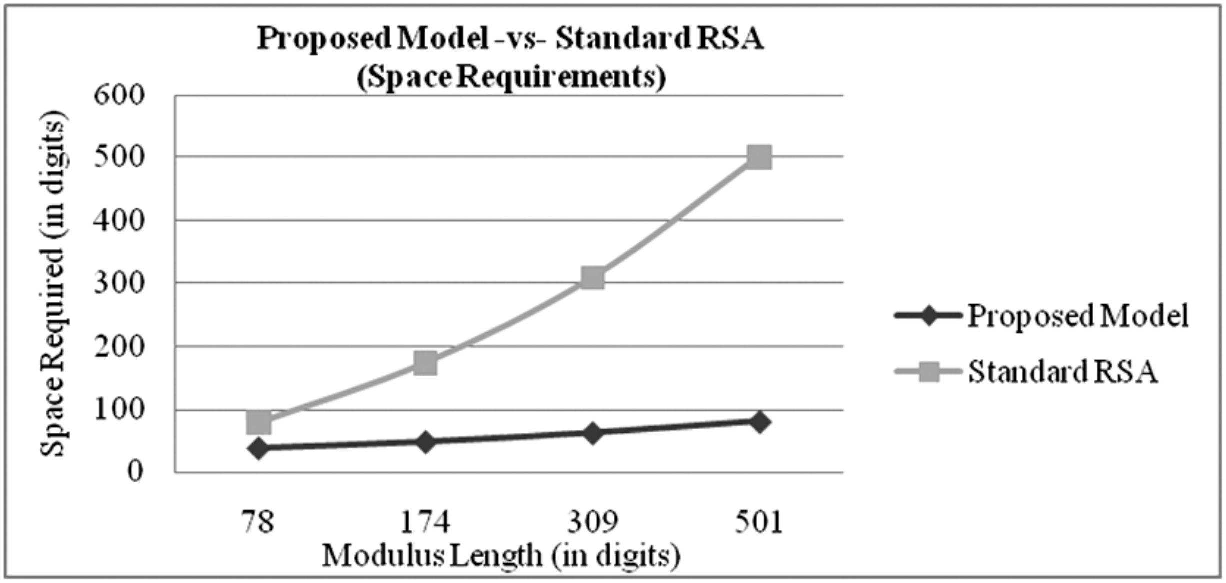

Figure 9: Space requirement of proposed model and standard RSA

4. CONCLUSION

The Security of the RSA solely depends on the key size i.e., "key generation algorithm", In RSA algorithm key generation depends on platform, i.e. if the platform (i.e. computer system) generates a strong key then the our cryptosystem is strong but in the RSA algorithm they don't focus on intelligent method to generate a big prime number "P" and "Q" so that "N" would be large (i.e. at least of size 100 digits). The proposed model ensures high security than RSA because it guarantees to generate large key. Hence, from the above observation we can see that the time complexity of key generation part of proposed model is same as standard RSA, but in proposed model decryption is done using Chinese Remainder Theorem hence decryption time of this is less than standard RSA. From previous sub-section we can observe that the proposed model is better in terms of space requirement than standard RSA.

REFERENCES

[1] Sun, Hung-Min, et al. "Dual RSA and its security analysis." IEEE Transactions on Information Theory 53.8 (2007): 2922-2933

[2] Swami, Balram, Ravindar Singh, and Sanjay Choudhary. "Dual Modulus RSA Based on Jordan-totient Function". Procedia Technology 24 (2016): 1581-1586

[3] S. Thajoddin & S. Vangipuram. "A Note on Jordan's Totients Function". Indian J. pure appl. Math. 19(12):1156-1161, December 1988.

[4] R. Rivest, A. Shamir and L. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", Communications of the ACM, 21 (2), February 1978, pages 120

[5] D. Boneh, "Twenty years of attacks on the RSA cryptosystem," *Notices of the American Mathematical Society*, vol. 46, no. 2, pp. 203–213,1999.

[6] D. Boneh and G. Durfee, "Cryptanalysis of RSA with private key d less than N ," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1339–1349 , Jul. 2000.

[7] D. Boneh, G. Durfee, and Y. Frankel, "An attack on RSA given a small fraction of the private key bits," in *Advances in Cryptology—ASIACRYPT'98*, ser. Lecture Notes in Computer Science, K. Ohta and D.Pei, Eds. New York: Springer, 1998, vol. 1514, pp. 25–34.

[8] D. Boneh and H. Shacham, "Fast variants of RSA," Crypto Bytes, vol.5, no. 1, pp. 1–9, 2002.

[9] D. Coppersmith, "Finding a small root of a univariate modular equation," in *Advances in Cryptology—EUROCRYPT'96*, ser. Lecture Notes in Computer Science, U. M. Maurer, Ed. New York: Springer,1996, vol. 1070, pp. 155–165.

[10] D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter, "Low-exponent RSA with related message," in *Advances in Cryptology—EUROCRYPT'96*, ser. Lecture Notes in Computer Science, U. M. Maurer, Ed. : Springer, 1996, vol. 1070, pp. 1–9.

[11] M. J. Hinek, "Another look at small RSA exponents," in *Topics in Cryptology-CT-RSA 2006*, ser. Lecture Notes in Computer Science, D. Point cheval, Ed. New York: Springer, 2006, vol. 3860, pp. 82–98.

[12] A. K. Lenstra, "Generating RSA moduli with a predetermined portion," in *Advances in Cryptology—ASIACRYPT'98*, ser. Lecture Notes in Computer Science, K. Ohta and D. Pei, Eds. New York: Springer,1998, vol. 1514, pp. 1–10.

[13] A. K. Lenstra, "Unbelievable security. Matching AES security using public key systems," in *Advances in Cryptology—ASIACRYPT'01*, ser. Lecture Notes in Computer Science, C. Boyd, Ed. New York: Springer, 2001, vol. 2248, pp. 67–86.

[14] G. Qiao and K.-Y. Lam, "RSA signature algorithm for microcontroller implementation," in *Smart Card Research and Applications,CARDIS'98*, ser. Lecture Notes in Comput. Sci., J.-J. Quisquater and B. Schneier, Eds. New York: Springer, 1998, vol. 1820, pp. 353–356.

[15] J.-J. Quisquater and C. Couvreur, "Fast decipherment algorithm for RSA public key cryptosystem,"*Electron. Lett.*, vol. 18, no. 21, pp. 905–907, Oct. 1982.

[16] H.-M. Sun and M.-E.Wu, "An approach towards Rebalanced RSA-CRT with short public exponent Cryptology", ePrint Archive, Report 2005/053, 2005 [Online]. Available: http://eprint.iacr.org/2005/053

[17] H.-M. Sun and C.-T.Yang, "RSA with balanced short exponents and its application to entity authentication," in *Public Key Cryptology—PKC 2005, Lecture Notes in Computer Science.* NewYork: Springer, 2005,vol. 3386, pp. 199–215.

[18] H.-M. Sun, W.-C. Yang, and C.-S. Laih, "On the design of RSA with short secret exponent," in *Advances in Cryptology—ASIACRYPT'99*,ser. Lecture Notes in Computer Science, K.-Y. Lam, E. Okamoto, andC. Xing, Eds. Berlin: Springer, 1999, vol. 1716, pp. 150–164.

[19]] S. A. Vanstone and R. J. Zuccherato, "Short RSA keys and their generation," *J. Cryptol.*, vol. 8, no. 2, pp. 101–114, Mar. 1995.

[20] Cryptography and Network Security: Principles and Practice by William Stallings .

[21] I. Niven and H. S. Zuckerman*, An Introduction to the Theory of Numbers.*New York: Wiley, 1991.

[22] H. W. Lenstra, "Factoring integers with elliptic curves," *Ann. Math.*,vol. 126, pp. 649–673, 1987.

[23] R. Rivest, A. Shamir, and L. Aldeman, "A method for obtaining digital signatures and public-key cryptosystems,"*Commun. ACM*, vol. 21, no.2, pp. 120–126, 1978.

[24] R. Rivest and R. Silverman, Are "Strong Prime" Needed for RSA?Cryptology ePrint Archive Report 2001/007, 2001 [Online]. Available:http://eprint.iacr.org/2001/007

[25] R. D. Silverman, "Fast generation of random, strong RSA primes,"*CryptoBytes*, vol. 3, no. 1, pp. 9–13, 1997.

[26] T. Takagi, "Fast RSA-type cryptosystem modulo p q," in *Advances in Cryptology-CRYPTO'98*, ser. Lecture Notes in Computer Science. New York: Springer, 1998, vol. 1462, pp. 318–326.

[27] Chinese Remaindering Based Cryptosystems in the Presence of Faults by Marc Joye, Arjen K.Lenstra, Jean-Jacques.

[28] A New CRT RSA Algorithm Secure Against Bellcore Attacks by Johannes Blomer , Martin Otto , Jean-Pierre Seifert

[29] Dhananjay Pugila, Harsh Chitrala, Salpesh Lunawat, P.M.Durai Raj Vincent,"An Efficient Envryption Algorithm Based On Public Key Cryptography", Dhananjay Pugila et.al / International Journal of Engineering and Technology (IJET)

[30] Vanstone, S.A., Zuccherato, R.J., "Short RSA keys and their generation". Journal of Cryptology8 (1995) 101-114