A Multisignature-like Scheme Based on RSA with CRT-Exponents

Hung-Min Sun*, Cheng-Ta Yang**, Mu-En Wu* and Vincent S. Tseng**

*Department of Computer Science
National Tsing Hua University
Email: hmsun@cs.nthu.edu.tw

**Department of Computer Science and Information Engineering
National Cheng Kung University
Email: zadayang@ismail.csie.ncku.edu.tw

***Abstract*— A digital multisignature is a normal digital signature of a message generated by multiple signers with knowledge of multiple private keys. In this paper, we point out that two CRT-equations can be totally independent in Rebalanced-RSA. This means that we can choose different public exponents when using Rebalanced-RSA. From this point of view, we proposed the new multisignature-like scheme based on RSA with CRT-Exponents.**

Keywords: RSA, Rebalanced-RSA, Multisignature, Public-key cryptography.

I. Introduction

An institution or organization usually makes up the committees for some special tasks. These committees need a chairman and some members. They often have a lot of paper output, e.g. : administrative decrees, policies, specifications, ... and so on. Some documents are very important and are approved by all members of committee. Conventionally, all group members sign the same message, we call this scheme multisignature.

The multisignature is a kind of group-oriented cryptography that was first introduced by Desmedt [2] in 1987. The property of group-oriented cryptosystem is that there are two or more participators assisting the others to achieve their work. The group has a security policy that requires a multisignature to be signed by all group members with the knowledge of multiple private keys. Any verifier can easily verify the accuracy of a given message based on the multisignature and all signers' public keys.

The practical implementation of RSA cryptosystem for multiple operations of a given message causes bit expansion problem inherently. There are many multisignature schemes proposed in the past based on RSA [4][6][8]. However, with these schemes, either the signers and signing order must be determined in advance, or the size of a multisignature grows proportional to the number of signers. Recently. Harn and Ren[5] point out an efficient multisignature scheme should possess two properties: fixed length and constant verification time.

In 1990, Wiener [15] proposed an RSA variant, which can further reduce the decryption cost again by choosing d such that both $d_p \equiv d(\mathrm{mod}\, p-1)$ and $d_q \equiv d(\mathrm{mod}\, q-1)$ are small. Since this RSA variant enables us to rebalance the costs of encryption and decryption, Boneh and Shacham [1] called it Rebalanced-RSA. In other words, we can speed up the CRT decryption by shifting the decryption cost to the encryption cost. However, one of the drawbacks of Rebalanced-RSA is the size of public exponent is almost as large as the RSA modulus, which causes the encryption inefficient. Consequently, Sun, Hinek, and Wu [14] modified the key-generation of Rebalanced-RSA to against this drawback. They proposed a method to construct CRT-equations in Rebalanced-RSA and called their method Generalized Rebalanced-RSA. Compared to the Rebalanced-RSA, Generalized Rebalanced-RSA reduces the size of public exponent while just raising the size of secret exponent slightly.

In this paper, we observe a property from the CRT-equations in Rebalanced-RSA. We point out that, in fact, two CRT-equations can be totally independent. This means, We can choose different public exponents when using Rebalance RSA. Furthermore, we propose a multisignature-like scheme based on Rebalanced-RSA. In our proposed scheme, the signers can sign the same message simultaneously and then combine all individual signatures into a multisignature. In addition, there exists an exponential operation in the multisignature verification procedure.

The rest of this paper is organized as follows: In the next section, we survey an RSA and its variant. The property of CRT-Equations in Rebalanced-RSA is presented in Section 3. Section 4 proposes a new multisignature-like scheme based on RSA with CRT-Exponents. There are some analyses in Section 5. Finally, some conclusions are made in Section 6.

II. Preliminary

In 1978, Rivest, Shamir and Adleman proposed new type of public-key cryptosystem, called "RSA cryptosystem", whose security is based on the difficulty of factoring a large integer [10]. In this section we introduce the basic RSA and its variants.

*a) **RSA Cryptosystem**:* Let N be the product of two large primes p and q. Let e and d be two integers satisfying $ed \equiv 1 \bmod \phi(N)$, where $\phi(N) = (p-1)(q-1)$ is the Euler totient function of N. In general, N is called the RSA modulus, e is the public exponent, and d is the secret exponent. To encrypt a plaintext message M, one computes the corresponding ciphertext $C \equiv M^e (\bmod N)$. To decrypt the ciphertext C, the legitimate receiver computes $M \equiv C^d (\bmod N)$.

*b) **CRT Decryption**:* Based on Chinese Remainder Theorem (CRT for short), Quisquater and Couvreur [9] proposed a fast decryption algorithm, called CRT decryption. Let $d_p \equiv d \mod (p-1)$ and $d_q \equiv d \bmod (q-1)$. The legitimate receiver decrypts by computing $C_p \equiv C^{d_p} \pmod p$ and $C_q \equiv C^{d_q} \pmod q$, and then using Chinese Remainder Theorem to combine C_p and C_q to recovery the plaintext M. The public exponent e satisfies $ed_p \equiv 1 \bmod (p-1)$ and $ed_q \equiv 1 \bmod (q-1)$, which are called CRT equations throughout this paper. Also, d_p and d_q are called CRT-exponents. In general, the CRT decryption is approximately 4 times faster than the decryption in standard RSA.

*c) **Rebalanced-RSA**:* Wiener [15] suggested an RSA variant, Rebalanced-RSA, to further speed up decryption by shifting the work to the encryption, but slow down encryption. This is why the variant called "rebalanced". The key generator first chooses two small CRT-exponents d_p and d_q, and then compute d satisfying $d_p \equiv d \bmod (p-1)$ and $d_q \equiv d \bmod (q-1)$ by using Chinese Remainder Theorem. Finally, it computes the public exponent e satisfying $ed \equiv 1 (\bmod (N))$. The features of Rebalanced-RSA is fast decryption but slow encryption. Such kind of RSA variant is especially suitable for the device executing decryption with small computational ability.

*d) **Generalized Rebalanced-RSA**:* Generalized Rebalanced-RSA was first proposed by Sun and Wu [13]. The complete version for discussing its security analysis is in shown [14]. Galbraith *et. al*. published the similar result in [3]. The main idea of Generalized Rebalanced-RSA is that we do not need to compute the private exponent d in the key generation. Since the decryption process just depends on the following two equations:

$$\begin{aligned} ed_p &= k_p(p-1)+1 \\ ed_q &= k_q(q-1)+1 \end{aligned} \tag{1}$$

we can choose any desired public exponent e to construct (Eq.(1)) instead of computing e by $e = d^{-1} (\bmod\ \phi(N))$. However, due to the security consideration, we cannot choose e and d arbitrarily. Sun, Hinek and Wu have provided the detailed security analysis in [14]. The similar results are also shown in [3]. Sun et. al. also provided the secure requirement when using Generalized Rebalanced-RSA. The parameters of Rebalanced-RSA should satisfy some inequalities, which will be shown later. In the next section we show an observation for CRT-equations in Rebalanced-RSA. This observation motivates us to separate two CRT-equations independently.

III. Observation

In CRT-equations of Rebalanced-RSA (see Eq.(1)), the public exponent e has to be restricted the same one due to the process of key-generation algorithm in Rebalanced-RSA. This fact also restricts the choices of e and d_p, d_q which causes the system to be time-consuming, i.e., inefficient encryption or inefficient decryption. Hence, we try to separate these two CRT-equations to make they are totally independent by assigning e_p or e_q to each CRT-equation, respectively. We modify Eq.(1) to the following format:

$$\begin{aligned} e_p d_p &= k_p(p-1)+1 \text{ and} \\ e_q d_q &= k_q(q-1)+1. \end{aligned}$$

In general, e_p, e_q, N are published and d_p, d_q, p, q must to be kept secret. The encryption/decryption procedures are shown in the following:

Encryption: To encrypt a plaintext message M, one computes the corresponding Ciphertext

$$\begin{aligned} C_p &\equiv M^{e_p} (\bmod N) \text{ and} \\ C_q &\equiv M^{e_q} (\bmod N). \end{aligned}$$

The Ciphertext is C_p and C_q.

Decryption: To decrypt the Ciphertext C_p and C_q, the legitimate receiver computes

$$\begin{aligned} M_p &\equiv C_p^{d_p} (\bmod p) \text{ and} \\ M_q &\equiv C_q^{d_q} (\bmod q). \end{aligned}$$

Then we can recovery the plaintext M form M_p and M_q by using Chinese Remainder Theorem.

Note that we should prove $M_p \equiv M \pmod p$ and $M_q \equiv M \pmod q$. We show it in the following:

$$\begin{aligned} M_p &\equiv C_p^{d_p} (\bmod p) \\ &\equiv [M^{e_p} (\bmod N)]^{d_p} (\bmod p) \\ &\equiv M^{k_p(p-1)+1} \pmod p \\ &\equiv M \pmod p \end{aligned}$$

The proof of $M_q = M \pmod q$ is similar to M_p and we ignore it here. Next, we show an application of this observation. We apply this technique to design a multisignature scheme based on RSA with CRT-exponents.

IV. Proposed Multisignature-like Scheme

In this section we present our multisignature-like scheme based on Rebalanced-RSA as follows. There are three phases in our proposed scheme: the key generation phase, the message signing phase, and the verification phase. We describe them as follows.

A. Key Generation Phase

Let $G = \{G_i | i = 1 \sim n\}$ be the group of n members and $U = \{U_i | i = 1 \sim k\}$ be the group of k signers. It denotes that U is the subset of G. In the signer's group U, there is a specified signer, called clerk. The clerk U_q of the signer's group is responsible for verifying all partial signatures signed by signers in U and combining them into a multisignature.

1) Initiation: In our system, each user becomes a formal member of the group G. The key generator runs the probabilistic polynomial algorithm to generate a random large prime p_i for $i = 1 \sim n$. Subsequently, the key generator chooses a random public key e_{p_i} and computes the private key d_{p_i} satisfying $e_{p_i} d_{p_i} \equiv 1 \bmod (p_i - 1)$ for $i = 1 \sim n$. Finally, assigns the key pair (e_{p_i}, d_{p_i}) to each member by a secret channel. The key generator keeps $p_1, p_2, ..., p_n$ secret in the system and publishes e_{p_i} for $i = 1 \sim n$.

2) Signer Secret Generation: Once a signer's group (U) is selected by the system, the clerk U_q must be decided. The key generator performs the following steps to produce the signer secret key.

1) Runs the probabilistic polynomial algorithm to generate a random large prime q and computes a large number
$$N = q \times \prod_{i=1}^{k} p_i.$$
2) Chooses a random public key e_q and computes the private key d_q satisfying $e_q d_q \equiv 1 \bmod (q - 1)$.
3) Computes D satisfying $D \equiv d_{p_i} \bmod (p_i - 1)$ for $i = 1 \sim k$ and $D \equiv d_q \bmod (q - 1)$ using the CRT.
4) Computes E satisfying $ED \equiv 1 (\bmod \phi(N))$, the key pair (E, D) is for the signer group U.
5) Computes $\bar{e} = h(e_{p_1}, e_{p_2}, \ldots, e_{p_k}, e_q)$, $h()$ is one-way hash function.
   Let
$$\alpha = \frac{E}{\bar{e}} \text{ and}$$
$$\beta = \frac{D}{d_q + d_{p_1} + d_{p_2} + \ldots + d_{p_k}} \times \alpha$$
   and transmits the key pair (e_q, d_q) and an authorization β to clerk U_q by a secret channel.

The key generator keeps q, D secret and publishes N, E and a one-way hash function $h()$. In our scheme, when a new signer's group is made up or new clerk is re-elected, the key generator runs a new prime $q^{'}$ and publishes a new $N^{'}$ to the new signer's group $U^{'}$, and the new clerk has the new key pair $(e_q^{'}, d_q^{'})$ and a new authorization $\beta^{'}$.

B. Message Signing Phase

To generate a multisignature, each signer U_i and the clerk U_q carry out the following steps to sign a same message M.

1) For each signer U_i : Computes $s_i \equiv H(M)^{d_{p_i}} (\bmod N)$ and sends s_i to the clerk U_q, $H()$ is a one-way hash function.
2) For the clerk U_q : After receiving of every signer's signature s_i, $i = 1, 2, \ldots, k$. The clerk computes $s_q \equiv H(M)^{d_q} (\bmod N)$ and constructs the multisignature
$$S \equiv \left(s_q \times \prod_{i=1}^{k} s_i \right)^{\beta} (\bmod N),$$
   the multisignature for message M is S.

Our proposed scheme allows each signer to sign the same message separately and independently, and then all individual signatures can be combined into a multisignature. In addition, the length of the multisignature is the same as the length of each signer's individual signature.

C. Verification Phase

To verify a multisignature S of message M of signers (the clerk and U_i, $i = 1, 2, \ldots, k$), any verifier uses the clerk's and signer's public key $e_q, e_{p_1}, e_{p_2}, \ldots, e_{p_k}$ and computes $\bar{e}' = h(e_{p_1}, e_{p_2}, \ldots, e_{p_k}, e_q)$ to verify the multisignature S by checking the following equality:
$$S^{\bar{e}'} \stackrel{?}{\equiv} H(M) (\bmod N)$$
If it holds, the multisignature is valid, otherwise it is invalid.

From the above verification algorithm, the number of modulo exponentiations is one. Therefore, the verification time of each multisignature is fixed.

Theorem 1: If $S^{\bar{e}'} \equiv H(M) (\bmod N)$, then S is the multisignature of M.

Proof: With the knowledge of the public key $e_{p_1}, e_{p_2}, \ldots, e_{p_k}, e_q$, a verifier can compute $\bar{e}' = h(e_{p_1}, e_{p_2}, \ldots, e_{p_k}, e_q)$. we have

$$\begin{aligned}
S^{\bar{e}'} &\equiv \left(\left(s_q \times \prod_{i=1}^{k} s_i \right)^{\beta} (\bmod N) \right)^{\bar{e}'} \\
&\equiv (H(M)^{(d_q + d_{p_1} + d_{p_2} + \ldots + d_{p_k}) \times \beta})^{\bar{e}'} (\bmod N) \\
&\equiv (H(M)^{(d_q + d_{p_1} + d_{p_2} + \ldots + d_{p_k}) \times \frac{D}{d_q + d_{p_1} + d_{p_2} + \ldots + d_{p_k}} \times \alpha})^{\bar{e}'} (\bmod N) \\
&\equiv (H(M)^{D\alpha})^{\bar{e}} (\bmod N) \\
&\equiv (H(M)^{D \times \frac{E}{\bar{e}}})^{\bar{e}'} (\bmod N).
\end{aligned}$$

According to $\bar{e} = h(e_{p_1}, e_{p_2}, \ldots, e_{p_k}, e_q)$, we can obtain
$$S^{\bar{e}'} \equiv H(M)^{DE} (\bmod N).$$
With $ED \equiv 1 (\bmod \phi(N))$, we have
$$S^{\bar{e}'} \equiv H(M) (\bmod N).$$
■

V. Algebraic Analysis: Lattice Attack

We use the technique of lattice attack to find the suitable parameter for our proposed cryptosystem. First define the notation "n_X" be the bit-length of the parameter X , and n be the bit-length of the an RSA modulus N. From the two CRT-equations, we have
$$\begin{aligned}
e_p d_p - 1 &= k_p (p - 1) \\
e_q d_q - 1 &= k_q (q - 1)
\end{aligned} \tag{2}$$

Multiplying them yields

$$e_p e_q d_p d_q - e_p d_p - e_q d_q + 1 = k_p k_q (p-1)(q-1), \quad (3)$$

which suggests that we look for small solutions of the polynomial

$$f_{e_p}(x, y, z) = x[(N+1) - y] + e_q z) - 1 \bmod e_p, \quad (4)$$

since $(x_0, y_0, z_0) = (k_p k_q, p+q, d_q)$ is a root of f_{e_p}. Note that we just consider the equation (Eq.(3)) $\bmod e_p$ but ignoring $\bmod e_q$ because the method of $\bmod e_q$ is the same situation for $\bmod e_p$. Now we consider the following useful theorem, which was proposed by Coppersmith.

Theorem 2: Let $f(x_1, \ldots, x_r)$ be a linear polynomial with integer coefficients. Let $X_1, \ldots, X_r$ be positive integers. Given $(y_1, \ldots, y_r) \in \mathbb{Z}^r$ satisfying $|y_1| < X_1, \ldots, |y_r| < X_r$, if $(y_1, \ldots, y_r)$ is a root of f modulo N and $\prod_{i=1}^{r} X_i < N$ then we can compute $(y_1, \ldots, y_r)$ in polynomial time. $\Diamond$

Now, define $X = 2^{(n_{e_p}+n_{d_p}-n/2)+(n_{e_q}+n_{d_q}-n/2)}$, $Y = 2^{n/2}$ and $Z = 2^{n_{d_q}}$ be the upper bounds of x_0, y_0, and z_0, respectively. That is, $|x_0| \leq X$, $|y_0| \leq Y$ and $|z_0| \leq Z$. According to Theorem 2, we have to set

$$XYZ < 2^{n_{e_p}}.$$

Thus,

$$n_{e_p} + n_{d_p} + n_{e_q} + n_{d_q} - \frac{n}{2} + n_{d_q} < n_{e_p},$$

which is reduced to

$$n_{d_p} + n_{e_q} + 2n_{d_q} < \frac{n}{2}. \quad (5)$$

Similarly, the method of mod e_q leads to the inequality

$$n_{d_q} + n_{e_p} + 2n_{d_p} < \frac{n}{2}. \quad (6)$$

The above two inequalities Eq.(5)(Eq.(6) show that we can not choose the parameter as small as desired. In average each parameter should be chosen larger than $\frac{n}{8}$ bits for the security consideration.

VI. Conclusions

We observe that two CRT-equations can be totally independent in Rebalanced-RSA. That is to say, we can choose different public exponents when using Rebalanced-RSA. Furthermore, we propose a multisignature-like scheme based on Rebalanced-RSA. Our proposed scheme is an efficient multisignature-like scheme that allows each signer to sign the same message parallel and all individual signatures can be combined into a multisignature. Moreover, a verifier just need an exponential operation to verify the multisignature.

References

[1] D. Boneh and H. Shacham, "Fast Variants of RSA," CryptoBytes, 2002, Vol. 5, No. 1, Springer, 2002.

[2] Y. Desmedt, "Society and Group Oriented Cryptography: A New Concept," In Advances in Cryptology, Proc. Of Crypto '87, Springer Verlag, Berlin, pp.120-127, 1988.

[3] S. D. Galbraith, C. Heneghan and J. F. McKee, "Tunable balancing of RSA", Proceedings of ACISP'05, LNCS, Vol. 3574, pp. 280-292, 2005.

[4] L. Harn and T. Kiesler, "New scheme for digital multisignature," Electron. Lett., vol. 25, no. 15, pp. 1002–1003, July 1989.

[5] L. Harna, J. Renb,"Efficient identity-based RSA multisignatures," computers & security, vol. 27, pp12 – 15, 2008

[6] T. Kiesler and L. Harn, "RSA blocking and multisignature schemes with no bit expansion," Electron. Lett., vol. 26, no. 18, pp. 1490–1491, Aug. 1990.

[7] L. M. Kohnfelder, "On the signature reblocking problem in public-key cryptography", Commun. ACM, 21, (2) pp.179, 1978

[8] T. Okamoto, "A digital multisignature scheme using bijective public-key cryptosystems," ACM Trans. Comput. Syst., vol. 6, no. 8, pp. 432–441, Nov. 1988.

[9] J. J. Quisquater and C. Couvreur, "Fast decipherment algorithm for RSA public key cryptosystem," Electronic Letters, vol. 18, pp.905-907, 1982.

[10] R. Rivest, A. Shamir and L. Aldeman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM , vol. 21, No.2, pp.120-126, 1978.

[11] H.-M. Sun, W.-C. Yang and C.-S. Laih, "On the Design of RSA with Short Secret exponent," Advances in Cryptology-ASIACRYPT'99, LNCS 1716, pp.150-164, 1999.

[12] H.-M. Sun and C.-T. Yang, "RSA with Balanced Short Exponents and Its Application to Entity

[13] 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. Authentication," Public Key Cryptography 05, 2005.

[14] H.-M. Sun, M. J. Hinek, and M.-E. Wu. On the design of Rebalanced-RSA, revised version of [13]. Technical Report CACR 2005-35, Centre for Applied Cryptographic Research, 2005. [ONLINE]. Available: http://www.cacr.math.uwaterl oo.ca/techreports/2005/cacr2005-35.pdf.

[15] M. J. Wiener, "Cryptanalysis of RSA with short secret exponents," IEEE Transactions on information Theory, IT-36, pp.553-558, 1990.