Ballot Casting Assurance via Voter-Initiated Poll Station Auditing

Josh Benaloh
Microsoft Research

June 28, 2007

Abstract

The technology for verifiable, open-audit elections has advanced substantially since research on this topic began a quarter century ago. Many of the problems are well-understood and have solid solutions. Ballot casting assurance — the problem of ensuring that a programmatically encrypted ballot matches the intentions of an individual human voter — has recently been recognized as perhaps the last substantial obstacle to making this technology fully viable. Several clever schemes have been developed to engage humans in interactive proofs to challenge and check validity of each ballot cast, but such a high standard may be neither practical nor necessary. If done properly, substantial integrity can be obtained by giving voters and observers the *option* to challenge ballot validity without requiring all voters to do so. This option can be made unobtrusive so as to not interfere with the normal process for most voters, but there are numerous risks and subtleties that necessitate a careful examination of the process. This paper identifies some heretofore unobserved issues with this "simple" method of casting ballots and describes a detailed process that mitigates all known threats. In doing so, it provides a blueprint for how verifiable, open-audit elections can reasonably be conducted in practice.

1 Introduction

Verifiable, open-audit election technologies allow voters to ensure that their own votes are cast as intended and that *all* votes are counted as cast. The "counted as cast" portion of this process has been the subject of cryptographic inquiry since the early 1980s and a plethora of strong and practical methods are available. (See [Chau81], [DLM82], [CoFi85], [Bena87], [PIK93], [BeTu94], [SaKi95], [CGS97], [BJR01], [FuSa01], [Neff01], [GZBJJ02], [JJR02], [Grot03], [Chau04], [Furu04], [Chau05], [CRS05], [PBD05], and [Bena06] for a select sampling of work in this area.) Although there are some notable exceptions, the process frequently begins with voters encrypting their votes and then submitting their encrypted votes via a public mechanism. These encrypted votes are then counted by a cryptographic process which produces the tally together with a proof of its correctness — while protecting the privacy of individual voters.

The step in which voters transform their intentions into encrypted votes has proven to be far more stubborn than initially expected. If voters use their own devices to capture and encrypt their intentions, then they are subject to coercion as well as the risk of viruses and other threats to the integrity of their devices. If voters use devices managed by entities of their choosing, many of the integrity threats can be mitigated, but voters are still subject to coercion. To prevent coercion, voters should only use devices that are under the control of election authorities, but how then are voters to gain assurance that their intentions are being accurately captured?

Several recent schemes have been devised to engage voters in interactive proofs to verify the integrity of votes that have been encrypted on their behalf. A great deal of cleverness has been applied to reduce the burdens that would normally be born by voters involved in a cryptographic exchange. Neff's "MarkPledge" scheme ([Neff04]) achieves very high assurance — enabling each voter to detect any malfeasance on the part of a voting device with probability well in excess of 99%.[1] Chaum and others have introduced several

[1] The notion of ballot casting assurance was formalized in [AdNe06].

designs that enable detection of malfeasance with 50% probability (see, for instance, [Chau04] and [Chau05]). Although 50% assurance on each ballot may seem low, it still means that any attempt to alter anything more than a very small number of votes would almost certainly be detected. The primary drawback of these designs is that although they provide very strong assurance, they require voters to do more than what is required to cast a vote in a traditional system – in some cases, these extra burdens are fairly cumbersome.

Another approach to ballot casting assurance is to audit sample ballots which are *not* cast. Assurance is derived from the random selection of which ballots to challenge. The level of assurance varies with the fraction of ballots challenged. This approach is used in the Prêt à Voter scheme of Chaum, Ryan, and Schneider ([CRS05]) and in the *direct ballot casting* scheme of [Bena06]. Their common reliance upon auditing of uncast ballots is one of the few similarities between these two schemes.

Prêt à Voter is paper-based, restricts the form of the ballot to a list of options which must be independently randomized on each ballot, and performs auditing on blank ballots. Direct ballot casting is designed for use with electronic voting systems (e.g. *Direct Recording Electronic* (DRE) Systems), does not restrict the form of ballots, and performs auditing on fully marked ballots. There are numerous trade-offs between paper-based and electronic voting systems, and there is clear benefit to having solid verifiable voting technologies for both kinds of system.

The direct ballot casting approach taken in [Bena06] requires virtually nothing more of voters than that to which they are already accustomed. Instead of requiring voters to take steps to verify the accuracy of their ballots, voters are unobtrusively offered an option to do so. As with Prêt à Voter, the level of assurance achieved depends upon the number of voters (and others[2]) who choose to participate in a validation process. Furthermore, this validation process is extremely simple and intuitive. Even if only 1% of ballots are verified, then malicious devices would be very unlikely to succeed in altering more than a few hundred ballots without being detected. This can make a significant difference in a small election; but in an election with 100 million ballots cast, there is virtually no chance of plausibly altering enough votes to affect the outcome — especially since any such alterations would have to be attempted at ballot casting time, well before any tallying.

While direct ballot casting is simple and effective, the approach is somewhat fragile and the process must be carefully delineated. For example, if a voting device knows or has reason to believe that a particular ballot is unlikely to be audited, then it can alter that ballot at will. For this reason, it is essential that the voting device not have any information on the identity of a voter or other user during an election. There are other threats to the integrity of ballots produced in this manner, and this paper will identify several such threats and describe a process to circumvent all identified threats

2 Direct Ballot Casting

Abstractly, using voter-initiated auditing as suggested in [Bena06] to achieve ballot casting assurance is a simple process. Voters and other interested parties arrive at a polling station and use an interactive ballot encryption device[3] to obtain one or more encrypted ballots. No identification is necessary to use these ballot encryption devices, and they may be used as many times as desired by any individual (subject, of course, to resource constraints). Subsequently, eligible voters cast encrypted ballots by identifying themselves to poll workers through a sign-in process and providing (one of) the encrypted ballots produced by a ballot encryption device. Any uncast ballot can be *opened* to provide assurance as to its legitimacy.[4]

While this process seems simple, closer examination reveals numerous subtle risks — both procedural and security-related.

2.1 Process Issues

One principal incompatibility between this basic audit process and established practice is that polling sites for multiple precincts are frequently co-located requiring multiple ballots to be offered at the same site. Even where precincts are not

[2]As will be further explained later, officials, observers, and any voter or non-voter who cares to do so can increase assurance by verifying ballots at will.

[3]Such a device may appear much like a Direct Recording Electronic (DRE) device that many voters are accustomed to today.

[4]The detailed mechanism for opening an encrypted ballot depends on the specifics of the encryption which, in turn, depends upon the ballot counting system to be used. An encrypted ballot can either be opened immediately by the ballot encryption device that created it or at a later time by *election trustees.*

co-located, different ballots may be required for different voters — for instance during primaries in jurisdictions where voters need to be registered with a political party in order to participate.

To accommodate multiple ballot choices, it is not reasonable in practice to assume that voters will be able to obtain correct ballots without identifying themselves to poll workers, and this must be adequately addressed by any serious proposal.

Additionally, there must be a suitable medium for carrying encrypted ballots from the ballot encryption device to poll workers and to enable the voter and poll worker to each retain a copy of any encrypted ballot that is cast.

Finally, it may not be practical to expect voters to understand an audit process. To audit a particular ballot, a voter may need to either indicate to a poll worker that a given ballot is to be audited and wait for the ballot to be opened by other sources at a later time or return to the same ballot encryption device which was used to create the ballot (there may well be more than one such device in a poll site) and present the ballot to have it opened on the spot. These processes may be excessively cumbersome — even though only a small fraction of voters would be likely to choose to engage in them.

2.2 Integrity

To maintain integrity within this design, it is crucial that a ballot encryption device not receive any information that may be used to identify a voter or indicate that a particular ballot may be more or less likely to be audited.

A device that is certain that a particular ballot will not be audited is free to alter the contents of that ballot at will. Even a statistical advantage is sufficient to call the integrity of an election into question.

It is therefore essential that ballot encryption devices not have access to voter identities and that any communications with these devices be strictly limited to prevent them from receiving external information that may indicate the likelihood of particular user auditing ballots.

2.3 Coercion Issues

Coercion and vote-buying in elections can be subtle and insidious. Some tricks are well-known, but vigilance is particularly required whenever a new mechanism is considered.

2.3.1 Unauthorized Encryption

A simple technique for vote buying or coercion is to encrypt ballots off-site — out of the control of voting authorities. If voters can be given their encrypted ballots in advance by a coercive agent, the agent can simply check the list of encrypted ballots to ensure that these ballots have been cast.

This is a simple attack which must be addressed by any election system based upon public display of encrypted ballots.

2.3.2 Chain Voting

A well-known risk of many voting technologies is chain voting. A vote-buyer obtains a blank ballot — perhaps by signing in as a legitimate voter — and then completes the ballot as desired. The vote-buyer then leaves the polling site with this ballot and goes to a pre-arranged location. Vote sellers arrive periodically at the arranged location and each receive a completed ballot. Each vote seller takes a completed ballot into the poll site, signs in, and receives a blank ballot. The voter does *not* complete this ballot. Instead, after a suitable delay, the voter casts the completed ballot received from the vote buyer. The voter then leaves the poll site with the blank ballot and returns this blank ballot to the vote buyer in exchange for cash or considerations — enabling the chain to be continued at will.

A voter buyer can check public voting records after the fact to ensure that vote sellers actually cast their completed ballots rather than spoiling them.

2.3.3 One Balloter — Many Voters

Even if all ballots are properly encrypted on devices under the control and observation of poll workers, the basic auditing design does not limit the use of these devices. Thus, a single individual could create many individual ballots and then give one to each of many legitimate voters. The coercer or vote-buyer can then check that these ballots are included in the tally and act accordingly.

This attack is similar to chain voting but doesn't require chains since a coercer or vote-buyer is explicitly allowed to create as many completed ballots as desired.

2.3.4 Commitment Coercion

Many election schemes based upon encryption include steps where commitments are made as to

the contents of ballots and voters are asked to issue *random* challenges in response to those commitments. However, forcing a voter to make a challenge decision based upon a commitment value creates an opportunity for coercion. For instance, if an encryption device commits to a particular encrypted ballot, the coerced voter can be instructed to compute a function based on the encrypted ballot and to use the result of this function as a challenge. The detailed instantiation varies depending upon the scheme used.

It may seem that such coercion is not an issue with schemes based on ballot auditing since there is no explicit voter challenge. But there is an implicit voter challenge in the decision of which ballot to cast.

For example, if the number of encrypted ballots created by a voter in a direct ballot casting system can be observed (perhaps by observing the time spent using a ballot encryption device), then a coercer can employ the following strategy. The coercer instructs the voter to create a ballot according to the voting wishes of the coercer. If the encrypted ballot produced by the device satisfies a particular predicate (for instance, if the encrypted ballot value is odd), then the instructions are to cast that ballot. If the ballot does not satisfy the predicate, then the instructions are to have it opened and to create and cast a second ballot according to the wishes of the coercer — regardless of the second ballot's value.

A coerced voter might be able to cast a ballot that does not match the wishes of the coercer, but the voter entails some risk of exposure in doing so. For instance, one strategy for a voter would be to encrypt a ballot according to the voter's own wishes and then compute the predicate. If this ballot satisfies the predicate, the voter can then cast it without being exposed. If, however, the predicate fails, the voter will not be able to have this ballot opened without its contents being exposed to the coercer. The voter might attempt to create a second ballot (this time according to the wishes of the coercer). If the predicate is *not* satisfied by the second ballot, the voter *might* be able to switch the two ballots, have the second opened, and cast the first.[5] If, however, the second ballot satisfies the predicate, the voter will be exposed as having disobeyed the coercer.

If the coercer chooses a predicate with a probability of p, a coerced voter that attempts to disobey the coercer according to the above strategy will be exposed with probability $p(1-p)$. Other strategies are possible, but within this context a coercer can always ensure there is at least a 25% chance of exposing any attempt by a voter to defy the coercion, and this probability is likely more than sufficient to achieve good compliance.

As with other risks, this threat must be acknowledged and mitigated by any credible ballot casting system.

[5] This is effective only if the ballots contain no public timestamp or other ordering information.

3 Voter-Initiated Auditing: The Process in Detail

The following process is designed to address all of the threats and vulnerabilities in direct ballot casting identified above. It is believed to offer a complete solution to the ballot casting problem which is unobtrusive to voters and enables strong guarantees of election integrity, protection from coercion, and unrestricted ballot form. As with all ballot integrity systems based upon auditing of uncast ballots, the level of assurance depends upon the number of auditing events that take place. If no ballots are challenged, there is no reason to have any faith in the integrity of the election tally. However, audit events are so simple that any voter, observer, or official inspector can trigger an audit event by just creating an encrypted ballot and then having the ballot opened. This can be done at any time and at any polling site during an election. Even a modest number of audit events can create an extremely high level of assurance in the integrity of a large election.

The process described herein seems simple, but it is also quite fragile. Small changes in the details or process order can enable some of the potential vulnerabilities that have been identified.

The process assumes the availability of ballot encryption devices that could appear to users as virtually identical to typical electronic voting devices in common use. These devices should have displays capable of presenting a ballot to a voter and means for accepting preference information from voters. These ballot encryption devices should be capable of receiving and reading *ballot-type* cards which can convey information about the ballot type to be used and should also be able to write an encrypted ballot onto these cards. They should also have paper printers or other similar means of providing voters with short receipts. The printer should be constructed so that a receipt can be partially printed without the voter being able

to see what has been printed.[6] Vote encryption devices should be configured with the public key(s) necessary to encrypt ballots as required by the ballot counting scheme to be used and should each contain an individual authenticated signature key (one per device). Ballot encryption devices should include counters which can be incremented whenever a ballot is created.

Poll workers should have a supply of (blank) ballot-type cards. They should also have access to appliances capable of reading from and writing onto these cards. The card appliances should include counters which are incremented with each use as well as displays that can present these counter values to poll workers.

A voter proceeds as follows.

1. After arriving at a polling site, a voter goes to a poll worker and is identified to determine the appropriate ballot type.

2. The poll workers prepares an appropriate ballot-type card for the voter which should include the voter's ballot-type information together with the current value of the counter in the card reader/writer appliance. The poll worker records the counter value and gives the card to the voter.[7]

3. Next, the voter proceeds to the vote encryption device and inserts the ballot-type card.

4. The voter is then presented with a ballot of the type indicated on the card and interacts with the device to select preferences.

5. Upon completion of this interaction, the voter is asked whether the voter is finished or wishes to make changes. When a voter indicates having finished, the voter's selections are encrypted with the public key(s) of the election and the encrypted ballot (or a short cryptographic hash thereof) is printed on the paper receipt but not yet visible to the voter.

6. The voter is then asked whether this vote should be cast.

   - If the voter responds "yes", the device digitally signs the encrypted vote together with the counter value on the ballot-type card and its own counter value, writes the signed value onto the card, and also adds the signature to the paper receipt which is then made available to the voter.[8]
   - If the voter responds "no", the device opens the ballot by printing on the receipt the raw contents of the ballot together with any random seed data used in the encryption process.

7. A voter wishing to cast the ballot removes the card, returns to the poll worker who, if the device digital signature verifies[9] and the application counter matches the value previously entered, records the encrypted ballot from the card as corresponding to the voter, and erases the signed values from the card to enable reuse. The voter may take the paper receipt(s) home.

The encrypted ballots recorded by poll workers should be posted on a public site — enabling voter verification of these ballots against the encrypted ballots on each receipt.[10] Failure to post these encrypted ballots is evidence (although not proof) of improper poll worker practice, since voters might sometimes fail to return their authenticated ballot cards to poll workers.

The encrypted ballots can be processed using any of a wide variety of encrypted ballot counting protocols. Such protocols are well understood and independent of the results described herein.

4 Resistance to Threats

How does this approach stand up to the potential vulnerabilities and concerns previously identified?

Process Issues Since legitimate voters visit poll workers prior to visiting ballot encryption devices, and since poll workers identify voters and provide tokens with appropriate ballot-type information,

[6] This may be accomplished by printing the receipt behind an opaque screen or face down beneath a transparent barrier.

[7] A voter, observer, or inspector may ask for any ballot type at any time during the process, but only legitimate voters will be able to cast votes and only of the types to which they are entitled.

[8] Note that signing the counter values together with the encrypted ballot does *not* threaten voter privacy since all of this data will be associated with an identified voter anyway. It is the encryption of the ballot that protects voter privacy, and the identifying data will be removed as part of the subsequent vote counting process that tallies the results without providing direct decryptions.

[9] The signature verification could optionally be preformed later as part of the ballot counting process.

[10] The posted ballots can include full voter identification or, depending upon local requirements, may simply be identified by posted sequence counter values.

the process of ensuring that voters are using the correct ballots is similar to that in common current practice.

These ballot-type cards also provide a means of conveying encrypted votes from the vote encryption devices to poll workers.

Finally, although the auditing process is sufficient to provide strong integrity guarantees, it is largely implicit. Voters who do not care to participate in the audit process do not even see it, and any one who cares to initiate an audit event can easily do so and leave with a receipt which can be checked at any subsequent time by third parties, independently-written software, or directly by a voter.

Integrity To maintain integrity, it is essential that ballot encryption devices not receive any information about voter identity. The devices are isolated and should have no means of remote input.

There does exist a channel by which a device could receive information about voters. The cards used to indicate ballot types could be rigged to include information about whether or not a voter is likely to initiate an audit event. For instance, a poll worker could have two varieties of cards — one of which is used when, in the poll worker's estimation, the voter is unlikely to initiate an audit, and one to indicate that the ballot should be properly encrypted.

While such an attack is possible in theory, it is a retail attack that would require collusion between many poll workers and at least one equipment manufacturer to have the potential to influence a large election. Furthermore, an attack of this sort would leave substantial evidence since there would need to be numerous cards in many hands that could each trigger device malfeasance. Any such card could be taken to an affected device during an election to demonstrate improper performance.

Unauthorized Encryption Since all ballots produced on authorized ballot encryption devices include authenticated digital signatures, any cast ballots are verified as originating in an appropriate ballot encryption device.[11]

Chain Voting Ordinary chain voting is not an issue in this design since it involves unauthorized marking of blank ballots which is prevented by the use of digital signatures.

One Balloter — Many Voters The attack wherein a single individual can create many legitimate ballots and distribute them to many voters is prevented by the secondary use of the ballot-type cards. An encrypted ballot that is eligible for casting must be presented on the ballot-type card, and these cards can be limited to carry only one vote at a time. Non-voting observers and suspicious voters may also obtain ballot-type cards, but only one at a time, and these cards should be returned to poll workers whether or not a vote is cast.

A more elaborate attack is effected by an attempt to create duplicate cards that can be used to carry authorized (signed) votes out of a poll site to be given to additional (coerced) voters. This is prevented by the signed sequential counter values written onto the ballot-type cards. If the signed counter value does not match the counter value issued to the voter on the ballot-type card (and recorded by the poll worker when the card was issued to the voter), the encrypted ballot should not be accepted.[12]

Commitment Coercion This attack is mitigated by the physical barrier (either an opaque screen or face down printing under glass) to prevent a voter from seeing the encrypted ballot data before deciding whether or not a ballot is to be marked as for casting.

This threat remains very subtle, and, on first analysis, it may seem as though this mitigation is inadequate. After all, once an encrypted vote has been written onto the ballot-type card, a voter will see the paper receipt and can still take action based on the arithmetic value of the encrypted ballot. The vote will not be cast until the voter returns the card to a poll worker, and the voter could still choose not to do so (perhaps by de-

[11] A signing key extracted from an authorized device could be used to impersonate an authorized device. Appropriate security should be in place to prevent private key disclosure. An insider attack could make signing keys available to unauthorized parties. However, this is just a special case of a broader class of attacks in which a voting device is corrupted by insiders to compromise voter privacy. All known voting devices are subject to such attacks by a variety of means including hidden cameras, invisible ink, fingerprinting, and numerous other technologies.

[12] It is not necessary that the poll worker appliance be capable of parsing and verifying a digitally signed counter value. The counter value can also be provided in the clear and recorded together with the signed vote. Subsequent integrity checks could be used to ensure that the digital signature verifies and any ballot that does not verify can be excluded at that time.

stroying the card, by claiming an error and asking to have the ballot spoiled in favor of a new one, or simply by leaving the poll site without returning to a poll worker at all). The fact, however, is that the risk is not precisely in a voter using the arithmetic value of an encrypted ballot as a basis to decide whether or not to cast the ballot. The actual risk is that a voter my use the arithmetic value of an encrypted ballot as a basis to decide whether or not to have a ballot opened. Under normal circumstances, these two decisions are the same since an encrypted ballot would normally either be cast or opened (and never both). However, the specific risk is that a coercer may be able to view the contents of an encrypted ballot by forcing a voter to open it rather than cast it. A coercer gets no advantage by forcing an unopened ballot to not be cast.[13]

The key here is that the voter's opportunity to have an encrypted ballot opened is limited to a period in which the voter cannot view the arithmetic value of the encrypted ballot. Preventing a voter from using this value to decide whether or not to have a ballot opened is sufficient to eliminate this threat.

Other Potential Attacks While there is no proof that this list of vulnerabilities is comprehensive, it seems to include all known threats that are unique to this voter-initiated auditing approach to ballot casting. No further vulnerabilities are known.

5 Conclusion

The detailed process for ballot casting via voter-initiated auditing described within this paper is practical and entirely consistent with current practices for both voters and poll workers. It does not restrict the ballot form in any way and is believed to address all known threats that might impact integrity or allow coercion. While the level of assurance is not as great as in some other ballot casting designs, it is still much greater than anything found in current use and should be more than adequate for most elections. Any one of numerous cryptographic ballot counting procedures can be used to complete an election.

[13] A coercer, of course, can intimidate a voter into not casting a ballot at all. This is a real threat in any election, but it is well beyond the scope of this work.

References

[AdNe06] **Adida, B.** and **Neff, C.A.** "Ballot Casting Assurance" *Proceedings of the 2006 Electronic Voting Technology Workshop.* Vancouver, BC (Aug. 2006). Available at http://usenix.org/events/evt2006/tech/.

[Bena06] **Benaloh, J.** "Simple Verifiable Elections" *Proceedings of the 2006 Electronic Voting Technology Workshop.* Vancouver, BC (Aug. 2006). Available at http://usenix.org/events/evt2006/tech/.

[Bena87] **Benaloh, J.** "Verifiable Secret-Ballot Elections." *Yale University Ph.D. Thesis YALEU/DCS/TR-561.* New Haven, CT (Dec. 1987).

[BeTu94] **Benaloh, J.** and **Tuinstra, D.** "Receipt-Free Secret-Ballot Elections" *Proceedings of the* 26^{th} *ACM Symposium on Theory of Computing.* Montreal, PQ (May 1994) 544–553.

[BJR01] **Bruck, S.**, **Jefferson, D.**, and **Rivest, R.** "A Modular Voting Architecture ("Frogs")." *Workshop on Theory of Elections.* Tomales Bay, CA (Aug. 2001).

[Chau04] **Chaum, D.** "Secret-Ballot Receipts: True Voter-Verifiable Elections." *IEEE Security & Privacy 2* 1, (Feb. 2004), 38–47.

[Chau05] **Chaum, D.** "Recent Results in Electronic Voting." *Fronteers in Electronic Elections.* Milan, Italy (Sep. 2005).

[Chau81] **Chaum, D.** "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms." *Communications of the ACM 24,* 2, (Feb. 1981), 84–88.

[CGS97] **Cramer, R.**, **Gennaro, R.**, and **Schoenmakers, B.** "A Secure and Optimally Efficient Mult-Authority Election Scheme." *Proceedings of Eurocrypt '97.* Konstanz, Germany (May 1997) 103–118.

[CoFi85] **Cohen (now Benaloh), J.** and **Fischer, M.** "A Robust and Verifiable Cryptographically Secure Election Scheme." *Proceedings of the* 26^{th} *IEEE Symposium on Foundations of Computer Science.* Portland, OR (Oct. 1985), 372–382.

[CRS05] **Chaum, D.**, **Ryan, P. Y. A.**, and **Schneider, S.** "A Pracital Voter-Verifiable Election Scheme." *Proceedings of European Symposium on Research in Computer Security.* Milan, Italy (Sep 2005) 118–139.

[DLM82] **De Millo, R.**, **Lynch, N.**, and **Merritt, M.** "Cryptographic Protocols." *Proceedings of the* 14^{th} *ACM Symposium on Theory of Computing.* San Francisco, CA (May 1982), 383–400.

[Furu04] **Furukawa, J.** "Efficient, Verifiable Shuffle Decryption and Its Requirement of Unlinkability." *Proceedings of PKC 2004.* Singapore (Mar. 2004) 319–332.

[FuSa01] **Furukawa, J.** and **Sako, K.** "An Efficient Scheme for Proving a Shuffle." *Proceedings of Crypto 2001.* Santa Barbara, CA (Aug. 2001) 368–387.

[Grot03] **Groth, J.** "A Verifiable Secret Shuffle of Homomorphic Encryptions." *Proceedings of PKC 2003.* Miami, FL (Jan. 2003) 145–160.

[GZBJJ02] **Golle, P.**, **Zhong, S.**, **Boneh, D.**, **Jakobsson, M.**, and **Juels, A.** "Optimistic Mixing for Exit-Polls." *Proceedings of Asiacrypt 2002.* Queenstown, New Zealand (Dec. 2002) 451–465.

[JJR02] **Jakobsson, M.**, **Juels, A.**, and **Rivest, R.** "Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking." *Proceedings of the 2002 USENIX Security Symposium.* San Francisco, CA (Aug. 2002), 339–353.

[Neff01] **Neff, C.A.** "A Verifiable Secret Shuffle and its Application to E-Voting." *Proceedings of the ACM Conference on Computer and Communications Security.* Philadelphia, PA (Nov. 2001), 116–125.

[Neff04] **Neff, C.A.** "Practical High Certainty Intent Verification for Encrypted Votes." Available at http://www.votehere.com/documents.php.

[PBD05] **Peng, K.**, **Boyd, C.**, and **Dawson, E.** "Simple and Efficient Shuffling with Provable Correctness and ZK Privacy." *Proceedings of Crypto 2005.* Santa Barbara, CA (Aug. 2005) 188–204.

[PIK93] **Park, C.**, **Itoh, K.**, and **Kurosawa, K.** "Efficient Anonymous Channel and All/Nothing Election Scheme." *Proceedings of Eurocrypt '93.* Lofthus, Norway (May 1993), 248–259.

[SaKi95] **Sako, K.** and **Kilian, J.** "Receipt-Free Mix-Type Voting Scheme." *Proceedings of Eurocrypt '95.* St. Malo, France (May 1995) 394–403.