Sigma Ballots

Stefan Popoveniuc[1] and Andrew Regenscheid[2]

[1]KT Consulting
Gaithersburg, MD, USA
stefan@popoveniuc.com

[2]Computer Security Division
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD, USA.
andrew.regenscheid@nist.gov

Abstract: We present Sigma ballots, a new type of ballot designed to be used in secure elections. Sigma ballots use the random order of candidates introduced by Prêt à Voter, combined with the confirmation codes of Scantegrity II. These ballots can be produced by a DRE machine with a slightly modified VVPAT, or can be similar to optical scan ballots. Sigma ballots work in conjunction with existing publicly verifiable tallying schemes to allow for end-to-end verifiability. The advantages of Sigma ballots include an easy check for correct printing, the possibility of keeping a fixed order of candidates when selections are made, automated creation of receipts, no extra marks added to the ballot after it is cast, the ability to be hand counted, and voters only needing to know a valid confirmation code to file a complaint.

1 Introduction

A new class of voting systems was developed in the last couple of years which allows for a unique level of public scrutiny of the declared totals. These systems, known as *end-to-end verifiable voting systems*, allow voters to check that their ballots were cast and recorded as they intended, and allow anyone to check that all the recorded ballots have been correctly tallied. They offer security properties radically different from any voting system used in elections today.

While, in theory, many end-to-end verifiable voting systems have great properties, in practice, they suffer from known weaknesses. Some of them may be difficult to use by voters [PH06], others may be difficult to implement in practice [CD04], some may be too slow for very large elections [CRS05], while others may be vulnerable to attack [CD08].

Recently, binding elections have been run using end-to-end verifiable systems [EA07, AB09]. Scantegrity II [CD08] has been used in a public election, to elect the mayor and city council of Takoma Park, MD. While Scantegrity II has many desirable security properties, it suffers from a series of problems, many of them being acknowledged after the Takoma Park election.
In this paper, we present Sigma ballots, a new type of ballot which can be used in conjunction with existing publicly verifiable tallying schemes to create end-to-end verifiable voting systems that are not vulnerable to many attacks faced by existing systems.

1.1 Motivation and Related Work

A number of end-to-end verifiable voting systems have been proposed [EA07, AB09, CD08, AR06, CRS05]. Many of these systems have known vulnerabilities.

A well-known attack on Scantegrity II is to misprint the ballots. For example, if we assume that a certain voter is going to vote for Alice, but the inside attacker is a supporter of Bob, then the attacker may print next to Alice the confirmation code that corresponds to Bob. The voter fills in the oval next to Alice's name and gets a confirmation code that she thinks is for Alice, when in fact it is for Bob. The tallying mechanism is going to correctly transform this confirmation code into a vote for Bob. This attack is possible because voters are not able to directly distinguish improperly printed Scantegrity II ballots from correctly printed ones.

The typical way of mitigating this attack is to allow the voter to choose two ballots, one to vote, and one to spoil and audit the printing on it. This approach is theoretically sound, but in practice there are multiple disadvantages. First, the approach adds time and complexity to the voting process. Second, voters need to take the fully marked ballots home, and check them against the data on a bulletin board. This potentially violates current election practices, as ballot accounting procedures in many jurisdictions prevent voters from leaving the polling place with a ballot, even spoiled ballots. Third, the approach is highly dependent on procedures followed both by the voter and election officials [KJ07].

Another option is to have a designated auditor that comes and chooses a random set of ballots to be audited for correct printing. This solution requires a trusted auditor, as well as a secure chain of custody for the audited ballots.

The same print audit problem exists in other voting systems, e.g., Prêt à Voter [CRS05], Scratch&Vote [AR06], or, more generally, voting systems in which the ballot does not consist of two or more symmetrical parts, such as PunchScan [PH06].

Another issue with Scantegrity II is that voters are asked to create their receipts by hand. They have to write down the serial number of the ballot along with the confirmation codes for each ballot question. This task can be time consuming and error-prone.

A third security problem identified with Scantegrity II is the possibility of the voting system transforming a no-vote into a valid vote, or a valid vote into an over-vote, by adding extra marks to the ballot after it was cast. Since the voter cannot prove that she does not know the codes for the marks she did not make, the voter cannot prove that she was not the one that made the marks which were in fact added by the system afterwards. This security issue is unique to end-to-end verifiable voting systems where the voting receipt is a proof of knowledge, rather than a partial copy of a cast ballot.

1.2 Contribution

This paper presents Sigma ballots, a new type of ballot to be used to create secure voting systems. Sigma ballots use the random order of candidates introduced by Prêt à Voter, combined with the confirmation codes of Scantegrity II. These ballots can be produced by a DRE machine with a slightly modified Voter Verifiable Paper Audit Trail (VVPAT) printer, or can be similar to optical scan ballots. For illustration purposes this paper provides an example (see section 5) for how to implement verifiable ballot tallying using techniques from Scantegrity II [CD08].

Similar to PunchScan, but without suffering from its indirection problems, the proposed Sigma ballots are two parts symmetrical ballots, with any of the parts containing the same amount of information. The voter may use any of the parts to check for correct printing, without being able to prove how she voted.

Sigma ballots can be used to automatically create a receipt, without the voter needing to write down anything by hand.

Sigma ballots also solve the problem of improperly invalidating cast ballots by giving the voter a digitally signed receipt, covering all and only the selection on the voter's ballot. The voter can now prove that extra marks have been added to her ballot after it was cast by presenting her signed receipt.

2 Description of Sigma ballots

We start by describing what a Sigma ballot looks like. In section 3, we detail how the Sigma ballot can be created using either a DRE with a VVPAT printer, or an optical scan system.

The design of the Sigma ballot uses ideas from the Prêt à Voter ballot and the Scantegrity II confirmation codes. Sigma ballots are filled-in ballots, clear text, with marks next to the candidates the voter selected. Voters can inspect a Sigma ballot to verify that their choices are correctly represented, by checking the names next to the marks. Also, Sigma ballots can be counted by hand.

Figure 1 shows a Sigma ballot. On the left side of the ballot is a list of candidates, in a permuted order on each ballot[1]. The order of the candidates on each ballot is publicly committed to before the election and may be different for different ballots. On the right, there is a mark for each candidate the voter selected.

The voter can check that the marks appear only next to the candidates she voted for. If not, the voter can start creating another Sigma ballot (no harm was done). This check is similar to asking the voter to verify that the Voter Verified Paper Audit Trail (VVPAT) contains her choice in a DRE+VVPAT system.

Each mark that appears next to the candidates has a confirmation code assigned to it. All the confirmation codes are printed at the bottom of the ballot. A public commitment ties the confirmation code to the marks next to the candidates, i.e., to the position where marks appear at.

Instructions are printed at the bottom of the ballot about how the voter can check that her vote was correctly recorded. There are also two bar codes containing digital signatures. One signature covers the confirmation codes and the order of the candidates (the left side); the other covers the confirmation codes and the position where marks appear at (the right side).

Like in Scantegrity II, the voter only sees the confirmation codes for the candidates she selected, creating a knowledge-based receipt. If the voter notices that her confirmation codes are not correctly posted on the public bulletin board, knowledge of a valid confirmation codes is sufficient to file a complaint.

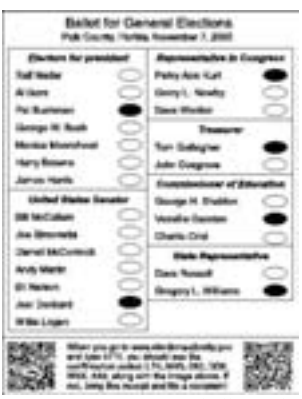

Figure 1: A Sigma ballot. The order in which the candidates are printed may be different on different ballots. The confirmation codes are not associated with candidates or marks.

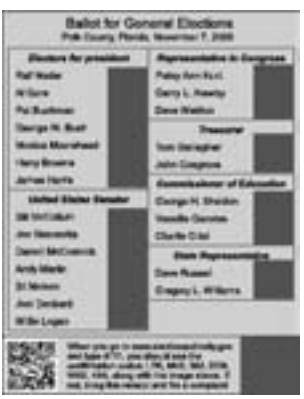

Figure 2: Receipt produced by photocopier 1. The order of the candidates is visible, but no marks are visible, so an observer cannot tell how the voter voted.

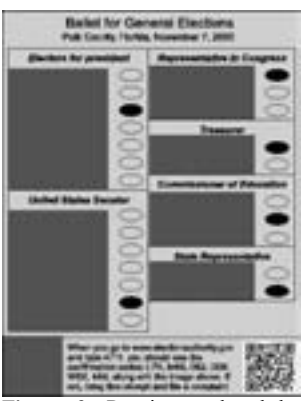

Figure 3: Receipt produced by photocopier 2. The marks are visible, but the order of the candidates is hidden, so an observer cannot say which candidates the marks correspond to.

[1] The name "Sigma ballot" comes from having a permutation represented by the Greek letter σ.

There are two photocopiers in the polling place, which are used to produce a receipt from a Sigma ballot. On her way out, the voter may choose one of the two photocopiers and place her ballot in it. The scanning portion of the photocopiers is partially blackened out by an opaque template (i.e., black tape), such that, for the first photocopier, the template hides (i.e., does not allow to be copied) the portion with the marks (Figure 2). For the second photocopier, the template hides the portion with the order of the candidates (Figure 3). The copy produced by the photocopier becomes the voter's receipt, while the Sigma ballot is deposited into a ballot box.

If the voter chooses the first photocopier, she obtains the order in which the candidates appeared on the voted ballot along with the confirmation codes (see Figure 2). Since no marks for any of the candidates are visible, and since the confirmation codes may be different for different ballots, inspecting this receipt does not reveal the choices the voter made. If the voter chooses the second photocopier, the voter obtains the position of the marks along with the confirmation codes (see Figure 3). Since the order of the candidates may be different on different ballots, the positions where the marks appear do not reveal the chosen candidates. Therefore, no matter which photocopier the voter uses, she gets a receipt that does not reveal how she voted.

The bar code at the bottom of the ballot contains a digital signature of the receipt to avoid voters being able to manufacture fake receipts, and to avoid having the system adding more marks after the ballot is cast. The verification of the correctness of the digital signature is part of future work.

To simplify things, two digital signatures are on the initial Sigma ballot. Depending on which photocopier is used, one of them is covered, such that the receipt only contains the appropriate digital signature.

All the receipts are posted on a public bulletin board and the voter may check it and compare her receipt to the posted one. If the receipt does not appear on the bulletin board, or if it is not correctly posted (e.g., different confirmation codes, different order of the candidates, or different position of the filled-in marks), the voter can show her physical receipt, which is irrefutable proof that the bulletin board contains invalid information. The posted receipts can be used by a publicly verifiable tallying scheme (see section 5) to produce vote totals which are proven to come from the posted receipts, and thus from the choices the voters made.

2.1 Pre-election setup

Before the election, a set of commitments is published for each ballot. For each confirmation code, a commitment that ties the confirmation code to a coded vote is made. The coded vote is the input to a verifiable tally mechanism (can be a mixnet [PH06] or homomorphic tallier [AR06]).

For each ballot, a commitment to the order of the candidates is published before the election. If the voter uses the first photocopier getting her receipt with the order of the

candidates, this commitment is opened, and anybody (not just the voter), can check that the receipt posted on the public bulletin board is consistent with the commitment that ties each candidate with the position it appears at.

Commitments that tie marks at certain marked positions to confirmation codes are also published for each possible marked position. For each confirmation code on each ballot, the system publishes a commitment that ties the confirmation code to the position that should be marked when this confirmation code is printed on the chit. If the voter uses the second photocopier, the system opens the commitment that binds the confirmation code on the receipt to the position of the mark on the receipt. Anybody can check that, on the posted receipts, the marks appear at the positions indicated by the opened commitments and that the confirmation codes do correspond to these positions.

We assume that the system that produces the Sigma ballot does not know a priori which photocopier is chosen by the voter. If, on a particular ballot, the system modifies either the order of the candidates, or the confirmation codes, then the system has a 50% chance of not getting caught (because there is a 50% chance that the voter chooses the photocopier that makes a copy of the part that was not cheated on). Assuming the voters' choices of photocopiers are independent, the probability of not detecting any misprinted ballots decreases exponentially with the number of misprinted ballots.

2.2 Advantages of Sigma ballots

Sigma ballots have three major advantages. First, it should be relatively easy for the voters to check if the paper ballot contains a vote for the candidate that they voted for: locate a mark and simply read the name of the candidate to the left of the mark. Second, by giving the voters the choice to put their ballot in any of the two photocopiers, the voter performs an automatic print audit of their ballot. In some cases the voters check that the order of the candidates is correct, and in the other cases the voters check that the confirmation codes correspond to the marked positions. Third, the voter does not have to create a receipt by hand, since the confirmation code is already printed on the stub of the ballot, which is photocopied and included in the receipt.

Voters that are not interested in getting a receipt can simply ignore the photocopiers and walk out, but not before depositing the Sigma ballot into the ballot-box. To ensure that the ballots are correctly printed, it is not necessary that all voters get a receipt from one of the photocopiers, but only that a statistically significant, unpredictable fraction do.

Depending on the predictability of the confirmation codes, the lack of a paper receipt may *not* prevent the voter from checking the public bulletin board, just like in Scantegrity II. If correct confirmation codes on any given ballot are difficult to guess by voters, then the voter's knowledge of the confirmation codes may be sufficient to file a complaint if the confirmation code is not correctly posted on the bulletin board. A voter that provides a confirmation code that is unpredictable, and that was previously committed to, has probably discovered that her correctly cast ballot is not correctly posted on the bulletin board.

3 Producing the Voted Ballot

Sigma ballots are ballots that are already filled-in; they already contain the will of the voter. In this section, we present a few ways in which Sigma ballots can be created. One option is to use a DRE connected to a printer (VVPAT). A second option is to have an optical scan paper ballot that is a combination of Prêt à Voter and Scantegrity II ballots.

3.1 Ballot Marking Devices–DREs with VVPAT

Probably the easiest way to produce Sigma ballots is to use a ballot marking device. This device can look like a DRE, where voters can make their selection using a touch screen and have the liberty to choose the ballot language, font size, contrast, etc. The same device can serve multiple ballot styles.

The order in which the candidates are presented to the voter can be standardized and can be the same for all ballots (such that it is consistent with local electoral law). After the voter made all her selections and inspected the review screen, she presses the "Print Sigma Ballot" button. The DRE has a regular office printer attached to it which prints a Sigma ballot. The voter inspects the print-out to see if marks appear next to the candidates she voted for. If this is not the case, she spoils the Sigma ballot and uses the DRE again to make her selections and to produce another Sigma ballot. Otherwise, the voter walks over to the area where the photocopiers are, following the process described in section 2.

The Sigma ballot can be viewed as a Voter Verifiable Paper Audit Trail (VVPAT). But the VVPAT is not printed under glass and the voter can photocopy part of it. The DRE always prints the Sigma ballot, just like it always prints a VVPAT, regardless if the voter will take the Sigma ballot to the photocopier or not. As soon as the Sigma ballot is out of the printer and is inspectable by the voter, the voter can simply memorize the confirmation codes next to the selected candidates. Later, if the voter does not see the confirmation codes on the public bulletin board, she may still file a complaint, and knowledge of the codes may be sufficient. The voter only knows the confirmation codes for the candidates she selected, thus knowing some other valid confirmation code would mean that either the vote guessed the code (which should be difficult if the codes are unpredictable), or the bulletin board contains an incorrect confirmation code.

By looking at the Sigma ballot, the voter gets a receipt based on "something you know," i.e., the confirmation codes. The voter may also get a "something you have" receipt, a paper receipt, if she uses one of the photocopiers. The extra check that the paper receipt allows the voter to do is to ensure that the association between candidates and confirmation codes on her Sigma ballot is correct. This association has two parts: candidates to positions and positions to confirmation numbers. The voter can check that either the order of the candidates is correct, or that the marks are correctly assigned to the confirmation codes.

Because of the digital signature, neither the voting system nor the voter can add more marks to the receipt after it comes out of the photocopier. Having a physical receipt (as opposed to a "something you know") precluded the voting system from adding more marks to the cast ballots, which may be used as an attack to transform a blank ballot into a voted one, or a voted one into an over-voted one (as is the case in Scantegrity II).

At the end of the voting day, the DREs can provide tallies for fast reporting. Moreover, the Sigma ballots can be used in a hand recount, since each Sigma ballot is a clear text ballot. A third count is provided by an existing publicly verifiable tallying mechanism such as the ones used by Scratch&Vote [AR06], PunchScan [PH06] or the Scantegrity II [CD08] (presented in section 5).

3.1 Optical Scan

A Scantegrity II [CD08] ballot is an optical scan ballot in which, next to each candidate there is an oval printed in invisible ink. The voter fills in the oval next to her desired candidate, and the chemicals in the pen react with the invisible ink printed in the oval, such that the ovals turns mostly black, except for a confirmation code that stays white, and thus becomes visible. The voter can record the confirmation code, in essence creating a receipt for her vote. The paper ballots can be scanned or counted by hand.

A practical problem with a Scantegrity II ballot is ensuring that codes are printed next to the correct candidates. Scantegrity II allows the voter to receive two ballots, one to fully mark and audit the printing on it, and the other one to cast. In practice, since performing the print audit is an extra burden, voters do not perform print audits. In this case, a designated auditor is needed for performing the print audit, which may be problematic.

A Sigma ballot is a Scantegrity II ballot with candidates in randomized order. This solves the print audit problem by allowing the voter to choose one of the two photocopiers to create her receipt and check the correctness of half the printing on her ballot.

Another shortcoming of the Scantegrity II ballots is that voters must create their own receipts, by writing down the confirmation numbers revealed when marking the ovals, or remembering them. Sigma ballots address this too. Assume the voter is allowed to place her Sigma ballot in one of the photocopiers, get her copy, but also get back the Sigma ballot. The voter then deposits the ballot she got back into an optical-scan system, which has a printer attached to it. The voter places the copy she got from the photocopier in the paper feed of this printer, such that the printer will print on this copy. The optical scanner detects the marks from the ballot and prints the confirmation codes on the copy that is in the printer. Therefore the voter does not need to write down the confirmation codes by hand.

The above technique is based on the assumption that the scanner does not know if the voter used the first or the second photocopier (i.e., the photocopier cannot signal the scanner). If the voter used the second photocopier, the copy already contains the confirmation codes, since in a Scantegrity II ballot the codes are revealed when the oval is filled-in by the voter. If the scanner would produce different confirmation codes, then the voter would have irrefutable proof that the scanner printed incorrect confirmation codes. If the voter used the first photocopier, the copy contains the order of the candidate, but without any confirmation codes. In this case the scanner can print incorrect confirmation codes without being detected. But since it is assumed that the scanner does not know what information is already printed on the voter's copy, the chance of printing incorrect confirmation codes and not getting caught decreases exponentially with the number of ballots cheated on.

One can also envision a system in which the scanner is used before the photocopiers: the voter puts the Sigma ballot in a scanner that checks for under-votes and over-votes and also prints the confirmation codes at the bottom of the ballot (a copy of the ballot can also be produced instead of printing at the bottom of the original ballot). Then the voter gets back the ballot and goes to one of the photocopiers, like in the DRE setting. The voter always gets the confirmation numbers, since they were printed by the scanner at the bottom of the ballot. The voter can also check that the scanner wrote the confirmation codes correctly (i.e., it detected the marks correctly), by simply inspecting the output of the optical scanner.

4 Formalization of Sigma Ballots

We present a formal model of Sigma ballots. For simplicity, we model a single race and we assume that there is a candidate "No Vote," which is selected by default if the voter does not select any candidate. Let $\boldsymbol{C}$ be the set of candidates. Let c be the cardinality of the set $\boldsymbol{C}$, and let[2] $\boldsymbol{Z}_c$ be the set of numbers from zero to $c-1$. Let $\boldsymbol{N}$ be the set of all possible confirmation codes, and let $\boldsymbol{E}$ be the set of coded votes that a publicly verifiable tallying scheme takes as input. We assume that the cardinality of $\boldsymbol{N}$ is large.

A Sigma ballot is defined by three functions:

1. $\sigma : \boldsymbol{C} \rightarrow \boldsymbol{Z}_c$ representing the association between the candidates and the position they appear at. σ is a bijective function.
2. $\pi : \boldsymbol{Z}_c \rightarrow \boldsymbol{N}$ representing the association between positions and confirmation codes. π is an injective function. We assume that it is difficult to guess $y \in \boldsymbol{N}$ such that $\exists!\ x \in \boldsymbol{Z}_c$ such that $\pi(x)=y$.
3. $\phi : \pi(\boldsymbol{Z}_c) \rightarrow \boldsymbol{E}$ representing the association between confirmation codes and coded votes. ϕ is an injective function.

A Sigma ballot transforms a clear text vote (a candidate) into a coded vote by composing the three functions $\phi \circ \pi \circ \sigma$.

[2] We abuse the $\boldsymbol{Z}_c$ notation to simply mean the set of numbers from zero to $c-1$ instead of the set of residues modulo c.

The protocol follows the following steps, for each ballot:

1. The election authority computes in secret ϕ, π and σ.
2. The election authority computes and publishes:
    a. A commitment to the entire function σ.
    b. For each $x \in \boldsymbol{Z}_c$, a commitment to $(x, \pi(x))$
    c. For each $x \in \pi(\boldsymbol{Z}_c)$ a commitment to x
    d. For each $x \in \pi(\boldsymbol{Z}_c)$ a commitment to $(x, \phi(x))$
3. The election authority prepares a publicly verifiable tallying function $\boldsymbol{D}$ such that $\forall x \in \boldsymbol{C}$, $\boldsymbol{D} \circ \phi \circ \pi \circ \sigma(x) = x$. The preparation may involve publishing commitments, keys, etc. depending on the particular $\boldsymbol{D}$. One can say that a sigma ballot encrypts a clear text vote x into a coded vote y and $\boldsymbol{D}$ decrypts y back to x. A sample $\boldsymbol{D}$ is described in section 5.

To check that $\forall x \in \boldsymbol{C}$, $\boldsymbol{D} \circ \phi \circ \pi \circ \sigma (x)=x$, a public auditor chooses a statistically significant number of ballots and asks the election authority for the information such that the equation $\boldsymbol{D} \circ \phi \circ \pi \circ \sigma (x)=x$ can be publicly checked. This is the very first step of the protocol and is done before Election Day, before ballots are printed.

The next step is to produce Sigma ballots and receipts, using one of the protocols described in section 3.

After the voter obtains her receipt, the following commitments are opened:

1. If the receipt contains the order of the candidates, the commitment to the entire function σ is opened.
2. If the receipt contains the position x of the marks, the commitment to $(x, \pi(x))$ is opened.
3. For the confirmation code x which is always on the receipt, the commitment $(x, \phi(x))$ is opened.

If a voter complains that she does not see the correct confirmation codes posted on the public bulletin board, she is asked to provide the confirmation codes that she thinks should be on the bulletin board. Then the election authority opens all commitments to x, $\forall x \in \pi(\boldsymbol{Z}_c)$. If the confirmation code provided by the voter is not among the opened ones, then the voter must be wrong. If it is among the revealed ones, and since the confirmation codes are difficult to simply guess, then, if a statistically significant fraction of voters provide confirmation codes that are among the committed ones, this becomes an indication of malfunction.

If the voter does not see her paper receipt correctly posted on the public bulletin board, i.e., the order of the candidates or the position at which the marks appear is not the same, then the voter can bring her paper receipt as irrefutable proof of malfeasance.

Anybody can inspect the bulletin board and check that the commitments are consistent with the posted receipts, i.e., with the order of the candidates or with the association between confirmation codes and the marked positions. Also, anyone can check the commitments to the confirmation codes themselves or the commitments to the association between confirmation codes to coded votes.

5 One way to produce the tally

Inspired by Scantegrity II [CD08], we briefly describe an example of a function ***D*** that allows everyone to check that all the receipts have been correctly tallied. This scheme is not a contribution of this paper, and it is presented only for completeness.

Let N be the number of ballots in an election and let c be the number of candidates on a ballot. Consider three tables (see Figure 4): table R contains coded votes, table T contains clear text votes that are countable by anyone and table D connecting R with T. R is a matrix with N rows and c columns, each row represents the coded votes of a ballot. R is a matrix with c rows and N columns, each row representing a candidate. An element (i, j) is either marked or not marked in R and T. A mark in T corresponds to a vote for a candidate. D is a set with N * c elements. Figure 4 gives an example of the three tables for an election with six ballots and two candidates.

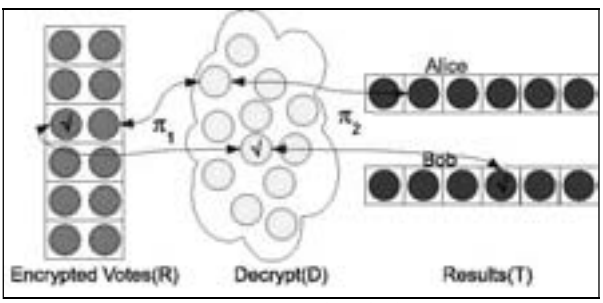

Figure 4: Pointer-based mixnet

The tables are connected by two permutations, π_1 and π_2. π_1 connects R with D: $D_k=R_{\pi 1 (k)}$, where k is some canonical representation of (i, j), e.g., k=(c-1)*i+j. Similarly, π_2 connects D with T: $T_k=D_{\pi 2 (k)}$.

The properties of the permutations may be formalized as follows: let π_1: $\boldsymbol{Z}_{n \times c} \rightarrow \boldsymbol{Z}_{n \times c}$ be bijective and let π_2: $\boldsymbol{Z}_{n \times c} \rightarrow \boldsymbol{Z}_{n \times c}$ be bijective such that no two coded votes belonging to the same ballot initially (in the same row in table R) are mapped to two elements belonging to the same candidate (the same row in table T):

$$\forall\ i,j,\ i \neq j \text{ having } [\ i / c\ ] = [j / c] \rightarrow [\pi_2 (\pi_1 (i)) / b] \neq [\pi_2 (\pi_1 (j)) / b] \qquad \text{Equation 1}$$

where [x] represents the greatest integer less or equal to x. The function ***D*** that provides a universally verifiable tally function is $\boldsymbol{D} = \pi_2 \text{ o } \pi_1$.

Initially, the election authority publishes commitments to each mapping done by π_1 and π_2, along with the commitments needed for the Sigma ballots, including the commitments that tie in the confirmation coded to the coded vote (the indexes in the R table). To check the correctness of this step, an auditor can request some statistically significant number of ballots to have their commitments opened. When a cast ballot is received, the election authority opens the commitment that ties the confirmation code to the coded vote in the R table. After the polls close and the index in the R, D and T are marked, the final audit checks that one of the two properties hold, at random: $D_i=R_{\pi 1(i)}$ or $D_i=T_{\pi 1(i)}$ and that the properties of the two permutations π_1 and π_2 hold, i.e., it checks

that both π_1 and π_2 are injective functions and that Equation 1 holds for each of the revealed pairs of π_1 or π_2. Because the voting system cannot predict which permutation will be checked, a successful audit implies that the coded votes have been correctly transformed into clear text votes with high probability. Privacy is preserved, since no complete link is revealed from the R table to the T table, but only links from either R to D, or from D to T.

6 Conclusions

We have presented a new type of filled-in ballot which has confirmation codes like Scantegrity II and the order of the candidates permuted like Prêt à Voter. The advantages of Sigma ballots combine the ability to easily check that they have been correctly printed with the ability to file a complaint without the need for the voter to present physical evidence. At the same time, Sigma ballots solve some of the issues of Scantegrity II, such as adding marks after the ballots have been cast, or needing to create receipts by hand. Sigma ballots produce a "something you know" receipt to check the correct recording of the cast ballot and a "something you have" receipt to check the correctness of printing.

We described two ways in which Sigma ballots can be produced: using a DRE+VVPAT or using an optical scan Scantegrity II ballots. The DRE+VVPAT approach seems to be the most promising one, since it combines the advantages of having a robust and precise interface with the availability of hand countable paper ballots, and on top of that, the publicity verifiable tallying method.

Bibliography

[AR06] Adida, B, and R. Rivest. 2006. Scratch & vote. Self-contained paper-based cryptographic voting. In *WPES '06. Proceedings of the 5th ACM workshop on privacy in electronic society*, 29– 40. New York, NY, USA: ACM Press.

[AB09] Adida, B. et al. 2009. Electing a university president using open-audit voting. Analysis of real-world use of helios. In *Electronic voting technology workshop/workshop on trustworthy elections*. Usenix.

[CD04] Chaum, D. 2004. Secret-ballot receipts. True voter-verifiable elections. *IEEE Security and Privacy* January/February: 38–47.

[CRS05] Chaum, D., P. Y. A. Ryan, and S. Schneider. 2005. A practical voter-verifiable election scheme. In *ES-ORICS*, *volume 3679 of lecture notes in computer science*, ed. Sabrina De Capitani di Vimercati, Paul F. Syverson, and Dieter Gollmann, 118–139. Springer (http://www.springerlink.com/content/ebrbl9kc81bhx98j/).

[CD08] Chaum, D. et al. 2008. End-to-end verifiability for optical scan election systems using invisible ink confirmation codes. In *EVT'07. Proceedings of the USENIX/accurate electronic voting technology workshop*. USENIX Association.

[EA07] Essex, A. et al. 2007. Punchscan in practice. An e2e election case study. In *IAVoSS workshop on trustworthy elections (WOTE 2007)*. University of Ottawa, Canada.

[KJ07] Kelsey, J. et al. 2007. Some random attacks on paper-based e2e systems. http://kathrin.dagstuhl.de/files/Materials/07/07311/07311.KelseyJohn.Slides.pdf/. (accessed 17 November 2008).

[PH06] Popoveniuc, S., and B. Hosp. 2006. An introduction to PunchScan. In *IAVoSS Workshop On Trustworthy Elections (WOTE 2006)*. Robinson College, Cambridge.