Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 741031, 6 pages
http://dx.doi.org/10.1155/2015/741031

Research Article

Electronic Voting Protocol Using Identity-Based Cryptography

Gina Gallegos-Garcia[1] and Horacio Tapia-Recillas[2]

[1]*Sección de Estudios de Posgrado e Investigación, Escuela Superior de Ingeniería Mecánica y Eléctrica, Instituto Politécnico Nacional, Avenida Santa Ana 1000, San Francisco, Culhuacan, Coyoacan, 04430 México City, DF, Mexico*
[2]*Departamento de Matematicas, Universidad Autónoma Metropolitana Iztapalapa, San Rafael Atlixco 186, Vicentina, Iztapalapa, 09340 México City, DF, Mexico*

Correspondence should be addressed to Gina Gallegos-Garcia; gganig@hotmail.com

Received 9 February 2015; Revised 4 May 2015; Accepted 6 May 2015

Academic Editor: Ting-Yi Chang

Electronic voting protocols proposed to date meet their properties based on Public Key Cryptography (PKC), which offers high flexibility through key agreement protocols and authentication mechanisms. However, when PKC is used, it is necessary to implement Certification Authority (CA) to provide certificates which bind public keys to entities and enable verification of such public key bindings. Consequently, the components of the protocol increase notably. An alternative is to use Identity-Based Encryption (IBE). With this kind of cryptography, it is possible to have all the benefits offered by PKC, without neither the need of certificates nor all the core components of a Public Key Infrastructure (PKI). Considering the aforementioned, in this paper we propose an electronic voting protocol, which meets the privacy and robustness properties by using bilinear maps.

1. Introduction

Since 1964, considerable efforts have been made to improve the efficiency of election processes that has brought, as a consequence, a wide range of proposals on such topic.

Electronic voting has been mentioned in different media as the use of computers or computerized voting equipment to cast ballots in an election, which nowadays are a reasonable alternative to conventional elections and other opinion expressing processes [1–5]. Roughly speaking an electronic voting protocol, used to develop an electronic voting process, involves three main entities: voters, registration authorities, and counting authorities who interact with each other during four main phases: registration, authentication, voting, and counting [6, 7], from which authentication is out of our scope.

In order to use an electronic voting protocol inside an electronic voting process, it should satisfy several properties [8]. However, proposed protocol meets privacy and robustness properties by using bilinear maps.

(i) Privacy: a vote must be kept secret from any coalition of authorities.

(ii) Robustness: the protocol can be developed even if there are entities who do not give correct information. In other words, this property is against dishonest users.

In this paper a voting protocol based on bilinear maps [9, 10] satisfying privacy, uncoercibility, and robustness is proposed. The paper is organized as follows: in Section 2 some intractable problems on finite groups are recalled. The security of the proposed protocol is based on these intractable problems. In Section 3 the proposed protocol is presented. An analysis of privacy and robustness properties is given in Section 4. Obtained results are showed in Section 5. Section 6 presents concluding remarks and final references are listed.

2. Preliminaries

Let $(G_1, +)$ be a cyclic group of order m written additively. With such a group G_1, the following hard cryptographic problems are defined:

(i) Discrete Logarithm Problem (DLP): given $P, P' \in G_1$, find an integer n such that $P = nP'$ whenever such integer exists.

(ii) Computational Diffie-Hellman Problem (CDHP): given a triple $P, aP, bP \in G_1$ for $a, b \in \mathbb{Z}_m$, find the element $(ab)P$.

(iii) Decision Diffie-Hellman Problem (DDHP): given a quadruple $P, aP, bP, cP \in G_1$ for $a, b, c \in \mathbb{Z}_m$, decide whether $c \equiv ab (\mathrm{mod}\ m)$ or not.

We assume throughout the paper that DLP and CDHP are intractable, which means that there does not exist a Polynomial Time Algorithm to solve them with nonnegligible probability. When the DDHP is easy but the CDHP is hard on the group G_1, G_1 is called a Gap Diffie-Hellman (GDH) group. Such a group can be found on supersingular elliptic curves or hyperelliptic curves over finite fields [11, 12]. The proposed electronic voting protocol can be built on any GDH group.

3. The Proposed Electronic Voting Protocol

The protocol is divided into three phases: setup, voting, and counting. In the setup stage the key pairs to be used during the voting and counting phases are generated. The generation of these key pairs involves the participation of n entities E_i, where $1 \le i \le n$ [12–14]. Each entity broadcasts and receives specific information by using Shamir's secret-sharing scheme in order to generate its public and private shares [15]. In the voting phase voters encrypt votes and ask a blind signature [13, 14]. In the counting phase, a Combining Entity reconstructs the signatures of the votes and verifies and decrypts them [13, 14, 16, 17].

The Combining Entity, who does not have any private key, decrypts the votes by combining decryption shares, which are generated by each entity E_i, after which the votes are counted and the tally is published.

The three phases are detailed as follows.

3.1. Setup Phase

(1) Let $(G_1, +)$ and $(G_2, *)$ be cyclic groups of the same order q which is assumed to be a prime number, with $G_1 = \langle P_1 \rangle$, and let $\hat{e} : G_1 \times G_1 \to G_2$ be a nondegenerated bilinear mapping. Let $H_1 : \{0,1\}^* \to G_1$ and $H_2 : G_2 \to \{0,1\}^n$ be two hash functions. This information is known to all entities E_i, $i = 1, 2, \ldots, n$, where $n \le q-1$. Furthermore, each entity E_i chooses a binary string, an element of $\{0,1\}^k$, corresponding to information identifying this entity, for example, an e-mail address, an IP address, and telephone number. The entity E_i sends information to each E_j to generate the public encryption key P_{pub} and its respective private decryption key d as follows:

(a) Entity E_i randomly selects $a_{i0} \in \mathbb{Z}_q^*$, keeps it in secret, and broadcasts $a_{i0}P_1$.

(b) Entity E_i randomly picks up a polynomial $f_i(x) = a_{i0} + a_{i1}x + \cdots + a_{it-1}x^{t-1} \in \mathbb{Z}_q[x]$ of degree $\le t-1$ such that $f_i(0) = a_{i0}$. The integer t is taken sufficiently large.

(c) E_i computes and broadcasts $a_{ij}P_1$ for $j = 1, 2, \ldots, t-1$ and sends $f_i(j)$ to each E_j for $j = 1, 2, \ldots, n$, where $j \ne i$.

(2) After E_i receives $f_j(i)$ from entity E_j, $j = 1, 2, \ldots, n$, $j \ne i$, it does the following:

(a) E_i verifies $f_j(i)P_1$ by checking that $f_j(i)P_1 = (\sum_{k=0}^{t-1} i^k a_{jk})P_1$, for each $j = 1, 2, \ldots, n$, $j \ne i$. If the check fails, E_i broadcasts a complaint against E_j.

(b) It computes its private share $d_i = \sum_{k=1}^{n} f_k(i)$ and keeps it in secret. This d_i may be considered as an element of $\mathbb{Z}_q$.

Each E_i calculates its public share $P_{\mathrm{pub}_i} = d_iP_1 \in G_1$ and computes the public encryption key $P_{\mathrm{pub}} = \sum_{i=1}^{n} a_{i0}P_1 \in G_1$.

(3) With the above calculations, the public key is $P_{\mathrm{pub}} = dP_1$ and its respective private key, that is distributed to every entity E_i, is $d = \sum_{i=1}^{n} a_{i0}$.

(4) Let ID be the binary sequence identifying the receiver, also called Combining Entity, and let $P_{\mathrm{pub}_{\mathrm{ID}}} = H_1(\mathrm{ID}) \in G_1$; all entities E_i compute their private encryption private share $d_{\mathrm{ID}_i} = a_{i0}P_{\mathrm{pub}_{\mathrm{ID}}}$.

(5) In order to generate the signature and verification key pair, each entity E_i sends the following information to each E_j. This is done by using the same (additive) group $G_1 = \langle P_1 \rangle$ as follows:

(a) Entity E_i randomly selects $b_{i0} \in \mathbb{Z}_q^*$, keeps it in secret, and broadcasts $b_{i0}P_1$.

(b) It picks up randomly a polynomial $g_i(x) = b_{i0} + b_{i1}x + \cdots + b_{it-1}x^{t-1} \in \mathbb{Z}_q[x]$ of degree $\le t-1$ such that $g_i(0) = b_{i0}$. Note that the polynomials $f_i(x), g_i(x)$, despite having the same degree, are different.

(c) It computes and broadcasts $b_{ij}P_1$ and sends $g_i(j)$ to each E_j for $j = 1, 2, \ldots, n$, $j \ne i$.

(6) After E_i receives $g_j(i)$ from entity E_j, $j = 1, 2, \ldots, n$, $j \ne i$, it does the following:

(a) E_i verifies $g_j(i)P_1$ by checking that $g_j(i)P_1 = (\sum_{k=0}^{t-1} i^k b_{jk})P_1$. If the check fails, E_i broadcasts a complaint against E_j.

(b) It computes its private share $s_i = \sum_{k=1}^{n} g_k(i)$ and keeps it in secret. The element s_i can be regarded as an element of $\mathbb{Z}_q$.

(c) Then, each E_j calculates its public share $Q_i = s_iP_1$ and computes the public verification key $Q = (\sum_{i=1}^{n} b_{i0})P_1$.

(7) With the above calculations, the public key is $Q = (\sum_{i=1}^{n} b_{i0})P_1$; it means that $Q = sP_1$ and its respective private key that is distributed to every entity E_i is $s = \sum_{i=1}^{n} b_{i0}$.

3.2. Voting Phase

(1) Let $\hat{e} : G_1 \times G_1 \rightarrow G_2$ be the bilinear pairing mentioned above. To encrypt a vote as a message, the voter chooses an option v and selects $0 \neq r \in \mathbb{Z}_q$. Then, it codifies v as an element of $\{0,1\}^n$. After that, the voter selects any $P_{\text{pub}_{\text{ID}}}$ and computes one scalar multiplication and one bilinear pairing obtaining the encrypted vote given by (U, W), where $U = rP_1 \in G_1$ and $W = v \oplus H_2(\hat{e}(P_{\text{pub}}, P_{\text{pub}_{\text{ID}}})^r) \in \{0,1\}^n$.

(2) The voter gets the blinded encrypted vote v' by choosing randomly $0 \neq b \in \mathbb{Z}_q$ and calculating $v' = b(U + H_1(W))$. After that, v' is sent to each entity E_i in order to ask for an i-shadow-blind signature to each entity E_i, with $1 \leq i \leq n$.

(3) Each entity E_i computes $\sigma_i' = s_i v'$ and sends it back to the voter. Since $v' \in G_1$, $\sigma_i' \in G_1$ as well.

(4) The voter calculates the *i-shadow-signature* of each entity E_i by computing $\sigma_i = b^{-1}\sigma_i' = s_i(U + H_1(W))$. Since σ_i' is an element of G_1 so is σ_i.

(5) Considering a storage device, the vote (U, W) and the *i-shadow-signatures* are stored as $(U, W \| \sigma_1 \| \cdots \| \sigma_n)$, where σ_i is computed as in the previous step.

3.3. Counting Phase

(1) To rebuild and verify the signature of each vote, the independent Combining Entity proceeds as follows:

(a) It selects a subset $S \subseteq \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$ of t shadow-signatures, that is, $|S| = t$, and computes $\sigma = \sum_{i \in S} L_i \sigma_i$, where L_i denotes the Lagrange coefficient associated with the polynomial $g_i(x)$ given by $L_j = \prod_{j \in S, j \neq i}(-j/(i-j))$ ([17]). Observe that in particular $\sum_{i \in S} s_i L_i = s$.

(b) It verifies the signature by checking that $\hat{e}(\sigma, P_1) = \hat{e}(U + H_1(W), Q)$.

(2) To decrypt the votes, the procedure is as follows:

(a) Each entity E_i calculates its decryption share $\hat{e}(U, d_{\text{ID}_i})$ for every vote cast and sends to the Combining Entity, who selects a set $T \subseteq \{\hat{e}(U, d_{\text{ID}_1}), \hat{e}(U, d_{\text{ID}_2}), \ldots, \hat{e}(U, d_{\text{ID}_n})\}$ of t decryption shares and computes $g = \prod_{i \in T} \hat{e}(U, d_{\text{ID}_i})^{I_i}$, where I_i denotes the Lagrange coefficient associated with the polynomial $f_i(x)$ given by $I_j = \prod_{j \in S, j \neq i}(-j/(i-j))$ ([17]).

(3) Once g is determined, the vote is decrypted by computing $v = W \oplus H_2(g)$.

(4) The votes are counted and the tally is published. The voter can check if its vote was counted by comparing its receipt with the announced results.

4. Properties Analysis

4.1. Privacy. The proposed electronic voting protocol meets the privacy property by using a threshold encryption scheme and its respective signature version, which is probably secure under the Computational Bilinear Diffie-Hellman Problem. With this, only the Combining Entity, jointly with at least t entities, is the only one who is able to decrypt votes and verify signatures during the counting stage. The correctness is shown as follows from the signature verification in Section 3.3:

$$\begin{aligned}
\hat{e}(\sigma, P_1) &= \hat{e}\left(\sum_{i \in S} L_i \sigma_i, P_1\right) = \prod_{i \in S} \hat{e}(\sigma_i, P_1)^{L_i} \\
&= \prod_{i \in S} \hat{e}\left(b^{-1}\sigma_i', P_1\right)^{L_i} = \prod_{i \in S} \hat{e}\left(b^{-1} s_i v', P_1\right)^{L_i} \\
&= \prod_{s_i \in S} \hat{e}\left(s_i\left(U + H_1(W)\right), P_1\right)^{L_i} \\
&= \prod_{i \in S} \hat{e}\left(U + H_1(W), P_1\right)^{s_i L_i} \\
&= \hat{e}\left(U + H_1(W), \left(\sum_{i \in S} s_i L_i\right) P_1\right) \\
&= \hat{e}\left(U + H_1(W), sP_1\right) = \hat{e}\left(U + H_1(W), Q\right)
\end{aligned} \tag{1}$$

and from the decryption votes, also in Section 3.3:

$$\begin{aligned}
g &= \prod_{i \in T} \hat{e}\left(U, d_{\text{ID}_i}\right)^{I_i} = \hat{e}\left(rP_1, \left(\sum_{i \in T} a_{i0} I_i\right) P_{\text{pub}_{\text{ID}}}\right) \\
&= \hat{e}\left(rP_1, dP_{\text{pub}_{\text{ID}}}\right) = \hat{e}\left(P_1, P_{\text{pub}_{\text{ID}}}\right)^{rd} \\
&= \hat{e}\left(dP_1, P_{\text{pub}_{\text{ID}}}\right)^r = \hat{e}\left(P_{\text{pub}}, P_{\text{pub}_{\text{ID}}}\right)^r.
\end{aligned} \tag{2}$$

Then,

$$W \oplus H_2(g) = W \oplus H_2\left(\hat{e}\left(P_{\text{pub}}, P_{\text{pub}_{\text{ID}}}\right)^r\right) = v. \tag{3}$$

4.2. Robustness. The proposed electronic voting protocol meets robustness property by using bilinear properties in such way that each entity E_i has to prove, in a noninteractive way, the equality of two inverses of the isomorphism $f_{P_1} = \hat{e}(P_1, \cdot)$ induced by the bilinear map $\hat{e}$.

To do this, each entity E_i chooses a random $R \in G_1$ and computes $w_1 = \hat{e}(P_1, R) \in G_2$, $w_2 = \hat{e}(U, R) \in G_2$ and a hash e of the tuple $(\hat{e}(U, d_{\text{ID}_i}), \hat{e}(P_{\text{pub}}, P_{\text{pub}_{\text{ID}}}), w_1, w_2)$.

Then, entity E_i computes $V = R + e_{d\text{ID}_i} \in G_1$ and joins the tuple (w_1, w_2, e, V) to its share in order that other entities E_i can check that

$$\begin{aligned}
\hat{e}\left(P_1, V\right) &= \hat{e}\left(P_1, R\right) \hat{e}\left(P_{\text{pub}}, P_{\text{pub}_{\text{ID}}}\right)^e, \\
\hat{e}\left(U, V\right) &= \hat{e}\left(U, R\right) \hat{e}\left(U, d_{\text{ID}_i}\right)^e.
\end{aligned} \tag{4}$$

Both equalities hold as we can see as follows:

$$\begin{aligned}
\widehat{e}(P_1, V) &= \widehat{e}(P_1, R)\,\widehat{e}\left(P_{\text{pub}i}, P_{\text{pub}_{\text{ID}}}\right)^e,\\
\widehat{e}\left(P_1, R + ed_{\text{ID}_i}\right) &= \widehat{e}(P_1, R)\,\widehat{e}\left(P_{\text{pub}i}, P_{\text{pub}_{\text{ID}}}\right)^e,\\
\widehat{e}(P_1, R)\,\widehat{e}\left(P_1, d_{\text{ID}_i}\right)^e &= \widehat{e}(P_1, R)\,\widehat{e}\left(d_i P_1, d_{\text{ID}_i} d_i^{-1}\right)^e,\\
\widehat{e}(P_1, R)\,\widehat{e}\left(P_1, d_{\text{ID}_i}\right)^e &= \widehat{e}(P_1, R)\,\widehat{e}\left(P_1, d_{\text{ID}_i}\right)^e,\\
\widehat{e}(U, V) &= \widehat{e}(U, R)\,\widehat{e}\left(U, d_{\text{ID}_i}\right)^e,\\
\widehat{e}\left(U, R + ed_{\text{ID}_i}\right) &= \widehat{e}(U, R)\,\widehat{e}\left(U, d_{\text{ID}_i}\right)^e,\\
\widehat{e}(U, R)\,\widehat{e}\left(U, ed_{\text{ID}_i}\right)^e &= \widehat{e}(U, R)\,\widehat{e}\left(U, d_{\text{ID}_i}\right)^e,\\
\widehat{e}(U, R)\,\widehat{e}\left(U, d_{\text{ID}_i}\right)^e &= \widehat{e}(U, R)\,\widehat{e}\left(U, d_{\text{ID}_i}\right)^e.
\end{aligned} \tag{5}$$

4.3. Security Analysis. In the proposed protocol we assume that any attacker who wishes to break the privacy in the proposed electronic voting protocol is fully aware of the public key and any algorithms that may be used as part of the protocol. The information that is denied to the attacker is the private key for encryption during the voting phases.

The nature of the relation between the public and private keys means that it is possible for any asymmetric scheme to achieve a perfect notion of security. Public keys, by definition, must contain enough information to compute their associated private key. In such case it may be theoretically possible to recover the private key from the public key; it is not computationally feasible to do so. Considering that and that we cannot derive definite mathematical statements about the security of the protocol, we do prove that a reduction exists between the difficulty of breaking the protocol and the difficulty of solving a well-studied mathematical problem.

The reductionist approach is used to prove the security in our protocol relying on assumptions about the hardness of some mathematical problems. All of this is made in order to prove its security. We give some definitions as follows.

Definition 1. Given two groups G_1 and G_2 of the same prime order q, a bilinear map $\widehat{e} : G_1 \times G_1 \to G_2$, and a generator g of G_1, the Decisional Bilinear Diffie-Hellman Problem (DBDHP) in $(G_1, G_2, \widehat{e})$ is to decide whether $h = \widehat{e}(g, g)^{abc}$ given $(g, g^a, g^b, g^c) \in G_1^4$ and an element $h \in G_2$.

Definition 2. Given two groups G_1 and G_2 of the same prime order q, a bilinear map $\widehat{e} : G_1 \times G_1 \to G_2$, and a generator g of G_1, the Computational Bilinear Diffie-Hellman Problem (CBDHP) in $(G_1, G_2, \widehat{e})$ is to compute $h = \widehat{e}(g, g)^{abc}$ given $(g, g^a, g^b, g^c) \in G_1^4$.

In other words, security of proposed protocol is based on hardness assumptions for problems in groups equipped with a pairing. The advantage of solving such assumptions is given as follows.

Definition 3. The advantage of an algorithm A in solving the Bilinear Diffie-Hellman Problem (BDHP) in (G_1, G_2) is

$$\text{Adv}_A^{\text{BDHP}}(k) = \Pr\left[A\left(g, g^a, g^b, g^c\right) = \widehat{e}(g, g)^{abc}\right], \tag{6}$$

where $a, b, c \overset{\$}{\leftarrow} Z_q^*$ and we assume that parameters (G_1, G_2, e, q, g) as output by the algorithm *PairingGen* on input 1^k are given to A as additional inputs. We say that the BDHP is hard in (G_1, G_2) if no Polynomial Time Algorithm that solves the BDHP in (G_1, G_2) has a nonnegligible advantage, as a function of the security parameter k.

Definition 4. The advantage of an algorithm A in solving the Decisional Bilinear Diffie-Hellman Problem (DBDHP) in (G_1, G_2) is

$$\begin{aligned}
\text{Adv}_A^{\text{DBDHP}}(k) = \Big|&\Pr\left[A\left(g, g^a, g^b, g^c, \widehat{e}(g, g)^{abc}\right) = 1\right]\\
&- \Pr\left[A\left(g, g^a, g^b, g^c, Z\right) = 1\right]\Big|,
\end{aligned} \tag{7}$$

where $a, b, c \overset{\$}{\leftarrow} Z_q^*$ and $Z \overset{\$}{\leftarrow} G_2$. Moreover, we assume that parameters (G_1, G_2, e, q, g) as output by the algorithm *PairingGen* on input 1^k are given to A as additional inputs. We say that the DBDHP is hard in (G_1, G_2) if no Polynomial Time Algorithm that solves the DBDHP in (G_1, G_2) has a nonnegligible advantage, as a function of the security parameter k.

Definition 5. The advantage of an algorithm A in solving the Computational Bilinear Diffie-Hellman Problem (CBDHP) in (G_1, G_2) is

$$\begin{aligned}
&\text{Adv}_A^{\text{CBDHP}}(k)\\
&\quad = \left|\Pr\left[A\left(g, g^a, g^b, g^c, \widehat{e}(g, g)^{abc}\right) = h\right]\right|,
\end{aligned} \tag{8}$$

where $a, b, c \overset{\$}{\leftarrow} Z_q^*$ and $h = \widehat{e}(g, g)^{abc} \overset{\$}{\leftarrow} G_2$. Moreover, we assume that parameters (G_1, G_2, e, q, g) as output by the algorithm *PairingGen* on input 1^k are given to A as additional inputs. We say that the CBDHP is hard in (G_1, G_2) if no Polynomial Time Algorithm that solves the CBDHP in (G_1, G_2) has a nonnegligible advantage, as a function of the security parameter k.

Considering the aforementioned, to break our protocol from the privacy point of view, first of all, attacker must break the atomic primitives our cryptographic protocol is based on in addition to getting nonnegligible advantage in the above definitions.

5. Results

In order to get a comparison between the proposed protocol and related work, results are shown from two points of view; Table 1 shows the first one, which is viewed from the total number of PKI components that the proposed protocol

Table 1: Proposed protocol does not need any component of a PKI.

Protocol	PKI Component 1	PKI Component 2	Privacy and robustness based on
Cramer et al. [1]	0	0	DHP
Mu et al. [2]	$1 * V$	0	DHP
Ohkubo et al. [3]	$1 * v$	0	DHP
Baudron et al. [4]	$1 * L$	0	RC
Gallegos-García et al. [5]	0	1	CBDHP
Proposed protocol	0	0	CBDHP

Table 2: Cryptographic operations developed in the proposed protocol.

Op.	Ohkubo et al. [3]	Cramer et al. [1]	Baudron et al. [4]	Mu et al. [2]	Gallegos-García et al. [5]	Proposed protocol
$+$	3	1	$2(i-2)+2$	1	0	$(t-1)*2$
x	$16+(t-1)$	$12+(i-1)$	$L*10+8+2(t-1)$	6	0	$(t-1)*2$
x^y	13	$19+n+n*i$	$L(n!+13)+9+t$	11	0	0
x^{-1}	2	3	$L*2$	1	1	1
$P+Q$	NA	NA	NA	NA	0	0
nP	NA	NA	NA	NA	$2+1*v+3*i$	$2*((4*i)+(1*j)+(1*j*t-1)+(1*i*n))+(1*i)+(2*v)*(1*v*i)$
Hash	5	NA	NA	NA	$(1*i)+(3*v)$	$(1*i)+(3*v)$
$\hat{e}$	0	0	0	0	$(1*v)+(1*i)+2$	$(1*v)+(1*i)+2$

would use to develop a voting process. In such table PKI Component 1 and PKI Component 2 mean certification and trust authorities, respectively. Both of them are main components in a PKI. In that table it is possible to see that the number of components required increases depending on the number of voters participating in the voting protocol. Moreover, the proposed electronic voting protocol meets privacy and robustness based on Diffie-Hellman problems, which become as secure as [5] and more secure than [1–4], as [5] reports. In this sense CBDHP means Computational Bilinear Diffie-Hellman Problem.

On the other hand, the second point of view is from the computations needed to develop the proposed protocol, which depends on the number of cryptographic operations used in comparison with the proposed one. Operations considered are modular addition ($+$), modular multiplication ($*$), exponentiation (x^y), inversion (x^{-1}), point addition ($P+Q$), and scalar multiplication (nP). Moreover, v means voter and parameter n represents the total number of shareholders who participate during the voting process with $1 \leq i \leq n$ and t denotes the threshold that the voting protocol considers for counting stage. It is important to say that our protocol involves operations based on groups, finite fields, and field extensions, which are made by using polynomials to represent the field elements that bilinear maps use.

In Table 2 it is possible to see that even though the proposed protocol does not involve exponentiations and point additions, it does use several computations of bilinear maps, which involves additions and multiplications over a finite field and its extensions, a technique called tower fields. However, even though the proposed protocol has the highest computational cost, bilinear maps can be addressed by using cryptoaccelerators, which efficiently develops such kind of cryptographic operations. The inclusion of such processors is considered to be cheaper and preferred than the components of a Public Key Infrastructure.

6. Conclusion

Electronic voting protocols that include as main construction blocks blind signatures and homomorphic and secret sharing techniques have been developed in last years. In this paper we present a protocol that is based on blind signatures and secret sharing techniques, using blind signatures and encryption schemes as the main construction blocks. The main difference with protocols proposed to date is that its functionality is based on bilinear maps and secret sharing schemes, which are used jointly with their respective properties to meet expectations of privacy and robustness. Bilinear maps develop high cost operations which can be addressed by using cryptoaccelerators to efficiently develop this sort of operations. As a result, we eliminate the need of implementing a Public Key Infrastructure (PKI).

In addition the proposed protocol is based on the difficulty of solving the Computational Diffie-Hellman Problem (CDHP) and the Bilinear Diffie-Hellman Problem (BDHP); due to its construction it can be found on supersingular elliptic curves or hyperelliptic curves over finite fields; as a consequence no algorithm exists as yet capable of solving such problems in polynomial time.

According to what was mentioned above, it is easy to see that proposed protocol highlights the balance between security and efficiency. In other words, from the security point of view, the proposed protocol is based on the difficulty of solving the Computational Diffie-Hellman Problem (CDHP) and the Bilinear Diffie-Hellman Problem (BDHP). From the efficiency point of view, we eliminate the need of implementing the components of a Public Key Infrastructure (PKI) and leave as consideration the development of cryptographic operations by using cryptoaccelerators.

The protocol presented here could be used, for instance, in a voting system based on Direct Recording Electronic (DRE) systems, which provides authentication of the voter's identity based on official documents presented to the electoral authority.

Moreover, the voter's receipt could be used to meet requirements of verifiability and accuracy. Thus, in order to verify if the votes were recorded and counted, the receipt should appear on a bulletin board in which it is displayed together with the final tally. If any voter does not find his/her hash value on the bulletin board, he/she can register a complaint with election officials.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors thank the Instituto Politecnico Nacional and the Consejo Nacional de Ciencia y Tecnología. The research for this paper was financially supported by Project Grant no. SIP-2014-RE/123, CONACyT 216533.

References

[1] R. Cramer, R. Gennaro, and B. Schoenmakers, "A secure and optimally efficient multi-authority election scheme," in *Advances in Cryptology—EUROCRYPT '97*, vol. 1233 of *Lecture Notes in Computer Science*, pp. 361–376, Springer, Berlin, Germany, 1997.

[2] Y. Mu and V. Varadharajan, "Anonymuos secure e-voting over a network," in *Proceedings of the 14th Annual Computer Security Applications Conference*, pp. 293–299, IEEE Computer Society, 1998.

[3] M. Ohkubo, F. Miura, M. Abe, A. Fujioka, and T. Okamoto, "An improvement on a practical secret voting scheme," in *Information Security*, vol. 1729 of *Lecture Notes in Computer Science*, pp. 225–234, Springer, Berlin, Germany, 1999.

[4] O. Baudron, P. Fouque, D. Pointcheval, G. Poupard, and J. Stern, "Practical multi-candidate election system," in *Proceedings of the 20th Annual ACM Symposium on Principles of Distributed Computing (PODC '01)*, pp. 274–283, ACM, 2001.

[5] G. Gallegos-García, R. Gómez-Cárdenas, and G. I. Duchén-Sánchez, "Identity based threshold cryptography and blind signatures for electronic voting," *WSEAS Transactions on Computers*, vol. 9, no. 1, pp. 62–71, 2010.

[6] Internet Policy Institute, *Report of the National Workshop on Internet Voting: Issues and Research Agenda*, Internet Policy Institute, 2001.

[7] T. Kohno, A. Stubblefield, A. D. Rubin, and D. S. Wallach, "Analysis of an electronic voting system," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 27–40, IEEE, May 2004.

[8] D. A. Gritzalis, "Principles and requirements for a secure e-voting system," *Computers and Security*, vol. 21, no. 6, pp. 539–556, 2002.

[9] P. Barreto, H. Kim, and M. Scott, "Efficient algorithms for pairing-based cryptosystems," in *Advances in Cryptology—Crypto '02*, vol. 2442 of *Lecture Notes in Computer Science*, pp. 354–368, Springer, 2003.

[10] J. L. Beuchat, N. Brisebarre, J. Detrey, E. Okamoto, M. Shirase, and T. Takagi, "Algorithms and arithmetic operators for computing the ηT pairing in characteristic three," *IEEE Transactions on Computers*, vol. 57, no. 11, pp. 1454–1468, 2008.

[11] H. Cohen and F. Gerhard, *Handbook of Elliptic and Hyperelliptic Curve Cryptography*, Chapman & Hall/CRC, Taylor & Francis, 2006.

[12] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology—CRYPTO 2001: Proceedings of the 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19–23, 2001*, vol. 2139 of *Lecture Notes in Computer Science*, pp. 213–229, Springer, Berlin, Germany, 2001.

[13] B. Libert and J.-J. Quisquater, "Efficient revocation and threshold pairing based cryptosystems," in *Proceedings of the 22nd Annual ACM Symposium on Principles of Distributed Computing (PODC '03)*, pp. 163–171, July 2003.

[14] D. Liem, F. Zhang, and K. Kim, "A new threshold blind signature scheme from pairings," in *Proceedings of the Symposium on Cryptography and Information Security*, Hamamatsu, Japan, 2003.

[15] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[16] Y. Desmedt, "Threshold cryptosystems," in *Advances in Cryptology—AUSCRYPT '92*, J. Seberry and Y. Zheng, Eds., vol. 718 of *Lecture Notes in Computer Science*, pp. 1–14, Springer, Berlin, Germany, 1993.

[17] A. Boldyreva, "Efficient threshold signature, multisignature and blind signature schemes based on the Gap-Diffie-Hellman-Group signature scheme," in *Proceedings of the International Workshop on Public Key Cryptography (PKC '03)*, vol. 2139 of *Lecture Notes in Computer Science*, pp. 31–46, Springer, 2003.