Cryptanalysis and Security Enhancement on the Generation of Mu-Varadharajan Electronic Voting Protocol

Vahid Jahandideh[1], Amir S. Mortazavi[1], Yaser Baseri*[2], Javad Mohajeri[2]
[1] *Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran,*
[2] *Electronics Research Center, Sharif University of Technology, Tehran, Iran.*

Abstract

Mu and Varadharajan proposed an electronic voting scheme and claimed that their scheme authenticate the voters, protect the anonymity of them, and detect the identity of double voters. Due to some weaknesses in Mu-Varadharajan scheme, several modified schemes have been proposed by Lin et al., Hwang et al., Rodriguez-Henriquez et al. and Asaar et al.; however this paper shows that these schemes suffer from some weaknesses in fulfilling the pointed properties. For this purpose, we get Hwang et al. scheme as a case study and apply our new attack on it. Also we consider the applicability of the attacks on other pointed schemes. In addition, we present a new scheme and show that the scheme resists against the proposed attacks without loosing efficiency.

Key words:
Anonymity of voter, Unforgeability of ticket, Perceptibility of double voting, Security of voting, Attack prevention.

---

1. Introduction

By the need of democracy in developing human societies and by developing communication networks and internet, it is necessary to produce new

---

*Tel:+98-21-66164906, Fax:+98-21-66030318

Email address: `v_jahandideh@ee.sharif.edu`, `sa_mortazavi@ee.sharif.edu`, `yaser_baseri@alum.sharif.ir`, `mohajer@sharif.edu` (Vahid Jahandideh[1], Amir S. Mortazavi[1], Yaser Baseri[2], Javad Mohajeri[2])

Preprint submitted to Applied Mathematics and Computation *August 30, 2009*

secure e-voting schemes that satisfy some requirements such as anonymity of voters, unforgeability of tickets and perceptibility of double voters.

Electronic voting protocols could be classified by their approach into three categories: 1) blind signature based electronic voting schemes [1], [2], [3], [4]; 2) homomorphic encryption based electronic voting schemes [5], [6]; 3) electronic voting schemes which use randomization such as the schemes which employ mixnets [7], [8]. One of the first voting protocols which is based on blind signature and used to claim that it could protect voter's anonymity, authenticate the voters and detect the identity of double voters, is proposed by Mu and Varadharajan [9]. They also claimed that their scheme is suitable for large scale elections. In 2003, Chien et al. [10] showed that there are some weaknesses in security of Mu-Varadharajan's schemes such as identifying the owner of a cast ballot by authorities, imperceptibility of double voting and forgeablity of a ballot by anyone without being authenticated. In 2003 Lin et al. [11] proposed an improvement on Mu and Varadharajan's protocol. They have tried to improve the weakness that voters could successfully vote more than once without being detected. Their proposed scheme does not require any special voting channel and it is claimed that the scheme is able to detect vote duplicity effectively. Yang et al. in 2004 [12] proposed another improvement on Mu and Varadharajan's protocol. Although their scheme is resistant to the attacks which has been proposed in [10], it can not determine the identity of double voter. Hwang et al. in 2005 represented an attack on Lin et al. protocol in [13]. They also showed that the modification of Lin et al. allows the authentication server to identify the voters of published tickets so that voters will lose their privacy. Hwang proposed a new scheme to solve this problem and thus enhance the security [13]. It was claimed that the protocol satisfies the privacy of voters and detects double voting. In 2008 Asaar et al. [14] showed that the Hwang et al. scheme has the weakness that eligible voter with a valid ticket could vote more than once without being detected. They also proposed a new improvement on it. However it is possible to show that their improvement has the same weaknesses as the weaknesses which are proposed in this paper. Furthermore in 2007 F. Rodriguez-Henriquez et al. [15] proposed another improvement over the Mu and Varadharajan e-voting protocol which suffers from the same problem as Hwang et al. scheme. Moreover, in 2008 Asaar et al. [16] proposed another scheme based on Lin et al. scheme. Although, their scheme resist against the proposed attacks in [10], it is vulnerable to some attacks which are presented in section 2.2.

In this paper first we review Hwang et al. scheme and its failures in section 2. Then in section 3 we propose a new modified scheme to solve the weaknesses of the former schemes and show that the new scheme won't be affected by the proposed attack without losing efficiency and requirement of the former one. The conclusion is presented in section 4.

2. Hwang et al. scheme and its failures

In this section first we describe the protocol proposed by Hwang et al. in subsection 2.1. Then in subsection 2.2 we present some attacks on anonymity of voter, unforgeability of ticket and perceptibility of double vote in this scheme. The applicability of these attacks to some other schemes on the race of Mu-Varadharajan scheme are parented.

2.1. Hwang et al. scheme

The Hwang et al.'s anonymous electronic voting scheme consists of the following participants: Voters (V), an Authentication Server (AS), Voting Servers (VS), a Ticket Counting Server (TCS), and a Certificate Authority (CA). For convenience, some necessary notation are defined below:

- $(e_x, n_x), d_x$: 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.

2.1.1. The voting and ticket obtaining phase

(a) A voter V chooses two blind factors b_1, b_2 in $\mathbb{Z}_{n_{AS}}^*$ and two random numbers k_1 and r in $\mathbb{Z}_p^*$. Then, V computes w_1, w_1', w_2 and w_2' as follow:

$$\begin{aligned} w_1 &= g^r b_1^{e_{AS}} \bmod n_{AS} \\ w_1' &= h^r b_1^{e_{AS}} \bmod n_{AS} \\ w_2 &= g^{k_1} b_2^{e_{AS}} \bmod n_{AS} \\ w_2' &= h^{k_1} b_2^{e_{AS}} \bmod n_{AS} \end{aligned} \tag{1}$$

Then, 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 first verifies the validity of the certificate and timestamp, then validate the signature $((w_1\|w_1'\|w_2\|w_2'\|t)^{d_V}) mod\, n_V$. After passing all verifications, AS chooses a unique random number k_2 and computes:

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

Where $a_1 = g^r$, $a_2 = h^r$, $y_1 = g^{k_1+k_2}$, and $y_2 = h^{2k_1+k_2}$. Then, the messages $\{AS, V, w_3, (w_4\|w_5\|w_6\|w_7\|t)^{e_V}\, modn_V\}$ are delivered to V. Note that AS also stores k_2 along with V's identity in it's database.

(c) V obtains k_2 by decrypting w_3. Thus, V can calculate y_1 and y_2 by using g, h, k_1 and k_2. Furthermore, V also computes the signatures s_1, s_2, s_3 and s_4 by removing the blinding factors b_1 and b_2 from w_4, w_5, w_6 and w_7 as follow:

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

(d) V applies the ElGamal digital signature scheme [17]to sign the voting content m. Let $x_1 = k_1 + k_2$ and $x_2 = 2k_1 + k_2$ be the private keys of ElGamal digital signature and y_1, y_2 be the corresponding public keys such that $y_1 = g^{k_1+k_2} mod\, p$ and $y_2 = h^{2k_1+k_2} mod\, p$. V generates two signature (a_1, s_5) and (a_2, s_6) of the voting content m by using the following equations:

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

Then 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:

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

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:

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

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 to 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. Moreover, if the voter uses the same parameters to sign different voting contents, TCS and AS can cooperate to find the malicious voter as follows. Assume that TCS discovers a voter using the same parameters y_1, y_2, a_1 and a_2 to sign two different voting contents m and m'. Then TCS can calculate

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

Finally, TCS can identify the malicious voter by searching AS's database to find out which voter is associated with the unique random number k_2. Then the TCS publishes the valid tickets and counts them.

2.2. Weaknesses of this scheme

In this subsection we present some attacks on Hwang et al. scheme and show that the scheme is vulnerable to the claimed properties (i.e. anonymity of voter, unforgeability of tickets and perceptibility of double voter). Furthermore applicability of these attacks on other schemes of the family of Mu-Varadharajan scheme are shown in table 1.

2.2.1. The attacks on anonymity of voter

First attack. In this scheme since parameters w_1 and w'_1 are blinded with the same blinding factor, when 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, AS can compute the value $z_1 = \frac{w_1}{w'_1} = \frac{g^r}{h^r}$ for the voter and store it beside the identity of him. On the other hand at the end of voting process, when TCS publishes tickets AS can compute the value $z_2 = \frac{a_1}{a_2} = \frac{g^r}{h^r}$ for each ticket which is published by ticket counting server. By matching the values of z_1 and z_2, AS can determine the owner of each vote m.

Since, Assar et al. [14] has similar structure to this scheme, the attack can also be employed on theie scheme. However, since Lin et al. scheme [11], Assar et al. scheme [16] and Rodriguez-Henriquez et al.[15], have used different blinding factors in different equations, computing the value similar to z_1 is impossible and this attack can not be employed on them.

Second attack. Suppose that TCS has published all cast tickets. In order to find 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 can perform the following procedure to identify the owner of this ticket:

1. AS select a record $\{V', k'_2\}$ from its own database and computes the value r' as follow:
$$r' = \frac{s_5 s_6 k'_2 - m(2a_1 s_6 - a_2 s_5)}{s_5 - 2s_6} \tag{8}$$
If the equation $a_1 \overset{?}{=} g^{r'}$ holds then $r' = r$ and V' is the owner of this ticket; else AS chooses another record from its own data base and compute the equation (8) while the owner of this vote is determined.
2. This step is done while the owner of all tickets are determined.

This attack can be employed on [11], [16], [13], [14] and [15] in similar ways.

2.2.2. The attacks on unforgeability of tickets

First attack. Hwang et al. scheme has weakness with respect to a kind of multiplicative attack. With this attack a malicious voter by a valid ticket can make forged tickets as much as he desires without detection. After ticket obtaining phase, malicious voter choose four numbers δ_1, δ_2, δ_3, δ_4 arbitrary and compute the value of a'_1, a'_2, y'_1, y'_2, s'_1, s'_2, s'_3, s'_4 as forged duplicate values of a_1, a_2, y_1, y_2, s_1, s_2, s_3, s_4 by the following procedure:

$$\begin{aligned}
a'_1 &= a_1 \times g^{\delta_1 . e_{AS}} \, mod \, n_{AS} \\
a'_2 &= a_2 \times h^{\delta_2 . e_{AS}} \, mod \, n_{AS} \\
y'_1 &= y_1 \times g^{\delta_3 . e_{AS}} \, mod \, n_{AS} \\
y'_2 &= y_2 \times h^{\delta_4 . e_{AS}} \, mod \, n_{AS} \\
s'_1 &= s_1 \times g^{\delta_1} \, mod \, n_{AS} \\
s'_2 &= s_2 \times h^{\delta_2} \, mod \, n_{AS} \\
s'_3 &= s_3 \times g^{\delta_3} \, mod \, n_{AS} \\
s'_4 &= s_4 \times h^{\delta_4} \, mod \, n_{AS} \\
x'_1 &= x_1 + \delta_3 . e_{AS} \\
x'_2 &= x_2 + \delta_4 . e_{AS} \\
s'_5 &= {x'_1}^{-1} \times (m a'_1 - (r + \delta_1 . e_{AS})) \, mod \, (p-1) \\
s'_6 &= {x'_2}^{-1} \times (m a'_2 - (r + \delta_2 . e_{AS})) \, mod \, (p-1)
\end{aligned} \tag{9}$$

Finally the forged duplicate ticket would be $T' = \{s'_1 \parallel s'_2 \parallel s'_3 \parallel s'_4 \parallel s'_5 \parallel s'_6 \parallel a'_1 \parallel a'_2 \parallel y'_1 \parallel y'_2 \parallel m\}$. This ticket is passed through all validations which will be done:

$$\begin{gathered}
\begin{cases} AS \times a_1 = s_1^{e_{AS}} \, mod \, n_{AS} \Rightarrow \\ AS \times a'_1 = {s'_1}^{e_{AS}} \, mod \, n_{AS} \end{cases} \\
\begin{cases} AS \times a_2 = s_2^{e_{AS}} \, mod \, n_{AS} \Rightarrow \\ AS \times a'_2 = {s'_2}^{e_{AS}} \, mod \, n_{AS} \end{cases} \\
\begin{cases} AS \times y_1 = s_3^{e_{AS}} \, mod \, n_{AS} \Rightarrow \\ AS \times y'_1 = {s'_3}^{e_{AS}} \, mod \, n_{AS} \end{cases} \\
\begin{cases} AS \times y_2 = s_4^{e_{AS}} \, mod \, n_{AS} \Rightarrow \\ AS \times y'_2 = {s'_4}^{e_{AS}} \, mod \, n_{AS} \end{cases}
\end{gathered} \tag{10}$$

This attack can be employed on [14] in a similar way. Furthermore by using one generator g instead of two generator g and h, this attack can be employed on [11], [16] and [13] in a similar way. However since in [15] the computations are done in different interconnected modules, while computation of multiplications, we can not forge new valid ticket similar to this scheme.

Second attack. Two malicious voters whose tickets are $T = \{s_1 \parallel s_2 \parallel s_3 \parallel s_4 \parallel s_5 \parallel s_6 \parallel a_1 \parallel a_2 \parallel y_1 \parallel y_2 \parallel m\}$ and $T' = \{s'_1 \parallel s'_2 \parallel s'_3 \parallel s'_4 \parallel s'_5 \parallel s'_6 \parallel a'_1 \parallel a'_2 \parallel y'_1 \parallel y'_2 \parallel m'\}$ can collude and make the forged 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 follow:

$$\begin{array}{l}
s''_1 = (\frac{s_1}{s'_1})^{c_1} \times s_1 \, mod \, n_{AS} \\
s''_2 = (\frac{s_2}{s'_2})^{c_2} \times s_2 \, mod \, n_{AS} \\
s''_3 = (\frac{s_3}{s'_3})^{c_3} \times s_3 \, mod \, n_{AS} \\
s''_4 = (\frac{s_4}{s'_4})^{c_4} \times s_4 \, mod \, n_{AS} \\
a''_1 = g^{c_1(r-r')+r} \\
a''_2 = h^{c_2(r-r')+r} \\
y''_1 = g^{c_3(k_1+k_2-k'_1-k'_2)+k_1+k_2} \\
y''_2 = h^{c_4(k_1+2k_2-k'_1-2k'_2)+k_1+2k_2}
\end{array} \tag{11}$$

This thicket is passed through the validation of the protocol.

In [15] since the computations are done in different modules, while computation of divisions,we can not forge new valid ticket similar to this scheme. However by using one generator g instead of two generator g and h, this attack can be employed on [11], [16], [13], and [15] in a similar way. Furthermore this attack can be obtained on [14].

Third attack. A malicious voter can exchange the values of a_1, s_1 with y_1, s_3 respectively and compute new values for s_5 as $s_5 = r^{-1}.(my_1 - (k_1 + k_2)) \, mod \, (p-1)$. By these changes he will be able to produce new forged ticket which passes the validations of the protocol.
In a similar way by exchanging the values of a_2, s_2 with y_2, s_4 respectively and computing new value for s_6 as $s_6 = r^{-1}.(my_2 - (2k_1 + k_2)) \, mod \, (p-1)$ he can produce another forged ticket.
Also by simultaneously applying the mentioned changes he can produce another forged ticket.

Moreover since the exchanging the sequence of the parameters in a valid ticket, is not checked in [14], [11], [16], [13], and [15], This attack can be employed on them in a similar way.

2.2.3. The attack on perceptibility of double voter

In the scheme, it is supposed that the voter is trusted to choose exponents in the form described in equation (1). However, a malicious voter can deviate from this form, choose other value for k_1 such as k'_1 and compute the value of $w'_2 = h^{k'_1} b_2^{e_{AS}}$ and fulfill the process of acquiring a valid ticket. In this situation, if a malicious voter votes more than once, TCS will not be able to determine the identity of him.

In Hwang scheme since the voter is trusted to choose exponent on the generators g and h in his share of the keys in the prescribed manner, this attack is applicable on it. The similar situations are held in [14]. However since the schemes [11], [16], [13], and [15] use only one generator g instead of two generators g and h, and consequently there is no assumption on the exponents of the generators which the voters must be trusted to obey them. So the attack is not applicable on them.

Schemes	**Attacks on anonymity**		**Attacks on unforgeability**			**Attacks on perceptibility**
	First attack	Second attack	First attack	Second attack	Third attack	First attack
Line et al. [11]	×	✓	✓	✓	✓	×
Asaar et al. [16]	×	✓	✓	✓	✓	×
Hwang et al. [13]	✓	✓	✓	✓	✓	✓
Asaar et al. [14]	✓	✓	✓	✓	✓	✓
Rodriguez-Henriquez et al. [15]	×	✓	×	×	✓	×

Table 1: Comparing resistance of different schemes on the race of Mu-Varadharajan scheme against the proposed attacks(✓: applicable, ×: unapplicable)

3. The new electronic voting scheme

In the new scheme in order to get vulnerability to the first and second attacks on unforgeability of tickets, multiplicative structure of the keys in the signatures is replaced by additive structure and the generators g and h are chosen in a way that the discrete logarithm of them be unknown. Furthermore, in previous schemes the malicious voter can choose exponents in such way that no authority could trace him if double voting occurs. In order to protect the scheme against the third attack on unforgeability of tickets, we allowed the voter to choose exponent as he desire and by introducing new key structure, the perceptibility of double voting is satisfied. In addition, in the new scheme by effectively blinding the signatures of AS and increasing the number of unknown parameters, our new scheme resists the attacks on anonymity of voter.

3.1. Our proposed scheme

Our proposed scheme also has composed of three phases of the former scheme. Before starting the protocol, voters must get their certificate from certificate authority. Let $\mathbb{Z}_p^*$ be a multiplicative group such that solving discrete logarithm problem in it is hard. Furthermore suppose that q be a large prime number such that $q|p-1$. In our scheme the certificate authority chooses a generator g of order q for multiplicative group $\mathbb{Z}_p^*$ and stores it beside the other information in voter's certificate.

3.1.1. Voting and ticket obtaining phase

Before starting this phase certificate authority chooses a generator h of order q in $\mathbb{Z}_p^*$ for authentication server and publishes it. Then the following steps are done:

(a) A voter V chooses three blinding factors b_1, b_2 and b_3 as well as two random numbers r_1 and r_2 such that $r_1, r_2 < q/2$. Then, V computes w_1, w_2 and w_3 by the following equations:

$$\begin{aligned} w_1 &= (g^{r_1} + h^{r_2}).b_1^{e_{AS}} \, mod \, n_{AS} \\ w_2 &= g^{r_1} b_2^{e_{AS}} \, mod \, n_{AS} \\ w_3 &= h^{r_2} b_3^{e_{AS}} \, mod \, n_{AS} \end{aligned} \tag{12}$$

After that, the voter sends $\{V, AS, Cert_V, t, w_1, w_2, w_3, ((w_1\|w_2\|w_3\|t)^{d_V}) \, mod \, n_V\}$ to AS.

(b) AS first verifies the validity of the timestamp t and the certificate $Cert_V$ and then by using the certificate it verify the signature $((w_1\|w_2\|w_3\|t)^{d_V}) \, mod \, n_V\}$. If all verifications are successful, AS chooses $k_i, k_j \in \mathbb{Z}_q^*$ such that $k_i, k_j < q/2$ randomly and computes $k_2 = k_i + k_j$ in a way that k_2 is a fresh value and has not been already used and computes:

$$\begin{aligned} w_4 \;&= (k_i\|k_j\|t)^{e_v} \, mod \, n_v \\ w_5 \;&= (AS \times (w_1))^{d_{AS}} \, mod \, n_{AS} \\ &= (AS \times (a_1 + a_2))^{d_{AS}} \times b_1 \, mod_{AS} \\ w_6 \;&= (AS \times (w_2 w_3 + w_2 w_3.g^{k_i} h^{k_j}))^{d_{AS}} \, mod \, n_{AS} \\ &= (AS \times (a_1 a_2 + y_1 y_2))^{d_{AS}}.b_2 b_3, mod_{AS} \end{aligned} \tag{13}$$

Where $a_1 = g^{r_1}$, $a_2 = h^{r_2}$, $y_1 = g^{r_1+k_i}$, and $y_2 = h^{r_2+k_j}$. Then, AS delivers the messages $\{AS, V, w_4, t, (w_5\|w_6\|t)^{e_V} \, mod \, n_V\}$ to V. Note that AS also records k_2 along with V's identity in it's database.

(c) V decrypts w_4 to obtain k_i and k_j. Thus, V can calculate y_1 and y_2

by multiplying a_1 and a_2 with g^{k_i} and h^{k_j} respectively. In addition, V also computes the signatures s_1, s_2, and s_3 by the following equations:

$$\begin{aligned} s_1 &= w_5 \times b_1^{-1} \, mod \, n_{AS} \\ &= (AS \times (a_1 + a_2))^{d_{AS}} \; mod \, n_{AS} \\ s_2 &= w_6 \times b_2^{-1} b_3^{-1} \, mod \, n_{AS} \\ &= (AS \times (a_1 a_2 + y_1 y_2))^{d_{AS}} \; mod \, n_{AS} \end{aligned} \tag{14}$$

(d) V 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 $x_1 = r_1 + k_i$ and $x_2 = r_2 + k_j$ be the corresponding private keys, such that $y_1 = g^{x_1} mod \, p$ and $y_2 = h^{x_2} mod \, p$. V generates two signatures (a_1, s_5) and (a_2, s_6) of the voting content m by using the following equations:

$$\begin{aligned} s_3 &= x_1^{-1}(ma_1 - r_1) \, mod \, q \\ s_4 &= x_2^{-1}(ma_2 - r_2) \, mod \, q \end{aligned} \tag{15}$$

Then the voting ticket can be computed as

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

Note that due to condition imposed on r_1, r_2, k_i and k_j, we are sure that x_1^{-1} and x_2^{-1} exist.

3.1.2. The voting and tickets collecting phase

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

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

$$\begin{aligned} AS \times (a_1 + a_2) &\overset{?}{=} s_1^{e_{AS}} mod \, n_{AS} \\ AS \times (a_1 a_2 + y_1 y_2) &\overset{?}{=} s_2^{e_{AS}} mod \, n_{AS} \end{aligned} \tag{16}$$

If the above equations hold, VS further verifies the signatures (a_1, y_1, s_3) and (a_2, y_2, s_4) of the voting content m by checking the following equations:

$$\begin{aligned} y_1^{s_3} a_1 &\overset{?}{=} g^{ma_1} mod \, p \\ y_2^{s_4} a_2 &\overset{?}{=} h^{ma_2} mod \, p \end{aligned} \tag{17}$$

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

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

3.1.3. The tickets counting phase

Upon receiving all tickets from the Voting Servers, TCS first verifies if there are double voting tickets by checking if the values of a_1, a_2, y_1, y_2 of one thicket has not occurred in another thicket in one of the following form:

$$\begin{aligned} T' = \; & \{s'_1 \parallel s'_2 \parallel s'_3 \parallel s'_4 \parallel a_1 \parallel a_2 \parallel y_1 \parallel y_2 \parallel m'\} \\ & \{s'_1 \parallel s'_2 \parallel s'_3 \parallel s'_4 \parallel y_1 \parallel a_2 \parallel a_1 \parallel y_2 \parallel m'\} \\ & \{s'_1 \parallel s'_2 \parallel s'_3 \parallel s'_4 \parallel a_1 \parallel y_2 \parallel y_1 \parallel a_2 \parallel m'\} \\ & \{s'_1 \parallel s'_2 \parallel s'_3 \parallel s'_4 \parallel y_1 \parallel y_2 \parallel a_1 \parallel a_2 \parallel m'\} \end{aligned} \tag{18}$$

if one of these forms occurs in another thicket, double voting occurs and TCS can compute the value of k_2 corresponding to the malicious voter as follows:

$$\begin{aligned} & x_1 = \tfrac{m'a_1 - ma_1}{s'_3 - s_3} \, mod \, q \\ & x_2 = \tfrac{m'a_2 - ma_2}{s'_4 - s_4} \, mod \, q \\ & r_1 = ma_1 - s_3 x_1 \, mod \, q \\ & r_2 = ma_2 - s_4 x_2 \, mod \, q \\ & k_i = |x_1 - r_1| \, mod \, q \\ & k_j = |x_2 - r_2| \, mod \, q \\ & k_2 = k_i + k_j \, mod \, q \end{aligned} \tag{19}$$

in which $|X|$ means that:

$$|X| = \begin{cases} X \, mod \, q & if \, X < q/2 \\ q - X \, mod \, q & other \, wise \end{cases}$$

Then, TCS can identify the malicious voter by cooperating with AS and searching AS's database to find out which voter is associated with the unique random number k_2.

3.2. Analysis of the new scheme

3.2.1. Security analysis of the scheme

The problem of previous works is their multiplicative structure that causes them vulnerable to multiplicative attack which is presented in this paper. In our scheme by replacing multiplicative structure of the keys in the signatures by additive structure and choosing g and h by certificate authority, in a way that no one knows the discrete logarithm of them, we could make our scheme resistant against the first and second attacks on unforgeability of tickets.

Since k_i and k_j must be less than $q/2$ for all voters, by the help introducing the operation $|X|$, we can obtain the identity of malicious voters who employ the third attack on unforgeability of tickets. On the other hand in previous schemes the voter was trusted to choose exponent in his share of the keys in the prescribed manner. However malicious voter can choose exponents in such way that no authority can trace him if double voting occurs. In order to preclude this attack we allowed the voter to choose exponent as he desires and by introducing new key structure the perceptibility of double voting claimed in previous schemes satisfied. The other essential future of the e-voting is protecting anonymity of voters. In this paper it is showed that anonymity of voter can easily be removed without double voting. However in the new scheme by effectively blinding the signatures of AS and increasing the number of unknown parameters, the scheme is resistant against the proposed attacks.

3.2.2. Efficiency of the new scheme

By comparing number of multiplications and number of exponentiations we found that our scheme is more efficient than Hwang scheme. Table 2 expresses the comparison between our scheme and Hwang scheme.

Schemes	**Multiplication**			**Exponentiation**		
	Phase1	Phase2	Phase3	Phase1	Phase2	Phase3
Hwang et al.	18	8	6	22	8	0
Our Scheme	16	8	10	18	6	0

Table 2: Comparing efficiency of Hwang scheme and our scheme

4. Conclusion

In this paper we discussed the weaknesses of Hwang scheme in satisfying the properties of unforgeability of the tickets and perceptibility of double voter and anonymity of voters. We mentioned the applicability of these attacks on other scheme in the generation of Mu-Varadharajan protocols. Furthermore we proposed a new scheme and showed that this scheme beside the resistance against the attack which have been proposed until now, satisfy the anonymity and unforgeability of the thicket.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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.

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

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

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

[8] 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.

[9] 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.

[10] H.Chien, J.Jan and Y.Tseng. Cryptanalysis on Mu-Varadharajan's E-voting Schemes. *Appl. Math. Comput* 139 (2-3):525–530, 2003.

[11] 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.

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

[13] 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.

[14] M. Asaar, J. Mohajeri and M. Salmasizadeh. Security modification for the Hwang-Wen-Hwang 's e-voting scheme. *Proceedings of International Conference on Security and Management (SAM'08)*, 139:486–490, 2008.

[15] 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.

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

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