A Study on the Ubiquitous e-Voting System for the Implementation of e-Government

Choong Sik Kim[1], Chang Duk Jung[2], Seo Yeong Ha[3] and Chan Hyuk Park[4]

[1]*Director of Korea Agency for Culture and Promotion in Seoul, Seoul, South Korea*
[2]*Dept. of computer and information Korea University, Seoul, South Korea*
[3]*Director of Smart Broadcasting System, Seoul, South Korea*
[4]*Dept. of software Korea University, Seoul, South Korea*

ljw0525@hanmail.net, jcd12345@daum.net, mellinda@hanmail.net, churchoffire@daum.net

Abstract

The Global IT revolution is growing rapidly. Government and business have to be ready to meet the increased demand for effective and secure online services. With the E-Government practicing, day-by-day the public demand is also increasing simultaneously. Now this present moment, one of important research part is secure E-Voting for E-Government service, but for this important factor or Government Issue, it needs information privacy for secure information transaction of citizen's opinions and secure authentication. This paper has analyzed several approaches E-voting protocols, those are implemented with many digital signature mechanisms and maintained many types of cryptographic rules, which are main factor for information privacy. In this paper we discussed them with a view to voter anonymity and protection from manipulations. The paper then developed an algorithm designed to guarantee anonymity of the voter and to avoid the risk of manipulation of votes. In this paper the proposed algorithm is based upon the strict separation of voter's registration and submission of votes, which means that certain information has to be stored on a secure storage media. This paper discusses the secure security criteria and possible implementation options for such secure storage.

Keywords*: E-Government, E-Voting, ubiquitous, security, cryptographic, authentication*

1. Introduction

Each legal citizen has the right and duty to elect senators by the election for their country. And a voter must cast a vote by himself. Paper-form elections have been traditionally used. People normally go to certain places, show their valid identification cards to obtain ballot papers, then write down their opinions and finally cast their ballot papers into ballot boxes. In this process, election officers have to know the real identification of each voter who participates in the vote. To prevent a voter from voting multiple times, the officers have to check whether they have already voted. This process is not only very inefficient but it also violates the voter's privacy. By rapid developing of E-Government issues, E-Voting has been developed, where computers are used in the election process enabling greater effectiveness. Especially, on-line voting is trend in voting system development. Voters do not need to go to a central place to cast their ballots; instead they can stay at home or at any other place with Internet connection to cast their ballot papers. Obviously, electronic voting or on-line voting [1] is desirable since it is convenient for voters and speeds up the whole election process. But the main issue is security or information privacy for citizen's democratic rights and

anonymity of voter's. Many topics are involved in order to guarantee the desired level of security. [2] Lists four essential core properties those are desirable in almost any election system.

Our main focus is to guarantee anonymity of the voter and to avoid the risk of manipulation of votes for voter's privacy in network-based election system, called E-Voting. In this paper, we have tried to focus on information privacy with Secure E-Voting system, because E-Voting is one of the main issues where information transaction and privacy is needed with the help of crypto algorithm and blind signature mechanism. In this paper, in **Section 2** we have discussed the several existing E-Voting protocols in different approaches in the related work section. In **Section 3,** we described our proposed secure algorithm for secure E-Voting which is offering voter's anonymity, risk of manipulation and storage idea for storing tokens which will keep the secure authentication for voter and, in **Section 4**, we discussed our proposed method's performances.

2. Related Works

Chaum proposed an Anonymous Channel (MIX NET) [7] protocol for E-Voting where the original message is encrypted with the public keys from several servers and then passed from one server to the next server, each decrypting with its private key and passing the message on the next server in large batches with different order. In Anonymous Channel (MIX NET) E-Voting protocol, there are several MIX phases like $\boldsymbol{mix_1 \ldots\ldots mix_n}$. The reason of having several of such phases is to achieve robustness, otherwise would not be guaranteed because decryption mix-net does not have a final decryption phase. Rather, the initial encryption phase **E** encrypts its inputs by applying a concatenation of **k** encryption operations to each input; each mix peels off one of these encryptions by applying a corresponding decryption algorithm; it then mixes all its decrypted inputs by applying a secret random permutation to them. Thus, this scheme has the structure of an onion; **E** builds the onion, and each *mix* peels off one layer of the onion. Anonymous Channel also gives a solution to the problem of anonymous users but in this case at least one server has to be honest. On the other hand, when the number of server is large the protocol can become slow and also have the possibility to fake the vote.

Extensions of the original scheme can be found in [8] and in [6]. Later approaches in [9] and [10], apart from algorithmic improvements, add much to the stability and performance of the protocol and the computational effort in the client is reduced considerably (one collective key instead of several consecutive keys); however, it still has to be analyzed and tested in prototype implementations, whether the basic difficulties in MIX nets have been completely addressed.

The Homomorphism protocol, as Homomorphic based protocols in generally has limited scalability, as they tend to limit the poll to several options, which the voter has two options to choose from {1, -1}. The vote is cast as a binary YES/NO vote, encoded following a homomorphism scheme, and submitted to a number of ballot box servers. Due to the homomorphism, the summary count of YES/NO votes is possible without having to know the individual votes [11]. This advantageous property is also the main problem of the approach: only binary votes can be cast. Although its privacy is good and follows homomorphic encrypt algorithm but it is not suitable for real environment for E-Voting because of its two options YES/NO.

Later approaches in [12], it is based on homomorphic threshold encryption; the voter encrypts his vote and sends it in to the authorities that tally the votes. If voters can send in arbitrary plaintexts then they can cheat. It is therefore important that they attach an argument of knowledge of the plaintext being a correctly formed vote. Typically, those arguments are

honest verifier zero-knowledge arguments that are made non-interactive using the Fiat-Shamir heuristc. Security is argued in the random oracle model. All the currently existing homomorphic E-Voting schemes are based on additive homomorphism but presently it has been proposed on multiplicative homomorphism [13]. This news E-Voting scheme is more efficient.

ANDOS protocols provide a sender-anonymous channel. They emulate the anonymous purchase of a bit string [14]. In the *Two Agency Protocol* developed by Nurmi, Salomaa, and Santean [15], the responsibilities of validating registered voters and computing and publishing the results of the election are divided between two agencies, as in the simplistic scheme. In the first phase of this protocol, the validator sends a large prime identification tag to each of *n* voters who have previously reported an intention to vote. The validator then sends the tallier a list of all *n* identification tags (with no record of the corresponding voters). In the second phase, each voter *B* sends the tallier the pair $(t_B, h_B(t_B, v_B))$ [where t_B is the voter's tag, h_B is a cryptographic hash function of two variables, and v_B is the vote]. The tallier then publishes $h_B(t_B, v_B)$. *B* responds by sending the tallier the pair (t_B, h_B^{-1}), allowing the tallier to determine v_B. When the voting period is over, the tallier publishes a list of each v_B and it's corresponding $h_B(t_B, v_B)$. At this point each voter can confirm that his or her vote was counted properly. In phase 3, any voter who discovers that his or her vote was lost or not counted properly can protest by submitting the triple. Because $h_B(t_B, v_B)$ was published in phase 2, the tallier cannot deny the error. In phase 4 (optional), voters can change their votes by repeating the procedures in phases 2 and 3 with a different hash function. One of the biggest problems with this protocol is that if the validator and tallier collude they can determine the mapping between *B* and v_B

The *One Agency Protocol* is identical to the Two Agency Protocol, except for the tag distribution procedure in phase 1. When voters is not satisfied or cannot see their vote then they could be challenge their vote to tallier and tallier then distributes the secret tags to justifying the votes. This could be solves the collusion problem, but not the vote-buying[1] problem.

Fujioka, Okamoto and Ohta proposed in 1993 [16], blind signature based E-Voting protocol called FOO's protocol. Previously, Chaum invented the blind signature method in 1982. Considering the RSA based blind signature, the registrar (the authority that does the signing) has the set *(N,d,e)* based on RSA [where p and q prime numbers and N=(p.q); e and d are the public encryption key and decryption key]. The voter wishes to blind the ballot x by calculating *f(x)*, where $f(x) = x*R^e$ [where *R* is a randomly conjured number]. Let the signing function be $S(w) = (w^d) mod\ N$. The blinded ballot $x*R^e$ is sent to the registrar who cannot determine the true value of x and performs $S(f(x)) = (x*R^e)^d = x^d*R$. Then, the unblinding function $g(Y) = Y/R$. So once x^d*R is returned to the voter, the voter may unblind it by performing $g(S(f(x))) = x^d*R/R = x^d = S(x)$, thus the voter now has a signed ballot.

The registrar has no way of tracing the ballot even if the tallier publishes it and back to the voter. For every blinded ballot $y_1 = x_1*R_1^e$ and $y_2 = x_2*R_2^e$ and corresponding votes x_1, x_2 there is a random value $R^{\wedge}$ such that $y_2 = x_1(R^{\wedge})^e$ (specifically - $R^{\wedge} = (x_2/x_1)^d*R_2$. Since R_1, R_2 are random and secret the registrar cannot reasonably determine the matching any better than

[1] The last issue was addressed by Niemi and Renvall in a later paper, but the algorithm pre-supposes the use of a secure voting booth and involves high computational efforts [17]

guessing it. So, this blind signature method is so secured and can give guarantee of voter's anonymity.

Blind signature protocol is famous and we also followed this protocol. Nevertheless, although FOO's one-stage smart card based E-voting system is well formatted and well secured in application side, a problem arises when the administration of the registration and the ballot box servers collude. In this case, it is possible to break the anonymity as well as to vote for voters that are entitled to vote but do not do so. Another problem is if the browser based application (*e.g.*, a java applet provided by the registration to perform the registration step) fraudulently stores the IP address for each blindly signed ballot paper and passes this information to the ballot box and, thus, vote can be forged.

3. Proposed Algorithm for Secure Two-Doors E-Voting

Voters established secret and authentic communication channels with the Trust Center, through an authenticated key exchange [18] using the certification facilities provided by an independent cryptography infrastructure. One is for secure registration and another for secure voting. The use of secret sharing scheme removes the possibility of dishonest members affecting the voting.

For Voter's privacy and anonymity our proposed Secure Two-Doors E-Voting system has divided into two phases.

Registration Phase (First Door). In this phase, every voter has to be provided with a smart card holding his/her private key. The correspondent public key must be properly certified and the certificate published in a public directory. Before the Election Day, voters should have to complete the registration process. Voters can register even at a time when the list of candidates is not complete yet. During registration period voter will get authenticate tokens from Registration Server and Trust Center which have to need to cast the vote during Election Day on Voting phase. So voter's can store those tokens into the Smart Card or devices.

Voting Phase (Second Door). After finish the days of registration phase, the Election Day will come and that time Voting Phase will be start. In our E-Voting System we have divided the day of two phases only for voter's privacy and anonymity and avoid the risk of manipulations. Apart from the fact that anonymity can be guaranteed to the voter, if he uses different terminals (IP addresses) for registration and submission of the vote, the server administration of the registration and the ballot box collude, votes cannot be forged, as a valid vote also has to be authenticated by a trust center. On Election day voter's need to use the authenticate tokens those have been stored into secure storage media during registration period.

Before going to next section let us introduce with some notations:

Pin : The secret Personal Identification Number required to activate the Voter's Smart card.

SmtC: Smart Card.

STS : Mutual authentication and authenticated Key exchange protocol.

BP : Ballot Paper

B : Ballot box Server

R : Registration Server

V : voter

m , '*m*: Symmetric crypto key

$S^{\{V,R,B,T\}}_{\{priv,pub\}}$: The voter's, the registration's, the ballot box server's and the Trust Center's signature key pair.
$K^{\{V,R,B,T\}}_{\{priv,pub\}}$:The voter's, the registration's, the ballot box server's and the Trust Center's key pair for encryption.

3.1 Registration Phase (First Door)

- **Step 1.** Activate the Smart Card by proving the correct *Pin*.
- **Step 2.** Voter generates a random token *t.*
- **Step 3.** Prepare the *t* for blind signature and add a text for applying the E-Vote.
- **Step 4.** Sign it with voters private signature key: S^{V}_{priv}*(blinded(t), "I want to give E-Vote")*.
- **Step 5.** Then the message is encrypted with *R'*s (Registration server) public key and sent it to registration server: $K^{R}_{pub}[S^{V}_{priv}$*(blinded(t), "I want to give E-Vote")]*.
- **Step 6.** Registration server verifies the voter by resolving the public signature key of the voter's. If verification is successful then the registration server signs the blinded *t* and makes σ_R*(blinded(t))*.
- **Step 7.** Then the registration server signs σ_R*(blinded(t))* with *R*'s private signature key and again encrypt the message with voter's public encryption key and send it to the voter: $KV_{pub}[SR_{priv}\ \sigma_R$ *(blinded(t))]]*.
- **Step 8.** Voter gets the token *t,*σ_R*(t).*
- **Step 9.** Voter issued a second token **τ,** blind it and obtains the blindly signed $\sigma_T(\tau)$ from the trust center followed by same mechanism [**Step 1** to **Step 7**]**.**

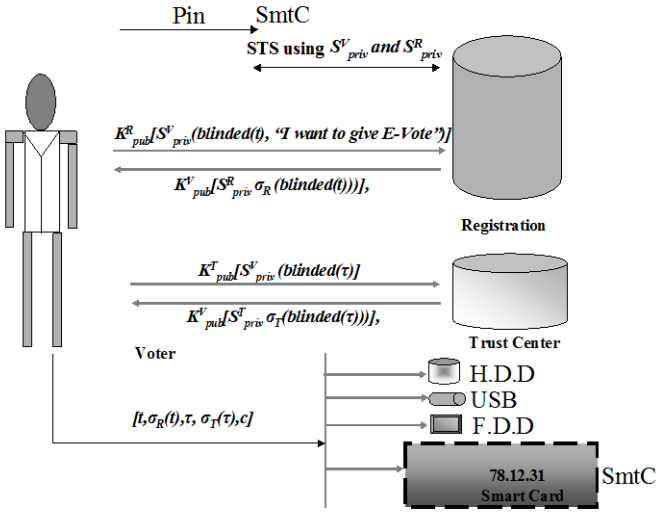

Figure 1. Registration Process (First Door). Voter is doing registration by generating random token number with blind signature mechanism and getting temporary tokens at last storing them to storage.

- **Step 10.** At the end of registration process voter obtains the temporary tokens and constituency information $[t,\sigma_R(t),\tau,\ \sigma_T(\tau),c]$ both tokens are needed to cast a vote on election day.

In most elections, voters will be organized in constituencies *c*, this information is also sent back to the voter and has to be submitted on election day to indicate in which constituency the vote is to be counted. To avoid possible manipulation of *c* the blind signature keys used for $\sigma_R(t)$can be made specific to the constituency. Hence the clear-text *c* submitted on Election Day and the authentication token issued by the registration have to point to the same *c*.

3.2 Voting Phase (Second Door)

On the Election Day the voter sends the tokens to the ballot box server to obtain a ballot paper.

- **Step 1.** Voter generates an asymmetric key *m* , '*m* for secure communication**.**
- **Step 2.** Voter adds *T*, the identification of the trust center or CA, for resolving the blind signature.
- **Step 3.** Voter encrypts the temporary token with ballot server's public encryption key and send it to ballot server: $K^B_{pub}\ [c,T,m,t,\sigma_R(t),\tau,\ \sigma_T(\tau)]$.
- **Step 4.** Ballot box decrypts the message to resolve the signatures $\sigma_R(t)$and $\sigma_T(\tau)$.
- **Step 5.** If the token is OK to authenticate the voter then the ballot server issues an empty ballot sheet and encodes it with the symmetric key *(m(BP))* and sends it to the voter: $m(S^B_{priv}(BP))$.
- **Step 6.** Voter decrypts it with '*m* and fills out the ballot paper.
- **Step 7.** Voter again send the ballot sheet with tokens by encrypting with the Ballot servers public key: $K^B_{pub}\ [c,T,m,t,\sigma R(t),\tau,\ \sigma T(\tau),BP]$.

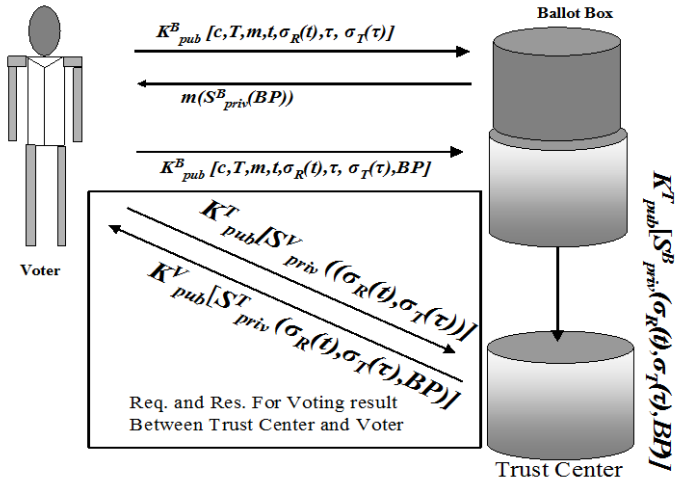

Figure 2. Voting Process (Second Door). Voter is requesting for ballot paper to the ballot box server with temporary tokens, which are supplied from registration process. After authentication the right voter then ballot box is responding to the voter with ballot paper and then again voter is filling the ballot paper and sending to the ballot box server with previous tokens.

4. Performances and Evaluation

Here we have depicted performances and evaluations following to the voting principles.

4.1 Computation Analysis

We have done an experiment by using PC (1.5 GHz Pentium 4 CPU with 256 MB RAM) to test the software performance. The simulation is performed with ASP script, visual Studio-6(C++) platform and with SSL v3 secure channel. The simulator divided into Registration process, Voting process and Trust center process.

The scheme is implemented and got the average time by using 1024 bit key length. When the 1024 bits key length is used, the voter needs $30008x10^{-3}$seconds for registration process and $30056x10^{-3}$ for voting process. For Trust Center process, its processing time is same as registration process.

Table 1. Computation time of our proposed Two-Doors E-Voting scheme

Key length *(bit)*	Registration process	Trust Center Process	Voting Process	Total
1024	$30008x10^{-3}$sec	$30008x10^{-3}$ sec	$30056x10^{-3}$ sec	$90072x10^{-3}$ sec

Performance of the scheme has shown in Table 1. Every voter needs $90072x10^{-3}$ seconds during computation time.

4.2 Performance Analysis

In this section, we have analysed our E-Voting Scheme with some analysed properties. The following properties are included in Table 2: Anonymity, Avoid manipulation of vote, Variability, Real Environment and Storage.

Table 2. Performance analysis of our Proposed Two Doors Secure E-Voting System

	Privacy & anonymity	Avoid Manipulation	Verifiability	Real Life environment	storage
Our Scheme	√	√	√	√	√

On the above Table 2, we have analyzed our scheme with following properties and (√) tic mark is indicating that our scheme is satisfied properly with following properties. According to the (Section 4.1) discussions we can measure (Table 2) analysis. If any users uses different terminals (IP address) for registration and submission of vote and the server administration of the registration and the ballot box collude but votes cannot be forged. Because as a valid vote also has to be authenticated by the Trust Center. By this way our Two-Doors E-Voting System could be guaranteed the anonymity of the voters. Although our method and also others existing methods (See Section 2 on related works) can give security and maintain anonymity of voters but we cannot give proper surety, because everything depends on Trust Center. If trust center collude then privacy of voter will not satisfy with any E-Voting

schemes. It can be solved if we choice a right and reliable trust center or CA. In our method, there has another advantages is "Storage". Before voting day a voter can registered and in voting day (another day) he/she can use his/her identity tokens, which were found from registration process and stored them to storage. This storage facility also is helpful for maintaining the privacy of voter. In verifiability property case, Trust Center strongly can satisfy it. By getting tokens with blind signature mechanism from registration process the besides privacy and anonymity our scheme also avoid the voting manipulation.

5. Conclusions

The main key-points of this paper were on privacy and anonymity of voters. In particular we are interested in E-Voting. In this paper voter's anonymity, privacy and avoiding the risk of manipulation are very important concerns for building a secure E-Voting system. In a nation all citizens have their rights and voting is one of them, to focus their democracy. Government has the duty to ensure democracy for citizens. In these times, E-Government is a new research area and secure E-Voting systems can play an important role to ensure citizens democracy rights in E-Government.

In this paper we proposed an algorithm for secure E-Voting (the two Two-Doors E-Voting system) and discussed about possible secure storage media. One of the main concerns in E-Voting is the possibility of fraudulent manipulations of the voter's PC or voting terminal. Our proposed algorithm mainly based on blind signature's mechanism can protect and give guarantee of voters' anonymity and has the capability to avoid the risk of manipulation of votes. This mechanism can efficiently satisfy the need of keeping privacy of the citizen's information. The Government can make National ID cards with Pin protected digital signature, which is affiliated by Trust Center or CA (Certificate Authority). This link is implemented by the combining the public key of digital certificate and the registry number, where the Government digitally signs the combination. Then, there is no need for voters to specific register for election and it could be more secured.

References

[1] A. D. Rubin, "Security Considerations for Remote Electronic Voting over the Internet", **(2001)** June 22, http://avirubin.com/e-voting.security.pdf.
[2] L. F. Cranor and R. K. Cytron, "Design and Implementation of Practical Security-Conscious Electronic Polling System", Washington University Report WUCS-96-02, **(1996)**.
[3] D. Chaum, "Secret-Ballot Receipts: True Voter-Verifiable Elections", IEEE Security and Privacy, **(2004)** January/February.
[4] J. Benaloh and D. Tuinstra, "Receipt-Free Secret –Ballot Elections", in STOC, **(1994)**, pp. 544-338.
[5] V. Niemi and A. Renvall, "How to Prevent Buying of Votes in Computer Elections", ASIACRYPT'94, LNCS 917 (Springer-Verlag), **(1994)**, pp. 125-132.
[6] K. Sako and J. Kilian, "Receipt-Free Mix-Type Voting Scheme", EUROCRYPT'95, LNCS 921 (Springer-Verlag), **(1995)**, pp. 393-403.
[7] D. Chaum, "Untraceable electronic mail return addresses and digital pseudonyms", Communications of the ACM, vol. 24, no. 2, **(1981)**, pp. 84-88.
[8] C. Park, K. Itoh and K. Kurosawa, "All/Nothing Election Scheme and Anonymous Channel", Lecture Notes in Computer Science, vol. 765, Advances in Cryptography Euro crypt 93, Berlin, Springer Verlag, **(1994)**, pp. 248-259.
[9] M. Abe, "Universally Verifiable Mix-Net with Verification Work Independent of the Number of Mix-Centers", Advances in Cryptology - EUROCRYPT '98, Springer-Verlag, Berlin, **(1998)**, pp. 437-447.
[10] M. Jakobsson, A. Juels and R. L. Rivest, "Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking", Proc. Usenix Security, Usenix Assoc., **(2002)**, pp. 339-353.
[11] R. Cramer, R. Gennaro and B. Schoenmakers, "A Secure and optimally Efficient Multi-Authority Election Scheme", Advances in Cryptology-EUROCRYPT'97, Lecture Note in Computer Science, vol. 1233, Springer-Verlag, Berlin, **(1997)**, pp. 103-118.

[12] J. Groth, "Non-interactive Zero-Knowledge Arguments for Voting", ACNS 2005, **(2005)**.
[13] K. Peng, R. Aditya, C. Boyd, E. Dawson and B. Lee, "Multiplicative Homomorphic E-Voting", INDOCRYPT'04, LNCS (Springer-Verlag), Chennai, India, **(2004)**, pp. 61.
[14] G. Brassard, C. Crepeau and J. -M. Robert, "All-or- Nothing Disclosure of Secrets", Lecture Notes in Computer Science, vol. 263, Advances in Cryptology; Crypto 86, Berlin, Springer Verlag, **(1987)**, pp. 234-238.
[15] A. Salomaa, "Verifying and Recasting Secret Ballots in Computer Networks", Maurer, H.A. (ed.): New Results and New Trends in Computer Science, Springer-Verlag, Berlin, **(1991)**, pp. 283-289.
[16] A. Fujioka, T. Okamoto and K. Ohta, "A Practical Secret Voting Scheme for Large Scale Elections", Advances in Cryptology – AUSCRYPT92, Springer-Verlag, Berlin, **(1993)**, pp. 244-251.

Authors

Kim Choong Sik

Social development graduate school of art & culture at Choong-Ang University. Complete the CEO course of business administration at Yonsei University. Director of Korea Liberal arts Center Director of Korea Agency for Culture and Promotion in Seoul.