Double Voter Perceptible Blind Signature Based Electronic Voting Protocol

Yaser Baseri[1], Amir S. Mortazavi[2], Maryam Rajabzadeh Asaar[3], Mohsen Pourpouneh[4],Javad Mohajeri[5]

[1,5] *Electronics Research Center,Sharif University of Technology,Tehran, Iran.*
[2,3] *Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran.*
[4] *Department of Mathematics, Shahid Beheshty University, Tehran, Iran.*

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. scheme as one of the latest one and showing its weaknesses, we propose a new voting scheme which is immune to the weaknesses of previous schemes without loosing efficiency. The scheme, is based on a special structure, which directly use the identity of 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.

Key words: Electronic voting, Anonymity of voter, Unforgeability of ticket, Perceptibility of double voting, Security of voting, Blind signature.

1. Introduction

Nowadays, computers are almost everywhere and they are used for many purposes. One of these purposes is electronic voting. Using computer net-

Email address: `yaser_baseri@alum.sharif.ir, sa_mortazavi@ee.sharif.edu, asaar@ee.sharif.ir, m.pourpouneh@mail.sbu.ac.ir, mohajer@sharif.edu` (Yaser Baseri[1], Amir S. Mortazavi[2], Maryam Rajabzadeh Asaar[3], Mohsen Pourpouneh[4],Javad Mohajeri[5])

works and 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 [4], [11], [24], [15], homomorphic encryption based electronic voting schemes [3], [17], and the schemes which use randomization such as the schemes that employ mixnets [6], [7]. 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 [18]. 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 signature [9], 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. [5] showed that Mu-Varadharajan's schemes suffers from 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. [16] proposed an improvement on Mu and Varadharajan's scheme. They improved the weakness that voters could successfully vote more than one without being detected. The proposed scheme did not require any special voting channel and detect double voting effectively. Yang et al. in 2004 [23] proposed another improvement on Mu-Varadharajan's scheme. Although their scheme is resistance against the attacks which has proposed in [5], it can not determine the identity of double voters. In 2005, Hwang et al. [12] represented an attack on Lin et al. 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. protocol. However Hwang et al. scheme had some weakness in fulfilling the claimed properties [13]. Furthermore, Asaar et al. [1] proposed one more scheme based on Lin et al. scheme. Their scheme resists against the attacks which have been proposed in [16]. In 2007 F. Rodriguez-Henriquez et al. [22] proposed another improvement over the Lin et al. scheme. They presented a fully functional RSA/DSA-based e-voting protocol for online elections. They presented a weakness of Lin et al. 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 [8]. These changes guarantee that independently choosing values by the voter and authentication server would not have undesirable effect on the ticket obtaining procedure. In 2010, Jahandideh et al. [13] showed that all of Lin et al. [16], Yang et al. [23], Hwang et al. [12], Rodriguez-Henriquez et al. [22] and Asaar et al. [1] protocols suffer from some weaknesses. One of the latest schemes which have been proposed in this category is Asadpour et al. protocol [2]. 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 suffer from some other weaknesses beside the weaknesses they have counted for 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 occurring double voting 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's anonymity, detection of double voters and prevention of the attacks which have 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 vote 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

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

and Representation problem. Finally, in section 5, we show a comparison between the efficiency of our scheme and Asadpour et al. scheme and show than our proposed scheme is more efficient than those scheme.

2. Asadpour et al. scheme and its failures

In this section first we describe the protocol proposed by Asadpour et al. in subsection 2.1. Then in subsection 2.2 we present some attacks to the protocol.

2.1. Asadpour et al. scheme

The 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.
- p: a large prime number, which is a public system parameter.
- g, h: are 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:

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

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

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

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

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 messages $\{AS, V, w_3, (w_4\|w_5\|w_6\|w_7\|t)^{e_V}\, mod\, n_V\}$ to V and store 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:

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

(d) V applies the ElGamal digital signature scheme [9]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 signature (a_1, s_5) and (a_2, s_6) using the following equations:

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

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:

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

If all of the above equations hold, VS further 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:

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

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 have 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:

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

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

2.2. *Weaknesses of the scheme*

In this subsection, we show that the proposed scheme of Asadpour et al. get affected by some weaknesses. Beside the weakness in fulfilling the property of perceptibility of double voter, which have been mentioned in their paper, the scheme suffer from weaknesses in protecting the anonymity of voters. In this part, we present two attacks and show that the scheme doesn't provide anonymity of honest voter property.

2.2.1. *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 $w_1' = h^r b_1^{e_{AS}} \underline{m}od n_{AS}$, AS is able to compute 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. *Second attack*

After publishing tickets on the bulletin board, AS has access to the information of all tickets. On the other hand, AS have 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 select a record $\{V', k_2'\}$ from its own database and computes the value r' as follows:

$$r' = \frac{s_5 s_6 k_2' - m(2a_1 s_6 - a_2 s_5)}{s_5 - 2s_6} \, mod\, p \tag{8}$$

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

3. The new electronic voting scheme

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 $g_1^{u_v} \, mod \, n_{AS}$.

- 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.

- $\|$: the operation of concatenation.

- t: timestamp.

Note that the used RSA system for AS is based on 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 have been used in some other protocols such as Ferguson electronic cash protocol [10]. 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 signature of the authentication server which is in the ticket and then sends the ballot to the ballot counting server, 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. First phase: voting and ticket obtaining phase

(a) The voter select 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:

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

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

(b) AS first verifies the validity of the certificate, timestamp and value of A by using 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:

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

Finally, the message $\{((w_3 \| w_4 \| w_5 \| t)^{e_V}) \, mod \ n_V\}$ 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:

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

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

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

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. 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 have been forged in the protocol.

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

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

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

3.3. Third phase: tickets counting phase

Upon receiving all tickets from the voting servers, TCS verify if there are double voting has been 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, d', it computes the identity of the voter as by the following equations:

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

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 out 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 nnumbers 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 the 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 equation 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 any things 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 know the 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 voter 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 protocol in the case of double voting clears the identity of double voter, it is trivial 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 voter.*

Proof. Note that AS is the only authority which accesses to the identification information of each voter during the voting process. Furthermore, it only accesses to blinded values of s_2 and s_3 and the value of A, However, since VS and TCS have access to pure value 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 cooperation of AS, VS and TCS. Furthermore, since, the number of unknown parameters are more than the number of equations in the equation 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 honest voter. So, the anonymity of voters is hold 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 is depended on the values of s_1 and s_2, changing the value of s_3, lonely, is invaded. Furthermore, since the value of B is optional, forging of B is not valuable. So the only way reminded, 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, lonely, 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 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 voting process in the case of double voting. □

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

Proof. By previous two lemmas, it is impossible to forge an extra ticket beside tickets of voters. The only leak of the protocol is the one, which has mentioned in the proof of previous lemma. However, also, in this case, 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 which used in our scheme and Asadpour schemes. As it is shown, the proposed voting scheme is more efficient than Asadpour et al. scheme.

Schemes	Multiplication	Exponentiation
Asadpour et al.	30	35
Our Scheme	11	21

Table 1: Comparing efficiency of our scheme with Asadpour et al. schemes

6. Conclusion

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

References

[1] M. Asaar, J. Mohajeri and M. Salmasizadeh. Another security improvment over the Lin et al.'s electronic voting scheme. *International Journal of Electronic Security and Digital Forensics*, 1(4):413–422, 2008.

[2] M. Asadpour and R. Jalili. Double Voting Problem of Some Anonymous E-Voting Schemes. *JOURNAL OF INFORMATION SCIENCE AND ENGINEERING* , 25: 895–906, 2009.

[3] J. Benaloh. Verifiable Secret Ballot Elections. ,Ph.D. Thesis, Yale University, 1987.

[4] D. Chum. Elections with Unconditionally-Secret Ballots and Disruption Equvalnt to Breaking RSA. *In Advances in Cryptology-Eurocrypt'88*, 330:177–182, LNCS, Springer, 1988.

[5] H.Chien, J.Jan and Y.,Tseng. Cryptanalysis on MuVaradharajan's E-voting Schemes. *Appl. Math. Comput* 139 (23):525-530, 2003.

[6] D. Chaum. Secret-Ballot Receipts: True Voter-Verifiable Elections. *IEEE Securityand Privacy* , 2004.

[7] D. Chaum, P. Rayan and S. Schneider. A Practical, Voter-Verifiable Election Scheme. *inProc. 10th European Symposium on Research in Computer Security (ESORICS'05)*, 3679:118–139, LNCS, Springer, 2005.

[8] FIPS 186-2, Digital Signature Standard, National Institute of Standards and Technology(NIST), http://csrc.nist.gov/publications/fips/October 2001.

[9] T. ELGamal. A public-key cryptosystem and a signature scheme based on discrete logarithm. *IEEE Transactions on Information Theory*, 31:469–472, July 1985.

[10] N. Ferguson. Single Term Off-Line Coins. *Workshop on the theory and application of cryptographic techniques on Advances in cryptology* , 318–328, Springer-Verlag, 1994.

[11] A. Fujioka, T. Okamoto and K. Ohta. A Practical Secret Voting Scheme for Large Scale Elections. *Advances in Cryptology-AUSCRYPT '92*, 718:244–251, LNCS, Springer, 1988.

[12] S. Hwang, H. Wen and T. Hwang. On the security enhancement for anonymous secure e-voting over computer network. *Computer Standards and Interfaces*, 27(2):163–168, 2005.

[13] V. Jahandideh, A. Mortazavi, Y. Baseri, J. Mohajeri. Cryptanalysis and security enhancement on the generation of Mu-Varadharajan electronic voting protocol. *International Journal of Electronic Governance*, 3(1):72–84, 2010.

[14] D.Juels, M.Luby and R.Ostrovsky. Security of blind digital signatures. *Proceedings of international conference on the theory and applications of cryptography and information security:Advanced in cryptology*, 1294:150–164, LNCS, Springer, 1997.

[15] B. Lee, C. Boyd, E. Dawson, K. Kim, J. Yang and S. Yoo. Providing Receipt-Freeness in Mixnet-Based Voting Protocols. *Proceedings of Information Security and Cryptology (ICISC'03)*, 2971:245–258, LNCS, Springer, 2004.

[16] I. Lin, M. Hwang, and C. Chang. Security enhancement for anonymous secure e-voting over a network. *Computer Standards and Interfaces*, 25:131–139, 2003.

[17] M. Hirt and K. Sako. Efficient Receipt-Free Voting Based on Homomorphic Encryption. *Advances in Cryptography-Eurocrypt 2000*, 1807:539–556, LNCS, Springer 2000.

[18] Y. Mu and V. Varadharajan. Anonymous secure e-voting over a network. *Proceedings of the 14th Annual Computer Security Application Conference. IEEE Computer Society*, pages 293–299, 1998.

[19] T. Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. *Proceedings of crypto92*, 740:31-53, LNCS, Springer verlog, 1993.

[20] D. Poincheval and J. stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 13:361-395, 2000.

[21] D. Poincheval and J. stern. Provably secure blind signature schemes. *Proceedings of international conference on the theory and applications of cryptography and information security:Advanced in cryptology, Asiacrypt96*, 1163:255-263, LNCS, Springer verlog, 1997.

[22] F. Rodrguez-Henrquez, D. Ortiz-Arroyo and C. Garca-Zamora. Yet another improvement over the Mu-Varadharajan e-voting protocol. *Computer Standards and Interfaces*, 29:471–480, 2007.

[23] C. Yang , C. Lin and H.Yang Improved anonymous secure e-voting over a network. *INFORMATIM & SECURITY*, 15(2):185–191, 2004.

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