A Multiauthority Electronic Voting Protocol Based upon a Blind Multisignature Scheme

Jian-Liang Lin† Hsiu-Feng Lin†† Chih-Ying Chen† and Chin-Chen Chang †††,

Feng Chia University, Taichung, Taiwan 40724, R.O.C.

Summary
In this paper, we propose a blind multisignature scheme based on an extension of the RSA cryptosystem (called the ERSA system). Taking advantage of the scheme, we then present a new multiauthority electronic election system, which can meet all of the following requirements: eligibility, collision free, vigorousness of authorities**,** accuracy, privacy, verifiability, robustness, fairness, and prevention of ticket-buying or extortion. In addition, the computations among voters are independent without any global computation, so it is suitable for conducting a large-scale general election.

Key words:
Blind multisignature, multiauthority, electronic election system

Introduction

The design of an efficient electronic election or voting system is one of the most significant research issues in modern and future Internet applications. Compared with traditional election activities, electronic voting system is far more efficient and convenient. However, it cannot be widely adopted unless the following security requirements are all satisfied.

1. Eligibility: Only eligible voters can vote and each one can only vote once.
2. Collision Free: Each legitimate voting ticket can be uniquely identified.
3. Vigorousness of Authorities: No authority can add or subtract extra ballots to the final tally.
4. Accuracy: Any authority cannot succeed in altering the voting strategy of any voter.
5. Privacy: No one can determine for whom other voters vote.
6. Verifiability: Any voter can verify that his/her own vote has been taken into account in the final tally. Everyone can verify whether the tally published by each authority is identical or not.
7. Robustness: No malicious voter or authority can disrupt the voting procedure.
8. Fairness: The intermediate result of the election will not be leaked out.
9. Prevent Ticket-Buying and Extortion: No voter can tell any third party what his voting strategy is.

The concept of the electronic election was first introduced by Chaum [3].

Following Chaum's proposal, a lot of solutions were subsequently put forward. However, in some of the suggested solutions, the computations of voters are not independent and if any voter stops following the protocol during the voting, the election is disrupted [5,23,29]. Also, each voter needs to perform a global computation, hence they are not suitable for large-scale elections [6,13,22]. In addition, most solutions contain only one authority. The common drawbacks of a single authority electronic election system include: (1) malicious authority may add some extra ballots to the tally, (2) it is hard to prevent ticket-buying and extortion [9,12,16,17,21]. It is pointed out in [15,16,18] that the most effective way to cope with these drawbacks is to design a system consisting of more than one authority. Although some multiauthority election systems have been suggested in the past few years [1,7,8], they are still not suitable for a general election because the intentions of voters are only expressed as "Yes" or "No". Recently, based upon a blind threshold signature technique, Juang et al. [18] also proposed a multiauthority system in which the intention of a voter may be any one of more than two options. Nevertheless, it still can not prevent the problem of ticket-buying and extortion.

In this paper, we are concerned about the design of an efficient multiauthority election system. Based upon an extension of the RSA cryptosystem, named ERSA system that proposed by Feng [10], we shall first propose a blind multisignature scheme. Then, taking advantage of the proposed scheme and ERSA cryptosystem, we also present a new multiauthority electronic election system that can meet all security requirements listed above.

The rest of this paper is organized as follows. Section 2 gives a brief review of the RSA and ERSA cryptosystems. Section 3 introduces our ERSA-based blind multisignature scheme. Section 4 describes the protocol of our multiauthority electronic election system. Section 5 presents security analysis of our voting system. Finally, conclusions and future research issues are given in Section 6.

Manuscript received December 5, 2006.
Manuscript revised December 25, 2006.

2. A Brief Review of RSA and ERSA Cryptosystems

In the RSA cryptosystem [25], each user is required to obtain two large primes p and q, and a value d, relatively prime to $\varphi(n)$, for $n = p \times q$; and make public n and e, the inverse of d modulo $\varphi(n)$. A message m is encoded as an integer less than n, and the encrypted message is $C \equiv m^e \pmod n$, where (n, e) is the public key of the receiver. To decrypt the ciphertext C, the receiver may use his private key d to compute $C^d \pmod n$. Having $C^d \equiv m^{ed} \equiv m^{t\varphi(n)+1} \equiv m \pmod n$ by the Euler's theorem [26], the plaintext is recovered.

The security of the RSA system primarily relies on the computational infeasibility of factoring the used modulus n into a product of p and q [25]. Accordingly, to resist some factoring attacks, p and q must be chosen carefully as strong primes [14, 20]. Besides, in order to withstand a ciphertext attack given by Simmons and Norris [28], p and q must be as small as possible such that (p-1, q-1), denoting the greatest common divisor of p-1 and q-1 (since p-1 and q-1 are both even, the minimum value for (p-1,q-1) is 2.)

Nevertheless, the need to protect the secrecy of the factors of the used modulus n disable the direct adaptation of the RSA scheme to a group-oriented or distributed communication environment. Since a group-oriented or distributed cryptosystem always involves many players, it is desirable to use a "universal" modulus for all players to simplify computing complexities. In the RSA cryptosystem, however, if the modulus n is universal and if each player has to know the factoring of n to decide his secret key, there will be no secret among all internal members.

To make RSA system more for the design of cryptosystems in a group-oriented or distributed communication environment, Feng [10] proposed an extension of the RSA system, called the ERSA cryptosystem. Vectors $< e_1, e_2, \cdots, e_r >$ and $< d_1, d_2, \cdots, d_r >$ whose inner product satisfies $e_1d_1 + e_2d_2 + \cdots + e_rd_r \equiv 1 \pmod{\varphi(n)}$ are used instead of e and d, as the encryption and decryption keys, respectively, where $n = p \times q$ is the same as that appears in the original RSA system. It is seen that some previously suggested RSA variants with additive share keys [2,11,24] are in fact special cases of ERSA system. Also, when r=1, the ERSA system is just the same as the original RSA system.

ERSA scheme can also achieve secure communication between two individuals. The ciphertext for a message m is computed as a vector $C = < c_1, c_2, \cdots, c_r >$, where $c_i \equiv m^{e_i} \pmod n$ and $(n, < e_1, e_2, \cdots, e_r >)$ is the public key of the receiver. The plaintext for C can be reconstructed by computing $\prod_{i=1}^{r} c_i^{d_i} \equiv \prod_{i=1}^{r} m^{e_i d_i} \equiv m^{e_1d_1+e_2d_2+\cdots+e_rd_r} \equiv m \pmod n$. In this case, ERSA seems to have no superior over the original RSA system in terms of the encryption/decryption time and the size of the ciphertext is r times of that of an RSA system. Nevertheless, since vector encryption/decryption keys are introduced, by distributing $d_1, d_2, \cdots, d_r$ to r individuals in a communication network as their secret keys or shared secret keys, the ERSA scheme is thus more adapted for the design of cryptosystem in a group-oriented or distributed communication environment than the original RSA system. Based on ERSA, Feng [10] has presented a generalized group-oriented cryptosystem. Still based on ERSA, we shall propose a blind multisignature scheme later in this paper.

The security of the ERSA system is almost the same as that of RSA system provided that all parameters p, q, n, $e_1, e_2, \cdots, e_r$ and $d_1, d_2, \cdots, d_r$ are properly determined. Similarly, to ensure that the factoring of $n = p \times q$ is intractable and to withstand some known ciphertext attacks, both p and q must be chosen as strong primes such that (p-1, q-1)=2. However, unlike the original RSA system where there is only one public value e, the public values e_i's, $1 \le i \le r$, of the ERSA system must be chosen to let $(e_i, e_j) > 1$ for $i \ne j$ to avoid direct derivation of a message m from a system of equations such as $c_i \equiv m^{e_i} \pmod n$ and $c_j \equiv m^{e_j} \pmod n$ by the Euclidean algorithm.

Accordingly, the essential problem concerning the ERSA system is that if there exists an efficient algorithm that can determine a set of proper parameters p, q, n, e_i and d_i, $1 \le i \le r$, such that (1) both p and q are strong primes (2) (p-1, q-1)=2 and (3) $(e_i, e_j) > 1$ for $i \ne j$ and $e_1d_1 + e_2d_2 + \cdots + e_rd_r \equiv 1 \pmod{\varphi(n)}$. Though it is not an easy fortification, an efficient algorithm for generating all proper ERSA parameters has already been given in [10]. The readers who are interested in this algorithm may refer to [10].

3. An ERSA Based Blind Multisignature Scheme

The concept of blind signature scheme was introduced by Chaum in 1982 [4]. In a blind signature system, a signer shall have no idea of what he signs. It means a signer must not be able to find a relationship between some blinded and unblinded parameters. This property is usually referred as the unlinkability property. Accordingly, blind signatures are widely used to construct anonymous electronic election schemes [9,12,15,16,18].

As we have pointed out earlier in this paper that the most effective method to prevent the authority of a single-authority voting system being cheated is to develop a multiauthority voting system and an efficient blind multisignature is indispensable. Accordingly, based on the concept of the ERSA cryptoscheme, we shall propose such a blind multisignature scheme in this section [16].

Suppose that there are r signers A_i's, $1 \le i \le r$, a signature requester denoted as B, and a trusted key generating center (KGC). Then, the generation and verification of our blind multisignature scheme can be described as follows.

Key Generating Phase:

1. The KGC generates a set of ERSA parameters including $(n, e_1, e_2, \cdots, e_r, d_1, d_2, \cdots, d_r)$ satisfying $(e_i, e_j) > \alpha$ if $i \ne j$, and $e_1 d_1 + e_2 d_2 + \cdots + e_r d_r \equiv 1 \pmod{\varphi(n)}$.
   (Note that according to the ERSA parameters generating algorithm given by Feng [10], α is a prime satisfying $\alpha \equiv 1 \pmod 4$ and can be made arbitrarily large.)
2. The KGC publishes n and distributes e_i and d_i to each A_i, $1 \le i \le r$, as his public and private keys, respectively.

Blind Multisignature Generation Phase:

Suppose B wants A_i, $1 \le i \le r$, to sign a message m blindly, where $m \in Z_n = \{0,1,2,\ldots,n-1\}$.

1. B determines two large strong primes p and q such that it is computationally infeasible to factor the value of their product.
2. For each $1 \le i \le r$, B computes $R_{1i} \equiv p^{e_i} m \pmod n$ and $R_{2i} \equiv q^{e_i} m^{-1} \pmod n$, and sends (R_{1i}, R_{2i}) to A_i.
3. Once receiving (R_{1i}, R_{2i}) from B, each A_i, $1 \le i \le r$, computes $w_{1i} \equiv (R_{1i})^{d_i} \pmod n$ and $w_{2i} \equiv (R_{2i})^{d_i} \pmod n$ as his blind signature for m. Then he sends (w_{1i}, w_{2i}) back to B.
4. After receiving all pairs (w_{1i}, w_{2i}) from A_i, $1 \le i \le r$, B computes W_1, W_2 and T as :

$$W_1 \equiv \prod_{i=1}^{r} w_{1i} \pmod n,$$

$$W_2 \equiv \prod_{i=1}^{r} w_{2i} \pmod n,$$

$$T \equiv p^{-1} W_1 \pmod n,$$

where T is served as the blind multisignature of m from A_i, $1 \le i \le r$, and (W_1, W_2) is preserved for verifying the blind multisignature.

Blind Multisignature Verification Phase:

After obtaining the values of W_1, W_2 and T, B can make sure the validity of T by checking whether $W_1 W_2 \equiv pq \pmod n$. If it holds, the blind multisignature T is proved to be correct.

Blindness Discussion and Security Analysis:

Observe Step (3) of the signature generation phase to see if each A_i, $1 \le i \le r$, computes w_{1i} and w_{2i} with his genuine private key d_i. Then in Step (4) we will have

$$W_1 \equiv \prod_{i=1}^{r} w_{1i} \equiv p^{e_1 d_1 + e_2 d_2 + \cdots + e_r d_r} m^{d_1 + d_2 + \cdots + d_r}$$

$$\equiv p\ m^{d_1 + d_2 + \cdots + d_r} \pmod n,$$

and

$$W_2 \equiv \prod_{i=1}^{r} w_{2i} \equiv q^{e_1 d_1 + e_2 d_2 + \cdots + e_r d_r} m^{-(d_1 + d_2 + \cdots + d_r)}$$

$$\equiv q\ m^{-(d_1 + d_2 + \cdots + d_r)} \pmod n,$$

for $p^{e_1d_1+e_2d_2+\cdots+e_rd_r} \equiv p \pmod n$ and $q^{e_1d_1+e_2d_2+\cdots+e_rd_r} \equiv q \pmod n$ according to the encryption/decryption principle of the ERSA cryptosystem. In this case, we also have, in Step (4), $T \equiv p^{-1}W_1 \equiv m^{d_1+d_2+\cdots d_r} \pmod n$. Accordingly, if $W_1W_2 \equiv pq \pmod n$ holds, the blind multisignature T for the message m is indeed verified.

The blindness of $T \equiv m^{d_1+d_2+\cdots d_r} \pmod n$ can be seen as follows. Each A_i, $1 \le i \le r$, is unable to see the value of the message m from R_{1i} or R_{2i} due to unknown of the values of the blinding factors p and q. On the other hand, even if all A_i's, $1 \le i \le r$, can cooperate to compute the values of W_1 and W_2, and obtain the value of $p \times q$ from computing $W_1W_2 \pmod n$, they still cannot get the values of p or q. This is because, in Step (1) of the signature generation phase, p and q are chosen as two strong primes by B such that it is computationally infeasible to factor the value of their product. Consequently, only when the signature requester B himself holds the blinding factors p and q can he verify, unblind, and obtain the valid blind multisignature T.

The security of the above proposed scheme is based on the ERSA system which is guaranteed by the computational infeasibility of factoring the used modulus. An attacker is hard to forge a legitimate blind multisignature unless he knows the factoring of n.

On the other hand, if some A_i applies a fake secret key to sign R_{1i} and R_{2i}, $W_1W_2 \equiv pq \pmod n$ cannot be correct and the blind multisignature will not be verified. Hence, no A_i can forge a legitimate individual blind signature for m. It is also impossible for any A_i to sign m in R_{1i} and m^{-1} in R_{2i} with a fake key while sign p^{e_i} in R_{1i} and q^{e_i} in R_{2i} with his genuine key because he is unable to separate p^{e_i} and m from R_{1i}; and q^{e_i} and m^{-1} from R_{2i}. Consequently, that all signers cooperatively forge a legitimate blind multisignature is hard to realize.

In addition, after receiving the blind multisignature T for m, the signer cannot succeed to forge another legitimate blind multisignature pair (m', T') by computing $m' = a^{e_i}m$ and $T' = aT$, where a is any random integer. Now, we have $\prod_{i=1}^{r} T_i' \equiv T' \pmod n$ where $T_i' \equiv (m')^{d_i} \pmod n$, $\prod_{i=1}^{r} (T_i')^{e_i} \neq m \pmod n$. Accordingly, the singers can easily detect the forgery by checking together whether $\prod_{i=1}^{r} T_i^{e_i} \equiv m \pmod n$ holds or not.

Consequently, given a message and blind multisignature pair (m,T), all signers A_i's, $1 \le i \le r$, can cooperate to verify that T is indeed a blind multisignature of m by first computing $T_i \equiv m^{d_i} \pmod n$, $1 \le i \le r$, individually, and checking together whether $\prod_{i=1}^{r} T_i \equiv T \pmod n$ and $\prod_{i=1}^{r} T_i^{e_i} \equiv m \pmod n$ are true. However, due to the blindness of T, each A_i is unable to learn when and for whom T is produced.

4. A New Multiauthority Electronic Election System

In this section, based on the ERSA blind multisingnature scheme that we suggested in the last section and the ERSA cryptosystem suggested by Feng [10], we shall propose a new multiauthority electronic election system which will meet all security requirements mentioned in Section 1.

4.1 Basic Assumptions

The underlying assumptions of our protocol include:

1. Every legitimate voter has his own RSA keys.
2. Every authority has his own RSA keys.
3. No two voters can stay in one voting booth at the same time, but only one.
4. The KGC (key generating center) is trustworthy.
5. All authorities will not conspire simultaneously; namely, there must exist at least one honest authority.
6. Both the discrete logarithm problem and the factorization problem are computationally infeasible.

4.2 The Protocol

There are four participants in our system, i.e., a key generating center (KGC), r authorities (A_i, $1 \le i \le r$), voters and a set of voting booths. The key generating center (KGC) is responsible for generating the system parameters for all authorities, while the authorities are responsible for checking the identity of each voter, issuing

blind voting tags to legitimate voters, collecting, opening, counting and publishing the votes. The aid of voting booths is indispensable. Although it is very convenient for a voter to register and vote electronically from anywhere, it cannot prevent ticket buying and extortion. The reason is that the buyer has to get involved in the voting process to see how the voter votes [15,18]. Accordingly, in order to prevent ticket buying and extortion thoroughly, voting booths are necessary. The voting protocol of our system consists of four phases described exhaustively as follows.

4.3 The Voting Protocol

The details of voting protocol of our multiauthority electronic election system will be given as follows.

Phase 0 (Initialization Phase)

1. The key generating center (KGC) selects and publishes a large prime P such that P-1 has at least one large prime factor, and a fixed primitive root g in the field GF(P).
2. The KGC generates a set of ERSA parameters (N_A , $e_{A_1}, e_{A_2}, \cdots, e_{A_r}$, $d_{A_1}, d_{A_2}, \cdots, d_{A_r}$) and distributes each pair of e_{A_i} and $d_{A_i}, 1 \le i \le r$, to the i-th authority A_i as his ERSA public key and private key, respectively.
3. Each A_i, $1 \le i \le r$, selects a secret random integer $x_{A_i} \in Z_P$ and publishes the value $y_{A_i} \equiv g^{x_{A_i}} \pmod{P}$.
4. Each A_i, $1 \le i \le r$, publishes the list of legitimate voters.

Phase 1 (Registration Phase)

In this phase, each voter should identify himself to each authority that he is a real legal voter and has not cast his vote yet. At the same time, he should apply the ERSA blind multisignature technique to get a blind voting tag from each authority. The detailed procedure can be described as follows.

Step 1: [V's term]

1. Voter V determines two strong primes p and q such that it is computationally infeasible to factor the value of their product.
2. (2.1) V selects a secret integer m_v in Z_{N_A}.
   (2.2) For each $1 \le i \le r$, V computes the blinded message $R_{1i} \equiv p^{e_{A_i}} m_v \pmod{N_A}$ and $R_{2i} \equiv q^{e_{A_i}} m_v^{-1} \pmod{N_A}$, where e_{A_i} is the ERSA public key of A_i.
3. (3.1) For each $1 \le i \le r$, V computes the signature $S_i \equiv (ID_v \parallel R_{1i} \parallel R_{2i})^{d_v^*} \pmod{n_v^*}$, where ID_v is the identification number and (n_v^*, d_v^*) is V's RSA secret key.
   (3.2) V sends $(ID_v \parallel S_i \parallel t)$ to each authority A_i, $1 \le i \le r$, where t denotes a timestamp and $\parallel$ denotes the concatenation of bit-strings.
   (Note that in order to prevent the messages from being maliciously tampered, V sends S_i instead of $(ID_v \parallel R_{1i} \parallel R_{2i})$ in Step (3.2). However, we shall assume $n_v^* > N_A$ to avoid the reblocking problem [19,25,27].)

Step 2: [A_i 's, term, $1 \le i \le r$]

1. Receiving V's registration request, each authority A_i, $1 \le i \le r$, has to validate V's digital signature and checks his database to see if V has not registered yet.
2. If 1. holds, A_i records V's registration, uses his ERSA private key d_{A_i} to sign R_{1i} and R_{2i} blindly as $w_{1i} \equiv (R_{1i})^{d_{A_i}} \equiv p^{e_{A_i} d_{A_i}} m_v^{d_{A_i}} \pmod{N_A}$ and $w_{2i} \equiv (R_{2i})^{d_{A_i}} \equiv q^{e_{A_i} d_{A_i}} m_v^{-d_{A_i}} \pmod{N_A}$, respectively , then sends them back to V.

Step 3: [V's term]

1. After getting w_{1i} and w_{2i} from A_i, $1 \le i \le r$, V computes
   $$W_1 \equiv \prod_{i=1}^{r} w_{1i} \pmod{N_A},$$
   $$W_2 \equiv \prod_{i=1}^{r} w_{2i} \pmod{N_A},$$
   and $T_v \equiv p^{-1} W_1 \pmod{N_A}$.
2. V checks whether $W_1 W_2 \equiv pq \pmod{N_A}$ holds. If it holds, V accepts T_v as the blind multisignature of m_v. And (m_v, T_v) is preserved as V's legitimate

blind voting tag, which will be served as a permission of casting in the voting phase.

Phase 2 (Voting and Verifying Phase)

Once the voter V obtains the valid blind voting tag (m_v, T_v) from the authorities, he constructs and casts his voting tickets to each authority.

Step 1: [V's term]

1. V determines his voting strategy denoted as Z_v.
2. (2.1) V determines an ERSA modulus N_v.
   (2.2) V selects a secret integer $x_v \in Z_P$ and computes
   $y_v \equiv g^{x_v} \pmod P$ and $d_i \equiv y_{A_i}^{x_v} \pmod P$,
   such that $d_i \neq d_j$ for $i \neq j$ and $(d_i, \varphi(N_v)) = 1$ for $1 \le i \le r$.
3. Applying the algorithm given by Feng [10], V determines a set of integers $e_{v_1}, e_{v_2}, \cdots, e_{v_r}$ satisfying $(e_{v_i}, e_{v_j}) > \alpha$ if $i \neq j$, where α is a large prime of the form $\alpha \equiv 1 \pmod 4$, and $e_{v_1} d_1 + e_{v_2} d_2 + \cdots + e_{v_r} d_r \equiv 1 \pmod{\varphi(N_v)}$.
4. V encrypts the voting strategy Z_v into r pieces $C_i \equiv Z_v^{e_{v_i}} \pmod{N_v}$, $1 \le i \le r$.
5. For each $1 \le i \le r$, V constructs a voting ticket for A_i as
   $U_i = (m_v \| T_v \| C_i \| e_{v_i} \| N_v \| y_v \| \theta_v \| \theta'_v \| t)$,
   where θ_v and θ'_v are two selected random integers between 1 and the minimum value of $\{n^*_{A_i} \mid 1 \le i \le r\}$, where $n^*_{A_i}$ is an RSA modulus for A_i.
6. For each $1 \le i \le r$, V encrypts each voting ticket U_i into
   $U'_i \equiv U_i^{e^*_{A_i}} \pmod{n^*_{A_i}}$,
   and sends it to A_i, where $(n^*_{A_i}, e^*_{A_i})$ is the RSA public key of A_i.

(Note that the purpose of encrypting U_i into U'_i is to prevent a malicious attacker to forge a legitimate voting ticket from intercepting the blind voting tag (m_v, T_v). Consequently, we have to set $n^*_{A_i} > N_A$, $n^*_{A_i} > P$ and $n^*_{A_i} > N_v$ to avoid the possible reblocking problem [19,25,27].)

Step 2: [A_i's, term, $1 \le i \le r$]

It is seen that each voting ticket casted out by V is untraceable and does not reveal any information about V while each authority can still ensure the authenticity of the received voting ticket. The authenticating process is as follows.

1. After receiving the voting ticket U'_i, each A_i decrypts it by his own RSA secret key $d^*_{A_i}$ and checks whether the blind voting tag (m_v, T_v) has been used before to avoid double voting. All voting tickets that contain the same blind voting tag will be published and not be accepted by any authority.
2. Each A_i, $1 \le i \le r$, checks the validity of (m_v, T_v) through the following procedure.
   (2.1) Each A_i computes $T_i \equiv m_v^{d_{A_i}} \pmod{N_A}$ and broadcasts it to other A_js, $1 \le j \le r$ and $j \neq i$.
   (2.2) Each A_i checks whether $\prod_{i=1}^{r} T_i \equiv T_v \pmod{N_A}$ and $\prod_{i=1}^{r} T_i^{e_{A_i}} \equiv m_v \pmod{N_A}$. If both are correct, then the validity of (m_v, T_v) has been confirmed.

Phase 3 (Counting and Publication Phase)

After the deadline of vote casting, all authorities start to decipher V's voting strategy cooperatively with the ERSA group-oriented decryption technique [10]. The details are as follows.

Step 1: [A_i's, term, $1 \le i \le r$]

1. Each A_i, $1 \le i \le r$, computes
   $d_i \equiv y_v^{x_{A_i}} \pmod P$,
   $B_i \equiv C_i^{d_i} \pmod{N_v}$,
   $k_{ij} \equiv y_{A_j}^{x_{A_i}} \pmod P$, $1 \le j \le r$ and $j \neq i$,
   $G_{ij} = E_{k_{ij}}(B_i)$, $1 \le j \le r$ and $j \neq i$,

and sends G_{ij} to other A_j s, $1 \le j \le r$ and $j \ne i$, where $E_{k_{ij}}$ is an encryption operation using the secret key k_{ij}.

2. (2.1) On receiving all G_{ij}, $1 \le j \le r$ and $j \ne i$, each A_i computes $B_j = D_{k_{ij}}(G_{ij})$ and retrieves V's voting strategy as $Z_{vi} \equiv \prod_{j=1}^{r} B_j \pmod{N_v}$.
   (2.2) Each A_i, $1 \le i \le r$, checks whether $Z_{vi}^{e_{vi}} = C_i \pmod{N_v}$. If it holds, A_i accepts Z_{vi} as V's voting strategy Z_v.
   Otherwise, it may indicate that there exists at least one maliciously cheating authority. We will discuss this case later in Section 5.
3. Each A_i, $1 \le i \le r$, publishes the value Z_{vi} as the voting strategy of V.

5. Security Analysis

In this section, we are going to prove that our voting system can satisfy all the security requirements including eligibility, accuracy, privacy, verifiability, robustness, fairness and prevention of ticket-buying and extortion as follows.

Theorem 1(Eligibility): Only eligible voters can vote and can vote only once.

Proof. The eligibility of voter V is checked by each authority in Step 2-(1) of Phase 1. Meanwhile, multiple registrations from the same voter can also be checked and prohibited by Step 2-(1) of Phase 1. Then, double voting (voting with the same blind voting tag) is checked and prohibited in Step 2-(1) of Phase 2.

Theorem 2(Collision Free): Each legitimate voting ticket can be uniquely identified.

Proof. It is required that each m_v be of the form $m_v \equiv r_1 \cdot r_2^{r_3} \pmod{N_A}$ in Step 1-(2.1) of Phase 1, where r_1, r_2 and r_3 are random numbers in Z_{N_A}, it is quite unlikely that two distinct voters have the same blind voting tag. Therefore, each legitimate voting ticket can be uniquely identified.

Theorem 3(Vigorousness of Authorities): No authority can add or subtract extra ballots to/from the final tally.

Proof. Under the assumption that authorities do not conspire simultaneously, one can easily prevent any malicious authority from adding or substracting extra ballots to/from the final tally. A simple method is to check in Step 1-(3) of Phase 3 and see whether the tally published by each authority is identical or not.

Theorem 4(Accuracy): No authority can alter the voting strategy of any voter.

Proof. Suppose that in Step 1-(1) of Phase 3, the authority A_i is malicious and uses a false key x'_{A_i} to compute $d'_i \equiv y_v^{x'_{A_i}} \pmod{P}$, $B'_i \equiv C_i^{d'_i} \pmod{N_v}$ and send $G'_{ij} = E_{k_{ij}}(B'_i)$ to other authorities to try altering the voter V's genuine voting strategy. In this situation, the cheating can be detected by the verification in Step 1-(2.2) of Phase 3. Once it happens, the authorities have to publish one of the values $\{\theta_v, \theta'_v\}$ that are involved in the voting ticket of V. When seeing his θ_v (or θ'_v) on the announcement, V has to prove that he is the owner of the voting ticket by sending θ'_v (or θ_v) to the authorities without revealing his real identity. Then, V has to send some extra information, according to the cheater detection and fake shadow key correction procedures of the ERSA cryptosystem suggested in Feng [10], to help detecting the malicious authority (cheater) and correcting the fake shadow key d'_i. As a result, V's original voting strategy can finally be recovered.

Theorem 5(Privacy): No one can determine for whom others voted.

Proof. It is impossible that two voters can stay in one voting booth at the same time, so no one (including a ticket-buyer) can get to know how others vote. In addition, the legitimacy of a vote comes from the valid blind voting tag. Due to the blindness preserved by the ERSA based blind multisignature scheme, even the authorities do not know to whom a blind voting tag belongs. The blind voting tags and the voting tickets thereafter have been sent to each authority anonymously. There is no way to trace them back to the originated voters. Hence, the privacy of the voters is preserved.

Theorem 6(Verifiability): Any voter can verify whether his own vote has been taken into account in the final tally. And everyone can verify whether the tally published by each authority is identical or not.

Proof. Under the assumption that all authorities do not conspire simultaneously, any voter can simply perform Step 1-(3) of Phase 3 to check whether the tally published by each authority is identical or not, and thus see if his own vote has been counted in the final tally.

Theorem 7(Robustness): Not any malicious voter or authority can disrupt the voting procedure.

Proof. According to the blindness discussion and security analysis of Section 3, it is for a malicious voter or authority to forge a legitimate blind multisignature unless he knows the factoring of N_A. That is, it is impossible for a voter or authority to cast more than one legitimate vote. In addition, because of unknown of the voters' RSA private keys, any authority is unable to make false registrations for the voters who do not register in the registration phase Further, the ERSA-based blind multisignature is used to distribute the power of a single authority to several persons, so that voters can abstain from voting after their registrations.

Theorem 8(Fairness): The intermediate result of the election will not be leaked out.

Proof. Without all other authorities' cooperation, no single authority can open a legal vote on his own. However, if all authorities conspire to open a vote before the deadline of vote casting, the fairness property will be violated. Since this contradicts to our assumption, the fairness can be preserved.

Theorem 9(Prevent Ticket-Buying and Extortion): Any voter cannot prove to any third party what his voting strategy is.

Proof. Since the publication of each authority does not reveal any relationship between a voter and his vote, and any voter can verify that his own vote has been taken into account in the final tally by just checking whether the tally published by each authority is identical or not. Accordingly, any voter has no evidence to show anyone else whom he voted to.

6. Conclusions

It has been pointed out that the most effective way to prevent the authority of a single authority voting system from being cheated is to develop a multiauthority voting system. However, due to the lack of an efficient blind multisignature scheme, so far there still has no efficient multiauthority voting system. In this paper, based on an extension of the RSA cryptosystem (called the ERSA system), we first proposed an efficient blind multisignature scheme. Taking advantage of this blind multisignature scheme, we also developed a new multiauthority electronic election system. Having been proved with experiments, our voting system can meet all security requirements of being a soundly electronic election system. Significantly, we have solved the vital problem of cheating by the authority and the problem of ticket–buying, which inherently exist in most single-authority electronic voting systems suggested previously.

In addition, our system has a number of practical properties, including (1) voters can abstain from voting after the registration phase. (2) The computations among voters are independent without the need of any global computation, so the system is suitable for conducting large-scale general elections. (3) Voters can determine the encryption /decryption keys of votes themselves. Especially, property (3) significantly prevents any voter from maliciously accusing the authorities of having altered his voting strategy.

References

[1] J. Benaloh, D. Tuinstra, "Receipt free secret ballot elections," Proc, of the 26th Annual ACM Symp. on the Theory of Computing, pp. 544-553, 1994.

[2] C. Boyd, "Digital multisignatures," Cryptography and Coding, pp.241-246, Claredon Press, 1989.

[3] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," Communications of the ACM, Vol.24, No.2, pp. 84-88, 1981.

[4] D. Chaum, "Blind signatures for untraceable payments," Advances in Cryptology-Crypto'82, Springer-Verlag, pp. 199-203, 1983.

[5] D. Chaum, "Elections with unconditionally-secret ballots and disruption equivalent to breaking RSA," Advances in Cryptology-EUROCRYPT'88 Proceedings, Lecture Notes in Computer Science, Vol. 330, (C. G. Gunther, Editor), Springer-Verlag, pp. 177-182, 1988.

[6] D. Chaum, C. Crepeau, and I. Damgard, "Multiparty unconditionally secure protocols," Proceedings of the 20th Annual ACM Symposium on Theory of Computing, pp. 11-19, May 1988.

[7] R. Cramer, M. Franklin, B. Schoenmakers, and M. Yumg, "Multi-authority secret-ballot elections with linear work," Advances in Cryptology-EUROCRYPT'96, Springer-Verlag, pp. 72-83, 1996.

[8] R. Cramer, R. Gennaro, and B. Schoenmakers, "A secure and optimally efficient multi-authority election scheme," Advances in Cryptology: Proc. of EuroCrypt'97, Lecture Notes in Computer Science, Vol. 1233, Springer-Verlag, pp. 103-118, 1997.

[9] C. I. Fan, C. L. Lei, and C. Y. Chang, "An efficient election scheme for resolving ties," International Computer Symposium, Workshop on Cryptology and Information Security, Taiwan, R.O.C, pp. 95-100, 1998.

[10] Y. M. Feng, "Extended RSA based generalized group-Oriented Cryptosystem and signature system," MS Thesis, Institute of Electrical Engineering, Feng Chia University, Jan. 1999.

[11] Y. Frankel, "A practical protocol for large group-oriented networks," Eurocrypt'89, pp. 56-61, 1989.

[12] A. Fujioka, T. Okamoto, and K. Ohta, "A practical secret voting scheme for large scale elections," Advances in Cryptology: Proc. Of AusCrypt'92, Lectures Notes in Computer Science, Vol. 718, Springer-Verlag, pp. 244-251, 1992.

[13] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game or a completeness theorem for protocols with honest majority," Proceedings of the 19th Annual

ACM Symposium on Theory of Computing, pp. 218-229, May 1987.
[14] J. Gordan, "Strong RSA key," Electronics Letters, Vol.20, pp. 514-516, 1984.
[15] S. I. Hwang, "An electronic voting system with enhanced security," International Computer Symposium, Workshop on Cryptology and Information Security, Taiwan, R.O.C, pp. 87-94, 1998.
[16] W. S. Juang and C. L. Lei, "A secure and practical electronic voting scheme for real world environments," IEICE Trans. On Fundamentals, Vol. E80-A, No.1, pp. 64-71, Jan. 1997.
[17] W. S. Juang, C. L. Lei, and C. I. Fan, "A collision free secret ballot protocol for computerized general elections," International Computer Symposium, Taiwan, R.O.C., pp. 309-314, 1994.
[18] W. S. Juang, C. L. Lei, and P. L. Yu, "A verifiable multi-authorities secret election allowing abstaining from voting," International Computer Symposium, Workshop on Cryptology and Information Security, Taiwan, R.O.C., pp. 101-108, 1998.
[19] L. M. Konfelder, "On the signature reblocking problem in public-key cryptosystem," Comm. ACM, Vol. 21, p. 179, 1978.
[20] C. S. Laih, W. C. Yang and C. H. Chen, "Efficient method for generating strong primes with constraint of bit length," Electronics Letters, Vol. 27, No. 20, pp. 1807-1808, 1991.
[21] H. Nurmi, A. Salomaa, and L. Santean, "Secret ballot elections in computer networks," Computers & Security, Vol. 10, pp. 553-560, 1991.
[22] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computation," Proceedings of the 20th Annual ACM Symposium on Theory of Computing, pp. 1-10, May 1988.
[23] C. P. Pfleeger, "Security in computing," PrenticeHall, Inc, 1989.
[24] T. Rabin, "A simplified approach to threshold and proactive RSA," Crypto'98, pp. 89-104, 1998.
[25] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public key cryptosystem," Communications of the ACM, Vol. 21, No. 2, pp. 120-126, 1978.
[26] K. H. Rosen, "Elementary number theory and its applications," Second Edition, Addison-Wesley Publishing Company, pp. 299-342, 1987.
[27] M. Shimada and K. Tanaka, "Blocking method for RSA cryptosystem without expanding cipher length," Electronics Letters, Vol. 25, No.12, pp. 773-774, 1989.
[28] G. J. Simmons and J. N. Norris, "Preliminary comments on the M.I.T. public key cryptosystem," Cryptologia, Vol. 1, pp. 406-414, 1977.
[29] A. Yao, "Protocols for secure communications," Proceedings of the 23rd Annual IEEE Symposium on the Foundations of Computer Science, pp. 160-164, 1982.