DESIGN AND ANALYSIS OF NOVEL VERIFIABLE VOTING SCHEMES

Yernat Yestekov

Thesis Prepared for the Degree of

MASTER OF SCIENCE

UNIVERSITY OF NORTH TEXAS

December 2013

APPROVED:

Mahadevan Gomathisankaran, Major Professor
Bill Buckles, Committee Member
Armin Mikler, Committee Member
Saraju Mohanty, Committee Member
Barret Bryant, Chair of the Department of Computer Science and Engineering
Costas Tsatsoulis, Dean of the College of Engineering
Mark Wardell, Dean of the Toulouse Graduate School

Yestekov, Yernat. Design and analysis of novel verifiable voting schemes. Master of Science (Computer Science), December 2013, 52 pp., 23 tables, 17 figures, 41 numbered references.

Free and fair elections are the basis for democracy, but conducting elections is not an easy task. Different groups of people are trying to influence the outcome of the election in their favor using the range of methods, from campaigning for a particular candidate to well-financed lobbying. Often the stakes are too high, and the methods are illegal.

Two main properties of any voting scheme are the privacy of a voter's choice and the integrity of the tally. Unfortunately, they are mutually exclusive. Integrity requires making elections transparent and auditable, but at the same time, we must preserve a voter's privacy. It is always a trade-off between these two requirements. Current voting schemes favor privacy over auditability, and thus, they are vulnerable to voting fraud.

I propose two novel voting systems that can achieve both privacy and verifiability. The first protocol is based on cryptographical primitives to ensure the integrity of the final tally and privacy of the voter. The second protocol is a simple paper-based voting scheme that achieves almost the same level of security without usage of cryptography.

Copyright 2013

by

Yernat Yestekov

TABLE OF CONTENTS

Page

CHAPTER 1

INTRODUCTION

> *The will of the people shall be the basis of the authority of government; this will shall be expressed in periodic and genuine elections which shall be by universal and equal suffrage and shall be held by secret vote or by equivalent free voting procedures.*
>
> Article 21, Universal Declaration of Human Rights, 1948

One of the basic rights of people in democracy is the right to choose their political leaders in regular, free and fair elections. Since the time when ancient Greeks implemented a democracy, conducting elections was a challenging task. Vote selling and buying, bribery and corruption are a constant part of any election. The recent history of elections in US gives a broad overview of fraud techniques from simple vote selling and buying to intricate mechanisms and ingenious methods of fraud. To prevent a voting fraud heavy penalties were established and new voting technologies such as numerous ballot and ballot box designs, lever machines, punch cards, optical scan and digital record machines were implemented. While they provided additional security, it was proven that they failed to provide necessary level of security and became a new point of failure [1].

In this thesis two voting protocols are presented. The first protocol is electronic only and uses cryptographic primitives to proof the result of elections mathematically. The second protocol represents a more practical approach that can be expanded both to paper and electronic ballots. We begin with the overview of tallying methods, then continue with requirements to voting protocols and why such requirements make

designing a voting protocol a tough problem. Then, a short overview of the voting protocols is given. Finally electronic and paper voting protocols are presented.

1.1 A Historical Overview of Voting

The collaborative team of researchers from MIT and Caltech known as the Voting Technology Project proposed a new measurement of the performance of a voting equipment termed as “residual vote rate.” Residual vote rate is a number of votes that were not included in the final tally for the particular race due to problems in voting equipment. It includes ballots that don’t have any marks or mark wasn’t counted by tallying machine (undervote), ballots that have more marks than it is allowed (overvote) and simply lost ballots that weren’t counted for any reason [2].

1.1.1 Public Voting

From Ancient Greece till the early 1800s, public voting was the main voting scheme. It was also known as a *viva-voce* literally translated from Latin as “with living voice” or “by word of mouth.” Voters simply pronounced their preferences in public, while election clerks made written records to the pollbook. In the beginning of the 19th century party tickets were introduced. It was any piece of paper on which voter wrote down his preferences. The name “party ticket” appeared due to the fact that ballots were produced and distributed by political parties (Fig. 1.1).

1.1.2 Modern Paper-Based Voting

---

[1] Figures from 1.1 to 1.5 by Douglas W. Jones. "A Brief Illustrated History of Voting" Photograph. 2001. http://homepage.cs.uiowa.edu/~jones/voting/pictures/, http://homepage.cs.uiowa.edu/~jones/voting/optical/ (last accessed July 21, 2013), unless stated otherwise.

In the absence of a voting privacy, the vote selling was spreading. People were demanding reforms to prevent the fraud. In 1858, Australia introduced a novel voting scheme, where paper ballots were printed and stored in advance by the state (Fig. 1.2). Each eligible voter received a ballot to vote in a private voting-booth. After that ballots were casted into ballot boxes. The current paper-based voting hasn't changed a lot since 1892 when the Australian ballot was used in the United States for the first time. The Australian ballot had an important property of a ballot secrecy that heavily changed further evolution of voting systems.

It didn't take a lot of time to reveal weaknesses in the Australian ballot scheme. Ballots were stuffed, election officials were corrupted to count the votes in favor of a desired candidate. New ballot boxes design or private booths couldn't stop the fraud. A good example that the system was totally compromised was a famous politician William "Boss" Tweed. Once he said "As long as I count the votes, what are you going to do about it?" Later he was convicted for stealing NY state budget funds in the amount estimated between $1 to $8 billion in 2010 dollars.

1.1.3 Lever Machines

The Industrial Revolution that happened in the 19th century changed societies in Western Europe and the United States. Many aspects of daily life were modernized by mechanisms that replaced human labour. The Revolution also influenced the voting system. First lever voting machines were used in 1892 in Lockport, NY. Inside of private booth, voter pulls down a lever, after that a mechanism inside triggers the counter to

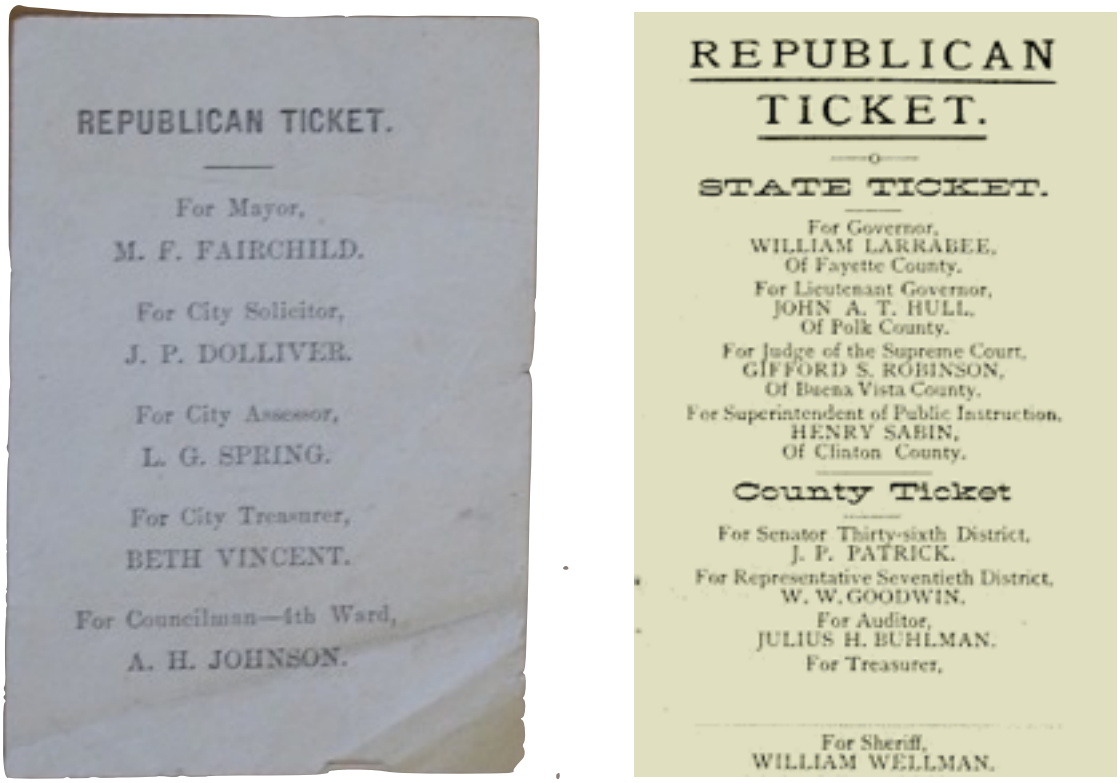

Figure 1.1: Early paper ballots. On the left side, Republican party ticket from an 1880 municipal election in Iowa. On the right side, a ticket from an 1888 general election in Iowa. [1]

add one vote and then, all levers return to the default position (Fig. 1.3). By the end of 1920s large U.S. cities adopted the lever machines.

Lever machines were widely used until 1980. As any complex mechanical device they had a lot of problems with broken counters and levers. Because voter didn't have any access to a machine nobody could verify if his vote was counted as intended.

1.1.4 Punch Card Machines

The further evolution of computing mechanisms evolved to punch-cards. As early computers punch card machines counted holes in different positions. The voter had to

[2] Public domain image via Smithsonian Institution. http://americanhistory.si.edu/vote/votingmachine.html

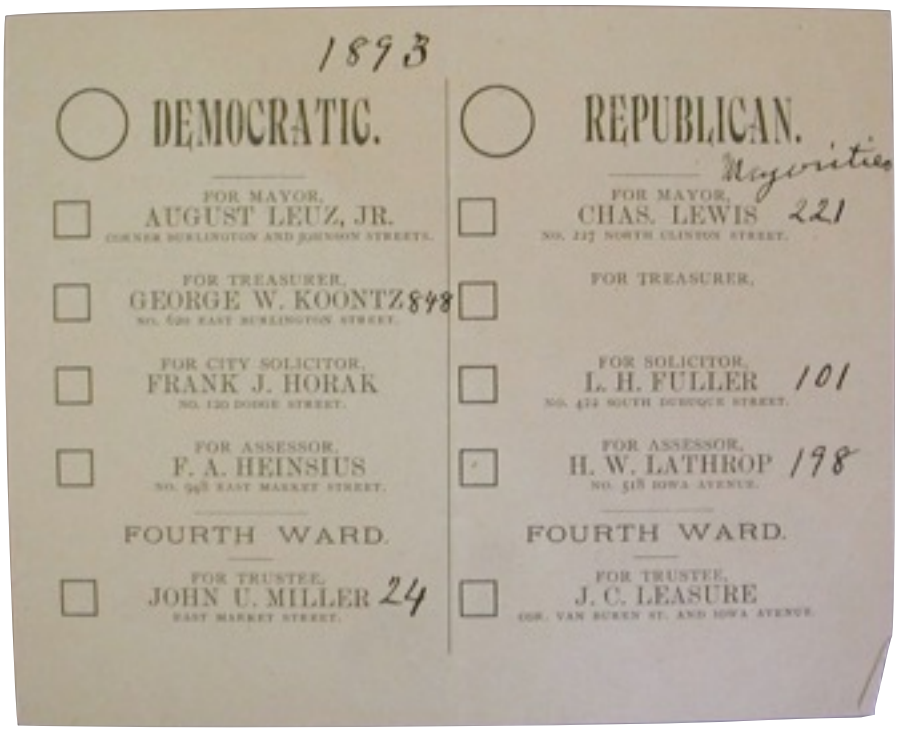

Figure 1.2: The Australian ballot. The ballot from an 1893 Iowa City municipal election.

punch a card (ballot) in a proper place to make a hole (Fig. 1.4). This hole was counted as a vote. While it sounds quite simple, the punch cards had one major problem called a chad, a piece of paper that wasn't removed completely from a card. Such chads prevented machines to count votes appropriately.

The presidential race between Democrat Al Gore, Jr. and Republican George W. Bush in 2000 was one of the most controversial and close-call election in the American history. Since the outcome wasn't clear the U.S. Supreme Court defined a winner by recounting of ballots in Florida. The final tally showed that Bush lost the popular vote by 500,00 votes [3], but won the Electoral College by a margin of 483 votes [4]. The study [5] showed that punch card machines had higher average residual vote rates than any other voting system.

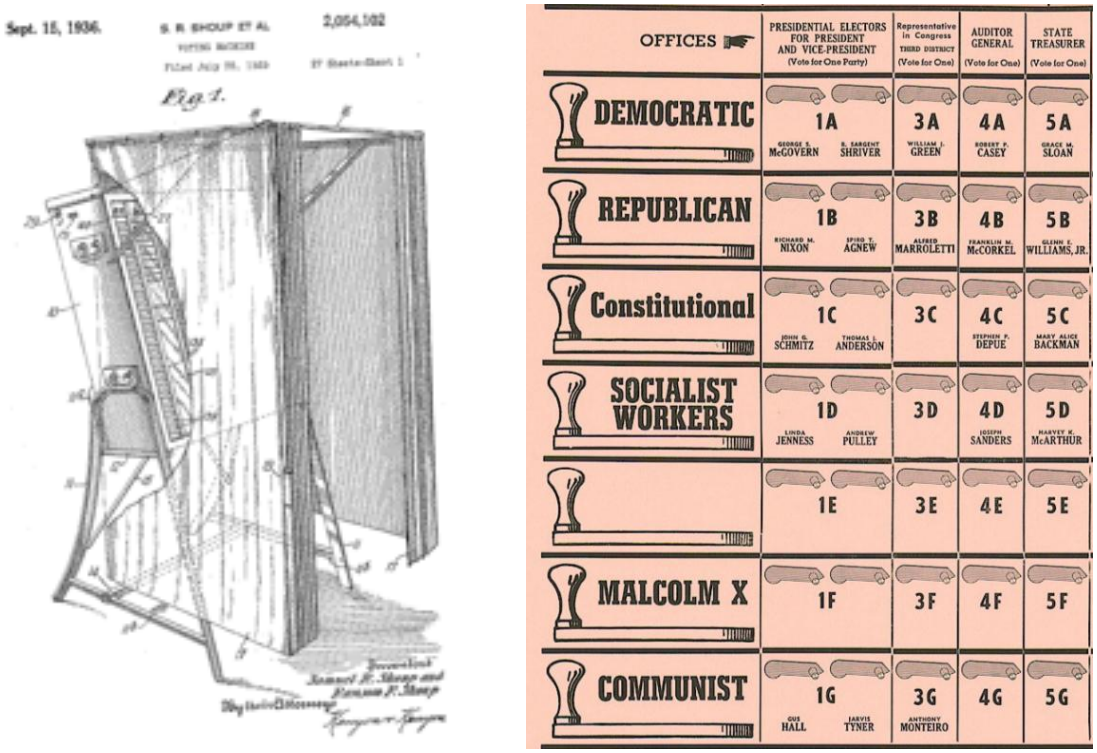

Figure 1.3: The lever machine. On the left side,Patent illustration for a lever voting machine. On the right side, lever voting ballot layout. Philadelphia, 1972. [2]

1.1.5 Optical Scan Machines

In optical scan voting scheme, voter should mark his choice by filling a bubble on a paper ballot using a pencil or pen. Then the ballot is scanned and tallied by an optical scan machine (Fig. 1.5).

There are two types of op-scan machines. One is a precinct-based, where ballots are scanned and tallied immediately. Another one is central-count, where ballots first collected using ballot boxes, then transferred to a central location and tallied there.

The main problem with op-scan machines is that scanner couldn't recognize the marks properly. As it was discovered later, voters can understand the phrase "fill the bubble" completely different. One voter would put an X mark, the other would fill it completely.

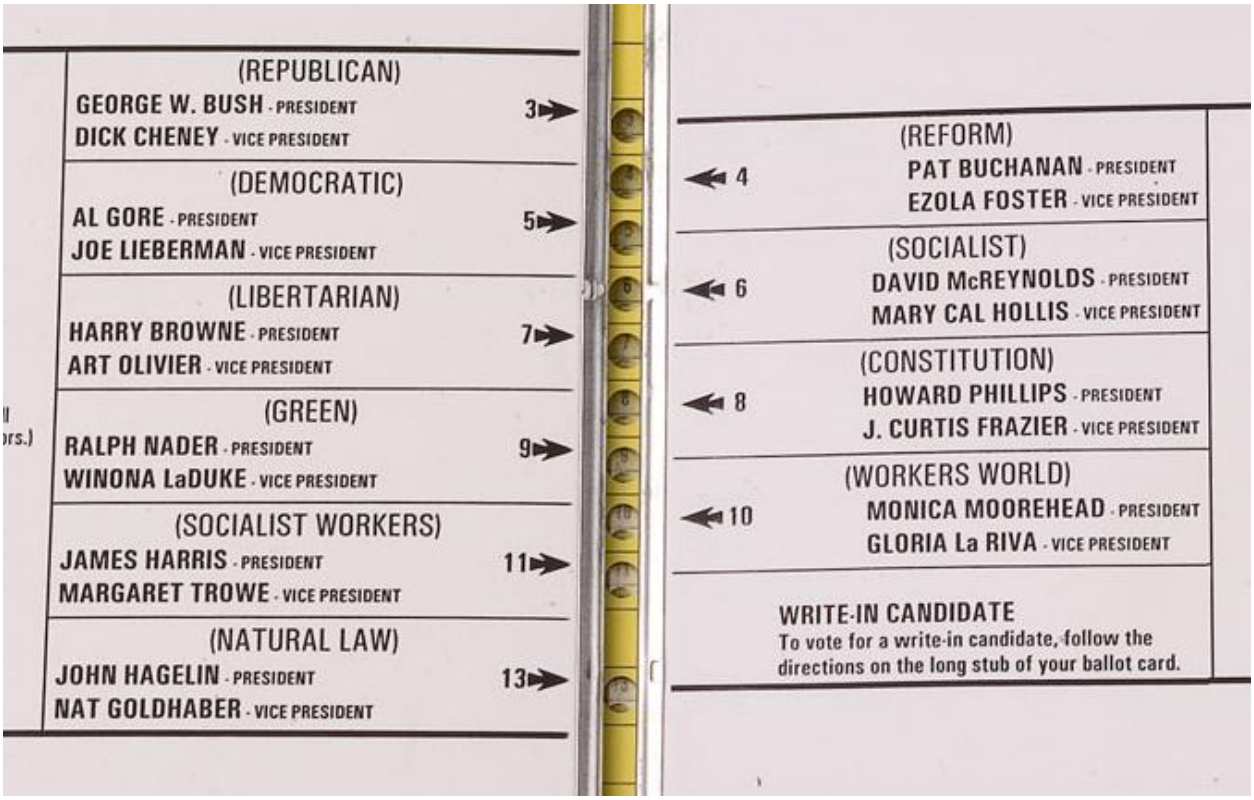

Figure 1.4: A butterfly ballot. The presidential elections in 2000, Florida. [3]

1.1.6 Direct-Recording Electronic (DRE) Voting System

As the Industrial Revolution lead to introduction of lever machines, the expansion of computers lead to new voting systems known as direct recording electronic machines. DRE machines are very similar to ATM machines, where the user has a small touch screen and selects different options (Fig. 1.6). Inside the DRE machines are typical computers with such typical problems to them as buggy and malicious software, hardware failures, and etc. Even the main problem with DRE is similar to lever machines! In case of any error, a voter doesn't have any receipt to verify that his vote was counted properly. For example, the study of one of the Diebold DRE machines concluded that elections can be easily tampered despite the cryptography used to

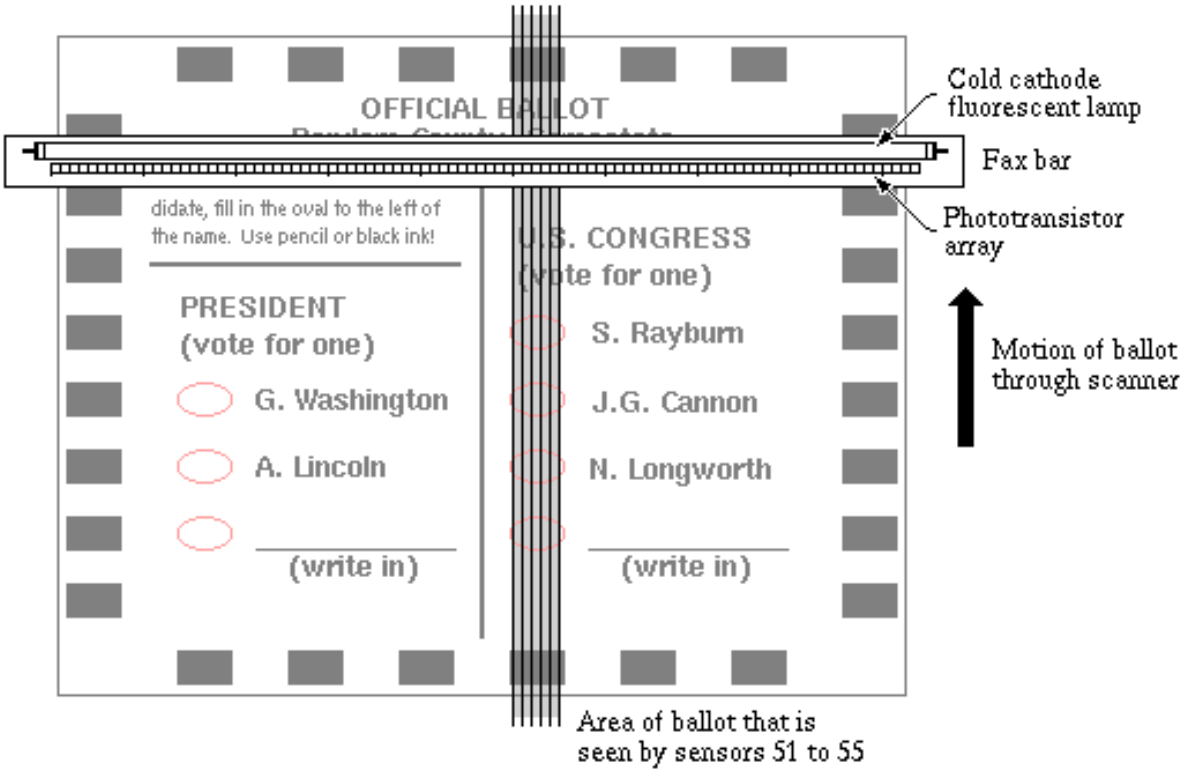

Figure 1.5: An optical scan machine. The process of scanning a ballot.

protect data [6]. Also surprisingly, DRE machines have about the same number of residual vote rates as the leader in inaccuracy - punch card machines [5].

1.1.7 The Voter-Verified Paper Audit Trail (VVPAT)

Kevin Thompson in his classical paper “Reflections on Trusting Trust” said [7]: “You can’t trust code that you did not totally create yourself.... No amount of source-level verification or scrutiny will protect you from using untrusted code.... A well-installed microcode bug will be almost impossible to detect.” Therefore, nobody can be assured that DRE machines are secure and don’t contain any malicious software. As a reaction to it, in 1992 a VVPAT was proposed. The VVPAT is similar to a cash register tape, except VVPAT is shown behind a glass to user and left within a machine. The voter

Figure 1.6: A Direct-Recording Machine. The process of scanning a ballot. [4]

makes choices on the screen, at the end a record is printed on a tape. The voter should visually check the correctness of the record and in case of mistake, cancel his vote. If everything is correct, than voter should confirm his vote.

1.2 Tallying Methods

Any democracy relies on elections as the representation of voters intents and preferences. While such an assumption is true in general, the outcome of elections may be heavily influenced by the design of an electoral system. An overwhelming variety of electoral systems exists, but plurality rule remains the most popular choice for its simplicity and intuitivity. The actual choice of a tallying procedure should be made by election officials depending on the criteria of optimality in a given decision setting. One

system can encourage voters participation. The other one may discourage the formation of bipartisan system. The third will increase chances to win for one of the sides, etc.

As it was noted before, most countries use a first-past-the-post (FPP) or in other words winner-takes-all voting for the presidential elections. Research showed that FFP is susceptible to gerrymandering, spoiler effect and other “attacks.” Gerrymandering is the process of rebounding voting districts in advantage of one specific party or candidate. The spoiler effect is an effect when votes are split between candidates of similar ideology. Spoiler candidate is drawing votes from a candidate with a similar ideology and therefore causing an opposite candidate to win the race, even if the opposite candidate represents a minority of a voters. Kenneth Arrow in 1951 [8] showed that people in power are tend to preserve a status quo. If current method of voting brought them an election win, there is no reason to change it with an unknown outcome on a next elections.

While we agree that such problems exist, an optimal solution is not a goal of this research. It is more related to political science than a computer science. For further reading [9], [10] are recommended.

- *Plurality rule:* A candidate which has been given a most of votes is a winner.
- *Approval voting:* Each candidate is approved or disapproved by a voter. Most approved candidate is a winner.
- *Condorcet system:* Candidates are compared to each other pairwise. A candidate with a majority is a winner [11].
- *Instant run-off (IRV) or Single transferable vote (STV):* Voters rank candidates in order of preference. If any candidate receives a majority vote, thus declared as a

winner. Otherwise, a candidate with a least number of votes eliminated from election. Votes from ballots that ranked eliminated candidate as a first preference, are transferred to candidates ranked on the ballot as a second preference and so on, until one of the candidates receives a majority vote [9].

- *Borda voting:* Voters rank candidates in order of preference. Each rank has a corresponding number of points. For example, there are n candidates. First candidate will receive n-1 points, second candidate n-2, etc. After all votes being counted, a candidate with a most of points is a winner.
- *Range voting:* Voters assigned some score within a range [e.g. 0 to 9] to each candidate. After all votes being counted, a candidate with a highest score is a winner.

1.3 Supervised vs Remote Voting

In terms of privacy, any remote voting scheme has a fundamental vulnerability. Election officials do not control the remote environment. Therefore, no matter what measures are implemented to secure the ballot secrecy, the adversary can violate it by standing behind a voter and observing his choice.

On the other side, supervised voting provides controlled environment where election officials can ensure the privacy of the voter. Also the security of many voting schemes is based on following strict procedures. For instance, the requirement to present a photo ID before receiving a voting ballot.

Despite the fundamental vulnerability for vote selling and coercion, remote voting becomes more popular among the US voters. While it has clear advantages to voters

overseas, the popularity of absentee ballots, early and mail in voting rise the concerns. The Voting Technology Project recommends to discourage any attempts to engage a broader coverage of remote voting [2].

1.4 Write-In Candidates

We should also note that many tallying methods allow write-in candidates. The write-in candidate is a person or a party who was selected by a voter, but wasn't listed on the ballot. Therefore voter enters candidate's name himself [12]. However, write-ins add unnecessary complexity not related to security directly such as the ability of optical-scan machines to recognize handwriting and properly read write-in candidates. So, we decided to leave ability to write-in a candidate in proposed schemes for further research.

CHAPTER 2

PRELIMINARIES

2.1 Why Voting is Hard?

Advancements in technology for the last 25 years completely changed the way we act in many fields, from listening to music and shopping to management of critical infrastructure. You can send an instant message to a person on an opposite side of the world and it will cost you nothing. You can have a video call during the flight over the Atlantic. One swipe of the credit card in the middle of China would initiate a complex money transfer from U.S. to Hong-Kong bank. Robots make most of the transactions on a stock exchange and make decisions in a fraction of a second. In the world where, banks are able to manage trillions of transactions each day, immediately reflecting it on accounts for a possible audit, development of a secure voting system shouldn't be a hard problem. In the common sense, it shouldn't be, but B. Scheiner [13] and later B. Adida [14] noted that such analogies are wrong. To prevent voter coercion in elections, anonymity is required. Any voting system must eliminate relationships between the voter and his ballot to ensure that no one is able to trace how the voter voted. Such property is known as the ballot secrecy requirement.

Compare it to the banking where banks has a sender and a recipient of a transaction. Bank keep record of all transactions and can audit the transaction history to verify its correctness. In case of a mistake, bank can easily solve the problem, e.g. return money to a sender. In voting, we know a recipient of a transaction (vote) but we don't know a sender. Even if we would able to detect a fraudulent transaction, we can't perform an audit and distinguish between legitimate and fraudulent votes in the tally. B.

Adida [14] perfectly described how banking and aviation would look like if it would have the same requirements as voting:

> If voting is compared to banking, then one should imagine a banking system where the bank cannot know the customer's balance, and even the customer cannot prove her balance to her spouse, yet somehow she receives enough assurance that her money is safe. If voting is compared to aviation, then one must imagine that pilots are regularly trying to crash the plane, and that we must ensure that they are almost always unsuccessful, even though, in this imaginary world, plane crashes are particularly difficult to detect. These significant additional constraints lead to a clearer appreciation of the challenges faced by voting system designers.

2.2 Key Properties for Voting Protocols

2.2.1 Integrity

Election results should be correct and not altered in any way. Therefore each stage of the elections must be honest [15], [16], [17]. [17] provides a good summary of each stage:

- Cast as intended: the vote recorded by the voting device (paper ballot, DRE, optical scan and etc) should match the vote that the voter intended to cast. It is more a usability requirement than a security one.
- Recorded as cast: a process of processing votes should not change the votes itself. It can change only the form of it. For example, scanning a paper ballot to electronic record.
- Counted as recorded: recorded ballots are counted correctly.

2.2.2 Privacy (Confidentiality)

A precise definition of privacy is given in [18]: Let two voters Assiya and Bob cast the ballots. Let v_1 and v_2 denote their votes. There are two possible cases exist: Assiya

casts v_1 and Bob casts v_2, Assiya casts v_2 and Bob casts v_1. If a third party cannot distinguish between the cases, then voting is private.

Two forms of privacy are exist:

- Weak form: nobody can associate casted vote with a voter.
- Strong form: nobody can associate casted vote with a voter, even if the voter wants to reveal it.

A voting system should be also coercion resistant and provide receipt-freeness.

- Coercion resistant means that the coercer is not able to determine if the voter casts a vote in a particular way.
- Receipt-freeness means that voters can't prove to the third party how they voted. Voters should not have or be able to construct any evidence of how they voted [41]. But the voter can have a receipt that doesn't reveal any information about his vote. Receipt-freeness is important to prevent vote selling [17].

2.2.3 Verifiability

Another requirement for voting protocol that is close to integrity property is a verifiability. The correctness of a final tally and processing of votes should be verifiable and auditable. If anyone can check that all votes in a final tally was counted properly and no votes were changed during the voting it is called universal verifiability. If only a voter is able to check that his/her vote was counted as intended and included in the final tally, it is called voter verifiability. If each stage of the election can be verified that is called end-to-end verifiability [16], [19], [20].

2.2.4 Other properties

There are other properties of a voting protocols that are important, but out-of-scope of this paper:

- Voter authentication
    - Only authorized voters can participate in election
    - Each voter should vote only authorized number of times
- Enfranchisement: all authorized voters have the opportunity to vote
- Accessibility: voters with disabilities should be able to cast the vote.
- Usability: the election system should be easy to run by election officials and to use by the voters

As it was noted before, any voting scheme should satisfy to several mutually exclusive requirements such as privacy and verifiability. Verifiability requires elections to be transparent as much as possible. At the same time we should avoid any mechanism that comprises a voter's privacy. It is always a trade-off between these two requirements. In addition, there is no trusted third party that can perform the audit and verify the outcome of elections. And what makes situation even worse that different groups of voters and parties do not trust each other. There is a long record of revolutions and riots started from doubts about correctness of the result of elections. Therefore verification mechanisms should be clear and understandable to convince as many people as possible.

2.3 Cryptographic Primitives

2.3.1 Pseudo-Random Number Generator (PRNG)

A PRNG is an algorithm for generating a set of numbers similar to the set of truly random numbers. It is called pseudo-random due to the fact that numbers are not truly random, but generated from the seed number. Important property of PRNG is that it will produce the identical set of numbers every time it was initiated by the same seed.

2.3.2 Hash Function

Hash function is an algorithm that derives a single fixed length value (hash) from an arbitrary long sequence of data. One of the important properties of hash function is that it is practically impossible to modify the message without changing the hash. Therefore, the hash is used to verify the integrity of the message. Proposed cryptography-based voting scheme use hash function to ensure the integrity of the ballot.

2.3.3 Public-Key Cryptography

Prior to the 1970s, cryptographic protocols were based on idea of the shared secret. Two parties should share the secret or in other words, the key before transmitting encrypted messages. It is an easy task, if you can encrypt a next key with a previous one, but how to establish a secret if no secret is shared in advance? Sending a secret without protection over an insecure channel is a risky business. Therefore, the best choice would be to have a physical meeting to share the key.

Also, each user should have a shared key with each party he communicates. Imagine for a second that you’re owner of a popular service visited by millions of people

around the world each day. Even if would solve a problem of distributing initial keys, you would have to maintain a million of keys!

In 1979, Whitfield Diffie and Martin Hellman presented their revolutionary paper “New Directions in Cryptography” [21]. They proposed to perform encryption and decryption using different keys. The key for encryption should be made public and available to anyone, while decryption key should be kept secret. This approach was named the public-key cryptography. It has solved the problem of distribution of a shared secret and maintaining different keys for each engaged party.

2.4 Current Approaches to Voting Schemes

Let’s consider the current approaches to voting from classical voting system to systems based on cryptographic primitives.

2.4.1 Classical Voting System

The classical voting system is quite simple. On election day, a voter marks his preferences on the preprinted ballot (can be both paper or electronic), cast the ballot to a ballot box or a machine and leaves the precinct. The security of election is based on assumption that voter should trust to the election officials and people who count the tally. As soon as the ballot falls into the ballot box, a voter establish the trust to a chain of custody. There is no way to verify that his ballot was counted as intended. As most of the current voting systems used in elections, classical voting scheme preserve privacy against verifiability (See Fig. 2.1).

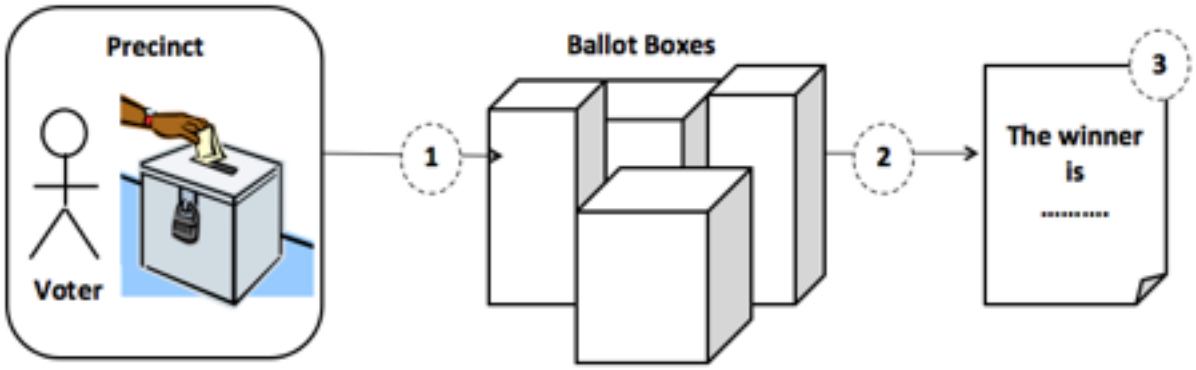

Figure 2.1: A classical voting scheme. (1) A voter marks his choice on the ballot and casts it to the ballot box. (2) Ballot boxes are collected and tallied. (3) The victor is announced.

History showed that voting mechanisms that were introduced to solve the problem of voting security miserably failed. Mechanisms not only didn't help against the fraud, but became a new point of failure.

The recent Caltech/MIT Voting Technology Project report [2] summarized research and analysis for the past 12 years since the controversial 2000 presidential election (Al Gore vs George W. Bush). They provided several recommendations for improving the administration and technology of elections in the United States. In general, they suggest to standardize the voter registration and verification across the states, improve the work of polling places and poll worker training and expand support for research on voting field. They are also recommend to limit usage of absentee balloting, mail and internet voting due the significant risks for voter privacy and possibility of fraud. The most interesting part for us is the voting technology suggestions. Instead of setting security standards for election equipment they suggest a legislation mandating effective election auditing and statistically meaningful post-election audits.

2.4.2 Non-Cryptographic Schemes

Non-cryptographic schemes represent a small group of voting systems that are based on different ballot designs and methods to verify a voter's choice. They usually provide a nice level of end-to-end verifiability, but still too vulnerable for practical implementation. For instance, the "ThreeBallot" voting scheme uses a paper ballot that separated into three independent parts. The voter marks all three ballots in particular way and leave the copy of one of the ballots as a receipt to verify his vote later. Election officials publish the final tally and all casted ballots. The voter can check if the ballot that he took as a receipt is included in the tally. The receipt do not reveal any information about voter's choice because of certain restrictions on how the ballots should be filled out [35]. Several vulnerabilities were identified in ThreeBallot, as in other non-cryptographical schemes. In general, such schemes do not achieve the same level of security that more advanced cryptographic schemes can provide.

2.4.3 Cryptographic Schemes

Most of novel voting schemes proposed by researchers include some kind of cryptography, because cryptography provides efficient methods and primitives for data integrity and anonymization. My proposed cryptography-based scheme is highly influenced and based on ideas from two cryptographic schemes: Civitas [16] and Pret-a-Voter [17].

Civitas is the end-to-end verifiable scheme designed for remote electronic voting. It uses zero-knowledge proofs for voting registration and to ensure the honesty of tellers. It also uses RSA algorithm as public-key cryptography for keys generation. The

overall security of the scheme is strong, but requires two weak assumptions: a secure registration protocol and a scalable vote storage system.

In contrast, Pret-a-Voter uses cryptographic primitives for supervised paper voting. The PAV ballot designated in such way that a voter handle it as typical paper ballot for optical-scan machine. But the implementation of visual cryptography provides required security and verifiability.

CHAPTER 3

THE PAPER-BASED VOTING SCHEME

3.1 Scheme Overview

3.1.1 Assumptions

Both proposed voting schemes require certain assumptions to run correctly. General assumptions are those which apply to both schemes. Specific assumptions apply only to the scheme described in this chapter.

General assumptions:

- Pseudo-random number generator (PRNG) is trusted.
- In general, election officials are honest.
- The adversary is malicious third party trying to influence on the outcome of elections.

Specific assumptions:

- Protocol is developed to be a supervised voting.
- Transmission channels are secure and untappable.
- Electoral roll (the list of eligible voters) is publicly available and valid.

3.1.2 Setup

Ballot. A voting ballot printed on a single sheet of paper with a perforation line in the middle (Fig. 3.1). This line will allow later separate a ballot in two part: candidates part and verification codes part.

Candidates part contains names of the candidates with an op-scan bubble, later filled by a voter. Candidates are placed in random order.

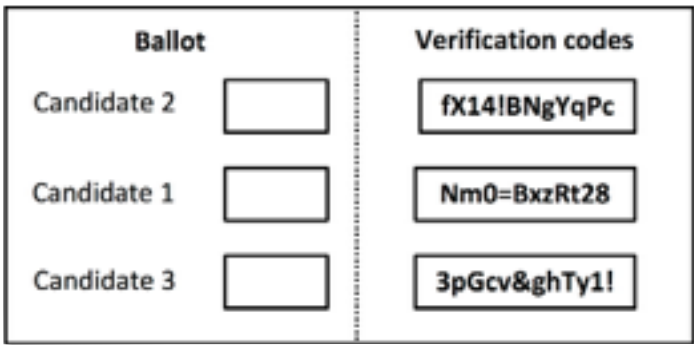

Figure 3.1: A voting ballot. A left side of the ballot consist of candidates in a random order. A right side has a several verification codes and later will play the role of a reciept.

Verification codes part contains preprinted pseudorandomly generated codes. Each code should be unique and shouldn't be repeated at any other ballot. Codes are unrelated in any way to candidates or candidates order. Therefore, number of codes can be equal or greater than the number of candidates. For security reasons, number of codes less than a number of candidates is not recommended. Codes part will later play a role of a receipt to verify that a particular vote was properly tabulated and is present at the final tally. Verification codes for each election on each ballot are uniformly, pseudorandomly and independently selected from the set of possible codes generated by a pseudorandom number generator from a secret seed [17], [22]. The seed must be kept secret so no one else is able to generate the same codes. Unavailability to generate the set of possible verification codes protects against attacks based on fabricated receipts that are explained later in security analysis section.

Public Bulletin Board. The public bulletin board is a general description for robust, authenticated public broadcast channel [14]. The bulletin board is used to publish information required to perform the audit of the elections. The board should be write-in only. Once information is published it cannot be changed, removed or altered in

any way. While we do not cover the exact scheme for public bulletin board, there are known algorithms ready for implementation [23].

Optical Scan Machine. An optical scan machine uses optical scanner to recognize marks in an op-scan bubble on the ballot. In our scheme, the op-scan machine should recognize the mark for an intended candidate, the position of the mark, and marked confirmation code. Last two fields are immediately published on the public bulletin board. We assume that the channel between bulletin board and op-scan machine is secure. The actual vote is saved in secret on the memory device. Before elections several procedures should be performed to ensure the security of the device. The internals of the optical scan machine must be protected and sealed. The election officials should carefully test the op-scan machine to verify that it works properly.

3.1.3 Voting

Figure 3.2 gives an overview of the scheme. The voter casts his or her vote by putting a mark to an op-scan bubble opposite the name of the preferred candidate. In the same way the voter marks a code. After the voter indicated his choice, he inserts a ballot to an optical scanning machine. The machine records the voter's choice, position of the candidate in an order and the code (ex: Candidate 1, position #2, code: fX14! BNgYqPc)(Fig. 3.3). Later the voter separates the ballot into two pieces, he or she casts the candidates part by putting it into a usual ballot box. The voter leaves the verifications codes part as a receipt for possible audit (Fig. 3.4).

3.1.4 Tallying

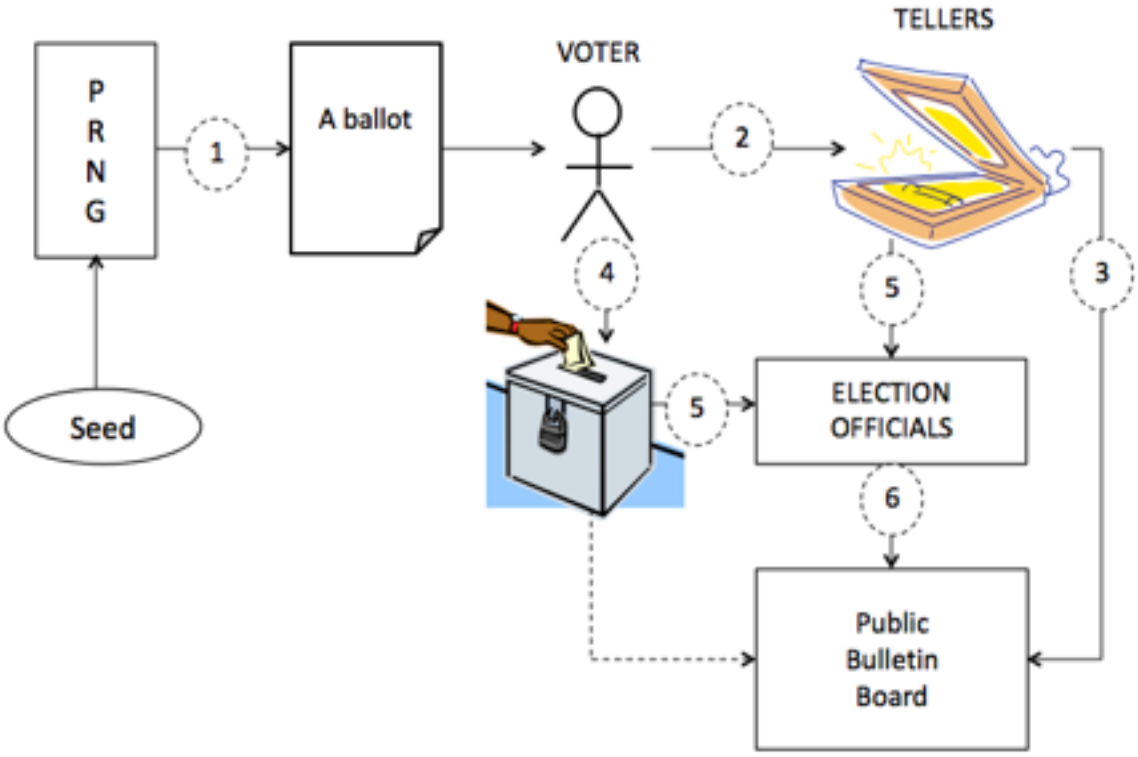

Figure 3.2: The paper-based scheme. (1) PRNG generates a set of verification codes using a secret seed. (2) Voter marks his preferences and scan the ballot. (3) An optical scan machine immediately sends a verification code and position of chosen candidate to the public bulletin board. (4) Voter casts the left part of the ballot to the ballot box. (5) Election officials tabulate the tally using records from an optical-scan machine and paper ballots. (6) Then election officials announce the outcome of the race. (7) Voter can verify a position of his candidate

After voting, election officials reveal the outcome of elections based on data from optical scan machines and a victor is publicly announced.

Election officials also publish the list of voters who actually voted and a 2 column table on the public bulletin board. Unique codes are placed in the first column of the table. The second column is related to the code position of the candidate that was chosen by the voter. Number of rows should be equal to the number of voted individuals.

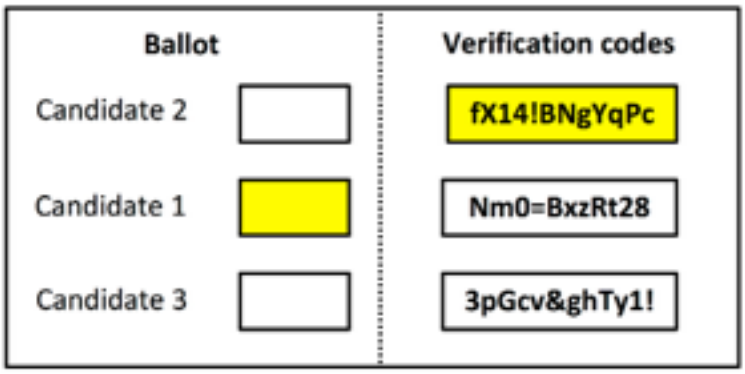

Figure 3.3: Marked ballot. A voter marked his preferences using a yellow marker.

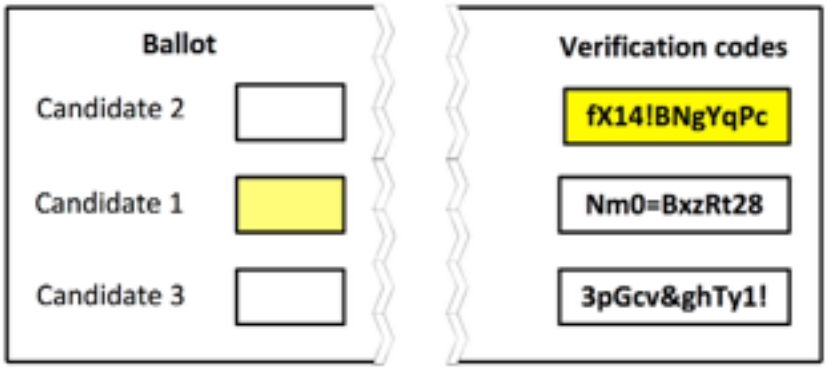

Figure 3.4: Separated ballot. After scanning the ballot, the voter separates it into two parts. Left side is casted to the classic ballot box. Right side left as a receipt to verify the presence of the voter's choice in the final tally.

Voters
Assiya
Bob
...
...
...
Zach

Codes	Position
fX14!BNgYqPc	2
Bqm3(9/cRt44	3
...	...
...	...
...	...
vh$v*DP9x&p2	1

Figure 3.5: Audit tables. First table is a list of all voter participated at the elections. Second table has a verification code marked by some voter and a position of the candidate he voted for.

3.1.5 Audit

Any voter can check the presence of his chosen unique code and position of the candidate he voted for on the public bulletin board. If the bulletin board doesn't contain a code or the position is different from what he or she voted for, then the voter can file a complaint (Fig. 3.5).

Any person or organization can count the total number of voters and compare it to the total number of codes. If they do not match, election officials should initiate the manual recount or rescan of the cast of paper ballots.

While individual verifiability is possible, total public verifiability is not yet achievable.

3.2 Security Analysis

We must perform a security analysis of proposed scheme to check if it satisfies requirements for a voting system. Two main requirements are an integrity of the election outcome and privacy of the voter. We consider a verifiability as a mechanism to ensure the integrity.

3.2.1 Integrity

Modification or elimination of ballots. We assume that final tally was tabulated twice. First time by optical-scan machine at the time when a voter casts his ballot. Second time when election officials perform an election audit by recounting paper ballots. Let's imagine that adversary was able to modify ballots on the memory card of an optical scan machine. On each casted ballot, the adversary changes marked

candidates but leaves position number and verification code unaltered. Voter's audit wouldn't reveal it because election officials will publish only positions and verification codes. But recount of the paper ballots will detect it. The same will happen if adversary would change only paper ballots, leaving the optical scan machine data unaltered. Therefore, the adversary should modify both paper ballots and data on optical scan machines to succeed. This attack is possible, but requires to have an access to op-scan machine internals and ability to change ballot boxes or ballots themselves. It would be practically impossible to realize this attack without the help from corrupted election officials, which contradicts to our assumption that they act honest.

Ballots stuffing. Let assume that the adversary was able to stuff either paper ballots or votes on op-scan machines. This attack would be easily detected because the number of paper ballots and ballots tabulated by op-scan machine wouldn't match. Even if the adversary was able to add additional votes to both op-scan machines and ballot boxes, the total number of votes should match the number of voters who actually casted their ballots. Both numbers are publicly announced on the bulletin board. To achieve success, the adversary should add more voters on the list. Otherwise fraud will be detected.

Now let's assume that the adversary was able to add some voters' names to the list of actually voted citizens. If the adversary adds legitimate voters, who didn't vote then there is a probability that someone will detect that person who didn't participate in election, appeared on the list of voters. It is detectable with high probability, but unfortunately, not guaranteed [24].

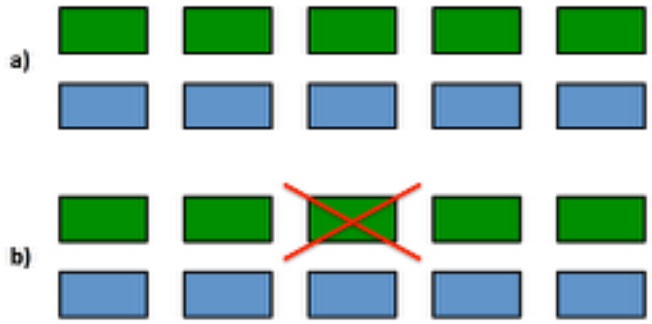

Figure 3.6: Attack on legitimacy. (a) The result of a election is a tie. (b) The adversary attacks the legitimacy of the result of one of the districts, where the green candidate will win with a high probability. Depending on a policy, votes from that district can be not included in the final tally. Instead of original tie, the blue candidate win the race.

The adversary can add fake voters. Then auditors should check the list of voters who actually voted to the list of the registered voters. Any mismatch should raise a concern about integrity.

Attack on legitimacy. The goal of adversary can be not only adding votes to a preferred candidate, but also eliminating votes from an opposite party. On the figure 3.6 10 voting districts are pictured. In five districts, a green candidate won. In other five, a blue candidate won. Knowing the districts where green candidate would win the elections the adversary votes in one of them. Later, he can claim that he didn't vote and his name was included in the list of voters by somebody else. If there would be a relatively high number of such voters, the legitimacy of the result in this particular district can be a question. Depending on a policy, votes from that district can be excluded from the final tally. To prevent such an attack we need to have some confirmation that voter casted his or her ballot. For instance, it can be a stamp printed in the passport at the time of receiving a ballot. As a result the voter will not be able to vote and pretend later he didn't. He will be required to present a passport without the stamp. Of course, he can loose the passport, but one voter can't question the legitimacy of the result. It there

would be many voters like that, it will be really suspicious that many voters simultaneously lost their passports after elections.

Vote selling. To coerce a voter an adversary should have some confirmation that the voter voted in a way the attacker demanded. The receipt (part of the ballot with confirmation codes) do not reveal any information on how the voter voted, except the position of the candidate. The adversary cannot verify if voter marked position of desired candidate because it is assigned randomly at each ballot. Therefore vote selling is impossible.

Paying for receipts. The adversary may obtain receipts from voters to change later the votes corresponding to the codes at receipts. Now knowing that without receipt a voter won't be able to protest a result of tally, the receipt-buyer alters the vote, the data from an optical scan machine and replaces ballots with of opposite candidates to the intended one. Such attack is possible, but requires the adversary to have access to all voting equipment. As it was discussed before, it is only possible if election officials are corrupted. Most of other voting schemes including cryptographic ones, vulnerable to this attack. Educating voters to keep their receipts or discard them correctly considered by researchers as possible countermeasure [25], [19].

Chain voting [25]. In the chain voting attack the adversary must obtain a blank ballot. One can be stolen before elections or counterfeited. A ballot can also be legitimately obtained at the precinct, but not used immediately. Now the adversary marks the ballot for a desired candidate and hands it to a coerced voter. Voter enters a precinct, receives a blank ballot, but casts the one he received from adversary. After leaving a precinct, he gives recently obtained blank ballot to adversary. Now an attacker

can repeat the cycle. In my scheme, chain voting can be easily prevented by a simple procedure. When the voter pick-ups his ballot, one of the verification codes should be recorded by election officials. After voter casts his ballot he must present the part of ballot with confirmation codes. Election officials check the presence of previously recorded code verifying that casted ballot is the one received before by this voter. If confirmation codes do not match, authorities should initiate an investigation. Unfortunately, voter will be able to cast pre-marked ballot, because checking procedure happens after casting the ballot. But most of the elections will tolerate a relatively small number of coerced votes without affecting the outcome of elections. If not, election officials can perform the check before voter casts the ballot. In this case, voter privacy should be protected by exposing only confirmation codes part.

Randomization attack [26]. The potential vulnerability occurs if an adversary forces the voter to vote only for the specific position no matter what candidate it is. If the voter provides the adversary with the confirmation code, the adversary can later check if the voter marked an intended position. The outcome of such attack depends on the number of candidates participating in the race. It is most efficient in case of two parties. The adversary have 50% chance that a desired candidate will present at the chosen position and marked by a coerced voter. Chances of success are getting lower as the number of participating candidates increase. The potential solution in case of bipartisan election, is to generate a number of fake candidates. Therefore, the probability of success of voting for the candidate that the adversary insists decreases. But the downside, is that we also decrease the chance for voting for voter's desired candidate.

3.2.2 Privacy

Voter privacy is the second main requirement to the voting scheme. It is important because voter privacy prevents vote selling. Even if the voter wants to reveal his choice, there must be no evidence to support his words. Otherwise, voter could sell his vote or force someone to vote in a particular way. Let's analyze the privacy of the paper-based scheme starting from a receipt.

Receipt. As we discussed in Section 3.1.2 codes are unrelated in any way to candidates or candidates order. Even after voter marked one of the codes, it doesn't have any relationship to the voter's choice. The code is just a unique ID to verify if the ballot was counted properly. Therefore, we conclude that receipt doesn't violate voter privacy.

The op-scan machine. The most possible critical point to attack is the op-scan machine, because it contains all necessary information to identify the voter's choice. An adversary can install a malicious software far before the elections, and as we know from Ken Thompson's article [7] it almost impossible to find out if the software of a machine is compromised. A possible solution is to divide the functionality between two op-scan machines. One saves a position of a candidate and a chosen confirmation code, another one reads the name of the marked candidate. In general, any electronic equipment can be compromised what makes this attack almost unavoidable.

3.3 Usability

The usability of the paper-based scheme is close to conventional voting systems. In addition to marking one candidate, a voter must also mark one of the confirmation

codes. No complex machine is used, only a simple optical scanner. Separation of the candidates part and cast it to a ballot box is also very simple. We must admit that even if separately each action is very simple, in general, number of procedures and following them can look more complex. In conclusion, this scheme provides almost the same usability level comparing to conventional voting systems, but provides voter verifiability and ensures integrity.

3.4 Discussion and Further Improvements

The security analysis showed that our paper-based scheme is quite secure, provides a voter verifiability and robust to most of the vulnerabilities described before and in [26], [17].

Further improvements should concentrate on achieving the universal verifiability. Combined with an existing voter verifiability the scheme will become end-to-end verifiable.

One of the weakest assumptions of the scheme is that election officials act honest. Usually its officials who are coerced by the adversary. To eliminate such an assumption we may add an additional level of security using cryptographic primitives. Similar procedures were implemented in Scantegrity II [22] and Pret-a-Voter [17].

CHAPTER 4

THE CRYPTOGRAPHY-BASED SCHEME

4.1 Scheme Overview

The continued research for the last 25 years in electronic voting brought many proposals and schemes [27], [12], [28], [29], [22], [30], [31], [20]. While many of them are theoretically secure and end-to-end verifiable, just a few had an experimental deployment. A little success in such a sensitive for the modern world area shows that secure electronic voting is a tough problem. Attempts to tackle this problem using cryptography primitives resulted in common approaches for solving subproblems such as anonymizing votes by mixnets [32], public key cryptography for integrity purpose, etc. Our scheme is not an exception. We suggest zero-knowledge proofs for voter's authorization and public key encryption to ensure the integrity of the ballots. In general, this scheme was inspired by Clarkson et al [16] and Pret-a-Voter protocol [17].

4.1.1 Agents

- Voters - eligible voters with an ID containing some secret.
- The registrar - an entity that authorizes voters, checks their eligibility and provides a voting token.
- Tellers - entities that play role of remote electronic ballot boxes.
- Election officials - an entity that finalizes the tally and announce the outcome.
- Auditors - individuals or organizations that want to verify the outcome of elections.

4.1.2 Assumptions

- The registrar acts honestly.
- Each voter has an ID with a public - private key pair.
- The registrar can securely authorize the voter and his\her eligibility to vote.
- Voter can easily connect to legitimate entities.
- At least one honest teller is available.

4.1.3 Setup

First, the registrar publishes a list of participating candidates on the public bulletin board. At the same time the registrar generates a set of voting tokens using a pseudo-random number generator (PRNG) with a secret seed.

Second, the election officials publish their public key and a list of authorized tellers. Tellers can be the representatives of the participating candidates, independent entities or state authorities, whereas election officials represent the state representatives or the election assistance committee (EAC) (Fig. 4.1).

4.1.4 Voting

A check-in phase from a classical voting scheme is represented by authorization procedure. A voter connects to the registrar to verify his identity. The registrar initiates an authorization protocol. In the section 4.2, an example of such protocol is presented. It uses zero-knowledge proofs to verify the voter identity, but the actual implementations may use any other method that can provide a secure authorization. If the voter is successfully authorized, the registrar randomly assigns one of previously generated

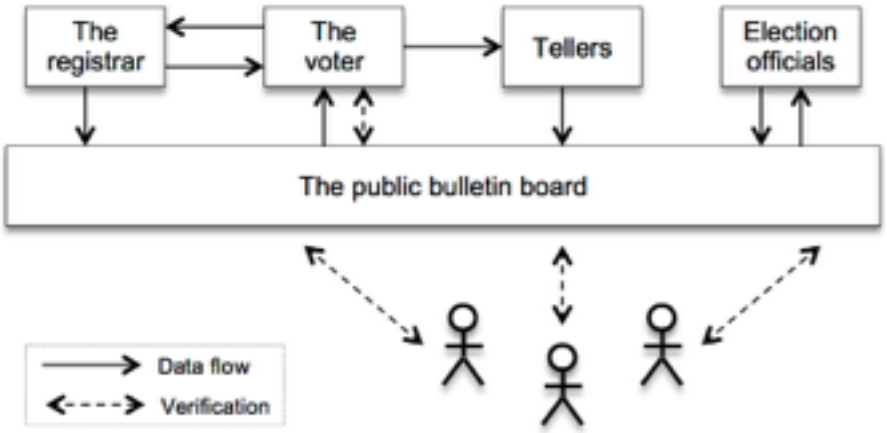

Figure 4.1: A high level concept of cryptography-based voting scheme. The registrar authorizes the voter, who generates the ballot using information from the public bulletin board (PBB). Then the voter casts the ballot by sending it to tellers. Tellers maintain voting chains, which published at the PBB after election is closed. Election officials tabulate the tally and announce the winner. All information on the PBB is publicly observed and can be audited.

tokens. At each race, each eligible voter should receive only one token, despite a number of check-ins. Each token can be assigned at most once.

After obtaining the token the voter looks at the public bulletin board to obtain an election official's public key and choose a candidate along with tellers. To cast a vote voter must generate a ballot. The ballot consist of the vote, the token encrypted by a public key of the authorities, and a hash of the "vote + token" pair (Fig. 4.2). The voter casts his ballot by sending it to all available tellers.

As soon as the election begins, tellers start to accept ballots. Each time they receive a ballot, they add it to the end of a voting chain similar to one used in Bitcoin [33]. The voting chain is a chain of ballots, where each ballot store the hash of a previous ballot (Fig 4.3).

4.1.5 Tallying and auditing

Figure 4.2: A ballot structure. A vote part consist a chosen candidate. Then it followed by a ciphertext, a token encrypted by the public key of election officials. Last field, is a hash of the vote and the token.

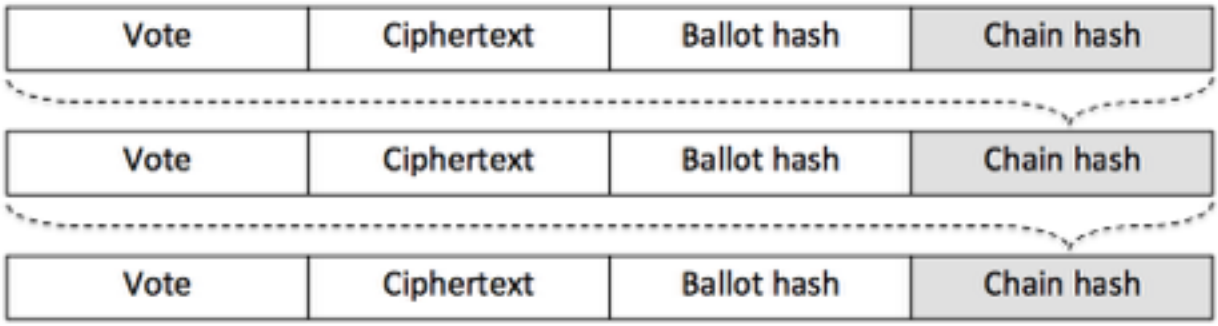

Figure 4.3: A voting chain structure. A hash of previous ballot is added to each new ballot in the chain.

As voting finished, the registrar waits until all tellers publish their chains on the bulletin board. Then the registrar publishes the seed and the list of assigned tokens, so anyone can validate votes in the chains.

Election officials and auditors start to tabulate the tally by the following procedure:

1. Generate the set of tokens using a published seed.
2. Compare it to the list of assigned tokens and verify it's correctness.
3. Election officials decrypt ciphertexts using the private key and retrieve tokens from ballots. Since the auditors don't have the private key, they encrypt assigned tokens and compare them to ciphertexts.
4. Remove all ballots with tokens that are not included in the set of valid tokens.
5. Compute hash for each vote and token pair.
6. Compare computed hashes with hashes from chains.

7. Remove all votes where hashes do not match.
8. Remove all votes that have the same valid token, but different choices (overvote).
9. Remove duplications. For each token at most one vote is retained.
10. Ballots that left considered as valid ballots. Count them.
11. Publicly announce a victor. Only election officials can publish the final tally and determine the victor.
12. Voters can verify that their votes were counted as intended and included in the final tally using their tokens. Auditors can tabulate ballots and verify the correctness of the outcome. In case of disagreement with official results voters or auditors should file the protest.

4.2 Technical Details

Proposed scheme uses cryptographic primitives as building blocks. Verification mechanisms are based on PRNG property to generate a set of unique tokens and verification codes from some secret number. Voters encrypt their tokens using public key cryptography to prevent tampering by an adversary (Fig. 4.4). In this section we review algorithms for three cryptographic primitives: public key and hash algorithms, and zero-knowledge proofs. Pseudo-random number generator was described earlier in the section 3.2.

4.2.1 Authorization protocol

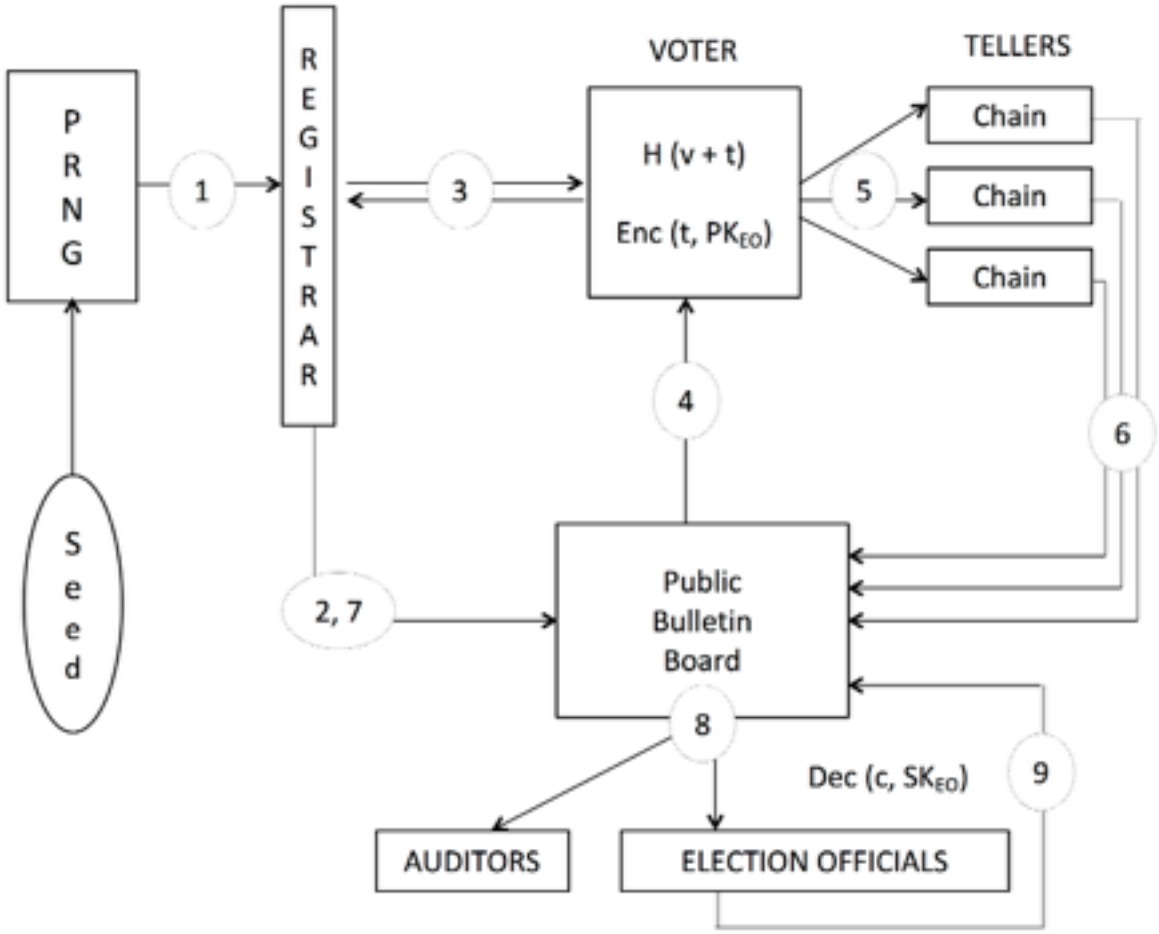

Figure 4.4: A detailed cryptography-based voting scheme. (1) PRNG generates a set of tokens using a secret seed. (2) A registrar publish a list of participating candidates on the public bulletin board (PBB). (3) The registrar authorizes a voter and provides with a voting token. (4) The voter obtains a public key of election authorities, denoted as P_{EO}, a list of candidates and authorized tellers. (5) The voter generates a ballot and sends it to tellers. (6) Tellers maintain voting chains until the end of election. Then they published chains on PBB. (7) As soon as all tellers published their chains, the registrar publish a secret seed and a list of assigned tokens. (8) Auditors and election officials tabulate a tally. (9) A winner is publicly announced.

The authorization protocol is presented only as an example for possible solution and can easily exchanged to any other scheme. The protocol uses Schnorr's zero-knowledge proof protocol [34].

The trusted third party known as Key Authentication Center (KAC) verifies the identity of the voter and generates an identification number *I* (containing name, address,

ID–number etc.) and generates a signature *S* for the pair (*I, v*) where *v* denotes user's public key. KAC also publish primes *p* and *q*, a base *α* for the discrete logarithm such that $\alpha q = 1 \pmod{p}$ and a one–way hash function *h*. The protocol is following:

- The voter:
  - picks random *r*.
  - computes *x* such that $x = \alpha^r \pmod{p}$.
  - computes hash of *x*, *hash(x)*.
  - sends *I*, *v*, *(S)*, and *hash(x)* to the registrar.
- The registrar:
  - picks random *e*.
  - sends *e* to the voter.
- The voter:
  - computes *y* such that $y = r + se \pmod{q}$.
  - sends *y* to the registrar
- The registrar:
  - computes $x' = \alpha^y v^e \pmod{p}$
  - check that *hash(x) = hash (x')*

4.2.2 Public-key algorithm

Classic encryption protocols are based on secret that should be shared before the transmission. In 1976, Whitfield Diffie and Martin Hellman proposed the revolutionary public key encryption, where the secret is derived from a public-private key pair [21]. It was only theorically possible until 1977 when Rivest, Shamir, and

Adleman introduced first implementation named after them [35]. Currently, the RSA algorithm is the most popular and widely adopted public key algorithm. While our scheme can use any public key algorithm, we choose RSA for its convenience.

The security of RSA algorithm is based on integer factorization problem. Given two random prime numbers, p and q, it's easy to find their product, $n = p * q$. But given only n, its hard to recover the prime factors.

Key generation. The RSA algorithm generates public-private key pair by the following procedure [36]:

- Generate two random prime numbers p and q of the same bit-length.
- Compute $n = p * q$. n is a modulus for public and private keys.
- Compute $\phi(n) = (p - 1)(q - 1)$
- Choose an integer e between 3 and $\phi(n)$. e is the public key exponent.
- Determine d such that $d * e \equiv 1 \pmod{\phi(n)}$. d is the private key exponent.
- Produce:
  - The public key: (n, e).
  - The private key: (n, d).

In the scheme (Fig. 4.4), PK_{EO} denotes the public key and SK_{EO} denotes the private (secret) key of election officials.

Encryption. The ciphertext denoted by c, the message by m. In our case, message is a voting token denoted by t.

- $c = Encrypt(m) = m^e \bmod n$

Decryption. Election officials decrypt the ciphertext to obtain a token.

- $m = Decrypt(c) = c^d \bmod n$

4.2.3 Hash function

We implement the set of cryptographic functions designed by National Security Agency and known as SHA-2. SHA stands for Secure Hash Algorithm and consists of four functions with a different length of hashes, ranging from 224 to 512 bits. The exact explanation of SHA-2 function is quite lengthy, therefore is absent here. For those who are interested in details, the official description from National Institute of Standards and Technology (NIST) is recommended [37].

4.3 Security Analysis

4.3.1 Integrity

The cryptography-based scheme provides both voter and public verifiability. In the thesis, we present the informal and intuitive arguments, without providing a formal analysis that requires an additional research.

An adversary pretends to be a voter during verification. While the attack is theoretically possible, the realization depends on many factors such as verification protocol, the strength of ID, security of the channel and etc. We are assuming that each voter has an ID with public-private key pair. It can be embedded into a hardware (e.g. smart cards) to prevent distribution of keys. Severe penalties and economic incentives can be introduced to discourage voters to transfer hardware to the third party. For instance, the Republic of Estonia implemented the national ID cards with private keys stored inside the chip [38]. Keys are used at the general election through internet and for legally binding digital signatures. Therefore, selling the ID card is gainless.

Some tellers are malicious. Suppose all tellers except one are malicious. The adversary can tamper incoming ballots by changing candidates to the preferred one. Such fraud will be easily detected by election officials and auditors, because computed hashes won't match hashes in the ballot. The adversary can't generate a valid hash without knowing a related token. While tallying such ballots will be discarded. The real ballot will still be counted because one honest teller is still present in the system. This teller will publish a valid ballot with a proper hash. Instead the adversary can start to simply reject votes. Again, if at least one honest teller available, votes will be counted. To ensure that the ballot will be received by at least one honest teller, the ballot must be casted to all tellers.

Denial of Service attack on tellers. DDoS attack is a popular weapon to make service unavailable to legitimate users. Suppose the adversary successfully attacked all honest tellers, except one. At the previous subsection we proved that until our assumption about one honest teller present in the system is valid, the system will work and be able to count all votes correctly. System will be compromised only if all tellers are malicious, which contradicts to the assumption.

An adversary simulates the election entities (the registrar, tellers, public bulletin board). In real deployment the adversary can poison DNS entries to redirect voters to malicious bulletin board or registrar. Again, such attack is possible, but out of a scope of this thesis. It also violates our assumption that voter can easily connect to legitimate election entities.

Voter's machine is compromised. Attacks of this type includes trojan and backdoors installed on a voter's machine, malicious clients, and etc. Malicious software

can compromise the process of generating ballot. Several measures should be enforced to increase assurance in the software integrity. The voting client is open-sourced and audited by experts. Voting may require a smart card to generate a ballot. Machines can be pre-certified as trusted voting clients. But even all strict measures implemented, we can't be sure that either software or hardware doesn't contain vulnerabilities. As Ken Thompson noted in his well-known article "Reflections on trusting trust" [7]: "You can't trust code that you did not totally create yourself.... No amount of source-level verification or scrutiny will protect you from using untrusted code.... A well-installed microcode bug will be almost impossible to detect."

Man-in-the-Middle attack. The election officials can change marked candidate in the ballot by a man-in-the-middle attack. To prevent it, a secure channel must be used for transmissions between voters and tellers, and between tellers and a public bulletin board.

Both the election officials and teller are malicious and collude to commit fraud. The election officials can decrypt the ballots while malicious tellers will tamper the votes. It is very dangerous attack because such fraud would pass undetectable. Even if voters will protest the outcome, hashes will prove that encrypted token and vote wasn't touched. Both election officials and tellers can pretend that they're honest and there is no direct evidence of their fraud. To prevent such an exceptional, but possible situation election officials and auditors should collectively generate a public-private key pair where public key is common, but parties hold a share of a private key. Therefore, all the parties (all auditors and election officials) should be engaged in decrypting the ballots.

4.3.2 Privacy

Voter privacy is the second main requirement to the voting scheme. As it was noted before two forms of privacy exist. Weak form: nobody can associate casted vote with a voter. And strong form: nobody can associate casted vote with a voter, even if the voter wants to reveal it. Voter privacy is important to prevent vote selling. Even if the voter wants to reveal his choice, there must be no evidence to support his words. Otherwise, voter could sell his vote or force to vote in a particular way.

Unfortunately, in terms of privacy, any remote voting scheme has a fundamental vulnerability. Election officials do not control the remote environment. Therefore, no matter what measures are implemented to secure the ballot secrecy, the adversary can violate it by standing behind a voter and observing his choice.

Assuming that remote location is secure, the adversary can violate voter's privacy only by revealing the relationship between specific token and a voter. There are two entities that contain such information: the registrar and a voter's machine. Since we are assuming that the registrar is honest, then only the machine is left. As we noted before, there is no possibility to guarantee that machine is totally secure.

4.4 Usability

At first glance, our scheme seems complex and not user-friendly. While it is partially true, from the voter perspective it is quite simple. The voter should run the program, authenticate himself, make a choice, and send ballot to the tellers. All the complexity such as zero-knowledge proofs and ballot generation can be hide behind the scene. Of course, theoretically it can help the adversary to compromise the client's

machine. But in practice, most of the voters won't go deep inside to ensure the security, but just prefer more convenient options.

4.5 Further Improvements

The scheme is novel and there is a large room for improvement. For example, sometimes voters are forced by the adversary to vote in a specific way. A predefined fake vote can be introduced to help voters satisfy the adversary, but at the same time does not execute his orders. Also as it was mentioned in security analysis section, a distributed key generation scheme can help to prevent the collaboration between election officials and tellers [39], [21]. Election officials and auditors should collectively generate a public-private key pair whereas the public key is common, but parties hold a share of the private key. Therefore, all auditors and election officials should be engaged in decrypting the ballots.

4.6 Conclusions

At the beginning, we noted that voting is a tough problem and requires intricate schemes to achieve mutually exclusive properties. Most of the schemes require to satisfy weak assumptions to ensure the security, integrity and privacy. Our schemes are not exclusion. While they do provide a nice level of verifiability, both schemes should be viewed more as an academic research, than a practical proposal. Just recently first academic schemes were deployed to the field [9].

Also we should note that despite the fundamental vulnerability for voter coercion, the remote voting increases in popularity. It is less time-consuming, and more

convenient for the most of voters. The academia should continue the research in the field of remote voting and propose better alternatives to the traditional remote voting schemes focusing more on usability and convenience rather than security and verifiability. At the current state, most of the proposed schemes are far better than mail in voting in terms of integrity and privacy.

REFERENCES

1. Gumbel, Andrew. *Steal this vote: Dirty elections and the rotten history of democracy in America*. Nation Books, 2005.
2. Alvarez, Michael R., Katz, Jonathan N., Stewart III, Charles, Rivest, Ronald L., Ansolabehere, Stephen, Hall, Thad E., "Voting: What Has Changed, What Hasn't, and What Needs Improvement." *Voting Technology Project report*, California Institute of Technology / Massachusetts Institute of Technology, 2013.
3. Statistics of the presidential and congressional election of November 7, 2000, http://clerk.house.gov/member_info/electionInfo/2000election.pdf
4. Florida Department of State. Official Results of the November 7, 2000 General Election, 2000. http://election.dos.state.fl.us/elections/resultsarchive/SummaryRpt.asp?ElectionDate=11/7/2000&Race=PRE&DATAMODE=.
5. Ansolabehere, Stephen, and Charles Stewart Stewart. "Residual votes attributable to technology." *Journal of Politics* 67, no. 2 (2005): 365-389.
6. Feldman, Ariel J., J. Alex Halderman, and Edward W. Felten. "Security analysis of the Diebold AccuVote-TS voting machine." (2006).
7. Thompson, Ken. "Reflections on trusting trust." *Communications of the ACM* 27, no. 8 (1984): 761-763.
8. Arrow, Kenneth J. *Social choice and individual values*. Vol. 12. Yale university press, 2012.
9. Levin, Jonathan, and Barry Nalebuff. "An introduction to vote-counting schemes." *The Journal of Economic Perspectives* 9, no. 1 (1995): 3-26.
10. Nurmi, Hannu. *Comparing voting systems*. Dordrecht: Reidel, 1987.

11. Condorcet, Marquis de. "Essay on the Application of Analysis to the Probability of Majority Decisions." *Paris: Imprimerie Royale* (1785).
12. US Election Assistance Commission. "Voluntary voting system guidelines." *US Election Assistance Commission* (2005).
13. Schneier, Bruce. Crypto-gram newsletter. http://www.schneier.com/crypto-gram-0102.html#10 (February 2001). Last accessed July 20, 2013.
14. Adida, Ben. "Advances in cryptographic voting systems." PhD diss., Massachusetts Institute of Technology, 2006.
15. Adida, Ben, and C. Andrew Neff. "Ballot casting assurance." In Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop 2006 on Electronic Voting Technology Workshop, pp. 7-7. USENIX Association, 2006.
16. Clarkson, Michael R., Stephen Chong, and Andrew C. Myers. "Civitas: Toward a secure voting system." In *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pp. 354-368. IEEE, 2008.
17. Ryan, Peter YA, David Bismark, J. A. Heather, Steve A. Schneider, and Zhe Xia. "The prêt à voter verifiable election system." *IEEE transactions on information forensics and security* 4, no. 4 (2009): 662-673.
18. Kremer, Steve, and Mark Ryan. "Analysis of an electronic voting protocol in the applied pi calculus." In *Programming Languages and Systems*, pp. 186-200. Springer Berlin Heidelberg, 2005.
19. Ryan, Peter YA, and Thea Peacock. "Prêta voter: a systems perspective." *University of Newcastle, Tech. Rep. CS-TR-929* (2005).

20. Sako, Kazue, and Joe Kilian. "Receipt-free mix-type voting scheme." In *Advances in Cryptology—EUROCRYPT'95*, pp. 393-403. Springer Berlin Heidelberg, 1995.
21. Diffie, Whitfield, and Martin Hellman. "New directions in cryptography." Information Theory, IEEE Transactions on 22, no. 6 (1976): 644-654.
22. Chaum, David, Richard Carback, Jeremy Clark, Aleksander Essex, Stefan Popoveniuc, Ronald L. Rivest, Peter YA Ryan, Emily Shen, and Alan T. Sherman. "Scantegrity II: end-to-end verifiability for optical scan election systems using invisible ink confirmation codes." EVT 8 (2008): 1-13.
23. Heather, James, and David Lundin. "The append-only web bulletin board." In *Formal Aspects in Security and Trust*, pp. 242-256. Springer Berlin Heidelberg, 2009.
24. Aslam, Javed A., Raluca A. Popa, and Ronald L. Rivest. "On estimating the size and confidence of a statistical audit." In *Proc. 2007 USENIX/ACCURATE Electronic Voting Technology Workshop (EVT'07)*. 2007.
25. Jones, Douglas W. "Chain voting." In Workshop on Developing an Analysis of Threats to Voting Systems, National Institute of Standards and Technology. 2005.
26. Karlof, Chris, Naveen Sastry, and David Wagner. "Cryptographic voting protocols: A systems perspective." In *USENIX Security Symposium*, vol. 12, p. 39. 2005.
27. Benaloh, Josh C., and Moti Yung. "Distributing the power of a government to enhance the privacy of voters." In Proceedings of the fifth annual ACM symposium on Principles of distributed computing, pp. 52-62. ACM, 1986.
28. Fujioka, Atsushi, Tatsuaki Okamoto, and Kazuo Ohta. "A practical secret voting scheme for large scale elections." In Advances in Cryptology—AUSCRYPT'92, pp. 244-251. Springer Berlin Heidelberg, 1993.

29. Hirt, Martin, and Kazue Sako. "Efficient receipt-free voting based on homomorphic encryption." In Advances in Cryptology—EUROCRYPT 2000, pp. 539-556. Springer Berlin Heidelberg, 2000.
30. Jakobsson, Markus, Ari Juels, and Ronald L. Rivest. "Making Mix Nets Robust For Electronic Voting By Randomized Partial Checking." In USENIX security symposium, pp. 339-353. 2002.
31. Neff, C. Andrew. "A verifiable secret shuffle and its application to e-voting." In Proceedings of the 8th ACM conference on Computer and Communications Security, pp. 116-125. ACM, 2001.
32. Chaum, David L. "Untraceable electronic mail, return addresses, and digital pseudonyms." Communications of the ACM 24, no. 2 (1981): 84-90.
33. Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." Consulted 1 (2008): 2012.
34. Schnorr, Claus-Peter. "Efficient signature generation by smart cards." Journal of cryptology 4, no. 3 (1991): 161-174.
35. Rivest, Ronald L., Adi Shamir, and Len Adleman. "A method for obtaining digital signatures and public-key cryptosystems." Communications of the ACM 21, no. 2 (1978): 120-126.
36. Kaliski, Burt. "The Mathematics of the RSA Public-Key Cryptosystem." RSA Laboratories (2006).
37. US NIST. “Descriptions of SHA-256, SHA-384 and SHA- 512”. (2001),http:// csrc.nist.gov/encryptionishs/sha2S6-3X4-SI2.pdf (Last accessed: July 22, 2013).

38. Republic of Estonia. Digital signatures act. http://www.riigiteataja.ee/ert/act.jsp?id=694375, 2000.
39. Brandt, Felix. "Efficient cryptographic protocol design based on distributed El Gamal encryption." In Information Security and Cryptology-ICISC 2005, pp. 32-47. Springer Berlin Heidelberg, 2006.
40. Burton, Craig, Chris Culnane, James Heather, Thea Peacock, Peter YA Ryan, Steve Schneider, Sriramkrishnan Srinivasan, Vanessa Teague, Roland Wen, and Zhe Xia. "Using Prêta Voter in Victorian State elections." In Electronic Voting Technology Workshop/Workshop on Trustworthy Elections. 2012.
41. Rivest, Ronald L., and Warren D. Smith. "Three voting protocols: ThreeBallot, VAV, and Twin." In Proceedings of the USENIX Workshop on Accurate Electronic Voting Technology, vol. 16. 2007.