# Security Enhancement for Anonymous Secure E-Voting over a Network

Iuon-Chang Lin† Min-Shiang Hwang‡ Chin-Chen Chang†

Department of Computer Science and Information Engineering†
National Chung Cheng University
Chaiyi, Taiwan, R. O. C.

Department of Information Management‡
Chaoyang University of Technology
168 Gifeng E. Rd., Wufeng,
Taichung County, Taiwan 413, R.O.C.
Email: mshwang@cyut.edu.tw
Fax: 886-4-23742337

**Abstract**

An electronic voting system makes it possible for the voters to cast their ballots over the computer network. Hence, voters can participate in elections without having to go to the polling places, which is more convenient and efficient. To design a practical voting scheme, Mu and Varadharajan have recently proposed an anonymous secure electronic voting scheme to be applied over the network. It does not only protect voters' privacy and prevent double voting, but also suits large scale elections. However, the scheme has a weakness in security; that is, some voters may still double vote without being detected and may even reveal information they should not. In this paper, we shall show this weakness and improve the scheme to increase the protection against fraudulence.

*Keywords:* Blind signatures, cryptography, electronic voting, ElGamal public key cryptosystem

## 1 Introduction

Broadly speaking, electronic voting systems can be classified into two categories: without anonymous channel and with anonymous channel. In both voting systems, the voters can cast their ballots over networks such as the Internet. This is more convenient and efficient than conventional voting approaches because people can vote without showing up in the polling places. Furthermore, the systems cannot invade voters' privacy. The intention of each voter must be kept secret. To deal with the property of privacy, the first type uses the framework of multi-authority to hide the content of each vote [7, 8]. Secret sharing schemes and zero-knowledge proof are often used to guarantee that no single authority can determine any individual vote. However, the voters have to communicate with several authorities in the voting process, and the system requires more computation in the vote counting procedure. This is unfavorable especially when it comes to large-scale elections. Therefore, to design a practical and efficient voting scheme for large-scale elections, the choice is usually with anonymous channel.

Having consulted several anonymous voting schemes [6, 9, 12], we decide on the privacy protection type as hiding the voters' identities and delivering the contents of the votes to the authority publicly. Such systems must satisfy the following requirements [14, 15].

- Anonymity of voters: The identity of the voter is untraceable.
- Accuracy of votes: Every one can verify the validity of the votes and assure that the votes cannot be altered, duplicated, and forged by any others (including the authority).

‡Responsible for correspondence: Prof. Min-Shiang Hwang

- Collision freedom: To guarantee that all legal votes differ from each other.
- Tally correctness: The final tally must be equal to the total number of the valid votes.
- Verifiability: Each voter can check if his/she own vote is published and tallied.
- Double voting detection: If double voting occurs, the authority can find who the voter is.

However, these schemes [6, 9, 12] require such cryptographic techniques as the blind signature [1, 3, 4] or some anonymous communication channels [2, 5, 16]. According to Karro's study [13], anonymous schemes have some weaknesses. One of the weaknesses is that, if two or more voters happen to choose the same keys, then these originally valid votes will be the same. It does not meet the requirement of collision freedom, and hence the tally center would decide that they are fraudulent votes and cross them out. Another weakness is that the anonymous channels are hard to implement over the Internet because the users would then be forced to browse in certain sites.

Mu and Varadharajan have recently proposed an anonymous, secure electronic voting scheme [15] which satisfies all the requirements for electronic voting. Mu and Varadharajan's scheme is based on the ElGamal digital signature algorithm and is suitable for large-scale elections. The authors claim that the system can detect the identities of the voters who send in wrongful votes on the condition of double voting. So far, many techniques [6, 9, 10, 12, 14, 17] have been proposed to perform electronic voting, but they cannot detect the illegal voters. Only Mu and Varadharajan's scheme can accomplish this mission. However, we shall show that Mu and Varadharajan's scheme has a deadly weakness in security that legal voters can actually successfully vote more than once without being detected.

In this paper, we shall show the weakness of Mu and Varadharajan's scheme and propose an improved version to overcome the weakness and also to satisfy all the requirements for electronic voting. The organization of the paper is as follows. In the next section, we shall briefly review Mu and Varadharajan's scheme and show the weakness of their scheme. In Section 3, we shall propose a new secure anonymous electronic voting scheme and analyze the security of the improved scheme. Finally, we shall summarize the benefits of our scheme in the last section of this paper.

## 2 Mu and Varadharajan's anonymous electronic voting scheme and its weakness

Mu and Varadharajan [15] have proposed two versions of an anonymous secure electronic voting scheme to be applied over the network. Both versions are based on the ElGamal digital signature [11] without any anonymous channel. One of the two assumes that the Authentication Server (AS) is trusted, and therefore it does not generate any voting ticket without the voter's consent. In addition, it does not leak out any information to the Voting Servers (VS) or Ticket Counting Server (TCS). The other version assumes that the AS is not to be trusted, which is closer to the truth. We shall emphasize the version that assumes the AS is not to be trusted.

### 2.1 Mu and Varadharajan's anonymous secure electronic voting scheme

In Mu and Varadharajan's voting scheme, participants are the following parties: Voters, an Authentication Server, Voting Servers, a Ticket Counting Server, and a Certificate Authentic. The scheme is composed of three phases: the voting ticket obtaining phase, the voting and tickets collecting phase, and the tickets counting phase.

For the rest of this paper, $V$ will denote the Voter, $AS$ the Authentication Server, $VS$ the Voting Servers, $TCS$ the Ticket Counting Server, $CA$ the Certificate Authentic, $t$ the timestamp, $p$ a large prime number, and $\|$ a concatenation of bits. Before performing the voting processes, each participant $x$ has a long term RSA [18] key pair $\{e_x, d_x\}$ and a product of two large prime numbers $n_x$, where $e_x \times d_x \bmod \phi(n_x) = 1$. For example, a voter $V$ holds an RSA key pair $\{e_V, d_V\}$ and $n_V$; $AS$ holds an RSA key pair $\{e_{AS}, d_{AS}\}$ and $n_{AS}$. Furthermore, the eligible voters hold a long-term voting certificate ($Cert$) delivered from CA. The certificate is signed by the CA's secret key, and the contents include a serial number, the voter's identity, the CA's identity, the RSA public key, the life-cycle time, and a timestamp.

1 **The voting ticket obtaining phase**

1. Before sending $Cert$ to prove that he/she is an eligible voter, a voter first chooses a blind factor $b$ and three random numbers $g$, $r$, $k_1 \in Z_p^*$, where $Z_p^*$ is the set of all positive integers smaller than and relatively prime to $p$. Then, the voter computes the parameters $a$, $x_1$, $x_2$, and $x_3$ by using the following equations:

$$\begin{aligned} a &= g^r \bmod p, \\ x_1 &= g b^{e_{AS}} \bmod n_{AS}, \qquad (1) \\ x_2 &= g^{k_1} b^{e_{AS}} \bmod n_{AS}, \quad \text{and} \qquad (2) \\ x_3 &= a b^{e_{AS}} \bmod n_{AS}. \end{aligned}$$

The voter sends $\{V, AS, Cert_V, (x_1\|x_2\|x_3\|t)^{d_V} \bmod n_V\}$ to AS.

2. $AS$ first verifies the validity of the certificate and validates the signature $(x_1\| \; x_2\| \; x_3\| \; t)^{d_V} \bmod n_V$. Then $AS$ chooses a random number $k_2$ and computes:

$$\begin{aligned} x_4 &= (k_2\|t)^{e_V} \bmod n_V, \\ x_5 &= (x_1^{3k_2} x_2^2 x_3)^{d_{AS}} \bmod n_{AS} \\ &= (y_1 y_2 a)^{d_{AS}} b^{3(k_2+1)} \bmod n_{AS}, \end{aligned}$$

where $y_1 = g^{k_1+k_2}$ and $y_2 = g^{k_1+2k_2}$. The messages $\{AS, V, x_4, (x_5\|t)^{e_V} \bmod n_V\}$ are delivered to $V$. Note that the parameter $k_2$ is different for each voter and that $AS$ stores $k_2$ along with $V's$ identity in its database.

3. $V$ obtains $k_2$ by decrypting $x_4$. Thus, $V$ can calculate $y_1$ and $y_2$. Furthermore, $V$ can derive $s$ by removing the blind factor $b^{3(k_2+1)}$ following the equation below:

$$\begin{aligned} s &= x_5 \times b^{-3(k_2+1)} \\ &= (y_1 y_2 a)^{d_{AS}} \bmod n_{AS}. \end{aligned}$$

$s$ is the signature for the product $y_1 y_2 a$. The signatures $s_1$ and $s_2$ of the voting content $m$ can be generated only from $V$ by the ElGamal digital signature algorithm [11]:

$$s_1 = (k_1 + k_2)^{-1}(ma - r) \bmod p - 1, \quad \text{and} \tag{3}$$
$$s_2 = (k_1 + 2k_2)^{-1}(ma - r) \bmod p - 1. \tag{4}$$

Then $V$ can obtain the voting ticket $T = \{a\|g\|y_1\|y_2\|s\|s_1\|s_2\|m\}$.

2 **The voting and tickets collecting phase**
After obtaining a valid voting ticket from $AS$, $V$ can cast the vote to the Voting Server $VS$ over the network. The main task of $VS$ is to guarantee the validity of the voting ticket.

1. $V$ sends the voting ticket $T$ to $VS$.

2. $VS$ decrypts $T$ and verifies the validity of $a, y_1, y_2$ by $AS's$ signature $s$. Then, $VS$ checks the correctness of the signatures $s_1$ and $s_2$ on $m$ by the following equations:

$$y_1^{s_1} a = g^{ma} \bmod p, \quad \text{and} \tag{5}$$
$$y_2^{s_2} a = g^{ma} \bmod p. \tag{6}$$

If the result of the above verification is positive, then $VS$ can make sure the ticket $T$ is valid. $VS$ stores all the voting tickets cast in and sends this batch to the Ticket Counting Server ($TCS$) over the network.

3 **The tickets counting phase**
All the Voting Servers send their tickets to the Ticket Counting Server ($TCS$). However, $V$ may use the same parameters $a, g, k_1, k_2$ to sign another voting content $m'$ and send the new ticket $T = \{a\|g\|y_1\|y_2\|s\|s_1'\|s_2'\|m'\}$ to a different $VS$. For double voting detection, $TCS$ checks the part parameters $a, g, y_1, y_2$ of $T$ to see whether they have been repetitively used. If the parameters have been used more than once, it means there is a case of double voting. $TCS$ has the ability to find who the voter is by the following equations:

$$k_1 + k_2 = \frac{m'a - ma}{s_1' - s_1} \bmod (p-1), \quad \text{and} \tag{7}$$
$$k_1 + 2k_2 = \frac{m'a - ma}{s_2' - s_2} \bmod (p-1). \tag{8}$$

From the above two equations, $TCS$ can obtain the parameter $k_2$, and hence $TCS$ can find the identity of the illegal voter. In this phase, the main tasks of $TCS$ are to count the tickets and to prevent any voter from casting more than once in different voting servers.

## 2.2 The weakness of Mu and Varadharajan's scheme

In this subsection, we shall show that Mu and Varadharajan's scheme has a weakness in security that a voter can in fact vote more than once without being detected. The attacks are described as follows.

**Attack 1:**
In Step 1, voter $V$ can obtain a valid ticket $T = \{a\|g\|y_1\|y_2\|s\|s_1\|s_2\|m\}$. The voter can successfully vote more than once in the following process. First, $V$ computes $g'$, $y'_1$, $y'_2$, $a'$ as

$$\begin{aligned} g' &= g^{c_0} \bmod p, \\ y'_1 &= g'^{(k_1+k_2+c_1)c_0^{-1}} \bmod p, \\ y'_2 &= g'^{(k_1+2K_2+c_2)c_0^{-1}} \bmod p, \\ a' &= g'^{(r+c_3)c_0^{-1}} \bmod p, \end{aligned}$$

where $c_0$, $c_1$, $c_2$, and $c_3$ are integers, $c_1, c_2, c_3 \neq 0$ and $c_1 + c_2 + c_3 = 0$. Therefore, $V$ can generate a new ticket $T' = \{a'\|g'\|y'_1\|y'_2\|s\|s'_1\|s'_2\|m\}$, where $s'_1$ and $s'_2$ are the signatures of $m$ created with the keys $(k_1+k_2+c_1)c_0^{-1}$ and $(k_1 + 2k_2 + c_2)c_0^{-1}$, respectively.

$$\begin{aligned} s'_1 &= ((k_1 + k_2 + c_1)c_0^{-1})^{-1}(ma' - (r + c_3)c_0^{-1}) \bmod p - 1, \quad \text{and} \\ s'_2 &= ((k_1 + 2k_2 + c_2)c_0^{-1})^{-1}(ma' - (r + c_3)c_0^{-1}) \bmod p - 1. \end{aligned}$$

In the voting and tickets collecting phase, $V$ can send the new ticket $T'$ to $VS$. $VS$ first verifies signature $s$ and checks the validity of $a', y'_1, y'_2$ following the equation:

$$\begin{aligned} s^{e_{AS}} &= y_1 y_2 a \bmod n_{AS} \\ &= y'_1 y'_2 a' \bmod n_{AS}. \end{aligned}$$

$VS$ then verifies the validity of $s'_1$ and $s'_2$ following Equations (5) and (6). Thus, $VS$ believes the ticket $T'$ is valid and sends it to $TCS$. For double voting protection, $TCS$ checks the parameters $a', g', y'_1, y'_2$ and decides that they have not been used more than once. Thus, the attack can succeed without being detected. Even if $VS$ detects that the signature $s$ has in fact been used before, it still cannot detect the identity of the illegal voter by Equations (7) and (8).

**Attack 2:**
Similar to Attack 1, $V$ first chooses a random number $h$ and computes $g' = g^h$, $a' = a^{h^2}$, $y'_1 = y_1^{h^2}$, $y'_2 = y_2^{h^2}$, and $s' = s^{h^2}$. Furthermore, the signatures $s'_1$ and $s'_2$ of $m$ can be computed by Equations (3) and (4), respectively.

$$\begin{aligned} s'_1 &= (k_1 + k_2)^{-1} h^{-1}(ma' - hr) \bmod p - 1, \quad \text{and} \\ s'_2 &= (k_1 + 2k_2)^{-1} h^{-1}(ma' - hr) \bmod p - 1. \end{aligned}$$

Therefore, $V$ can generate a new voting ticket $T' = \{a'\|g'\|y'_1\|y'_2\|s'\|s'_1\|s'_2\|m\}$, where $s'_1$ and $s'_2$ are computed by Equations (3) and (4) using the keys $(k_1 + k_2)h$ and $(k_1 + 2k_2)h$, respectively. $VS$ then believes $T'$ is a valid ticket because the following verifications hold.

$$\begin{aligned} (s')^{e_{AS}} &= (y_1 y_2 a)^{h^2} = y'_1 y'_2 a' \bmod n_{AS}, \\ y_1'^{s'_1} a' &= g'^{ma'} \bmod p, \\ y_2'^{s'_2} a' &= g'^{ma'} \bmod p. \end{aligned}$$

Using the above two attacks, a voter can vote more than once and can remain un-detected.

# 3 The improvement on Mu and Varadharajan's anonymous electronic voting scheme

In this section, we shall propose a secure electronic voting scheme for enhancing the security of Mu and Varadharajan's scheme, and we shall also analyze the security of the proposed scheme to prove its security. This new scheme can overcome the weakness of Mu and Varadharajan's scheme as described in Section 2.2.

### 3.1 The improved scheme

The proposed scheme is also composed of three phases with the some notations as are described in Section 2.1. The details of the improved scheme are as follows:

1 **The voting ticket obtaining phase**

1. A voter chooses two blind factors $b_1$ and $b_2$ as well as two random numbers $k_1$ and $r$. With these parameters, the voting system computes $w_1$ and $w_2$ by using the following equations:

$$\begin{aligned} w_1 &= g^r b_1^{e_{AS}} \bmod n_{AS}, \ and \\ w_2 &= g^{k_1} b_2^{e_{AS}} \bmod n_{AS}, \end{aligned}$$

where $g \in Z_p^*$ is the system's public parameter. Then, the voter sends $\{V, AS, Cert_V, t, w_1, w_2, ((w_1 \| w_2 \| t)^{d_V} \ mod \ n_V)\}$ to AS.

2. $AS$ first verifies the validity of the certificate and validates the signature $(w_1\|w_2\|t)^{d_V}$ mod$n_V$. If the verification result is positive, $AS$ can make sure that the received parameters are correct. Then $AS$ chooses a random number $k_2$ is different for each voter and computes:

$$\begin{aligned} w_3 &= (k_2\|t)^{e_V} \bmod n_V \\ w_4 &= (w_1 \times AS)^{d_{AS}} \bmod n_{AS} \\ &= (a \times AS)^{d_{AS}} b_1 \bmod n_{AS}, \\ w_5 &= (w_2 \times g^{k_2} \times AS)^{d_{AS}} \bmod n_{AS} \\ &= (y_1 \times AS)^{d_{AS}} b_2 \bmod n_{AS}, \\ w_6 &= (w_2^2 \times g^{k_2} \times AS)^{d_{AS}} \bmod n_{AS} \\ &= (y_2 \times AS)^{d_{AS}} b_2^2 \bmod n_{AS}, \end{aligned}$$

where $a = g^r$, $y_1 = g^{k_1+k_2}$, and $y_2 = g^{2k_1+k_2}$. The messages $\{AS, V, w_3, (w_4\|w_5 \|w_6\|t)^{e_V} \bmod n_V\}$ are delivered to $V$. Note that $AS$ stores $k_2$ along with $V$'s identity in its database.

3. $V$ obtains $k_2$ by decrypting $w_3$. Thus, $V$ can calculate $y_1$ and $y_2$. Furthermore, $V$ can calculate the signatures $s_1$, $s_2$, and $s_3$ by removing the blind factors following the equations below:

$$\begin{aligned} s_1 &= w_4 \times b_1^{-1} = (a \times AS)^{d_{AS}} \bmod n_{AS}, \\ s_2 &= w_5 \times b_2^{-1} = (y_1 \times AS)^{d_{AS}} \bmod n_{AS}, \\ s_3 &= w_6 \times b_2^{-2} = (y_2 \times AS)^{d_{AS}} \bmod n_{AS}. \end{aligned}$$

4. The voter applies the ElGamal digital signature scheme to sign the voting content $m$. Let $y_1$ and $y_2$ be the public keys of the ElGamal Cryptosystem, and $x_1 = k_1 + k_2$ and $x_2 = 2k_1 + k_2$ be the corresponding secret keys, such that $y_1 = g^{k_1+k_2} \ mod \ p$ and $y_2 = g^{2k_1+k_2} \ mod \ p$. The two signatures $(a, s_4)$ and $(a, s_5)$ of the voting content $m$ can be generated from the following equations

$$\begin{aligned} s_4 &= x_1^{-1}(ma - r) \bmod p - 1, \ and \\ s_5 &= x_2^{-1}(ma - r) \bmod p - 1, \end{aligned}$$

respectively. Then $V$ can obtain the voting ticket as $T = \{s_1 \|s_2\|s_3 \|s_4\|s_5 \|a\|y_1 \|y_2 \|m\}$.

2 **The voting and tickets collecting phase**

1. $V$ sends the voting ticket $T$ to VS.

2. $VS$ verifies the validity of $a$, $y_1$, and $y_2$ by checking the following equations:

$$AS \times a \stackrel{?}{=} s_1^{e_{AS}} \bmod n_{AS}, \tag{9}$$

$$AS \times y_1 \stackrel{?}{=} s_2^{e_{AS}} \bmod n_{AS}, \ and \tag{10}$$

$$AS \times y_2 \stackrel{?}{=} s_3^{e_{AS}} \bmod n_{AS}. \tag{11}$$

If all are positive, $VS$ also verifies the correctness of the signatures $(a, s_4)$ and $(a, s_5)$ on $m$ by checking the following equations:

$$\begin{aligned} g^{ma} &\stackrel{?}{=} y_1^{s_4} a = \bmod p, and \\ g^{ma} &\stackrel{?}{=} y_2^{s_5} a = \bmod p, \end{aligned}$$

respectively. If both the verifications turn out positive, $VS$ can make sure the ticket $T$ is valid. $VS$ stores all the voting tickets cast in and sends this batch to $TCS$ over the network.

[scale=.6]fig1.eps

Figure 1: The flowchart of the improved scheme

3 **The tickets counting phase**
All the Voting Servers send their tickets to $TCS$. $TCS$ publishes the tickets and counts them. Besides that, it is also responsible for detecting double voting. The double voting detection process is as follows.Assume that a voter uses the same parameters $y_1$, $y_2$, and $a$ to sign another voting content $m'$ and sends the new voting ticket to a different $VS$. For double voting detection, $TCS$ first checks $y_1$, $y_2$, and $a$ of $T$ to see whether they have been used before. If the parameters have been used more than once and the voting contents are not the same, it means that there is a case of double voting. $TCS$ has the ability to find who the voter is by figuring out the following equations:

$$\begin{aligned} x_1 &= \frac{m'a - ma}{s_4' - s_4} \bmod (p-1), and \\ x_2 &= \frac{m'a - ma}{s_5' - s_5} \bmod (p-1). \end{aligned}$$

From the above two equations, $TCS$ can obtain the parameters from computing $x_2 - x_1 = (2k_1 + k_2) - (k_1 + k_2) = k_1$. Consequently, $k_2$ can be easily recovered by computing $x_1 - k_1 = k_2$, and hence $TCS$ can find the identity of the illegal voter. Since the contents of votes are open, the complexity problem in counting votes in large scale systems is overcome in these systems.

The flowchart of the improved scheme is in Figure 1.

## 3.2 The security of the improved scheme

Our improved scheme is also based on the ElGamal digital signature and satisfies all the electronic voting requirements. The proposed scheme enhances the security and overcomes the weakness of Mu and Varadharajan's scheme. In the following analysis, we shall show that our scheme can resist the previous attacks and the parties conspiring attack.

3 **Resisting the previous attacks**
Assume that a voter wants to forge the parameters $a$, $y_1$, and $y_2$ like what happens in Attacks 1. However, it is impossible to make corresponding signatures $s_1$, $s_2$, and $s_3$ that satisfy Equations 9, 10, and 11, respectively. The voter does not know the AS's private key $d_{AS}$. Hence, these forged parameters still cannot escape the verifications.

In the second attack, any voter can also make a forged signature $s_2' = (AS' \times y_2')^{d_{AS}}$ easily, but he/she cannot generate the correct signatures on the voting content $m$. For instance, suppose $s_2' = s_2^2 = (AS^2 \times y_2^2)^{d_{AS}}$ and $y_2' = AS \times y_2^2$. Hence the parameter $y_2'$ can pass the verification. However, the voter cannot obtain the corresponding secret key $x_1'$ due to the difficulty of computing discrete logarithms. Without the secret key, the voter cannot generate the correct signature. Therefore, Attacks 2 cannot do any harm to the improved scheme. A voter can never generate another valid vote.

- **Resisting the parties conspiring attack**
First, we assume that two or more voters conspire to obtain a new signature from individual signatures. Suppose that two voters $V_1$ and $V_2$ have valid signatures $(s_{11}, s_{12}, s_{13})$ and $(s_{21}, s_{22}, s_{23})$, respectively. They conspire to obtain a new signature of $(s_1', s_2', s_3')$ by the following equation:

$$s_i' = s_{1i} \times s_{2i} \bmod n_{AS}$$

where $i = 1, 2, \cdots, 4$. However, to compute the corresponding parameters $r'$, $x_1'$, and $x_2'$, the voters have to solve discrete logarithms.

On the other hand, we assume that $AS$, $VS$, and $TCS$ are not to be trusted in our scheme. AS maybe leak out the request information to VS or $TCS$. Even if $AS$, $VS$, and $TCS$ have all the voting information, they still cannot trace the identities of the voters. In addition, if $AS$, $VS$, and $TCS$ try to conspire to forge a valid voting ticket, they still cannot do any harm to this scheme. The reason is that $T$ is public, which means everyone can check wheter the total number of voting tickets is equal to the total number of registered voters. Thus, our scheme can prevent the parties conspiring attack.

Furthermore, a voter may try to use the same secret key to sign a different content $m'$. However, the scheme also has the ability to detect who is using the same secret key to double vote as described in the tickets counting phase. Therefore, the improved scheme can prevent a voter from voting more than once.

## 4 Conclusion

In this article, we have proposed a secure electronic voting scheme that overcomes the weakness of Mu and Varadharajan's scheme. In addition, the scheme also meets all the electronic voting requirements: anonymity of voters, accuracy of voters, collision freedom, tally correctness, verifiability, and double voting detection. So far, there are few methods to meet all the electronic voting requirements, especially double voting detection. Moreover, the scheme is suitable for large-scale elections and does not require any special voting channel.

## References


[1] J. Camenisch, J. Piveteau, and M. Stadler, "Blind signatures based on discrete logarithm problem," *Advances in Cryptology, EUROCRYPT'94*, pp. 428–432, Lecture Notes in Computer Science, 950, 1994.

[2] D. Chaum, "Untracable electronic mail, return addresses, and digital pseudonyms," *Communication of the ACM*, vol. 24, pp. 84–88, 1981.

[3] D. Chaum, "Blind signatures for untraceable payments," *Advances in Cryptology, CRYPTO'82*, pp. 199–203, 1982.

[4] D. Chaum, "Blind signatures system," *Advances in Cryptology, CRYPTO'83*, pp. 153–156, 1983.

[5] D. Chaum, "The dinning cryptographers problem: Unconditional sender and recipient untraceability," *Joural of Cryptography*, vol. 1, pp. 65–75, 1988.

[6] D. Chaum, "Elections with unconditionally secret ballots and disruption equivalent to breaking RSA," *Advances in Cryptology, EUROCRYPT'88*, pp. 177–182, 1988.

[7] R. Cramer, M. Franklin, B. Schoenmakers, and M. Yung, "Multi-authority secret bollat elections with linear work," *Advances in Cryptology, EUROCRYPT'96*, pp. 72–83, Lecture Notes in Computer Science, 1070, 1996.

[8] R. Cramer, R. Gennaro, and J. Borrell, "A secure and optimally efficient multi-authority election scheme," *Advances in Cryptology, EUROCRYPT'97*, pp. 103–117, Lecture Notes in Computer Science, 1233, 1997.

[9] L. Cranor and R. Cytron, "Sensus: A security-conscious electronic polling system for the Internet," *Proceedings of the International Cnference on System Sciences*, Hawaii, 1997.

[10] G. Dini, "Electronic voting in a large-scale distributed system," *Networks*, Vol. 38, pp. 22–32, 2001.

[11] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. IT-31, pp. 469–472, July 1985.

[12] A. Fujioka, T. Okamoto, and K. Ohta, "A practical secret voting scheme for large-scale elections," *Advances in Cryptology, AUSCRYPT'92*, pp. 244–251, Lecture Notes in Computer Science, 718, 1993.

[13] J. Karro and J. Wang, "Towards a practical, secure, and very large scale online election," *Proceedings of the 15th Annual Computer Security Applications Conference, CACSAC'99*, pp. 161–169, 1999.

[14] C.L. Lei and C.I. Fan, "A universal single-authority election system," *IEICE Transactions on Fundamentals*, vol. E81-A, no. 10, pp. 2186–2193, 1998.

[15] Y. Mu and V. Varadharajan, "Anonymous secure e-voting over a network," *Proceedings of the 14th Annual Computer Security Applications Conference, CACSAC'98*, pp. 2936–299, 1998.

[16] A. Pfitzmann, "A switched/broadcase ISDN to decrease user oberability," *Proceedings of the International Zurich Seminar on Digital Communication*, pp. 183–190, 1984.

[17] I. Ray and N. Narasimhamurthi, "An Anonymous Electronic Voting Protocol for Voting Over the Internet," *Proceedings of Third International Workshop on Advanced Issues of E-Commerce and Web- Based Information Systems*, pp. 188–190, 2001.

[18] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, pp. 120–126, Feb. 1978.