A Framework for Receipt Issuing, Contendable Remote Poll-Site Voting

Prashanth P. Bungale
prash@cs.jhu.edu

Swaroop Sridhar
swaroop@cs.jhu.edu

Department of Computer Science
The Johns Hopkins University, Baltimore, MD 21218, U. S. A.

Abstract

This paper presents a new framework for an electronic voting system in which a voter can vote not only from his home poll-site, but from any poll-site, in a manner that ***guarantees total voter anonymity and privacy****. The core concentration of our work is the design of a mechanism in which a voter can be given a* ***receipt*** *to acknowledge his vote and at the same time preventing any occurrence of vote-selling or voter coercion. The voter shall be able to verify his vote – from anywhere – after election results have been published. If deemed necessary, the voter shall be able to* ***anonymously contend*** *the election results from any election office.*

Much of the present literature views receipt-freeness as the necessity for precluding vote-selling and voter coercion. They present a "clear" picture of a tradeoff between the "mutually exclusive" issues of receipt issuance and voter security. The solution we propose overcomes this tradeoff and thus ensures the voter's confidence in that his vote has been counted as cast, without compromising voter security and privacy.

1. Introduction

People all over the world are starting to take a hard look at their voting equipment and procedures, and trying to figure out how to improve them. There is a strong inclination towards moving to Electronic Voting in order to enhance voter convenience, increase voter confidence and voter turnout. However, there are serious technological and social aspects that come into play while designing the voting system, which need to be addressed.

The main focus of our work is addressing the open question of providing a mechanism in which a voter can be given a *receipt* to acknowledge his vote (which facilitates vote confirmation and vote contention), and at the same time prevent any occurrence of vote-selling or voter coercion. Much of the present literature views receipt-freeness as the *necessity* for precluding vote-selling and voter coercion. There is a "clear" picture presented, of a tradeoff between the "mutually exclusive" issues of receipt issuance and voter security. We have overcome this tradeoff and thus ensured the voter's confidence in that his vote has been counted as cast, without compromising voter security.

2. Related Work

Electronic voting refers to the use of computers or computerized voting equipment to cast ballots in an election. Sometimes, this term is used more specifically to refer to voting that takes place over the Internet. Electronic systems can be used to register voters, tally ballots, and record votes [1].

The Caltech/MIT Voting Technology Project [2] came into being in order to develop a new voting technology in order to prevent a recurrence of the problems that threatened the 2000 U. S. Presidential Elections. The report assesses the magnitude of the problems, their root causes and how technology can reduce them. They address a wide range of "What is" issues including voting procedures, voting equipment, voter registration, polling places, absentee and early voting, ballot security, cost and public finance of elections, etc. They then propose a novel "What could be" framework for voting technology (that moves away from monolithic voting structures), and propose that a process for innovation be setup. The framework is "A Modular Voting Architecture ("Frogs")" [3,4,5] in which vote generation is performed separately from vote casting, and the "Frog" forms a permanent audit trail, the importance of which cannot be over-stressed. Here, the vote generation machine can be proprietary whereas the vote casting machine must be open-source and thoroughly verified and certified for correctness and security. Finally, the report provides a set of short-term and long-term recommendations on the various issues related to voting.

In "Electronic Voting" [6], Rivest addresses some issues like the "secure platform problem" and the (im)possibility of giving a receipt to the voter. He also provides some personal opinions on a host of issues including the striking dissimilarity between e-commerce and e-voting, the dangers of adversaries performing automated, wide-scale attacks while voting from home, the need for extreme simplicity of voting equipment, the importance of audit-trails, support for disabled voters, security problems of absentee ballots, etc.

The NSF Internet Voting Report [7] addresses the feasibility of different forms of Internet voting from both the technical and social science perspectives, and defines a research agenda to pursue if Internet voting is to be viable in the future. It groups Internet voting systems into three general categories as follows:

- *Poll-site Internet voting*: It offers the promise of greater convenience and efficiency in that voters could cast their ballots from any poll site, and the tallying process would be both fast and certain. More importantly, since election officials would control both the voting platform and the physical environment, managing the security risks of such systems is feasible.
- *Kiosk voting*: Voting machines would be located away from traditional polling places, in such convenient locations as malls, libraries, or schools. The voting platforms would still be under the control of election officials, and the physical environment could be modified as needed and monitored (e.g., by election officials, volunteers, or even cameras) to address security and privacy concerns, and prevent coercion or other forms of intervention.
- *Remote Internet voting*: It seeks to maximize the convenience and access of the voters by enabling them to cast ballots from virtually any location that is Internet accessible. While this concept is attractive and offers significant benefits, it also poses substantial security risks and other concerns relative to civic culture. Current and near-term technologies are inadequate to address these risks.

The report presents some findings on the feasibility of each of these categories and provides research recommendations for the long-term future. It also identifies criteria for election systems and addresses the technological and social science issues.

The California Internet Voting Report [8] suggests a strategy of evolutionary rather than revolutionary change towards achieving the goal of providing voters with the opportunity to cast their ballots at any time from any place via the Internet. The report defines four distinct Internet voting models – Internet voting at voter's polling place, Internet voting at any polling place, Remote Internet voting from County computers or kiosks, Remote Internet voting from any Internet connection – and the corresponding technical and design requirements that must be met when implementing any of the stages. It addresses the advantages, implementation and security issues of each of the four stages. They believe that additional technical innovations are necessary before remote Internet voting can be widely implemented as a useful tool to improve participation in the elections process and that current technology however would allow for the implementation of new voting systems that would allow voters to cast a ballot over the Internet from a computer at any one of a number of county-controlled polling places in a county. Finally, the report presents the findings and recommendations of the task force on policy issues.

An extensive survey of e-voting technology has been provided in "e-Voting Security Study" [9]. It provides a survey of recent academic and commercial projects in the area, in addition to the area's prominent academics' personal views and testimonies regarding the issues. It identifies threats, potential sources of attack and possible methods of attack in such voting systems. It also identifies security objectives and requirements of an electronic voting system.

The foundation of much of the academic work in the area of remote voting is a paper by Fujioka, Okamoto and Ohta (FOO) [10]. It gives a mathematical framework for a secure election that involves an administrator, and a counter and the voter connected by an anonymous channel. Practically focused projects build on the blind voting protocol proposed in this paper. Sensus [11] uses blind signatures to ensure that only registered voters can vote and that each registered voter votes exactly once, while at the same time maintaining voter's privacy. It allows voters to verify independently that their votes were counted correctly and anonymously challenge the results, should their votes be miscounted. Another project called E-VOX [12] at MIT implements a simplified, user-friendly version of the FOO framework using Java, Netscape and JDBC (Java Database Connectivity). This system is still involved in teaching and research and was used for an Undergraduates Association election at MIT in 1999. "Multiple

Administrators for Electronic Voting” [13] improves this further by distributing the authority among multiple administrators to prevent vote forging.

“An untraceable, universally verifiable voting scheme” [14] presents a remote voting scheme that applies the technique of blinded signature to a voter's ballot so that it is impossible for anyone to trace the ballot back to the voter. They achieve the desired properties of privacy, universal verifiability, convenience and untraceability, *but* at the expense of *receipt-freeness*.

The E-Poll (Electronic Polling System for Remote Voting Operations) project [15] investigates broadband mobile communications based on the UMTS standard for providing the E-Poll network with the required bandwidth and security. This makes it possible to use E-Poll kiosks anywhere, within a private, reliable and protected network. The voter-recognition system is based on an innovative smart card with an embedded biometric fingerprint reader, which performs voter recognition with absolute security. An ergonomic kiosk facilitates use by disabled people.

The FREE e-democracy project [16] is dedicated to creating the GNU.FREE Internet Voting system and also advocating Free Software, which is non-partisan and non-commercial in origin.

[17] presents a system for secure electronic voting which does not rely on persistent network connections between polling places and the vote-tallying server. They build the system on a disconnected (or, more accurately, an intermittently connected) environment, which behaves well in the absence of network connectivity.

“Security Criteria for Electronic Voting” [18] considers some basic criteria for confidentiality, integrity, availability, reliability, and assurance for computer systems involved in electronic voting. After an assessment of the realizability of those criteria, it concludes that, operationally, many of the criteria are inherently unsatisfiable with any meaningful assurance.

In [19], Rubin identifies the new risks brought about by introducing the state-of-the-art technology into the election process, which may not be worth taking. The major security risks identified include those at the voting platform – including malicious payload (attack programs, remote administration and monitoring toolkits, etc.) and delivery mechanism (worms, viruses and bugs, active content downloaded automatically, etc.) – and the communications infrastructure – including (distributed) denial of service attack, DNS server attack, etc. He also identifies security issues in social engineering and in using specialized devices.

3. Requirements for an Electonic Voting System

In the following subsections, we identify the requirements for an electronic voting system, that will have to be met by the framework that we propose later.

3.1. Functional Requirements

1. **Mobility**: The voter should not be restricted to cast his ballot at a single poll-site at his home precinct. He shall be able to vote from any poll-site within the country, not only from his home poll-site.

2. **Convenience**: The system shall allow the voters to cast their votes quickly, in one session, and should not require many special skills or intimidate the voter (to ensure *Equality of Access to Voters*).
3. **Transparency**: Voters should be able to possess a general knowledge and understanding of the voting process.
4. **Eligibility**: Only authorized voters, who are registered, should be able to vote.
5. **Uniqueness**: No voter should be able to vote more than once.
6. **Auditability**: It should be possible to verify that all votes have been correctly accounted for in the final election tally, and there should be reliable and demonstrably authentic election records, in terms of physical, permanent audit trail (which should not reveal the user's identity in any manner).
7. **Voter Confirmation**: The voter shall be able to confirm clearly how his vote is being cast, and shall be given a chance to modify his vote before he commits it.
8. **Receipt Issuance**: The system shall issue a receipt to the voter acknowledging his choices *if and only if it can be ensured that vote-coercion and vote-selling are prevented*, so that he may verify his vote at any time and also contend, if necessary.
9. **Provisional Ballots**: The voter shall be able to vote with a provisional (electronic) ballot if he has some registration problems, which could be counted if verified by the authorities later.
10. **Documentation and Assurance**: The design, implementation, and testing procedures must be well documented so that the voter-confidence in the election process is ensured.
11. **Cost-effectiveness**: The resulting election system should be affordable and efficient.

3.2. Security Requirements

1. **Voter Authenticity:** Ensure that the voter must identify himself (with respect to the registration database) to be entitled to vote.
2. **Registration**: The voter registration shall be done *in person* only. However, the computerized registration database shall be made available to polling-booths all around the nation.
3. **Voter Anonymity**: Ensure that votes must *not* be associated with voter identity.
4. **System Integrity**: Ensure that the system cannot be re-configured during operation.
5. **Data Integrity**: Ensure that each vote is recorded as intended and cannot be tampered with in any manner, once recorded (i.e., votes should not be modified, forged or deleted without detection).
6. **Secrecy / Privacy**: No one should be able to determine how any individual voted.
7. **Non-coercibility and No Vote-selling**: Voters should not be able to prove to others how they voted (which would facilitate vote selling or voter coercion).
8. **Reliability**: Election systems should work robustly, without loss of any votes, even in the face of numerous failures, including failures of voting machines and total loss of network communication. The system shall be developed in a manner that ensures there are no malicious codes or bugs.
9. **Availability**: Ensure that system is protected against accidental and malicious denial of service attacks. Also, setup *redundant communication paths* so that availability is ensured.
10. **System Disclosability**: The core of the system, especially the vote-casting equipment, shall be *open-source*, so that it can allow external inspection and auditing.
11. **Simplicity**: The system shall be designed to be extremely simple, as complexity is the enemy of security.

12. **Testing and Certification**: The system should be tested by experts with respect to all of the security considerations, so that election officials have the confidence that the system meets the necessary criteria.
13. **System Accountability**: Ensure that system operations are logged and audited.
14. **Operator Authentication and Control**: Ensure that those operating and administering the system are authenticated and have strictly controlled functional access on the system.
15. **Distribution of Authority**: The administrative authority shall not rest with a single entity. The authority shall be distributed among multiple administrators, who are known not to collude among themselves (e.g., different political parties).

4. Design and Analysis

In this section, we present the protocol flows and an analysis of the various components of our design. We do this by walking through the various phases of the election process.

4.1. Voter Registration

Voter registration shall be done preferably *in person* at the various offices before the Election Day. The officer shall register the person's identification (Signature or ID no., etc.) and the offices for which he is eligible to vote, in the database.

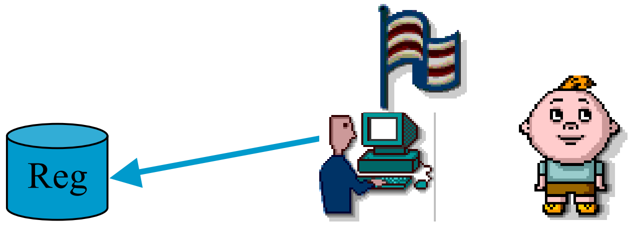


Figure 4.1: Voter Registration

4.2. Voter Authentication and Vote Initialization

On the Election Day, the officer at the poll site authenticates the voter against the registration database using some kind of identification information presented to him. Once the voter is authenticated, the officer initializes a memory stick with the voter's identification information and information about the ballot to be used, then, signs it with his private key, and hands it over to the voter.

In the case of remote poll-site voting, the officer at the remote poll-site gets the voter's registration information from the home database and uses this to authenticate the voter. He then requests and receives the identification information, signed by the officer at the voter's home poll-site (**S_{oH}**), and initializes a memory stick with it (**ID** + S_{oH}(ID)), which is finally handed over to the voter.

Analysis: Since the memory stick is signed by an election officer, the voter cannot forge votes, for example, by bringing lots of memory sticks with him to the poll-site. Moreover, the privacy and anonymity of the voter are totally uncompromised, as the memory stick is signed by the officer at his *home poll-site*, irrespective of where the voter actually casts his vote.

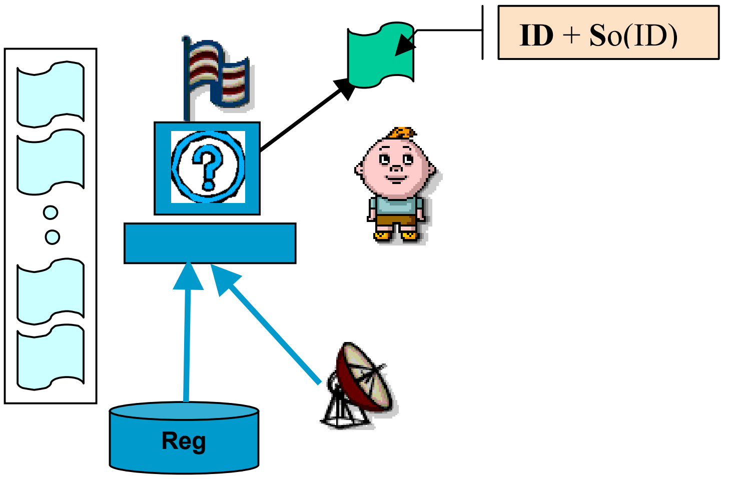


Figure 4.2: Voter Authentication and Vote Initialization

4.3. Vote Generation and Verification

4.3.1. The Voting Machine

The voting machine comprises of an input unit to record the voter's choices, a simple display unit and a memory stick reader/writer. It has sufficient computation power to execute encryption and other necessary cryptographic algorithms. It also has ports connected to network hosts, which help communicate with other voting machines. Lastly, it has output ports connected to the tallier(s).

The voting machine is designed to be simple, is completely open-source and is subject to thorough testing, verification and certification. The voter can be *sure* that the vote choices recorded in his memory stick are exactly as displayed on the display unit of the machine.

We require that both the vote generation and the vote casting machines be verified and certified (See *Appendix A for the reasons*). However, we do maintain the separation between the vote generation and vote casting phases. The modules may still be generated and verified independently. Nevertheless, since both of them need to be certified, we intend to club their functionality into a single machine – the voting machine. This greatly simplifies the voting process for the voter, since he does not have to take his memory stick back and forth between the vote generation and vote casting machinery.

4.3.2. Vote Generation

The voter takes the initialized memory stick and inserts it into a voting machine, placed such that his privacy is ensured. The machine presents the voter with the concerned ballot (downloading it, if necessary, from his home poll-site), and provides him with an easy-to-use and unambiguous interface to generate his vote. The interface provided may also cater to the needs of multilingual voters and disabled (e.g., blind) voters.

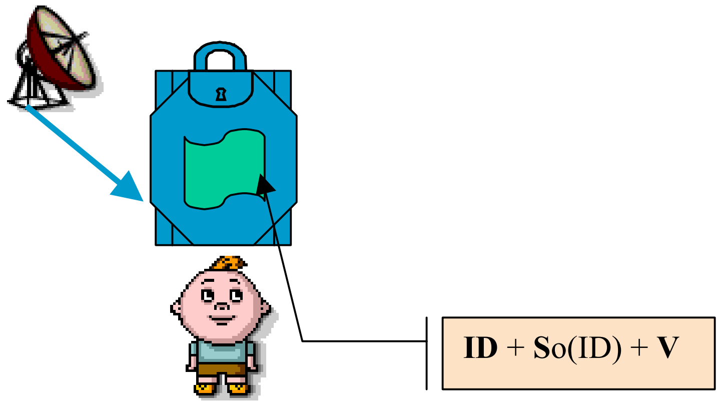


Figure 4.3: Vote Generation

4.3.3. Vote Verification

After the voter has made his choices, the machine finally displays all the information recorded by *retrieving* it from the memory stick. He then verifies the vote as displayed by the machine. If the voter is not satisfied with his current choices, he can revert back, correct his vote, and then re-verify his vote. This correction can be made as many times as he wishes.

Analysis: Since the voting machine is completely open-source and is subject to thorough testing, verification and certification, the voter can be *sure* that the vote choices recorded in his memory stick are exactly as displayed on the display unit of the machine. Also, since the voter is able to verify and correct his vote, we achieve the goal of reflecting the correct intention of the voter in the recorded vote.

4.4. Vote Casting, Audit Trail Generation, Issue of Receipts and Vote Tallying

4.4.1. Home poll-site voting

After the voter satisfactorily confirms his vote, he CASTS the vote, at which time the following actions are performed:

i) The vote, 'V', present on the memory stick, is signed by the officer's private key and by the private keys of many observers (possibly, mutually adversarial parties such as political parties) to get $\mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)_{\ldots}$*.

ii) A copy of this signed vote, $\mathbf{V} + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)_{\ldots,}$ is recorded on another memory stick, which is stored internally to form a part of the (anonymous) permanent physical audit-trail.

iii) Another copy of the signed vote is sent to the vote tallier, which will then take this vote into account for tallying. In addition to the vote tallier machine kept by the election office, the copy of the signed vote can optionally be broadcast to multiple vote tallier machines belonging to various observers (again, mutually adversarial parties). Note that the voter's identification information is not stored either on the audit trail or on the copies sent to the vote talliers.

iv) The vote - '**V**', along with all the signatures on the voter's memory stick are encrypted using the officer's public key to get $\mathbf{E}_o\{V + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)_{\ldots}\}$.

v) Now, a (unique) random number '**N**' is generated, and $\mathbf{E}_o\{V + \ldots\}$ generated above is further encrypted using '**N**' as the symmetric key, to get $\mathbf{A}_N\,[\mathbf{E}_o\{V + \ldots\}]$.

* - 'E' stands for public-key encryption; 'A' stands for symmetric-key encryption; 'S' stands for signature

vi) The final content of the memory stick is now "$\mathbf{ID} + \mathbf{S}_o(\mathrm{ID}) + \mathbf{A}_N\,[\mathbf{E}_o\{V + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)_{\ldots}\}]$". This is then permanently sealed (possibly by blowing a fuse on the memory stick) before the memory stick is finally handed over to the voter.

vii) A print-out of the number '**N**' and the name of the candidate voted for, is issued to the voter. This, along with the memory stick handed over in step (vi) above, forms the ***receipt*** for his vote cast. The print-out facilitates vote-verification and the memory stick facilitates vote-contention.

viii) Now that the voter has been issued a receipt for his vote cast, even containing the name of the candidate for whom he voted, this receipt could be abused for vote-selling / coercion. In order to prevent this from occurring, in addition to the correct receipt, the machine also issues *fake* receipts – identical to the legitimate one – which are print-outs of some more (unique) random numbers (n_1, n_2, n_3, …) with the names of all the other candidates. This step is necessary to prevent voter coercion and vote selling.

ix) Finally, all the (n_i, v_i) pairs, generated above are transmitted to the database.

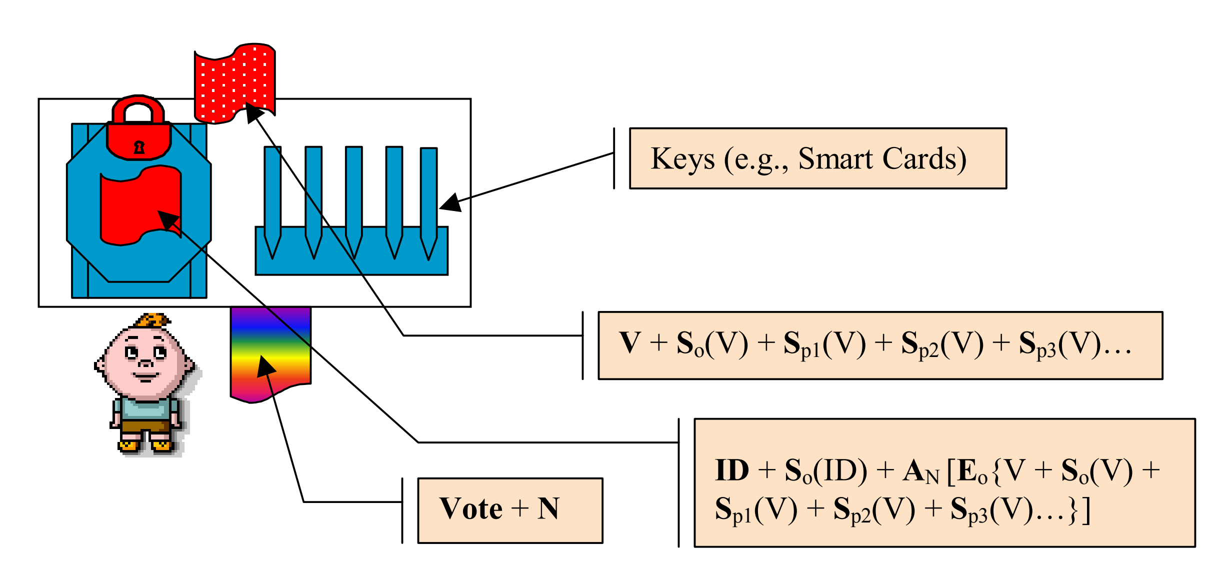

Keys (e.g., Smart Cards)
$V + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)\ldots$
$\mathbf{ID} + \mathbf{S}_o(\mathrm{ID}) + \mathbf{A}_N\,[\mathbf{E}_o\{V + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)\ldots\}]$
Vote + **N**

Figure 4.4: Vote Casting and Audit Trail Generation

Analysis:

i) Since the vote is signed by many observers – in addition to the election officer – who have no reason to collude amongst themselves, vote forging / ballot stuffing is prevented.

ii) Since the voter's identification information is not tagged on to his vote anywhere else other than the memory stick that the voter takes away with him, the anonymity and privacy of the voter are ensured.

iii) Since there is a physical audit trail, accurate recounts in the case of contested elections are facilitated.

iv) Since there are redundant talliers, incorrect tallies can be detected.

Since the political parties' talliers do not receive the 'N' values corresponding to the votes, there is no possibility of them knowing who voted for whom.

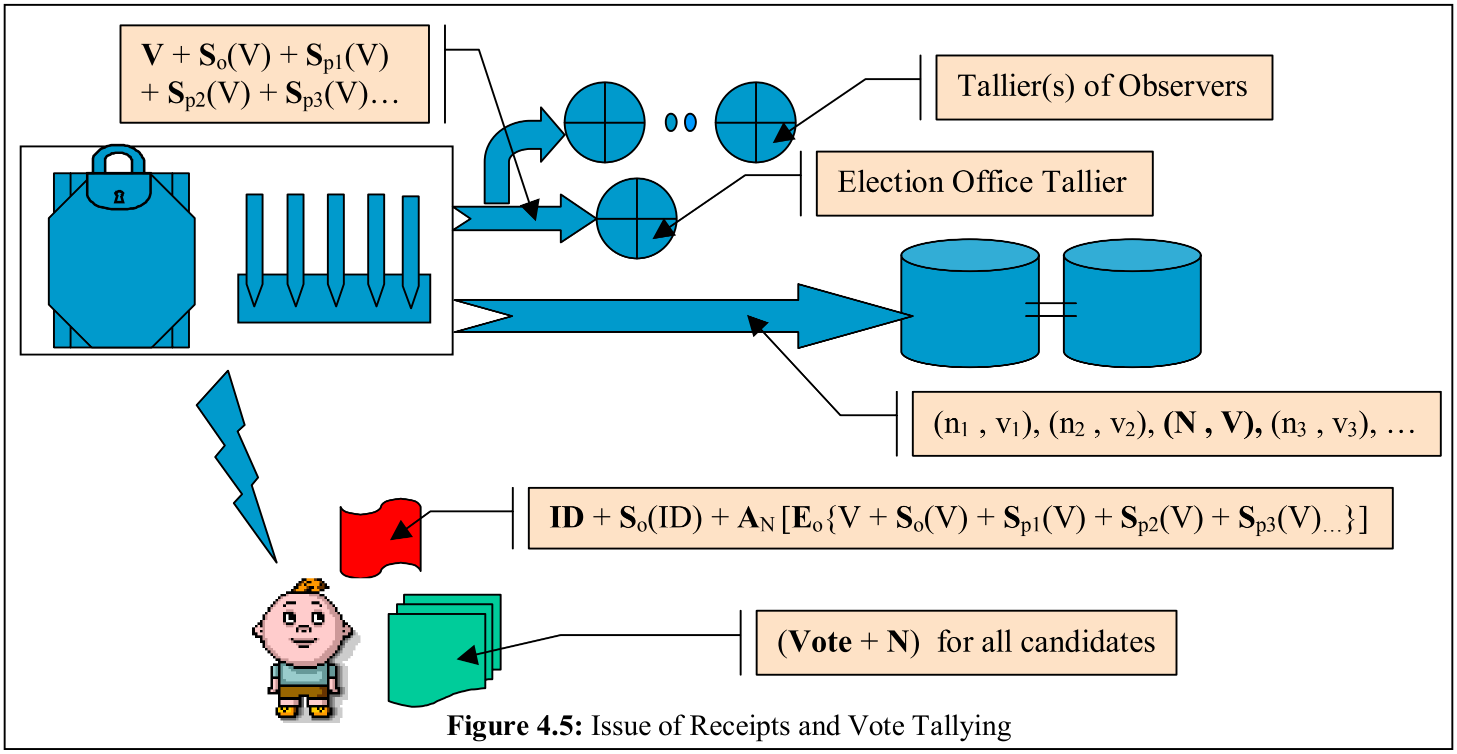

$\mathbf{V} + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)\ldots$
Tallier(s) of Observers
Election Office Tallier
$(n_1, v_1), (n_2, v_2), \mathbf{(N, V)}, (n_3, v_3), \ldots$
$\mathbf{ID} + \mathbf{S}_o(ID) + \mathbf{A}_N [\mathbf{E}_o\{V + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)_{\ldots}\}]$
(**Vote** + **N**) for all candidates

Figure 4.5: Issue of Receipts and Vote Tallying

v) Because:
  a. The vote is encrypted by the officer on the memory stick handed over to the voter
  b. Fake receipts (print-outs) identical to the legitimate one are issued

  There is no possibility that he can prove to somebody else that he has voted in some way. Thus, voter coercion and vote-selling are prevented.

vi) As all the (n_i, v_i) pairs are sent to the database each time a vote is cast, there is no way to tell which (n, v) pair is the real/valid one from the database.

vii) Since the memory stick is sealed, possibly by blowing a fuse or something similar to that, the vote can never be modified.

viii) At the time of vote contention (when the voter takes his memory stick to contend the way his vote was counted), since the vote on the memory stick is encrypted by the number 'N', known only to the voter, it is not possible even for the election officer to view the vote.

ix) Since the database of (n_i, v_i) is maintained in a redundant fashion, reliability is ensured.

4.4.2. Remote poll-site voting

In the case of remote poll-site voting, after the voter satisfactorily confirms his vote, he casts the vote, at which time the following actions are performed:

i) The vote, '**V**', present on the memory stick, along with the already signed identification information, is encrypted with the home poll-site officer's public key to get $\mathbf{X} = \mathbf{E}_{oH}$ [**ID** + $\mathbf{S}_{oH}$(ID) + **V**]. **X** is then signed by the remote poll-site officer to get $\mathbf{S}_{oR}$(X). **X** + $\mathbf{S}_{oR}$(X) is then sent to the home poll-site's voting machine through the network host (along a private network, with redundant connectivity).

ii) The home poll-site (*local*) officer verifies the signature and then decrypts the message using his private key. Next, the vote, '**V**', is signed by the local officer's private key and by the private keys of many local observers to get $\mathbf{S}_{oH}(V) + \mathbf{S}_{p1H}(V) + \mathbf{S}_{p2H}(V) + \mathbf{S}_{p3H}(V)_{\ldots}$.

iii) A copy of this signed vote is locally recorded on a memory stick, which is stored internally to form a part of the (anonymous) permanent physical audit-trail at the home poll-site.

iv) Another copy of the signed vote is sent to the local vote tallier(s), which then take this vote into account for tallying, as described in section 4.4.1.

v) The vote - ‘**V**’, along with all the signatures are encrypted using the local officer’s public key to get $\mathbf{E}_{oH}\{V + \ldots\}$.

vi) Now, a (unique) random number ‘**N**’ is generated, and the above $\mathbf{E}_{oH}\{V + \ldots\}$ is further encrypted using ‘**N**’ as the symmetric key, to get $\mathbf{A}_N\ [\mathbf{E}_{oH}\{V + \ldots\}]$. Also random numbers, n_1, n_2, n_3, … are generated for all other candidates.

vii) The encrypted vote, the identification information, along with the various signatures, and all of the (n_i, v_i) pairs – of which (N, V) is the first – are then encrypted with the remote officer’s public key to get $\mathbf{Y} = \mathbf{E}_{oR}\ [\mathbf{ID} + \mathbf{S}_{oH}(ID) + \mathbf{A}_N\ [\mathbf{E}_{oH}\{V + \mathbf{S}_{oH}(V) + \mathbf{S}_{p1H}(V) + \mathbf{S}_{p2H}(V) + \mathbf{S}_{p3H}(V)_{\ldots}\}] + \{(N, V), (n_1, v_1), (n_2, v_2), \ldots\}]$. This encrypted message is then signed by the local officer. $\mathbf{Y} + \mathbf{S}_{oH}(Y)$ is then sent to the remote poll site’s voting machine through the network host.

viii) This is then decrypted at the remote poll-site (after verifying the signature) by using the officer’s private key. The “$\mathbf{ID} + \mathbf{S}_{oH}(ID) + \mathbf{A}_N\ [\mathbf{E}_{oH}\{V + \mathbf{S}_{oH}(V) + \mathbf{S}_{p1H}(V) + \mathbf{S}_{p2H}(V) + \mathbf{S}_{p3H}(V)_{\ldots}\}]$” so obtained, is stored onto the voter’s memory stick and permanently sealed before the memory stick is finally handed over to the voter.

ix) A receipt print-out corresponding to each of the (n_i, v_i) pairs (received in the message) is issued to the voter in that order. The first receipt, along with the memory stick handed over in step (viii) above, form the RECEIPT for his vote cast.

Analysis:

i) Since all the phases of the voting process (from obtaining the ballot to vote casting, audit trail and vote tallying) are performed virtually as though the voter is voting from his home poll-site, duplicates can be detected at the home poll-site. Thus, the voter *cannot* vote at multiple remote poll-sites.

ii) Complete voter privacy and anonymity is ensured as the vote signing, random number generation, vote storage, audit trail formation, are all performed at his home poll-site and there is no way to tell where the voter actually ‘cast’ his vote.

iii) Since the messages across the network are encrypted in a manner proven to be cryptographically secure, voter confidentiality is maintained.

iv) Since messages are authenticated, bogus requests are not entertained.

v) Since ID is carried with each message, multiple requests for the same ID can be dishonored. This prevents replay attacks.

vi) Since the information is transmitted over a private network, possibilities of attack are limited.

vii) Since redundant connectivity is maintained, availability is sustained.

viii) Since Network connectivity is provided by the network host, the voting machine, as such, can be kept simple.

4.5. Use of Memory Sticks

The memory stick used in the whole of our design is assumed to be just a simple medium to store the encrypted vote reliably. It does not need to have any form of processing power or any “smart” features. Since the plaintext of the vote can be encrypted in multiple-byte blocks, a 10K memory stick (costing around 20 cents according to [2]) would be more than sufficient to store the encrypted vote. We feel that this option is thus very much feasible. Also, we note that the Caltech/MIT report [2] says:

"We imagine that the election office purchases frogs in bulk in blank, uninitialized form. Thus, frogs may be considerably cheaper than printed paper, or optical-scanned ballots. A blank frog may be a blank piece of paper, a blank memory card costing 20 cents or less, or some other medium with suitable properties. We expect that some form of electronic memory will eventually be the favored representation of a frog."

Lastly, if there is some other cheaper medium, that can hold the encrypted vote reliably, it is welcome to be used.

4.6. Publishing (n, v)

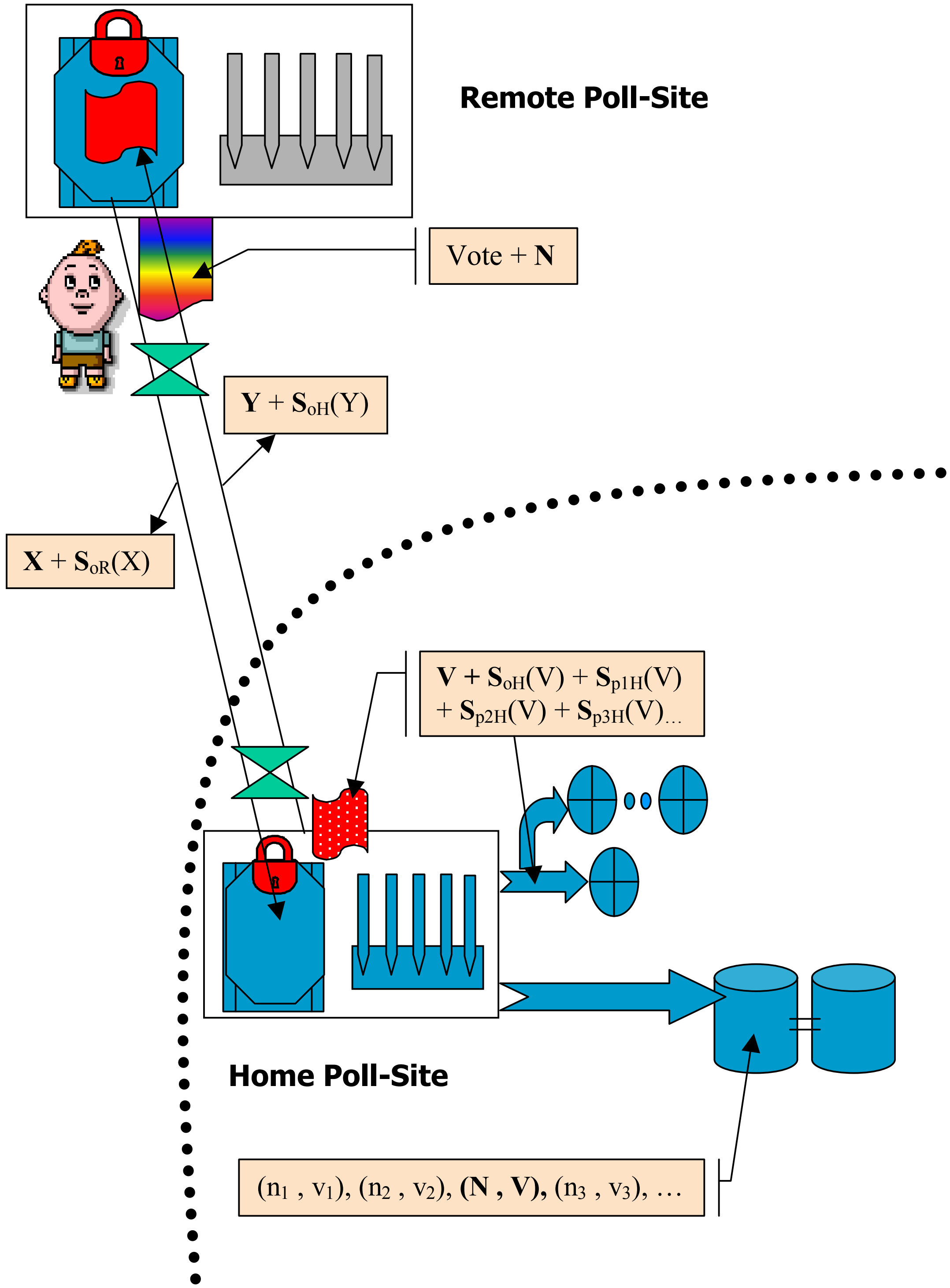
After the voting period ends, at some pre-announced point of time, all the (n, v) pairs are published on the Internet and at kiosks – placed at strategic locations such as public offices and shopping malls – maintained by the election office. The actual information published is nothing but the pairs of (Number, Name of Candidate Voted for). Here, the pair may correspond to a legitimate vote that was taken into account for tallying, or to a fake vote that is just present to prevent voter coercion and vote selling. The voter then checks the published information to see if the vote corresponding to the number on the legitimate receipt issued to him is correct. If so, the voter is sure that his vote has been counted as he had cast it. If not, the voter can decide to contend his vote using the memory stick.

Analysis: Since the (n, v) database is being published both on the Internet and at strategically placed, election-office-controlled kiosks, equality of access to all voters is ensured (as the system does not assume Internet access to be available to all the voters). Also, since the fake vote pairs are also published, again, voter coercion and vote-selling are prevented. Finally, since the information is published on multiple replicated servers, it alleviates the problem caused by denial of service (DOS) attacks.

4.7. Vote Contention

If the voter decides to contend his vote, he takes his memory stick and goes to the nearest election office to do so. Here, he is first authenticated with respect to the memory stick that he is carrying, by checking if the identification information is signed properly on the memory stick, and next verifying if that information is the same as the identification information presented by the voter to the officer.

Once authenticated, the voter then goes to the vote contending machine (which is again placed such that the voter’s privacy is ensured) and inserts his memory stick. This is also a thoroughly verified and certified machine. In fact, the same voting machines can be reused in a different functional mode. The machine compares the vote on the memory stick (after decrypting it with the number ‘N’ entered by the user, and then decrypting this with the officer’s private key, and finally verifying the signatures) with the published vote. If different, a complaint is lodged with the commissioner or some other higher authority that there was an inconsistency in the vote tally and thus, there may be a need to order a recount of the votes (using the audit-trail) in order to get the correct, accurate picture of the election results.


Remote Poll-Site
Vote + **N**
Y + S_{oH}(Y)
X + S_{oR}(X)
V + S_{oH}(V) + S_{p1H}(V) + S_{p2H}(V) + S_{p3H}(V)...
Home Poll-Site
(n_1 , v_1), (n_2 , v_2), **(N , V),** (n_3 , v_3), …

Legend:

$\mathbf{X} = \mathbf{E}_{oH}\ [\mathbf{ID} + \mathbf{S}_{oH}(\mathrm{ID}) + \mathbf{V}]$

$\mathbf{Y} = \mathbf{E}_{oR}\ [\mathbf{ID} + \mathbf{S}_{oH}(\mathrm{ID}) + \mathbf{A}_{N}\ [\mathbf{E}_{oH}\{V + \mathbf{S}_{oH}(V) + \mathbf{S}_{p1H}(V) + \mathbf{S}_{p2H}(V) + \mathbf{S}_{p3H}(V)_{\ldots}\}] + \{(N, V), (n_1, v_1), (n_2, v_2), \ldots\}]$

Figure 4.6: Remote Poll-Site Voting

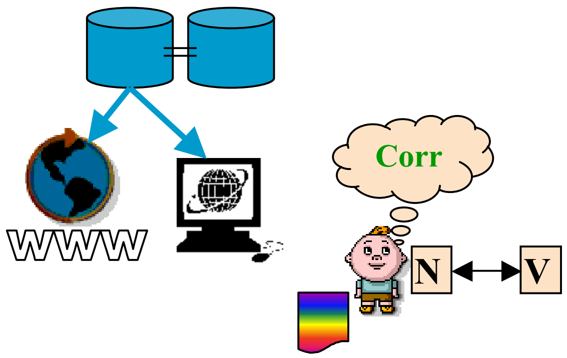
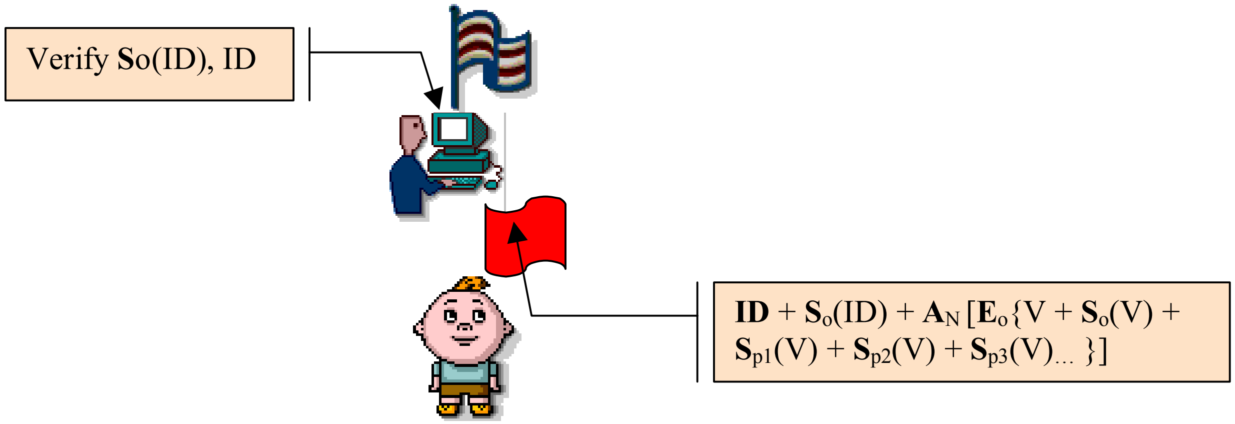
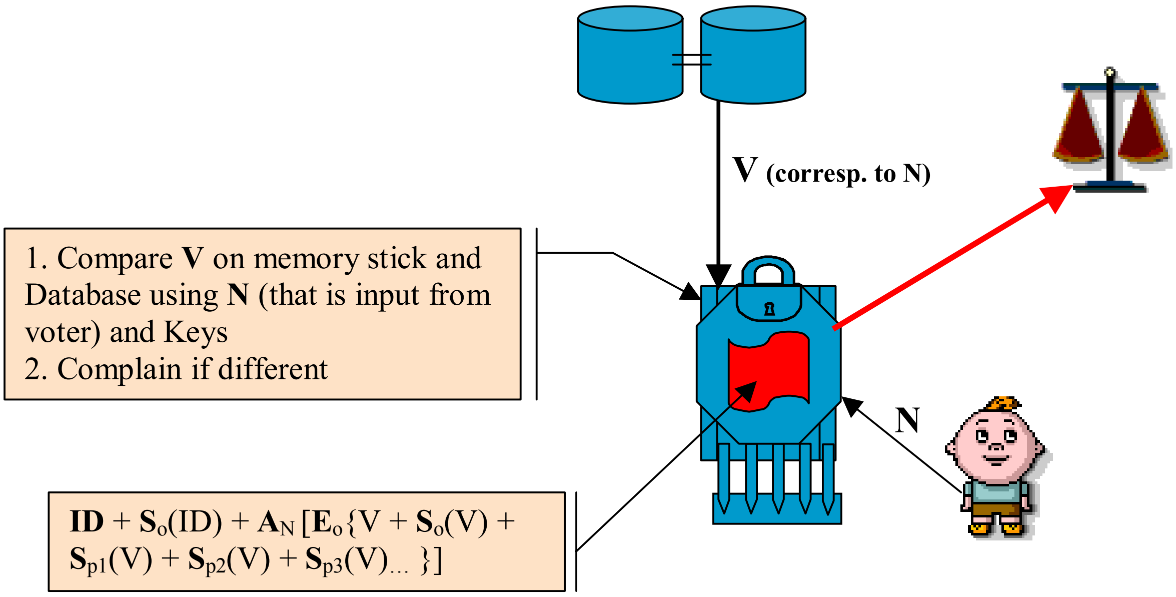


Figure 4.7: Vote Tally Verification



Verify So(ID), ID
$\mathbf{ID} + \mathbf{S}_o(\mathrm{ID}) + \mathbf{A}_N\,[\mathbf{E}_o\{V + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)\ldots\}]$

Figure 4.8: Vote Contention - Part 1



V (corresp. to N)
1. Compare **V** on memory stick and Database using **N** (that is input from voter) and Keys
2. Complain if different
N
$\mathbf{ID} + \mathbf{S}_o(\mathrm{ID}) + \mathbf{A}_N\,[\mathbf{E}_o\{V + \mathbf{S}_o(V) + \mathbf{S}_{p1}(V) + \mathbf{S}_{p2}(V) + \mathbf{S}_{p3}(V)\ldots\}]$

Figure 4.9: Vote Contention - Part 2

In the case of remote poll-site voting, the authentication phase would be the same. However, during the actual contention phase, since the vote present on the memory stick can be decrypted only by using the home-poll-site election officer's key, the encrypted vote, $\mathbf{A}_N$ [$\mathbf{E}_o$ {V + … }], along with the number '**N**' entered by the voter, are sent to the voter's home-poll-site's voting machine, after being signed by this machine, and encrypted using the destination's public key. At the home-poll-site, the message is decrypted; the signature of the source is then verified. The encrypted vote is now decrypted using its private key, and then the various signatures are verified. The vote is then compared with the published vote. Finally, a message about the result of the contention is sent to the source machine, which then displays it to the voter.

Analysis: Since the vote contention is performed virtually as though he were doing it from his home-poll-site, he can contend at *any* poll-site. Also, since the comparison, and complaint lodging are performed with respect to 'N', anonymous contention is ensured.

5. Conclusion

In this paper, we have presented a novel framework for an electronic remote poll-site voting system. The key issue addressed by our work has been the issue of receipt to the voter to acknowledge his vote, which facilitates vote confirmation and vote contention, and at the same time preventing any occurrence of vote-selling or voter coercion. Even though much of the present literature views receipt-freeness as the *necessity* for precluding vote-selling and voter coercion, we have overcome the tradeoff and thus ensured the voter's confidence in that his vote has been counted as cast, without compromising voter security. Also, our design successfully meets *all* of the identified requirements.

Implementation of a prototype of our design suggested that the voting process was not phenomenally different from the point of view of the voter, except for the enormous convenience benefits and the new facilitation of the issue of receipts. The voter need not go through a training process in order to use the proposed voting system. Thus, our system is backward-compatible with the systems in place as of now, which facilitates a smoother transition.

6. Acknowledgements

We thank Dr. Aviel D. Rubin, Associate Professor at the Department of Computer Science and Technical Director of Information Security Insititute at The Johns Hopkins University, for his help and valuable feedback.

7. References

[1] "*Electronic Voting*," Encyclopedia of Computers and Computer History, prepared by Lorrie Faith Cranor and edited by Raul Rojas, published by Fitzroy Dearborn, 2001.

[2] "*Voting – What is, What Could be*," Caltech/MIT Voting Technology Project (VTP) Report, July 2001.

[3] "*A Modular Voting Architecture ("Frogs")*," Shuki Bruck, David Jefferson, and Ronald L. Rivest, August 2001.

[4] "*Comments of Professor Ronald L. Rivest*", Caltech/MIT VTP Press Conference, July 16, 2001, http://theory.lcs.mit.edu/~rivest/publications.html.

[5] "*Testimony given before the U.S. House Committee on Administration*", Ronald L. Rivest, May 24, 2001, http://theory.lcs.mit.edu/~rivest/publications.html.

[6] "*Electronic Voting*," Ronald L. Rivest, Technical Report, Laboratory for Computer Science, Massachusetts Institute of Technology.

[7] "*Report of the National Workshop on Internet Voting: Issues and Research Agendas*," Internet Policy Institute, Sponsored by the National Science Foundation, Conducted in cooperation with the University of Maryland and hosted by the Freedom Forum, March 2001.

[8] "*A Report on the Feasibility of Internet Voting,*" California Internet Voting Task Force, January 2000.

[9] "*e-Voting Security Study*," E-Democracy Consultation, U. K. Cabinet Office, http://www.edemocracy.gov.uk/library/papers/study.pdf.

[10] "*A Practical Secret Voting Scheme for Large Scale Elections*," A. Fujioka, T. Okamoto, and K. Ohta, Advances in Cyptology - AUSCRYPT '92.

[11] "*Sensus: A Security-Conscious Electronic Polling System for the Internet*," Lorrie F. Cranor and Ron K. Cytron, Proceedings of the Hawai`i International Conference on System Sciences, January 7-10, 1997, Wailea, Hawai`i, USA.

[12] "*Secure Electronic Voting Over the World Wide Web*," Master's Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, May 1999.

[13] "*Multiple Administrators for Electronic Voting*," Bachelor's Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, May 1999.

[14] "*An Untraceable, Universally Verifiable Voting Scheme*," Professor Philip Klein, Seminar in Cryptology, December 12, 1995.

[15] http://www.e-poll-project.net/

[16] http://www.free-project.org/

[17] "*Secure Voting Using Disconnected, Distributed Polling Devices*," David Clausen, Daryl Puryear and Adrian Rodriguez, Department of Computer Science, Stanford University.

[18] "*Security Criteria for Electronic Voting*," Peter G. Neumann, 16th National Computer Security Conference, Baltimore, Maryland, September 20-23, 1993.

[19] "*Security Considerations for Remote Electronic Voting*," Aviel D. Rubin, Communications of the ACM, Vol. 45, No. 12, December 2002.

Appendix A: Why require even the Vote Generation Machine to be certified/verified?

The AMVA (A Modular Voting Architecture) framework presented in the Caltech/MIT Voting Technology Project [2] suggests that the vote generation machine can be proprietary and only the vote casting machine must be open-source and thoroughly verified and certified for correctness and security. However, we think this may *not* be a reasonable proposition. This is because it can lead to serious privacy breaches if the election official can collude with the producer of vote generation machines. The attack would proceed as follows:

> Each time the officer authenticates a voter, he records the time against his identification (He knows the identification information as he is the one who authenticated the voter). He also notes which vote generation machine the voter went on to vote. The vote generation machine being non-certified and hence – presumably under the maximally hostile assumption – malicious, will record the choices of each voter against a time vector. The officer can then correlate these pieces of information to obtain the votes of almost all voters!

The election officers *cannot* be trusted as they usually are volunteers or temporary workers paid by the hour. Also, we cannot say "Let there be many observers from different political parties so that they do not collude amongst themselves." While this is true in case of vote tallying, where they have no reason to and will not collude in interest of their individual good, this is not the case here. All parties may collude to find "who voted for whom?". Moreover, all voting machines cannot be masked off from the officers and the security because of security implications "behind the screen".

Above all, the voter's privacy is of utmost concern. If a vote tally is incorrect, it can be resolved by a recount, or in the worst case, by re-election. However, if the voter's choices go public, he has lost his privacy – for life! There may be serious criminal consequences if this ever happens. This arena is too sensitive to allow non-certified devices at *any* point. Therefore, we hold that even the vote generation machine *shall be verified and certified*.