Improved Semantically Secured Variant of RSA Public Key Cryptosystem

Sushma Pradhan, Birendra Kumar Sharma
School of Studies in Mathematics, Pt. Ravi Shankar Shukla University, Raipur, Chhattisgarh, India

Article Info

Article history:

Received Aug 12, 2015
Revised Nov 20, 2015
Accepted Nov 26, 2015

Keyword:

Batch RSA
DRSA
Encryption
M-prime RSA
QC-RSA
RSA cryptosystem
RSA-CRT
Semantic security

ABSTRACT

Boneh and Shacham gave a nice survey on four variants (Batch RSA, Multi-Prime RSA, Multi-Power RSA, and Rebalanced RSA). Batch RSA and Multi-Prime RSA were then combined to increase the decryption/signature generation performance. Here in this paper we further tried to increase the encryption/ signature verification performance. The proposed scheme is semantically secure also.

Corresponding Author:

Sushma Pradhan,
School of Studies in Mathematics,
Pt. Ravi Shankar Shukla University,
Raipur, Chhattisgarh, India.
Email: sushpradhan@gmail.com

1. INTRODUCTION

The RSA algorithm [8] was publicly described in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman at MIT. In cryptography, RSA is an algorithm for publickey cryptography. It is the first algorithm known to be suitable for signing as well as encryption, and one of the first great advances in public key cryptography. RSA is widely used in electronic commerce protocols, and is believed to be secure given sufficiently long keys and the use of up-to-date implementations. Public-key cryptography utilizes an asymmetric cryptography technique with two keys, one public and one private. The keys are derived from the multiple of two large prime numbers. The private key can only be deduced from the public key by factoring the large multiple. RSA security comes from the difficulty in factoring very large numbers. Techniques for factoring numbers are improving, but the speed of all depend on the size of the number, which means they still take significant time. While the possibility exists that one day there will be an extraordinary leap in our ability to factor large numbers, it is unlikely and offers a minimal threat to RSA.

The RSA cryptosystem due to Rivest, Shamir and Adleman [8] is one of the most popular public key cryptosystem and widely used to ensure privacy and authenticity of elctronic data. Several variants have been developed to enhance the property of RSA cryptosystem. In 2002, Boneh and Shacham [2] gave a very nice comparison of the variants of RSA (Batch RSA [1], MPrime RSA [4], MPower RSA [10], and Rebalanced RSA [11]). All these variants are improving the decryption/ signature verification performance. Their work was further extended by Caesar [3] in 2003. He combined the two variants of RSA, MPrime RSA and Rebalanced RSA and gave the performance by improving the decryption/ signature generation speed by 27 times to RSA and by 4.8 times to RSA with CRT for 2048 bit moduli. It is further improved by the combined approach of Batch RSA and Mprime RSA (i.e. BM-Prime RSA [9]). Here in this paper we extending the work of BM-Prime RSA, we are trying to increase the encryption speed with less compromising the decryption speed of BM-Prime RSA.

The scheme is providing better performance; besides this it is providing semantic security, whereas BM-Prime RSA is not having semantic security. This is achieved with the help of DRSA [5] cryptosystem which is having the property of semantic security. The new scheme is also giving better performance as compared to DRSA cryptosystem.

In second section of the paper RSA cryptosystem and its variants are explained. In third section DRSA cryptosystem is reviewed. Then the proposed scheme is given to improve the encryption performance of BM-Prime RSA and the security analysis is given with the comparison of proposed scheme with BM-Prime RSA and DRSA. The paper is then concluded in the last section.

2. WORKING OF RSA

In RSA [8] two keys are required, first is e and N (n bits) public and second is a number d that is kept secret. In order for A to send a message to B, A looks up B's public values and, if the message is M (written as a number), then A divides the message into pieces of size less than N and sends $C = M^e \bmod n$. Then B decodes by $M = C^d \bmod n$. The security of the system lies in the choices of the public and private keys.

a) Key Generation of RSA
   Step1. Two large random primes (p and q) of n/2 bits are generated such that their product N = pq is
   Step2. N = pq and φ (N) = (p − 1)(q − 1).
   Step3. Select an integer e (1 < e < φ (N)), such that gcd (e, φ (N)) = 1
   Step4. Now Compute d, 1 < d < φ (N), such that e d =1(mod φ (N)).
   Public key= (N, e) and Private Key= (N, d).

b) Encryption of RSA
   C=cipher text, M= plaintext
   Step1. Represent the plaintext message as a positive integer M.
   Step2. $C = M^e \bmod n$.

c) Decryption of RSA
   Step1. $M = C^d \bmod n$.
   Step2. Extract the plaintext from the corresponding integer message M.

In 1982, Quisquater and Conver [7] introduced a fast deciphering algorithm to speed up the decryption process via Chinese Remainder Theorem, which was called QCRSA. It improves the standard RSA by the factor of 4. In 2002 Boneh and Shacham [2] gave the comparison of RSA variants (Batch RSA [1], MPrime RSA [4], MPower RSA [10] and Rebalanced RSA [11]) to enhance the performance of RSA cryptosystem. Here these variants are reviewed. RPrime [3] given by Caeser in 2002 is also given here. There comparisons are explained in [6]. Then the combined approach of Batch RSA and MPrimeRSA (BM-Prime RSA [9]) is described.

2.1. Batch RSA

Batch RSA variant [1] was introduced in 1989. Fiat [1] has shown that, using small public exponents e_1 and e_2 for the same modulus N, it is possible to decrypt two ciphertexts for approximately the price of one. Fait generalized the above observation to the decryption of a batch of b RSA ciphertext. We have b pairwise relatively prime public keys $e_1, e_2 .... e_b$, all sharing a common modulus N. we have b encrypted messages $C_1, C_2 ... C_b$, where Ci is encrypted using the exponent ei. We wish to compute $M_i = C_i^{1/e_i}$ for $i = 1,..b$. Fiat described this b-batch by processing a binary tree for small values of b $(b \le 8)$. One sets $e = \prod_i e_i$ and $A_0 = \prod_i C^{e/e_i}$ (where the indices range over $i = 1,..b$). Then one calculates $A = A_0^{1/e} = \prod_{i=1}^{b} C^{1/e_i}$. For each I one computes Mi as:

$$M_i = C_i^{1/e_i} = \frac{A^{\alpha_i}}{C_i^{\alpha_i - 1/e_i} . \prod_{j \ne i} C_j^{\alpha_i / e_i}}$$

The Batch decryption is extended by QCRSA [8]. Here b-batch requires b modular inversions whereas Fiats tree-based method requires 2 b modular inversions, but fewer auxiliary multiplications. Note that since b and the ei's are small, the exponents in above equation are also small.

Batch RSA decrypts simultaneously b messages with the approximate cost of a single exponentiation (of order of N) and some small exponentiations (using public exponents). According to Fiat with standard 1024-bit keys, batching improves performance significantly. With b = 4, RSA decryption is enhanced by a factor of 2.6, with b = 8, by a factor of almost 3.5.

2.2. Multi-Prime RSA

Multi-Prime RSA [10] was introduced in 1998. The RSA modulus was modified so that it consists of k primes p1, p2; ....; pk instead of using only two. The algorithms of Key generation, Encryption and Decryption are described as:

a) Key generation of Multi-Prime RSA

The key pairs (public and private) are generated according to the following steps (here k is the number of primes be used in the variant):

Step1. Compute k distinct primes $p_1, p_2,...p_k$ each (n/k) bits in length such that $N = \prod_{i=1}^{k} p_i$

Step2. Compute e and d such that $d = e^{-1} \bmod \phi(N)$, where gcd(e, φ(N)) = 1 and φ(N) = $\prod_{i=1}^{k} (p_i - 1)$

Step3. For 1 ≤ i ≤ k, compute di = d mod (pi − 1).

Public key = < N, e >,
Private Key = < N, d1, d2...dk >

b) Encryption of Multi-Prime RSA

Encryption is same as in the original RSA, thus $C = M^e \bmod N$.

c) Decryption of Multi-Prime RSA

The decryption is an extension of the Quisquater- Couvreur method. To decrypt a ciphertext C,

Step1. Calculate $M_i = C^{d_i} \bmod p_i$ for each i, 1 ≤ i ≤ k.

Step2. Apply the CRT to the Mi.'s is to get $M = C^d (\bmod N)$.

d) Performance of Multi-Prime RSA

The theoretical speedup [6] of this variant to the CRT RSA is given as follows:

SCRT= $(2.(n/2)^3)/(k.(n/k)^3) = k^2/4$.

2.3. BM-Prime RSA (Combined Approach of Batch RSA and Multi-Prime RSA)

In this approach [9] Batch RSA [1] and Multi Prime RSA [4] are combined to improve the decryption performance. The general idea of this scheme is to use the key generation algorithm of Batch RSA with only two primes of n/k bits length and decryption algorithm of Multi-Prime RSA. The three algorithms for the new scheme are as follows:

a) Key Generation

Let N be the RSA modulus n and k be the batch size.

Step1. Compute b distinct primes $p_1, p_2,...p_k$, each one (n/k) bits in length and

$$N = \prod_{i_1}^{k} p_i$$

.

Step2. Compute e and d such that $d = e_i \bmod(N)$, where $\gcd(e, \phi(N)) = 1$.

$$\Phi(N) = \prod_{i=1}^{k} (p_i - 1).$$

Step3. For $1 \le i \le k$, compute $d_i = d \bmod (p_i - 1)$

Public Key= $< n, e_{1,} e_2 .... e_k >$; Private Key= $< n, d_{1,} d_2 .... d_k >$.

b) Encryption

We have b encrypted messages $C_1, C_2, ..., C_k$ where ci is encrypted using the exponent ei, i.e,

$$C_1 = M_1^{1/e_1} \bmod N$$
$$C_2 = M_2^{1/e_2} \bmod N$$
$$.$$
$$.$$
$$.$$
$$C_i = M_i^{1/e_i} \bmod N \qquad 1 \le i \le k$$

c) Decryption

To decrypt a cipher text c, First calculate $M_i = C_i^{1/e_i} \bmod p_i$ for each I ,$1 \le i \le k$. Next, apply the CRT to the Mi.'s is to get $M = C^{1/e_i} \bmod N$. The CRT step takes negligible time compared to the k-exponentiation.

d) Performance of BM-Prime RSA

We compare the decryption work using the above scheme to the work done when decrypting a normal RSA cipher text. Recall that RSA decryption using CRT requires two full exponentiations modulo n/2 bits numbers. In our BM-Prime decryption requires k full exponentiation modulo n/k bits numbers. BM-Prime RSA, which for 2048-bits moduli got a gain of 30 percent with relation to Rebalanced RSA and is there-fore about 27 times faster than original RSA.

2. DRSA CRYPTOSYSTEM

The standard RSA and other variants are not semantically secure. Pointchevel [5] gave a new Dependent RSA (DRSA) problem and proposed a cryptosystem with the property of semantic security. The three algorithms of key generation, encryption and decryption are of DRSA cryptosystem are explained below.

a) Key Generation

The key generation for DRSA [5] scheme is same as that for the standard RSA scheme. To generate keys in DRSA scheme, the user chooses two large primes p and q and computes N = pq. User then determines an integer e less than and relatively prime to φ(N) and computes an integer d such that ed = φ(N). The public key and the secret key for the user R is (e, N) and d respectively. The prime p and q are also kept secret.

b) Encryption

To encrypt any plaintext $M \in Z_N$, sender S first randomly selects an integer $l \in Z_N^*$ and sends the complete cipher text (C_1, C_2) to the receiver R. Where,

$$C_1 = l^e \bmod N$$
$$C_2 = M(l+1)^e \bmod N.$$

c) Decryption

To decrypt the cipher text (C_1, C_2), receiver R first computes

Step1. $C_1^d \bmod N = l$.

Step2. $M = (C_2 / (l+1)^e) \bmod N$.

3. NEW SCHEME

3.1. Algorithm

Here BM-prime RSA [9] is further extended to increase the encryption/signature verification performance with a very less compromising the decryption. In this approach the DRSA [5] cryptosystem is used with BM-Prime RSA. The Key generation, Encryption and the Decryption process are as follows.

a) Key Generation

Let N be the RSA modulus n and k be the batch size.

Step1. Compute k distinct primes $p_1, p_2, ...p_k$ each (n/k) bits in length such that $N = \prod_{i=1}^{k} p_i$

Step2. Compute e and d such that $d = e^{-1} \bmod \phi(N)$, where gcd (e, φ(N)) = 1 and φ(N) = $\prod_{i=1}^{k} (p_i - 1)$

Step3. For 1 ≤ i ≤ k, compute di = d mod (pi − 1).

Public key = < N, e >,
Private Key = < N, d1, d2...dk >

b) Encryption

To encrypt any message $M \in Z_N$, sender S chooses a random integer $h \in Z*_N$ and computes

1. $C_1 = (h+1)^e \bmod N$
2. $C_2 = Mh^{-1} \bmod N$

Cipher text (C_1, C_2) is sent to the receiver.

c) Decryption

The steps in this algorithm are as follows:

Step1. Compute $h_{1=}C_1^{d_p} \bmod p$ and $h_{2=}C_1^{dq} \bmod q$.

Step2. Thus $h_1^e = C_1 \bmod p$ and $h_2^e = C_1 \bmod q$.

Step3. Compute h1, such that $(h_1)^e = C_1 \bmod p_k - 1$

Step4. Using CRT, compute h such that $h = h_1 \bmod p_k - 1$ and $h = h_2 \bmod q$. Then $h = C_1^d \bmod N$.

Step5. From h we can now finally calculate $M = C_2^h \bmod N$.

4.2. Semantic Security

Semantic security is a widely-used definition for security in an asymmetric key encryption algorithm. For a cryptosystem to be semantically secure, it must be infeasible for a computationally-bounded adversary to derive significant information about a message (plaintext) when given only its cipher text and the corresponding public encryption key. The proposed cryptosystem is semantically secure against chosen plaintext attack. This we can say on the basis that in order to determine any information about the plaintext M from the cipher text (C_1, C_2), the adversary needs to have information about $h^{-1}(\bmod N)$, where this h is a randomly chosen element in $Z*_N$. We cannot calculate the value of h without knowing the value of the secret key d. Even the value of $h^{-1}(\bmod N)$ can't be calculated even if we know partial information about h. Thus the given new scheme is semantically secure.

4.3. Computational efficiency

In the proposed scheme, the random values h can be taken in advance, thus the user has the values $(h+1)^e$ and h^{-1} computed well in advance. Thus time consumed during the encryption phase in the new scheme will be lowered. Because the value of e is very large, so it saves a large amount of time during encryption. In this case, the encryption process requires only one multiplication modulo N and that is quite affordable. Whereas in BM-Prime RSA [9] scheme, having large value of the encryption exponent e, the encryption cost is very high. As compared to only one multiplication modulo N in new scheme, BM- Prime RSA requires one exponentiation to the power e modulo N, resulting in poor performance of encryption side. Thus it can be concluded that the new scheme encryption phase has better performance than BM-Prime RSA and of course than any other variants of RSA. This encryption performance enhancement is at the cost of slight increase in decryption cost. Besides other computation the proposed scheme requires to compute one extra multiplication modulo N at the decryption process. Thus the decryption speed is computationally as expensive as BM-Prime RSA.

In BM-Prime RSA ne ≈ n, so it cost very high in Encryption process. But this factor is not involved in the new scheme, that's why the encryption is very low as compared to BM-Prime approach. This result is a small increase in cost of decryption side approach.

Semantic Security: The proposed scheme is semantically secure due to randomness added in the comutation, but BM-Prime RSA is not semantically secure.

4.4. Comparison with DRSA Scheme

4.4.1. Decryption Phase

Our proposed scheme is computationally less expensive than DRSA cryptosystem. In our proposed scheme, On average it is required to compute less than one exponent to the power d modulo N and one multiplication modulo N. Whereas in DRSA, decryption process requires computing one exponentiation to the power e and to the power d modulo N, one inversion and one multiplication under modulo N. Hence our proposed scheme is more efficient than that of the DRSA scheme.

4.4.2. Encryption Phase

The efficiency of encryption process of DRSA and our proposed scheme both is same whereas decryption cost is lower in new scheme as compared to DRSA.

4. CONCLUSION

In this paper RSA public cryptosystem is explained with its combined variants of Batch RSA and Multi-Power RSA, i.e., BM-Prime RSA. BM-Prime RSA gave the maximum decryption/signature generation performance of all the variants of RSA (Batch RSA, Multi-prime RSA, Multi-Power RSA, Rebalanced RSA, and RPrime RSA). But this performance is at the cost of very high encryption/ signature verification cost. The devices with constrained resources, like PDAs, are not always used for decryption/signature generation. There is the requirement that these constraint devices are used for encryption/ signature verification as well. In that case the use of BM-Prime RSA will not be giving good performance. DRSA is explained to the semantic security to the system. The proposed approach in the paper gives better encryption performance at the cost of a small decrease in decryption side. Besides it provides the semantic security to the system which is not provided by BM-Prime RSA. The scheme is proven to be better than BM-Prime RSA as well as to DRSA.

REFERENCES

[1] A. Fiat., “Batch RSA”, *Advances in Cryptology: Proceedings of Crypto’89*, pp. 435-175, 1989.

[2] D. Boneh and H. Shacham, “Fast variant of RSA”, *RSA laboratories*, 2002.

[3] Cesar Alison Monticro Paixao, “An Efficient Variant Of The RSA Cryptosystem”, *Cryptology ePrint Archive*, pp. 159, 2003.

[4] Collins T, Hopkin D, Langford S. and Sabin M., “Public Key Cryptographic Apparatus And Method”. US patent #5, 848,159, 1997.

[5] David Pointcheval, “New Public Key Cryptosystem Based On The Dependent RSA Problem”, *Eurocrypt99 LNCS*, Springer-Verlag, Vol. 1592, pp.239-254, 1999.

[6] A.A. Mamun, M. M. Islam, S.M. M. Romman, and A.H.S.U Ahmad, “Performance Evaluation of Several Efficient RSA Variants”, *IJCSNS*, Vol.8 No.7, pp. 7-11, July 2008.

[7] J.Quisquater, Couvruur, “Fast Decyhpering Algorithm for RSA Public Key Cryptosystem”, *Electronics Letters*, Vol. 01, pp. 905-907, 1982.

[8] R.L.Rivest, A. Shamir and L. Adlemann, “A Method for Obtaining Digital Signature and Public Key Cryptosystem”, *Communication of the ACM*, Vol. 2, pp.120-126, 1978.

[9] Sushma Pradhan, Birendra Kumar Sharma, “An Efficient RSA Cryptosystem with BM-PRIME Method”, *IJINS*, Vol.2, No.1, pp. 438-443, February 2013.

[10] T.Takagi, “Fast RSA-Type Cryptosystem Modulo pkq”, *Crypto98*, Vol-1462 of LNCS, pp. 318-326, 1998.

[11] M. Wiener, “Cryptanalysis of Short RSA Secret Exponents”, *IEEE Transactions on Information Theory*, Vol. 36, No. 3, pp. 553-558, 1990.

BIOGRAPHIES OF AUTHORS

Sushma Pradhan received the B.Sc, M.Sc and M.Phill degree in Mathematics Pt. Ravishankar Shukla University, Raipur, Chattigarh, India in 2002, 2004 and 2007. She joined School of Studies in Mathematics, Pt. Ravishnakra Shukla University, Raipur, India for her Research work. She is a life time member of Cryptology Research Society of India (CRSI). Her area of interest is Public Key Cryptography and Integer factorization Problem.

Birendra Kumar Sharma Professor, School of Studies in Mathematics, Pt.Ravishankar Shukla University Raipur (C. G.) India. He has been working for long time in the field of Non Linear Operator Theory and currently in Cryptography. He and his research scholars work on many branches of public key cryptography. He is a life member of Indian Mathematical Society and the Ramanujan Mathematical Society.