Experimental Design of Worldwide Internet Voting System using PKI

Kwangjo Kim, Jinho Kim, Byoungcheon Lee, and Gookwhan Ahn

Abstract— **We have designed an Internet voting system applicable for worldwide voting which is based on Ohkubo *et. al.*'s scheme [23] combined with Public Key Infrastructure (PKI). To the best of our knowledge, this is the first trial to serve secure Internet voting system to the world. In our system, voter's privacy is guaranteed by using blind signature and mix-net, and robustness is provided through the threshold encryption scheme. By employing Java technology, we propose a way of typical implementation for internet voting system. Furthermore, PKI permits worldwide key distribution and achieve "one certificate/one vote" policy. Therefore, anyone can participate in the voting if he gets a certificate from Certificate Authority (CA). By the joint work between Korean and Japanese teams, the implementation aims to select MVPs in 2002 FIFA World Cup Korea-Japan^{TM} in easy and friendly manner for any Internet user to participate and enjoy Internet voting.**

Keywords— **Internet voting, PKI, security, cryptography, blind signature, CA.**

I. Introduction

VOTING is one of efficient methods for decision making in any society. The research on electronic voting through Internet will play a very important role for the progress of democracy. If a secure and convenient electronic voting system is provided, it will be used more frequently to collect the opinion of eligible voters for many political and social decisions through cyberspace.

As new services like e-commerce, e-cash, and e-government using cryptographic primitives become popular over the Internet, the possibility of electronic voting over the Internet also attracts great interest. If the Internet voting is sufficiently easy and comfortable, many people can easily participate in voting over the Internet. It is considered as a good solution for the recent problem of decreasing the participation rate in voting with which all nations over the world are confronted.

The state of California [3] has introduced a shadow election test of Internet voting system for the public election in Conta Costa County. Recently, Caltech-MIT joint project [30] has started in 2000 to develop reliable and uniform US voting machine due to the problems that threatened the 2000 American presidential election in Florida [29].

Internet voting systems must meet security requirements such as anonymity, privacy, completeness, fairness, verifiability, and receipt-freeness. These requirements make Internet voting much more challenging than other electronic commerce or electronic government applications. Although there has been strong interest for Internet voting as a good solution to make voting more accessible and convenient, it is widely believed that there is no practical Internet voting system satisfying all these requirements of electronic voting together with good efficiency.

In this research, we design Internet voting system using public key infrastructure (PKI). Our proposed Internet voting system satisfies most of important security requirements and has efficiency and flexibility. Although this system is not quite the first trial, we believe it is the first user-friendly, secure Internet voting system using PKI which is open to general people over the Internet. In our system, voter's privacy is guaranteed by using blind signature and mix-net, and robustness is provided through the threshold encryption scheme. By employing Java language suitable for the Internet, we can implement a user-friendly web interface for the voting system and a downloadable voting applet. Furthermore, we use PKI for worldwide key distribution and achieve "one certificate/one vote" policy. Therefore, anyone can participate in the voting if he gets a certificate from CA.

The rest of this paper is organized as follows: In Section 2, we will review security requirements of Internet voting system discussed in the literature. We will briefly describe related works done till now and cryptographic components to design Internet voting system in Sections 3 and 4, respectively. In Section 5, we will discuss the general architecture of our voting system and show how each stages are designed. In Section 6, we will introduce typical implementation including building stuffs, cryptographic library and network connections. In Section 7, we sketch a typical voting applet to select MVPs of 2002 FIFA World Cup Korea-Japan^{TM} implemented till now. Finally, concluding remarks will follow in Section 8.

II. Security Requirements

Many extensive researches on electronic voting have been conducted and now an extensive list of security requirements for electronic voting is available. In general, we can classify the security requirements of electronic voting protocol into the following two criteria [6], [16], [19], [20], [21], [33]:

Basic Requirements

- Privacy: All votes should be secret.
- Completeness: All valid votes should be counted correctly.
- Soundness: Anyone cannot disturb the voting.
- Unreusability: All voters can vote only one.

The authors are with IRIS (International Research center for Information Security), Information and Communication University (ICU), 58-4 Hwaam-dong, Yusong-gu, Taejon, 305-732, Korea.
E-mail: {kkj,kman,sultan,tachyon}@icu.ac.kr
Web-site: http://www.iris.re.kr

- Eligibility: Anyone who is eligible can vote.
- Fairness: Noting can affect the voting.

Most electronic voting systems must meet these basic requirements.

Extended Requirements

- Walk-away: The voter need not to make any action after voting.
- Robustness: The voting system should be successful regardless of partial failure of the system.
- Universal verifiability: Anyone can verify the validity of the whole voting process.
- Receipt-freeness: Voter should not be able to prove his or her vote to a buyer. Voter does not have any receipt for the vote to prevent vote-selling.

The universal verifiability and receipt-freeness are of great cryptographic interest, but will cost a lot for real implementation in practice. To the best of our knowledge, there is no single ideal scheme which satisfies all the requirements described above. Safevote, Inc. [4] presents also a set of requirements of voting system applicable to paper, electronic and Internet voting together.

Since our goal is to design and implement a Internet voting system to be useful in practice, some security requirements like universal verifiability and receipt-freeness can be ignored. Our design mainly focuses to provide high efficiency and low delay in Internet voting to normal users with meeting all the basic requirements as well. This enables more voters from any place can join our voting system at any time.

III. Related Works

Numerous researches have been done to construct secure and efficient voting systems: schemes based on homomorphic encryption [5], [11], [12], [32], schemes based on Mix-net [1], [2], [24], [31], and schemes based on blind signature [9], [16], [23]. However, there is no perfect solution satisfying every requirements together with high efficiency. The schemes based on homomorphic encryption are applicable only to yes-no voting. The schemes based on Mix-net cost very much to guarantee that every ballot is opened correctly when the number of voters is large. One of the standard schemes using blind signature was proposed by Fujioka *et. al.* [16] ("FOO92" in short) in 1992. FOO92 use blind signatures to satisfy the privacy and unreusability property, and the bit-commitment scheme to realize the fairness property. It is efficient in computation and is applicable to multiple choices in very flexible way. But a considerable obstacle caused by using bit-commitment scheme in FOO92 is that all voters have to join the ballot counting process. This means that it is not practical to apply FOO92 for the real world, since each voter must stay until all other voters complete the voting stage. An improved scheme was proposed by Ohkubo *et. al.* [23] ("OMAFO99" in short) in 1999. They proposed a practical blind signature-based voting scheme that allows voters to walk away once they finish casting their votes. They use a threshold encryption scheme to solve walk-away problem instead of using bit-commitment protocol. In this scheme the ballots are encrypted with server's public key and the threshold decryption of ballot is conducted by multiple servers. We will explain this scheme in more detail later.

On the other hands, there have been several practical implementations as a trial for replacing ordinary voting by electronic voting.

By Cranor and Cytron [10], a practical, secure and private system called Sensus for polling (conducting surveys and elections) over computer network has been designed and implemented by expanding FOO92. Since Sensus is a direct implementation of FOO92, each voter can not walk away until all other voters complete their voting. Moreover, It is assumed that voters communicate with the administrator and the tallier via a special anonymous channel that can transmit messages to both sides. This channel is not implemented in Sensus.

A research group at the Laboratory for Computer Science, MIT, had implemented a system called EVOX [18] based on FOO92. While the system has been used at MIT for Undergraduate Associates elections, EVOX still possesses certain vulnerabilities using only single administrator. In 1999, a second branch [14] of the EVOX system has been created which uses multiple administrators for vote signing. This improve the security by preventing the administrator from forging votes. In the EVOX system, they use an Anonymizer which can be regarded as a single mix server to realize anonymous channel. Furthermore, the walk-away problem is solved by sending the vote and commitment at the same time. But, the completeness property can be broken by the malicious tallier, since only one tallier count votes.

These implementations have same drawback, key distribution that is a traditional problem in cryptography. They assume that all necessary public keys and private keys are generated and stored in a secure way. The assumption can be an obstacle to apply these implementations to the real world. We use PKI to solve key distribution problem. This can be a good way to expand the number of voters easily over the Internet.

IV. Cryptographic Components

To enhance the readability of this paper to the readers, we will describe cryptographic components briefly which are used to design our Internet Voting Systems.

ElGamal Public Key Encryption

The ElGamal public key encryption is a widely used public key cryptosystem whose security is based on the intractability of the discrete logarithm problem [15] and Diffie-Hellman problem [13]. First, an entity A creates a public key and its corresponding private key as follows: generates large primes p and q with $q|p-1$ and a generator g of Z_p^* of order q, and creates a pair of keys, $x \in Z_q^*$ and $y = g^x \bmod p$. Other entity B can encrypt a message m for A as $(G, M) = (g^a, m \cdot y^a)$ where a is a random number. A can decrypt the ciphertext (G, M) to get the original message $m = M/G^x \bmod p$.

Rijndael Block Cipher

In our system, we use a hybrid scheme [22] to handle long size of input message that exceed the modules size of underlying public key encryption. We choose a block cipher, Rijndael [26] which was chosen as Advanced Encryption Standard (AES) algorithm recently by NIST. It is said that Rijndael resists against all known attacks, providing speed and code compactness on a wide range of platforms and simplicity of design.

Schnorr Signature

The Schnorr signature [25], [27] is used in our system. Key generation for the Schnorr signature scheme are as follows: an entity A generates large primes p and q with $q|p-1$ and a generator g of Z_p^* of order q, and creates a pair of keys, $x \in Z_q$ and $y = g^{-x} \bmod p$. A publish y and keeps x secret. The signature of a message m is a form as (e, s), where $r = g^k \bmod p$ with a random $k \in Z_q^*$, $e = H(m, r) \bmod q$ and $s = k + ex \bmod q$. We employ SHA-1 as a hash function H. Other entity B can verify the signature as $e = H(m, g^s y^e \bmod p)$.

Schnorr Blind Signature

Blind signature schemes [8] are two party protocols between a sender A and a signer B. The purpose of a blind signature is to prevent the signer B from observing the message it signs and its resulting signature.

Our voting system use a blind signature proposed by Schnorr [25]. Key generation for the Schnorr blind signature is the same as the Schnorr signature. To get the signature of a secret message m, A asks B to initiate a communication. B chooses a random $k \in Z_q^*$, and sends $r = g^k \bmod p$ to A. A blinds this value with two random elements $\alpha, \beta \in Z_q^*$ into $r' = rg^{-\alpha}y^{-\beta} \bmod p$, computes $e' = H(m, r') \bmod q$ and sends $e = e' + \beta \bmod q$ to B. B returns s such that $g^s y^e = r \bmod p$. Finally, A computes $s' = s - \alpha \bmod q$. This way, the pair (e', s') is a valid Schnorr signature of m since it satisfies $e' = H(m, g^{s'} y^{e'} \bmod p)$.

Threshold Scheme

In 1979, Shamir at first defined the notion of threshold scheme [28] (often called a secret sharing scheme). The (k, n) threshold scheme proposed by Shamir is to divide a secret D into n pieces $D_1, D_2, \ldots, D_n$ in such a way that:

- Knowledge of any k $(k < n)$ or more D_i pieces makes D easily computable.
- Knowledge of any $k-1$ or fewer D_i pieces leaves D completely undetermined.

A piece of a secret is called share. The scheme originally proposed by Shamir is based on polynomial interpolation.

Anonymous Channel

An anonymous channel is a kind of communication channel that hides the traffic pattern of the network. It hides the correspondences between the senders and messages. D. Chaum introduced a secure anonymous channel called a mix-net [7]. The mix-net consists of a series of mix servers, which for inputs of a list of ciphertexts output a permuted list of items. The outputs are either the permuted plaintexts that correspond to the input ciphertexts, or the permuted ciphertexts that correspond to the same plaintexts as the input ciphertexts. The important function of mix-net is that the anonymity of the output is guaranteed as long as at least one server works honestly.

Although a universally verifiable mix server can be used for providing reliable proof of correctness, it is very complex and inefficient. We choose a simple mix server using hybrid scheme [22] that combine asymmetric key exchange and symmetric encryption. Rijndeal is used as a symmetric encryption in our system as stated earlier.

PKI

Public key cryptography plays an important role in providing security services such as confidentiality, authentication, digital signatures, and integrity. Public key cryptography uses a pair of keys: public and private. These keys are mathematically related, but the private key cannot be derived from the public key. The public key can be known by anyone, but the private key is kept secret by its owner. For the public key cryptography to be widely used in applications, the ability to verify the authenticity of public key is required. This is achieved by the use of certificate, which provides a means to bind a public key to its owner. The certificate contains certification information such as owner's name, the associated public key, and validity period, and are issued by a trusted Certification Authority (CA).

V. Internet Voting System Architecture

In this Section, we describe overall architecture and detailed protocol steps of our voting system. We assume that the voters trust the admin server completely, and anybody can post, but nobody can erase or overwrite the data once written in the bulletin board. We use some cryptographic primitives such as threshold encryption, digital signature, blind signature, and mix-net as stated before. As shown in Fig. 1, Internet voting system architecture consists of six basic entities; voter, admin, bulletin board, mix server, tally server and certification authority.

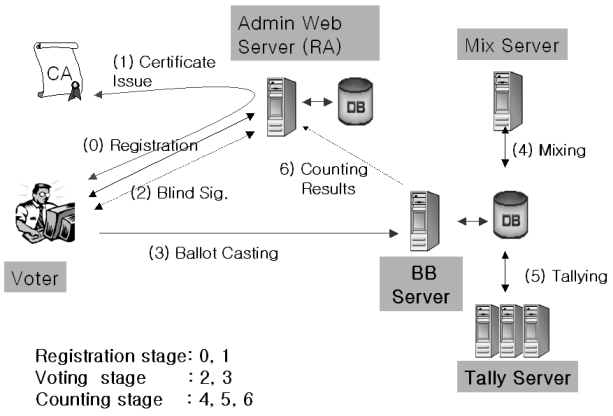


Fig. 1. Internet voting system architecture.

Throughout this paper we use the following notations:
V_i: voter i
C_i: V_i's certificate

AS: Admin server
M: Mix-server
T: Talliers (T_j denotes the tallier j)
BB: Bulletin board and ballot box
DB: Database
E_T: T's threshold encryption scheme
D_T: T's threshold decryption scheme
E_M: M's encryption scheme
D_M: M's decryption scheme
S_i: V_i's signature scheme
S_A: AS's signature scheme
B: Blinding procedure
UB: Unblinding procedure
ID_i: V_i's identification
v_i: Vote of V_i

A. Voting protocol

We choose OMAFO99 for our Internet voting system since it is efficient in terms of computation and its typical implementation is available for large scale voting. The basic security of our system is based on the difficulty of solving discrete logarithm uniformly.

OMAFO99 inherits most of the security properties of FOO92, but the major differences are the number of talliers and the use of threshold encryption scheme instead of using bit-commitment scheme. The relation between the voter's identity and ballot is sealed by the blind signature scheme. Ballot is sent through an anonymous channel, so no one can violate voter privacy. Unreusability and eligibility also hold under the assumption that no voter can break the blind signature scheme and the ordinary digital signature scheme. The most important improvement is the walk-away property which the voters may leave away after casting their votes. They need not send any information to talliers to open their ballots because they encrypt their votes with the tallier's public key. Furthermore, fairness is also assured because malicious talliers that are less than threshold t can not decrypt the ballots in the middle. Robustness is assured under assumption that the number of colluding authorities don't exceed a predetermined threshold. However, universal verifiability and receipt-freeness is not guaranteed in OMAFO99.

B. Internet Voting Stages

As shown in Fig. 1, our system consists of three stages: registration, voting, and counting stages. Before beginning these stage, system parameters and all entities' key pairs except voters should be generated and distributed by using PKI.

Registration Stage

(R1) V_i accesses AS to download registration form and inputs his information required for certificate issuing. The information is encrypted with AS's public key and is sent to AS. Then AS checks that V_i has the right to vote after decrypting the information. If V_i doesn't have the right, AS gives an error message. Otherwise, AS gives V_i the right to download key generation applet.

(R2) After downloading key generation applet and generating key pairs, V_i keeps his private key in safe storage and sends his public key to AS to request C_i.

(R3) AS requests C_i issuing to CA. CA issues a certificate to V_i. V_i stores C_i in safe storage.

Voting Stage

(V1) After downloading login applet to enter voting stage, V_i provides authentication data (ID and password). AS checks whether the voter has already voted or not. If V_i had already voted, AS rejects the authorization. Otherwise, AS gives V_i the right to download voting applet.

(V2) After downloading the voting applet, V_i selects vote v_i of his choice and encrypts v_i with T's public key of the threshold encryption scheme as $x_i = E_T(v_i)$. V_i blinds x_i as $e_i = B(x_i, r_i)$, where r_i is a randomly chosen blinding factor. V_i signs e_i as $s_i = S_i(e_i)$ and sends (ID_i, e_i, s_i) to AS.

(V3) AS verifies the signature s_i of message e_i. If s_i is valid, then AS signs e_i as $d_i = S_A(e_i)$ and sends d_i to V_i. At the end of the voting stage, AS announces the number of voters receiving AS's signature, and publishes the final list as (ID_i, e_i, s_i).

(V4) V_i retrieves the desired signature y_i of ballot x_i by $y_i = UB(d_i, r_i)$. V_i checks whether y_i is AS's signature for x_i. If this check fails, V_i claims it by showing that (x_i, y_i) is invalid.

(V5) V_i encrypts (x_i, y_i) with the encryption key of the mix-net as $c_i = E_M(x_i, y_i)$ and sends the resulting doubly encrypted ballot to BB via a sender authenticated public channel (signed message and certificate by the voter are sent).

(V6) BB checks the signature of the posted message by using V_i's certificate.

Counting Stage

(C1) M decrypts the list of c_i and outputs the list of (x_i, y_i) in random order.

(C2) T_j verifies the signature y_i of x_i. If the verification fails, T_j claims that y_i is not a valid signature of x_i by publishing (x_i, y_i). If more than t talliers agree with T_j on (x_i, y_i), M has to prove in zero-knowledge that (x_i, y_i) is the correct result of decryption of c_i. T_j checks the proofs issued by M. If the checks fails, it means that M issued wrong proof. If all proofs are valid, which means that M works fairly, then the voter has cast an invalid vote. Thus, the vote is excluded from further steps of the counting stage.

(C3) T cooperatively decrypts ballot x_i and retrieves vote v_i as $v_i = D_T(x_i)$ in communication via the bulletin board. T publishes the voting results by using BB.

C. System Architecture

As mentioned before, the system consists of six entities; V_i, AS, BB, M, T, and CA. Since only a person having certificate can get the right to vote, V_i should register at AS to get C_i before voting. The registration stage in Fig. 2 shows the certificate issuing procedures in detail.

AS is responsible for verifying the voter's right to vote and authenticating the ballot. It must permit only one vote

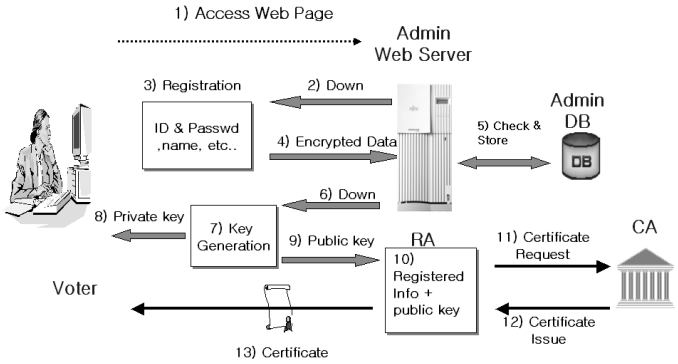


Fig. 2. Registration stage.

per eligible voter. AS can play the role of a registration authority (RA) when issuing C_i. After successful registration, AS requests CA to issue C_i for each registered person. After receiving C_i from CA and storing it, AS sends C_i to V_i. C_i is issued in simplified form based on X.509v3.

After receiving C_i, V_i can vote by using voting applet which is executed in web browser. When the program is downloaded after entering his ID and password, V_i simply needs to click on his choice and click on the "Vote" button. Then, the applet communicate automatically with AS to get AS's blind signature. Finally, doubly encrypted ballot, V_i's signature, and C_i are posted to BB.

BB is used as a public communication channel that can be read by any entity. Each legitimate entity can write message only on its designated section. No entity can erase any information from BB. When receiving encrypted ballot and V_i's signature, BB verifies V_i's signature and store the encrypted ballot to DB.

After the voting time is over, M mixes doubly encrypted ballots by voters and decrypts these mixed ballots. T receives encrypted ballots from M and opens them by using threshold decryption protocol. Finally, T publishes the results by using BB.

VI. Typical Implementation

In order to implement Internet voting system efficiently, we try to use built-in components produced by Korean security industries and extend their functions to meet our objectives.

We choose to use the CA server by KSIGN [36], one of CA vendors in Korea, and the Java crypto library J/LOCK by STI [38]. Insol Soft [34] is responsible for web interface for voters and SECUi.COM [37] provides security management of main computer and security measures of network. Imai laboratory at the University of Tokyo is responsible for checking correctness and vulnerability of our system.

A. Servers

AS and BB can be implemented on Unix system using Apache as a web server and Tomcat as a servlet container and JavaServer Pages™ (JSP) implementation. We have developed the main part of AS and BB by using JSP, JDK1.2, and Java crypto library. Oracle DB is used for AS to manage a huge number of information of all voters. BB also use an independent DB to handle ballots. Since JDBC (Java Database Connectivity) and standard SQL queries are used for handling DB, we can use other database systems such as Informix, Oracle, Sybase, Microsoft, and so on. M and T are implemented in C language on a Linux system. We use only one mix server for efficiency.

The system environments can be as follows:

- M, T : Intel Pentium II 866Mhz, Linux kernel 2.2.17
- AS, BB, CA : Sun Ultra-Sparc 550Mhz, Solaris 2.8
- DB : Oracle 8.0

B. Voting Applet

The voting applet is a downloadable program code and is executed in voter's web browser supporting Java, which contains necessary information to support the actual candidate selections. The voter does not need to download any code ahead of time. The voting stage in Fig. 3 shows the action of the voting applet.

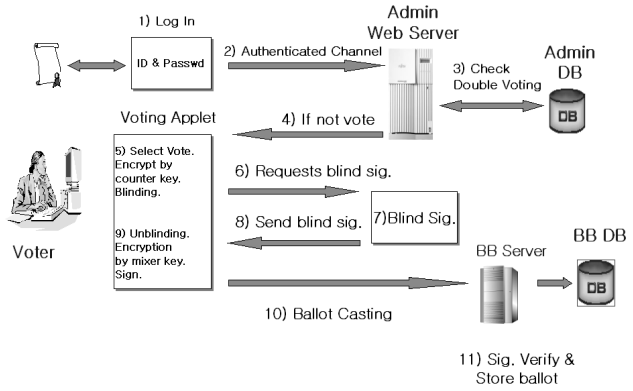


Fig. 3. Voting stage.

The voting applet needs permission to open connections to multiple addresses and to access a secret file containing voter's private key. A simple and secure way to achieve this is using the functionality of signed applet in JDK which allows safe downloading and execution of the applet. Because Java applet is running inside sandbox, there is no need to worry about system security.

Now, the target system where the voting applet can operate is limited to Window OS on PC because the location of the secret key can be stored safely.

C. Cryptographic Library

In addition to J/LOCK by STI, we utilize the previously existing crypto-library in C language implemented by NTT [17], which has once used to implement Internet voting system. The J/LOCK library includes a class and interface hierarchy for keys, encryption, decryption, blinding, and signature. We have implemented the necessary C-library for ElGamal public key cryptosystem and Rijndael. Other cryptographic primitives such as threshold encryption scheme, Schnorr digital signature scheme,

Schnorr blind signature scheme, and mix-net scheme are implemented.

D. Network Connections

Java provides basic networking functionality. This includes the ability to create sockets which are flexible and sufficient for general communication. Also, it supports RMI (Remote Method Invocation) which is the action of invoking a method of a remote interface on a remote object. In the Java platform's distributed object model, a remote object is one whose methods can be invoked from another Java virtual machine, potentially on a different host. Since objects are serializible, we can send object streams through sockets or RMI. By using this mechanism, we can make secure connections without SSL connection.

We offer two types of connections, simple and secure connections. A simple connection is provided directly by JDK sockets, but a secure connection is provided by a cryptographic layer.

VII. Application

Currently, we are applying our Internet voting system to select MVPs of 2002 FIFA World Cup Korea-Japan^{TM}, which will be held from May 31 to June 30, 2002 at major cities in Korea and Japan. This application aims to demonstrate electronic voting technology to the world in easy and friendly manner with joint work by Korean and Japanese teams.

Because this system will be open to general people all over the world via the Internet, a huge amount of data handling are expected. To overcome this bottleneck, we will use supercomputer running by KISTI [35], Korea.

As an example, the voting applet illustrated in Fig. 4 provides two choices, the MVP and the best goalkeeper. To vote for a player, voter firstly choose the country where the player belongs and then selects a player in the player list of the country.

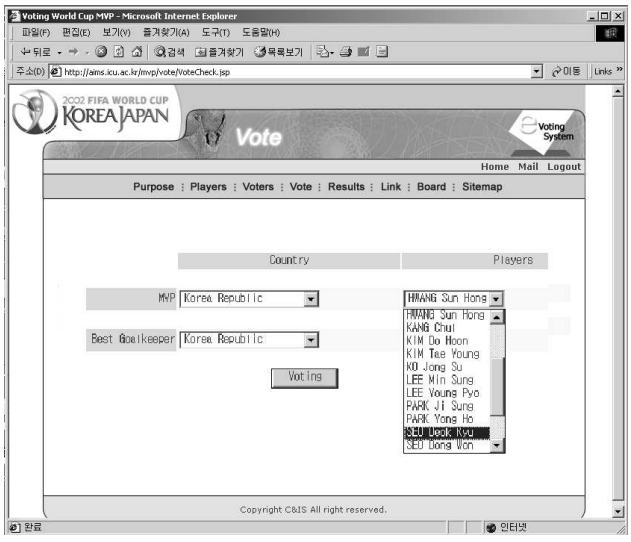


Fig. 4. Voting applet for 2002 worldcup Korea-Japan.

We expect the Internet voting system will be served at http://mvp.worldcup2002.or.kr.

VIII. Concluding Remarks

We have designed the Internet voting system using PKI. To the best of our knowledge, this is the first user-friendly, secure Internet voting system using PKI. Now we have finished to implement our voting system inside LAN. Further works like porting job to supercomputer, system security measures, network security measures, and performance test need to be done. In our system, voter's private key will be stored in the hard disk of PC in a secure way assuming that most voters use PC, but it can be vulnerable to any malicious attack. If smart card is used to store voter's private key, the problem will be improved a lot. Since voters will connect to our system in remote way from all over the world, it is very hard to authenticate voters in non face-to-face situation. An efficient authentication mechanism for eligible voters is under development now. Also, the fault tolerance of voting host, traffic bottleneck due to limited bandwidth and computing power and the maximum number of voters to be allowed need to be carefully designed.

If we succeed in serving the Internet voting system to the world, we can yield very important experiences such as:

- Worldwide level application of cryptographic technology.
- Contribution to the development of information security related-industry such as PKI.
- Typical example of international cooperation between 2 countries: Korea and Japan.
- Many feedbacks and valuable lessons to the planned Internet voting systems, etc.

References

[1] M. Abe, *Universally verifiable mix-net with verification work independent of th number of mix-servers*, Advances in Cryptology-Eurocrypt'98, LNCS Vol. 1403, pp.437–447, Springer-Verlag, 1998.

[2] M. Abe, *Mix-network on permutation networks*, Advances in Cryptology-Asiacrypt'99, LNCS Vol. 1716, pp.258–273, Springer-Verlag, 1999.

[3] The BELL Newsletter on Internet Voting, The Bell, Vol.1 No.4, Safevote Inc., Aug. 2000.

[4] *Internet Voting Requirements*, The Bell, Vol.1 No.7,p3, Safevote Inc., Nov. 2000, http://www.thebell.net/papers/vite-reg.pdf

[5] J. Benaloh, *Verifiable secret-ballot elections*, Ph.D. thesis, Yale University, Department of Computer Science, YALEU/CDS/TR-561, December 1987.

[6] J. C. Benaloh and D. Tuinstra, *Receipt-free secret ballot elections*, Proc. of 26th ACM STOC, pp.544–553, 1994.

[7] D. Chaum, *Untraceable electronic mail, return addresses, and digital pseudonyms*, Comm. ACM 24, pp.84–88, 1981.

[8] D. Chaum, *Blind Signatures for Untraceable Payments*, Advances in Cryptology-Crypto'82, pp.199–203, Springer-Verlag, 1983.

[9] D. Chaum, *Elections with unconditionally- secret ballots and disruption equivalent to breaking RSA*, Advances in Cryptology-Eurocrypt'88, LNCS Vol.330, pp.177–182, Springer-Verlag, 1988.

[10] L.F. Cranor and R.K. Cytron, Sensus : A security-conscious electronic polling system for the Internet, Proc. of the Hawai International Conference on System Sciences, Jan. 7–10, 1997, http://www.research.att.com/ lorrie/pub/hicss/

[11] R. Cramer, M. Franklin, B. Schoenmakers, and M. Yung, *Multi-authority secret ballot elections with linear work*, Advances in Cryptology-Eurocrypt'96, LNCS Vol.1070, pp.72–83, Springer-Verlag, 1996.

[12] R. Cramer, R. Gennaro, and B. Schoenmakers, *A secure an optimally efficient multi-authority election schemes*, Advances in Cryptology-Eurocrypt'97, LNCS Vol.1233, pp.103–118, Springer-Verlag, 1996.

[13] W. Diffie and M. Hellman *New directions in cryptography*, IEEE Trans. on Information Theory Vol.22, No.6, pp.644–654, 1976.

[14] Brandon William DuRette, *Multiple administrators for electronic voting*, Bachelor Thesis, MIT, May 1999, http://theory.lcs.mit.edu/~cis/theses/DuRette-bachelors.pdf

[15] T. ElGamal, *A public key cryptosystem and a signature scheme based on discrete logarithms*, Advances in Cryptology-Crypto'84, LNCS Vol.196, pp.10–18, Springer-Verlag, 1985.

[16] A. Fujioka, T. Okamoto and K. Ohta, *A practical secret voting scheme for large scale election*, Advances in Cryptology-Auscrypt'92, LNCS Vol.718, pp.248–259, Springer-Verlag, 1993.

[17] A. Fujioka, M. Abe, M. Ohkubo, and F. Hoshino, *An Implementation and an Experiment of a Practical and Secure Voting Scheme*, Proc. of SCIS2000, C48, Okinawa, Japan, Jan. 26–28, 2000.

[18] M. Herschberg, *Secure electronic voting using the world wide web*, Master's Thesis, MIT, June 1997. http://theory.lcs.mit.edu/~cis/voting/herschberg-theis/

[19] B. Lee, and K. Kim, *Receipt-free electronic voting through collaboration of voter and honest verifier*, Proceeding of JW-ISC2000, pp.101–108, Jan. 25-26, 2000, Okinawa, Japan.

[20] M. Maichels and P. Horster, *Some remarks on a recirpt-free and universally verifiable mix-type votins scheme*, Advances in Cryptology-Asiacrypt'96, LNCS Vol.1163, pp.125–132, Springer-Verlag, 1996.

[21] V. Niemi and A. Renvall, *How to prevent buying of voters in computer elections*, Advances in Cryptology-Asiacrypt'94, LNCS Vol.917, pp.164–170, Springer-Verlag, 1994.

[22] M. Ohkubo, M. Abe *A Length-Invariant Hybrid Mix*, Advances in Cryptology-Asiacrypt'00, LNCS Vol.1976, pp.178–191, Springer-Verlag, 2000.

[23] M. Ohkubo, F. Miura, M. Abe, A. Fujioka and T. Okamoto, *An Improvement on a Practical Secret Voting Scheme*, Information Security'99, LNCS Vol.1729, pp.225–234, Springer-Verlag, 1999.

[24] C. Park, K. Itoh, and K. Kurosawa, *Efficient anonymous channel and all/nothing election scheme*, Advances in Cryptology-Eurocrypt'93, LNCS Vol.765, pp.248–259, Springer-Verlag, 1994.

[25] D. Pointcheval and J. Stern, *Security Arguments for Digital Signatures and Blind Signatures*, Journal of Cryptology, LNCS Vol.13, Num.3, pp.361–396, Springer-Verlag, 2000.

[26] J. Nechvatal, E. Barker, and at. al., *Report on the Development of the Advanced Encryption Standard (AES)*, NIST publication, NIST, Oct. 2000.

[27] C. P. Schnorr, *Efficient Identification and Signatures for Smart Cards*, Advances in Cryptology-Crypto'89, LNCS Vol.435, pp.235–251, Springer-Verlag, 1990.

[28] A. Shamir, *How to share a secret*, Comm. ACM Vol.22, pp.612–613, 1979.

[29] M. I. Shamos, *What's happening in Florida ? Bugs in Computerized Voting*, CMU Distinguished Lecture, Nov., 2000.

[30] CALTECH-MIT/Voting Technology Project, Dec, 2000, http://www.vote.caltech.edu/

[31] K. Sako and J. Killian, *Receipt-free Mix type voting scheme - a practical solution to the implementation of a voting booth*, Advances in Cryptology-Eurocrypt'95, LNCS Vol.921, pp.393–403, Springer-Verlag, 1995

[32] K. Sako and J. Killian, *Secure voting using partially compatible homomorphisms*, Advances in Cryptology-Crypto'94, LNCS Vol.839, pp.411–424, Springer-Verlag, 1994

[33] B. Schoenmakers, *A simple publicly verifiable secret sharing scheme and its application to electronic voting*, Advances in Cryptology-Crypto'99, LNCS Vol.1666, pp.148–164, Springer-Verlag, 1999

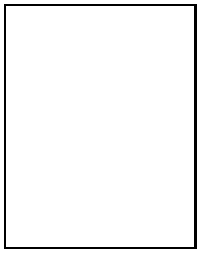
[34] Insol Soft, In *http://www.insolsoft.com/index2.html*

[35] Korea Institute of Science and Technology Information, In *http://www.kisti.re.kr*

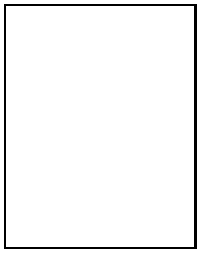
[36] KSIGN Co., Ltd. KSignCA. In *http://www.ksign.com*

[37] SECUi.COM In *http://www.secui.com*

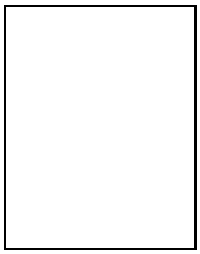
[38] Security Technology Inc. (STI), *J/LOCK* In *http://www.stitec.com*



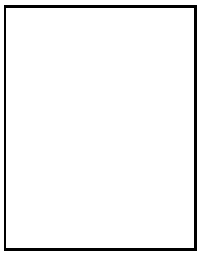
Kwanjo Kim received the B.S. and M.S. degrees in Electronic Engineering from Yonsei University, Seoul, Korea in 1980 and 1983 respectively. In 1991, he received Ph.D. degree in Electrical and Information Engineering at Yokohama National University, Japan. In 1979, he joined ETRI (Electronics and Telecommunications Research Institute) and had been engaged in research and development of various applications of cryptographic technology. In 1998, he moved to ICU (Information and Communications University), Taejon, Korea, as a faculty member. He is currently an associate professor in information security group and a director of IRIS (International Research center for Information Security) in ICU. He served as program co-chair of Asiacrypt'96 and the program chair of PKC2001 and served as program committee member of numerous international conferences. His research interest includes all fields of cryptography and information security. He is a director of IACR (International Association for Cryptologic Research) and a member of KIISC (Korea institute of information security and cryptoloy), IEICE and IEEE. He serves as an editor of JCN and IJIS.



Jinho Kim received the B.S. degree in Mathematics and Computer Science from Pohang University of Science and Technology (POSTECH), Pohang, Korea in 2000. He is currently a M.S. student in the information security group and a researcher of IRIS in ICU, Taejon, Korea. His research interest includes electronic voting system and information security.



Byoungcheon Lee received the B.S. and M.S. degrees in Physics from Seoul National University, Seoul, Korea in 1986 and 1988, respectively. He worked for LG Cable Co. from 1988 to 1993 and for LG corporate institute of technology from 1993 to 1998 as a researcher. He is currently a Ph.D. student in the information security group and a researcher of IRIS in ICU, Taejon, Korea. His research interest includes cryptography and information security. He is a member of KIISC (Korea institute of information security and cryptoloy).



Gookwhan An received the B.S. degree in Computer Engineering from Hongik University, Seoul, Korea in 1992. He worked for Samsung Data Systems from 1993 to 1999. He is currently a M.S. student in the information security group and a researcher of IRIS in ICU, Taejon, Korea. His research interest includes computer and network security.