RSA Public Key Cryptosystem using Modular Multiplication

G.A.V.Ramachandra Rao
Dept. of computer science & Technology
Sri Chaitanya Engineering College Visakhapatnam, India

P.V. Lakshmi
Dept. of Information Technology
GIT, GITAM University, Visakhapatnam, India

N. Ravi Shankar
Dept. of Applied Mathematics
GIS, GITAM University, Visakhapatnam, India

ABSTRACT
In the rapid development of contemporary information technology, security has become important technique in many applications including Virtual Private Network (VPN), electronic commerce ,secure internet access etc. The security of public key encryption such as RSA scheme relied on the integer factorization problem. The security of RSA algorithm is based on a positive integer n, because each transmitting node generates pair of keys such as public and private. Encryption and decryption of any message depends on positive integer n. Where, the positive integer n is the product of two prime numbers and pair of key generation is depend on these prime numbers. In the paper [11], an algorithm for modular multiplication for public key cryptosystem is presented. This method is based on the following two ideas: (i) The remainder in regard to n can be constructed from the remainder with modulus (2n+1) and the remainder with modulus (2n+2). (ii) It often happens that 2n+1can easily be factorized, even if n is a prime number or n is difficult to be factorized into prime factors. The changed modulus value will be stated, which might be the one of the modulus factor i.e., (2n+1). Even if the hacker factorizes this new modulus value, they can't be searched out the original decryption key (d). Incapability to find the original decryption key, the factorization is insignificant. This proposed method helps to overcome the weakness of factorization found in RSA.

General Terms
Remainder multiplication, RSA Cryptosystem.

Keywords
Pubic key cryptosystem; modular multiplication; RSA Cryptosystem; modulus factor

1. INTRODUCTION
With the rapid development and the increasing popularity of the Internet technology, e-commerce application has also developed rapidly. The main issue related to reliable communication is security. The confidential communication in which online banking, transactions through ATM, business transactions using credit card and debit card etc. need security, because majority of transaction is held through IP cloud. An internet protocol cloud (IP Cloud) is an internet protocol network used to carry data traffic. In IP cloud the data is open and there is chance of being hacked. The essential requirement in security is to hide information from hackers. To secure information different kinds of crypto graphic techniques are used like message authentication code, symmetric and asymmetric cryptography, digital signature, hash function etc. [5]. There are number of methods introduced to secure the data communication in IP cloud or over the Internet. The hackers are so well-groomed that they can approach the information. To secure the information over an unsafe channel first public key cryptography was presented by Diffie and Hellman [4]. They introduced a protocol for exchanging information over an unsecure channel. After this different public key cryptosystems were introduced but the most known and suitable for both encryption and signing is RSA algorithm. This algorithm is planned to be the first great development in public key cryptography systems. However, RSA algorithm is secured only if long key strategy is kept [6, 10]. Rao et al.[11] proposed a new modular multiplication method to reduce the computational time of RSA cryptosystem. Rao et al.[12] proposed a novel Modular Multiplication Algorithm and its Application to RSA decryption.

This paper proposes an effective method to choose the modulus value to protect it from factorization risks of original RSA algorithm. The reason behind this circumstances is that the modified modulus factor (M_f) i.e., (2n+1) is false and value of encryption key (***e***) and decryption key (***d***) depend upon actual modulus (***n***). It also increase limit of plain text more than actual modulus value up to changed modulus factor " M_f " . Experimental results verify that the proposed scheme improves the security of RSA algorithm.

The rest of this paper is organized in six sections. Section 2 presents remainder multiplication algorithm. Section 3 presents computational complexity of each method, Section 4 briefly discusses RSA methodology and its limitations. Section 5 presents our proposed scheme to improve the security in RSA algorithm. Section 6 gives the comparison between RSA and proposed scheme. The conclusion and future work are presented in Section 7.

2. REMAINDER MULTIPLICATION ALGORITHM
In this section, the Remainder Multiplication method [11] is explained with an example. Let

$$y = xu \bmod n \quad (1)$$

Then y can be expressed as follows:

$y_1 \geq y_2 \quad y \equiv 2y_1 - y_2 \pmod n$ (2)

$y_1 < y_2 \quad y \equiv 2y_1 - y_2 + 2 \pmod n$. (3)

The positive integers (2 n + 1) and(2 n + 2) are decomposed into products of mutually prime factors. Let

$$2n+1 = \prod_{i=1}^{m} p_i \quad (4)$$

$$2n+2=\prod_{i=1}^{m_1} q_i \quad (5)$$

Assuming that moduli (2n + 1) and (2 n + 2) are decomposed as above, the next algorithm receives x such that $0 \le x \le (2n-1)^2$, and outputs y = xu mod $(p_1,...,p_{m_1})$

Algorithm mulmod(x, p, y)

Input : x, u, $0 \le x \le (2n-1)^2$, $0 \le u \le (2n-1)^2$,p = $(p_1 \cdots p_{m_1})$

Output: y = xu mod $(p_1 \ldots p_{m_1})$

Step 1 : Calculate $x_i = x \bmod p_i$, $u_i = u \bmod p_i$, $i = 1, \ldots, m_1$.
Step 2 : Calculate $a_i = x_i u_i$, $i = 1, \ldots, m_1$.
Step 3 : Calculate $a_i = a_i \bmod p_i$, $i = 1, \ldots, m_1$.
Step 4: Calculate y by Chinese remainder theorem (a, p, y)

The algorithm mulmod is used. y_1 = xu mod (2n+ 1) and y_2 = xu mod (2n+ 2) are obtained by mulmod (x, p, y_1) and mulmod (x, q, y_2) , respectively.

Example 1:
Consider the RSA cryptography example in the paper [1].Let n = 1386 and $X = x^2$ .Then 2n+1 = 2773 = 47 × 59 and 2n+2 = 2774 = 38 × 73.When x = 920 , y_1 = X mod (2n+1) =846400 mod 2773 = 635 and y_2 = X mod (2n+2)= 846400 mod 2774 = 330. In this case, $y_1 \ge y_2$. Applying Eq.(2), y ≡ 2(635) - 330 + (mod 1386) ≡ 1270 - 330 mod (1386) ≡ 940 (mod 1386) ≡940 is obtained. This agrees with the result of direct calculation of 920^2 mod 1386 = 940.

Example 2: consider the same RSA cryptography as in example-1 Let n = 1386 as preliminary computations 2n+1 and 2n+2 are decomposed. i.e 2n+1 = 2773 = 47 × 59 and 2n + 2 = 2774 = 38 × 73. Let p_1 = 47 ,p_2 =59 ,q_1 = 38 , and q_2 = 73. As the exponentiation computation assume that x = 920 and $y = x^2$ mod n is to be calculated.
The computation procedure for mulmod (920, (47×59),y_1) is shown in the following.
Step 1 **:** $x_1 = 920 \bmod 47 = 27$
$x_2 = 920 \bmod 59 = 35$
Step 2 **:** $a_1 = 27^2 = 729$
$a_2 = 35^2 = 1225$
Step 3 **:** $a_1 = 729 \bmod 47 = 24$
$a_2 = 1225 \bmod 59 = 45$
Step 4 **:** Solving the following system of congruence equations y_1 = 635 (in Example 1) is obtained and $y_1 \equiv 24 \bmod 47$ and $y_1 \equiv 45 \bmod 59$. Similarly y_2 = 330 is obtained and $y_2 \equiv 26 \bmod 38$ and $y_2 \equiv 38 \bmod 73$ from mulmod (920,(38 ×73),y_2) from y_1 = 635 and y_2 = 330 hence $y_1 \ge y_2$ then apply the equation
$Y \equiv 2y_1 - y_2 + (\bmod n)$ implies Y ≡ 2 (635) - 330+ (mod 1386) . Therefore, Y ≡ 940 mod 1386 = 940 is obtained. It was already seen in Example 1.

3. EVALUATION OF COMPUTATIONAL COMPLEXITY

By the consideration of security, the bit length of modulus should be 8192 bits in order to make the operations secure. We will calculate the total number of operations for each method.

3.1 Traditional Method

The total number of decryption operation for traditional method can be represented as :
MOD_E (d, n) = 1.5 × *l*(d) [M(*l*(n)) + 2 Mod (*l*(n)) +1]
= 1.5 × 8192[M (8192) + 2 Mod (8192) +1]
= 4573200384 clock cycles.

3.2 Hwang et al Method

In this method the total number of operations is given by: 4 MOD_E (d/4, n/4) + 8MOD_E (d/8, n/8) + 4[A(2048) + 2M(2048) + Mod(2048)] + 4[A(1024) + 2M(1024) + Mod(1024)] + 2[A(4096) + M(4096) + Mod(4096)] + A(4096) + 2M(4096) + Mod(4096) = 618359917 clock cycles.

3.3 CRT Method

In this method, the bit length of two distinct primes is equal. So, the total number of decryption operation for this method can be expressed as:
2 MOD_E(d/2, n/2) + 3 A (4096) + 2M (4096) + Mod (4096) = 1480580885 clock cycles.

3.4 Rao et al. Method

In this method the total number of operations is given by
8 MOD_E (d/8, n/8) + 16 MOD_E (d/16, n/16) + 4[A (256) + 2M (256) + Mod (256)] + 4[A (128) + 2M (128) + Mod (128)] + 2[A (512) + M (512) + Mod (512)] + A(512) + 2M (512) + Mod(512) = 2715648 clock cycles.

The notations are used for the above computational complexity of operations are

(3.4.1) MOD_E (d, n) denotes the computational complexity of modular exponentiation (x^y modn).

(3.4.2) M (x), A(x) and Mod(x) denotes the computational complexity of multiplication, addition and modulus operations with the bit length of operand is x.

(3.4.3) *l* (x) denotes the bit length of x.

(3.4.4) SH denotes shift operator.

Table I. shows the number of CPU clock Cycles for realizing the RSA decryption with the following parameters for bit length of modulus is 8192 bits on four different methods.

The traditional decryption method takes 4573200384 clock cycles, while the decryption method based on the Hwang et al method takes 618359917 clock cycles, and the decryption method based on the CRT method takes 1480580885 clock cycles. Significantly, the Rao et al method takes only 2715648 clock cycles. The results are listed in Table I**.**

Table I The Execution Time for each Method

	Traditional method	Hwang et al. Method	CRT Method	Rao et al Method
Estimation	4.573200384E9	6.18372205E8	1.480580885E9	1.89812126E8
Simulation (8192 bits)	2.286600192E9	3.09390445E8	1.480581269E9	9.4911902E7

Table II from [12] shows the computational complexity of multiplication ,modulo and addition operations. The clock cycles for the above three operations using different bit lengths and the values are listed in the Table II.

Table II The clock cycles for M(x), Mod(x), and A(x)

Bit Length	MUL	MOD	ADD
8192	76171	147998	256
4096	24963	47759	128
2048	8107	15136	64
1024	2595	4657	32
512	811	1362	16
256	243	363	8
128	67	80	4
64	15	11	2
32	1	1	1

Figure -1 depicts the performance and time complexity of the above methods .

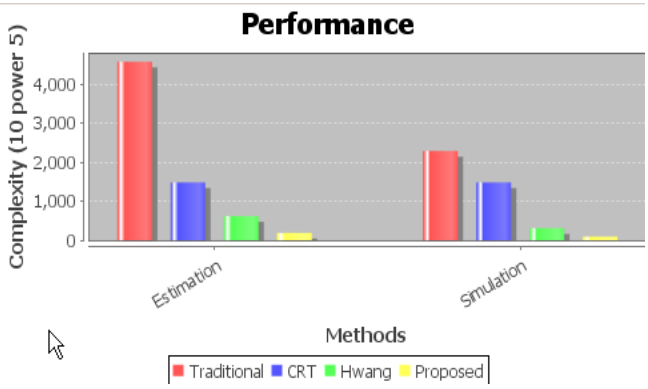

Fig. 1 shows the Time Complexity of each method.

4. RSA METHODOLOGY AND ITS LIMITATIONS

RSA is a commonly adopted public key cryptosystem which was proposed by Rivest, Shamir and Adleman in 1977 at Massachusetts Institute of Technology (MIT),[1]. The security of RSA rests on the exertion to factorize the big numbers of modulus. The size of modulus value is 1024 bits while the recommended length is 2048 bits as 640 bits key is not secure [7]. RSA uses two pairs of related keys (public key) ku = {e,n }for encryption and (private key) kr ={d, p,q} for decryption. Let's consider how these keys are generated and RSA works.

RSA cryptosystem:

Step 1: p and q are two relatively prime and large random numbers.

Step 2: A positive integer n is defined as a product of p and q.

Step 3: Eulers value of φ (n) = (p −1) (q −1).

Step 4: Choose e such that 1 < e <φ (n) and gcd (e,φ (n)) = 1

Here, d is computed by d = e-1 mod φ (n)

Step 5: In RSA e and n are public keys and d and (p, q) are

private keys so the plaintext M is encrypted by:1 < M < n and

C = Me mod n.

Step 6: And the cipher text C is decrypted by M = Cd mod n

In RSA, by choose a suitable value for "e" such that 3 and 17 will improve the speed of encryption procedure, but will cause more computational complexity on decryption side [8]. Resultantly, this situates on mathematically unstable ground like other cryptographic techniques . Different restrictions could be observed and several successful attacks are developed to break this algorithm [3].The major topic is related to its factorizing. If the process factorization should be done then whole algorithm is broken. Also, mathematically attacks can get the information about RSA keys by other resources [2]. Therefore its security relax on unconfirmed suppositions. RSA has no more guarantees that the secrets it guards will remain secure. Its center has decrepit every bother, the best minds of cryptography has work out [2].

5. PROPOSED METHOD (PMD)

The proposed scheme suggests a modification in RSA algorithm to resolve the subject of RSA factorization. This scheme solves this problem by changing actual modulus value "n" into false modulus "M_f" illustrates the procedure of generating fake modulus "M_f" and various steps of proposed approach are enumerated as under:

Step l: In first step choose two prime numbers p and q. In the third step selects the value of "e" and select "d" such that [e.d mod φ(n)]=1

Step 2: The second step toward this approach is to compute (possible values of e) such that (p-1) and (q-1) are relatively prime. gcd[e,φ(n)]=1,where 1<e < φ(p,q).

Step 3:In the third step selects the value of "e" and select "d" such that [e. d mod φ(n)]=1

Step 4: In this step, finds the "Ge" Where "Ge" is the modulus factor which are multiplied by "2n+1" and produce false modulus " Mf ". The new modulus " Mf " is used in place of actual modulus "n" and do the process of encryption and decryption.
Step 5: In this step approach calculates "Mf" that is the product of modulus factor "n" and "Ge" as
$Mf = Ge = (2n+1)$ (6)
Where "Mf" = false modulus factor value, n = product of two prime numbers p and q also "Ge" = modulus factor of n.
Step 6: Public key ku = {e, Mf} and private key = {d, Mf}

EXAMPLE 1: Here, we present an example-1 for the proposed modified method (PMD) in RSA encryption and decryption. We have used artificially small parameters to clarify the concept. However, the method is applicable in general to all suitable selected parameters.

1) Select two prime numbers p and q such that p = 71 and q = 37
2) Calculate $n = p*q = 71 * 37 = 2627$ (7)
3) Calculate $\varphi(n) = (p-1)*(q-1) = 70 * 36 = 2520$ (8)
Next we calculate "e" such that $\gcd[e,\varphi(n)] = 1$, where $1 < e < \varphi(p,q)$ and e is the co-prime to $\varphi(n)$
4) If e = 29 then d = 869
5) Then select G_e = (2n+1) = 5255 (already explained in modular multiplication algorithm one of the factor) = 5255 = 5*1051 = product of prime factors
6) For different value of e and G_e is different then $M_f = G_e$ = (2n+1)= M_f = (2n+1) = 5255
7) Public key ku = {29, 5255} and private key = {869,5255}
8) Now plain text = M where $o \le M \le 2n-1$ and ciher text = $c = M^e \bmod M_f$ and the Decipher as $M = C^d \bmod M_f$

The encryption and decryption process are done by using the modulus factor and the results are listed in the Table III.

Table III: Proposed method(PMT) Encryption and Decryption process

Plaintext (M)	Encryption	Decryption
36	$36^{29} \bmod 5255 = 2326$	$2326^{869} \bmod 5255 = 36$
42	$42^{29} \bmod 5255 = 2267$	$2267^{869} \bmod 5255 = 42$
920	$920^{29} \bmod 5255 = 2030$	$2030^{869} \bmod 5255 = 920$
2630	$2630^{29} \bmod 5255 = 2870$	$2870^{869} \bmod 5255 = 2630$
5253	$5253^{29} \bmod 5255 = 908$	$908^{869} \bmod 5255 = 5253$

EXAMPLE -2: We have used artificially small parameters to clarify the concept. Select two prime numbers p and q such that p = 7 and q = 11 and n = 77 , $\Phi(n) = 60$. If e = 13 then d = 37. Then select G_e = (2n+1) = 155 (already explained in modular multiplication algorithm one of the factor) = 155 = 5*31 = product of prime factors.

For different value of e and G_e is different then $M_f = G_e$ = (2n+1) = 155 . Now Public key ku = {13, 155} and private key = {37,155} and the plain text = M where $o \le M \le 2n-1$ and ciher text = $c = M^e \bmod M_f$ and the Decipher as $M = C^d \bmod M_f$

The encryption and decryption process are done by using the modulus factor and the results are listed in the Table IV.

Table IV : Proposed method (PMT) Encryption and Decryption process

Plaintext (M)	Encryption	Decryption
15	$15^{13} \bmod 155 = 120$	$120^{37} \bmod 155 = 15$
36	$36^{13} \bmod 155 = 36$	$36^{37} \bmod 155 = 36$
99	$99^{13} \bmod 155 = 99$	$99^{37} \bmod 155 = 99$
135	$135^{13} \bmod 155 = 145$	$145^{37} \bmod 155 = 135$
146	$146^{13} \bmod 155 = 106$	$106^{37} \bmod 155 = 146$

6. PERFORMANCE COMPARISON BETWEEN PROPOSED METHOD (PMT) & RSA

In this two main problems have been discussed related to RSA and it has been shown that these can be overcome by using the proposed method.
Problem of factorization**:** If anyone is able to factorize the modules(n) in RSA, can easily decrypt the message.However, in our case the result produced by "G_e" will produce false modulus "M_f". This is publically announced. If anyone tries to factorize the "M_f", he can't find out the original value of decryption key, because it is a false value, and result produced by its will not give accurate result.
Figure-2 and Figure 3 represent the comparison results between RSA and the proposed method (PMT) ,to find the inverse with the help of public key pair. In RSA use KU ={ e,n} and in PMT use KU = { e,M_f }.

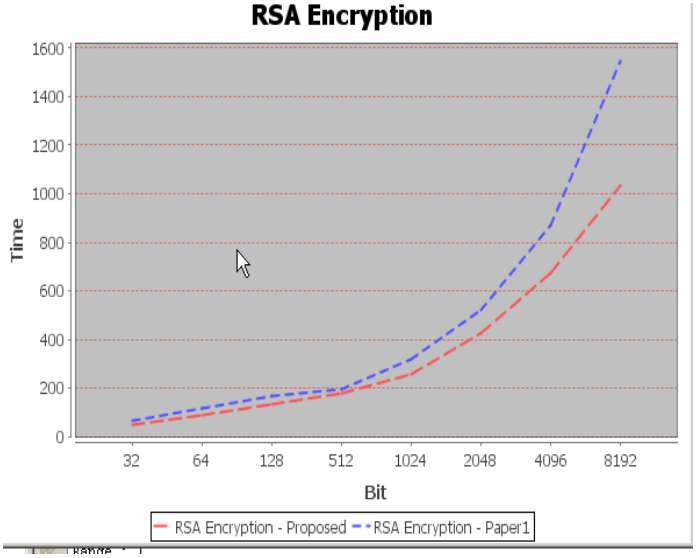

Fig.2 The performance of RSA encryption and the proposed method (PMT)

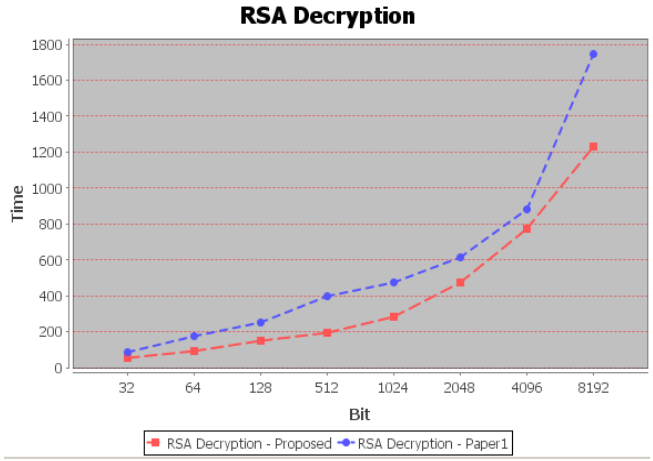

Fig .3: The performance of RSA Decryption and the proposed method (PMT).

Elimination of plain text: According to RSA, it will not be larger than the modulus (n) value is $0 \leq M \leq 2n-1$. Also, in the proposed method no extra calculation is needed. Once the "G_e" is found, the remaining algorithm works like RSA. Extra burden with much more computation complexity is needed by hacker to break up the proposed method and the table -5 shows the comparison between RSA and PMT.

Table V: Comparison between RSA and proposed method (PMT)

RSA	PMT
Publicly announced original modulus	Publicly announced false modulus
e and d computed from publicly announced modulus	e and d computed from publicly secret modulus
Limit of plain text is $0 \leq m < n$	Limit of plain text is $0 \leq m < M_f$
Only two possible factors are required to factorize modulus p*q or q*p	Only two factors are required to factorize modulus p*q or q*p
The strength of variables are only two p and q	The strength of variables are only three p and q and G_e

7. CONCLUSION

In this paper, the RSA algorithm is examined to reduce its boundaries. The proposed method gets better the security risk and limitation on sending plaintext. The major disadvantage of RSA is factorization; if the hackers factorize the modulus (n), then whole RSA security device will be opened from this key of factorization. In the proposed method, we publically declare the value of "M_f" which is totally false. This method makes it more expansive for hackers to find the unique value of modulus (n). And the prospect work is related to continuation of "G_e". It may be discover that how to define exactly these special numbers and what may be the principle to select them most correctly.

8. ACKNOWLEDGEMENTS

The authors would like to thank the referees for providing very helpful comments and suggestions.

9. REFERENCES

[1] R.Rivest, A.Shamir, and L.Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," vol. 21 (2), pp.120-126, 1978. .

[2] S.Robinson,"Still Guarding Secrets after Years of Attacks, RSA Earns Accolades for its Founders," SIAM News, Vol. 36, Number 5, June 2003.

[3] M.Frunza,and L.Scripcariu ,"Improved RSA Encryption Algorithm for increased Security of Wireless Networks" Signals, Circuits and Systems,2007,ISSCS 2007, International Symposiumon. Vol.2. IEEE,2007

[4] D.Lerch Hostalot, "Factorization attack on RSA," hakin9 3/2007. www.en.hackin9.org

[5] Al Hasib, Abdullah, and Abul Ahsan Md Mahmudul Haque. "A comparative study of the performance and security issues of AES and RSA cryptography." Convergence and Hybrid Information Technology, 2008. ICCIT'08. Third International Conference on. Vol. 2. IEEE, 2008.

[6] J. M. Jordan, and P. J. Flinn, "Using the RSA Algorithm for Encryption and Digital Signatures," July 9, 1997.

[7] Fahn, R., and M. J. B. Robshaw. Results from the RSA Factoring Challenge. Technical Report TR-501, Version 1.3, RSA Laboratories, 1995.

[8] Hwang, R. J., Su, F. F., Yeh, Y. S., & Chen, C. Y. (2005, March). An efficient decryption method for RSA cryptosystem. In Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on (Vol. 1, pp. 585-590). IEEE.

[9] B.Schneier "Applied cryptography protocol, algorithms, and source code in C" 2ndedition, John Wiley&sons,Inc,1996.

[10] Koblitz, Neal. "Elliptic curve cryptography." Mathematics of Computation 48,177 (1987).

[11] Rao, GAV Rama Chandra, P. V. Lakshmi, and N. Ravi Shankar. "A New Modular Multiplication Method in Public Key Cryptosystem." International Journal of Network Security 15.1 (2013): 23-27.

[12] Rao, GAV Rama Chandra, P. V. Lakshmi, and N. Ravi Shankar, "A Novel Modular Multiplication Algorithm and its Application to RSA Decryption", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 6, No 3, November 2012.