ABSTRACT

Title of thesis: AN ANALYSIS OF CHAUM'S VOTER-VERIFIABLE ELECTION SCHEME

Julie Ann Staub, Master of Science, 2005

Thesis directed by: Professor Jonathan Katz
Department of Applied Mathematics

Chaum's Voter-Verifiable election scheme introduces a new direction for electronic voting. The scheme eliminates the need to trust any machinery or authority, and instead relies on mathematical proof to certify the trustworthiness of an election. Audits at every stage of the election create transparency that should restore voter confidence in the election process. We survey and categorize the field of electronic voting, and place Chaum's scheme within this context. We then define a framework of formal requirements of a voting system. We present Chaum's scheme itself, and give an analysis. Based on our technical analysis, we find the scheme to be secure. However, after considering other implementation concerns, we recognize various minor obstacles limiting its widespread adoption in today's elections. Despite this, we believe that the substance of the scheme is promising and maybe an improved, simpler variant might better suit future elections.

AN ANALYSIS OF CHAUM'S VOTER-VERIFIABLE ELECTION SCHEME

by

Julie Ann Staub

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2005

Advisory Committee:

Professor Jonathan Katz, Chair/Advisor
Professor Benjamin Bederson
Professor Lawrence Washington

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction to Electronic Voting

1.1 Motivation

Free and fair governmental elections are of critical importance in a democracy. Since they are very high-profile, elections have long been a target of fraud and corruption [55, 63]. And because the national constituency is so large, yet elections so infrequent, there is a high likelihood of error. Therefore, it is crucial to have a trustworthy election scheme that delivers accurate results amidst errors and malicious attacks.

Electronic voting with precinct-based electronic machines has been suggested as an improvement over the way we currently conduct elections. Proponents argue that the major benefits include increased efficiency, convenience, and voter anonymity, in conjunction with decreased long-term cost and labor [16, 45]. Plus, an electronic user interface can alert the voter of and prevent under- or over-voting, as well as facilitate voting for those with various disabilities.

Critics, however, suggest that there are problems inherent in the idea of using electronic devices to aid in elections [41, 45]. The major complaint is that there is no true way of controlling, tracking, or auditing the behavior of an electronic machine. The machines act like black boxes. Voters currently have no way of ensuring that their vote has been recorded correctly or counted, and the public has no way of

verifying that the tally was computed correctly. Without having a way of verifying an election's validity, the results cannot be fully trusted. Tampering with either the machine or the votes is not completely detectable. In addition, most electronic voting machines today do not even have a mechanism that provides a meaningful recount; this fact forces us to place trust in the machine itself.

Another reality of electronic voting machines is that large-scale fraud could be accomplished more easily, and with significantly less effort in comparison to other voting methods. Manipulation of electronic data or machinery can potentially be done remotely and efficiently, and could affect more votes than an attack on physical ballots or machinery at a polling station. Furthermore, an attack on an electronic voting machine has a higher chance of going completely undetected, due to the lack of an auditing mechanism.

Many have suggested equipping the existing electronic voting machines with printers, to produce a receipt of each ballot. These receipts would be retained by the polling station. Although this does provide a way to conduct a recount, thereby creating an auditing mechanism, these paper audit trails "provide, at best, a short term fix to a fundamentally flawed approach" [91]. Relying on manual recounting somewhat undermines the advantages of an electronic system.

The real motivation behind the use of paper receipts is that of verifiability: with physical receipts the tally can be audited, and therefore no trust needs to be placed in the electronic machines themselves. This quality is of crucial importance to the integrity of an election. The correctness of a software program, and therefore the behavior of an electronic machine, is known to not be provable. If an electronic

voting scheme is designed to eliminate the necessity for trust in the machines, then it potentially can administer a fully auditable electronic election. We claim that a scheme introduced in 2002 by David Chaum [19] accomplishes such a feat in a novel, elegant manner.

With the use of transparency in all stages of Chaum's Voter-Verifiable election scheme, it is possible to audit each step, detecting error or fraud with all but *negligible* probability (described in Section 1.3). The advantage of having these verification procedures is that any post-election question about its legitimacy is answered in the form of proof. If the election is *proven* to be trustworthy, then there is no possibility for controversy. There is also no need to perform a recounting of ballots, which is still integral to our current election practices.

The central goal of the scheme is that of acceptance and trust by all, especially the voters, thereby restoring confidence in the quality of our elections and the value of each vote. It provides sufficient transparency to *verify the results* of an election, rather than needing to verify the election procedure or equipment. However, by eliminating the need to trust anything physical, we instead will need to trust certain properties of mathematics and cryptography, and be satisfied with *probabilistic* integrity of the final tally; rather than assuring absolute correctness of the tally, the scheme can only assure its accuracy with overwhelming probability.

One major advantage of Chaum's scheme is that it has been made available for public scrutiny. Experts and all other interested parties are in fact encouraged to evaluate and criticize the scheme. The intent is to expose any flaws or weaknesses, and subsequently work towards improving the scheme. This is in contrast with

the trend of most other poll station electronic voting systems, whose proprietors have claimed that it is necessary to keep the details secret for purposes of securing intellectual property. However, the existence of open source systems is exactly what many experts believe lead us to better solutions in technology. In fact, Chaum's work has already inspired many others to conduct related research, some of which we mention in Section 4.4.

1.2 Definitions

We define an election by its purpose: an *election* is a process to obtain accurate data representing a set of participants' answers to a posed question. A *vote* is what physically represents a participant's answer to a particular question. A vote consists of a selection, generally from a predetermined set of answers, called *candidates*. Sometimes a vote contains a selection which is not an element of the predetermined list, and is called a *write-in* vote. One or more votes are combined into a structure called a *ballot*. An eligible, authenticated participant in an election is called a *voter*. We call each question in an election a *race*, and therefore each race has a set of candidates, potentially receiving votes from voters.

A *voting scheme* is a protocol which has a means of receiving votes as input, and produces an output which is a tally of the votes cast. Therefore, it is a method for conducting an election. The tally may result in a *decision*. The decision can, for example, be the assignment of an individual to a public office, or the institution of a referendum. In the event of a referendum vote, the set of candidates would consist

of "yes" or "no".

A voting scheme, as defined above, can refer to any method that can successfully manage an election. The voting schemes that have been used historically are called *traditional* voting schemes. Some examples of these schemes are those which use ordinary paper ballots, lever systems, or punch-card ballots. In contrast, an *electronic* voting scheme, or *e-voting* scheme is one that makes use of electronic devices to conduct an election. Votes are recorded electronically and possibly tallied electronically as well. Electronic voting schemes can be further classified into *remote* voting schemes or *poll station* voting schemes.

The term *remote* electronic voting refers to the subset of electronic voting schemes which assume that voters are connected remotely through an electronic network. All setup, communications, and computations are done electronically, with computers acting as proxies for all voters and other players in the election. A further subset of remote e-voting schemes are those that utilize the internet to conduct an election. These schemes are referred to as *internet* voting schemes or *i-voting* schemes. In contrast to remote voting, *poll station* electronic voting schemes require that voters cast their votes from electronic voting machines at physical, central locations. The machines record the votes electronically. It is also possible for the machines to count, transmit, or tally the votes electronically as well. The distinction between remote voting and poll station voting protocols is central to this paper, and will be further discussed in Chapter 2.

A *Direct Recording Electronic machine*, or *DRE-machine*, refers to an electronic device whose purpose is to record votes. Voters' selections are recorded di-

rectly into an electronic machine. The DRE-machine may also be able to tally the votes. We distinguish a DRE-machine from other electronic machines that merely tally, such as the devices used to count punch-card ballots. A subset of these DRE-machines are those which employ touch-screen technology as the interface to the voter. These touch-screen DRE-machines will be the focus of our discussions on remote e-voting and therefore will be subsequently referred to simply as DRE-machines.

Figure 1.1 shows the hierarchy of the voting schemes just discussed.

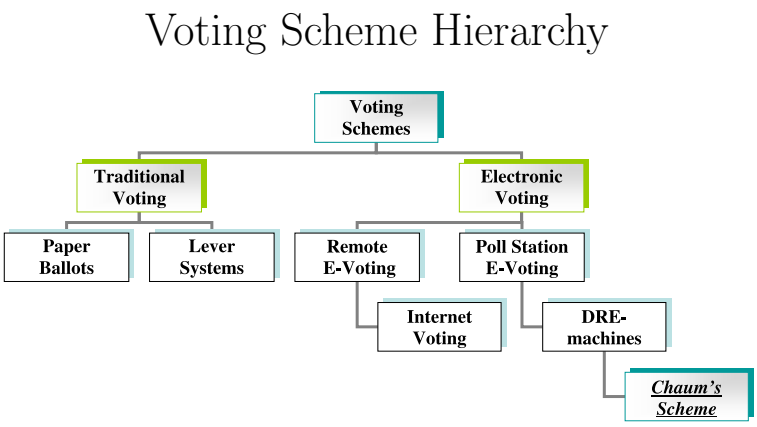


Figure 1.1: The categorization of voting schemes.

Ideally, a voting scheme should create an accurate mapping from voter intent to tallied vote. The steps in an election between each voter's decision on the posed question and the announcement of the tally are called *translation* steps. The more complicated the election process, and therefore the more translation steps, the more places there are for breaches of security. These include malicious fraud or corruption as well as non-malicious errors. As a result, the more simple and transparent a voting scheme is, the easier error detection is [96].

In this paper, we choose to focus on *governmental* elections which are large-scale and binding. Certainly, other election types exist and are interesting to study. Other such types include private shareholder elections and university elections, as well as polls and surveys. Our focus is a result of the recent controversy in the United States over the validity of the DRE-based electronic voting systems in the 2000 and 2004 general elections. We choose to follow Chaum's lead by researching and analyzing ways to improve such elections.

Voting Styles

In an election, the *voting style* mandates the number of candidate selections that constitute a vote. There are numerous different types of voting styles [88]. Here we will distinguish between those which are applicable to our discussion.

- 1-out-of-2 voting (yes/no voting) - There are only two candidates, typically yes or no. The vote is $v = i$, where $i \in \{0, 1\}$

- 1-out-of-L voting - There are L candidates, and the voter selects exactly one of them. The vote is $v = i$, where $i \in \{1, \ldots, L\}$.

- k-out-of-L voting - There are L candidates, and the voter may choose a subset of size at most k of those L. The vote is $v =< v_1, v_2, \ldots, v_m >$, where $m \leq k \leq L$ and for $1 \leq i, j \leq m$, $i \neq j$, $v_i \in \{1, \ldots, L\}$ and $v_i \neq v_j$.

- Preferential voting - There are L candidates, and the candidates are numerically ranked according to the voter's preference. Thus, the vote is

$v =< v_1, v_2, \ldots, v_L >$, where for $1 \leq i, j \leq L$, $i \neq j$, $v_i \in \{1, \ldots, L\}$ and $v_i \neq v_j$. The order of the v_i is such that the voter prefers v_i to v_j for $i \leq j$.

- Write-in voting - The voter inputs a vote v $\notin \{1, \ldots, L\}$. The vote v is stored as a string of letters, representing the name of an individual, for example.

For any given *race*, the first four voting style options are disjoint, yet the last option, write-in voting, may be integrated into any one of the first four. In an election, there may be multiple races on the ballot, and voting styles may vary from race to race.

The most flexible of voting schemes would support any of the above voting styles. We will see however, that not all proposed schemes can handle write-in voting, and some can only handle 1-out-of-2 voting. Furthermore, not all schemes are reasonably efficient at all types. This provides us with a criterion that distinguishes the effectiveness of voting schemes.

Communication Channels

Every electronic voting scheme relies on some type of communication channel(s) between the voters and other players in the election. These channels can either be realized through cryptographic or physical means, depending on the circumstance. The following definitions of types of communication channels have been modified from [11] and [88].

- Anonymous channel - a communication channel between two parties where the sender of a message remains anonymous.

- Untappable channel - a physically secure one-way communication channel for which a third party is unable to read or alter a message from the sender. Furthermore, neither the sender nor receiver is able to prove the content of the message to a third party.

- Untappable anonymous channel - a channel that both maintains the anonymity of the sender and renders any interception or alteration of a message impossible. It is physically secure, and the content of a message cannot be proved to a third party.

- Voting booth - a two-way untappable, anonymous communication channel. It is the theoretical equivalent of a physical booth. The communications between the two parties cannot be read or altered by a third party, and the sender remains anonymous.

- Public bulletin board - a public web site, or physical equivalent. Every legitimate player has permission to "read" messages from and "append" messages to the bulletin board. However, no party is able to modify or delete any information.

1.3 Requirements

Every voting scheme must satisfy certain requirements in order to be trusted in a real election. Clearly, we will require some notion of security. We can define security rigorously by describing a set of security properties. Thus, the security of

a particular scheme is measured by the degree to which it adheres to these defined requirements. Beyond security, there are a number of other criteria that distinguish voting schemes from each other.

The two foremost and crucial requirements of the security of modern-day voting schemes are accuracy and anonymity. The difficulty of the electronic voting problem, however, is a result of the conflicting nature of these two properties. Chaum has conjectured that it is impossible to achieve unconditional anonymity and accuracy simultaneously [19]. Therefore, a relaxation is necessary for at least one of these.

Experts disagree on the exact prioritized order of the requirements of voting schemes. There is a set of criteria that is considered absolutely necessary for an implementable scheme to achieve, while the remaining requirements are viewed as highly desirable. We, like [13, 17, 61, 68], categorize the requirements into two sets: basic and extended. The schemes that satisfy the basic requirements will thereby satisfy our definition of a *secure* electronic voting scheme. Those requirements which are deemed highly desirable are met by each scheme with differing degrees. Thus, we can evaluate and compare schemes based on how well they achieve these requirements.

Basic Requirements

1. **Accuracy** or **Integrity**

   In an election, the tally should correctly reflect the total number of legitimate votes cast for each candidate. Any votes that are falsely created or

modified should not be counted in the tally.

We note that perfect accuracy is not reasonably possible for an election of significant scale [16]. Furthermore, since unconditional accuracy and anonymity are competing properties, we allow for a probabilistic assurance of accuracy. We thus define a scheme to be accurate if the probability of undetected integrity fraud for n votes is *negligible*, that is, it decreases exponentially in n.

2. **Anonymity** or **Ballot Secrecy**

A scheme preserves anonymity if the probability of recreating the mapping from any voter to her vote is non-negligibly better than a random guess.

In practice, it is acceptable to reveal partial results of an election, which compromises the anonymity of sets of voters. For example, it is reasonably permissible to reveal tallies according to precinct in national elections. It is also impossible to maintain anonymity in the case that every voter selects the same candidate.

Beyond providing privacy on behalf of the voter, an anonymous scheme also prevents against certain fraud. If, after the election, a voter can prove to a third party that she voted a certain way, her vote would be susceptible to vote-buying and/or coercion. Benaloh coined the term *receipt-freeness* to describe this property [11]. A scheme is receipt-free if it is impossible to reconstruct a provable receipt of a vote *outside* the voting booth. It is important to emphasize that this property does not prevent the creation of a receipt *in the*

voting booth, as long as it is not readable outside the booth. Instead, it does not allow a voter to possess or create proof of her vote *after* the election.

It is worth noting that it is impossible to recover from a compromise in anonymity [16]. Since it is a matter of information flow, once a vote's identity has been revealed, anonymity cannot be regained. However, since anonymity can only be compromised after a vote is cast, the tally remains unchanged, and thus integrity is maintained. Therefore, compromises in anonymity will not necessarily invalidate a particular election.

3. **Universal Verifiability** or **Auditability**

   Any interested party should be able to independently verify the integrity of the election tally. This should include verification along each translation step of the election. Although it is important for an individual to have the power to verify the correctness of her vote (known as *voter-verifiability*), the verification becomes *universal* only when each individual has the power to verify the correctness of every legitimate vote cast in an election.

Extended Requirements

The following *desirable* qualities focus more on the reliability, practicality and acceptability of a voting scheme. They are not typically emphasized in the technical literature, yet play a vital role in the implementation of theoretical schemes. We will revisit these requirements, as well as the basic requirements, in Section 3.4 when we analyze Chaum's scheme in this context.

1. **Robustness**

   The voting scheme should have the capacity to tolerate *partial* failure or fraud at any stage of the translation process. The scheme should be able to continue the election without termination and produce an accurate tally regardless of these failures. Furthermore, when error or fraud is detected, the voting scheme should have a mechanism or procedure in place to contain and, ideally, correct the error. Note that it may be impossible to isolate the exact votes which were corrupted without sacrificing anonymity.

   From the dependability perspective [16], absolute tolerance of faults is not feasible. Instead, we can measure the robustness of a scheme based on how well it handles various failures. Also, timely determination of the tally is crucial to the confidence placed in a given election. Therefore, it is more desirable to have a real-time recovery mechanism in place rather than after-the-fact recovery mechanisms like manual recounts.

2. **Efficiency**

   A voting scheme should produce an accurate tally in a timely fashion after the close of an election, regardless of any error or fraud.

3. **Flexibility**

   A voting scheme should be able to adapt smoothly to a new election, new ballot, or new precinct. Schemes that support all voting styles are the most flexible.

4. **Certifiability**

A voting scheme must provide a way to be evaluated and tested by *Independent Testing Authorities*, or *ITA*s. A scheme should not be declared certified before undergoing and passing a strict evaluation of its adherence to a predetermined set of security requirements. If a proprietor insists on keeping the details of the scheme closed-source to the public, the minimal requirement should be that all is revealed to the ITAs.

5. **Scalability**

The size of an election should not affect the security properties of the scheme. The most scalable scheme should securely and *efficiently* handle any number of votes, where the meaning of *efficiently* depends on the requirements of the particular election.

6. **Usability**

A voting scheme should be easy for people to use. It should be convenient, quick, and simple. It should not require a voter to learn any new skills in the voting booth, perform complex tasks, nor be involved in too many phases of the process. The less amount of work required on behalf of the voter, the better, as long as trust in the accuracy of the tally is maintained.

7. **Accessibility**

A voting scheme must be made accessible to all eligible voters. This set includes the elderly, the disabled, the computer illiterate, minorities, those

from differing socioeconomic backgrounds, as well as non-native English language speakers.

1.4 Outline of this Thesis

We provide an overview of various types of electronic voting schemes in Chapter 2. In Chapter 3 we introduce Chaum's Voter-Verifiable election scheme and follow an example vote through the election process. We present the scheme in the context of the formal definitions, and analyze it based on the requirements defined. Chapter 4 discusses the non-technical obstacles that impede the implementation of Chaum's scheme in real elections.

Chapter 2

Overview of Electronic Voting Schemes

Beginning in the early eighties, a large number of technical papers have appeared in the field of electronic voting. A comprehensive list of proposed theoretical electronic voting schemes is presented in [4], [47], or [61]. Most of the research in cryptographic e-voting protocols follow a remote voting model. That is, the research has focused on schemes that rely entirely on an electronic network to conduct the election. However, very few of these schemes are known to have been implemented. Furthermore, it appears that none have been implemented in a real, binding election of significant scale.

One major concern with remote voting that does not appear to be solvable is that of vote-buying and voter coercion. Without physical security to assure privacy of the voter at the time of vote casting, anonymity cannot be guaranteed. You can imagine a situation where a coercer or vote-buyer is physically present during the vote casting stage, or even impersonating another voter, thus controlling that vote. Note that this is a problem with any remote voting protocol, not just an electronic one; any form of absentee voting, for example, poses this same problem.

There are other limitations of remote electronic voting, including the security of the network platform. These obstacles are discussed in Section 2.1.6 in the context of internet voting, yet are applicable to any network. It is for these reasons that

the cryptographic, remote e-voting schemes found in the literature have not been successfully implemented in governmental elections. Still, it is of great importance to study these schemes, as their results have impacted other schemes, including that of Chaum. In this chapter, we will overview both remote and poll station electronic voting schemes.

2.1 Remote E-Voting Schemes

Remote electronic voting can be categorized into schemes which are based on a mix-net, blind signatures, or homomorphic encryption. Each of these general techniques is the basis for multiple schemes in the literature. There is no one technique that is universally better than the others; the strength of each depends on the particular application. Each individual scheme optimizes a different set of extended requirements. Furthermore, each has its own assumptions, both physical and cryptographic. Therefore, the appropriateness of any particular scheme will depend on the assumptions and requirements associated with a particular election.

2.1.1 Schemes based on a Mix-net

The first paper to introduce the idea of a mix-net, as well as an electronic voting protocol altogether, is that of Chaum in 1981 [22]. The goal of a mix-net is to accept a set of inputs and anonymize them via a secret shuffling process, such that the outputs cannot be traced back to their corresponding inputs. This creates an anonymous channel. In an application of a mix-net to electronic voting, the inputs

are encrypted votes, and the outputs are the corresponding plaintext votes. A series of *(Mix) Servers*, or *Trustees*, lie between the inputs and outputs of the mix-net. Each Server partially decrypts each vote in the set with its own private key, then performs a secret shuffle to the set of partially decrypted votes. Then the Server forwards all of the votes to the next Server, who functions in a similar manner, until the last Server in the mix-net has fully decrypted each vote. The result is an untraceable path from input to output. In the context of voting, that means that it is impossible to reconstruct the one-to-one correspondence between voter and vote, thus preserving anonymity.

In any mix-net scheme, it is imperative to verify the actions of the Servers in order to ensure integrity of the decrypted votes. To do this, there must be an auditing process. The Servers must produce proofs of correctness of their computations. Achieving this, while still maintaining anonymity of the votes, is difficult. The process is inherently inefficient, and many attempts have been made to produce methods to increase the efficiency of this step.

Technical papers on the subject can be further categorized into three areas: improvements on mix-nets themselves [1, 2, 3, 50, 51, 52, 53, 74, 76, 78, 104], electronic voting schemes based on mix-nets [14, 39, 42, 64, 72, 82, 93], and attacks on particular mix-nets [68, 69, 84].

There are certainly advantages and disadvantages of schemes based on mix-nets. Accuracy, anonymity, and universal verifiability can all be achieved through this methodology, and therefore these schemes can be considered theoretically secure. One main advantage of mix-net based schemes is that they can support write-

in votes, which enhances their flexibility and distinguishes them from most other schemes. The major disadvantage of these schemes is their efficiency. The tallying process is considered to be extremely slow in most mix-net schemes. Otherwise, the efficiency bottleneck is in the proofs of correctness of the Trustees' work during the mixing phase. In either case, e-voting schemes based on mix-nets are relatively inefficient in computation, as well as communication. Evaluation based on the remainder of the extended requirements can only be done on a case-by-case basis. This evaluation would need to be conducted in the context of an implementation model.

2.1.2 Schemes based on Blind Signatures

E-voting schemes based on blind signatures are closely related to those based on a mix-net. Modeled after Chaum's paper [23] in 1983, the basic protocol consists of four main phases [79]: registration/authorization, voting, claiming, and tallying. During the registration stage, the *Administrator* issues a blind signature to each voter if presented with a commitment on a valid vote. For the voting stage, the voter sends her *unblinded* vote along with the Administrator's signature to the public bulletin board via an anonymous channel. The voter can verify that her vote appears on the board during the claiming phase and publicly dispute it if not. Finally, the votes found on the board are verified and tallied in the tallying stage. The contents of the bulletin board are universally verifiable, therefore so is the tally.

This class of schemes does support write-in votes, as well as all other voting styles. However, they are considered less practical because it is either the case that

trust must be in placed in the Administrator, or an anonymous channel is assumed. These schemes are simple, efficient, and flexible, but cannot provide receipt-freeness; the voter's blinding factor can be used to prove how she voted. Thus, verifying the voters' blinding factors would compromise the anonymity requirement, resulting in the impossibility of universal verification.

For publications of such schemes, see [21, 23, 38, 77, 79, 80, 94]. Also, two implementations of a blind signature based e-voting scheme are discussed in Section 2.1.5.

2.1.3 Schemes based on Homomorphic Encryption

There are several schemes that are based on homomorphic encryption. The foremost disadvantage of these schemes is that they do not support write-in votes. On the other hand, these schemes perform considerably faster than other types, mostly due to the speed in the tallying phase. Indeed, these schemes are the best choice for yes/no elections.

Fundamental to these schemes is the homomorphic property:

$$E(m_1) + E(m_2) = E(m_1 + m_2),$$

where E represents encryption and m_1, m_2 are messages [37]. The $E(m_1) + E(m_2)$ is a calculation in a group G, whereas $E(m_1 + m_2)$ is a calculation in a group H. The '+' is a group operator corresponding to each group, and may be different for G and H [5].

In words, the property states that the sum of two encrypted messages is equal

to the encryption of their sum [75]. In the context of an election, the messages are the votes, and the encryption scheme is some public key encryption scheme that possesses the homomorphic property. Some known examples are [81], and [13, 25].

The exploitable advantage of the homomorphic property is that the vote tally can be computed and verified without knowing the content of the individual votes. After the election, the encrypted votes are combined into a single, encrypted, quantity. The authorities then decrypt this tally, in the group H. Due to the homomorphic property, this quantity should equal the quantity resulting from the decryption of each of the individual votes in group G. In this way, tallying is done without learning the individual values of the votes. Thus, anonymity is maintained. Since the values of the votes are never revealed, it is necessary to validate the correctness of these votes. For this purpose, non-interactive zero-knowledge proofs are used.

Homomorphic encryption schemes can also be made robust by creating threshold variants, where the decryption is shared between multiple servers. Therefore, if failure occurs for some minority of the servers, termination of the protocol is not necessary.

The application of the homomorphic property to electronic voting is first attributed to Benaloh [13]. The property of receipt-freeness was also first introduced in this paper, but the author's claim that his scheme is receipt-free was later proven incorrect by [47] in 2000.

Some examples of schemes based on homomorphic encryption are [6, 11, 12, 13, 25, 26, 27, 28, 32, 37, 47, 61, 62, 95, 97].

2.1.4 Hybrid Schemes

Each of the three aforementioned remote e-voting techniques have their advantages, yet they also have their limitations. Mix-nets tend to be extremely inefficient, homomorphic encryption lacks the flexibility for supporting write-in votes and preferential voting efficiently, and blind signatures make universal verification impossible without compromising the anonymity requirement [58]. There are a couple of innovative, hybrid-style election schemes that combine techniques from multiple methods. The resulting schemes satisfy certain requirements better than any one technique could.

- A hybrid scheme published by Golle et al. [42] combines homomorphic encryption with a mix-net. The homomorphic property is used to initially produce an efficient, yet unofficial, tally. Because the accuracy of this initial tally is not satisfactorily secure, the backup mode is later deployed in order to produce the official tally. This second, mix-net mode is inefficient, yet extremely accurate. Although universal verifiability is not realized, a modified, yet practical variant called *public verifiability* is. This refers to the property that only the inputs which are well-formed according to the scheme's requirements can be verified by a third-party, as opposed to verifiability for all possible inputs.

- A hybrid mix-net technique based on ElGamal encryption was developed in 2002 by Boneh and Golle [14]. The idea of the resulting scheme, like that of [42], is to be able to announce a quick, unofficial result of an election, then obtain an official, yet inefficient, tally of the votes later. In order to produce

the initial quick result, the mixing technique used can only guarantee accuracy with a *high* probability, but not with *overwhelmingly high* probability. As a tradeoff, the inefficient perfect-correctness proofs are foregone, and replaced with less stringent, "almost entirely correct" proofs. The official results of the election can be thoroughly computed later, using proofs for perfect correctness, without having to mix the votes again. The result is that efficiency is emphasized over perfect accuracy, while still maintaining anonymity.

- Vector ballots [58], combine a mix-net with homomorphic encryption. The ballot is split into two fields, plus an indicator. The first field is for recording votes which are strictly from the race's predetermined set of candidates, while the second field is for recording write-in votes. The indicator is flagged only in the case of a write-in vote. Efficient homomorphic encryption is used to process and tally the set of all non-write-in votes, and produces an unofficial result. The write-in votes are tallied later, off-line, by a mix-net. This is acceptable since in almost every case, the write-in ballots do not determine the winner of the election.

2.1.5 Implemented Schemes

There are a few documented cases where remote cryptographic e-voting schemes have been implemented, but none in an election of significant size, nor in a governmental election. Two such schemes are SENSUS, by Cranor [29], and E-Vox, by Herschberg [46]. Both were used in small, university-based elections in the late

1990's, the former at Washington University and the latter at MIT. Both are modified versions of the blind signature scheme by Fujioka et al. [38], and were implemented by students. Neither of these schemes would be suitable for the elections that this paper focuses on, yet they provide interesting starting points for practicality considerations of the theoretical remote e-voting schemes. For an idea of the scale of the capacity of these schemes, the original election conducted with E-Vox could handle approximately 100 students with *reasonably* fast servers in 1997 [46].

2.1.6 Internet Voting

Over the past decade or so, researchers have given serious consideration to remote voting over the internet. Utilizing the vast capacity of the internet has multiple perceived benefits, including increased convenience and flexibility, along with reduced cost and labor. The hope is that i-voting may also result in increased voter turnout. Certainly, internet voting is an attractive possibility. However, there are some serious obstacles that prevent the facilitation of a secure election over the internet. The foremost obstacle is the same as that of any remote voting system: the privacy of the voter at the moment of vote casting cannot be guaranteed in a remote setting. Threats of vote buying or voter coercion are a serious concern and violate the anonymity requirement. Even with the ingenious solutions that cryptography provides us, there is no way to ensure that a voter will vote in physical privacy in an internet based election (or in any other remote voting based election).

A third obstacle stems from the fact that a scheme must deal with the existing platform of the internet itself. In practice, the internet is simply not secure

enough to use in an election. The reader may wonder why an election cannot be conducted satisfactorily since we have an (arguably) successful implementation of financial transactions over the internet. It turns out that elections differ from financial transactions in a number of critical ways. First, the receipt-freeness requirement of elections prevents the creation of take-home receipts (unless they are encrypted), which are crucial for disputes in financial transactions. Second, because an election is conducted during one day only, there are time constraints in real elections that are not as pertinent in financial transfers. For this reason, denial of service attacks over the internet are a major threat. During a time-restricted election, the density of attempted attacks on the election servers would likely be greater than regular attacker behavior on other sensitive servers. The stakes of an election are sometimes worth more to certain individuals or groups than a major financial transaction. While a denial of service attack is not likely to change the results of an election, it can certainly cause a significant disruption to the process. For these and other reasons, voting over the internet is a harder problem than that of conducting financial transactions.

Even though the physical equivalence of a denial of service attack is present in current voting practices (for example, a power outage at a polling station), the effects of such an attack on an internet election web site can be much more widespread. Also, fraud in poll station voting is much more likely to be contained to a particular precinct or small geographic region, whereas with i-voting, these geographic boundaries do not exist. Furthermore, the geography of the attacker plays little or no role in i-voting; the threat of distant or foreign attacks is more real.

Other possible internet-based attacks include web site spoofing, automated vote buying, and PC viruses. Each of these is a real and potentially catastrophic threat. More potential threats include man-in-the-middle attacks, insider attacks, software bugs, and client-side computer vulnerabilities [54]. For any attack on an i-voting system, detection is more obscure and even if detected, voter confidence would be greatly affected.

A fourth obstacle of internet voting is the lack of a Public Key Infrastructure, or PKI [41]. Authenticating an eligible voter before giving her the appropriate vote-casting privileges over the internet often requires the use of the voter's own public/private key pair. This assumes that each voter has a cryptographic key and that the entire infrastructure is secure and widespread, which is not the case today. Distributing keys to all eligible voters is a difficult problem that is possible to overcome, yet unlikely to be solved in the near future.

Another problem is that the testing and certification of internet voting may be much more difficult. Not all the equipment, such as operating systems and browsers, will be controlled by the election board. There is increasing disparity, rather than standardization, in the use of these products.

Furthermore, it can be argued that some of the *perceived* advantages of remote internet voting would not come true. For example, some proponents of an i-voting system claim that one advantage would be increased voter turnout. They argue that the convenience of being able to vote from your own home would raise the percentage of voters voting in a single election. We argue against this; we believe that while i-voting may encourage an increased turnout for a *particular* set of voters,

the overall turnout may not change much. This is due to the notion of the digital divide. It is probable that those who belong to the portion of the divide with poor computer accessibility will be further separated and possibly disenfranchised by i-voting technology. Also, the perceived reduction in cost of implementing an i-voting system compared to current systems may not be realistic. The cost of buying and maintaining voting servers, standardized databases and routing systems, along with other costs, need to be considered [17].

Another deterrent is that the result of a United States Department of Defense $22 million research effort on remote internet voting was quite pessimistic. Project SERVE (Secure Electronic Registration and Voting Experiment), was halted in early 2004 after the researchers concluded that the proposed experiment was too insecure to carry out. The proposal was aimed to facilitate remote PC-based voting for overseas military and other personnel in the 2004 primary and general elections. The preliminary research reported that all security vulnerabilities of DRE-based systems still exist in an i-voting context, with the addition of multiple other serious security concerns.

For further reading on the prospects of internet voting, see [41, 54, 87, 89, 99].

2.2 Poll Station E-Voting Schemes

Only recently have electronic systems been used in poll station elections. Recall that we classify schemes as electronic only if votes are *cast* onto an electronic device. Thus far, the schemes of this type have been under intense scrutiny by

security experts. Only in the past decade or so have researchers focused on rigorously ensuring that the security requirements are properly met. We discuss various approaches below.

2.2.1 Typical DRE-machines

The current class of DRE-based schemes are causing widespread controversy. They lack universal verifiability; most do not even have a rigorous auditing mechanism. The integrity of the tally is not provable, and real security vulnerabilities have not been properly considered or handled. The networks that connect multiple DRE-machines together are subject to attacks that could disrupt an election, and the insider threat is significant. Nevertheless, there has not yet been a proven case of fraud detected. A handful of elections conducted with DRE-machines have spawned suspicion of malicious fraud [48], and a few DRE-machines have detected non-malicious errors [65]. The lack of detection, however, certainly does not prove the absence of fraud.

The crux of the issue, however, is that most companies refuse to publicize the source code for their products. They claim that the security of their systems depend on their secrecy, and also that they want to protect their intellectual property from competitors. Because of this, voting and security experts are unable to properly evaluate these schemes. Allowing experts to freely study the code could increase the chances of both exposing and fixing any vulnerabilities. Alternatively, their evaluation could lead to a proof or verification of the scheme's security.

In the United States, several companies are in the DRE-based electronic voting

business. The following is a non-inclusive list of companies whose DRE-machines have been used in national elections: Diebold Election Systems, Sequoia Voting Systems, Election Systems and Software, Avante International Technology, MicroVote General Corporation, and Hart InterCivic.

Despite Diebold Election Systems' insistence on keeping their code closed-source, a collection of code was inadvertently leaked in 2002 that appears to be a part of the company's AccuVote-TS voting technology. This sparked a published analysis in 2003 by a group of computer security experts, led by Rubin [60]. The report highlighted many vulnerabilities, including unauthorized privilege escalation, incorrect use of cryptography, vulnerabilities to network threats, and poor software development processes. The affect of this report has been heightened awareness, controversy, and even bitter debate over the security of DRE-machines in general. This was the report that started it all: it fueled intense discussions and debate over how secure our electronic elections have been since then, and prompted many to recommend against the use of Diebold's products [67], as well as other DRE-machines.

2.2.2 Frog Voting

The concept of frog voting is attributed to Bruck, Jefferson, and Rivest [15]. It consists of a novel *modular* voting architecture. The main idea is to separate the voting procedure into two distinct stages: vote-generation and vote-casting. This modular system would provide better security exactly where security is most crucial: at the vote-casting stage. The vote-casting module in this framework could

be designed to be extremely simple and transparent, minimizing the trust placed in electronic devices. The "frog" is a physical token given to the voter during an election. The frog is initialized by an election official with pertinent election-specific information. The frog, most likely a simple memory card with "read" and "write" capabilities, would then be placed in a vote-generation device by the voter. This device can be as complex as desired, with the goal of usability in mind. Its only function is to provide an interface with the voter to make her selections and review them. The voter then transfers the frog to the vote-casting equipment. This device is required to be as simple and transparent as possible; its functions are to give the voter a final chance to review her vote, cryptographically sign the confirmed data file on the frog, then "freeze" all of the data on the frog. The freezing prevents any further alterations from being made. The vote is now stored in two places: the physical frozen frog is dropped into a special receptacle, and an electronic copy is transmitted via a serial port to one or more vote storage units. The former of these storage options provides an audit trail for recounts. The latter is capable of efficiently tallying the votes in a universally verifiable manner.

2.2.3 VoteHere's VHTi

The company VoteHere has developed technology that complements any existing e-voting DRE-based scheme [74]. Their product, VHTi, is a voter-verified auditor. It adds technology that can provably verify the accuracy of the election tally. The scheme is based on homomorphic encryption in a modified ElGamal cryptosystem. Voters using VHTi are given a physical token, which contains a blank

ballot. After voting, a set of verification codes (which prove to voter that her vote was recorded correctly, yet reveal no information about the content of the vote to others) are printed on a receipt and retained by the voter. The assignment of the codes corresponds to a random permutation that is specific to each ballot, which the DRE-machine had committed to before the voting had begun. The voter can later verify on an election web site that her receipt gets posted correctly. Audits are performed on the operations of the DRE-machines by checking their commitments on the random permutations. More information can be found at www.votehere.net. This technology seems to be quite promising. Like Chaum's scheme, the VHTi uses both encrypted receipts and has similar goals.

2.3 Summary

Thus far, there is no practical, provably secure e-voting scheme that can be trusted in a large-scale, binding, governmental election. The remote e-voting schemes are of great theoretical importance, but have not yet proven to be practical in an election of this type. On the other hand, the current poll station e-voting schemes either seem too vulnerable or are still in the stages of development. Chaum's scheme is a poll station e-voting protocol that makes excellent use of cryptography and other mathematical properties in order to achieve a level of transparency and verifiability not yet seen by other poll station schemes, while still meeting many of the practical concerns that most remote e-voting systems lack. We present this innovative scheme in Chapter 3.

Chapter 3

Chaum's Voter-Verifiable Election Scheme

3.1 Features

Chaum's Voter-Verifiable election scheme provides a novel, elegant means to conduct elections. Due to the widespread controversy over the trustworthiness of the current DRE-based electronic voting schemes, the voting community currently has focused on creating transparency in elections. Chaum's scheme does just that: the final tally can be mathematically proven to be secure through end-to-end auditing mechanisms. The proofs are conducted publicly, creating total transparency for every translation step of the election. The scheme holds all parties accountable, thereby eliminating the need to trust any person or device.

Since all processes in Chaum's scheme are conducted electronically, election security can be indisputably verified in real-time. This is a clear advantage over any scheme that either relies on lengthy manual recounts for election verification, or one that does not meaningfully verify results at all (like many current DRE-based schemes). No e-voting scheme before Chaum's has provided a practical and transparent universally verifiable scheme.

3.2 Overview of the Election Procedure

We present an overview of Chaum's Voter-Verifiable election scheme in this section.

Chaum's scheme is designed to run on any existing DRE-machine. Each machine needs to be equipped with a special purpose printer unit, yet this is the only hardware modification necessary. Special software would be created in order to meet the design elements of the scheme. The following steps give an overview of the election process.

1. At the DRE-machine, an authenticated voter selects her desired candidate and submits it as her vote.

2. The DRE-machine prints a matrix of pixels onto each of two transparent sheets of paper, called *receipts*. When aligned correctly, these receipts reveal a plaintext image of the chosen candidate's name. This image is called the *Ballot image*. Each receipt alone looks like pixels of random noise and is considered an encrypted representation of the vote since it reveals no information about the vote. The DRE-machine also prints additional information onto each receipt that is necessary for decryption.

3. The voter verifies her selection at this time; if the *Ballot image* shown is incorrect, the voter may cancel this receipt and start over.

4. The voter randomly selects one receipt to keep as an encrypted receipt and informs the DRE-machine of her selection of this *chosen receipt*.

5. The DRE-machine prints a pair of digital signatures on both receipts. These signatures are later used to test the authenticity of the chosen receipt.

6. The voter separates the two receipts and keeps the chosen receipt. The other, *unchosen receipt*, gets physically destroyed in a shredder to prevent reconstruction of the *Ballot image* outside the voting booth. Likewise, the corresponding electronic copy will be destroyed from the machine's memory.

7. Outside the polling station, the voter or a third party can verify both the authenticity of the chosen receipt and the correctness of the decryption information printed by the DRE-machine using a small, hand-held *scanner device*.

8. For each legitimate vote cast, an electronic copy of the *chosen* receipt is posted by the DRE-machine to a public web site. Voters can verify that a copy of their receipt, now called their *vote*, has been correctly posted to the web site. The presence of a vote in this *initial receipt batch* effectively ensures its correct inclusion in the final tally.

9. A group of election *Trustees* collectively, yet sequentially, transform a batch of votes taken from the web site into their corresponding plaintext *Ballot image*. Each Trustee partially decrypts each vote and applies a secret permutation to all votes in the batch before passing control to the next Trustee. By distributing the work among the Trustees as such, the anonymity of the votes is maintained. The output of each Trustee's operations are posted to the web site for use in the auditing stage.

10. The fully decrypted votes in this *final receipt batch* are tallied using dependable software. This can be done redundantly by competing software packages in order to ensure accuracy.

11. An audit is performed on the operations of the Trustees. As a result, either integrity fraud is detected for one or more Trustees, or the election is certified to be accurate.

3.3 Chaum's Scheme

We present Chaum's scheme by following an example vote through the procedure. This illustration should be sufficient for understanding the security analysis of the scheme presented in Section 3.4, which is our main focus. Full details of the scheme can be found either in the original paper by Chaum [19], or in an analysis by Bryans and Ryan [16]. Other related publications include [24, 90, 91, 102], and [103].

We divide the description of the scheme into four main sections: receipt construction, receipt verifications, decryption, and audits on the decryption.

3.3.1 Receipt Construction

The construction of the receipts is based on visual cryptography [71], which is a method to encrypt an image by visually obscuring it. The plaintext image is only revealed when two separate image layers are superimposed. When separated however, each image layer looks like random noise. One of these layers is considered the ciphertext and the other the key. Then, decryption of the ciphertext is only possible

to those who hold the secret decryption key. The role of visual cryptography in Chaum's scheme is to provide an encrypted receipt to the voter that proves that a vote is correctly included in the tally, yet the content of the receipt is unreadable outside of the voting booth. In this way, the scheme ensures accuracy while maintaining receipt-freeness.

The building block of Chaum's adaptation of visual cryptography is called a *pixel symbol*. One pixel symbol is a square unit divided into four smaller squares, two of them colored white and the other two colored black. The two squares of identical color are always diagonal from each other, creating just two possible pixel symbols, as shown in Figure 3.1. We assign one of the pixel symbols the binary digit 0 and the other one the binary digit 1.

0 **1**

Figure 3.1: The two possible pixel symbols for visual cryptography.

Next, Figure 3.2 diagrams all possible results of visually overlaying any pair of these pixel symbols. We define $\bigoplus_v$ to be the **v**isual operator [16]. Notice that the only two images, called *stacked symbols*, that result from this operation can be classified as either *semi-transparent* or *opaque*. Any two of the same pixel symbol produce a semi-transparent stacked symbol while different pixel symbols result in an opaque stacked symbol. We assign the binary digit 0 to either of the semi-transparent stacked symbols and the binary digit 1 to an opaque stacked symbol.

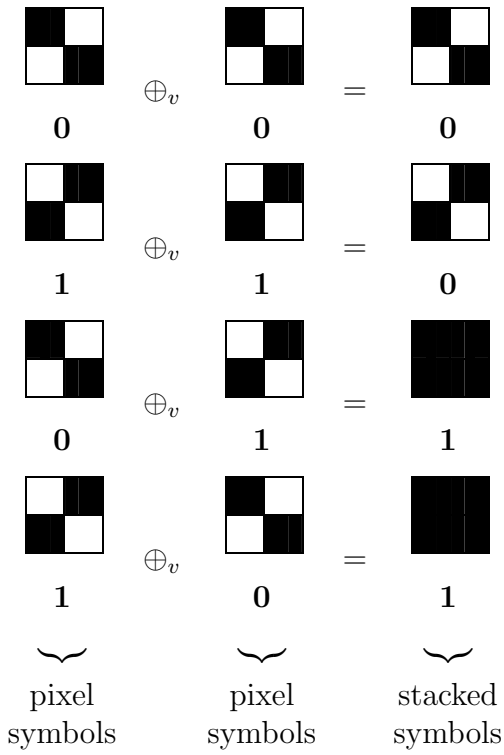


Figure 3.2: The output from the visual operator.

Constructing the *Ballot image*

We now consider a $m \times n$ matrix with mn-many *stacked symbols* as its entries. In order to represent a particular plaintext English letter in this matrix, semi-transparent stacked symbols are placed within the matrix in an arrangement that forms the closest representation of the typewritten image of the letter. The remaining squares in the background of the matrix are filled with opaque stacked symbols. Although not a visually perfect representation of the intended plaintext letter, this allocation of stacked symbols will produce a readable, indisputable version of it. We refer to this matrix which contains the plaintext as the *Ballot image* matrix. Figure 3.3, replicated from [19], shows how a vote for the letter e would be represented in an 8×7 *Ballot image* matrix of stacked symbols.

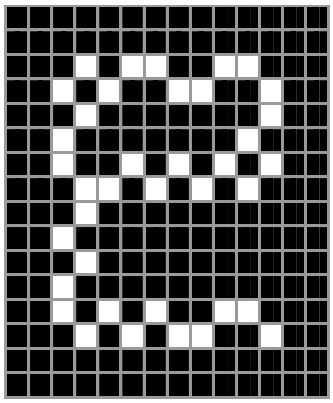


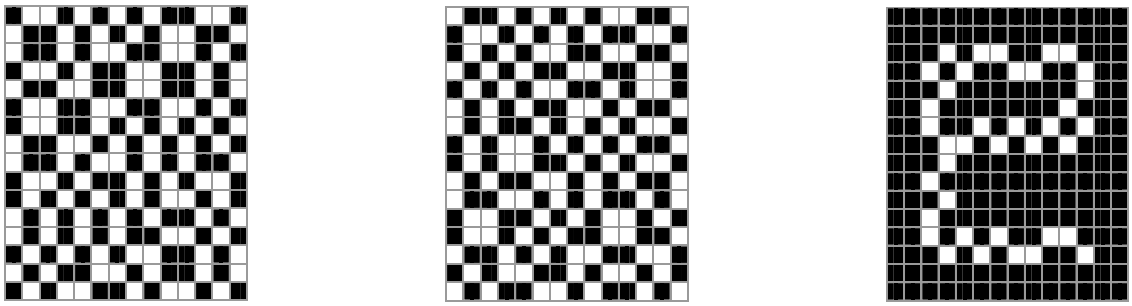
Figure 3.3: The *Ballot image* matrix for the vote e.

Constructing the *Layers*

In order to encrypt this vote, the DRE-machine must create two receipt *Layers*, top and bottom, such that when overlaid, they reveal the *Ballot image*. Figure 3.4 illustrates a possible pair of receipt *Layers*, denoted ${}^{t}Layer$ and ${}^{b}Layer$, that visually combine to form the *Ballot image* for the vote e.

In order to illustrate the procedure for creating ${}^{t}Layer$ and ${}^{b}Layer$, as well as illustrating the remainder of the voting scheme, we will use a different, simpler example vote. In this new example, the *Ballot image* matrix will be of dimension 4×2. Figure 3.5 displays the *Ballot image* representing the vote C. Formally, the goal is to construct ${}^{t}Layer$, ${}^{b}Layer$ such that ${}^{t}Layer \oplus_{v} {}^{b}Layer = Ballot\ image$.

The DRE-machine will perform a series of steps in order to construct ${}^{t}Layer$ and ${}^{b}Layer$. The specific construction of these *Layers* ensures that decryption will be possible. Recall that inside the voting booth, the voter chooses one receipt (containing a *Layer*) to retain and the other, unchosen, receipt (containing the other *Layer*) gets destructed. The chosen receipt is posted to the election web



$^{t}Layer$ $^{b}Layer$ *Ballot image*

Figure 3.4: The receipt *Layers* for the vote e (when overlaid, they reveal the *Ballot image*).

site and passed to the Trustees. The task of the Trustees, then, is to collectively reconstruct the *Layer* from the unchosen receipt, making decryption of the vote possible.

The example vote C will be used during the remainder of the description of Chaum's scheme. We let the number of Trustees be two. The number of Rounds of encryption and decryption should be double the number of Trustees (for an explanation, see Section 3.3.4), and thus there are four Rounds for our example. The functions *hash* and $hash'$ are public cryptographic hash functions, whose composi-

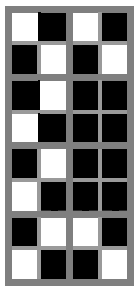


Figure 3.5: The *Ballot image* for the vote C.

tion outputs a binary string of length $\frac{mn}{2} = \frac{4 \cdot 2}{2} = 4$. Each ballot has a unique serial number, q. The DRE-machine has private signature keys ${}^{t}s$ and ${}^{b}s$, corresponding to the top and bottom, respectively.

Here are the steps that the DRE-machine follows in order to construct ${}^{t}Layer$ and ${}^{b}Layer$.

- First, the *Ballot image*, composed of stacked symbols, is converted to its equivalent binary matrix *Ballot* (See Figure 3.6). Then this matrix is *checkerboarded* into two $m \times \frac{n}{2} = 4 \times 1$ matrices, ${}^{t}B$ and ${}^{b}B$. Every alternating matrix entry, corresponding to every square of one color on a checkerboard, is used to construct ${}^{t}B$ and the remaining entries are used to construct ${}^{b}B$. The matrices ${}^{t}B$ and ${}^{b}B$ are subsequently represented as binary strings (see Figure 3.7). Formally,

$$ {}^{t}B_{i,j} := Ballot_{i,\, 2j-(i \bmod 2)} $$

$$ {}^{b}B_{i,j} := Ballot_{i,\, 2j-((i+1) \bmod 2)} $$

For our example we have:

$$ {}^{t}B = (0100) $$

$$ {}^{b}B = (0010) $$

- For both the top and bottom, four variables d'_l are prepared (one for each Round of encryption/decryption). The digitally signed serial number q, along with an index, are input into the function $hash$. For $1 \leq l \leq 4$,

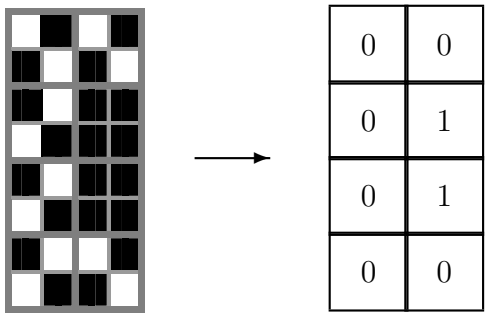


Figure 3.6: The binary matrix *Ballot* for the vote *C*.

$$^{t}d'_l := hash(\{q\}_{t_s}, l)$$

$$^{b}d'_l := hash(\{q\}_{b_s}, l)$$

- Next, these d'_l variables are the input into the second hash function, $hash'$. This results in four variables d_l for the top layer and four for the bottom. For $1 \leq l \leq 4$,

$$^{t}d_l := hash'(^{t}d'_l)$$

$$^{b}d_l := hash'(^{b}d'_l)$$

For our example, we created sample output of this step:

$$^{t}d_1 = (0011) \qquad ^{b}d_1 = (0010)$$

$$^{t}d_2 = (1110) \qquad ^{b}d_2 = (0110)$$

$$^{t}d_3 = (1100) \qquad ^{b}d_3 = (1110)$$

$$^{t}d_4 = (0110) \qquad ^{b}d_4 = (1000)$$

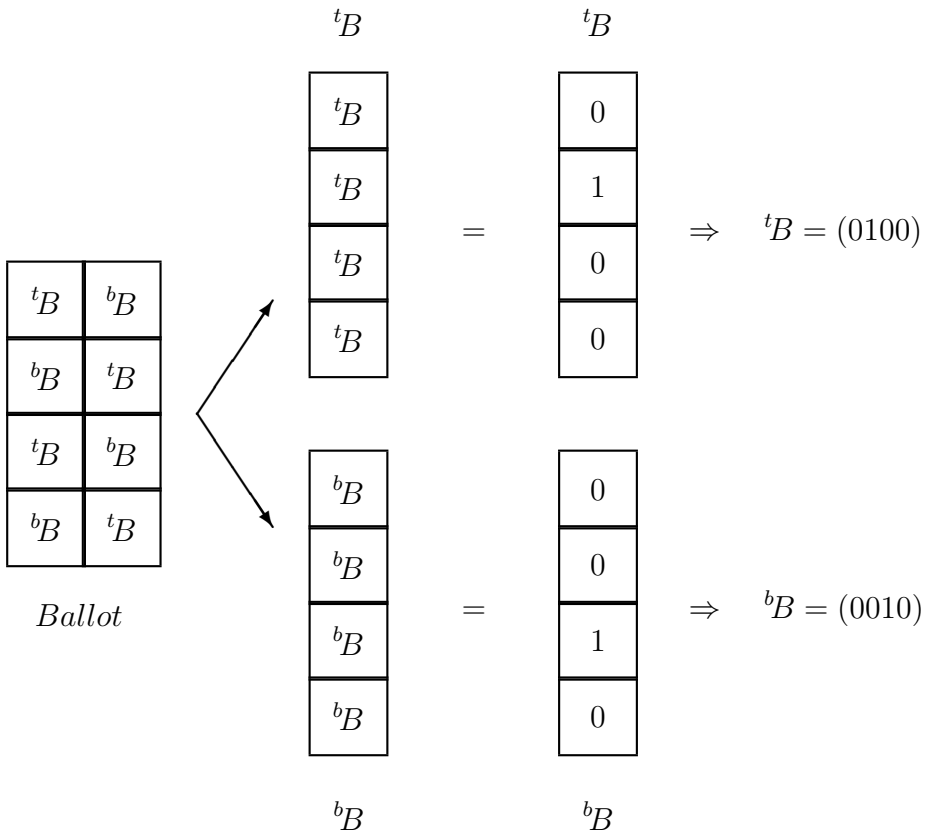


Figure 3.7: The *checkerboarding* process of the *Ballot*.

- Then, for both the top and bottom, the four d_l variables are *xor*'d together to form a binary string W. For $1 \leq l \leq 4$,

$$^tW := \bigoplus {}^td_l$$

$$^bW := \bigoplus {}^bd_l$$

For our example, we have:

$$^tW = (0111)$$

$$^bW = (0010)$$

- Next, new binary strings ^bR and ^tR are formed according to the relationship

$$^{t}R := {}^{b}W \oplus {}^{t}B$$

$$^{b}R := {}^{t}W \oplus {}^{b}B$$

For our example, we have:

$$\begin{array}{rr} {}^{b}W = (0010) & {}^{t}W = (0111) \\ \oplus \underline{{}^{t}B = (0100)} & \oplus \underline{{}^{b}B = (0010)} \\ {}^{t}R = (0110) & {}^{b}R = (0101) \end{array}$$

- Lastly, (see Figure 3.8)

$^{t}Layer$ is formed by *checkerboarding* ^{t}R and ^{t}W

$^{b}Layer$ is formed by *checkerboarding* ^{b}W and ^{b}R

Formally,

$$^{t}Layer_{i,\,2j-((i+1)\,mod\,2)} := {}^{t}W_{i,j}$$

$$^{t}Layer_{i,\,2j-(i\,mod\,2)} := {}^{t}R_{i,j}$$

and

$$^{b}Layer_{i,\,2j-((i+1)\,mod\,2)} := {}^{b}R_{i,j}$$

$$^{b}Layer_{i,\,2j-(i\,mod\,2)} := {}^{b}W_{i,j}$$

Constructing the *Dolls*

Since the *Layer* not chosen by the voter gets destroyed, there must be some information printed on the chosen *Layer* that allows the Trustees to reconstruct the

unchosen *Layer*, and therefore the vote. The DRE-machine creates objects known as *Dolls* that encrypt this information, and prints the *Dolls* on both receipt *Layers*.

For both the top and bottom, the *Dolls* encrypt the d' variables that were formed during the receipt *Layer* construction. First, d'_1 is encrypted with the public encryption key e_1 corresponding to the Trustee in Round 1. The resulting value, $Doll_1$, along with d'_2, is encrypted with e_2 (the public encryption key of the Trustee in Round 2) in order to form $Doll_2$. This is continued for every subsequent Round.

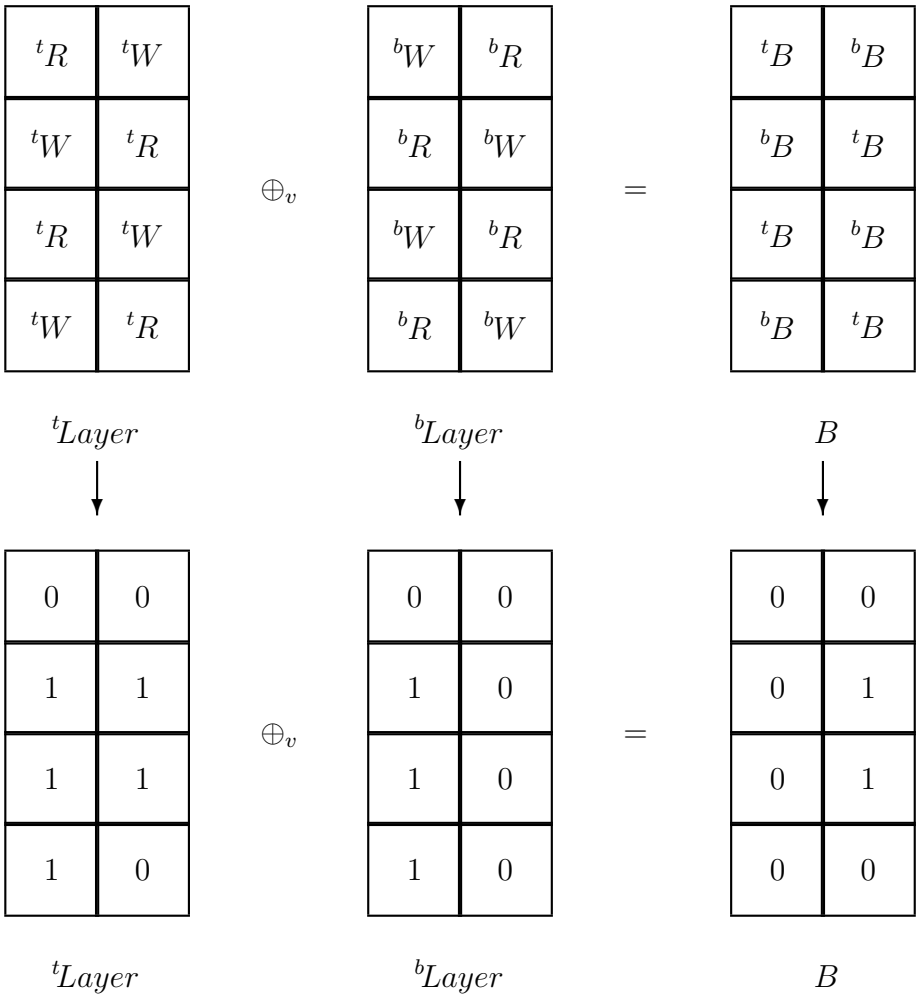


Figure 3.8: The receipt *Layers* for the vote C.

Through these layers of encryption, the *Dolls* hide the d' variables, which are needed to reconstruct the unchosen *Layer*. Therefore, only through the cooperation of all Trustees (each applying its corresponding decryption key in the proper Round) can the *Dolls* be decrypted, and thus the unchosen *Layer* recovered.

For both the top and bottom, the DRE-machine constructs

$$Doll_1 = e_1(d'_1)$$

$$Doll_2 = e_2(d'_2, Doll_1)$$

$$Doll_3 = e_3(d'_3, Doll_2)$$

$$Doll_4 = e_4(d'_4, Doll_3) = Doll$$

such that

$${}^tDoll := e_4({}^td'_4, e_3({}^td'_3, e_2({}^td'_2, e_1({}^td'_1))))$$

$${}^bDoll := e_4({}^bd'_4, e_3({}^bd'_3, e_2({}^bd'_2, e_1({}^bd'_1))))$$

3.3.2 Receipt Verifications

The following is a list of all information that is printed onto the receipts.

TOP RECEIPT

- ${}^tLayer$ - top receipt *Layer*
- q - unique serial number
- ${}^tDoll$ - top *Doll*
- ${}^bDoll$ - bottom *Doll*
- $\{q\}_{t_s}$ - the serial number, digitally signed
- $\{{}^tLayer, q, {}^tDoll, {}^bDoll, \{q\}_{t_s}\}_{t_o}$ - all receipt information, digitally signed by the

DRE-machine with its "overall" signing key corresponding to the top

BOTTOM RECEIPT

- ${}^{b}Layer$ - bottom receipt $Layer$
- q - unique serial number
- ${}^{t}Doll$ - top $Doll$
- ${}^{b}Doll$ - bottom $Doll$
- $\{q\}_{b_s}$ - the serial number, digitally signed
- $\{{}^{b}Layer, q, {}^{t}Doll, {}^{b}Doll, \{q\}_{b_s}\}_{b_o}$ - all receipt information, digitally signed by the DRE-machine with its "overall" signing key corresponding to the bottom

For the remainder of our discussion, we assume the voter had chosen to retain the **top** receipt. Therefore, the bottom receipt is the unchosen receipt, and is destroyed. The following are a series of verifications that can be performed by the voter.

1. Still inside the voting booth, the voter can verify that the two overlaid receipt $Layers$ reveal the proper $Ballot\ image$ (which should be an image of the name of the candidate she intended to vote for). Formally, she can verify that

$$ {}^{t}Layer \oplus_{v} {}^{b}Layer = Ballot\ image. $$

2. Once outside the voting booth, the voter can, with the aid of a scanner device, perform additional verifications on her chosen receipt. Since she had chosen the top receipt, the scanner device has input

$$ < {}^{t}Layer, q, {}^{t}Doll, {}^{b}Doll, \{q\}_{t_s}, \{{}^{t}Layer, q, {}^{t}Doll, {}^{b}Doll, \{q\}_{t_s}\}_{t_o} >. $$

Note that the function $hash$, the DRE-machine's inverse signature keys ${}^{t}s^{-1}$ and ${}^{t}o^{-1}$, and the Trustees' encryption keys e_l for $1 \leq l \leq 4$, are all public. The scanner device:

(a) Computes $\tilde{q} = \{\{q\}_{t_s}\}_{t_s{}^{-1}}$, and checks that $\tilde{q} = q$

(b) Computes

$\{{}^{t}\widetilde{Layer}, \tilde{q}, {}^{t}\widetilde{Doll}, {}^{b}\widetilde{Doll}, \{\tilde{q}\}_{t_s}\} = \{{}^{t}Layer, q, {}^{t}Doll, {}^{b}Doll, \{q\}_{t_s}\}_{t_o}\}_{t_o{}^{-1}}$ and checks that ${}^{t}\widetilde{Layer} = {}^{t}Layer, \tilde{q} = q, {}^{t}\widetilde{Doll} = {}^{t}Doll$,
${}^{b}\widetilde{Doll} = {}^{b}Doll$, and $\{\tilde{q}\}_{t_s} = \{q\}_{t_s}$

(c) Computes

- ${}^{t}d'_l = hash(\{q\}_{t_s}, l)$ for $1 \leq l \leq 4$
- ${}^{t}\widehat{Doll} = e_4({}^{t}d'_4, e_3({}^{t}d'_3, e_2({}^{t}d'_2, e_1({}^{t}d'_1))))$

and checks that ${}^{t}\widehat{Doll} = {}^{t}Doll$

The first two checks verify the digital signatures of the DRE-machine. These checks ensure that the receipt was created by an authentic DRE-machine, and not forged. The last check verifies that ${}^{t}Doll$ was formed correctly.

We note that since the value of $\{q\}_{b_s}$ is not known to the scanner device (in fact it has been destroyed with the bottom receipt), it is impossible to verify the construction of ${}^{b}Doll$. This means that if the DRE-machine had falsified ${}^{b}Doll$, it will go undetected. However, if the voter had instead chosen to retain the bottom receipt, ${}^{b}Doll$ would be verified and ${}^{t}Doll$ would not. A falsified $Doll$ will be decrypted differently than the correct $Doll$, and will result in a different *Ballot image*. There-

fore, this is one way (but is the only way) that a DRE-machine can alter a vote. If the DRE-machine falsifies one of the *Dolls*, then, depending on which receipt the voter chooses, there is a 50% chance of it being detected.

The following security properties are a result of the aforementioned checks conducted by the scanner device.

- A forged receipt is always detected by the scanner device since the digital signatures are verified, and are assumed to be secure.
- For one vote, if one of the *Dolls* is falsified, the chance of the fraudulent DRE-machine going undetected is $\frac{1}{2}$.
- For each of n votes, if one of the *Dolls* is falsified, the chance of the fraudulent DRE-machine going undetected is $\frac{1}{2^n}$. Equivalently, a fraudulent DRE-machine that alters n votes is detected with probability $1-\frac{1}{2^n}$. This probability is considered *overwhelmingly* high for even a small value of n. For example, if $n = 8$, fraud is detected with approximate probability .996, or approximately 99.6% of the time.

At the close of the election, an electronic copy of every chosen receipt is posted to the election web site by the DRE-machine. This *initial receipt batch* becomes the input into the next phase of the election, the decryption process.

3.3.3 Decryption

A mix-net is used in Chaum's scheme in order to maintain anonymity of the votes during decryption. It is possible, instead of using a mix-net, to place all trust

in one Trustee to solely perform the decryption. But then that one Trustee would know the correspondence between every voter and her vote. Since that is clearly undesirable, a mix-net is implemented to distribute the decryption computations among a series of Trustees.

The input into the mix-net is the *initial receipt batch* consisting of a set of encrypted votes read from the election web site. All identifying information, namely the serial number, is stripped from the votes before entering the mix-net. The batch of encrypted votes are directed through the series of Trustees, with each Trustee partially decrypting each vote and performing a secret shuffling on each *intermediate batch* of votes. The output from the Trustee in the final Round will be the *final receipt batch* that consists of the original plaintext *Ballot images* for these votes. This final output, along with the output of each Trustee during the intermediate stages, is posted to the web site for use during the auditing procedure.

Each Trustee possesses a pair of private decryption keys and uses each key once when performing two Rounds of partial decryption of the votes. The reason that each Trustee is responsible for two Rounds becomes clear in the explanation of the auditing procedure in Section 3.3.4.

To continue with our example, recall that the top receipt (which includes ${}^{t}Layer$) was the one chosen by the voter, and therefore is posted to the web site. It is thus the job of the Trustees to collectively reconstruct ${}^{b}Layer$, so that the *Ballot image* can be revealed.

Recall that the *Ballot image*, composed of stacked symbols, has a corresponding binary matrix, *Ballot* (see Figure 3.6). Also, the *Ballot* is the *checkerboarding*

of the two binary matrices ${}^tB$ and ${}^bB$. These binary matrices can also be represented as binary strings (see Figure 3.7). Therefore, the Trustees' goal of reconstructing the *Ballot image* is equivalent to reconstructing the binary strings ${}^tB$ and ${}^bB$. Due to how the *Dolls* were formed, this is actually not possible. Instead of being able to reconstruct both ${}^tB$ and ${}^bB$, the Trustees will only be able to reconstruct ${}^tB$, the binary string that corresponds to the chosen receipt. Although ${}^tB$ does not reproduce the original *Ballot image* altogether, it reproduces a *checkerboarded* half of it. In a real implementation of the scheme, the *Ballot image* matrix is much larger than 4×2. Instead, the size of the matrix would likely be on the order of 100×200 pixels. Therefore, reconstructing half of the pixels (in a checkerboard design) is enough to distinguish the names of different candidates from one another. This means that when the Trustees collectively reconstruct ${}^tB$, it is sufficient for determining the vote.

Recall that the following values are associated with the top receipt:

$$< {}^tLayer, q, {}^tDoll, {}^bDoll, \{q\}_{t_s}, \{{}^tLayer, q, {}^tDoll, {}^bDoll, \{q\}_{t_s}\}_{t_o} >$$

Before passing these values to the last Trustee, the serial number q is stripped off, along with all other extraneous information. Recall (from Section 3.3.1) that ${}^tLayer$ was formed by *checkerboarding* the ${}^tR$ and ${}^tW$ bits. Therefore, the ${}^tR$ bits can be extracted from the ${}^tLayer$. The only values required for decryption are $< {}^tR, {}^bDoll >$, and these are passed to the last Trustee to begin the decryption procedure.

Recall that the DRE-machine had constructed ${}^tR$ such that

$$^{t}R := {}^{b}W \oplus {}^{t}B.$$

Since the Trustee knows the value of ${}^{t}R$, and wants to reconstruct the value of ${}^{t}B$, it will need to obtain the value of ${}^{b}W$ (since ${}^{t}B = {}^{t}R \oplus {}^{b}W$). We will show that indeed ${}^{b}W$ *can* be sequentially computed by the Trustees (using ${}^{b}Doll$).

Also recall that ${}^{b}R$ had been constructed according to

$$^{b}R := {}^{t}W \oplus {}^{b}B.$$

Notice that the Trustee could also determine the value of ${}^{t}W$ (from ${}^{t}Layer$), and therefore would simply need to reconstruct ${}^{b}R$ in order to reveal ${}^{b}B$ (since ${}^{b}B = {}^{b}R \oplus {}^{t}W$). However, there is no way for the Trustees (or anyone else) to learn the value of ${}^{b}R$ without possessing ${}^{b}Layer$, which was discarded with the bottom receipt. Each *Doll* contains enough information necessary to reconstruct W from the corresponding receipt, but nothing about R can be learned from either *Doll*.

The Trustees' task has thus been reduced to reconstructing the binary string ${}^{b}W$. Recall that

$$^{b}W := \bigoplus {}^{b}d_l \text{ for } 1 \leq l \leq 4.$$

To set aside some notation, since

$$^{t}B = {}^{t}R \oplus {}^{b}W$$

$$^{t}B = {}^{t}R \oplus {}^{b}d_4 \oplus {}^{b}d_3 \oplus {}^{b}d_2 \oplus {}^{b}d_1$$

we can define

$$^{t}R_4 = {}^{t}R$$

$$^{t}R_3 = {}^{t}R_4 \oplus {}^{b}d_4$$

$$^tR_2 = {}^tR_3 \oplus {}^bd_3$$

$$^tR_1 = {}^tR_2 \oplus {}^bd_2$$

$$^tR_0 = {}^tR_1 \oplus {}^bd_1$$

so that $^tR_0 = {}^tB$.

Now, we illustrate the procedure of one Round of decryption by showing the operations of the Trustee in the fourth, or last, Round.

Upon input $< {}^tR, {}^bDoll > = < {}^tR_4, {}^bDoll_4 >$, the Trustee performs the following steps.

(i) For each vote, the Trustee

- Computes $\{{}^bD_4\}_{e_4^{-1}} = {}^bd'_4, {}^bD_3$

  (Recall that ${}^bD_4 := e_4({}^bd'_4, \underbrace{e_3({}^bd'_3, e_2({}^bd'_2, e_1({}^bd'_1)))}_{{}^bD_3})$)

- Computes $hash'({}^bd'_4) = {}^bd_4$

- Computes $^tR_3 = {}^tR_4 \oplus {}^bd_4$

(ii) The Trustee performs a secret shuffling of all votes in the batch. The shuffling of the votes through the entire mix-net is illustrated in Figure 3.9.

(iii) For each vote, the Trustee passes $< {}^tR_3, {}^bDoll_3 >$ to the Trustee in Round 3.

The Trustees in the remaining Rounds subsequently follow the analogous procedure. For Round l, the Trustee receives the input $< {}^tR_l, {}^bDoll_l >$ and produces the output $< {}^tR_{l-1}, {}^bDoll_{l-1} >$.

Here are the results of the 4 Rounds of decryption for our example.

Round 4:	**Round 3:**
${}^tR_4 = (0110)$	${}^tR_3 = (1110)$
$\oplus\ {}^bd_4 = (1000)$	$\oplus\ {}^bd_3 = (1110)$
${}^tR_3 = (1110)$	${}^tR_2 = (0000)$

Round 2:	**Round 1:**
${}^tR_2 = (0000)$	${}^tR_1 = (0110)$
$\oplus\ {}^bd_2 = (0110)$	$\oplus\ {}^bd_1 = (0010)$
${}^tR_1 = (0110)$	${}^tR_0 = (0100)$

Since ${}^tR_0 = (0100) = {}^tB$, the Trustees have successfully reconstructed ${}^tB$, which is enough to determine the vote.

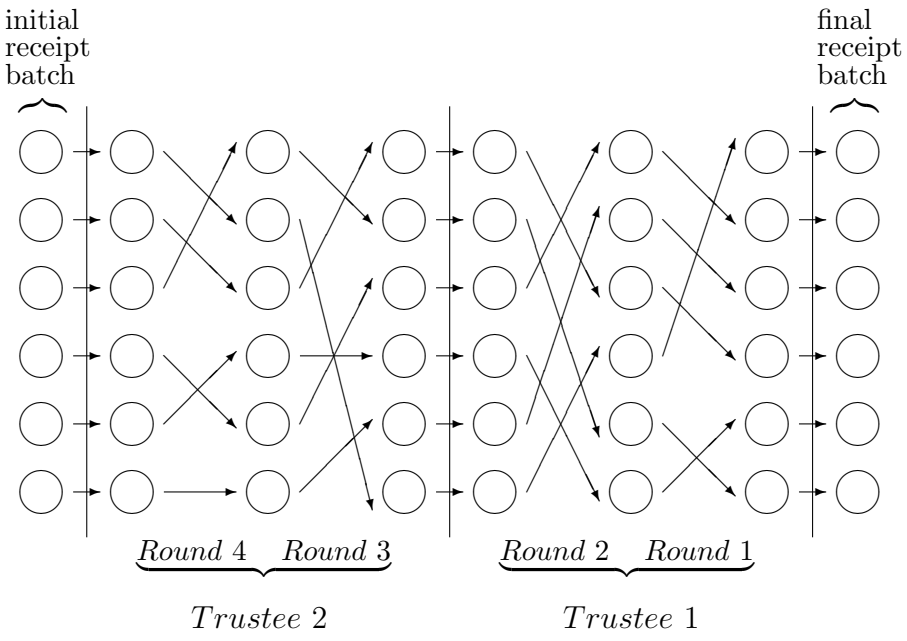


Figure 3.9: The shuffling of votes in a mix-net with 2 Trustees.

It still appears that we need to trust that the Trustees have performed all computations honestly, without altering the content of any votes and thus jeopardizing integrity. To eliminate this need to trust them, the scheme implements an auditing procedure on the operations of the Trustees called randomized partial checking [50].

3.3.4 Audits on the Decryption

The randomized partial checking (RPC) methodology is a way of ensuring integrity in a mix-net. It is an auditing procedure on the actions of the Trustees that helps provide universal verification for the votes.

The RPC auditing occurs after the completion of the decryption phase. Each Trustee is expected to reveal half of the "links" from each of its two secret shuffles in order for an *Auditor* to verify that its operations were performed correctly. In reality, the Auditor can be any interested third party and thus the operations of the Trustees can be verified universally. The links to be revealed are chosen as follows. First, a random half of the links from the first shuffle are chosen by a third party *Authority* through a public lottery. It is crucial, however, that the Trustee not know which "links" are chosen until after its mix-net operations have been completed. Instead, if the Trustee were aware of which links would be audited, it could cheat undetected on all votes *not* selected by the chosen links. After the set of links is chosen from the first secret shuffle, a *disjoint* set of links is chosen from the second shuffle. As a result, it is impossible to trace the path of any one vote through the operations of this Trustee. This is necessary for securing anonymity of the votes and is the reason that each Trustee is involved in two stages of the mix-net. This is illustrated

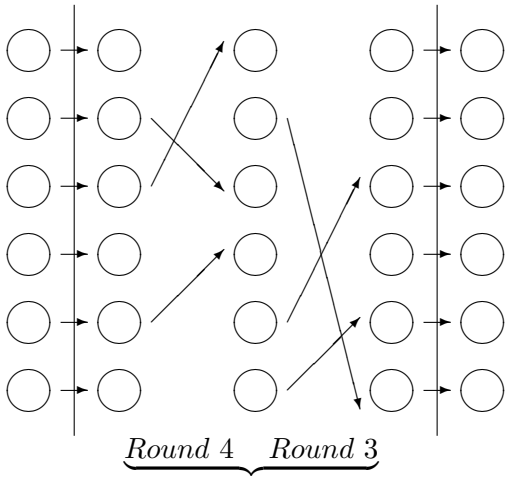
in Figure 3.10.

In order to conduct the auditing without compromising anonymity of the votes, only half of each Trustee's operations are checked. Because of this, the integrity is only verified in a probabilistic sense. If a Trustee alters one vote, the auditing procedure will detect fraud with probability $\frac{1}{2}$, since each vote has that same probability of being chosen for audit. Then if a Trustee alters n votes, fraud is detected with probability $1 - \frac{1}{2^n}$. The probability of detecting a fraudulent Trustee is thus analogous to that of detecting a fraudulent DRE-machine, which we determined was *overwhelmingly* high, even for a small value of n.

Continuing with our example, we illustrate the auditing process by explaining the procedure for the Trustee in Round 4.

For each vote audited, the Trustee must publish:

- the "link" $< {}^{t}R_4, {}^{b}Doll_4 > \rightarrow < {}^{t}R_3, {}^{b}Doll_3 >$
- ${}^{b}d'_4$



$Trustee\ 2$

Figure 3.10: Auditing the mix-net: The set of links revealed in Round 3 are disjoint from those revealed in Round 4.

Then the Auditor (for each audited vote):

1. Computes

   - ${}^{b}\hat{d}_4 = {}^{t}R_4 \oplus {}^{t}R_3$

     (Recall that the Trustee in Round 4 computed ${}^{t}R_3 := {}^{t}R_4 \oplus {}^{b}d_4$)

   - ${}^{b}d_4 = hash'({}^{b}d'_4)$

   and checks that ${}^{b}\hat{d}_4 = {}^{b}d_4$.

2. Computes

   - ${}^{b}\widetilde{Doll}_4 = e_4({}^{b}d'_4, {}^{b}Doll_3)$

     (Recall that the Trustee computed $\{{}^{b}Doll_4\}_{e_4^{-1}} = ({}^{b}d'_4, {}^{b}Doll_3)$)

   and checks that ${}^{b}\widetilde{Doll}_4 = {}^{b}Doll_4$.

In the first check, the Auditor verifies that the ${}^{t}R_4$ value is indeed linked to the ${}^{t}R_3$ value that the Trustee passes to the Trustee in Round 3. The second check verifies that the input ${}^{b}Doll_4$ is indeed linked to the output ${}^{b}Doll_3$.

This Auditing procedure is analogously repeated for each vote selected by the lottery for audit. This is exactly half of the votes in each Round.

At the end of the auditing phase, there are two possible scenarios. Either fraud has been detected, or the absence of detected fraud guarantees integrity with overwhelmingly high probability. If fraud has been detected, it needs to be handled procedurally according to pre-specified regulations for the particular election. If no fraud is detected, the election is said to be *certified*. At this point, the tally

can be computed from the decrypted votes that were the output of the mix-net (Section 3.3.3). The tally (computed electronically) can be conducted publicly and redundantly, ensuring its integrity.

3.4 Security Analysis

In this section, we analyze Chaum's scheme according to the framework of requirements outlined in Section 1.3. We describe how the scheme satisfies the objectives of a secure e-voting system. We follow the aforementioned categorization into basic and extended requirements.

Basic Requirements

1. **Accuracy** or **Integrity**

   Accuracy of the election tally is verified through a number of auditing procedures. Unconditional accuracy is not a property of Chaum's scheme; rather, probabilistic accuracy is attained. Any erroneous vote has a probability of at most $\frac{1}{2}$ of going undetected. Although that does not seem to be satisfactory, the probability of n votes going undetected diminishes in n. More precisely, n fraudulent votes will go undetected with probability at most $\frac{1}{2^n}$. Practically speaking, this is negligible. For example, rarely do as few as 8 votes change the outcome of a race. If 8 votes are altered, the probability that no fraud is detected is at most $\frac{1}{2^8} \approx 0.004$, or a 0.4% chance.

   The assurance of accuracy may be reduced if the voters' choice of receipts

in the voting booth is biased. When given a choice of top or bottom, the populace may have a natural bias to choose one of these more often than the other. This lack of randomness could improve the DRE-machine's chances of predicting the unchosen receipt, thus improving its chances of falsifying this receipt without detection. The solution would be to use a good source of randomness as a proxy for the voter to make this choice, yet this is admittedly difficult. Another accuracy concern that is undetectable by the scheme is ballot-stuffing (false vote creation) by the DRE-machine. This can be detected procedurally by simply counting the number of votes cast at the polling station and verifying that this number matches the number of receipts on the election web site.

One nice property of the scheme is that any legitimate claim that a particular vote is not included in the final tally can be verifiably confirmed (with a probabilistic guarantee), yet any false claim can be rightfully dismissed. Any receipt not generated by an authentic DRE-machine can be indisputably identified by the scanner device, preventing voters or others from forging receipts. Also, a voter cannot falsely claim that her receipt was not posted or posted incorrectly on the web site. Possession of the physical receipt is necessary in order to place a dispute. Therefore, by simply checking the web site it is trivial to refute this false claim.

2. **Anonymity** or **Ballot Secrecy**

The scheme protects anonymity by encrypting the receipts and dis-

tributing the decryption to a series of Trustees. Due to the secret shuffling of the votes in a batch, the mapping between input and output for every Trustee is obscured. A rigorous stochastic analysis of the decryption process by Gomulkiewicz, Klonowski, and Kutylowski [43] shows that the connection between the encrypted votes and their plaintexts "remains almost purely random". The scheme is also receipt-free since there is no way to reconstruct proof of a vote's content outside of the voting booth.

Anonymity can only be compromised in the following two ways. First, if any one party possesses all private decryption keys of the Trustees, the plaintext votes can be recovered and linked to individual voters. That would require either all Trustees to collude or an adversary to break the encryption scheme, and neither are assumed to occur. Second, the DRE-machine itself may secretely opt to not discard its electronic copy of the unchosen receipt. With the two receipts, it can store the plaintext vote, and use the serial number to link it to a particular voter. However, current DRE-machines are capable of an analogous attack on anonymity [103]. There is no way to verify that any DRE-machine actually discards the information linking voter to vote.

It is important to note that breaches in the anonymity requirement do not affect the tally. Since the scheme is receipt-free, voters can vote without pressure from a third party. Their votes should correctly reflect their intentions. Then once a vote is cast, a leak in privacy does not change the content of the vote. Thus, the tally should correctly reflect voter intent, regardless of

the breach in anonymity.

3. **Universal Verifiability** or **Auditability**

Chaum's scheme does not quite meet the strict definition of universal verifiability because it fails to provide end-to-end auditing on every vote. Instead, it audits *most* of the actions on the votes, which we describe below. This makes the verifiability of the scheme probabilistic, although the chance that enough votes were altered to change the election outcome is negligible.

Not *every* voter can be assured that her vote was recorded correctly by the DRE-machine. In the first place, recall that an attempt by the DRE-machine to alter one vote will be detected by the scanner device with probability $\frac{1}{2}$, yet with probability $1 - \frac{1}{2^n}$ for n altered votes. Secondly, a vote altered by a Trustee has the same chances of being detected during the randomized partial checking: with probability $\frac{1}{2}$, or $1 - \frac{1}{2^n}$ for n votes. Despite the fact that these probabilities are negligible for even a small value of n, any *one* vote may be inaccurately included in the final tally with probability $\frac{1}{2}$. Overall, the integrity of the final tally is overwhelmingly verifiable, yet for any one vote the accuracy assurance is not very strong.

There are no other ways that cheating can go undetected. There are multiple checks that indisputably detect fraud or error in all other translation steps. For instance, anyone with possession of a valid encrypted receipt can verify that it is correctly included in the initial receipt batch. Otherwise, error has irrefutably occurred and should be reported. If necessary, an electronic

copy of the voter's physical encrypted receipt can be added to the initial receipt batch manually. Also, votes added or destroyed by the Trustees during the mixing phase are immediately detectable by simply comparing the number of inputs to outputs for each Trustee. In addition, any interested third party can conduct or observe the RPC auditing as well as the ensuing tallying, ensuring that no cheating occurs during these phases.

Extended Requirements

1. **Robustness**

   As proposed, Chaum's scheme is not very robust. If one or more of the Trustees is unavailable, the election is halted. Since only a particular Trustee knows its private key, decryption cannot resume unless that Trustee participates properly. However, a key-sharing threshold technique could be incorporated into the protocol, where participation of any t out of k Trustees, but no fewer, is necessary for decryption of the votes.

   Recall that the auditing of Trustees occurs after the votes are decrypted. If the tally is deemed inaccurate at this point due to fraudulent or erroneous Trustees, the mix-net simply needs to restart. Fortunately, a re-vote is unnecessary since the input to the mix-net remains the same for another decryption attempt. For any particular Trustee, if error is detected on only a few votes, it may be tolerated. If the number of erroneous votes detected could not affect the outcome of the election, those votes become insignificant.

Recall that error or fraud by the DRE-machines can be detected during the election by a scanner device outside the polling station. If a receipt generated by an authentic DRE-machine is found to be invalid, the voter may simply re-vote at that time according to the election policy in place. If enough invalid receipts are traced to a particular DRE-machine, it should be rendered unusable, and the election can proceed with the remaining DRE-machines. If the stored copies of the valid receipts are lost or corrupted by the DRE-machine, the physical receipts retained by the voters can act as input into the mix-net process instead.

2. **Efficiency**

To our knowledge, no time estimates exist for Chaum's scheme. We conjecture, based on its comparative (lack of) complexity to other mix-net schemes, that the work of each Trustee is straightforward and quick. The audits on the Trustees are done efficiently: only half of the operations are checked, and each audit consists of strong evidence, rather than a complete proof, of correctness. The speed of these proofs is significantly better than the zero-knowledge proofs used by other mix-nets. The work of both the Trustees and the Auditors does increase with the number of voters, but dividing the votes into manageably sized batches helps offset any time increase.

3. **Flexibility**

Chaum's scheme supports *all* voting styles listed in Section 1.2, ranking it among the most flexible of voting schemes. In fact, it is already in a minority

of e-voting systems for supporting write-in votes, and may be the only poll station voting scheme that provides universal verification for write-in votes [103]. The scheme can also easily handle provisional ballots efficiently by placing them into a separate batch. Since the law relaxes privacy requirements for provisional ballots, every illegitimate vote in the provisional batch can be individually removed, and the remainder of votes in the provisional batch added to the final tally.

4. **Certifiability**

One elegant property of Chaum's scheme is that trust in the equipment or software is not necessary. We can be assured that fraud or error by the DRE-machine or the Trustees will be detected with overwhelming probability. Therefore the security of these components need not rely on pre-election certification. Instead of having to certify the equipment or software, Chaum's scheme certifies the *results* of an election. The tally is proven correct by the properties of the universal verification procedures. This idea of *proving* the accuracy of a tally is a noteworthy accomplishment of a poll station e-voting protocol, and is the standout feature of Chaum's scheme.

5. **Scalability**

The stochastic analysis of Chaum's scheme [43] mentioned under the anonymity requirement resulted in the following corollary: "For achieving [a] high security level a constant number of stages [Trustees] is enough no matter how large the population of voters is." The authors concluded this after com-

paring the "probability distribution of the permutations linking the encoded votes with the decoded votes given the information revealed by randomized partial checking" to the uniform distribution. Therefore, it is unnecessary to alter the number of Trustees depending on the number of voters. The result is that the scale of the election should not significantly affect its efficiency. Another benefit of keeping the number of Trustees constant is that the Trustee infrastructure can also remain in tact for all elections. We note here that the Trustees can be representatives from opposing political parties, for example, in order to prevent collusion by all of them.

6. **Usability**

The usability of Chaum's scheme relies heavily on the user interface of the corresponding DRE-machine. The more consideration that is given to usability during the design of the user interface, the more comfortable voters will be while voting. Once inside the voting booth, the voting process typically takes two to three minutes using a DRE-machine. With the added tasks of reviewing, selecting, printing, and destructing receipts, we expect that time to increase only slightly per voter. The real bottleneck for poll station voting is the time spent on authenticating voters and not on voting itself.

Voters are expected or encouraged to participate in multiple additional duties beyond those of traditional voting schemes. From an usability perspective, the effectiveness of this depends on the reaction of different voters and may or may not be advantageous to the scheme. The ideal situation is a

voting scheme that realizes the concept of "vote-and-go" while simultaneously ensuring that voters can legitimately and effortlessly trust the integrity of the tally. Since Chaum's scheme does not realize "vote-and-go", we conjecture that there may be differing reactions: one set of voters will find the additional duties tedious or even annoying, resulting in a lack of participation. Another set of voters may be enthused or flattered by the level of involvement they are given, *feeling* that their participation in the election is important or dutiful as a citizen. Additionally, there may be voters, albeit a minority, that fully understand the intricacies of the scheme and happily participate, *knowing* exactly why their efforts are in fact increasing the security of the scheme. There really is no way to accurately predict the populace's reaction to usability concerns. The best measure would be to survey voters after implementation.

7. **Accessibility**

It is appropriate to evaluate the accessibility of the DRE-machine providing the user interface for Chaum's scheme. The best machines of today provide satisfactory access for the disabled, including the visually impaired and the physically handicapped. Most DRE-machines also provide access to non-English speakers.

Chapter 4

Obstacles to Adoption

In this chapter, we explore the obstacles which would be necessary to overcome prior to the implementation of Chaum's scheme to a real, binding, large-scale governmental election. The obstacles include various social, legal, and economic barriers.

4.1 Social Issues

Although the technical aspects of an electronic voting scheme are central to its worth, a scheme will never be adopted unless it is viewed as trustworthy. Not only do the election boards and ITAs need to trust it, but, arguably more importantly, the electorate itself must believe in a scheme's trustworthiness.

Chaum's scheme requires trust in the mathematics. Those who are interested and have the background to understand the mathematical proofs can safely believe in the scheme's trustworthiness, but can the general public do that? It is unknown whether the public would be more willing to trust the word of mathematicians rather than the current situation: trusting the proprietors of the DRE-machines.

According to [16], it is "not enough for the system to be dependable, it must also be seen to be dependable. [Chaum's] scheme is complex and difficult to understand. To what extent could 'the average voter' understand the scheme and believe

the claims? To what extent would assurances of experts suffice? How easy would it be to undermine public confidence...?" The authors of [16] are planning to conduct a number of trials that address these sociological questions, which should prove useful to our evaluation.

There was a field study conducted and published by Bederson and Herrnson in 2002 [10] that sought to capture the electorate's reaction to touch-screen style DRE-machines. The results indicated that the reaction to the Diebold Accu-Vote TS machines during these trials was that "most of the voters ... responded favorably to it." The most relevant question asked in the questionnaire was whether the voters "trusted that the system recorded the vote they intended to cast." Of the responses, 85% reported trust in the system, while 7% reported moderate trust and the remaining 8% indicated they did not trust or only somewhat trusted the system.

Unfortunately, there have not been any known studies evaluating a prototype version of Chaum's scheme. The results of [10] lead us to believe that generally, voters are open to DRE-machine technology. The authors were reportedly left "optimistic, but concerned" regarding electronic voting systems. For Chaum's scheme, it is important to consider whether the scheme is too technically complex for voters to trust it. This seems to be the major obstacle for its public acceptance. Although all steps are transparent, it requires a sophisticated level of expertise and exposure to the technical aspects of the voting problem in order for the scheme's merits to be fully understood. There have been recent efforts that address this issue by aiming to produce simplified variants of Chaum's scheme. There is work being developed independently by van de Graaf [102], Vora [103], and Bryans and Chaum [24]. These

schemes seem very promising, and are likely to be viable options in the future. But even still, gaining widespread acceptance is a problem of education and marketing, and only a substantial, widespread effort is likely to be effective.

4.2 Legislative Issues

In 2002, Congress passed the Help America Vote Act (HAVA), an election reform bill of significant proportions. The bill appropriated three and a half billion dollars for new voting equipment across the country. The funds are offered to states that used paper, lever, or punch card systems in 2000 and choose to upgrade their election technology to DRE-machine or optical scanning systems by 2006. States that comply with HAVA's regulations would receive four thousand dollars for every qualifying precinct in the state [101]. As evidenced by HAVA, DRE-machine electronic voting has permeated its way into the mainstream, and appears to be permanent.

The effects of HAVA are both good and bad for Chaum's Voter-Verifiable election scheme. On one hand, Congress' endorsement of DRE-machine technology helps DRE-based schemes in general earn public acceptance. Also, sales of DRE-machines have grown as a result of HAVA funds. Having DRE-machines already in place benefits Chaum's scheme since it is really an add-on to DRE-based systems.

On the other hand, the compliance deadline for states to be able to receive HAVA funds is January 1, 2006. Chaum's scheme has not yet made it to the market, and will likely not by the deadline. Therefore, all states that were interested

in upgrading their voting technologies will have no financial incentive after 2006 to purchase the necessary components for Chaum's scheme. Budget constraints have historically caused states to only comply with the minimal election requirements, and HAVA does not insist on the level of verification that Chaum's scheme provides. Due to this, it is likely that Chaum's scheme will only be adopted if stricter legislation were passed or other political pressures were to persist that would require universal verification for DRE-based schemes.

4.3 Economic Issues

The affordability of any scheme is certainly a concern. Luckily, the bulk of the costs of Chaum's scheme lie in the DRE-machines themselves. Because of HAVA, DRE-machines have already become popular, and are likely to become more widespread by 2006. Therefore in many cases the cost of adopting Chaum's scheme would consist of relatively small add-on costs, including special purpose printers, transparent paper, and shredders for the receipts. There is also the cost of designing, creating, and maintaining election web sites. There are multiple other recurring costs that are not seen as additional costs, as they are no different than the costs of other current election schemes. A few examples of these reoccurring costs include those of the personnel, polling stations, and certification of the equipment associated with an election.

4.4 Expert Opinion

Experts appear to be reacting quite positively to Chaum's scheme [66]. The concept of a scheme that provides provable security to an election is indisputably desirable. It seems that a new era of election technology has begun, with this idea being fundamental to these developing schemes.

Bryans and Ryans, in their 2003 dependability analysis [16], state that "we believe that the Chaum voting scheme comes closer than any other scheme we are currently aware of to meeting all of [our stated] requirements." Ryan also commented that the scheme's "technical (mathematical) core appears robust" [92]. Furthermore, Gomulkiewicz et al. report that Chaum's scheme is a "fairly practical scheme designed to meet the demands mentioned" in their paper and that there "is a strong argument for using such a scenario in practice, provided that all technical problems (special printers and so) are solved" [43].

4.5 Conclusion

After careful analysis of numerous aspects surrounding the voting problem, we conclude that it will be difficult for Chaum's scheme, in its original proposed form, to be adopted in elections. The social, legislative, and economic factors surrounding election decisions limit the scheme's likelihood of implementation. However, the scheme is innovative and its central ideas are notably promising. It has shifted the focus of poll station e-voting research towards a new direction: that of provable security through cryptography. As with the development of any new technology,

refinement of an original idea is necessary in order to improve upon its weaknesses. Chaum's scheme is of great importance, and its simpler variants seem to be headed toward acceptance for real elections. The scheme already meets a strong definition of security. Consequently, if a variant is able to sufficiently address the non-technical concerns, in particular the issue of complexity, it will likely be considered for adoption in a real, binding, large-scale governmental election.

BIBLIOGRAPHY

[1] M. Abe. *Mix-Networks on Permutation Networks*. In ASIACRYPT '99, pgs. 258-273. Springer-Verlag, LNCS 1716, 1999.

[2] M. Abe and F. Hoshino. *Remarks on Mix-Network Based on Permutation Networks*. In PKC '01, pgs. 317-324. Springer-Verlag, LNCS 1992, 2001.

[3] M. Abe. *Universally Verifiable Mix-net with Verification Work Independent of the Number of Mix-servers*. In EUROCRYPT '98, pgs. 437-447. Springer-Verlag, LNCS 1403, 1998.

[4] A. Acquisti. *Receipt-free Homomorphic Elections and Write-in Ballots*. Technical Report TR-2004/105, International Association for Cryptologic Research, 2004. Available at http://eprint.iacr.org/2004/105.

[5] R. Aditya, C. Boyd, E. Dawson, K. Viswanathan. *Secure E-Voting for Preferential Elections*. In EGOV '03, pgs. 246-249. Springer-Verlag, LNCS 2739, 2003.

[6] J. Adler, W. Dai, R. Green, C. A. Neff. *Computational Details of the VoteHere Homomorphic Election System*. Available at http://www.votehere.net.

[7] J. Bannet, D. Price, A. Rudys, J Singer, and D. Wallach. *Hack-A-Vote: Security Issues with Electronic Voting Systems*. IEEE Security and Privacy, pgs.32-37. Vol. 2 No. 1, Jan/Feb 2004.

[8] L. Barlow. *A Discussion of Cryptographic Protocols for Electronic Voting*. Oregon State University, 2003. Available at http://islab.oregonstate.edu/koc/ece399/f03/final/barlow2.pdf.

[9] O. Baudron, P.-A. Fouque, D. Pointcheval, G. Poupard, and J. Stern. *Practical Multi-Candidate Election System*. In PODC '01, pgs. 274-283. ACM Press, 2001.

[10] B. Bederson and P. Herrnson. *Usability Review of the Diebold DRE System for Four Counties in the State of Maryland*. Available at http://www.capc.umd.edu/rpts/MD_EVoteMach.pdf.

[11] J. Benaloh and D. Tuinstra. *Receipt-Free Secret-Ballot Elections*. In STOC '94, pgs. 544-553, 1994.

[12] J. Benaloh and M. Yung. *Distributing the Power of a Government to Enhance the Privacy of Voters*. In PODC '86, pgs. 52-62. ACM Press, 1986.

[13] J. Benaloh. *Verifiable Secret-Ballot Elections*. PhD Thesis, Yale University, Department of Computer Science, September 1987.

[14] D. Boneh and P. Golle. *Almost Entirely Correct Mixing With Applications to Voting*, In Proc. of the 9th ACM-CCS Conference, pgs. 68-77. ACM Press, 2002.

[15] S. Bruck, D. Jefferson, and R. Rivest. *A Modular Voting Architecture ("Frogs")*. Caltech/MIT Voting Technology Project working paper, 2001. Available at http://www.vote.caltech.edu/Reports/vtp_WP2.pdf.

[16] J. Bryans and P. Ryan. *A Dependability Analysis of the Chaum Voting Scheme*. Technical Report CS-TR-809, Newcastle University School of Computing Science, 2003.

[17] M. Burmester and E. Magkos. *Towards Secure and Practical E-Elections in the New Era*. In Advances in Information Security, Vol. 7: **Secure Electronic Voting**, pp. 63-76. Kluwer Academic Publishers, 2003.

[18] J. Camp, A. Friedman, and W. Bowman. *Voting, Vote Capture, and Vote Counting Symposium, Electronic Voting Best Practices: A Summary*. 2004. Available at http://www.ljean.com/files/ABPractices.pdf.

[19] D. Chaum. *Secret Ballot Receipts: True Voter-Verifiable Elections*. IEEE Security and Privacy, 2(1): pgs. 38-47, Jan/Feb 2004.

[20] D. Chaum. Presentation: *Secret-Ballot Receipts: True Voter-Verifiable Elections*. DIMACS Workshop on Electronic Voting, 2004. Available at http://www.dimacs.rutgers.edu/Workshops/Voting/slides/chaum.ppt.

[21] D. Chaum. *Elections with Unconditionally Secure Ballots and Disruption Equivalent to Breaking RSA*. In EUROCRYPT '88, pgs. 177-182. Springer-Verlag, LNCS 330, 1988.

[22] D. Chaum. *Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms*. Communications of the ACM, 24(2), pgs. 84-88, 1981.

[23] D. Chaum. *Blind signatures for Untraceable Payments*. In Advances of Cryptology - Crypto '82, pgs. 199-203. Plenum Press, 1983.

[24] D. Chaum, P. Ryan, S. Schneider. *A Practical, Voter-verifiable Election Scheme*. CS-TR-880, Newcastle School of Computing Science, 2004. Available at http://www.cs.ncl.ac.uk/research/pubs/trs/papers/880.pdf.

[25] J. Cohen and M. Fischer. *A Robust and Verifiable Cryptographically Secure Election Scheme*. In FOCS '85, pgs. 372-382. IEEE, 1985.

[26] J. Cohen. *Improving Privacy in Cryptographic Elections*. Ph.D. Dissertation, Yale University, Dept. Of Computer Science, 1986.

[27] R. Cramer, M. Franklin, B. Schoenmakers, and M. Yung. *Multi-Authority Secret-Ballot Elections with Linear Work*. In EUROCRYPT '96, pgs. 72-83. Springer-Verlag, LNCS 1070, 1996.

[28] R. Cramer, R. Gennaro, and B. Schoenmakers. *A Secure and Optimally Efficient Multi-Authority Election Scheme*. In EUROCRYPT '97, pgs. 103-118. Springer-Verlag, LNCS 1233, 1997.

[29] L. Cranor and R. Cytron. *Design and Implementation of a Security-Conscious Electronic Polling System*. Washington University Computer Science Technical Report WUCS-96-02, 1996.

[30] L. Cranor. *Electronic Voting: Computerized Polls May Save Money, Protect Privacy*. ACM Crossroads Student Magazine, 2(4), 1996. Available at http://www.acm.org/crossroads/xrds2-4/voting.html.

[31] L. Cranor. *Voting after Florida: No Easy Answers*. Ubiquity, 1(47). ACM Press, 2001.

[32] I. Damgård and M. Jurik. *A Generalisation, a Simplification and Some Applications of Pallier's Probabilistic Public-Key System*. In PKC '01, pgs. 119-136. Springer-Verlag, LNCS 1992, 2001.

[33] D. Dill, R. Mercuri, P. Neumann, and D. Wallach. *Frequently Asked Questions about DRE Voting Systems*. Available at http://www.verifiedvoting.org/drefaq.asp.

[34] G. Dini. *A Secure and Available Electronic Voting Service for a Large-scale Distributed System*. In Future Generation of Computer Systems, 19(1), pgs. 69-85, 2003.

[35] B. DuRette. *Multiple Administrators for Electronic Voting*. Bachelors Thesis, Dept. of Computer Science and Electrical Engineering, MIT, 1999.

[36] D. Evans and N. Paul. *Election Security: Perception and Reality*. IEEE Security and Privacy, pgs. 24-31, Jan/Feb 2004.

[37] P.-A. Fouque, G. Poupard, and J. Stern. *Sharing Decryption in the Context of Voting or Lotteries*. In FINANCIAL CRYPTO '00, pgs.90-104. Springer-Verlag, LNCS 1962, 2000.

[38] A. Fujioka, T. Okamoto, and K. Ohta. *A Practical Secret Voting Scheme for Large Scale Elections*. In AUSCRYPT '92, pgs. 244-251. Springer-Verlag, LNCS 718, 1992.

[39] J. Furukawa and K. Sako. *An Efficient Scheme for Proving a Shuffle*. In CRYPTO '01, pgs. 368-387. Springer-Verlag, LNCS 2139, 2001.

[40] J. Furukawa, H. Miyauchi, K. Mori, S. Obana, K. Sako. *An Implementation of a Universally Verifiable Electronic Voting Scheme Based on Shuffling*. In Financial Crypto '02, pgs. 16-30. Springer-Verlag, LNCS 2357, 2002.

[41] E. Gerck, C. A. Neff, R. Rivest, A. Rubin, and M. Yung. *The Business of Electronic Voting*. In Financial Crypto '02, pgs. 243-268. Springer-Verlag, LNCS 2339, 2002.

[42] P. Golle, S. Zhong, D. Boneh, M. Jakobsson, and A. Juels. *Optimistic Mixing for Exit-Polls*. In ASIACRYPT '02, pgs.451-465. Springer-Verlag, LNCS 2501, 2002.

[43] M. Gomulkiewicz, M. Klonowski, and M. Kutylowski. *Rapid Mixing and Security of Chaum's Visual Electronic Voting*. In ESORICS '03, pgs. 132-145. Springer-Verlag, LNCS 2808, 2003.

[44] D. Gritzalis (Ed.). **Secure Electronic Voting**, Advances in Information Security, Vol. 7. Kluwer Academic Publishers, 2003.

[45] D. Gritzalis. *Principles and Requirements for a Secure E-Voting System*. In Computers & Security, pgs. 539-556. Vol. 21 No.6, 2002.

[46] M. Herschberg. *Secure Electronic Voting Using the World Wide Web*. Masters Thesis, Dept. of Electrical Engineering and Computer Science, MIT, 1997.

[47] M. Hirt and K. Sako. *Efficient Receipt-free Voting Based on Homomorphic Encryption*. In EUROCRYPT '00, pgs. 539-556. Springer-Verlag, LNCS 1807, 2000.

[48] M. Hout, L. Mangels, J. Carlson, and R. Best. *Working Paper: The Effect of Electronic Voting Machines on Change in Support for Bush in the 2004 Florida Elections.* 2004. Available at http://verifiedvoting.org/downloads/election04_WP.pdf.

[49] K. P. Iversen. *A Cryptographic Scheme for Computerized General Elections*. In CRYPTO '91, pgs. 405-419. Springer-Verlag, LNCS 812, 1992.

[50] M. Jakobsson, A. Juels, and R. Rivest. *Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking*. In USENIX '02, pgs. 339-353, 2002.

[51] M. Jakobsson. *A Practical Mix*. In EUROCRYPT '98, pgs. 448-461. Springer-Verlag, LNCS 1403, 1998.

[52] M. Jakobsson. *Flash Mixing*. In PODC '99, pgs. 83-89. ACM, 1999.

[53] M. Jakobsson and A. Juels. *Millimix: Mixing in Small Batches*. DIMACS Technical Report 99-33, 1999.

[54] D. Jefferson, A. Rubin, B. Simons, and D. Wagner. *A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE)*. 2004. Available at http://www.servesecurityreport.org.

[55] D. Jones. *A Brief Illustrated History of Voting*. Updated in 2003. Available at http://www.cs.uiowa.edu/~jones/voting/pictures.

[56] A. Juels and M. Jakobsson. *Coercion-Resistant Electronic Elections*. 2002. Available at http://www.eprint.iacr.org/2002/165.

[57] J. Katz, S. Myers, R. Ostrovsky. *Cryptographic Counters and Applications to Electronic Voting*. In EUROCRYPT '01, pgs. 78-92. Springer-Verlag, LNCS 2045, 2001.

[58] A. Kiayias and M. Yung. *The Vector-Ballot E-Voting Approach*. In Financial Crypto '04, pgs. 72-89. Springer-Verlag, LNCS 3110, 2004.

[59] A. Kiayias and M. Yung. *Self-Tallying Elections and Perfect Ballot Secrecy*. In PKC '02, pgs. 141-158. Springer-Verlag, LNCS 2274, 2002.

[60] T. Kohno, A. Stubblefield, A. Rubin, and D. Wallach. *Analysis of an Electronic Voting System*. Johns Hopkins Information Security Institute Technical Report TR-2003-19, 2003.

[61] B. Lee and K. Kim. *Receipt-free Electronic Voting Scheme with a Tamper-Resistent Randomizer*. In ICISC '02, pgs. 405-422, 2002.

[62] B. Lee and K. Kim. *Receipt-free Electronic Voting Through Collaboration of Voter and Honest Verifier*. Proceedings of JW-ISC2000, pgs. 101-108, 2000. Available at http://www.citeseer.nj.nec.com/lee00receiptfree.html.

[63] F. Lehoucq. *Electoral Fraud: Causes, Types, and Consequences.* Annual Review of Political Science, pgs. 233-256. Vol. 6, June 2003.

[64] E. Magkos, M. Burmester, V. Chrissikopoulos. *Receipt-freeness in Large-Scale Elections Without Untappable Channels*. In IFIP Conference on E-Commerce/E-businesss/E-Government, pgs. 683-694. Kluwer Academic Press, 2001.

[65] M. McCarthy. *Machine Error Gave Bush Extra Ohio Votes.* Associated Press, November 5, 2004. Available at http://www.truthout.org/docs_04/110604W.shtml.

[66] M. McGaley. *Report on DIMACS Workshop on Electronic Voting - Theory and Practice*. 2004. Available at http://dimacs.rutgers.edu/Workshops/Voting/e-voting-final.pdf.

[67] R. Mercuri. *A Better Ballot Box*? IEEE Spectrum, Vol. 39, pgs. 46-50, 2002.

[68] M. Michels and P. Horster. *Some Remarks on a Receipt-free and Universally Verifiable Mix-Type Voting Scheme*. In ASIACRYPT '94, pgs. 125-132. Springer-Verlag, LNCS 1163, 1996.

[69] M. Michels and P. Horster. *Cryptanalysis of a Voting Scheme*. In Communications and Multimedia Security II, pgs. 53-59. Chapman and Hall, 1996.

[70] M. Milroy. *The November 2nd, 2004 Election: Unresolved Questions.* Available at http://cagreens.org/longbeach/unresolved.htm.

[71] M. Naor and A. Shamir. *Visual Cryptography.* In EUROCRYPT '94, pgs. 1-12. Springer-Verlag, LNCS 950, 1995.

[72] C. A. Neff. *Detecting Malicious Poll Site Voting Clients*, 2003. Available at http://www.votehere.net/vhti/documentation/psclients.pdf.

[73] C. A. Neff. *Election Confidence: A Comparison of Methodologies and Their Relative Effectiveness at Achieving It.* 2003. Available at http://www.votehere.net/papers/ElectionConfidence.pdf.

[74] C. A. Neff. *A Verifiable Secret Shuffle and its Application to E-Voting.* In CCS '01, pgs. 116-125. ACM Press, 2001.

[75] C. A. Neff. *Conducting a Universally Verifiable Electronic Election Using Homomorphic Encryption.* 2000. Available at http://www.votehere.net.

[76] W. Ogata, K. Kurosawa, K. Sako, and K. Tatatani. *Fault Tolerant Anonymous Channel.* In ICICS '97, pgs. 440-444. Springer-Verlag, LNCS 1334, 1998.

[77] M. Ohkubo, F. Miura, M. Abe, A. Fujioka, and T. Okamoto. *An Improvement on a Practical Secret Voting Scheme.* In ISW '99, pgs. 225-234. Springer-Verlag. LNCS 1729, 1999.

[78] M. Ohkubo and M. Abe. *A length-invariant hybrid mix.* In ASIACRYPT '00, pgs. 178-191. Springer-Verlag, LNCS 1976, 2000.

[79] T. Okamoto. *Receipt-free Electronic Voting Schemes for Large Scale Elections.* In Security Protocols Workshop '97, pgs. 25-35. Springer-Verlag, LNCS 1361, 1997.

[80] T. Okamoto. *An Electronic Voting Scheme.* Proc. of IFIP '96, Advanced IT Tools, Chapman and Hall, pgs. 21-30, 1996.

[81] P. Paillier. *Public-Key Cryptosystems Based on Composite Degree Residuosity Classes.* In EUROCRYPT '99, pgs. 223-238. Springer-Verlag, LNCS 1592, 1999.

[82] C. Park, K. Itoh, and K. Kurosawa. *Efficient Anonymous Channel and All/Nothing Election Scheme.* In EUROCRYPT '93, pgs. 248-259. Springer-Verlag, LNCS 765, 1993.

[83] H. Peterson, P. Horster, and M. Michels. *Blind Multisignature Schemes and Their Relevance to E Voting.* In 11th Annual Computer Security Applications Conference, pgs. 149-155. IEEE Press, 1995.

[84] B. Pfitzmann. *Breaking an Efficient Anoynomous Channel.* In EUROCRYPT '94, pgs. 332-340. Springer-Verlag, LNCS 950, 1995.

[85] B. Pfitzmann, M. Waidner. *Unconditionally Untraceable and Fault Tolerant Broadcast and Secret Ballot Election*. Hildesheimer Informatik-Berichte, Institut für Informatik, Universität Hildesheim, 1992.

[86] M. Radwin. *An Untraceable, Universally Verifiable Voting Scheme*. 1995. Available at http://www.radwin.org/michael/projects/voting.pdf.

[87] R. Rivest. *Electronic Voting*. In Financial Cryptography '01, pgs. 243-268. Springer-Verlag, LNCS 2339, 2002.

[88] Z. Rjaskova. *Electronic Voting Schemes*. Masters Thesis, Dept. of Computer Science, Comenius University, 2002.

[89] A. Rubin. *Security Considerations for Remote Electronic Voting Over the Internet*. Communications of the ACM, Vol. 45, pgs. 39-44, 2002.

[90] P. Ryan and J. Bryans. *A Simplified Version of the Chaum Voting Scheme*. Technical Report CS-TR 843, Newcastle School of Computing Science, 2004.

[91] P. Ryan. *A Variant of the Chaum Voter-verifiable Scheme*. Technical Report CS-TR 864, Newcastle School of Computing Science, 2004. To appear in WITS 2005.

[92] P. Ryan. Presentation: *Towards a Dependability Case for the Chaum Voting Scheme*. DIMACS Workshop on Electronic Voting, 2004. Available at http://dimacs.rutgers.edu/Workshops/Voting/slides/ryan.ppt.

[93] K. Sako and J. Kilian. *Receipt-free Mix-type Voting Scheme*. In EUROCRYPT '95, pgs. 393-403. Springer-Verlag, LNCS 921, 1995.

[94] K. Sako. *Electronic Voting Schemes Allowing Open Objection to the Tally*. In Transactions of IEICE, pgs. 24-30. Vol. E77-A No.1, Jan. 1994.

[95] K. Sako and J. Kilian. *Secure Voting Using Partially Compatible Homomorphisms*. In CRYPTO '94, pgs. 248-259. Springer-Verlag, LNCS 839, 1994.

[96] B. Schneier. *Voting Security and Technology*. IEEE Security and Privacy, pg. 84, Vol. 2 No. 1, Jan/Feb 2004.

[97] B. Schoenmakers. *A Simple Publicly Verifiable Secret Sharing Scheme and its Application to Electronic Voting*. In CRYPTO '99, pgs. 148-164. Springer-Verlag, LNCS 1666, 1999.

[98] B. Schoenmakers. *Compensating for a Lack of Transparency*. In Proc. of 10th Conference on Computers, Freedom & Privacy, pgs. 231-233. ACM, 2000.

[99] M. Shamos. *Paper v. Electronic Voting Records - An Assessment*. 2004. Mimeo, Carnegie Mellon University. Available at http://euro.ecom.cmu.edu/people/faculty/mshamos/paper.htm.

[100] C.E. Shannon, *Communication Theory of Secrecy Systems*. Bell System Technical Journal, No. 28, pgs. 656-715, 1949.

[101] United States Congress. *The HAVA Bill of 2002*. Text available at http://www.fec.gov/hava/law_ext.txt.

[102] J. van de Graaf. *Adapting Chaum's Voter-Verifiable Election Scheme to the Brazilian System*. WSeg 2004 IV Workshop em Seguranç de Sistemas Computacionais, Brazil. Available at http://www.ppgia.pucpr.br/~maziero/pesquisa/wseg/2004/2958.pdf.

[103] P. Vora. *David Chaum's Voter Verification using Encrypted Paper Receipts*. 2004. Available at http://eprint.iacr.org/2005/050.pdf.

[104] D. Wikstrom. *A Universally Composable Mix-Net*. In TCC '04, pgs. 317-335. Springer-Verlag, LNCS 2951, 2004.