**A Practical and Secure Electronic Election System**

Yunho Lee and Dongho Won

E-voting is expected to offer prominent advantages over traditional voting methods. Advantages include faster tallying, greater accuracy, prevention of void ballots, and lower cost. However, many experts express concerns about the potential for large-scale fraud. Recently, many paper based end-to-end (E2E) voter verifiable systems providing individual verifiability and universal verifiability have been proposed. These systems, unlike previous voting systems, are used in polling booths and without access to trusted computing devices at the time of voting. In this paper, we propose a practical and secure E2E voter verifiable system using a paper receipt based on cryptographic technologies.

**Keywords:** Electronic voting, DRE, cryptographic receipt, individual verifiability, universal verifiability.

**I. Introduction**

Plain paper-based voting is the most widely used method of casting votes world-wide. Plain paper-based voting systems include plain paper ballot voting systems, punch card voting systems, and optical mark-sense voting systems. The alleged problems with plain paper ballots are the time required to count them and the danger of incorrect counting results. Alternatives such as punch cards and optical mark-sense voting systems require voters to cast ballots before officials can tabulate votes. However, a punch card voting system is not completely free from counting problems, as evidenced in Florida’s 2000 US Presidential election. Similarly, with an optical mark-sense voting system, voters commonly mark a ballot in a way that cannot be scanned correctly.

While many experts believe that a paperless voting system, that is, e-voting system, offers definite advantages over plain paper voting systems, others express concerns about the potential for large-scale fraud. Many current e-voting systems are not based on cryptographic techniques and require voters to trust them. Voters cannot be assured that the vote cast accurately reflects their intent unless voting machines provide proof or confirming evidence of their honesty; however, in current machines, the mechanism that records the vote is hidden in the code of the machine. Thus, Mercuri [1] insisted that an e-voting machine should produce paper audit trails enabling voters to examine their vote and that the paper audit trails should be deposited to a secure ballot box for later recounting. The former examination is for individual verifiability, and the later recounting is for universal verifiability.

Although Mercuri’s system can limit voter distrust in voting machines, it increases the maintenance costs and election complexity for the poll workers. Moreover, any discrepancies between results reported by the voting machine and the paper audit count would likely lead to controversy. Obviously, the
cause of this discrepancy is that a vote cast is recorded by two independent mechanisms. We propose that a vote cast should be recorded only by an electronic mechanism and, the paper audit trail retained by the voter should not be used for recounting but for verification that a vote was recorded correctly.

Much research has focused on universal verifiability, and many efficient and secure schemes have been proposed based on a mix-network or blind signature [2], [3]. However, in our opinion, studies of efficient and secure methods of achieving individual verifiability are insufficient. Satisfying individual verifiability assures that votes are recorded and counted correctly. The most effective way of increasing voters’ confidence is to provide audit trails to voters that can be retained as receipts [4]-[6]. A receipt is a proof to show that a vote cast has been recorded correctly. Nevertheless, the receipt should not be used to prove a voter’s vote; this property is termed receipt-freeness. One simple method to validate a receipt is to decrypt the content. However, this method is impractical for the following two reasons:

i) For elections, a decryption key, which could be distributed to several authorities, would be reconstructed only during tallying. We fear, though, that decryption could be abused for vote buying or selling.

ii) Even if the decryption key is available only at the voting booth, voters must trust in the decryption device to verify the voting machine. Obviously, it is unreasonable to require voters to trust a device at the voting booth.

In 2002, Chaum proposed a method for a voter-verifiable paper trail based on visual cryptography [4], [5]. The scheme enables a voter to be convinced that the voting machine correctly recorded her intent with a probability of 50% and prevents voters from proving how they voted (eliminating the danger of using the method as a tool for vote buying/selling). However, Chaum’s scheme requires special printers and lacks a high verification probability. Although several schemes for voter-verifiable paper trails satisfying individual verifiability and receipt-freeness were subsequently proposed, they require preprocessings, such as the creation of a large number of ballot forms or a codebook [7]-[9], and for voters to trust additional devices [10], [11].

In this paper, we propose a practical and secure scheme to issue and verify receipts using the well-known cut-and-choose method. In the proposed scheme, the receipt verification is composed of two distinct verifications. The first compares two code words printed on the ballot, the second verifies receipt after leaving the polling station. The advantages of our scheme are as follows.

First, only one comparison between two strings is required by a voter in the voting booth. That is, voters do not have to put trust in any devices in the polling station. Furthermore, the strings to be compared are composed of only two alphanumeric characters.

Second, anyone can be assured that the encrypted vote cast reflects the voter’s intent with probability of at least 50% via a simple and public method. Conversely, only the voter who saw the ballot form can know the voter’s intent, so the proposed system satisfies individual verifiability and receipt-freeness.

The remainder of this paper is organized as follows. Section II reviews the previous works, Prêt à Voter and PunchScan. In sections III and IV, we propose a practical and secure e-voting scheme and improvement of it, respectively. Section V analyzes its security and presents comparison results.

II. Previous Works

Over the past decades, numerous voting systems have been proposed: Helios [12], ThreeBallot [13], Civitas [14], VoteBox [15], SpeakUp [16], Scantegrity [17], and so on. In this section, we will briefly review Prêt à Voter [10] and PunchScan [18].

1. Prêt à Voter

In 2005, Chaum and others proposed a practical voter-verifiable election scheme, Prêt à Voter [10]. Their scheme uses a more conventional representation of the vote, that is, ballot forms with the candidates listed in the left column, and the voter choices marked in an adjacent right column.

A. Election Setup

An authority prepares a large number of ballots, significantly more than the number of the electorate. The candidate order in a ballot should be randomized and unpredictable. A ballot also contains information encrypted by the public key(s) of the predetermined teller(s), and the candidate ordering can be reconstructed using that information (Fig. 1).
B. Casting the Vote

A voter marks her ‘✓’ in the usual way at a voting booth. Suppose that she selects the second ballot (index = 2) and casts her vote to the candidate “Alice.” The ballot will be as shown in Fig. 2(a). She then removes the left hand strip (for shredding), and feeds the right hand strip into the voting machine (a kind of scanner) (Figs. 2(b) and 2(c)).

C. Tallying

The teller(s) should reconstruct the candidate ordering for each vote cast for tallying. The reconstruction of candidate ordering can be done by decrypting the information.

D. Analysis

There are three serious drawbacks to this scheme.

i) The candidate order printed on the ballot is not fixed.

ii) The teller(s) should decrypt the transmitted information.

iii) Voters are required to trust the teller(s) in the polling station.

The decryption operation should never be performed during the voting period because the decryption oracle can be misused by voters (attackers) to prove their selections. Furthermore, the fact that a voter has to trust her ballot form’s integrity is not ideal. In a polling station, a voter should never have to trust any devices.

Additionally, a large number of ballots, significantly more than the number of the electorate, should be prepared carefully. The cost for ballot preparation would be significant because each ballot would contain randomized candidate ordering and a unique ciphertext of the ordering.

2. PunchScan

In 2005, Popoveniuc and Hosp proposed a voter verifiable e-voting scheme, PunchScan [18]. There are three phases of the PunchScan, the pre-election phase, election day, and the post-election phase.

A. Pre-election Phase

During this phase, the election authority (EA) generates ballots and candidates check them. A ballot is composed of two pages, the top page and bottom page (Fig. 3).

B. Election Day

On an election day, a voter enters the voting booth and marks her favorite candidate on the ballot and chooses to keep one page of the ballot. The selected page is scanned, recorded, and made public, and the other one should be shredded (Fig. 4).

C. Post-election Phase

After closing the election, the election is audited and proofs are carried out to ensure the integrity of the election. For more details, refer to [18].

D. Analysis

A large number of ballots are required to make the system secure, and the system cannot prevent voters from spoiling their votes.
III. Proposed Technique

In 2010, Lee and others proposed a secure electronic voting system based on paper receipts satisfying individual verifiability and universal verifiability. In this section, we review the basic process of the system. For more details, see [19].

Suppose that there are \( n \) candidates. Then, the election authorities assign two public symbols, \( S_i^L \) and \( S_i^R \), to each candidate \( i (1 \leq i \leq n) \). Let \( E(\cdot,\cdot) \) and \( D(\cdot) \) denote probabilistic encryption/decryption, for example, the ElGamal cryptosystem.

1. Basic Ballot Casting Procedure

A voter enters the voting booth after successful voter-identification. She then casts her ballot using the following procedures.

i) For each candidate \( i (1 \leq i \leq n) \), direct recording electronic (DRE) computes
\[
e_i^L = E(S_i^L, r_i^L)
\]
and
\[
e_i^R = E(S_i^R, r_i^R),
\]
where \( r_i^L \) and \( r_i^R \) are random numbers, and displays \( n \) candidates’ names.

ii) The voter chooses her preferred candidate \( j (1 \leq j \leq n) \) by touching the DRE screen.

iii) DRE displays \( e_j^L \) and \( e_j^R \) alongside the selected candidate \( j \).

iv) The voter randomly selects \( e_j^L \) or \( e_j^R \). Let \( e_j^\prime = e_j^L \) and \( e_j^\prime = e_j^R \) if she selects \( e_j^L \). Otherwise, \( e_j^\prime = e_j^R \) and \( e_j^\prime = e_j^L \).

v) DRE records \( e_j \) as her encrypted vote and prints \( e_j \) on a receipt for her. Additionally, DRE prints \( n \) encrypted symbols \( e_i^R \) along with the corresponding random numbers \( r_i^R \) on the receipt according to the candidate order if \( e_j = e_j^L \). Otherwise, DRE prints \( n \) \( e_j^L \) and \( r_i^L \) on the receipt.

vi) The voter should check that \( e_j \) and \( e_j^\prime \) are printed correctly and \( e_j^\prime \) is printed at the \( j \)-th row.

vii) After voter verification, DRE prints its digital signature \( \sigma \) on the receipt to ensure receipt validity.

viii) The voter verifies the digital signature with the help of the election officials or scrutineers. If it is valid, she keeps the receipt and can check the correctness of the encrypted vote later.

ix) Finally, DRE posts \( e_j \) along with her receipt \( id \) on the official election web site.

Note that the voter can start the ballot casting procedure over at any step until the DRE prints its digital signature on her receipt. Figure 5 shows an example of a receipt [20].

2. Receipt Verification

After having left the polling place, a voter can check, using software she trusts – even software she may have written herself – that her intended vote has been correctly posted and tallied. She may also give her receipt to third parties she trusts, for example, civic groups, parties, or NGOs, and the designated party will verify the receipt on her behalf.

Receipt verification is described as follows.

i) For \( 1 \leq i \leq n \), check if \( e_i^R = E(S_i^R, r_i^R) \) holds where \( S_i^R \) is publicly known.

ii) Check if \((id, e_j)\) are posted on the official election web site.

The voter can be assured that \( j \) is an encryption of her intended vote with a probability 50% and will be counted correctly if the above verifications are successful.

3. Naked-Eye Comparison

In general, the length of a ciphertext exceeds 1,024 bits that requires more than 170 alphanumeric characters to be represented. Obviously, no voter can compare such long strings in front of the DRE and thus, a string to be compared should be
short, 2 to 4 characters. We incorporate a special function, that is, reduction function $R : \{0, 1\}^* \rightarrow \{0, 1\}^8$, to convert ciphertext to the corresponding code to tackle this problem (Fig. 6). More specifically, the function $R$ is defined by $R(x) = W(h(x))$, where $W$ refers to the 10-bit word exclusive OR function and $h$ denotes a secure hash function, for example, SHA-1.

A 10-bit string can be encoded as two characters based on Table 1.

One serious problem of $R$ is that $R$ is not collision resistant; hence, DRE can easily find $r'$ satisfying $E(S', r) = E(S', r')$ where $S \neq S'$. For example, suppose that the voter selects the third candidate and the DRE displays two corresponding codes $c_3^L = R(E(S_3^L, r_3^L))$ and $c_3^R = R(E(S_3^R, r_3^R))$ to cheat the voter. If the voter chooses the left code, then DRE reveals $r'$ satisfying $c_3^L = R(E(S_3^L, r'))$. Otherwise, DRE reveals $r'$ satisfying $c_3^R = R(E(S_3^R, r'))$.

To prevent this problem, the DRE should print all the ciphertexts on the receipt before the voter’s selection. This commitment can prevent the DRE from cheating the voter since it is computationally infeasible to find $w'$ or $w''$ satisfying $E(S_3^L, c_3^L) = E(S_3^L, r')$ or $E(S_3^L, c_3^R) = E(S_3^L, r')$. The ciphertexts in the commitment should not be printed in the order of the candidate list to prevent vote-buying or vote-selling. One easy way to permute ciphertexts is to sort all the ciphertexts [21]-[23].

IV. Improvement of Proposed Technique

Though the scheme of Lee and others provides both individual verifiability and universal verifiability, its security highly depends on the voters’ carefulness, especially naked-eye comparisons. However, it would be impractical to require voters to verify the well-formedness of each candidate’s codes at the very instant of voting.

In this section, we improve the scheme of Lee and others to make the system more practical and user-friendly. The improved scheme’s main advantage over the previous scheme is user-friendliness. For the previous scheme, a voter should keep an eye on the codes printed on the receipt and check whether the codes are the same as the displayed codes on the screen because DRE prints codes on it after the voter’s selection. However, for the improved scheme, this kind of problem does not exist because the receipt (ballot) is printed before the voter’s selection. Furthermore, a voter has to remember the only one preferred code before casting her vote and this would greatly increase the usefulness of the scheme.

1. Overall Election Process

The overall election process of the proposed scheme can be described as follows (Fig. 7).

**Step 1.** Voter identification. The electoral officer checks the voter’s eligibility. Various identification schemes can be used in this stage.

**Step 2.** Ballot creation. A ballot should be issued to the eligible voter.

**Step 3 (optional).** Ballot verification. The well-formed ballot can be checked by the independent auditor (scrutineer), and the verified ballot should be discarded. Note that this is optional because ballot verification can be achieved in step 6 implicitly.

**Step 4.** Casting vote. At this stage, the voter casts her vote using the DRE. After casting the vote, she tears off the ballot to create the receipt. Note that the receipt can be either the left-hand or right-hand side according to the voter’s random selection.

**Step 5.** Shredding. Before leaving, the voter hands the election officer the opposite side of the receipt. The officer inserts it into a paper shredder to destroy it.

---

### Table 1. Bits and their corresponding characters.

<table>
<thead>
<tr>
<th>Bits</th>
<th>Char</th>
<th>Bits</th>
<th>Char</th>
<th>Bits</th>
<th>Char</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000</td>
<td>A</td>
<td>01000</td>
<td>J</td>
<td>10000</td>
<td>R</td>
</tr>
<tr>
<td>00001</td>
<td>C</td>
<td>01001</td>
<td>K</td>
<td>10001</td>
<td>S</td>
</tr>
<tr>
<td>00010</td>
<td>D</td>
<td>01010</td>
<td>L</td>
<td>10010</td>
<td>T</td>
</tr>
<tr>
<td>00011</td>
<td>E</td>
<td>01011</td>
<td>M</td>
<td>10011</td>
<td>U</td>
</tr>
<tr>
<td>00100</td>
<td>F</td>
<td>01100</td>
<td>N</td>
<td>10100</td>
<td>V</td>
</tr>
<tr>
<td>00101</td>
<td>G</td>
<td>01101</td>
<td>O</td>
<td>10101</td>
<td>W</td>
</tr>
<tr>
<td>00110</td>
<td>H</td>
<td>01110</td>
<td>P</td>
<td>10110</td>
<td>X</td>
</tr>
<tr>
<td>00111</td>
<td>I</td>
<td>01111</td>
<td>Q</td>
<td>10111</td>
<td>Y</td>
</tr>
<tr>
<td>01000</td>
<td>J</td>
<td>01000</td>
<td>J</td>
<td>10000</td>
<td>R</td>
</tr>
</tbody>
</table>

*Note that ‘B’, ‘I’, and ‘Z’ are not used because they could be easily confused with ‘8’, ‘1’, and ‘Z’.*
2. Voter Experience

A. Definitions and Notations

In this paper, we will use the following definitions and notations.

- \( E(\cdot, \cdot), D(\cdot) \): ElGamal public key encryption/decryption functions
- \( \mathbf{s}^L_i, \mathbf{s}^R_i \): The \( i \)-th candidate’s public symbols for the left and right hands, respectively
- \( r^L_i, r^R_i \): The random numbers for the \( i \)-th candidate that are used to encrypt the \( i \)-th candidate’s symbols
- \( R(\cdot) \): Reduction function (Fig. 4)
- \( P(\cdot) \): Permutation and inverse permutation functions

A voter enters into the voting booth after successful voter identification. She then casts her ballot using the following procedures. We will describe the improved voting procedure with an example. Suppose there are four candidates, that is, \( n=4 \).

B. Ballot Composition

After successful voter identification, the ballot issuing system computes \( \mathbf{s}^L_i, \mathbf{s}^R_i \) for each candidate \( i=1, \ldots, n \). Note that each \( \mathbf{s}^L_i \) and \( \mathbf{s}^R_i \) should be unique. \( 2^n \) codes \( \mathbf{s}^L_i \) and \( \mathbf{s}^R_i \), and their corresponding random numbers \( r^L_i \) and \( r^R_i \) are encoded into two 2D barcodes \( L_1 \) and \( R_1 \) for the left-hand and the right-hand sides of the ballot, respectively.

The other two barcodes \( L_2 \) and \( R_2 \) are composed of the following information:

\[
\begin{align*}
L_2 & : \mathbf{P}(r^L_1, r^L_2, \ldots, r^L_n), \\
R_2 & : \mathbf{P}(c^R_1, c^R_2, \ldots, c^R_n).
\end{align*}
\]

Finally, after composing four barcodes, the ballot issuing system calculates the digital signatures for each barcode and appends each signature to the corresponding barcode.

A voter is given her ballot by the ballot issuing system; the ballot looks like that displayed Fig. 8, where \( n=4 \). Note that candidate \( i \)'s left and right codes are computed by \( R(c^R_i) \) and \( R(c^L_i) \), respectively, for example, \( C9=R(c^R_9) \).

Suppose that the voter’s preferred candidate is the second, then she only has to select one code \( C9 \) or \( 7G \) at random, that is, the left or the right, and she does not have to consider the other codes and their position.

C. Checking If a Ballot is Well Formed

Obviously two barcodes pairs (\( L_1, R_2 \)) and (\( R_1, L_2 \)) should contain the same ciphertexts except the order of those. We define the well-formedness of a ballot as follows.

Definition. (Ballot well-formedness). A ballot is said to be well formed if the following two conditions are satisfied.

i) All ciphertexts of \( L_1 \) (\( R_2 \)) are contained in \( R_1 \) (\( L_2 \)), and the orders of ciphertexts of \( L_1 \) and \( R_2 \) differ.

ii) All ciphertexts of \( R_1 \) (\( L_2 \)) are contained in \( L_1 \) (\( R_2 \)), and the orders of ciphertexts of \( R_1 \) and \( L_2 \) differ.

Verifying the well-formedness of the ballots can be achieved by simple software using the barcode reader. However, it should be conducted by non-government organizations, such as civic groups or the opposition parties, to scrutinize the election.

Note that since no decryptions are required to verify how well-formed a ballot is, the verified ballot can be used to cast a vote that is different from other e-voting systems \([11],[17] \).

D. Casting Vote

A voter casts her vote at the front of the DRE using the ballot. The ballot casting procedure is described as follows.
Fig. 10. Final ballot form reflects voter’s intent.

**i)** The voter inserts the ballot into the DRE.

**ii)** DRE scans the barcodes and checks the digital signatures of the ballot issuing system.

**iii)** After verification of the digital signatures, DRE displays the ballot on the touch screen which looks like Fig. 9.

**iv)** The voter selects one code of her preferred candidate randomly, that is, the left or the right. She only has to remember the selected code. For example, the voter remembers ‘C9’ if her preferred candidate is the second, and she selects the left code.

**v)** DRE prints the selected code and its corresponding ciphertext on the opposite side of the voter’s selection, and appends its digital signature. Finally, DRE marks the selected hand of the ballot to be shredded by printing the predetermined word, such as ‘void’ or ‘discard.’ Suppose the voter has selected ‘C9,’ then the final ballot will look like Fig. 10. Note that barcode F contains the corresponding ciphertext $C_j(=C_j^L)$ and digital signature.

**vi)** After the voter’s confirmation of her selection, DRE posts the selected ciphertext accompanied by the ballot id to the public Web bulletin board.

**E. Shredding**

The voter checks the ballot to see if the printed code is the same as her choice on the opposite side of her selection. If it is incorrect, she notifies the electoral personnel that the receipt is incorrect and restarts the voting procedure. She then tears off the ballot to create a receipt and hands the electoral officer the selected side of the ballot, that is, the left side ballot in the case of the preceding example. The officer inserts it into a paper shredder to destroy it (Fig. 11).

Fig. 11. Receipt production. Opposite side (the left side in case of the above figure) should be shredded securely.

**F. Receipt Verification and Ballot Counting**

If the voter takes the right side of a ballot as her receipt, then the receipt verification can be done by checking the following conditions.

**i)** Each code on the receipt should be unique.

**ii)** For $i = 1, ..., n$, the $i$-th candidate’s code $\text{code}_i^R$ should be the same as $R(E(S_i^r, r_i^S))$ using $r_i^S$ in the barcode R1 and publicly known $S_i^r$.

**iii)** The ciphertext $C_j$ in the barcode F should be in the barcode R2 and $R(C_j) = \text{code}_j^L(=C9)$.

**iv)** The ciphertext $C_j$ should be published to the public Web bulletin board.

If the receipt verification is successful, the voter can be assured the $C_j$ is an encryption of her selection with a probability of 50%.

After all ballots have been published to the official election web site, the ballot tabulation stage begins. In ballot tabulation, the election authorities execute a publicly verifiable multistage mix-net, where each authority privately executes a particular stage of the mix-net [2]. To preserve anonymity, the authorities strip each ballot of its id before it enters the mix-net. Each stage of the mix-net takes as input a set of encrypted ballots, partially decrypts or re-encrypts them, and randomly permutes them. The final result of the mix-net is a set of plaintext ballots that can be publicly counted but cannot be linked to the encrypted ballots or the voter identity. In cryptographic voting protocols, the mix-net is designed to be universally verifiable: the authority provides a proof any observer can use to confirm that the protocol has been followed correctly. This means a corrupt authority cannot
surreptitiously add, delete, or alter ballots. Note that ballot counting is beyond the scope of this paper.

V. Security and Performance Analysis

1. Security Model

In general, an e-voting protocol should satisfy the following requirements.

**Receipt-freeness.** No one can obtain or is able to construct a receipt proving the content of a ballot. Voting schemes should provide receipt-freeness to prevent voters from providing their vote cast, and hence to thwart vote-buying and coercion.

**Verifiability.** There are two kinds of verifiabilities, individual verifiability and universal verifiability [7]. The former means that each voter should be able to satisfy him/herself that the recorded ballot has been captured correctly and the later means that anyone should be able to satisfy him/herself that the recorded ballots are counted correctly.

**Anonymity.** No one can link the decrypted votes to voters.

We will discuss only receipt-freeness and individual verifiability because anonymity and universal verifiability can be satisfied by incorporating mix-nets [2]. Note that the proposed scheme supports any kind of traditional mix-nets or onion mix-nets. Additionally, we will show that the improved scheme satisfies end-to-end (E2E) verifiability according to the Popoveniuc’s checks [24].

2. Individual Verifiability

Individual verifiability can be measured by fraud detection probability. Obviously, the probability of fraud detection should be sufficiently high, for example, at least 50%. Assume that all ballots are verified by the trusted auditor(s), then there are two kinds of attackers, the ballot issuing system and the DRE. In case of a malicious ballot issuing system, it could make the symbol $s_j^k$ or $s_j^s$ be recorded differently from a voter’s actual selection ($j \neq k$) by composing barcodes maliciously. For simplicity, let us assume that there are two candidates and a voter prefers the first candidate. Then, the malicious system could compose the ballot in two ways, the first of which is shown in Fig. 12.

However, such an attack can always be detected because the ballot is not well formed and is detected by the auditor. The other form of cheating is to compose the ballot based on the prediction of the voter’s selection (Fig. 13).

Although the ballot is well formed, if the voter selects the right code, and takes the left side as her receipt, the cheating will be detected later because she checks if the following two equations hold:

$$E(s_j^r, r^r) = E(s_j^k, r^k),$$

$$E(s_j^r, r^r) = E(s_j^k, r^k).$$

Thus, the probability of cheating going undetected is 1/2. DRE could also launch an attack, since it knows the voter’s selection. However, it cannot modify the barcodes, other than barcode F (Fig. 10). The only form of cheating to be launched by the DRE using a well-formed and valid ballot, is to find a random number $r$ satisfying $E(s_j^r, r^r) = E(s_j^k, r^k)$. However, finding such $r$ is computationally infeasible. Thus, we can say that the improved e-voting system’s fraud detection probability is at least 50%.

3. Receipt-Freeness

We can say that the improved e-voting system satisfies the receipt-freeness if [19] satisfies the receipt-freeness and vice-versa because a receipt is composed of the same information. Thus, the following theorem can be derived directly from [19].

**Theorem.** If there exists a secure probabilistic encryption scheme in the sense of indistinguishability and a cryptographically secure pseudo-random number generator (CSPRNG), and a voting machine generates random numbers
Table 2. Comparison results.

<table>
<thead>
<tr>
<th>Category</th>
<th>Prét à Voter</th>
<th>PunchScan</th>
<th>Proposed Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security ( p ) (fraud detection)</td>
<td>( p = 1 - (b - t) / b )</td>
<td>( p = 1 - (b - t) / b )</td>
<td>( p = 1 / 2 )</td>
</tr>
<tr>
<td>Number of ballots ( b )</td>
<td>( b \approx 2t )</td>
<td>( b \approx 2t )</td>
<td>( b = t ) **</td>
</tr>
<tr>
<td>Ballot preparation</td>
<td>Required*</td>
<td>Required*</td>
<td>Not required</td>
</tr>
<tr>
<td>Candidate order</td>
<td>Not fixed</td>
<td>Fixed</td>
<td>Fixed</td>
</tr>
<tr>
<td>Trust assumption</td>
<td>Yes (teller)</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Receipt-freeness</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Write-ins support</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

\( t \): the number of electorate

**: All ballots should be prepared in the election setup stage.

using CSPRNG, then no one can get any partial information about the voter’s choice from a receipt.

**Proof.** The ElGamal encryption scheme [25] is secure in the sense of indistinguishability. Therefore, the ElGamal encryption scheme is semantically secure and is secure under a chosen plaintext attack. Let \( A \) be a set of encrypted values printed on a receipt and \( B \) a set of encrypted values computed by the voter herself. Obviously, no one can distinguish \( A \) from \( B \) if the voting machine generates random numbers using CSPRNG. Moreover, several secure CSPRNGs exist [26]. Thus, an attack trying to obtain information from a receipt can be thought to be a form of chosen plaintext attack. An attacker cannot learn any partial information about the voter’s choice because the ElGamal encryption scheme is secure under a chosen plaintext attack [19].

The theorem implies that a voter can prove her selection to others only if she can predict the random numbers used by the DRE.

4. E2E Verifiability

Popoveniuc and others proposed a definition for E2E verifiability of public elections, and suggested a set of properties that collectively define them [24]. In this subsection, we briefly analyze the improved scheme’s conformance to the Popoveniuc’s requirements. The scheme satisfies two requirements, “recorded as cast” and “consistency,” because it gives voters receipts which are posted on a public bulletin board [24].

- Cast ballots are well formed. Since the scheme cast ballot cannot contain any negative votes, and over-votes can be detected by anyone who inspects the receipts published on the public bulletin board, “cast ballots are formed” is satisfied.
- Tallied as recorded. The scheme uses a publicly verifiable mix-network to ensure that all the recorded ballots have been tallied. Therefore “tallied as recorded” is satisfied.
- Each recorded ballot is subject to the “recorded as cast” check. Suppose two voters are given the same ballot. It may happen that they choose different candidates; thus, anyone can detect that the same receipt is posted twice on the public bulletin board with different selections. Thus, “each recorded ballot is subject to the recorded as cast check” is satisfied.

In conclusion, the improved scheme is E2E verifiable.

5. Another Attack

In [27], Kelsey and others considered the security of the various E2E voting systems against attacks to tamper with election results and to permit the buying or coercion of votes. Among these attacks, one of the most powerful attacks for vote buying or coercion is forged ballots. An attacker provides voters forged ballot halves to destroy instead of actual ballot halves. This threat can be mitigated by letting the DRE shred the unselected ballot halves automatically after voter confirmation or checking serial numbers before shredding by election officers.

6. Comparison Results

With Prêt à Voter, a voter can be assured that the vote cast reflects her intent with probability \( (b - t) / b \), where \( b \) is the total number of ballots and \( t \) is the number of electorate. To
provide at least 50% security, there should be at least two times more ballots than required for the electorate.

Though PunchScan and Scantegrity offer greater familiarity of a widely used vote-counting system, they also require a large number of ballots to be prepared and audited. Neither of the systems supports write-ins voting. One disadvantage of the proposed system is that the voting machine can know the voter’s choice. Table 2 summarizes the comparison results.

VI. Conclusion

A voter verifiable audit trail is the most effective way to make electronic elections trustworthy. However, if we use receipts for physical audit trails only for recounts, as in Mercuri’s system, we lose the main advantages of e-voting, such as faster tallying and greater accuracy. A receipt should be used to ensure individual verifiability, not for recounting.

In this paper, we proposed a practical and secure e-voting scheme preserving the secrecy of votes cast and providing receipt-freeness and voter verifiability. The proposed scheme provides two major advantages. The first advantage is that voters do not have to put trust in any devices in the polling station. Furthermore, the strings to be compared are composed of only two alphanumeric characters. The second advantage is that anyone can be assured that the encrypted vote cast reflects the voter’s intent with probability of at least 50% via a simple and public method.

References


Yunho Lee received his BS, MS, and PhD from Sungkyunkwan University, Rep. of Korea, in 1991, 1993, and 2008, respectively. After working at Korea Telecom as a member of the technical staff from 1993 to 2000, he worked for KBS Internet as the director of the technical support team for 5 years. He is currently a professor at the Department of Cyber Security & Police, Gwangju University, Rep. of Korea. His interests include cryptology and information security. He is particularly interested in electronic voting, digital watermarking and fingerprinting, and key exchange protocol.

Dongho Won received his BE, ME, and PhD from Sungkyunkwan University, Rep. of Korea, in 1976, 1978, and 1988, respectively. After working at the ETRI from 1978 to 1980, he joined Sungkyunkwan University in 1982, where he is currently a professor of the School of Information and Communication Engineering. His interests include cryptology and information security. He was the president of the Korea Institute of Information Security & Cryptology (KIISC) in 2002.