Aperio: High Integrity Elections for Developing Countries

Aleks Essex[1], Jeremy Clark[2], and Carlisle Adams[1]

[1] University of Ottawa
aesse083@site.uottawa.ca,
cadams@site.uottawa.ca
[2] University of Waterloo
j5clark@cs.uwaterloo.ca

Abstract. This article presents an 'end-to-end' integrity verification mechanism for use in minimally equipped secret paper-ballot election environments. The scheme presented in this paper achieves high integrity properties without interfering with the traditional marking and tabulation procedures of paper-ballot elections. Officials and auditors can respectively generate and independently verify 'end-to-end' audit trails, with office stationery and entirely without cryptographic or mathematic computations.

1 Introduction

Aperio
-*verb* (Latin)
1. to reveal, uncover, lay bare.

Practical proposals for "end-to-end" election verification mechanisms have received much attention recently for their provision of election integrity *independent* of the physical chain-of-custody of ballots, achievable in part through the issuance of privacy-preserving receipts to voters [1,4,5,6,10]. These systems have been primarily cryptographic and largely inspired by the concept of anonymizing mixnets [3]. They also generally rely heavily on technology at the polling place and specialized technical knowledge for the verification process. These systems find their application predominantly in election environments with a pre-existing infrastructure of electronic (*e.g.,* optical scan) equipment. However the election environments which arguably would benefit the most from the end-to-end integrity properties—in developing democracies—are ones in which a technological infrastructure for voting is either not present, or not practical.

Three proposals were made in [14] for systems that do not directly utilize cryptography. However the first two proposals, ThreeBallot and VAV require the voter to mark the ballot in an arguably unintuitive way and still propose the use of some electronic equipment (to validate the ballot). The third and more elegant proposal, Twin, requires no electronic equipment or special ballot marking or

D. Chaum et al. (Eds.): Towards Trustworthy Elections, LNCS 6000, pp. 388–401, 2010.

tabulation procedures, although does carry an inherent custody assumption that the "floating receipt" being issued to the voter is a *valid copy* of the ballot of another (anonymous) voter. In response, we propose Aperio, a simple paper-based election integrity verification mechanism for minimally equipped election environments, similar in intent to Twin, but with end-to-end integrity properties that do not rely on chain-of-custody.

2 Preliminaries

2.1 Integrity Properties

"*End-to-end*" (E2E) verification, as initially proposed in [7] offers two positive guarantees of integrity to voters:

1. Their ballot is included unmodified in the *same* set of ballots that get tallied,
2. The tally produced from this set is correct.

A less stringent requirement, "*software independence*" (SI), has also been proposed: "A voting system is software-independent if an undetected change or error in its software cannot cause an undetectable change or error in the election outcome" [13]. By definition, this term is not applicable to environments which do not utilize software components. However the problem has a physical analog to the so-called "chain of custody" of ballots. For our purposes we will defined a new term "*custody independence*" (CI) to be defined as: "A voting system is custody independent if an undetected change or error in its physical paper trail cannot cause an undetectable change or error in the election outcome."

In informal terms, this requirement states that any entity that can manage to exchange the legitimate ballot box for another (rigged) one, cannot do so undetectably. Therefore any system that is end-to-end verifiable can be shown to exhibit the CI property. In the case of Twin [14], the first end-to-end criterion is not directly satisfied by the floating receipt model it employs.

2.2 Roles

In the following section we roughly define four sets of roles within the election, for which its members are not necessarily distinct (*e.g.,* a voter can also function as a verifier or poll official).

- **Election Trustees.** In addition to the existing requirements for the administration of a paper-ballot election, trustees oversee the generation, commitment and decommitment of the audit commitment lists of the corresponding ballots.
- **Verifiers.** An (unspecified) partnership of concerned entities, including potentially candidates and their representatives, non-governmental organizations, voter advocacy groups and other election observers act to verify the integrity (*i.e.,* correctness) of the election outcome *independently* of physical access to the official paper trail via the mechanism presented in the following section.

- **Voters.** Those who vote in the election. In addition to voting, voters are given the option of retaining a (privacy-preserving) receipt.
- **Poll Officials.** Those responsible for administering the election at the polling location with tasks that include voter authentication, ballot distribution and casting.

3 Basic Paper Scheme

3.1 Ballot Format

In a conventional paper-ballot election, a voter is issued a single sheet of paper—a ballot. Under the Aperio scheme, a voter is instead issued a "ballot assembly" which consists of a paper ballot sheet, a receipt sheet and any number of audit sheets stacked and joined in such a way that only the top ballot layer is visible to the voter. These layers are defined as follows:

- A "*ballot*" is used to describe any paper *Australian ballot*[1] with the specific property that the list of candidates or proposals is printed in an independently random order across the set of ballots in an election,
- a "*receipt*" is used to describe a sheet of paper, equivalent in size and layout to a "ballot," but without a candidate list, and additionally a unique serial number, and
- an "*audit sheet*" is used to describe a sheet of paper, equivalent in size and layout to a "ballot" but without a candidate list. It does not contain a serial number, but has pre-printed regions in which to write one. Additionally provided is a region in which to mark a "commitment reference number."

For simplicity of the following description, we will consider the base-case ballot assembly which includes *two* audit sheets. We will refer to the assembly's layers by a standard office-paper color pallet, in which the ballot, receipt, and two audit sheets are assigned the colors "white," "canary," "goldenrod," and "pink" respectively (see Figure 1). The sheets are stacked in such a way that voter-made marks on the ballot sheet will be transfered to the other sheets using carbon-copy, or alternatively NCR brand (carbonless copy) paper (see Figure 2).

3.2 Initial Setup

As in other of E2E systems [1,4,5,6,10], the entity responsible for printing ballots is generally entrusted with voter privacy. Although there likely exist protocols under which ballots could be printed in either a threshold-trust or even a fully oblivious manner, for simplicity we describe the following operations as being performed by a single privacy-entrusted entity.

[1] "An official ballot printed at public expense on which the names of all the candidates and proposals appear and which is distributed only at the polling place and marked in secret" [2].

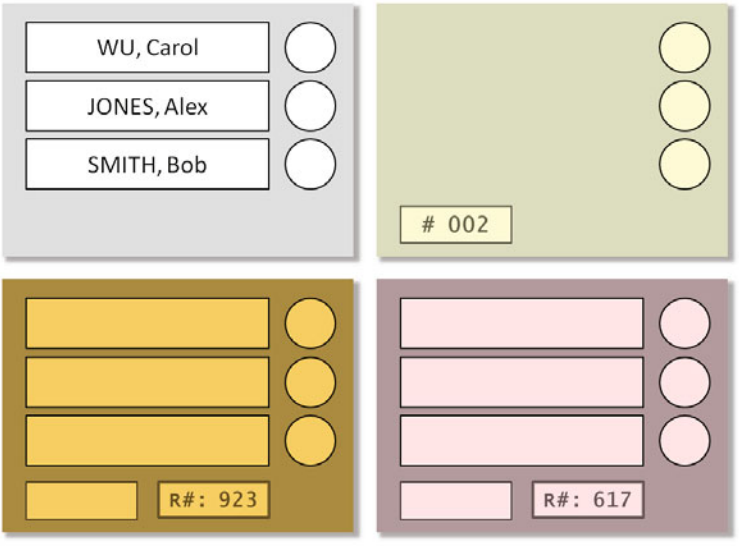


Fig. 1. Ballot Assembly: (Top left) Paper "Australian" ballot with random candidate order (*white*). **(Top right)** Receipt with unique serial number (*canary*). **(Bottom left)** Audit sheet with "commitment reference number" (*goldenrod*). **(Bottom right)** Audit sheet with independently assigned "commitment reference number" (*pink*).

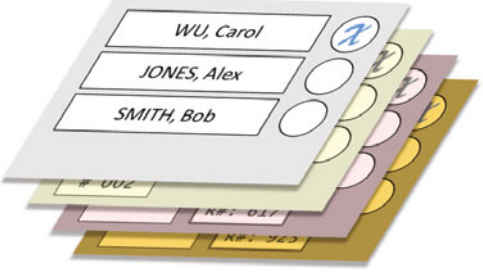


Fig. 2. Ballot Assembly (Exploded View): Marks made on the top "ballot" layer are transfered to the "receipt" and "audit sheet" layers using "carbon-copy" style paper

Generating Ballots. In order to preserve voter privacy, the association of candidate order and serial number must be secret and arbitrary, such that knowledge of one in no way implies knowledge of the other. Consider a stack of b ballots, each with an independently randomly printed candidate order. Likewise consider a stack of b receipts each with unique serial number. An arbitrary association can be formed by drawing the top ballot and receipt sheets from the respective stacks and joining (*e.g.,* stapling) them together. Additionally the (blank) audit sheets are joined to the ballot and receipt sheets, constituting a specific instance of a ballot assembly. This is repeated to create b independent ballot assemblies.

Generating Commitment Lists. A "commitment list" is defined as a list of b rows pre-printed with a monotonically increasing set of b "commitment reference numbers." Next to each commitment reference number is a region, initially blank, that will contain an associated value. We define two types of commitment lists:

- **Receipt commitment lists** contain a set of b distinct commitment reference numbers, each with a (randomly) associated *serial number*
- **Ballot commitment lists** contain the same set of b distinct commitment reference numbers, each with a (randomly) associated *candidate list ordering.*

To generate the commitment lists, we begin by considering a particular audit trail color (*e.g.,* pink). The *pink receipt commitment list* and *pink ballot commitment list* are generated as follows:

1. A ballot assembly is drawn from the stack and the serial number s is noted. A non-replaced random number $i \in b$ is selected (*e.g.,* on a slip of paper drawn from a hat),
2. The pink audit sheet of the ballot is exposed, and the number i is written in the commitment reference number space,
3. On the *pink receipt commitment list*, the number s is written in the blank space beside commitment reference number i,
4. On the *pink ballot commitment list*, the candidate order o is written in the blank space beside commitment reference number i.

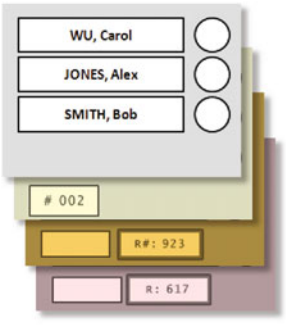
These steps are performed on all ballot assemblies. The pink audit trail is now complete and the *goldenrod receipt commitment list* and *goldenrod ballot commitment list* (and any additional audit trail colors) can be generated in the same manner. An example ballot assembly and corresponding entries in the commitment lists is depicted in Figure 3.

Committing. For an election with two audit trails (pink and goldenrod), the election trustees generate the *pink receipt*, *pink ballot*, *goldenrod receipt* and *goldenrod ballot commitment* lists. They lock these values in time (*i.e.,* commit to them) through the following procedure:

1. Each commitment list is placed in its own appropriately labeled tamper-evident document envelope and sealed,
2. The trustees present the sealed envelopes to the verifiers who are given the opportunity to inspect the exterior of the envelope and sign on the flaps,
3. The envelopes are returned into the custody of the trustees.

3.3 Print Audit Selections

In order to ensure the commitment reference numbers of the ballot assemblies point to the same candidate orderings/serial numbers appearing on the ballot and receipt layers, some ballot assemblies will be selected by the verifiers for a "print audit." Procedure could vary between jurisdictions, but one recommendation would be for the print audit selections to be made in conjunction with



Receipt Commitments	
Ref#	Serial #
...	...
923	002
...	...

Ballot Commitments	
Ref#	Candidate List
...	...
923	WU, JONES, SMITH
...	...

Receipt Commitments	
Ref#	Serial #
...	...
617	002
...	...

Ballot Commitments	
Ref#	Candidate List
...	...
617	WU, JONES, SMITH
...	...

Fig. 3. Commitment Lists: For *each* audit trail (goldenrod and pink in this case) two commitment lists are generated – a **receipt** list and a **ballot** list, each randomly associating serial numbers and respectively candidate orderings with a distinct commitment reference number

the random spot checks of the poll registration book at the polling place during election day (as is the procedure in many paper ballot elections today). A verifier would select a ballot at random from the stack of ballots, and the poll worker would mark the ballot as spoiled (*e.g.,* by punching a hole through the layers). The spoiled ballot would then be given to the verifier to retain for a later auditing procedure.

3.4 Voting

Voting is conducted in accordance with the jurisdiction's procedures for paper ballot elections. An eligible and authenticated voter is issued a ballot assembly by a poll official and is directed to a voting booth. The voter marks the ballot assembly as they normally would in any conventional paper ballot election. They return the ballot assembly to the poll official who first inspects the assembly to ensure the respective layers are still sealed together. The official then separates and distributes the ballot layers in the following way: the ballot layer is cast in the ballot box. The receipt (canary) layer is issued to the voter as a receipt. The pink and goldenrod audit sheets are cast into "pink" and "goldenrod" audit boxes respectively.

3.5 Election Outcome

After the close of the polls, the election results are tallied normally, in accordance with the pre-existing procedures of the jurisdiction for paper ballot elections using contents of the ballot box and is referred to as the "official tally." At the close of the polls, the "pink" and "goldenrod" audit boxes are relinquished into custody of the verifiers.

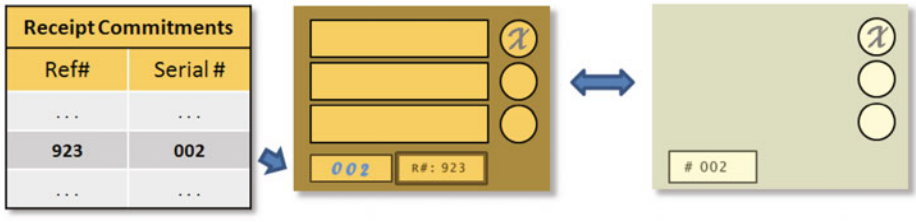


Fig. 4. Reconstituting the Receipt Audit Trail: The Decommited (goldenrod) receipt commitment list (**Left**) can be used to reconstitute receipts from the corresponding (goldenrod) audit trail (**Middle**). The reconstituted receipt audit trail can be cross-referenced against voter receipts (**Right**).

3.6 Decomitting

A coin is flipped in public. If the outcome is heads, the pink audit box is selected to become the ballot audit trail and the goldenrod audit box is selected to become the receipt audit trail. If the outcome is tails, the opposite envelopes are selected. Trustees respond by releasing (*i.e.,* decommiting) the corresponding commitment envelopes through the following procedure:

1. Trustees relinquish selected commitment envelopes into the custody of the verifiers,
2. Verifiers inspect envelopes for the presence of their signatures on the flap and for the absence of evidence of physical tampering of the envelope,
3. Trustees destroy (*e.g.,* shred) the remaining unselected commitment envelopes.

In the following explanation of the receipt and tally audit, for clarity, we will assume a case in which *heads* was the outcome of the coin flip—meaning goldenrod and pink were selected to become the receipt and ballot audit trails respectively.

3.7 Receipt Audit

Using the contents of the goldenrod audit trail box in conjunction with the goldenrod receipt commitment list, a receipt trail can be reconstituted in the following way (see Figure 4):

1. A goldenrod audit sheet is drawn from the goldenrod audit trail box. The commitment reference number $i \in b$ is noted,
2. The i-th row of the *goldenrod receipt commitment list* is consulted, and corresponding serial number s is noted,
3. The number s is written into the blank serial number space on the audit sheet.

These steps are performed on all goldenrod audit sheets. The reconstituted receipts can be cross-referenced against voter receipts to ensure the records match.

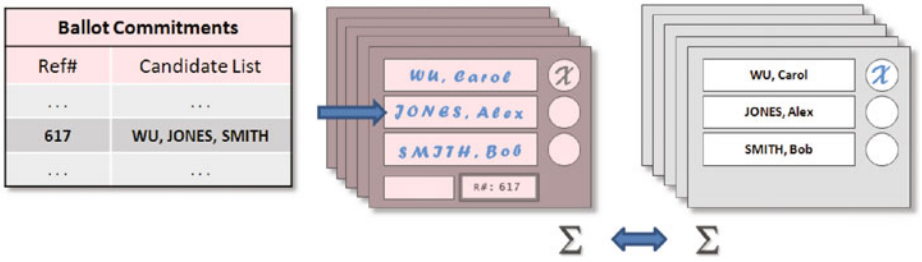


Fig. 5. Reconstituting the Ballot Audit Trail: The Decommited (pink) ballot commitment list (**Left**) can be used to reconstitute ballots from the corresponding (pink) audit trail (**Middle**). The reconstituted ballot audit trail can be tallied and the totals cross-referenced against the official tally (**Right**).

Voters could optionally give their receipts to the verifiers to cross-reference on their behalf, or alternatively the verifiers could publish the reconstituted receipt trail in a public venue (*e.g.,* newspaper) with which voters could check for themselves.

3.8 Tally Audit

Using the contents of the pink audit trail box in conjunction with the pink ballot commitment list, a ballot can be reconstituted in the following way (see Figure 5):

1. A pink audit sheet is drawn from the pink audit trail box. The commitment reference number $j \in b$ is noted,
2. The j-th row of the *pink ballot commitment list* is consulted, and corresponding candidate list ordering o noted,
3. The candidates are written in order o into the blank candidate name spaces on the audit sheet.

These steps are performed on all pink audit sheets. The reconstituted ballots can be tallied and cross-referenced against the official tally to ensure a match.

3.9 Print Audit

Recalling the randomly selected (spoiled) ballots from section 3.3, the correctness of a ballot assembly's printing can be verified in the following way:

1. For a given ballot assembly, candidate order o and serial number s are noted,
2. The ballot assembly's pink and goldenrod audit layers are reconstituted into ballot and receipt audit layers as described in sections 3.7 and 3.8 to recover the o' and s' pointed to by the respective commitment reference numbers,
3. The printing of this given ballot assembly is correct if $o = o'$ and $s = s'$.

4 Security Analysis

In this section, we describe how Aperio meets the E2E integrity criteria as defined in section 2. Further, we analyse the attack vectors that an adversary could use to attempt to corrupt the results of an election and demonstrate the protections offered by Aperio to thwart these attacks.

4.1 A Positive Assertion of Security

Let an unmarked Aperio ballot assembly be the tuple $\langle o, s, c_p, c_g \rangle$ for candidate order, serial number, commitment reference number of the pink sheet, and commitment reference number of the goldenrod sheet. Let ρ denote the position marked by the voter on each element of the ballot assembly. For the following discussion, again consider the instance in which the *pink receipt commitment list* and the *goldenrod ballot commitment list* were selected to be decommitted (although the following security properties are invariant to any particular selection). The audit process establishes the following facts:

1. The voter's receipt contains $\langle s, \rho \rangle$. By matching the voter's receipt to the *receipt commitment list*, it can be verified that ρ' (of row s of the receipt commitment list) matches the ρ on the voter's receipt. Therefore, the voter's mark is included unmodified in the collection of ballots—the first E2E criterion.
2. The print audit verifies that s and c_p printed on a ballot are the same as in the commitment reference sheet and additionally,
3. Verifies o and c_g are the same as on the ballot reference sheet.
4. Since 2 and 3 are dependent on a random decision, it is probabilistic that s and c_g also are consistent between the printed ballots and reference sheets, and additionally,
5. It is probabilistic that o and c_p also are consistent between the printed ballots and reference sheets. If the printed ballots are not consistent, this would be detected with probability $1-(1-Y)^{x-1}$ where Y is the percentage of receipts checked and x is the number of audit sheets.
6. By combining facts 2 and 5, or similarly 3 and 4, we infer that s and o on the sheets are consistent with what the voter saw in the polling booth.
7. By combining 1 and 6, the voter is assured that the same ρ at s on their receipt is in the ballot commitment list somewhere beside the same o that was on their ballot.
8. Finally, given 7, the voter can generate a correct tally for all votes using the ballot reference sheet proving that the collection of ballots is tallied correctly—the second property of an E2E election.

The indirectness of this proof prevents the voter from proving which candidate they voted for to a coercer or someone wishing to purchase their vote. The tally that was generated to provide fact 8 can also be compared to the official tally generated using the original paper ballots for additional assurance.

4.2 Prevented Attacks

In addition to the positive security properties already outlined, it may be also useful to demonstrate a set of attacks it is not susceptible to.

- **Adding or removing ballots.** In order to increase the number of votes for a candidate, an adversary may "stuff" the ballot box with extra ballots for their candidate of choice. Alternatively, given that elections are local events and that correlations often exist between a locality and a political preference, an adversary may attempt to destroy cast ballots in a discriminatory manner by choosing locations based on expected political preference. We assume that a voter registration list is maintained with the number of voters who cast ballots in the election. This information may also be corroborated by election observers. The number of ballots and audit sheets should be identical between them and should also be identical to the total voters on the registration.
- **Modifying ballots.** A more sophisticated adversary would avoid upsetting the number of cast ballots by either replacing existing ballots with new ballots marked for their candidate or modifying the marks on existing ballots. Since a ballot assembly is distributed between distinct boxes, the tuple $\langle o, s, c_p, c_g \rangle$ becomes unlinked and subsequently unassociable once cast into the respective boxes. If an adversary modifies or replaces o on a ballot, then the audit tally will not match the tally generated from adding up all the top layers. Its not in the interest of an adversary to modify c_p or c_g, as there is no way of knowing which candidate a vote is for and which it will be mapped to if modified. Furthermore, there is only a 50% chance that the tally will change at all—alternatively the box will be used to decommit receipts. In either case, such modifications will be detectable either because the tallies will not match, or the receipts will not match. Since c_g and c_p hide o, the voter cannot change both an o on a ballot and, say, c_g on a goldenrod audit sheet in a way that consistently changes both tallies, should the goldenrod ballot commitment list be opened.
- **Misprinting ballots.** An adversary could also misprint ballot assemblies before voting day, or mix-and-match sheets from one ballot assembly with sheets of another. Most of these attacks will not affect the tally (only the receipts) and there is only one method for potentially modifying the tally in an undetectable way: switch both o and one audit sheet, say c_g, with another ballot with a different order o', where $o \neq o'$. If the *goldenrod ballot commitment list* is selected to be decommited, the two tallies will match, otherwise the attack will be detected. Furthermore, since the adversary has no guarantee of which voter will receive the misprinted ballot, votes cannot be predictably directed to a particular favored candidate. This attack is marginal and can be further mitigated by using more than two audit layers, which exponentially decreases the probability of a successful execution of this attack.
- **Forced randomization attack.** An adversary could coerce a voter into returning with a receipt with a particular position marked. Since the adversary

has no knowledge of o, this is equivalent to forcing the voter to throw away their vote by voting for a random candidate as demonstrated in [12]. Since a random vote will be given to each candidate with equal probability, coercing a random votes will have the same effect as coercing a voter to not vote at all—the latter likely being easier to execute.

- **Chain voting.** An adversary who can capture an unmarked ballot can execute a chain voting attack where they fill out the unmarked ballot for a candidate they choose, then coerce a voter into casting the adversary's ballot in place of the unmarked ballot the voter receives, and demand that the voter return to them the voter's unmarked ballot so they can execute the attack *ad infinitum*. This can be mitigated by the use of a detachable, uniquely marked "counterfoil." As the voter registration official issues the ballot, the number n on the counterfoil is noted. When the voter returns, the official verifies the presence of the counterfoil and notes the number n'. If $n = n'$, the voter has not exchanged ballots. The counterfoil is then detached and discarded, and the ballot assembly and the casting procedure continues as in section 3.4. As an option, the counterfoil could be additionally utilized to initially seal the ballot layers together to prevent anyone from viewing s, c_g and c_p beforehand.
- **Pattern Attacks.** A voter wishing to sell their vote could mark their ballot in a pattern that is likely to be uniquely identifiable and then point out its location to the buyer in the ballot commitment list. This can be mitigated by partitioning the audit into subaudits: *i.e.,* having separate commitment lists for each contest or small collections of contests. Exactly how to partition a ballot to marginalize a pattern attack can be determined statistically—*i.e.,* [8].

5 Privacy

In this section we describe how the scheme protects voter privacy. With the exception of the privacy entrusted entity that generates and prints ballot assemblies, no information about how a voter votes becomes known to the other entities (*i.e.,* voter, poll officials, verifiers) with the following justification:

- **Ballot assembly.** It is easy to see that through the decommiting process, $\langle c_p, c_g \rangle \Rightarrow \langle o, s \rangle$, and therefore $\langle c_p, c_g, \rho \rangle \Rightarrow \langle o, s, \rho \rangle$, which is the basis of the print audit of *spoiled* ballot assemblies. Privacy is preserved on these ballot assemblies given $\rho = \emptyset$. Assuming poll procedure is followed, then $\langle c_p, c_g \rangle$ will be physically unlinked by the poll official and logically unlinked by the mixing in the audit trail ballot boxes. It is central to voter privacy that neither the voters, poll officials or verifiers see $\langle c_p, c_g \rangle$ leading up to, and during, the ballot casting process. The assembly layers might be sealed (*e.g.,* glued) together as previously suggested by a tear-off "counterfoil."
- **Ballot receipt.** Given that $\langle o, s \rangle$ were randomly selected in section 3.2, and known only to the privacy entrusted entity, then to all other entities $\langle s \rangle \not\Rightarrow \langle o \rangle$ and therefore $\langle s, \rho \rangle \not\Rightarrow \langle o, s, \rho \rangle$ given receipt $\langle s, \rho \rangle$.

– **Receipt Commitment Lists.** Given *ballot commitment lists* $\mathcal{CB}_g = \{\langle c_{g_1}, o_1\rangle, \cdots, \langle c_{g_b}, o_b\rangle\}$ and $\mathcal{CB}_p = \{\langle c_{p_1}, o_1\rangle, \cdots, \langle c_{p_b}, o_b\rangle\}$ and *receipt commitment lists* $\mathcal{CR}_g = \{\langle c_{g_1}, s_1\rangle, \cdots, \langle c_{g_b}, s_b\rangle\}$ and $\mathcal{CR}_p = \{\langle c_{p_1}, s_1\rangle, \cdots, \langle c_{p_b}, s_b\rangle\}$, but given that only one of $\langle \mathcal{CB}_g, \mathcal{CR}_p\rangle$ and $\langle \mathcal{CR}_g, \mathcal{CB}_p\rangle$ are ever made public, it is easy to see $\langle c_{g_i}, o_i\rangle \not\Leftrightarrow \langle c_{g_i}, s_i\rangle$ and $\langle c_{p_i}, o_i\rangle \not\Leftrightarrow \langle c_{p_i}, s_i\rangle$ and thus $\langle c_{g_i}, \rho_i\rangle \not\Rightarrow \langle o_i, s_i, \rho_i\rangle$ and likewise $\langle c_{p_i}, \rho_i\rangle \not\Rightarrow \langle o_i, s_i, \rho_i\rangle$.

5.1 Prevented Attacks

Under the privacy properties, a link cannot be established between a vote and a receipt, and therefore the receipt cannot be used to prove how a voter voted. That is to say any observer, given only the ballot receipt, cannot "guess" a voter's selections with non-negligible advantage. These privacy properties address the following attacks:

– **Vote Buying.** A voter who votes a certain way, following an arrangement made between the voter and any entity *a priori* to voting, cannot subsequently prove how they voted. This effectively relegates vote buying to, at best, conventional threats, or at worst, to the previously mentioned forced-randomization attack.
– **Retribution.** A subtly different threat, often overlooked in literature, is the circumstance in which no precursory voting agreement exists, yet a possibility of retribution exists if the voter's selections become known *a postiori* to voting. Concern for future retribution would be legitimate cause for a voter to vote differently than intended, and therefore is arguably as serious a concern as vote buying and must likewise not be facilitated by any proposed receipt scheme.

6 Extensions and Other Applications

6.1 Multiple Contests

The specific system presented in this paper considered a small, single contest race. Under this scheme, uniquely patterned receipts are unlikely to occur under the so-called "Short Ballot Assumption" of [14]. In the case of cryptographic E2E systems, elections with multiple contests have attempted to mitigate against pattern attacks by partitioning the anonymizing network by contest [11]. It is easy to see the physical analog of this technique for a multi-contest ballot would be to either to perforate the ballot assembly such that the assembly could be separated by contest at casting time, or simply to have one assembly per contest (a technique already employed by some jurisdictions).

6.2 Automation of Verification Process

Although an election environment may not provide for optical scanners at the polling place, it may be reasonable to assume some computing capability on

behalf of the trustees and verifiers. In this circumstance, the trustees might print the commitment reference numbers on audit sheets as well as the commitment lists in an optically readable format to speed the verification process. In this scheme, the same audit operations would be undertaken, except by a computer.

6.3 Implications for Cryptographic Schemes

The anonymizing network model introduced by Aperio is fundamentally simpler than the mixnet model employed by many cryptographic schemes in the sense that it does not propagate mark states through mix nodes, and therefore does not directly require a scheme such as *randomized partial checking* [9] to verify the network's correctness. This could replace the anonymizing network of a system such as Scantegrity [4] which employs multiple instances of a two stage permutation-network, instead, with multiple (independently shuffled) instances of $\langle \mathcal{CR}, \mathcal{CB} \rangle$. This change would likely decrease the total number of verification operations, and arguably decreases the overall conceptual complexity of the verification process.

7 Conclusion

In this document we have introduced Aperio, a mechanism to provide custody independent verification of election results without the use of electronic equipment or special skill requirements for voting, tallying, or verification. Our intent through this proposal is twofold. First, we hope to cultivate discussion on practical techniques for strengthening the democratic process in developing counties and other minimally equipped or minimally funded election environments. Second, it is our hope that Aperio will be useful as a more broadly-accessible educational tool for E2E concepts, especially for the purpose of growing understanding and acceptance of its various cryptographic counterparts.

References

1. Adida, B., Rivest, R.L.: Scratch & vote: self-contained paper-based cryptographic voting. In: WPES 2006: Proceedings of the 5th ACM workshop on Privacy in electronic society, pp. 29–40 (2006)
2. Ballot, A.: Merriam-webster dictionary. Online
3. Chaum, D.: Untraceable electronic mail, return addresses, and digital pseudonyms. Commun. ACM 24(2), 84–90 (1981)
4. Chaum, D., Essex, A., Carback, R., Clark, J., Popoveniuc, S., Sherman, A.T., Vora, P.: Scantegrity: End-to-end voter verifiable optical-scan voting. IEEE Security and Privacy Magazine (May/June 2008)
5. Chaum, D., Ryan, P.Y.A., Schneider, S.A.: A Practical, Voter-verifiable, Election Scheme. Technical Report Series CS-TR-880, University of Newcastle Upon Tyne, School of Computer Science (December 2004)
6. Chaum, D., van de Graaf, J., Ryan, P.Y.A., Vora, P.L.: Secret ballot elections with unconditional integrity. Technical report, IACR Eprint (2007), `http://eprint.iacr.org/`

7. United States Election Assistance Commission. 2005 voluntary voting system guidelines (December 2005), http://eac.gov/vvsg_intro.htm
8. Henry, K., Stinson, D., Sui, J.: The effectiveness of receipt-based attacks on three-ballot. Technical report, Centre for Applied Cryptographic Research, University of Waterloo (2007)
9. Jakobsson, M., Juels, A., Rivest, R.L.: Making mix nets robust for electronic voting by randomized partial checking. In: USENIX Security 2002, pp. 339–353 (2002)
10. Popoveniuc, S., Hosp, B.: An Introduction to Punchscan. In: Preproceedings of the 2006 IAVoSS Workshop on Trustworthy Elections, Robinson College, Cambridge, United Kingdom. International Association for Voting System Sciences (2006)
11. Popoveniuc, S., Stanton, J.: Undervote and pattern voting: Vulnerability and a mitigation technique (June 2007)
12. Popoveniuc, S., Stanton, J.: Buying random votes is as hard as buying no-votes. Cryptology ePrint Archive, Report 2008/059 (2008), http://eprint.iacr.org/
13. Rivest, R.L., Wack, J.: On the notion of "software independence" in voting systems. DRAFT Version (July 28, 2006)
14. Rivest, R.L., Smith, W.D.: Three Voting Protocols: Threeballot, VAV, and Twin. In: Usenix/Accurate EVT (August 2007)