The ISC Int'l Journal of Information Security

ISeCure

January 2011, Volume 3, Number 1 (pp. 1–8)

http://www.isecure-journal.org

Double Voter Perceptible Blind Signature Based Electronic Voting Protocol

Yaser Baseri [a,b,*], Amir S. Mortazavi [c], Maryam Rajabzadeh Asaar [c], Mohsen Pourpouneh [d], and Javad Mohajeri [a]

[a] *Electronics Research Center, Sharif University of Technology,Tehran, Iran.*
[b] *Department of Computer Science and Engineering, Payame Noor University, Tehran, Iran.*
[c] *Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran.*
[d] *Department of Mathematics, Shahid Beheshti University, Tehran, Iran.*

ARTICLE INFO.

Article history:
Received: 7 September 2009
Revised: 5 October 2010
Accepted: 7 October 2010
Published Online: 15 January 2011

Keywords:
Electronic Voting, Anonymity of Voter, Unforgeability of Ticket, Perceptibility of Double Voting, Security of Voting, Blind Signature

ABSTRACT

Mu *et al.* have proposed an electronic voting protocol and claimed that it protects anonymity of voters, detects double voting and authenticates eligible voters. It has been shown that it does not protect voter's privacy and prevent double voting. After that, several schemes have been presented to fulfill these properties. However, many of them suffer from the same weaknesses. In this paper, getting Asadpour *et al.*'s scheme as one of the latest ones and showing its weaknesses, we propose a new voting scheme which is immune to the weaknesses of previous schemes without losing efficiency. The scheme, is based on a special structure, which directly uses the identity of the voter, hides it in that structure and reveals it after double voting. We also, show that the security of this scheme depends on hardness of RSA cryptosystem, Discrete Logarithm problem and Representation problem.

1 Introduction

Nowadays, computers are almost everywhere and they are used for many purposes. One of these purposes is electronic voting. By using computer networks and the Internet, traditional voting can be substituted by electronic voting, which speeds up election process, decreases costs and facilitates voting process.

Electronic voting schemes can be classified into three types: blind signature-based electronic voting schemes [1–4], homomorphic encryption-based electronic voting schemes [5, 6], and the schemes which use randomization such as the schemes that employ mixnets [7, 8]. In the schemes based on blind signature, the voter first gets a token which is a blindly signed message unknown to any one except him, and then sends his token together with his vote anonymously.

One of the first schemes which is based on blind signature and used to claim that it can detect double voters, relates to Mu and Varadharajan [9]. They also claimed that their scheme is suitable for large scale elections. They have proposed two versions of their electronic voting scheme based on the ElGamal digital

* Corresponding author.
Email addresses: yaser_baseri@alum.sharif.ir (Y. Baseri), sa_mortazavi@ee.sharif.edu (A. S. Mortazavi), asaar@ee.sharif.ir (M. R. Asaar), m.pourpouneh@mail.sbu.ac.ir (M. Pourpouneh), mohajer@sharif.edu (J. Mohajeri).
ISSN: 2008-2045

signature [10], to be applied over network without any anonymous channel. One of these schemes assumes that the authentication server is trusted, and therefore it does not generate any voting ticket without the voter's consent. In this version, the authentication server does not leak out any information to the voting server or ticket counting server. The other version assumes that authentication server is not trusted, which is closer to truth. In 2003, Chien *et al.* [11] showed that Mu-Varadharajan's schemes suffer from some weaknesses including: 1) the authentication server can easily identify the owner of a cast ballot, 2) a valid voter can vote more than one without being detected, 3) any one can forge ballot without being authenticated.

In 2003, Lin *et al.* [12] proposed an improvement on Mu and Varadharajan's scheme. They improved the weakness that voters could successfully vote more than once without being detected. The proposed scheme did not require any special voting channel and it is claimed that the scheme is able to detect double voting effectively. Yang *et al.* in 2004 [13] proposed another improvement on Mu-Varadharajan's scheme. Although their scheme is resistant to the attacks which have been proposed in [11], it can not determine the identity of double voters. In 2005, Hwang *et al.* [14] represented an attack on Lin *et al.*'s protocol. They showed that the Lin *et al.*'s modification allows the authentication server to identify the voters of published tickets so that voters will lose their privacy. They also proposed a new scheme to solve this problem and enhance the security. They used two generators so that after publishing all cast tickets by ticket counting server, authentication server could not trace the owner of the tickets. By these changes they tried to improve the privacy of voters in Lin *et al.*'s protocol. However, Hwang *et al.*'s scheme had some weaknesses in fulfilling the claimed properties [15]. Furthermore, Asaar *et al.* [16] proposed one more scheme based on Lin *et al.*'s scheme. Their scheme resists to the attacks which have been proposed in [12]. In 2007, F. Rodriguez-Henriquez *et al.* [17] proposed another improvement over the Lin *et al.*'s scheme. They presented a fully functional RSA/DSA-based e-voting protocol for online elections. They presented a weakness of Lin *et al.*'s scheme arising from the structure of ElGamal digital signature. For preventing the proposed weakness, they substituted the ElGamal digital signature employed by other protocols with DSA signature [18]. These changes guarantee that independently-chosen values by the voter and authentication server would not have undesirable effects on the ticket obtaining procedure. In 2010, Jahandideh *et al.* [15] showed that all of Lin *et al.*'s [12], Yang *et al.*'s [13], Hwang et al.'s [14], Rodriguez-Henriquez *et al.*'s [17] and Asaar *et al.*'s [16] protocols suffer from some weaknesses. One of the latest schemes which has been proposed in this category is Asadpour *et al.*'s protocol [19]. Using hash functions, they proposed a new scheme and claimed that their scheme is immune to some of their attacks. However, as we show in this paper, it suffers from some other weaknesses beside the weaknesses they have counted for, in their schemes.

In this paper, we review Asadpour *et al.*'s protocol as one of the latest improvements on Mu *et al.*'s protocol and describe its weaknesses in Section 2. Furthermore, we propose a new scheme which hides the identity of voter in the structure of blind signature and reveals it after double voting takes place in a different way in Section 3. In the proposed scheme, hiding the identity of voter in the structure of blind signatures, we use a construction for authentication of voters, protection of voters' anonymity, detection of double voters and prevention of the attacks which have been presented until now on this family of protocols. In this structure, we use the identity of voter directly and hide it in that structure and reveal it, if a malicious voter has voted twice or more. According to Pointcheval's definition of restrictive blind signature[20] [1] , we can enumerate the used signature scheme as restrictive blind signature. Next, in Section 4, we present the security analysis of the scheme and show that the security of our system could be reduced to the security of RSA cryptosystem and difficulty of Discrete Logarithm problem and Representation problem. Finally, in Section 5, we show a comparison between the efficiency of our scheme and Asadpour *et al.*'s scheme and show that our proposed scheme is more efficient than their scheme.

2 Asadpour *et al.*'s Scheme and its Failures

First we describe the protocol proposed by Asadpour *et al.*'s in Section 2.1. Then in Section 2.2 we present some attacks to the protocol.

2.1 Asadpour *et al.*'s Scheme

Asadpour *et al.*'s electronic voting scheme consists of the participants including Voters (V), an Authentication Server (AS), Voting Servers (VS), a Ticket Counting Server (TCS), and a Certificate Authority (CA). In order to describe the protocol, we use the following notations:

- $(e_x, n_x), d_x$: the RSA public/private key pair of participant x.
- $Cert_x$: the public-key certificate of participant x, which is signed by CA.

[1] Those blind signatures which hide a specific structure, such as the identity, are called *restrictive blind signature*.

- p: a large prime number, which is a public system parameter.
- g, h: two different elements in $\mathbb{Z}_p^*$ which are also public system parameters.
- $\|$: the operation of concatenation.
- t: timestamp.
- $Hash$: a one way hash function.

2.1.1 The Voting and Ticket Obtaining Phase

(a) Voter V chooses three blind factors $b_0, b_1 and b_2$ in $\mathbb{Z}_{n_{AS}}^*$ and two random numbers k_1 and r in $\mathbb{Z}_p^*$. Then, V computes w_0, w_1, w_1', w_2 and w_2' by the following equations:

$$\begin{aligned} \mathcal{H}_{lnk} &= Hash(g^r, h^r) = Hash(a_1, a_2) \\ w_0 &= \mathcal{H}_{lnk}.b_0^{e_{AS}} \bmod n_{AS} \\ w_1 &= g^r b_1^{e_{AS}} \bmod n_{AS} \\ w_1' &= h^r b_1^{e_{AS}} \bmod n_{AS} \\ w_2 &= g^{k_1} b_2^{e_{AS}} \bmod n_{AS} \\ w_2' &= h^{k_1} b_2^{e_{AS}} \bmod n_{AS} \end{aligned} \quad (1)$$

Next, the voter sends $\{V, AS, Cert_V, t, w_1, w_1', w_2, w_2', (w_1\|w_1'\|w_2\|w_2'\|t)^{d_V} \bmod n_V\}$ to AS.

(b) AS verifies the validity of the certificate, timestamp, and the signature $((w_1\|w_1'\|w_2\|w_2'\|t)^{d_V}) mod\, n_V$. Getting all the verifications passed, AS chooses a unique random number k_2 for the voter and computes:

$$\begin{aligned} w_3 &= (k_2\|t)^{e_V} \bmod n_v \\ w_4 &= (w_1 \times w_0)^{d_{AS}} \bmod n_{AS} \\ &= (a_1 \times \mathcal{H}_{lnk})^{d_{AS}} \times b_0 \times b_1 \bmod n_{AS} \\ w_5 &= (w_1' \times w_0)^{d_{AS}} \bmod n_{AS} \\ &= (a_2 \times \mathcal{H}_{lnk})^{d_{AS}} \times b_0 \times b_1 \bmod n_{AS} \\ w_6 &= (w_2 \times g^{k_2} \times w_0)^{d_{AS}} \bmod n_{AS} \\ &= (y_1 \times \mathcal{H}_{lnk})^{d_{AS}} \times b_0 \times b_2 \bmod n_{AS} \\ w_7 &= (w_2'^2 \times h^{k_2} \times w_0)^{d_{AS}} \bmod n_{AS} \\ &= (y_2 \times \mathcal{H}_{lnk})^{d_{AS}} \times b_0 \times b_2^2 \bmod n_{AS} \end{aligned} \quad (2)$$

Where $a_1 = g^r$, $a_2 = h^r$, $y_1 = g^{k_1+k_2}$, and $y_2 = h^{2k_1+k_2}$. Subsequently, AS sends the message $\{AS, V, w_3, (w_4\|w_5\|w_6\|w_7\|t)^{e_V} \bmod n_V\}$ to V and stores k_2 along with V's identity in its database.

(c) Decrypting w_3, V obtains k_2 and using g, h, k_1 and k_2, he calculates y_1 and y_2. Furthermore, removing the blinding factors b_0, b_1 and b_2 from w_4, w_5, w_6 and w_7, he computes the signatures s_1, s_2, s_3 and s_4 as follows:

$$\begin{aligned} s_1 &= w_4 \times b_1^{-1} \times b_0^{-1} \, mod\, n_{AS} = (a_1 \times \mathcal{H}_{lnk})^{d_{AS}} \, mod\, n_{AS} \\ s_2 &= w_5 \times b_1^{-1} \times b_0^{-1} \, mod\, n_{AS} = (a_2 \times \mathcal{H}_{lnk})^{d_{AS}} \, mod\, n_{AS} \\ s_3 &= w_6 \times b_2^{-1} \times b_0^{-1} \, mod\, n_{AS} = (y_1 \times \mathcal{H}_{lnk})^{d_{AS}} \, mod\, n_{AS} \\ s_4 &= w_7 \times b_2^{-2} \times b_0^{-1} \, mod\, n_{AS} = (y_2 \times \mathcal{H}_{lnk})^{d_{AS}} \, mod\, n_{AS} \end{aligned} \quad (3)$$

(d) V applies the ElGamal digital signature scheme [10] to sign the voting content m. Let $x_1 = k_1 + k_2$ and $x_2 = 2k_1 + k_2$ be the private keys and y_1 and y_2 be the corresponding public keys of ElGamel system, i.e., $y_1 = g^{k_1+k_2} mod\, p$ and $y_2 = h^{2k_1+k_2} mod\, p$. V generates two signatures (a_1, s_5) and (a_2, s_6) using the following equations:

$$\begin{aligned} s_5 &= x_1^{-1}(ma_1 - r) \, mod\, (p-1) \\ s_6 &= x_2^{-1}(ma_2 - r) \, mod\, (p-1) \end{aligned} \quad (4)$$

Finally, the voting ticket can be computed as

$$T = \{s_1 \parallel s_2 \parallel s_3 \parallel s_4 \parallel s_5 \parallel s_6 \parallel a_1 \parallel a_2 \parallel y_1 \parallel y_2 \parallel m\}$$

2.1.2 The Voting and Tickets Collecting Phase

(a) V sends the voting ticket T to VS.

(b) VS validates a_1, a_2, y_1, and y_2 by checking the following equations:

$$\begin{aligned} \mathcal{H}_{lnk} \times a_1 &\stackrel{?}{=} s_1^{e_{AS}} mod\, n_{AS} \\ \mathcal{H}_{lnk} \times a_2 &\stackrel{?}{=} s_2^{e_{AS}} mod\, n_{AS} \\ \mathcal{H}_{lnk} \times y_1 &\stackrel{?}{=} s_3^{e_{AS}} mod\, n_{AS} \\ \mathcal{H}_{lnk} \times y_2 &\stackrel{?}{=} s_4^{e_{AS}} mod\, n_{AS} \end{aligned} \quad (5)$$

Furthermore, VS verifies the signatures (a_1, y_1, s_5) and (a_2, y_2, s_6) of the voting content m by checking the following equations:

$$\begin{aligned} y_1^{s_5} a_1 &\stackrel{?}{=} g^{ma_1} mod\, p \\ y_2^{s_6} a_2 &\stackrel{?}{=} h^{ma_2} mod\, p \end{aligned} \quad (6)$$

If both verifications succeed, VS stores T in its database.

(c) After the voting time expired, VS sends all the collected tickets to TCS.

2.1.3 The Tickets Counting Phase

Upon receiving all tickets from the Voting Servers, TCS first verifies if there are double voting tickets by checking y_1, y_2, a_1 and a_2 for every ticket and see whether they have been repetitively used. If these parameters appear in more than one ticket, the owner of this ticket has voted twice or more. In cooperation

with AS, TCS finds the malicious voter. When TCS discovers a voter who has used the same parameters y_1, y_2, a_1 and a_2 to sign two different voting contents m and m', it calculates k_2 using the following equations:

$$\begin{aligned} x_1 &= \tfrac{m'a_1 - ma_1}{s'_5 - s_5} \bmod (p-1) \\ x_2 &= \tfrac{m'a_2 - ma_2}{s'_6 - s_6} \bmod (p-1) \\ k_1 &= x_2 - x_1 = (2k_1 + k_2) - (k_1 + k_2) \\ k_2 &= x_1 - k_1 \end{aligned} \tag{7}$$

Searching AS's database and associating the unique number k_2 with the malicious voter, TCS is able to identify him. Finally, the TCS publishes the valid tickets and counts them.

2.2 Weaknesses of the Scheme

In [15], we have proposed some attacks on Lin *et al.*'s [12], Yang *et al.*'s [13], Hwang *et al.*'s [14], Rodriguez-Henriquez *et al.*'s [17] and Asaar *et al.*'s [16] protocols and shown that they suffer from some weaknesses. In this section, we show that some of these attacks are applicable to Asadpour *et al.*'s protocol. By representing two attacks on the anonymity of the voter, we show that beside the weakness in fulfilling the property of perceptibility of double voter which is mentioned in their paper, Asadpour *et al.*'s scheme suffers from some weaknesses in protecting the anonymity of the voter.

2.2.1 The First Attack

Since parameters w_1 and w'_1 are blinded with the same blinding factor for each voter, i.e., $w_1 = g^r b_1^{e_{AS}} \bmod n_{AS}$ and $w'_1 = h^r b_1^{e_{AS}} \bmod n_{AS}$, AS is able to compute the proportion of them and consequently the proportion of g^r and h^r for each voter. On the other hand, when tickets get published on the bulletin board at the end of voting process, AS is able to compute the proportion of a_1 and a_2 and consequently the proportion of g^r and h^r in $mod\, n_{AS}$. Matching these two proportions, AS is able to determine the owner of each vote m.

2.2.2 The Second Attack

After publishing tickets on the bulletin board, AS has access to the information of all tickets. On the other hand, AS has allocated the value of k_2 for each voter and stored it in its database beside the identity of each voter. Suppose that AS would be interested in finding the owner of the ticket $T = \{s_1 \parallel s_2 \parallel s_3 \parallel s_4 \parallel s_5 \parallel s_6 \parallel a_1 \parallel a_2 \parallel y_1 \parallel y_2 \parallel m\}$, AS selects a record $\{V', k'_2\}$ from its own database and computes the value r' as follows:

$$r' = \tfrac{s_5 s_6 k'_2 - m(2a_1 s_6 - a_2 s_5)}{s_5 - 2s_6} \bmod p \tag{8}$$

Since $a_1 = g^r \bmod p$, if the equation $a_1 \stackrel{?}{=} g^{r'} \bmod p$ holds, then $r' = r$ and V' is the owner of this ticket; else AS chooses another record from its own database and redoes this procedure until the owner of this vote is identified.

3 Emergence of Dynamic Environments

Our electronic voting environment involves at least the following parties: voters (V's), an authentication sever (AS), voting servers(VS's), a ticket counting server (TCS) and a trusted certificate authority (CA). For convenience, some necessary notations are defined below:

- $(e_i, n_i), d_i$: the RSA public/private key pair of participant except AS.
- $(e_{AS}, n_{AS}), 1/e_{AS}$ and $(e'_{AS}, n_{AS}), 1/e'_{AS}$: the two RSA public/private key pairs of AS such that $e_{AS} > e'_{AS}$.
- $Cert_x$: the public-key certificate of participant x, which is signed by CA.
- g_1, g_2: two publicly known elements of the same large prime order l in $\mathbb{Z}^*_{n_{AS}}$.
- u_v: which is unique for each voter and is unknown to others.
- ID_v: the identity of the voter which is certified by certificate authority and is equal to $g_1^{u_v} \bmod n_{AS}$.
- b_1 and b_2: two blind factors in $\mathbb{Z}^*_{n_{AS}}$, which are relatively prime to n_{AS}.
- $\mathcal{H}$: a one way hash function.
- $\parallel$: the operation of concatenation.
- t: timestamp.

Note that the used RSA system for AS is based on the difficulty of computation of v'th root of numbers in $\mathbb{Z}^*_n$, such that $n = p * q$ and p, q are two large prime numbers. The public exponent of the RSA system is e, a reasonably large prime, and ciphertexts are computed as e'th exponent of plaintexts. For decryption, decryptor computes e'th root of ciphertexts. Every one who knows the factorization of n is able to compute e'th root of numbers and consequently able to decrypt ciphertexts. Hence, here, no one except AS knows the factorization of n. This type of cryptosystem has been used in some other protocols such as Ferguson's electronic cash protocol [21]. Furthermore, for security enhancement and preventing some security attacks based on homomorphic property, we use two different pairwise keys for AS.

The scheme consists of three phases: 1) voting preparation, in which the voter authenticates himself and gets a valid ticket from the authentication server, 2)

voting and collecting ballot, in which the voter sends the ballot to a voting server, then the voting server verifies the eligibility of the voter by checking the signature of the authentication server which is in the ticket and then sends the ballot to the ballot counting server, and 3) counting ballots in which the ballots are counted and double voters are detected. In this section, we describe each phase in detail.

3.1 The First Phase: Voting and Ticket Obtaining Phase

(a) The voter selects two blind factors b_1 and b_2 and three random numbers x_1, $x_2 \in \mathbb{Z}^*_{e'_{AS}}$ and $s \in \mathbb{Z}^*_{e_{AS}}$ and computes A, A', B, w_1, w_2 as follow:

$$\begin{aligned} A &= g_1^{u_v} g_2 \bmod n_{AS} \\ A' &= A^s \bmod n_{AS} \\ B &= g_1^{x_1} g_2^{x_2} \bmod n_{AS} \\ w_1 &= B b_1^{e'_{AS}} \bmod n_{AS} \\ w_2 &= (A' + B) b_2^{e_{AS}} \bmod n_{AS} \end{aligned} \tag{9}$$

Then, the voter sends $\{Cert_V, A, w_1, w_2, t, ((A\|w_1\|w_2\|t)^{d_V}) \bmod n_V\}$ to AS.

(b) AS first verifies the validity of the certificate, timestamp and value of A by using the certificate, identity of the voter and public information. It also, validates the signature $((A\|w_1\|w_2\|t)^{d_V}) mod\, n_V$. After passing all verifications, AS computes the following equations:

$$\begin{aligned} w_3 &= A^{1/e_{AS}} \bmod n_{AS} \\ w_4 &= w_1^{1/e'_{AS}} \bmod n_{AS} \\ w_5 &= w_2^{1/e_{AS}} \bmod n_{AS} \end{aligned} \tag{10}$$

Finally, the message $\{((w_3\|w_4\|w_5\|t)^{e_V}) \bmod n_V\}$ is sent to V.

(c) Decrypting the received value, V will get access to the signature of AS on A and blinded signatures of AS on B and $A' + B$. V computes the signatures of AS on A', B and $A' + B$ as follow:

$$\begin{aligned} s_1 &= w_3^s \bmod n_{AS} = A'^{1/e_{AS}} \\ s_2 &= w_4 / b_1 \bmod n_{AS} = B^{1/e'_{AS}} \\ s_3 &= w_5 / b_2 \bmod n_{AS} = (A' + B)^{1/e_{AS}} \end{aligned} \tag{11}$$

Then he chooses his vote and computes the values of d, r_1 and r_2 using the following equations:

$$\begin{aligned} d &= \mathcal{H}(A', B, s_1, s_2, s_3, vote, nonce) \bmod e_{AS} \\ r_1 &= d u_v s + x_1 \bmod e_{AS} \\ r_2 &= d s + x_2 \bmod e_{AS} \end{aligned} \tag{12}$$

Finally, the voting ticket could be computed as

$$Ticket = \{A', B, vote, s_1, s_2, s_3, d, r_1, r_2, nonce\} \tag{13}$$

3.2 The Second Phase: Voting and Tickets Collecting Phase

(a) V sends the voting ticket $Ticket$ to VS.

(b) VS verifies the signatures s_1, s_2, s_3 using the information available in the ticket. It also, verifies the following equation to ensure that no item has been forged in the protocol.

$$g_1^{r_1} g_2^{r_2} \stackrel{?}{=} A'^d B \bmod n_{AS} \tag{14}$$

If the validation holds, VS stores $Ticket$ in its database.

(c) After the voting time expires, VS sends all the collected tickets to TCS.

3.3 The Third Phase: Tickets Counting Phase

Upon receiving all tickets from the voting servers, TCS verifies if double voting has occurred or not. This affair is done by checking the parameters A' and B of tickets and detecting if they have been repeatedly used. If these parameters appear in more than one ticket, the voter has voted twice or more. If the TCS finds the same items A and B in two or more tickets (i.e., $\{A', B, vote, s_1, s_2, s_3, d, r_1, r_2\}$ and $\{A', B, vote, s_1, s_2, s_3, d', r'_1, r'_2\}$), then by using the relation between r_1, r_2, d and consequently between r'_1, r'_2, and d', it computes the identity of the voter by the following equations:

$$\begin{aligned} u_v &= \tfrac{r_1 - r'_1}{r_2 - r'_2} \bmod e_{AS} \\ ID_v &= g_1^{u_v} \bmod n_{AS} \end{aligned} \tag{15}$$

Finally, the TCS counts the valid tickets and publishes them in the bulletin board to give insurance to voters that their tickets have been counted.

4 Security Analysis of Our Electronic Voting Scheme

In this section, we prove the correctness of our voting system to fulfill the claimed properties. Note that for proving the correctness of the protocol, we assume the difficulty of solving some problems and unforgeability of certifications.

Assumption 1. Factorization of large numbers is a hard problem.

Assumption 2. RSA problem is a hard problem.

Note that the security of the RSA cryptosystem is based on two mathematical problems: the problem of factoring large numbers and the RSA problem.

Assumption 3. Discrete logarithm problem is a hard problem.

Assumption 4. Representation problem is a hard problem.

Lemma 1. A voter has the ability to provide correct values of r_1 and r_2 with respect to d which could pass the verifications of voting and ticket obtaining phase, if and only if he knows a representation of A' and B with respect to g_1 and g_2.

Proof. Suppose that a voter knows a representation of A' and B with respect to g_1 and g_2. Then, he knows the values of u, x_1 and x_2. Consequently, he can compute the values of d, r_1 and r_2 from (12). Conversely, suppose that a voter does not know a representation of A' and B with respect to g_1 and g_2. Then, he does not know anything about u, x_1 and x_2. Consequently, he can not provide valid values for d, s, r_1 and r_2. □

Lemma 2. A voter can use a ticket, if and only if he knows a representation of A' and B with respect to g_1 and g_2.

Proof. According to the previous lemma, a voter knows a representation of A' and B with respect to g_1 and g_2 if and only if he can provide correct values of r_1 and r_2 with respect to d in voting and ticket obtaining phase. Furthermore, a voter can make and use a ticket, if and only if he provides the correct values of d, r_1 and r_2 for his own ticket. □

Theorem 1. The proposed scheme achieves the requirement of eligibility of voters.

Proof. According to the previous lemma, a voter can vote, if and only if he knows a representation of A' and B with respect to g_1 and g_2. Furthermore, before getting the signature of AS on A' and B, eligibility of the voter has been passed by checking the validity of his own certificate by the authentication server. It means that only eligible voters can get a ticket which could pass the voting process. □

Theorem 2. The proposed scheme achieves the requirement of perceptibility of double voters.

Proof. Since the computation of ticket counting server in the third phase of the protocol in the case of double voting clears the identity of the double voter, it is evident that this property is satisfied by the protocol. □

Lemma 3. If a voter follows the protocols and does not double vote, no authority could specify the identity of the voter.

Proof. Note that AS is the only authority which accesses the identification information of each voter during the voting process. Furthermore, it only accesses blinded values of s_2 and s_3 and the value of A. However, since VS and TCS have access to pure values of s_2 and s_3 and blinded values of A, i.e. A', there is no relation between each cast ticket and the information which is given to AS by the voters. Hence, it is impossible to find the identity of voters even by the cooperation of AS, VS and TCS. Furthermore, since the number of unknown parameters are more than the number of equations in (12), it is impossible for TCS to find the owner of tickets. □

Theorem 3. The proposed scheme achieves the requirement of anonymity of voters.

Proof. According to the previous lemma, no one can specify the identity of the honest voter. So, the anonymity of voters holds in the protocol. □

Lemma 4. No voter by himself is able to forge the ticket without detection.

Proof. Suppose that a voter could forge a ticket. Then, the forged ticket is provided by changing in value of one of the signed amounts $s_1 = sign_{AS}(A')$, $s_2 = sign_{AS}(B)$, $s_3 = sign_{AS}(A' + B)$. Since the value of s_3 depends on the values of s_1 and s_2, changing the value of s_3, only, is invaded. Furthermore, since the value of B is optional, forging B is not valuable. So the only way which remains, is forging the signature of s_1 and applying the required changes on s_3. The only way to forge the value of s_1 is using the homomorphic property of RSA cryptosystem. In this case, due to optional value of s in $A' = A^s$, applying this change does not have any value. □

Lemma 5. It is impossible to forge an extra ticket to vote with.

Proof. Similar to the proof of the previous lemma, the only way to forge a ticket is to change its value of A using the homomorphic property. As presented in the previous lemma, a voter, alone, is not able to forge A. So the only way to change the value of A is the cooperation of some malicious voters together to add their own values of u and get the signature of AS on new values of A and $A' + B$ by an eligible voter instead of his own values. However, the forged ticket is identified at the end of the voting process in the case of double voting. □

Theorem 4. The proposed scheme achieves the requirement of unforgeability of tickets.

Proof. By the previous two lemmas, it is impossible to forge an extra ticket beside the tickets of voters. The only leak of the protocol is the one, which has been mentioned in the proof of the previous lemma. However, in this case too, it is impossible to forge an extra ticket. □

5 Efficiency of the Scheme

Table 1 shows the comparison of the number of multiplications, exponentiations and hash functions used in our scheme and Asadpour textitet al.'s scheme.

Table 1. Comparing efficiency of our scheme with Asadpour *et al.*'s scheme.

Schemes	Multiplication	Exponentiation
Asadpour *et al.*'s scheme	30	35
Our scheme	11	21

As it is shown, the proposed voting scheme is more efficient than Asadpour *et al.*'s scheme.

6 Conclusions

In this paper, we considered one of the last voting protocols in the generation of Mu Varadharajan protocol and showed its weaknesses. Furthermore, we contributed an electronic voting which is immune to the weaknesses of the previous works. In order to hide the identity of the voter and detect it in the case of double voting, we contributed a special structure which hides identities and by that we generated a protocol which protects the anonymity of voters, detects the identity of double voter and authenticates eligible voters with more efficiency than the previous one, Asadpour *et al.*'s protocol. The security of the new protocol was also considered.

Acknowledgements

This paper has been supported by Research Deputy Chancellor of Sharif University of Technology.

References

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

[2] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta. A Practical Secret Voting Scheme for Large Scale Elections. In *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques: Advances in Cryptology*, ASIACRYPT '92, pages 244–251. Springer-Verlag, 1993.

[3] Juang Wen-Shenq and Lei Chin-Laung. A Secure and Practical Electronic Voting Scheme for Real World Environments. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 80(1):64–71, 1997.

[4] Byoungcheon Lee, Colin Boyd, Ed Dawson, Kwangjo Kim, Jeongmo Yang, and Seungjae Yoo. Providing Receipt-freeness in Mixnet-based Voting Protocols. In *Proceedings of Information Security and Cryptology (ICISC'03), volume 2971 of LNCS*, pages 245–258. Springer, 2003.

[5] Josh Daniel Cohen Benaloh. *Verifiable Secret-Ballot Elections*. PhD thesis, 1987.

[6] Martin Hirt and Kazue Sako. Efficient Receipt-Free Voting Based on Homomorphic Encryption. In *Proceedings of the 19th International Conference on Theory and Application of Cryptographic Techniques*, EUROCRYPT'00, pages 539–556. Springer-Verlag, 2000.

[7] David Chaum. Secret-Ballot Receipts: True Voter-Verifiable Elections. *IEEE Security and Privacy*, 2(1):38–47, January 2004.

[8] David Chaum, Peter Y. A. Ryan, and Steve A. Schneider. A Practical Voter-Verifiable Election Scheme. In *ESORICS*, pages 118–139, 2005.

[9] Y. Mu and V. Varadharajan. Anonymous Secure E-Voting Over a Network. In *Proceedings of the 14th Annual Computer Security Applications Conference*, pages 293–299, Washington, DC, USA, 1998. IEEE Computer Society.

[10] Taher El Gamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In *Proceedings of CRYPTO 84 on Advances in Cryptology*, pages 10–18. Springer-Verlag, 1985.

[11] Hung-Yu Chien, Jinn-Ke Jan, and Yuh-Min Tseng. Cryptanalysis on Mu–Varadharajan's E-Voting Schemes. *Applied Mathematics and Computation*, 139(2-3):525–530, July 2003.

[12] Iuon-Chang Lina, Min-Shiang Hwangb, and Chin-Chen Chang. Security Enhancement for Anonymous Secure E-Voting over a Network. *Computer Standards and Interfaces*, 25(2):131–139, 2003.

[13] C. Yang, C. Lin, and H. Yang. Improved Anonymous Secure E-Voting over a Network. *Information and Security*, 15(2):181–198, May 2004.

[14] Sheng-Yu Hwang, Hsiang-An Wen, and Tzonelih

Hwang. On the Security Enhancement for Anonymous Secure E-Voting over Computer Network. *Computer Standards & Interfaces*, 27(2):163–168, 2005.

[15] Vahid Jahandideh, Amir S. Mortazavi, Yaser Baseri, and Javad Mohajeri. Cryptanalysis and Security Enhancement on the Generation of Mu-Varadharajan Electronic Voting Protocol. *International Journal of Electronic Governance*, 3(1): 72–84, 2010.

[16] Maryam Rajabzadeh Asaar, Javad Mohajeri, and Mahmoud Salmasizadeh. Another Security Improvement over the Lin *et al.*'s Electronic Voting Scheme. *International Journal of Electronic Security and Digital Forensic*, 1(4):413–422, November 2008. ISSN 1751-911X.

[17] F. Rodríguez-Henríquez, Daniel Ortiz-Arroyo, and Claudia García-Zamora. Yet Another Improvement over the Mu-Varadharajan E-Voting Protocol. *Compututer Standards and Interfaces*, 29(4):471–480, May 2007.

[18] National Institute of Standards and Technology. *FIPS PUB 186-2: Digital Signature Standard (DSS)*. 2000.

[19] Mahdi Asadpour and Rasool Jalili. Double Voting Problem of Some Anonymous E-Voting Schemes. *Journal of Informaion Science and Engineering*, 25(3):895–906, 2009.

[20] Security Arguments for Digital Signatures and Blind Signatures. *Journal of Cryptology*, 13(3): 361–396, 2000.

[21] Niels Ferguson. Single term off-line coins. In *Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology*, EUROCRYPT'93, pages 318–328, Secaucus, NJ, USA, 1994. Springer-Verlag.

Yaser Baseri received his BSc degree in Computer Science from Shahid Beheshti University, Tehran, Iran, in 2005 and MSc Degree in Computer Science from Sharif University of Technology, Tehran, Iran, in 2007. Currently, he is a faculty member at Computer Science and Engeering Department of Payame Noor University. His research interests include formal method, cryptography, and network security.

Amir S. Mortazavi received his BSc degree in Electrical Engineering in 2008 from Tabriz University. He is currently continuing his master level in cryptography field in Sharif University of Technology. His major research interests include information security, cryptographic protocols, and speech processing.

Maryam Rajabzadeh Asaar received her BSc degree in Electrical Engineering from Shahid Bahonar University, Kerman, Iran, in 2004 and her MSc degree in Electrical Engineering from Sharif University of Technology, Tehran, Iran in 2007. She is now a PhD student in Electrical Engineering Department of Sharif University of Technology. Her research interests include design and analysis of cryptographic protocols and network security.

Mohsen Pourpouneh was born in Isfahan, Iran on May 30, 1978. He is currently pursuing his BSc degree in Computer Science in Mathematics Department, at Shahid Beheshti University of Tehran. His research interests include cryptography and steganography.

Javad Mohajeri received the BSc degree from Isfahan University in 1986 and MSc degree from Sharif University of Technology in 1989, both in Mathematics. Since 1990, he has been a faculty member at Electronics Research Center of Sharif University of Technology. His research interests include cryptography and data security. He is author/co-author of over 50 research articles in refereed Journals/Conferences. He is one of the founding members of Iranian Society of Cryptology.