Electronic Voting Algorithm and Its Algebraic Formation

M. Mesbahuddin Sarker[1] and Dr. Md. Sharif Uddin[2]

[1]Associate Professor,
Institute of Information Technology, Jahangirnagar University,
Savar, Dhaka Bangladesh
[2]Professor, Mathematics,
Dept. of Mathematics, Jahangirnagar University,
Savar, Dhaka, Bangladesh.

Abstract

Vote security, secrecy and confidence are necessary in any electoral process. Electronic voting is one of the most valuable exploratory areas for the pursuance of a secure e-Government transaction environment. This system is increasingly used in electoral processes ranging from specialized stand alone machines, up to complete paperless and remote voting system. Therefore this system has to follows a very complex set of security protocols. This paper analysis several algorithms to implement an electronic voting systems and discusses with a view to voter anonymity and proper protection from manipulations.

Key Words: Cryptography, Encryption-Decryption, Blind Signature, Zero Knowledge Proofs.

1. INTRODUCTION

Electronic voting promises the possibility of a convenient, efficient and secure facility for recording and tallying votes. It can be used for a variety of types of elections, from small committees or on-line communities through to full-scale national elections. The traditional process of election is quite tedious, time consuming, cumbersome because voter's come in person and vote at pre-assigned voting booth. But in era of networking and internet, we can overcome this problem by using the electronic voting system. This system is expected to make our modern social life more convenient, efficient, inexpensive and without disturbing the daily routine life. More potentially secure system could be implemented, based on formal protocols that specify the messages sent between the voters and administrators. Such protocols have been studied for several decades aiming to provide security properties, which go beyond those that can be achieved by paper-based voting systems. Generally, any e-Voting systems consist of six [Triinu 2007] main phases such as (figure 1), also known as Conventional Voting System [Kalaichelvi et. al. 2011]:

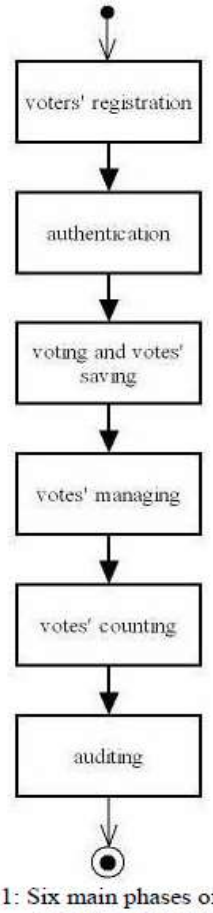

Fig.1: Six main phases of e-Voting systems

i. **The voters' registration** - is a phase to define voters for the e-Voting system and give them authentication data to log into the e-Voting system.
ii. **The authentication** - is a phase to verify that the voters have access rights and franchise.

iii. **The voting and vote's saving** - is a phase where eligible voters cast votes and e-Voting system saves the received votes from voters.
iv. **The votes' managing** - is a phase in which votes are managed, sorted and prepared for counting.
v. **The votes' counting** - is the phase to decrypt and count the votes and to output the final tally.
vi. **The auditing** - is a phase to check that eligible voters were capable to vote and their votes participate in the computation of final tally.

2. ALGEBRAIC FORMATION OF E-VOTING REQUIREMENTS

The minimum requirements that every outline of electronic voting should satisfy are the followings [Cabello et. al. 2007]:

i. **Anonymity:** It should be impossible to link the ballot with the voter which casts it. None of the three authorities that participate in the electoral process can determine the vote of a voter V_i. The authority of certification U_o knows the bit sequence C_i but it is impossible to determine v_j because he doesn't know B_i. The authority of authentication U_1 only knows the bit sequence B_i and consistently any forecast on the vote of V_i will not have a probability over 0.5. Finally, the authority of recollection U_2, knows P_i but he does not have any information about C_i since the only data he knows is the XOR sum

$$C_1 \oplus C_2 \oplus ... \oplus C_N$$

ii. **Completeness:** Only the eligible voters are allowed to vote. This property remains guaranteed since the authority of certification U_o takes charge of providing digital certificates to the registered voters and to make the blind signature of the different votes P_i.
iii. **Correctness:** Each voter should can to check that the own vote has been considered appropriately. Each voter V_i can verify that its vote has been considered since the bit sequence P_i is published by the U_2 authority. In addition it is possible to verify the final result of the recount since also sequence C is made public.
iv. **Uniqueness:** Each legal voter can vote only once. By the own construction of the algorithm, each one of the voters can cast only vote valid.

3. ELECTRONIC VOTING PROTOCOL

There are many e-Voting protocols have been done successfully. Among them are Cryptographic Voting Protocols [Karlof et.al. 2005], A Novel in E-Voting of Egypt [Abo-Rizka et.al. 2007] and A Simple Protocol for Yes-No Electronic Voting [Cabello et. al. 2007]. Though Chaum presented the first e-Voting scheme [Chaum 1981]. For the construction of votes we will use the bit operation XOR, there are four parts implied in this scheme:

i. **Voters**: $V_1, V_2 ....., V_N$. They are the main actors of any electoral process. Every voter must emit one and only one anonymous vote that is assessed correctly by the pertinent authorities.

ii. **Authority of certification**: U_o. It is a third trusted party whose mission is to provide digital certificates to the legitimate voters registered and to carry out the blind digital signatures of the votes.

iii. **Authority of authentication**: U_1. It is a third trusted party whose mission is to authenticate the registered voters and to provide them of the necessary tools to emit their vote in a proper way.

iv. **Authority of collection**: U_2. It is a third trusted party responsible for collecting votes, to verify its validity, to store them and finally to carry out the recount of them. It is, therefore, the only entity that has permission for the deciphered of the votes.

4. ALGEBRAIC FORMATION OF E-VOTING PROTOCOLS

i. The authority of certification U_0 emits a digital certificate to each one of the legal registered voters.

ii. Each voter V_i is identified by the authority of authentication U_1, which validates its digital certificate and sends a random sequence of bits $B_i \in F_2^N$ to the voter.

iii. Each voter V_i constructs his/her vote, $v_i \in F_2^N$, as follows:

- If V_i votes for option 1, than: $\mathrm{v_i} = \mathrm{B_i} \oplus (0,....,0, \overset{\text{i th bit}}{1}, 0, ..., 0)$.
- If V_i votes for option 2 than: $\mathrm{v_i} = \mathrm{B_i} \oplus (0,....,0, \overset{\text{i th bit}}{1}, 0, ..., 0)$.

iv. Each voter V_i randomly chooses a bit sequence $C_i \in F_2^N$ and computes: $P_i = v_i \oplus C_i$.

v. The authority U_0 makes the blind signature of P_i, P_i^*, and returns it to V_i, which obtains, when recovering it, $S(P_i)$.

vi. Each voter V_i sends to the authority U_0 the bit sequence $C_i \in F_2^N$.

vii. Each voter V_i sends to the authority U_2 his/her vote signed by U_0 : $S(P_i)$.

viii. The authority U_0 computes:

$$C = C_1 \oplus C_2 \oplus ... \oplus C_N \in F_2^N$$ and sends it to the authority U_2.

ix. The authority computes:

$$B = B_1 \oplus B_2 \oplus ... \oplus B_N \in F_2^N$$ and sends it to the authority U_2.

x. The authority U_2 verifies the validity of the different votes deciphering $S(P_1), ..., S(P_N)$, obtaining $P_1, ..., P_N$.

xi. The authority U_2 computes:

$$P = P_1 \oplus P_2 \oplus ... \oplus P_N$$

$$P \oplus C = v_1 \oplus v_2 \oplus ... \oplus v_N = v.$$

xii. The authority U_2 calculates the number of votes obtained by option 1 simply computing the Hamming distance of bit sequences *v* and *B*. That is:

Number of votes of option 1: $d_H(v, B)$,

Number of votes of option 2: $N - d_H(v, B)$.

xiii. Finally, the authority U_2 publishes the bit sequences $P_{1,\ldots,}P_N$ together with C

[Cabello et. al. 2007]

5. E-VOTING ALGORITHM

Several algorithms for electronic voting have been developed and we here describe the following algorithms such as Mixnet, Homomorphic and Blind signature. A new E-voting system using the way of cloud technology [Pankaj 2014] is highly secure *secured for identifying the voter b*ut this system is very expensive because the system is not established one server of E-voting system rather than this system is installing many servers of the E-voting system over various geographical locations of India.

5.1. Algorithm for Mixnets Based e-Voting

Chaum [Chaum 1981] introduces the concept of a mix-net that is built up from several linked servers called mixes. Each mix randomizes input messages and outputs the permutation of them, such that the input and output messages are not linkable to each other. This approach is very easy to understand because the sequence of cryptographic building blocks closely resembles how a classic paper based voting occurs too. Several steps of this approach are described below (figure 2):

i. The voter prepares the plaintext ballot and encrypts it so that only he himself is able to decrypt it. He also calculates so called zero-knowledge [Goldwasser et. al. and Jean-Jacques et.al. 1989] proofs to assure that the encrypted vote is in fact a valid vote.

ii. The voter then authenticates himself with the Voting Authority, who checks that the voter is eligible to vote.

iii. The voter receives the signed vote back and decrypts it. He now holds a plaintext ballot which is signed by the voting authority. This is what blind signature means: the voting authority is able to sign the plaintext contents of the vote, even if it is encrypted.

iv. The voter now encrypts the vote with the public key used for the elections. He then sends the vote through an anonymizing mix-net. This would be a network of independently operated computers, each of which will somehow shuffle the incoming votes and then send them in a different order to the next node in the mix-net. Each link in the mix-net could also include its own encryption-decryption, on top of the encryption the voter already applied to the plaintext vote.

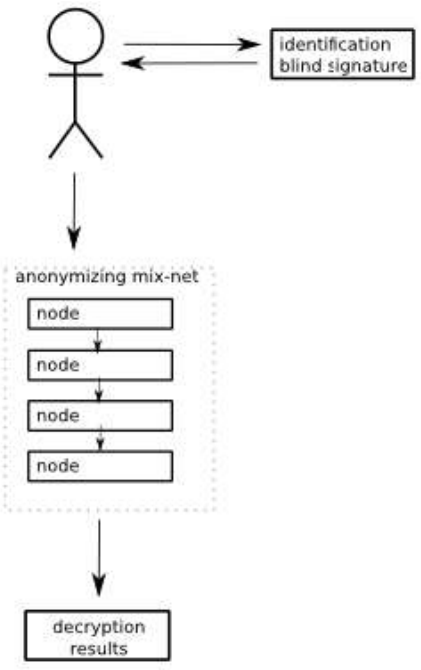

Fig.2: Mixnet based e-Voting algorithm

v. When voting has closed, all votes are decrypted. To ensure that nobody can decrypt any votes ahead of time.
vi. Plaintext votes are counted.

Fig. 3 presents a schematic diagram of a generic mixnet. At stage $i(Mix_i)$, a batch of inputs are received and transformed using either decryption or encryption, permuted and parallely transferred to stage $i+1$. Based on the cryptographic transformation used, the mixnet is called decryption mixnet or re-encryption mixnet [Krishna et. al. 2006]. We describe them below.

5.1.1 Decryption mixnet/ Re-encryption mixnet

Let K_i be the public key of the ith stage. Let the sender of the mixnet be a voter and the receiver be an authority, in a voting scheme. A sender V_j concatenates a message m_j with a random string r_j as $(m_j \| r_j)$, then encrypts as, $E_{K_1}(E_{K_2}(...(E_{K_l}(m_j \| r_j))...))$ and broadcasts it Mix_i with private key K_i^{-1}, receives inputs as, $E_{K_i}(E_{Ki+1}(...(E_{K_l}(m_j \| r_j))...))$ from Mix_{i-1}. Another decryption algorithm is developed by Swaminathan [Swaminathan et. al. 2012]

However, the mixnet algorithm can be described as follows [Krishna et. al. 2006](figure 3):

Input:

$E_{K_1}(E_{K_2}(...(E_{K_l}(m_j \| r_j))...))$; j = 1, …, n.

For i = 1, …, l.

For j = 1, …, n.

Step 1:

Decrypt as $D_{K_i} E_{K_i}(E_{Ki+1}(...(E_{K_l}(m_j \| r_j))...)) = E_{Ki+1}(...(E_{K_l}(m_j \| r_j))...)$.

Step 2:

Lexicographically order all decrypted quantities obtained in Step 1.

Output :

$\{m_j\}_R$, a batch of mixed messages that cannot be traced back to senders.

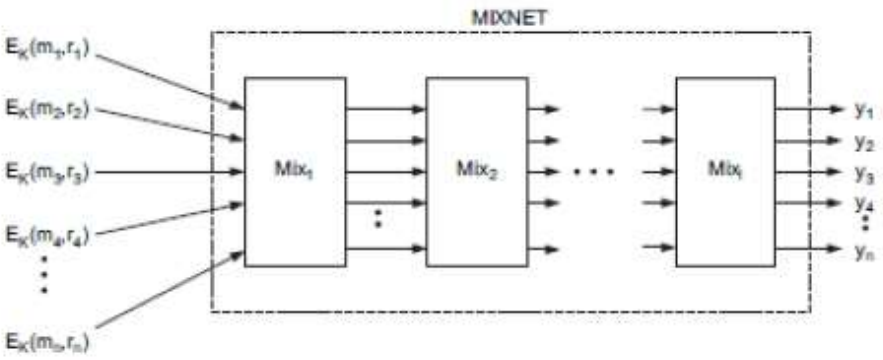

Fig.3: A schematic diagram of a generic mixnet with l mixes and n inputs

5.2. Algorithm for Homomorphic e-Voting

The homomorphic encryption was proposed by Cramer [Cramer et al. 1997] and it takes advantage of the characteristic properties of the homomorphic encryption to provide verifiability to the electronic vote schemes without contributing any information on the individual votes. Homomorphism is an algebraic property particularly useful in electronic voting schemes because it allows applying operations on sets of encrypted ballots without need of decrypting them [Laure et. al.]. In electronic voting schemes, this notion is used as follows: Let M is a plain-texts groups and C be a cipher-texts group. Then, we say that the encryption scheme is $(\otimes, \oplus)$-homomorphic if for any instance E of the encryption scheme, given $c_1 = E_{r1}(m_1)$ and $c_2 = E_{r2}(m_2)$, there exists an r such that: $c_1 \otimes c_2 = E_r(m_1 \oplus m_2)$. Such encryption schemes are often used in electronic voting protocols, for example, to compute the tally without decrypting each vote and therefore guarantee the privacy of voters. According to Hingo [Hingo 2013], algorithm based on homomorphic encryption would work as follows (figure 4):

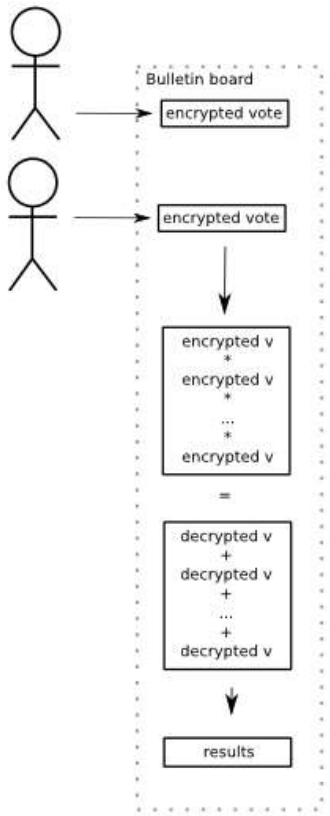

Fig.4: Homomorphic encryption based e-Voting algorithm

i. The voter prepares a plaintext ballot and encrypts it with a homomorphic encryption algorithm. He also provides zero-knowledge proofs that the contents of the encrypted ballot are a valid ballot. He also signs the ballot, identifying himself.
ii. The encrypted vote, the proofs and the signature are all posted on a public, non-erasable bulletin board. Therefore the verifiability of this approach seems to be well taken care of.
iii. After voting has closed, the voting authorities will multiply all votes with each other. Again this happens in public, and of course anyone could do the same multiplication.
iv. The voting authority then takes the result of the multiplication and decrypts that. Individual votes are never decrypted.
v. Anyone can see the result. Anyone can also verify that the result is the plaintext of the encrypted result.

5.3. Algorithm for Blind Signature e-Voting

The concept of blind signatures was first introduced by Chaum [Chaum 1982] to design the first e-case protocols. Later, they were used by Fujioka [Fujioka et. al. 1993]. A voting authority authenticates a message, usually an encrypted vote, without knowing the contents. Even if later the (un-blinded) signature is made public, it is impossible to connect the signature to the signing process, *i.e.* to the voter. Schemes based on blind signatures usually use anonymous channels in order to send the un-blinded signature and the encryption of the ballot to a voting authority, assuring the anonymity of the sender. Blind signature is often used to get a token from the authority [Fujioka Atsushi et.al. 1992]. The voter gets a signature

from the authority of his ballot and then he is able to cast his ballot. It is used to achieve eligibility.

5.4 Blind Signature

A blind signature allows somebody for instance an authority to sign an encrypted message without decrypting it. Once the message signed and resent to the sender, he has a signed version of his vote by the authority and a guarantee that his vote has not been seen [Andrea 2011]. The generic notations to describe blind signature scheme scheme [Zuzana 2002] with message space M is a 5-tuple $(\eta, \chi, \sigma, \delta, \Gamma)$, where

- η is a polynomial-time probabilistic algorithm, that constructs the signer's public key (*pk*) and its corresponding secret key (*sk*);
- is a polynomial-time blinding algorithm, that on input a message $m \in M$, a public key *pk* and a random string *r*, constructs a blind message m^1;
- σ is a polynomial-time signing algorithm, that on input a blind message m^1 and the secret key *sk* constructs a blind signature s^1 on m^1;
- δ is a polynomial-time retrieving algorithm, that on input a blind signature s^1 and the random string *r* extract a signature *s* on *m*;
- Γ is a polynomial-time signature-verifying algorithm that on input a message signature pair (*m*,*s*) and the public key *pk* outputs either yes or no.

6. CONCLUSION

There are some potential weaknesses in mixnet algorithm such as how do we know that the nodes in the mix-net don't cheat. They could for example drop some votes and don't forward them. Since we know the total amount of votes cast, the voting authority would see that some votes are missing. The corrupt nodes could then also duplicate the same amount of votes to make the total match. Still, what if there are votes missing? Who would we blame? Therefore, the nodes in the mix-net are required to publish some mathematical proofs that verify that they forwarded all messages correctly, without of course revealing how exactly they shuffled the messages. Such proofs make the process more verifiable. But still, some robustness issues remain in these schemes. On the other hand, the homomorphic algorithm approaches the universal verifiability and robustness aspects. There's no mixing and shuffling and hidden channels going on, rather everything happens in public, but without compromising the privacy of an individual voter. The main drawback in these schemes is that the algorithm will restrict the format of the plaintext ballot. Typically it can only be a number, or some kind of a bitmap.

7. REFERENCE

1. Abo-Rizka M., and Ghounaim H. (2007). A Novel in E-voting in Egypt. *IJCSNS International Journal of Computer Science and Network Security*, VOL.7, No.11. International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.4, July 2011.
2. Andrea Huszti (2011). A homomorphic encryption-based secure electronic voting scheme. Faculty of Informatics. University of Debrecen. Hungary.

3. Cabello A.B. Pardos, Hernández A. Encinas , Hoya S. White, Martín A. del Rey and Rodríguez G. Sánchez (2007). A Simple Protocol for Yes-No Electronic Voting. IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.7, July 2007.
4. Chaum D. (1982). Blind Signatures for Untraceable Payments, CRYPTO '82, *Plenum Press*, 1982,199-203.
5. Chaum D. (1981). Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms, Communications of the ACM 24(2), 1981, 84-88.
6. Cramer R., Gennaro R., Schoenmakers B. (1997). A secure and optimally efficient multi-authority election schem." Advances in Cryptology-Eurocrypt 97, LNCS vol. 1233, 103-118, 1997.
7. Fujioka Atsushi, Tatsuaki Okamoto, and Kazuo Ohta (1992). A practical secret voting scheme for large scale elections. In ASIACRYPT '92: Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques, pages 244–251, London, UK, 1992. Springer-Verlag.
8. Fujioka A, T. Okamoto, K. Ohta (1993). "A practial secret voting scheme for large scale elections" Advances in Cryptology-Asiacrypt 92, LNCS vol. 718, pp. 248-259, 1993.
9. Goldwasser S., Micali S., and Rackoff C. (1989). The knowledge complexity of interactive proof systems. SIAM J. Comput., 18(1):186–208, 1989.
10. Hingo (2013). On cryptographic e-voting algorithms in general. OpenLife.cc (Open Life: The Philosophy of Open Source).
11. Jean-Jacques Quisquater, Louis Guillou, Marie Annick, and Tom Berson (1989). How to explain zero-knowledge protocols to your children. In CRYPTO '89: Proceedings on Advances in cryptology, pages 628–631, New York, NY, USA, 1989. Springer-Verlag New York, Inc.
12. Kalaichelvi And Dr. RM. Chandrasekaran (2011). Design And Analysis of Secured Electronic Voting Protocol. Journal of Theoretical and Applied Information Technology, Vol. 34 No.2.
13. Karlof C., Sastry N., and Wagner D. (2005). Cryptographic voting protocols: A Systems perspective, *14th USENIX Security Symposium*, pp. 33-49.
14. Krishna Sampigethaya, Radha Poovendran (2006). A framework and taxonomy for comparison of electronic voting schemes. Computers & Security 25(2006)137–153. available at www.sciencedirect.com
15. Laure Fouard, Mathilde Duclos, and Pascal Lafourcade. Survey on Electronic Voting Schemes. VERIMAG, 2 avenue de Vignate, 38610 Grières, France.
16. Pankaj Kumar Malviya (2014) E-Voting System Using Cloud in Indian Scenario. International Journal of Engineering Science & Advanced Technology] ISSN: 2250-3676, Volume-3, Issue-3, 171-175.
17. Swaminathan B. and Cross Datson Dinesh J. (2012). Highly Secure Online Voting System with Multi Security using Biometric and Steganography. International Journal of Advanced Scientific Research and Technology Issue 2, Volume 2 ISSN: 2249-9954.
18. Triinu Mägi (2007). Practical Security Analysis of E-voting Systems. Master Thesis. Faculty of Information Technology. Department of Informatics. Tallinn University of Technology. Tallinn.
19. Zuzana Rjaskova (2002). Electronic Voting Schemes. PhD thesis, Comenius University, Bratislava, 2002.