Towards Trustworthy e-Voting using Paper Receipts*

Yunho Lee, Kwangwoo Lee, Seungjoo Kim, and Dongho Won**

Information Security Group,
Sungkyunkwan University,
300 Cheoncheon-dong, Suwon-si, Gyeonggi-do, 440-746, Korea
{leeyh,kwlee,skim,dhwon}@security.re.kr
http://www.security.re.kr

Abstract. Current electronic voting systems are not sufficient to satisfy trustworthy elections as they do not provide any proofs or confirming evidences of their honesty. This lack of trustworthiness is the main reason why e-voting is not widely spread even though e-voting is expected to be more efficient than the current plain paper voting. Many experts believe that the only way to assure voters that their intended votes are casted is to use paper receipts. In this paper, we propose an efficient scheme for issuing receipts to voters in e-voting using the well-known divide-and-choose method. Our scheme does not require any special printers or scanners, nor frequent observations to voting machines. In addition to that, our scheme is more secure than the previous ones.

Key words: e-voting, voter verifiable e-voting, paper receipts

1 Introduction

Current electronic voting systems require voters to trust them. Voters should believe that the voting machines do not cheat even though they do not provide any proofs or confirming evidences of their honesty. This so called *Black-Box Voting* is the greatest obstacle to conduct electronic voting.

In [8], R.Mercuri stated that fully electronic systems do not provide any way that the voter can truly verify that the ballot cast corresponds to that being recorded, transmitted, or tabulated as many other experts did. Moreover, any programmer can write code that displays one thing on a screen, records something else, and prints yet another result. There is no known way to ensure that this is not happening inside of a voting system. The most effective way to decrease the trust voters must place in voting machine software is to let voters physically verify that their intent is recorded correctly, for e.g., voter verifiable paper trails.

* This work was supported by the University IT Research Center Project funded by the Korea Ministry of Information and Communication.

** Corresponding author : Dongho Won(dhwon@security.re.kr)

In 2002, R.Mercuri proposed a method for voter verifiable ballots [11]. In her method, after a voter has finished making selections using a voting machine, the machine prints out a paper ballot that contains the voter's selections for each choice. The printed ballot is kept behind a window to prevent voters from having any opportunity to tamper with it. If voters examine and approve the ballot, the voting machine drops the printed ballot into an opaque ballot box. While voter verifiable paper ballots eliminate the need to trust the voting machine, the need to support printing and collecting of paper ballots increases the maintenance costs and election complexity for the poll workers [15].

In 2002 and 2004, D.Chaum proposed a method to provide voters with a coded receipt that reflects their vote but does not reveal it to anyone else [10, 14]. In his scheme, the voting machine prints the coded receipt on the two separable layers using visual cryptography [5]. When laminated together, they reveal the voter's choices, but each separated layer is meaningless dots. The voter verifies the laminated receipt and then selects one of the two layers to retain as receipt. The other layer should be surrendered to a poll worker and shredded. The voting machine could cheat if it knows which layer will be selected by the voter in advance, thus the chance is $\frac{1}{2}$. The costs to implement D.Chaum's scheme is relatively high as it is required that all voting machines to be equipped with special printers.

A.C.Neff and J.Alder proposed another method to provide voters with a coded receipt [12]. In their scheme the receipt is printed with a codes, i.e., encrypted values, for each selection according to the codebook which is generated by election authority prior to the election.

The printed codes are meaningless to anyone other than the voter, who sees the codes displayed by the voting machine. The validity of the displayed codes can be assured by observers at randomly selected times throughout the election who act as voters and audit the displayed codes. Obviously, observers should have the same codebook and keep its secrecy. Thus, the chance that the voting machine could cheat is $\frac{c}{l+c}$ where c and l are the number of observations and the number of voters respectively.

An e-voting scheme should be verifiable by voters whether casting and counting ballots are performed correctly or not. A verifiable e-voting involves the following two distinct checks: [12]

- (**Check 1**) A voter should be able to satisfy him/herself that the voted ballot is captured correctly (*cast-as-intended* or *individual verifiability*); and
- (**Check 2**) Anyone should be able to satisfy him/herself that the voted ballot is counted correctly (*counted-as-casted* or *universal verifiability*).

The second check preserving anonymity can be satisfied by using various methods for e.g., a provably secure mix-networks. However, in case of the first check, it is not easy to verify it without voter verifiable receipts because no one can trust voting machines. While a voter verifiable receipts are used for the first check, nobody can prove his or her vote to anyone even to him or herself in order to prevent *vote-buying* and *selling*.

For the first check of the above mentioned, we argue that an e-voting scheme should satisfy the following two requirements.

- (**Requirement 1**) No devices or workers in polling places should be trusted.

- (**Requirement 2**) No decryptions should be performed during voting period.

There is no need to explain why the requirement 1 is necessary. The decryption in requirement 2 means that any operations which can be misused to prove a voter's casted vote. Furthermore, for the sake of voter's confidence, the validity of receipt can be assured not only by prefixed devices or softwares but also by any devices or softwares made by herself at anytime.

In 2005, D.Chaum *et el.* proposed a practical voter-verifiable election scheme, Prêt *à* Voter[16]. Their scheme provides a number of advantages, for e.g., the voting machine can not learn the voters' choices and the vote casting process is similar to that of the current paper voting. However, their scheme does not meet the above two requirements.

Our Contribution In this paper, we propose an efficient method for issuing receipt to a voter for voter verifiable e-voting using the well-known *divide-and-choose* method. Our scheme does not require voting machines to be equipped with special printers, observations to voting machines, and secret codebooks different from the D.Chaum's scheme or A.C.Neff's scheme. Moreover, our scheme is more secure than the previous ones and voters do not need to trust any devices or election personnel in voting booths.

The advantages of the proposed scheme can be listed as follows.

1. **Security** The fraud detection probability of a voting machine is more than $\frac{1}{2}$.
2. **Reliability** A voter does not have to trust somebody in order to verify her/his receipt. Anyone can make software for verifying receipts.
3. **Efficiency** Our scheme does not need election observers to audit voting machines in polling places.
4. **Implementation** Our scheme is easy to implement because it does not require any special printers or preprocessing.

The rest of this paper organized as follows. Section 2 outlines cryptographic primitives for our schemes. Section 3 describes the D.Chaum's election scheme, Prêt *à* Voter. Section 4 presents an efficient receipt issuing scheme and we analyze security of the schemes in section 5. Section 6 presents future directions and finally, this paper is concluded in Section 7.

2 Previous Work

2.1 Y.Lee's Scheme

In 2005, Y.Lee *et al.* proposed

D.Chaum, P.Y.A.Ryan, and S.Schneider proposed a practical voter-verifiable election scheme, Prêt *à* Voter[16]. Their scheme uses a more conventional representation of the vote, i.e., ballot forms with the candidates listed in left column, and the voter choices marked in an adjacent right column.

The 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 the information from which the candidate ordering can be reconstructed, encrypted by the public key(s) of the predetermined teller(s)(See Fig.1).

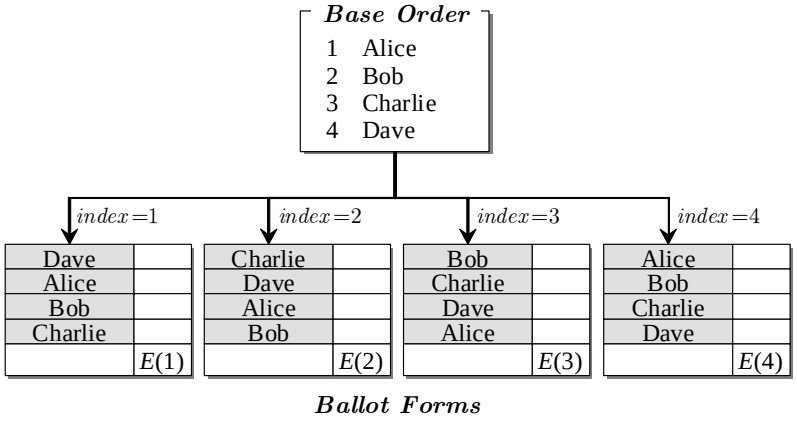

Fig. 1. Ballot forms

Casting The Vote After registering and authenticating a voter, she is asked to select a ballot form. She marks her '$\surd$ ' 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 look like Fig.2-(a).

She now removes the left hand strip(for shredding), and feed the right hand strip into the voting machine(a kind of scanner)(See Fig.2-(b) and (c)). She retains the right hand strip as a receipt and can check whether her vote is recorded correctly via public bulletin board.

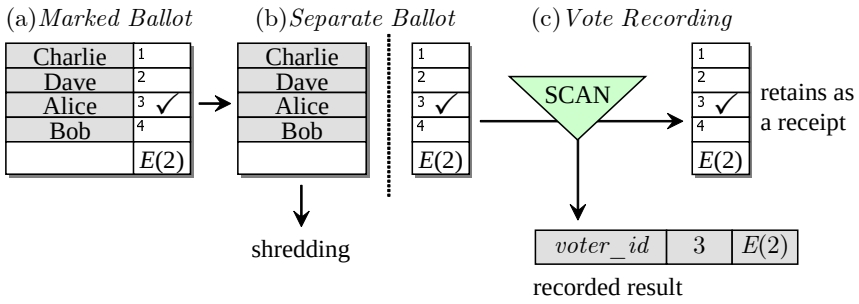

Fig. 2. Casting the vote

Tallying For tallying, the teller(s) should reconstruct the candidate ordering for each casted votes. The reconstruction of the candidate ordering can be done by decrypting the information. For example, suppose that the casted vote is $(3, E(2))$, then the voter's choice v can be calculated by the following equation:

$$v = 3 - D(E(2)) (\text{mod } 4) = 3 - 2 (\text{mod } 4) = 1 (\text{Alice}).$$

Auditing the Process Obviously one of the most concerned problem is the discrepancy between the candidate order printed on a form and the buried information. [16] stated that the most satisfactory method is that a voter gives the information printed on a ballot form, the teller(s) return the candidate ordering. For example, a voter selects b ballot forms at random and nominates $b-1$ for checking. If the $b-1$ checks go through okay, the voter can trust the remaining ballot form is well-formed with probability $\frac{b-1}{b}$.

Two serious drawbacks of this auditing method are;

1. the teller(s) should decrypt the transmitted information, and
2. the voter should trust an auditor in the polling station.

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, it is less acceptable that a voter should trust an auditor in order to trust her ballot form's integrity. In polling station, a voter should never trust any devices or election workers other than herself.

3 Proposed Scheme

A receipt issued to a voter which can be took out of the polling place increases voter's confidence that the ballot was *cast-as-intended* and *counted-as-casted* as she can verify the whole processes of election at any time using her receipt.

In this paper, we propose a receipt issuing scheme which does not require voters to trust specific device(s) in the polling station and decryption operations during the voting period. The whole e-voting procedure is depicted in Fig. 3.

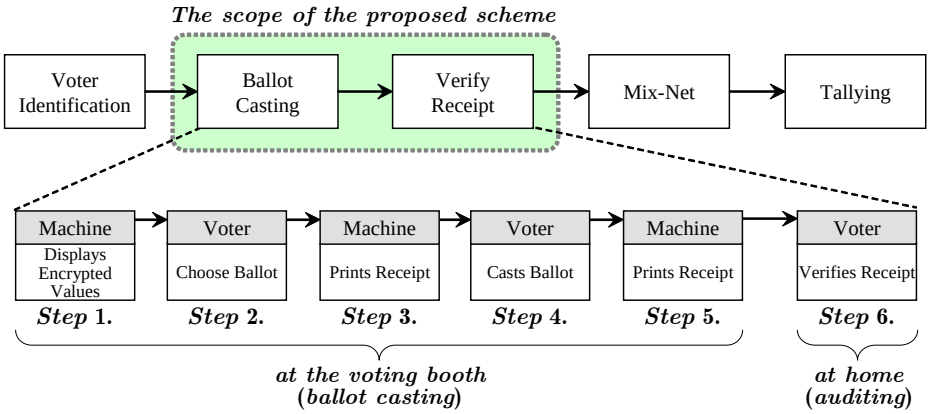

Fig. 3. Overall procedure for e-voting and the scope of the proposed scheme

A voter can be assured that her vote is *cast-as-intended* by her own computations of ElGamal encryption or by using a public verification server at anytime. Also a voter can verify that her vote is *counted-as-casted* by comparing her encrypted choice with the result registered to the public web bulletin board and verifying ballot shuffling by mix-net.

3.1 Notations

- n : The number of candidates
- $E(\cdot,\cdot)$, $D(\cdot,\cdot)$: ElGamal encryption and decryption
- v : voter's choice$(1 \leq v \leq n)$
- w : random number for ElGamal encryption
- e : encrypted choice $(e = E(v, w))$

3.2 Ballot Casting

The detailed procedure of the proposed scheme can be described as follows.

1. Voting machine displays n encrypted pairs $(e'_j, e''_j) = (E(j, w'_j), E(j, w''_j))$ where w'_j and w''_j are random numbers for $j(j = 1, ..., n)$.
2. For $j = 1, ..., n$, a voter randomly selects e'_j or e''_j for each j.
3. Voting machine prints n unselected values and their corresponding random numbers on the paper receipt as proofs. Voter should check that the n printed values are the same as on the screen.
4. Voter casts her ballot by choosing $v(v \in \{1, ..., n\})$.
5. Voting machine prints the v^{th} encrypted value from the selected values in step 2. Voter should check that the printed value is the same as on the screen.

Please note that candidates' codes $j(1 \leq j \leq n)$ can be replaced by unique numbers that is generated by way of a publicly defined process. For the proposed scheme, the chance that the voting machine's cheating would go undetected is $\frac{1}{2^{n-1}}$. Fig. 4. depicts an example in case of $n = 4$.

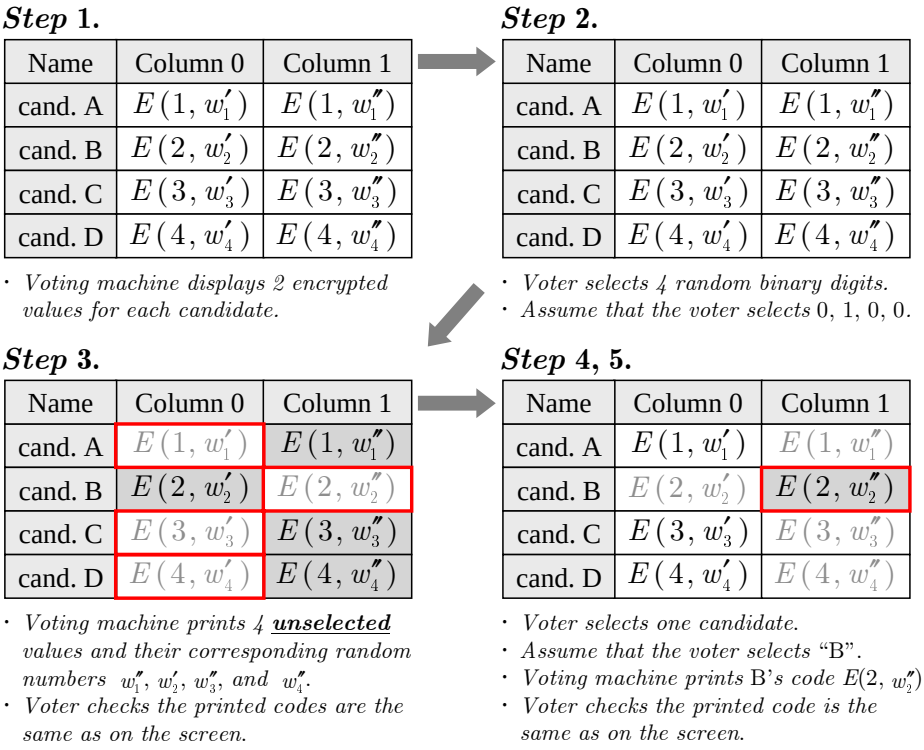

Fig. 4. Ballot casting procedures in case of 4 candidates.

3.3 Receipt Verification

After finishing voting, the voter retains her receipt which is look like the left side of Fig. 5.

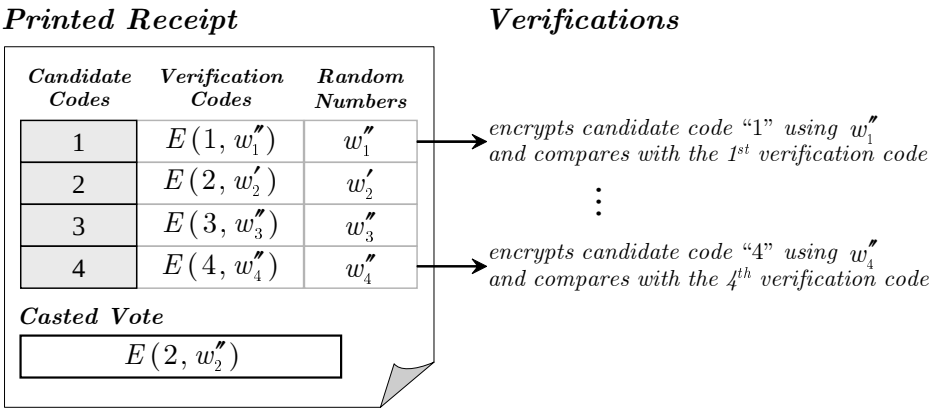

Fig. 5. A sample receipt with 4 candidates and verification procedure.

Here is a brief description of verification of a receipt. A voter encrypts each candidate code j with the corresponding random number printed on the receipt using ElGamal encryption and verifies the encrypted results are equal to the corresponding verification codes. This verification can be done easily by any voters because it requires n simple encryptions and there are numerous open source codes for ElGamal encryption. Thus voters do not have to trust any specific devices or election workers in order to verify their receipts.

For $j = 1, ..., n$, if all the encrypted results are equal to the corresponding verification codes, voter can be assured that her vote was casted as intended with the probability of $1 - \frac{1}{2^{n-1}}$. In Fig. 5, a voter should check whether the following 4 equations hold.

$$E(1, w_1'') = \text{the } 1^{\text{st}} \text{ verification code} \quad (1)$$
$$E(2, w_2') = \text{the } 2^{\text{nd}} \text{ verification code} \quad (2)$$
$$E(3, w_3'') = \text{the } 3^{\text{rd}} \text{ verification code} \quad (3)$$
$$E(4, w_4'') = \text{the } 4^{\text{th}} \text{ verification code} \quad (4)$$

3.4 Mixing and Tallying

Every casted votes are posted to the public web bulletin board. After closing election, the votes are shuffled through the mix-nets to preserve anonymity and then decrypted. There are numerous proposals for these mixing and tallying and these are beyond the scope of this paper.

4 Security Analysis and Comparisons

The security of a method for voter verifiable receipt can be measured by the chance of detecting a voting machine's fraud and the receipt-freeness of a receipt.

Receipt-freeness Receipt-freeness means that no one can obtains or is able to construct a receipt proving the content of a ballot.

Theorem 1. *If the ElGamal encryption scheme is not secure in the sense of indistinguishability, then there exists a probabilistic polynomial time TM that solves the DDH(Decision Diffie-Hellman) problem with overwhelming probability.*

Proof. Please refer to [6]. □

Theorem 1 implies that the ElGamal encryption scheme is secure because it is highly believed that the DDH problem is intractable. The Lemma 1 follows immediately from the theorem 1.

Lemma 1. *If there exists a secure CSPRNG(Cryptographically Secure Pseudo-Random Number Generator) and a voting machine generates random numbers using CSPRNG, then no one can get any partial information about the voter's choice from a receipt.*

Proof. An encryption scheme secure in the sense of indistinguishability is semantically secure[2]. Thus ElGamal encryption scheme is semantically secure and is secure under chosen plaintext attack. Let A be a set of encrypted values printed on a receipt and B be a set of encrypted values computed by a voter herself. Obviously no one can distinguish A from B if the voting machine generates random numbers using CSPRNG. Moreover, there are several secure CSPRNGs for e.g., [4]. Thus an attack trying to obtain any information from a receipt can be thought as a kind of chosen plaintext attack. However, an attacker can not learn any partial information about the voter's choice because the ElGamal encryption scheme is secure under chosen plaintext attack. □

Fraud Detection If a voting machine can predict a voter's random selection in step 2, it can alter her choice easily without being detected. Assume that there are 4 candidates and a voting machine predicts the voter's random selection as "0100"(column 0, column 1, column 0, and column 0), then it would simply displays 8 codes as Fig. 6-(b) other than Fig. 6-(a).

As you can see Fig. 6-(b), if the prediction is correct then the first candidate will be recorded despite the voter's choice. Thus the fraud detection probability is $1 - \frac{1}{2^{(n-1)}}$.

(a) No Cheating

Name	Column 0	Column 1
cand. A	$E(1, w_1')$	$E(1, w_1'')$
cand. B	$E(2, w_2')$	$E(2, w_2'')$
cand. C	$E(3, w_3')$	$E(3, w_3'')$
cand. D	$E(4, w_4')$	$E(4, w_4'')$

(b) Cheating

Name	Column 0	Column 1
cand. A	$E(1, w_1')$	$E(1, w_1'')$
cand. B	$E(2, w_2')$	$E(1, w_2'')$
cand. C	$E(1, w_3')$	$E(3, w_3'')$
cand. D	$E(1, w_4')$	$E(4, w_4'')$

Fig. 6. Cheating machine's screen.

Name	Column 0	Column 1	Column 2
cand. A	$E(1, w_1')$	$E(1, w_1'')$	$E(1, w_1''')$
cand. B	$E(2, w_2')$	$E(2, w_2'')$	$E(2, w_2''')$

Fig. 7. Strengthening security in case of $n = 2$.

Strengthening Security Suppose that $n = 2$ which is the minimum number of candidates, then the probability is only $\frac{1}{2}$ which is the same as D.Chaum's scheme proposed in 2002 and is not so high enough. In this case, we can make it more secure by incorporating additional columns. If we use 3 columns the probability will be increased to $\frac{2}{3}$(See Fig. 7).

Table 1 illustrates a comparison of fraud detection probabilities and efficiencies of the proposed scheme, A.C.Neff's scheme, and two D.Chaum's schemes.

Table 1. Comparison of fraud detection probabilities and efficiencies

	Proposed Scheme	A.Neff's Scheme(2003)	D.Chaum's Scheme(2002)	Chaum's Scheme(2005)
Fraud detection	$1 - \frac{1}{2^{n-1}}$	$1 - \frac{c}{l+c}$	$\frac{1}{2}$	$\frac{b-1}{b}$
Preprocessing	none	codebook	none	encrypted ballot
Special devices	none	none	special printers	none
Human familiarity	normal	normal	high	high
Trust assumption	none	observers	none	auditors

n : The number of candidates(≥ 2).
l : The number of voters.
c : The number of observations.
b : The number of unused ballots from which voter choose her ballot.

5 Future Directions

Though the proposed scheme provides numerous advantages, it also has two drawbacks. The one is that it is not easy to compare verification codes printed on a receipt with the ones displayed on the screen. And the other is that the scheme requires voters to select numerous random selections. In this chapter, we briefly explain how to tackle these drawbacks. These are the subject of current research.

5.1 Making the Receipt Easy to Compare

The typical length of a verification code is 1,024 to 2,048 bits. If we use a cryptographically secure hash function such as SHA-1, we can reduce the length of a code to 160 bits. However, it is still not easy to compare 28-character-long codes within seconds(See Fig. 8).

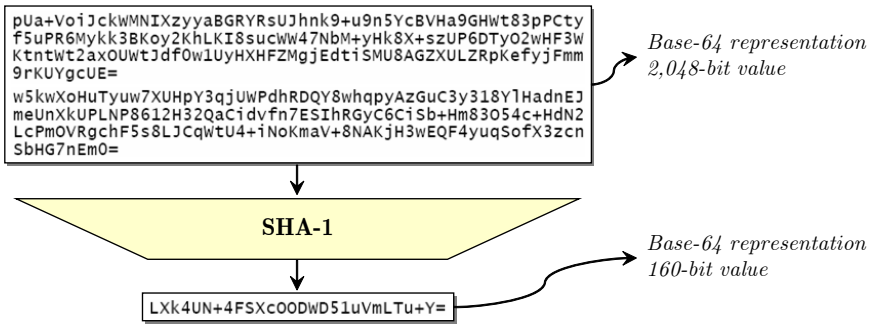

Fig. 8. Making a verification code short for easy compare using SHA-1

A reasonable length of a verification code is 8 to 10 bits which can be represented by up to two characters. However, for security reason, it is well known that a hashed code should be at least 160-bit long. Therefore, we have to find a way to generate much shorter codes without sacrificing its security.

5.2 Minimizing Random Selections

The proposed scheme requires a voter to n random selections of verification codes. Psychologically, it is well accepted that humans are notoriously bad at random selections. For the proposed scheme, n random binary selections can be translated into one random selection of numbers ranging from 2^0 to 2^n. For example, suppose that there are 4 candidates, then a voter selects only one number from 1 to 16(See Fig. 9).

Though the revised method is useful, it can not be used if there are lots of candidates. For example, if there are 10 candidates, it is impractical to display

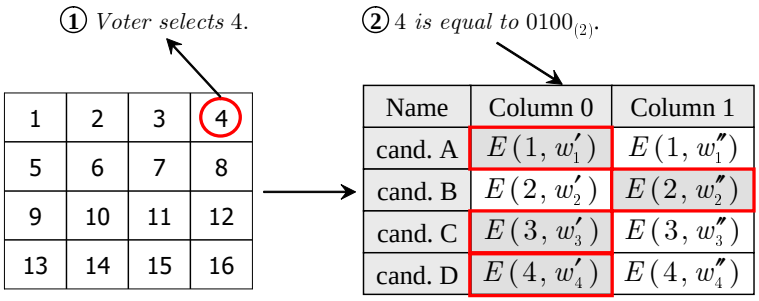

Fig. 9. Revised random selection method(16 means 0)

$2^{10} = 1,024$ numbers within a screen. Thus we have to make a more user-friendly and efficient selection method.

6 Concluding Remarks

As many experts said, voter verifiable audit trail is the most effective way to make trustworthy electronic elections. In this paper, we presented an efficient scheme for issuing cryptographic receipts in e-voting. Our scheme does not require any special printers or scanners nor frequent observations to voting machines. In addition to that, our scheme is more secure than the previous ones.

Though our scheme is more secure than the previous ones, there are some issues in our scheme. For e.g., psychologically, it is well accepted that humans are notoriously bad at random selections. Thus, random selections should be eliminated as possible. As compared with D.Chaum's scheme, our scheme requires a number of times more random selections. However this can be lessened if we modify the selection method. Another issue is that the comparison of verification codes displayed on the screen and printed on the receipt is difficult as the codes are too long. Even though if we hash the verification codes using SHA-1, the code length is about 160 bits. These issues are the subject of current research.

References

1. D.Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms," *Comm. of the ACM*, vol. 24, no. 2, pages 84–88, Feb. 1981.
2. S.Goldwasser and S.Micali, "Probabilistic Encryption," *Journal of Computer System Sciences(JCSS)*, vol. 28, no. 2, pages 270–299, Apr. 1984.
3. T.ElGamal, "A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Trans. on Information Theory*, vol. IT-31, no. 4, pages 469–472, 1985.
4. L.Blum, M.Blum, and M.Shub, "A Simple Secure Unpredictable Pseudo-ramdom Number Generator," *SIAM Journal on Computing*, Vol. 15, pages 364–383, 1986.

5. M.Naor and A.Shamir, "Visual Cryptography," *Proc. of Advances in Cryptology(Eurocrypt'94)*, LNCS 950, pages 1–12, 1995.

6. Y..Tsiounis and M.Yung, "On the Security of ElGamal Based Encryption," *Public Key Cryptography 98(PKC'98)*, LNCS 1431, pages 117–134, 1998.

7. P.Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes," *Proc. of Advances in Cryptology(Eurocrypt'99)*, LNCS 1592, pages 223–238, 1999.

8. R.Mercuri, "Rebecca Mercuri's Statement on Electronic Voting," `http://www.notablesoftware.com/RMstatement.html`, 2001.

9. A.C.Neff, "A Verifiable Secret Shuffle and Its Application to E-Voting," *Proc. of the 8th ACM Conference on Computers and Communications Security(CCS-8)*, pages 116–125, 2001.

10. D.Chaum, "Secret-Ballot Receipt and Transparent Integrity," *working draft*, May. 2002.

11. R.Mercuri, "A Better Ballot Box?," *IEEE Spectrum Online*, pages 46–50, Oct. 2002.

12. A.C.Neff and J.Adler, "Verifiable e-Voting: Indisputable Electronic Elections at Polling Places," `http://www.votehere.net/vhti/documentation/VH_VHTi_WhitePaper.pdf`, VoteHere Inc., 2003.

13. P.Golle, M.Jakobsson, A.Juels, and P.Syverson, "Universal Re-encryption for Mixnets," *CT-RSA 2004*, LNCS 2964, pages 163–178, 2004.

14. D.Chaum, "Secret-Ballot Receipts: True Voter-Verifiable Elections," *IEEE Security and Privacy Magazine*, vol. 2, no. 1, pages 38–47, Jan. 2004.

15. D.Evans and N.paul, "Election Security: Perception and Reality," *IEEE Security and Privacy Magazine*, vol. 2, no. 1, pages 24–31, Jan. 2004.

16. D.Chaum, P.Y.A.Ryan, and S.Schneider, "A Practical Voter-Verifiable Election Scheme," *Proc. of 10th European Symposium on Research in Computer Security(ESORICS2005)*, LNCS 3679, pages 118–139, 2005.

17. Y.Lee, K.Lee, S.Kim, and D.Won, "Efficient Voter Verifiable E-Voting Schemes with Cryptographic Receipts," *Proc. of IAVoSS Workshop On Trustworthy Elections(WOTE2006)*, pages ??–??, Cambridge, United Kingdom, 2006.