A New Implementation of
a Dual (Paper and Cryptographic) Voting System

Jonathan Ben-Nun[1], Niko Farhi[1], Morgan Llewellyn[2], Ben Riva[1], Alon Rosen[3], Amnon Ta-Shma[1], Douglas Wikström[4]

[1]Tel Aviv University
Israel
jonathan.bennun@gmail.com
{nikofarh | benriva | amnon@tau.ac.il}

[2]IMT Lucca
Italy
morgan.llewellyn@imtlucca.it

[3]IDC Herzliya
Israel
alon.rosen@idc.ac.il

[4]KTH Stockholm
Sweden
dog@csc.kth.se

Abstract: We report on the design and implementation of a new cryptographic voting system, designed to retain the "look and feel" of standard, paper-based voting used in our country Israel while enhancing security with end-to-end verifiability guaranteed by cryptographic voting. Our system is dual ballot and runs two voting processes in parallel: one is electronic while the other is paper-based and similar to the traditional process used in Israel. Consistency between the two processes is enforced by means of a new, specially-tailored paper ballot format. We examined the practicality and usability of our protocol through implementation and field testing in two elections: the first being a student council election with over 2000 voters, the second a political party's election for choosing their leader. We present our findings, some of which were extracted from a survey we conducted during the first election. Overall, voters trusted the system and found it comfortable to use.

1 Introduction

The foundations of modern cryptographic voting systems were laid out in the 1990s, introducing powerful techniques such as homomorphic tallying and mixing networks. Almost all early work assumes that the voter has access to some trusted computational device while voting. In 2004, Chaum [Ch04] and, independently, Neff [Ne04] proposed cryptographically secure voting systems in which the voter has access to no computational device at the time of voting. Since then, most research has focused on such bare-handed, end-to-end verifiable voting systems.

In 2008, Benaloh [Be08] suggested dual voting. In Benaloh's system, the voter fills in a plaintext ballot and a scanning machine reads it to produce a printed plaintext ballot, which is cast into a ballot box, together with a cryptographic encryption, which is uploaded to a public web page, and an electronic receipt, which the voter may take home. The system is end-to-end verifiable using standard cut-and-choose techniques.[1]

There are several advantages to dual voting. Cryptographic voting, in general, is more vulnerable than paper-based voting to global failures and attacks. We can demonstrate this with a simple global failure. Many cryptographic protocols use a k-out-of-n threshold encryption scheme. It may happen that (accidently or deliberately) too many keys are lost, in which case the whole election is compromised. Paper-based systems are, in contrast, more resistant to global failures. Thus, dual-voting systems supply the stronger guarantees of end-to-end verifiability characteristic of electronic cryptographic voting while retaining paper's resiliency against global failures.

Another major advantage of dual voting is psychological. Dual-voting systems often retain the look and feel of paper-based systems, which makes these systems more familiar to and trusted by voters, who are used to paper-based voting. Furthermore, we saw time and again that people trust paper, probably because paper is something you can hold and read on your own. The fact that our system offers a paper backup made it easier for the Merez party to decide to use our system.

In dual-ballot systems, an adversary wishing to commit election fraud would need to break both the paper-based and the cryptographic systems.[2] On the downside, it is enough to break one system to breach privacy.

Finally, it should be noted that in dual-ballot systems it must be decided in advance when to count which system. Indeed, in some states (like California) the law requires to count paper ballots, while in others, only a sample is required. We find the following options reasonable:

- Use the paper-based system as backup only for disaster recovery, e.g., when private keys are lost or when the bulletin board goes down during the election.
- Count both systems (for all polling stations or for a sample of them) and if they substantially differ, conduct an official investigation.

[1] In fact, Benaloh's system may be seen as a triple voting system, where the scanner tallies the scanned votes in addition to the electronic and paper tallying.

[2] In most cryptographic systems the integrity guarantee is unconditional, even against all-powerful adversaries, and so it is often heard that cryptographic systems cannot be undetectably forged. However, it should be noted that the cryptographic guarantee is given only provided certain assumptions hold, e.g., the authenticity of the bulletin board is assumed.

While the theory of cryptographic voting is extensive, and quite well understood, not many cryptographic voting systems have been tested in practice. Helios [Ad08, Ad09], which is a web-based voting system, has been used in several elections totaling more than 25,000 voters. Prêt-a-Voter was tested at the University of Surrey Student Union elections in 2007 [Bi09]. We mention that a recent version of Prêt-a-Voter [LR08] also supports dual voting. Punchscan was used at the University of Ottawa in 2007 [EC07]. Scantegrity II was used at the Takoma Park, Maryland municipal elections in 2009, serving over 1,700 voters [Ca10]. Scantegrity II also supports dual-voting. With the exception of Helios, all the other systems use pre-prepared ballots.

A common criticism of cryptographic voting systems concerns the usability issue. It is often said that cryptographic voting systems are too complicated for the common voter. In this work we set to design and implement a dual ballot system that retains the look and feel of paper-based elections in our country, trying to prove that such systems do not suffer from usability issues. We implemented a bare-handed, end-to-end verifiable, dual (paper and electronic) system with ballots printed on-demand (as opposed to pre-prepared ballots). Our design is closest to Benaloh's system [Be08] and has been adapted to Israel‘s paper-based system.

Our system was successfully tested twice. It was first used in an the Interdisciplinary Center's student council election held in May 2011 and then again in Merez's party leader election held in February 2012. We summarize our experience as follows:

> **IDC's Election:** The Interdisciplinary Center (IDC) is a non-profit college with around 6,000 registered students; 2,097 students voted in the election. We counted both the electronic and paper-based systems and discovered minor differences between the two tallies, most likely attributed to mistakes in the hand-counted paper tally. 481 voters checked their receipts online.3 We had only two complaints about missing receipts, which we attribute to scanning errors.
>
> We also asked voters to fill in a questionnaire about the voting experience, asking about their understanding of the voting process and their satisfaction from it. The results show that the majority of survey respondents thought the voting process was clear and simple and possessed a high degree of confidence in their vote being counted. We report on the survey results in Section 4.2. It should be kept in mind, though, that most of the voters were young and often technologically savvy students.
>
> **Merez's election:** Merez is a small political party in Israel and has about 3% of the seats in parliament. The party council, with about 950 representatives, elects the party's leader. There was a high turnout at the elections with approximately 830 voters (88% of registered voters). Many of the voters were over 50 years old. Due to limited resources, we did not run a questionnaire at the election, but we received enthusiastic feedback from many voters and officials, with the party's secretary-general saying over 60 representatives called him to say how good it was to use our voting system.

[3] We gave the voters an incentive to verify their vote online.

We believe the fact that our system retains the look and feel of current paper-based voting systems helped people accept it and made them think of the dangers and promises of electronic voting. We hope that our experiment will help facilitate the transition from paper-based voting to more sophisticated systems supporting end-to-end verifiability.

2 Desired Properties

The most crucial property required of electronic voting systems is *integrity*, meaning that it is impossible to falsify election results. Another crucial property is *privacy,* meaning that no one can link a voter to his or her vote, and even further, a voter cannot prove to someone, what his or her vote was. Such a system is known as *coercion-free* or *incoercible* and helps reduce the chances of vote buying.

A system is *voter-verifiable* if any voter can verify that his/her vote was correctly recorded and is included in the tally. A system is universally-verifiable if *anyone* can verify that all recorded votes are properly tallied. A system having both properties is *end-to-end* verifiable.

One can roughly divide the new voting systems into two classes: voting systems where ballots are pre-prepared before election day [Ch04,RP05,FCS06,AR06,Chb08,Cha08] and voting systems where ballots are printed on-demand in the voting booth behind curtains [Ne04,MN06,Be06,Be08, SDW08]. On-demand systems often have easy, user-friendly interface for the voter (often using touch screens). Regarding privacy, with print-on-demand voting the voter often has to enter his or her choices into the voting machine - thus losing privacy with respect to the voting machine, whereas pre-prepared ballots avoid this problem. On the other hand, when ballots are printed in advance it is crucial to guarantee that these ballots are kept secret (for instance, that the ballots are not photocopied by an adversary) leading to the *chain of custody* problem. Another privacy issue in print-on-demand systems is the possibility of subliminal channels where the booth leaks information about the votes to outsiders. For example, the booth can pick randomness that would create a ciphertext whose last bits would also encode the candidate. [FB09,AN09,GGR09] These resources show how to mitigate these types of attacks.

3 The Protocol

Our protocol is based on the protocols from Benaloh [Be06, Be08]. Since the voting booth in our protocol prints ballots on-demand, we protect against subliminal channels by splitting some of the booth's functionality to external smart cards (see Appendix A for further details.)

Our system uses standard cryptographic primitives used in other cryptographic voting protocols. More specifically, we use the following protocols: ElGamal encryption scheme [Ga85]; Pedersen's (t, n) -threshold ElGamal encryption scheme[Pe91, Pe92], in which any t parties can decrypt a message but no $t - 1$ parties can; Cramer et al.'s three round, honest-verifier zero-knowledge proof system [CDS94], proving an ElGamal ciphertext c is an encryption of a message m from a given set of possibilities$m_1, \dots, m_l$; the Fiat-Shamir heuristic to transform public-coin, zero-knowledge proofs to non-interactive ones; and we use a *universally verifiable* mix-net producing non-interactive, zero-knowledge correctness proofs. We chose to use a mix-net rather than homomorphic tallying because mix-nets support a wider range of voting schemes.

3.1 Trust Model

Assumptions assuring integrity: We assume the polling station workers are semi-honest, i.e., they will not allow someone to upload encrypted votes or to cast plaintext votes that were not legitimately cast by voters.

Assumptions assuring incoercibility (and privacy): We assume the voting booth will remain integrous, not collaborating with any coercer or with any of the smart cards it uses. We further assume that the smart cards are manufactured by different companies and are not able to collaborate amongst themselves. We also assume that the smart cards can be initialized only once and their internal memory cannot be read or modified externally. Last, we assume there is no dishonest subset of the mix-net parties large enough to be able to decrypt messages.

3.2 High-level Description

The voter first enters the polling station and identifies herself to the polling station committee. Once cleared, the voter proceeds to the voting booth and makes her selection on a touch screen. The voting machine then prints a *dual-ballot*. At this point in the process the voter can either audit the machine, or, use the ballot for casting (i.e., we employ Benaloh's [Be06] *cast-or-audit* method).

Our dual-ballot is a paper note, divided into two detachable parts: the electronic ballot and the physical (*plaintext*) ballot (see Figure 1). The electronic ballot contains the encrypted vote along with a digital signature certifying the electronic ballot. The physical ballot shows the actual vote printed on it. It can be folded in half and then sealed using a standard adhesive, thereby hiding the plaintext inside.

If the voter intends to cast the ballot, the voting machine prints ”For Casting” on the ballot (see Figure 1). The voter then folds and seals the physical ballot (see Figure 3) and exits the voting booth. The electronic ballot is scanned by the polling station committee and the information is uploaded to the public electronic bulletin board. The committee stamps both parts of the ballot and detaches them in front of the voter. The physical ballot is cast into the ballot box and the electronic ballot is taken home by the voter as a receipt (see Figure 4).

If the voter intends to audit the ballot, the voting machine prints additional audit information on the ballot (see Figure 2). Audit ballots allow one to check the consistency of the voting machine, and inconsistent audit ballots serve as a proof that a given voting machine does not function correctly. Audit ballots cannot be used for voting; to cast an actual vote, the voter must re-enter the voting booth.

Tallying: Once the polling stations close, the electronic tallying process takes place publicly on the bulletin board. The tallying is performed using cryptographic tools, such as mix-nets and zero-knowledge proofs. Manual tallying of the paper ballots may be performed at the polling station once it is closed. The decision whether to count/sample the paper ballots or not is left to the discretion of the officials organizing the elections. A policy defining when paper ballots will be tallied should be published prior to the elections.

A detailed description of the protocol appears in Appendix A.

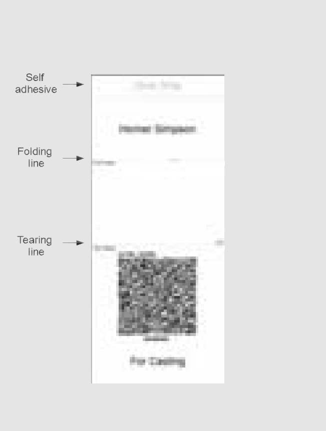


Fig. 1: Dual-ballot before folding. Since it is for casting, there is no barcode in the lower part of the ballot

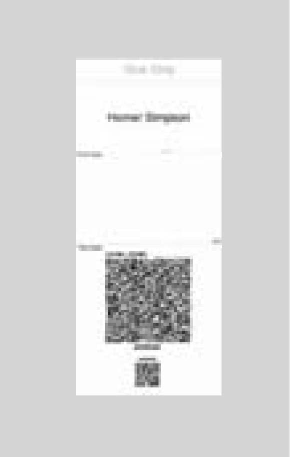
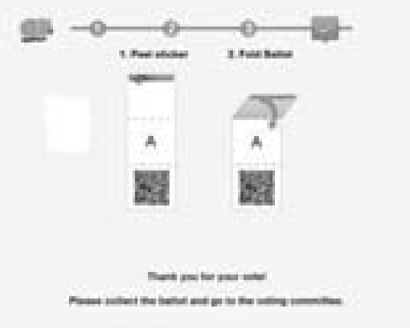


Fig. 2: Audit ballot. The audit information is printed in the barcode in the lowest part of the ballot



Thank you for your vote!

Please collect the ballot and go to the voting committee.

Fig. 3: Folding a ballot

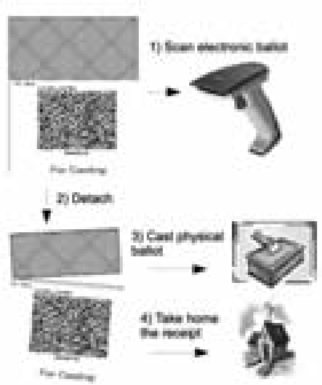


Fig. 4: Casting

3.3 Implementation

According to the protocol, the machine has to commit to the encryption before knowing whether or not the ballot has been audited. To implement this, the printer output slot is protected by a partially transparent plastic cover that lets the voter see the partially-printed ballot without seeing what is printed on it. This also prevents using the ciphertext as a source of randomness for coercion.

An important implementation detail concerns the choice whether to audit the ballot or not. At first, we asked each voter if he or she would like to audit the ballot. We discovered that many voters were confused by that question. As a result we decided to hide the ballot-auditing feature from common voters. Instead, in our implementation the audit option can be invoked by pressing a hidden button while the ballot is printed (see Figure 5). The rationale behind this is the fact that it is sufficient to audit approximately 2-3% of the ballots, and this can be done by designated auditors. That way, we simplify the voting experience for the common voter without sacrificing the security of the system.

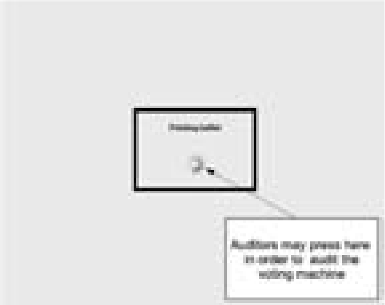

Printing ballot
Auditors may press here in order to audit the voting machine

Fig. 5: Screenshots of the printing window with the hidden audit button

We advertised this procedure on the web page so that more sophisticated voters could also participate in the auditing process.

Our website displayed encrypted votes and some additional information about the election like explanations about the voting, auditing and tallying processes, all public keys, the mix-net proofs of correctness, the uploaded votes file and signature, and election results. Voters can also use the website to find their votes inside the vote file.
For the mix-net, we use Verificatum [Ve11], which is a free and open source implementation of an ElGamal based mix-net. Most of the code is written in Java, but arithmetic code is also available for improved speed. For more details about the protocol itself we refer the reader to Wikström [Wi11]. We are currently in the final stages of writing an independent verifier for the proofs generated by Verificatum.
We also wrote an open source Android application allowing voters to audit their votes more easily. The application allows voters to take a picture of the ciphertext part of the

ballot and the audit part of the ballot (if it exists) using the smart phone's camera. The application verifies that the signatures on the ballot are correct. If the ballot is an audit ballot, the app would ensure that the ciphertext was generated using the randomness specified in the audit part. If it is "For Casting", the app verifies the ciphertext information is posted correctly on the website.

3.4 Unimplemented Functionality

The protocol uses smart cards to mitigate a subliminal channel attack. However, we had neither the time nor the resources to build and test a system with smart cards. Instead, we simulated the smart card functionality. We hope to add actual smart cards in later versions of the system.

In our original design, the polling stations would only upload the new votes to the website. To make sure the website would not remove chunks of votes from the list, the posted votes were to be protected by Merkle Hash Tree [Me87]. However, due to time restrictions, and the fact that we supported only one polling station, we decided to upload all votes to the website.

4 Usability and Related Issues

The IDC elections took place for three consecutive days, from May 17th to 19th. There were several simultaneous races: In addition to races for the student council president, vice president, and elections for representatives of 27 special tracks, 78 candidates competed for 56 available seats on the student council. About 2,097 voted in the election out of about 6000 registered voters (approximately 33%). Most of the voters were students in their early 20s. On average, it took a voter 1-2 minutes to vote, comprised of about 30 seconds of interacting with the polling station worker before voting, one minute using the voting machine, and another 30 seconds of interaction with polling station workers after voting. Once polling stations close, the mix-net was run on a single machine. The whole process took slightly less than 20 minutes and the election results were announced 45 minutes after the closing of the polling station on the last day of the elections. No contentions were filed.

In order to educate potential voters about the system, in both elections the voting process was explained in advance on a website. Furthermore, one of the developers stood at the entrance of the polling station and explained the polling process, defining exactly what they had to do once inside the polling station. We also made large posters clarifying the process and posted them outside the polling station.

4.1 Lessons learned

Many voters (in both elections) did not fold their ballots at all or folded them incorrectly, without explicitly being told the proper technique. This was partly due to an insufficient ballot design, which made it possible to fold the ballot in two different ways. When one of the system developers demonstrated the proper folding method for voters before entering the voting booth, the error rate virtually dropped to zero.

We also explained the dangers of DRE voting, i.e., where a computer simply stores the votes internally, to interested voters. Voters quickly understood the issue and many of them told us they feel better knowing they can actually *see* their vote in plaintext. Many voters (especially the younger ones) enjoyed voting with the new technology, and as a result, were more open-minded to learn about the system. Since the usability of electronic voting also depends on the voters' enthusiasm and understanding, we believe these two reactions are positive if one considers large-scale deployment of the system.

4.2 The Questionnaire

In the first election, we asked voters to fill in an on-line questionnaire. (We did not have a questionnaire in the second election because of limited resources.) The online questionnaire was composed of 10 questions: two administrative, six about the voter's understanding of the voting process and his or her satisfaction, and two about the perceived privacy and integrity of the system. In addition, we also conducted random exit surveys. In total, 481 voters participated in the survey, 403 of them answering the on-line survey and 78 the exit survey. The survey response rate was just under 23.4%. About 37% of those who answered were female and 62% were male, with 4 voters declining to state their gender. In general, survey participants were well -distributed among seven fields of study. The majority (about 73 %) of survey participants verified their ballots.

Information on a voter's satisfaction with the voting process was captured via the survey question: "Thinking about your overall experience at the polls today, how satisfied are you with your voting experience?" Responses to this question are posted in Table 1. Over 85% of respondents reported being satisfied.

	Very satisfied	*Satisfied*	*Somewhat dissatisfied*	*Very dissatisfied*	*Don't know*
On-line survey	45.2%	49.6%	2.2%	1.0%	2.0%
Exit survey	62.9%	34.6%	0.0%	2.5%	0.0%

Table 1: Voter Satisfaction

[4] The high participation rate is due to a lottery of two campus parking lots (a desirable bonus) among those who participated.

Voter opinion over the simplicity of the voting process is located in Table 2. The majority of survey respondents believed the voting process was clear and simple. Across all survey participants, 60% of respondents strongly agreed that the voting process was clear and simple; with just over 1% of respondents strongly disagreeing. About three-quarters of survey respondents reported understanding why the ballot was separated.

	Strongly Agree	Agree Somewhat	Neither Agree nor Disagree	Disagree Somewhat	Strongly Disagree
Did not verify	68.5%	20.8%	8.4%	1.5%	0.8%
Verified ballot	56.1%	29.6%	8.9%	4.0%	1.4%

Table 2: The Voting Process Was Clear and Simple

Given that many voters viewed the process as rather straightforward, it is not surprising that voters possessed a high degree of confidence in their votes being counted. Relative to previous studies of voter confidence in U.S. elections, voter confidence was extremely high with 95.1% of voters expressing a high level of confidence [AHL08].

Despite high levels of voter satisfaction, the survey did highlight two areas for future improvement. Approximately 15% of respondents reported encountering a problem or asking for assistance during the voting process. Through a follow-up question, respondents identified folding the ballot as the most commonly encountered difficulty (36% of identified problems). At 14% of the reported problems, the second most cited difficulty was the online verification process. Participants were asked to state the one task which they would like to improve. Out of a list of 9 fixed choices, and one write-in option, 33% of survey respondents selected verifying their ballot on the Internet. These issues are currently being addressed by the design team, and we anticipate future versions of the system to encounter significantly fewer user issues.

In conclusion, voters exhibited high levels of satisfaction and confidence with the system. A clear majority of voters found the voting process simple and uncomplicated which is particularly important when implementing a new e-voting system. Given the unfamiliarity of the concept of vote verification, it is reassuring that most voters were confident and comfortable with the technology. Finally, survey and observational analysis revealed a significant portion of voters encountered problems with the ballot design, especially the folding, which clearly needs to be improved.

Appendix A: Detailed Description of the Protocol

A.1. Setting up the election

The mix-net parties jointly generate a master public key using the distributed key generation of the threshold ElGamal cryptosystem. Let G, q, g be the public parameters and let h be the generated threshold ElGamal public key.
The bulletin board and all polling station committee computers generate signature key pairs. We assume that the bulletin board public key is known to all participants.
Last, the election officials initialize two smart cards SC_1, SC_2 for each voting booth. The initialization of smart card SC_i consists of the generation of a unique identification number id_i and the generation of a signature key pair (possibly the same for all booths) and setting the internal counter $rnd_{cnt_i} = 1$. Also, the election public-key is stored on the card along with the list of valid candidates. All the smart cards' public keys are stored on the bulletin board.

A.2. Election day

Voting: The voter enters the polling station and identifies herself. Once cleared by the poll workers, the voter enters the voting booth. The voter votes v using a touch screen.
Denote the smart cards by SC_1, SC_2. The booth itself is a deterministic machine that cannot generate randomness. The booth requests randomness from the smart cards (to avoid the subliminal channel problem). Each smart card $i \in \{1, 2\}$ increases its internal counter by one rnd_{cnt_i} and returns a message consisting of [rnd_{cnt_i}, r_i, g^{r_i},] $Signature_{SC_i}(id_i|rnd_{cnt_i}\|g^{r_i})$ where g is the generator from the election public key and r_i is uniformly random.
The booth encrypts the vote by $c = Enc_h(v, r_1 + r_2)$. It also generates a non-interactive zero-knowledge proof π_c that c is an encryption of a valid vote (using 1-out-of-l zero-knowledge proof). The booth sends [rnd_{cnt_1}, rnd_{cnt_2}, c , π_c] to SC_1 (SC_1 is chosen before the election day, e.g. the smart card with lower ID number). The smart card verifies that the proof π_c is valid for c , and that its internal counter sig_{cnt_1} is smaller than rnd_{cnt_1}. If everything is sufficiently verified, the smart card sets its internal counter to $sig_{cnt_1} = rnd_{cnt_1}$ and returns
[$Signature_{SC_1}(id_1|rnd_{cnt_1}|rnd_{cnt_2}\|c)$]. Otherwise it will display an error message. (We need the 1-out-of-l zero-knowledge proof to prevent the voting machine from leaking previous votes in the encrypted message, thereby violating voter privacy.)

The booth prints the first and second parts of the ballot (see Figure 1). More specifically, in the physical ballot part it prints v and in the electronic ballot it prints:

$$\begin{array}{l} id_1, id_2 \\ rnd_{cnt_1}, rnd_{cnt_2} \\ g^{r_1}, Signature_{SC_1}(id_1|rnd_{cnt_1}|g^{r_1}) \\ g^{r_2}, Signature_{SC_2}(id_2|rnd_{cnt_2}|g^{r_2}) \\ c = Enc_h(v, r_1 + r_2), Signature_{SC_1}(id_1|rnd_{cnt_1}|rnd_{cnt_2}\|c) \end{array}$$

The counters are used to prevent chain voting and a re-use of randomness.

We shielded the printer output such that the voter could see that a ballot had been printed but it cannot be extracted before the voter chooses whether or not to audit the ballot.
We note that by using the information printed in the electronic ballot, anyone can verify that the encryption was computed with randomness that was produced by the smart cards. That can be checked simply by verifying all signatures and computing $g^{r_1}g^{r_2}$ and comparing it with the first element $Enc_k(v, r_1 + r_2)$.
Now, the voter can (but does not have to) audit the voting machine to verify that the ballot was produced properly. If the voter wishes check it, she presses "Audit the Machine" on the touch screen. Otherwise, the voter presses "Cast".

Auditing the machine: The booth prints "Audit information: r_1, r_2" at the bottom of the ballot. After the voter exits the booth, the poll-workers verify that all signatures are valid and that the randomness counters are equal and increased by one over the counters of previously casted ballots. By using the randomness printed as audit information the poll workers can verify that the ciphertext printed on the electronic part of the ballot really encrypts the plaintext printed on the other part. If so, they stamp the ballot and the voter can return to the booth to continue her voting. The voter may also verify those properties at home.

Casting: If the voter presses "Cast" the booth prints "For Casting" at the bottom of the ballot. The voter folds the first part of the ballot. Next, the voter leaves the voting booth and presents her folded ballot to the poll workers. The poll workers verify that her ballot has not yet been detached. They scan the electronic ballot, verify its signatures and randomness counters, stamp both parts of the ballot, and detach the physical ballot from the electronic one. All of this is done in front of the voter. The physical ballot is publicly put into the ballot box and the stamped electronic part is uploaded to the bulletin board and returned to the voter as receipt.
The voter then leaves the polling station with the electronic ballot.

A.3. Tallying

After the election is over, the mix-net at every polling station takes all the encrypted votes $c_1, c_2, \dots, c_N$ and passes them through a (re-encryption) mix-net. The mix-net is made of n mixes, each one belongs to a different party. After the last mix outputs a list of ciphertexts, $c'_1, c'_2, \dots, c'_N$, a verifiable threshold decryption is executed by t parties. The result of this decryption is the tally result for this specific polling station.
The physical ballots may also be counted according to the policy of the officials organizing the elections.

A.4. Auditing

Auditability of casting: The voter can check whether her casted electronic vote is posted correctly on the bulletin board. Also, she can choose to audit the voting machine and receive an audit ballot that she can check at her home, using her own computer. Because the machine has to commit to the ballot by printing it before it knows whether it is audited or not, the machine has to decide whether to "cheat" or not before knowing whether the ballot will be audited.

Auditability of tallying: Universal verifiability of the tallying is achieved using the standard primitives of verifiable shuffles and verifiable threshold decryption. Anyone can download a program to check those proofs using his or her own computer. Anyone with sufficient knowledge can write a program to verify those proofs themselves.

Cross checking: At the end of the election we get two parallel systems that can validate each other. The decision whether or not to count the paper-based system should be determined before the election takes place.

Bibliography

[Ad08] Ben Adida. Helios: web-based open-audit voting. In USENIX Security Symposium, 2008.

[Ad09] Ben Adida, Olivier Pereira, Olivier DeMarneffe, and Jean jacques Quisquater. Electing a university president using open-audit voting: Analysis of real-world use of Helios. In EVT/WOTE, 2009.

[AHL08] R.Michael Alvarez, Thad E. Hall, andMorgan H. Llewellyn. Are Americans Confident Their Ballots Are Counted? The Journal of Politics, 70(03):754–766, 2008.

[AN09] Ben Adida and C. Andrew Neff. Efficient receipt-free ballot casting resistant to covert channels. In EVT, 2009.

[AR06] Ben Adida and Ronald L. Rivest. Scratch and Vote: Self-Contained Paper-Based Cryptographic Voting. In WPES, 2006.

[Be06] Josh Benaloh. Simple verifiable elections. In EVT, 2006.

[Be08] Josh Benaloh. Administrative and Public Verifiability: Can We Have Both? In EVT, 2008.

[Bi09] David Bismark, James Heather, RogerM. A. Peel, Steve Schneider, Zhe Xia, and Peter Y. A. Ryan. Experiences Gained from the first Pret A Voter Implementation. REVOTE, 2009.

[Ca10] Richard Carback, David Chaum, Jeremy Clark, John Conway, Aleksander Essex, Paul S. Herrnson, Travis Mayberry, Stefan Popoveniuc, Ronald L. Rivest, Emily Shen, Alan T. Sherman, and Poorvi L. Vora. Scantegrity II municipal election at Takoma Park: the first E2E binding governmental election with ballot privacy. In USENIX conference on Security, 2010.

[CDS94] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols. In CRYPTO, 1994.

[Ch04] David Chaum. Secret-Ballot Receipts: True Voter-Verifiable Elections. IEEE Security & Privacy, 2(1):38–47, 2004.

[Cha08] David Chaum, Richard Carback, Jeremy Clark, Aleksander Essex, Stefan Popoveniuc, Ronald L. Rivest, Peter Y. A. Ryan, Emily Shen, and Alan T. Sherman. Scantegrity II: end-to-end verifiability for optical scan election systems using invisible ink confirmation codes. In EVT, 2008.

[Chb08] David Chaum, Aleks Essex, Richard Carback, Jeremy Clark, Stefan Popoveniuc, Alan Sherman, and Poorvi Vora. Scantegrity: End-to-End Voter-Verifiable Optical-Scan Voting. IEEE Security and Privacy, 6:40–46, 2008.

[EC07] Aleks Essex and Jeremy Clark. Punchscan in practice: an E2E election case study. In WOTE, 2007.

[FB09] Ariel J. Feldman and Josh Benaloh. On subliminal channels in encrypt-on-cast voting systems. In EVT, 2009.

[FCS06] Kevin Fisher, Richard Carback, and Alan T. Sherman. Punchscan: Introduction and system definition of a high-integrity election system. In WOTE, 2006.

[Ga85] Taher El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In CRYPTO, 1985.

[GGR09] Ryan W. Gardner, Sujata Garera, and Aviel D. Rubin. Coercion resistant end-to-end voting. In FC, 2009.

[LR08] David Lundin and Peter Y. A. Ryan. Human Readable Paper Verification of Pret a Voter. In ESORICS, 2008.

[Me87] Ralph Merkle. A Digital Signature Based on a Conventional Encryption Function. In CRYPTO. 1987.

[MN06] Tal Moran and Moni Naor. Receipt-Free Universally-Verifiable Voting With Everlasting Privacy. In CRYPTO, 2006.

[Ne04] Andrew Neff. Practical High Certainty Intent Verification for Encrypted Votes. 2004.

[Pe91] Torben P. Pedersen. A Threshold Cryptosystem without a Trusted Party (Extended Abstract). In EUROCRYPT, 1991.

[Pe92] Torben P. Pedersen. Distributed Provers and Verifiable Secret Sharing Based on the Discrete Logarithm Problem. PhD thesis, 1992.

[RP05] Peter Ryan and Thea Peacock. Pret a Voter: a System Perspective. Technical Report 929, University of Newcastle upon Tyne, School of Computing Science, Apr 2005.

[SDW08] Daniel Sandler, Kyle Derr, and Dan S. Wallach. VoteBox: a tamper-evident, verifiable electronic voting system. In USENIX Security Symposium, 2008.

[Ve11] Verificatum project, 2011. http://www.verificatum.org.

[Wi11] Douglas Wikström. Verificatum, 2011. In preparation.