Anonymous Electronic Voting System with Non-Transferable Voting Passes

ROSANNA Y. CHAN, JONATHAN C. WONG, ALEX C. CHAN
Department of Information Engineering, The Chinese University of Hong Kong

Key words: Electronic voting, Receipt-free, security protocol, cryptography, voting pass, non-transferable

Abstract: This paper describes a new receipt-free electronic voting system for an open network. In our system, the voters vote with anonymous voting passes, which is motivated by the E-Cash protocol [1]. We have modified the E-Cash protocol such that the pass is non-transferable. A modified commitment scheme is also employed so that the administrator cannot change the vote even if it knows the content of the ballot. Our system is designed for realising the digitalization of large-scale elections conducted by the government. Various aspects involved in an election are considered during protocol design so that the potential cheats of either the voters or the administrators are prevented.

1. INTRODUCTION

Since the first proposal of the cryptographic voting protocol in 1981 [2], a number of researches have been done on the electronic voting systems. Like many digital services, electronic voting brings much convenience to the society and lowers the cost of traditional elections. However, the systems must be supported by very strong security services in order to be fair and accepted by the general public. Efficiency is also an important factor for a practical system.

The definition of a secure voting system is given in [3], in which seven requirements namely completeness, soundness, privacy, unreusability, eligibility, fairness and verifiability are listed. However most of the early electronic voting schemes are not receipt-free. The receipt generated by the election administrator can be used as an evidence of having made a particular vote and therefore enables one to sell his vote to the others. This problem induced the concept of receipt-freeness in an electronic voting system. Such a concept, which is firstly proposed in [4], is

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35515-3_53

S. Qing et al. (eds.), *Information Security for Global Information Infrastructures*

regarded as a milestone in electronic voting protocol design. A number of receipt-free electronic voting schemes have been proposed in which the receipts were eliminated by different means. In [4] and [5], physical assumptions are made so that receipt-freeness is achieved. While in [6], [7] [8] and [9], the authentication and voting processes are separated. A piece of authentication document is generated with which the voter can vote anonymously. However, we find out that in these systems, the voting right can be transferred because the authentication document is not related to any secret information of the voters. This is a severe threat to the fairness of a voting system.

In this paper, we propose a new electronic voting scheme in which non-transferable voting passes are issued to the legitimate voters. Only the voting pass holder can vote. If the voters sell their voting passes, they suffer the disclosure of their private keys. This deters the transaction of voting right.

Our system does not rely on special physical assumptions nor network configuration because the security is supported by strong cryptographic algorithms. A modified commitment scheme is employed so that the voters cannot deny making a particular vote; the administrator cannot change the votes either, even if it can read the contents of the ballot. Unlike the scheme proposed in [3], the voters in our system are not required to submit a vote-opening key after the polling period. However, they can check their voting record at a Trusted Third Party (TTP).

The organization of the rest of this paper is as follows. In section 2, we describe the overall model of our system. Two innovative cryptographic protocols, the anonymous authentication protocol as well as the anonymous commitment protocol are introduced in section 3. While the entire electronic voting protocol is described in section 4. Section 5 gives the security analysis of our system, followed by the conclusion in section 6.

2. THE PROPOSED ELECTRONIC VOTING SYSTEM

2.1 The Proposed Election Model

Our system is designed for realizing the digitalization of large-scale elections conducted by a government. Arbitrary voting schemes are allowed. It is assumed that the community is supported by a Public Key Infrastructure (PKI), where each voter possesses a public-private key pair and one can generate a liable digital signature with his own private key. The parties involved include the Voters, the Registrar, Administrator, Collector and a trusted third party.

Voters. They refer to the group of people who have the right to vote in a particular election.

Registrar. It is the server responsible for voter verification and anonymity service provision.

Administrator. It is the server responsible for conducting the polling process and announcing the results.

Collector. It is the server responsible for collecting the ballots. It is the digital analogy of the vote-collecting box.

Trust Third Party (TTP). This is an independent party who is responsible for handling disputes. The court is an example.

All of the Registrar, Administrator and the Collector are servers reside in the government or any parties that is responsible for holding the election. The Registrar and the Administrator may not necessarily be two separated servers. There is also a public counter, which is universally readable. The number shown represents the number of ballots accepted and its initial value is set to zero.

There are three phases in the election. The Registration Phase, the Polling Phase and also the Vote Opening Phase.

The Registration Phase. In this phase, the valid voters registrar for the election and obtain an anonymous voting pass from the Registrar.

The Polling Phase. During this phase the voters vote with their anonymous passes obtained in phase one. Each voter can only vote once. This phase can be divided into two sub-phases. Namely the voter authentication phase and the poll-casting phase. The polling phase can also be regarded as the polling period. The polling phase and the registration phase can run simultaneously. After the Administrator accepted a valid vote from the voter, the public counter will be increased by one.

The Vote Opening Phase. This phase begins after the polling phase. The result is computed and published.

In our system, the voters can vote in any locations with any devices that can be connected to the network. However, they are required to vote in a voting station for absolute uncoercibility.

3. TWO CRYPTOGRAPHIC PROTOCOLS

This section explains two cryptographic protocols specially designed for our voting scheme. In section 3.1 we describe an authentication protocol which is innovated by the E-Cash protocol [1]. In section 3.2 we describe a modified commitment protocol.

3.1 Protocol One – The Anonymous Authentication Protocol

The anonymous authentication protocol is designed for supporting the non-transferable anonymous voting pass of our system. We have modified *the check withdrawal transaction protocol* in [1] so that the following desirable properties can be achieved.

1. Alice can prove the knowledge of her secret key anonymously.
2. Anyone who passes the random challenge must possess the knowledge of Alice's secret key. This is to ensure that Alice cannot transfer the pass to other parties or she risks the disclosure of her private key.
3. Unlike the original protocol in [1], the secret information is not released to the Administrator during the registration phase.

We are going to describe how Alice obtains a non-transferable pass from Bob. Let Alice's private key be u and the corresponding public key be v, where $u \cdot v \equiv 1$ $(mod\ \Phi(q))$ and q is the published modulus [10]. We also define the operation $h(x)$ as

$$h(x) = \alpha^x \ (\text{mod } q)$$

where α is a published element in GF(q). $h(x)$ is a one way function. It is computationally difficult to calculate x from $h(x)$ [11].

The description of Protocol One is as follows:

1. Alice first generates k candidates (called T_i) in the following way. She generates the random integers a_i, b_i, c_i and r_i for $i = 1$ to k. Then she computes:
   $$x_i = g(a_i{+}u,\ b_i)$$
   $$y_i = g(-a_i,\ c_i)$$
   $$and \qquad T_i = r_i^e{\cdot}f(x_i,\ y_i)\ (mod\ n)$$
   where (e, n) is the public key of Bob. f, g are some two-argument collision-free functions. a_i+ $(-a_i) \equiv 0$ $(mod\ \Phi(q))$, where $(v,\ q)$ is the public key of Alice.
2. Alice presents the k T_i to Bob.
3. Bob randomly splits the integers 1, …, k into two ordered sets, S and S', each consists of k/2 elements. He then sends S to Alice.
4. Alice sends the values of $h(-a_i)$ and r_i for all $i \in S$ to Bob.
5. Bob generates a k/2-bit binary random challenge vector $z = (z_0, z_1, \ldots, z_{k/2})$ and sends it to Alice. The i^{th} bit in z corresponds to the i^{th} element in S.
6. Alice responds according to the following rule:
   when $z_i = 0$, Alice sends Bob a_j+u, b_j and y_j;
   when $z_i = 1$, Alice sends Bob $-a_j$, c_j and x_j.
   where j is the i^{th} element in S.
7. Bob verifies accordingly:
   when $z_i = 0$, Bob checks if $(h(a_j{+}u){\cdot}h(-a_j))^v \equiv \alpha\ (mod\ q)$. He then checks if the corresponding T_i are valid;
   when $z_i = 1$, Bob checks if the $h(-a_i)$ presented by Alice previously can be derived from -a_i. He thén checks if the corresponding T_i are valid.
8. If T_i is valid for all $i \in S$, Bob signs on the other k/2 unopened T_i with his private key d and returns to Alice:
   $$\prod_{i\in S'} T_i^d \ (\text{mod } n)$$
9. Alice removes the blinding factors and form:

$$\prod_{i \in S'} f(x_i, y_i)^d (\bmod n)$$

which is Bob's signature on the pass

$$\Psi = \prod_{i \in S'} f(x_i, y_i) (\bmod n)$$

Notice that steps (4) to (7) are necessary in order to ensure that Alice's private key, u, is included in the pass Ψ.

When Alice shows her pass to Bob, the following steps will be taken:

10. Alice presents the anonymous pass Ψ to Bob.
11. Bob checks if the signature on Ψ is valid.
12. Bob generates a $k/2$-bit binary random challenge vector $z = (z_0, z_1, \ldots, z_{k/2})$ and sends it to Alice. The i^{th} bit in z corresponds to the i^{th} element in S'.
13. Alice responds according to the following rule:
    when $z_i = 0$, Alice sends Bob a_j+u, b_j and y_j;
    when $z_i = 1$, Alice sends Bob $-a_j$, c_j and x_j.
    where j is the i^{th} element in S'.
14. Bob checks if Ψ can be derived from the partial openings. If so, authentication will be granted.

Notice that Alice remains anonymous to Bob because both Ψ and the partial openings do not contain plain information on Alice's identity. Furthermore, one must know the values of a_i+u, $-a_i$, b_i, and c_i for all i in S' in order to succeed in the challenge on Ψ. This means that the proving party is able to access Alice's private key u since

$$a_i+u+(-a_i) \equiv u \ (mod\ \Phi(q))$$

for any i in S'.

Therefore if Alice sells her pass to a third party, she is releasing her private key u as well.

3.2 Protocol Two – Anonymous Commitment

In our system, the voter signs and commits on the vote anonymously. Neither the voter can deny a committed vote nor the administrator can change an accepted vote. This is achieved by the anonymous commitment protocol. The identity of a voter is only revealed to the TTP when disputes occur.

We are going to illustrate how Alice commits on a message m using her private-public key pair (u, v) anonymously.

Let $H(x)$ be a multiplicative homomorphic one-way hash function as described in [12], and we define the two-argument hash function $H(x, y)$ as $H(x{\cdot}y)$. Due the multiplicative homomorphic property,

$$H(x, y) = H(x{\cdot}y) = H(x){\cdot}H(y).$$

To commit on m, Alice submit the followings to Bob:

$$(m||H(m)||H(Sign_u(m))||H(m, Sign_u(m))$$

where $Sign_u(m)$ denotes Alice's signature on m using her private key u. Bob accepts the commitment if $H(m){\cdot}H(Sign_u(m)) = H(m, Sign_u(m))$. Notice that Alice remains anonymous.

We give a brief security analysis of protocol two here.

1. Alice cannot deny the committed message m because Bob cannot generate $H(Sign_u(m))$.
2. Bob cannot change m because he cannot generate a valid $H(Sign_u(m'))$ on m'.
3. Alice cannot cheat by submitting the followings
   $$(m||H(m)||H(Sign_u(m'))||H(m, Sign_u(m'))$$
   to Bob but claiming m is modified from m' by Bob. This is because the TTP will ask Alice to sign on m', which she claims the original message is. It is easy to find out that $H(Sign_u(m'))\cdot H(m') \neq H(m, Sign_u(m'))$.

Both Protocol One and Protocol Two will be employed in our voting protocol. This will be introduced in the next section.

4. THE ELECTRONIC VOTING PROTOCOL

The main innovation in our protocol is the employment of a non-transferable anonymous voting pass, which is motivated by the E-Cash scheme [1]. In addition, the modified commitment scheme eliminates the submission of a vote-opening key by the voters. In this section we describe the proposed E-Voting protocol phase by phase.

4.1 The Registration Phase

Suppose Alice is a valid voter in the election. Before she polls, she needs to register for the election and obtain an anonymous voting pass from the Registrar. The Registrar holds a list of names of the valid voters. Step (1) to (9) of Protocol One is run, with Bob being replaced by the Registrar. The resulting pass Ψ is the voting pass.

4.1.1 The Polling Phase

Two successive sub-phases are involved in the polling phase. Namely the voter authentication and vote-casting processes. The Administrator is responsible for the execution of this phase.

4.1.2 Voter Authentication

The authentication process is done by step (10) to (14) in Protocol One. With Bob being replaced by the Administrator. Authentication is granted upon the successful verification on an unused Ψ.

4.1.3 Poll-Casting

The poll-casting phase is run immediately after the voter authentication phase. Protocol Two is run, with Bob being replaced by the Administrator. Also, the message *m* is replaced by the ballot β. Alice submits the following encrypted token

$$Encrypt_c(\beta||H(\beta)||H(Sign_u(\beta))||H(\beta, Sign_u(\beta))$$

to the Administrator, where $Sign_u(\beta)$ denotes the signature of β with u, the private key of Alice. The entire token is encrypted with c, the asymmetric encryption key of the Collector. The Collector is the digital analogy of submitting the ballot into the voting box.

After accepting the ballot from the valid voter, the public counter will be increased by one. A reference number is also recorded and saved at the Administrator.

The voter authentication sub-phases and the vote-casting sub-phase are two non-separable processes. If interruption occurs, the first sub-phase should be run again. This is necessary in order to ensure that the ballot is submitted by the valid pass holder.

4.2 Vote-Opening Phase

After the polling period, the Collector sends the decryption key c' to the Administrator. The Administrator decrypts the tokens collected in the vote-casting phase, it also verifies and counts the votes.

The final result is calculated. A result list, which is shown below is published.

Table 1. List of ballots published after the election

	Received Ballots
1	$(\beta_1\|\|H(\beta_1)\|\|H(Sign_{u1}(\beta_1))\|\|H(\beta_1, Sign_{u1}(\beta_1))$
2	$(\beta_2\|\|H(\beta_2)\|\|H(Sign_{u2}(\beta_2))\|\|H(\beta_2, Sign_{u2}(\beta_2))$
3	$(\beta_3\|\|H(\beta_3)\|\|H(Sign_{u3}(\beta_3))\|\|H(\beta_3, Sign_{u3}(\beta_3))$
.	.
.	.
n	$(\beta_n\|\|H(\beta_n)\|\|H(Sign_{un}(\beta_n))\|\|H(\beta_n, Sign_{un}(\beta_n))$

Given *n* is the total number of ballots collected, the first column shows the reference number of a particular ballot; and the second column lists the content of the ballot in concern. Notice that *n* should equal to the final reading on the counter. It is easy to check if the Administrator has varied any ballots. A particular voter can also check if her vote has been counted correctly. This will be discussed in next section.

5. SECURITY ANALYSIS

Our electronic voting protocol satisfies the basic security requirements listed in some of the previous electronic-voting papers, such as those listed in [3] and [13]. In additional, our protocol is a receipt-free scheme, in which the voters cannot sell the votes to the others using the receipt. Also, neither the voter can deny a committed vote nor the administrator can change an accepted vote.

5.1 Basic Security Requirements

According to [3], the basic security requirements in an election system include completeness, soundness, privacy, unreusability, eligibility, fairness and verifiability. In this subsection, we analyze the security of our systems in terms of these parameters.

Completeness. In [3], completeness is interpreted as "All valid votes are counted correctly". Our system satisfies the completeness requirement because the Administrator cannot change or drop an accepted vote. The final number shown on the public counter indicates the number of ballots collected. Any drop of votes can be detected by the incorrectness of the counter. Moreover, our system prevents the Administrator from miscounting the votes because it is possible to determine if the voting result is correct from the published table. Also, the voters are allowed to check their voting record at a Trust Third Party (TTP) after the vote-opening phase.

When a voter wants to check her voting record, she presents Ψ together with the request to the TTP. Upon receiving the request, the TTP verifies Ψ using step (9) to (13) in protocol two, with Alice and Bob being replaced by the voter and the TTP respectively. If the verification is successful, it asks the Administrator for the reference number of the corresponding vote. This reference number will not be signed by the TTP and it is passed to the voter. She can check against the published list and see if her vote has been counted correctly. The voter remains anonymous.

As illustrated in Protocol Two, any change of vote by the Administrator can be detected. Further actions will be taken if disputes occur.

Soundness. Soundness refers to the inability of a dishonest voter to disrupt the voting. In our system, the voter cannot deny a committed vote, nor she can frame up the Administrator for changing his vote. This has been explained in section 3.2 and we do not repeat here.

Privacy. Privacy ensures a voter's vote being kept secret such that no one other than the voter knows her choice. Since the voting passes in our system are anonymous, voter privacy is provided. The voter's identity is only revealed to a trusted third party when dispute occurs.

Unreusability. In our system, no one is able to vote twice. Any reuse of the voting pass is disallowed. The Administrator makes sure that every pass is used only once during the voter authentication phase.

Eligibility. In an eligibility-supported voting system, anyone who is not allowed to vote cannot vote. In our system, the voting right of a voter is validated during the registration phase. Therefore the voting pass is only issued to the eligible voters. Multiple Registrars can be employed [14] in order to prevent a cheating Registrar from issuing voting passes to ineligible voters.

Moreover, the voting passes in our system are non-transferable. No ineligible voters can vote by buying a voting pass.

Fairness. In a fair voting system, nothing can affect the voting [3]. Our system is a fair system. Firstly, the Administrator cannot change a committed ballot. Secondly, a voter cannot sell her vote using the receipt because our system is receipt-free. Thirdly, the voting passes are non-transferable so that the unauthorized parties cannot vote. Lastly, the final result is published and can be verified.

Verifiability. According to [3], the verifiability is interpreted as whether there are any parties who can falsify the result of voting. In our system, the overall election results as well as the ballots are published. In addition, the voters are allowed to check their votes. Therefore our system supports the verifiability requirement.

5.2 Receipt-freeness

In some early electronic voting proposal such as [2], [3], [15] and [16], the voters are able to show to a third party how the vote is cast. Such systems are not receipt-free and they are against the fairness requirement of a voting system.

A definition of receipt-freeness has been given in [5]. According to this definition, an electronic voting system is receipt-free if the voter V_i can cast a vote $v_i^* \neq v_i$ and is accepted by the coercer, who wants to interfere the voting decision of V_i; while v_i is the favorite vote of the coercer. Our system is a receipt-free system because the voter has no evidence to prove how her vote has been cast. Unlike the scheme proposed in [3], the Administrator issues no signed document on how the vote has been received. Also, the bulletin board lists only the received ballots without the voter information. Therefore the voters cannot show how they vote in the election.

Even if the voter checks her vote at the TTP after the election, no receipt will be generated and she still cannot prove her vote to the coercer. Notice that the checker must possess the corresponding private key resides in Ψ as explained in section 3.1. Therefore only the voter in concern can check her voting record and she cannot prove her vote to a third party directly.

5.3 Non-transferability of Voting Right

We believe that the voting right must not be transferable in an eligible voting system. This requirement, however, is violated in a number of electronic voting proposals such as [6], [7], [8] and [9].

We have proposed the non-transferable voting pass in our paper. The voting pass contains the information of the private key of the holder. Our algorithm has been carefully designed so that the pass issuer can ensure that the private key is included in the pass, while no information about the private key is revealed. As illustrated in section 3.1, one must knows the values of a_i+u, $-a_i$, b_i, and c_i for all i in S' in order to succeed in the challenge on the voting pass. This means that the proving party must be able to access Alice's private key u since

$$a_i+u+(-a_i) \equiv u \ (mod\ \Phi(n'))$$

for any i in S'.

Therefore if Alice sells her pass to a third party, she is releasing her private key u as well. This explains why the voting pass is non-transferable.

6. CONCLUSION

In this paper, we have pointed out the requirement of non-transferability of voting right. Two cryptographic protocols, the anonymous authentication protocol and the anonymous commitment protocol, are presented. We have proposed a new electronic voting system in which the voters vote with the non-transferable voting passes, such voting passes are achieved by the anonymous authentication protocol. In addition, our system is receipt-free and it satisfies the basic security requirements include completeness, soundness, privacy, unreusability, eligibility, fairness and verifiability.

Acknowledgements

The authors would like to acknowledge Professor Victor K. W. Wei for his valuable opinions and the supervision of the research.

References

[1] David Chaum, Amos Fiat and Moni Naor, "Untraceable Electronic Cash," Advances in Cryptology – Proceedings of CRYPTO'88, 1988.

[2] David Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms," Communications of the ACM 24, 2, 1981.

[3] Atsushi Fujioka, Tatsuaki Okamoto, Kazuo Ohta. "A Practical Secret Voting Scheme for Large Scale Election," Advances in Cryptology – Proceedings of AUSCRYPT'92, 1992.

[4] J. Benaloh and D. Tuinstra, "Receipt-Free Secret-Ballot Elections (Extended Abstract)," Proceedings of the 26th Annual ACM Symposium on the Theory of Computing, 1994.

[5] Tatsuaki Okamoto, "Receipt-Free Electronic Voting Schemes for Large Scale Elections," Proceedings of Security Protocols'97, 1997.

[6] Michael J. Radwin, "An Untraceable, Universally Verifiable Voting Scheme," Seminar in Cryptology, 1995.

[7] Yi Mu and Vijay Varadharajan, "Anonymous Secure E-Voting over a Network," Proceedings of 14th Annual Computer Security Applications Conference, 1998.

[8] Qi He and Zhongmin Su, "A New Practical Secure e-Voting Scheme," Proceedings of SEC'98, 1998.

[9] A. Riera, J. Borrell and J. Rifa, "An Uncoercible Verifiable Electronic Voting Protocol," Proceedings of SEC'98, 1998.

[10] Rivest, R.L., A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, v. 21, n. 2, 1978.

[11] K. S. McCurley, "The Discrete Logarithm Problem", Cryptology and Computational Number Theory, v.42 of Proceedings of Symposia in Applied Mathematics, 1990.

[12] Josh Cohen Benaloh, "Secret Sharing Homomorphisms: Keeping Shares of a Secret Secret (Extended Abstract)," Proceedings of CRYPTO'86.

[13] Ronald Cramer, Matthew Franklin, Berry Schoenmakers and Moti Yung, "Multi-Authority Secret-Ballot Elections with Linear Work," Proceedings of EUROCRYPT'96, 1996.

[14] B. DuRette, "Multiple Administrators for Electronic Voting, " MIT Thesis, 1999.

[15] J. Cohen et al., "A Robust and Verifiable Cryptographically Secure Election Scheme," Proceedings of the 26th IEEE Symposium on Foundations of Computer Science, 1985.

[16] Josh Benaloh and Moti Yung, "Distributing the Power of a Government to Enhance the Privacy of Voters," Proceedings of 5th ACM Symposium on Principles of Distributed Computing, 1986.