An Anonymous Electronic Voting Protocol for Voting Over The Internet[*]

Indrajit Ray[†] Indrakshi Ray[†] Natarajan Narasimhamurthi[‡]
[†]Department of Computer and Information Science
[‡]Department of Electrical and Computer Engineering
University of Michigan-Dearborn
4901 Evergreen Road, Dearborn, MI 48128

Abstract

In this work we propose a secure electronic voting protocol that is suitable for large scale voting over the Internet. The protocol allows a voter to cast his or her ballot anonymously, by exchanging untraceable yet authentic messages. The protocol ensures that (i) only eligible voters are able to cast votes, (ii) a voter is able to cast only one vote, (iii) a voter is able to verify that his or her vote is counted in the final tally, (iv) nobody, other than the voter, is able to link a cast vote with a voter, and (v) if a voter decides not to cast a vote, nobody is able to cast a fraudulent vote in place of the voter.

1 Introduction

Secure electronic voting requires the exchange of untraceable yet authentic messages. Broadly two different approaches have been proposed: (i) approaches that require complex encryption schemes [1, 6, 7, 10], and (ii) approaches that require an anonymous channel [2, 5, 8, 9, 11, 12, 13, 14] that is used to cast the ballot as an untraceable message. The protocol we propose does not require any complex cryptographic schemes. Our protocol is similar to the ones in [8, 9] but does not need an anonymous channel. Voting is similar to a guest ftp session. The session may, at best, be traced back to an IP address but not to a voter.

Researchers have identified a set of requirements for a secure electronic voting protocol [8]:

1. **Accuracy**: (i) A cast vote can not be altered. (ii) An invalid vote is not counted. (iii) Each voter has the guarantee that his/her ballot is counted.

2. **Democracy**: (i) Only a eligible voter participate. (ii) Each voter can cast only one vote.

3. **Privacy**: A ballot can not be linked back to the voter who cast it.

4. **Verifiability**: Each voter can verify that his/her vote is counted.

5. We identify an additional property which we call **No Unauthorized Proxy**: If a voter decides not to cast his/her ballot, no party can take advantage of this and cast a forged ballot.

The protocol we describe next satisfies all of the above properties.

2 The Protocol

We assume that the following are available:

1. Hard-to-invert permutations: A permutation of a finite set of numbers whose inverse is hard to compute.

2. (Blind) Signature on messages: A verifyable transformation of a message which can only be generated by the signing entity. Using publicly available information, any one can verify the signature. In a blind signature, the signing entity signs a message without knowing its contents [3, 4]. The message that is submitted for blind signature can be freely published without revealing the actual message.

3. Secure Transit: An encryption scheme that ensures privacy and integrity of messages in transit.

2.1 Protocol description

The voting protocol employs three, not necessarily trusted, agents for successful operation:

1. *BD*: A ballot distributor who prepares blank ballots and distributes one to each voter.

[*]This work has been partially supported by the National Science Foundation under grant EIA 9977548

2. *CA*: A certifying authority who verifies eligibility, certifies ballots and ensures that a voter gets only one certified ballot.

3. *VC*: A vote compiler who tallies the votes and announces the results.

The agents may collude with each other or with a voter to perpetrate fraud. If a fraud is suspected, then the protocol ensures that the fraud can be proved. When an agent colludes with a voter, it will only affect that voters ballot. Before the voting process, a voter registers with some voter registration authority. This authority prepares a list of registered voters and issues a certificate for each registered voter that contains the voter's identity and public key.

For this discussion we assume that for any party, X, X_e represents the party's encryption key and X_d, the decryption (signing) key. The voting proceeds as follows:

1. *Blank ballot distribution*: *BD* ⟶ *V*: [{y, [h(y), BD_d]}, V_e], [h(voter certificate),BD_d]. When a voter electronically authenticates himself, *BD* provides a signed blank ballot and a signed digest of the voter certificate. The blank ballot is a message of two fields (i) the ballot serial number field, *y* and (ii) *BD* signed digest of the ballot serial number, [h(y),BD_d].

2. *Generating a voter mark*: The Voter performs a one-way permutation of the serial number to generate a unique voter mark *m*.

3. *Voter certification*:

   (a) *V* ⟶ *CA*: [{m×[r,CA_e], [h(m×[r,CA_e]), V_d], V}, CA_e], [{*V*, voter-certificate, [h(voter-certificate), BD_d]}, CA_e], m is the voter mark generated by the voter.

   (b) *CA* ⟶ *V*: [{[{m×[r, CA_e]}, CA_d]}, V_e]

   The voter gets a blind signature on the voter mark. The blinded voter mark has the voter's signature on it, which authenticates the voter at the certifying authority.

4. *Vote casting*:

   (a) *V* ⟶ Public FTP site: [{{vote, [m, CA_d]}, h(vote, [m, CA_d])}, VC_e]

   (b) Public FTP site ⟶ *VC*: [{{vote, [m, CA_d]}, h(vote,[m, CA_d])}, VC_e]

   (c) *VC* ⟶ Public FTP site: [h(vote, [m,CA_d]), VC_d]

   (d) Public FTP site ⟶ *V*: [h(vote, [m, CA_d]), VC_d]

   When the voter receives the signed salted voter mark, he removes all identifying marks, creates a filled ballot (vote – a message of fixed length and pre-determined format) and transmits the *CA*-signed voter mark and vote to *VC* using a protocol similar to anonymous ftp from a voting kiosk. The *CA*-signed voter mark together with the corresponding filled ballot is hence forth called the marked-ballot.

5. *Vote counting*: Once the voting period is over, *VC* publishes in a public place all the cast ballots and announces the results. In addition, *CA* publishes in a public place all the salted voter marks that were sent to it for signature, and *BD* publishes the number of blank ballots distributed and their serial numbers.

2.2 Properties

First suppose that no fraud has been perpetrated. Since all the ballots are published, and each ballot has the voter mark *m*, any voter can verify that his ballot has been counted. The voter mark being a one-way permutation of the serial number, it is not possible to trace the voter back via the voter mark. In addition, presence of *CA*'s signature on the voter mark ensures that each voter gets to cast only one vote. Also, uniqueness of *m* ensures that a voter can cast only once. Finally, for every ballot that is cast, *CA* should have a salted voter mark signed by a registered voter. This prevents unauthorized proxies. Based on these properties, it is possible to show that the protocol satisfies the requirement set forth earlier.

Next consider the case when a fraud has been perpetrated. In the absence of fraud, the following will be true:

1. Every ballot that is published by *VC* will have *CA*'s signature.

2. For every marked-ballot that is published by *VC*, there should be a corresponding unsigned salted voter mark that was submitted to *CA* for signature. Each of those must have the signature of an eligible voter and the signature of *BD*. Also, voter marks submitted for blind signature are publicly available.

Since marked-ballots that *VC* publishes must contain *CA*'s signature, *CA* must be involved in any fraud. *CA* and *VC* by colluding can produce a marked-ballot with the necessary signatures. However, there will not be a corresponding salted voter mark that has been submitted for *CA*'s signature. Similarly *BD* and *CA* can collude to cast a fraudulent ballot with the necessary signatures. However, in both these two cases, when the final ballot is published, the number of ballots that are counted would exceed the number of voter marks that were submitted to *CA* for signature by valid voters. Thus, in these cases, where only two of the three entities collude, fraud is easily detected. This observation is equally valid even if all the three agents collude. Thus, in order to perpetrate fraud, *CA* must generate spurious signed marked-ballots, and *VC* must substitute valid ballots with spurious ones. Detection of such a fraud requires the active participation of the voter. A voter, who does not see his ballot in the final tally will detect the fraud and prove it by submitting, anonymously, a copy of his ballot signed by *CA*. A corresponding ballot will not be found among the published ballots.

3 Conclusion

In this work we present a secure electronic voting protocol that is suitable for large scale voting over the Internet. The protocol satisfies the core properties of secure voting systems – namely accuracy, democracy, privacy and verifiability. Further the protocol ensures that if an eligible voter decides not to cast a vote (not the same as a voter choosing to abstain), nobody is able to cast a fraudulent vote in place of the voter – a property we call *no unauthorized proxy*. We are aware of three shortcomings: Since the voter can identify his ballot, we can not prevent vote buying. Secondly, if several voters, after obtaining *CA*'s signature, decide not to cast their ballot, then the three agents can cast fraudulent ballots. This fraud cannot be detected if the number of fraudulent ballots is less than the number of signed ballots that were not cast. If such a fraud is detected, then proving it will require the cooperation of the voters who did not cast their signed ballots. Finally, we allow a cast ballot to be traced back to an IP address (not to a voter). By using public voting kiosk we can avoid the IP address to be linked to a voter. We are currently looking into methods to address these issues.

References

[1] J. Benaloh and M. Young. Distributing the Power of a Government to Enhance the Privacy of Voters. In *Proc. of the 5th ACM Symposium on Principles of Distributed Computing*, pages 52–62, August 1986.

[2] C. Boyd. A New Multiple Key Cipher and an Improved Voting Scheme. In *Advances in Cryptology – EUROCRYPT '89*, volume 434 of *Lecture Notes in Computer Science*, pages 617–625. Springer-Verlag, Berlin, 1989.

[3] D. Chaum. Blind Signatures for Untraceable Payments. In *Advances in Cryptology – CRYPTO '82*, pages 199–203. Springer-Verlag, Berlin, 1983.

[4] D. Chaum. Security Without Identification: Transaction Systems to Make Big Brother Obsolete. *Communications of the ACM*, 28(10):1030–1044, October 1985.

[5] D. Chaum. Elections with Unconditionally Secret Ballots and Disruption Equivalent to Breaking RSA. In *Advances in Cryptology – EUROCRYPT '88*, volume 330 of *Lecture Notes in Computer Science*, pages 177–182. Springer-Verlag, Berlin, 1988.

[6] R. Cramer, M. Franklin, B. Schoenmakers, and M. Young. Multi-Authority Secret-Ballot Elections with Linear Work. In *Advances in Cryptology – EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 72–83. Springer-Verlag, Berlin, 1996.

[7] R. Cramer, R. Gennaro, and B. Schoenmakers. A Secure and Optimally Efficient Multi-Authority Election Scheme. In *Advances in Cryptology – EUROCRYPT '97*, volume 1233 of *Lecture Notes in Computer Science*, pages 103–118. Springer-Verlag, Berlin, 1997.

[8] L. F. Cranor and R. K. Cytron. Design and Implementation of a Practical Security-Conscious Electronic Polling System. Technical Report WUCS-96-02, Dept. Of Computer Science, Washington University, January 1996.

[9] A. Fujioka, T. Okamoto, and K. Ohta. A Practical Secret Voting Scheme for Large Scale Elections. In *Advances in Cryptology – AUSCRYPT '92*, volume 718 of *Lecture Notes in Computer Science*, pages 244–251. Springer-Verlag, Berlin, 1993.

[10] K. R. Iversen. A Cryptographic Scheme for Computerized General Elections. In *Advances in Cryptology – CRYPTO '91*, volume 576 of *Lecture Notes in Computer Science*, pages 405–419. Springer-Verlag, Berlin, 1992.

[11] W. S. Juang and C. L. Lei. A Secure and Practical Electronic Voting Scheme for Real World Environments. *IEICE Transaction on Fundamentals of Electronics, Communications and Computer Science*, E80-A(1):64–71, January 1997.

[12] T. Okamoto. An Electronic Voting Scheme. In *Proc. of the IFIP Workshop on Advanced IT Tools*, pages 21–30. Chapman & Hall, 1996.

[13] T. Okamoto. Receipt-Free Electronic Voting Schemes for Large Scale Elections. In *Proc. of Security Protocol Workshop – Paris*, pages 25–35, 1997.

[14] C. Park, K. Itoh, and K. Kurosawa. Efficient Anonymous Channel and All/Nothing Election Scheme. In *Advances in Cryptology – EUROCRYPT '93*, volume 765 of *Lecture Notes in Computer Science*, pages 248–259. Springer-Verlag, Berlin, 1993.