Secret-Ballot Receipts and Transparent Integrity

Improving voter confidence and 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 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 others how they voted. These laws are aimed at preventing "improper influence," 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 might do to surreptitiously change votes. Not only can receipts and this level of integrity enhance voter confidence, but they eliminate the need for trusted voting machines.

The current requirement that each voting machine be highly-trusted has real disadvantages. Both private and public sectors have generally rejected such proprietary "black box" technology in favor of open-platform solutions, which are suitable for the receipt system. The standards and high volumes of open platforms allow much lower cost and higher quality, with better availability and upgradeability, as they create competitive markets in hardware and software components. Receipts also improve reliability over trusted devices, not only because

failures are detected during voting before it is too late to prevent lost votes, but also because receipts can ensure votes count no matter what happens to the machines or "data cartridges" currently used to transport votes. And open platform solutions can, instead of being stored in specially protected warehouses most of the time, be used for various purposes year-round in public places like schools and libraries.

Inability under the current approach to reconcile secrecy and security concerns has led to problems beyond trusted devices and lack of receipts. Destruction of audit-chain data, for example, is mandated by current design specifications for privacy reasons; receipts allow effective system audit, even though all local data is either published or expunged. 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 increase turnout by letting people vote with full privacy from any polling place within their county, independent of their home precinct. Courts can also surgically add or remove the votes of particular categories of voters; being unable to do this today forces them to either throw out all ballots or determine winners themselves.

Voting with the new approach

After making your choices on a touch screen or the like, when using this new approach to vote, a small printer that looks like those at cash registers prints the body of your receipt. This printout shows your vote and only your vote. The names of those candidates you chose, office sought and party affiliation, would be listed as well as your choice on any ballot questions. (Please see figure 1.) Included would be any allowed "write-ins" or choices you made, such as with "open primaries" or "instant-runoff voting". There could also be warnings about contests or questions not voted. (As detailed later, there is a security feature, such as an unbroken blackbackground around the text, that voters should also check for at this point.) You are then asked whether or not you agree with the receipt so far; and, if you don't agree you can amend your vote and try again.

Fig. 1: Example 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 the two-layer receipt. Overall security hinges on your freedom to choose, even though it is an arbitrary decision, which layer you want to keep. Once you've chosen, a further inch or so is printed and the then complete form is automatically cut off and presented to you. (See figure 2.)

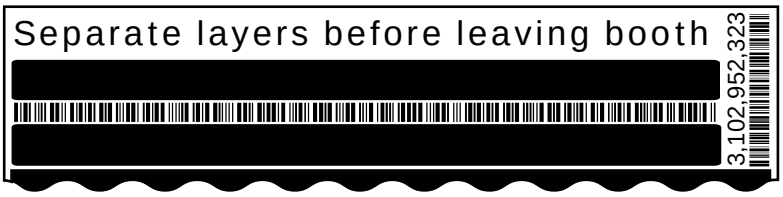

Fig. 2: Last inch printed, not yet separated.

As you separate the two layers, the votes on each transform into an unreadable and seemingly random pattern of tiny squares printed on a translucent plastic material—it was the light passing through the combination of still-laminated layers that showed your choices. The special receipt printers used differ from ordinary single-color receipt printers mainly 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.

The last inch printed, however, contains some text messages that are readable only when the layers are viewed separately. Whichever layer you had selected to keep, 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 "Voter must surrender this layer to poll worker" (see Figure 4). On the way out, you hand the poll worker the layer marked for them. They make sure they got the right layer and as you watch they insert it into a small transparently-housed paper shredder in which it is destroyed. The receipt layer you took, however, includes your votes in a safely-coded form and is sent electronically by the voting machine for posting on the official website.

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

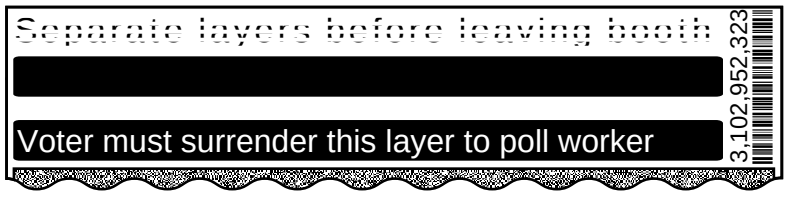

Fig. 4: Separated layer to be shredded.

Outside the polling place 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 coded data to a consistency check, detailed later, and confirming them when online). If they were ever to detect an invalid receipt, incorrect operation of election equipment would be irrefutably indicated (a false alarm could be immediately dispelled by any other checker), hopefully before any unwitting recipients of invalid receipts had already left the polling place. You could send your receipt to your favorite political party or group for checking. You can even, on the official website, look up the page for the range of serial numbers that includes your receipt, and check for yourself that it has been posted correctly.

After the polls close, and all agreed receipts are posted on the website, a series of encrypted process steps leading up to the tally is also posted. Then randomly-selected samples of it are decrypted and posted. The choice of samples is made so that it does not reveal so much information as to compromise privacy. The samples do reveal enough, though, that anyone can run a simple open-source program, available from major political parties and other sources, that checks them against the published process steps to be quite confident that the tally correctly resulted from exactly the votes encoded in the posted receipts.

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? First all three questions are considered for a general audience. After the first question, answers to the second and third questions are combined for each of three aspects: the receipts, the tally process, and the cryptography. Finally the system is described using some mathematical formalism and the properties are proved.

Properties of the new system

- If your receipt is 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 posted, it would be 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.
- There are only two ways that a system, no matter how incorrectly it operates, would have a chance of changing a voter's ballot without being detected: (1) printing a bad layer and hoping that the voter chooses the other layer; or (2) creating a bad sub-step

among the tally process steps. 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. Larger examples are 10 improper ballots will be discovered in all but less than one in a 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 combination readable and the separated layers not is simply that the printer has printed the black on them in different patterns. If you were to look at a single square or "pixel" of the receipt, what you would see is that each layer covers that pixel with one of two symbols. Only two "pixel symbols" are used. The main thing about the two pixel symbols is that they are opposites of each other: one positive and the other negative; where one is clear the other is black and vice versa. So when the same pixel symbol is printed on both layers, and they are aligned, one directly above the other, light is able to pass through all the parts where they are clear. But when the two different patterns are used, each on its own layer, any clear on one is blocked by black on the other and so the combination appears totally opaque. (See figure 5.)

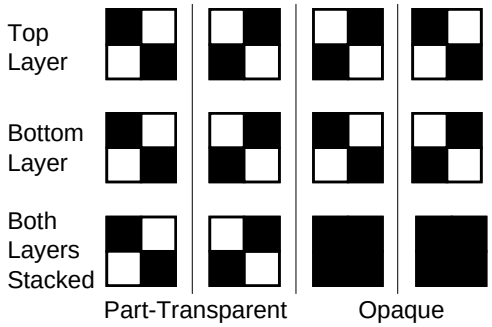

Fig. 5: The two pixel symbols and the effect of overlaying them.

This technique can be used to encode information on a first sheet so that it can only be read by those with a second sheet that is a key (as has been suggested by Naor and Shamir [1995]). You start with a key sheet that is divided into pixels, the pixel symbol for each pixel being one of the two, but with each chosen apparently at random. Then to encode your message in a second sheet you simply choose each of its pixel symbols accordingly: if you want light to shine through for that pixel location, you choose the same pixel symbol as is printed there on the key sheet; and if you don't want light to come through there, you choose the opposite symbol.

In most printing technologies today, ordinary text is simply printed by breaking each character up into a grid of pixels, some are printed as black while the others are not printed at all. For a receipt: instead of leaving the background unprinted, opaque pixel symbol

combinations are used; and instead of black ink for the letters, partially translucent pixel symbol combinations give a gray effect (where brightness of the gray depends on backlighting).[1] Thus, each pixel of a letter that would be black in a newspaper would be printed with matching pixels; and everywhere without ink on the newspaper would be with differing pixel symbols. (See Figure 6.)

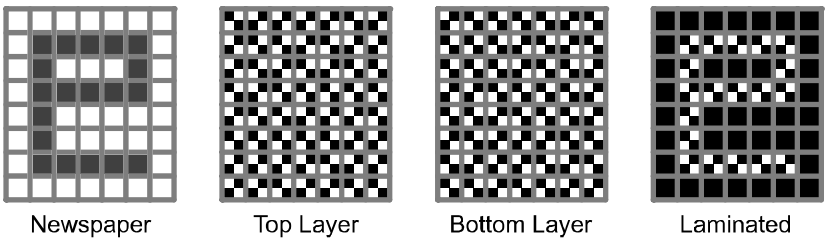

Fig. 6: The letter "e" in standard and receipt printing.

For the receipts, the pixels on the code sheet are in effect random (as explained later). These key pixels will be called "white" pixels, suggesting that they are somehow neutral and safe since they are not based on any vote information. The pixels that are opposite the white pixels, those chosen to encode the text of the candidate names when combined, will be called "red" pixels, suggesting their somewhat more sensitive nature. When the white pixels are laminated with the red pixels, the result, printed in partly-translucent gray on a black background, is the "ballot image"—a definition of the voter's vote that has been accepted by the voter in a visible text form.

If all the red pixels were on one layer and all the white on the other, you could not take your choice of layers. So each layer is arranged like a checkerboard of red and white squares on the tablecloth in a typical bistro. To line each red pixel on one layer up with a white pixel on the other layer, one checkerboard is in a standard orientation and the other is as if rotated a quarter turn. (See Figure7.) Even half the pixels will uniquely determine each character in the ballot image, especially as more and finer pixels are used.

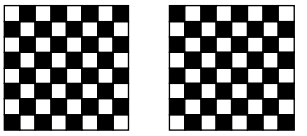

Fig. 7: Red pixels opposite white pixels

The coding technique, called the "one-time pad," has been proven by Shannon [1951]to be unbreakable, assuming the key is random. The key here, the white pixels, is not random but in practice believed "indistinguishable" from random except to a set of "trustees" who are collectively the guardians of ballot secrecy (as will be explained later). Intuitively, if you have

[1] So-called "direct 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 "dry-peal" adhesive keeps the two layers together while loosing all tack once delaminated.

only the single-layer receipt and are staring at a particular “red” pixel on it, all you learn is that the combined visible pixel would be clear if the opposite white pixel were the same as that red pixel and opaque if it were different—but knowing nothing about the apparently random white pixel’s value means you cannot infer anything about whether or not the combination would be clear.

Central to integrity of the system is ensuring that the ballot image is correctly contained in the receipt. The red and white pixels of each layer are seen optically and agreed on by the voter. The only other part of a receipt that determines the ballot image is the coded forms of the white pixels, called “coded dolls” (as detailed in the next section). There are two different coded dolls, one that encodes the white pixels of each layer. Copies of both dolls are printed on one layer but the pixel symbols are flipped on the opposite layer so that the result combines to opaque pixels. Thus, when the voter checks that the opaque background comprised of these pixels is unbroken (as mentioned earlier), they are actually verifying that both layers contain exact copies of the same dolls. The correctness of a doll can be checked using a “seed” number (detailed later), which allows the whole doll to be re-constructed and checked completely against its white pixels. The seed is printed on the same layer together with the doll’s white pixels, so that the combination can be checked if the layer is selected. Although the doll used to encode the ballot image on the layer the voter keeps is itself never actually checked, layers printed with dolls that would not check must be printed before the voter chooses which layer to keep, and so would be caught half the time.

How and why tabulating works

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 agreed and posted. Before beginning processing to produce the tally, each receipt is stripped down to its essential “pair” of values—the red pixels and the coded doll of the opposite layer. These pairs are then processed by the trustees in a verified way to yield the final tally—the list of ballot images.

The core of the tallying system, the coded doll mentioned above, is analogous to a special version of the famous “Russian nesting dolls,” one nested inside the next in series. These particular nesting dolls are like KGB agents, each holding her unique random code sheet in her hands. The sizes of the dolls are standardized and, for each size, there is a different key that

unlocks all the dolls of that size. Consider the trustee that has the key for, say, the 10 inch dolls. To process such a doll, he first unlocks the 10 inch doll using his key and removes the 9 inch doll it contains. At this point he can see two code sheets, one from each doll. He combines the two sheets to produce a new code sheet, pixel by pixel, as follows: where light shines through the combination of sheets, one pixel symbol is printed on the new sheet and where it does not, the other symbol is printed.[2] He then places this combined code sheet in the hands of the 9 inch doll and destroys the 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 outputs them as a batch. Before the batch is output, however, it is randomly mixed, so that nothing about the correspondence between the dolls in the input and those in the output is revealed by the order in which they appear in the batches.

Now consider how all this can be put to use. Suppose the original doll maker faithfully chooses all the sheets in all the dolls at random, but for one complete big doll keeps copies of all the sheets of the dolls it contains. Instead of keeping these each on a separate sheet, he can combine them into a single sheet, one pair of sheets at a time (or all at once using some simple arithmetic) to form a single "white" sheet. This is the same sheet as will be seen that would result were that doll to be processed successively by all the trustees.

Now further suppose the doll maker wants to publish information contained on a "blue" sheet without it being traceable back to a particular one of the big dolls. What he does is determine a "red" sheet such that when it is optically combined with the white sheet, it results in the blue sheet. Then the complete big doll is given the red sheet to hold (all the other big dolls get different red sheets). Once the batch passes through processing by all the trustees, the final output batch includes one of the tiny solid-wood dolls holding exactly the information on the blue sheet. It can be read by laminating a sheet that has all one type of pixel symbol everywhere.

Actually, if all dolls were made this way, they could each be used to publish anonymous messages by anyone who forms a red sheet from a big doll's white sheet and enters the pair comprising the doll and sheet into the initial input batch. Moreover, only if all the trustees were

[2] 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.

to divulge detailed records of what they did, could anyone trace individual blue sheets back to the original big dolls.

The analogy's red and white sheets correspond, of course, to the red and white pixels. The blue sheet is then the ballot image. 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 doll using the respective published keys, but only each trustee can in turn strip off the layers it has the keys to.

Suppose further that a way is needed so that the trustees would likely get caught if they were to improperly change sheets during processing. An inspection would be conducted after the processing. It would require the trustees to release complete and detailed video tapes of the processing, but only for selected dolls. To prevent these from allowing the blue sheets to be traced backwards to the respective red sheets, each trustee processes, say, two successive batches of the series. After all the processing, a lottery-style draw selects half the dolls in the first input batgch to have their videos released. These dolls are spared having their videos of the second processing revealed, but none of the other dolls are. Exact tracing is thus prevented because only one video is released per doll. Each sheet a trustee improperly changes, however, has a 50% chance of being selected for release on video, so the odds stack up pretty fast! With encryption instead of a videotape, the code sheet originally held by the doll that is output is all that has to be released, because it is easy to check that applying the public key to the combination of original sheet and output doll yields the input doll.

The "coded doll" mentioned earlier is just the digital version of the big doll without its red code sheet. Actually, the coded dolls are not freely chosen by the voting machine, but are determined cryptographically from a seed that is a unique kind of signature on the ballot serial number. That way, the coded dolls are guaranteed to be unique, cannot leak any information, and, when their layer is checked for consistency outside the polls they can be completely generated afresh and checked as corresponding exactly to their white pixels.

The codes used and what they achieve

A "digital signature" is 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, when the verifier outside the polling place scans your receipt, part of the consistency they can check for immediately is whether the signature is: valid, from an authorized voting booth, and consistent with everything printed. If it does not check, the physical receipt itself is immediate evidence of the failure. If the receipt does check, however, you know that the receipt cannot be credibly denied a place among the ones to be posted and tallied.

Cryptographic techniques are generally divided between those that are "unconditionally secure" and those that are merely "computationally secure". The former are rare, like the one-time pad with random key described above, and cannot be broken, even if an adversary were to apply infinite computing power. Almost all cryptographic algorithms, however, 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 do such computation using resources available today (since many systems, including international high-value wire transfer, rely on such codes). Such standard cryptographic building blocks, like those used widely by browsers when accessing secure websites today, are enough to build the systems as described here.

Computational systems are used here to protect ballot secrecy and privacy. In view of technological advances, such as miniature cameras, sensors and transmitters, the privacy and secrecy of voting in polling places cannot practically be maintained by any absolute standard today. The use here of codes protecting receipts and posted data, 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: 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, are subject to ballot number recording or marking and automatically captures fingerprints; and theoretical limits are generally believed to force a choice in applicable cryptographic systems between unconditional integrity and unconditional privacy.

Thus the present system is arguably optimal: it protects privacy, of votes in receipts and during processing, computationally, according to best practices; while 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 the proper public keys of the system, 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 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 function h (which outputs binary sequences of length $mn/2$) as follows: ${}^{z}W_{i,j} = ({}^{z}d_k \oplus {}^{z}d_{k-1} \oplus \ldots \oplus {}^{z}d_1)_{(mj-m)+i}$, where ${}^{y}d_i = h({}^{y}s(q),i)$. 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): ${}^zD_l = e_l({}^zd_l \ldots e_2({}^zd_2,(e_1({}^zd_1))$, where $1 \le l \le k$ and for convenience ${}^zD = {}^zD_k$.[3]

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 ${}^xL$ and all of ${}^yD$ are included in the tally input batch, comprised of "pairs" ${}^xB_k = {}^xR, {}^yD = {}^yD_k$, that can be written here as B_k, D_k. Each such pair is transformed, through a series of k mix operations (Chaum [1981]), into a corresponding ballot image ${}^zB$. The l'th mix transforms each pair B_l, D_l in its input batch into a corresponding B_{l-1}, D_{l-1} pair 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, and then applying $B_{l-1} = d_l \oplus B_l$. The k'th mix performs the same operation on each pair, but since ${}^zB_0 = {}^zB$ and D_0 is empty, the result may be written as B.

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, 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 8.)

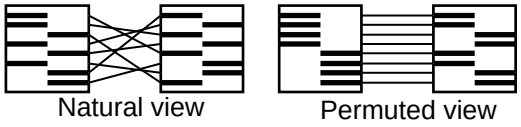

Fig. 8: A trustee's 4 mixes of 8 pairs.

THEOREM: Ballot images are revealed only in encrypted form by any properly-formed selected layer and its resulting processing.

***Proof** (sketch):* Of the six components of the selected layer $<{}^xL,q,{}^tD,{}^bD,{}^xs(q),{}^xo({}^xL,q,{}^tD,{}^bD,{}^zs(q))>$, only the first depends on the ballot image B. The bits of ${}^xL$ are partitioned among those

[3] 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 pair of symbols, along with their mapping to the respective physical layer, that the voter chooses between and that would be printed before layer selection in a way that hides them until after the layers are separated.

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 pairs 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 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 pair of 4-tuples from an instance of step 2, one is selected and 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 ${}^{y}s$ 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 pairs in any of its output batches will be detected in at least one pair is $1–2^{-u}$.

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

THEOREM: For the mixes of any trustee, the prescribed opening of pairs 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 "duo" 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 duo 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 duos free to be any particular output.

Conclusion

The present work demonstrates that receipt-based voting systems offer much higher reliability and unconditional integrity. Since the platform can be open, yielding lower cost and greater scalability, they can be more rapidly and widely deployed. These systems can also facilitate higher turnout, as well as needed improvements in adjudication. Moreover, and perhaps in the end most importantly, they can greatly improve voter confidence.

The huge anticipated 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 may be whether the process can be opened in time to new types of systems that are fundamentally better.

Acknowledgements

To be supplied.

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*, in "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, pages 656–715, 1949.