A Practical Electronic Voting Protocol Using Threshold Schemes

Ahmad Baraani-Dastjerdi
Josef Pieprzyk
Reihaneh Safavi-Naini

The Centre for Computer Security Research
Department of Computer Science
University of Wollongong
Wollongong, NSW 2500, AUSTRALIA

e-mail: [g9272860, josef, rei].uow.edu.au

May 27, 1994

Abstract

This paper presents a novel secret voting scheme which fully conforms to the requirements of large scale elections. The participants in the scheme are voters, candidates, an administrator, and a counter. The scheme uses threshold encryption to preserve the privacy and accuracy of the votes against dishonesty of voters, candidates, the administrator, and the counter. It also ensures verifiability, fairness, and soundness of the voting process and hence neither administrator nor candidates, or the counter is capable of producing false tally, affecting the voting result, or corrupting/disrupting the election.

A Practical Electronic Voting Protocol Using Threshold Schemes

1 Introduction

The development of cryptographic techniques allow us to "computerize" many areas of human activities. Voting is an important human activity in a democratic society. The problem of providing a secure system which ensures privacy of the voters and accuracy of the votes, and is suitable for application in large scale voting has been the subject of research for a number of years. Work has been done from both theoretical and practical points of view [1, 2, 9, 5, 15, 4, 6, 10, 11, 12, 3]. In some proposal, the whole voting procedure is controlled by the voters. To ensure the correctness of voting results usually many interactions among voters are necessary. However, in a real voting system there is no interaction among voters and hence such schemes are only of theoretical interest but are not practical.

In this paper, our goal is to design a *secure and practical* voting scheme. Our work is inspired by the work of Fujioka, Okamoto, and Ohta [3]. However, in the system proposed by Fujioka et. al, the validity of the election depends only on two parameters: the number of eligible voters announced by the administrator, and the number of valid votes received by the counter and hence it is required that all the registered voters cast their votes and no voter abstain from voting. In real life, this assumption is too rigid and unrealistic. There is always a possibility that a voter intentionally or accidentally does not forward his/her vote to the counter even after applying for registration. But failure of a single voter should not disrupt the entire election. We propose a scheme that overcomes this drawback and retains all the properties of Fujioka et. al. scheme.

Our scheme includes voters, candidates, the counter, and the administrator. The basic assumptions of the scheme are: (1) voters, candidates, and the counter can communicate over an anonymous channel, (2) the signature schemes are secure, and (3) the majority of candidates (that is at least half of the candidates) are honest. Under these assumptions, the scheme is robust and the privacy of the voters, soundness of voting, and fairness of casting are ensured, i.e., the votes, no subset of voters can disrupt or corrupt the election, and no one can benefit from the intermediate result of voting.

The organization of this paper is as follows. In Section 2, a brief review of voting protocols to date is presented. Next, we present notations and assumptions which are used in this paper (Section 3). In Section 4, the proposed voting scheme is presented. In Section 5, the security of the proposed scheme is analyzed. Conclusions are given in Section 6.

2 Related Work

In order to design a secure election protocol that provides secrecy and authenticity, two approaches for the transmission of untraceable yet authentic ballots are proposed:

1. the ballot be sent in an encrypted form, and
2. the ballot be sent through an anonymous communication channel.

In first approach proposed by Benaloh and Yung [5], Cohen and Fischer [15], and Iversen [4] using encryption techniques.

The second approach, proposed by Chaum [6], Ohta [10], Asano, Matsumoto and Imai [11], Boyd [12], and Fujioka, Okamoto and Ohta [3], uses an anonymous communication channel, and an administrator, and provides unconditional security against tracing the votes.

The use of an anonymous communication channel was first proposed by Chaum [16, 6]. His voting scheme provides unconditional security against tracing the votes but failure of a single voter can disrupt the election. However the failure can be traced, and the election will restart without the faulty voter. This scheme is not practical for large scale elections.

Boyd [12] proposed a voting scheme based on a multiple key cipher which is simply defined by the group of exponentiation transformations in a prime field known to be oneway. The scheme ensures that the votes can not be forged, and privacy of the votes are preserved, provided that the voters can deliver their votes anonymously. However, the main problem with this scheme is the ability of the government to see, and even worse produce a false tally by adding votes of its own choice.

Iversen [4] proposed a cryptographic voting scheme using encryption technique. The participants in the scheme are the voters, the candidates and the government. It is assumed that the voters and the candidates communicate via a broadcast channel. The scheme ensures independence of the voters and unlike [2, 1] the scheme does not require them to be presenet at the same time or to go through several phases together. Moreover, no global computation is needed. The scheme preserves the privacy of the votes against any proper subset of dishonest voters, and/or any proper subset of dishonest candidates, including government. Also it ensures that no subset of voters can corrupt or disrupt the election. The essential drawback of this scheme is that if all the centers (candidates) conspire the privacy of the voters is violated. Moreover this scheme is less practical for large scale elections, since it requires much communication and computation when the number of voters is large.

Fujioka et.al.[3] have proposed a voting scheme which overcomes most of the previous problems such as fairness, privacy, and practicality. Their scheme is the most suitable scheme for large scale elections, since the communication and computation overhead is fairly small even if the number of voters is large. However, the main problem with their scheme is that it requires all the registered voters to cast their votes and no voter abstain from voting. In fact, the failure of a single voter will disrupt the whole election process which makes the scheme impractical in real life.

3 Notations and Assumptions

3.1 Notations

In this paper, we use the following notations.

T: A trusted authority.

V_i: Voter i.

v_i: Vote of the i-th voter.

ID_i: Identification of the i-th voter.

A: Administrator.

$\widehat{C}$: Counter.

C_j: j-th Candidate.

N: The number of candidates, the voters can vote for.

m_A: The number of the registered voters announced by the administrator.

m_{C_j}: The number of votes announced by the j-th candidate.

$m_{\widehat{C}}$: The number of the valid votes received by the counter.

z_i : Pseudo-random number obtained from pseudo-random function and used as pseudo identity of the i-th voter.

$S_i()$: i-th voter's signature scheme.

k_{C_j}: Partial key of the candidate C_j.

K^{-1}: Decryption key of the threshold encryptsystem.

$E_{k_{C_j}}()$: The partial encryption function of the candidate C_j with partial key k_{C_j}.

$z_i' = D_{K^{-1}}(z_i)$: Sealed pseudo-identity of the i-th voter by decryption function of the share cryptogram scheme.

B(v_i, k_i): Bit-commitment scheme for vote v_i using the secret key k_i.

$S_A(e_i)$: Administrator's signature.

$X(x_i, r_i)$: Blinding technique for ballot x_i and random r_i.

$R_A(d_i, r_i)$: Retrieval technique for the blind signature.

3.2 Assumptions

We assume that: (i) all communication between the counter and the voters or between candidates and the voters are via secure anonymous communication channels [16]; (ii) every voter has a digital signature scheme [14]; the administrator has a blind signature scheme [7]; (iii) there exists a secure the bit-commitment scheme [13]; (iv) there exists a secure threshold encryption scheme with N partial encryption functions controlled by the corresponding set of partial keys k_{C_j} $(1 \leq j \leq N)$ held by the n candidates such that for a given message m $E_{k_{C_j}}(m)$ is a share of the cryptogram; the cryptogram $E_K(m)$ can be restored from, t out of N, $E_{k_{C_j}}(m)$ is and be decrypted by $D_{K^{-1}}(E_K(m)) = m$ [17, 18].

Furthermore, we assume that every candidate, the counter and the adminstrator, maintains a public board (a public board can be thought of as restricted shared memories). A public board can be read by all the participants in the voting scheme but can only be written by its owner by appending new messages. Finally , we assume that the majority of candidates, t out of N ($t \geq \frac{N}{2}$), are honest.

4 The Election Scheme

Our election scheme is based on a simple paradigm, which we present first. Then, we give a complete description of the protocol.

4.1 Election Paradigm

The basic election paradigm includes four stages as shown in Figure 1. The participants consist of voters V_i, candidates C_j, an administrator A, and a counter $\widehat{C}$. We also require a trusted party T whose role is limited to the preparation phase of the schemes.

1. Preparation. A trusted party generates N secret keys of the threshold encryption scheme distributes the partial keys to the candidates and a corresponding key K^{1} to the counter.
The voters' pseudo identities decrypted by K^{-1} and sent via an untraceable channel.

2. Registration. It consists of two steps:
   a. Voter V_i creates x_i and makes the message e_i , that hidden ballot, and sends it to A for A's blind signature.
   b. Administrator A signs the message e_i and sends the signature to the voter.

3. Voting.
   a. The signed ballot x_i is sent to all N candidates anonymously by voter Vi.
   b. Each candidate computes his/her partial cryptogram share on the signed ballots and decrypted psuedo identities and then (c_j , w_j) to the counter.
   c. The counter computes all t of N combinations c_j and w_j to get the ballot and z_i . If the majority of the ballot computed is the same and psuedo identity z_i is valid then a roster of valid ballot is made and published.

4 Opening.
   a. Voters send their encryption key to the counter anonymously.
   b. Counter opens votes and counts them and announces the result.

Figure 1: The election Paradigm.

4.2 Structure of Proposed Scheme

The complete voting scheme, based on the paradigm in Figure 1, is as follows.

4.2.1 Preparation Stage

Before the election date, the following steps are completed by the trusted authority T.

1. A list of all eligible voters is prepared such that it is universally accepted by the candidates and voting authorities. The list, say $L1$, is stored in a read-only memory and is available only to the administrator.

2. T generates a pseudo-random number z_i, using a secure pseudo-random generator for, each listed voter and stores it in a scrambled order in a read-only memory accessable only to the counter such that he/she is unable to find any relation between this list and the list of voters, $L1$.

3. T generates N partial keys of the threshold encryption scheme k_{C_j} for the N candidates (C_j $1 \leq j \geq N$). He/she also generates the correct decryption key K^{-1} for the counter and delivers it securely to his/her. The threshold parameter t, candidates' partial secret keys k_{C_j}, and decryption key K^{-1} are kept secret.

4. The pseudo-identities (z_is) are sealed, $z_i' = D_{K^{-1}}(z_i)$, and sent to the voters using an untraceable and secure channel.

4.2.2 Registration Stage

Registration process has two phases.

- *Phase 1 executed by the voter V_i:*

  2a.1 Select vote v_i and compute the ballot using a random key k_i giving x_i=B(v_i, k_i).

  2a.2 Compute the message e_i using the blinding technique $e_i = X(x_i, r_i)$.

  2a.3 Sign e_i, $s_i = S_i(e_i)$, and send the triple $< ID_i, e_i, s_i >$ to the administrator.

- *Phase 2 executed by the administrator A:*

  2b.1 Check the voter V_i's eligibility:

  2b.1a Check voter V_i has the right to vote.

  2b.1b Check through the list of registered voters to make sure that V_i has not already applied for signature.

  2b.1c Check the signature s_i of message e_i is valid.

  2b.2 If all checks of the previous steps are valid then:

  2b.2a Add $< ID_i, e_i, s_i >$ into the administrator's public board.

  2b.2b Sign e_i, $d_i = S_A(e_i)$, and send A's certificate to V_i.

  2b.3 If any check of step 2b.1 failed then reject the transaction.

  2b.4 At the end of voting time, publish the number of the registered voters m_A.

Figure 2 illustrates this stage.

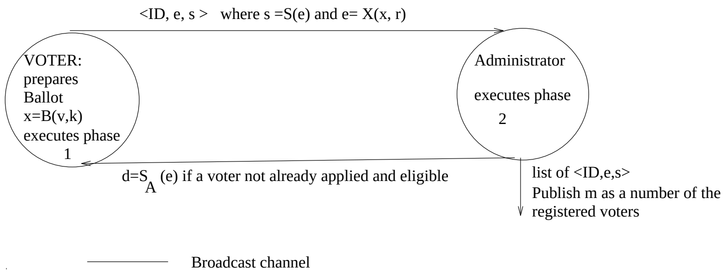

Figure 2: Registration stage.

4.2.3 Voting Stage

Voting process involves three phases. Each candidate and the counter has a public board which is readable by all participants but writable only by the owner:

- *Phase 1 executed by the voter V_i:*

  3a.1 Retrieve administrator's signature y_i on the ballot x_i using $y_i = R_A(d_i, r_i)$.

  3a.2 Check A's signature. If the check succeeds then send $< z_i', x_i, y_i >$ to all the N candidates ($C_j; 1 \leq j \geq N$), through anonymous communication channels. Otherwise, publish $< x_i, y_i >$ as an invalid ballot.

- *Phase 2 executed by the candidate C_j.*

  3b.1 Check the tuple $< x_i, y_i >$ is not received before.

  3b.2 Check the signature y_i of the ballot x_i using A's signature verification algorithm.

  3b.3 If the checks succeed then encrypt $x_i||y_i$ to obtain c_j, $c_j = E_{k_{C_j}}(x_i||y_i)$, encrypt z_i' to obtain w_j, $w_j = E_{k_{C_j}}(z_i')$, and append $< z_i', x_i, y_i >$ to the public board and send $< x_i, y_i > ||c_j||w_j$ to the counter ($||$ denotes concatenation).

  3b.4 If the checks fail then reject transaction and publish $< z_i', x_i, y_i >$.

  3b.5 In the end, publish the number of the valid votes m_{C_j}.

- *Phase 3 executed by the counter $\widehat{C}$.*

  3c.1 Select all $< x_i, y_i > ||c_j||w_j$ with the same prefix $< x_i, y_i >$. If they are less than t, then publish $< x_i, y_i >$ as an invalid ballot; otherwise extract c_j and w_j.

  3c.2 Repeat Steps 3c.2a and 3.c2b until t equal $x_i||y_i$'s and z_i's are found:

  3c.2a Combine t elements out ofthe N extracted c_j's and decrypt with the key K^{-1} to compute $x_i||y_i$,

  3c.2b Create t elements out of the N extracted w_j's to compute the pseudo-identity z_i.

3c.3 Check the signature y_i is valid; the pair $< x_i, y_i >$ is not received before and z_i is valid.

3c.4 If the checks are correct then x_i is a valid ballot; include $< x_i, y_i >$ as the lth row of the counter's public board. Otherwise, reject the transaction and publish $< z_i', x_i, y_i >$.

3c.5 In the end, publish the number of the valid received votes $m_{\widehat{C}}$.

Figure 3 illustrates this stage.

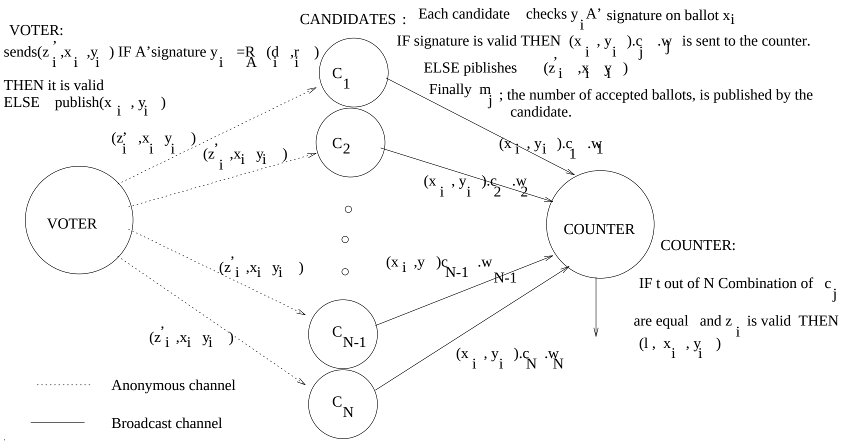

Figure 3: Voting stage

4.2.4 Opening Stage

Opening stage has two phases:

- *Phase 1 executed by the voter V_i:*

  4a.1 Check at least t out of the N of the m_{C_j}s are equal to $m_{\widehat{C}}$ and are less than m_A.

  4a.2 Check the ballot $< x_i, y_i >$ is in the list published by the counter.

  4a.3 If checks succeed then send $< l, k_i >$ to the counter through an anonymous communication channel. Otherwise, open $< x_i, y_i >$ as the valid ballot and its signature.

- *Phase 2 executed by the counter $\widehat{C}$:*

  4b.1 Open the ballot x_i, retrieves the vote v_i; the key k_i is appended to the lth row of the counter's public board allowing everyone to see the vote.

  4b.2 Count votes and announce the result.

Figure 4 illustrates the opening stage.

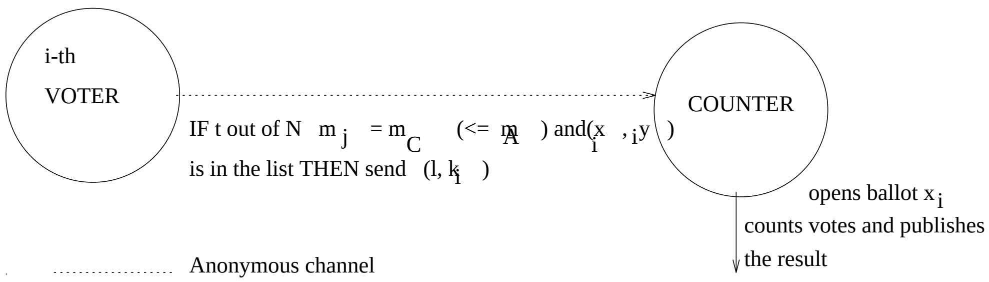

Figure 4: Opening stage

5 Security

In this section, we sketch briefly the proof that the scheme presented in Section 4 has the essential properties of a secure election scheme.

Theorem 5.1 *(Completeness): The ballot of an honest voter is accepted by the honest candidates and the counter with probability one.*

Proof: For a valid vote, the checks performed by the candidates and the counter in Phase 2 and 3 of the voting process ensure that the valid ballot will always be accepted by the honest candidates and the counter and hence the result of the casting is trustable.

Theorem 5.2 *(Privacy): Even if all the participants (the administrator, the counter, and the candidates) conspire, the privacy of the votes is preserved.*

Proof: The relationship between the voters and their ID is hidden by the blind signature scheme and the psuedo-random function used, and the ballot x_i and key k_i are sent through anonymous channel and can not be traced. Furthermore, since if a voter claims the disruption by the administrator or the counter, he can keep his vote v_i secret and show the pair $< x_i, y_i >$ only. Hence, the voter's privacy is ensured.

Theorem 5.3 *(Soundness): If the majority of the candidates are honest, a voter or a conspiring group of the candidates can not disrupt the election.*

Proof: The only ways to disrupt the election is for voters to send invalid votes, and/or for candidates not to pass forward valid vote to the counter or sends duplicate votes. The duplicate ballots can be detected in Phase 2 and 3 in the Voting Stage. Since we assume that the majority of the candidates are honest, therefore in Phase 3 of the Voting Process only valid votes will be accepted.

Theorem 5.4 *(Unreusability): Given that the blind signature scheme and the threshold scheme are secure, no voter can vote more than once without being able to break blind signature scheme.*

Proof: To vote more than once, a voter must have valid tuples of the ballot, the signature, and sealed pseudo-identity. This means that he/she has obtained one signature and sealed pseudo-identity through the proper procedures and created extra pairs himself/herself. Thus, he/she is able to break the blind signature, and/or the threshold encryption scheme. This contradicts the assumptions.

Theorem 5.5 *(Eligibility): Under assumption that the signature schemes and the pseudo-random function are secure, only the eligible voters are able to vote.*

Proof: Assume that a dishonest person can vote. Then, the check is performed by the administrator in Step 2b.1a must be successful. This means that the dishonest person must be able to create a valid pair of the ballot and the signature himself/herself. This clearly contradicts security of the signature schemes used are un-breakable. Furthermore, the check is performed by the counter in Step 3c.4 must be valid. This means that a valid sealed pseudo-random identity is created by the dishonest voter. This is not possible.

Theorem 5.6 *(Verifiability): Assume that the signature schemes and the threshold scheme are secure, the majority of candidates are honest, and the pseudorandom function is secure. Then, the published tally is equal to the actual result of the election.*

Proof: An election may be disrupted by the dishonesty of the candidates, and/or the counter, and/or the administrator.

Disruption by the counter happens when he/she fails to add a valid ballot to the public board. This disruption can be easily detected by showing the validity of $< x_i, y_i >$ by the voteri and the t out of the N candidate's lists prepared in Phase 2 of Voting Stage.

The only possible disruption by the candidate is to send a dummy ballot. To do so, the dishonest candidate must be able to create a valid tuple of ballot, signature, and sealed pseudo-identity by himself. This contradicts our assumption. However if this happens, the dummy ballot will only be counted as a valid ballot if the dummy vote can be computed from the combination of t ballots. This contradicts the assumption that the majority of candidates are honest and threshold scheme is secure. Thus the invalid ballot can be revealed and moreover the dishonest candidate can be identified easily.

The administrator can not also produce dummy vote. To do so, the administrator must be able to create a valid sealed dummy pseudo-identity. This contradicts our assumptions.

Theorem 5.7 *(Fairness): The counting of votes does not affect the voting.*

Proof: Since counting is done after the Voting Stage is completed, so it is impossible that the counting of the votes affect the voting.

6 Conclusion

This paper proposed a secure and practical election scheme for large general voting that provides robustness and verifiability as well as privacy for voters and voting fairness.

References

[1] 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.

[2] 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.

[3] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta, "A Practical Secret Voting Scheme for Large Scale Elections", AUSCRYPT'92.

[4] K. R. Iversen , "A Cryptographic Scheme for Computerized General Elections", in Advances in Cryptology-CRYPTO'91, Lecture Notes in Science 576 Springer-Verlag Berlin pp405-419 (1992).

[5] J. Benaloh, and M. Yung, "Distributing the Power of a Government to Enhance the Privacy of Votes", Proceeding of the 5th ACM Symposium on Principles of Distributed Computing pp.52-62 (Aug.1986).

[6] D. Chaum, "Elections with Unconditionally Secret Ballots and Disruption Equivalent to Breaking RSA", in advances in Cryptology EUROCRYPT'88, Lecture Notes in Computer Science 330, Springer-Verlag, Berlin, pp 177-182 (1988).

[7] D. Chaum, "Security Without identification: Transaction Systems to Make Big Brother Obsolete", Communications of the ACM, Vol. 28, No. 10 pp. 1030-1044. Oct. 1985.

[8] D. Chaum, "The Dinning Cryptographers Problems: Unconditional Sender and Recipient Untraceability", Journal of Cryptology, Vol. 1, No. 1, pp. 65-75, 1988.

[9] Charles P. Pfleeger, "Security In Computing" , 1989.

[10] K. Ohta, "An Electrical Voting Scheme using a single Administrator", (in Japanese) , 1988 Spring National Convention Record, IEICE, A-294(Mar., 1988).

[11] T. Asano, T. Matsumoto, and H. Imai, "A Study on Some Schemes for Fair Election Secret Voting", (in Japanese), The proceeding of the 1991 Symposium on Cryptography and Information Security, SCIS91-12A(Feb., 1991) .

[12] Colin Boyd , "A New Multiple Key Cipher and an Improved Voting Scheme", in Advances in Cryptology-EUROCRYPT'89- Proceeding volume 434 Lecture Notes in Computer Science pp 617-625 Springer-Verlag.

[13] M. Naor, "Bit Commitment Using Pseudo-Randomness", in Advances in Cryptology-CRYPTO'89, Lecture NOTES in Computer Science 435, Springer-Verlag, Berlin, pp128-136 (1990).

[14] W. Diffie, and M. E. Hellman, "New Direction in Cryptography", IEEE Transactions on Information Theory , Vol. IT-22, NO.6, Nov.1976, pp644-654.

[15] J. D. Cohen, and M. J. Fischer, "A Robust and Verifiable Cryptographically Secure Election Scheme", 26th Annual Symposium on Foundations of Computer Science, IEEE, pp372-382 Oct. 1985.

[16] D. Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", Communications of the ACM Vol.24 No.2 pp84-88 Feb. 1981.

[17] A. Shamir, "How to Share a Secret", Communications ACM, Vol. 22, No. 11, pp. 612-613, 1979.

[18] G. R. Blakley, "Safeguarding Cryptographic Keys", in Proceedings NCC, AFIPS Press, Montvale, N. J., Vol. 48, pp. 313-317, 1979.