2006

Electronic voting system for RIT Student Government elections

Sungho Maeung

Master of Science Thesis Proposal

An Internet-Based Voting System for Student Government Elections

Sungho Maeung
<sxm3011@cs.rit.edu>
Computer Science Department
Rochester Institute of Technology

November 14, 2003

Abstract

Recent studies [2, 3] argue that traditional voting systems do not encourage increased voter participation due to constraints in time, location, accuracy, and, accessibility. To ensure the rights of a democratic society and to enhance and secure the voting rights of citizens by surpassing all the limitations of the traditional voting system, the development of an electronic voting system is an attractive solution. Research on secure electronic voting systems has been conducted for at least the past two decades. We propose to develop an electronic voting system, called the Rochester Institute of Technology Student Government Election System (SGEES) based on Damgard et al [1]. This voting scheme will use efficient honest-verifier zero-knowledge, which, unlike previous election schemes [3, 4], are both easy to compute and to verify for both voters and authorities. Our proposed electronic voting system will allow convenient and confident voting while maintaining the accuracy of election results. This project will address the security requirements for electronic voting over the Internet, including privacy, completeness, soundness, receipt-freeness, and universal verifiability. In particular, we will research the feasibility of the voting scheme and protocols by studying three related cryptographical theories: homomorphic encryption, efficient honest-verifier zero-knowledge proofs, and threshold decryption cryptosystem.

Contents

1 Introduction

1.1 Goals and Objectives of Research

The concept of Internet-based voting promises convenience for voters and an inexpensive voting method for a modern democratic society. Researchers have proposed [3, 4] voting schemes that, theoretically speaking, support Internet-based voting. However, due to the difficulty of implementing the necessary security requirements, only a few actual schemes for supporting Internet-based voting have been developed, and these make impractically, strong assumptions about the voting environment, such as anonymous or untappable communication channels [12]. In order to implement an electronic voting system, the following security concerns must be addressed: voter's privacy, anonymity, completeness, soundness, unreusability, eligibility, robustness, universal verifiability, and, receipt-freeness. The most crucial of these security properties to practical voting schemes are universal verifiability and receipt-freeness [12]. One way to satisfy these security requirements is to adopt a set of cryptographic protocols and a bulletin board [1]. A bulletin board allows voters to correspond with trusted authorities via public channels. Previous approaches based on this framework include the homomorphic encryption scheme [4] and the blind signature and mix-net scheme [3]. However, the later two schemes may introduce significant delay and difficulty in implementation.

We propose to study and implement a practical voting system based on homomorphic public-key cryptosystems with threshold decryption using efficient honest-verifier zero-knowledge proofs for correctness of encrypted votes. Furthermore, we state the theoretical soundness of our scheme and propose the implementation of this scheme for the RIT Student Government Election System. In the process of implementation, we will explore how to improve the voting scheme [1] to satisfy all security requirements. We will also address and document the limitations in all components of our system. We expect that this project will lead to a publication and/or presentation at a conference.

2 Cryptographic preliminaries

This section describes of the security requirements of cryptographic primitives for the Internet-based voting system.

2.1 Security requirements

In order to implement the Internet-based voting system, the following security requirements must be addressed.

Eligibility: Only an eligible voter can cast a vote, and each voter can only cast a single vote. The vote of an invalid voter is not counted.

Privacy: Votes remain anonymous; each individual vote is protected against coercion. Different ballots are indistinguishable irrespective of the contained votes.

Completeness: All valid votes must be secret and be counted in the final tally.

Unreusability: A voter cannot vote twice.

Robustness: The voting system must be working properly even though the partial failure of the system occurs.

Universal Verifiability: A third party can check whether or not ballots are correctly cast, and only invalid ballots are ignored.

Receipt-Freeness: A voter cannot prove which candidate he vote. Consequently, voters are not able to sell their votes to the buyers.

2.2 Homomorphic encryption

Homomorphic encryption establishes universal verifiability in the election system by generating a new encrypted vote from the product of two votes. The "product" of the ciphertexts of any two votes is the ciphertext of the "sum" of the votes. The idea behind the homomorphic property is that ciphertext of the sum of the ciphertexts of the votes can be decrypted each encrypted vote. Results of the election can, with the help of the appropriate private key, be computed efficiently [4]. A general definition of the notation is

as follows. For any instance E of the encryption scheme and given messages, m_1 and m_2.

$$E(m_1) \otimes E(m_2) = E(m_1 \oplus m_2) \tag{1}$$

Homomorphic encryption is important to the construction of voting protocols. The ElGamal style cryptosystem can be modified to this homomorphic encryption under the generalized Decision Diffie-Hellman assumption (DDH) and using the standard binomial expansion [1].

2.3 Bulletin board

A bulletin board is a public broadcast channel with memory, which is used our voting system for all communication between voters and authorities. Any information on the bulletin board can be read by any third party and it can be monitored publicly. The bulletin board, however, does not allow any party to erase any information on the board. Only a valid voter can append an encrypted vote in his/her part of the board section. Each column of the bulletin board for a voter consists of four fields: challenge, response, ballot, and a proof as follows.

Challenge field: The verifier posts the challenge value.

Response field: Voter gives the response value to the verifier.

Proof field: The verifier propels the proof of validity for the final ballot.

Ballot field: Voter posts the final ballot.

The bulletin board will execute the interactive proof of validity with voters in the challenge, response, and proof fields during the voting scheme. At the end of the validity proof, the bulletin board posts the proof of validity of the final ballot on the ballot field. In order to gain access to the various fields of the bulletin board, each voter must be identified by a digital signature.

2.4 Zero-Knowledge Proofs

Zero-knowledge proofs based on an interactive proof system allow a prover to convince it has secret knowledge to a verifier without revealing the prover's secret itself. In the voting system, a prover (voter) wants to convince the verifier (tallying) of the correctness of an encrypted vote in such a way that

the verifier cannot learn from the prover's ballot information. In general, zero-knowledge proofs are not suitable for this purpose due to the number of rounds with large computational complexity. Required to reach a conclusion of Damgärd, proposed efficient honest-verifier zero-knowledge proofs of knowledge having a constant number of rounds. His results can also be applied to non-interactive proofs such as those having a cryptographic hash function h based on Fiat-Shamir heuristic $\sum$-protocols [1]. Zero-knowledge proofs can also show the validity of votes encrypted via a homomorphic encryption scheme.

2.5 Threshold decryptions system

In a (t, n) threshold protocol, a trusted party selects a secret key, S, and distributes shares of S to n members. Any group of t members which pool their shares should then be able to recover the secret S. A (t,n) threshold scheme based on Lagrange interpolation was developed by Shamir [13]. If the members are less than t of members, then they cannot receive any information about the secret S. The main components of a threshold decryption model are as follows.

i. A key generation protocol by a trusted third authority. The trusted third authority distributes the private/public keys voters and tallying servers.

ii. A decryption protocol by threshold ElGamal cryptosystem. To decrypt a ciphertext c a main tallying server forwards the ciphertext to the tallying servers. Using their shared secret keys, each tallying server runs the decryption algorithm and outputs a part of decryption c with the validity proofs. Finally, the main tallying server uses the combining algorithm to decrypt the ciphertext if the partial decryptions are valid.

3 System Architecture

The Internet-based voting system is divided into three main components: registration, validation, and tallying. Distributed servers will support the voting scheme for the RIT SGEES as shown in Figure 1. Our system will have the following features.

1. A voter's client application provides a user-friendly interface.
2. Trusted authorities will allow the voters to be authenticated via their votes, using zero-knowledge proofs.
3. The votes will be securely transmitted from the voters to the tallying servers via the bulletin board without revealing any voter's private information.
4. The tallying servers will count the ballots securely and will print the results of the election on the bulletin board.
5. Client will be executed on the PC of the Internet enabling the communication with the authority, to cast the vote and verify the final tally.
6. Distributed servers will verify the voter's vote, collect, and count them correctly.

Both Java and Relational Database Management System (RDBMS) will be used for carrying out the development and storing encrypted ballots and user's information. Our project will be portable to multiple platforms.

3.1 Registration

Before the election (as shown in Figure 2), a voter must prove his identity and eligibility. We will use the ElGamal cryptosystem, which is based on Decision Diffie-Hellman Assumptions along with the Homomorphic Encryption Property suggested by Damgärd et al [1].

The scenario of the registration is as follows: each voter can encrypt a ballot with the public key of a trusted tallying server. A trusted admin server distributes the public and private keys. Before the voter submits an

encrypted ballot to the bulletin board, the trusted registration server will authenticate the voter's eligibility and ensure the uniqueness of the vote. In the process, the homomorphic encryption model will be used.

3.2 Validation

After registration (as shown in Figure 3), the validation process on the bulletin board ensures that each encrypted vote is valid. The correctness of the encrypted vote will be validated using honest-verifier zero-knowledge proofs [3, 4, 5, 9, 10, 11] that reveal nothing about the secret information of the voter. We will address the following properties of honest-verifier zero-knowledge proofs.

- **Completeness:** The Verifier will accept the proof with very high probability if the Prover knows secret information, called a witness, from which a proof can be built.
- **Special Soundness:** If the Prover does not know any witness, and performs any probabilistic algorithm. The Verifier will reject a proof attempt with a very high probability.
- **Honest-verifier zero-knowledge:** It is possible to generate a transcript, indistinguishable from a valid proof protocol, without interacting with the Prover. It ensures that an honest verifier does not gain any knowledge about the witness that is available to the Prover.

3.3 Tallying

At the end of the voting period (as shown in Figure 4), all votes are counted correctly and securely by decrypting the final encrypted ballot. This can be done by secret-sharing scheme among a set of authorities. In a (t,n) threshold cryptosystem [5, 6, 7, 8], it is required that a private key is shared among n tallying servers, and the decryption is possible only when at least t tallying servers cooperate with each other. The idea behind a threshold cryptosystem is to keep the private key with a fault-tolerant technique and to distribute the functionality of cryptographic protocols to establish robustness. The tallying process can be shared among n tallying servers by using a (t,n) threshold public key decryption system. Each one of the n tallying servers has a share of the private key. And each voter encrypts his vote with the

public-key of the tallying servers. The final tallying server can decrypt the encrypted vote by using t tallying servers which are cooperating with each other. It provides privacy of the votes and accuracy of the tally if at least t tallying servers are provided.

4 Deliverables

4.1 Source code with javadoc documentation

1. All cryptographic protocols in the election system
2. Registration mechanism
3. Bulletin board mechanism
4. Tallying servers mechanism
5. Trusted servers mechanism

4.2 Documentation and analysis of security considerations

1. Complete technical specification and documentation of the design and implementation of the election system and cryptographic algorithms, including justification for choices made.
2. Analysis of security considerations such as homomorphic encryption, honest-verifier zero-knowledge proofs, and threshold cryptosystem.
3. Formal conclusions regarding scalability and security of the system, with justification for all conclusions and results collected.
4. Working demonstration of the voting system and performance analysis for computational and communication complexities.
5. Five tallying servers, one registration and authentication server, and one bulletin board server at workstation level will be used in the Computer Science Department labs.

4.3 Experiments & Testing

The experiments and testing of the electronic voting system will consist of three parts: registration, bulletin board, and tallying. Each part will be included in the evaluation of the cryptographic protocols corresponding to the activity tested as follows.

1. Registration
   The registration's tests will include evaluations to verify eligible voters, store the voter's information, and interact with the trusted admin server to gain public keys for the voters.

2. Bulletin board
   The bulletin board's tests will include the activity evaluations: proving the validity of encrypted votes, processing zero-knowledge properties, and interacting with trusted admin server to verify there were no double votes under zero-knowledge proofs.

3. Tallying servers
   The tallying's tests will comprise the activity evaluations: homomorphic encryption, shares of private key among the tallying servers, and decryptions of the final encrypted votes.

4.4 Draft thesis contents

I. Abstract

II. Introduction

III. Related works

IV. Cryptographic primitives

V. Design specification

VI. Implementation

VII. Results

VIII. Result analysis

IX. Conclusions

X. Appendix

XI. References

4.5 Proposed Schedule

Winter 2003, Research	
Activity Description	Estimated Dates
Analysis of existing electronic voting systems	Dec 1 - Dec 20
Feasibility of known theoretical voting schemes	Dec 21 - Jan 22
Research cryptographic protocols and schemes	Jan 23 - Feb 22
Spring 2004 , Implementation and experiments	
Implementation of the proposed election system	Feb 23 - April 10
experiments and testing	April 11 - May 5
Summer 2004, writing thesis	
Writing thesis, final experiments	May 6 - July 10
Thesis defense	By the end of July

References

[1] I. Damgard, J. Groth, and G. Salomonsen. *The theory and implementation of an electronic voting system.* In D. Gritzalis, editor, Secure Electronic Voting. Kluwer Academic Publishers, 2002.

[2] CALTECH-MIT Voting Technology Project : Retrieved on October 27, 2003 from http://www.vote.caltech.edu/.

[3] A. Fujioka, T. Okamoto and K. Otha. *A practival secret voting scheme for large scale elections.* Advances in Cryptology - AUSCRYPT '92, Springer Verlag LNCS series, pp. 244-251.

[4] R. Cramer, R. Gennaro, B. Schoenmakers: *A Secure and Optimally Efficient MultiAuthority Election Scheme.* Proceedings of EUROCRYPT '97, Springer Verlag LNCS series, pp. 103-118.

[5] A. Menezes, P. Oorschot and S. Vanstone, CRC *Handbook of Applied Cryptography.* CRC Press, 1996.

[6] Y. Desmedt, Y. Frankel. *Threshold cryptosystem.* Advances in Cryptology - Crypto '89, Springer Verlag LNCS series, pp. 307-315.

[7] T. Pedersen. *A Threshold Cryptosystem without a Trusted Party.* In D.Davies editor, Advances in Cryptology - EUROCrYPT'91, Springer Verlag LNCS series, 1991.

[8] I. Ingemarsson, G. J. Simmons. *A protocol to set up shared secret schemes without the assistance of a mutually trusted party.* Advances in Cryptology- EUROCRYPT'90, Springer Verlag LNCS series, pp. 266-283.

[9] D. Stinson. *Cryptography: Theory and Practice - 1st edition.* CRC Press, 1995.

[10] O. Goldreich. *Foundation of Cryptography - Basic Tool.* Cambridge University Press, 2001.

[11] M. Wenbo. *Modern Cryptography : Theory and Practice.* Prentice-Hall, Inc, 2004.

[12] M. Hirt and K. Sako. *Efficient receipt-free voting based on homomorphic encryption*. Advances in Cryptology - EUROCRYPT'00, Springer Verlag LNCS series, 2000.

[13] A. Fiat and A. Shamir. *How to prove yourself: Practical solutions to identification and signature problems*. In Advances in Cryptology - CRYPTO'86, Springer Verlag LNCS series, pp. 186-194.