

A Secure and Efficient Voter-Controlled Anonymous Election Scheme

Thomas E. Carroll
Dept. of Computer Science
Wayne State University
5143 Cass Avenue, Detroit, MI 48202.
tec@cs.wayne.edu

Daniel Grosu
Dept. of Computer Science
Wayne State University
5143 Cass Avenue, Detroit, MI 48202
dgrosu@cs.wayne.edu

Abstract

We propose an electronic voting (e-voting) scheme that combines user-centric mix networks with voter-verifiable receipts. Unlike traditional mixnet-based e-voting schemes, our scheme empowers voters; the voters themselves decide the degree of anonymity required. Voters requiring a greater degree of anonymity obtain it by performing several protocol iterations. The proposed scheme utilizes incoercible, voter-verifiable receipts. It is robust as no reasonable-sized coalition can interfere with the correct operation. Finally, it is efficient as the number of transmitted messages increases linearly with the number of voters.

1. Introduction

Elections are the cornerstone of democracies. Citizens cast their vote as an extension of their beliefs, desires, and hopes. When the outcome is questioned because of improprieties, malfunctions, etc. citizens lose faith in the system. A recent study [16] suggests that the outcomes for a large proportion of past elections were manipulated.

Recently, the world encountered the notorious Florida recounts of the 2000 US Presidential election. Florida used an antiquated punch card system which resulted in a large number of spoiled votes. The initial tally was close resulting, by law, in an automatic recount. Poll officials needed to inspect every card and attempt to make a determination of voter's intent. The resulting process was highly subjective. In many cases, the intent was determined by the degree to which a chad was detached from its card. Several recounts and lawsuits ensued. In the end, voters felt disenfranchised because of the belief that all votes should be counted.

Even the gold standard of voting, paper ballots, has defects. Due to various factors, the intent remains open to interpretation which can influence the election outcome. These difficulties resulted in the US federal government mandating the modernization of election systems. Many

states chose optical systems, but a few decided on *direct recording electronics (DRE)*. DREs are small computer devices that voters interact with through a simple interface. Even though DREs appear as a revolutionary leap in election systems, in the rush to get to market, the systems were flawed resulting in a host of new problems. In one instance [12], approximately 3000 votes were miscounted. Further complicating the matter, the systems did not provide an audit trail, hence making the recounts impossible. Some states are mandating that the units be supplemented with a paper printer that produces receipts that the voter verifies and then deposits in a sealed box. If a recount is necessary, the receipts are tallied instead of the digital results.

There exists several electronic voting (*e-voting*) schemes in the literature. The schemes have varying degrees of security and practicality. In general, there appears to be an inverse relationship between security and practicality, i.e. systems that are highly secure have little practicality and systems that are highly practical are minimally secure.

We are proposing an e-voting scheme that has a good balance of security and practicality. The voters determine the degree of anonymity which they require. Voters requiring greater anonymity obtain it by performing additional protocol iterations. The scheme produces ballots that are publicly tallied. This is beneficial as (i) individuals can identify if their vote was counted; (ii) recounts are simple; and (iii) the trust is enhanced.

We define a set of criteria in order to characterize voting systems. The set of criteria is divided into two broad categories: *practicality* and *security*. A system is *practical* if it is convenient, applicable to a wide-range of technologies, scalable, flexible, and accessible.

The *security* category consists of the following criteria:

1. *Democratic*. Only eligible voters can cast votes and no voter can cast more than one vote.
2. *Private*. All votes should remain private while voting is in progress.
3. *Accurate*. No vote can be altered, tampered, duplicated, or eliminated without being detected.

4. *Fair*. No observer can gain any knowledge about the partial tally before the votes are counted.

5. *Robust*. Faulty behavior of a reasonable sized coalition of participants is tolerable. No coalition of voters can affect the election and faulty (malicious) voters are detected.

6. *Voter verifiable*. A voter can determine that his vote was correctly counted.

7. *Universal verifiable*. Any observer, passive or otherwise, can be convinced that the final tally is correctly computed from the votes that were cast.

8. *Incoercible*. No voter should be able to prove the value of his vote to another party.

Related work. Electronic voting schemes can be divided into four categories: schemes based on mixnets [4], schemes based on blind signatures [9], schemes based on homomorphic secret sharing [3], and schemes based on homomorphic encryption [7].

Chaum [4] was the first to introduce mixnets. Chaumian mixes are simple RSA decryption mixes in which every server along the route between sender and receiver decrypts one layer of the message. Another type of mix is the re-encryption mix [10] that randomizes based on re-encryption. This mix type has greater resilience to failure than the Chaumian mixes. Acquisti [1, 2] proposed the concept of *user-centric mixnets* in which voters manage their privacy requirements. Voters collaborate with a third-party in order to exchange ballots among themselves. This technique has increased resilience over earlier systems. Chaum [6] supplemented mixnets with visual cryptography. This was the first model that supplied incoercible receipts to the voter. Among other attributes, the receipts permit the voter to verify that his vote is tallied. Vora [18] describes an implementation of the system. To reduce complexity and cost, Ryan and Bryans [15] designed a simpler visual encoding using a pair of aligned strips in place of visual cryptography. A strip contains a single row of symbols that when aligned with its partner reveals the vote.

Chaum [5] introduced blind signatures as a method to authenticate a message without knowing the content of the message. The signature is unlinkable: a signer cannot derive the correspondence between the signing process and the publicly-available signature. An election system by Fujioka *et al.* [9] solves the problem of anonymously validating votes by utilizing blind signatures. Several other systems use blind signatures (e.g. [14]), but all the systems experience the same drawback; voters should not abstain after the registration phase.

Contribution. We propose an election scheme that combines Acquisti's [2] user-centric mixnets with Chaum's [6] voter-verifiable receipts. The system in its entirety exhibits several positive attributes.

The user-centric mixnets empower voters as they deter-

mine themselves the degree of anonymity. Voters requiring a greater degree of anonymity achieve it by performing several protocol iterations. Additionally, these mixnets demonstrate better fault tolerance. During the mix progression, voters discover faulty parties and compensate by engaging other operational parties.

Voter-verifiable receipts require reformulation when combined with the user-centric mixnet. The proposed receipts successfully maintain most of the properties of the originals.

The scheme is robust as no reasonable-sized coalition can interfere with the correct operation. Furthermore, the scheme is efficient as the number of transmitted messages increases linearly with the number of voters.

Organization. The paper is structured as follows. In Section 2 we present the primitives on which our election scheme is based. In Section 3 we present and detail our proposed election scheme. In Section 4 we discuss the merits and costs. In Section 5 we draw conclusions and present future directions.

2. Preliminaries

2.1 Cryptographic Primitives

Public key cryptography, also called asymmetric key cryptography, is a family of cryptographic algorithms which use two keys. One key is the private key that must be kept secret, while the other key is the public key which is advertised. The keys function oppositely; when the public key is used to encrypt a message, the private key must be used to decrypt the ciphertext.

The ElGamal encryption cryptosystem [11] is a family of asymmetric key cryptographic algorithms exploiting the intractability of the discrete logarithm problem. In the usual case, ElGamal is performed over a multiplicative subgroup \mathbb{G} of order q in \mathbb{Z}_p^* where p, q are large primes.

- *Key generation (KG)*: Output key set $(PK, SK) = [(p, g, h = g^x), x]$ for large prime p and $g, h \in \mathbb{Z}_p$.
- *Encryption (E)*: Input comprises a message m , a public key (p, g, h) , and a random encryption factor $k \in \mathbb{Z}_p$. The output is a ciphertext $C = (G, M) = (g^k, mh^k)$. We write $C = E_{PK}(m, k)$ or $C = E_{PK}(m)$ for brevity.
- *Decryption (D)*: Input is a ciphertext $C = (G, M)$ under public key (p, g, h) . Compute $m = M/G^x$. We write $m = D_{SK}(C)$.

The ElGamal cryptosystem supports a (t, n) robust threshold scheme [8]. The purpose of a robust threshold

cryptosystem is the fault-tolerant sharing of the private key such that messages can be decrypted when $t \leq n$ trustees cooperate; any coalition of $t - 1$ or fewer trustees cannot reconstruct the key and thus cannot decrypt messages. This scheme is *resilient*; any coalition of at most $n - t$ trustees can not influence the correctness of the other trustees.

- *Key generation (KG)*: Output key $(PK, (SK_1, SK_2, \dots, SK_{n-1}, SK_n)) = [(p, g, h = g^x), (x_1, x_2, \dots, x_{n-1}, x_n)]$ for large primes $p, g, h \in \mathbb{Z}_p$, and $x = \sum_{j \in \mathcal{T}} x_j \lambda_{j, \mathcal{T}}$, $\lambda_{j, \mathcal{T}} = \prod_{l \in \mathcal{T} - \{j\}} \frac{1}{t-j}$ where \mathcal{T} is the set of trustees. The key generation step is executed by a third party who distributes the results to the given participants.
- *Decryption (D)*: Input is a ciphertext $C = (G, M)$ under public key (p, g, h) . Trustee T_j broadcasts $w_j = G^{x_j}$ and proves in zero-knowledge that $\log_g h_j = \log_G w_j$. Compute $m = M/G^x$ for $g^x = g^{\sum_{j \in \mathcal{T}} x_j \lambda_{j, \mathcal{T}}}$.

2.2 Mix networks

Mix networks (mixnets) are mechanisms to unlink message senders from receivers. Chaum [4] proposed mixnets in the context of an anonymous email system. The function of mixnets is to randomize a sequence of mutated messages such that the inputs and outputs are unlinkable. Messages are mutated by either encrypting, decrypting, or re-encrypting them.

The Chaumian variant of mixnets uses “*onion encryption*” and RSA decryption. In a mixnet, there are n mix servers $M_1, \dots, M_j, \dots, M_n$, each with key set (PK_j, SK_j) for $j = 1, \dots, n$. When a message m is to be transmitted anonymously, m is encrypted as $E_1(\dots E_j(\dots E_n(m)))$ and then transmitted to M_1 .

M_j waits until it receives several encrypted messages. Upon reaching a threshold on the number of messages, it removes one layer of encryption from m :

$$D_j(E_j(E_{j+1}(\dots E_n(m)))) = E_{j+1}(\dots E_n(m))$$

then shuffles and transmits the encrypted messages to M_{j+1} . The final server M_n removes the remaining layer of encryption, shuffles the batch, and transmits m to the recipient.

Traditionally, users have minimal input in the mixnet processing. Users requiring more anonymity than what is provided cannot obtain it. *User-centric mixnets* [2] solve the problem. These mixnets require that the users participate in the mix. The advantage is that users are empowered in the sense that a user requiring more anonymity can achieve it by performing several mix iterations. Additionally, these mixnets demonstrate better fault tolerance. Users

discover faults among the participants and compensate by engaging other participants.

2.3 Visual Cryptography

Naor and Shamir [13] introduced visual cryptography to conceal images without cryptographic computations. The cryptosystem works by encoding the plain text message m into a printed page of ciphertext and n transparencies that encode the key. The message m is visually observed when the ciphertext and $k \leq n$ transparencies are aligned, even though individually they are indistinguishable from random noise. The technique in effect is the one-time pad, which Shannon [17] demonstrated to be information-theoretic secure.

Chaum [6] adapted the concept into voter-verifiable receipts. Upon completion of voting, the voter is presented his vote summary on a printout. The printout is composed of two layers of superimposed transparencies. When the layers are separated, the vote becomes indiscernible from random noise. The layers are divided into square grids of equal parts where each square contains one of two pixel symbols. The pixel symbols are 2×2 square grids with the squares on a diagonal filled. The symbols are reverses of each other; where one is black, the other is white, and vice versa. When the layers are properly aligned, each symbol has a paired symbol on the opposite layer. A square in the resulting printout is grayed when the layers have the same symbol and is opaque when the layers have different symbols.

The printout is crafted by first generating a random ciphertext (white sheet) and then choosing the key pixels on the key sheet (red sheet) to obtain the image. To eliminate the possibility of faked layers, red pixels must be dispersed between the layers. This is achieved by randomly swapping half the pixel pairs between layers. After swapping, both $m \times n$ layers contain $(m \times n)/2$ white pixels and $(m \times n)/2$ red pixels. The layers are digitized. Both layers contain 4-tuples (L^z, q, D_N^t, D_N^b) , where L is the $m \times n$ matrix representation of the layer, q is the serial number, D is a doll, and z is either t for the top layer or b for the bottom layer. The dolls contain information to generate half the random values. The dolls are protected by encryption and are decrypted in a N layer mixnet. The user verifies that the ballot image $B = L_t \oplus L_b$ and that the last three tuple components are the same on both layers. The voter commits to his vote by selecting a layer and destroying the other.

The ballot image is restored by the mixnet operation. The duo T_N, D_N is the input of the mix, where $T_N = L^x$, $D_N = D_N^y$, and $x \neq y$. The trustee M_j removes a layer from D_j resulting in D_{j-1}, h_j , where h_j is the receipt contribution from T_j . The trustee computes $T_{j-1} = T_j \oplus h_j$ and forwards T_{j-1}, D_{j-1} to trustee M_{j-1} . After the final

trustee, $T_0 = B_z$, where B_z is a ballot image half.

3. The Proposed Scheme

In describing our voting scheme we use the following notations:

i) V_i is voter i . $\mathcal{V} = \{V_1, V_2, \dots, V_n\}$ is the set of n voters.

ii) T_i is trustee i . $\mathcal{T} = \{T_1, T_2, \dots, T_s\}$ is the set of s trustees.

iii) $E_{PK}(m)$ is the encryption of message m under PK .

iv) $Sig_{SK}(m)$ is the secure digital signature of m under private key SK . $S_{SK}(m) = (m, Sig_{SK}(m))$ is the digitally signed message m under private key SK .

Players. There are four players in the system: *voters*, *facilitator(s)*, *bulletin board(s)*, and *trustees*. A voter should be able to determine that his vote was counted and that it is anonymous. The facilitators and bulletin boards ensure anonymity. The trustees are responsible for ensuring the vote tally.

A facilitator is associated with a bulletin board to which he can post. The bulletin board is immutable, i.e. messages cannot be modified once they are posted to the board. The board is divided into slots, each slot is independent of the others.

Receipts. The receipts as proposed by Chaum [6] require reformulation to be successfully integrated with user-centric mixnets. The visual encoding scheme remains unchanged as in Section 2.3. What differs is that both layers are represented as a 3-tuple $(L^c, q, E_{PK}(B, k))$, where $c \in \{top, bottom\}$, q is the serial number, and $B = L^{top} \oplus L^{bottom}$ is the ballot image. The random encryption factor is obtained as: $k = h(S_{PK}(q))$, where $S_{PK}(q)$ is the digitally signed serial number q under public key PK and h is a public one-way function.

The voter verifies that the receipt is generated correctly: (i) He checks that $L^{top} \oplus L^{bottom} = B$. (ii) He confirms that $q, E_{PK}(B)$ are identical on both layers. (iii) He evaluates $E_{PK}(B)$. If any of the checks fails, it is undeniable evidence that the polling station malfunctioned.

The ballot image is reconstructed by removing L^c, q , and by submitting $E_{PK}(B)$ to the trustees. The trustees decrypt $E_{PK}(B)$ using the threshold robust ElGamal algorithm (Section 2.1).

Keys. Players are required to have public/private key sets. If (PK, SK) is the key set, PK and SK represent the public and private keys respectively. It is implicitly understood that a *public key infrastructure (PKI)* exists and all public keys are registered to it. We use the following notation for the key sets:

i) $(PK_{\mathcal{T}}, [SK_{T_1}, SK_{T_2}, \dots, SK_{T_s}])$: key set for the trustees \mathcal{T} , where T_j is trustee j .

ii) (PK_F, SK_F) : key set for the facilitator.

iii) $(PK_{V_i}^t, SK_{V_i}^t)$: t^{th} key set for voter V_i . No one, except V_i , knows that a link exists between $PK_{V_i}^t, PK_{V_i}^{t+1}$, and V_i .

Tokens. When the system deems a voter eligible, it transmits a token to the voter. A token is a unique *receipt of eligibility*. A token is generated for an identity and thereafter is linked to a voter. Tokens are inputs to the mix where the identity-token relationships are severed.

A token is *redeemed* for a vote submission. The token is authenticated before accepting the vote. Authentication includes verifying that the token is *well-formed* and unredeemed. Token and token construction must satisfy the following constraints: (i) Tokens must be difficult to counterfeit. By extension, they must be easily verified. (ii) Tokens must offer *replay protection*. (iii) Tokens are valid only for the election for which they are generated. A token crafted for election A cannot be used for election B and vice versa.

As previously stated, tokens are the objects exchanged during the mix. Beside as a receipt of eligibility, tokens are necessary because, unlike ballots, they are *equivalent*. Tokens have the same value to all parties, similar to how a quarter is worth \$0.25 to everyone. Ballots, on the other hand, are dependent on the vote and the voter; a vote for George W. Bush has a vastly different value to a Republican than to a Democrat.

In the following we present our voting scheme. The sequence of messages exchanged by the participants in this scheme is presented in Figure 1.

Initialization: The trustees initialize the robust threshold ElGamal cryptosystem. A third party generates $(PK_{\mathcal{T}}, [SK_{T_1}, SK_{T_2}, \dots, SK_{T_s}])$ and distributes the share SK_{T_j} to trustee $j = 1, 2, \dots, s$. The trustees publish $PK_{\mathcal{T}}$. Finally, the trustees generate the tokens that will be used for the election.

Eligibility: The voter interacts with a polling station at his predetermined precinct. He presents his identity *IDENT* to the polling station¹. The polling station transmits *IDENT* to the facilitator where it is verified. If necessary, the facilitator invokes a challenge². If *IDENT* is eligible, the facilitator sends token *TOK* to the voter.

Voting: The voter votes using the interface supplied by the polling station. The station encodes the vote (Section 2.3) and presents a printout. If the summary satisfies

¹The identity can be stored or recalled from magnetic strip cards, smart cards, RFID chips (*radio frequency identification*), or biometrics. The traditional eligibility process is easily amendable to support the system requirements.

²Challenges are dependent on the authentication scheme. They could be as simple as requesting the voter's home address or a pass phrase.

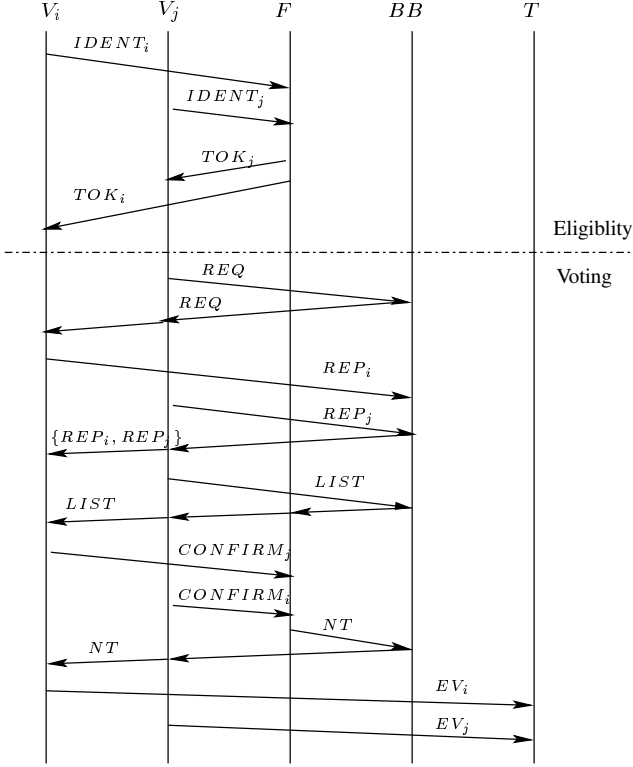


Figure 1. The sequence of messages exchanged by the participants.

the voter, the voter commits by choosing a layer as the receipt and destroying the other. The polling station commences the mix operation.

A voter can perform several mix iterations depending on his personal anonymity requirement. A mix iteration requires the usage of a single bulletin board slot; no two iterations will share a slot. In the following, t denotes the iteration number.

1.) The voter V_j posts $REQ = S_{SK_{V_j}^t}(PK_{V_j}^t)$ to the bulletin board. This message is a request to exchange tokens with other voters.

2.) Voter V_i observes the request and posts $REP_i = E_{PK_{V_j}^t}(S_{SK_{V_i}^{t+1}}(PK_{V_i}^{t+1}, REQ))$.

The message expresses the willingness of V_i to exchange his token. If several REQ exists, V_i randomly chooses one.

3.) At some point in time, V_j proceeds with the transaction. He generates

$$LIST = (S_{SK_{V_j}^t}([PK_{V_{\pi(1)}}^{t+1}, PK_{V_{\pi(2)}}^{t+1}, \dots, PK_{V_{\pi(l)}}^{t+1}]))$$

and posts it to the bulletin board and transmits it to F ,

where $l \leq n$ and π is a private one-way permutation. The voters compiled into $LIST$ are selected based on voters transmitting valid messages from step 2. In practice, $LIST$ is thought of as a *transaction identifier*.

4.) V_i searches for his public key in $LIST$, where $i = 1, 2, \dots, n$. If he finds his public key, he transmits

$$CONFIRM_i = S_{SK_{V_i}^{t+1}}(LIST, E_{PK_F}(TOK_{V_i}^t))$$

to F . If he fails to find his public key, he continues the protocol from step 7.

5.) F waits until it receives $CONFIRM_i$ from all V_i published in $LIST$. If F does not receive the message from all V_i in $LIST$, the protocol terminates; all V_i continue from step 7. F authenticates all tokens $TOK_{V_i}^t$ and confirms that they have yet to be redeemed. F “redeems” $TOK_{V_i}^t$ and obtains $TOK_{V_i}^{t+1}$ for $i = 1, 2, \dots, l$. As stated above, only V_i knows the linkage $PK_{V_i}^t, PK_{V_i}^{t+1}$, and himself. Hence, F cannot link $TOK_{V_i}^{t+1}$ to V_i .

6.) F posts $NT = S_{SK_F}(E_{PK_{V_{\pi(1)}}^{t+1}}(TOK_{V_{\pi(1)}}^{t+1}),$

$$E_{PK_{V_{\pi(2)}}^{t+1}}(TOK_{V_{\pi(2)}}^{t+1}), \dots, E_{PK_{V_{\pi(l)}}^{t+1}}(TOK_{V_{\pi(l)}}^{t+1}))$$

The position of $TOK_{V_i}^{t+1}$ is determined by the index of $PK_{V_i}^{t+1}$ within $LIST$. V_i retrieves $TOK_{V_i}^{t+1}$.

7.) Optionally, V_i participates in another mix iteration starting from step 1.

8.) V_i transmits

$$EV_i = S_{SK_{V_i}^{t+1}}(PK_{V_i}^{t+1}, TOK_{V_i}^{t+1}, q, E_{PK_T}(B))$$

to the trustees \mathcal{T} via a secure channel, thus completing the mix.

Tallying: After the polls close, all provisional and/or contested voting is resolved. L^c, q are stripped from the receipts and L^c , ordered by q , are posted to the official, publicly-accessible poll website. The trustees decrypt $E_{PK}(B)$ using the robust threshold ElGamal algorithm (Section 2.1) and post B to the previously mentioned website. The votes are publicly counted from the ballot images.

4. Properties and Complexity

Properties. The proposed scheme satisfies the criteria presented in Section 1 as follows:

1. *Democratic.* The facilitator ensures the eligibility of voters and grants an eligible voter a single token. Since vote submission requires the redeeming of a token and a voter is granted a single token, duplicate votes are detected and excluded.

2. *Private*. Facilitators cannot link voter identity to the tokens. By extension, facilitators cannot link voter identity to a vote. Trustees have no means to link voter identity, token, and vote. Even if the facilitators and the trustees collude, they have no means to determine the link between voter and vote. Finally, if the utilized public key cryptosystem is secure, than the vote value is protected.

3. *Accurate*. There are several points at which accuracy can be determined. If the printout or the receipt has errors, it is undeniable evidence that the system is corrupt. If the receipt is not posted to the official poll website and the voter has a receipt, it is undeniable evidence that the system is corrupt.

4. *Fair*. The tallying follows voting. Additionally, the votes are protected by a (t, n) threshold robust cryptosystem. A coalition of t faulty trustees is required to void fairness.

5. *Robust*. A token is necessary to engage with the system. The only method a malicious voter has is the manipulation of tokens. These tokens are detected and rejected by the system. A faulty facilitator is detected during vote submission when redeeming tokens. As stated above, the votes are protected by a (t, n) robust threshold cryptosystem. As long as there exists a coalition of $t - 1$ correct trustees, the vote will remain secure.

6. *Voter verifiable*. The voter can verify that his vote is properly encoded and that his vote is included in the final batch of votes to be tallied.

7. *Universal verifiable*. The votes are tallied in public where any observer can verify the tally himself.

8. *Incoercible*. The voter cannot use his receipt to prove his vote value. L^c appears random and is indiscernible from random noise. L^c, q are detached from B , thus a relationship does not exist between the receipt and the ballot image.

Communication Complexity. The message count for the various phases is as follows: the eligibility phase requires two messages; the mix phase requires $n + 2l + 3 \leq 3n + 3 \approx 3n$ messages per iteration; and one message to communicate the vote to the trustees. If there are n possible voters and each voter participates in k iterations, the overall message communication complexity is $O(nk)$.

Due to space constraints, we are not able to present here a case study and a detailed analysis. They will be presented in an extended version of this paper.

5. Conclusion

We have presented a voter-controlled, voter-verifiable e-voting scheme. The main advantage of this scheme is that it empowers voters; voters requiring higher degrees of anonymity can achieve them. The incoercible, voter-verifiable receipts maintain the properties of the original

Chaumian receipts. The scheme is efficient in terms of message complexity as it increases linearly with respect to the number of voters. In future work we will further investigate the implementation details and provide a prototype implementation.

References

- [1] A. Acquisti. An anonymous, fair voting/recommendation system. Technical report, School of Information Management and Systems, UC Berkeley, 2002.
- [2] A. Acquisti. An user-centric MIX-net protocol to protect privacy. In *Proc. of the Workshop on Privacy in Digital Environments: Empowering Users*, Nov. 2002.
- [3] J. Benaloh. *Verifiable Secret-Ballot Elections*. PhD thesis, Yale University, 1987.
- [4] D. Chaum. Untraceable electronic mail, return address, and digital pseudonym. *Communications of ACM*, 24(2):84–88, Feb. 1981.
- [5] D. Chaum. Blind signatures for untraceable payments. In *CRYPTO '82*, pages 199–203. Plenum Press, 1982.
- [6] D. Chaum. Secret-ballot receipts: True voter-verifiable elections. *IEEE Security & Privacy*, 2(1):38–47, Jan./Feb. 2004.
- [7] R. Cramer, R. Gennaro, and B. Schoenmakers. A secure and optimally efficient multi-authority election scheme. In *EUROCRYPT '97*, volume 1233 of *LNCS*, pages 103–118. Springer-Verlag, 1997.
- [8] Y. Desmedt and Y. Frankel. Threshold cryptosystems. In *Proc. of Advances in Cryptology – CRYPTO '89*, volume 435 of *LNCS*, pages 307–315. Springer-Verlag, 1990.
- [9] A. Fujioka, T. Okamoto, and K. Ohta. A practical secret voting scheme for large scale elections. In *AUSCRYPT '92*, volume 718 of *LNCS*, pages 224–251. Springer-Verlag, 1993.
- [10] M. Jakobsson. Flash mixing. In *Proc. of the 18th ACM Symposium on Principles of Distributed Computing (PODC '99)*, pages 83–89. ACM, 1999.
- [11] A. J. Menezes, P. C. von Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, Inc., 1996.
- [12] L. Monteagudo Jr. and H. Gao. Some votes miscounted in primary, officials say. *The San Diego Union-Tribune*, Apr. 8 2004.
- [13] M. Naor and A. Shamir. Visual cryptography. In A. D. Santis, editor, *Proc. Advances in Cryptography (EUROCRYPT '94)*, volume 950 of *LNCS*, pages 1–12. Springer-Verlag, 1995.
- [14] H. Petersen, P. Horster, and M. Michels. Blind multisignature schemes and their relevance to electronic voting. In *Proc. of the 11th Annual Computer Security Applications Conference*, pages 149–155. IEEE Press, 1995.
- [15] P. Y. A. Ryan and J. W. Bryans. A simplified version of the Chaum voting scheme. Technical report, School of Computing Science, University of Newcastle upon Tyne, UK, May 2004.
- [16] M. Shamos. Theory v. practice in electronic voting. In *DIMACS Workshop on Electronic Voting – Theory and Practice*, Rutgers University, New Jersey, May 26–27 2004.
- [17] C. E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28(4):656–715, 1949.
- [18] P. Vora. David Chaum's voter verification using encrypted paper receipts. In *DIMACS Workshop on Electronic Voting – Theory and Practice*, Rutgers University, New Jersey, May 26–27 2004.