Secret-Ballot Receipts and Transparent Integrity

Better and less-costly electronic voting at polling places

David Chaum

Introduction

Current electronic voting machines at polling places do not give receipts. These machines instead require each prospective voter to trust them—without any proof or confirming evidence—to correctly record each vote and include it in the final tally. Receipts could let voters be sure that their intended votes are counted. But receipts have been outlawed generally because of the "secret ballot" principle, which forbids voters from taking anything out of the polling place that could be used to show how they voted to others. These laws are aimed at preventing "improper influence" of voters, such as vote-selling and various forms of coercion.

Introduced here is a new kind of receipt. In the voting booth, it is as convincing as any receipt. And once the voter takes it out of the booth, it can readily be used to ensure that the votes it contains are included correctly in the final tally. But it cannot be used in improper influence schemes to show how the voter voted.

The system incorporating the receipts can be proven mathematically to ensure integrity of the election against whatever incorrectly-behaving machines or people might do to surreptitiously change votes. Not only can receipts and this level of integrity greatly enhance voter satisfaction and confidence, but they eliminate the need for trusted voting machines.

The current requirement that full faith be placed in each voting machine, independent of whether it is deserved, has real disadvantages. Both private and public sectors have rejected such proprietary "black box" technology for almost everything but voting in favor of open-platform solutions, like PC's, which are suitable for the system proposed here. The high volumes and standards of open platforms allow significantly lower cost and higher quality, with better availability and upgradeability, because of the competitive markets in hardware and software. The receipts also improve robustness over devices that must be trusted, not only because failures can be detected at the polls in time to prevent lost votes, but also because the votes that receipts contain can be counted no matter what happens to the machines, which currently use costly proprietary hardware redundancy to store and transport votes. And open platform hardware, instead of being stored in special warehouses most of the time, could even be used for various purposes year-round in places such as schools and libraries.

All the security provisions currently employed around voting are not only costly, but what they can achieve is limited. Destruction of audit-chain data within the machine, for example, is mandated by current design specifications for privacy reasons. In contrast, receipts allow significantly more efficient and

more effective integrity assurance, even though all local data is either published or expunged.

Inability under the current approach to reconcile secrecy and security needs has also led to problems of functionality. Contested and provisional ballots, where poll workers challenge the rights of voters to vote, today require separate handling and counting that singles them out for reduced privacy protection. Just as the system presented here can seamlessly include all such votes, it can lift the requirement that voters must vote from their home precinct, ensuring access while improving convenience and turnout. (Even inter-jurisdiction voting becomes workable.) Courts can also surgically add or remove the votes of particular fine-grained categories of voters; being unable to do this today forces them to choose between calling a re-vote, throwing out all ballots or determining winners themselves.

Voting with the new approach

After making your choices on a touch screen (or by whatever input means), when you vote using the new approach, a small printer that looks like those at cash registers prints your receipt. This printout shows your vote and only your vote. The names of those candidates you chose, office sought and party affiliation, are listed as well as your choice on any ballot questions. (Please see figure 1.) Included are any allowed "write-ins" and other choices you made, such as with "open primaries," "none-of-the-above" options or "instant-runoff" voting.[1] There could also be warnings about contests or questions not voted. Once all your votes are printed, you are prompted by the machine to decide whether you agree with the receipt. If you don't agree with it you can amend your vote and print a new receipt.

Fig. 1: Line of receipt for a particular candidate.

If you do agree with the receipt, you are asked to indicate whether you wish to take the top or the bottom "layer" of what is at this stage a two-layer receipt. The special receipt printers used differ from ordinary receipt printers in that instead of just printing on the top side of the form, they can also simultaneously print separate but aligned graphics on the bottom side of the form.

Your freedom to choose at this point which layer you will take, even though it is an arbitrary decision, is key to keeping the system honest. Only after you've indicated your choice of layer, a further inch or so of the receipt is printed. Then both layers, still laminated together, are automatically cut off and released to you. (See figure 2.)

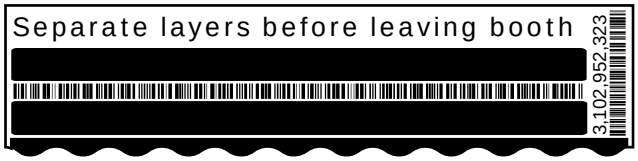

Fig. 2: Last inch printed, not yet separated.

As you separate the two layers, the image of the votes changes into an unreadable and seemingly random pattern of tiny squares printed on each of two layers of translucent plastic material—it was the light passing through the sandwiched layers only in places where the mutual relationship is such that *both* have no printing that made your choices visible. Neither layer is readable on its own, because of this special graphical encryption, as explained later. But each layer separately and safely encodes exactly your vote as you saw it.

The last inch of the form is different because its layers have messages that are readable after they are separated. The layer you selected to keep as your receipt, whether top or bottom, would bear a message like "Voter keeps this privacy-protected receipt layer" (see figure 3), while the other layer would state something like

[1] Graphics like candidate names as written by each voter and party symbols, such as are used in some countries, can be included as well on receipts.

Fig. 3: Separated layer selected by voter to keep.

"Voter must surrender this layer to poll worker" (see figure 4).

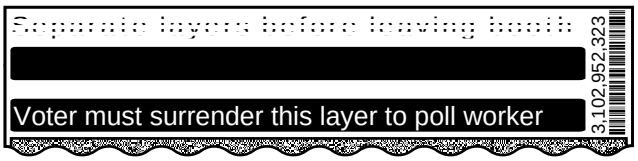

Fig. 4: Separated layer to be shredded.

You take both layers out of the booth. At the exit of the polling place you hand the poll worker the layer that is marked to be surrendered. The poll worker makes sure that it is indeed the layer marked for surrender and then, for your protection and as you watch, destroys it in a small transparently-housed paper shredder. The other layer is finally your receipt that you can take away with you. An electronic version of exactly this same final receipt, which as mentioned encodes your vote, is also kept by the voting machine until it is successfully sent in for posting on the official election website.[2]

Your receipt can safely be shown for checking to anyone, such as various political, governmental, public-interest or media organizations. Outside the polling place, for instance, you might find one or more groups, such as the League of Women Voters, prepared to check your receipt for you if you wish. They simply scan it and immediately let you know that it is authentic and correct (by subjecting the receipt's printed image and its coded data to a consistency check, detailed below, and later also ensuring that it is correctly posted online when it should be[3]). If an invalid receipt were ever detected, incorrect operation of election equipment would be irrefutably indicated, but a false alarm could readily be dispelled by any other checker.

Once the polls close, the digital form of the receipts are sent from the polling place, electronically or by transport of physical media; no digital record is kept of the shredded layers.

You could, if you wish, privately look up on the official election website the page that includes your receipt among others, by entering part of its serial number. You would then be able to check for yourself that it has been posted correctly by, for instance, printing it out and overlaying the two and seeing that they are the same. (You need not run consistency checking software, since anyone can do this for all posted receipts, as mentioned later.) You could also simply provide the original or its image by fax or photocopy to others who might also check it.

The definitive set of receipts to be counted is, at some point after the close of polls, posted on the website as a list called the "receipt batch." The final output of the election, the "tally batch," is also similarly posted. It contains exactly the same number of items, but each is a readable plaintext image of the ballot just as the voter saw it (from which anyone can compute the totals with simple software). To protect privacy and ballot secrecy, the receipt and tally batches are not in the same order, thereby hiding the correspondence between receipts and ballot images. But to ensure that there does in fact exist a one-to-one correspondence—i.e, that ballots were not inserted, deleted, or changed—a chain of intermediate batches between the receipt batch and tally batch is used.

[2] An example way to handle voters refusing to surrender layers is for the exit shredder (based on barcode reading of the ballot serial number, which additionally prevents shredding the wrong layer and allows spoiling of "missing" receipts), to give a sticker with a key needed to decrypt the receipt signatures. This also lets voters claiming their choice of layer was switched to safely be issued the other last inch.

[3] Revealing receipt copies only after the official version is published, and then widely, helps ensure that all receipts are posted correctly and consistently. Even if, for instance, only a randomly-selected 50% of receipts are ever circulated, a published list has a chance of half of being contradicted for each improper entry. The dilution of overall probabilities by low receipt percentages may not be significant except for extremely close contests, which can adaptively require higher percentages.

Once all these batches are created and published, randomly-chosen samples of linkages between items of batches adjacent in the chain are decrypted. The choice of samples is made so that it does not reveal so much that it compromises privacy. The samples do reveal enough, though, that they can be checked against the published batches by running a simple, open-source program downloadable from any of multiple suppliers (which can also check the consistency of each entry in the receipt batch). This check, explained later, lets anyone be quite confident that the receipt batch must have correctly yielded the tally batch.

Much effort is currently spent trying to assure integrity of elections, including: extensive closed reviews and testing of software and hardware, government and party observers, so called "logic and accuracy" testing before and after each election, elaborate security measures, as well as costly and time-consuming audits and recounts. With the receipt system, whatever level of effort is applied would yield far greater assurance and reduce overall cost. Moreover, checking becomes easy enough that ordinary voters can truly verify their own ballots.

It is important to ask, as for any security system: What are the properties claimed? How does the mechanism work? and What is the proof that the mechanism really ensures the properties? All three questions are considered for a general audience, beginning with the first question. Answers to the second and third questions are introduced in three parts: the receipts, the tabulating process, and the cryptography. In the final section, the system is described more formally and the properties are proved.

Properties of the receipt system

The receipt system ensures these main properties:

- If your receipt is correctly posted, you can be sure (with acceptable probability, see last bullet item below) that your vote will be included correctly in the tally.
- If your receipt were not properly posted, it would be physical evidence of a failure on the part of the election system and any refusal by officials to post it would be an irrefutable admission of a breakdown in the election process.
- Your receipt cannot be decoded by anyone, or otherwise linked to your vote, except by decrypting with (or breaking) all the secret keys of which each trustee has its own.
- There are only two ways that a system, no matter how incorrectly it operates, would have a chance of changing a voter's correctly-posted ballot without being detected: (1) printing an incorrect layer and hoping that the voter chooses the other layer; or (2) incorrectly performing a step among the tally process steps and hoping that step will not be selected for audit. For each voter's ballot and with either approach, the chance that it would go undetected is one half. Thus, the chance that two ballots will be changed without anything being discovered is only a quarter, three ballots an eighth, and so on. Changes in just 10 ballots, as a larger example, will avoid detection less than one in 1,000 times and changes in just 20 ballots will avoid detection less than one in 1,000,000 times.

How and why the receipts work

What makes the laminated layers readable and the separated layers meaningless is the mutual relationship of the special patterns printed in black on each translucent plastic layer. All the printing on both layers is divided into a grid of squares, called "pixels locations." Every pixel location is printed with one of two "pixel symbols," like a large Tic-Tac-Toe board that is all filled in. The main thing about the two pixel symbols used is that, while half the pattern of each is clear and the other half is black, they are reverses of each other: where one is clear the other is black and vice versa. So when the two different pixel symbols are aligned one directly on top of the other as when laminated, any clear

on one is blocked by black on the other—the lamination appears totally opaque. But when the same pixel symbol is printed on both layers, and they are aligned one above the other, all the clear parts are directly over each other and part of the light is thus able to shine through. (See figure 5.)

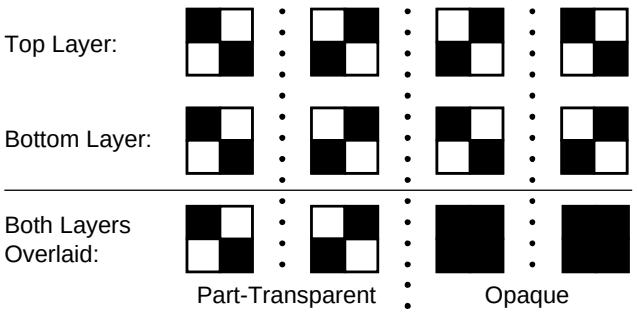

Fig. 5: The two pixel symbols, separate & overlaid.

This technique can be used to encode information on one sheet so that it can only be read by someone with a second sheet, as was first proposed by Naor and Shamir [1995]. It will be useful to have names associated with the two sheets: the first will be called "white" and the second "red"; these colors have no more graphic significance than they might tint the two translucent sheets so they can be easily recognized.

You start with the white sheet, divided into its grid of pixel locations, each pixel location with a separately-chosen random pixel symbol printed on it. When two sheets are "laminated" together, the grids are exactly lined up; each pixel location on one sheet has a "paired" pixel location at the same coordinates on the other sheet, so that the two are exactly on top of each other. Now to encode your message in the red sheet you simply choose each of its pixel symbols accordingly: if you want light to shine through for that pixel location when laminated, you choose the same pixel symbol as its paired pixel on the white sheet; and if you don't want light to come through at that location, you choose the other symbol.

In most printing technologies today, ordinary text is simply printed by creating a grid of pixel locations in which some are printed fully with black ink while the other pixel locations get no ink. For receipt printing: instead of leaving the background without ink, non-matching (i.e. opaque) pixel symbol combinations are paired; and instead of full black ink for the letters, matching (i.e. partly-clear) pixel-symbol combinations are paired giving a gray effect whose brightness depends on the backlighting.[4] (See Figure 6.)

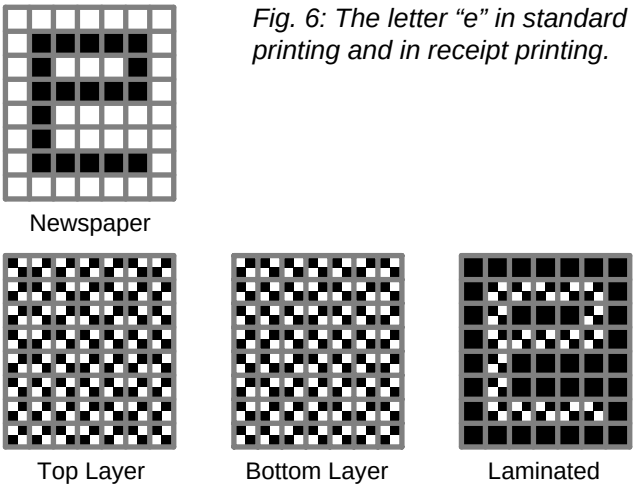

Fig. 6: The letter "e" in standard printing and in receipt printing.

For receipts, when the layers are still laminated, printing of all the choices made by the voter is thus in a kind of gray on a black background. It is called the "ballot image"—a definition of the voter's vote that has been accepted by the voter in a visible plaintext form.

Since the vote should be encoded in each layer separately, "red pixels" are needed on both layers. The solution is based on the observation that swapping two paired pixel symbols between the layers leaves the laminate visually unchanged. So pairs in half of the pixel locations, say, in a checkerboard pattern, are swapped. If the pixels were tinted, instead of separate red and a white layers, they would look like the red and white table cloths in a typical bistro.

[4] So-called "thermal" printers, deployed at most checkout counters these days, have twice to three times the resolution needed here, but this can be used to neatly frame pixels and forgive mechanical alignment error between the ceramic printheads that run the width of the paper on top and bottom. A clear "fugitive" adhesive keeps the two layers together and is not sticky when delaminated.

The coding technique used to encrypt the ballot image is the so-called "one-time pad." It has been proven by Shannon [1951] to be unbreakable, assuming the key is random. The keys used, the white pixels, are not random but are believed in practice "indistinguishable" from random except to a set of "trustees" who are collectively the guardians of ballot secrecy (as will be explained later). So if you have only a single-layer final receipt and are staring at a particular "white" pixel on it, you learn nothing. Similarly, a "red" pixel only tells you that the lamination would have been partly clear if the paired white pixel were to match the red pixel and opaque if it did not—but knowing nothing about which pixel symbol was paired means you cannot infer anything about whether or not the combination was partly clear or opaque.

Receipts should encode exactly the votes seen by the respective voter. It is technically possible, however, that the voter would see one set of choices in the laminated receipt, but the receipt layer the voter takes would actually encode other choices. The only way this could happen though is if at least one layer is invalid. If both layers are invalid, then no matter which one the voter takes, it will not pass checking and be caught. If only one layer is invalid, and it is not selected by the voter, it would not be checked, just shredded. But essential to security, as mentioned above, is that the voter chooses which layer to take only *after* the printer finishes printing the votes. So, even a single invalid layer must first be printed and then it has a fifty-fifty chance of being selected by the voter and caught.

How and why tabulating works

The tabulating process starts with an agreed receipt batch and produces a final tally batch of ballot images. Once the polls are closed, any contested or provisional voting should be resolved and all the receipts to be included in the tabulating process should be posted electronically as the official definitive receipt batch[5]. Then the first trustee can produce the first "intermediate batch" from the receipt batch. After that, a trustee forms the second intermediate batch from the first intermediate batch, and so forth, until the last trustee forms the tally batch from the last intermediate batch. (See figure 7.)

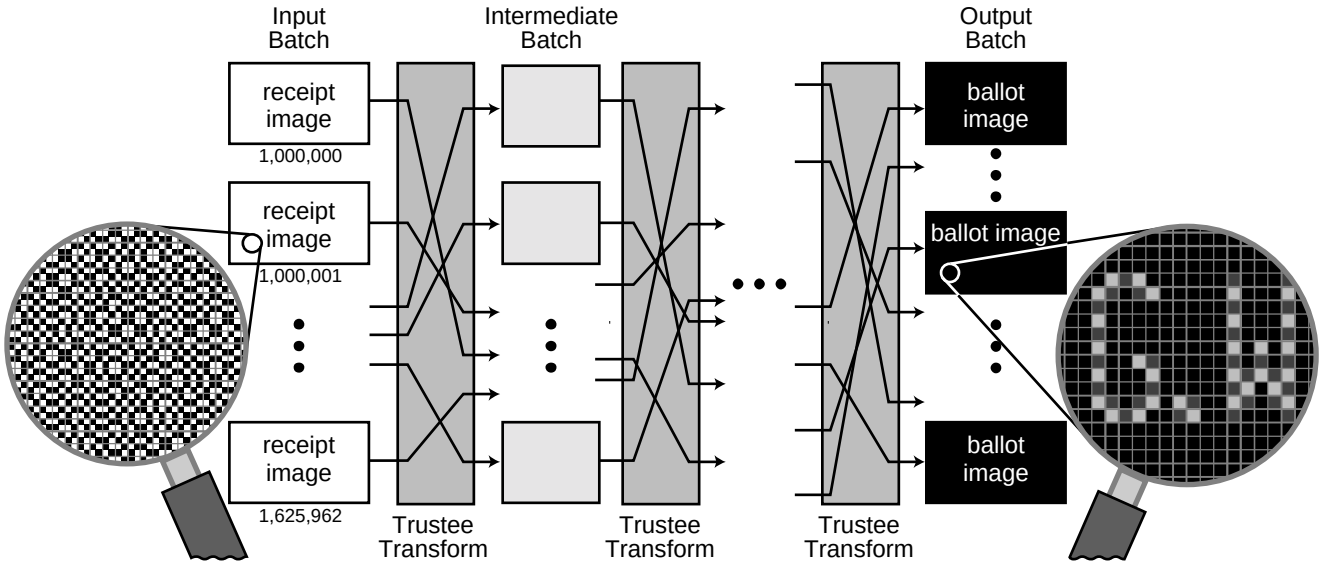

Fig. 7: Overall tabulating process from serial-numbered receipts through trustee-operated mixes to ballot images that are posted and then tallied. The vertical ellipsis "..." indicate the batch items not shown and the horizontal ellipsis the additional trustees. (The darker ballot-image pixels are inferred from the lighter ones.)

[5] A preliminary tally can be formed, before contested and provisional ballots are included, that omits a random

Privacy is ensured by the trustees changing the coding, as well as the order of items, from one batch in the chain to the next. Security is ensured by requiring each trustee to release some random samples of linkages, which establishes that items have been correctly transferred from batch to batch.

This method of processing the votes, following the chain of intermediate batches all the way from the receipt batch through to the final tally batch, and its audit, can be described in more depth by analogy with special "Russian nested dolls." Each batch is a collection of these dolls. The receipt batch, for instance, is a batch of complete "big" dolls, each with all its smaller dolls neatly nested within it. The next batch, the first intermediary batch is the same except that the outermost doll has been removed from each big doll. This continues all the way until the tally batch, at the other end of the chain, in which only the tiny, solid-wood innermost dolls remain. Thus, all batches have the same number of outermost dolls. Within a batch, all outermost dolls are the same size and each contains all its own smaller dolls.

The special nesting dolls used are like secret agents, each doll holding her unique random "code sheet" in her hands. The sheet is just a grid of pixels printed using the two pixel symbols. Each doll is also physically locked closed by a combination lock. For each size doll, there is a different secret combination that unlocks all the dolls of that size and that is know only to a single corresponding trustee.

Consider the trustee that has the secret combination for, say, the locks of the 10-inch dolls. To process an individual doll in the batch of 10-inch dolls, he first unlocks it using his secret combination and removes its contents, a 9-inch doll. At this point he has two code sheets, one from the 10-inch and one from the 9-inch doll. He combines the two sheets to produce a new code sheet as follows: for every pixel location where light shines through the two sheets when stacked, one pixel symbol is printed on the new sheet and everywhere it does not, the other symbol is printed.[6] He then places this combined code sheet in the hands of the 9-inch doll and destroys the empty 10-inch doll along with both old code sheets.

Once all the 10-inch dolls have been processed in this way into 9-inch dolls with new code sheets, he randomizes their order and outputs them as a batch. The trustee with the secret combination for the 9-inch dolls then takes this batch as input and processes it into a batch of 8-inch dolls, and so on.

To use all this for a kind of election, the sheet held by each big doll is formed in a special way from all the sheets of the dolls nested within it. Suppose the original doll maker faithfully chooses sheets for all the dolls inside a big doll at random, but makes copies of all the sheets. Instead of keeping these copies each on a separate sheet, he can combine them into a single sheet for that big doll, one pair of sheets at a time (or all at once using some simple arithmetic[2]). The result is a single "white" sheet for that big doll. This can be thought of as a kind of "adding in" of the coding of all the sheets that will later be "subtracted back out" in stages as the dolls are processed.

Now suppose voters, staying with the analogy, each have one of the special dolls and want to use it to vote with privacy. Each determines a "red" sheet such that when it is optically combined with the white sheet of the doll it results in the desired ballot image. Then each voter gives the big doll its red sheet to hold and places it in the initial batch of big dolls. Once the batch of dolls passes through processing by all the trustees, the final output batch is of tiny solid-wooden dolls each holding a sheet that reveals a ballot image—because all of the code sheets added in to form the white

selection of ballots that will be included in the final tally, so as to obscure the provisional/contested votes.

[6] When each of the two pixel symbols is viewed as a binary digit, 1 or 0, combining (any number of) sheets is just adding up the 1's and calculating the remainder after dividing by 2 and then printing the corresponding symbols on the new sheet.

sheet that influenced the red sheet have finally been subtracted back out. (Laminating a sheet with one pixel symbol copied everywhere makes it easy to read the revealed ballot image.)

To provide integrity, a way is needed to catch any trustee if it were to change sheets improperly during processing. A solution requires trustees to release complete and detailed audit trails of the processing, say, as video tapes, but only for some selection of dolls. So that half the tapes can be released without compromising ballot secrecy, trustees each process more than one of the batches, say, two successive batches of the chain. (Tracing any tiny doll back to a big doll should not be possible, even for a collusion of all but one trustee.) After all the processing, a lottery-style draw selects half the dolls in the first input batch of a trustee to have their videos released. These dolls would not have videos of the second processing revealed, while all the other dolls would. (See figure 8.) Exact tracing is thus prevented because only one video is released per doll for the two adjacent batches. Still, each time a trustee improperly forms a batch item there is a 50% chance of this being selected for release on video, so the odds of being caught stack up pretty fast!

Returning to the receipt system, the analogy's red and white sheets correspond, of course, to the whole set of red and white pixels (though without checkerboarding). The analog of a locked wooden doll container is so-called "public-key encryption," in which anyone can encrypt a message, using a published public key, but only the holder of the corresponding private key, the trustee, can decrypt it. Thus any voting machine can successively form the layers of a (digital) doll using the respective published keys, but only a trustee can in turn strip off its layers using its private keys.[7] With encryption as the mechanism, instead of a videotape, the code sheet originally held by the output doll is all that has to be released (because it is easy to check that the input doll can be re-constructed by

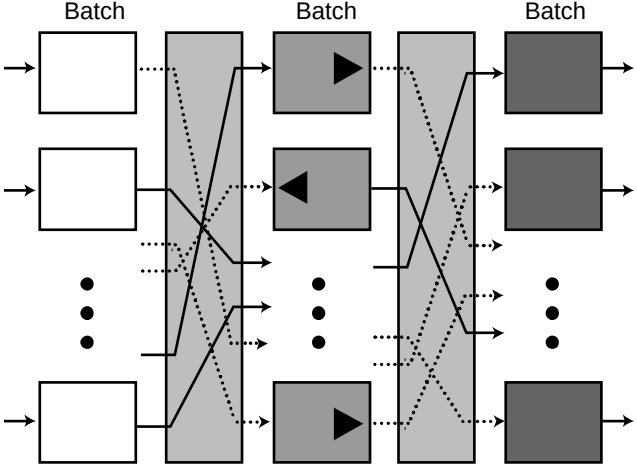

Fig. 8: Single trustee with triangles & broken lines showing links whose details are released in an audit.

applying the public key to the combination of original sheet and output doll).

A receipt uses two dolls, and both are printed on each layer.[8] One doll is checked completely by being re-constructed from values printed on the last inch of the receipt layer. The other doll and the red pixels comprise a "duo" that travels together through the chain of batches in the tally process. All batches are made up completely of such duos. Each batch is processed by a trustee removing encryption from the duo's doll and applying coding to what started out as its red pixels. By the time the duo reaches the tally batch, there is nothing left of the doll and the red pixels have become the plaintext ballot image.

The codes and what they achieve

One kind of coding mentioned already is the "digital signature." These are printed in barcode on the last inch of the receipt layer. Such signatures, while only computationally secure (as will be defined below), have been given legal standing in many countries, and are used as irrefutable proof that they were formed by the legitimate signer. Thus, a verifier outside the polling place having scanned your receipt can immediately check, among other things, three aspects of the signature: is it valid, was it made

[7] Various known redundancy and key-sharing techniques provide resiliency in case some trustees don't participate.

[8] Multiple and preferably differently-coded copies of the dolls printed on one layer use pixel symbols opposite the copies on the other layer, creating a uniform opaque background whose absence would easily be noticed by the voter, ensuring that the two are identical. Serial numbers are printed in this same way as well as readably in the last inch. Additional error detecting and correcting coding in the black areas gives robustness for the entire image.

by an authorized voting station, and does it correctly cover all the data printed. If the signature does not so check, the physical receipt itself is immediate evidence of failure. If the receipt does check, however, you know that the receipt cannot be credibly denied a place in the definitive receipt batch.

Cryptographic techniques are generally divided between those that are "unconditionally secure" and those that are merely "computationally secure." The former, like the one-time pad with random key described above, cannot be broken, even if an adversary were to apply infinite computing power. However, almost all cryptographic algorithms used are of the computationally secure type, and are known in principle to be breakable if enough computing power were to be applied. Most likely though, no criminal has been able to find a way to make such computations using resources available today (since many systems, including international high-value wire transfer, rely on such codes and are still in place). Such standard cryptographic building blocks, which are also like those used widely by browsers when accessing secure websites today, are enough to build the systems described here.

Computationally-secure encryption is used here to form the layers of dolls, which ultimately encrypt the data in receipts and batches, and thus to protect privacy and ballot secrecy. Because of current surveillance technology, such as sensors like miniature cameras and emanations receivers, as well as memory and transmitters, the confidentiality of what transpires in voting booths cannot in practice be held to any absolute standard. But after voting, the codes protecting receipts and posted batches, which are only readily linkable to ballot numbers and not people, can easily be at least as good as those protecting comparable and much more identifiable, sensitive and detailed data traveling on networks today. Moreover, technical provision of privacy has its limits in voting, some examples of which are as follows: most US voter addresses and party affiliations are a matter of public record; the more help a device gives a voter the harder it is to keep it from learning who they vote for (though here the devices need not be able to retain data between votes); even the "gold standard" of voting systems, manual paper ballots, is subject to marking or ballot-number recording and automatically captures fingerprints; and theoretical limits are believed generally to force a choice in cryptographic systems between unconditional integrity and unconditional privacy.

Thus the present system is arguably optimal. It protects privacy, by encrypting votes in receipts and published batches, computationally. And it protects integrity of the tally, by enforcing probabilities of detecting tampering, unconditionally.

More formally

The system presented is in two "phases," a "voting" phase followed by a "tally" phase. First consider the voting phase, which is comprised of a number of instances. Each instance is in up to 6 successive steps: (1) the prospective "voter" supplies a "ballot image" B; (2) the system responds by providing two initial 4-tuples: $<{}^{z}L,q,{}^{t}D,{}^{b}D>$, each printed on a separate "layer," the "top" layer with $z=t$ and the "bottom" with $z=b$; (3) the voter "verifies," using the optical properties of the printing, that ${}^{t}R \oplus {}^{b}W = {}^{t}B$ and ${}^{b}R \oplus {}^{t}W = {}^{b}B$ as well as that the last three components of the 4-tuple are identical on both layers; (4) the voter either aborts (and is assumed to do so if the optical verification fails) or "selects" the top layer $x=t$ or the bottom layer $x=b$; (5) the system makes two digital signatures and provides them in a 2-tuple $< {}^{x}s(q), {}^{x}o({}^{x}L,q,{}^{t}D,{}^{b}D,{}^{x}s(q)>$; and (6) the voter or a designate does the "consistency check" that (a) the digital signatures of the 2-tuple check, using agreed public inverses of the system's private signature functions ${}^{x}s$ and ${}^{x}o$, with the unsigned version of the corresponding values of the selected 4-tuple as printed on the selected layer and (b) that ${}^{x}D$, and the half of the elements of ${}^{x}L$ that should be, are correctly determined by ${}^{x}s(q)$.

More particularly, the relations between the elements of the 4-tuples and the 2-tuple are defined as follows. The m by n binary matrices ${}^{z}L$ are determined by the "red" bits ${}^{z}R$ and "white" bits ${}^{z}W$ (both m by $n/2$, n even), in a way that depends on whether $z=t$ or $z=b$: ${}^{t}L_{i,2j-(i \bmod 2)} = {}^{t}R_{i,j}$, ${}^{t}L_{i,2j-(i+1 \bmod 2)} = {}^{t}W_{i,j}$, ${}^{b}L_{i,2j-(i+1 \bmod 2)} = {}^{b}R_{i,j}$, ${}^{b}L_{i,2j-(i \bmod 2)} = {}^{b}W_{i,j}$, where $1 \le i \le m$ and $1 \le j \le n/2$. The red bits are determined by the ballot image and the paired white bits of the opposite layer: ${}^{x}R \oplus {}^{y}W = {}^{x}B$. The white bits are themselves determined (as is checked in the sixth step above) by the cryptographic pseudo-random sequence functions h and h' (whose composition yields binary sequences of length $mn/2$) from the signature on the serial number q as follows: ${}^{z}W_{i,j} = ({}^{z}d_k \oplus {}^{z}d_{k-1} \oplus \ldots \oplus {}^{z}d_1)_{(mj-m)+i}$, where ${}^{z}d_l' = h({}^{z}s(q), l)$ and ${}^{z}d_l = h'({}^{z}d_l')$. The "dolls" are also formed (and checked in step 6) from the ${}^{z}d_l'$ using the public key encryption functions e_l whose inverse is known to one of the trustees (as will be described): ${}^{z}D_l = e_l({}^{z}d_l' \ldots e_2({}^{z}d_2', (e_1({}^{z}d_1'))$, where $1 \le l \le k$ and for convenience ${}^{z}D = {}^{z}D_k$. [9]

Now consider the tally phase, which takes its input batch from the outputs of an agreed subset of voting instances that reached step 6. For each such instance, only half of ${}^{x}L$ and all of ${}^{y}D$ are included in the tally input batch, comprised of "duos" ${}^{x}B_k = {}^{x}R, {}^{y}D = {}^{y}D_k$, that can be written here as B_k, D_k. Each such duo is transformed, through a series of k mix operations (Chaum [1981]), into a corresponding ballot image ${}^{z}B$. The l'th mix transforms each duo B_l, D_l in its input batch into a corresponding B_{l-1}, D_{l-1} duo in its lexicographically-ordered output batch, by first decrypting D_l using its secret decryption key corresponding to e_l, extracting d_l' from the resulting plaintext, applying h', and finally applying $B_{l-1} = d_l \oplus B_l$. The k'th mix performs the same operation on each duo, but since ${}^{z}B_0 = {}^{z}B$ and D_0 is empty, the result may be written as B. [10]

The k mixes are partitioned into contiguous sequences of four among a set of $k/4$ trustees, where k is divisible by 4. The input batch size is, for simplicity, also assumed divisible by 4. After all the mixing is done, half the tuples in each batch are selected for "opening." This approach is inspired by the work of Jakobsson, Juels, and Rivest [2002]. A random public draw, such as is used for lotto (at which commitment copies of all batches can be distributed on write-once media, such as CD or DVD), allows these choices to be assumed independent and uniformly distributed. The tuples selected for opening depend on the order within each trustee's four mixes: in the first mix, half of all tuples are chosen; in the second, all those not pointed to by those opened in the first mix are opened; in the third, opened are half those pointed to by those opened in the second mix and half that are not; and for the fourth mix, as with the second, those tuples not pointed to by the previous mix are opened. (See Figure 9.)

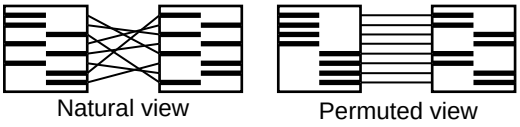

Fig. 9: A trustee's 4 mixes of 8 pairs.

THEOREM: Ballot images are revealed only in encrypted form by any properly-formed selected layer and its resulting processing until they appear in the tally batch.

***Proof** (sketch):* Of the six components of the selected layer $<{}^{x}L, q, {}^{t}D, {}^{b}D, {}^{x}s(q), {}^{x}o({}^{x}L, q, {}^{t}D, {}^{b}D, {}^{z}s(q))>$, only the first depends on the ballot image B. The bits of ${}^{x}L$ are partitioned among those of ${}^{x}R$, that depend on B, and ${}^{x}W$ that do not. Since the ${}^{x}R_{i,j}$ are included in the first input batch, it is sufficient to only consider these batches. The ${}^{x}W$ are the other component of the duos in the input batches, but each is encrypted by some e_i and can therefore be ignored. Each B_l, $1 \le l \le k$, appears in

[9] To keep a voter's choice of layer, which is revealed to the poll workers, from determining the type of ballot image, as well as to prevent bias in voter preference for particular layers, a function of the dolls can determine a mapping between the physical layers and a pair of symbols that the voter chooses between. The symbols would be printed before layer selection in a way that hides them until after the layers are separated.

[10] Multiple doll pairs allow separate ballot images per contest and/or question.

its respective input batch summed modulo 2 with each d_p, $l \le p < k$. Thus, each time any particular bit of B appears in an input batch it appears ⊕'ed with a separate pseudorandom bit that is not present in any following batch.

THEOREM: If, for a selected and an unselected 4-tuple from an instance of step 2, the selected 4-tuple does check in step 6 and there exists a 2-tuple that would check in a step 6 with the unselected 4-tuple, then the doll of the unselected layer, as printed on the selected layer, is correctly formed and determines all white pixels printed on the unselected layer.

Proof *(sketch):* The serial number q and the doll values ${}^{t}D$ and ${}^{b}D$ are all printed on both layers identically, as verified by the voter in step 3. The doll ${}^{y}D$ in the 2-tuple of the unselected layer is properly formed from q, by application of the functions ${}^{y}s$, h, and e, because the unselected 4-tuple would satisfy the consistency check in the hypothetical step 6. Similarly, the white bits ${}^{y}W$ are correctly determined by q, through application of the functions ${}^{y}s$, h, and h'; and since the ${}^{y}W$ would be checked in the hypothetical step 6 as consistent with those printed on the unselected layer, they are also determined by q. So it remains to show that ${}^{y}D$ determines ${}^{y}W$. But the encryption e is bijective and so ${}^{y}D$ determines the ${}^{y}d_i'$ which determine ${}^{y}W$.

THEOREM: The probability that a trustee that improperly forms u distinct duos in any of its output batches will be detected in at least one duo is $1–2^{-u}$.

Proof *(sketch):* The duos in the first batch of a trustee that are opened are selected independently of any control by the trustee and an opened duo is either correct or not. The probability of detection is thus ½ for each improperly-formed duo in that batch. Because those values opened were all correct, the half chosen for the next batch is selected independent of any improperly-formed duo, and so on inductively.

THEOREM: For the mixes of any trustee, the prescribed opening of duos does not reveal a restriction on the correspondence between any individual input and output.

Proof *(sketch):* It is easy to see that the restriction imposed by an odd numbered batch followed by an even numbered batch, a "doubleton" of batches, requires that each of the two known halves of the inputs results in a respective known half of the outputs. (This could reveal something about an individual input and output, such as whether the input could correspond to a particular unique output.) A next doubleton that exactly splits each output partition of its predecessor across its own input partitions does enforce the restriction that exactly half the members of an input partition are in each output partition, but leaves any particular input to the two doubletons free to be any particular output.

Conclusion

The present work demonstrates that voting systems with receipts, like those introduced here, reduce the cost of integrity while raising its level dramatically and making its assurance open to all interested parties. Robustness is similarly more cost-effective and raised to a level where it can also be ensured by voters, assuming they can access a functioning booth. Privacy and secret-ballot protections can easily meet current best practices and are arguably practically optimal.

Since the platform can be open, yielding lower hardware cost, the systems can be more rapidly and widely deployed. (The cost of the printers is expected to be less than the hardware cost saving, not even including savings in total cost of ownership or from multiple uses.) Improved functionality of the systems facilitate accessibility and higher turnout, as well as needed improvements in adjudication. Moreover, and perhaps in the end most importantly, these systems can truly give the kind of voter satisfaction being called for—while particularly improving voter confidence.

The huge expected burst of Federal subsidy for voting systems in the US, and related programs in major states, could end up cementing in place, for a long time to come, the current approach responsible for the very problems the subsidies are intended to address. The real challenge for democracy is whether the process can be opened in time to these new types of systems that are fundamentally better.

Acknowledgements

It is a pleasure to acknowledge Ron Rivest, who served as a superb sounding board for ideas. The "WOTE" workshop was also very stimulating. Later, Jim Dolbear and Lori Weinstein provided a lot of help. Detailed comments from Josh Benaloh, Paul Craft, David Jefferson, Doug Jones, and Andreu Riera as well as feedback from Jeremy Bryans, Dan Boneh's group, Stuart Haber, Robert Naegele, Peter Ryan, and Adi Shamir were also helpful.

References

1. Chaum, D. *Untraceable electronic mail, return addresses, and digital pseudonyms*. Communications of the ACM 24, 2 (February), pp. 84–88, 1981.

2. Jakobsson, M., Juels, A., and Rivest, R.L., *Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking* (submitted).

3. Naor, M. and Shamir, A. *Visual Cryptography*. "Advances in Cryptology – Eurocrypt '94", A. De Santis Ed., Vol. 950 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, pp. 1–12, 1995.

4. Shannon, C. E. *Communication theory of secrecy systems*. Bell System Technical Journal, pp. 656–715, 1949.